PhD Dissertation, University of Manitoba

A Framework for Collaborative Planning and Investigations of Decision Support Tools for Hydro Development

by

Michael J. Bender

Department of Civil and Geological Engineering

A Thesis Submitted to the Faculty of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

September, 1996

ISBN 0-612-16089-0

Name Michael J. Bender

Dissertation Abstracts International and *Masters Abstracts International* are arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation or thesis. Enter the corresponding four-digit code in the spaces provided.

Civil Engineering — SUBJECT TERM

0543 — SUBJECT CODE

UMI

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS
Architecture 0729
Art History 0377
Cinema 0900
Dance 0378
Fine Arts 0357
Information Science 0723
Journalism 0391
Library Science 0399
Mass Communications 0708
Music 0413
Speech Communication 0459
Theater 0465

EDUCATION
General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0273
Bilingual and Multicultural 0282
Business 0688
Community College 0275
Curriculum and Instruction 0727
Early Childhood 0518
Elementary 0524
Finance 0277
Guidance and Counseling 0519
Health 0680
Higher 0745
History of 0520
Home Economics 0278
Industrial 0521
Language and Literature 0279
Mathematics 0280
Music 0522
Philosophy of 0998
Physical 0523
Psychology 0525
Reading 0535
Religious 0527
Sciences 0714
Secondary 0533
Social Sciences 0534
Sociology of 0340
Special 0529
Teacher Training 0530
Technology 0710
Tests and Measurements 0288
Vocational 0747

LANGUAGE, LITERATURE AND LINGUISTICS
Language
General 0679
Ancient 0289
Linguistics 0290
Modern 0291
Literature
General 0401
Classical 0294
Comparative 0295
Medieval 0297
Modern 0298
African 0316
American 0591
Asian 0305
Canadian (English) 0352
Canadian (French) 0355
English 0593
Germanic 0311
Latin American 0312
Middle Eastern 0315
Romance 0313
Slavic and East European 0314

PHILOSOPHY, RELIGION AND THEOLOGY
Philosophy 0422
Religion
General 0318
Biblical Studies 0321
Clergy 0319
History of 0320
Philosophy of 0322
Theology 0469

SOCIAL SCIENCES
American Studies 0323
Anthropology
Archaeology 0324
Cultural 0326
Physical 0327
Business Administration
General 0310
Accounting 0272
Banking 0770
Management 0454
Marketing 0338
Canadian Studies 0385
Economics
General 0501
Agricultural 0503
Commerce-Business 0505
Finance 0508
History 0509
Labor 0510
Theory 0511
Folklore 0358
Geography 0366
Gerontology 0351
History
General 0578
Ancient 0579
Medieval 0581
Modern 0582
Black 0328
African 0331
Asia, Australia and Oceania 0332
Canadian 0334
European 0335
Latin American 0336
Middle Eastern 0333
United States 0337
History of Science 0585
Law 0398
Political Science
General 0615
International Law and Relations 0616
Public Administration 0617
Recreation 0814
Social Work 0452
Sociology
General 0626
Criminology and Penology 0627
Demography 0938
Ethnic and Racial Studies 0631
Individual and Family Studies 0628
Industrial and Labor Relations 0629
Public and Social Welfare 0630
Social Structure and Development 0700
Theory and Methods 0344
Transportation 0709
Urban and Regional Planning 0999
Women's Studies 0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES
Agriculture
General 0473
Agronomy 0285
Animal Culture and Nutrition 0475
Animal Pathology 0476
Food Science and Technology 0359
Forestry and Wildlife 0478
Plant Culture 0479
Plant Pathology 0480
Plant Physiology 0817
Range Management 0777
Wood Technology 0746
Biology
General 0306
Anatomy 0287
Biostatistics 0308
Botany 0309
Cell 0379
Ecology 0329
Entomology 0353
Genetics 0369
Limnology 0793
Microbiology 0410
Molecular 0307
Neuroscience 0317
Oceanography 0416
Physiology 0433
Radiation 0821
Veterinary Science 0778
Zoology 0472
Biophysics
General 0786
Medical 0760

EARTH SCIENCES
Biogeochemistry 0425
Geochemistry 0996
Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoecology 0426
Paleontology 0418
Paleozoology 0985
Palynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES
Environmental Sciences 0768
Health Sciences
General 0566
Audiology 0300
Chemotherapy 0992
Dentistry 0567
Education 0350
Hospital Management 0769
Human Development 0758
Immunology 0982
Medicine and Surgery 0564
Mental Health 0347
Nursing 0569
Nutrition 0570
Obstetrics and Gynecology 0380
Occupational Health and Therapy 0354
Ophthalmology 0381
Pathology 0571
Pharmacology 0419
Pharmacy 0572
Physical Therapy 0382
Public Health 0573
Radiology 0574
Recreation 0575
Speech Pathology 0460
Toxicology 0383
Home Economics 0386

PHYSICAL SCIENCES

Pure Sciences
Chemistry
General 0485
Agricultural 0749
Analytical 0486
Biochemistry 0487
Inorganic 0488
Nuclear 0738
Organic 0490
Pharmaceutical 0491
Physical 0494
Polymer 0495
Radiation 0754
Mathematics 0405
Physics
General 0605
Acoustics 0986
Astronomy and Astrophysics 0606
Atmospheric Science 0608
Atomic 0748
Electronics and Electricity 0607
Elementary Particles and High Energy 0798
Fluid and Plasma 0759
Molecular 0609
Nuclear 0610
Optics 0752
Radiation 0756
Solid State 0611
Statistics 0463

Applied Sciences
Applied Mechanics 0346
Computer Science 0984
Engineering
General 0537
Aerospace 0538
Agricultural 0539
Automotive 0540
Biomedical 0541
Chemical 0542
Civil 0543
Electronics and Electrical 0544
Heat and Thermodynamics 0348
Hydraulic 0545
Industrial 0546
Marine 0547
Materials Science 0794
Mechanical 0548
Metallurgy 0743
Mining 0551
Nuclear 0552
Packaging 0549
Petroleum 0765
Sanitary and Municipal 0554
System Science 0790
Geotechnology 0428
Operations Research 0796
Plastics Technology 0795
Textile Technology 0994

PSYCHOLOGY
General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

Nom ____

Dissertation Abstracts International est organisé en catégories de sujets. Veuillez s.v.p. choisir le sujet qui décrit le mieux votre thèse et inscrivez le code numérique approprié dans l'espace réservé ci-dessous.

____ ☐☐☐☐ U·M·I

SUJET CODE DE SUJET

Catégories par sujets

HUMANITÉS ET SCIENCES SOCIALES

COMMUNICATIONS ET LES ARTS
Architecture 0729
Beaux-arts 0357
Bibliothéconomie 0399
Cinéma 0900
Communication verbale 0459
Communications 0708
Danse 0378
Histoire de l'art 0377
Journalisme 0391
Musique 0413
Sciences de l'information 0723
Théâtre 0465

ÉDUCATION
Généralités 515
Administration 0514
Art 0273
Collèges communautaires 0275
Commerce 0688
Économie domestique 0278
Éducation permanente 0516
Éducation préscolaire 0518
Éducation sanitaire 0680
Enseignement agricole 0517
Enseignement bilingue et multiculturel 0282
Enseignement industriel 0521
Enseignement primaire. 0524
Enseignement professionnel 0747
Enseignement religieux 0527
Enseignement secondaire 0533
Enseignement spécial 0529
Enseignement supérieur 0745
Évaluation 0288
Finances 0277
Formation des enseignants 0530
Histoire de l'éducation 0520
Langues et littérature 0279
Lecture 0535
Mathématiques 0280
Musique 0522
Orientation et consultation 0519
Philosophie de l'éducation 0998
Physique 0523
Programmes d'études et enseignement 0727
Psychologie 0525
Sciences 0714
Sciences sociales 0534
Sociologie de l'éducation 0340
Technologie 0710

LANGUE, LITTÉRATURE ET LINGUISTIQUE
Langues
Généralités 0679
Anciennes 0289
Linguistique 0290
Modernes 0291
Littérature
Généralités 0401
Anciennes 0294
Comparée 0295
Mediévale 0297
Moderne 0298
Africaine 0316
Américaine 0591
Anglaise 0593
Asiatique 0305
Canadienne (Anglaise) 0352
Canadienne (Française) 0355
Germanique 0311
Latino-américaine 0312
Moyen-orientale 0315
Romane 0313
Slave et est-européenne 0314

PHILOSOPHIE, RELIGION ET THÉOLOGIE
Philosophie 0422
Religion
Généralités 0318
Clergé 0319
Études bibliques 0321
Histoire des religions 0320
Philosophie de la religion 0322
Théologie 0469

SCIENCES SOCIALES
Anthropologie
Archéologie 0324
Culturelle 0326
Physique 0327
Droit 0398
Économie
Généralités 0501
Commerce-Affaires 0505
Économie agricole 0503
Économie du travail 0510
Finances 0508
Histoire 0509
Théorie 0511
Études américaines 0323
Études canadiennes 0385
Études féministes 0453
Folklore 0358
Géographie 0366
Gérontologie 0351
Gestion des affaires
Généralités 0310
Administration 0454
Banques 0770
Comptabilité 0272
Marketing 0338
Histoire
Histoire générale 0578
Ancienne 0579
Médiévale 0581
Moderne 0582
Histoire des noirs 0328
Africaine 0331
Canadienne 0334
États-Unis 0337
Européenne 0335
Moyen-orientale 0333
Latino-américaine 0336
Asie, Australie et Océanie 0332
Histoire des sciences 0585
Loisirs 0814
Planification urbaine et régionale 0999
Science politique
Généralités 0615
Administration publique 0617
Droit et relations internationales 0616
Sociologie
Généralités 0626
Aide et bien-àtre social 0630
Criminologie et établissements pénitentiaires 0627
Démographie 0938
Études de l' individu et de la famille 0628
Études des relations interethniques et des relations raciales 0631
Structure et développement social 0700
Théorie et méthodes. 0344
Travail et relations industrielles 0629
Transports 0709
Travail social 0452

SCIENCES ET INGÉNIERIE

SCIENCES BIOLOGIQUES
Agriculture
Généralités 0473
Agronomie. 0285
Alimentation et technologie alimentaire 0359
Culture 0479
Élevage et alimentation 0475
Exploitation des péturages 0777
Pathologie animale 0476
Pathologie végétale 0480
Physiologie végétale 0817
Sylviculture et faune 0478
Technologie du bois 0746
Biologie
Généralités 0306
Anatomie 0287
Biologie (Statistiques) 0308
Biologie moléculaire 0307
Botanique 0309
Çellule 0379
Écologie 0329
Entomologie 0353
Génétique 0369
Limnologie 0793
Microbiologie 0410
Neurologie 0317
Océanographie 0416
Physiologie 0433
Radiation 0821
Science vétérinaire 0778
Zoologie 0472
Biophysique
Généralités 0786
Medicale 0760

SCIENCES DE LA TERRE
Biogéochimie 0425
Géochimie 0996
Géodésie 0370
Géographie physique 0368
Géologie 0372
Géophysique 0373
Hydrologie 0388
Minéralogie 0411
Océanographie physique 0415
Paléobotanique 0345
Paléoécologie 0426
Paléontologie 0418
Paléozoologie 0985
Palynologie 0427

SCIENCES DE LA SANTÉ ET DE L'ENVIRONNEMENT
Économie domestique 0386
Sciences de l'environnement 0768
Sciences de la santé
Généralités 0566
Administration des hipitaux 0769
Alimentation et nutrition 0570
Audiologie 0300
Chimiothérapie 0992
Dentisterie 0567
Développement humain 0758
Enseignement 0350
Immunologie 0982
Loisirs 0575
Médecine du travail et thérapie 0354
Médecine et chirurgie 0564
Obstétrique et gynécologie 0380
Ophtalmologie 0381
Orthophonie 0460
Pathologie 0571
Pharmacie 0572
Pharmacologie 0419
Physiothérapie 0382
Radiologie 0574
Santé mentale 0347
Santé publique 0573
Soins infirmiers 0569
Toxicologie 0383

SCIENCES PHYSIQUES
Sciences Pures
Chimie
Genéralités 0485
Biochimie 487
Chimie agricole 0749
Chimie analytique 0486
Chimie minérale 0488
Chimie nucléaire 0738
Chimie organique 0490
Chimie pharmaceutique 0491
Physique 0494
PolymÇres 0495
Radiation 0754
Mathématiques 0405
Physique
Généralités 0605
Acoustique 0986
Astronomie et astrophysique 0606
Electronique et électricité 0607
Fluides et plasma 0759
Météorologie 0608
Optique 0752
Particules (Physique nucléaire) 0798
Physique atomique 0748
Physique de l'état solide 0611
Physique moléculaire 0609
Physique nucléaire 0610
Radiation 0756
Statistiques 0463

Sciences Appliqués Et Technologie
Informatique 0984
Ingénierie
Généralités 0537
Agricole 0539
Automobile 0540
Biomédicale 0541
Chaleur et ther modynamique 0348
Conditionnement (Emballage) 0549
Génie aérospatial 0538
Génie chimique 0542
Génie civil 0543
Génie électronique et électrique 0544
Génie industriel 0546
Génie mécanique 0548
Génie nucléaire 0552
Ingénierie des systämes 0790
Mécanique navale 0547
Métallurgie 0743
Science des matériaux 0794
Technique du pétrole 0765
Technique minière 0551
Techniques sanitaires et municipales 0554
Technologie hydraulique 0545
Mécanique appliquée 0346
Géotechnologie 0428
Matières plastiques (Technologie) 0795
Recherche opérationnelle 0796
Textiles et tissus (Technologie) 0794

PSYCHOLOGIE
Généralités 0621
Personnalité 0625
Psychobiologie 0349
Psychologie clinique 0622
Psychologie du comportement 0384
Psychologie du développement 0620
Psychologie expérimentale 0623
Psychologie industrielle 0624
Psychologie physiologique 0989
Psychologie sociale 0451
Psychométrie 0632

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

COPYRIGHT PERMISSION

A FRAMEWORK FOR COLLABORATIVE PLANNING AND INVESTIGATIONS OF DECISION SUPPORT TOOLS FOR HYDRO DEVELOPMENT

BY

MICHAEL J. BENDER

A Thesis/Practicum submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Abstract

Management of water resources can be a contentious issue. International conflicts can be ignited from controversy concerning the use of water. On the local and regional scale, though, there is opportunity to work with people who would be affected by management of water resources. The planning of water development projects has evolved from the state of partial economic analysis, to more holistic mitigation of externalities. Environmental licensing processes have expanded the number of participants for the selection of alternatives. This may not be the most efficient framework for internalizing those issues which are outside the domain of the project proponent.

A collaborative planning process involves stakeholders in the conceptual design stage of a project. It is viewed as a potentially efficient prelude to an environmental licensing process. If participants in the collaboration are able to work toward consensus before an adversarial licensing process, this framework may resolve many issues which inhibit water resource development. As an approach to achieving sustainability, collaborative planning focuses on the concept of consensus, and attempts to achieve that goal by moving away from the reference point of conflicting opinions.

This dissertation attempts to define a suitable collaborative planning framework for hydroelectric development, and apply it within 3 decision support system modules: selection of evaluation criteria, generation of alternatives, and evaluation of alternatives.

The criteria selection module incorporates the concept of grounded theory to base the interaction of decision makers on measurable facts about the problem domain. These grounded facts form the basis of linking the value systems of participants to the technical management issues of planning a project.

The alternative generation module embeds GIS capabilities within a decision support system that supports feedback using expert systems. It allows participants to interactively experiment with different technical alternatives, and automatically generate the consequences of selecting different technical options based on available expertise.

The process of evaluating alternatives is supported by a fuzzy compromise approach which attempts to preserve the transparency and intuitiveness of the compromise programming technique for multicriteria decision-making, while incorporating various sources of uncertainty. The approach is based on arithmetic operations on fuzzy sets using the extension principle. Feedback to decision makers is either in a visual form representing the range of possible performance of an alternative, or in the form of rankings. Rankings are made with one of two fuzzy ranking measures, both of which employ parametric control to show the impact of different outlooks on the part of decision makers.

Acknowledgements

Although the name below the title for this dissertation suggests that there is a single author, many people have contributed and deserve recognition.

My academic advisor, Professor Slobodan Simonovic (Simon) has been a mentor to me for many years now. His original idea for a multidisciplinary focus to an engineering thesis, back in 1992, was something that was very different. I was intrigued and interested. I couldn't escape the lure of possibly contributing to the mindset of water resource managers. Since 1992, multidisciplinary efforts to manage water resource systems have exploded onto the scene under the guise of sustainable development and public participation. Simon and I have raced to keep up, but also to add something to the mix. I think that we have succeeded. Simon, all those pep talks have paid off! We have produced a non-traditional civil engineering PhD thesis which deals with many of the technical complexities of water resources problems, but also recognizes many of the external issues which engineers have (more or less) neglected for years. As the dust settles on this work, and collaborative planning processes become common place, I hope that we can smile and say that we were part of that history. I will look forward to continued collaboration and contact in the coming years. You are a man of great integrity and deserve all the respect in the world. You have provided me with the opportunities which every young person so desperately needs, in order to be noticed. And I will never forget those times when you supported and defended me against unscrupulous colleagues.

Members of my examining committee have been very active in supporting, encouraging, and working with me. Professor Barbara Lence (Civil & Geological Engineering) had her ear bent on more occasions than she probably cares to remember. Professor Gary Johnson (Agriculture Economics and Farm Management) has influenced my understanding of resource policies, and educated me on how different disciplines think and work. I had never realized, before I met Gary, how many ways there are to say the same thing! The fact that professionals like engineers and economists can have such different perspectives continues to motivate my research.

Manitoba Hydro has been a key contributor. Their support, in the form of a graduate scholarship and technical support (digital geographic data), are greatly appreciated. Denis De Pape, Dennis Windsor, and Per Stokke always showed great interest and enthusiasm in the project. Manitoba Hydro also eagerly provided a case study for which to apply the collaborative framework ideas and decision support tools. Their moral support, along with Canada Department of Fisheries and Oceans, helped to settle the direction of this research.

Where is a graduate student without his comrades? I have enjoyed the company and the support of many people at the FIDS lab (Rob, Sri, Zsolt, Kwame, Glen, and others), the secretaries in the civil engineering office (Judy, Norma, Melanie, Ingrid, and Shirley), my raquetball partners - my daily diversionaries (Jeff, Jim, Greg), and many others.

My Christian family has also been instrumental over the years. Most recently, my men's group (Mike, Greg, Clayton, and Steve) has encouraged me to be a promise keeper and a man of integrity in my profession as well as in my home and spiritual pilgrimage.

Doing a PhD degree close to home has its benefits. My parents have always been very supportive. They raised me to believe that I am capable of accomplishing anything. That kind of positive reinforcement is rare, I feel, but absolutely necessary for me to persevere and continue to believe in myself.

When I was trying to decide whether to pursue a PhD, I was engaged to a great country girl who married me in spite of the fact that I was a student. Marilyn has always been my greatest supporter. She has loved me, pushed me, scolded me, and held me through the trials that seem to accompany PhD degrees. Now that

Emily is here, our lives have changed again. My girls give me diversion, which is sorely needed when the work tries to follow you home every night. Thank you for your patience and love.

Table of Contents

List of Figures

Chapter 6 Integration of GIS and ES Decision Support Tools in the Exploration of Alternatives

Chapter 7 Evaluation of Alternatives using a Fuzzy Compromise Approach

List of Tables

Chapter 1 Introduction

1.1 Problem Statement

In a global community with a heightened awareness of social and environmental impacts, there is a growing need for proponents of water resource development, and water resource managers, to formulate their alternatives within a more complete socioeconomic model. There is a great interest in public participation for development initiatives, but it is traditionally limited to public relations, or impact mitigation after proposals have been developed. Proposed plans are developed to be economically efficient. Restoration and mitigation costs can be very expensive when added to plans after they have been developed. Addition of mitigation costs may result in other plans becoming more efficient for the proponent.

Manitoba Hydro is a proponent that experiences a great amount of external pressure to respond to stakeholders (affected or interested parties) concerning impacts on the environment caused by activities such as hydroelectric development. One way of reducing tensions between proponents and stakeholders is to allow them more direct participation in the early planning of a potential project (before licensing).

The concept of collaborative planning is a novel idea that has seen limited use because of the complexities that prevent people with different backgrounds from communicating. This research explores a number of tools and techniques toward a creative computer-assisted planning environment for stakeholder participation. The premise, for potential stakeholder interaction with the proponent in early planning, is that creating ground for a common understanding will enhance motivation for creative solutions to mitigate potential impacts.

1.2 Project Scope

An example problem has been selected to demonstrate ideas and the roles of decision tools for collaborative planning, based on possible hydroelectric development by Manitoba Hydro of the Wuskwatim and Three Point Lake area on the Burntwood River system northwest of Thompson in northern Manitoba.

Manitoba Hydro has selected the Wuskwatim Lake area as a site with good hydroelectric potential and water supply reliability, among other reasons (Manitoba Hydro, 1987). Manitoba Hydro has a mandate for providing electricity to Manitobans at a reasonable cost and with a high degree of reliability (Manitoba Hydro, 1990). This includes planning for future demands. The Wuskwatim generating station is one of several sites being considered for development.

Like all forms of power generation, hydroelectric power produces external impacts to various ecosystems. However, it is promoted as being "cleaner" than most forms of power. Manitoba has invested heavily into the use of hydroelectric power because of the abundance of water in the province. Most of the generating capacity is in isolated areas where the Nelson River drains into Hudson's Bay. This is mostly a pristine environment, where few people live, with abundant natural resources. Traditionally, the local population has relied on hunting, trapping, and fishing to maintain a subsistence economy. Villages in the region are mainly dependent on mining and forestry operations.

Development of northern water supplies have caused a number of impacts to the natural resources and cultural integrity of the region during the past 30 years. Some impacts were not sufficiently mitigated, but in fairness, many of the impacts were either not understood, expected, or considered valuable. The changing social attitudes, along with alert and even militant opposition from traditional communities, environmental groups, and government regulatory agencies, have brought many new concerns to the project licensing process.

Manitoba Hydro works through a long process of review before a proposal is acceptable for licensing. A proposal for developing a generating station, such as at Wuskwatim Lake, passes through a variety of regulatory agencies - each with a particular agenda. Then, finally, the proposal is presented to the Public Utilities Board, which includes a number of public hearings and debates. Throughout the lengthy licensing process, a large number of stakeholders or interested parties are given an opportunity to assess the proposal. It is possible that a number of changes are required, and the entire project may sometimes be in jeopardy.

The costs of performing in-depth environmental studies, and of allowing flexibility in project design for mitigating or avoiding adverse impacts, is quickly becoming a necessity for many development proponents. It is (or will be) the responsibility of the proponent to perform these tasks efficiently. How these development projects are handled in the early stages of planning is of growing concern.

Collaborative planning decision support systems are a form of tool which may enhance the participation process through the use of computer integration of modelling and analysis capabilities. Input is restricted to a structured form, but allows flexible access to information and available expertise. This dissertation focuses on the potential use of different decision support tools, taking into account the potential planning framework for stakeholder participation.

1.3 Paper Summary

This research presents a framework for facilitating collaboration with stakeholders in the early planning phases of project conceptualization. The planning approach demands a synthesis of physical modelling with subjective priorities of various stakeholders. It stresses careful, deliberate, definition of the physical scope of planning, and allows individual objectives to be explored in a cooperative, multidisciplinary environment.

Application of the collaborative planning approach, in the form of a decision support system (DSS), is presented as a Collaborative Planning Support System (CPSS). The basis for its functionality is the integration of modelling and decision tools, a management system for allowing adaptation of design proposals, a method of administering project goals and objectives, and a technique for providing feedback to stakeholders involved in the decision process.

Chapter 3 introduces many of the available tools for collaborative decision support that are implemented within the CPSS presented below. The various aspects and behaviour of decision support systems are examined and related to the purpose of supplying collaborative decision support.

The CPSS is introduced in Chapter 4 in terms of a conceptual systems approach for modelling collaborative decisions. The feedback mechanisms for modelling dynamic systems are discussed, and the CPSS application, in terms of the 3 modules presented in Chapters 5, 6, and 7, are described using the systems approach.

In Chapter 5, technical expertise, and available data, are moulded into the role of relating physical and contextual realms by providing a set of base level indicators called grounded facts. These grounded facts are then used to build an individual's goal structure, and ground it, or lay its foundation, within physical modelling and expertise. Within this structured approach for collaborative planning, the quality of decision support is subject to various disciplines of knowledge including engineering, biology, sociology, economics, or politics.

Chapter 6 presents a possible form of technology integration for generation of alternatives. In addressing a forum for stakeholder participation, object-oriented management of database information is combined with artificial intelligence techniques. These tools, along with spatial data management and analysis, are integrated and applied to surface water resource management questions. While each technology is available for specialized tasks, a state-of-the-art hybrid has enormous potential for assisting in decisions related to water resources. Many of the assumptions for application and difficulties in achieving full integration are explored.

An adaptation of multicriteria decision-making is presented in Chapter 7. Providing a multicriteria systems management tool, adapted to specialized demands of collaborative planning support, is the purpose of a fuzzy compromise approach. Identification of a structured process for stakeholder participation in early planning stages produces unique circumstances and opportunities for decision-making. In the fuzzy compromise multicriteria technique there is a great opportunity for various forms of uncertainty to be expressed explicitly.

As an introduction to the collaborative framework and the applications which follow, the role of sustainability is examined, as it relates to water resources planning and management in Chapter 2. This includes a short explanation of the origin of sustainability issues from our current economic paradigm, evolving approaches to analysis, and a philosophy of collaborative integrated multidisciplinary decision-making as one approach for achieving sustainability.

Chapter 2 A Framework for Achieving Sustainability

The following chapter presents many of the concepts and background necessary for understanding and justifying the approaches and techniques developed in later chapters. It also defines (in sections 2.4 and 2.6) the underlying principles which motivate the contributions of this dissertation, which are to identify collaborative planning decision support tools within a consensus sustainability framework.

Any discussion of sustainability, related to the management of water resources, must begin with a traditional discussion of the management of renewable resources. As a renewable resource, water rights are allocated according to our understanding of efficiency - normally implemented as least cost. Market economics has governed our approach to allocating resources. Unfortunately, we have always operated from a partial economic model. Because of this fact, there exists externalities to our decision models. The underlying issue in discussions of how to achieve sustainability is how to internalize the socio-ecological concerns which we now recognize as valuable.

The market economic origins of present-day sustainability ethics for renewable resources are examined in section 2.1, followed by an experiment designed to explore different approaches to internalizing issues (section 2.2). In this experiment, the traditional market economic model for producing optimal policies in a dynamic system is challenged by a multiobjective approach that places our economic decision model within a different role. The observations from this experiment underline the complexities that exist in the process of internalizing issues within a dynamic renewable resource system. Together, sections 2.1 and 2.2 provide the historical and theoretical background for examining many of the issues related to sustainability.

2.1 Renewable Resources

A simple definition of a renewable resource, in the context of water, is a mass or energy source subject to constant or periodic flux (Conrad and Clark, 1987). Other renewable resources, such as animals, are also subject to flux but are dependent on much different mechanisms. Fish populations are governed by an annual growth function subject to fishing pressures. Water quantity and quality is governed by input from precipitation events and subject to water demand pressures.

The supply of surface water from rainfall or snowmelt, as well as groundwater, can be generally classified as a common property renewable resource. Water flows from one political jurisdiction to another, across one land owner's property to another, from one country to another. Supplies of water may satisfy a variety of needs. However, a number of needs may compete for the same water.

2.1.1 Property Rights

Property rights are considered a stream of benefits and costs that affect the allocation and use of resources. Conflicts over water resource development plans occur because of discrepancies or omissions in the set of rights for each stakeholder. These rights define a set of operating rules which govern the use of water. Economically, the level that these rules are developed tends to be the least cost (maximum benefit) property rights solution (Demsetz, 1967). The application of the least cost solution can occur anywhere on a scale from open access to private ownership. Open access refers to a resource for which there are no rules of use. Netting (1976) presented the famous example of grazing on the swiss commons where the addition of extra cattle by a single farmer did not directly cost the farmer the complete impact to the grassland, but was absorbed by all farmers. Changes in the environment and cultural values led to the adoption of rules governing the addition of cattle to the grassland. Netting considers this limited allocation of rights to be common instead of open.

Demsetz (1967) considers the level of privatization, moving away from open access to complete allocation of rights, to be dependent on the relative benefits realized from privatizing, subject to the transaction costs of adding extra rules. Costs may be in the form of negotiation, contracting, or enforcement. Any attempt to resolve conflicts by assigning more explicit or additional rules for resource utilization must consider the costs of achieving an ideal state. The economic solution suggests that an acceptable compromise or consensus property rights allocation will be found to be efficient when transaction costs are considered.

2.1.2 Common Property Externalities

Conflicts in water resource development are usually the result of impacts that are external to the project proponents, but affect others who use the resource or use the derivatives of that resource. These externalities always exist for a property rights system that is not completely private (ie. at some level of common property allocation of rights). It is the role of the legal system or a licensing board to determine at what level externalities are to be internalized. Internalization of both positive and negative impacts demands that users or beneficiaries of resource allocation bear the burden of their role in resource impact.

It is usually not efficient to internalize all possible property rights issues. Coase (1960) stated that externalities should be dealt with in an economically optimal manner, and that the relative allocation of rights does not matter, as long as conflicts are resolved at minimal social (or aggregate) cost. Coase, however, refers to a perfect market system where there are no transaction costs, among other assumptions. Baumol and Oates (1988) suggest that internalizing impacts is only desirable if the result is a pareto-optimal move toward the social welfare frontier. That is, for cases where all objectives improve. In practice, this is shown in a fishery problem presented by Hardin (1968) where the maximum economic benefit may not be at the point of maximum sustainable yield, (the open access solution) but at the point where marginal benefits equals marginal costs. It is a move to improve the maximum benefit. However, it is only a single objective. The welfare frontier is a composite of any number of objectives. The limiting assumption in this economic analysis is that the optimal management of a single firm or market entity can be directly transformed for use by all the firms in a geographic region acting as a single firm. In reality, each user of the resource acts independently according to limited information within the property rights structure. Each firm has its own objective of economic efficiency (among other more abstract or intangible objectives). Some or all of the affected firms will be in conflict - and a pareto-optimal move will not be possible. Only when noneconomic benefits are measured will pareto-optimal moves be possible, but these are not all applicable to the economic system of markets, and some are difficult to quantify in any units of measure.

2.1.3 Nonmarket Valuation

Recent discussions by economists and noneconomists alike have centred around expanding economic analysis to include an assessment of overall social welfare or standard of living. This includes social values related to our environment. Noneconomic ideas do not centre around how to come to a more advanced property rights structure as part of an economic policy, but they discuss the policy framework for improvement or preservation of noneconomic systems.

Assessing social welfare is a difficult task. Many noncommensurate goods do not relate to traditional markets. There are many identified forms of nonmarket benefits, such as *use*, *option*, *altruistic*, *preservation*, *bequest*, *intrinsic*, and *existence* (Brookshire et al, 1986). The terminology varies, but this list is representative. Benefits such as from recreation and pure existence values are difficult to measure and justify, but it is a current topic of discussion among economists, particularly for water resource benefits (Brookshire and Smith, 1987; Madariaga and McConnell, 1987; etc.).

A common approach in economic policy analysis is to use a nonmarket valuation method to determine a perceived cost or benefit for resources and property rights that are external to a market economy. Valuation methods include the *travel cost method*, *hedonic price method*, and the *contingent valuation method*. Travel cost and hedonic methods use available market data to estimate or develop surrogates for nonmarket values, while contingent valuation is more dependent on survey techniques and market simulation. Contingent valuation is a popular method. It uses survey techniques to estimate a demand function for a resource, either by calculating a willingness of individuals to pay for use of a resource (such as fishing rights), or a willingness to receive payment for transferring a property right (such as a payment by a polluter to residents who live with the environmental effects). There are several techniques for contingent valuation: a bidding game, open-ended questions, payment questions, dichotomous-choice format surveys, and contingent ranking. Techniques for willingness to pay always underestimate demands for resources while willingness to receive studies overestimate demands (Bishop and Heberlein, 1990). Willingness to pay is a conservative estimate of demands, and willingness to receive is a risk averse estimate of nonmarket demands. However, most contingent valuation studies related to water management focus on willingness to pay estimates of noncommensurate resource uses (Harpman et al, 1993; Bohm et al, 1993; Carson and Mitchell, 1993).

Economic analysis is able to describe the values of decision makers, but only in hindsight. Decisions suggest implicit monetary values to such things as cultural integrity and recreational uses. Economics has tools for approximation of these values as a predictive tool (contingent valuation is an example), but they can be extremely uncertain depending on the method used to derive the values (eg. survey technique). Part of the decision process can involve identification of base values such as: "in order to have made that choice, you must have to value this property right at least *X*". This is a slightly different question than the willingness to pay types of answers provided by implicit valuation of resources. Economics can calculate this value X, but there is no place for this subjective decision-making in the economic paradigm of conservation of scarce resources.

Utilization of nonmarket techniques have a number of shortcomings. There are enough techniques available to potentially justify or prevent any proposal by a proponent. This is because there is no consensus on values associated with nonmarket goods such as natural aquatic environments. Many of the forms of nonmarket benefits are interchangeable or interdependent. In addition to this, the validity of using these techniques within a traditional benefit-cost analysis has been questioned (Brookshire et al, 1986).

Noneconomist perspectives may be moving away from traditional economic theory in circumstances where the market system fails to account for noncommensurate resources, and value systems not explicitly defined within the property rights structure. For example, many temporal issues remain external to the market system as long as perceived marginal benefits are insufficient to offset the marginal costs of bringing future social values, and the possible restrictions of future options, to bear on our present economies. We can watch an example of this as the industrial world desperately clamps down on emissions that contribute to the destruction of the ozone layer, invoking large added costs to producing some goods and a heavy burden on some economies. What is merely implied, is the economic interpretation of how markets work, and how open access or common property resources are treated within our market system.

2.2 An Illustration of Internalization Techniques for Dynamic Renewable Resource Policies.

The field of renewable resources economics has explored descriptions of resources moving from open access responses to optimal utilization management (Anderson, 1993), and have begun to examine temporal equity considerations (Burton, 1993) and other tradeoffs associated with coordinating policy with more traditional markets (Johnson et al, 1990). This has led to discussions of knowledge integration and decision-making (Roots, 1992; Cairns et al, 1994; and Costanza et al, 1992). The evolution of multidisciplinary decision-making is developing from theoretical mathematical approaches such as Klopatek et al (1983), which follows traditional mathematical economic modelling, to an idea of flexibly-applied indicators to integrate the process of assessing ecological health - a concept of growing popularity with ecologists (Costanza et al, 1992).

Our theoretical market system demands significant institutional development via changes in policy and in the property rights structure toward more comprehensive rules of use for private property, common property, and open access resources. In the following example, an expansion of the traditional economic model, along with a multiobjective framework which incorporates temporal decision-making, are used to provide perspective on renewable resource management issues. Comparisons are made of different policy paradigms with representative modelling. The example does not include arguments related to difficulties in applying policy principles such as justification of internalizing matters of the ecosystem or problems in valuation of nonmarket resources.

2.2.1 Optimal Dynamic Management of Renewable Resources

Resource allocation models for continuous-time optimal dynamic management are familiar in resource economics. Analysis of market conditions and management strategy is possible by formulating an optimization problem and considering maximization using a Lagrangian method. A standard form of problem is to maximize the net present value flowing from the use of a resource for a given temporal domain, subject to a description of how the resource changes in time. The general case for a single resource (one state variable), and one control variable in a free state system, to be maximized for total net present value, is given below as a Gordon-Schaffer fishery model (Conrad and Clark, 1987) for a discrete system, where V_t is a value function at some time t which is dependent on both the state variable, X_t, and the control variable, Y_t. ρ is the discount factor (3) in terms of a discount rate, δ, and f represents the equation of motion for the state variable.

$$\max z = \sum_{t=0}^{T} V_t \rho^t \tag{1}$$

$$s.t. \;\; x_{t+1} - x_t = f \tag{2}$$

$$\rho = (1-\delta)^{-t} \tag{3}$$

The maximum principle, comprised of necessary conditions for the present value Hamiltonian, is used to solve the problem given a convex decision space. A Lagrangian method can be used to examine conditions for economic efficiency through use of the Hamiltonian function (Conrad and Clark, 1987). This solution involves a present value shadow price for the state variable given by the Lagrange multiplier. Economists, in many cases, are more interested in the current value of the shadow price for the state variable, obtained by maximizing the current value Hamiltonian. Some assumptions for this analysis are: pricing is not controlled within the scope of the model; there is a given wealth distribution, institutional arrangements, and property rights allocation that will be unaffected by changes in management of the state variable; there is free access to enter and exit the market, and effort can be adapted to different industries with no cost or time delay; etc.

An example is a simple model of a fishery resource where X_t is the state variable, a stock of fish subject to an amount of effort, E_t, (control variable) in harvesting the fish (defined as a number of fishing vessels in use). The value function (4) used defines p as the price of fish, H_t as the harvest function for fish (which is dependent on both the stock and the effort toward harvesting) defined in the example as (5) resulting in an equation of motion as given in (6). F is a natural growth function for the fish stock, q (used in (5)) is a scalar coefficient, and C_E is the marginal cost of effort for catching the fish.

$$V_t = pH_t - C_E E_t \tag{4}$$

$$H_t = qX_t E_t \tag{5}$$

$$X_{t+1} - X_t = F - H_t \tag{6}$$

Assuming that a steady state solution to this problem exists, the optimal policy for operating the fishery, based on maximizing the net present value, can be found in the form of a switching function for "bang-bang" control of E_t. This policy type essentially acts to push the stock level to its optimum as fast as possible. Transformation of the solution defines the analytical form for an efficient price (7). F' is the partial derivative of F with respect to X. By evaluating model conditions, economic policy can be implemented to ensure efficient economic use of the resource.

$$p = \frac{C_E}{qX}\left[1 + \frac{F}{X(\delta - F')}\right] \tag{7}$$

2.2.2 Sustainable Investment Model

When economic market efficiency is the guiding principle, additional subsets of economic conditions can be added. One is intergenerational equity, another is ecological integrity. Economic efficiency, intergenerational equity, and ecological integrity, are three objectives for a more complete paradigm for resource management. These conditions can be used to define an expanded market system in terms of market price for a "sustainable" resource as described in (8) (Young, 1992).

$$p = MC_S + MC_{LES} + MC_P + MC_{LFO} + MC_{LEV} + MC_A + MC_{KRD} \tag{8}$$

- MC_S is the marginal cost of supplying the resource,
- MC_{LES} is the marginal cost of replacing lost ecosystem support,
- MC_P is the marginal cost of any pollution that the resource use imposes on other people,
- MC_{LFO} is the marginal cost of offsetting lost future options,
- MC_{LEV} is the marginal cost of offsetting lost existence values,
- MC_A is the marginal compensation for additional costs associated with the provision of positive nonmarket benefits (retaining or creating future options and retaining existence values for the community), and
- MC_{KRD} is the marginal cost of capital associated with resource development.

Both MC_{LFO} and MC_{LEV} reflect social costs from losses in ecosystem diversity and resilience.

Applying this ecosystem paradigm to the previously defined fishery model, terms are added to the value function to reflect the sensitivity of the environment to decreasing populations, biomass, or diversity, and penalizes according to the potential degradation in future generations. The following 4 conditions might be added to the model:

- Mitigation of pollution as a function of the effort that producing the pollution, given as $-C_pE_t$.
- Additional development or restoration costs as a function of the stock population = $-f_D(D(X_t))$, where $D(X_t)$ is a development function (9) similar to the natural growth function which can be included in the equation of motion for the stock, with respect to the environmental carrying capacity, X_c, such that:

$$D \propto \frac{1}{(X_c - X_t)} \tag{9}$$

- Existence value of the fish stock $= f_E(X_t)$ as a function of the population.
- Costs associated with reducing future development options, or for evolutionary change (10):

$$f_o \propto \frac{1}{X}(1 - \rho^t) \tag{10}$$

Substituting $\{G(X(t),t) = f_E - f_D - f_O\}$ and solving for price, p, we find that price, as shown in (11) remains in its original form of a summation of marginal costs (Bender et al, 1994).

$$p = \frac{C_E + C_P}{qX}\left\{1 + \frac{\left[F + D - \left(\frac{qX}{C_E + C_P}\right)G'\right]}{X(\delta - F' - D')}\right\} \tag{11}$$

This new price definition is consistent with the components of price described earlier. Expansion of the model for internalizing nonmarket impacts causes marginal costs associated with use of the resource to increase assuming the G' term (the partial derivative of G with respect to X) is negative, or does not overpower the addition of D or C_P.

In the perfect market dynamic model, the marginal cost has a current marginal cost of extracting resources and a marginal user cost. The marginal user cost results from using the asset (the fish stock) at its current level rather than allowing that asset to grow to a higher level. By modifying the fishery model, based on application of the sustainable investment model, expanded definitions of cost include pollution impacts on other people, alternative resource development, and various costs that reflect impacts on the ecosystem. The net result to the model is an increase in marginal costs. These increased marginal costs result in increased fish stocks through reductions in overall effort towards fishing. As marginal costs increase, effort in the fishery decreases regardless of ownership structure, and stock levels increase toward the environmental carrying capacity.

Describing the internalization of unallocated property rights, using traditional resource economic concepts, allows economists to examine market effects. The example is very simple. Complications such as multiple fish species and competing fish species as described by Clark (1976) may not allow analytical solutions to easily be found. Adding growth constraints for other species serves to add Lagrangian multiplier terms to the Hamiltonian equation. More state variables will exist, as well as additional control variables. In the example, the form of solution remains consistent as the model becomes more complicated with additional valuation components, and an expanded equation of motion. Further steps for model expansion includes additional constraints, and more state and control variables.

2.2.3 Multiobjective Welfare Model

A Multiobjective decision framework may discuss problems in terms of a welfare model, defining welfare efficiency, substituting a welfare function, W, for monetary benefits, V. The model is dependent on the definition of welfare - which requires relative valuation of socio-ecological components. The efficiency condition in the welfare model (the objective function) considers the practice of discounting future values. Normally, the discount rate (considering implications of tradeoffs over time) is equivalent to some real cost of borrowing or using capital. An efficient policy is to maximize the value with regard to future values over the life span to be considered. In general, greater discount rates result in greater long-term degradation of the resource as policy is geared toward immediate returns. Smaller discount rates are generally associated with preserving the resource.

Choosing a scalar value discount rate to reflect social values may not differ from the previous interest discount. The optimal path may vary, but the inevitable end associated with the governing decision paradigm may be unaltered. For example, farmers are constantly presented with the problem of soil degradation. Tradeoffs between maximum yield in the short term and lower yields over the long run, force farmers to decide on practices that affect both their present financial requirements, and their longevity. Because of the difficulty in assessing future market and social values, farmers may operate on a particularly short time frame, using intensive practices to maximize financial gain without regard to the end state of the resource. Other farmers may choose a more conservative, set of practices in an effort to either extend the operational time frame, or as an attempt to maintain the end state of soil resources for future operations.

Of course, considerable uncertainty exists in evaluating future values and in choosing the best option at some point in the future - assuming we are aware of future consequences. Uncertainty breeds risk averse decision-making, which implies a practical or perceived cost of capital at a very high rate. In choosing a lower bound on the resource at the end state, we are selecting a level of conservation or safety factor. In doing so, the resource is driven to the lower bound.

To make operational decisions, we must also consider the time frame to restrict our analysis. The first impression is to extend the time frame, but it can also be argued that a shorter time frame may be more appropriate, depending on the policy definition, because our social values may change over time as understanding of our surroundings improves. Our conscious analytical choice of time frame defines the problem of intergenerational equity.

One of the possible ways for dealing with intergenerational equity issues is a multiobjective framework. It requires definition of objectives of future generations. Extension of the welfare model is required in order to adjust to the multiobjective structure. The fishery objective function can be expressed as the weighted combination of future values (12) such that the sum of weights equals one (13).

$$\max \sum_{t=0}^{T} w_t V_t \rho^t \tag{12}$$

$$\text{where} \sum_{t=0}^{T} w_t = 1 \tag{13}$$

This is not a traditional form of showing a multiobjective model. The different objectives are related to time as opposed to physical objectives. Application of (12) produces a set of nondominated solutions, as opposed to a single optimum followed by a subjective process to select one of the nondominated solutions as a best compromise solution. Assigning weights to define a system of likely scenarios may allow tracking of different policy perspectives within the dynamic model, although an analytical economic interpretation of this multiobjective paradigm is difficult. Rearrangement of our fishery model includes defining a dynamic optimization problem with two control variables (E_t and w_t), and a constraint to ensure the integral of w_t is equal to 1. The multiobjective framework cannot suggest optimal weights. Policy decisions remain outside model analysis. A similarly designed dynamic analysis problem may bias all weight to the initial time frame, or evenly distribute the weights (assuming steady state conditions exist).

2.2.4 Robust Resource Allocation Paradigm

Uncertainty in selecting objectives for future generations, as well as selecting a combination of weights may result in selection of a best compromise solution which does not explicitly internalize concerns. Some aspects of resource management indicate that the best compromise solution concept should be replaced with the concept of a most robust solution (Simonovic, 1989). Simonovic has demonstrated (in the field of water resources) that the idea of combining the sensitivity analysis of the multiobjective solution to objective values and preference (weight) structures results in the replacement of a best compromise solution with the most robust solution. The most robust solution is defined as a multiobjective alternative selection least sensitive to changes in the objectives and preference structure. If a number of alternatives are available in which the dynamic model is solved for a variety of circumstances, a compromise programming procedure can be used to test the relative quality of decisions under a number of weight combinations and distance measures. It is important to note that this conceptual difference in applying the multiobjective analysis provides a direct way for incorporating intergenerational equity issues in sustainable development and resources management.

Robust decision-making is not unlike selecting a site based on ecosystem sensitivity to the proposed form of development. The remaining portion of the problem lies in identifying the direction of search for new or improved alternatives. Optimists, such as those in the area of multicriteria decision analysis suggest the possibility of "super-optimal" cooperative solutions to the problem of environmental degradation for sustainable welfare. These types of solutions are technical ingenuities. A special level of understanding must exist for a person to suggest a super-optimal resolution to a dispute. It depends on the formulation of the problem, definitions, and language. Super-optimal solutions are not available with economic analysis. Expanded systems approaches that include economic considerations, and even discuss consequences in economic terms, may provide the framework for fostering creative project decision-making (Brill, 1979).

2.2.5 Fishery Management Example

In an effort to apply the concepts discussed in this paper, a discrete renewable resource economic, spawner-recruit stock resource model for harvesting of Antarctic Blue Whales (Spence, 1974; Conrad and Clark, 1987, Bender et al, 1994) has been selected for examining ecosystems which have market values, and also naturally regenerate. The form of the model's natural growth or regeneration pattern is shown in Figure 1, including the environmental carrying capacity (136,422) and 2 sample initial stock levels (30,000 and 100,000), positioned above and below the stock level for maximum sustainable yield (maximum slope). The validity of the model for policy development of Blue Whale harvesting is irrelevant. The purpose of using such a model in this context is not to provide a real application, but to describe general conditioning for internalizing unallocated property rights, and possibly shed light on innovative approaches toward resolving development issues and policy questions.

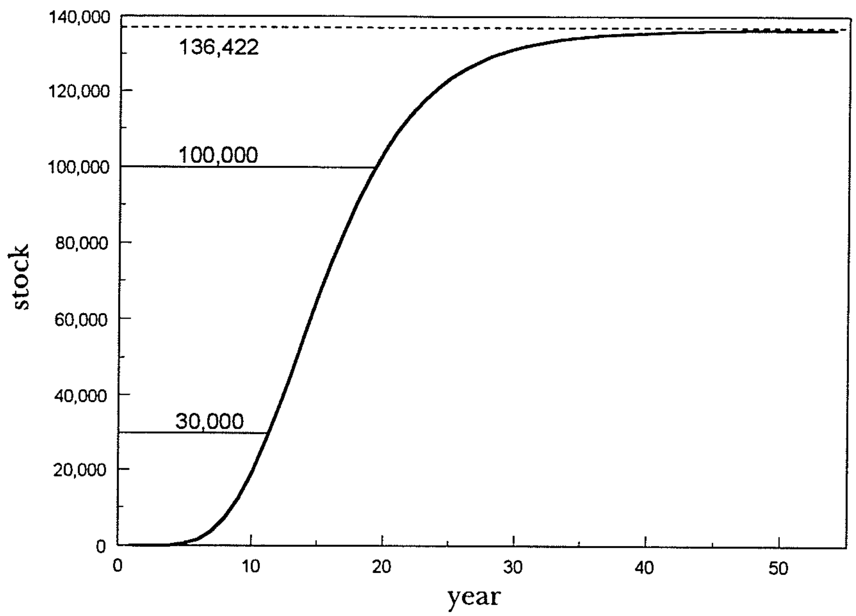

Figure 1. Natural growth function for fishery.

The Spence (1974) model is a discrete equivalent of the economic analysis presented earlier. The theoretical Hamiltonian approach is replaced by numerical solutions. Equations (14 to 17) describe the Spence model for the discrete model case. The value function, V_t, is given by (14). The objective function is then (15), where ρ is the discount factor (ρ=1.0 implies no discount or interest rate). The control variable effort, E_t, along with stock size, influences the harvest, Y_t (16). The stock size for the next time step can then be calculated by including the harvest component with the natural growth function (17).

$$V_t = pY_t - cE_t \tag{14}$$

$$M = \sum_{t=0}^{T} \rho^t V_t \tag{15}$$

$$Y_t = aX_t^b(1 - e^{-qE_t}) \tag{16}$$

$$X_{t+1} = aX_t^b - Y_t \tag{17}$$

Four unique scenarios were envisioned to encompass the potential decision space for policy development (Figure 2). They range from single decisions for constant harvesting effort (scenario 1), a "bang-bang" switching policy that uses 2 levels of effort to initially drive the stock to an optimal level, using either maximum or zero effort up to time ***S***, and then maintain the stock at that level (scenario 2), flexible decision-making on an annual basis (scenario 3), and finally segments of constant effort adjusted at regular intervals (scenario 4).

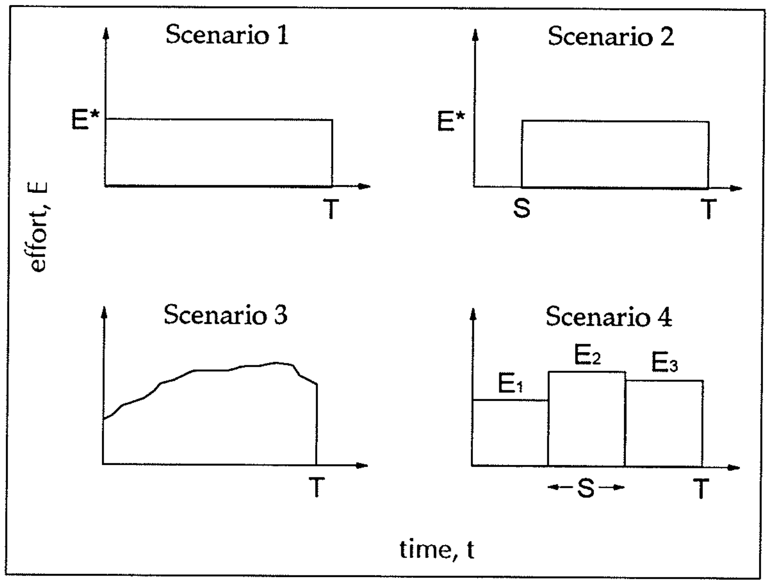

Figure 2. Policy decision scenarios.

For each of these scenarios, a number of optimal paths may be chosen based on initial resource conditions, planning horizon, and discounting rates. Scenario 1 contains only 1 decision variable, ***E****. The other scenarios, however, are much more complex and the shape of the decision spaces are difficult to visualize. An initial examination of optimal solutions for scenarios 2 to 4, as well as scenario 1 is necessary for reference material to baseline model behavior. The Box Complex nonlinear search (Beveridge and Schechter, 1970) was used to find optimal model conditions for a variety of discount rates and time horizons. A 50 year time horizon was chosen to examine 3 policy paradigms. It proved to be a reasonably long enough time frame for producing stable results.

Scenario 1 understandably produced the poorest economic results because only a single decision is allowed. If the initial fish stock is very low the amount of harvesting effort is restricted for a long time even if the stock level grows quickly. Scenario 1 is not flexible enough to take advantage of improved conditions in the future. Rather, it is a very conservative approach for initially low stocks, and degrading for high initial stocks and finite time horizons. Scenario 2 improves on scenario 1 because it allows depleted stocks to regenerate to some optimum before making a final decision on harvesting practices.

Scenario 3 produces the most efficient economic benefits. This should not be surprising because scenario 3 provides the most flexible decision framework. It allows new decisions for amount of effort each year including zero effort or maximum effort depending on stock size, whatever is found to be efficient. Finally, scenario 4 provides a decision-making framework somewhere between scenarios 2 and 3. As scenario 4 approaches S=1 it becomes scenario 3 and as S increases to some ideal length, the benefits more closely resemble scenario 2. The flexibility of scenario 3 is ideal, but quite impossible to maintain in terms of intensive, real-time, policy alternative selection. Scenario 2 is in many cases the intention of policy-makers because it is seen to be a practical approach that produces efficient results. A balance must be struck between the practical and impractical because most project decisions are made with finite horizons in mind.

A scenario 4 framework may provide an effective policy-making platform that bridges impracticalities in both time horizon and alternative selection intensity. In order to examine this possibility more closely, a comparison is made between economic solutions, solutions with increased marginal costs based on a sustainable investment model (Young, 1992), and a multiobjective (MO) welfare model after Simonovic (1989) to find the most robust solution. Economic solutions are generated using the Box complex nonlinear search algorithm for a 50 year horizon based on Spence's model (Spence, 1974). The sustainable investment model adjusts the original economic model by increasing marginal costs such that the cost of effort increased by 25%. Both of these models generated solutions for a number of discount factors {1.0, 0.95, 0.85, 0.5}. Not surprisingly, fish stocks are consistently higher for the sustainable investment model solutions, except for extreme discounting cases.

For the multiobjective framework to be compared with economic and sustainable investment results, a series of scenario 4 solutions are developed in which decisions are made every 2, 5, or 10 years denoted by $\boldsymbol{S}$. For a 50 horizon, 5 decisions are needed for $\boldsymbol{S}$=10 years. These 3 alternative forms of scenario 4 were examined for various degrees of uncertainty, expressed as the discount factor. This produces a multiobjective problem with 3 alternatives ($\boldsymbol{S}$ = 2, 5, 10) and 4 criteria ($\boldsymbol{\rho}$ = 1.0, 0.95, 0.85, 0.5). Compromise Programming (Goicoechea et al., 1982) was then used to choose the most robust economic solution given the number of criteria, and a variety of weight combinations. The objective function values within each criteria is shown in Table 1 for $\boldsymbol{S}$=2, $\boldsymbol{S}$=5, and $\boldsymbol{S}$=10 rows. Varying the weights changed emphasis on decision-making from no discounting to extremely high discounting. This provides for a range of possible resource states in terms of property rights, because the discount factor may represent a social discount factor, or describe the state of the fishery market from open access ($\boldsymbol{\rho}$=0.5) to impossibly perfect circumstances ($\boldsymbol{\rho}$=1.0).

The multiobjective model choice is made independently for the low initial stock case (30,000 fish) and the high initial stock case (100,000 fish), and without arbitrary or artificial environmental constraints. It is interesting to note that for low initial stock (a state of serious environmental degradation), it proved to be more robust to make decisions on a short time interval (every 2 years) while a high initial stock (a pristine state with few impacts to the natural ecosystem) recommended making decisions on a longer time interval (every 5 years). For an initial stock of 30,000 choosing $\boldsymbol{S}$=2 insured that positive actions were made more quickly. Choosing S=5 for initial stock of 100,000 prevented short term depletion of fish stocks and served as a hedge against great uncertainties or discounting.

Examining the stock for the 3 types of model solutions discussed (traditional economic model, sustainable investment model, and the multiobjective welfare model) suggests that even though the multiobjective model is expressed entirely in terms of readily available economic measures, it produced more stable, and overall improved environmental conditions with negligible economic losses. Figures 3 and 4 are examples of these comparisons. In Figure 3, the multiobjective model appears more environmentally-safe. Figure 4 adds an element of uncertainty in terms of discounting. This causes increased variability in fish stocks, especially for economic model solutions, which is consistent with an overall trend of increasing volatility for greater discounting. In Figure 4, the multiobjective model appears to be more desirable than the purely economic model solution because of more stability in stock behaviour. Added stability appears to be a consistent advantage for multiobjective model solutions over the economic model. In terms of the fish stock, the multiobjective model generally provides at least as high a stock as in the economic solutions. Of course, increases in harvesting costs in the sustainable investment model solution always produce more desirable environmental effects, but at a great economic cost to the fishermen. Table 1 shows the fishery's net economic benefits to be cut in half for sustainable investment framework compared to the economic paradigm.

It might be said that regulating the frequency of fish harvest policy decisions is a compromise solution between the economic and Young solutions. If the economic solution is too negligent, and sustainable investment is not attainable, then the most robust may be sustainable for the welfare model.

Table 1. Fishery solutions.

	$X_o = 30,000$				$X_o = 100,000$			
	1	0.95	0.85	0.5	1	0.95	0.85	0.5
Economic	716.2	183.9	34.5	0.5	854	304.7	117.9	39.9
Young	416.2	108.6	16	0	493.6	171	58.3	16
S=2	698.3	174.1	27.1	0	867.6	307.7	116.2	27.8
S=5	679.1	174.4	24	0	858.3	307.5	116.3	33.9
S=10	678.3	172.4	18.8	0	771.4	258.2	115.2	35.5

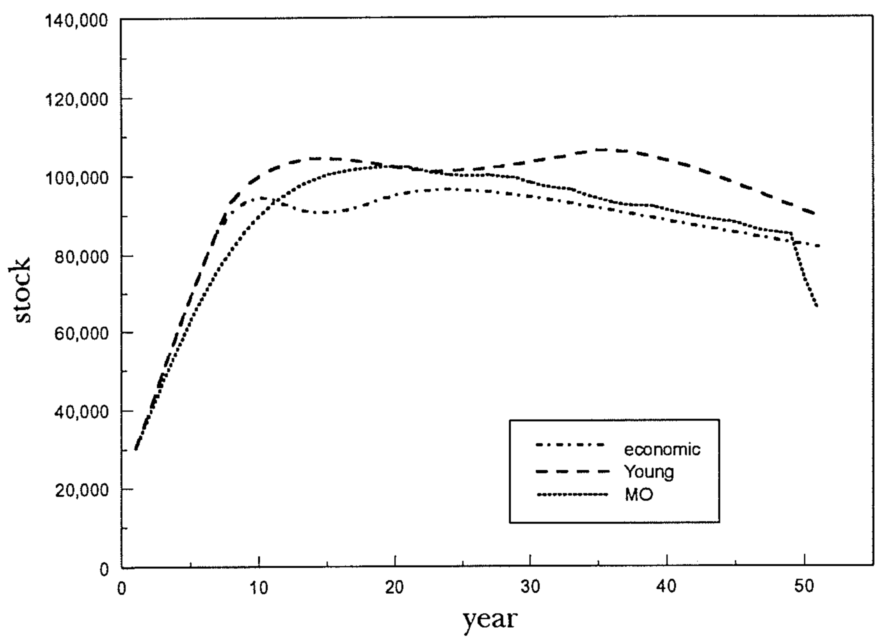

Figure 3. Comparisons for discount factor of 1.0, initial 30000.

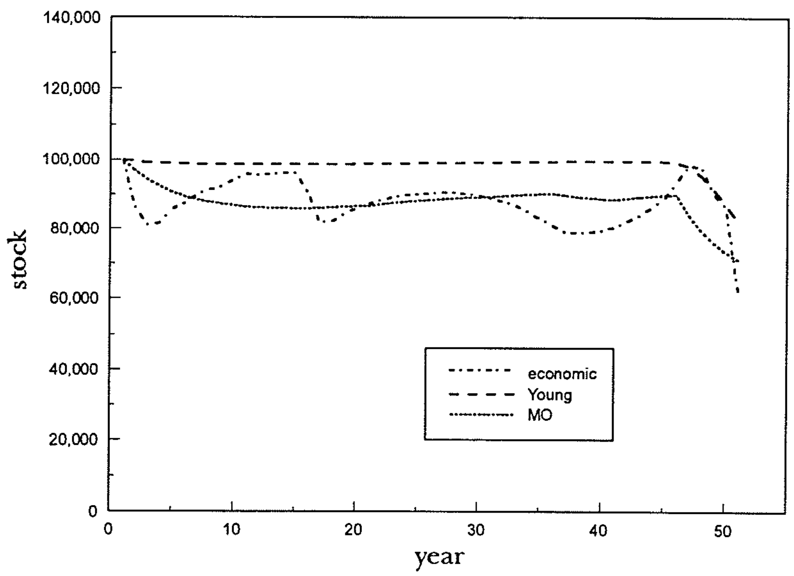

Figure 4. Comparisons for discount factor of 0.95, initial 100000.

2.2.6 Potential Policy Implications

Many difficulties exist in defining and implementing policies for dynamic renewable resources. Expanding economic analysis to a more generalized ecosystem approach requires a great deal of complication. Issues of uncertainty and risk are difficult to adequately address, the interactive behaviour of ecosystem components may not be well defined, and market effects from resource use may not be appreciated.

In a fishery, internalizing issues of concern alters the problem to a question of sustainable fish stock level, which may differ significantly from maximum sustainable yield. Solution of the sustainable fishery model demands satisfaction of criteria for all ecosystem components. Otherwise, the solution is not strictly sustainable, only potentially sustainable, increasing the risk of welfare reduction. Once a decision has been reached regarding property rights and market perfection, then all costs related to resource use must be incorporated at their appropriate levels.

This discussion in resource economics is based on partial system analysis. The advantage of considering noneconomic paradigms is that they are usually able to handle more complicated nonmarket systems. The potential acceptance of decision-making frameworks such as a multiple objective assessment of equity tradeoffs may signal the necessity for changes in our interpretation of economics and the market system. In application, decision-making may prefer the noneconomic paradigm while policy-making may prefer to track effects using an economic interpretation of social values.

Of the possible decision situations explored in the example, initial stock varied from very low to very high, and discounting varied (either real or perceived) from very low to very high. Whether quantitative knowledge is known about the behaviour of resources under either natural conditions and under human pressures, temporal manipulation of the way decisions are made may be a powerful tool. Management alternatives spawned from this knowledge may turn out to be considerably cheaper and more subtle in the control of individual actions.

In terms of practical decision-making, policy measures may need to restrict increasing changes in effort (ie. control entrance to markets). This forces not only a limitation on effort and stock effects but also to foster longer term decision-making. A stable fishing community may actually learn to circumvent many property rights transfer costs by regulating themselves to preserve their heritage and solve problems associated with use of the commons which are no longer considered an open access resource. In the context of (8), the fishing community may push the price toward the sustainable investment case by restricting harvests once they appreciate some of the added benefits of maintaining larger fish populations.

In using a multiobjective model, the form of the question to be answered was changed from "What physical aspects apply?" to "What temporal aspects apply?". In doing so, an entirely different set of alternative choices is created. This new decision-making problem can be viewed as one attempt to change the thought patterns of decision makers toward developing innovative, super-optimal solutions to complex problems of management policy. Allowing robust solutions to be accepted as a viable alternative selection process can also open the door to other creative management approaches. We shouldn't neglect these types of approaches if management models serve as a means of alternative generation and quantification for further selection negotiations. They are merely filling out the decision space.

2.3 Sustainability Issues

Recent discussions have advocated the adoption of sustainable development policies to ensure that future generations will enjoy a standard of living at least as high as today (WCED, 1987). Economic definitions are seen as one aspect in a holistic sustainable development paradigm. To a large degree, the formation of policies for sustainable development has been spurred on by the highly publicized paper: "Our Common Future" (WCED, 1987). Since then, engineers, geographers, economists, ecologists, architects, and politicians have been trying to shed some light on a practical definition of what it means to sustainably develop (Goodland et al, 1991).

Sustainable development is a popular rallying point for many academic disciplines, professional industries, regulatory agencies, and interest groups. Although the precise agenda for sustainable development is debatable, it has been used to justify a number of decisions. References to intergenerational equity, and other accepted aspects of sustainable development are too vague to be of practical use. Cultural, geographical, and political variations change the scope and scale of decisions in terms of resources to be managed, the decision makers involved, and the objectives used to assess tradeoffs among alternatives.

2.3.1 Multidisciplinary Perspectives

Economists have been studying sustainability issues from an economic perspective since at least 1952, developing definitions for optimal economic activity in terms of optimum state of conservation:

> *Somewhere, in conservation, an economically optimum distribution of rates of use over time is reached. This distribution we call the 'optimum state of conservation'. (Ciriacy-Wantrup, 1968)*

More recently Tietenberg operationalized this definition as dynamic efficiency, stating that:

> *An allocation across **n** time periods is dynamically efficient if it maximizes the present value of net benefits that could be received from all the possible ways of allocating those resources over the **n** periods. (Tietenberg, 1992)*

Beyond the above basic definitions economists have been investigating economic effects from implementation of environmental policy through use of theoretical models. Xepapadeas (1992) discussed effects of environmental policies in the form of emission charges or limits and was able to show behavioural differences in the short and long run. Barrett (1992) considered valuation of various environmental indicators to challenge whether environmental preservation would be desirable. Amir (1992) echoes this suspicious attitude by suggesting that, through a theoretical economic analysis, development decisions always result in negative environmental impacts. His comments arise from considering a more complete economic analysis which includes flows of externalities outside the firm. There is a need to consider these market impacts from potential policy implementation if a practical sustainable development paradigm is to be realized.

An example framework for achieving sustainability is found in Young (1992), which takes an ecological viewpoint in an economic framework, in defining required model components for the calculation of a resource price. The assumption made is that all the perceived important nonmarket ecological factors can be internalized. In this paradigm, the method of internalization is similar to that discussed by Baumol and Oates (1988) in their theory of externalities. Producers of negative externalities are required to pay to offset or alleviate effects. Beneficiaries of positive externalities must compensate for their use. Development of rules of use for open access or common property resources identifies additional benefits to be realized from removal of some externalities. Young does not discuss the costs involved with the exchange of property rights. Extensive transaction costs may be incurred in determining the sorts of required mitigation and restoration measures for externalities such as poor water quality. These costs may be identified through explicit definition of the necessary conditions for sustainability. Finally, the monitoring networks required to enforce new rules of use for common property resources needs to be implemented.

Geographers have joined the sustainability discussion from the vantage point of spatial and temporal implications of decisions (Pierce, 1992; Yin and Pierce, 1993; Niu et al, 1993). These are difficult problems to assess. Spatially, problems take on different meaning at different scales. Reasons for making a decision for a local river basin may be at odds with policy within the political boundaries, or misrepresented at a global scale. Temporally, difficulties arise from trying to represent unborn generations in establishing equity for decision options within long time frames.

The concerns of geographers led them to identify spatially or socially disparate focus for sustainable development. Global aspects of wealth distribution between countries is tempered with questions of local community viability. Two distinct, although not mutually exclusive, viewpoints are identified. One is utilitarian. The other is ecological. The global-scale conflict between these view points is income redistribution versus income growth. The local-scale conflict is in terms of the specific environmental aspects that are desirable. Together, global and local issues question how to improve overall social welfare.

Utilitarians are often mistaken for economists or exploitists. Ecologists are emerging from an environmentalist stereotype. Both, from different directions, are moving toward a common goal (Regier and Bronson, 1992). Interdependent systems and relational links between different ecosystems is the common understanding that drives both extremes toward looking for relevant information. Unfortunately, it is extremely difficult to assemble, interpret and communicate relevant information. The approach to tackle this problem can be to develop reasonable indicators for ecosystem health (Costanza et al, 1992), although many indicators such as economic ones can be very misleading (Tinbergen and Hueting, 1991).

Integrated environmental management is a stark contrast to theoretical economics. Integrated management of ecosystems include environmental, social, and economic systems. Integrated approaches consider economics as one part of a much larger picture (Iverson and Aston, 1994). It also introduces the human element in a leading directional role (Geller, 1994). Cairns et al (1991, 1994) provides a number of case studies discussing problems and approaches for integrating decision-making concerns at the industrial, regulatory, and social levels.

Ecosystem health (Costanza et al, 1992) relies on complex studies and correlation of variables to determine broad indicators for ecosystems. These indicators are utilized within a framework that defines ecosystem health in terms of: *dynamism*, *relatedness*, *hierarchy*, *creativity*, and *differential fragility*. Dynamism and relatedness recognize that processes within systems are constantly in flux, and are not independent. The hierarchy of processes, which may contain discrete objects, is organized mainly by temporal and spatial scale. Creativity results from energy flowing through systems of processes allowing self-organization by repetition and duplication. Finally, differential fragility acknowledges that each of these interrelated systems of processes react differently to external (human) disruptions.

The benefit of multidisciplinary indicators is that they adapt to multiobjective and group decision tools. The inclusion of multiobjective analysis in decision-making paradigms is not a new concept for Sustainable Development. Systems analysis interpretation envisions a holistic approach (Haimes, 1992) that includes multiobjective analysis, risk analysis, impact analysis, scope consideration for selection of multiple decision makers, and accounting for interaction among the various ecosystem components.

2.3.2 Sustainability in Water Resources Management

The water resources management community has begun to assess the applicability of sustainability ethics for planning and operation of water projects (Loucks, 1994). Several countries have, or are currently developing guidelines or criteria for sustainability. In Canada, the Canadian Society of Civil Engineers have presented and are debating guidelines for sustainable practices (Johnson et al, 1994; Mitchell and Shrubsole, 1994). Although they are general and often redundant, they represent a real effort toward addressing environmental, social, and economic impacts of engineering projects. The same has been done in other countries such as Australia (Institution of Engineers, 1992). The focus in these guidelines is to expand the scope of responsibility for engineers, including water resource engineers, to consider multidisciplinary impacts and include interdisciplinary specialists (Loucks, 1994).

Water resources presents problems in analysis and management that may be unique from most other management systems. The types of assumptions, uncertainty, and interactions with other physical, social, or economic systems can pose challenging questions for management strategies and policy paradigms. The use of water has implicationss for nearly every ecosystem imaginable. Water can determine where people live, and it can instigate wars. Water can heal the sick, or devastate everything in its path. Water has direct links with every biological ecosystem on the planet, is essential for many social systems, and eventually filters into most economic markets.

The analysis of water systems is dependent on 2 general conditions. One is that all quantitative interactions are either known or negligible. The other is that our analytical mechanisms are a reasonable facsimile of reality, for transport or mixing for example. These 2 assumptions cover the basic limits in the hydrologic cycle. Analysis, based on these assumptions must accommodate uncertainty in both the supply and demand of water, which generates implications of risk in each decision.

We typically consider water demand for human or societal consumption, but there are many ecosystem processes that exert a direct demand for water. Infiltration and evapotranspiration are two simple examples that are normally included in hydrologic models. Animals, small organisms, and vegetation also exert demands. Uncertainty in all of these demands affects the temporal and spatial availability of water. However, uncertainty in demand can be limited or reduced with intervention.

Uncertainty in water supply is a much more difficult problem. Freshwater is supplied to a watershed or river basin either through rainfall, or snowmelt, which in turn is transported and stored either on the surface or as ground water. The amount and timing of supply is extremely uncertain. Engineers can spend a great amount of time trying to reduce the uncertainties in water supply to limit associated risks. It is a difficult, and somewhat unique task. Correlation of supply variation is difficult because it is entirely driven by random events and transported by mechanisms with different lag times. Variations for both quantity and quality of water occur for daily periods, in seasonal or annual cycles, and are affected at much longer periods by global processes. Every decision concerning the management of water includes all of the uncertainties and associated risks for each ecosystem that depends on the supply of water.

Water resources planning recognizes the interconnectedness of ecosystems, and the need for collaborative efforts to make multiobjective decisions. Current conceptual thinking about sustainability in developing water resources relates to handling risk (Plate, 1993; Haimes, 1992) and preventing adverse conditions. Risk is an issue in water resources due to uncertainties in water supply through rainfall runoff or snowmelt, subject to water demands and losses. This natural uncertainty translates to risk of system failures such as power and potable water shortages. Future planning may introduce promotion of positive impacts while reducing negative effects.

Recent directions in project evaluation includes objectives which are difficult to quantify in commensurate units. This is especially true for social and environmental impacts. For engineers planning water development, project alternatives are traditionally screened by benefit-cost analysis. The highest economically-rated project is selected. Lund (1992) notes instabilities in using benefit-cost ratios in certain circumstances. More stable evaluation of project economics may be achieved with net present value calculations. Multiple objectives with noncommensurate units of measure are a challenge with the net present value approach, but analysis of renewable resources is common (Conrad and Clark, 1987).

2.3.3 Sustainability Implications on Project Planning

There is a gap between abstract definitions of sustainability and the practical issues of each problem. One way to resolve this gap is through more detailed criteria for defining an agenda for sustainability (Simonovic et al, 1995). There may exist a set of measurable criteria that could allow assessment of overall sustainability, and be sensitive to the scope and scale of a decision context. Besides the difficulty in finding and compiling these criteria, the question is: who decides? Who is allowed to choose the criteria? A government body? A global organization? Should there be sets of criteria ranging from local interests to global interests? Another direction toward sustainability is to prepare processes that foster creativity in the selection or preparation of alternatives. Consideration of processes for creative decision-making suggests that stakeholder participation is important for achieving sustainability.

Another possibility for societies to move toward sustainability is to prepare processes and tools that foster creativity in the selection or preparation of alternatives. Processes for creative decision-making may span disciplines and problem types. Tools will very likely be specific to a problem type. Development of processes and tools for creative decision-making, more than anything else, suggests that participation of stakeholders in the decision process is important for achieving sustainability. Sustainable decision-making that relies on criteria or indicators can only achieve stakeholder participation as far as selecting weights for each criteria - assuming regulation of indicators/criteria allows for manipulation.

Conceptually, sustainable design of projects can be treated differently from sustainable management of infrastructure. Assuming that we have done a poor job of decision-making for sustainability, a practical application of sustainability in development and management of resources will need to focus on 2 types of problems. One is for design of new projects, the other type is remediation of existing infrastructure. Certainly, its much more flexible to plan a new project than to work around previous decisions. At least, a more efficient and sustainable set of solutions are expected. In reality, there may never be such a simple division of project types. The number of prior decisions produces a set of constraints that grows with the number of decisions that have already been made, and project classification may become fuzzy. If there exists a specific set of sustainability criteria for the problem, the difficulty in achieving satisfaction of those criteria increases with the number of constraints on the project. Of course, in all problems there will be legal, political, economic and technological constraints that may be independent of prior project decisions.

For projects that can generally be defined as remediation of existing infrastructure, options are limited to technologically creative solutions for satisfaction of sustainable criteria. New devices, arrangements, or processes may be developed to achieve design specifications due to a lack of options. Creativity of a different sort is required for new projects, where impacts tend to be broad and remedial work focuses more on specific impacts and causes. For new projects, technological innovation is less likely for the simple reason that impacts are too widely varied to focus on a specific technology to solve the problem. Potential improvement of solutions is found in the integration of disciplines using existing technologies. Other projects that are difficult to classify as new or remedial may be solved by a combination of technological and interdisciplinary creativity. Figure 5 demonstrates some differences between project types. Project types in Figure 5 are compared according to several categories or conditions. For example, one would expect existing projects to include a large number of prior decisions, have a relatively large number of constraints in terms of options, and impacts can be expected to be well-defined (specific).

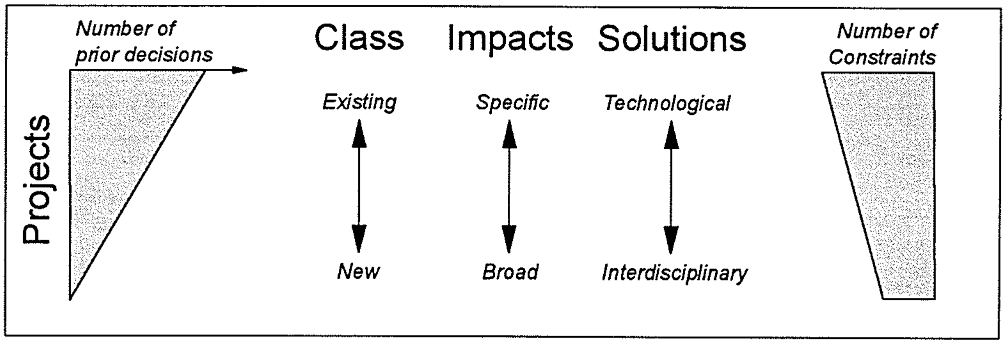

Figure 5. Contrasting types of projects.

The technological creativity used to produce sustainable practices for existing projects is heavily dependent on expertise and insight from a professional in the appropriate discipline. An interdisciplinary solution for a new project may require that options be made available, and that an appropriate team concept or group planning process be in place. The availability of interdisciplinary design options may be very complex and extensive, and a number of difficulties are related to communication between disciplines. The planning process however, depending on its desired effects, has a great potential for inspiring decision tools that specialize in making multiobjective decisions with an unspecified number of discrete alternatives.

The question of compiling a sustainable regional (or local) alternative in a proponent-initiated circumstance is quite different from a public decision problem of choosing from a list of prepared alternatives. Proponent-driven planning is a subclass of public planning because a proponent presents one proposal to a licensing board. In public decision-making, alternatives for development are generally predefined, and decisions are reduced to a question of suitable weighting based on a number of decision criteria. Many forms of multicriteria decision-making tools are available to evaluate tradeoffs between predefined proposals. However, the multidisciplinary development of alternatives, and criteria for iterative alternative rating is quite another problem. In many instances, a project is initiated by a proponent who wishes to develop within a very specific part of the potential decision space. This type of proposal represents a single alternative (or a small number of alternatives). In the simplest case, the public planning problem becomes a choice of 2 alternatives: the proponent's proposal, and the *status quo* (or rejection of proposal).

A proponent of development would like to propose a single proposal within a portion of the decision space for some resource. A proponent proposal must compete with other alternatives, including the option of leaving the resource undeveloped. Sustainability criteria are supplied by some form of project review process that may allow licensing based on interests external to the proponent such as political, social, and environmental issues. This approach will need to provide at least the potential for sustaining those aspects of life that seem important to people outside the interests of the proponent.

For a proponent, building a single proposal demands a complex selection of components with wide-ranging implications that may need to be addressed before venturing on to the licensing process. There is a need in the decision process, even if the scope pertains only to hydroelectric development, to identify and address some of the possibilities besides the currently framed proposal to improve the probability of acceptance at licensing.

Because proponent-driven planning is a subcomponent of the public planning problem, it is up to the proponent to formulate an alternative with the added objective of satisfying public goals. The proponent must compete with other alternatives, but also with the *status quo* option. As public involvement in the project licensing process becomes more powerful, a balance in project planning objectives tends to be shifted toward stakeholders that are traditionally in an adversarial position relative to the proponent. This adds implications of the public decision problem to the early planning process, and may foster more cooperative proposals as opposed to partial analysis and difficult adversarial negotiations. In other words, proponents will need to be more aware of the shifting public goals to stave off rejection at the time of licencing or at a future relicensing.

2.3.4 The Evolution of Sustainability

How can sustainability be achieved? How do we measure sustainability? What combination of axioms or criteria should be used? Most are not yet operationalized, and quite nebulous. Environmental Integrity, although a noble cause, is difficult to define in an operational form. Intergenerational equity is nearly incomprehensible except as a combination of more realistic terms such as reversibility, resiliency, or robustness. Unlike the above concepts, economic efficiency is somewhat understood and is commonly applied - yet it is subject to many externalities with real or potential economic values not counted in the justification of development. Even if they are counted, special care must be taken to assign proper "perspective" or weight.

Other concepts used to assess planning decisions in terms of sustainability include risk, equity, reversibility, resiliency, and vulnerability (Hashimoto et al, 1982; Simonovic et al, 1995). Risk has measurable qualities, as long as the proper risk events are identified. Equity has meaningful forms of evaluating the distribution of benefits, but what about equity between an energy consumer and a resident affected by the flooding of a reservoir for hydropower? To evaluate equity, commensurate (dis)benefits must be compared. Reversibility, resiliency, and vulnerability are similar. All three are subject to time frame considerations. Of course, in geologic time scales, (almost) everything is reversible. Another axiom for sustainability is robustness. Robustness implicitly handles some of the above sustainability axioms such as risk, and it is a reasonably intuitive result of sensitivity analysis. It also has flaws, however. One is the assumption of optimality as the driving force behind evaluating performance. That is, a robust solution performs reasonably well under a range of possible conditions. A robust solution must be judged against an ideal state (relative to other solutions). What will be our definition of an ideal state?

2.4 Consensus Sustainability

The practical implications of sustainable development seem to be elusive. Metrics for measuring the level of sustainability are eagerly sought. It is very difficult to capture the important features, and maintain many of the valued details, of environmental and social systems when complete or appropriate economic accounting is attempted. A much more transcendent and flexible metric may be *consensus*. Consensus as a sustainability metric describes the level at which stakeholders are satisfied with a solution to a question. Consensus assumes that an appropriate group of stakeholders is able to collaborate in assessing proposed solutions to environmental problems, or development initiatives. It also assumes that the collective best a group of stakeholders has to offer implicitly provides insight to the needs of future generations. This section describes an approach for assessing a level of consensus. The benefits are: estimation of areas of common understanding; and clarification of conflicting values.

2.4.1 Concept of Consensus Sustainability

Consensus, as the concept for promoting sustainability in decision making, is a criteria quite unlike many of the other axioms previously described. Consensus has no units of measure. It is measured in a brief moment of time, but may implicitly consider future events and uncertainties. Consensus is a high level indicator, dependent on value judgements which may in turn depend on lower level indicators derived from facts concerning problem characteristics.

The definition for consensus in Webster's Dictionary is:

A general agreement in opinion.

It relies on a qualitative and subjective opinion, and the qualifying condition is a general agreement. Who is making the agreement? How well do they need to agree?

Sustainability can be defined in many ways. If consensus leads to sustainability, what is consensus sustainability in an operational form? Let us start by giving the following definition for consensus as it relates to sustainability:

> *Consensus is an equitable compromise which is robust with regard to a) resource management uncertainties, and b) stakeholder perspectives.*

This definition is not yet operational, but its constituent parts might be manageable. There are some assumptions which also need to be made. It is assumed that appropriate stakeholders have been included in the decision process. By stakeholder, we refer to interested parties which may be impacted in some way by any decision that is made (a political choice). The second major assumption is that all stakeholders voluntarily cooperate in the decision making process. Of course, this is dependent on those involved. An appropriate set of stakeholders could bring all of the important issues to bear on the decision, circumventing the economic market model which does not handle externalities (actually the market model would explain this as internalizing the issues).

Traditional group techniques for multicriteria decision analysis make extensive use of subjective weighting to show level of importance for both different criteria and also for the decision makers (stakeholders). The result of these 2 sets of weights is the reduction of a difficult multiobjective problem to a straightforward optimization. In the discrete problem, the weighted alternatives need only be ranked in reference to some ideal solution.

There exists a choice in approach. Decision makers may explore their own values in choosing individual weighting schemes for criteria. For each experimental choice in value, however, ranking alternatives is still a multicriteria problem. The choice to be made is whether to cast judgement on the importance of an individual, or continue to explore solutions in search of better consensus among participants. The following example from Bender and Simonovic (1996d) illustrates the consensus approach.

2.4.2 Degree of Consensus Example

An illustrative example in water resources planning can be found in Simonovic (1989). In the former republic of Yugoslavia, there existed a set of alternative solutions to be applied to a system of reservoirs in Serbia. Each alternative is judged using a set of criteria which are rated by a set of decision makers. The problem size is as follows:

- 6 alternatives
- 8 criteria
- 6 decision makers

An ordinal ranking of the alternatives is achieved using the distance metrics of compromise programming (Zeleny, 1982) which provide a strong ranking of the alternatives. Compromise programming will be described in more detail in Chapter 7. For the purposes of this degree of consensus example, it is enough to know that compromise programming is a multicriteria decision analysis technique that produces a ranking of alternatives. Three distance measure definitions are used to evaluate alternatives, defined by an exponent, p:

- p=1
- p=2
- p=10

In order to apply this distance metric technique, decision makers must choose a weight to describe the importance of each criteria. Unfortunately, there are several decision makers - each with their own priorities. As each decision maker uses their individual set of weights, the rankings may change. In fact, as the form of distance metric changes, the rankings may also change. The choice of alternative is no longer a straightforward decision that results in a strong ordinal ranking. Each set of weights and each choice of distance measure provide a strong ranking, but there are uncertainties in ranking related to subjective priorities. In Simonovic (1989), alternative 5 was eventually selected based on rankings similar to Table 2. For the most part, the differences in ranking for the 6 sets of weights (provided by the decision makers) is not very dramatic. There are some discrepancies, but there are no further steps in the decision process and alternative 5 is selected because it appears to be robust in terms of choice of priorities and choice of distance measure.

Table 2. Ranking of Alternatives using Compromise Programming.

Distance measure	Decision maker	Rank for alternative 1	2	3	4	5	6
p=1	1	6	3	5	4	2	1
	2	6	5	3	4	2	1
	3	6	3	5	4	2	1
	4	3	6	5	4	2	1
	5	6	5	3	4	2	1
	6	6	3	5	4	2	1
p=2	1	5	6	3	4	2	1
	2	5	4	2	6	3	1
	3	5	3	6	4	2	1
	4	5	3	6	4	2	1
	5	5	4	2	6	3	1
	6	3	6	5	4	2	1
p=10	1	5	4	6	3	2	1
	2	5	2	4	6	1	3
	3	5	4	2	6	3	1
	4	5	4	6	3	2	1
	5	2	5	4	1	6	3
	6	5	4	6	3	2	1

In a consensus-based approach for achieving sustainability, the decision process becomes iterative, using feedback to evaluate progress in discussions among decision makers. The previous distance metrics can be used to assess degree of consensus among decision makers. Degree of consensus indicates the level of agreement with the ordinal ranking of each alternative. That is, the worst alternative may have a high degree of consensus because everyone agrees that it is the worst alternative! Of course, a composite overall degree of consensus can also be found.

The following are 5 measures for degree of consensus found in the literature (Kuncheva, 1994):

$$\gamma^1 = 1 - \min_{i \neq j} \left| w_i x_i - w_j x_j \right|,\ i,j = 1,\ldots n \qquad (18)$$

$$\gamma^2 = 1 - \max\left|w_i x_i - w_j x_j\right|,\ i,j = 1,\ldots n \tag{19}$$

$$\gamma^3 = 1 - \tfrac{1}{n}\sum_{i=1}^{n}\left|w_i x_i - \tfrac{1}{n}u\right| \tag{20}$$

$$\gamma^4 = 1 - \tfrac{2}{n(n-1)}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left|w_i x_i - w_j x_j\right| \tag{21}$$

$$\gamma^5 = 1 - \max\left|w_i x_i - \tfrac{1}{n}u\right|,\ i = 1,\ldots n \tag{22}$$

$$u = \tfrac{1}{n}\sum_{i=1}^{n} w_i x_i \tag{23}$$

where n is the number of decision makers, x_i is the distance metric for decision maker i, w_i provides parametric control and possible weighting of decision makers, and $\gamma^k \in [0, 1]$ is the degree of consensus measure for an alternative, indexed by $k \in [1, 5]$.

The degree of consensus indicates the relative strength of ranking. In cases where complete transitivity is not achieved in the rankings, a weak ranking exists. Degree of consensus suggests the relative degree of transitivity in the rankings. Table 3 shows the degree of consensus for each alternative in the case study, for all 5 consensus measures. Each measure captures a certain aspect of level of agreement. For example, measure γ^1=1 if at least 2 decision makers agree on the rank (actually, the value of the distance metric). The values for γ^k vary, but they all appear to reasonably represent the level of agreement in the rank of each alternative.

Table 3. Degree of Consensus measures for Simonovic (1989).

Distance measure	Alternative	Consensus measure 1	2	3	4	5
p=1	1	1	0.75	0.94	0.9	0.86
	2	0.99	0.81	0.95	0.92	0.9
	3	0.99	0.8	0.95	0.91	0.9
	4	1	0.9	0.97	0.96	0.94
	5	1	0.87	0.96	0.94	0.92
	6	1	0.84	0.95	0.93	0.91
p=2	1	1	0.73	0.94	0.89	0.86
	2	0.98	0.79	0.94	0.91	0.88
	3	0.98	0.73	0.92	0.88	0.84
	4	1	0.87	0.97	0.95	0.93
	5	1	0.83	0.96	0.93	0.91
	6	1	0.79	0.93	0.9	0.86
p=10	1	0.99	0.69	0.88	0.85	0.79
	2	1	0.66	0.88	0.82	0.82
	3	1	0.67	0.9	0.85	0.77
	4	1	0.73	0.93	0.89	0.82
	5	1	0.75	0.94	0.9	0.85
	6	1	0.68	0.9	0.86	0.78

Using consensus as the measure for sustainability, decision makers have the opportunity to explore their values with different sets of weights to find a robust solution. Decision makers also have an opportunity to evaluate the strength of their decisions as negotiations progress. Encouragement of iterative, interactive feedback to a negotiation process is motivated by possible spontaneous creativity in resolving differences of opinion. Other searches may identify clustering or grouping of individuals in terms of their ranking. Advanced use of degree of consensus may even identify aspects of the system as candidates for adaptation, as an attempt to improve the nondominated frontier of solutions toward more sustainable solutions.

2.4.3 Summary

Degree of consensus, as a measure for achieving sustainability, provides insight into the level of agreement between the set of interested or affected stakeholders. The iterative process which this measure promotes may also provide insight into specific issues for which to focus planning resources.

The consensus measure of sustainability is dependent on the inclusion of relevant stakeholders. A major assumption is that this group of participants in the decision process represent all of the related sustainability concerns about the environment, future agendas, system uncertainties, etc. The participants themselves become the instruments of sustainability. Degree of consensus monitors progress in the search for a sustainable choice, or for creative change.

2.5 Frameworks for Resolving Conflicts

Consensus sustainability assumes that stakeholders are voluntarily copperative. This is certainly not always the case. Even if stakeholders cooperate, there may be many sources of conflict between them. The ability to resolve conflicts is of primary concern for a paradigm of consensus sustainability. There are many frameworks for resolving conflicts. The most obvious is the current judicial system. Quite frankly, lawyers fees are one transaction cost which proponents to development would like to avoid!

2.5.1 Alternative Dispute Resolution

Alternative Dispute Resolution (ADR) is a general title given to methods of resolving conflicts outside the judicial system. A standard text on the subject is Bacow and Wheeler (1984). In many cases, sending a conflict through the court system can be extremely expensive. As a result, alternative methods for resolving conflicts outside the courts is growing rapidly. In water resources management, conflict resolution techniques are actively pursued (Priscoli, 1988). There are several general categories, many specialized forms of each, and hybrid techniques which combine aspects of some of the main categories. The types of interests that must be met to achieve a durable solution have been outlined by Priscoli (1990):

- substantive - money, time, resources
- procedural - the way something is done
- relational - how one is treated, or conditions for ongoing relationships

Probably the most basic form of ADR is *Negotiation*, which is based on the conflicting parties speaking directly and coming to an acceptable compromise. All other ADR processes involve intervention of some form. The least intrusive form of third party involvement is *Mediation*, which adds to the negotiation a neutral party to help work through some of the issues, and has been applied to hydropower development disputes (Moore, 1991). However, the mediator has no power to impose a solution. Mediation is a technique that has also been used to involve public interest groups and settle licensing disputes in water resources development (Moore, 1991). Negotiation is another method (Coughlan et al, 1993). Both mediation and negotiation are more direct approaches to resolving disputes. They tend to come into use before more formal third party involvement such as some form of arbitration hearing. There is currently some effort to explore ways of changing the questions that are being asked. Negotiated Rule-Making (Fiorini, 1990) is an adaptation of negotiation that asks the conflicting parties to resolve policy questions. The difficulty with settling on a single set of statements that defines overall policy is that it becomes too general, with broad implications in political theory, economic theory, etc.

Some techniques distribute decision-making authority by using an arbitrator. *Arbitration* and other similar techniques such as Private Judging demand that the conflicting parties agree to abide by an arbitrator's ruling and, in effect, give up some rights to pursue more legal action. Other ADR techniques involve more outside parties. Summary Jury Trials, Mini-Trials, and more contemporary processes such as Citizen's Panels (Crosby, 1987), use unaffected individuals to present opposing cases or available options in the hope of getting unbiased responses. Citizen's Panels place authority with the public through representatives who are chosen randomly. The randomness is a direct attempt to remove some of the relational dynamics of direct negotiation. In that way, citizen's panels differ from a public utilities board such as in the province of Manitoba. A public utilities board is similar to traditional arbitration or mediation. A similar approach, Citizen Advisory Groups, has been explored by the US Army Corps of Engineers (Priscoli, 1975). Some of these latter techniques, such as mini-trials, are not strictly alternatives to the court system. In many cases they are simply preludes to legal action, or even the first steps toward a court case.

Another possibility is the generation of better solutions through cooperative, or joint problem solving. Fisher et al (1991) provide a general framework for negotiation. Fisher et al (1991) focus on integrative negotiation, or cooperative consensus generation. It involves exploring the decision space in a logical, rational manner. Most importantly, it provides a forum for increased understanding of different perspectives or values, and enhances communication between the disputants. It achieves these philosophical goals by avoiding bottom lines, and strategic positioning. They provide 5 points for improving the quality of discussion towards an agreement:

- Separate the people from the problem - individuals tend to bargain from predetermined positions and value systems. Provide a setting that minimizes judgmental stereotyping, and adversarial poses.
- Focus on interests, not positions - bargaining over positions, or prices, may not allow the underlying problem to be examined.
- Work together to create options - examine those underlying issues without threats related to bargaining power or leverage.
- Use objective criteria - maintain focus on the constructive examination of problem issues.
- Know your best alternative to a negotiated agreement (BATNA) - be aware of the incentives to negotiate such as potential gains in time and reductions in transaction costs for transferring property rights.

2.5.2 Game Theory

Game theory is a mathematical construct for analyzing conflicts and consequences of decisions, presented more thoroughly by Luce and Raiffa (1957). Game Theory is the basic tool for evaluation of multiobjective multiple decision maker problems, where multiobjective analysis generally handles single decision maker problems. Games can be cooperative or noncooperative, where cooperative game analysis allows for binding agreements and the formation of coalitions (Isreal et al, 1994). Conflict analysis is an extension of metagame analysis developed by Howard (1971). Application of conflict analysis requires identification of players and their options at a specific point in time, construction of preference vectors, and stability analysis.

Decision analysis or conflict analysis is an extension of game theory, a prescriptive tool that describes the reasoning behind strategic plays within a zero-sum, or fixed-sum game where the goal is to maximize one's own gains (Goldberg et al, 1985). Typically, negotiation can be broken down to two aspects, integrative and distributive (Bingham, 1986). Game theory emphasizes distributive negotiation. Fisher et al (1991) focus on integrative negotiation, where a play within the decision game is a pareto-optimal move within the decision space, benefiting both or all players. Decision analysis deals with distributive bargaining which assumes one person's gain is offset by losses to another player. The biggest problem with successful development of integrative negotiation is reducing the influence of distributive bargaining. The 5 steps in the process of Fisher et al (1991) are designed to minimize, or even remove, distributive effects. Whether this can be accomplished is a matter of effective mediation.

Game theory applications in water resources management have a limited, but growing history. Rogers (1969) applied game theory for evaluation of alternative river basin management strategies. Other applications concentrated on allocation of costs in multiple purpose water resource project development (Isreal et al, 1994). Recent focus has been resolution of water resource conflicts (Lussier, 1989; Chadderton, 1992), environmental negotiations, dispute resolution, and regulation enforcement (Kilgour et al, 1992; Fukuyama et al, 1994).

Another, more interactive approach to decision making using conflict analysis, has been developed using graph theory concepts (Kilgour et al, 1987; Fang et al, 1993; Fang et al, 1989; Fang and Hipel, 1988) to reformulate normal noncooperative metagame analysis. Stability analysis, applied in a similar manner to many branch-and-bound search techniques, has been shown to be very powerful within the graph theory architecture. It has been demonstrated in water resource environmental disputes such as in Fang and Hipel (1988) for the Garrision Diversion Project.

2.5.3 Public Participation

Public involvement in project decision-making is becoming increasingly important as an integral part of the planning process. In water resources planning, public involvement is an important aspect because water is a common property resource, and there may be many interest groups who are affected by changes in quality, quantity, or distribution.

Communications and cooperation with interested or affected stakeholders is also becoming increasingly prevalent with increased populations and development of available resources. In Tennessee, the Tennessee Valley Authority (TVA) has moved from controlling water to managing the water resource, including the management of diverse water-related issues affecting people (Herrin and Whitlock, 1992). There are many utilities such as TVA that are now making a great effort to proactively involve and educate the public, including Manitoba Hydro. How to approach this is a current topic of discussion (Vearil, 1992; Creighton et al, 1983), but is not a new one.

Public perception and attitude has been viewed as a planning tool for some time (Dasgupta, 1976; Beatty and Pierce, 1976) in water resource development. Priscoli (1990) notes that public involvement is not strictly educating the public, but also being educated by the public.

Collaborative participation is seen as a direct approach to conflict resolution similar to negotiation or mediation. The idea of collaborative negotiation focuses on the following types of interest to satisfy in order to attain a durable settlement (Lincoln, 1986):

- Substantive - needs, money, time, resources.
- Procedural - needs for specific types of behaviour.
- Relationship or psychological - how one is treated or conditions for ongoing relationships.

Collaborative problem solving is a relatively unstructured ADR approach that philosophically fits between negotiation and mediation (Priscoli, 1990). Public involvement efforts in general, is embracing more direct contact and discussion with stakeholders. The idea of using more collaborative ADR approaches is becoming more popular. This includes the incorporation of decision support systems (Vearil, 1992) but also debates about the form of negotiation (Susskind and Cruikshank, 1987) and role of the participants (Wilds and Lamb, 1986).

Mediation or a collaborative framework in encouraging participation and communication, demands that the role of mediator (Susskind et al, 1987; Fiorino, 1990; Clark et al, 1991):

- Understand perspectives of stakeholders
- Provide active direction
- Produce accepted facts, developed collaboratively
- Provide feedback
- Consider alternatives
- Be open and flexible

A public hearing process is one attempt to represent the interests of all affected parties before a project is licensed, to force proponents to provide justification for development and answer to any external concerns. The weakness in such a process is that the proposed development plan has little flexibility in its presentation form. This makes it difficult to accommodate legitimate concerns, and may force the proponent to endure a large expense to alter the proposal or to produce a new plan. The resulting process may be extremely costly in time and money for all participants. For the most part, this procedure is adversarial, and equitable compromises are difficult to make. A public utilities board for hydro development is a form of arbitration. Complete satisfaction of all stakeholders may be unreasonably difficult to accomplish, but the forum for participation may facilitate common understanding and an increased possibility of finding a more socially efficient solution.

Any forum for communication must facilitate or alleviate many intangibles, if an alternative dispute resolution mechanism is to be applicable to the complexities and diversity of water planning problems. Each person participtating in a negotiation setting brings their own unique blend of skills, perception, and understanding of decision options. How stakeholders interact, or are allowed to interact may have an affect on the success of communication between parties. There are several aspects that have been identified as being necessary for successful negotiation (Burkhardt et al, 1994; Ryder and Taylor, 1993; Susskind and Cruikshank, 1987):

- Definition of issues
- Urgency
- Need to negotiate

- Representativeness
- Technical clarity
- Power
- Sense of fairness

Personal bias can be described by attitude and cognitive style, and may impact the progress of collaborative efforts, depending on the decision making structure (Miller, 1983). Cognitive style refers to how people approach problems. Problem solving abilities can be further described using "conceptual level" which is a person's capacity to cope with complexity, conflict, and uncertainty. The influence of bias on planning can range from considering a restricted range of alternatives to misplacing emphasis on issues. In general, stakeholders may limit a group's ability to reach consensus about a problem (Miller, 1985; Miller, 1983) depending on:

- Self insight
- Cognitive style
- Understanding of the nature of the problem

One impediment to progress in negotiations is the aspect of trust. Stakeholders can be extremely suspicious of each other, and of the results of field surveys or model analysis. Even if results are presented honestly, the design of studies may be biased and leave gaps in important areas. The form of communication is crucial in dealing with this important problem. Computer-aided planning, for example, is a realistic option for a number of negotiated problems. The attraction toward computers is their data handling capacity. Computers are supposed to be cold, hard, unbiased pillars of journalistic presentation. Unfortunately, they are easily manipulated. Trust centers around trust in data and trust in motives. Extensive use of computers may serve to flare up concerns about data, especially if a limited number of stakeholders have access and the inclination to manipulate in their favour.

2.6 Collaborative Planning as a Conflict Resolution Technique

2.6.1 Multidisciplinary Problem-solving

How do we approach the design of projects for sustainable management of water resources? Engineering for sustainable water resource planning and management is a project-oriented problem. Whether the project is public or privately proposed, or whether it is merely a small cog in a master plan, planning is related by project. And each project must fit with overall goals and social values. What is it that we must sustain? There is usually more than one valid answer, and each project may have a unique set of answers. That is, a project can have a case specific set of goals and an entirely unique value system. Planning procedures and guidelines need to be flexible and adaptive to changing cases, and their unique blend of social benefits. It may be possible to compile a set of ecological indicators, compare them with net present value calculations and determine a level of sustainability. It wouldn't need to be an arbitrary measure. It could include weights to indicate relative importance of factors according to the specific case study. But then, the weights are only meaningful if the appropriate stakeholders decide, because if the weights are arbitrary the answer is arbitrary! In that case, the best approach may be to remove the stakeholders and use sensitivity analysis to choose a robust set of weights to reach a compromise!

Robust is a word that fits well with ideas of ecological diversity, environmental integrity, ecosystem health, intergenerational equity, and so on. Robust is also a word that engineers and water planners can chew on. How do you sell 'robust' to the public? It's a well known fact that there are no right answers in developing water resources. The simple reason is that you can please some of the people some of the time, but you can't give everyone everything they want. Hopefully they can all get what they need!

In order for people to get what they need, 2 things need to happen. First, a dialogue must develop between the proponent and the interested or affected parties. Second, we must develop our understanding of how different aspects of the same problem are related. This involves technological integration and innovation. Unfortunately, there are very few engineers who are trained as biologists, and there are very few biologists who have a background in sociology or geography. Until multidisciplinary expertise develops to the point of producing meaningful answers, we must rely more on abstract values and understanding of our environment. This requires all the relevant disciplines and stakeholders to speak to each other and listen to each other, which at the same time promotes interdisciplinary learning. The result may not be sustainable, but at least its not arbitrary.

A few experienced people have accumulated expertise in negotiating and mitigating issues in water planning. This wisdom is a valuable professional commodity because they have become adept at approaching circumstances appropriately, and finding solutions that are acceptable to stakeholders. These people can never be replaced, but (as good as they are) it is difficult to keep up with technology and experiences of other people. Who is commissioned to apprentice under experts so that accumulated expertise is not lost? It would be a great shame if each expert, generation after generation, accumulates interdisciplinary knowledge independent of each other. Societal advances are made possible by the implementation of technological infrastructure that grows with each discovery (unless the infrastructure itself breaks down). Continuity in knowledge accumulation for resource development is very important today. The sense of urgency to save a number of ecosystems demands that our time and experience be used wisely and collectively.

Traditionally, biologists complain that engineers and decision makers ignore environmental impacts of resource development. Likewise, engineers and politicians are known to retort that biological, environmental, or ecosystem knowledge is not in a suitable form for use in structured planning analysis.

Both cases are true to some extent. There is a wealth of biological and ecosystem knowledge and expertise. There are also many engineers and decision makers who are very aware of the importance of ecosystem interference. The missing ingredient is the relative impact (known at the planning stage) of resource development on the workings of various ecosystems. This is difficult to assess, mainly because evaluation of impacts can be case specific. Up to now, impacts are mainly measured after development. Mitigation, then, becomes more expensive and resource development is less efficient. Only now are impact statements referring to previous (similar) cases of resource development impact such as the Grande-Baleine project in northern Quebec (Hydro-Quebec, 1993). This is the starting point of integrating historical experience in project planning regarding environmental impact. In order to communicate among groups, the engineer needs to become more of an multidisciplinary problem solver (Priscoli, 1991) if engineers are to find a place within emerging paradigms for planning water projects.

Who should be proposing frameworks for decision-making integration of economic and ecosystem issues? There seems to be a useful niche for collaborative planning in a number of disciplines and industries, but what discipline is capable of delivering an acceptable social, economic, ecological, technical planning framework for improving public involvement? Any approach must consider a number of both technical and nontechnical issues. There will be many multidisciplinary demands placed on any subsequent practical tool for implementing the approach.

Engineers may be a reasonable choice of discipline for thinking about the necessary ingredients to be delivered into planning decisions. For particular proponents, the domain or scope of decision-making is limited by economic incentive such as an electric utility recognizing the hydroelectric potential, at a particular location, to meet consumer demands for power. The problem of design choice is a technical one. Realistically, it is an engineering problem. Regardless of the economic, ecological, or social impacts that are the motives for collaborative planning, the essential decisions are engineering. The challenge to the engineering profession is whether it is capable of developing a suitable forum for dealing with the outside issues. Engineers have acted as liaison for the hard sciences such as chemistry, physics, and mathematics, to the benefit of society. Over the years, this has also grown into roles in management and even policy-making. It can be argued that engineers are the natural candidate for filling multidisciplinary demands of finding a collaborative planning framework for proponents.

Engineers, by themselves, don't have all the answers. In fact, they never have! However, the definition of engineering that suggests they are flexible enough to tackle poorly-defined multidisciplinary problems. Websters dictionary defines engineering as:

> *The art and science by which natural forces and materials are utilized in structures or machines.*

In many ways, the definition for architecture is similar (and suits the needs of designing methods for finding sustainable practices or solutions):

> *The art, profession, or science of designing and constructing.*

The purpose of engineering is to devise ways of applying science for practical problems. This is not limited to physics, but should be applicable for zoology and other environmentally related disciplines.

Economics suggests an expanded use of its principles to internalize impacts. Ecology, on its own, has no conception of market demands. Geographers are also involved in defining sustainability for society. They are in many ways the interface between sociology (along with other social sciences) and environmental sciences. Architects, too, must satisfy abstract social needs in designing practical solutions. Engineering, however, is the only problem solving discipline with the technical background for meeting proponents and stakeholders from other disciplines on common ground.

2.6.2 Collaborative Planning Concept

Collaborative planning is a general planning framework for stakeholder participation in resource management. A collaborative approach to resolving conflicting views about development proposals is one attempt to internalize many diseconomies of adversarial negotiations in a project licensing process. The interest in public participation and the need to resolve disputes in an equitable and efficient manner provides the motivation for pursuing such an approach in water resource management proposals.

Kearns et al (1995) advocates collaborative forms of planning for hydro development, providing opportunities for agreement through stakeholder participation. Prendergast (1995) calls for localized assessment of priorities and needs by including additional stakeholders and reaching a consensus to defuse outrage associated with perceived risks, and to provide more equitable decision-making on a case-by-case basis.

Use of alternative dispute resolution methods in the licensing process may ensure proponent cooperation in the conceptual development of proposals. The proponent is then faced with the problem of minimizing the costs of compliance in a strategic game. It's possible that many project externalities may be resolved with quality stakeholder participation. In terms of moving toward improved overall social benefits, the transaction costs for more structured stakeholder involvement are relatively small compared to the potential benefits of social well being, equity for future options, and other desirable aspects of sustainable development. The condition for this to be true is the implementation of an efficient framework for collaborative planning of project alternatives.

Who is the collaborative process being developed for? Is it for a government regulatory agency, or maybe a public interest group? If there are good legal and moral reasons for using a collaborative planning approach, the framework is best suited in the hands of the development proponent. The reason for incorporating a collaborative planning framework in the project design process is to increase public involvement in planning decisions affecting issues external to a proponent. A proponent may be motivated to use such a framework to minimize the risk of proposal rejection or regulatory demand of expensive revisions (to minimize the cost of implementing a proposal).

Benefits of including stakeholders in early planning of potential projects:

1. To prepare more acceptable (overall) proposals.
2. To reduce the risk of licence rejection.
3. To promote understanding between stakeholders and proponent.
4. To pursue more efficient and creative mitigation of issues and satisfaction of stakeholder objectives.

For a potential stakeholder in a collaborative planning environment, there is also incentive to participate with a proponent to include their perspective in the planning process, and also to minimize the risk of project licencing without a preferred amount of input.

Without insight to the attitudes of participants, there may also be significant difficulties in allowing stakeholder involvement in early planning. They are generally:

1. Many stakeholders have difficulty with technical concepts and data outside of their discipline.
2. Available technical diversity is not conveniently packaged to foster creative exploration of options in a limited time frame.
3. Lack of understanding of the goals and perspectives of stakeholders.
4. Lack of trust between proponent and stakeholders.
5. There is no right answer.
6. Stakeholders may not be cooperative (for strategic or personal reasons).

A collaborative planning approach assumes proactive stakeholder participation in the early stages of conceptual design. That is, a specific proponent would like to propose a plan within a portion of the possible decision space for some resource, and include insights to external issues within an initial proposal. The proponent must be able to compete with other plans, including the option of leaving a resource undeveloped. A collaborative process for developing a proposal for future use of resources can be seen as iterative in nature, or cyclical in the way data gathering and analysis are interpreted, and how a proposal is adapted by decision makers in an experimentation with different alternatives (Bender and Simonovic, 1994), as depicted in Figure 6.

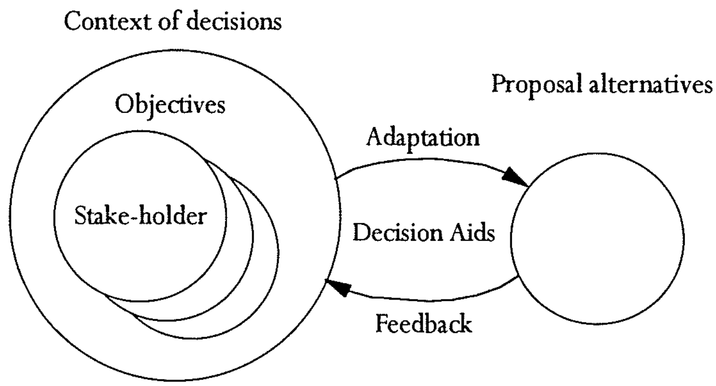

Figure 6. Conceptual collaborative planning process.

Collaborative planning places external stakeholders within the planning process as unique entities. The addition of stakeholders such as environmental regulatory agencies, social group representatives, or competing resource users, complicates decisions by expanding the viewpoints that are considered. The collaborative planning concept is an approach for addressing impacts and choices at the beginning of a proposal development process. By including various stakeholders in an exploration of the relevant decision space, common understanding and creative solutions may emerge to resolve potential disputes.

The use of relational links in this cyclical process is crucial to collaborative planning in terms of formalizing the present state of experience and available knowledge in the project domain. Relational linking is the mechanism by which communication can occur between a system of resource attributes, and a set of stakeholder goals. A system of goals and preferences can be completely abstract from the physical system that describes an ecosystem. These goals and preferences define the context of a decision. A construct to relate the decision context and the physical system can also be developed to reflect available knowledge and experience. Links provide an avenue for describing relationships, indicating consequences, and stating options.

A physical resource system may be altered by development and intervention from technical options. An understanding of how resource attributes interact with other objects, systems, or processes is the fundamental motivation for pursuing models of physical resource systems. The changes in distribution, quality, and use of water over time is an example of the types of interaction that can be described. Other interesting applications are the implications of changes in the water resource system on biosystems that rely on water for existence.

In the human context of decision-making, there are objectives or goals, and motives that affect the choices that are made. A context system manages the interactions and interdependencies of each context object in a decision maker's goal structure. The context system contains priorities, motives, objectives, and their relationships, ranging from practical consequences to strategic motivation.

Each stakeholder or decision maker will ultimately describe a completely unique structure and language for the context of a decision. Just as different ecosystems operate within a unique system of relationships compared to other ecosystems, each decision maker operates from a unique set of issues. Negotiation to determine a single goal structure from which to make a decision reduces the amount of information available to make that decision. The single goal structure makes the decision very straightforward, but the process for preparing the single goal structure may be extremely difficult or inequitable.

Interaction between contextual and physical systems allows structured involvement of multiple stakeholders to supplement current understanding of potential local or regional development policy objectives, and criteria for meeting objectives by changing basic system properties, policy statements, and parameter options for analysis of tradeoffs. The resulting planning framework is focused on fostering creative choices in preparing a development proposal, based on common understanding, collaboration, and multidisciplinary technologies.

The combination of context and physical systems is a challenge to integrate multiple disciplines and perspectives in a decision framework. In a cooperative environment, with access to relevant expertise, collaborative planning processes may flourish. The remaining challenge will then be to properly organize, access, and present information in a timely and appropriate manner. The most appropriate manner will likely be specific to the decision context and the characteristics of the relevant physical systems. A framework and relevant tools will be explored in the following chapters as an attempt to address some of the issues in applying collaborative planning processes with the help of decision support systems.

Chapter 3 Integrated Decision Support Tools

This chapter builds on the idea of implementing a collaborative planning process using decision support tools and techniques. Given the motivation of achieving consensus sustainability through collaborative planning, some decision support tools may be more applicable than others, and the form of decision support tool application may impact the effectiveness of integrated decision support.

In order to develop and implement a collaborative planning decision support system, an assumption is made that decisions require support which spans multiple disciplines, and encourages the perspectives of different stakeholders. This integrated approach, as an expansion of our economic interpretation and as encouraged by ecologists, defines a very particular form of decision support. Systems approaches for analyzing processes and decisions, spatial and temporal interpretation of data and analysis, and flexible interaction with available knowledge are all part of a holistic approach to facilitating stakeholder participation in the decision process.

The chapters that follow (chapter 4,5,6,7) describe a prototype application of a decision support system (DSS) for collaborative planning. In particular, chapter 4 introduces the conceptual decision support system. As an introduction to those chapters, some of the basic tools used to develop the DSS are explored here in chapter 3.

3.1 Systems Analysis

The following section is not meant as an exhaustive review of systems analysis techniques and philosophy. Discussion is skewed toward those applications relevant to integrated decision support for multiple decision makers in areas of water resources planning and management.

3.1.1 System Optimization

The use of optimization methods is well known in water resources. Management of water reserves for municipal, agricultural, or environmental supply is accomplished using various optimization techniques. An optimization problem has 2 parts: an objective statement; and constraining factors. The solution is an efficient allocation of resources in terms of an objective variable. Other decision variables are manipulated to create the best combination of the objective variable.

In decision-making, optimization is used to make decision recommendations, or to help decision makers understand implications of choices. An early example of linear programming in resource allocation planning is Liang (1976), where implications of group collaboration in planning are assessed for using traditional optimization to find "the answer". Noninteractive optimization denies planners the opportunity of observing impacts from alternative plans, robs them of a feeling of participation, and forces planners to specify all standards and other constraints in advance (Liang, 1976).

Liebman (1976) follows up by declaring public problems to be "wicked", and that optimization answers are useful only if the problem is completely understood.

> *In wicked problems, ... the set of alternatives is too large, too diverse, and too little agreed upon. (Liebman, 1976).*

Liebman's exasperation comes from the realization that there is no single right answer to questions of resource allocation when there are legitimate external issues. Conflicts over planning decisions resulted from different opinions about the objective(s). The conflict resolution problem becomes a question of what set of criteria defines the "best" alternatives choices?

The planning problem quickly expands from an optimization problem to a multiple objective problem. Broad-based questions of equity were explored in water management problems as early as 1976 for water quality management alternatives (Brill et al, 1976), in which tradeoffs were examined between economic efficiency and equity definitions. There is no shortage of objectives for planning problems. One certainty in planning is that the analytical solution will always be the result of partial analysis. The impact of even a single unaccounted objective in a multiobjective assessment will likely project a best compromise solution on the inferior region of the original problem (Brill, 1979). Even if all objectives of stakeholders are accounted for, there usually exists too many local optimum solutions along the noninferior frontier to acceptably choose a single option. Using rigorous systems analysis techniques to provide solutions may trigger additional sources of conflict.

Brill (1979) offers an alternative use for optimization methods in making planning decisions, especially for resource allocation. Brill proposed that they be used as tools within a planning process. The purpose of the optimization and analysis techniques is reduced from the role of making recommendations, to facilitator of creativity and inventiveness. A resulting tool box of models would use system simulation and optimization models to generate alternatives and facilitate evaluations by decision makers in an interactive process.

3.1.2 Multiobjective Analysis

Management of complex decision problems rarely involves a single objective. Multiple objective decisions do not have an optimal solution, unless one solution completely dominates every other solution for every objective. This does not usually happen in water resource management. As a result, there has been great efforts made to develop techniques for assessing tradeoffs between alternatives based on using more than one objective. In 3 decades of multiobjective research, efforts have been made in 3 levels of making a multiobjective decision:

- objective quantification.
- generation of alternatives.
- plan selection.

Early work focused on alternative generation, providing decision makers with a complete spectrum of nondominated solutions. Contemporary research into multiobjective analysis has shifted away from continuous theoretical models, and have explored issues in evaluating discrete alternatives for plan selection, including techniques related to multicriteria decision analysis (MCDA), and multiattribute utility analysis (MAUT).

Plan formulation concentrates on 2 general approaches for generating nondominated alternatives. One is a weighting approach. Each objective value is assessed, and combined using weights to offset noncommensurate units and to express relative importance of each objective. By varying the weights, a set of nondominated solutions emerges by solving each new optimization problem. Another technique is the constraint method. It assesses, or attempts to optimize each objective individually while restricting other objectives to maintain minimum standards. The constraint method is more flexible than the weighting method because weighting demands that the efficiency (nondominated) frontier be strictly convex relative to a decision maker's indifference curve.

In terms of plan selection, the group of techniques known as multicriteria decision-making (MCDM) methods deal with selecting a discrete alternative from a list of options. The techniques developed for these types of problems are based on one of the following philosophies:

- outranking.
- distance.
- utility.

Outranking techniques such as ELECTRE (Benayoun et al, 1966) methods use indicators like concordance and discordance to make judgements in a search for a highly-rated alternative for most criteria yet are not completely unacceptable for any criteria. Distance-based methods use a notion of geometric best to determine the "closest" option to an ideal point. Multiattribute utility (MAUT) methods rely on values of relative objective satisfaction, where the alternative with the highest-rated utility is preferred. All methods are reasonably robust in evaluating sensitivity to changing parameters. However, in terms of decision maker input, outranking and distance methods require only weights and scales. Utility methods require more in-depth input.

Selection of a discrete alternative can be generalized for different decision types. Some decisions involve a single decision maker and multiple criteria in an effort to choose from a list of alternatives, known as single participant multiple criteria problems. Other decisions may involve multiple participants yet only a single criteria to judge a list of options. This can be referred to as multiple participant single criteria problems. Multiple participant single criteria problems are not very common, and can easily be converted to a single participant multiple criteria problem. A multiple participant multiple criteria problem (Hipel et al, 1993) is the parent problem class from which single participant multiple criteria and multiple participant single criteria problems are subproblems.

There are plenty of options when it comes to choosing a multiobjective method. Cohon and Marks (1975) provided an early comparison of models and suggested the surrogate worth tradeoff method for water resources problems because of its interactive nature (Haimes and Hall, 1974). Hobbs et al (1992) compare multicriteria methods for their appropriateness, ease of use, and validity for water planning decisions. Hobbs suggests that simpler transparent methods, or no formal method at all, are preferred by experienced planners. In comparing goal programming, ELECTRE, the analytic hierarchy process (AHP), and both additive and multiplicative utility functions - utility functions appeared to be a suitable choice for water problems based on ease of use, and data limitations for assessment of risk in terms of value judgements. Tecle (1992) compares 15 of the best known methods. Goicoechea et al (1992) provides an experimental evaluation of 4 methods for water resource planning problems. Harboe (1992) compares several multiobjective methods for reservoir operation, based on the form of weighting (prior, post, interactive). Multiobjective techniques have been extensively explored in water resource planning (Keeney and Wood, 1977; Loucks et al, 1981; Gershon and Duckstein, 1983; Kindler, 1988; Simonovic, 1989; Hipel, 1992; Ko et al, 1992; Thiessen and Loucks, 1992; and many others).

Zeleny (1992) observes circumstances in which multicriteria methods may have reduced visibility and usefulness. These are:

- Time pressure reduces the number of criteria to be considered.
- As the problem definition becomes more complete and precise, fewer criteria are needed.
- Autonomous decision makers are bound to use more criteria than those being controlled by a strict hierarchical decision system.
- Isolation from the perturbations of changing environment reduces the need for multiple criteria.
- The more complete, comprehensive and integrated knowledge of the problem the more criteria will be used - but partial, limited and non-integrated knowledge will significantly reduce the number of criteria.
- Cultures and organizations focused on central planning and collective decision making rely on aggregation and the reduction of criteria in order to reach consensus.

Most of the basic multicriteria techniques, and reviews of those methods, do not incorporate aspects of multiple decision makers, or conflict resolution. One application, however, is Fang et al (1993) which applies a graph theory conflict resolution technique. Other researchers have made modifications to basic multicriteria methods to allow for group decision-making (Neely et al, 1976; Lotfi et al, 1991; Bogardi and

Duckstein, 1992; Hamaleinen et al, 1992). Sainfort et al (1990) found that a cooperative group environment can effectively use multicriteria methods to reach consensus. Nagel (1990) presents a similar multicriteria tool for negotiation. Liu (1984) provides a review of the basic MCDA techniques available to group decision making. The shortcoming of most multicriteria methods is that they rely on a prior articulation of preferences. The difficulty for group decision-making is that conflicts arise, and complicate the evaluation process by tying decision makers to their articulation of preference. Prior articulation methods are typified by an effort to aggregate criteria of decision makers and reduce the problem to a single participant multiple criteria problem.

Exceptions to prior articulation are methods that employ progressive articulation of preferences. These are the true interactive conflict-capable multiobjective methods. One example is the step method (STEM). When progressive articulation methods are included within a comparison of techniques, they are not usually rated highly because of the amount of information and time that is required by decision makers. They are based on an algorithmic approach such as:

1 Find a noninferior (nondominated) solution,
2 Modify the solution according to reactions of the decision makers, and
3 Repeat until satisfaction or termination.

Computational constraints can no longer be used to discard techniques. Many of these boundaries simply do not exist anymore. Focus is now on how to effectively approach consensus-seeking between multiple stakeholders with multiple objectives (Hamaleinen et al, 1992). Uber et al (1992) follow up on the tool kit approach discussed by Brill (1979) to restrict the role of mathematical solutions in complex decision problems. Instead of gathering all available information for procuring a recommendation, multicriteria methods and other techniques are used to generate additional insight into decision consequences, as well as tradeoffs. This may lead to an increase in decision information, or simply a transition from one form of output to more efficient feedback.

3.2 Decision Support Systems

Decision support incorporates system management decision-making aids, expert systems, statistical tools, specialized models, and data management. When some or all of these elements are applied within a computer application for the purpose of assisting a decision maker, the result is referred to as a decision support system (DSS). A convenient collection of early research in DSS applications can be found in Loucks and da Costa (1989). Most DSSs focus on a small number of technologies or tools to assist in decision-making. The common element of DSS applications is the gravitation toward graphical representation of output for decision-makers. Normally, the analytical tools that comprise the functionality of a DSS are screened from view by more simple interpretation and summary of data. Graphical interfaces are seen as a key ingredient for DSSs to be able to respond to decision maker needs (Dougherty, 1994).

DSSs that are developed to assist in decisions related to a specific problem type typically combine most of the potential components of a DSS. They include tools for a specific form of problem, but also provide support for data intensive tasks specific to the problem type. Systems analysis tools are commonly used to provide management decision support. Input is provided by statistical tools, specialized models, and database management systems. One example is presented by Simonovic and Savic (1989) for reservoir management. Graphical interfaces mask expert systems that evaluate the decision context and choose an analytical tool from a tool kit of potential management models. Each model has data requirements supplied by different databases. Options and advice from the expert system supplement model output for comprehensive decision support based on experience and current technology.

Another example is an application for environmental management (Janssen, 1992). It provides access to several multiobjective decision aids for formulating criteria and weights to assess tradeoffs between alternative choices and present results in a graphical way. The purpose of the software, DEFINITE, is to form the management core of a comprehensive DSS to support environmental problem solving. This involves interaction with databases, simulation models, various analysis methods, as well as management optimization procedures. The functionality of the DSS is built around the selected forms of graphical output, and its ability to interact with other software.

3.2.1 Decision Support System Characteristics

Important characteristics of DSSs for sustainable management of water resources include accessibility, flexibility, facilitation, learning, interaction and easy of use (Simonovic, 1995). The following discussion of DSS characteristics will address specific requirements of sustainable water resources planning and management.

(A) *Problem Identification*

Sustainable water resource management contains a number of semi-structured and non-structured problems. The management problem which can be well formulated in an algorithmic way (a computer program), is called well-structured. Decisions in this case are straightforward because alternative solutions are known. If the management problem involves lack of data or knowledge, non-quantifiable variables, and a very complex description, then it is called semi- or non-structured. Structuring of the problem, in this case, must be done by the human in the man-machine system.

Because judgment and intuition are critical in examining and resolving many water resource problems, an effective DSS involves problem identification. This process includes searching the decision making domain for future problems that need to be anticipated and solved. Future opportunities can be identified and implemented to address the long-term consequences of current decisions, defined as the second component of the sustainable water resources management context.

(B) *Problem Formulation (learning)*

Before trying to implement principles of sustainable development, a water resource DSS may be used in situations in which there is a clear problem definition. However, the concept of a 'problem' as it relates to sustainable development may be expanded to include two perspectives (Landry et al, 1985):

- Problem as objective reality.
- Problem as mental construct.

In the first case, a problem is viewed as unsatisfactory objective reality discovered by observations and facts. The stakeholder or expert has to define the problem. As a problem exists objectively, all participants in the decision process see it in the same way (even if there are different alternative solutions). Here, problem formulation is a preliminary step to DSS design. The second case presents an alternative view, considering a problem to be a subjective presentation conceived by a participant confronted with reality perceived as unsatisfactory. Here, common threshold values have to be defined by the different participants in the decision making process before another procedure can take place. This approach requires integration of the problem formulation process into the context of a DSS. The emphasis is shifted from the analysis phase. It is important to note that problem formulation in sustainable development is more a social process than a technical one.

(C) *Adaptability*

A DSS environment allows a number of 'what if' questions to be asked and answered. A benefit of using DSSs is that a number of decisions can be tried without having to deal with the consequences. In this way, DSSs can guide stakeholders through the most optimistic, the most pessimistic, and in-between scenarios.

The ability to ask 'what if' questions to establish areas of uncertainty, and to recognize the sensitivity of results to varying assumptions, stimulates creative and analytical processes of collaborative decision making. The process provides a common ground for communication between stakeholders. Since the stakeholders can use the tool directly, higher quality decisions can be made on a more timely basis.

(D) *Facilitation*

The integration and administration of mathematical models within a general framework could be identified as the primary purpose of DSSs. For problem identification and problem solving, decision makers deal with analysis. This fact underlines the need for DSS modeling capabilities in:

- Retrieval of data.
- Execution of *ad hoc* analysis.
- Evaluation of consequences of proposed actions.
- Proposal of decisions.

Typical models that include database management system functions as data queries and data manipulation, range from simple arithmetic functions and statistical operations to the ability to call up optimization and simulation models. The scope of a DSS is in the integration of such different facilities. The philosophical idea of using a DSS is to integrate different fields of science, and put weight on social circumstances which may decide or influence problem definitions and solution approaches.

Decision support systems for water resources planning and management have access to a plethora of modelling and analysis tools, a well developed set of evaluation tools, applied artificial intelligence techniques, and emerging visual aid capabilities. Geographic Information Systems (GIS) and tool kits of analysis modules provide the basis for a powerful aid to stakeholders. The design of user interfaces can enable multidisciplinary stakeholders to contribute insight and evaluate circumstances without expertise in all of the related specialized fields. If a stakeholder asks about the effect of changing the full supply level of a proposed reservoir on the amount of flooded stream areas or erodibility of the reservoir shoreline - for the purpose of assessing potential fisheries needs - GIS tools can regenerate flooding sequences, change dam or dyke configurations, and calculate relevant parameters. Parameters can then be input to modules to recalculate power capacity or reliability of power generation, and compare potential life cycle impacts to fisheries resources, etc. In a modular form of storing knowledge about impacts and technical alternatives, project-related modules can be purchased or developed to supply advice or suggestions. One example is a module to supply knowledge of potential fish migration disruptions and relevant technical options for either mitigating or circumventing the problem. Another module may supply design alternatives for hydroelectric generating stations, depending on site conditions.

(E) *Interaction*

The ability of a DSS to interact with its users is also an important issue. A DSS must answer 'what if' questions and provide potential solutions. The user-machine interface provides answers that stakeholders can understand, when such information is needed, under their direct control. Therefore, DSSs are intended to help stakeholders throughout the process of identifying and solving their problems. The merging of output with the subjective judgment of participants in water resource decision making processes provide a better foundation for making effective decisions.

One must acknowledge that the form of interface has a great impact to the user's ability to assimilate information from the DSS, or even to understand the available options in using the DSS. The concern is especially prevalent within a multiple stakeholder environment with multidisciplinary expertise. Different scientific disciplines are known to make use of different forms of presentation for data and knowledge. Satisfying the quircks of all potential users is a challenge which must also be addressed.

3.2.2 Expert Systems

Expert systems are a branch of the artificial intelligence field that specializes in the mundane task of encoding experience and processes for making decisions. Knowledge is encoded in Boolean logic and accessed by searching mechanisms called inference engines. Five phases in expert system design are: identification, conceptualization, formalization, implementation, and testing. Describing expert systems this way tends to cloud the essense of expert system application. Most computer programs can handle the IF-THEN-ELSE architecture that expert systems use to encode knowledge. The unique advantage is derived by the inferencing capabilities of expert systems. Two types are used: backward and forward chaining. Backward chaining searches for information if it is required while forward chaining is directed to the relevant information. In general, backward chaining uses IF statements as search mechanisms, and forward chaining acts on THEN statements. The unique power that backward chaining brings to expert systems is the modularity in knowledge dissemination. Each rule in a knowledge base may be given a very specific scope and aspect of a knowledge domain, and does not need to address its place in the broader problem scope. Consistency in language is necessary for the expert system to function.

The use of expert systems in describing operating policies for reservoirs and other water management problems is an approach that easily adapts to system simulation and experimentation of decision rules. Simonovic (1991) outlines general areas applicable to expert system technologies. One example is the use of interest satisfaction relationships, defined within an expert system, to describe regulatory decision-making on Lake Ontario (Eberhardt, 1994). An expert system application for a water resource design problem for fish passage can be found in Bender et al (1992). Like many design problems, rules of thumb are popular for facilitating choices. Fish passage is no exception. Bender et al (1992) encodes rules of thumb within the Boolean architecture, and integrates the knowledge, in the typical expert system manner, with both backward and forward inferencing mechanisms. Other examples of expert systems in water management problems can be found in Simonovic and Savic (1989), and Simonovic (1992). Applications for environmental screening of alternatives have also used expert systems. An example is Fedra et al (1991).

3.2.3 Object-oriented Modelling and Design

Object-oriented design is current state-of-the-art for computer modelling. The main benefit for using object-oriented techniques in system management is that the behaviour of each object is self-contained, and completely modular. Each object is able to perform specific tasks, and knows how to find and interact with other certain types of objects. In this way a system of objects or processes can be built up, modified and managed with much less effort than more traditional programming structures in civil engineering such as with FORTRAN.

Object-oriented design components are made up of 2 types of elements: objects and links. Every object is linked with other objects forming a system of discrete entities, transferring information through the links and processing information by the existing behaviour of the objects.

Objects are normally organized in a hierarchical fashion. A group of objects that behave in the same way, or have some common features, may belong to a parent object called a class. In turn, each class can belong to still other classes. In the same way, individual objects may be composed of children objects that each describe different aspects of the parent object.

The links that connect parent and child objects are one form of bond between objects. Other bonds describe how information relates to different objects. Bonds may be directional, they may be constrained, they may possess specific attributes, and they may define multiple associations.

An object-oriented approach is generally defined by 4 properties: *identity*, *classification*, *polymorphism*, and *inheritance* (Rumbaugh et al, 1991). Objects, by being discrete, have identity. Possession of groups of objects by a parent object is classification. Polymorphism is the ability for an action to adapt to different groups or classes of objects. That is, 2 objects may behave completely differently if asked the same question. Finally, inheritance refers to the ability of objects to share attributes and behaviour based on hierarchical parent-child relationships.

Some software development tools for expert systems incorporate some qualities of object-oriented design. Nexpert Object (Neuron Data, 1993) is one example. Attributes, values, and behaviour can be inherited from classes to objects. Objects can belong to other objects. However, object modelling is limited by the fact that Nexpert cannot define links with the same flexibility. Linking objects with Nexpert Object is limited to possession - where one class or object is a parent of another object or class.

Water resource applications have used object architectures mostly for management of reservoirs. STELLA and EXTEND are 2 popular examples that allow graphical objects to be created and linked. Keyes and Palmer (1992) used STELLA to demonstrate drought planning policy scenarios for a multiple reservoir system.

The scope for resource allocation or development decisions can be described as managing the flux of certain decision variables. The domain of each decision variable, such as surface water, can be described in terms of a system of discrete components, states, and processes. Within the scope of a decision, there may be several interdependent systems that are of concern. It is for these systems that available analysis or modelling capabilities can be applied. Physical system management includes evaluation of a static system, and adaptive description for changes in the system due to development.

One application example for physical system management is an object-oriented approach to managing hydraulic network information. The components and structures that compose the present and possible future physical system (of lakes, streams, and structures) must be defined and organized in such a way that options may be added, moved, deleted, or adjusted by the users. An accepted method of accomplishing this is the organization of potential system properties into general classes with which new components or structures called objects can be associated. Object properties must be defined carefully because the system may include natural, structural, economic, or even social properties. The list of possible alternatives within the physical network are dependent on the scope of the problem and the scope of the potential decision space which will be explored. One of the more difficult dimensions of scope that must be included is the temporal variability of any physical system. Of course the broader the scope, the more complicated the definition and management of system components will be.

The purpose of interactive system development is to be able to ask questions of the system such as "What would happen to power reliability and fish habitat suitability if I reduce reservoir flooding and add an upstream weir or control structure?" Normally this is a difficult question to answer. However, the formulation of reliability problems is available, system simulation and forecasts can be made available, operating rules can be adjusted, etc. Facilitating collaborative planning requires the inclusion of a number of technical tools, and flexibility for adding new tools to analysis of the physical system. If the properties of the physical system are properly defined, a modelling and analysis tool kit (Brill, 1979) for simulating flow sequences, geomorphologic changes, and environmental quality parameters becomes possible. It is also important that geographic data form an integral part of the physical network, and that GIS tools supplement the tool kit available for system analysis.

By providing an integrated facility, the interactive environment moves from an abstract, representative, understanding to the more realistic visualisation of system behaviour that allows stake-holders to focus on their particular concerns. In collaborative planning, ease of use provided by these facilities will give users the power to investigate options. In today's computing environments it is possible to integrate this power. Software development is moving quickly in the direction of modular programming, and object-oriented structures that supply the processing and graphical power required for complex, yet flexible, modelling.

Decision makers must take advantage of this new resource. Development of such software allows users to verify statements, and explore more possibilities than otherwise possible in a limited time frame.

An object-oriented model for network management assumes that individual objects are connected or linked according to the flow of a resource such as water or money. Money flows by transaction, and water flows by gravity. Both are resources that can be managed for the benefit of other resources such as people, wildlife, etc. The condition of the resource chosen to be managed, or the state of the resource system, is described in terms of discrete objects such as a bank account or a restaurant which stores or enables the flow of money. A model describing the interactions and values of the discrete objects can then be used to feed input information to any number of technical or management tools for system analysis.

In the object-oriented model of hydraulic system characteristics, the forms of surface water are grouped into classes such as stream or lake. A *class* is defined as the governing bahaviour for a group of objects, where an *object* is an individual such as Lake Winnipeg. The class, lake, would comprise the set of all the lakes in the hydraulic system including Lake Winnipeg. Included in the definition of each class is a list of the properties needed to define the characteristics of an object in that class. Stream class, for example, has the properties slope and drainage area. Any objects that belong to the stream class automatically inherit the properties of the stream class, but not necessarily the actual value of the property. Obviously, each stream will differ in channel slope. Figure 7 is a simple system of objects (triangles) that are linked to form a small hydraulic system.

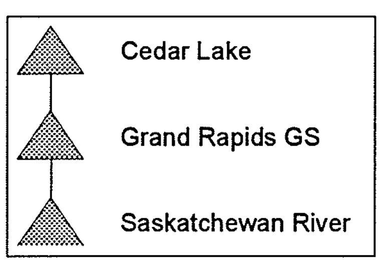

Figure 7. Simple water management system.

Specific objects such as Grand Rapids generating station, which belongs to the dam class, and is hydraulically linked to Cedar Lake and Saskatchewan River, can be complex systems themselves. Grand Rapids generating station can be subdivided into a number of sub-objects or *components*. There are a number of options for a dam. If it is to be developed as a generating station, then a power house object must be added. Inclusion of a power house demands other components such as intake, tailrace, and possibly a penstock or similar watercourse supply. Figure 8 is a logical schematic of possible options for a dam.

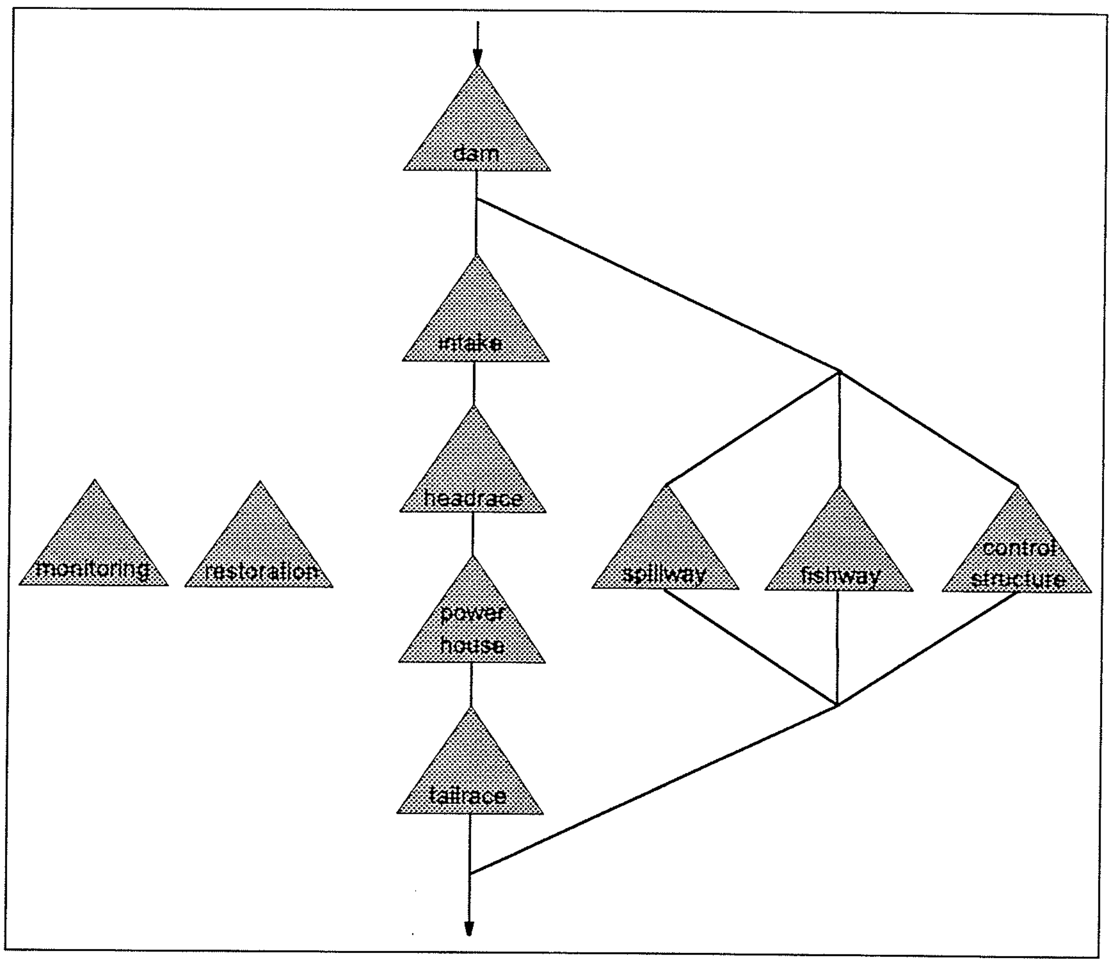

Figure 8. Dam object alternatives and flow of water.

Each component, then, inherits unique properties from a governing class in the same way that Grand Rapids generating station inherited properties from the dam class. The spillway inherits properties from a spillway class. If necessary, the spillway can be defined with a number of optional components. A hierarchy is established, consisting of objects, components, the components of components and so on. Each of which is treated as an object, and inherits properties from a governing class. The hierarchy forms a number of *logical links*, not to be confused with *hydraulic links*. A list of hydraulic links may include a number of logical links because at a dam, water flows from the intake through the power house, and out the tailrace to another hydraulic object. However, logical links do not include hydraulic links. A dam is structurally independent from a stream.

Classes of objects, also belong to general categories of classes. Lake class, for instance, belongs to the storage category which is a class itself. Lake Winnipeg, then, is an object that belongs to lake class, which in turn belongs to storage class. Storage objects tend to have properties such as volume. Volume is one property that the lake class inherits from the storage class. Volume can then be inherited to Lake Winnipeg. Lake Winnipeg is defined as a lake instead of a generic storage bacause there are distinct types of storage vehicles that contain additional properties. An estuary is a form of storage that demands properties concerned about the diurnal effects of tide.

Objects that are created define a hydraulic system with inherited properties and behaviour. There are 2 types of properties: static and dynamic. A classification property for an object is a *static property*. The stage of the lake, however, is a *dynamic property* that changes over time. Static properties describe the general characteristics of objects, while dynamic properties track values related to the changing water distribution in the system.

Once a physical system has been prepared, it should be able to operate independently from other systems. There may be external conditions that trigger actions and changes within the system, but it should be essentially self-contained. All types of resource flux should be described within the object model. The only way it is restricted is through external inputs of the same resource, unless the object model is global. Ideally, several physical systems can be used to simulate and manipulate different resources. Water is one resource, money is another resource that manages flux from one place to another or from one form to another in terms of both storage and flow.

3.2.4 Geographic Information Systems

Spatial information system (SIS) is a broad term that encompasses a plethora of tools and techniques to examine and analyze temporally referenced spatial data sets. SIS focus on the use of nondestructive testing of spatial attributes, such as aerial photography and satellite imaging of the electromagnetic spectrum, but also includes discrete sampling from traditional networks of measurement sites. General spatial tools such as Geographical Information Systems (GIS) are commonly used to manipulate spatial data and analyze.

Geographic Information Systems (GIS) are tools for the storage and analysis of spatial and temporally variable data. GIS provides an architecture for efficiently storing large amounts of data which may be of 2 dimensions or more. Associated with GIS software are analytical tools for manipulating and comparing spatial data. Burrough (1993) is a good reference for GIS.

To make accurate use of geographic data, several concepts must be understood about the earth and our representations of it. The earth's surface has curvature which causes distortions to flat map projections. In addition, the earth is not a perfect circle. Generally, the earth is a spheroid or ellipsoid due to the rotation of the earth - causing a slight bulge at the equator and flattening at the poles.

There are over 250 approaches for addressing the difficulties of flattening the earth into a manageable projection. They attempt to address 2 issues. One is *conformity* (orthomorphicity), which expresses the projection's ability to preserve spheroidal shapes. The other is *equivalency* (homolographicity) which describes the projection's preservation of area size. The relative tradeoff between these 2 objectives has produced the vast array of projection approaches.

Datums from which map values are calculated are numerous. The US, Canada, and Mexico have adopted the *Geodetic Reference System* (GRS) 1980. The GRS80 ellipsoid is commonly used with the *Universal Transverse Mercator* (UTM) grid system for Canadian digital mapping services.

The *National Topographic Service* (NTS) provides digital maps across Canada. Provincial agencies also provide more detailed surveys. Maps are available in the UTM coordinate system, using the GRS80 ellipsoid, at 1:50000 scale or smaller (eg. 1:500000, 1:2000000) covering the entire country. More populated regions have larger scale maps (1:20000). They are available in digital form, depicting features such as lakes, rivers, wetlands, roads, buildings, boundaries, elevation contours, etc.

The storage of digital data within a GIS can be in either of 2 forms: *vector* and *raster*. A raster system stores data in grid of pixels or cells. Each cell shares its border with surrounding cells, and values can be given to each cell. Vector data representation of spatial data uses points to define locations which in turn are used to define lines. Multiple lines can then be used to define an area or polygon. All of which can be described by a small number of points.

Vector data is most efficient for storage or transfer of data. For example, the province of Saskatchewan can be described by 4 points which connect lines that define a single polygon area. Accuracy of a vector map at different scales is not an issue. However, analysis of vector data is conceptually difficult and relatively limited.

Raster data can become enormously cumbersome for fine-scaled cell sizes in a large map area. For example, if a 30km by 30km map area is represented using a 30m by 30m cell size, the number of cells needed to cover the entire region is 1,000,000 (without using various data compression techniques). A reduction in cell size, or the addition of other map layers will increase the storage requirements of a raster database dramatically. The primary benefit of using raster images is to take advantage of the number of analysis tools available for raster data representation. Most forms of analysis easily adapt to raster images.

The types of digital maps which use either of the given data types, include various topographic features such as vegetation or soil types, land use types, digital elevation models, other areal forms of spatial data, and site (or point) data. Area data has values or category identifiers for each cell. Site data may also have values or categories, but are also normally linked to a relational database which stores properties and other complex relationships.

Analysis of GIS data includes the following types of tools:

1 cell-specific ... reclassification and algebraic analysis
2 layer-specific ... region analysis (areas), neighbourhood analysis (distances)

Reclassification and algebraic calculations act on each cell of a raster image, or compares the same cell of different map layers. Region analysis compares adjacent cells, for the identification of spatial properties such as area, length, or shape. Neighbourhood analysis is concerned with more complex comparisons of adjacent cells to calculate distances. Calculation of buffer areas, travel costs, and spatial interpolation techniques such as *kriging* are examples of neighbourhood analysis.

The application of GIS has benefitted analysis and decision making for several types of problems. These include site selection or classification, network problems, natural resource management, and more recently in the areas of multiobjective decision analysis and conflict resolution. Site selection includes landfill placement and other land acquisition problems such as purchasing land for airports. Network analysis includes the location of network elements and the routing of flows through the network such as locating new roads or power lines. Natural resource management is a form of GIS application with widespread use from forest and wildlife habitat management to water quality management and environmental impact assessment. More conceptually - advanced uses of GIS include the evaluation of tradeoffs for multiobjective spatial problems, or analysis of risks and decision uncertainties. Application of these uses with relevant spatial analysis tools have also begun to play a role in conflict resolution problems with multiple stakeholders. The visual aspects of feedback from GIS, and the relative ease of preparing initial feedback from analysis, makes GIS a powerful tool for communicating ideas to diverse groups of people.

Geographic information systems have been used mostly in land management applications such as forestry planning, or agricultural land assessment. They have also found application in natural resource management, particularly in classifying wildlife habitat. Water resource management applications are limited. Digital elevation models in a GIS environment have been used to delineate drainage basins, and calculate potential reservoir storage. For the most part, though, water resources relies on discrete ground measurement sites, and sequential models to describe the transfer of water. Data is collected on the ground for use by case specific system management models, not for validation of spatial inferencing by remote sensing.

The use of GIS in decision support of water resources planning and management problems can be very beneficial when properly integrated with other tools such as relational database management systems (RDBMS), expert systems (ES), and existing modelling or analysis tool kits (Bender, 1995). Relational databases provide access to data for the generation of damage curves from flooding, for example. Expert systems provide a means of applying decision rules such as operating policies for reservoirs, or to provide advice based on existing technologies or experience. Other modelling or analysis tool kits may be necessary to calculate certain system-wide parameters, or to evaluate the flow of resources. Walsh (1993) promotes the synthesis of GIS in decision support systems for water resource management problems, and identifies interdisciplinary collaboration as a challenge to overcome.

In the area of hydrologic modelling, GIS has been applied to (DeVantier and Feldman, 1993):

1 Rainfall-runoff modelling
2 Floodplain management and flood forecasting
3 Erosion prediction and control
4 Water quality prediction and control
5 Drainage utility implementation

A simple water resource application is to visualize the consequences of developing a river reach with hydroelectric power generation. If a dam or some other obstacle is placed in a watershed, what will be the flooding consequences of that action? What volume of storage will be gained? What area of land will be flooded? Will buildings or agricultural areas be displaced? Will spawning areas be flooded? Will migrating species be disrupted? There is specialized software for the generation of backwater effects - which is the desired output for the flooding level upstream of the dam. However, for simple visualization of flooding effects early in the planning process, GIS tools can be very useful for showing inundated areas and experimenting with flooding scenarios - without calculating backwater effects.

GIS is fast becoming the central tool in many decision support systems (DSS). Its analytical power is very broad. In most water resource management projects, however, the data, analysis, and knowledge required to produce results and aid in the decision making process is more complex than merely spatial. Temporal aspects can be addressed in many GIS packages, but the systems approach to managing stocks and flows of resources is completely foreign to GIS which does not normally discretize conceptual systems as much as other analytical tools.

3.2.5 Graphical User Interfaces

A graphical user interface (GUI) is a method of interaction from which a computer user can perform tasks with a program. It is, in psychological terms, an artifact that we use to simplify the completion of task (Carroll, 1991). A GUI can take on many forms in terms of the set of concepts that are used to trigger responses. It is a command line language with pictures instead of prompts, although it may include prompts. It is a step-by-step approach that allows cheating! Computer graphics, and user-interaction is limited only by the imagination of the developers. Decision-making and learning about decision options can be supplemented by using GUI tools. One example is presented by Marlatt et al (1993) where the US Army has experimented with video simulation as part of an environmental decision-making process.

The use of graphical displays is very accessible, and expected from software users. Software development tools normally include a set of graphical display tools. The programmer then supplies the functionality, using a programming or scripting language. The set of graphical display tools differs depending on the software development tool, the graphical environment, and the operating system. Generally, tools are composed of windows, buttons, text edits, or hybrid combinations. Graphing facilities, and drawing facilities may also be available, along with specialized input devices modelled after familiar medium (such as dials, thermometers, etc.). One example of a development tool is Neuron Data's Open Interface (Neuron Data, 1993). It possesses a set of display tools called widgets. Functionality is provided by a scripting language based on C, that is attached to each widget, and automatically added to C code generated by the development tool. The appearance and behaviour of each widget, including how it interacts with other widgets is generated when the developer "draws" the widget on the screen.

The purpose of using graphical tools is to provide an intuitive approach for communicating with other software. Open Interface can be extended to include communication libraries for an expert system development tool, Nexpert Object, and links to a relational database such as Oracle. Using a mouse to click on a button in the GUI may trigger rules to be assessed in an expert system, or data to be transferred and displayed. The GUI developer can also build additional functionality directly into the C code generated by Open Interface. Other benefits include the ability to access external software such as operating system scripts (or batch files), executable programs, and independent software packages.

The use of GUIs is gaining attention in water resource management for reasons of decision complexity. Issues can be data intensive, involving diverse computational features, controlled by complex decision rules, yet relying on the decision makers understanding of conceptual issues. The ability of a GUI to aid in managing data and knowledge without clouding the conceptual issues is its most rewarding attribute.

Chapter 4 Conceptual Design of a Collaborative Planning Support System

In recent years there has been increasing emphasis placed on helping decision makers make well informed decisions and involving all stakeholders (affected parties and agencies) in the decision making process from initial problem conception. The need is much greater in fields where problems are poorly structured, as often happens in water resources. As a result, decision support systems have become an essential subsystem within the framework of broader management information systems. The difference between the two is that a management information system provides information to solve problems (usually recurring), and a decision support system (DSS) positions stakeholders in the center of the decision making process, providing help in solving both *ad hoc* problems, as they arise, and recurring problems (Mittra, 1986). Object-oriented programming approaches (OOP) are bringing water resources planning a step closer to the ultimate goal of managing river basins as complete holistic dynamic systems (Palmer et al, 1993; Reitsma et al, 1993).

This chapter presents a conceptual Decision Support System (DSS) for collaborative group planning of hydroelectric development projects, which will be more specifically developed in later chapters. The framework for planning is described with a systems approach for dynamic modelling of the decision process.

4.1 An Introduction to Collaborative Decision Support Systems

The new ethic of sustainability not only reinforces but also extends the main principles of water resources management. Decision support systems have their role in the implementation of these principles into water resource management practice. What is emerging, despite the growing differences, is a new commitment based on fundamental linkages between environmental protection and management, economic development, and the social well-being of people. This three-dimensional approach to sustainability places water resource engineering in a new perspective (Simonovic, 1995). Solutions are required which are not only good for the environment but also for poverty alleviation and wealth creation.

The use of public participation approaches is a growing trend for proponents of resource development. Environmental consciousness has led to an increasingly difficult process for licensing proposed development of water and other resources. Opposition and competition are forcing many development proponents to consider innovative approaches for resolving disputes early in the planning process - long before a final design is prepared for environmental impact assessment and public review.

Many costs to the proponent can potentially be saved by seeking an adequate level of consensus among stakeholders before proceeding with detailed design and preparation of projects such as hydroelectric generation stations. If a detailed proposal is rejected during the licensing process, losses are incurred by the proponent. If changes to the proposal are necessary, or issues of clarification cause delays during licensing, there may also be significant costs to a proponent. Other issues include the efficient collection of data related to potential environmental impacts, which may benefit from clarification of issues early in the planning process.

Decision support systems are beginning to play a greater role in enabling stakeholders to contribute to decisions with technical complexities. Group decision support systems have begun to address some of the cognitive issues, relationship concerns, and technical communication aspects of working with diverse multidisciplinary groups (Bui, 1987; Andrews, 1990; Andrews, 1991; Andrews, 1992). A review of computer-aided group decision making can be found in Lewis (1993). One of the most interesting sources of advancement in group decision support is being made in distributed artificial intelligence frameworks for cooperative problem solving, which are being driven by large corporations and communication systems (Shaw and Fox, 1993).

Collaboration between development proponents and diverse sets of stakeholders is one approach which may address alternative selection problems, but may also be allowed to generate creative alternatives based on multidisciplinary experience. The flexibility of a decision support system to provide adequate support for such problems may further subdivide the classes of decision support and group decision support systems, creating a niche for collaborative planning support systems.

A collaborative framework for a water resources planning and management decision support system is envisioned as a multiple stakeholder planning environment, in which stakeholders have access to each of three components (*decision context*, *system behaviour*, and *alternative evaluation*) through the use of a DSS. Stakeholders provide *context* to planning decisions. For example, the inclusion of a fisheries regulatory agency may suggest that the proponent is willing to discuss options for mitigating or circumventing fisheries impacts. The domain of a planning scenario also includes knowledge concerning the *behaviour* and interrelationships involved with the system to be developed. Knowledge about system behaviour is used to generate planning alternatives and future scenarios, or to assist stakeholders in alternative creation. In order to *evaluate* alternatives, in a multiple stakeholder environment, some form of decision support is needed to effectively integrate the context of stakeholder perspectives and the results of analyzing system behaviour. Integration of decision context and system behaviour is the motivation for preparing approaches to decision support that promote consensus among stakeholders.

The complexity of potential decision support configurations demands that special care be taken in choosing both the architecture and approach for decision support, with due regard to the goals of pursuing automated support for planners in a multiple stakeholder environment. For example, multiobjective analysis tools such as multicriteria decision aids can perform very useful functions in assisting stakeholders assess tradeoffs. However, they demand very specific types of information which may not be compatible with all of the stakeholders' cognitive abilities. Stakeholders may not think in terms of maximizing a function, or may not understand the consequences of assigning certain preferences.

4.1.1 Collaborative Planning Support System Architecture

Collaborative planning support is not about providing solutions but empowering stakeholders to explore solutions. Sustainable decisions require a multidisciplinary, open access, framework for problem-solving. This demands a great leap in technology integration because there is no formal practice for negotiation and mitigation of interdependent impacts and goals in poorly-defined technical problems. A decision support system (DSS) approach to building potential negotiation and planning tools for achieving sustainability can be described as a collaborative planning support system (CPSS). A CPSS takes advantage of computing facilities for database management, knowledge dissemination, modelling or response simulation, and decision analysis. Components that can be expected within a CPSS are graphical user interfaces, database management systems, geographic information systems (GIS), expert systems (ES), and tool kits of models and analysis techniques. CPSS integrates available computer technologies with modelling and analysis tools in a friendly environment incorporating database capabilities that are designed to allow relevant feedback and accumulate knowledge within the problem domain (Figure 9).

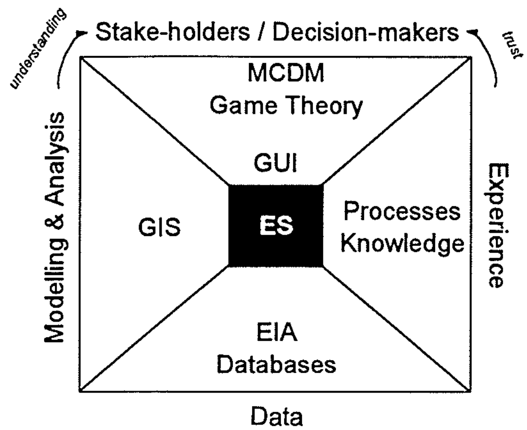

Figure 9. Collaborative planning decision support tools.

Extensive use of computers makes it possible for more thorough use of available data. Most importantly, it allows an iterative planning process to emerge by making analysis and evaluation more convenient. Learning through iterative changes in planning options creates an atmosphere that promotes the impacts on goals or issues that affect stakeholders.

4.1.2 Role of Collaborative Decision Support

Collaborative planning support systems fall in a class of alternative dispute resolution between negotiation and mediation. If a mediator is to be used, there are a number of aspects to consider in choosing the mediator: the forum for negotiation, mediation style, the personality of the mediator, the mediator's experience in the domain of the conflict, and cost of the mediator (Susskind et al, 1987). CPSS is not meant to take the place of a mediator. Rather, it supplements the qualities of a mediator. For example, it helps to clarify the workings of the negotiation forum by formalizing acceptable forms of communication and input. Style and personality are still unique qualities of a mediator, but experience in certain situations can be passed to the CPSS so that knowledge and experience of several mediators can be combined to provide a thorough set of inferences to problem situations. Concerning cost, the cost of mediation may rise with computer aids, but there are other benefits that include better communication with technical staff, and rigid guidelines for the form and extent of EI statements.

Collaborative planning among stakeholders, using a DSS forum for communication, is not a replacement for environmental impact assessment. It is meant to work with environmental assessment in an iterative fashion. Some field studies are necessary to provide an environmental inventory and a basic ecosystem understanding. CPSS then forces stakeholders to clarify their goals, objectives, and perspectives. Within a grounded problem domain, input by stakeholders defines a set of relevant questions or data needs to be fulfilled. Environmental impact statements can then be designed to answer those questions. In this way, superfluous and redundant research is not performed - saving both time and money. In the iterative approach, each iteration leads to more detailed questions that demand more specific studies. The tricky part is not allowing the case where iterative question answering makes environmental impact studies inefficient. This may signal a change in the way environmental impact assessment is approached. (general/broad to detailed/specific).

Constructing an acceptable plan demands that technical and social experience be made available to make inferences about solutions to potential impacts. Integration with a database of previous projects can help to alleviate some of the problems of potential risk and uncertainty. The difficulty comes from the fact that each case may be unique. Projects, however, have similarities as well as differences. Results from previous projects and studies can be compared with the circumstances of the project planning. Additional rules of thumb, based on experience or analysis, can be used to supplement previous knowledge. The method of comparing project scenarios, and ecosystems, is key to accessing relevant data. Feedback to stakeholders can be either quantitative or qualitative in nature. Without a meaningful project database it is still possible to provide feedback from empirical knowledge and results from analysis.

Describing the characteristics and changes in a resource system includes integrated modelling capabilities, and a tool kit composed of analytical models or information processing technologies. A tool kit may include specialized computational routines, access to external analysis facilities, tools for presentation, an architecture capable of simulating operating policies, management models for decision analysis, and others. It is not enough to make all the separate entities of the tool kit available. An effective tool kit moulds the various tools for flexible management of system attributes. For example, if the full supply level of a dam is changed, references to attributes will change: such as flooded area of reservoir, volume storage in the reservoir. Storage can be easily calculated if stage-storage curves have been computed, but if dykes are added to preserve a ravine from flooding and possible erosion - then more general spatial analysis tools may be required. Preparing how and when to use tools in a proactive fashion allows more straightforward adaptation to proposed details of a plan.

Extensive use of computers in water resources planning makes it possible for more efficient use of available information. It also allows an iterative planning process by making analysis and evaluation more convenient. The concept of collaborative planning is an idea that has seen limited use because of the complexities that prevent people with different backgrounds from communicating. A Collaborative Planning Support System (CPSS) integrates available computer technologies with modelling and analysis tools in a friendly environment. It is aimed at enhancing the communication between a proponent for resource development and affected or interested parties (stakeholders).

Collaborative planning support does not provide solutions but empowers participants in the process to explore solutions and reach consensus. CPSS enhances learning through interactive changes in planning alternatives, and by assisting the integration of goals, or issues of importance for process participants, into the planning process. Our belief is that this type of planning environment is effective in identifying areas of common understanding among participants.

4.2 Conceptual Group Decision Processes

Decision-making, or the art of making choices, is a process that involves both objective and subjective aspects. Figure 10 demonstrates subjective and objective components along a path from choice to decision. Subjective aspects are composed of a conceptual understanding of the management system, experience, and a set of values. Whether it is a conscious effort or not, these aspects affect decisions. Tools for making decisions such as multiobjective analysis techniques form an objective component of decision-making, along with data and modelling.

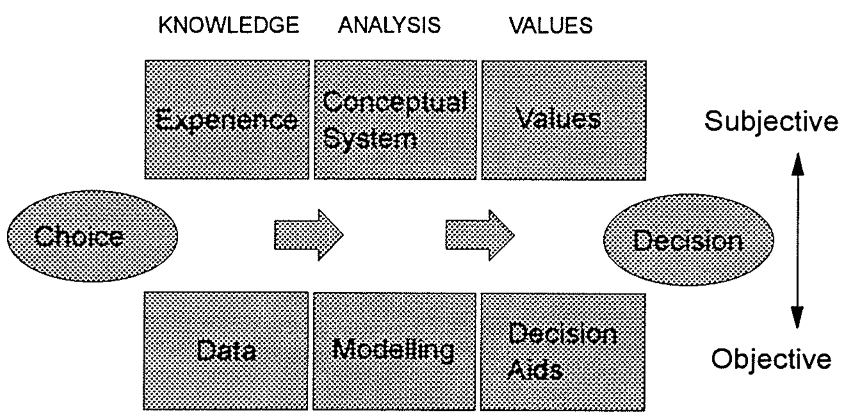

Figure 10. Conceptual Decision Path

4.2.1 Public Decision-Making

The following section will avoid a lengthy discussion on the role of formal analysis in public decision-making. Scientists have understood since the early '70s that public decision-making demands the consideration of subjective values and tradeoffs, and also that problems need to be decomposed to their constituent parts to undercut rhetoric (Keeney and Raiffa, 1974) and shift the emphasis from bargaining positions to issues (Fisher et. al., 1991). The objective in this chapter is to consider systems analysis on a higher level, as a driving principle of collaborative public decision making. In this way, the important conceptual role of formal systems analysis in public decision-making can be shown.

Set of accepted conditions for the development of a systems approach to modelling dynamic group decision processes:

- Decision problems in the public domain are complex.
- Systems analysis framework needs subjective inputs. Systems analysis with subjective input is called decision analysis (Simonovic and Bender, 1996).
- Decision analysis aims at promoting good decision making, not at substituting decision makers.
- Public decision making strives to find equitable solutions for all participants in the process.
- Decision analysis framework combines (a) systems decomposition; and (b) systems synthesis.

In many instances of public decisions, a proponent initiates the process of potential change. This proponent-initiated process is typified by environmental licensing processes and public hearings. An alternative initiated by the proponent is inspected, debated, and then either accepted, rejected, or altered.

Public decision processes such as the one described above are historically "reactive" in nature. The stakes are high, and any changes to the given alternative are expensive. Although the public may exercise some control over the rejection of alternatives which are generally poor, the "reactive" process does not provide opportunities to efficiently improve alternatives.

Collaboration and public involvement early in the planning process of potential alternatives is a growing trend. It is more "proactive" in nature than historical public decision processes. There are many benefits to pursuing a collaborative process, such as alternatives which are more overall acceptable, and alternatives which are able to effectively or creatively mitigate potential adverse impacts. The difficulty in applying "proactive" approaches is that they demand interaction and integration of technical disciplines and social perspectives.

who represent the range of relevant perspectives. Choosing an appropriate group may also be a sensitive issue in applying collaborative decision processes.

4.2.2 Collaborative Decisions

Multicriteria decision tools and other multiobjective systems analysis techniques provide personality to the results of modelling analysis, and attempt to add meaning to the subjectivity of value systems by describing in quantitative measures some of the values and preferences of stakeholders. Decision tools in a collaborative planning framework may be used to manage and monitor many of the menial descriptors of a decision process. In traditional management decisions, a decision maker may formulate the decision choice in a tabular format for use in a multicriteria technique. There are usually choices, and criteria or issues by which to judge the different options. The decision maker may perform sensitivity analysis of the relative importance of different issues to establish a decision frontier. In a collaborative planning problem, there is no single formulation, but numerous choices each with a set of issues which may be dependent on other issues.

The traditional multicriteria approach can emulate the multiple problem type, depicted in Figure 11, by enlarging the problem to include all issues, and all options for all choices. To evaluate any sub-problem, unnecessary issues and options can be removed. This form of emulation, however, cannot perform simultaneous sensitivity of the interdependent set of planning choices. Advances in the decision sciences will be made on focusing on the collaborative planning decision problems, because present techniques are inadequate and fulfill only partial roles.

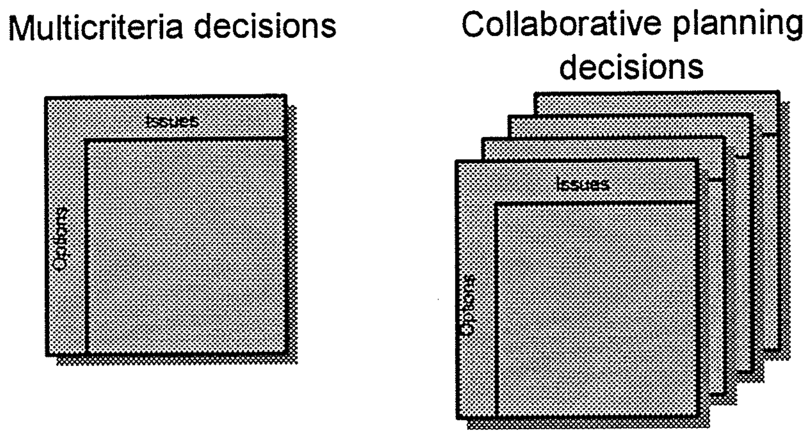

Figure 11. Multiple Problem Emulation by Multicriteria Techniques

Many techniques for multiobjective analysis are available to evaluate tradeoffs between predefined proposals. However, the multidisciplinary development of alternatives, and criteria for iteratively rating experimental alternative adaptions by multiple stakeholders is much more complex. Framing a proposal considers alternatives within a decision tree. At each branch there are choices which lead to still other choices. There are branch and bound methods to deal with this type of problem. The difficulty for proponents is the uncertainty of the impact (or shadow) one branch may cast on another when viewed under certain lighting conditions. The success of a collaborative planning approach is measured by the inspiration of multiobjective tools that assess inter-related alternatives on multiple branches.

Literature in group decision-making almost completely ignores the fact that creating an alternative is extremely difficult, especially for technically diverse groups and nontechnical stakeholders. Most techniques presented for group decisions are limited to aggregating preferences of multiple decision makers. It is generally assumed that individual stakeholders have evaluated alternatives *a priori*, either by directly entering rankings, or by manipulating weights and available preference information. The literature also assumes that the best discrete alternatives have been made available for discussion. There are techniques for generating alternatives, but mostly for either continuous decision spaces, or using branch and bound algorithms. Neither are interactive by nature.

Bales and Strodtbeck (1951) identified 5 main types of functional problems faced by groups:

- Orientation.
- Evaluation - different values and interests need personalized framework to individually express wishes and feelings.
- Control - each party agrees to work toward finding directions to reach consensus among alternatives.
- Tension management.
- Integration of group members.

People haven't changed much since the '50s. Neither have the problems faced by groups. This is quite evident in the interest in group decision-making techniques, and in the arena of public decision-making methods.

4.3 Systems Approach to Dynamic Modelling

4.3.1 Theoretical Systems Framework

A systems approach is a discipline for seeing wholes. It is a framework for seeing interrelationships rather than things, for seeing patterns of change rather than static snapshots for seeing processes rather than objects. It is a set of general principles, and also a set of specific tools and techniques. A structure, or theory is essential if we are to effectively interrelate and interpret our observations of the problem domain. However, public decision making is being seen as simply lacking an integrating structure, knowledge is a mere collection of observations, practices and conflicting incidents. Systems approach is a discipline for seeing the "structures" that underlie complex domains. The main concept of systems approach is the concept of "feedback" which is emerging as the basis for structuring our observations of complex systems and their economic, social, political and environmental implications.

A basic assumption of systems approach is that the dynamic behaviour exhibited by a system is produced by the structure of the system. What is structure? The answer is that it consists of many components combined into infrastructures, which are then equipped with feedback loop relationships. It is feedback loops that enable infrastructures to realize their full potential for generating dynamics. In particular, feedback relationships enable infrastructures to generate behaviours, as well as to produce responses to policy initiatives and other forms of human intervention. While infrastructures define the range of behaviour patterns, the particular kind of feedback relationships that are superimposed on the infrastructure will determine which of these patterns will be realized. The feedback relationship is the basic building block within the system boundary. The feedback loop is a path coupling decision, action, level (or condition) of the system, and information, with the path returning to the decision point.

4.3.2 Feedback Paradigm

In our discussion, a decision process is one that controls any system action. The decision is based on the available information; the decision controls an action that influences a system level; and new information arises to modify the decision stream. Every decision is made within a feedback loop. The decision controls action which alters the system levels which influence the decision. A decision process can be part of more than one feedback loop.

There are currently two types of dynamic modelling paradigms understood by systems analysis. One is negative feedback, which forces system transformation toward an external goal - economic efficiency, for example. The concept of an external (artificial) goal is a source of model fluctuation and instability, due to dynamic overcompensation or undercompensation. In many cases external goals for complex systems are only partial representations of the system scope, and may be inadequate.

The second systems modelling paradigm is the concept of positive feedback. Positive feedback behaves in a similar manner to many natural growth processes in which the system feedback instigates growth away from an external goal or reference point. In other words, there is no predetermined optimal solution or normal behaviour pattern. Model behaviour results from initial conditions.

4.3.3 Dynamic Feedback Processes

A feedback relationship is a closed-loop circle of cause-and-effect. There are two distinct types of dynamic feedback processes:

- reinforcing (or amplifying, or positive)
- balancing (or stabilizing, or negative)

Reinforcing feedback processes are the engines of growth. Positive feedback "adds to", or reinforce change leading a process to compound or spiral, causing it to gain momentum as it goes. Acceleration may occur in both directions. Both, good news and bad news reinforcing loops accelerate so quickly that they often take people by surprise. But pure accelerating growth or decline rarely continues unchecked in nature, because reinforcing processes rarely occur in isolation. Eventually, limits are encountered which can slow growth, stop it, divert it or even reverse it. These limits are one form of "balancing feedback".

Balancing feedback operates whenever there is a goal-oriented behaviour. This type of loop negates change. Push a condition in one direction and a negative feedback will cause it to bounce right back in the other. Negative feedback relationships thus seek to maintain conditions in line with fixed goals. Nature loves a balance. However, many times human decision makers act contrary to these balances. Balancing feedback processes are everywhere. They underlie all goal-oriented behaviour.

Planning creates long-term balancing processes. What makes balancing process so difficult in public decision making is that the goals are often implicit, and no one recognizes that the balancing process exists at all. Balancing loops are, in general, more difficult to see than reinforcing loops.

4.4 Dynamic Model of a Collaborative Planning Process

Instead of looking into the details of decision making framework we would like to point out the importance and benefits of a systematic approach for the development of the decision making framework, applied to planning hydroelectric water management. This approach has been used for the development of a Decision Support System for collaborative planning of hydroelectric power generation (Bender and Simonovic, 1996c), presented in the following section of this chapter.

4.4.1 Model Framework

A Collaborative Decision Support System is designed for interaction and participation of a group of stakeholders with a project proponent early in the planning process, before an alternative is presented for licensing (Bender and Simonovic, 1994). Application of decision support in this form can benefit from adopting the conceptual systems approach of using "positive" feedback to push away from areas of conflict, and by describing the dynamic interaction of stakeholders with the decision process using "balancing" and "reinforcing" feedback mechanisms.

A framework for planning hydroelectric water management policy is demonstrated as a positive system feedback approach, which assumes that the proponent and stakeholders are interested in moving away from the reference point of conflict about a decision. Public policy decisions may be modelled to push toward external goals. They may also be modelled to push away from an undesirable reference point. Public decisions may involve environmental licensing and public hearings, which add to the complexity of project planning. The negative feedback approach tends to publicly debate proposed solutions, adjusting them until they reach an external goal based on an overall acceptability indicator. This approach is conceptually limiting. The positive feedback systems approach initiates changes to a proposal away from the initial reference point of conflict or disjoint value systems. Its direction and pace are flexible, which may be desirable properties for group behaviour.

Direct involvement of stakeholders in a group planning context early in the conceptual planning stages is seen as a significant step toward achieving the external goals of the traditional negative feedback process, but also creates a potential for surpassing those limiting and partial modelling objectives. Without formally internalizing concerns which are external to a proponent, involvement of stakeholders brings to bear many of the externalities simply by expressing their value systems and impressions of performance for an alternative.

Within this collaborative decision process, the role of systems analysis tools is somewhat altered from their traditional use. The traditional systems approach to problems has been to simulate, optimize, or choose a compromise solution based on tradeoffs between conflicting objectives. However, systems thinking is evolving into concepts that may help us understand how to approach complex technical problems that affect or involve people.

Systems analysis tools should be used to facilitate good or creative decisions, not to recommend a "best" solution (Brill, 1979). In an approach based on more contemporary systems thinking, the group of stakeholders control the decision process and settle on a recommendation. Systems analysis tools no longer recommend an alternative. They provide "balancing feedback" that help people to understand the behaviour of alternatives and the implications of choices.

In no way does collaborative decision support attempt to control the pace or direction of the decision process. The direction is set by stakeholders as they experiment with different combinations of technical options or with different value judgements and attitude toward existing alternatives. The pace is controlled by the ability of stakeholders to cooperate, disseminate knowledge, assimilate knowledge, and by their degree of interest in specific details about different aspects of the problem domain.

4.4.2 Dynamic Model Functionality

The reference point for dynamic model functionality is the initial system state. Initial conditions represent the *status quo*, which probably is contributing to conflicting opinions about change. Using a positive feedback process, the planning process dynamically searches for decisions which are less conflicting than before. There is no optimal decision, or minimum standards as a final external goal. The greatest assumption of the positive feedback process is that less conflict is better! Of course, this assumption also makes other assumptions such as the inclusion of all relevant stakeholders into the process. Likewise, if degree of consensus (A) is the complement of degree of conflict (C) and both are measured on the range [0,1] (such that A=1-C), then group consensus about a decision is an implicit external goal.

To model the dynamic processes of reinforcing and balancing feedback in public decisions (or public evaluation of decisions), we can learn from the tendencies of the traditional "reactive" approaches of accepting or rejecting proposals. People, in many cases, decide that they are either for or against an alternative. The result is a polarization of perspectives. The alternatives, judgement criteria, and any attempt to recommend a best solution are "reinforcing" factors which are used to justify bargaining positions.

Balancing feedback, or stabilizing dynamic factors, are facilitated by knowledge transfer and empowerment of stakeholders. As stakeholders improve their understanding of the decision context, they have an opportunity to make a more informed proposal. The decision context includes the value systems of people, but also includes the relationship those value systems have with the chosen alternatives. For example, a valued fishery can be impacted in several ways by several aspects from proposed development. Understanding those links is vital to understanding the context of a decision. Figure 12 shows the conceptual flow of information in the decision process: stakeholder input to the problem context and domain (scope of technical alternatives), and feedback in the decision process to the stakeholders.

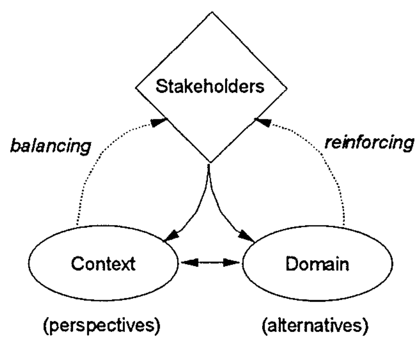

Figure 12. Sources of balancing and reinforcing feedback.

Providing balancing feedback, and facilitating the understanding of the various links that help define the problem context, is essentially a knowledge base problem. In the form of computer software decision support, knowledge bases take on the shape of expert systems or some other type of applied artificial intelligence technique. Knowledge bases in a human environment take on the shape of experience, technicians, and consultants. In whatever form, knowledge bases are resources for stakeholders with different technical backgrounds and different "technical languages" for describing their value systems.

4.4.3 Decision Support Description

The Collaborative DSS presented in this paper uses a decision process which contains three main modules:

1 Criteria selection.
2 Alternative generation.
3 Decision evaluation.

Inputs by stakeholders to the DSS are one of the following types:

- Values.
- Technical options.
- Impressions of alternative performance.

Output to the stakeholders mirrors the above inputs:

- Problem context.
- Alternative behaviour.
- Decision robustness.

The first type of input, stakeholder values, express the scope of concern for each person. Output from value elicitation is a "problem context", defined collectively. The second type of input, technical options, demands that options are made available. Alternative behaviour is the output resulting from specification of a set of technical options, as described by a tool box of models and analysis techniques. Alternative behaviour includes benefits and costs, but also describes aspects of uncertainty such as future discounting and health risks. Finally, input of impressions about alternative behaviour can be used to output the robustness of a possible decision. While technical options carry performance uncertainties, decision makers carry uncertainties about their values and perceptions. In this way, different groups result in different levels of communication, mutual understanding, and cooperation.

The criteria selection module acknowledges that the choice of judgement criteria is variable. Individuals may differ greatly, and they may also (unknowingly) be redundant. The choice of judgement criteria and their relative weight in assessing alternatives can be delicate. Figure 13 illustrates the process of feedback to stakeholders as they explore choices in criteria. The input of stakeholder values, in the form of facts or properties of the problem domain considered to be important, are accessed by knowledge bases which suggest relevant criteria (Simonovic and Bender, 1996). This process of inducing judgement criteria from known or measurable facts is similar to grounded theory approaches in the social sciences (Glaser and Strauss, 1967).

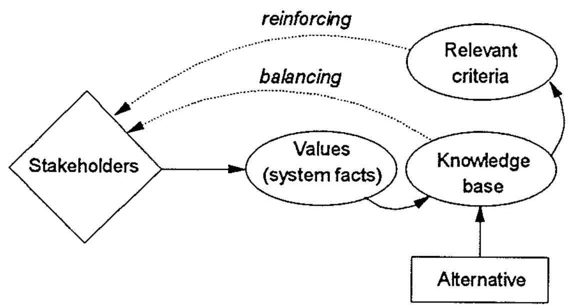

Figure 13. Feedback in the criteria selection process.

Feedback in the criteria selection process is both balancing and reinforcing. The choice of criteria may "reinforce" the opinions of stakeholders, by encouraging additional bias. However, the description of reasoning by the knowledge base acts to "balance" subsequent changes by explaining degree of importance and potential impact on valued facts.

Alternative generation within the group setting assumes an iterative, flexible, modelling posture. Stakeholders are able to specify technical options from the problem domain. Knowledge bases are then used to determine appropriate model analysis given the context of the problem (the selected judgement criteria). The model analysis, in turn, describes the behaviour of the alternative (Figure 14).

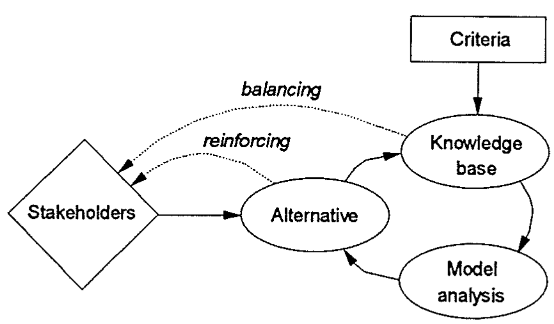

Figure 14. Feedback in the alternative generation process.

For example, a technical option of building a dam at a specific location with water levels raised to a given stage, model analysis determines the cost of construction, flooded area, storage capacity, hydroelectric generation potential, and the uncertainties of alternative behaviour. Impacts from flooding and altered water regimes are addressed using modelling tools such as geographical information systems. Stakeholders are able to interactively adjust the system to visualize changes, designed to improve understanding of system behaviour.

Alternative behaviour is likely to "reinforce" the direction of subsequent choices in technical options. Knowledge bases must "balance" the behaviour of the stakeholders by explaining how the models reach their conclusions, and how appropriate analysis is driven by the given context of the problem.

Decision evaluation is a process of examining tradeoffs and exploring the sensitivity of decisions to uncertainties in alternative behaviour. As well, the prospect of making a choice illuminates any uncertainty in criteria selection. The value systems that suggest appropriate judgement criteria also suggest degrees of risk aversion and level of aggressiveness for judgements and decisions made by stakeholders. Robustness of alternatives to uncertainty in judgement criteria is a desirable quality for many decisions. Another aspect of robustness is the robustness of a decision to different types of stakeholders - each willing to operate at unique levels of risk aversion and aggressiveness.

In the process of multiobjective analysis shown in Figure 15, experimental alternatives are ranked in terms of decision robustness relative to apparent issues and preference structures. This is accomplished with a generalized fuzzy compromise approach (Bender and Simonovic, 1996a) that identifies consensus (or lack of conflict) in terms of a collectively-defined displaced (infeasible) ideal solution.

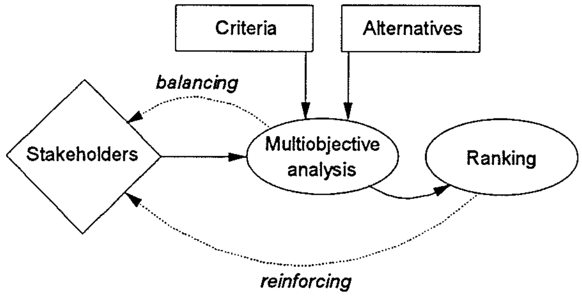

Figure 15. Feedback in the decision evaluation process.

The formal multiobjective approach provides a framework designed to "balance" the "reinforcing" implications of seeing which alternatives are ranked higher! It provides structure and a specific form of expressing both judgements and degree of subjectivity.

4.5 Summary

4.5.1 Use of Systems Approach

The initiation of collaborative planning processes may benefit from the conceptualization of stakeholder participation using a systems approach. Overall, the positive feedback mechanisms of iteration and experimentation allow alternatives to be generated, assessed, and improved. stakeholders are also able to explore their value systems, gain insight on potential impacts, and evaluate the collective judgement of participants.

The reference point for pursuing collaborative planning is the state of conflict (or *status quo*) among stakeholders. The process does not have an ultimate external goal, except the implicit goal of consensus. The motivation for pursuing this form of approach is the potential of discovering creative solutions from combining the disjoint aspects of stakeholder perspectives.

Implementation of collaborative decision support for public decisions is limited by several factors.

- Acceptance, trust by participants.
- Learning time for using decision support tools.
- Accumulation of domain knowledge.

However, if these limitations can be overcome, we feel that the risks of proposal rejection and costs of planning will be reduced, and that more creative solutions will emerge.

4.5.2 Collaborative Decision Support Implementation Issues

Collaborative decision support for planning processes can certainly be viable, effective method of planning projects, and a systems approach for conceptualizing the various aspects of a dynamic planning framework can adequately identify the suitable roles of modelling and analytical tools. The greatest obstacles to implementation are the sociological and psychological baggage of the stakeholders who are participating. The processes and framework discussed in this chapter assume that stakeholders voluntarily participate within a cooperative environment, where a development proponent proactively pursues input by other people.

Another issue in implementation of collaborative decision support is the identification of an appropriate scope for the problem domain, which is related to the issue of choosing appropriate stakeholders for participation. Generally, all relevant perspectives should be represented. The danger comes from redundancy, and the potential overemphasis of some perspectives. Likewise, if the range of technical options are too restricted to empower some valid perspectives, the result is an implicit underemphasis of some perspectives. Likewise, if the range of technical options are too restricted to empower some valid perspectives, the result is an implicit underemphasis of some perspectives. Applications of collaborative decision support should be encouraged to avoid the common practice of weighting individual decision makers. This is a difficult task, but the feedback processes in the criteria selection module provide an example of the role of heuristics (experience) to suggest relevant criteria based on an examination of stakeholder values.

There is a great amount of flexibility in how inputs are secured from stakeholders, and how output is presented. The primary interest in providing decision support is to disseminate knowledge. Typically, the more intuitive and transparent, the better! Awareness of this issue may impact the necessary power, but if they are viewed as a black box they may contribute elements of distrust. In addition, visually appealing techniques such as techniques based on fuzzy sets may actually preserve more information than more complex analytical methods. The degree of balancing feedback is dependent on the effectiveness of interaction between stakeholders and the chosen decision tools.

Probably the most ill-defined task for any collaborative decision support application is the design of knowledge bases and their function in the feedback process. They must provide technical expertise and experience in the problem domain, in the form of suggestions, explanations, or various insights. Technicians, analysts, and experts would normally provide these functions when the collaborative process relies on multidisciplinary teams of participants. However, as collaborative planning for a specific problem domain relies more heavily on automated decision support tools, issues of knowledge base design become more pronounced. Heuristics and technical expertise must be organized in such a way that people from different backgrounds and areas of expertise are able to learn about the interrelationships between actions and impacts given the different perspectives. In other words, the quality of knowledge base organization directly contributes to the quality of balancing feedback in the decision process.

Chapter 5 Selecting Evaluation Criteria using a Grounded Approach

Within the framework of a decision support system for multiple stakeholders, introduced in the previous chapter, special attention can be devoted to the problem of determining evaluation criteria. An independent decision support module is developed for selection of appropriate evaluation criteria. Its description is provided in the following section. Evaluation criteria, formulated as objectives, are usually expressed in a very abstract form. Examples are: maximization of net benefits; minimization of negative environmental effects; maximization of social well being of people; etc. On the abstract level many of the objectives carry adversarial connotations. Conflict between development and environment concerns can be used as an example. However, the authors believe that in every adversarial situation there is ground for compromise if communication between stakeholders can be maintained. Maintaining communication includes elimination of language barriers. Each stakeholder should be able to express their goals and preferences (in their own "language"). Moreover, these preferences should be clear to other participants in the process.

The evaluation process in a collaborative decision support environment is based on feedback provided by the system during the search for a socially desirable management decision. Generally, a large group of stakeholders results in complexity, potential conflicts and high transaction costs, assuming a solution can be found to reach consensus. To assess tradeoffs between solution alternatives, a set of knowledge must be available which refers to the accumulated experience.

This chapter provides an example of one collaborative planning support system (CPSS) module to determine relevant project evaluation criteria based on the selection of important system elements by stakeholders (Simonovic and Bender, 1996). The application incorporates knowledge related to hydroelectric development. In preparing a decision support tool for experimentally determining project evaluation criteria, stakeholders can be provided with opportunities to learn and contribute early in the planning phases of potential resource development projects. The approach used to foster communication and understanding between stakeholders (in the context of hydroelectric development) is viewed as an important direction for pursuing various sustainability axioms.

5.1 Context Management

The thought of working side-by-side with other stakeholders from a number of scientific disciplines and ethical backgrounds can invoke all sorts of logistical nightmares. Facilitating stakeholder involvement is not an easy task. Each stakeholder will probably have a unique set of language references for framing development options. Two or more people working together will never agree on everything. Each will have a different perspective, a different agenda of goals or objectives, and different moral or ethical priorities. Negotiations between stakeholders tend to be adversarial because there exists a set of judgement criteria which is entirely unique to each person. This makes it difficult to explain and demonstrate a proposed development plan. It's difficult to grasp all the physical changes to a system. It's much more difficult to grasp all the impacts this will have on the world of a stakeholder.

Other issues in stakeholder involvement are more subtle. Trust is something that is difficult to earn. Stakeholder confidence in the information and recommendations provided by other stakeholders or decision makers may be suspicious at best, and contributes both to persistence of bias and reactions within a zero-sum game attitude. The posturing that is a part of game theory is something that is difficult to avoid because it is a natural evolution of decision making when multiple stakeholders are involved in a problem with more than one valid objective (Hipel, 1992). Any collaborative approach must attempt to remove some of the barriers that prevent free information transfer and creativity in the search for acceptable development proposals.

In a collaborative planning approach, stakeholders do not waste time debating a suitable set of criteria or objectives. Instead, each stakeholder is allowed the freedom of expressing their perspectives within individual goal structures, with open access to viewing other stakeholder goal structures. Interactive exploration of technical decision options will help advance user understanding of their own priorities as well as other stakeholders. If each stakeholder is allowed the flexibility of changing both the proposal and their own proposed goal structure, trial and error learning may advance user understanding and encourage dialogue between stakeholders.

Flexible multiple user visualisation is important for collaborative planning because there is no "right answer" proposal, nor is there an absolute objective structure. Goal structures can be extremely complex, and diverse for any given case because they reflect the thought patterns of individuals. Keeney (1992) provides a number of examples of goal structures. They are not necessarily hierarchical. They range from broad strategic goals to very explicit tactical issues, but can be interrelated in any fashion. The only requirement is that each objective be tied to the rest of the goal structure through one or more contextual or relational links that describe why an objective is important or what it is related to.

There are 2 reasons for allowing individuality in the expression of decision context, for flexibility in learning about your own values, and as a basis for making claims. Keeney (1992) demonstrates how people may have difficulty expressing their abstract values and motives for liking or not liking something. As stakeholders receive feedback from changes in a development proposal, each person has an opportunity to witness how their value system is impacted by changing physical attributes. Through making changes to their own goal structure, stakeholders learn about their own motivations. Making each stakeholder unique, and accountable for their own value system, it is possible to better understand the motivation of others. There is a danger that people will withold information because of it. At the same time, if claims are made, or decisions are chosen, the reasons for making those choices must be evident in the value system if stakeholders are participating honestly.

The description of context for each stakeholder easily adapts to an object-oriented framework for managing hierarchies of goals. Actually, a goal structure does not need to be strictly hierarchical. It can be loosely judged as hierarchical, but it is entirely possible that cyclical patterns and odd dependencies emerge. Poh et al (1994) describe this general process of categorization of issues in terms of a knowledge representational scheme that uses influence diagrams, a tool of decision analysis, and inheritance/abstraction hierarchies, an application of artificial intelligence. Abstraction hierarchies in general are discussed in terms of planning decisions in Knoblock (1993).

An object-oriented goal structure defines goals as objects. Each of these context objects are then described in terms of their relationship with other context objects, using parent or child links. Parent links refer to "why" a goal is important. Child links generally refer to "how" a goal is satisfied or "what" components comprise the parent goal. In this way, a structure emerges that generally contains abstract high-order parents, and technical low-order children.

By choosing to focus on a specific goal, a stakeholder can assess how certain physical variables and impacts contribute to broad implications. However, the goal structures of stakeholders are limited in scope by the detail in expression. On its own, context management has no method for collaboratively accessing the tool kit of models and the results of choices on a physical resource system.

5.1.1 Relational Links

Relational linking is the methodology by which decision makers view impacts of changes in the physical systems on their description of objectives or goals. Links between *physical* and *context* systems provide an avenue for feedback from updated planning proposals. Automated linking is a concept that allows decision makers to remain within the language constraints of their objective structure, while accessing multidisciplinary technical expertise.

Effective structuring of the decision problem demands a compilation of available and accessible technical alternatives. The quality of relational links between the physical system and the objective structures is dependent on the extent of technical experience. The needs for this decision resource consists of a historical inventory of methods or techniques, and interdependent or interrelated consequences of each inventory option. From a multidisciplinary inventory, technical support may suggest a list of potential baseline indicators. They represent an inclusive list of possible areas of concern that are direct outcomes from including a component within the physical system. This extensive endeavour demands both data intensive environmental assessment and technical insight from experienced professionals. Some of the knowledge may be in place before environmental assessment, and some studies may be irrelevant. The greatest problem with technical support is that data is not normally in an applicable form for predictive analysis or management model application. Biological data in particular may conform to biological standards, but the data in most cases can only be used for qualitative summary and is normally not built into any ecological taxonomy of quantitative decision variables and objectives.

The challenge of understanding technical information, for many of the stakeholders, may be a very serious limitation to their ability to contribute to the collaborative planning process. This is essentially a language issue. It is related to the challenge of assimilating natural language descriptions of objectives and constraints, which are used to model technical input for management decisions (Smith, 1993). The technical objectives and constraints are the language of the engineers and other planners, while many stakeholders have either a unique set of objectives and constraints or a completely abstract set of goals. There are 2 choices to consider for communication links. One is to use a language parser, a text-based tool for describing statements of objectives or their constraints in a natural vocabulary. The other option is to develop a graphical interface such that the choices for the user has a restrictive vocabulary and order of appearance, dependent on previous selections (Dougherty, 1994). Both forms of language use result in approximately the same language solution. That is, restrictive vocabulary and grammar.

To enable integration of technologies and feedback, a natural language system of descriptors, used in conjunction with quantified results from analysis and comparisons may be an efficient means of fostering real feedback to stakeholders. Language references will need to be restrictive to produce a viable product in a reasonable time frame. However, in a well organized model of knowledge handlers, implemented in an object-oriented framework, there are options for easily expanding the terms of reference. The remaining question is: How do the language references and technological integration refer to the stakeholders and their context of goals?

5.1.2 Knowledge Representation

Within the discipline of Artificial Intelligence, the study of expert systems provides a form of knowledge representation that may be effective in describing the interrelationships that we wish to present. Expert systems are good at describing both quantitative and qualitative relationships. They can retain knowledge in a modular form, preventing over-complexity when reviewing or updating the knowledge elements. Questions concerning expert systems as the initiating tool for managing relational links are: what will be the architecture for rules in a general object-oriented approach, and what language defines the information to be used throughout the user context? There are practical difficulties in developing and maintaining knowledge bases. Conditions to trigger rules can be very complex and conclusions can be dependent on database information, the results of models, or even other rules. Most rules contain a focus of objects and/or properties as well as a test of some kind. Within the focus and test, though, can be a wide variety of queries. These and other considerations need to be addressed in the planning phase of an expert system.

Completeness and appropriateness of a language system for an expert system is difficult to ascertain. The problem domain must be clearly defined (eg. hydro, fish passage). In choosing language domain descriptors, the type of expected feedback must be considered. Some examples are:

- Values passed (quantitative object properties).
- Qualitative interpretation of impact.
- References to similar cases or experience.
- Advice for potential alternative actions.

5.2 Grounded Approach

The organization of feedback from physical systems to contextual hierarchies in a collaborative planning approach may benefit from an analogy in the qualitative research of social sciences. Sociologists, in qualitative studies, induce theory about social processes by compiling available facts, and building on elements of known fact into higher-order abstractions. This inductive process is referred to as grounded theory (Glaser and Strauss, 1967).

5.2.1 Grounded Theory

Grounded theory is in direct confrontation to classical sociology and anthropology which tended to make assumptions and deduce evidence to support claims on the agenda. Deduction is not unlike the process of mitigating social and environmental impacts of water resource development. Instead of asking what domain of knowledge and important issues should be used to judge a proposal, the traditional approach is to assume partial (economic) analysis as the means of choosing efficient solutions. This is followed by attempts to efficiently justify decisions through mitigation of external impacts. The deductive approach in the social sciences led to stifling processes for incorporation of new information into existing theories. Inflexibility led to catastrophic paradigm shifts after unbearable buildup of unclassifiable information (Kuhn, 1962).

Grounded theory removes the constraints of having to prove previous statements and satisfy prior expectations. It is essentially a search for the best available paradigm, valid until new information comes available. Observations, or any valid knowledge about a problem domain, are gathered without preconcieved ideas about relative value. Abstract theory, described by core categories of behaviour or process, are then constructed from the ground up. Hence the terminology. Where debates about deductive theories focus on inclusion or possible exclusion of data, inductive theory processes constantly debate the valid use of data.

The use of grounded theory is an iterative process because descriptive constructs typically develop through many iterations and changes as the researcher finds better ways of organizing facts into groups and processes. As the researcher develops a better understanding of the material, higher-order abstractions begin to emerge. The result may be unique to the researcher, due to previous experience, perceptions, and expertise. Any bias in the governing mental constructs, however, is somewhat constrained by the use of factual evidence to initiate the explanation of preferences and processes.

Constructs from domain knowledge tend to be hierarchical, although they can be loosely structured and even co-dependent. Abstraction of grounded theory occurs in multiple dimensions, ultimately falling within global core categories. Typically, there may be 2 dimensions that a researcher may stress. One may be to follow chains of causal dependencies in events or behaviour. The other dimension may group facts into taxonomic or typological constructs. These 2 dimensions may intersect each other and interact, but for the most part they are unique in scale and scope. Other potential dimensions of abstract constructs will generally follow one of these patterns to describe processes of various proportion. There is no single accepted methodology for using grounded theory. Only the general form of the methodology is consistent. Examples of grounded approaches can be found in Glaser (1978), Spradley (1980), and Lofland and Lofland (1971).

5.2.2 An Illustration on the use of Grounded Information in Collaborative Planning

Analysis tools require input, and result in output of some form. The output of a modelling or analysis tool is a set of grounded facts that are known about a system. Raw output is not always the ultimate form of grounded knowledge. Conditional combinations of grounded facts contribute to other grounded facts. Together, analysis output and conditional combinations form a grounded layer of domain knowledge. A list of valued ecosystem components, indicators, or parameter values, form the grounded basis for describing stakeholder goals.

The analogy of grounded theory can be used to organize the flow of feedback in a collaborative approach by allowing multidisciplinary goals and preferences to be directly linked to components in the physical system. Each user must describe issues in terms of quantifiable grounded facts and build a unique set of preference constructs toward the relevant core issues. This results in a set of language constraints (as combinations of grounded facts) unique to each stakeholder, and allows various stakeholders freedom of expression for participating in evaluating alternatives.

Presently, there is no adequate mechanism for defining objectives (as a dynamic set of language constraints), and relating those preferences to available expertise and analysis facilities. Grounded knowledge acts as a foundation toward understanding abstract core issues. Discussions develop from known facts rather than adversarial negotiating positions. Grounded objective structures allow users to define their own language system as part of a process to define values and issues. Objectives in unique language systems are defined as an aggregation of grounded facts. This allows unbiased comparisons of objectives by assessment relative *satisfaction*, or *advice* toward adaptation based on experience related through grounded knowledge.

Are valued ecosystem components (VEC) or environmental indicators the equivalent of grounded facts? Certainly, that depends on how they are defined. The effectiveness of using indicators also depends on the extent that basic variables and object properties are used to justify goals. Some ecosystem components refer to the project as a whole, others are specific to individual objects, or dependent on data availability.

Below are examples of some types of valued ecosystem component for hydro development with environmental considerations such as fish passage:

- Power_capacity.
- Flooded_area.
- Fish_population.
- Migratory_delay.
- Project_construction_cost.
- Habitat_suitability.
- Index of Biotic Integrity (IBI).

A knowledge based approach to qualifying grounded relationships demands that indicators be defined and embedded in rule sets. For instance, since it is impossible to count the number of fish (although you can estimate roughly), a grounded fact called *fish_population* may be a qualitative variable with subjective (fuzzy) terms to describe it such as healthy, reduced, depleted, or extinct. Certain hydraulic conditions may be known (suspected) to cause changes in fish population dynamics. Flooding of known spawning areas may result in degradation in the qualitative description of *fish_population*.

Object attributes for grounded facts are also useful for relating information. Terms in rule conditions such as *migration_route*, or *spawning_habitat* are higher-order language references. A *migration_route*, for instance is defined by the fish species, location in the hydraulic network, and time frame, as well as direction of travel. If tagging studies have been used (or local knowledge is extensive) to define migration routes, a database with this information can be loaded into the domain knowledge base within the class *migration_route*. Instances of this class would then have properties: species; maturity; location (an object reference); start date; end date, and direction. If the knowledge is more detailed, other properties such as temperature can be used to define the starting and ending dates. In this case, more properties for the *migration_route* class are not needed, only methods that search other rules or external models to find the dates which are dependent on certain conditions.

A grounded layer of knowledge, that rule conditions examine, eventually emerges. A migration route is one type of grounded fact that, when incorporated into a rule set, defines a modular knowledge base for a particular domain of issues. There may be many migration routes with various types of concerns. However, a valued ecosystem component such as *project_construction_cost* is a single instance defined by the properties of many components. Throughout modules for providing feedback, there are global variables (like project_construction_cost) and local or intermediate ecosystem components (such as *migration_route* which may contribute to the calculation of *migratory_delay*, another indicator).

A number of knowledge modules are needed for any multidisciplinary problem. For a problem scope of hydro development and management of environmental concerns such as fish passage needs, there needs to be one or more knowledge modules for each area of concern. There is likely numerous modules for hydro development related to hydraulic characteristics, power generation technologies, economic flows of goods and services, etc. Each module can be developed and validated independently. In the case of fish passage, a likely author of the knowledge base is a government regulatory agency such as the federal Department of Fisheries & Oceans or a provincial Department of Natural Resources (in Canada).

Grounded context relationships are linked by single facts, but grounded knowledge is organized within a layer of intermediate interdependent indicators. Figure 16 conceptualizes the grounded knowledge layer between contextual system of values or objectives, and a system of physical resources. The relative thickness of this layer is dependent on definitions of grounded variables. The above cases demonstrate some potential complexity of the grounded layer.

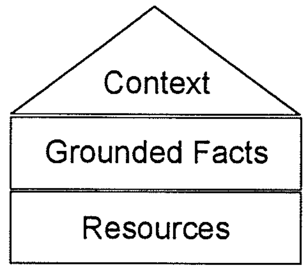

Figure 16. Grounded layer of facts relative to contextual and physical systems.

The assessment of ecosystem aspects has been, and will be, an important task in project evaluation. In the VEC form of grounded facts for relational links, is there room for environmental impact assessment study results that measure a number of environmental variables? How are these variables related to structural changes in the watershed? Obviously, changes in water regime (and other factors) lead to habitat changes in water temperature or quality, and to the vegetation that grows in the altered environment. Models of these variables, and their impacts on more abstract ecosystem components are needed to make value judgements about development options. Each case study will deal with unique ecosystem properties, but there must be at least general trends to follow. Ecosystem models of this form are extremely difficult to use for prediction, especially because ecosystems are also able to adapt and evolve to accommodate change in unsuspecting ways.

One of the areas where environmental impact assessment will have to adjust for implementing collaborative paradigms of proactive mitigation is to focus on potentially reliable indicators of ecosystem health. If models of ecosystem interactions are not viable tool kit options to be included in the list of grounded facts, then value judgements need to be made directly from the context of a decision maker. Basic stream flow information and expectations of changing flow regime will need to be accessed as grounded facts. Then, by using satisficing relationships, a simple general model of expected impact can be qualified and evaluated.

In this way, many simple models can be built independent of the tool kit, and specific to a case study problem.

How are indicators used to relay information through user context hierarchies? Individual users, when selecting a particular goal, trigger a hypothesis to evaluate in the knowledge base. A backward chaining form of inference engine then attempts to pass rules relevant to the goal involved. Because the goals are "grounded", the user actually creates part of the backward chaining path to coordinate model execution and access to technical expertise. An exhaustive search needs to be made of the knowledge modules. The inference engines of most development tools are capable of operating under those conditions.

5.2.3 Grounded Feedback

Many forms of feedback are possible if information can be transferred via linking and satisficing relationships, and different grouping relationships can take place such as objectives, users, or system states. The same applies to decision aids. Summary of feedback or reformulation of the objective structures into a standard multicriteria technique are both possible. Many forms of prior weighting and interactive techniques are possible from the grounded context.

In an object-oriented knowledge representational scheme, a user may initiate feedback by focusing on part of the context such as one of the core issues or categories of concern. The user-focused goal is defined within the goal structure as an object. A query of the object's descendents (children, grandchildren, etc.) will reveal the extent of grounded facts that are relevant to the stakeholder. The user-defined relationship between each of the focused goal's descendents determines the relative magnitude of association with each grounded fact.

Grounded facts themselves, in a collaborative planning architecture, are objects. For quantitative grounded fact objects, a value may be required. Backward chaining through the physical system is the natural searching mechanism to find methods or rule sets that determine the object value. In the case of project_construction_cost, queries are made of each object in the physical system that has inherited the property *construction_cost*. If that data has not been made available, decision makers are prompted for missing values. Qualitative grounded objects require descriptive satisfaction of an object value. They will likely incorporate a larger knowledge module than quantitative objects to satisfy the same need because rules for every potential combination of circumstances is required to determine the qualitative variables.

The quality of feedback is dependent on the extent of knowledge modules, and subjective descriptions are limited by the available grounded facts. Relational links to goals defined by stakeholders help to point out weaknesses in the knowledge domain when the grounded facts are insufficient to describe priorities. Descriptions and explanations for each of grounded facts can be passed to the user. Feasible or recommended technical alternatives can also be made available.

Another form of feedback is made possible by the quantitative values of grounded objects and their relationships with broader goals. A stakeholder may assign an indication of importance to descriptive feedback, or relative goal satisfaction. Other uses include translation of selected portions of a goal structure for use with management decision aids. Multicriteria decision tools can be used in the preparation of alternative ranking. More complex use of management decision aids could relate multiple users.

The use of modelling tool kits, conceptualization of physical systems, management of decision maker objectives, and a grounded approach to relating impacts, is a convenient paradigm for providing feedback. It also allows experimentation with other areas of interest for planning decisions. One growing interest is in the generalization of model form for analysis of different types of problems (Laskey and Lehner, 1994; Wellman et al, 1992). This is essentially handled by plugging in different models and knowledge bases into the collaborative planning approach. Another growing interest is learning models to extract generalizations from collections of cases (Laskey and Lehner, 1994), usually applying a Bayesian approach. Growing databases of cases can be accessed within a collaborative planning environment, presented according to similarity statistics with the current case, and evaluated by decision makers. Existing bayesian assessment of similarity can also be implemented within the grounded layer of physical indicators.

In the development of a decision support module for the selection of evaluation criteria, the following theoretical principles are used:

- Each criteria can be decomposed into a set of issues (criteria develop from the inspection of relevant issues).
- Each issue can be described in terms of quantifiable grounded facts.
- Each stakeholder has a unique set of preference structures toward the relevant issues.

5.3 Modelling Stakeholder Interaction

The typical group decision problem has been presented by Szidarovszky et al (1986) in terms of a decision space, $\boldsymbol{R}$, and a consequence space, V. Calculation of a utility function for the j^{th} individual, v_j, is dependent on defining an individual's subjective probability density function, p_j, which represents the uncertainty by which consequences will occur for alternative $\boldsymbol{i}$. The problem then becomes a question of determining the aggregation rules for subjective probability distribution functions and utility functions. Consequences must be defined, as well as alternatives. Decision makers must supply the set of p_j and v_j. Most techniques circumvent this complicated task by asking simpler questions such as "Which alternative is preferred?" or "How do you rate this alternative?". In many cases, the consequence space is reduced to a number of valued criteria.

Szidarovszky defines a problem with 2 distinct and separate dimensions: decision space and consequence space. Can collaborative decisions be effective if the context of a problem is treated separately from the actual planning problem? The systems approach in the previous chapter, and the discussion of context management in the previous section, imply that collaborative processes would benefit if stakeholder interaction were motivated by an integrated decision model, where the context of a decision is an important element in framing alternatives. The following decision model is presented after Bender and Simonovic (1995).

5.3.1 Integrated Decision Model

Consider the decision space, $\boldsymbol{R}$, for a particular set of resources, X. $\boldsymbol{R}$ represents the entire realm of possibilities for managing a resource. A discrete solution within the decision space is denoted as a single system state, $\boldsymbol{i}$. System states can be described in terms of properties including temporal as well as spatial.

By initiating development in collaboration with a specific set of stakeholders, J, such that

$$J = \{j : j \in J\} \tag{24}$$

and a proponent isolates the decision space to a set of system states, I such that:

$$I = \{i : i \in I \subset R\} \tag{25}$$

The selection of a stakeholder group narrows the scope of possibilities to be considered. The tendency resulting from increasing J generally results in an increase in the set I, where I also represents the problem domain.

Searching for a socially desirable management decision for a resource requires the consideration of every state within R, which is always greater than the set I considered. If an absolutely complete set of stakeholders are accommodated within a planning framework, the set of system states, I, approaches the set R. This also assumes that a proponent is reduced to a role of stakeholder, allowing other proponents to participate. Generally, a large group of stakeholders results in complexity, potential conflicts, and high transaction costs (many externalities involved in assigning property rights of stakeholders incur transaction costs), assuming a solution can be found to reach consensus. For this reason, social management decisions should not consider R, but a considerably reduced decision space, I. This limits the size of the planning group. An optimal size of planning group is difficult to ascertain. There are tradeoffs in enlarging the available planning states, I, and also for reducing I towards only those states desirable to the proponent.

To determine the acceptability of a system state, i, each stakeholder, j, within the collaborative planning group, J, must be capable of accurately representing their views, goals, and preferences. To assess tradeoffs between system states, a set of knowledge must be made available which refers to the accumulated experience and current technologies in the problem domain, I.

The set of possible plans or system states, $\boldsymbol{I}$, is composed of 2 aspects, $\boldsymbol{I} = (\boldsymbol{I_p}, \boldsymbol{I_c})$. A system state is an arrangement of alternative components or strategies, and also the perspective by which a proposal is judged. $\boldsymbol{I_p}$ is the set of physical components, and management alternatives selected to comprise a proposal. $\boldsymbol{I_c}$ is the set of contextual variations used to frame proposals. $\boldsymbol{I_c}$ is made up of both objectives of stakeholders and the links used to define their interdependencies. A significant change in this definition of decision space, compared to standard descriptions, is the idea that the context of a decision is considered as part of the decision space, and a part of any proposal. A change in perceived context results in a completely unique solution.

The set of stakeholders, $\boldsymbol{J}$, provide the context or complete set of objectives, $\boldsymbol{O}$, as sets of objectives relevant to each stakeholder, $\boldsymbol{S_j}$, such that:

$$O = \{s : s \in S_j \subset O \subset I\} \tag{26}$$

Similarly, a link, $\boldsymbol{l}$, joins 2 objectives within the set of links for a stakeholder, $\boldsymbol{L_j}$, such that:

$$L = \{l : l \in L_j \subset L \subset I\}, O \cap L = 0 \tag{27}$$

Properties that describe system state can be calculated throughout the decision space in terms of the set of grounded facts, $\boldsymbol{G}$. $\boldsymbol{g}$ is a grounded fact, or set of grounded facts, such as the monetary cost of constructing a hydroelectric generating station, and $\boldsymbol{G}$ is the complete set of grounded facts including parameters related to fish passage.

For any system state, $\boldsymbol{i}$, there exists a valid set of advice, $\boldsymbol{A}$, for each stakeholder, $\boldsymbol{j}$, such that there is a (vector) set of advice for each stakeholder objective denoted by $\boldsymbol{A_{i,j}}$. Likewise, each set of advice, $\boldsymbol{A}$, is related to a set of available components, strategies or technologies, and belongs to the set of knowledge, $\boldsymbol{K}$, applicable to the problem domain, $\boldsymbol{I}$ such that:

$$A = \{a : a \in A \subset K\} \quad (28)$$

Expert system approaches may appropriately disseminate domain knowledge subject to both design circumstances and the context of an inquiry.

There also exists a value that describes the level of satisfaction of a stakeholder for any given objective, $V_{i,s}$, which is based on the set of grounded facts linked to the objective, s, by stakeholder, j, evaluated at state i. The assessment of V provides input to decision aids for evaluation of group planning progress.

A description of context for each stakeholder easily adapts to an object-oriented framework for managing goal hierarchies. An object-oriented goal structure defines goals as objects. Each context object is described in terms of its relationship with other context objects, using links between parent and child objects. Parents refer to "why" a child is important. Children generally refer to "how" a parent is satisfied or defined. In this way, a structure emerges that contains abstract high-order parents, and technical low-order children. On its own, though, context management has no direct access to results of technical choices on a physical resource system.

Links between objectives form parent-child relationships. L_j includes the following attributes: parent objective, child objective, and relationship between 2 objectives where p is the parent objective and c is the child objective as:

$$l = (s^p, s^c, \phi^p \langle c \rangle) \quad (29)$$

The sphere of influence for a stakeholder, in terms of objectives and links, defines a family of objectives. A decision maker may focus on a specific abstract objective to define a subproblem. A subproblem includes all grounded knowledge facts explicitly and implicitly linked to the objective at issue.

To enable feedback, and eventual use of decision aids, several mechanisms must be in place. Grounded facts must obtain values from analysis tools. Grounded facts must be able to define objectives. Child objectives must be able to pass values to parents. These three mechanisms are all necessary for evaluating relative alternative satisfaction. They may also be applied to numerous other group planning indicators. The process of creating alternatives and experimenting with options can be aided by an additional mechanism, the specification of relevant group advice.

Grounded facts obtain quantitative values from system properties from results of analysis of system properties. Unlike the object-oriented management of objectives, grounded facts are not governed by strict links. They are created, presumably, by knowledgeable experts. A *method* describes the data and calculations required to find the value of a grounded fact, either directly or by invoking analysis tools to prepare the required output.

The most basic of child objectives is a grounded fact. By linking a grounded fact to a stakeholder's objective, the units of a grounded value must be transformed to the amount of satisfaction that the stakeholder enjoys from the value of the grounded fact. This is accomplished by a simple satisficing relationship, such that:

$$V_{i,s} = \phi^{s}\langle g_i \rangle \tag{30}$$

To be able to define a satisficing relationship, upper and lower bounds are required for possible grounded fact values. The resulting link acts as a conversion from real values into contextual constructs. Using child objectives to define more abstract or strategic parent objectives involves the same form of satisficing as with grounded facts. However, in using satisficing as a measure of utility, upper and lower bounds are always *1.0* and *0.0* respectively.

Special attention must also be paid to the common case where multiple children belong to a common parent. There are many options for aggregating the satisfaction relationships from multiple children in the general form of:

$$V_{i,s} = \Phi^{c}[\phi^{s}\langle g_i \rangle] \tag{31}$$

Some examples of aggregation functions are: maximum, minimum, average, bounded sum, product, average difference, weighted average, etc. One candidate for a default aggregation mechanism is the product. Improvements to the worst child component of a parent will have the greatest overall improvement of satisfaction of the parent. Unfortunately, 3rd or 4th generation parents will have extremely low aggregate satisfaction. If the average is used, there is no information regarding the disparity between satisfaction of child objectives. The maximum is overly optimistic. The minimum is very conservative, although pessimism is more desirable than optimism for tracking satisfaction of many stakeholders. Ultimately, an aggregation mechanism must be chosen.

The above mechanisms allow stakeholders to determine their perceived satisfaction with an existing alternative. Changes to an alternative may be desirable or necessary. The event of choosing a means of adapting a proposal may be supplemented by supplying advice. The mechanism for supplying advice as a list of viable alternatives for each stakeholder, A_j, is based in boolean logic. It also encompasses forms of manipulation to sets of advice such as determining the intersection of every stakeholder's list of advice, such that a list of recommendations is defined by $\cap A$, within K (Figure 17). A less restrictive list of recommendations can be found by $\cup A$. By manipulating the list of advice for each stakeholder, it is also possible to rank options according to the perspective of individual stakeholders.

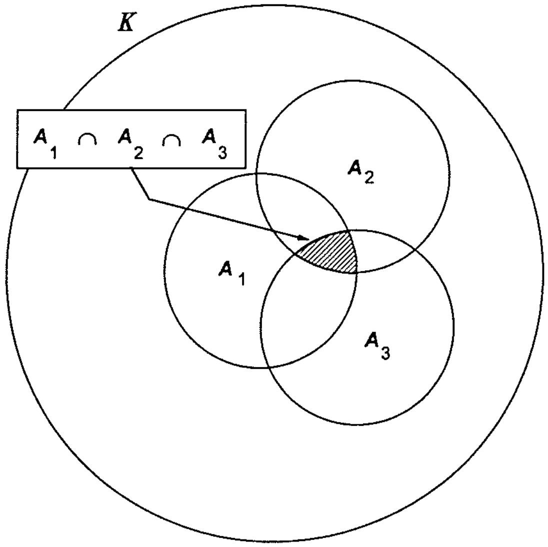

Figure 17. Common recommendations to a stakeholder group.

Recommendations (advice) provide a source of branching in the process of creating alternatives within the decision space, *I*. From each system state, there may exist several possible alternative measures or even several combinations of available options (Figure 18). Each forms an alternative branch which arrives at a different state. These alternatives can be chosen interactively, or generated automatically based on the perspectives of stakeholders.

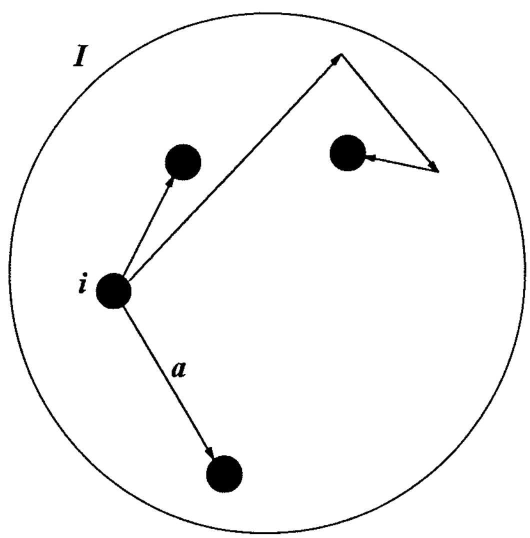

Figure 18. Using advice to create alternatives and move to other system states.

5.3.2 Summary of Stakeholder Interaction Model

The ability to aggregate links from grounded facts to any abstract objective allows additional evaluation of preferences, and subsequent aggregation of stakeholder preferences toward indicators of consensus. Some measures for alternatives by individuals or groups are: satisfaction, ranking, outranking relationships, concordance, discordance, inconsistency, reliability, resiliency, vulnerability, etc. Ranking can be cardinal or ordinal.

Comparisons can also be made between objectives of a single stakeholder for the same alternative, between alternatives for the same stakeholder objective, between stakeholders for a set of rankings of alternatives, etc. Aggregation of decision maker preferences is a complex task. There are numerous measures and techniques for standard problem types which assess a number of alternatives for each stakeholder's given set of criteria. There are also applications for multiple decision makers using game theory, but these are predominately adversarial in nature.

A general assumption for group decision aids is that selection automation is the desired purpose. There has been enough work in the areas of social choice theory, cognitive approaches, alternative dispute resolution, and other disciplines, to suggest the form of information and type of feedback to foster creative decision making. Decision support, using the collaborative planning approach described in this paper, allows flexible involvement of stakeholders in the conceptual design stage of a proposal by removing language constraints for nontechnical participants, providing feedback that is relevant to each stakeholder, and by providing a facility that is capable of encoding and disseminating domain knowledge.

The purpose of attempting to redefine the role of decision support for planning water projects is to effectively use preference information and perspectives of stakeholders to search for the most robust solution. Decision aids need to evaluate tradeoffs over both space and time, but also between stakeholders.

Only then will solutions be socially optimal. The approach presented here provides a way of examining priorities, communicating priorities, understanding each other, and confronting each other.

5.4 Decision Support Module for Selecting Evaluation Criteria

A decision support module for selecting evaluation criteria is a first attempt at developing an adequate mechanism for defining relevant criteria and relating those preferences to available expertise and analysis tools (Simonovic and Bender, 1996). A conscious attempt is made to integrate the context of the decision model to the selection of alternatives using grounded theory. Grounded facts act as a foundation toward understanding abstract core issues. Discussions develop from known facts rather than adversarial negotiating positions.

5.4.1 Architecture

The collaborative planning support system module for selecting project evaluation criteria assumes an architecture that emphasizes a cyclical use in which the end of processing is entirely dependent on the wishes of stakeholders. Figure 19 demonstrates the general process. There are four specific components, through which information passes. The stakeholders, a list of grounded system facts, a knowledge base (KB), and a potential list of evaluation criteria or objectives are the components developed in the module presented in this section.

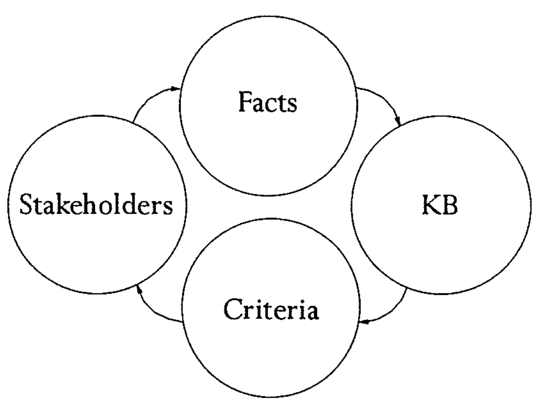

Figure 19. Processing in the Evaluation Criteria Selection module.

The first component of the evaluation criteria selection module allows multiple stakeholders to participate. They are expected to voluntarily participate in a cooperative, learning environment. Stakeholders supply information to the module describing their individual values or perception of important issues. They collectively drive the processes of the module by initiating information and by adapting their mental mapping of values.

The central component of this module is the use of grounded facts to relate stakeholder values and issues to information useful to the assessment of tradeoffs. Stakeholders supply their value information by selecting important features from the grounded facts list. Each individual stakeholder maintains a list of facts, and collectively, stakeholders are allowed to duplicate selections. In fact, duplication of facts in the collective list suggests areas of agreement between stakeholders.

A knowledge base is used to pass or filter the selected facts into the final component, which is a list of potential planning objectives. Once the selected lists of grounded facts are 'translated' into relevant, multidisciplinary planning objectives, the entire process may resume upon inspection by the stakeholders. Individuals may change their mind, or learn something from reviewing other facts lists, or gain insight from inspecting a suggested list of evaluation criteria.

5.4.2 Rulebase

The knowledge base used to relate grounded facts to management objectives is composed of rules generated, presumably, by experts from the relevant areas which compose the given problem domain (roughly defined by the proponent and the scope of the proposal). Since trust can be a major concern among stakeholders, who may traditionally be antagonists, preparation of this knowledge base in not a 'black box' type. An open forum for viewing the knowledge base, and the source of the rules, is made available.

Rules that trigger the relevance of an objective from a selected fact may be of several types, depending on the use of decision support. There are generally two classes of relationship between the facts and objectives. One is to trigger objectives if the fact is necessary for the evaluation of the objective. This, however, is more suitable to a summary of data requirements for an objective. The second, and more meaningful form of relationship between facts and objectives is based on impact. The impact relationship can be further divided to positive or negative impact relationships. The approach pursued for this experiment is a 'positive importance' or 'more is better' approach to each fact, and selection of objectives which will impact positively. An important objective may then be selected by considering whether it will help achieve a positive attitude for the selected fact.

The example application presented in this paper contains many simplifications, and may not accurately represent the relationships between criteria and facts. The purpose for this experiment is not to pretend that our knowledge is perfect. The form of knowledge base is not unlike many environmental assessment studies. Environmental assessments may attempt to prepare a large table of potential impacts to valued ecosystem components by specific components included in the proposal. In many cases the assessment agenda pursues a qualitative answer to each impact, either positive or negative, significant or insignificant. The application presented here is simplified, but it demonstrates the communication potential and use of a knowledge base of relationships within a decision support tool using a grounded approach.

A rule developed for a fact is a simple one-to-one relationship with an criteria. If a fact is selected, and there exists a simple rule that relates the fact to an criteria, then the criteria is considered to be somewhat important. A single fact may point to many criteria (using many rules), and many facts may point to a single criteria. This simple knowledge base can be managed as a relational database. A more complex arrangement may also be used in the traditional expert system format to provide detailed conditional knowledge concerning potential impact relationships. The knowledge incorporated in this example is provided in Table 4.

Table 4. Rules to trigger relevant criteria from a list of important facts.

Evaluation Criteria	Grounded Fact
MAX_SPECIES_habitat_suitability	SPECIES_range
MAX_SPECIES_habitat_suitability	SPECIES_LIFECYCLE_habitat
MAX_SPECIES_population	SPECIES_population
MAX_benefit_cost_ratio	BC_ratio
MAX_benefit_cost_ratio	STRUCTURE_lifespan
MAX_benefit_cost_ratio	inflation_rate
MAX_employment	employment_rate
MAX_energy_rel	RESERVOIR_volume
MAX_energy_rel	STRUCTURE_maximum_stage
MAX_energy_rel	STRUCTURE_minimum_stage
MAX_energy_rel	energy_capacity
MAX_energy_rel	energy_price
MAX_energy_rel	energy_supply_reliability
MAX_gross_national_product	inflation_rate
MAX_net_present_value	NPV
MAX_net_present_value	STRUCTURE_lifespan
MAX_net_present_value	discount_rate
MAX_operational_rel	STRUCTURE_lifespan
MAX_water_supply_rel	RESERVOIR_volume
MAX_water_supply_rel	STRUCTURE_minimum_stage
MAX_water_supply_rel	WATER_commercial_demand
MAX_water_supply_rel	WATER_domestic_demand
MAX_water_supply_rel	water_price
MAX_water_supply_rel	water_supply_reliability
MIN_FLOW_discharge_target_dev	FLOW_discharge
MIN_FLOW_discharge_var_target_dev	FLOW_discharge
MIN_FLOW_discharge_var_target_dev	FLOW_morphology
MIN_SPECIES_migratory_delay	SPECIES_LIFECYCLE_habitat
MIN_SPECIES_migratory_delay	SPECIES_population
MIN_SPECIES_migratory_delay	SPECIES_range
MIN_SPECIES_migratory_delay	STRUCTURE_fish_passage
MIN_STORAGE_flooded_area	STRUCTURE_maximum_stage
MIN_STORAGE_flooded_area	STRUCTURE_minimum_stage
MIN_STORAGE_flooded_area	land_COVER
MIN_STORAGE_flooded_area	land_USE
MIN_WATER_QUALITY_violations	WATER_domestic_demand
MIN_WATER_QUALITY_violations	WATER_quality
MIN_WATER_shoreline_erosion	FLOW_morphology
MIN_WATER_shoreline_erosion	FLOW_runoff_coefficient
MIN_WATER_shoreline_erosion	WATER_erodibility
MIN_WATER_stage_target_dev	RESERVOIR_volume
MIN_WATER_stage_target_dev	WATER_stage

MIN_construction_area	FLOW_morphology
MIN_construction_cost	construction_cost
MIN_energy_cost	STRUCTURE_maximum_stage
MIN_energy_cost	energy_price
MIN_flood_damage	population_density
MIN_flooded_agriculture	land_USE
MIN_flooded_archeological_sites	cultural_heritage
MIN_flooded_recreational_area	land_USE
MIN_flooded_vegetation	FLOW_runoff_coefficient
MIN_flooded_vegetation	land_COVER
MIN_flooded_wetlands	FLOW_runoff_coefficient
MIN_flooded_wetlands	land_COVER
MIN_health_risks	medical_capacity
MIN_length_of_flooded_streams	FLOW_morphology
MIN_length_of_flooded_streams	FLOW_runoff_coefficient
MIN_length_of_flooded_streams	SPECIES_LIFECYCLE_habitat
MIN_maintenance_cost	annual_benefits
MIN_maintenance_cost	maintenance_cost

The suggestion of relevant criteria is intended to be applied within a multicriteria framework for assessing tradeoffs between alternatives. Specification of criteria weights is necessary for the evaluation of noncommensurate criteria. The combination of selected facts by stakeholders can be used to imply weights for the objectives in many forms. A simple form of determining weights is demonstrated by this module for evaluation criteria. Much more complicated techniques, and potentially more representative techniques, are possible. The following equation represents a simple aggregation of the number of occurrences in which an objective has been triggered, normalized by the total number of rules triggered for all criteria. The resulting weight assumes that each selected fact carries the same level of importance to stakeholders, and that impact from a criteria on a fact is on the same scale.

$$w_I = \frac{\sum_j \sum_k \phi_j^I \cdot \theta_k^j}{\sum_i \sum_j \sum_k \phi_j^i \cdot \theta_k^j} \tag{32}$$

w_I weight for criteria ***I***

$\phi_j^i \equiv$ boolean reference by fact ***j*** to criteria ***i***

$\theta_j^i \equiv$ boolean reference by user ***k*** to fact ***j***

5.4.3 Learning process

The use of this collaborative planning support system module application is intended to serve a cooperative, voluntary, attempt to promote improved understanding and communication between stakeholders. Uncooperative games are not well served in this environment. In other words, for potentially uncooperative stakeholders, there are too many opportunities to manipulate the results. The selection of facts are not policed in any way, and the justification of a combination of facts (in a multiattribute form) has not been applied here.

The communication module is designed to promote learning among stakeholders by experimentation at the level of their value system. There are many applications of experimentation with technical scenarios. Andrews (1992) is one related example. The range of scenarios should be determined by the range of values that are brought to bear on the eventual decision. Determining the range of values can be served by 3 forms of learning by stakeholders. One is for individual stakeholders to clarify their own set of values as they relate to alternatives for decision criteria. Another form is learning about the values of other stakeholders. Finally, many stakeholders may benefit from learning about consequences of their value system to the process of alternative evaluation.

The result of using the support system module is not intended to 'handcuff' participants to a particular decision, or trap them into a certain definition of their values. If communication breaks down, the status quo of adversarial responses, political games, and legal actions still remains. The purpose is motivated by a need to circumvent many of these expensive forms of choosing a plan for developing a resource. Communication is the key in this application. Common understanding, and a potentially greater level of consensus is the desired result.

5.4.4 Illustrative Example

A prototype application has been developed for determining evaluation criteria using a grounded multiple stakeholder framework. A small UNIX-based software module was prepared in C using SmartElements™ by Neuron Data on a UNIX Workstation. Its functionality results from the 4 windows shown in Figure 20. They are the *CPSS* (which is an acronym for Collaborative Planning Support System), *Stakeholder Information*, *Stakeholder Facts*, and *Relevant Objectives* windows. Table 5 summarizes the functionality shown in Figure 20.

Table 5. Functionality description of Evaluation Criteria Selection module windows.

Window	Functionality
CPSS	open/close/save sessions, manage other windows
Stakeholder Information	add/delete stakeholders
	select stakeholder as current
Stakeholder Facts	add/delete important facts for a selected stakeholder
	view list of global facts (combined stakeholder list, number of instances of each)
Relevant Objectives	update list of relevant objectives, weighting
	summary of types of objectives

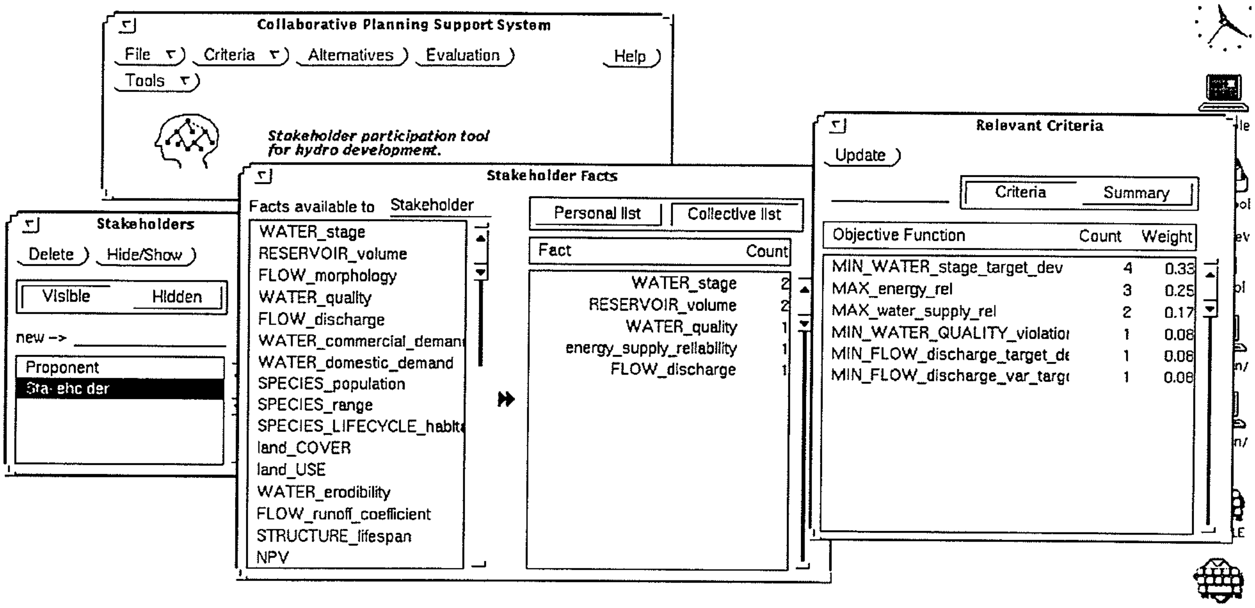

Figure 20. Collaborative session windows for the Evaluation Criteria Selection module.

A collaborative session is initiated by first defining the stakeholders involved in the Stakeholder Information window. The example in Figure 20 demonstrates a collaborative session between a proponent and a stakeholder. The proponent is initiating hydroelectric development of a river reach. The stakeholder represents an environmental regulatory agency.

Choosing available grounded facts by a selected stakeholder occurs from the left listbox within the *Stakeholder Facts* window. Many of the facts are independent of the case study, such as *energy_supply_reliability*. Other facts may refer to more than one physical object. For instance, *STRUCTURE_minimum_stage* refers to the minimum stage at hydraulic structures in the system. There may be more than one *STRUCTURE* (dam or weir, for example). Inclusion of another module for defining hydraulic systems of streams and lakes will allow selection of a specific water body. The *Personal Facts* listbox displays the facts chosen by the currently selected stakeholder. To the right is the *Global Facts* listbox which summarizes the collective list of facts chosen by all stakeholders, including the number of occurrences of each.

The example in Figure 20 shows that 5 different important system facts have been selected. The stakeholder, currently selected in Figure 20, is shown to have chosen 3 facts as important, referring to issues relevant to the environmental regulations stakeholder. The collective list of facts indicates that 2 of the facts are considered valuable by both the proponent and stakeholder. This area of common ground may be used to pursue more positive discussions about protecting or enhancing those aspects of the physical system. Another DSS module provides suggestions of technical options in the common areas of importance.

At any point during the selection of facts by stakeholders, the *Update* button on the *Relevant_Objectives* window can be depressed. This invokes the knowledge base to find all the relevant objectives, count the number of occurrences of each, and provide weights. The *Count* for each objective is the number of instances in which a selected fact triggered the identification of a relevant objective. A summary is also provided which classifies objectives as environmental, economic, or social - and aggregates the weights to give a general indication of bias toward a few traditionally adversarial agendas. Understanding the implications of all of the objectives may be difficult. Information provided by a summary, such as given, is more transparent even if it is not entirely accurate.

The example session divides objective weights mainly between minimizing deviations from stage target levels on water bodies and maximizing the reliability of energy supply or water supply. The knowledge base was able to suggest reasonable evaluation criteria with weights which also appear reasonable. In doing so, weights have been implicitly defined for each stakeholder, combined with preferences supplied by the grounded approach, producing an aggregation of appropriate criteria weighting from a potentially complex situation.

Criteria weights produced by the 2 participants appear to favour criteria related to environmental concerns. The social weighting appears very low. It may have been higher if a third participant were involved representing local residents or the government. In this case, water supply reliability is considered to be a social objective. In the knowledge base, energy reliability is provided as an economic criteria. As a criteria for evaluation, energy reliability could be reduced to a number of more specific objectives. Some could be viewed as economic criteria, while others may be considered social. In this way, energy reliability is not exclusively an economic liability, it has aspects related to social impact as well. The same can also be said of the objective concerned with water quality violations. To make those adjustments to the knowledge base, a small set of text files store all of the objective and fact definitions along with relationships between them.

Upon inspection of suggested evaluation criteria, stakeholders may review their own values, those of others, and ask questions by changing their list of facts. Several iterations of the example session produced a wide variety of criteria and weights in which environmental concerns were related by 30% to 70% of the weight and economic concerns ranged from 25% to 60% of the weight. As many as 12 different criteria were suggested, and criteria weighting became more evenly dispersed as more criteria were triggered by the knowledge base.

The stakeholders were able to explore the consequences of considering various facts as important, and observing the changes at the level in which tradeoff analysis tools are invoked. Experimentation may eventually settle on a consensus solution, although consensus is certainly not expected. The changing objective weights provide an opportunity to track the range of experimentation, and assess any trends during the learning process of stakeholders.

5.5 Summary

In water resources planning and management it has been recognized that successful completion of the process is directly related to the active involvement of affected parties and agencies, stakeholders. Their involvement is also essential because they carry the knowledge and experience necessary to arrive at effective alternatives. An object-oriented decision support system approach is implemented as an efficient tool for empowering stakeholders and providing support for collaborative water resources planning and management.

To facilitate the collaborative process, support is necessary to allow stakeholders express their preferences using their own language systems. A module for determining evaluation criteria uses grounded theory from the social sciences to build an objective structure which will represent the interests of all parties involved, for the purpose of fostering a consensus solution.

An illustrative example has been used to demonstrate the decision support module for determining evaluation criteria. Object-oriented development provides for easy use of the module, and integration with other DSS modules. Use of graphical tools is planned to enhance the presentation aspects of the module.

Chapter 6 Integration of GIS and ES Decision Support Tools in the Exploration of Alternatives

6.1 Introduction

Geographic Information Systems (GIS) are being used in almost all areas of research with spatial implications, and Expert Systems (ES) are an applied artificial intelligence technique which is commonly used. Their use in water resources planning and management is growing rapidly. The use of spatial analysis tools for early planning experimentation of hydrologic/hydraulic options is placed in the framework of a larger decision support system (DSS) framework. Integration of tools such as GIS and expert systems may benefit: the visualization of projects, the designation of field studies, and ultimately the inclusion of stakeholders in the planning process, as a development proponent pursues more effective means for stakeholder participation and conflict resolution.

Following the conceptual systems approach of managing feedback, the task of data management takes on a new role. System data, or physical data, includes:

- Description of problem domain.
- Characteristics or properties of region.
- Measurements from field studies (both included and missing).
- Model outputs.
- Technical options.
- Experience (with technical options, similar problem domain, site characteristics, etc.).

Experience, especially, is a key component to providing appropriate feedback to participants, although it may also be implicit in the organization of the other data.

Decision data is the complementary set of data to be recognized in the integration of tools. It tends to be more abstract, such as:

- Value systems.
- Technical background of participants.
- Preferences, opinions.

Management of this data can take on various forms, including relational databases. Some data are more appropriately stored in an object-oriented data management scheme, or within a spatial database. The different forms of data management offer unique benefits, based on implementation, but they are quite similar. Relational databases are organized into tables, records, and fields. Object-oriented databases are organized into classes, objects, and properties. One of the benefits of object-oriented databases is that they typically are able to access methods or models and take advantage of properties such as inheritance and polymorshism.

Access to models through the concept of attaching methods to a class of objects (polymorphism-like) allows seamless connection of translation models to convert from one unit to another, aggregation models to combine components into more abstract measures or indicators, and simulation models to investigate the behaviour of complex processes.

The following case study example uses object-oriented data management to define alternatives, connecting the technical option objects to GIS models.

6.2 Case Study Background

6.2.1 Site Information

A case study selected to apply integration of technologies for collaborative alternative generation at the proposed hydroelectric development site of Wuskwatim Lake, Manitoba. Wuskwatim Lake is on the Burntwood River system in northern Manitoba, west of the city of Thompson (Figure 21). Flow along the Wuskwatim reach of the Burntwood River is augmented by the diversion of water from South Indian Lake via the Rat River into Threepoint Lake (upstream of Wuskwatim). Proposed development sites in this area include Wuskwatim Lake at Taskinigup Falls, at Early Morning Rapids on the Burntwood (immediately upstream of Wuskwatim Lake), and at the Notigi control structure (upstream of Threepoint Lake). Two communities may be directly affected by development. Thompson is a city with a population of 14,000 people that live downstream of any development in the region. Nelson House is a First Nation community (population 1,500), upstream of Wuskwatim Lake on Footprint Lake near Threepoint Lake. They live in potentially flooded areas. Manitoba Hydro has identified the area as having a generating capacity of 360 MW of power (Manitoba Hydro, 1987).

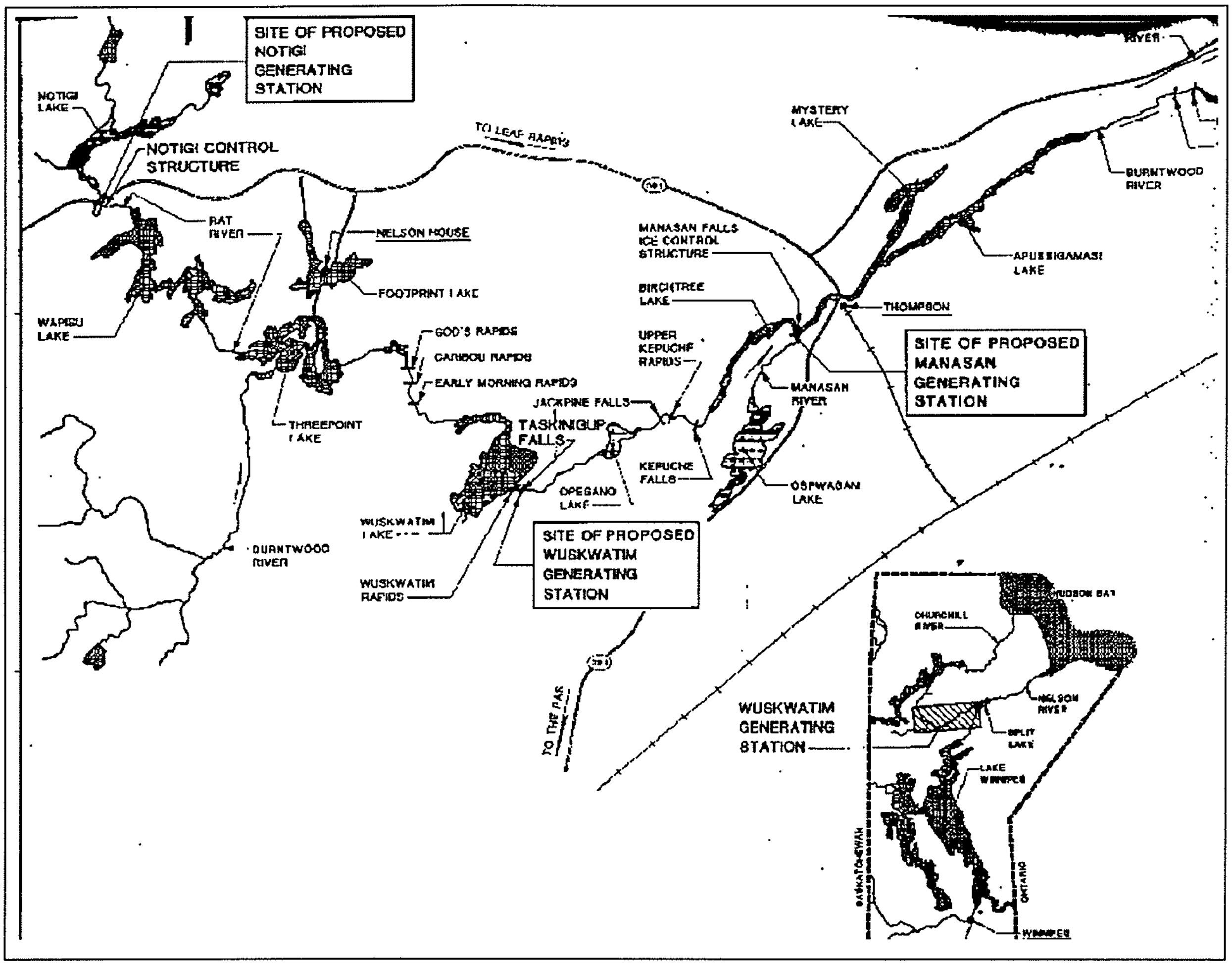

Figure 21. Case study area.

6.2.2 Development Proponent

Manitoba Hydro is an electric utility in the province of Manitoba. It manages, as a crown corporation of the province, a large system of regulated reservoirs, hydro-electric generating stations, thermal generating stations, transmission links throughout the province, and external transmission links to Manitoba. Manitoba Hydro operates from its mandate outlined in the Manitoba Hydro Act, which states:

> *The intent, purpose, and object of this Act is to provide for the continuance of a supply of power adequate for the needs of the province, and to promote economy and efficiency in the generation, distribution, supply, and use of power. (Manitoba Hydro, 1989)*

Using this Act as a guideline, Manitoba Hydro evaluates the energy needs of Manitoba in terms of consumer demand, and assesses the efficiency in which a reliable supply of energy is supplied. The achievement of Manitoba Hydro's mission, in the fulfillment of the Act, is described as the pursuit of several strategic objectives (Manitoba Hydro, 1989):

1 To provide a safe, adequate, economical and reliable supply of electricity to meet customer requirements.
2 To provide all customers with excellent service with particular focus on individual customer satisfaction.
3 To promote conservation of electricity when it can be achieved more economically than supply.
4 To develop and maintain a workforce with a high level of motivation, productivity and job satisfaction.
5 To improve productivity and quality in all segments of the business on a continuing basis.
6 To be recognized as a good corporate citizen which deals sensitively and fairly with the effects of its activities on communities and individuals.
7 To conduct all corporate activities in accordance with the principles of sustainable development.
8 To assure the Corporation's long-term financial integrity.
9 To secure beneficial extra-provincial agreements.

All activities of Manitoba Hydro may be described in terms of their role in satisfying one or more of these strategic objectives. The scope of this research is primarily concerned with strategic objectives 1,6,7,8, and potentially 9. Strategic objectives 1,8,9 are relatively straightforward to comprehend and pursue. However, objectives 6,7 are extremely subjective. Without stakeholder participation in making choices that affect these objectives, Manitoba Hydro can only guess whether they made the proper choices before moving through a licensing process.

Manitoba Hydro has prepared 2 initial design alternatives for the Wuskwatim Lake area. One option is to fully develop Wuskwatim with a high dam at Taskinigup Falls. Another option is to develop 2 low head generating stations, one at Taskinigup Falls, the other upstream of Wuskwatim Lake at Early Morning Rapids on the Burntwood River. A final design has not been chosen.

Manitoba Hydro would like to involve various stakeholders in the planning of environmentally sensitive features of development such as:

- Generating station option.

- Reservoir elevation.
- Operating mode for the generating station.
- Forebay clearing.
- Location of the permanent access road to the project site.
- Location of Birchtree station.
- Location of transmission lines.
- Mitigation, compensation, and enhancement programs.
- Monitoring.

6.2.3 Identification of Stakeholders

Manitoba Hydro has historically chosen to generate electricity primarily from the flow of water instead of using other sources of power such as nuclear power, or fossil fuels. The province of Manitoba is rich in hydroelectric potential and is sparsely-populated in many areas. Some of North America's largest lakes exist in Manitoba, and the Nelson River drains a large portion of North America into Hudson's Bay in Manitoba's north. Most of the generating capacity is in northern Manitoba where there are few people and many natural resources. Mining and forestry are the major industries of the region. Many areas are pristine wilderness and many communities have subsistence economies that are dependent on local hunting and fishing. Some generating capacity is already realized in northern Manitoba. A significant project is the Churchill River diversion which diverts water from the Churchill River system to existing generating stations in the Nelson River basin. Another proposed project which will not be built in the near future is the Conawapa generating station on the Nelson River.

There are several treaties and agreements in place to regulate the development of northern Manitoba water resources for hydro power. The most significant agreement, in terms of relevance for this case study, is the Northern Flood Agreement. It specifies constraints on development, with particular interest in South Indian Lake and the Churchill River diversion through the Rat River and Burntwood River systems to the Nelson River. The Northern Flood Agreement is a contract between the Government of Manitoba, Manitoba Hydro, and First Nations communities in the north. It includes Wuskwatim Lake and areas upstream and downstream of proposed hydro development in the area.

There are many potential planning participants identified as stakeholders for development near Wuskwatim Lake. They include the city of Thompson (downstream of Wuskwatim Lake), Nelson House First Nation (upstream of Wuskwatim Lake), and the Department of Fisheries & Oceans as a regulatory agency for fisheries interests. Any development near Wuskwatim Lake may impact the flow regime, water quality, and many geomorphological characteristics near Thompson. Nelson House may be subject to either direct flooding or increased water levels from backwater effects. Impacts associated with flooding may also affect Nelson House such as erosion and water quality problems.

The Department of Fisheries & Oceans, in an effort to address fisheries concerns, may consider impacts in terms of reservoir habitat, riverine habitat, and fish passage. Reservoir habitat may be altered from previous reservoir habitat and/or created from traditionally riverine habitat. Riverine habitat both upstream and downstream may be impacted. Most changes are assumed to occur downstream of the development site, but altered flow characteristics upstream of the site are caused by backwater effects.

Obstruction to fish migrations, to either upstream or downstream movements, may alter local populations of fish. Some species may disappear, while others may dominate. Changes or disruptions in species composition may alter ecosystem links. Changes in fish population may also impact local commercial and recreational fisheries at Nelson House or Thompson.

6.2.4 Project Licensing

There are also outstanding issues to be resolved between Manitoba Hydro and Nelson House First Nation, related to the Churchill diversion project which augments flow past Wuskwatim Lake. This complicates an already complicated procedure for project licensing. Presently, the federal environmental assessment and review process of pursuing development of a hydroelectric generating station can be described in 9 steps (FEARO, 1986):

1. Submission of a proposal, listing potential environmental issues and stakeholders (*a priori* environmental assessment investigations are encouraged and quickly becoming mandatory).
2. Screening of proposals to determine the need to mitigate environmental impacts or to modify the proposal.
3. Further investigation. Projects which pass screening may need further clarification of impacts before public hearings.
4. Referral to the Minister of the Environment for panel review.
5. Preparation of an environmental impact statement.
6. Public hearings on the environmental impact statement.
7. Report on proposal impacts and recommendations to address impacts.
8. Publication of report.
9. Licensing decision by the Minister of the Environment.

6.3 Integration of GIS Tools

In an attempt to avoid conflicts with stakeholders through the project licensing process, a collaborative planning process can be implemented to include relevant participants in the conceptual design stage. Decision support tools to experiment with different technical options can be a powerful visualization and knowledge transfer tool. GIS, as a viable and popular spatial analysis tool, is well-suited to be integrated with hydraulic and hydrologic processes.

6.3.1 GIS Database

Two digital NTS maps were selected for use in GIS applications (they are 63o09, and 63o10). They are 1:50000 scale UTM grid maps, in zone 14, using the GRS80 ellipsoid. The maps are adjacent to each other. Each map is approximately 30km x 30km. Map 63o10 contains areas upstream of Wuskwatim Lake, but not Notigi control structure or the Nelson House community. It also contains the majority of Wuskwatim Lake and the Rat River release point. Map 63o09 contains a portion of Wuskwatim Lake, and downstream areas of the Rat River, although not as far as Thompson.

A digital elevation model (DEM) has been developed from contour lines, a small set of available point elevation values, and known lake levels for some of the larger lakes. The accuracy of the DEM is not questioned at this point. It is discretized at 1m (vertical scale) intervals for 30m by 30m cell sizes, and is meant to be representative overall.

Other data in the database includes boundaries between land and surface water areas, wetlands, streams, rapids, and roads. A number of structures have also been digitized for possible inclusion in flooding experiments, including both the proposed Wuskwatim and Early Morning generating stations.

6.3.2 Flood Inundation Visualization

GIS exploration of flooding scenarios is one aspect of visual demonstration that may contribute to improved participation and understanding between various stakeholders.

The task of flood inundation is a complex task if hydraulic behaviour such as backwater effects are taken into account. Unfortunately, the determination of backwater demands a substantial amount of data. Backwaters are usually generated by the standard step method, using cross-section data for each reach. A typical procedure calculates the effect with external models, and simply displays the results using GIS. That procedure works fine for a river basin where the flooding is mainly on the flood plain of the river. For cases where a flood will inundate a variety of areas and land types, the cross-section data requirements become expensive and unmanageable.

For the purpose of visualizing a flooding scenario, especially for a large case study region (over 1000 km^2 in this case), it is relatively simple and straightforward to generate a flood without backwater. The results will not be completely accurate, but will be representative.

To generate a flood, the following procedure is used:

1. Combine the selected hydraulic structures (dams) as bitmap images with the digital elevation model (DEM). Hydraulic structures are treated as an area with a specified elevation.
2. Identify the upstream side of the hydraulic structure.
3. Specify an elevation for flooding.
4. Generate clumps of areas below the flood level.
5. Choose the appropriate clump as the reservoir.
6. Change the DEM and topographic maps appropriately.

A graphical interface has been developed in OpenWindows using SmartElements from Neuron Data to allow experimentation with different flooding scenarios. Structures such as dams or dykes can be added and removed. Reservoir levels can be adjusted. The size of the flooded area, and the added storage volume are also calculated.

6.3.3 Development of Alternatives

Generally, the experimental process of developing alternatives is iterative according to the sequence below:

1 Choose technical options (such as dam, reservoir stage).
2 Update model analysis.
3 Present results (save alternative).
4 Return to 1.

The following figure (Figure 22) shows the results of selecting technical options updating the model analysis, and presenting the results for a possible design proposal. The example in Figure 22 shows the interactive selection of 1 dam icon and 2 dyke icons on a small picture of the case study area. The selected dam location, Wuskwatim (at Taskinigup falls), is then set to a reservoir stage of 240m by the participants. This simple input defines the basic requirements for a technical alternative. Not visible in Figure 22 is the alternate approach of selecting from a previously-defined list of alternatives (remember Manitoba Hydro may already have conceptual designs being considered).

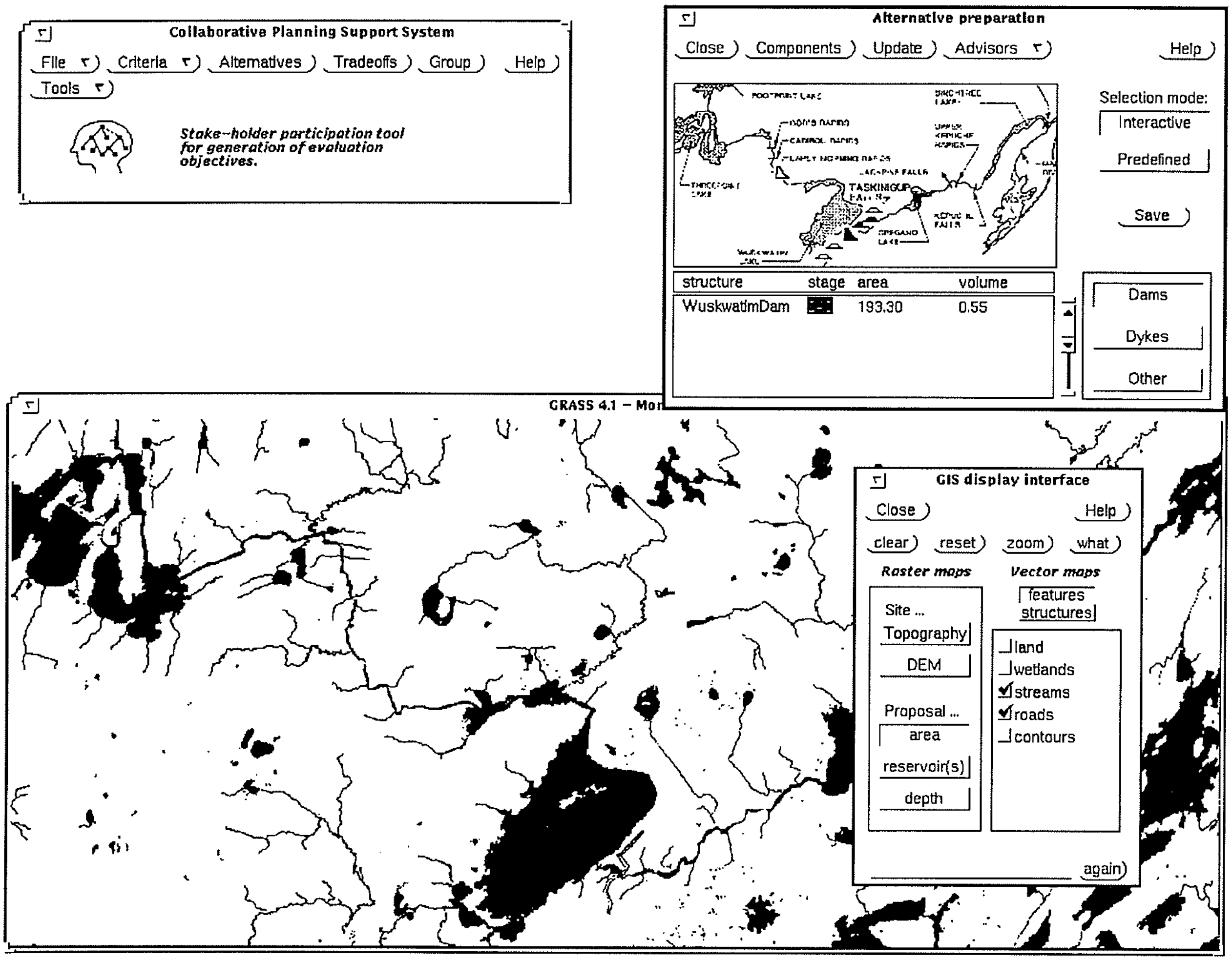

Figure 22. Example display for alternative generation decision support.

An update (clicking the *Update* button atop the window in the top right corner of the display) triggers the object-oriented database to collect the selected technical options, and submit them to relational database tables. The necessary GIS analysis tools are invoked, providing updates in the form of GIS maps of the flooded region. Other properties are also calculated, such as reservoir area (193.3 km^2) and reservoir volume (0.55 km^3).

The GIS display has also been automated with a custom interface. Original topography or the DEM can be displayed at any time to compare with the current flooded scenario. The new topographic area, the reservoir area, and reservoir depth can be shown. Other vector features and structures are also made available. For instance, streams are stored in vector format.

In this way, participants are able to interactively experiment with technical options, and view output of model analysis. The motivation is for participants with diverse backgrounds to understand the implications of different choices. The learning process is augmented by the visualization tools, and also by the interactive nature of experimentation. A new alternative can be updated within a couple of minutes. Participants are then able to see, in (near) real-time, how different technical options behave.

The form of decision support is very specific to stakeholder participation. It is also possible to generate a large number of scenarios to cover the likely range of alternatives to consider. From that database of generated alternatives, tradeoffs can be assessed and a selection made. However, in an automated generation of alternatives, there is typically one element missing. Facilitating creativity from the participants is the primary motivation of using an experimental learning process. In fact, it is the ultimate goal of any decision support system!

The selection of technical options shown in this example in no way reflects the position of Manitoba Hydro. Manitoba Hydro is interested in the creative contribution of stakeholders. Predefined alternatives may be under consideration, but they have not been presented in any detail in this document.

6.4 Application of Expert Systems

6.4.1 Prototype Expert System for Choosing the Design of a Hydroelectric Generating Station

As an example expert system (ES), a prototype hydropower development construction planning expert system has been developed. The hydropower construction ES encodes some basic hydropower design engineering experience at Manitoba Hydro, from a cooperative expert: Per Stokke, P.Eng. The purpose of the ES is to suggest a technical option such as a dam, along with its various components such as reservoir and powerhouse, and provide expert advice as to the type of dam and potential improvements that might be required such as water energy dissipation requirements, reservoir operating policy, and water intake positioning.

If a dam is to be created, an object is created within the Dam class, inheriting all the properties and behaviour associated with a dam. In turn, 4 components are also created as subobjects to the dam. They are:

- Reservoir.
- Spillway.
- Powerhouse.
- Release.

Each of these subobjects are in turn attached to relevant classes. For example, the spillway belongs to a class of objects called *Spillways*. The new spillway, in turn, inherits the properties and behaviour associated with spillways. In this way, an object-oriented model is built to describe the relationships between the dam and its surroundings. Other, nonstructural objects can also be associated with the dam.

6.4.2 Knowledge Base Description

The rule base of the expert system attempts to specify many of the design elements of the dam. For instance, a dam may be earth fill or rock fill if an embankment type of dam is chosen. An example rule is:

```
IF     the dam is an embankment type AND
       site excavation rock is not available AND
       a site borrow area is easily accessible
THEN   design the dam as earth filled
WHY    earth fill cost is low due to accessibility, compared to quarrying rock
```

In order to assign "earth fill" to the embankment type of dam, however, we must ensure that embankment is chosen or at least feasible. Backward chaining is used by the inference engine to search for rules to assign the dam to the embankment class of dams, such as the following:

```
IF     there are no frost concerns AND
       the experience of the planners has been with embankment dams AND
       the cost of earth fill (borrow material) is low
THEN   recommend an embankment type of dam
WHY    embankment dams are feasible (cost of earth fill) and preferred
```

Other rules are used to determine the relative cost of earth or rock fill for embankment dams. Likewise, rules attempt to determine properties and design requirements for the dam subobjects (*reservoir*, *release*, *spillway*, *powerhouse*), for instance:

```
IF     the experience of planners has been with either/both overflow and
orifice spillways AND
       the potential siltation in the reservoir is not high
THEN   recommend an overflow type of spillway
WHY    experience has been with overflow spillways, and flushing of sediment is
not a factor


IF     the available hydraulic head to the powerhouse is less than 25m
THEN   recommend a close couple type of powerhouse
WHY    close couple systems work well for low head stations


IF     the available hydraulic head to the powerhouse is less than 15m AND
       the powerhouse turbine unit capacity is less than 65MW
THEN   recommend a bulb turbine design
WHY    both head and turbine capacity are relatively low
```

A complete listing of the knowledge base is provided as an appendix.

Figure 23 shows the results of a consultation with the expert system through the CPSS interface. There are 2 active windows. The left window displays the recommended properties for design of the dam and hydroelectric generating station. Radio buttons provide access to properties of the different aspects of design. The right window is the *Session Control* window. Relevant questions are posed by the expert system. Subsequent recommendations are documented to the left in the *Property Display* window. Figures 23 and 24 show some of the recommendations for design based on an example consultation.

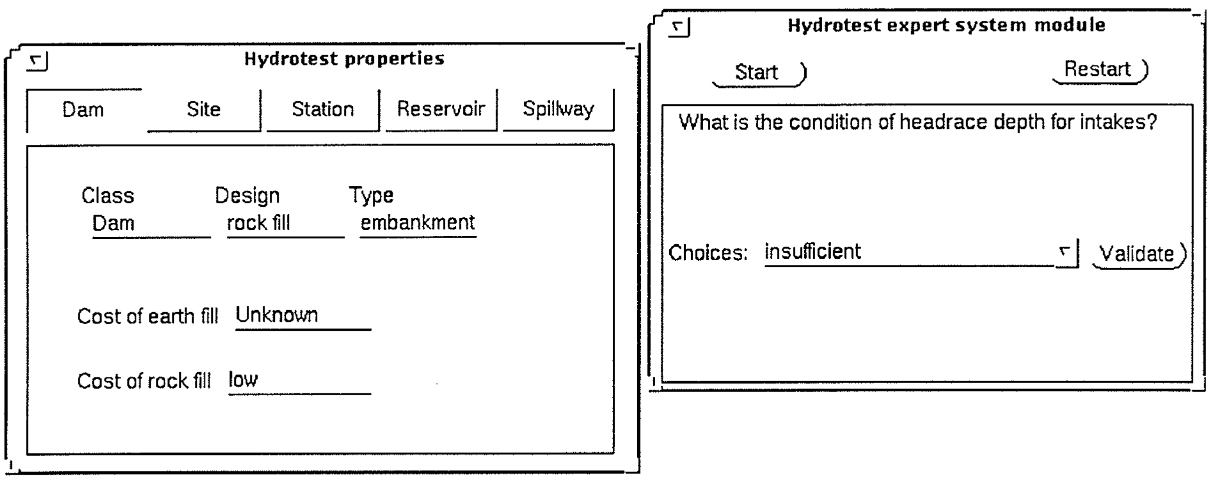

Figure 23. Expert system module interface.

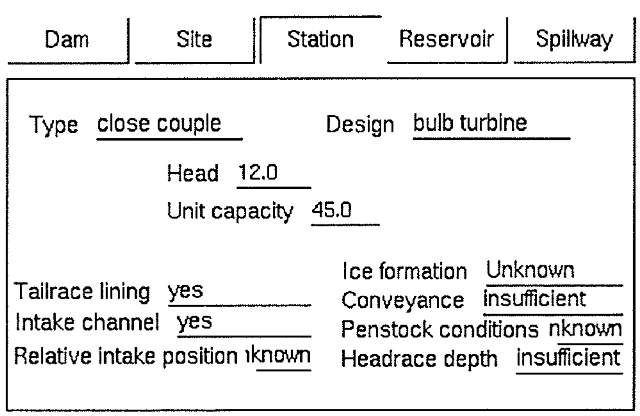

Figure 24. Example recommendation for a generating station design.

The hydropower construction expert system provides an example for the type of experience which can be provided by expert systems within a DSS. It is a sample utility, available for the specific (conceptual) design of technical options. Expert systems do not replace experience, but provide consistency and accessibility to knowledge. They may also provide decision making participants with the tools to generate realistic alternatives without being experts in multiple disciplines.

6.5 Summary

Geographic information systems and expert systems have many potential applications in water resources planning and management. A few of these have been touched on throughout this chapter. Of the many uses of DEMs and other topographic information, experimentation with flooding scenarios for the visualization of dyke requirements, or environmental impact assessment, is one simple yet powerful use of GIS in water resources planning. Another benefit to collaborative group planning techniques is the form of expert advice which enables detailed conceptual alternatives to be defined, including indications of cost and performance. The small knowledge base provided is an initial attempt to provide that service.

Chapter 7 Evaluation of Alternatives using a Fuzzy Compromise Approach

Multicriteria decision-making (MCDM), as a specialized field of Operations Research (OR), has been moving from optimization methods to more interactive decision support tools. Some of the areas of current and future development in the field have been identified by Dyer et. al. (1992). They include:

> *Sensitivity analysis and the incorporation of vague or imprecise judgements of preferences and/or probabilities in multiattribute situations and decisions under uncertainty in which states are multidimensional.*
>
> *Development of improved interactive software for multicriterion decision support systems, taking into account the findings of psychological research about biases and heuristics.*

The following chapter introduces a MCDM technique to evaluate the performance of discrete alternatives with uncertainties modelled as imprecise and vague. The fuzzy compromise approach is an attempt to address many of the lacking qualities in many MCDM techniques, where uncertainties and subjectivity are concerned.

7.1 Introduction

7.1.1 Compromise Programming

Multiobjective decision problems generally assume convexity of a decision frontier consisting of alternatives which are not entirely dominated by any other alternative. Figure 25 demonstrates a nondominated frontier in the decision space of 2 objectives for a maximization problem. For discrete alternative selection, the frontier surface reduces to a set of points, each representing an alternative (such as $\{A,B,C,D\}$ in Figure 25). The ideal point, where all objectives are able to achieve their greatest measure, is usually infeasible. The problem reduces to an evaluation of tradeoffs between efficient solutions.

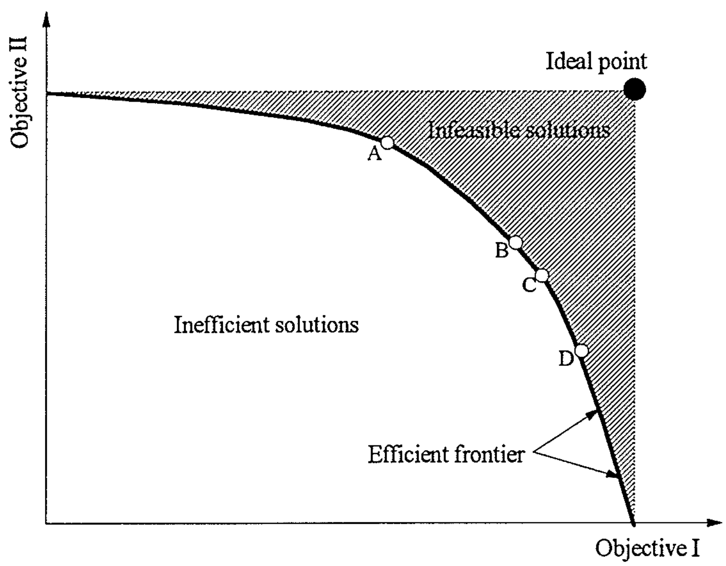

Figure 25. Multiobjective decision problems.

The concept of the displaced ideal was used by Zeleny (1973, 1982) to form compromise programming, a multiobjective technique which resolves multiple objectives into commensurable, unitless, distance metrics measured from an ideal point. In the discrete form of compromise programming, distance metrics can be calculated for each alternative - given an importance weight for each selection criteria. The result is a direct ranking of alternatives, valid for the selected weights and the chosen form of distance measurement. The following can be used to calculate a discrete compromise programming distance metric (L), otherwise known as the Minkowski distance:

$$L = \left[\Sigma_i \left\{ w_i^p \left(\frac{f_i^* - f_i}{f_i^* - f_i^-} \right)^p \right\} \right]^{\frac{1}{p}} \tag{33}$$

f_i is the value for criteria i, and f_i^*, f_i^- are the positive and negative ideal values for criteria i, respectively, where the term (f_i^*-f_i^-) serves to normalize the noncommensurate values of the different criteria. The weight, w_i, indicates relative importance of a criteria.

The distance from an ideal solution, L, is a function of the distance metric exponent, p. Typically, the Euclidean distance (p=2) is used to penalize large deviations from the ideal. However, the exponent can also carry an economic interpretation. The Hamming distance (p=1) results in a case of perfect compensation between criteria. For the Chebychev distance (p=∞), there is no compensation among criteria - the largest deviation from the ideal dominates the assessment.

Weights can be considered as the degree of importance or relevance. Although the use of weights may suggest tradeoff implications between criteria, practical applications in compromise programming use weights simply to place emphasis on important criteria (for example, see Simonovic, 1989). Subjective weighting may be an inefficient means of resolving commensurability issues, but it is a simple and interactive approach which cannot be proven invalid.

Weighting may or may not hold for the condition $\Sigma w_i = 1$. A useful rule of thumb, though, is to allow a range of criteria weights up to 1 order of magnitude. Another useful rule is to normalize the weights to a range [0,1], since the other term in the distance metric is also normalized to [0,1]. Although there is no formal proof for determining the benefits of these rules, they reduce the possibility of undue bias in the distance metric resulting from overcompensation. For example, if weights have a valid range [0,100] while the normalized criteria value differences (the second term in L) act on a range of [0,1], then the distance metrics would be dominated by the subjective weights.

Many of the traditional MCDM techniques, including compromise programming, attempt to preserve some level of transparency to problems. This is a valuable strength for decision makers. However, compromise programming (like most MCDM techniques) only makes use of a limited amount of information. Extensive sensitivity analysis is necessary to recommend any kind of recommendation with confidence. The marriage of a transparent technique such as compromise programming with fuzzy sets is an example of a hybrid decision making tool available to future planners.

7.1.2 Modelling Uncertainty

Uncertainty is a source of complexity in decision making which can be found in many forms. Typical types of uncertainty include uncertainty in model asumptions, and uncertainty in data or parameter values. There may also be uncertainty in the interpretation of results. While some uncertainties can be modelled as stochastic variables in a simulation, other forms of uncertainty may simply be vague or imprecise.

Traditional techniques for evaluating discrete alternatives such as ELECTRE (Benayoun et. al., 1966), AHP (Saaty, 1980), Compromise Programming (Zeleny, 1973; Zeleny, 1982), and others do not normally consider uncertainties involved in procuring criteria values. AHP inherently includes linguistic subjectivity, and has been applied to water resources problems (Palmer and Lund, 1985; Lund and Palmer, 1986).

Sensitivity analysis can be used to express decision maker uncertainty (such as uncertain preferences and ignorance), but this form of sensitivity analysis can be inadequate at expressing decision complexity. There have been efforts to extend traditional techniques, such as PROTRADE (Goicoechea et. al., 1982), which could be described as a stochastic compromise programming technique. A remaining problem is that not all uncertainties easily fit the probabilistic classification.

The theory of fuzzy sets, which is a theory of possibility, is not dissimilar to probability theory. In fact, they can be considered complementary. Fuzzy membership functions have a similar appearance to probability distribution functions (*pdf*). However, there are some inherent differences. A *pdf* provides the probability of specific values occurring. A fuzzy membership function acknowledges that we may not be completely sure what values we are talking about. Statistical precision can be independent of our classification of an event. For example, we may predict 90% probability of the occurence of a *good* value. What qualifies as a *good* value? Qualification of *good* can be subjective. Also, in many practical applications, there is not enough data to make probabilistic predictions with confidence. The dependence of stochastic applications on distribution functions can be restricting and misleading because of the intensity of data requirements. The difference between fuzzy and probabilistic functions is not always so clear. A fuzzy membership function may be used in place of a *pdf*, but the same data requirements are still relevant. In general, fuzzy sets provide an intuitive, and flexible framework for interactively exploring a problem that is either ill-defined or has limited available data.

There are typically 3 main forms of imprecision identified in fuzzy decision making (Ribeiro et. al., 1995):

- Incompleteness, such as insufficient data
- Fuzziness, where precise concepts are difficult to define
- Illusion of validity, such as detection of erroneous outputs (Tversky and Kahneman, 1990)

7.1.3 Fuzzy Decision-Making

The use of fuzzy representation of systems must also be considered for planning decisions involving multiple objectives with noncommensurate units and subjective definitions or impacts. Fuzzy set theory is almost 3 decades old. First introduced theoretically by Zadeh (1965), fuzzy sets gained a practical application method for system optimization problems when Bellman and Zadeh (1970) introduced the minimum operator. This enabled transformation and solution of equivalent nonfuzzy problems with linear programming. Although applications have since spread to nonlinear problems and multiobjective formulations, the basic solution concepts are still predominately based on the minimum operator. Other researchers have applied fuzzy sets within branch and bound search techniques to solve nonlinear problems. Surveys and reviews of fuzzy programming techniques can be found in Kandel (1986), Kacprzyk and Orlovski (1987), and Slowinski and Teghem (1990).

Fuzzy system descriptions have been applied in water resources planning decisions (Haimes, 1977; Slowinski, 1986), mostly for water supply planning problems including network problems. Fuzzy approaches attract a lot of attention in water resource management first of all because of uncertainties in discrete decisions that are affected by continuously variable inputs, but also because of empirical and poorly-defined goals for water supply, water quality, or other indirect measures such as recreational accessibility. Water supply problems entice fuzzy applications to be combined within multiobjective decisions for expert system decision support (Bardossy and Duckstein, 1992). Zimmerman (1987) presents frameworks and applications of decision-making with expert systems in a fuzzy environment. Contemporary research involving fuzzy systems are now exploring the incorporation of neural networks to solve complex fuzzy multiobjective problems (Sakawa, 1993) as one potential solution alternative to branch-and-bound methods. This example application is part of a research effort into multiobjective decision possibilities (Sakawa, 1993) and includes decision support applications to interactively explore solutions.

Fuzzy decision making techniques have addressed some uncertainties, such as the vagueness and conflict of preferences common in group decision making (Blin, 1974; Siskos, 1982; Seo and Sakawa, 1985; Felix, 1994; and others), and at least one effort has been made to combine decision problems with both stochastic and fuzzy components (Munda et al, 1995). Application, however, demands some level of intuitiveness for the decision makers, and encourages interaction or experimentation such as that found in Nishizaki and Seo (1994). Authors such as Leung (1982) and many others have explored fuzzy decision making environments. This is not always so intuitive to many people involved in practical decisions because the decision space may be some abstract measure of fuzziness, instead of a tangible measure of alternative performance. The alternatives to be evaluated are rarely fuzzy. Their performance is fuzzy. In other words, a fuzzy decision making environment may not be as generically-relevant as a fuzzy evaluation of a decision making problem.

Most fuzzy multicriteria methods either concentrate on multiobjective linear programming techniques, or experiment with methods based on fuzzy relations. Carlsson and Fuller (1996) provide a review of fuzzy multiple criteria decision making, and Ribeiro (1996) provides a very good review of fuzzy sets as they are applied to MCDM.

An intuitive, and relatively interactive, decision tool for discrete alternative selection, under various forms of uncertainty, would be a valuable tool in decision making - especially for applications with groups of decision makers. This chapter explores the application of fuzzy sets in conjunction with a standard MCDM technique, compromise programming. The adaptation of standard techniques to perform within the fuzzy framework demands a different set of operators. The following section describes techniques for making the necessary fuzzy arithmetic calculations when decision information is vague or imprecise. Compromise programming, as a crisp MCDM tool, is described. The application and use of fuzzy distance metrics are then developed and demonstrated as a fuzzy decision making tool.

7.1.4 Existing Fuzzy Applications using Displaced Ideals

The concept of a fuzzy displaced ideal was probably born with the comment by Carlsson (1982):

> *Zeleny's theory of the displaced ideal would ... be very useful in a fuzzy adaptation.*

Leung (1982) used the fuzzy ideal concept in multicriteria conflict resolution. Leung defines a fuzzy ideal solution, generates a membership function for each alternative (based on relative satisfaction or closeness to the ideal) and ranks alternatives based on the relative closeness to the ideal using distance metrics. In Leung's method, no weights are used, and the decision space is not defined by the criteria values, it is defined by the fuzzy membership (relative satisfaction) values. For this to occur, fuzzy sets representing level of satisfaction must be used to translate the criteria values. In order to accommodate conflict resolution, the decision space is treated as continuous - connecting the discrete (fuzzy) alternatives and searching for a location with the shortest distance to the fuzzy ideal.

Lai et al (1994) used distance metrics and the concept of a displaced ideal to reduce a multiobjective problem to a 2 objective problem. They are to:

- minimize the distance to an ideal solution
- maximize the distance to the worst solution.

Membership functions are assigned to the ideal and worst solutions to fuzzify the problem, weights are used to resolve the 2 remaining objectives. Decisions are reached by formulating the problem as a fuzzy linear programming problem, and solved in the standard Bellman and Zadeh (1970) approach.

An example of fuzzy compromise decision making can be found in Bardossy and Duckstein (1992), where a MCDM problem is evaluated using compromise programming with one of the criteria being qualitative and subjective. A codebook, a set of membership functions used to describe categories of subjective information, is established which translates a cardinal scale selection of the subjective criteria into a fuzzy set. Application of the extension principle to combine the single fuzzy criteria with other, quantitative, criteria is demonstrated graphically. Bardossy and Duckstein (1992) and a similar paper by Lee et al (1994) provide the only known examples of a fuzzy displaced ideal which is directly analogous to compromise programming. They do not provide the necessary framework for application of a general fuzzy compromise programming technique.

7.2 Fuzzy Arithmetic

Before describing the proposed fuzzy compromise approach for evaluating discrete alternatives, some properties of fuzzy sets must be examined.

7.2.1 Properties of Fuzzy Sets

The theory of fuzzy sets, initiated by Zadeh (1965), defines a fuzzy set, A, by degree of membership, $\mu(x)$, over a universe of discourse, X, as:

$$\mu_A(x) : X \rightarrow [0, 1] \tag{34}$$

Fuzzy sets are indications of a level of possibility, as opposed to probability. Figure 26 provides an example of a triangular fuzzy set, which is also normal and unimodal. Normality is satisfied by at least a single value with a possibility $\mu(x)=1$. Figure 26 shows a unimodal set because there is only one peak. The function which defines $\mu(x)$ is piecewise linear, but can be any function which satisfies the above equation.

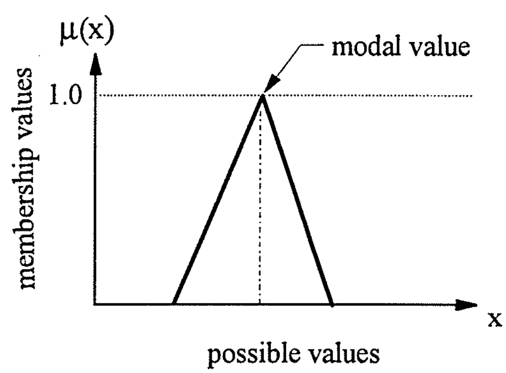

Figure 26. A fuzzy set.

One of the important characteristic properties of a fuzzy set is its degree of fuzziness. As the range of valid x values increases, the degree of fuzziness increases. Also, as more valid x values become more possible (higher membership values), the degree of fuzziness increases. There are many ways of expressing fuzziness. Two general measures of fuzziness are the energy measure and entropy measure defined below:

energy measure $E(A) = \int_x e[\mu_A(x)]dx$ (35)

where $e : [0, 1] \to [0, 1]$ increasing

entropy measure $H(A) = \int_x h[\mu_A(x)]dx$ (36)

where $h : [0, 1] \to [0, 1]$ increasing over $\left[0, \frac{1}{2}\right]$, decreasing over $\left[\frac{1}{2}, 1\right]$

Many of the operations on fuzzy sets use connectives called triangular norms: t-norms; and s-norms. t models the intersection operator in (nonfuzzy) set theory. It is defined by $t : [0, 1] \times [0, 1] \to [0, 1]$, and can be satisfied by any function which exhibits the following properties:

boundary conditions ... $x\,t\,\mathbf{0} = 0,\ \ x\,t\,\mathbf{1} = x$ (37)

monotonicity ... $x < x'$ and $y < y'$ implies $xty \le x'ty$ (38)

commutativity ... $xty = ytx$ (39)

associativity ... $(xty)tz = xt(ytz)$ (40)

Likewise, s models the union operator. It is defined by $s : [0, 1] \times [0, 1] \to [0, 1]$, and can be satisfied by functions with the following properties:

boundary conditions ... $x\,s\,\mathbf{0} = x,\ \ x\,s\,\mathbf{1} = \mathbf{1}$ (41)

monotonicity ... $x < x'$ and $y < y'$ implies $xsy \le x'sy$ (42)

commutativity ... $xsy = ysx$ (43)

associativity ... $(xsy)sz = xs(ysz)$ (44)

The **min** and **max** operators are commonly used for t and s respectively, although the family of valid triangular norms is endless.

Composition operators are also used to connect fuzzy sets in many operations. They include **sup** and **inf**. The operation, **sup**, is the supremum or maximum of its membership function over the universe of discourse. Likewise, **inf** refers to the minimum membership value over a universe of discourse. The combination of composition operators and connectives produces a powerful framework for many operations. **sup-*t*** compositions (**max-min**), and **inf-*s*** compositions (**min-max**) are 2 examples that are used in fuzzy arithmetic calculations.

There are many texts on fuzzy sets, including Dubois and Prade (1982), Zimmerman (1987), Sakawa (1993), and Pedrycz (1995). In particular, Pedrycz (1995) expands on valid triangular norms.

7.2.2 Arithmetic Operations

Arithmetic operations on fuzzy sets is very different from those on normal, crisp, numbers. Fuzzy algebra is made possible by the extension principle, which states that for $Y=f(X)$, $X(x)$ and $Y(y)$ are membership functions (equivalent to $\mu_x(x)$ and $\mu_y(y)$ respectively), there is:

$$Y(y) = \sup_{x \in X;\, y=f(x)} X(x) \tag{45}$$

where $f: X \rightarrow Y$

$y \in Y$

From this extension principle, fuzzy arithmetic can be described as:

$$\mu_{a+b}(x) = \sup_{y \in R} [\mu_a(x-y) \; t \; \mu_b(y)] \tag{46}$$

$$\mu_{a-b}(x) = \sup_{y \in R} [\mu_a(x+y) \; t \; \mu_b(y)] \tag{47}$$

$$\mu_{a \bullet b}(x) = \sup_{y \neq 0} [\mu_a(x/y) \; t \; \mu_b(y)] \tag{48}$$

$$\mu_{a/b}(x) = \sup_{y \in R} [\mu_a(yx) \; t \; \mu_b(y)] \tag{49}$$

$$\mu_{a^b}(x) = \sup_{y \neq 0} \left[\mu_a\left(x^{\frac{1}{y}}\right) \; t \; \mu_b(y) \right] \tag{50}$$

It is important to note that some fuzzy arithmetic operations are not possible for fuzzy sets defined over a universe of discourse which includes the valid value x=0. In particular, multiplication and exponential operations may not exist by the way in which the extension principle is applied.

In a simple arithmetic example to demonstrate the extension principle (Figure 27), $2 + \{\tilde{4}, \tilde{6}\} = \{\tilde{6}, \tilde{8}\}$. Other simple examples of fuzzy arithmetic operations can be found in Mares (1994) or Sakawa (1993).

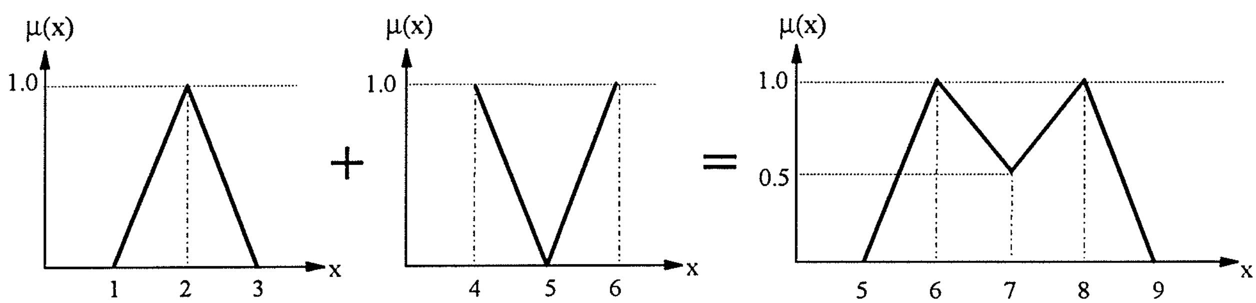

Figure 27. Fuzzy addition example.

In practice, there are various approaches to calculating the results of fuzzy arithmetic operations. To provide general analytical capabilities, there are at least 2 approaches. One is to assume a certain shape for the resulting fuzzy set. The other is to use brute force to find points along the resulting fuzzy set.

Assuming a known fuzzy set shape from a fuzzy arithmetic operation, the general characteristics of the fuzzy set can be demonstrated with an equation which matches the proper shape. This is the technique used in many applications that define left and/or right sided (*L*-, *R*-) fuzzy sets assuming a unimodal normal fuzzy set. The use of equations to define the characteristics is both visually and computationally pleasing when performing arithmetic operations. For example, the output result of adding 2 unimodal, normal fuzzy sets may be approximated using functions for the left and right side of the resulting set with a normal mode and boundaries defined by the nonfuzzy operations on the modes and boundaries of the 2 input fuzzy sets. If a shape cannot be assumed, more (computationally) drastic brute force measures may be necessary.

7.2.3 Brute Force Method

Calculating a fuzzy arithmetic operation with brute force can become convoluted, but is essentially a search of possible arithmetic combinations, taken at every valid point in the resultant set. The search may take the following form, for $X+Y=Z$ (where X,Y,Z are fuzzy sets):

1 Select a point z in Z.
2 Combine X and Y so that the operation on x and y results in z (eg. addition: $x + y = z$).
3 Select the minimum (if the **min** operator is used as the t-norm) membership value from the set $\{\mu_X(x), \mu_Y(y)\}$.
4 Return to 2 until all combinations are exhausted, assuming a Δy.
5 Choose the maximum (**sup**) membership value from the set of minimum (t-norm) operations accumulated in steps 2 through 4.
6 Return to 1 until Z is complete.
7 End search.

This brute force method can become tedious as Δz in the search approaches 0. A relatively coarse search can demonstrate the general characteristics, but accuracy can be a problem in corners of the resultant fuzzy set such as areas in the vicinity of modal values. Another issue is the relative coarseness of the tests between X and Y (step 2). If this test search pattern is too coarse, modal values and other distinct features may be missed.

There are many possible adaptations for the brute force method. One is a very coarse search supplemented by nonfuzzy operations on modal values. Another is to choose a technique for dynamically adjusting the coarseness of the search. Both will be examined.

Supplementing a coarse search with (crisp) modal arithmetic operations is adequate for relatively simple fuzzy sets such as triangular, or one-sided fuzzy sets. The resultant fuzzy set modal values can be assigned, and the coarse search can define the general shape of the remaining set. This technique, however, is unsatisfactory in cases where the fuzzy sets X and Y are not simple. Kinks and corners on X and Y, or situations of subnormality or multimodal properties, may translate to Z but be missed by the coarse search. Modal values and boundaries remain intact, but the general shape may not be indicative of the true shape.

Dynamically adjusting the coarseness of the search to suit the complexity of the fuzzy set is a general approach which can be used for any fuzzy arithmetic operation (Bender and Simonovic, 1996). Multimodal, subnormal, or odd-shaped fuzzy sets can be traced in such a way that test points are concentrated on distinct features while uninteresting features such as straight lines demand few test points. This approach is still essentially a brute force method but the number of test points required to accurately show distinct features may decrease dramatically.

An example adjustment pattern for the search can be generated from an objective to maximize the length of the fuzzy set, l**max**, where l is the length of the membership function, Z. This objective assumes that the brute force method will underestimate the curvature of Z at one or more points. For example, a standard brute force method may be satisfied without finding the mode (m, in Figure 28), resulting in a subnormal mode.

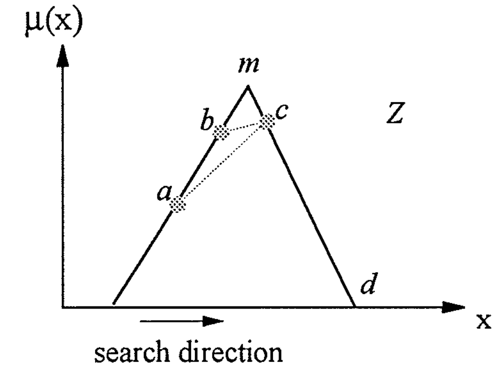

Figure 28. Maximize fuzzy set length search technique for the brute force method.

The following search procedure can be used to achieve the objective of maximizing the fuzzy set length (Bender and Simonovic, 1996):

1 Begin the search at one extreme value (z_{min}) of the fuzzy set, Z, and systematically work towards the other extreme value (z_{max}).
2 Examine 3 test points (a,b,c) at the default coarseness (very coarse), Δz. Test points are found by examining combinations of X and Y (t-norm tests) over the universe of discourse at a resolution Δy.
3 If the combined length of the segments is greater than the direct length between the endpoints, subject to a threshold accuracy (ε), there may exist a feature, m, which may have been missed. In this case, reduce the coarseness (Δz) and examine the area closer. Choosing a reduced step size can be accomplished with any number of methods. The simplest is the bisection method. If the difference in lengths is larger than a specified threshold (ε) reduce the coarseness by a factor of 2, disregard the points b and c, and continue testing with a as the starting point.
4 If the length difference is less than the threshold, ε, disregard b, keep c, and use c as the starting point for the next search. Reset Δz to the default coarseness.
5 Stop the search upon reaching d.

Parametric control over the brute force method using the *l***max** search technique includes the default coarseness of z test points (Δz), the coarseness of arithmetic tests on X and Y (Δy), and the accuracy threshold (ε).

The greatest gain in this search technique is the number of z test points needed to represent the characteristics of Z. A very small number of points are defaulted because: if more points are necessary to represent special features, more points are added automatically. The gain is large because each z test point requires a large number of experiments on X and Y.

While Δz may be coarse due to the l**max** search, accuracy of arithmetic operations also depend on the sensitivity parameter, Δy. The experiments on X and Y need to be extensive, so that modes and other key points are not missed. There is very little that can be done to reduce this number. If very accurate results are required, Δy must be very small.

One form of l**max** search transforms an accuracy threshold, ε, to a tolerance level (σ) - defined as a percentage of l_{ac}, as described by:

$$l_{ab} + l_{bc} = l_{ac} + \varepsilon, \quad \sigma = \tfrac{\varepsilon}{l_{ac}} \tag{51}$$

where the tolerance, σ, is usually small - less than 0.2. This allows a reasonable (and intuitive) interpretation of limits for the search. If σ=0.2, the search will reduce the coarseness of Δz in areas where ε is greater than 20% of the length, l_{ac}. Generally, the transformation of ε to σ prevents regression of Δz to 0 when the search method is trying to locate a corner. In other words, if σ is smaller, the search will be more strict about the desired accuracy. At some point, though, a very small σ will cause the fuzzy arithmetic operation to become convoluted, and the search technique breaks down.

The author found that, by incorporating the l**max** search, computational requirements decreased by 2 or even 3 orders of magnitude of test calculations for each fuzzy operation - while improving results at the same time. For example, in a simple fuzzy addition, Δz and Δy need to be very small for standard brute force. The number of resulting tests may be 500x5000=2,500,000 tests compared to 20x1000=20,000 tests for the brute force method with the l**max** search.

In summary, 2 approaches based on the brute force method are advocated for performing fuzzy arithmetic operations when there is little or no *a priori* knowledge about the shapes of the fuzzy sets. They are:

1 Limited brute force search supplemented with nonfuzzy calculations involving modal points.
2 Brute force method incorporating a search technique to reduce the necessary amount of brute force.

The first approach above is appropriate when the fuzzy sets are known to be normal and unimodal. If modality or normality are not known, the second approach must be used.

7.3 Fuzzy Distance Metrics

The following discussion of fuzzy distance metrics (Bender and Simonovic, 1996) is developed in the form of a general examination, independent of the types of fuzzy sets which an application might be limited.

7.3.1 Sources of Uncertainty

For compromise programming to address the vagueness in the decision maker's value system and criteria value uncertainty, a general fuzzy approach may be appropriate. Leung (1982) demonstrated the displaced ideal concept of compromise programming in a fuzzy decision environment. However, it is somewhat difficult to interpret sensitivity of criteria values. Bardossy and Duckstein (1992) demonstrate, in a limited way, the usefulnes of a fuzzy compromise programming technique for criteria which are subjective. Lai et al (1994) also works within the framework of compromise programming to demonstrate the fuzzification of the ideal point(s). These aspects of fuzziness may occur together, and in conjunction with other sources of uncertainty such as the weights, and the appropriateness of the selected distance measure (exponent).

Simply changing all inputs from crisp to fuzzy produces a definition for fuzzy compromise programming analogous to the crisp original. The multiobjective problem in Figure 25 can no longer consider a single point for the ideal solution, and each alternative now occupies a small region to various degrees. Measurement of distances between the fuzzy ideal and the fuzzy performance of alternatives can no longer be given a single value, because many distances are at least somewhat valid. Choosing the shortest distance to the ideal is no longer a straightforward ordering of distance metrics, because of overlaps and varying degrees of possibility. The fuzzy multiobjective problem, however, contains a great amount of additional information about the consequences of a decision - compared to the nonfuzzy counterpart.

A fuzzy distance metric possesses a valid range of values, each with a characteristic degree of possibility or membership, such that all possible values are a positive distance from the ideal solution (which also becomes fuzzy). Fuzzy inputs include the vagueness of criteria weights, vagueness of both positive and negative ideals, and vagueness in appropriate distance metric exponent. Of course, if any of the inputs are known with certainty, then L becomes less fuzzy.

The process of generating fuzzy sets for input is not trivial. Certainly, arbitrary assignment is simple and may cover the range of possibility, but it is possible to encode a lot of information and knowledge in a fuzzy set. The process of generating an appropriate fuzzy set, accommodating available data, heuristic knowledge, or conflicting opinions, should be capable of preserving and presenting information accurately both in detail an in general form. This topic is not addressed in any great detail in this paper. We consider appropriate techniques for fuzzy set generation to be specific to the type of problem being addressed, the availability of different types of information, and the presence of different decision makers.

By assuming fuzzy set membership functions for the various inputs to a distance metric calculation, a decision maker must make a number of assumptions. Normal fuzzy sets are considered. They acknowledge that there is at least one completely valid value, analogous to the expected value case for probabilistic experiments. In circumstances where at least one modal point cannot be found, it is usually better to assign multiple modal points than to assign low membership values across the range of possible values (the universe of discourse) - partly for the sake of interpreting evaluations. Multimodal fuzzy sets may consist of multiple modal points or a continuous range of modes. The choice of boundaries for the universe of discourse also makes assumptions about available knowledge on the universe of discourse. Boundary and modal point selection, along with the shape of the fuzzy sets, define a degree of fuzziness which hopefully represents the characteristic fuzziness of real world behaviour.

In (hopefully) describing real world, linguistic-type, interpretation of behaviour or validity, fuzzy sets describe a degree of possibility for valid values of a parameter. They do not, however, possess properties such as conditional probabilities for stochastic applications - at least, for simple applications. This fact is acceptable because typical sensitivity analyses explore all combinations of values anyway, and there is usually not enough information to form conditional properties. In an advanced fuzzy application, there is no reason not to provide conditional fuzzy sets.

7.3.2 Properties of Fuzzy Distance Metrics

Remaining discussions on properties of fuzzy distance metrics are for maximization problems. In other words, larger values for criteria are assumed to be better than smaller values, and that the ideal solution tends to have larger values than the alternatives.

It is possible, and may be desirable, to fuzzify all parameters within MCDM problem formulated with a framework using the displaced ideal concept. The following Figure 29 shows typical shapes of input fuzzy sets to be used for criteria values, weights, positive ideals, and negative ideals for compromise programming. The fuzzy sets shown are piecewise linear as: (a,b) one-sided linear, (c) triangular, (d) trapezoidal, or (e) conflicting which combines 2 triangular sets. Nonlinear fuzzy sets can also be used, but this selection typifies the different modal features.

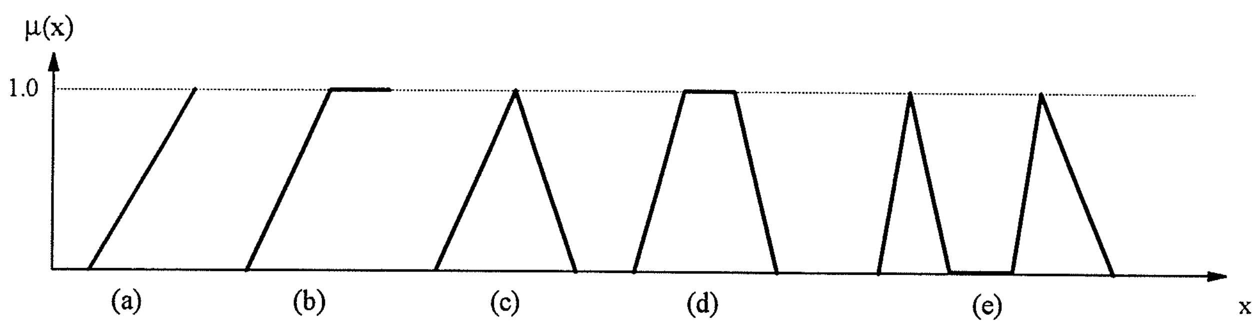

Figure 29. Typical fuzzy input shapes.

Fuzzification of criteria values is probably the most obvious use of fuzzy sets in decision making problems. There is a long history of published articles demonstrating decision problems with qualitative or subjective criteria. By and large, without moving to a completely different decision space such as with Saaty's AHP (1980), subjective criteria values are assessed along a numerical scale which is usually cardinal in nature. To capture the subtleties of relative performance of different alternatives from the perspective of a decision maker, there may not be enough choices. Likewise, if a large number of choices are provided - the appreciation of subjectivity in linguistic terms disappears. Fuzzy sets are able to capture many qualities of relative differences in perceived value of criteria among alternatives. Placement of modal values, along with curvature and skew of membership functions can allow decision makers to retain what they consider degree of possibility for subjective criteria values.

Quantitative criteria present some slightly different properties from qualitative criteria. It can be assumed that quantitative criteria are measured in some way, either directly or through calculation based on some model. They have stochastic properties which describe the probability of occurrence for values, based on future uncertainties for example. They also have some degree of imprecision in their measurement or modelling. In this way, quantitative criteria may have both stochastic and fuzzy properties. To prevent complication of many decision making problems, stochastic uncertainty may be adequately represented with fuzzy sets. In general, application of quantitative criteria within a fuzzy approach may assume that quantitative criteria are less fuzzy than qualitative criteria. However, there may be many exceptions to this rule!

Criteria weights are an aspect of most MCDM methods. Their assignment is completely subjective, usually with a rating on an interval scale. As a subjective value, criteria weights may be more accurately represented by fuzzy sets. Generating these fuzzy sets is also a subjective element. It may be difficult to get honest opinions about degree of fuzziness from a decision. It might actually be more straightforward to generate fuzzy sets for weights when multiple decision makers are involved! Then, at least, voting methods and other techniques are available for producing a composite, collective, opinion. Regardless, more information can be provided about valid weights from fuzzy sets than from crisp weights.

Membership functions for criteria values and criteria weights can both be expressed in 3 distinct forms (Figure 30). They are (a) uncertain (where: known with certainty - is a special case with a small degree of fuzziness), (b) unknown, and (c) conflicting. Both (b) and (c) produce a somewhat conflicting interpretation of valid behaviour.

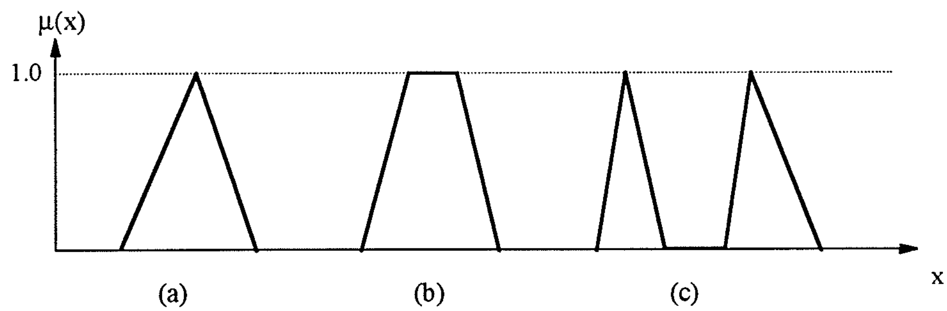

Figure 30. Fuzzy criteria values and weights.

Incorporation of vagueness in the ideal solution is an element which impacts rankings of alternatives. By incorporating fuzziness to the location of the ideal solution (both positive and negative), the valid area for the ideal point - in criteria space - affects the measurement of distance to the alternatives. For example, if profit is a criteria, then what is the ideal amount of profit? Typically, (crisp) compromise programming applications use the largest criteria value among the alternatives as the ideal value. This arbitrary placement is probably not valid, and also affects the relative distances to the overall ideal. In another example, if a subjective criteria is rated on a scale of {1,2,3,4,5}, with linguistic interpretations for each, and all alternatives are rated as {3} or {4}, then positive and negative ideals of {4} and {3} respectively, will not produce distance metrics indicative to overall alternative performance.

Figure 31 shows how positive and negative ideals can be expressed as one-sided fuzzy sets. The 3 choices are (a) certain, (b) uncertain, and (c) unknown. The uncertain (c) case can also be considered as a fuzzy goal. Improvement in criteria value does not improve the level of satisfaction because the goal has already been completely achieved at the initial modal value. The degree of certainty in which the ideals are known is expressed by the range of valid values. Positive and negative ideals may also be triangular or any other complex membership function, but they would typically assume that a larger value is less valid as the positive ideal solution, or that a smaller value is less valid as the negative ideal.

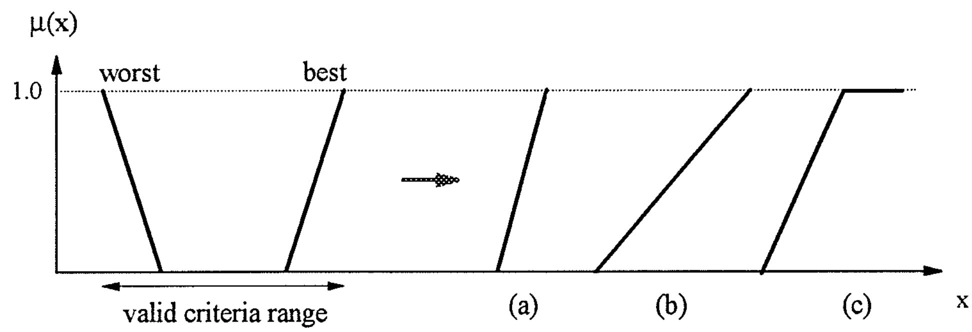

Figure 31. Range of valid criteria values as defined by fuzzy positive and negative ideals.

The distance metric exponent, p, is likely the most imprecise or vague element of distance metric calculation. There is no single acceptable value of p for almost any type of problem, and it can be easily misunderstood. Also, it is not related to problem information in any way except it provides parametric control over interpretation of distance. Fuzzification of the distance metric exponent, p, can take many forms but in a practical way it might be defined in one of the 4 choices shown in Figure 32. (a) and (c) suggest the common practice of using p=2. However, in (a), it is acknowledged that the distance metric exponent has a possibility of being as small as 1. (b) and (d) are the p=1 equivalent. Larger values of p may also be valid but fuzzy exponential operations for large exponents results in an unmanageable degree of fuzziness (range of possible values), making interpretation of the distance metric difficult.

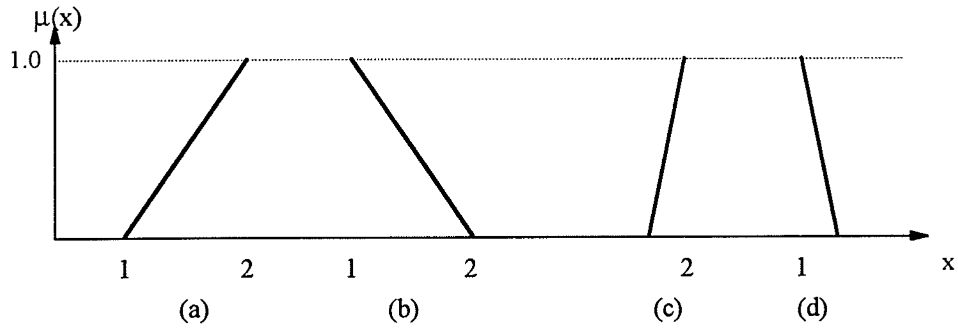

Figure 32. Fuzzy distance metric exponent.

The impact of fuzzy inputs on the shape of the resulting fuzzy distance metric is shown in Figures 31,32. Figure 33 shows typical shapes for L given triangular weights and criteria values, using different interpretations for p. Figure 34 shows the impact of unknown and uncertain weights or criteria values. For linear membership functions, areas about the modal value are impacted. The mode may be spread out, split, or both.

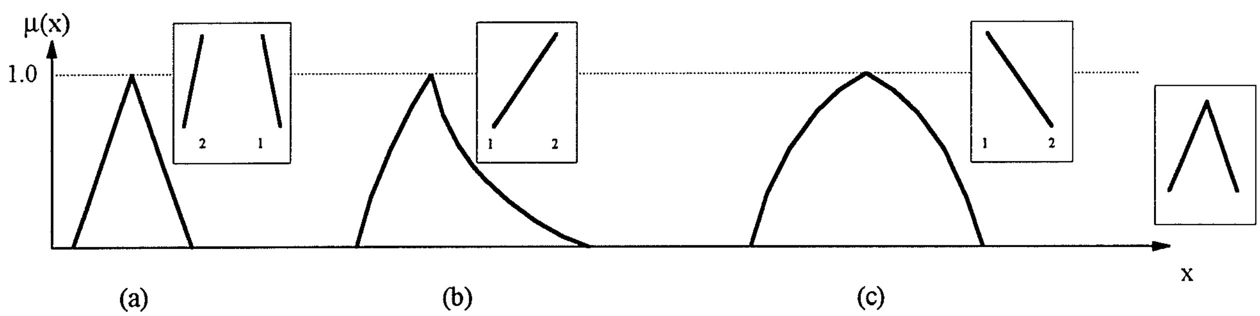

Figure 33. Fuzzy distance metrics for different fuzzy definitions of *p*.

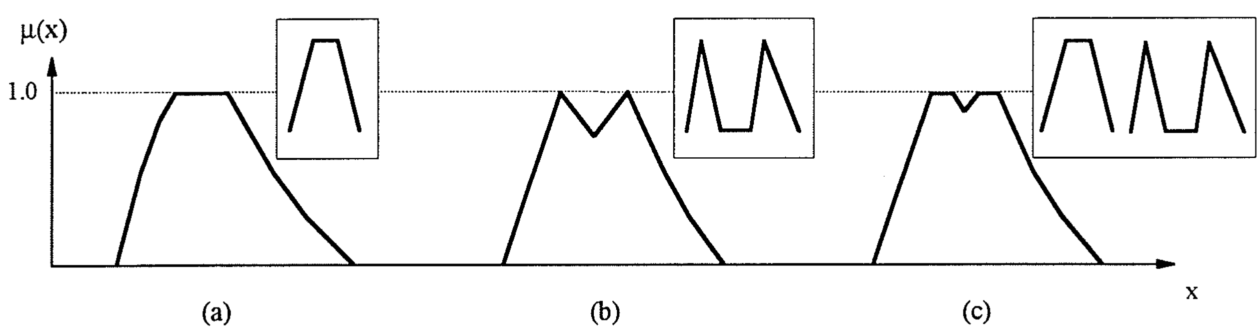

Figure 34. Fuzzy distance metrics for different multimodal criteria values and weights.

7.3.3 Summary

The benefits of adopting the general fuzzy approach to compromise programming are many. Probably the most obvious is the overall examination of decision uncertainty. Sources of uncertainty in criteria values are easily identified (especially for subjective qualitative criteria), but there is also uncertainty or vagueness in other inputs. These are typically ignored, or assessed using a sensitivity analysis. Each scenario is normally treated with equal probability unless special considerations are taken. Fuzzy compromise treats each potential scenario according to its degree of possibility.

Expressing possibility values with fuzzy inputs allows experience to play a significant role in the expression of input information. The shape of a fuzzy set expresses the experience or the interpretation of a decision maker. Conflicting data or preferences can also be easily expressed using multimodal fuzzy sets, making fuzzy compromise very flexible in adapting to group decision making.

Fuzzy criteria values reflect knowledge and confidence regarding the quality of data and models used to calculate criteria values. One assumption in using fuzzy criteria values is that quantitative criteria are generally less fuzzy than subjective criteria. This results in a major enhancement over many MCDM methods. One of the tendencies with evaluating problems containing both quantitative and qualitative criteria (especially for engineers, economists, and other "hard" sciences) may be to assign less weight to "soft" subjective criteria. Even criteria values known with certainty, in a single decision maker problem, may be uncertain when additional decision makers are considered who might disagree on the assessment of the criteria. Fuzzy compromise programming provides better options for expressing differences in subjectivity because uncertainty in relative importance is supplied by the weights and uncertainty in relative value is supplied by criteria values.

Fuzzy criteria weights can serve several purposes. Consideration of context-sensitive weighting produces a valid range of possible values. Ideally, the proper weight is conditional on the context of the decision and available information, but this is normally ill-defined and poorly understood. Choosing weights can then be considered as a fuzzy set of potentially valid weights for a single decision maker. Considering multiple decision makers results in fuzzy weighting, even if each decision maker is confident about their preferences.

The exponent used to define the distance metric indicates the level of compensation between criteria. The overall level of compensation may be fuzzy. Also, there is no single accepted distance metric. If a decision maker is unsure of how to penalize difference from an ideal solution, p can be defined over the range of possible values.

7.4 Selecting Acceptable Alternatives

7.4.1 Comparing Fuzzy Distance Metrics

Traditional (nonfuzzy) compromise programming distance metrics measure the distance from an ideal point, where the ideal alternative would result in a distance metric, L=0. In fuzzy compromise programming, the distance is fuzzy, such that it represents all of the possible valid evaluations, indicated by the degree of possibility or membership value. Alternatives which tend to be closer to the ideal may be selected. This fuzzified distance metric is analogous to a sensitivity analysis for the nonfuzzy case.

Figure 35 shows 2 Ls. If one alternative were to be chosen, the best alternative might be A, a reasonably intuitive choice. Simply consider that A and B have the same shape and degree of fuzziness, but A is shifted toward the origin - which is the ideal solution, assuming that a high membership value near x=0 is desirable. Choosing an alternative, however, is not usually so straightforward. If degree of fuzziness, or characteristic shape is different among the available alternatives, choosing the best compromise solution may be difficult.

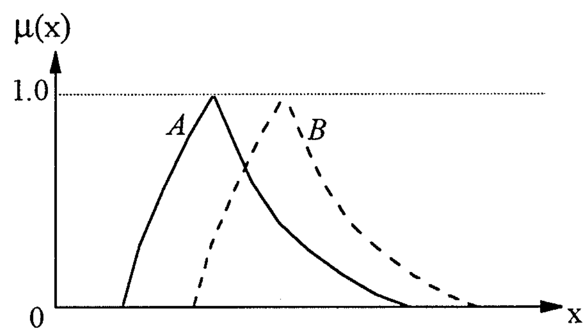

Figure 35. Fuzzy distance metric comparison for 2 alternatives.

As an attempt to standardize a procedure for judging which L is best among a set of alternatives, desirable properties can be defined. The most important properties are:

- Possibility values tend to be close to the ideal, x=0, distance.
- Possibility values have a relatively small degree of fuzziness.

Some other performance indicators might be:

- Modal values are close to the ideal.
- Possibility values tend to be far from poor solutions.

An experienced person may be able to visually distinguish relative acceptability among alternatives, but in cases with many alternatives where each L displays similar characteristics, it may be impractical or even undesirable to make a selection visually. A method for ranking alternatives, based on L, will make summary ranking information more accessible - automating many of the visual interpretations - and creating reproducible results. A ranking measure may also be useful in supplying additional insight into decision maker preferences, such as

- Distinguishing between relative risk aversion and optimism in rank selection by deicision makers
- Allowing adjustments in decision maker emphasis for relatively extreme possibilities.

An aspect of comparing fuzzy distance metrics is the possible occurrence of points of indifference between fuzzy sets. The fuzzy sets in Figure 35 show no *indifference points* between A and B, because the rising limb (L-side) of A is always valid on different x values from B, and likewise for the falling limb (R-side). If, however, the rising limb of A were to intersect the rising limb of B (ie. equal membership values at some point) - a point of indifference would exist, at least in relation to the rising limb. This concept of indifference may vary. In our discussion of equal membership at points along the rising and falling limbs of the fuzzy distance metric, interpretation of "best" depends on which side of the indifference point is considered to be interesting in the judgement of comparative best. In the special case where the modes are equal, while the rising and falling limbs vary drastically, selection of the mode as the point of interest in ranking the sets (similar to the expected value case in probability theory) will result in ranking the 2 fuzzy sets equally. Knowledge of these indifference points may not be directly evident by a ranking measure, but if parametric control is used to test the sensitivity of rankings - indifference points (depending on their location) may cause ranks to alter when fuzzy sets are examined under different "lighting" conditions.

Methods for ranking fuzzy sets are numerous. Selection of a method is subjective and specific to the form of problem and the fuzzy set characteristics which are desirable. A taxonomic examination of existing methods can be found in Bortolan and Degani (1985). Bortolan and Degani review an assortment of methods ranging from horizontal and vertical evaluation of fuzzy sets, to comparative methods. Some of these methods may independently evaluate fuzzy sets, while others use competition to choose among a selection list. Horizontal methods are those related to the practice of defuzzifying a fuzzy set by testing for a range of validity at a threshold membership value. Vertical methods tend to use the area under a membership function as the basis for evaluation, such as center of gravity. The comparative methods are those which introduce other artificial criteria for judging the performance of a fuzzy set, such as a fuzzy goal.

Horizontal methods are not explored here. The following 2 methods are vertical and comparative, respectively.

7.4.2 Weighted Center of Gravity Measure

Given the desirable properties of a ranking method for fuzzy compromise programming, one technique which may qualify as a candidate is the centroid method, as discussed by Yager (1981) in terms of its ability to rank fuzzy sets on the range [0,1]. The centroid method appears to be consistent in its ability to distinguish between most fuzzy sets. One weakness, however, is that the centroid method gives no indication to degree of fuzziness. It is unable to distinguish between fuzzy sets which may have the same centroid, but greatly differ in their degree of fuzziness. The weakness can be alleviated, somewhat, by the use of weighting which provides a source of parametric control over the ranking values. If high membership values are weighted higher than low membership values (in the calculation of a centroid), there is some indication of degree of fuzziness when comparing rankings from different weighting schemes. However, in the case of symmetrical fuzzy sets, weighting schemes will not distinguish relative fuzziness.

A weighted centroid ranking measure can be defined as follows:

$$\mathrm{WCoG} = \frac{\int g(x)\mu(x)^q dx}{\int \mu(x)^q dx} \tag{52}$$

where $g(x)$ is the horizontal component of the area under scrutiny, and $\mu(x)$ are membership function values. In practice, **WCoG** can be calculated in discrete intervals across the valid universe of discourse for L. The values $\mu(x)$ use the exponent, q, to exagerate the scale in each interval (in cases of $q>1$).

The weighted centroid (or Center of Gravity) method, **WCoG**, allows parametric control in the form of the exponent, q. This control mechanism allows ranking for cases ranging from the modal value ($q=\infty$) - which is analogous to an expected case or most likely scenario, to the center of gravity ($q=1$) - which signifies some concern over extreme cases. In this way, there exists a family of valid ranking values (which may or may not change too significantly). The final selection of appropriate rankings is dependent on the level of risk aversion from the decision maker.

Ranking of fuzzy sets with **WCoG** is by ordering from smallest to largest value. The smaller the **WCoG** measure, the closer the center of gravity of the fuzzy set to the origin. **WCoG** values are not restricted to a finite range because it is strictly a measure on valid x values. As a vertical method of ranking, **WCoG** values act on the set of positive real numbers.

7.4.3 Fuzzy Acceptability Measure

Another ranking method which shows promise is a fuzzy acceptability measure, **Acc**, based on Kim and Park (1990). Kim and Park derive a comparative ranking measure called the *Index of Optimism* (IO) which builds on the method of Jain (1976) using the Possibility measure (**Poss**) to signify an optimistic perspective, and supplements it with a pessimistic view similar to the Necessity measure (**Nec**).

The Possibility measure is formally known as the *degree of overlap* for fuzzy sets. It can be described as the possibility of something good happening. The optimism shown in high **Poss** values can be stated linguistically and mathematically as:

I would like to maximize the possibility of producing a good result.

$$\mathrm{Poss}(G, A) = \sup_{x \in R} [\mu_G(x)\ t\ \mu_A(x)] \tag{53}$$

where A is the fuzzy set defined by $A : X \rightarrow [0, 1]$ and G is a fuzzy goal, defined by $G : X \rightarrow [0, 1]$.

The Necessity measure gives a pessimistic view. It is formally known as the *degree of containment* when comparing fuzzy sets. The usefulness of **Nec** can be expressed in the following motivation and similar mathematical form as for **Poss**:

I am concerned about ensuring that something bad does not happen.

$$\mathrm{Nec}(G, A) = \inf_{x \in R} [\mu_G(x)\ s\ \overline{\mu_A}(x)] \tag{54}$$

where $\overline{\mu_A}$ is the complement ($1-\mu_A$) membership value.

The *Index of Optimism* (**IO**) method of Kim and Park (1990) can be defined in terms of the Possibility measure:

$$\text{IO}=\alpha\text{Poss}(G,A)+(1-\alpha)\left(1-\text{Poss}(\overline{G},A)\right) \tag{55}$$

The parameter, α (valid over the range [0,1]), is the expression of relative optimism or pessimism. α=0.5 results in equal weight given to both optimistic and pessimistic views. α=0.9 represents a very optimistic attitude. α=0.1 represents a very pessimistic behavioural pattern.

The method of Kim and Park is demonstrated in Figure 36, where A and B are fuzzy sets to be ranked. They are compared to G (and its counterpart, $\overline{G}$). The circles and squares represent the optimistic value and the pessimistic complement respectively (the pessimistic value is, of course: (1-**Poss**)). Kim and Park define 2 fuzzy goals, a positive (G) and a negative ($\overline{G}$), but since they restrict the range to that of A and B, the negative goal becomes the complement of the positive goal. Should this restriction be relaxed, the method would be very flexible.

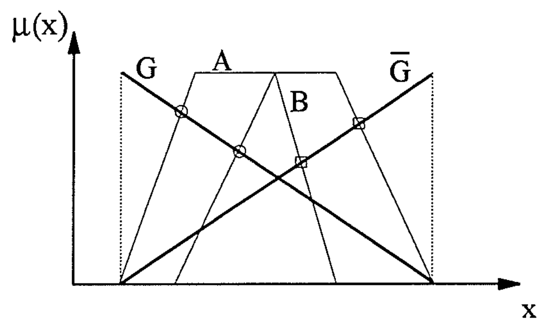

Figure 36. The Index of Optimism method of Kim and Park (1990).

The formulation of **IO** is in terms of **Poss** only, but can be restated as an acceptability measure (**Acc**):

$$\text{Acc}=\alpha\text{Poss}(G,A)+(1-\alpha)\text{Nec}(G,A) \tag{56}$$

This method is subjective only to the definition of G, doing away with the negative goal. Figure 37 demonstrates the **Acc** measure. It is, again, equivalent to Kim and Park, but can be defined in more simple (fuzzy) terms. To apply this Acceptability measure to the evaluation of alternatives, G (or goal) can be defined to have a modal value at x=0, and be decreasing on its range of positive real numbers. As a function, G may typically be linear, but can also be adapted to place more emphasis or less emphasis near the best value (x=0 for distance metrics). For example, the operations of concentration (**CON**=G^2) or dilation (**DIL**=$G^{½}$) can be used to promote gambling or risk averse behaviour respectively. The range of G is also subjective. One acceptable solution is to restrict the maximum possible value of G to the maximum possible L value, which is equivalent to the approach taken by Kim and Park. This, however, may be the subject of parametric control.

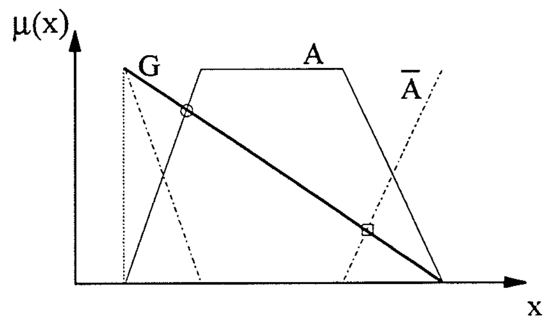

Figure 37. Acceptibility measure, Acc.

Parametric control with the Acceptability measure (**Acc**) is accomplished with the α weight and the choice of fuzzy goal, G. The α weight controls the degree of optimism and degree of pessimism as with the **IO**, and indicates (an overall) level of risk aversion. The choice of fuzzy goal, G, is not so intuitive. It should normally include the entire range of L, but it can be adjusted to a smaller range for the purpose of either exploring shape characteristics of L, or to provide an indication of necessary stringency. By decreasing the range of G, the decision maker becomes more stringent in that the method rewards higher membership values closer to the ideal. At the extreme degree of stringency, G becomes a nonfuzzy number that demands the alternatives be ideal.

Ranking of fuzzy sets using degree of Acceptability is accomplished by ordering **Acc** values from largest to smallest. That is, the fuzzy set with the greatest degree of Acceptability, given the relevant fuzzy goal, is most acceptable. **Acc** values are restricted on the range [0,1] since both the **Poss** and **Nec** measures act on [0,1], and the use of α essentially reduces the range of possible values by a factor of 2.

7.4.4 Comparison of ranking methods

In comparing the above ranking methods (**WCoG**, **Acc**) with those reviewed by Bortolan and Degani (1985), given the desirable properties of L, both are superior to the methods given in the review, and both methods produced similar results.

The problem with many available methods is that, although most are able to correctly identify the best fuzzy set, they may not be capable of distinguishing both degree of dominance and provide an ordinal ranking for more than 2 fuzzy sets. Many methods supplied ranking values, for example, as {1,0,0} for 3 fuzzy sets. Very little decision information is returned by those methods. Relative dominance among fuzzy sets is an important aspect for distinguishing between fuzzy distance metrics. Information of this type is provided by both **WCoG** and **Acc**.

Considering some of the intangibles of the 2 methods shown here, the weighted centroid method (**WCoG**) is more intuitive for a decision maker to understand. The idea of a centroid is easily understood, and visually intuitive. It's weakness in discerning between fuzzy sets with the same shape and modal value, yet with different degrees of fuzziness is offset, somewhat, by the unlikely event of having distance metrics with those properties. Fuzzy distance metrics may have very similar shapes considering that all alternatives are evaluated for the same fuzzy definition of p. They may also have similar modes, depending on criteria values. Degree of fuzziness, or at least some discrepancy in shape, provides the means by which the weighting parameter, q, is able to distinguish indifference points. In general, though, interpretation of indifference points is not usually very sensitive to the choice in q.

The Acceptability measure (**Acc**) provides more comprehensive, and possibly more relevant, parametric control over the interpretation of results. **Acc** is able to explore the "surface" of fuzzy distance metrics (fuzzy sets) with a meaningful interpretation of the variables used for parametric control (α, G). However, unlike the **WCoG** method which might easily be justified under scrutiny, the parameters for the **Acc** measure are difficult to justify if some combination is used to recommend an alternative. The appropriate use of **Acc** is strictly to determine sensitivity, if any, of alternative rankings to different attitudes displayed by a decision maker.

There are many different combinations of shapes and distinct intracacies among fuzzy distance metrics. One thing in common is the relative shape of the rising and falling limbs. They are dependent on the distance metric exponent definition, which must be consistent for all alternatives if a meaningful comparison is to occur. The differences are largely in degree of fuzziness, and modal characteristics. Large differences in input fuzziness can cause significant reduction or expansion of the valid range of distances from the ideal. Conflicting or unknown input may produce wider ranges where membership values of some alternatives remain close to unity. Regardless of the combination of characteristics for fuzzy distance metrics, both the **WCoG** and **Acc** methods produced similar results that corresponded with visual interpretation of fuzzy distance metric plots.

Both methods satisfy our desirable properties for ranking fuzzy distance metrics. Both may prove to be useful in a decision making problem with multiple alternatives. Choosing just one of these methods, or a completely different method (of which there are many), should be dependent on the desirable ranking properties of the given problem. In some cases, it may be advantageous to use more than one method as a form of verification.

7.5 Examples

The following examples of using the fuzzy compromise approach are taken from the field of water resources planning as found in Bender and Simonovic (1996). They are multicriteria decision problems, originally addressed using standard MCDM techniques such as ELECTRE and Compromise Programming to select a most desirable water management system alternative, either as a best compromise or as a robust choice. The problems all deal with subjective criteria which have been processed as discretized (nonfuzzy) values. They assume scalar values for subjective criteria evaluations, and criteria weights. They also, at most, deal with uncertainty in the criteria weights by experimenting with a few combinations. Each example will be redefined in fuzzy terms to demonstrate the fuzzy compromise technique.

7.5.1 Tisza River Example

The Tisza River basin in Hungary, east of Budapest, was studied by David and Duckstein (1976) for the purpose of comparing alternative water resource systems for long range goals. They attempt to follow a cost effetiveness methodology to choose from 5 alternatives, but many of the 12 criteria are subjective.

The last 8 criteria in Table 6 are subjective, and have linguistic evaluations assigned to them. The criteria for water quality, recreation, flood protection, manpower impact, environmental architecture, and development possibility are all considered on a scale with 5 linguistic options {*excellent*, *very good*, *good*, *fair*, *bad*}. The last 2 criteria are judged by different linguistic scales. First of all, international cooperation has a subjective scale {*very easy*, *easy*, *fairly difficult*, *difficult*}. Finally, the sensitivity criterion also uses a subjective scale with 4 categories (although one of them is not chosen) {*very sensitive*, *sensitive*, *fairly sensitive*, *not sensitive*}. No numeric values are provided by David and Duckstein (1976), but numeric differences along an interval scale are given so that a discordance index can be calculated for the ELECTRE method.

Table 6. Original values used in David & Duckstein (1976).

Criteria	Alternatives				
	I	II	III	IV	V
total annual cost	99.6	85.7	101.1	95.1	101.8
probability of water shortage	4	19	50	50	50
energy (reuse factor)	0.7	0.5	0.01	0.1	0.01
land and forest use (1000 ha)	90	80	80	60	70
water quality	very good	good	bad	very good	fair
recreation	very good	good	fair	bad	bad
flood protection %	good	excellent	fair	excellent	bad
manpower impact	very good	very good	good	fair	fair
environmental architecture	very good	good	bad	good	fair
development possibility	very good	good	fair	bad	fair
international cooperation	very easy	easy	fairly difficult	difficult	fairly difficult
sensitivity	not sensitive	not sensitive	very sensitive	sensitive	very sensitive

Issues of uncertainty are not addressed. Subjective criteria are assigned numeric values. Quantitative criteria do not address any stochastic uncertainties normally associated with modelling adequacy, data accuracy, or temporal instability. As well, additional criteria are listed, but are assumed to be handled implicitly.

The weighting of relative importance is also an issue of uncertainty. David and Duckstein (1976) provided criteria weights to calculate the concordance index of ELECTRE. Weights were supplied from the set of {1,2}. All criteria were weighted as 2 except land and forest use, manpower impact, development possibility, and sensitivity - which are given a weight of 1. In planning water resource systems, a single decision maker, regulatory agency, or interest group is rarely able to represent the interests of others that are impacted by changes in the system. The range of criteria used to evaluate alternatives is admirable, but the relative importance and the relationships between those criteria - defined by different perspectives of stakeholders - is an important, if not controlling, aspect of planning water resource systems. The technique used by ELECTRE somewhat alters the weighting issues in its use of a concordance index, and weights are not needed to calculate a discordance index, but it is not known what affect uncertainty in the weights has on assessing alternative tradeoffs.

As a conclusion, without pursuing a sensitivity analysis, David and Duckstein suggest that a mix of systems I and II would be appropriate - since they appear to somewhat dominate the other alternatives and show no overall domination over each other. Duckstein and Opricovic (1980) reached similar conclusions for the same system, using a different artificial scaling for subjective criteria.

A sensitivity analysis is implied by David and Duckstein (1976) to be the next logical step in the planning of the Tisza River basin. Changes to the data, weights, and time horizon are suggested. Although changes to the data may have probabilistic implications, criteria weights and certainly the impact of the time horizon are more vague because many values may be possible and entirely valid.

A useful improvement to evaluating water resource systems such as the Tisza River may be to treat uncertainties as fuzzy. Although fuzzy applications may not usually exhibit the same explicit definitions as stochastic uncertainties, it should suffice for long range planning problems.

The following (Figure 38) are the fuzzy definitions for linguistic terms used in assessing subjective criteria. Quantitative criteria are also fuzzified, but generally are less fuzzy.

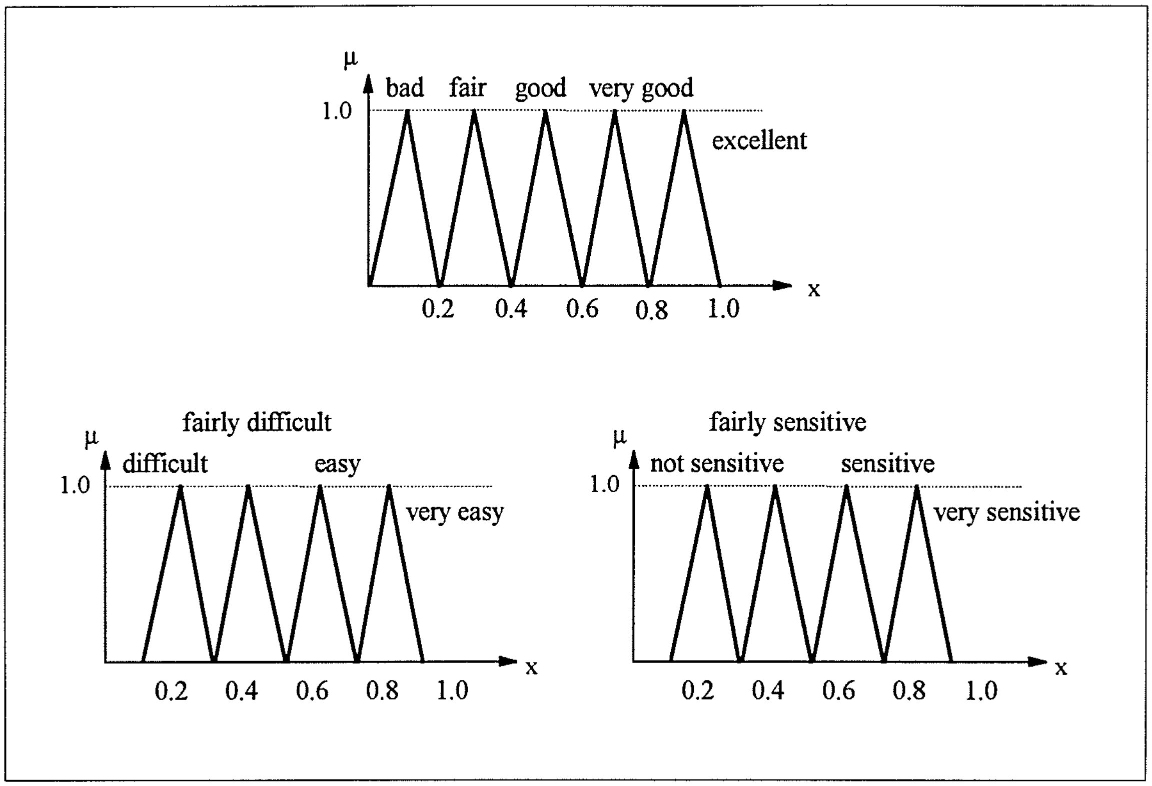

Figure 38. Fuzzy subjective criteria interpretation for the Tisza River problem.

Other fuzzy inputs include the expected ranges of criteria values (Figure 39a) and the form of distance metric or degree of compensation among criteria for different alternatives (Figure 39b). Criteria weights are fuzzified on a range of [0,1] by simple scaling of the weights used by David and Duckstein $\{1,2\} \rightarrow \{\tilde{.33}, \tilde{.66}\}$ (Figure 39c). All of the fuzzy inputs are treated in a simple form, exclusively normal and unimodal. They have either triangular or one-sided membership functions.

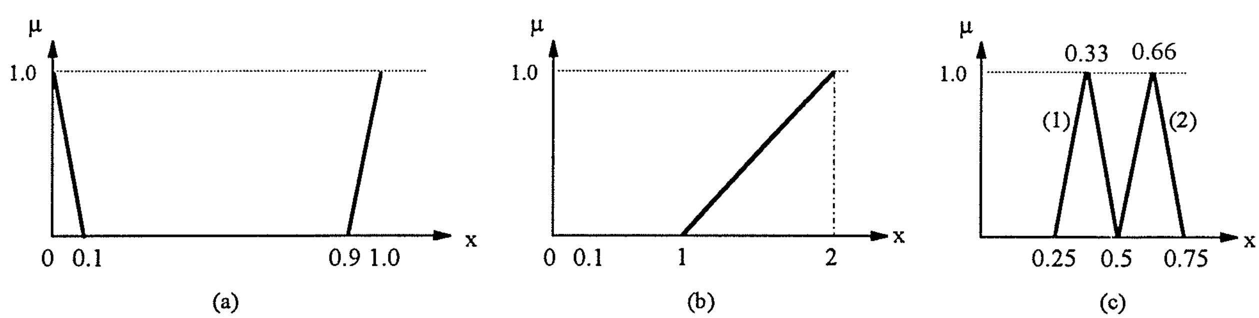

Figure 39. Fuzzy input for the Tisza River problem.

Assuming the fuzzy definition for the distance metric exponent (p), and knowing the form of criteria values and weights to be triangular, the resulting fuzzy distance metrics (L_i) posess the characteristic shape (Figure 40) of near linearity below the mode, and a somewhat quadratic polynomial curvature above the mode. Although the degree of fuzziness (range of valid distnces from the ideal solution) is similar for all 5 alternatives, some of the alternatives are clearly inferior.

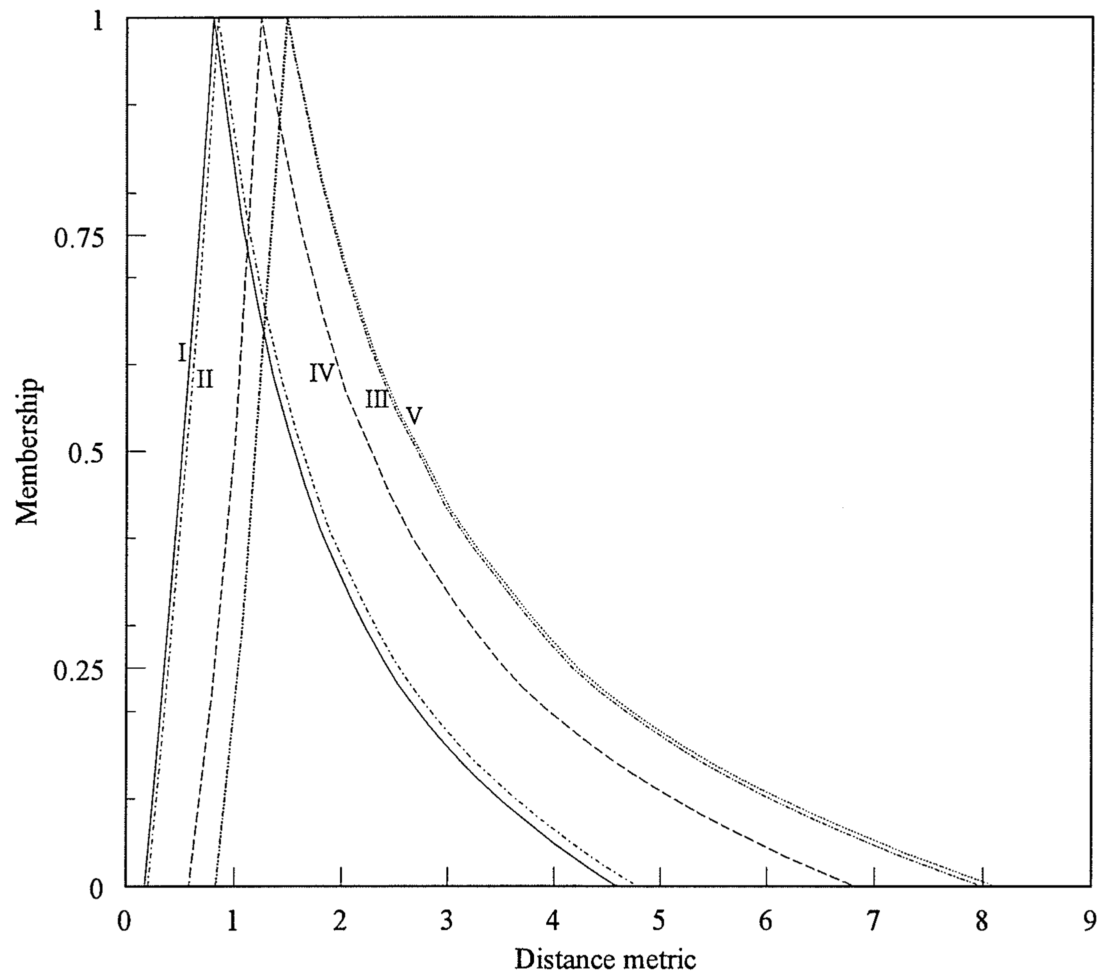

Figure 40. Distance metrics for the Tisza River problem.

Rankings of these alternatives is reasonably straightforward because of the simplicity of the shapes, and similarity in degree of fuzziness. Both the weighted centroid and acceptability measures produced expected results (Table 7) for arbitrary parameter settings on both methods. Rankings were insensitive to changes in levels of risk aversion, as would be expected from visual inspection. The resulting ranks confirm the findings of David and Duckstein (1976), that alternatives I and II dominate III, IV, and V.

Table 7. Tisza River alternative rankings from WCoG and Acc measures.

Rank	Alt	WCoG (p=1)	Acc (G:[0,8], α=0.5)
1	1	1.49	0.81
2	2	1.59	0.8
3	4	2.38	0.75
4	3	2.83	0.72
5	5	2.85	0.71

In a live case study with multiple decision makers, there are opportunities for a group emphasis to collectively adjust fuzzy input to the Tisza River problem. The rankings may change considerably because the values defined for this experiment are predominately simple triangular membership functions, given the form of nonfuzzy input data. Adjustments in relative fuzziness, and the emergence of conflicting opinions about valid criteria values or weights, may produce an entirely new outlook - one which may be sensitive to the level of risk aversion characterized by the decision maker.

7.5.2 Tucson Area Example

The problem described by Gershon et. al. (1982) is an interesting regional water management problem in the Tucson area of Arizona (USA). A number of flood control alternatives such as levee construction are candidates for combination with any one of a set of water supply alternatives - with each combination evaluated by a set of criteria. Ultimately, a combination of partial solutions will be chosen. The result is a large number of partial alternatives (25 in all), numbered according to Table 8.

Table 8. Index numbers for alternative systems.

Water supply actions	Flood control actions				
	Levee construction	Channelization	Reservoirs and dams	Flood plain managemen t	No action
Waste water reclamation	1	2	3	4	5
Groundwater development	6	7	8	9	10
Central Arizona project	11	12	13	14	15
Conservation and education	16	17	18	19	20
No action	21	22	23	24	25

Evaluation of alternatives is accomplished using ELECTRE. The inputs include the criteria weights, and criteria values. There are 13 criteria in all, 8 of which are subjectively assessed on the qualitative scale $\{a,b,c,d,e\}$ where a is best and e is worst for whatever context is relevant. Gershon et. al. (1982) make a point of using this qualitative scale whenever possible to alleviate issues involving units of measure. The criteria used in the Tucson area problem are shown in Table 9, along with the original weights used by Gershon.

Table 9. Evaluation criteria and weights for the Tucson area problem.

#	Criteria	w_i
1	Aquifer level, ft/yr	9
2	Water quality, urban	3
3	Water quality, agriculture	3
4	Expected flood losses	4
5	Expected flood frequency	5
6	Preservation designated areas	5
7	Effect on wildlife and vegetation	5
8	Implementation costs	2
9	Operations and maintenance costs	2
10	Indirect costs	2
11	Natural resource	2
12	Preservation of existing facilities	1.5
13	Creation of new opportunities	1.5

Robustness of the ELECTRE dominance relationships was evaluated with 2 approaches. One approach was to reverse the criteria relationships (so that an alternative with a previously high concordance index would revert to a low concordance, likewise for discordance) to see if the rankings would be reversed. The other approach was to change the interval scales and criteria weights. Besides the base case scales and weights, one other combination was tested - equal scales and weights for all criteria.

Gershon et. al. (1982) identified waste water reclamation (from the water supply list of possible actions) as a highly desirable component to the river basin development strategy. Development of reservoirs was identified from the flood control list. Accommodation of different weight combinations does not seem to be a priority when using ELECTRE. Gershon et. al. (1982) note that although the change in weights to equal weighting represents a dramatic shift in priorities, the ranks did not change significantly. This may be a quirk of the specific problem, but rankings which are insensitive to changes in weight appears to be a general observation about ELECTRE. Whether this characteristic is desirable for group planning remains to be seen.

Investigating the Tucson area planning problem with fuzzy compromise programming, the fuzzy definitions given in Figure 41 are used to describe the qualitative evaluations $\{a,b,c,d,e\}$. Quantitative criteria are also fuzzified, but with only a small range of uncertainty. Criteria weights are fuzzified on a scale of [0,1] instead of [0,10], and are given an arbitrary degree of fuzziness. For example, the fuzzy weight for criteria #1 is normal triangular unimodal linearly increasing over [0.85,0.9] and linearly decreasing over [0.9,1.0].

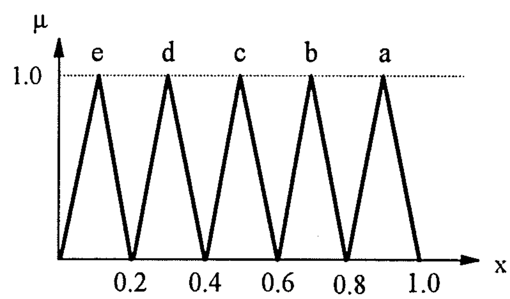

Figure 41. Fuzzy interpretation of qualitative criteria for the Tucson area problem.

Fuzzy distance metrics can be calculated based on the fuzzy inputs, for each of the 25 alternatives. It is very difficult to visually sort through so many alternatives. It may also be difficult to sort through the variety of ranks in Table 10 (for both **WCoG** and **Acc** measures) which samples a variety of levels for risk aversion. Certain alternatives do, however, exhibit consistently high rankings {2,4,17,19,22}. Other alternatives display an obvious sensitivity to changes in degree of risk aversion, such as {1,2,5,6,25}, at least in their relative ranking. This sensitivity is more apparent in **Acc** rankings, compared to **WCoG**.

Table 10. Rankings for the Tucson area problem.

	WCoG			Acc														
Rank	q			α, G:[0,6]					α, G:[0,4]					α, G:[0,3]				
	1	2	3	0.1	0.3	0.5	0.7	0.9	0.1	0.3	0.5	0.7	0.9	0.1	0.3	0.5	0.7	0.9
1	19	19	2	19	19	19	19	2	24	24	24	22	22	19	19	19	2	2
2	24	17	17	4	4	2	2	17	19	19	19	17	17	24	24	24	17	17
3	17	22	22	24	2	4	17	22	4	4	22	2	2	4	17	17	22	22
4	22	2	19	2	17	17	22	19	22	22	17	24	24	17	22	2	19	19
5	4	24	24	17	22	22	4	12	17	17	4	19	19	22	2	22	24	24
6	2	4	4	22	24	12	12	7	2	2	2	4	4	2	4	4	4	12
7	12	12	12	9	12	7	7	4	12	12	12	12	12	12	12	12	12	7
8	23	7	7	12	9	9	23	23	9	7	7	7	7	7	7	7	7	4
9	7	23	23	7	7	23	9	18	7	9	23	23	23	9	23	23	23	23
10	9	9	9	23	23	18	18	9	23	23	9	18	18	23	9	9	18	18
11	18	18	18	14	14	14	3	3	14	18	18	9	3	14	18	18	9	3
12	3	3	3	18	18	3	13	13	18	14	3	3	13	18	14	13	13	13
13	14	13	13	3	3	13	14	14	3	3	13	13	9	13	13	3	3	9
14	13	14	14	13	13	24	8	8	13	13	14	14	14	3	3	14	14	14
15	8	8	8	5	8	8	16	16	8	8	8	8	8	8	8	8	8	8
16	21	21	16	8	5	16	24	21	5	21	21	21	21	21	21	21	21	21
17	16	16	21	25	16	21	21	1	25	16	16	16	16	16	16	16	16	16
18	5	1	1	15	21	1	1	11	21	5	1	1	1	5	1	1	1	1
19	25	11	11	16	25	11	11	6	16	1	11	11	11	25	11	11	11	11
20	1	5	6	21	15	5	6	24	15	25	5	6	6	15	5	5	6	6
21	11	25	25	20	1	25	5	20	1	11	25	5	15	1	25	6	5	5
22	15	15	5	1	11	15	25	5	20	15	15	25	25	11	15	25	25	15
23	20	6	15	11	20	20	20	25	11	20	6	15	5	20	20	15	15	25
24	6	20	20	10	6	6	15	15	10	6	20	20	20	6	6	20	20	20
25	10	10	10	6	10	10	10	10	6	10	10	10	10	10	10	10	10	10

Summaries of Table 10 can be made for specific conditions. A simple accumulation of ranks is used in Table 11 to show the relative performance of different water supply and flood control measures. The result is specific for a goal, G, defined linearly decreasing on [0,4], and an α=0.5. The recommendations of Gershon, under these perceptions from the decision maker, are 3rd best for both water supply actions (waste water reclamation) and flood control actions (reservoirs and dams). Table 12 is a summary for more stringent alternative performance requirements, and a more risk averse decision maker. The water supply action ranks showed very little sensitivity to these changes in decision context. However, overall flood control preferences changed - making flood plain management techniques more desirable. This or similar techniques can be used to qualify extremes in risk aversion, and relative performance stringency, as a high level sensitivity analysis on a fuzzy distance metric.

Table 11. Ranking summary for Acc, G:[0,4], α=0.5.

	Flood control						
Water supply	LC	C	RD	FPM	NA	sum	rank
WWR	18	6	12	5	20	61	3
GWD	23	8	15	10	25	81	5
CAP	19	7	13	14	22	75	4
C&E	17	4	11	2	24	58	2
NA	16	3	9	1	21	50	1
sum	93	28	60	32	112		
rank	4	1	3	2	5		

Table 12. Ranking summary for Acc, G:[0,3], α=0.1.

	Flood control						
Water supply	LC	C	RD	FPM	NA	sum	rank
WWR	21	6	14	3	18	62	3
GWD	24	8	15	9	25	81	5
CAP	22	7	13	11	20	73	4
C&E	17	4	12	1	23	57	2
NA	16	5	10	2	19	52	1
sum	100	30	64	26	105		
rank	4	2	3	1	5		

The fuzzy compromise evaluation of the Tucson area alternatives demonstrates the ability of the fuzzy approach to indicate a level of uncertainty in the performance of alternatives, and to quantify implications of risk aversion tendencies by the decision maker(s). The fuzzy evaluation clarified these kinds of observations with degree of risk aversion from the decision maker.

7.5.3 Yugoslavia Example

A MCDM problem for choosing a water resource system management alternative in the former Yugoslavia has been documented by Simonovic (1989). Simonovic explored 2 systems with the same 8 criteria - 2 of which are quantitative, while the remaining 6 are subjective. System *S2* has 8 alternatives to choose from, and system *S1* has 6 alternatives. Alternatives for both systems are evaluated with compromise programming.

7.5.4 System *S2*

Table 13 lists the original criteria values used to evaluate *S2*. The subjective criteria are judged on a scale {1,2,3,4,5} interpreted as {*bad*, *fair*, *good*, *very good*, *excellent*}. The weights required to assess the distance metrics are given in Table 14. Size different sets of weights are given in all. Each represents a different emphasis in decision making. Any or all of the weight sets may be valid.

Table 13. Criteria input for Yugoslavian example, system *S2*.

	Criteria							
	1	2	3	4	5	6	7	8
Alternative	109din.	Gwh	subjective estimation					
1	-83.3	-3.4	3	4	3	5	3	4
2	-88.3	-3.5	3	2	4	4	4	4
3	-82.9	-3.3	5	4	3	4	3	4
4	-87.5	-3.4	4	3	3	3	5	4
5	-95.4	-3.7	2	5	4	5	3	4
6	-94.1	-3.4	5	3	3	4	4	4

7	-85.9	-3.4	4	4	3	5	3	3
8	-83.8	-3.4	5	2	5	3	5	3

Table 14. Sets of weights used to represent different emphasis for both system *S1*, and *S2*.

	Criteria							
weight set	1	2	3	4	5	6	7	8
1	1.2	1	0.7	0.7	0.5	1.8	1.1	1
2	1.5	1.5	0.8	0.5	0.3	1.2	1.3	0.8
3	1	1	1	1	1	1	1	1
4	1	1	1.4	1.3	1	1.2	0.3	0.7
5	1.2	1	0.7	0.7	0.5	1.1	1.8	1
6	1	1.2	0.7	0.7	0.5	1.8	1.1	1

Simonovic applies weight set 6 with p=2 for the distance metric exponent, and ranks alternative 1 the most desirable - as an example of using compromise programming. Certainly, there are many combinations which can be tested as part of a sensitivity analysis. Unfortunately, they provide no indication of relevance or degree of possibility. A fuzzy compromise application for *S2* may be beneficial.

Fuzzy definitions for the subjective criteria values shown in Table 13 are given in Figure 42. In the Tucson and Tisza River examples, subjective criteria values were interpreted using normal unimodal triangular membership functions which did not overlap at all. Each subjective term was given to be entirely unique and independent from any other subjective term in the fuzzy codebook. For system *S2* (and also *S1*), the range of valid x values is more continuous, there is less separation of fuzzy terms. For example, at x=2.5, the fuzzy terms *fair* (2) and *good* (3) are equally valid at membership, $\mu(x)$=0.5.

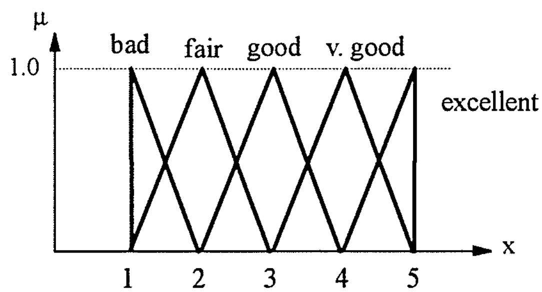

Figure 42. Fuzzy definitions for subjective criteria values, *S2*.

Figure 43 shows the collection of fuzzy criteria weights that are (loosely) defined as a combination of the 6 sets of nonfuzzy weights, as a collective opinion. Although these weight definitions are assumed, a voting technique can be used to generate the collective weights from a group of decision makers.

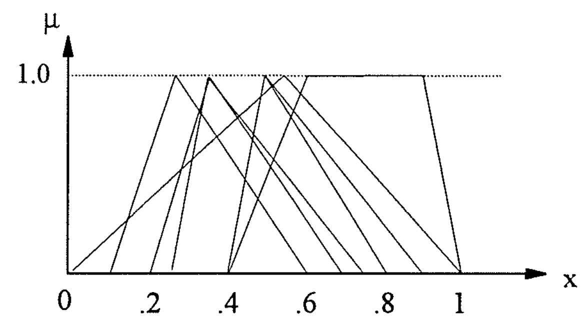

Figure 43. Example collective fuzzy weighting, criteria 2.

Upon definition of p as linearly increasing on the range [1,2] from a boundary {0} to the modal value {1}, the input necessary for fuzzy compromise programming is complete. Fuzzy distance metrics can be calculated with fuzzy arithmetic operations.

Selected fuzzy distance metrics are shown in Figure 44. Notice, by visual inspection, that alternative {6} is dominated by {1,3}. The alternatives {1,3} are very similar, with slightly different modal values - {1} slightly smaller than {3}. If one looks closely at {1,3}, you will notice that there are 2 indifference points -

one on the rising limb (L-side) at x=0.45, and one on the falling limb (R-side) at x=0.8. They suggest that ranking {1,3} is subjective. Also notice that {3} has a lesser degree of fuzziness.

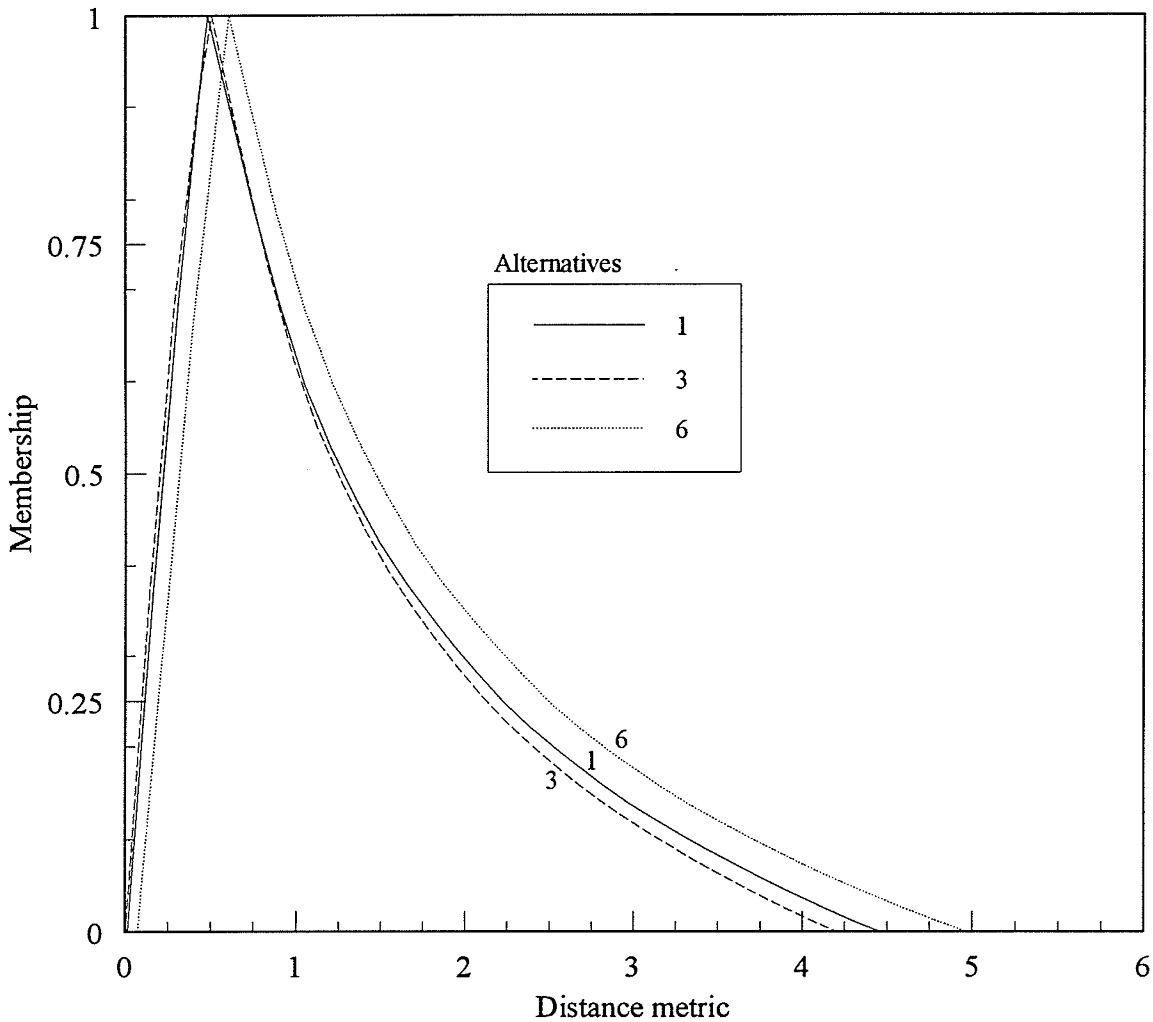

Figure 44. Fuzzy distance metrics: selected alternatives for *S2*.

Although the fuzzy weights have been defined in a collective fashion, it may be interesting to see the impact of alterations on the fuzzy inputs. For instance, there are many valid interpretations for specifying membership functions of fuzzy weights. The base case described above consists of predominately triangular input fuzzy sets, and a distance metric exponent defined as linearly increasing over [1,2].

Using the 6 sets of nonfuzzy weights as the basis for generating different scenarios, the following test patterns for collective interpretation of criteria weighting are explored:

1. Conflicting opinions (multimodal weights).
2. Unknown or very uncertain preferences (trapezoidal weights).
3. Both conflicting and unknown opinions.
4. Both conflicting and unknown opinions, with the distance metric exponent defined as linearly decreasing on the range [1,2] with a modal value at x=1.
5. Both conflicting and unknown opinions, with the distance metric exponent defined as nearly certain for p=2, linearly increasing on the range [1.9,2] with a modal value at x=2.

For *Test 1*, the weights for criteria {5,6} are treated as conflicting, where 2 distinct opinions emerge. The conflicting opinions expressed in the criteria 6 weight have modal values at x={0.5,0.9}, with valid membership values over the range [0.5,0.95] and a single point at x=0.7 where $\mu(x)$=0. Resulting ordinal rankings from the **Acc** measure are shown in Table 15 for α = {0.1, 0.3, 0.5, 0.7, 0.9} and G linearly decreasing from $\mu(x)$=1 to $\mu(x)$=0 over the range [0,4]. Although the 2 highest ranked alternatives {1,3} do not change order over the sensitivity analysis of both the base case weights and the *Test 1* case weights, there are several differences in ranking alternatives {2,4,6,7,8}.

Table 15. Ranking of alternatives for the Base case of fuzzy weights and *Test 1*.

Rank	Base case					Test 1				
	0.1	0.3	0.5	0.7	0.9	0.1	0.3	0.5	0.7	0.9
1	3	3	3	3	3	3	3	3	3	3
2	1	1	1	1	1	1	1	1	1	1
3	7	7	7	7	7	8	8	7	7	7
4	8	8	8	8	4	7	7	8	8	8
5	4	4	4	4	8	4	4	4	4	4
6	6	6	2	2	2	6	6	6	6	6
7	2	2	6	6	6	2	2	2	2	2
8	5	5	5	5	5	5	5	5	5	5

Instead of interpreting the weights as conflicting, criteria {6,7} may be interpreted to be very uncertain or unknown over part of the valid range (*Test 2*). The trapezoidal membership function chosen for the criteria

6 weight includes modal values on the range [0.6,0.9], and membership values restricted on the range [0.4,1.0]. The change in interpretation affects the rankings shown in Table 16 for the **Acc** measure.

Table 16. Ranking of alternatives for Test 2.

Rank	α				
	0.1	0.3	0.5	0.7	0.9
1	3	3	3	3	3
2	1	1	1	1	1
3	7	7	7	7	7
4	8	8	8	8	8
5	4	4	4	4	4
6	6	6	2	2	2
7	2	2	6	6	6
8	5	5	5	5	5

Test 3 on system *S2* uses conflicting weight definitions for criteria {5,6} and a trapezoidal membership function to signify unknown preferences about criteria {7}. This combined test uses the default membership function for p (Figure 45a). *Test 4* alters p as in Figure 43b. Finally, *Test 5* uses p defined by Figure 45c.

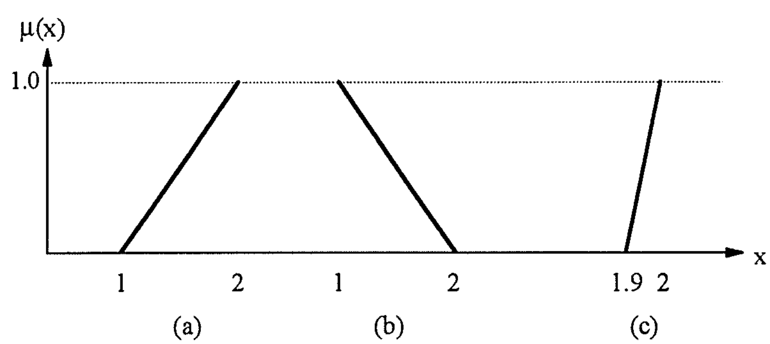

Figure 45. Definition of p for *Tests 3,4,5.*

A summary table of rankings using the **Acc** measure (Table 17) indicates some change in alternative ranking under the different distance metric definitions, and for varying degrees of risk aversion. Alternative 8, for example, is ranked as high as 2nd and as low as 5th.

Table 17. Acc rankings for *Tests 3,4,5*.

Rank	Test 3					Test 4					Test 5				
	0.1	0.3	0.5	0.7	0.9	0.1	0.3	0.5	0.7	0.9	0.1	0.3	0.5	0.7	0.9
1	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	1	1	1	1	1	**8**	1	1	1	1	1	1	1	1	1
3	8	8	7	7	7	1	8	8	8	7	7	7	7	7	7
4	7	7	8	8	4	7	7	7	7	8	8	8	8	8	4
5	4	4	4	4	**8**	4	4	4	4	4	4	4	4	4	**8**
6	6	6	2	2	2	6	6	6	6	2	2	2	2	2	2
7	2	2	6	6	6	2	2	2	2	6	6	6	6	6	6
8	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5

7.5.5 System *S1*

The other system examined by Simonovic (1989), *S1*, consists of 6 alternatives which are evaluated by the 8 criteria given for *S2*. Simonovic considered (nonfuzzy) input from 3 decision makers in a compromise programming sensitivity analysis on subjective criteria values, *p* definition, and the sets of weights provided for system *S2*. The sensitivity analysis suggested that alternatives {3,5} are reasonably robust in that they are consistently ranked at the top. Other alternatives that ranked high were {4,6}.

For system *S2*, criteria weights and the distance metric exponent were adjusted to reflect different interpretations of decision maker preferences in *Tests 1,2,3,4,5*. Another form of uncertainty or vagueness arises when several decision makers all cast judgement on subjective criteria values. Each judgement may be, in itself, vague or imprecise. Combining the opinions of several people may serve to strengthen the impression of a subjective criteria, or it may contribute additional vagueness. This additional vagueness is demonstrated in a fuzzy test case (*Test 6*) for system *S1*.

Fuzzification of the weights and criteria values remains the same as for the base case of *S2*. The resulting distance metrics are shown in Figure 46. Notice that alternative 1 (the thicker line) appears to be worse than the others. The shapes are all similar because of the definition of the exponent, p, and also because the weights and criteria values are predominately triangular. They differ, however, in modal values (expected case scenarios) and in degree of fuzziness.

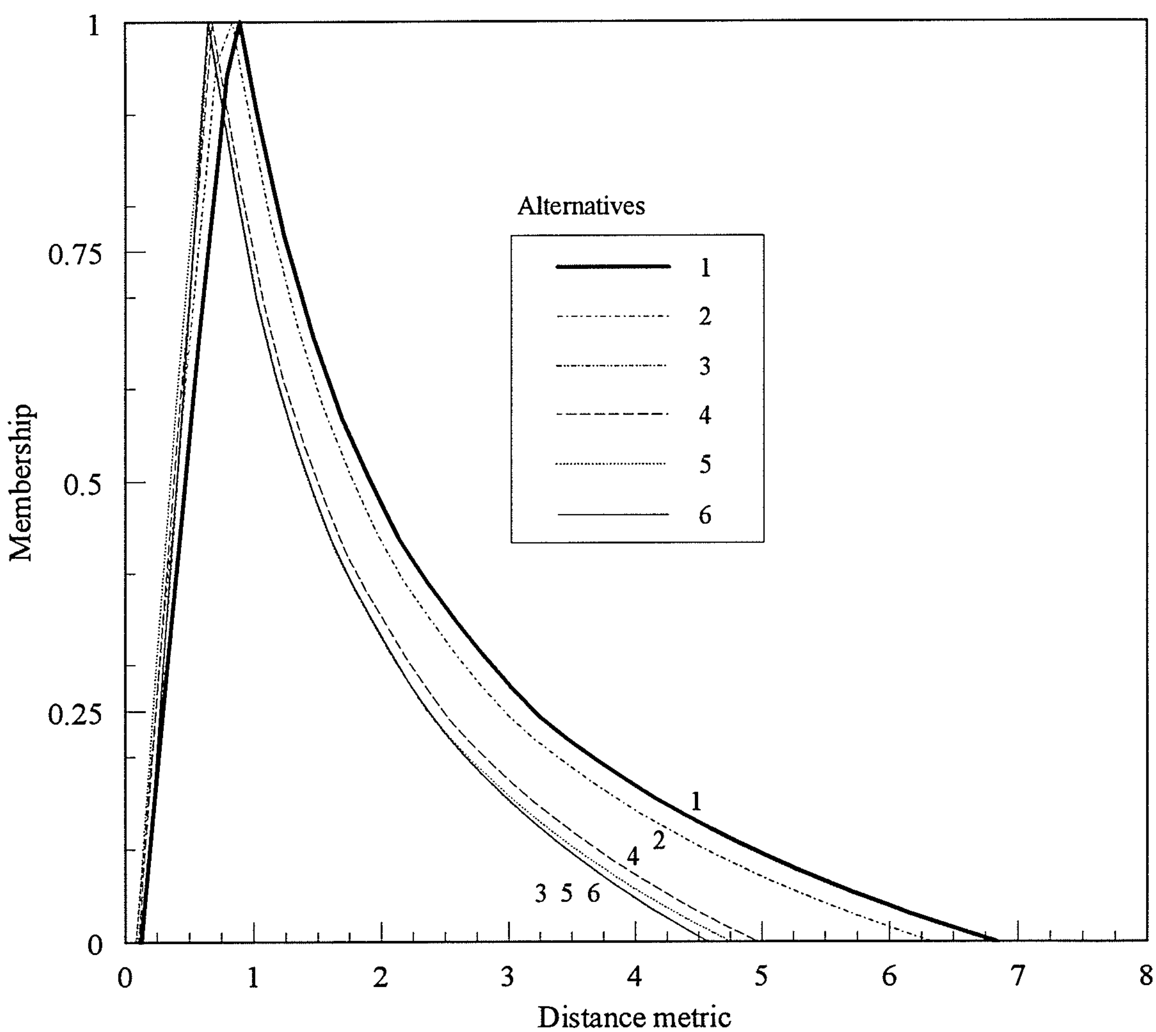

Figure 46. Fuzzy distance metrics for *S1*.

Simonovic (1989) provides (nonfuzzy) selection of subjective criteria values by 3 decision makers on the range of {1,2,3,4,5}. They agree completely on a few criteria values for certain alternatives, but they generally (at least partially) disagree on most subjective criteria values (Table 18).

Table 18. Nonfuzzy subjective criteria values, criteria 3 to 8, decision makers *A*,*B*,*C*.

Alt	A						B						C					
1	1	2	3	2	5	3	2	2	2	3	4	3	3	3	3	3	4	2
2	2	2	4	2	5	3	3	2	3	3	5	3	3	2	4	3	3	4
3	5	4	5	5	3	4	5	4	4	5	3	4	5	4	4	4	3	5
4	5	3	5	4	4	2	4	3	4	4	4	3	4	3	5	4	4	3
5	5	4	5	4	4	2	4	4	5	4	4	3	4	3	4	5	4	3
6	5	3	5	5	3	5	4	3	4	5	3	4	5	4	4	4	3	4

The fuzzy interpretation of these somewhat different opinions is based on the fuzzy definitions of subjective criteria values given earlier (Figure 41). For instance, the three opinions for alternative 3, criteria 4 are {4,4,4}. One can assume that this criteria value is quite certain about 4. There are many combinations of partial agreement/disagreement such as {3,3,4} which results in more vague membership functions. There is also an instance where all three opinions differ {1,2,3}. This difference of opinion has been interpreted as conflicting in nature. All three examples can be found below in Figure 47.

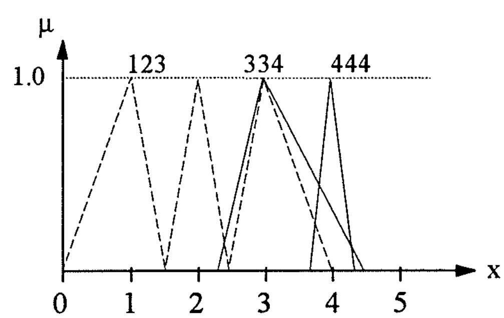

Figure 47. Collective criteria value definitions.

After redefining the fuzzy interpretation of the subjective criteria values to represent the combined opinions of the three decision makers, fuzzy distance metrics can be found. The difference in rankings between the original fuzzy interpretation, given by decision maker *A*, and the rankings realized from the collective opinion (using the **WCoG** measure to rank alternatives) is quite apparent (Table 19). All of the ranks vary except for alternatives {1,2} which are consistently poor, comparatively. Amazingly, by adding the element of collaboration between decision makers, alternative 4 changed from a rank of 4 (when rated by decision maker *A* only) to the highest rating among alternatives for the collective opinion case!

Table 19. WCoG ranks for *S1* base and test cases, *q*={1,2,3}.

Alt	A			A,B,C		
	1	2	3	1	2	3
1	5	5	5	3	4	4
2	6	3	3	5	5	5
3	3	6	6	4	3	6
4	4	4	4	6	6	3
5	2	2	2	2	2	2
6	1	1	1	1	1	1

7.6 Decision Support Module for the Fuzzy Compromise Approach

7.6.1 Module Description

To demonstrate the usefullness of this fuzzy compromise approach, a module has been added to the prototype collaborative planning decision support system. It includes data input from a variety of formats, parametric control over the brute force search algorithm, and flexible evaluation of alternative rankings. Figure 48 is a display of windows, including plots of different fuzzy sets, for the fuzzy compromise approach application. The windows in Figure 48 show the main components of the *Fuzzy Compromise Support System* module, generally moving from upper left to lower right.

The main components are:

1. Data input (upper left)
2. Distance metric calculation (right)
3. Ranking of alternatives (bottom)

The data input component allows fuzzy sets to be developed and altered for all the different sources of parametric uncertainty in the distance metric model. The distance metric calculation component is the control center for parametric control over the algorithms which optimize the distance metric calculation process. The component for producing alternative rankings allows access to 2 ranking measures, discussed earlier. Together they provide an application for including various sources of uncertainty in the decision process with the fuzzy compromise approach.

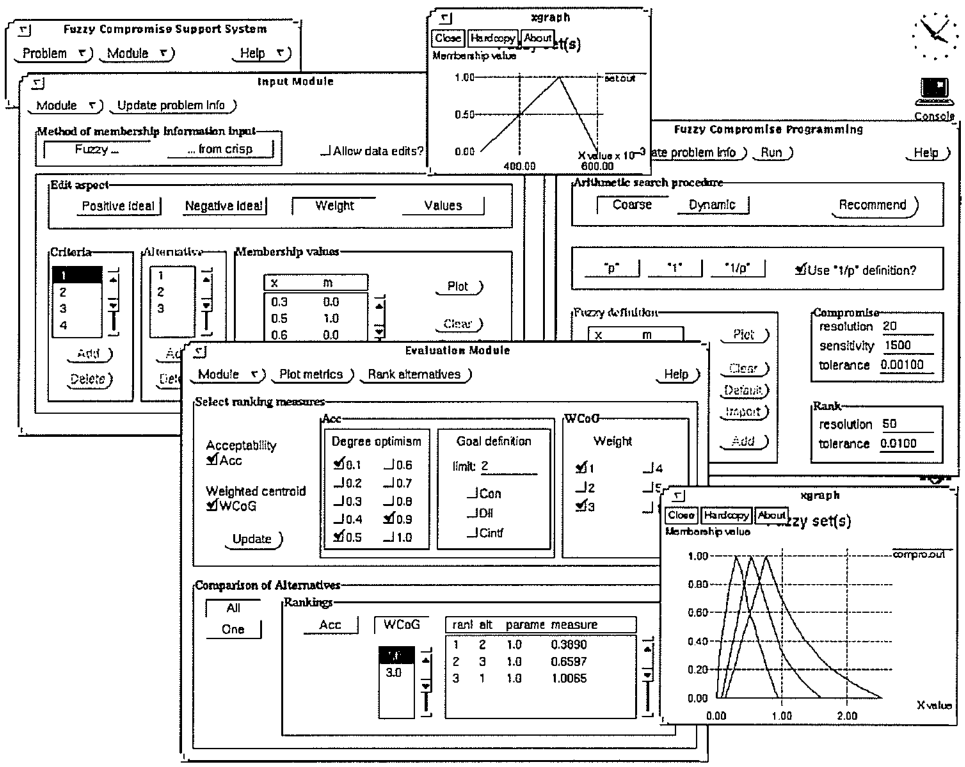

Figure 48. Display of fuzzy compromise approach application.

Fuzzy sets describe all of the parameters in the distance metric calculation. This includes: weights, criteria values, and the criteria ideals (both positive and negative). To allow flexible definition of fuzzy information, fuzzy sets are defined by points - assuming linear interpolation between points. The number of points are not limited. This allows the types of fuzzy sets such as conflicting opinions. A triangular fuzzy set is plotted from the data in a window near the data input component window.

To calculate distance metrics, there are a number of parameters which help to control the accuracy of the search algorithms used to calculate the fuzzy distance metrics. They are:

- Choice of coarse search (assuming unimodal) and general brute force,
- Distance metric exponent fuzzy definition,
- Resolution (number of search points on the distance metric fuzzy set),
- Sensitivity (controls the default number of search points on the fuzzy sets to be combined).
- Tolerance (controls the level of detail when searching for shape details on the distance metric fuzzy set),

Default values are automatically given. Use of custom values may produce better results for particular problems, but the optimal choice of parameter values is not always intuitive. Since a number of fuzzy arithmetic operations must be performed, the accuracy of a distance metric can be very dependent on the selection of search parameters. The most important attributes to preserve are modal values and other high membership values. The range of distance values will always be preserved, but the location and preservation of modal values can be difficult for the general brute force - even when using the dynamic search technique. A fuzzy distance metric is plotted in the lower part of the display (Figure 48).

To rank alternatives, both the centroid measure (**WCoG**) and the acceptability measure (**Acc**) are made available. Either or both measures can use a variety of parametric settings from which to generate rankings. Since different parametric settings can generate different rankings, sets of rankings can be displayed - listed according to ranking measure, or according to alternative. For example, if 5 different ranking measures are used (3 parametric settings for **Acc** and 2 for **WCoG**), listing ranks according to alternative will list the rank of a specified alternative for each of the selected measures. Listing ranks according to measure will list the ordering of alternatives for a specified measure. The evaluation module window in Figure 48 lists ranking according to measure, such that the alternatives are listed in order of rank for the selected **WCoG** measure with a weight of 1.0 .

7.6.2 Collaborative Decision Support Role of Fuzzy Evaluations

By allowing direct control over the definitions of fuzzy sets, stakeholders are able to experiment with different fuzzy definitions for parameter uncertainties. This promotes a better understanding of consequences for changes in accuracy when viewing the resulting rankings. One (hypothetical) benefit is that, on observing that there is no change in alternative ranking from improved accuracy of measuring water levels - there is no justified reason to use additional environmental impact assessment budget on improving the accuracy of water level measurements.

Each stakeholder brings a unique flavour to the planning process, in terms of their level of risk aversion and their interpretation of uncertainties in alternative performance. The interactive feedback to stakeholders about potential changes in ranking from different perceptions of uncertainty (ie. levels of risk aversion), may explain many idiosyncrasies in the opinions of different stakeholders.

Overall, the role of a fuzzy evaluation of alternatives is to promote stakeholder understanding of relative performance of alternatives, given multiple sources of uncertainty. The fuzzy compromise approach presented here is an example of a technique which can be relatively transparent and intuitive, and allows direct control by stakeholders.

7.7 Summary

The 3 case studies demonstrate many of the characteristics of fuzzy compromise programming as a MCDM technique. The Tisza River example showed consistency of results, compared to the ELECTRE method, without need for sensitivity analysis. Fuzzy compromise programming provided a degree of dominance in the ordering, for both the **WCoG** and **Acc** ranking measures. In the Tucson area example, a high level sensitivity analysis on the degree of risk aversion was used to effectively sort through a large number of alternative combinations in system management. The technique demonstrated its ability to provide intuitive decision information for different behavioural patterns of decision makers. Finally, the Yugoslavian example was used to explore different interpretations in creating membership functions for the inputs. System S2 was used to demonstrate sensitivity of rankings due to changes in the interpretation of distance metric exponent and criteria weight membership functions. System S1 follows by showing the effects on rankings from including other opinions about subjective criteria values. Multiple decision maker input demanded considerations for generating collective opinions.

More decision support information was made available by fuzzy compromise programming, compared to ELECTRE and compromise programming. Other MCDM methods have not been compared, such as Saaty's popular AHP method (Saaty, 1980) which uses pairwise comparisons to generate dominance relationships. A direct comparison with AHP is difficult because AHP relies on subjective comparisons and no criteria values are needed.

Fuzzy compromise programming has a number of comparative advantages over traditional (nonfuzzy) MCDM techniques. The most important is the direct, and often intuitive incorporation of vague and imprecise forms of uncertainty to the decision making process. In real decisions, many of the criteria are subjective in nature. By their very nature, subjective criteria are fuzzy. By allowing a degree of fuzziness, more realism is added to the evaluation without compromising on the technique's ability to disseminate alternative preferences. Similar observations can be made about criteria weights and the decision maker's interpretation of degree of compensation between criteria (p). All of which possess sufficient vagueness and imprecision to warrant scepticism when using traditional MCDM techniques.

By extending the compromise programming technique to a fuzzy approach, the differences in the distance metrics result in an adaptation of compromise programming from relative alternative desirability to relative desirable robustness of alternatives. The assessment of sensitivity to degree of risk aversion allows alternatives to be chosen which are robust to the type of decision maker(s), in addition to the more traditional robustness which is robustness to both criteria emphasis and criteria measurement.

At a time when group decision making is more common than single decision maker choices, fuzzy compromise programming facilitates collaborative exploration of the available alternatives and their associated risks. Collective opinions are incorporated by increasing (decreasing) the fuzziness of the inputs, and by locating ranges or multiple points of particular interest. Fuzzy membership functions are able to process this kind of information, and are also able to present it effectively and intuitively.

Numerical application of a fuzzy compromise approach can at best be intensive, and at worst be convoluted. The computational demands, however, have not been discussed in detail because any discussion of computational abilities for computers is dated as soon as it is written! Certainly, the method is computationally intensive, but that should not be a limiting factor.

Chapter 8 Conclusions

Given the problem of producing more effective proposals for licensing, in a climate of increased emphasis on sustainability ethics, and pressure to accommodate many traditional externalities, proponents of water resource development may consider adopting a framework of integration and cooperation with stakeholders. This dissertation explored the economic interpretation and ecosystem goals of water resources management, for the purpose of identifying a suitable framework from which to pursue effective planning of proposals. Consensus sustainability was identified as a potential candidate for the underlying decision paradigm, and a collaborative planning approach was identified as a candidate for collectively resolving conflicts in the early stages of planning.

A prototype decision support system, dubbed a Collaborative Planning Support System (CPSS), has been developed to demonstrate the possible implementation of integrated support for conceptual planning of hydropower development. Data from the Wuskwatim Lake case study in northern Manitoba was used as the basis for the data base given in chapter 5 (grounded approach for selecting evaluation criteria), and available spatial data for the Wuskwatim Lake area was used to develop GIS-based decision support for exploration of potential alternatives. Interviews with a senior design engineer at Manitoba Hydro produced the expert system application which provides support for design of a dam and generating station, to supplement the exploration of alternatives.

The 3 modules of the CPSS (criteria selection, exploration of alternatives, and alternative evaluation) are viewed as the basic problems to be addressed in any collaborative planning framework. In selecting criteria, the decision support approach attempted to address concerns of multiple value systems and representations of a decision problem when a group of stakeholders are involved. By exploring alternatives using online support from GIS and ES, users of the CPSS are able to experiment and visualize marginal differences between technical options. Finally, fuzzy evaluation of alternatives allows decision makers to incorporate multiple sources of uncertainty in the decision-making process, in a reasonably intuitive and straightforward manner. All attempts have been made in the development of the CPSS prototype to allow access and flexible control by stakeholders in all aspects, in a structured fashion. Although the CPSS is certainly not in a commercial form, it is meant to demonstrate the potential of integrated decision support for complex group planning issues.

8.1 Contributions

This dissertation may be considered an anomaly in the area of civil engineering. An engineering thesis does not normally cover such a broad area, with so little basic research in the content. But, then again, the pursuit of integrated planning for improved sustainability in management of water resources can be considered a very specific and challenging field, one that may not be properly exploited as yet. From this perspective, the dissertation makes several contributions. They are systematically outlined below. The last to be discussed, the fuzzy decision analysis technique, should be considered the most tangible contribution of this thesis.

8.1.1 Application of Integrated Decision Support for Sustainability in Managing Water Resources

Overall, appropriate tools for decision support were sought to achieve improved sustainability in managing water resources. The integration of disciplines and perspectives within decision support systems for the purpose of improved sustainability is an application which is quite unique, still, but is a very active research topic at this time. Issues in sustainability were examined, and tools were chosen based on need. In other words, "getting the right tool for the job" was used to justify the use of expert systems, GIS, object-oriented modelling, and fuzzy sets as appropriate tools for the types of issues which inhibit ecologically sound decisions. These tools were integrated for the purpose of collaborative group decision support, an application with a limited number of examples in the literature.

8.1.2 Identification of a Framework for Collaborative Planning

As one approach for achieving sustainability through better decision-making, a collaborative planning framework was defined using a systems approach. Many of the existing efforts to achieve sustainability are through more accurate economic evaluation or more holistic indicators. Pursuit of a collaborative planning concept for sustainability, based on the notion of consensus as an appropriate metric for sustainability, is a contribution which is quite unique in the literature.

The systems approach for describing the collaborative planning framework describes the decision process as iterative, and experimental - driven by different forms of feedback to the stakeholders/participants in the process. The entire set of tools and processes in the collaborative planning approach attempt to promote better decisions from decision makers. Model analysis and decision tools provide feedback, not recommendations.

8.1.3 Introduction of a Grounded Approach for Interaction with Stakeholder Value Systems

A new technique was introduced for selecting appropriate criteria for judging alternatives. In many instances, participants may have nontechnical backgrounds. Selection of traditional criteria and relevant weights to assign to the criteria, may be difficult. Also, there is very little control over undue bias in the selection of the weights. A novel technique has been developed which elicits input from decision participants in the form of elements or facts about the physical system which are deemed important by individuals. The method loosely follows the approach of grounded theory, from the field of qualitative sociological research, which attempts to induce abstract concepts and processes from the most basic and reproducible observations. By selecting basic facts about a system, the criteria selection technique implies links to a value system for each individual. Although further elicitation is not made to "map" the value systems of participants, the combination of facts that are selected may suggest a great deal of value information - not entirely shrouded by strategic moves. The method makes a conscious effort to produce relevant criteria without placing weights of relative importance on each of the decision-making participants.

The method relies on expertise and experience to generate a relevant set of criteria from the grounded facts. It assumes a certain context in which facts are being selected, and uses a knowledge base to trigger the inclusion of certain criteria. The complexity of the knowledge base is dependent on available expertise. Intuitiveness in the choice of criteria may not always be produced, but knowledge bases are inherently transparent. Participants are able to see which rules triggered which responses.

The method also relies on an aggregation mechanism to generate appropriate weights for the criteria. For this introduction to the method, a simple sum is maintained for each selected fact, and also for each triggered criteria. The weight is then a measure of how often available criteria are triggered by participants. The aggregation mechanism may be adapted for particular circumstances. For example, participants may be asked to give either ordinal or cardinal ranks to signify relative importance of facts. Criteria weights could then be normalized with respect to both individuals and the entire decision-making groups.

8.1.4 Introduction of a Fuzzy Multicriteria Decision Analysis Technique

The field of multicriteria decision-making (MCDM) has been actively pursuing fuzzy methods of decision analysis as an approach for dealing with various kinds of uncertainty. A contribution has been made by adapting a traditional MCDM technique, compromise programming (for discrete alternatives), using fuzzy inputs. The resulting fuzzy compromise approach expands on earlier attempts to fuzzify a single criteria by allowing all inputs (criteria values, criteria weights, ideal criteria values, and the form of distance metric) to be fuzzy. The approach is unique in that the rating of alternatives is in a fuzzy form, preserving a great deal of information about the expected bahaviour of an alternative and also the extremes of possible behaviour interpretation. Two ranking measures were explored for the purpose of defuzzifying the alternative ratings to produce a direct ranking of alternatives. One of these ranking measures, Acc, was developed specifically for this fuzzy compromise approach, as an extension to another ranking measure found in the literature. It's unique properties allow participants to examine possible degrees of risk aversion when ranking alternatives.

8.2 Opportunities for Future Research

There are a number of opportunities for further research to be generated from this dissertation on collaborative decision support. They are outlined below.

8.2.1 Knowledge Organization

The systems approach used to describe the decision process in a collaborative group setting relied heavily on the concept of balancing feedback coming from knowledge base elements. The effectiveness of knowledge bases to supply balancing feedback in the form of advice or past experience is dependent on the knowledge organizational scheme. It is possible to organize knowledge in different ways to suit the needs of different purposes. The implications and effectiveness of knowledge base structures in expert systems needs to be examined more thoroughly.

8.2.2 Model Integration

A limited number of models were integrated in the prototype collaborative planning decision support system. The effectiveness of decision support in potential future field studies with multiple participants may be substantially augmented by a concentrated effort to supply more technical expertise and analytical capacity. The use of object-oriented modelling is seen as a likely vehicle for managing augmented modelling support.

8.2.3 Fuzzy Information Elicitation

The fuzzy compromise approach presented earlier may provide powerful decision analysis benefits by encoding a plethora of information about parameter imprecision and participant perception in a relatively transparent MCDM method. The methods by which the fuzzy sets are generated is an area of research which should be examined. There are many techniques for encoding input to fuzzy sets. They can be classified as parametric and nonparametric. Parametric techniques closely follow traditional probabilistic methods. Nonparametric techniques can be further classified into horizontal and vertical methods. Different techniques for elicitation of fuzzy criteria values based on field measurements or model calculations should be treated differently from subjective opinions.

8.2.4 Case Study Applications

The thesis, as presented, contains a decision support system which remains largely untested in "live" scenarios with "real" participants. A group of interested stakeholders (from Manitoba Hydro, Department of Fisheries and Oceans, and the University of Manitoba) were represented throughout the research, but no live interaction with the CPSS was undertaken. This was due to time and budgetary constraints.

A series of applications would be extremely beneficial for understanding many of the intangible issues involved in stakeholder participation. This would also allow decision support tools to undergo more rigorous preparation. Preparation of available data, use of appropriate models, and inclusion of relevant expertise, are all parts of the technical management of a DSS which are specific to a case study. Another benefit of preparing decision support tools for applications is that, if properly managed, a library of model components and knowledge bases can be developed. Each knowledge base, for example, would pattern a very specific domain and would be designed to be compatible (in form) with other domain knowledge bases.

Chapter 9 References

Amir, S. 1992. The Environmental Cost of Sustainable Welfare. Resources for the Future discussion paper, Washington DC.

Anderson, L. 1993. Toward a Complete Economic Theory of the Utilization and management of Recreational Fisheries. *J. Environmental Economics and Management*, **24**:272-295.

Andrews, C. 1992. Spurring Inventiveness by Analyzing Tradeoffs: a public look at New England's electricity alternatives. *Environmental Impact Assessment Review*, **12**:185-210.

Andrews, C. 1991. Building Consensus on a shoestring: the efficacy of packaged models in joint fact-finding. In Klosterman, R. (ed), *Proceedings of the 2nd Int. Conference on Computers in Urban Planning and Urban Management*. Public Inst. for Computer Aided Planning, University of Akron, OH.

Andrews, C. 1990. Taking a Public - and Data-intensive - look at New England's Electric Power Planning Alternatives. In *Proceedings of the EPRI Information Technology for the Power Industry Conference*, Electric Power Research Institute, Palo Alto, CA.

Bacow, L., and M. Wheeler. 1984. *Environmental Dispute Resolution*. Plenum Press, New York.

Bales, R., and F. Strodtbeck. 1951. Phases in Group Problem Solving. *J. Abnormal Social and Psychology*, **46**:485-495.

Bardossy, A., and L. Duckstein. 1992. Analysis of a Karstic Aquifer Management Problem by Fuzzy Composite Programming. *Water Resources Bulletin*, **28**(1):63-73.

Barrett, S. 1992. Economic Growth and Environmental Preservation. *J. Environmental Economics and Management*, **23**:289-300.

Baumol, W., and W. Oates. 1988. *The Theory of Environmental Policy*. Cambridge University Press, Cambridge.

Beatty, K., and J. Pierce. 1976. Representation and Public Involvement in Water Resource Politics: a Comparison of Six Participant Types. *Water Resources Bulletin*, **12**(5):1005-1017.

Bellman, R., and L. Zadeh. 1970. Decision Making in a Fuzzy Environment. *Management Science*, **17**:141-164.

Benayoun, R., B. Roy, and B. Sussmann. 1966. ELECTRE: Une méthode pour quider le choix en présence de points de vue multiples, Note Trav. 49, Dir. Sci., *Soc. Econ. Math. Appl.*, Paris.

Bender, M., G. Johnson, and S. Simonovic. 1994. Sustainable Management of Renewable Resources: A Comparison of Alternative Decision Approaches. *Int. J. Sustainable Development and World Ecology*, **1**(2):77-88.

Bender, M., C. Katopodis, and S. Simonovic. 1992. A Prototype Expert System for Fishway Design. *Environmental Monitoring and Assessment*, **23**:115-127.

Bender, M., and S. Simonovic. 1995. Proponent and Stake-holder Interaction in Collaborative Water Resources Project Planning. In S. Simonovic, Z. Kundzewicz, D. Rosbjerg, and K. Takeuchi (ed). *Modelling and Management of Sustainable Basin-Scale Water Resource Systems*. IAHS publication 231.

Bender, M., and S. Simonovic. 1994. Collaborative Decision Support in Environmental Conflict Resolution. Proceedings: *UNESCO International Symposium on Water Resources Planning in a Changing World*, Karlsruhe Germany.

Bender, M., and S. Simonovic. 1996a. A Systems Approach for Collaborative Decision Support in Water Resources. *Proceedings of the IEEE Society on Social Implications of Technology ISTAS '96: technical expertise and public decisions*, Princeton, NJ.

Bender, M., and S. Simonovic. 1996b. Consensus as the Measure of Sustainability. *North American Water and Environment Congress '96*, Anaheim, CA.

Bender, M., and S. Simonovic. 1996. A Fuzzy Compromise Approach to Water Resources Systems Planning under Uncertainty. *Water Resources Research*, submitted for review.

Beveridge, G, and R. Schechter. 1970. *Optimization: Theory and Practice*. McGraw-Hill, New York.

Bidwell, R. 1992. Sustainability: the Link Between Conservation and Economic Benefits. *Environmental Impact Assessment Review*, **12**:37-47.

Bingham, G. 1986. *Resolving Environmental Disputes: A Decade of Experience*. The Conservation Foundation, Washington DC.

Bishop, R., and T. Heberlein. 1990. The Contingent Valuation Method. In Johnson, R.L., and G.V. Johnson, ed., *Economic Valuation of Natural Resources*.

Blin, J. 1974. Fuzzy Relations in Group Decision Theory. *J. Cybernetics*, **4**(2):17-22.

Bogardi, J., and L. Duckstein. 1992. Interactive Multiobjective Analysis Embedding the Decision Maker's Implicit Preference Function. *Water Resource Bulletin*, **28**(1):75-88.

Bohm, R., T. Essenburg, and W. Fox. 1993. Sustainability of Potable Water Services in the Philippines. *Water Resources Research*, **29**(7):1955-1963.

Bortolan, G., and R. Degani. 1985. A Review of some Methods for Ranking Fuzzy Subsets. *Fuzzy Sets and Systems*, **15**:1-19.

Brill, E. 1979. The Use of Optimization Models in Public-sector Planning. *Management Science*, **25**(5):413-422.

Brill, E., J. Liebman, and C. ReVelle. 1976. Equity Measures for Exploring Water Quality Managment Alternatives. *Water Resources Research*, **12**(5):845-850.

Brookshire, D., and K. Smith. 1987. Measuring Recreation Benefits: Conceptual and Empirical Issues. *Water Resources Research*, **23**(5):931-935.

Brookshire, D., L. Eubanks, and C. Sorg. 1986. Existence Values and Normative Economics: Implications for Valuing Water Resources. *Water Resources Research*, **22**(11):1509-1518.

Bui, T. 1987. *Co-op: a Group Decision Support System for Cooperative Multiple Criteria Group Decision Making*. Springer-Verlag, New York.

Burkardt, N., B. Lamb, and T. Waddle. 1994. Technical Understanding in Successful Environmental Negotiation. Proceedings: *ASCE Water Resources Planning and Management Division*.

Burrough, P. 1993. *Principles of Geographical Information Systems for Land Resources Assessment*. Clarendon Press, Oxford.

Burton, P. 1993. Intertemporal Preferences and Intergenerational Equity Considerations in Optimal Resource Harvesting. *J. Environmental Economics and Management*, **24**:119-132.

Cairns, J., and T. Crawford. 1991. *Integrated Environmental Management*. Lewis Publ. Inc., Michigan.

Cairns, J., and T. Crawford, and H. Salwasser, ed. 1994. *Implementing Integrated Environmental Management*. Lewis Publ. Inc., Michigan.

Carlsson, C. 1982. Tackling an MCDM problem with the help of some results from fuzzy set theory. *European J. Operational Research*, **10**:270-281.

Carlsson, C., and R. Fuller. 1996. Fuzzy Multiple Criteria Decision Making: recent developments. *Fuzzy Sets and Systems*, **78**:139-153.

Carroll, J. ed. 1991. *Designing Interaction: Psychology at the Human-Computer Interface*. Cambridge University Press, Cambridge.

Carson, R., and R. Mitchell. 1993. The Value of Clean Water: the Public's Willingness to Pay for Boatable, Fishable and Swimmable Quality Water. *Water Resources Research*, **29**(7):2445-2454.

Chadderton, R., R. Traver, and J. Rao. 1992. Hydraulic Modelling Support for Conflict Analysis: the Manayunk Canal Revisited. *Water Resource Bulletin*, **28**(6):1069-1076.

Ciriancy-Wantrup, S. 1968. *Resource Conservation: Economics and Policies*. U. of California Berkeley Division of Agricultural Sciences.

Clark, C. 1976. *Mathematical Bioeconomics: the Optimal Management of Renewable Resources*. Wiley and Sons, New York.

Clark, E., G. Bingham, and S. Orenstein. 1991. Resolving Water Disputes: Obstacles and Opportunities. RESOLVE report no. 23, *World Wildlife Fund* and *The Conservation Foundation*, Washington DC.

Coase, R. 1960. The Problems of Social Cost. *J. Law and Economics*, **3**:1-44.

Cohon, J., and D. Marks. 1975. A Review and Evaluation of Multiobjective Programming Techniques. *Water Resources Research*, **11**(2):208-219.

Conrad, J. and C. Clark. 1987. *Natural Resource Economics*. Cambridge University Press, Cambridge.

Cooter, R., and T. Ulen. 1988. *An Economic Theory of Property, in Law and Economics*. S. Foresman & Co, Glenview, Illinois.

Costanza, R., B. Norton, and B. Haskell. 1992. *Ecosystem Health: New Goals for Environmental Management*. Island Press, Washington, DC.

Coughlan, B., N. Burkardt, and D. Fulton. 1993. Assessing the "Need to Negotiate" in FERC Licensing Consultations: A Study of Two Hydropower Projects. *Environmental Impact Assessment Review*, **13**:331-351.

Creighton, J., J. Priscoli, and C. Dunning. 1983. Public Involvement Techniques: a Reader of 10 years Experiences at the Institute of Water Resources. *Institute for Water Resources research report* 87-R-1, US Army Corps of Engineers, Ft. Belvoir, Virginia.

Crosby, N. 1987. Citizens Panels - a New Democratic Process for Risk Management. Proceedings: *American Society for Public Administration.*

Dasgupta, S. 1976. Local Attitudes and Water Resources Development: a Canadian Case Study. *Water Resources Bulletin*, **12**(1):27-37.

David, L, and L. Duckstein. 1976. Multi-Criterion Ranking of Alternative Long-Range Water Resource Systems. *Water Resources Bulletin*, **12**(4):731-754.

Demsetz, H. 1967. Toward a Theory of Property Rights. *American Economic Review*, **57**:347-359.

DeVantier, B., and A. Feldman. 1993. Review of GIS Applications in Hydrologic Modelling. *ASCE J. Water Resources Planning and Management*, **119**(2):246-261.

Dougherty, D. 1994. Building Groundwater Optimizers that Respond to the needs of Decision-makers. Proceedings: *ASCE Water Resources Planning and Management* division annual conference.

Dubois, D., and H. Prade. 1978??. Operations on Fuzzy Numbers. *Int. J. Systems Science*, **9**(6):613-626.

Duckstein, L., and S. Opricovic. 1980. Multiobjective Optimization in River Basin Development. *Water Resources Research,* **16**(1):14-20.

Dyer, J., P. Fishburn, R. Steuer, J. Wallenius, and S. Zionts. 1992. Multiple Criteria Decision Making, Multiattribute Utility Theory: the next 10 years. *Management Science*, **38**(5):645-654.

Eberhardt, A. 1994. Lake Ontario Regulation Utilizing an Expert System Approach Constrained by Interest Satisfaction Relationships. Proceedings: *ASCE Water Resources Planning and Management Division* conference.

FEARO. 1986. *Initial Assessment Guide*. Federal Environmental Assessment Review Office, Canada Department of Supply and Services.

Fang, L., and K. Hipel. 1988. The Graph Model Approach to Environmental Conflict Resolution. *J. Environmental Management*, **27**:195-212.

Fang, L., K. Hipel, and D. Kilgour. 1993. *Interactive Decision Making: the Graph Model for Conflict Resolution*. Wiley and Sons, Inc., New York.

Fang, L., K. Hipel, and D. Kilgour. 1989. Conflict Models in Graph form: Solution Concepts and their Interrelationships. *European J. Operational Research*, **41**:86-100.

Fedra, K., L. Winkelbauer, and V. Pantulu. 1991. *Expert Systems for Environmental Screening: an application in the Lower Mekong basin.* Int. Institute for Applied Systems Analysis, Laxenburg, Austria.

Felix, R. 1994. Relationships between Goals in Multiple Attribute Decision Making. *Fuzzy Sets and Systems*, **67**:47-52.

Fiorino, D. 1990. Dimensions of Negotiated Rule-Making: Practical Constraints and Theoretical Implications, in Mills, M.K. ed, *Conflict Resolution and Public Policy*. Greenwood Press, New York.

Fisher, R., W. Ury, and B. Patton. 1991. *Getting to Yes: Negotiating Agreement Without Giving In*. Penguin Book, New York.

Fukuyama, K., D. Kilgour, and K. Hipel. 1994. Systematic Policy Development to Ensure Compliance to Environmental Regulations. *IEEE Transactions on Systems, Man, and Cybernetics*, **24**(9):1289-1305.

Geller, E. 1994. The Human Element in Integrated Environmental Management. In Cairns, J, T.V. Crawford, and H. Salwasser, ed., *Implementing Integrated Environmental Management*.

Gershon, M., and L. Duckstein. 1983. Multiobjective Approaches to River Basin Planning. *J. Water Resources Planning and Management*, ASCE, **109**(1):13-28.

Gershon, M., L. Duckstein, and R. McAniff. 1982. Multiobjective River Basin Planning with Qualitative Criteria. *Water Resources Research*, **18**(2):193-202.

Glaser, B. 1978. *Theoretical Sensitivity*. Sociology Press, Mill Valley, California.

Glaser, B., and A. Strauss. 1967. *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Aldine de Gruyter, New York.

Goicoichea, A., E. Stakhiv, and F. Li. 1992. Experimental Evaluation of Multiple Criteria Decision Models for Application to Water Resources Planning. *Water Resources Bulletin*, **28**(1):89-102.

Goicoichea, A., D. Hansen, and L. Duckstein. 1982. *Multiobjective Decision Analysis with Engineering and Business Applications*. Wiley and Sons, New York.

Goldberg, S., E. Green, and F. Sander. 1985. *Dispute Resolution*. Little Brown & Co., Toronto.

Goodland, R., H. Daly, S. El Serafy, and B. von Droste. 1991. *Environmentally Sustainable Economic Development: Building on Brundtland*. UNESCO, Paris.

Haimes, Y. 1992. Sustainable Development: A Holistic Approach to Natural Resource Management. *IEEE Transactions on Systems, Man, and Cybernetics*, **22**(3):413-417.

Haimes, Y. 1977. *Hierarchical Analyses of Water Resource Systems*. McGraw-Hill, New York.

Haimes, Y., and W. Hall. 1974. Multiobjectives in Water Resource Systems Analysis: the Surrogate Worth Trade-off Method. *Water Resources Research*, **10**(4):615-624.

Hamalainen, R., A. Salo, and K. Poysti. 1992. Observations about Consensus Seeking in a Multiple Criteria Environment. *IEEE Proceedings: Int. conference on Systems Sciences*, Kauai, Hawaii.

Harboe, R. 1992. Multiobjective Decision Making Techniques for Reservoir Operations. *Water Resources Bulletin*, **28**(1):103-110.

Hardin, G. 1968. The Tragedy of the Commons. *Science*, **162**:1243-1248.

Harpman, D., E. Sparling, and T. Waddle. 1993. A Methodology for Quantifying and Valuing the Impacts of Flow Changes on a Fishery. *Water Resources Research*, **29**(3):575-582.

Herrin, J., and A. Whitlock. 1992. Interfacing with the Public on Water-Related Issues: what TVA is doing. Proceedings: *Water Forum*, Baltimore Maryland.

Hipel, K. 1992. Multiple Objective Decision Making in Water Resources. *Water Resources Bulletin*, **28**(1):3-12.

Hipel, K., J. Radford, and L. Fang. 1993. Multiple Participant Multiple Criteria Decision Making. *IEEE Transactions on Systems, Man, and Cybernetics*, ??.

Hobbs, B., V. Chankong, W. Hamadeh, and E. Stakhiv. 1992. Does Choice of Multicriteria Method Matter? An Experiment in Water Resources Planning. *Water Resources Research*, **28**(7):1767-1779.

Howard, N. 1971. *Paradoxes of Rationality*. MIT Press, Cambridge.

Hydro-Quebec. 1993. *Grande-Baleine Complex Overview*. Hydro-Quebec, Canada.

Institution of Engineers. 1992. *Environmental Principles for Engineers*. National Committee on Environmental Engineering, Barton ACT, Australia.

Isreal, M., J. Lund, and G. Orlob. 1994. Cooperative Game Theory in Water Resources. Proceedings, *ASCE Water Resources Planning and Management* annual conference.

Iverson, D., and R. Aston. 1994. A new Role for Economics in Integrated Environmental Management. In Cairns, J., T.V. Crawford, and H. Salwasser, ed., *Implementing Environmental Management*.

Jain, R. 1976. Decision-making in the Presence of Fuzzy Variables. *IEEE Trans. Systems, Man, and Cybernetics*, **6**:698-703.

Janssen, R. 1992. *Multi-objective Decision Support for Environmental Management*. Kluwer Academic Publishers, Dordrecht Netherlands.

Johnson, C., R. Korol, and A. Perks. 1994. CSCE Guidelines for Civil Engineering Practice with a commitment to a Sustainable Future. *Canadian Civil Engineer*, **11**(5).

Johnson, S., R. Wolcott, and S. Aradhynla. 1990. Coordinating Agricultural and Environmental Policies Opportunities and Tradeoffs. *AEA Papers and Proceedings*, **80**(2):203-207.

Kacprzyk, J., and S. Orlovski (eds). 1987. *Optimization Models using Fuzzy Sets and Possibility Theory*. Reidel Publ. Co., Dordrecht.

Kandel, A. 1986. *Fuzzy Mathematical Techniques with Applications*. Addison-Wesley Publ. Co., Reading, Massechussetts, USA.

Kearns, K., C. Monahan, and A. West. 1995. Hydro Settlements: Providing an Opportunity for Agreement. *Hydro Review*, October.

Keeney, R. 1992. *Value-Focused Thinking: A Path to Creative Decisionmaking*. Harvard University Press.

Keeney, R., and H. Raiffa. 1974. A critique of formal analysis in public decision making. In Drake, A., R. Keeney, and P. Morse (ed.), *Analysis of Public Systems*, MIT Press, pp. 64-74.

Keeney, R., and E. Wood. 1977. An Illustrative Example of the use of Multiattribute Utility Theory for Water Resource Planning. *Water Resources Research*, **13**(4):705-712.

Keyes, A., and R. Palmer. 1992. Use of Interactive Simulation Environments for the Development of Negotiation Tools. Proceedings: *Water Forum*, Baltimore, Maryland.

Kilgour, D, L. Fang, K. Hipel. 1992. Game Theoretic Analyses of Enforcement of Environmental Laws and Regulations. *Water Resources Bulletin*, **28**(1):141-154.

Kilgour, D., K. Hipel, and L. Fang. 1987. The Graph Model for Conflicts. *Automatica*, **23**(1):41-55.

Kim, K., and K. Park. 1990. Ranking Fuzzy Numbers with Index of Optimism. *Fuzzy Sets and Systems*, **35**:143-150.

Kindler, J. 1988. On the Multiobjective Framework of Environmentally Sound and Sustainable Water Resources Management. *Water Resources Development*, **4**(2):117-123.

Klopatek, J., J. Krummel, J. Mankin, and R. O'Neill. 1983. A Theoretical Approach to Regional Environmental Conflicts. *J. Environmental Management*, **16**:1-15.

Knoblock, C. 1993. *Generating Abstraction Hierarchies: an Automated Approach to Reducing Search in Planning*. Kluwer Academic, Boston.

Ko, S., D. Fontane, and J. Labadie. 1992. Multi-objective Optimization of Reservoir Systems Operation. *Water Resources Bulletin*, **28**(1):111-127.

Kuhn, T. 1962. *The Structure of Scientific Revolutions*. University of Chicago Press.

Kuncheva, L. 1994. Pattern Recognition with a Model of Fuzzy Neuron using Degree of Consensus. *Fuzzy Sets and Systems*, **66**:241-250.

Lai, Y-J, T-Y Liu, and C-L Hwang. 1994. TOPSIS for MODM. *European J. Operational Research*, **76**:486-500.

Landry, M., D. Pascot, and D. Briolat, 1985. Can DSS evolve without changing our view of the concept of 'Problem'?, *Decision Support Systems*, Vol.1, No.1, pp.25-36.

Laskey, K., and P. Lehner. 1994. Metareasoning and the Problem of Small Worlds. *IEEE Transactions on Systems, Man, and Cybernetics*, **24**(11):1643-1652.

Lee, Y.W., M. Dahab, and I. Bogardi. 1994. Fuzzy Decision Making in Groundwater Nitrate Risk Management. *Water Resources Bulletin*, **30**(1):135-148.

Leung, Y. 1982. A Concept of a Fuzzy Ideal for Multicriteria Conflict Resolution. In M. Gupta, and E. Sanchez (ed.), *Fuzzy Information and Decision Processes*. Elsevier, New York.

Lewis, F. 1993. Decision-Aiding Software for Group Decision Making. In nagel, S. (Ed.), *Computer-Aided Decision Analysis: theory and applications*. Quorum Books, Westport Conneticut.

Liang, T. 1976. An LP-10 Model for Coordinating Multi-Group Inputs in Resource Planning. *Water Resources Bulletin*, **12**(3):601-624.

Liebman, J. 1976. Some Simple-Minded Observations on the Role of Optimization in Public Systems Decision-Making. *Interfaces*, **6**(4):102-108.

Lin, M-J. 1984. *Group Decision Making under Multiple Criteria - methods and applications*. Ph.D. dissertation, Kansas State University, Manhattan Kansas.

Lofland, J., and L. Lofland. 1971. *Analyzing Social Settings*. Wadsworth Publ., Belmont, California.

Lotfi, V., T. Stewart, and S. Zionts. 1991. An Aspiration-Level Interactive Model for Multiple Criteria Decision Making. *Computers and Operations Research*, ??.

Loucks, D.P. 1994. Sustainability: What does this mean for us and what can we do about it? Proceedings, Int. *UNESCO Symposium on Water Resources Planning in a Changing World*, Karlsruhe, Germany.

Loucks, D.P., and J. da Costa. 1989. *Decision Support Systems: Water Resources Planning*. Springer-Verlag, New York.

Loucks, D.P., J. Stedinger, and D. Haith. 1981. *Water Resource Systems Planning and Analysis*. Prentice-Hall Inc., London.

Lund, J. 1992. Benefit-Cost Ratios: Failures and Alternatives. *ASCE J. Water Resources Planning and Management*, **118**(1):94-105.

Lund, J., and R. Palmer. 1986. Subjective Evaluation: linguistic scales in pairwise comparison methods. *Civil Engineering Systems*, **3**:182-186.

Luce, D., and H. Raiffa. 1957. *Games and Decisions: Introduction and Critical Survey*. Wiley & Sons, New York.

Lussier, B., G. Mohr, and I. Goulter. 1989. Conflict Analysis of the Shoal Lake Subdivision. *Water Resources Bulletin*, **25**(1):111-116.

Madariaga, B., and K. McConnell. 1987. Exploring Existence Value. *Water Resources Research*, **23**(5):936-942.

Manitoba Hydro. 1990. *Manitoba Hydro Submission to the Public Utilities Board in Respect of Major Capital Projects*. Manitoba Hydro, Winnipeg, Canada.

Manitoba Hydro. 1989. *Perspective 2000*. Manitoba Hydro, Winnipeg, Canada

Manitoba Hydro, 1987. Report on Wuskwatim Generating Station: Stage III Concept Studies. Manitoba Hydro Generation and Transmission Planning Department report G&TPD 87-18E.

Manitoba Public Utilities Board. 1990. Report of the Public Utilities Board in respect of Major Capital Projects of Manitoba Hydro, Volume 1. Government of Manitoba, Department of Cooperative, Consumer and Corporate Affairs.

Mares, M. 1994. *Computation over Fuzzy Quantities*. CRC Press, London.

Marlatt, R., T. Hale, and R. Sullivan. 1993. Video Simulation as Part of Army Environmental Decision-making: Observations from Camp Shelby, Mississippi. *Environmental Impact Assessment Review*, **13**:75-88.

Miller, A. 1983. The Influence of Personal Biases on Environmental Problem-solving. *J. Environmental Management*, **17**:133-142.

Miller, A. 1985. Cognitive Styles and Environmental Problem-solving. *Int. J. of Environmental Studies*, **26**:21-31.

Mitchell, B., and D. Shrubsole. 1994. Reorienting to Achieve Sustainability in Canadian Water Management. *Canadian Water Resources Journal*, **19**(4):335-347.

Mittra, S., 1986. *Decision support systems: Tools and techniques*. John Wiley & Sons, New York, USA.

Moore, C. 1991. Corps of Engineers uses Mediation to Settle Hydropower Dispute. US Army Corps of Engineers, Institute for Water Resources Case Study 91-ADR-CS-6, Ft. Belvoir, VA.

Munda, G., P. Nijkamp, and P. Rietveld. 1995. Qualitative Multicriteria Methods for Fuzzy Evaluation Problems: an illustration of economic-ecological evaluation. *European J. Operational Research*, **82**:79-97.

Nagel, S. 1990. Multicriteria Dispute Resolution through Computer-aided Mediation. In Mills, M.K. ed., *Conflict Resolution and Public Policy*. Greenwood Press, New York.

Neely, W., R. North, and J. Fortson. 1976. Planning and Selecting Multiobjective Projects by Goal Programming. *Water Resources Bulletin*, **12**(1):19-25.

Netting, R. 1976. What Alpine Peasants Have in Common. *Human Ecology*, **4**:135-146.

Neuron Data Inc. 1993. *Nexpert Object: User's Guide*. Neuron Data, Palo Alto, California.

Neuron Data Inc. 1993. *Open Interface Elements: User's Guide*. Neuron Data, Palo Alto, California.

Nishizaki, I., and F. Seo. 1994. Interactive Support for Fuzzy Trade-off Evaluation in Group Decision Making. Fuzzy Sets and Systems, **68**:309-325.

Niu, W., J. Lu, and A. Khan. 1993. Spatial Systems Approach to Sustainable Development: A Conceptual Framework. *Environmental Management*, **17**(2):179-186.

Palmer, R., and J. Lund. 1985. Multiobjective Analysis with Subjective Information. *ASCE J. Water Resources Planning and Management*, **111**(4):399-416.

Palmer, R. , A., Keyes, and S. Fisher, 1993. Empowering Stakeholders Through Simulation in Water Resources Planning, in *Water Management for the '90s*, Edited by K. Hon, ASCE, 451-454.

Pedrycz, W. 1995. *Fuzzy Sets Engineering*. CRC Press, Boca Raton, Florida.

Pierce, J. 1992. Progress and the Biosphere: the Dialectics of Sustainable Development. *Canadian Geographer*, **36**(4):306-320.

Plate, E. 1993. Sustainable Development of Water Resources: A Challenge to Science and Engineering. *Water International*, **18**:84-94.

Poh, K., M. Fehling, and E. Horvitz. 1994. Dynamic Construction and Refinement of Utility-Based Categorization Models. *IEEE Transactions on Systems, Man, and Cybernetics*, **24**(11):1653-1663.

Priscoli, J. 1991. Environmental Ends and Engineering Means: becoming Environmental Engineers for the Nation and the World. National Resources Support Center, Institute for Water Resources working paper 91-ADR-WP-4, Ft. Belvoir, VA.

Priscoli, J. 1990. Public Involvement, Conflict Management, and Dispute Resolution in Water Resources and Environmental Decision Making. National Resources Support Center, Institute for Water Resources working paper 90-ADR-WP-2, Ft. Belvoir, VA.

Priscoli, J. 1989. Public Involvement, Conflict Management: means to EQ and Social Objectives. *ASCE J. Water Resources Planning and Management*, **115**(1):31-42.

Priscoli, J. 1988. Conflict Resolution in Water Resources: 2 404 general permits. *ASCE J. Water Resources Planning and Management*, .

Priscoli, J. 1975. Citizen Advisory Groups and Conflict Resolution in Regional Water Resources Planning. *Water Resources Bulletin*, **2**(6):1233-1243.

Regier, H., and Bronson, E. 1992. New Perspectives on Sustainable Development and Barriers to Relevant Information. *Environmental Monitoring and Assessment*, **20**:111-120.

Reitsma, R., A. Sautins, and S. Wehrend, 1993. RSS: A Construction Kit for Visual Programming of River Basin Models, in *Water Management for the '90s*, Edited by K. Hon, ASCE, 264-267.

Ribeiro, R. 1996. Fuzzy Multiple Attribute Decision Making: a review and new preference elicitation techniques. *Fuzzy Sets and Systems*, **78**:155-181.

Ribeiro, R., P. Powell, and J. Baldwin. 1995. Uncertainty in Decision-making: an abductive perspective. *Decision Support Systems*, **13**:183-193.

Rogers, P. 1969. A Game Theory Approach to the Problems of International River Basins. *Water Resources Research*, **5**(4):749-760.

Roots, E. 1992. Environmental Information - A Step to Knowledge and Understanding. *Environmental Monitoring and Assessment*, **20**:87-94.

Rumbaugh, J., M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen. 1991. *Object-oriented Modeling and Design*. Prentice Hall, Englewood Cliffs, New Jersey, USA.

Ryder, S., and J. Taylor. 1993. The Importance of Representativeness in Successful Negotiations. Report, National Ecology Research Center, National Biological Survey, Ft. Collins Colorado.

Saaty, T. 1980. *The Analytic Hierarchy Process*. McGraw-Hill, NY.

Sainfort, F.C., D. Gustafson, K. Bosworth, and R. Hawkins. 1990. Decision Support Systems Effectiveness: Conceptual Framework and Empirical Evaluation. *Organizational Behaviour and Human Decision Processes*, **45**:232-252.

Sakawa, M. 1993. *Fuzzy Sets and Interactive Multiobjective Optimization*. Plenum Press, New York.

Salewicz, and D.P. Loucks. 1989. Interactive Simulation for Planning, Managing and Negotiating. In Loucks, D.P., and U. Shamir, *Closing the Gap Between Theory and Practice*. IAHS publication #180, IAHS Press, Wallingford.

Seo, F., and M. Sakawa. 1985. Fuzzy Multiattribute Utility Analysis for Collective Choice. *IEEE Transactions on Systems, Man, and Cybernetics*, **15**(1):45-53.

Shaw, M., and M. Fox. 1993. Distributed Artificial Intelligence for Group Decision Support: integration of problem solving, coordination, and learning. *Decision Support Systems*, 9:349-367.

Simonovic, S. 1995. Decision Support Systems for Sustainable Management of Water Resources: 1. General Principles. Submitted to *Water International*.

Simonovic, S. 1992. Reservoir Systems Analysis: Closing the Gap between Theory and Practice. *ASCE J. Water Resources Planning and Management*, **118**(3):262-280.

Simonovic, S. 1991. Knowledge-based Systems and Operational Hydrology. *Can. J. Civil Engineering*, **18**(1):1-11.

Simonovic, S. 1989. Application of Water Resources Systems Concepts to the Formulation of a Water Master Plan. *Water International*, **14**(1):37-51.

Simonovic, S., and M. Bender. 1996. Collaborative Planning Support System: an approach for determining evaluation criteria. *J. Hydrology*, **177**:237-251.

Simonovic, S., and D. Savic. 1989. Intelligent Decision Support for Reservoir Management and Operations. *ASCE J. Computing in Civil Engineering*, **3**(4):367-384.

Simonovic, S., et al. 1995. Sustainability Criteria: an application to the hydropower industry. In ASCE 22nd Annual Conference of Water Resources Planning and Management Division - Integrated Water Resources Planning for the 21st Century, edited by M.F. Domenica. pp. 173-176.

Slowinski, R. 1986. A Multi-criteria Fuzzy Linear Programming Method for Water Supply System Development Planning. *Fuzzy Sets and Systems*, **19**:217-237.

Slowinski, R., and J. Teghem (eds). 1990. *Stochastic versus Fuzzy Approaches to Multi-objective Mathematical Programming Problems under Uncertainty*. Kluwer Academic Publ., Dordrecht.

Smith, L. 1993a. *Impact Assessment and Sustainable Resource Management*. Wiley, New York.

Smith, G. 1993b. Defining Real World Problems: A Conceptual Language. *IEEE Transactions on Systems, Man, and Cybernetics*, **23**(5):1220-1234.

Spence, M. 1974. Blue Whales and Applied Control Theory. In Zadeh et. al., ed., *System Approaches for Solving Mathematical Problems*. Vandenhoeck and Ruprecht, Zurich.

Spradley, J. 1980. *Participant Observation*. Harcourt Brace Jovanovich College Publ., New York.

Starr, M., and M. Zeleny. 1977. MCDM - State and Future of the Arts. *TIMS Stud. Management Sci.*, **6**:5-29.

Stauth, R., M. Sowman, and S. Grindley. 1993. The Panel Evaluation Method: an Approach to Evaluating Controversial Resource Allocation Proposals. *Environmental Impact Assessment Review*, **13**:13-35.

Susskind, L., and J. Cruikshank. 1987. *Breaking the Impasse: Consensual Approaches to Resolving Public Disputes*. Basic Books, New York.

Susskind, L., G. McMahon, and S. Rolley. 1987. Mediating Development Disputes: Some Barriers and Bridges to Successful Negotiation. *Environmental Impact Assessment Review*, **7**:127-138.

Szidarovszky, F., M. Gershon, and L. Duckstein. 1986. *Techniques for Multiobjective Decision Making in Systems Management*. Elsevier Publishing, Amsterdam.

Tecle, A. 1992. Selecting a Multicriterion Decision Making Technique for Watershed Resources Management. *Water Resources Bulletin*, **28**(1):129-140.

Thiessen, E., and D.P. Loucks. 1992. Computer Assisted Negotiation of Multiobjective Water Resource Conflicts. *Water Resources Bulletin*, **28**(1):163-177.

Tietenberg, T. 1992. *Environmental and Natural Resource Economics*. Harpers Collins Publ. Inc., New York.

Tinbergen, J., and R. Hueting. 1991. GNP and Market Prices: Wrong signals for sustainable economic success that mask environmental destruction. In Goodland, R., H. Daly, S. El Sarafy, and B. von Drost. *Environmentally Sustainable Economic Development: Building on Brundtland*. UNESCO, Paris.

Tversky, A., and D. Kahneman. 1990. Judgement under Uncertainty: heuristics and biases. In *Readings in Uncertain Reasoning*, edited by Shafer and Pearl.

Uber, J., E. Downey Brill, and J. Pfeffer. 1992. Use of Mathematical Programming Methods for Complex Systems. *ASCE J. Water Resources Planning and Management*, **118**(3):281-294.

Vearil, J., 1994. Public Involvement in Water Resource Management. Proceedings: *ASCE Water Resources Planning and Management* annual conference.

WCED. 1987. *Our Common Future*. World Commission on Environment and Development. Oxford University Press.

Walsh, M. 1993. Toward Spatial Decision Support Systems in Water Resources. *ASCE J. Water Resources Planning and Management*, **119**(2):158-169.

Wellman, M., J. Breese, and R. Goldman. 1992. From Knowledge Bases to Decision Models. *The Knowledge Engineering Review*, **7**(1):35-53.

Wilds, L., and B. Lamb. 1986. Role Analysis: use of the Legal Institutional Analysis Model for Instream Flows. Western Social Science Association Annual Meeting, Reno, Nevada.

Xepapadeas, A. 1992. Environmental Policy, Adjustment Costs, and Behaviour of the Firm. *J. Environmental Economics and Management*, **23**:258-275.

Yager, R. 1981. A Procedure for Ordering Fuzzy Subsets of the Unit Interval. *Information Science*, **24**:143-161.

Yin, Y., and J. Pierce. 1993. Integrated Resource Assessment and Sustainable Land Use. *Environmental Management*, **17**(3):319-327.

Young, M. 1992. *Sustainable Investment and Resource Use*. UNESCO and Parthenon, Paris.

Zadeh, L. 1965. Fuzzy Sets. *Information and Control*, **8**:338-353.

Zeleny, M. 1992. An Essay into a Philosophy of MCDM: a way of thinking or another algorithm? *Comput. Oper. Res.*, **19**:563-566.

Zeleny, M. 1982. *Multiple Criteria Decision Making*. McGraw-Hill, New York.

Zeleny, M. 1973. Compromise Programming. In Cochrane, J., and M. Zeleny (ed), *Multiple Criteria Decision Making*. University of South Carolina Press, Columbia USA.

Zimmerman, H. 1987. *Fuzzy Sets, Decision-making, and Expert Systems*. Kluwer Academic Publ., Dordrecht.

Chapter 10 Glossary

Acceptability (Acc) A ranking measure for fuzzy sets.

Alternative A unique plan or proposal.

Alternative dispute resolution (ADR)
The set of settlement techniques designed to resolve conflicts out of court.

Analytic Hierarchy Process (AHP)
A MCDA technique pioneered by Saaty (1980) based on pairwise comparisons.

Arbitration An ADR technique which employs a third party to assign property rights.

Class (or Subclass) A category of data elements, such as a reservoir.

Collaborative planning Voluntary inclusion of stakeholders in the early planning process of a project.

Collaborative Planning Support System (CPSS)
A DSS specialized for voluntary group selection of planning alternatives.

Common property Property rights with some level of controlled access, but not a complete allocation.

Compromise Programming
A MCDA technique pioneered by Zeleny (1973) based on distance metrics from an ideal.

Consensus General agreement in opinion; An equitable compromise which is robust with regard to a) resource management uncertainties, and b) stakeholder perspectives.

Criteria Issues of concern used to judge alternatives.

Dam A hydraulic structure used to block the flow of water.

Decision maker A person involved in the decision process.

Decision Support System (DSS)
An integrated set of tools, arranged to provide timely information and interactive use for decision makers.

Digital Elevation Model (DEM)
A measure of the surface elevation over a region.

Ecological Integrity An abstract indicator for measuring the intrinsic value of the environment.

Ecosystem health	A group of indicators used to describe the general condition of environmental processes.
ELECTRE	A MCDA technique pioneered by (Benayoun et al, 1966), based on outranking relationships.
Expert system (ES)	An artificial intelligence technique for modelling knowledge in boolean logic.
Externality	Impacts (either good or bad) which are external to a frame of analysis.
Fuzzy set	A measure of possibility for a value which is modelled as imprecise.
Game theory	The theory of games, or the technique for modelling strategic plays in negotiation processes.
Geographic Information System (GIS)	Data management and analysis system for spatial and temporal data.
Goal	A target level for a criteria.
Graphical user interface (GUI)	The front end display of a computer application.
Grounded theory	An approach for inducing (explaining) abstract processes and ideas from basic factual evidence.
Hamiltonian	A function used to describe a dynamic (economic) system.
Hydroelectric generating station	The hydroelectric power production unit usually associated with a dam.
Inheritance	The ability of an object to define its behaviour and properties based on a parent object or class.
Lagrangian method	Standard optimization technique for constrained optimization problems.
Manitoba Hydro	A power utility in the province of Manitoba which generates energy predominately from hydroelectric sources.
Mediation	An ADR technique employing a mediator to settle claims between parties.
Multicriteria decision-making (MCDM)	A set of decision-making tools for multiobjective problems with a discrete number of alternatives.

Multicriteria decision analysis (MCDA)

A set decision analysis techniques to analyzing multiobjective problems with a discrete number of alternatives.

Necessity A fuzzy set relation, degree of containment.

Negotiation An ADR technique for negotiating settlements, usually modelled within Game Theory.

Nondominated frontier The set of solutions in multiobjective space which are not entirely dominated by any other solution.

Noninferior surface See nondominated frontier.

Object-oriented A paradigm for data management consisting of classes, objects, and properties. It is used for programming, modelling, and design of computer programs and systems analysis.

Object (or Subobject) An individual data element, such as Lake Winnipeg.

Objective A performance evaluation function used to assess relative quality of solutions in multiobjective problems.

Pareto-optimal A relationship to the nondominated frontier in multiobjective analysis. A pareto-optimal move (or change) results in nondecreasing values of all objectives.

Participant A person or stakeholder involved in the decision process.

Polymorphism The ability of different classes of objects to behave differently under the same conditions.

Possibility A fuzzy set relation, degree of overlap.

Probability distribution function (*pdf*)

A function describing the inherent randomness of a parameter.

Property rights Streams of benefits and costs that affect the allocation and use of resources.

Proponent A champion for a cause, the party initiating a proposed development.

Raster GIS image data type based on pixels of uniform size.

Renewable resource Mass or energy sources subject to constant or periodic flux.

Reservoir A controlled lake or body of water.

Resilience The ability to rebound from change.

Reversibility The ability to return to a former state.

Robust The ability to perform well for many different possible conditions.

Sensitivity analysis The process of measuring the impact from changing system parameters.

Stakeholder An affected or interested party, including government regulators and the development proponent.

Subclass See Class.

Subobject See Object.

Technical option A structural element such as a dyke, or a reservoir operating rule.

Tradeoff A comparison of marginal differences between 2 alternatives.

Triangular norms A fuzzy set-theoretic construct for comparing fuzzy sets.

Valued Ecosystem Component (VEC)
An ecosystem process or object which has value (marketable or nonmarketable).

Vector GIS image data type based on the constructs of points, lines, and polygons.

Vulnerability The degree of openness to change from outside influences.

Weighted Center of Gravity (WCoG)
A ranking measure for fuzzy sets based on the centroid of the area beneath a fuzzy set.

Weights Degree of relative importance for a criteria, goal, or objective.

Wuskwatim Lake A lake in northern Manitoba, identified as a potential site for hydroelectric power generation.

Chapter 11 Notation

A	Set of advice; Degree of consensus
α	Weight
C	Marginal cost; Degree of conflict
c	Child objective; Coefficient
D	Development function
δ	Discount rate
E	Effort
Φ	Aggregation of satisficing relationships
F	Natural growth function
ϕ	Satisficing relationship; Boolean reference by a fact to a criteria
f	Function
f_i	Criteria value
f_i^*	Positive criteria ideal
f_i^-	Negative criteria ideal
G	Set of grounded facts; Fuzzy set representing a goal
γ	Consensus measure
g	Grounded fact
H	Harvest function for fish
I	Set of possible system states
J	Set of stakeholders
K	Set of domain knowledge

L	Set of links between objectives; Distance metric
l	Link between 2 objectives
M	Objective function
μ	Degree of (fuzzy) membership
MC	Marginal cost
O	Set of objectives
p	Resource price; Parent objective; Distance metric exponent
θ	Boolean reference by a user to a fact
q	Scalar coefficient; weighting exponent for **WCoG** measure
R	Decision space for a particular set of resources
ρ	Discount factor
S	Set of objectives relevant to a stakeholder
s	Stakeholder; Fuzzy triangular norm modelling the union operator
t	Time index; Fuzzy triangular norm modelling the intersection operator
V	Value function
W	Welfare function
w	Weight
X	State variable
x	Stock value; Value measurement

Chapter 12 Appendices

Appendix A. Internalization Experiments.

A.1 Methodology

An amount of literature produced by Shmuel Amir of *Resources for the Future*, in Washington DC, discusses the environmental costs of sustainable welfare (among other things). He maintains that sustainable development demands negative benefits to the environment, regardless of institutional changes. This includes transferral of externalities from one resource to another as each impact is brought to bear through policy development. He promotes partial solutions through expanded recognition of economic and ecosystem flows, and analyzes these systems within an entropy paradigm. Amir appears to be significant enough to examine more closely, but he is generally very critical of everything. I don't think he is threatening the direction of thinking in water resources. On the contrary, he may simply be adding a dimension of analysis that considers, using entropy, flows of resources to and from systems not directly involved in a decision process. This can also touch on cumulative and irreversible impacts of development.

In an effort to apply the concepts discussed in this thesis, a model has been selected for examination. A discrete renewable resource economic, spawner-recruit stock resource, model for harvesting of Antarctic Blue Whales (Spence, 1974; and Conrad, 1991 - *Note: Conrad is from Cornell U.*) provides a typical model form for examining ecosystems which have market values and also naturally regenerate. Examples of similar models are used with salmon stocks. The validity of the model for policy development of Blue Whale harvesting is irrelevant. The model simply provides a baseline for analysis.

Four unique scenarios were envisioned to encompass the potential decision spaces for policy development (see figure). They range from single decisions for constant harvesting effort (scenario 1), to flexible decision-making on an annual basis (scenario 3).

For each of these scenarios, a number of optimal paths may be chosen based on initial resource conditions, planning horizon, and discounting rates. Scenario 1 can be examined by manual calculations since it contains only 1 decision variable (DV). The other scenarios, however, are much more complex and the shape of the decision spaces are unknown. An initial examination of optimal solutions for scenarios 2 to 4, as well as scenario 1 is necessary for reference material to baseline model behavior. The Box Complex nonlinear search was used to find optimal model conditions. 109 solutions in total were found.

A number of problems with the Box search became evident. The search method experienced difficulties when initial solutions gave negative objective function (OF) values. Some results also showed problems in the optimal DV. Some solutions may only be locally optimal, or on a part of the decision space that is very insensitive to changes in DV. This is due to the fact that the OF varies at 5 orders of magnitude greater than the DV. Within the search formulation are variables that represent the sensitivity of both the OF and the DV, but in many cases selection of proper sensitivity levels is nearly impossible. Some of the solutions demand over 5000 iterations in the present form of the problem. When this search shortcoming is combined with an OF that uses a large discounting rate, solutions can be extremely volatile. $\rho=0.7$ was one of the discount rates explored. It proved to be a poor choice because the extremely discounted future values compounded the search algorithm's inability to distinguish between changes in DV. As a result, effort levels in some solutions jump from 0 to nearly 500. Expected effort levels should be less than 100 most of the time.

Some of the solutions appear to be reliable. The following is a brief summary of behaviors from the scenarios. The figures provided will tend to echo many of the comments.

A.2 Scenario 1 - constant effort harvesting

- For low X_o, increasing ρ results in increasing optimal ***E***, and decreasing stock equilibrium.
- For high X_o, increasing ρ results in decreasing optimal ***E***, and increasing stock equilibrium.
- For low X_o, increasing ***T*** causes increased ***E***, decreasing stock equilibrium.
- For high X_o, increasing ***T*** causes decreased ***E***, increasing stock equilibrium.
- Logically, increasing X_o provides flexible policy decision positions and encourages greater ***E***, but stock equilibrium decreases significantly.

A.3 Scenario 2 - 'bang-bang' form of steady-state control

- Increasing X_o allows harvesting to begin sooner.
- Increasing ***T*** has negligible effect on the harvesting starting point, ***S***, but allows more 'relaxed' harvesting (for a greater length of time).
- Changes in ρ (at the high end: $0.95<\rho<0.999$) have little effect on resulting optimal ***E***.

A.4 Scenario 3 - new effort decision are made every year

- Can be considered the most efficient solution set.
- ***E*** levels sometimes oscillate around the expected ***E***.
- At small or no discounting ($\rho=0.999$), increased X_o results in increased average ***E***. At short ***T***, ***E*** increases with time. At long ***T***, ***E*** establishes constant values around ***E***=30.
- At standard discounting ($\rho=0.95$), increased X_o also results in increased average effort. At small ***T***, ***E*** increases with time. At long ***T***, ***E*** establishes constant values around ***E***=40.
- At large discounts ($\rho=0.7$), solutions for ***E*** become volatile and unpredictable.

A.5 Scenario 4 - segments of constant effort

- High rates of discounting continue to give volatile results.
- In general, scenario 4 provides more stable solutions than scenario 3.
- ***E*** tends to increase steadily in time.
- The OF, net discounted benefits, decrease slightly for increased segment length ***S***.

A.6 Discussion

Application of the welfare model discussed by Loucks will effectively inhibit the effort in the end time step, and decrease efforts to conform with ecosystem needs. Resulting solutions will tend toward those in scenario 1, although by definition the effort must be continuously increasing to offset discounting effects. This is the moment where Amir (mentioned earlier) makes his point about necessary negative benefits to the environment. And this is where optimists such as those in the area of multicriteria decision making suggest the possibility of 'super-optimal' cooperative solutions to the problem of environmental degradation due to sustainable welfare. These types of solutions are technical ingenuities. A special level of understanding must exist for a person to suggest a super-optimal resolution.

One suggestion from systems analysis research (Simonovic) describes a decision process that searches for a least sensitive solution. The point of view is not unlike selecting a site based on ecosystem sensitivity to the proposed form of development. This is likely to adapt well to choosing a specific alternative from a list of specific possibilities. The remaining portion of the problem lies in identifying the direction of search for new or improved alternatives. It depends on the formulation of the problem, definitions, and language. Super-optimal solutions are not available with economic analysis. Expanded systems approaches that include economic considerations, and even discuss consequences in economic terms, may provide the framework for project decision-making.

In reducing the problem to its essential concept of transferring property rights, transaction costs can become a major contributor to the economic and social inefficiencies since negotiating positions can instinctively include an understanding of external costs whether they are driven by known impacts or by uncertainties. By describing not only the multiple objectives to be analyzed, but the form of the alternatives in consistent language, evaluation of acceptable tradeoffs among DMs can be the means of preparing input data to a single objective formulation for maximizing the consistently high ranking properties of established alternatives. For example, historical dealings with natives in Canada in regard to development of traditional native lands identified the production of jobs for natives as a high priority of any development proposal. This criteria was used as an objective along with economic efficiency. Finding solutions was similar to using a constraint method of evaluating multiobjective tradeoffs. Today, other criteria or objectives may be used to evaluate development. Unfortunately, they describe only the effects of the plan alternatives, and not the components of the plans themselves. Natives will be able to say why they like something, or why they don't. They are discussing only the effects of development. Choosing the least environmentally and socially sensitive alternative may be a good choice, but correlation of the components of various alternatives with tradeoff evaluation may identify specific components that inhibit consensus, or components that are driving consensus. In this way, essential components either for or against proposed development may be reformulated to produce super-optimal solutions. This type of correlation and optimization has the potential to enhance negotiation potentials by evaluating the level of mutual understanding or misunderstanding of technical concepts. The single decision variable would be a dimensionless correlation coefficient.

A.7 whale.for

```
C
      REAL STOCK(51),P,Q,A,B,C,SUM
      DIMENSION X(10,10), F(10)
C
      RO=0.95
      X0=30000
      N=10
      I=1
      DO 10 J=1,N
      X(I,J)=20
   10 CONTINUE
C
      P=7000.0
      C=875000.0
      A=8.4
      B=0.82
      Q=0.002
C
      SUM=0.0
      STOCK(1)=X0
      DO 500 J=1,N
      STOCK(J+1) = (A*STOCK(J)**B)*(1.0-(1.0-EXP(-Q*X(I,J))))
      SUM=SUM+(RO**J)*(P*(A*STOCK(J)**B)*(1.0-EXP(-Q*X(I,J)))-C*X(I,J))
  500 CONTINUE
C
      F(I) = SUM/1000000.0
C
      WRITE (*,*) X(I,1),STOCK(1)
      WRITE (*,*) X(I,5),STOCK(5)
      WRITE (*,*) X(I,10),STOCK(10)
      WRITE (*,*) F(I)
C
      STOP
      END
```

A.8 box.for

```
C
C     MAIN LINE PROGRAM FOR COMPLEX ALGORITHM OF BOX
C
      DIMENSION X(101,51), R(101,51), F(101), G(51), H(51), XC(51)
      INTEGER GAMMA
C
      NI = 5
      NO = 6
      OPEN(UNIT=NI,FILE='box4.in')
      OPEN(UNIT=NO,FILE='box4.out',STATUS='NEW')
C
   04 READ (5,*,END=911) RO,LT,LS,XO
      WRITE (6,009) RO,LT,LS,XO
  009 FORMAT (/,2X,F5.3,2X,I3,2X,I3,2X,F7.0)
C
      N = AINT((1.0*LT)/(LS*1.0))
      WRITE (6,*) N
      K = N+N
      M = N
      ITMAX = 10000
      IPRINT = 0
      IC = 0
      ALPHA = 1.3

      BETA = 0.001
      GAMMA = 5
      DELTA = 0.01
C
C     X(I,J) = EFFORT LEVEL IN DECISION TERM J
C
      DO 60 J=1,(N-1)
      X(1,J) = 0.1
   60 CONTINUE
      X(1,N) = 10.0
      DO 05 IR = 1,N
      DO 06 IS = 2,K
      R(IS,IR) = (((REAL(IR)-1)*REAL(IS))+REAL(IS))/(N*K)
   06 CONTINUE
   05 CONTINUE
C
      WRITE (NO,010)
  010 FORMAT (/,18X,24HCOMPLEX PROCEDURE OF BOX)
C      WRITE (NO,018)
C 018 FORMAT (/,2X,10HPARAMETERS )
C      WRITE (NO,011) N, M, K, ITMAX, IC, ALPHA, BETA, GAMMA, DELTA
C 011 FORMAT (/,2X,4HN = ,I2,3X,4HM = ,I2,3X,4HK = ,I2,2X,8HITMAX = ,
C     1I4,2X,4HIC = ,I2,/,2X,8HALPHA = ,F5.2,5X,7HBETA = ,F10.5,3X,
C     28HGAMMA = ,I2,3X,8HDELTA = ,F6.5)
      IF (IPRINT) 40, 50, 40
   40 WRITE (NO,012)
  012 FORMAT (/,2X,14HRANDOM NUMBERS)
      DO 200  J=2,K
      WRITE (NO,013) (J, I, R(J,I), I=1,N)
  013 FORMAT (3(2X,2HR(,I2,1H,,I2,4H) = ,F6.4,2X))
  200 CONTINUE
C
   50 CALL CONSX (N,M,K,ITMAX,ALPHA,BETA,GAMMA,DELTA,X,R,F,IT,
     1IEV2,NO,G,H,XC,IPRINT,XO,RO,LS)
C
      IF (IT-ITMAX) 20,20,30
   20 WRITE (NO,014) F(IEV2)
  014 FORMAT (/,2X,30HFINAL VALUE OF THE FUNCTION = ,E20.8)
```

```
C       WRITE (6,009) RO,N,XO
      WRITE (6,019) IT
  019 FORMAT (I4,2X,10HITERATIONS)
      WRITE (NO,015)
  015 FORMAT (/,2X,14HFINAL X VALUES)
      DO 300 J=1,N
      WRITE (NO,016) J, X(IEV2,J)
  016 FORMAT (2X,2HX(,I2,4H) = ,E20.8)
  300 CONTINUE
      GO TO 999
C
   30 WRITE (NO,017) ITMAX
  017 FORMAT (//,2X,38HTHE NUMBER OF ITERATIONS HAS EXCEEDED ,I4,10X,
     118HPROGRAM TERMINATED)
  999 GO TO 04
  911 STOP
      END
C
C
      SUBROUTINE CONSX (N,M,K,ITMAX,ALPHA,BETA,GAMMA,DELTA,X,R,F,IT,
     1IEV2,NO,G,H,XC,IPRINT,XO,RO,LS)
C     COORDINATES SPECIAL PURPOSE SUBROUTINES
C
C     ARGUMENT LIST
C
C     IT   = ITERATION INDEX
C     IEV1 = INDEX OF POINT WITH MIN. FUNCTION VALUE
C     IEV2 = INDEX OF POINT WITH MAX. FUNCTION VALUE
C     I    = POINT INDEX
C     KODE = CONTROL KEY USED TO DETERMINE IF IMPLICIT CONSTRAINTS
C                 ARE PROVIDED
C     K1   = DO LOOP LIMIT
C
C     ALL OTHERS PREVIOUSLY DEFINED IN MAIN LINE.
C
      DIMENSION X(101,51), R(101,51), F(101), G(51), H(51), XC(51)
      INTEGER GAMMA
C
      IT = 1
      KODE = 0
      IF (M-N) 20,20,10
   10 KODE = 1
   20 CONTINUE
      DO 40 II=2,K
      DO 30 J=1,N
   30 X(II,J) = 0.0
   40 CONTINUE
C
C     CALCULATE COMPLEX POINTS AND CHECK AGAINST CONSTRAINTS
C
      DO 65 II=2,K
      DO 50 J=1,N
      I = II
      CALL CONST (N,M,K,X,G,H,I)
      X(II,J) = G(J) + R(II,J)*(H(J)-G(J))
   50 CONTINUE
      K1 = II
      CALL CHECK (N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
      IF (II-2) 51, 51, 55
   51 IF (IPRINT) 52, 65, 52
   52 WRITE (NO,018)
  018 FORMAT (//,2X,30HCOORDINATES OF INITIAL COMPLEX)
      IO = 1
      WRITE (NO,019) (IO, J, X(IO,J), J=1,N)
```

```
  019 FORMAT (/,3(2X,2HX(,I2,1H,,I2,4H) = ,1PE13.6))
   55 IF (IPRINT) 56, 65, 56
   56 WRITE (NO,019) (II, J, X(II,J), J=1,N)
   65 CONTINUE
      K1 =K
      DO 70 I=1,K
      CALL FUNC (N,M,K,X,F,I,X0,RO,LS)
   70 CONTINUE
      KOUNT = 1
      IA = 0
C
C     FIND POINT WITH LOWEST FUNCTIONAL VALUE
C
      IF (IPRINT) 72, 80, 72
   72 WRITE (NO,021)
  021 FORMAT (/,2X,22HVALUES OF THE FUNCTION)
      WRITE (NO,022) (J, F(J), J=1,K)
  022 FORMAT (/,3(2X,2HF(,I2,4H) = ,1PE13.6))
   80 IEV1 =1
      DO 100 ICM=2,K
      IF (F(IEV1)-F(ICM)) 100,100,90
   90 IEV1 = ICM
  100 CONTINUE
C
C     FIND POINT WITH HIGHEST FUNCTION VALUE
C
      IEV2 = 1
      DO 120 ICM=2,K
      IF (F(IEV2)-F(ICM)) 110,110,120
  110 IEV2 = ICM
  120 CONTINUE
C
C     CHECK CONVERGENCE CRITERIA
C
      IF (F(IEV2)-(F(IEV1)+BETA)) 140,130,130
  130 KOUNT = 1
      GO TO 150
  140 KOUNT = KOUNT + 1
      IF (KOUNT-GAMMA) 150,240,240
C
C     REPLACE POINT WITH LOWEST FUNCTION VALUE
C
  150 CALL CENTR (N,M,K,IEV1,I,XC,X,K1)
      DO 160  JJ=1,N
  160 X(IEV1,JJ) = (1.0+ALPHA)*(XC(JJ))-ALPHA*(X(IEV1,JJ))
      I = IEV1
      CALL CHECK (N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
      CALL FUNC (N,M,K,X,F,I,X0,RO,LS)
C
C     REPLACE NEW POINT IF IT REPEATS AS LOWEST FUNCTION VALUE
C
  170 IEV2 = 1
      DO 190 ICM =2,K
      IF (F(IEV2)-F(ICM)) 190,190,180
  180 IEV2 = ICM
  190 CONTINUE
      IF (IEV2-IEV1) 220,200,220
  200 DO 210 JJ=1,N
      X(IEV1,JJ)=(X(IEV1,JJ) + XC(JJ))/2.0
  210 CONTINUE
      I = IEV1
      CALL CHECK (N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
      CALL FUNC (N,M,K,X,F,I,X0,RO,LS)
      GO TO 170
```

```
  220 CONTINUE
      IF (IPRINT) 230, 228,  230
  230 WRITE (NO,023) IT
  023 FORMAT (//,2X,17HITERATION NUMBER ,I5)
      WRITE (NO,024)
  024 FORMAT (/,2X,30HCOORDINATES OF CORRECTED POINT)
      WRITE (NO,019) (IEV1, JC ,X(IEV1,JC), JC=1,N)
      WRITE (NO,021)
      WRITE (NO,022) (I, F(I), I=1,K)
  228 IT = IT + 1
      IF (IT-ITMAX) 80,80,240
  240 RETURN
      END
C
C
      SUBROUTINE CHECK (N,M,K,X,G,H,I,KODE,XC,DELTA,K1)
C
C     ARGUMENT LIST
C
C     ALL ARGUMENTS DEFINED IN MAIN LINE AND CONSX
C
      DIMENSION X(101,51), G(51), H(51), XC(51)
C
   10 KT = 0
      CALL CONST (N,M,K,X,G,H,I)
C
C     CHECK AGAINST EXPLICIT CONSTRAINTS
C
      DO 50 J=1,N
      IF (X(I,J) - G(J)) 20,20,30
   20 X(I,J) = G(J) + DELTA
      GO TO 50
   30 IF (H(J)-X(I,J)) 40,40,50
   40 X(I,J) = H(J) - DELTA
   50 CONTINUE
C
      IF (KODE) 110,110,60
C
C     CHECK AGAINST THE IMPLICIT CONSTRAINTS
C
   60 NN = N + 1
      DO 100 J=NN,M
      CALL CONST (N,M,K,X,G,H,I)
      IF (X(I,J)-G(J)) 80,70,70
   70 IF (H(J)-X(I,J)) 80,100,100
   80 IEV1 = I
      KT = 1
      CALL CENTR (N,M,K,IEV1,I,XC,X,K1)
      DO 90 JJ=1,N
      X(I,JJ) = (X(I,JJ) + XC(JJ))/2.0
   90 CONTINUE
  100 CONTINUE
      IF (KT) 110,110, 10
  110 RETURN
      END
C
C
      SUBROUTINE CENTR (N,M,K,IEV1,I,XC,X,K1)
C
      DIMENSION X(101,51), XC(51)
C
      DO 20  J=1,N
      XC(J) = 0.0
      DO 10  IL=1,K1
```

```
   10 XC(J) = XC(J) + X(IL,J)
      RK = K1
   20 XC(J) = (XC(J)-X(IEV1,J))/(RK-1.0)
      RETURN
      END
C
C
      SUBROUTINE FUNC (N,M,K,X,F,I,X0,RO,LS)
C
      REAL P,Q,A,B,C,SUM
      DIMENSION X(101,51), F(101), STOCK(51)
C
      P=7000.0
      C=875000.0
      A=8.4
      B=0.82
      Q=0.002
C
      SUM = 0.0
      STOCK(1)=X0
      DO 500 J = 1,N
      CON = 1.0-EXP(-Q*X(I,J))
      DO 501 JJ = 1,LS
      JT = (J-1)*LS+JJ
      STOCK(JT+1) = (A*STOCK(JT)**B)*(1.0-CON)
      SUM = SUM+(RO**JT)*(P*(A*STOCK(JT)**B)*CON-C*X(I,J))
  501 CONTINUE
  500 CONTINUE
C
      F(I) = SUM/1000000.0
C
      RETURN
      END
C
C
      SUBROUTINE CONST (N,M,K,X,G,H,I)
C
      DIMENSION X(101,51),G(51),H(51)
C
C     Lower and upper bounds for DV
C
      DO 510 J=1,N
      G(J) = 0.0
      H(J) = 500.0
  510 CONTINUE
C
      RETURN
      END
```

Appendix B. Consensus Sustainability Experiment Programs.

B.1 compro.f

```
c ****************************************************************
c
c                       COMPROMISE PROGRAMMING
c                  by Mike Bender (Nov '95)
c
c     m     # alternatives ... i
c     n     # criteria ....... j
c     w     weights .......... kk
c     p     exponent ......... k
c
c ****************************************************************

        real w(10,50),zp(50),zn(50),zd(50),
     +z(50,20),l(10,10,20),p(10)

        integer m,n,np,alt(10,10,20),w_p,p_w

        open(1,file='compro.dat')
      open(2,file='ranks.dat')

      w_p=0
      p_w=1

c ****************************************************************
c    input block
c ****************************************************************

        read(1,*) nscheme
        read(1,*) m
        read(1,*) n
      read(1,*) np,(p(i),i=1,np)
      read(1,*) nw
      do 8 kk=1,nw
8       read(1,*) (w(kk,i),i=1,n)
      read(1,*) (zp(i),i=1,n)
      read(1,*) (zn(i),i=1,n)
      do 12 j=1,m
12      read(1,*) (z(i,j), i=1,n)

c ****************************************************************
c    print table heading
c ****************************************************************

      write(2,*) m
      write(2,*) nw

      write(*,*) 'm=',m
      write(*,*) 'n=',n
      write(*,*) ' rank     Lp     alt '
      write(*,*) '---------------------'

c ****************************************************************
c    calculate distance metrics
```

```
c *****************************************************************
c 3456789012345678901234567890123456789012345678901234567890123456789012

        do 10 i=1,n
10      zd(i)=zp(i)-zn(i)

      do 14 k=1,np
        write(*,*) 'p=',p(k)
        do 15 kk=1,nw
          write(*,100) (w(kk,i),i=1,n)
          do 20 j=1,m
            alt(kk,k,j)=j
            l(kk,k,j)=0.
            do 30 i=1,n
30    l(kk,k,j)=l(kk,k,j)+(w(kk,i)**p(k))*((zp(i)-z(i,j))/zd(i))**p(k)
            l(kk,k,j)=l(kk,k,j)**(1./p(k))
20        continue

          write(2,*) (l(kk,k,j),j=1,m)

c *****************************************************************
c    sort rankings (for the current 'p' value)
c *****************************************************************

          do 40 j=2,m
            a=l(kk,k,j)
            nalt=alt(kk,k,j)
            do 45 i=j-1,1,-1
              if (l(kk,k,i) .le. a) goto 47
              l(kk,k,i+1)=l(kk,k,i)
              alt(kk,k,i+1)=alt(kk,k,i)
45          continue
            i=0
47          l(kk,k,i+1)=a
            alt(kk,k,i+1)=nalt
40        continue

c *****************************************************************
c    print results (for the current 'p' value)
c *****************************************************************

          do 50 j=1,m
50          write(*,*) j, l(kk,k,j), alt(kk,k,j)

15      continue
14    continue

c *****************************************************************
c    print summary results
c *****************************************************************

      write(*,*)
      write(*,*) 'Summary of Alternative Performance Tradeoffs'
      write(*,*)
      write(*,*) 'Rank ->           ',(j,j=1,m)

      if (nscheme .eq. w_p) then
        do 60 kk=1,nw
          write(*,*)
c         write(*,100) (w(kk,i),i=1,n)
          write(*,190) kk
          do 70 k=1,np
70          write(*,150) p(k),(alt(kk,k,j),j=1,m)
60      continue
```

```
      else if (nscheme .eq. p_w) then
        do 80 k=1,np
          write(*,*)
          write(*,200) p(k)
          do 90 kk=1,nw
90          write(*,*) '       ',kk,'    | ',(alt(kk,k,j),j=1,m)
80      continue
      end if

100   format (50f5.2)
150   format ('     ',f5.1,'   |    ',20i3)
190   format ('w=',i3,'  p:')
200   format ('p=',f5.1,'  w:')
      stop
      end
```

B.2 consensus.f

```
c *****************************************************************
c
c                    Degree of Consensus
c               by Mike Bender (Jan '96)
c
c     m      # alternatives ...... i
c     ns     # stakeholders ...... j
c     gamma consensus measure ... l
c
c *****************************************************************
c 34567890123456789012345678901234567890123456789012345678901234567890l2

      real gamma(5,10),r(10,10),temp,mindif,maxdif,sumdif,u(10),w(10)
        integer m,ns

        open(1,file='ranks.dat')
        read(1,*) m
        read(1,*) ns
      do 2 j=1,ns
2       read(1,*) (r(i,j),i=1,m)

      close(1)

      do 45 j=1,ns
45      w(j)=1.

      greatest=0.
      do 40 i=1,m
        do 41 j=1,ns
41        if (r(i,j) .gt. greatest) greatest=r(i,j)
40    continue

      do 42 i=1,m
        do 43 j=1,ns
43        r(i,j)=r(i,j)/greatest
42    continue

      do 50 i=1,m
        u(i)=0.
        do 51 j=1,ns
51        u(i)=u(i)+w(j)*r(i,j)
50    continue

      do 100 i=1,m

c *****************************************************************
c   highest coincidence
c *****************************************************************
      mindif=u(i)
      do 10 j=1,ns
        do 11 k=1,ns
          temp=u(i)
          if (j .ne. k) temp=abs(w(j)*r(i,j)-w(j)*r(i,k))
          if (temp .lt. mindif) mindif=temp
11      continue
10    continue
      gamma(1,i)=1.-mindif

c *****************************************************************
c   highest discrepancy
c *****************************************************************
```

```
      maxdif=0.
      do 12 j=1,ns
        do 13 k=1,ns
          temp=0.
          if (j .ne. k) temp=abs(w(j)*r(i,j)-w(j)*r(i,k))
          if (temp .gt. maxdif) maxdif=temp
13      continue
12    continue
      gamma(2,i)=1-maxdif

c *****************************************************************
c   integral mean coincidence measure
c *****************************************************************
      sumdif=0.
      do 14 j=1,ns
14        sumdif=sumdif+abs(w(j)*r(i,j)-u(i)/real(m))
      gamma(3,i)=1-sumdif/real(m)

c *****************************************************************
c   integral pairwise coincidence
c *****************************************************************
      sumsumdif=0.
      do 16 j=1,ns-1
        do 17 k=j+1,ns
17        sumsumdif=sumsumdif+abs(w(j)*r(i,j)-w(j)*r(i,k))
16    continue
      gamma(4,i)=1-2./(ns*(ns-1.))*sumsumdif

c *****************************************************************
c   integral highest discrepancy measure
c *****************************************************************
      maxdif=0.
      do 18 j=1,ns
        do 19 k=1,ns
          temp=0.
          if (j .ne. k) temp=abs(w(j)*r(i,j)-u(i)/real(m))
          if (temp .gt. maxdif) maxdif=temp
19      continue
18    continue
      gamma(5,i)=1-maxdif

100   continue

c *****************************************************************
c   output block
c *****************************************************************
      write(*,*) 'm=',m
      write(*,*) 'ns=',ns
      write(*,*) ' alt  1     2     3     4     5'
      write(*,*) '-----------------------------------'
      do 90 i=1,m
90      write(*,95) i,(gamma(l,i), l=1,5)
95    format(2x,i2,5f6.3)

      stop
      end
```

B.3 compro.dat

```
1                      ! nscheme (0=w_p;1=p_w)
6                      ! m alternatives
8                      ! n criteria
3 1. 2. 10.            ! np p[k]
6                      ! nw # weight sets
1.2 1. .7 .7 .5 1.8 1.1 1.     ! w(i)
1.5 1.5 .8 .5 .3 1.2 1.3 .8
1. 1. 1. 1. 1. 1. 1. 1.
1. 1. 1.4 1.3 1. 1.2 .3 .7
1.2 1. .7 .7 .5 1.1 1.8 1.
1. 1.2 .7 .7 .5 1.8 1.1 1.
-300. -10. 5. 5. 5. 5. 5. 5.   ! zp(i)
-400. -20. 1. 1. 1. 1. 1. 1.   ! zn(i)
-307.6 -19.3 1 2 3 2 5 3       ! z(i) for alt j
-313.5 -17.6 2 2 4 2 5 3
-395.9 -14.5 5 4 5 5 3 4
-379.0 -13.7 5 3 5 4 4 2
-371.8 -14.0 5 4 5 4 4 2
-393.1 -14.7 5 3 5 5 3 5

-307.6 -19.3 2 2 2 3 4 3
-313.5 -17.6 3 2 3 3 5 3
-385.9 -14.5 5 4 4 5 3 4
-379.0 -13.7 4 3 4 4 4 3
-371.8 -14.0 4 4 5 4 4 3
-393.1 -14.7 4 3 4 5 3 4

-307.6 -19.3 3 3 3 3 4 2
-313.5 -17.6 3 2 4 3 4 3
-385.9 -14.5 5 4 4 4 3 5
-379.0 -13.7 4 3 5 4 4 3
-371.8 -14.0 4 3 4 5 4 3
-393.1 -14.7 5 4 4 4 3 4
```

B.4 rank.dat

```
 6
 6
   4.34620    3.94700    2.57580    3.14300    2.91160    2.48720
   4.13400    3.69250    3.08850    3.21500    3.02700    3.00150
   4.50600    3.89500    2.40900    2.91000    2.61800    2.40100
   5.13100    4.42000    2.05900    2.71000    2.34300    2.20100
   3.82120    3.42200    2.92580    3.14300    2.91160    2.83720
   4.51700    4.07200    2.47400    3.05900    2.84800    2.39500

   1.94264    1.80089    1.38653    1.41376    1.33177    1.37624
   1.93188    1.67835    1.73293    1.52656    1.44609    1.71238
   1.86967    1.61116    1.22359    1.30327    1.19395    1.26002
   2.22796    1.90891    1.13178    1.24690    1.07350    1.23801
   1.62233    1.44960    1.55883    1.41376    1.33177    1.54969
   2.03761    1.86798    1.26764    1.33580    1.27168    1.26880

   1.35342    1.35046    1.15088   0.956797   0.881166    1.11731
   1.39726    1.15047    1.43863    1.18519    1.07762    1.39672
   1.04803   0.864888   0.959192   0.828306   0.788443   0.931471
   1.40760    1.10917   0.959051   0.801782   0.721311   0.933629
  0.959425   0.857323    1.16030   0.956797   0.881166    1.12946
   1.36907    1.35268   0.959674   0.828111   0.789191   0.932101
```

Appendix C. CPSS Knowledge Bases.

C.1 cpss.scp - startup script for CPSS.

```
on event APPSTARTUP
      NOIR_LoadKB("cpss.tkb");
      NOIR_LoadKB("GIS.tkb");
      NOIR_LoadKB("alt.tkb");
      NOIR_LoadKB("multicriteria.tkb");
      NOIR_LoadKB("hydrotest/hydrotest.tkb");
      RLIB_LoadFile("cpss.dat");
      RLIB_LoadFile("FCP.dat");
      RLIB_LoadFile("hydrotest/hydrotest.dat");
      WIN_OpenByName("CPSS.Win1");
      NOIR_Suggest(NOIR_GetAtomId("START", NXP_ATYPE_HYPO), NXP_SPRIO_SUG);
      NOIR_Suggest(NOIR_GetAtomId("Load_objects", NXP_ATYPE_HYPO),
NXP_SPRIO_SUG);
      NOIR_Suggest(NOIR_GetAtomId("Initialize", NXP_ATYPE_HYPO),
NXP_SPRIO_SUG);
      NOIR_Knowcess();
end event
```

C.2 fcp.scp - startup script for the fuzzy compromise approach interface.

```
on event APPSTARTUP
      NOIR_LoadKB("multicriteria.tkb");
      RLIB_LoadFile("FCP.dat");
      WIN_OpenByName("Init.Win");
      NOIR_Suggest(NOIR_GetAtomId("Initialize", NXP_ATYPE_HYPO),
NXP_SPRIO_SUG);
      NOIR_Knowcess();
end event
```

C.3 hydro.scp - startup script for the hydropower dam design expert system.

```
on event APPSTARTUP
      NOIR_LoadKB("alt.tkb");
      NOIR_LoadKB("hydropower.tkb");
      RLIB_LoadFile("cpss.dat");
      WIN_OpenByName("Structures.Win1");
      WIN_OpenByName("Question.Win1");
      NOIR_Suggest(NOIR_GetAtomId("Load_objects", NXP_ATYPE_HYPO),
NXP_SPRIO_SUG);
      NOIR_Knowcess();
end event
```

C.4 cpss.tkb - main knowledge base for CPSS, including evaluation criteria selection module.

```
(@VERSION=  031)
(@PROPERTY= name         @TYPE=String;)
(@PROPERTY= class        @TYPE=String;)
(@PROPERTY= status       @TYPE=String;)
(@PROPERTY= ds           @TYPE=String;)
(@PROPERTY= us           @TYPE=String;)
(@PROPERTY= how          @TYPE=String;)
(@PROPERTY= why          @TYPE=String;)
(@PROPERTY= count        @TYPE=Integer;)
(@PROPERTY= sum_count    @TYPE=Integer;)
(@PROPERTY= weight       @TYPE=Float;)
(@PROPERTY= id           @TYPE=Integer;)
(@PROPERTY= id_fact      @TYPE=Integer;)
(@PROPERTY= id_obj       @TYPE=Integer;)
(@PROPERTY= id_class     @TYPE=Integer;)
(@PROPERTY= user         @TYPE=String;)
(@PROPERTY= fact         @TYPE=String;)
(@PROPERTY= OF           @TYPE=String;)
(@PROPERTY= mulVal       @TYPE=String;)

(@CLASS=    UtilityObjects          (@PROPERTIES=fact user))
(@CLASS=    StatisticalObjects      (@PROPERTIES=mulVal count sum_count))
(@CLASS=    Facts        (@PROPERTIES=fact id_fact id_class count))
(@CLASS=    Objectives   (@PROPERTIES=OF id_obj id_class count weight))
(@CLASS=    F_O_Relationships (@PROPERTIES=id_obj id_fact count))
(@CLASS=    Users             (@PROPERTIES=name))
(@CLASS=    VisibleUsers)
(@CLASS=    HiddenUsers)
(@CLASS=    UserFacts)
(@CLASS=    ImportantFacts)
(@CLASS=    ImportantRelationships)
(@CLASS=    ImportantObjectives)
(@CLASS=    FactsSubClasses         (@PROPERTIES=id_class class))
(@CLASS=    ObjectivesSubClasses    (@PROPERTIES=id_class class))
(@CLASS=    ObjectivesSummary (@PROPERTIES=id_class class weight))

(@OBJECT=   select            (@CLASSES=UtilityObjects))
(@OBJECT=   remove            (@CLASSES=UtilityObjects))
(@OBJECT=   new               (@CLASSES=UtilityObjects))
(@OBJECT=   hide              (@CLASSES=UtilityObjects))
(@OBJECT=   unhide            (@CLASSES=UtilityObjects))
(@OBJECT=   numFacts          (@CLASSES=StatisticalObjects))
(@OBJECT=   numRelationships  (@CLASSES=StatisticalObjects))
(@OBJECT=   numObjectives     (@CLASSES=StatisticalObjects))
(@OBJECT=   START             (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   Find_Objectives   (@PROPERTIES=Value      @TYPE=Boolean;))

(@META=     |Facts|.count                 (@INITVAL=  0))
(@META=     |F_O_Relationships|.count     (@INITVAL=  0))
(@META=     |Objectives|.count            (@INITVAL=  0))
(@META=     |Objectives|.weight           (@INITVAL=  0.00))
(@META=     |ObjectivesSummary|.weight    (@INITVAL=  0.00))
(@META=     |StatisticalObjects|.count    (@INITVAL=  0))
(@META=     |StatisticalObjects|.sum_count      (@INITVAL=  0))
(@META=     |UtilityObjects|.user         (@INITVAL=  ""))

(@METHOD=   IfChange
      (@ATOMID=new.user;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@LHS=
```

```
            (CreateObject       (\new.user\)        (|Users|))
            (CreateObject       (\new.user\)        (|VisibleUsers|))
            (Assign             (new.user)          (\new.user\.name))
        )
)
(@METHOD=   IfChange
        (@ATOMID=select.user;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (DeleteObject       (<|Facts|>)         (|UserFacts|))
            (Member             (<<|Facts|>>)       (<\select.user\>))
            (CreateObject       (<<|Facts|>>)       (|UserFacts|))
        )
)
(@METHOD=   IfChange
        (@ATOMID=select.fact;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (=              (<|Users|>.name)    (select.user))
            (=              (<|Facts|>.fact)    (select.fact))
            (NotMember      (<|Facts|>)         (<|UserFacts|>))
            (CreateObject       (<|Facts|>) (\select.user\))
            (CreateObject       (<|Facts|>) (|UserFacts|))
            (CreateObject       (<|Facts|>) (|ImportantFacts|))
            (Assign             (<|Facts|>.count + 1)     (<|Facts|>.count))
            (=              (<|Facts|>.count) (1))
            (CreateObject       (<|Facts|>) (|ImportantFacts|))
            (Assign             (numFacts.count + 1)      (numFacts.count))
        )
)
(@METHOD=   IfChange
        (@ATOMID=hide.user;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (=              (<|Users|>.name)    (hide.user))
            (DeleteObject       (<|Users|>)         (|VisibleUsers|))
            (CreateObject       (<|Users|>)         (|HiddenUsers|))
            (Member             (<|Facts|>)         (<\hide.user\>))
            (Assign             (<|Facts|>.count - 1)     (<|Facts|>.count))
            (=              (<|Facts|>.count) (0))
            (DeleteObject       (<|Facts|>) (|ImportantFacts|))
            (Assign             (numFacts.count - LENGTH(<|Facts|>))
(numFacts.count))
        )
        (@RHS=
            (Assign             ("")                (select.user))
            (Assign             ("")                (hide.user))
        )
        (@EHS=
            (Assign             ("")                (select.user))
            (Assign             ("")                (hide.user))
        )
)
(@METHOD=   IfChange
        (@ATOMID=unhide.user;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (=              (<|Users|>.name)    (unhide.user))
            (DeleteObject       (<|Users|>)         (|HiddenUsers|))
            (CreateObject       (<|Users|>)         (|VisibleUsers|))
            (Member             (<|Facts|>)         (<\unhide.user\>))
            (Assign             (<|Facts|>.count + 1)     (<|Facts|>.count))
            (=              (<|Facts|>.count) (1))
            (CreateObject       (<|Facts|>) (|ImportantFacts|))
```

```
            (Assign             (numFacts.count + LENGTH(<|Facts|>))
(numFacts.count))
        )
        (@RHS=
            (Assign             ("")                (unhide.user))
        )
)
(@METHOD=   IfChange
        (@ATOMID=remove.user;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (Member             (<|Facts|>)         (<\remove.user\>))
            (Assign             (<|Facts|>.count - 1)    (<|Facts|>.count))
            (=          (<|Facts|>.count) (0))
            (DeleteObject       (<|Facts|>) (|ImportantFacts|))
            (Assign             (numFacts.count - LENGTH(<|Facts|>))
(numFacts.count))
        )
        (@RHS=
            (DeleteObject       (\remove.user\))
            (Assign             ("")            (select.user))
        )
        (@EHS=
            (DeleteObject       (\remove.user\))
            (Assign             ("")            (select.user))
        )
)
(@METHOD=   IfChange
        (@ATOMID=remove.fact;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (=          (<|Facts|>.fact)  (remove.fact))
            (DeleteObject       (<|Facts|>)         (\select.user\))
            (DeleteObject       (<|Facts|>)         (|UserFacts|))
            (Assign             (<|Facts|>.count - 1)    (<|Facts|>.count))
            (=          (<|Facts|>.count) (0))
            (DeleteObject       (<|Facts|>) (|ImportantFacts|))
            (Assign             (numFacts.count - 1)     (numFacts.count))
        )
)
(@METHOD=   Count
        (@ATOMID=ImportantRelationships;@TYPE=CLASS;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (=          (<|ImportantFacts|>.id_fact)        (SELF.id_fact))
            (Assign             (SUM(<|ImportantFacts|>.count))
(SELF.count))
        )
)
(@METHOD=   Count
        (@ATOMID=ImportantObjectives;@TYPE=CLASS;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (=          (<|ImportantRelationships|>.id_obj) (SELF.id_obj))
            (Assign             (SUM(<|ImportantRelationships|>.count))
(SELF.count))
        )
)
(@METHOD=   Init
        (@ATOMID=FactsSubClasses;@TYPE=CLASS;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (CreateObject       (\SELF.class\))
            (=          (<|Facts|>.id_class)    (SELF.id_class))
```

```
            (CreateObject     (<|Facts|>)       (\SELF.class\))
      )
)
(@METHOD=   Init
      (@ATOMID=ObjectivesSubClasses;@TYPE=CLASS;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (CreateObject     (\SELF.class\)
(|ObjectivesSummary|))
            (Assign           (SELF.class)      (\SELF.class\.class))
            (Assign           (SELF.id_class)
(\SELF.class\.id_class))
            (=          (<|Objectives|>.id_class)     (SELF.id_class))
            (CreateObject     (<|Objectives|>)        (\SELF.class\))
      )
)
(@METHOD=   Weight
      (@ATOMID=ImportantObjectives;@TYPE=CLASS;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (Assign           (SELF.count / (numObjectives.sum_count * 1.0))
(SELF.weight))
      )
)

(@METHOD=   Weight
      (@ATOMID=ObjectivesSummary;@TYPE=CLASS;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (=          (<|ImportantObjectives|>.id_class)  (SELF.id_class))
            (Assign           (SUM(<|ImportantObjectives|>.weight))
(SELF.weight))
      )
      (@EHS=
            (Assign           (0)               (SELF.weight))
      )
)

(@RULE=     R1
      @COMMENTS="Load the list of facts, objectives, and relationships.";
      (@LHS=
            (Execute    ("Message") (@STRING="@TEXT=Reading from database
(Retrieve) ..., @TRANSCRIPT";))
            (Retrieve   ("facts.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|Facts|;\

@PROPS=id_fact,id_class,fact;@FIELDS="id_fact","id_class","fact";))
            (Retrieve   ("objectives.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|Objectives|;\

@PROPS=id_obj,id_class,OF;@FIELDS="id_obj","id_class","OF";))
            (Retrieve   ("relate.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|F_O_Relationships|;\

@PROPS=id_obj,id_fact;@FIELDS="id_obj","id_fact";))
            (Retrieve   ("cat_facts.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|FactsSubClasses|;\

@PROPS=id_class,class;@FIELDS="id_class","class";))
            (Retrieve   ("cat_obj.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|ObjectivesSubClasses|;\

@PROPS=id_class,class;@FIELDS="id_class","class";))
      )
```

```
        (@HYPO=      START)
        (@RHS=
              (SendMessage       ("Init")     (@TO=<|FactsSubClasses|>;))
              (SendMessage       ("Init")     (@TO=<|ObjectivesSubClasses|>;))
        )
)
(@RULE=     R2
        @COMMENTS="Suggest relevant objectives, given the list of facts.";
        (@LHS=
              (DeleteObject      (<|Objectives|>)  (|ImportantObjectives|))
              (DeleteObject      (<|F_O_Relationships|>)
(|ImportantRelationships|))
              (Execute     ("Unify")
(@ATOMID=numFacts.count,<|ImportantFacts|>,<|F_O_Relationships|>;\
                                          @STRING="@EQUAL,
@TESTFROM=id_fact,@TESTTO=id_fact,\
                                          @TOLINK=ImportantRelationships";))
              (Execute     ("GetRelatives")  (@STRING="@ONELEVEL, @CHILDREN,
@RETURN=numRelationships.mulVal";\
                                                @ATOMID=ImportantRelationships;))
              (Execute     ("GetMultiValue")
(@ATOMID=numRelationships.mulVal;@STRING="@LENGTH,\
                                                @RETURN=numRelationships.count";))
              (Execute     ("Unify")
(@ATOMID=numRelationships.count,<|ImportantRelationships|>,<|Objectives|>;\
                                          @STRING="@EQUAL,
@TESTFROM=id_obj,@TESTTO=id_obj,\
                                          @TOLINK=ImportantObjectives";))
        )
        (@HYPO=      Find_Objectives)
        (@RHS=
              (SendMessage       ("Count")    (@TO=<|ImportantRelationships|>;))
              (SendMessage       ("Count")    (@TO=<|ImportantObjectives|>;))
              (Assign            (SUM(<|ImportantObjectives|>.count))
(numObjectives.sum_count))
              (SendMessage       ("Weight")   (@TO=<|ImportantObjectives|>;))
              (SendMessage       ("Weight")   (@TO=<|ObjectivesSummary|>;))
              (Reset             (Find_Objectives))
        )
        (@EHS=
              (SendMessage       ("Count")    (@TO=<|ImportantRelationships|>;))
              (SendMessage       ("Count")    (@TO=<|ImportantObjectives|>;))
              (Assign            (SUM(<|ImportantObjectives|>.count))
(numObjectives.sum_count))
              (SendMessage       ("Weight")   (@TO=<|ImportantObjectives|>;))
              (SendMessage       ("Weight")   (@TO=<|ObjectivesSummary|>;))
              (Reset             (Find_Objectives))
        )
)
```

C.5 alt.tkb - knowledge base for managing alternatives defined by stakeholders.

```
(@VERSION=  031)
(@PROPERTY= name                    @TYPE=String;)
(@PROPERTY= parent                  @TYPE=String;)
(@PROPERTY= class                   @TYPE=String;)
(@PROPERTY= type                    @TYPE=String;)
(@PROPERTY= design                  @TYPE=String;)
(@PROPERTY= add                     @TYPE=String;)
(@PROPERTY= delete                  @TYPE=String;)
(@PROPERTY= mulVal                  @TYPE=String;)
(@PROPERTY= max_stage               @TYPE=String;)
(@PROPERTY= min_stage               @TYPE=String;)
(@PROPERTY= file_flooded_point           @TYPE=String;)
(@PROPERTY= stage                   @TYPE=String;)
(@PROPERTY= res_vol                 @TYPE=Float;)
(@PROPERTY= res_area                @TYPE=Float;)
(@PROPERTY= file                    @TYPE=String;)

(@PROPERTY= C1                      @TYPE=String;)
(@PROPERTY= C2                      @TYPE=String;)
(@PROPERTY= C3                      @TYPE=String;)
(@PROPERTY= C4                      @TYPE=String;)
(@PROPERTY= C5                      @TYPE=String;)
(@PROPERTY= C6                      @TYPE=String;)
(@PROPERTY= C7                      @TYPE=String;)
(@PROPERTY= C8                      @TYPE=String;)
(@PROPERTY= C9                      @TYPE=String;)
(@PROPERTY= C10                     @TYPE=String;)

(@CLASS=    Structures              (@PROPERTIES= name class type design stage
file_flooded_point)
                                    (@SUBCLASSES= Dam Dyke Spillway Powerhouse

Reservoir Release Fishway)
)
(@CLASS=    Components              (@PROPERTIES= name class type design parent)
                                    (@SUBCLASSES= Spillway Powerhouse Reservoir

Release Fishway)
)

(@CLASS=    Dam                     (@PROPERTIES= res_area res_vol))
(@CLASS=    Dyke)
(@CLASS=    Spillway)
(@CLASS=    Powerhouse)
(@CLASS=    Reservoir)
(@CLASS=    Release)
(@CLASS=    Fishway)

(@CLASS=    Potential_structures)
(@CLASS=    Proposed_structures)
(@CLASS=    Proposed_dams)
(@CLASS=    Proposed_dykes)
(@CLASS=    Proposed_other)
(@CLASS=    Alternatives                 (@PROPERTIES= name C1 C2 C3 C4 C5 C6 C7 C8
C9 C10))
(@CLASS=    Selected_Alternatives)
(@CLASS=    Current_Alternative)

(@OBJECT=   alt                     (@PROPERTIES= name file mulVal))
(@OBJECT=   structure               (@PROPERTIES= add delete class))
(@OBJECT=   next_alt_num                 (@PROPERTIES=Value      @TYPE=Integer;))
(@OBJECT=   temp                    (@PROPERTIES=Value      @TYPE=String;))
```

```
(@OBJECT=   Load_objects            (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   Flood             (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   Compile_structures      (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   Decompile_structure     (@PROPERTIES=Value      @TYPE=Boolean;))

(@META=           alt.name          (@INITVAL=  ""))

(@METHOD=   Init
      (@ATOMID=Structures;@TYPE=CLASS;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (CreateObject     (SELF)                  (\SELF.class\))
            (=          (SELF.class)            ("Dam"))
      )
      (@RHS=
            (Assign           (STRCAT("Spillway_",SELF.name))
(temp))
            (CreateObject     (\temp.Value\)    (|Spillway|))
            (CreateObject     (\temp.Value\)    (SELF))
            (Assign           (temp.Value)      (\temp.Value\.name))
            (Assign           (SELF.name) (\temp.Value\.parent))
            (Assign           ("Spillway")
(\temp.Value\.class))
            (Assign           (STRCAT("Powerhouse_",SELF.name))   (temp))
            (CreateObject     (\temp.Value\)    (|Powerhouse|))
            (CreateObject     (\temp.Value\)    (SELF))
            (Assign           (temp.Value)(\temp.Value\.name))
            (Assign           (SELF.name) (\temp.Value\.parent))
            (Assign           ("Powerhouse")
(\temp.Value\.class))
            (Assign           (STRCAT("Reservoir_",SELF.name))    (temp))
            (CreateObject     (\temp.Value\)    (|Reservoir|))
            (CreateObject     (\temp.Value\)    (SELF))
            (Assign           (temp.Value)(\temp.Value\.name))
            (Assign           (SELF.name) (\temp.Value\.parent))
            (Assign           ("Reservoir")     (\temp.Value\.class))
            (Assign           (STRCAT("Release_",SELF.name))
(temp))
            (CreateObject     (\temp.Value\)    (|Release|))
            (CreateObject     (\temp.Value\)    (SELF))
            (Assign           (temp.Value)      (\temp.Value\.name))
            (Assign           (SELF.name) (\temp.Value\.parent))
            (Assign           ("Release")       (\temp.Value\.class))
            (Assign           (STRCAT("Fishway_",SELF.name))
(temp))
            (CreateObject     (\temp.Value\)    (|Fishway|))
            (CreateObject     (\temp.Value\)                      (SELF))
            (Assign           (temp.Value)      (\temp.Value\.name))
            (Assign           (SELF.name)       (\temp.Value\.parent))
            (Assign           ("Fishway") (\temp.Value\.class))
      )
)
(@METHOD=   IfChange
      (@ATOMID=structure.add;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (CreateObject     (\structure.add\)       (|Proposed_structures|))
            (Assign           (\structure.add\.class)       (structure.class))
            (Assign           (Compile_structures)
(Compile_structures))
      )
      (@RHS=
```

```
            (SendMessage        ("Add_proposed_structures")
(@TO=<\structure.add\>;))
            (Reset              (Compile_structures))
            (Reset              (structure.add))
      )
)
(@METHOD=   IfChange
      (@ATOMID=structure.delete;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (DeleteObject       (\structure.delete\)
(|Proposed_structures|))
            (Assign             (\structure.delete\.class)    (structure.class))
            (Assign             (Decompile_structure)
(Decompile_structure))
      )
      (@RHS=
            (SendMessage        ("Delete_proposed_structures")
(@TO=<\structure.delete\>;))
            (Reset              (Decompile_structure))
            (Reset              (structure.delete))
      )
)
(@METHOD=   Add_proposed_structures
      (@ATOMID=Components;@TYPE=CLASS;)
      (@FLAGS=PUBLIC;)
      (@RHS=
            (CreateObject       (SELF)
(|Potential_structures|))
      )
)
(@METHOD=   Delete_proposed_structures
      (@ATOMID=Components;@TYPE=CLASS;)
      (@FLAGS=PUBLIC;)
      (@RHS=
            (DeleteObject       (SELF)
(|Proposed_structures|))
            (DeleteObject       (SELF)
(|Potential_structures|))
      )
)
(@METHOD=   Save_Alt
      (@ATOMID=alt;@TYPE=OBJECT;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (DeleteObject       (<|Alternatives|>)
(|Current_Alternative|))
            (Assign             (STRCAT("alt_",INT2STR(next_alt_num)))
(alt.name))
            (CreateObject       (\alt.name\)                  (|Alternatives|))
            (CreateObject       (\alt.name\)
(|Current_Alternative|))
            (Assign             (alt.name)              (\alt.name\.name))
            (Assign             (STRCAT("Dams_",alt.name))    (alt.file))
            (Write              ("@V(alt.file)")
(@TYPE="NXPDB";@FILL=NEW;\
                        @PROPS=stage,res_area,res_vol;\
                        @FIELDS="stage(10)","res_area(15)","res_vol(15)";\
                        @ATOMS=<|Proposed_dams|>;))
            (Assign             (STRCAT("Dykes_",alt.name))   (alt.file))
            (Write              ("@V(alt.file)")
(@TYPE="NXPDB";@FILL=NEW;\
                        @PROPS=stage;\
                        @FIELDS="stage(10)";\
```

```
                        @ATOMS=<|Proposed_dykes|>;))
            (Assign             (STRCAT("Other_",alt.name))    (alt.file))
            (Write              ("@V(alt.file)")
(@TYPE="NXPDB";@FILL=NEW;\
                        @PROPS=name;\
                        @FIELDS="name(30)";\
                        @ATOMS=<|Proposed_other|>;))
        )
        (@RHS=
            (Assign             (next_alt_num + 1)       (next_alt_num))
            (Write              ("alternatives.nxp")
(@TYPE="NXPDB";@FILL=INSERT;\
                        @ATOMS=<|Current_Alternative|>;))
        )
)
(@METHOD=   Load_Alt
        (@ATOMID=alt;@TYPE=OBJECT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (<>         (alt.name)          (""))
            (SendMessage        ("Clear_Alt")            (@TO=alt;))
        )
        (@RHS=
            (Assign             (STRCAT("Dams_",alt.name))     (alt.file))
            (Retrieve   ("@V(alt.file)")  (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Proposed_dams|;\
                        @PROPS=name,stage,res_area,res_vol;\
                        @FIELDS="Name","stage","res_area(15)","res_vol(15)";))
            (Assign             (STRCAT("Dykes_",alt.name))    (alt.file))
            (Retrieve   ("@V(alt.file)")  (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Proposed_dykes|;\
                        @PROPS=name,stage;\
                        @FIELDS="Name","stage";))
            (Assign             (STRCAT("Other_",alt.name))    (alt.file))
            (Retrieve   ("@V(alt.file)")  (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Proposed_other|;\
                        @PROPS=name;\
                        @FIELDS="Name";))
            (CreateObject       (<|Proposed_dams|>)      (|Proposed_structures|))
            (CreateObject       (<|Proposed_dykes|>)     (|Proposed_structures|))
            (CreateObject       (<|Proposed_other|>)     (|Proposed_structures|))
        )
)
(@METHOD=   Clear_Alt
        (@ATOMID=alt;@TYPE=OBJECT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
            (DeleteObject       (<|Structures|>)  (|Proposed_structures|))
            (DeleteObject       (<|Structures|>)  (|Proposed_dams|))
            (DeleteObject       (<<|Structures|>>)       (|Proposed_dykes|))
            (DeleteObject       (<|Components|>)  (|Proposed_other|))
        )
)
(@METHOD=   Delete_Alt
        (@ATOMID=alt;@TYPE=OBJECT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
            (<>         (alt.name)              (""))
            (DeleteObject       (\alt.name\)                    (|Alternatives|))
            (Write              ("alternatives.nxp")
(@TYPE="NXPDB";@FILL=NEW;\
                        @PROPS=name;@FIELDS="Name(30)";\
                        @ATOMS=<|Alternatives|>;))
        )
```

```
        (@RHS=
                (Assign         (STRCAT("Dams_",alt.name))    (alt.file))
                (Execute    ("rm @V(alt.file)")             (@TYPE=EXE;))
                (Assign         (STRCAT("Dykes_",alt.name))   (alt.file))
                (Execute    ("rm @V(alt.file)")             (@TYPE=EXE;))
                (Assign         (STRCAT("Other_",alt.name))   (alt.file))
                (Execute    ("rm @V(alt.file)")             (@TYPE=EXE;))
        )
)
(@RULE=     Rf1
        @COMMENTS="Load the list of available structures.";
        (@LHS=
                (Retrieve   ("structures.nxp")      (@TYPE="NXPDB";@FILL=ADD;\
                            @CREATE=|Structures|;\
                            @PROPS=name,class,stage,file_flooded_point;\
                            @FIELDS="Name","class","stage","file_flooded_point";))
                (Retrieve   ("alternatives.nxp")    (@TYPE="NXPDB";@FILL=ADD;\
                            @CREATE=|Alternatives|;\
                            @PROPS=name;@FIELDS="Name";))
                (Execute    ("GetRelatives")  (@STRING="@ONELEVEL, @CHILDREN,
@RETURN=alt.mulVal";\
                                                  @ATOMID=Alternatives;))
                (Execute    ("GetMultiValue")
(@ATOMID=alt.mulVal;@STRING="@LENGTH,\
                                                  @RETURN=next_alt_num";))
                (Assign         (next_alt_num + 1)      (next_alt_num))
        )
        (@HYPO=     Load_objects)
        (@RHS=
                (SendMessage      ("Init")          (@TO=<|Structures|>;))
        )
)
(@RULE=     Rf2
        @COMMENTS="Set the flood input file, execute the flood script, and
retrieve reservoir volumes.";
        (@LHS=
                (Write          ("flood.nxp")      (@TYPE="NXPDB";@FILL=NEW;\
                            @PROPS=stage,file_flooded_point,class;\
                            @FIELDS="stage(6)","file(30)","class(10)";\
                            @ATOMS=<|Proposed_structures|>;))
                (Execute    ("gen_fl_file")          (@TYPE=EXE;))
                (Execute    ("flood < flood_file")   (@TYPE=EXE;))
                (Retrieve   ("volume.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|Dam|;\
                            @PROPS=res_area,res_vol;\
                            @FIELDS="res_area","res_vol";))
        )
        (@HYPO=     Flood)
        (@RHS=
                (Reset          (Flood))
        )
)
(@RULE=     Decompile_structure_1
        @COMMENTS="Detach dams from the Proposed_dams class.";
        (@LHS=      (=          (structure.class)         ("Dam")))
        (@HYPO=     Decompile_structure)
        (@RHS=      (DeleteObject     (\structure.delete\)
(|Proposed_dams|)))
)
(@RULE=     Decompile_structure_2
        @COMMENTS="Detach dykes from the Proposed_dykes class.";
        (@LHS=      (=          (structure.class)         ("Dyke")))
        (@HYPO=     Decompile_structure)
```

```
      (@RHS=      (DeleteObject     (\structure.delete\)
(|Proposed_dykes|)))
)
(@RULE=     Decompile_structure_3
      @COMMENTS="Detach other structures from the Proposed_other class.";
      (@LHS=      (=            (structure.class)
("Spillway","Powerhouse","Fishway","Reservoir","Release")))
      (@HYPO=     Decompile_structure)
      (@RHS=      (DeleteObject     (\structure.delete\)
(|Proposed_other|)))
)
(@RULE=     Compile_structures_1
      @COMMENTS="Attach dams to the Proposed_dams class.";
      (@LHS=      (=            (structure.class)       ("Dam")))
      (@HYPO=     Compile_structures)
      (@RHS=      (CreateObject     (\structure.add\)
(|Proposed_dams|)))
)
(@RULE=     Compile_structures_2
      @COMMENTS="Attach dams to the Proposed_dykes class.";
      (@LHS=      (=            (structure.class)       ("Dyke")))
      (@HYPO=     Compile_structures)
      (@RHS=      (CreateObject     (\structure.add\)
(|Proposed_dykes|)))
)
(@RULE=     Compile_structures_3
      @COMMENTS="Attach dams to the Proposed_other class.";
      (@LHS=      (=            (structure.class)
("Spillway","Powerhouse","Fishway","Reservoir","Release")))
      (@HYPO=     Compile_structures)
      (@RHS=      (CreateObject     (\structure.add\)
(|Proposed_other|)))
)
(@GLOBALS=
      @INHVALUP=FALSE;
      @INHVALDOWN=TRUE;
      @INHOBJUP=FALSE;
      @INHOBJDOWN=FALSE;
      @INHCLASSUP=FALSE;
      @INHCLASSDOWN=TRUE;
      @INHBREADTH=FALSE;
      @INHPARENT=FALSE;
      @PWTRUE=FALSE;
      @PWFALSE=FALSE;
      @PWNOTKNOWN=FALSE;
      @EXHBWRD=FALSE;
      @PTGATES=FALSE;
      @PFACTIONS=TRUE;
      @SOURCESON=TRUE;
      @CACTIONSON=TRUE;
      @VALIDUSER=FALSE;
      @VALIDENGINE=FALSE;
      @PFEACTIONS=FALSE;
      @PFMACTIONS=GLOBAL;
      @PFMEACTIONS=FALSE;
)
```

C.6 flood.tkb - knowledge base for modelling flood impacts.

```
(@VERSION=  031)
(@PROPERTY= name            @TYPE=String;)
(@PROPERTY= class           @TYPE=String;)
(@PROPERTY= mulVal                @TYPE=String;)
(@PROPERTY= max_stage       @TYPE=String;)
(@PROPERTY= min_stage       @TYPE=String;)
(@PROPERTY= file_flooded_point    @TYPE=String;)
(@PROPERTY= stage           @TYPE=String;)
(@PROPERTY= res_vol               @TYPE=String;)
(@PROPERTY= res_area        @TYPE=String;)
(@PROPERTY= file            @TYPE=String;)

(@CLASS=    Structures      (@PROPERTIES= name class max_stage min_stage
file_flooded_point))
(@CLASS=    Dam             (@PROPERTIES= stage res_area res_vol))
(@CLASS=    Dyke)
(@CLASS=    Proposed_structures)
(@CLASS=    Proposed_dams)
(@CLASS=    Proposed_dykes)
(@CLASS=    Proposed_other)
(@CLASS=    Alternatives          (@PROPERTIES= name))
(@CLASS=    Current_Alternative)

(@OBJECT=   next_alt_num          (@PROPERTIES=Value      @TYPE=Integer;))
(@OBJECT=   alt             (@PROPERTIES= name file mulVal))
(@OBJECT=   add_structure         (@PROPERTIES=Value      @TYPE=String;))
(@OBJECT=   remove_structure (@PROPERTIES=Value     @TYPE=String;))
(@OBJECT=   Load_objects          (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   Flood           (@PROPERTIES=Value      @TYPE=Boolean;))

(@META=           alt.name          (@INITVAL=  ""))

(@METHOD=   Init
      (@ATOMID=Structures;@TYPE=CLASS;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (CreateObject     (SELF)                  (\SELF.class\))
            (=          (SELF.class)            ("Dam"))
      )
      (@RHS=
            (Assign           (SELF.max_stage)  (SELF.stage))
      )
)
(@METHOD=   IfChange
      (@ATOMID=add_structure;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (CreateObject     (\add_structure.Value\)
(|Proposed_structures|))
      )
      (@RHS=
            (SendMessage      ("Add")                 (@TO=<|Dam|>;))
            (SendMessage      ("Add")                 (@TO=<|Dyke|>;))
            (Assign           ("")              (add_structure))
      )
)
(@METHOD=   IfChange
      (@ATOMID=remove_structure;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (DeleteObject     (\remove_structure.Value\)
(|Proposed_structures|))
```

```
        )
        (@RHS=
              (SendMessage        ("Remove")          (@TO=<|Dam|>;))
              (SendMessage        ("Remove")          (@TO=<|Dyke|>;))
              (Assign             ("")                (remove_structure))
        )
)
(@METHOD=    Add
        (@ATOMID=Dam;@TYPE=CLASS;)
        (@FLAGS=PUBLIC;)
        (@LHS=
              (=             (SELF.name)        (add_structure.Value))
        )
        (@RHS=
              (CreateObject       (SELF)                    (|Proposed_dams|))
        )
)
(@METHOD=    Add
        (@ATOMID=Dyke;@TYPE=CLASS;)
        (@FLAGS=PUBLIC;)
        (@LHS=
              (=             (SELF.name)        (add_structure.Value))
        )
        (@RHS=
              (CreateObject       (SELF)                    (|Proposed_dykes|))
        )
)
(@METHOD=    Remove
        (@ATOMID=Dam;@TYPE=CLASS;)
        (@FLAGS=PUBLIC;)
        (@LHS=
              (=             (SELF.name)        (remove_structure.Value))
        )
        (@RHS=
              (DeleteObject       (SELF)                    (|Proposed_dams|))
        )
)
(@METHOD=    Remove
        (@ATOMID=Dyke;@TYPE=CLASS;)
        (@FLAGS=PUBLIC;)
        (@LHS=
              (=             (SELF.name)        (remove_structure.Value))
        )
        (@RHS=
              (DeleteObject       (SELF)                    (|Proposed_dykes|))
        )
)
(@METHOD=    Save_Alt
        (@ATOMID=alt;@TYPE=OBJECT;)
        (@FLAGS=PUBLIC;)
        (@LHS=
              (DeleteObject       (<|Alternatives|>)
(|Current_Alternative|))
              (Assign             (STRCAT("alt_",INT2STR(next_alt_num)))
(alt.name))
              (CreateObject       (\alt.name\)                  (|Alternatives|))
              (CreateObject       (\alt.name\)
(|Current_Alternative|))
              (Assign             (alt.name)              (\alt.name\.name))
              (Assign             (STRCAT("Dams_",alt.name))     (alt.file))
              (Write              ("@V(alt.file)")
(@TYPE="NXPDB";@FILL=NEW;\
```

```
@PROPS=stage,res_area,res_vol;@FIELDS="stage(10)","res_area(15)","res_vol(15)
";@ATOMS=<|Proposed_dams|>;))
            (Assign              (STRCAT("Dykes_",alt.name))    (alt.file))
            (Write               ("@V(alt.file)")
(@TYPE="NXPDB";@FILL=NEW;\

@PROPS=max_stage;@FIELDS="max_stage(10)";@ATOMS=<|Proposed_dykes|>;))
      )
      (@RHS=
            (Assign              (next_alt_num + 1)      (next_alt_num))
            (Write               ("alternatives.nxp")
(@TYPE="NXPDB";@FILL=INSERT;@ATOMS=<|Current_Alternative|>;))
      )
)
(@METHOD=   Load_Alt
      (@ATOMID=alt;@TYPE=OBJECT;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (<>         (alt.name)        (""))
            (DeleteObject     (<|Structures|>)   (|Proposed_structures|))
            (DeleteObject     (<|Structures|>)   (|Proposed_dams|))
            (DeleteObject     (<<|Structures|>>)       (|Proposed_dykes|))
      )
      (@RHS=
            (Assign              (STRCAT("Dams_",alt.name))    (alt.file))
            (Retrieve   ("@V(alt.file)")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|Proposed_dams|;\

@PROPS=name,stage,res_area,res_vol;@FIELDS="Name","stage","res_area(15)","res
_vol(15)";))
            (Assign              (STRCAT("Dykes_",alt.name))   (alt.file))
            (Retrieve   ("@V(alt.file)")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|Proposed_dykes|;\
                  @PROPS=name,max_stage;@FIELDS="Name","max_stage";))
            (CreateObject     (<|Proposed_dams|>)      (|Proposed_structures|))
            (CreateObject     (<|Proposed_dykes|>)     (|Proposed_structures|))
      )
)
(@METHOD=   Delete_Alt
      (@ATOMID=alt;@TYPE=OBJECT;)
      (@FLAGS=PUBLIC;)
      (@LHS=
            (<>         (alt.name)              (""))
            (DeleteObject     (\alt.name\)                  (|Alternatives|))
            (Write            ("alternatives.nxp")
(@TYPE="NXPDB";@FILL=NEW;\

@PROPS=name;@FIELDS="Name(30)";@ATOMS=<|Alternatives|>;))
      )
      (@RHS=
            (Assign              (STRCAT("Dams_",alt.name))    (alt.file))
            (Execute    ("rm @V(alt.file)")           (@TYPE=EXE;))
            (Assign              (STRCAT("Dykes_",alt.name))   (alt.file))
            (Execute    ("rm @V(alt.file)")           (@TYPE=EXE;))
      )
)
(@RULE=     Rf1
      @COMMENTS="Load the list of available structures.";
      (@LHS=
            (Retrieve   ("structures.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|Structures|;\

@PROPS=name,class,max_stage,min_stage,file_flooded_point;\
```

```
@FIELDS="Name","class","max_stage","min_stage","file_flooded_point";))
            (Retrieve    ("alternatives.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|Alternatives|;\
                         @PROPS=name;@FIELDS="Name";))
            (Execute     ("GetRelatives")  (@STRING="@ONELEVEL, @CHILDREN,
@RETURN=alt.mulVal";\
                                           @ATOMID=Alternatives;))
            (Execute     ("GetMultiValue")
(@ATOMID=alt.mulVal;@STRING="@LENGTH,\
                                           @RETURN=next_alt_num";))
            (Assign           (next_alt_num + 1)      (next_alt_num))
      )
      (@HYPO=     Load_objects)
      (@RHS=
            (SendMessage      ("Init")          (@TO=<|Structures|>;))
      )
)
(@RULE=     Rf2
      @COMMENTS="Set the flood input file, execute the flood script, and
retrieve reservoir volumes.";
      (@LHS=
            (Write            ("flood.nxp")     (@TYPE="NXPDB";@FILL=NEW;\
                        @PROPS=stage,file_flooded_point,class;\
                        @FIELDS="stage(6)","file(30)","class(10)";\
                        @ATOMS=<|Proposed_structures|>;))
            (Execute    ("gen_fl_file")         (@TYPE=EXE;))
            (Execute    ("flood < flood_file")  (@TYPE=EXE;))
            (Retrieve   ("volume.nxp")
(@TYPE="NXPDB";@FILL=ADD;@CREATE=|Dam|;\
                                    @PROPS=res_area,res_vol;\
                                    @FIELDS="res_area","res_vol";))
      )
      (@HYPO=     Flood)
      (@RHS=
            (Reset            (Flood))
      )
)
```

C.7 gis.tkb - knowledge base for GIS interface.

```
(@VERSION=  031)
(@PROPERTY= g_command   @TYPE=String;)
(@PROPERTY= r_DEM       @TYPE=Integer;)
(@PROPERTY= r_topography      @TYPE=Integer;)
(@PROPERTY= v_land            @TYPE=Integer;)
(@PROPERTY= v_wetlands  @TYPE=Integer;)
(@PROPERTY= v_contours  @TYPE=Integer;)
(@PROPERTY= v_streams   @TYPE=Integer;)
(@PROPERTY= v_roads           @TYPE=Integer;)
(@PROPERTY= v_structure @TYPE=String;)
(@PROPERTY= alt_topography    @TYPE=Integer;)
(@PROPERTY= alt_res_depth     @TYPE=Integer;)
(@PROPERTY= alt_reservoir     @TYPE=Integer;)

(@OBJECT=   GRASS       (@PROPERTIES=r_DEM r_topography\
                         v_streams v_roads v_land v_wetlands v_contours\
                         alt_topography alt_res_depth alt_reservoir\
                         g_command))

(@META=     GRASS.g_command         (@INITVAL=  ""))
(@META=     GRASS.r_DEM       (@INITVAL=  0))
(@META=     GRASS.r_topography      (@INITVAL=  0))
(@META=     GRASS.v_land            (@INITVAL=  0))
(@META=     GRASS.v_streams         (@INITVAL=  0))
(@META=     GRASS.v_roads           (@INITVAL=  0))
(@META=     GRASS.v_wetlands  (@INITVAL=  0))
(@META=     GRASS.v_contours  (@INITVAL=  0))
(@META=     GRASS.v_structure (@INITVAL=  ""))
(@META=     GRASS.alt_topography    (@INITVAL=  0))
(@META=     GRASS.alt_res_depth     (@INITVAL=  0))
(@META=     GRASS.alt_reservoir     (@INITVAL=  0))

(@METHOD=   IfChange
      (@ATOMID=GRASS.g_command;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@RHS=
            (Execute    ("@V(GRASS.g_command)") (@TYPE=EXE;))
            (Assign           ("")  (GRASS.g_command))
      )
)
(@METHOD=   IfChange
      (@ATOMID=GRASS.r_DEM;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@RHS=
            (Execute    ("d.rast DEM")    (@TYPE=EXE;))
            (Assign           (0)   (GRASS.r_DEM))
      )
)
(@METHOD=   IfChange
      (@ATOMID=GRASS.r_topography;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@RHS=
            (Execute    ("d.rast water")  (@TYPE=EXE;))
            (Assign           (0)   (GRASS.r_topography))
      )
)
(@METHOD=   IfChange
      (@ATOMID=GRASS.v_streams;@TYPE=SLOT;)
      (@FLAGS=PUBLIC;)
      (@RHS=
            (Execute    ("d.vect map=streams color=blue")   (@TYPE=EXE;))
            (Assign           (0)   (GRASS.v_streams))
```

```
        )
)
(@METHOD=    IfChange
        (@ATOMID=GRASS.v_roads;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
                (Execute    ("d.vect map=roads color=brown")    (@TYPE=EXE;))
                (Assign              (0)    (GRASS.v_roads))
        )
)
(@METHOD=    IfChange
        (@ATOMID=GRASS.v_land;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
                (Execute    ("d.vect map=vegetation color=black")
(@TYPE=EXE;))
                (Assign              (0)    (GRASS.v_land))
        )
)
(@METHOD=    IfChange
        (@ATOMID=GRASS.v_wetlands;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
                (Execute    ("d.vect map=wetlands color=aqua")  (@TYPE=EXE;))
                (Assign              (0)    (GRASS.v_wetlands))
        )
)
(@METHOD=    IfChange
        (@ATOMID=GRASS.v_contours;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
                (Execute    ("d.vect map=contours.09 color=grey")
(@TYPE=EXE;))
                (Execute    ("d.vect map=contours.10 color=grey")
(@TYPE=EXE;))
                (Assign              (0)    (GRASS.v_contours))
        )
)
(@METHOD=    IfChange
        (@ATOMID=GRASS.v_structure;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
                (Execute    ("d.vect map=@V(Grass.v_structure) color=black")
(@TYPE=EXE;))
                (Assign              ("")   (GRASS.v_structure))
        )
)
(@METHOD=    IfChange
        (@ATOMID=GRASS.alt_topography;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
                (Execute    ("d.rast flood.water")  (@TYPE=EXE;))
                (Assign              (0)    (GRASS.alt_topography))
        )
)
(@METHOD=    IfChange
        (@ATOMID=GRASS.alt_res_depth;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
                (Execute    ("d.rast flood.depth")  (@TYPE=EXE;))
                (Assign              (0)    (GRASS.alt_res_depth))
        )
)
(@METHOD=    IfChange
```

```
        (@ATOMID=GRASS.alt_reservoir;@TYPE=SLOT;)
        (@FLAGS=PUBLIC;)
        (@RHS=
              (Execute    ("d.rast flood.reser")  (@TYPE=EXE;))
              (Assign           (0)   (GRASS.alt_reservoir))
        )
)
```

C.8 hydro.tkb - hydropower dam design expert system knowledge base.

```
(@VERSION=  031)
(@PROPERTY= name                    @TYPE=String;)
(@PROPERTY= class                   @TYPE=String;)
(@PROPERTY= design                        @TYPE=String;)
(@PROPERTY= type                    @TYPE=String;)
(@PROPERTY= cost_of_earth_fill            @TYPE=String;)
(@PROPERTY= cost_of_rock_fill       @TYPE=String;)
(@PROPERTY= rock_ballast                  @TYPE=String;)
(@PROPERTY= dissipation_requirement       @TYPE=String;)
(@PROPERTY= op_reliability                @TYPE=String;)
(@PROPERTY= intake_channel                @TYPE=String;)
(@PROPERTY= conveyance              @TYPE=String;)
(@PROPERTY= intake_position_recommendation      @TYPE=String;)
(@PROPERTY= tailrace_lining               @TYPE=String;)
(@PROPERTY= operating_policy_class        @TYPE=String;)
(@PROPERTY= excavation_rock               @TYPE=String;)
(@PROPERTY= borrow_area             @TYPE=String;)
(@PROPERTY= frost_consideration           @TYPE=String;)
(@PROPERTY= borrow_quality                @TYPE=String;)
(@PROPERTY= quarry_area             @TYPE=String;)
(@PROPERTY= rock_quality                  @TYPE=String;)
(@PROPERTY= amount_of_explosives          @TYPE=String;)
(@PROPERTY= site_slope              @TYPE=String;)
(@PROPERTY= dyke_foundation               @TYPE=String;)
(@PROPERTY= potential_siltation           @TYPE=String;)
(@PROPERTY= potential_debris        @TYPE=String;)
(@PROPERTY= site_geology                  @TYPE=String;)
(@PROPERTY= penstock_conditions           @TYPE=String;)
(@PROPERTY= headrace_depth                @TYPE=String;)
(@PROPERTY= ice_formation                 @TYPE=String;)
(@PROPERTY= head                    @TYPE=Float;)
(@PROPERTY= turbine_unit_capacity         @TYPE=Float;)

(@CLASS=    Options           (@SUBCLASSES= Spillway Powerhouse))
(@CLASS=    Dam         (@PROPERTIES= cost_of_earth_fill cost_of_rock_fill\
                         excavation_rock borrow_area quarry_area\
                         rock_quality amount_of_explosives\
                         site_geology frost_consideration))
(@CLASS=    Dyke        (@PROPERTIES= rock_ballast dyke_foundation))
(@CLASS=    Reservoir   (@PROPERTIES= operating_policy_class\
                         potential_siltation potential_debris))
(@CLASS=    Release)
(@CLASS=    Spillway    (@PROPERTIES= dissipation_requirement op_reliability\
                         penstock_conditions headrace_depth ice_formation\
                         head turbine_unit_capacity))
(@CLASS=    Powerhouse  (@PROPERTIES= intake_channel conveyance\
                         intake_position_recommendation tailrace_lining))
(@CLASS=    Structures)

(@OBJECT=   structure               (@PROPERTIES= name))
(@OBJECT=   dam_experience                (@PROPERTIES=Value
@TYPE=String;))
(@OBJECT=   spillway_experience           (@PROPERTIES=Value
@TYPE=String;))

(@OBJECT=   Build_structure               (@PROPERTIES=Value
@TYPE=Boolean;))

(@OBJECT=   Choose_dam_type               (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Choose_dyke_type        (@PROPERTIES=Value      @TYPE=Boolean;))
```

```
(@OBJECT=   Assess_dyke_ballast_needs     (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Assess_spillway_dissipation_needs   (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Assess_spillway_op_reliability      (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Choose_powerhouse_type        (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Choose_powerhouse_design      (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Assess_intake_channel_needs   (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Recommend_intake_position     (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Assess_tailrace_needs         (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Choose_operating_policy_class (@PROPERTIES=Value
@TYPE=Boolean;))

(@OBJECT=   Choose_dam_class        (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   Estimate_earth_fill_cost      (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Estimate_rock_fill_cost       (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Choose_spillway_class         (@PROPERTIES=Value
@TYPE=Boolean;))
(@OBJECT=   Assess_conveyance_capacity    (@PROPERTIES=Value
@TYPE=Boolean;))

(@RULE=     HP_Main1
      @COMMENTS="Instigate rules to fire.";
      (@LHS=
            (=    (<|Structures|>.name)         (structure.name))
            (=    (<|Structures|>.class)        ("Dam"))
      )
      (@HYPO=     Build_structure)
      (@RHS=
            (Assign           (Choose_dam_type) (Choose_dam_type))
            (Reset            (Choose_dam_type))
            (Reset            (structure.name))
            (Reset            (Build_structure))
      )
)
(@RULE=     HP_Main2
      @COMMENTS="Instigate rules to fire.";
      (@LHS=
            (=    (<|Structures|>.name)         (structure.name))
            (=    (<|Structures|>.class)        ("Dyke"))
      )
      (@HYPO=     Build_structure)
      (@RHS=
            (Assign           (Choose_dyke_type)
(Choose_dyke_type))
            (Assign           (Assess_dyke_ballast_needs)
(Assess_dyke_ballast_needs))
            (Reset            (Choose_dyke_type))
            (Reset            (Assess_dyke_ballast_needs))
            (Reset            (structure.name))
            (Reset            (Build_structure))
      )
)
(@RULE=     HP_Main3
      @COMMENTS="Instigate rules to fire.";
      (@LHS=
```

```
        (=    (<|Structures|>.name)         (structure.name))
        (=    (<|Structures|>.class)        ("Spillway"))
    )
    (@HYPO=     Build_structure)
    (@RHS=
        (Assign     (Assess_spillway_dissipation_needs)
(Assess_spillway_dissipation_needs))
        (Assign     (Assess_spillway_op_reliability)
(Assess_spillway_op_reliability))
        (Reset      (Assess_spillway_dissipation_needs))
        (Reset      (Assess_spillway_op_reliability))
        (Reset            (structure.name))
        (Reset            (Build_structure))
    )
)
(@RULE=     HP_Main4
    @COMMENTS="Instigate rules to fire.";
    (@LHS=
        (=    (<|Structures|>.name)         (structure.name))
        (=    (<|Structures|>.class)        ("Powerhouse"))
    )
    (@HYPO=     Build_structure)
    (@RHS=
        (Assign           (Choose_powerhouse_type)
(Choose_powerhouse_type))
        (Assign           (Choose_powerhouse_design)
(Choose_powerhouse_design))
        (Assign           (Assess_intake_channel_needs)
(Assess_intake_channel_needs))
        (Assign           (Recommend_intake_position)
(Recommend_intake_position))
        (Assign           (Assess_tailrace_needs)
(Assess_tailrace_needs))
        (Reset            (Choose_powerhouse_type))
        (Reset            (Choose_powerhouse_design))
        (Reset            (Assess_intake_channel_needs))
        (Reset            (Recommend_intake_position))
        (Reset            (Assess_tailrace_needs))
        (Reset            (structure.name))
        (Reset            (Build_structure))
    )
)
(@RULE=     HP_Main5
    @COMMENTS="Instigate rules to fire.";
    (@LHS=
        (=    (<|Structures|>.name)         (structure.name))
        (=    (<|Structures|>.class)        ("Reservoir"))
    )
    (@HYPO=     Build_structure)
    (@RHS=
        (Assign           (Choose_operating_policy_class)
(Choose_operating_policy_class))
        (Reset            (Choose_operating_policy_class))
        (Reset            (structure.name))
        (Reset            (Build_structure))
    )
)

(@RULE=     HP_Dam1
    @COMMENTS="Assign type of dam construction.";
    @WHY="Earth fill cost is low due to accessibility, compared to quarrying
rock.";
    (@LHS=
        (=    (\structure.name\.type)             ("embankment"))
```

```
          (=    (\structure.name\.excavation_rock)  ("not available"))
          (=    (\structure.name\.borrow_area)            ("easy access"))
      )
      (@HYPO=     Choose_dam_type)
      (@RHS=      (Assign             ("earth fill")
(\structure.name\.design)))
)
(@RULE=     HP_Dam2
      @COMMENTS="Assign type of dam construction.";
      @WHY="Rock fill construction is more efficient, because of steep angles
of friction.";
      (@LHS=
          (=    (\structure.name\.type)             ("embankment"))
          (=    (\structure.name\.excavation_rock)  ("available"))
      )
      (@HYPO=     Choose_dam_type)
      (@RHS=      (Assign             ("rock fill")
(\structure.name\.design)))
)
(@RULE=     HP_Dam3
      @COMMENTS="Assign type of dam construction.";
      @WHY="Rock fill is relatively cheaper than earth fill.";
      (@LHS=
          (=    (\structure.name\.type)             ("embankment"))
          (=    (\structure.name\.cost_of_rock_fill)      ("low"))
          (=    (\structure.name\.cost_of_earth_fill)     ("high"))
      )
      (@HYPO=     Choose_dam_type)
      (@RHS=      (Assign             ("rock fill")
(\structure.name\.design)))
)
(@RULE=     HP_Dam4
      @COMMENTS="Assign type of dam construction.";
      @WHY="Earth fill is relatively cheaper than rock fill.";
      (@LHS=
          (=    (\structure.name\.type)       ("embankment"))
          (=    (\structure.name\.cost_of_rock_fill)      ("high"))
          (=    (\structure.name\.cost_of_earth_fill)     ("low"))
      )
      (@HYPO=     Choose_dam_type)
      (@RHS=      (Assign             ("earth fill")
(\structure.name\.design)))
)
(@RULE=     HP_Dam5
      @COMMENTS="Assign class of dam construction.";
      @WHY="Embankment dams are generally safer under severe frost
conditions.";
      (@LHS=      (=    (\structure.name\.frost_consideration)
("yes")))
      (@HYPO=     Choose_dam_class)
      (@RHS=      (Assign             ("embankment")
(\structure.name\.type)))
)
(@RULE=     HP_Dam6
      @COMMENTS="Assign class of dam construction.";
      @WHY="Embankment dams are feasible (cost of earth fill) and preferred.";
      (@LHS=
          (=    (\structure.name\.frost_consideration)          ("no"))
          (=    (dam_experience)                     ("embankment"))
          (=    (\structure.name\.cost_of_earth_fill)           ("low"))
      )
      (@HYPO=     Choose_dam_class)
      (@RHS=      (Assign             ("embankment")
(\structure.name\.type)))
```

```
)
(@RULE=     HP_Dam7
      @COMMENTS="Assign class of dam construction.";
      @WHY="Embankment dams are feasible (cost of rock fill) and preferred.";
      (@LHS=
            (=    (\structure.name\.frost_consideration)          ("no"))
            (=    (dam_experience)                          ("embankment"))
            (=    (\structure.name\.cost_of_rock_fill)            ("low"))
      )
      (@HYPO=     Choose_dam_class)
      (@RHS=      (Assign           ("embankment")
(\structure.name\.type)))
)
(@RULE=     HP_Dam8
      @COMMENTS="Estimate relative cost of earth fill.";
      @WHY="Low earth fill cost due to accessibility and quality of borrow
material.";
      (@LHS=
            (=    (\structure.name\.borrow_area)                  ("easy
access"))
            (=    (\structure.name\.borrow_quality)         ("good"))
      )
      (@HYPO=     Estimate_earth_fill_cost)
      (@RHS=      (Assign           ("low")
(\structure.name\.cost_of_earth_fill)))
)
(@RULE=     HP_Dam9
      @COMMENTS="Estimate relative cost of earth fill.";
      @WHY="High earth fill cost due to poor access conditions.";
      (@LHS=      (=    (\structure.name\.borrow_area)                  ("poor
access")))
      (@HYPO=     Estimate_earth_fill_cost)
      (@RHS=      (Assign           ("high")
(\structure.name\.cost_of_earth_fill)))
)
(@RULE=     HP_Dam10
      @COMMENTS="Estimate relative cost of earth fill.";
      @WHY="High earth fill cost due to poor quality of borrow material.";
      (@LHS=      (=    (\structure.name\.borrow_quality)         ("poor")))
      (@HYPO=     Estimate_earth_fill_cost)
      (@RHS=      (Assign           ("high")
(\structure.name\.cost_of_earth_fill)))
)
(@RULE=     HP_Dam11
      @COMMENTS="Estimate relative cost of rock fill.";
      @WHY="Rock fill is available on-site.";
      (@LHS=      (=    (\structure.name\.excavation_rock)
("available")))
      (@HYPO=     Estimate_rock_fill_cost)
      (@RHS=      (Assign           ("low")
(\structure.name\.cost_of_rock_fill)))
)
(@RULE=     HP_Dam12
      @COMMENTS="Estimate relative cost of rock fill.";
      @WHY="Rock fill is not available on-site, but is easily accessible and
of good quality.";
      (@LHS=
            (=    (\structure.name\.excavation_rock)        ("not available"))
            (=    (\structure.name\.quarry_area)                  ("easy
access"))
            (=    (\structure.name\.rock_quality)                 ("good"))
            (=    (\structure.name\.amount_of_explosives)
("reasonable"))
      )
```

```
      (@HYPO=     Estimate_rock_fill_cost)
      (@RHS=      (Assign            ("low")
(\structure.name\.cost_of_rock_fill)))
)
(@RULE=     HP_Dam13
      @COMMENTS="Estimate relative cost of rock fill.";
      @WHY="Rock fill is not available on-site, but is easily accessible and
of good quality.";
      (@LHS=
            (=    (\structure.name\.excavation_rock)        ("not available"))
            (=    (\structure.name\.quarry_area)                  ("easy
access"))
            (=    (\structure.name\.rock_quality)                 ("good"))
      )
      (@HYPO=     Estimate_rock_fill_cost)
      (@RHS=      (Assign            ("low")
(\structure.name\.cost_of_rock_fill)))
)
(@RULE=     HP_Dam14
      @COMMENTS="Estimate relative cost of rock fill.";
      @WHY="Rock fill is not available or easily accessible.";
      (@LHS=
            (=    (\structure.name\.excavation_rock)        ("not available"))
            (=    (\structure.name\.quarry_area)                  ("poor
access"))
      )
      (@HYPO=     Estimate_rock_fill_cost)
      (@RHS=      (Assign            ("high")
(\structure.name\.cost_of_rock_fill)))
)
(@RULE=     HP_Dam15
      @COMMENTS="Estimate relative cost of rock fill.";
      @WHY="Rock fill is not available and quarry material is of poor
quality.";
      (@LHS=
            (=    (\structure.name\.excavation_rock)        ("not available"))
            (=    (\structure.name\.rock_quality)                 ("poor"))
      )
      (@HYPO=     Estimate_rock_fill_cost)
      (@RHS=      (Assign            ("high")
(\structure.name\.cost_of_rock_fill)))
)
(@RULE=     HP_Dyke1
      @COMMENTS="Choose type of dyke construction.";
      @WHY="Earth fill dykes are normally used on flat surfaces.";
      (@LHS=      (=    (\structure.name\.site_slope)             ("flat")))
      (@HYPO=     Choose_dyke_type)
      (@RHS=      (Assign            ("earth fill")
(\structure.name\.design)))
)
(@RULE=     HP_Dyke2
      @COMMENTS="Choose type of dyke construction.";
      @WHY="A rock fill dyke may be required due to slope conditions.";
      (@LHS=      (=    (\structure.name\.site_slope)             ("steep")))
      (@HYPO=     Choose_dyke_type)
      (@RHS=      (Assign            ("rock fill")
(\structure.name\.design)))
)
(@RULE=     HP_Dyke3
      @COMMENTS="Assess the need for ballast with a dyke.";
      @WHY="Rock ballast is recommended due to geological conditions being
relatively soft.";
      (@LHS=      (=    (\structure.name\.dyke_foundation)        ("soft")))
      (@HYPO=     Assess_dyke_ballast_needs)
```

```
       (@RHS=      (Assign          ("yes")
(\structure.name\.rock_ballast)))
)
(@RULE=     HP_Spillway1
      @COMMENTS="Choose a class of spillway.";
      @WHY="Experience has been with overflow spillways, and flushing of
sediment is not a high priority.";
      (@LHS=
             (=    (spillway_experience)
("overflow"))
             (=    ('Reservoir_'\structure.name\.potential_siltation)
("low"))
      )
      (@HYPO=      Choose_spillway_class)
      (@RHS=       (Assign          ("overflow")
('Spillway_'\structure.name\.type)))
)
(@RULE=     HP_Spillway2
      @COMMENTS="Choose a class of spillway.";
      @WHY="Experience has been with orifice spillways, and build-up of
surface debris is not a problem.";
      (@LHS=
             (=    (spillway_experience)                       ("orifice"))
             (=    ('Reservoir_'\structure.name\.potential_debris)
("low","none"))
      )
      (@HYPO=      Choose_spillway_class)
      (@RHS=       (Assign          ("orifice")
('Spillway_'\structure.name\.type)))
)
(@RULE=     HP_Spillway3
      @COMMENTS="Choose a class of spillway.";
      @WHY="Priority is in management of debris.";
      (@LHS=
             (=    (spillway_experience)                       ("overflow
and orifice","none"))
             (=    ('Reservoir_'\structure.name\.potential_debris)
("high"))
             (=    ('Reservoir_'\structure.name\.potential_siltation)
("low","none"))
      )
      (@HYPO=      Choose_spillway_class)
      (@RHS=       (Assign          ("overflow")
('Spillway_'\structure.name\.type)))
)
(@RULE=     HP_Spillway4
      @COMMENTS="Choose a class of spillway.";
      @WHY="Priority is in management of siltation.";
      (@LHS=
             (=    (spillway_experience)                       ("overflow
and orifice","none"))
             (=    ('Reservoir_'\structure.name\.potential_debris)
("low","none"))
             (=    ('Reservoir_'\structure.name\.potential_siltation)
("high"))
      )
      (@HYPO=      Choose_spillway_class)
      (@RHS=       (Assign          ("orifice")
('Spillway_'\structure.name\.type)))
)
(@RULE=     HP_Spillway5
      @COMMENTS="Assess the needs for energy dissipation methods with the
spillway";
```

```
      @WHY="Dissipation methods are required when downstream bed material is
erodible.";
      (@LHS=       (=    (\structure.name\.site_geology)
("erodible")))
      (@HYPO=      Assess_spillway_dissipation_needs)
      (@RHS=       (Assign           ("yes")
('Spillway_'\structure.name\.dissipation_requirement)))
      (@EHS=       (Assign           ("no")
('Spillway_'\structure.name\.dissipation_requirement)))
)
(@RULE=     HP_Spillway6
      @COMMENTS="Assess the operational reliability of a spillway.";
      @WHY="Overflow spillways are known to be very reliable.";
      (@LHS=       (=    ('Spillway_'\structure.name\.type)
("overflow")))
      (@HYPO=      Assess_spillway_op_reliability)
      (@RHS=       (Assign           ("good")
('Spillway_'\structure.name\.op_reliability)))
)
(@RULE=     HP_Powerhouse1
      @COMMENTS="Choose the generating station system.";
      @WHY="Close couple systems work well for low head stations.";
      (@LHS=       (<    ('Powerhouse_'\structure.name\.head)
(25)))
      (@HYPO=      Choose_powerhouse_type)
      (@RHS=       (Assign           ("close couple")
('Powerhouse_'\structure.name\.type)))
)
(@RULE=     HP_Powerhouse2
      @COMMENTS="Choose the generating station system.";
      @WHY="Penstock systems are normally used for high head stations.";
      (@LHS=
            (>    ('Powerhouse_'\structure.name\.head)                 (30))
            (=    ('Powerhouse_'\structure.name\.penstock_conditions)
("suitable"))
      )
      (@HYPO=      Choose_powerhouse_type)
      (@RHS=       (Assign           ("penstock")
('Powerhouse_'\structure.name\.type)))
)
(@RULE=     HP_Powerhouse3
      @COMMENTS="Choose the type of turbine power system.";
      @WHY="Because both head and expected turbine capacity are relatively
high.";
      (@LHS=
            (>    ('Powerhouse_'\structure.name\.head)                 (15))
            (>    ('Powerhouse_'\structure.name\.turbine_unit_capacity) (70))
      )
      (@HYPO=      Choose_powerhouse_design)
      (@RHS=       (Assign           ("vertical shaft turbine")
('Powerhouse_'\structure.name\.design)))
)
(@RULE=     HP_Powerhouse4
      @COMMENTS="Choose the type of turbine power system.";
      @WHY="Because both head and expected turbine capacity are relatively
low.";
      (@LHS=
            (<    ('Powerhouse_'\structure.name\.head)                 (15))
            (<    ('Powerhouse_'\structure.name\.turbine_unit_capacity) (65))
      )
      (@HYPO=      Choose_powerhouse_design)
      (@RHS=       (Assign           ("bulb turbine")
('Powerhouse_'\structure.name\.design)))
)
```

```
(@RULE=     HP_Powerhouse5
      @COMMENTS="Decide whether an intake channel is necessary.";
      @WHY="An intake channel will provide for a lack of natural conveyance.";
      (@LHS=
            (=    ('Powerhouse_'\structure.name\.conveyance)
("insufficient"))
      )
      (@HYPO=     Assess_intake_channel_needs)
      (@RHS=      (Assign           ("yes")
('Powerhouse_'\structure.name\.intake_channel)))
)
(@RULE=     HP_Powerhouse6
      @COMMENTS="Decide whether an intake channel is necessary.";
      @WHY="There is sufficient natural conveyance without an intake
channel.";
      (@LHS=      (=    ('Powerhouse_'\structure.name\.conveyance)
("sufficient")))
      (@HYPO=     Assess_intake_channel_needs)
      (@RHS=      (Assign           ("no")
('Powerhouse_'\structure.name\.intake_channel)))
)
(@RULE=     HP_Powerhouse7
      @COMMENTS="Determine if there is sufficient conveyance for water
intake.";
      @WHY="Headrace depth is sufficient, and ice formation in the forebay is
complete.";
      (@LHS=
            (=    ('Powerhouse_'\structure.name\.headrace_depth)
("sufficient"))
            (=    ('Powerhouse_'\structure.name\.ice_formation)
("yes"))
      )
      (@HYPO=     Assess_conveyance_capacity)
      (@RHS=      (Assign           ("sufficient")
('Powerhouse_'\structure.name\.conveyance)))
)
(@RULE=     HP_Powerhouse8
      @COMMENTS="Determine if there is sufficient conveyance for water
intake.";
      @WHY="Headrace depth is insufficient for necessary conveyance.";
      (@LHS=      (=    ('Powerhouse_'\structure.name\.headrace_depth)
("insufficient")))
      (@HYPO=     Assess_conveyance_capacity)
      (@RHS=      (Assign           ("insufficient")
('Powerhouse_'\structure.name\.conveyance)))
)
(@RULE=     HP_Powerhouse9
      @COMMENTS="Determine if there is sufficient conveyance for water
intake.";
      @WHY="Partial ice cover may negatively affect conveyance for water
intake.";
      (@LHS=      (=    ('Powerhouse_'\structure.name\.ice_formation)
("some")))
      (@HYPO=     Assess_conveyance_capacity)
      (@RHS=      (Assign           ("insufficient")
('Powerhouse_'\structure.name\.conveyance)))
)
(@RULE=     HP_Powerhouse10
      @COMMENTS="Determine if there is sufficient conveyance for water
intake.";
      @WHY="Headrace depth is sufficient, and ice formation in the forebay is
not a problem.";
      (@LHS=
```

```
            (=     ('Powerhouse_'\structure.name\.headrace_depth)
("sufficient"))
            (=     ('Powerhouse_'\structure.name\.ice_formation)
("no"))
      )
      (@HYPO=     Assess_conveyance_capacity)
      (@RHS=      (Assign           ("sufficient")
('Powerhouse_'\structure.name\.conveyance)))
)
(@RULE=     HP_Powerhouse11
      @COMMENTS="Recommend the relative positioning of the water intakes.";
      @WHY="Water intakes should be placed as high as possible for penstock
systems.";
      (@LHS=      (=     ('Powerhouse_'\structure.name\.type)
("penstock")))
      (@HYPO=     Recommend_intake_position)
      (@RHS=      (Assign           ("high")
('Powerhouse_'\structure.name\.intake_position_recommendation)))
)
(@RULE=     HP_Powerhouse12
      @COMMENTS="Determine if artificial hardening is required along the
tailrace.";
      @WHY="A lining is required if local materials are prone to erosion.";
      (@LHS=      (=     (\structure.name\.site_geology)
("erodible")))
      (@HYPO=     Assess_tailrace_needs)
      (@RHS=      (Assign           ("yes")
('Powerhouse_'\structure.name\.tailrace_lining)))
      (@EHS=      (Assign           ("no")
('Powerhouse_'\structure.name\.tailrace_lining)))
)
(@RULE=     HP_Reservoir1
      @COMMENTS="Select a type of reservoir operating policy.";
      @WHY="Run-of-river policies are typical for low head stations.";
      (@LHS=      (<     ('Powerhouse_'\structure.name\.head)
(18)))
      (@HYPO=     Choose_operating_policy_class)
      (@RHS=      (Assign           ("run-of-river")
('Reservoir_'\structure.name\.operating_policy_class)))
)
(@RULE=     HP_Reservoir2
      @COMMENTS="Select a type of reservoir operating policy.";
      @WHY="Peaking stations are economically efficient for high head
scenarios.";
      (@LHS=      (>     ('Powerhouse_'\structure.name\.head)
(25)))
      (@HYPO=     Choose_operating_policy_class)
      (@RHS=      (Assign           ("peaking")
('Reservoir_'\structure.name\.operating_policy_class)))
)
```

C.9 mcda.tkb - knowledge base for the fuzzy compromise approach interface.

```
(@VERSION=  031)
(@PROPERTY= name                @TYPE=String;)
(@PROPERTY= class               @TYPE=String;)
(@PROPERTY= type                @TYPE=String;)
(@PROPERTY= id                  @TYPE=Integer;)
(@PROPERTY= alt_id              @TYPE=Integer;)
(@PROPERTY= crit_id             @TYPE=Integer;)
(@PROPERTY= pos_id              @TYPE=Integer;)
(@PROPERTY= neg_id              @TYPE=Integer;)
(@PROPERTY= wt_id               @TYPE=Integer;)
(@PROPERTY= value_id            @TYPE=Integer;)
(@PROPERTY= distance_id         @TYPE=Integer;)
(@PROPERTY= rank                @TYPE=Integer;)
(@PROPERTY= x                   @TYPE=Float;)
(@PROPERTY= m                   @TYPE=Float;)
(@PROPERTY= parameter           @TYPE=Float;)
(@PROPERTY= goal                @TYPE=String;)
(@PROPERTY= measure             @TYPE=Float;)
(@PROPERTY= val                 @TYPE=Float;)
(@PROPERTY= code                @TYPE=String;)
(@PROPERTY= con                 @TYPE=Boolean;)
(@PROPERTY= dil                 @TYPE=Boolean;)
(@PROPERTY= cintf               @TYPE=Boolean;)
(@PROPERTY= c_res               @TYPE=Integer;)
(@PROPERTY= c_sens              @TYPE=Integer;)
(@PROPERTY= c_tol               @TYPE=Float;)
(@PROPERTY= c_npi               @TYPE=Boolean;)
(@PROPERTY= r_res               @TYPE=Integer;)
(@PROPERTY= r_tol               @TYPE=Float;)
(@PROPERTY= r_acc               @TYPE=Boolean;)
(@PROPERTY= r_wcog              @TYPE=Boolean;)

(@CLASS=    Problems            (@PROPERTIES= name))
(@CLASS=    Alternatives        (@PROPERTIES= name id distance_id))
(@CLASS=    Criteria            (@SUBCLASSES= IncludedCriteria ExcludedCriteria)
                                (@PROPERTIES= name id pos_id neg_id wt_id))
(@CLASS=    CriteriaValues      (@PROPERTIES= crit_id alt_id value_id))
(@CLASS=    Memberships         (@SUBCLASSES= MembershipValues)
                                (@PROPERTIES= id x m))

(@CLASS=    MembershipValues)
(@CLASS=    MembershipData)
(@CLASS=    IncludedCriteria)
(@CLASS=    ExcludedCriteria)

(@CLASS=    RankMethods (@SUBCLASSES= Acc WCoG SelectedRankingMeasures
SelectedAccMeasures SelectedWCoGMeasures)
                        (@PROPERTIES= class parameter))
(@CLASS=    Acc)
(@CLASS=    WCoG)
(@CLASS=    SelectedRankingMeasures)
(@CLASS=    SelectedAccMeasures)
(@CLASS=    SelectedWCoGMeasures)

(@CLASS=    Ranks       (@SUBCLASSES= AccRanks WCoGRanks CurrentRanks)
                        (@PROPERTIES= class alt_id rank measure parameter))
(@CLASS=    CurrentRanks)
(@CLASS=    AccRanks)
(@CLASS=    WCoGRanks)

(@CLASS=    Crisp       (@SUBCLASSES= CrispValue CrispPos CrispNeg CrispWt)
                        (@PROPERTIES= class crit_id alt_id val))
```

```
(@CLASS=    CrispPos)
(@CLASS=    CrispNeg)
(@CLASS=    CrispWt)
(@CLASS=    CrispValue)
(@CLASS=    CrispAlt1)
(@CLASS=    CrispAlt2)
(@CLASS=    CrispAlt3)
(@CLASS=    CrispFuzziness    (@PROPERTIES= class crit_id alt_id code))
(@CLASS=    CrispCodeBook     (@PROPERTIES= code id))

(@CLASS=    FuzzyOptions      (@PROPERTIES= con dil cintf))

(@OBJECT=   selected    (@PROPERTIES= crit_id alt_id id))
(@OBJECT=   set         (@PROPERTIES= c_res c_sens c_tol r_res r_tol \
                              c_npi r_acc r_wcog))
(@OBJECT=   Compro            (@CLASSES= FuzzyOptions))
(@OBJECT=   AccGoal           (@CLASSES= FuzzyOptions)
                        (@PROPERTIES= goal))
(@OBJECT=   MemberNode  (@PROPERTIES= x m))

(@OBJECT=   next_id_num       (@PROPERTIES=Value      @TYPE=Integer;))
(@OBJECT=   next_memb_num     (@PROPERTIES=Value      @TYPE=Integer;))
(@OBJECT=   next_crit_num     (@PROPERTIES=Value      @TYPE=Integer;))
(@OBJECT=   next_alt_num      (@PROPERTIES=Value      @TYPE=Integer;))
(@OBJECT=   next_val_num      (@PROPERTIES=Value      @TYPE=Integer;))
(@OBJECT=   memb_num    (@PROPERTIES=Value      @TYPE=String;))
(@OBJECT=   file_name   (@PROPERTIES=Value      @TYPE=String;))
(@OBJECT=   problem_name      (@PROPERTIES=Value      @TYPE=String;))
(@OBJECT=   problem_type      (@PROPERTIES=Value      @TYPE=String;))
(@OBJECT=   new_problem (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   load_problems     (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   load_problem      (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   load_nxp    (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   load_crisp  (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   load_fuzzy  (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   load_ranks  (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   search_procedure  (@PROPERTIES=Value      @TYPE=String;))
(@OBJECT=   run_compro  (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   run_rank    (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   save_nxp    (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   save_settings     (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   clear_problem     (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   load_default      (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   plot_metrics      (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   plot_set    (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   clear_members     (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   add_point   (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   delete_point      (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   add_set     (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   delete_set  (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   delete_criteria   (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   delete_alternative(@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   select_alt_val    (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   select_wt   (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   select_pos  (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   select_neg  (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   select_p    (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   select_1    (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   select_1p   (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   md_all      (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   import_set  (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   selected_rank_alt (@PROPERTIES=Value      @TYPE=Integer;))
(@OBJECT=   selected_rank_parm      (@PROPERTIES=Value      @TYPE=Float;))
(@OBJECT=   avail_ranks (@PROPERTIES=Value      @TYPE=String;))
```

```
(@OBJECT=   select_rank_alt        (@PROPERTIES=Value      @TYPE=Boolean;))
(@OBJECT=   select_rank (@PROPERTIES=Value      @TYPE=Boolean;))

(@OBJECT=   Initialize  (@PROPERTIES=Value      @TYPE=Boolean;))

(@META=           c_tol @FORMAT="u.00000d";)
(@META=           r_tol @FORMAT="u.0000d";)
(@META=           measure     @FORMAT="u.0000d";)
(@META=           search_procedure.Value  (@INITVAL="coarse"))
(@META=           next_id_num.Value (@INITVAL=1))
(@META=           next_memb_num.Value     (@INITVAL=1))
(@META=           next_crit_num.Value     (@INITVAL=1))
(@META=           next_alt_num.Value      (@INITVAL=1))
(@META=           next_val_num.Value      (@INITVAL=1))

(@METHOD=   IfChange
      (@ATOMID=load_problems;@TYPE=SLOT;) (@FLAGS=PUBLIC;)
      (@LHS=
            (Execute    ("find_problems") (@TYPE=EXE;))
            (Retrieve   ("fc_problems.nxp")     (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Problems|;@PROPS=name;@FIELDS="Name";))
      )
      (@RHS=      (Reset            (load_problems)))
)
(@METHOD=   IfChange
      (@ATOMID=clear_problem;@TYPE=SLOT;) (@FLAGS=PUBLIC;)
      (@LHS=
            (DeleteObject     (<|Alternatives|>))
            (DeleteObject     (<|Criteria|>))
            (DeleteObject     (<|CriteriaValues|>))
            (DeleteObject     (<|Memberships|>))
            (DeleteObject     (<|Crisp|>))
            (Reset            (next_id_num))
            (Reset            (next_memb_num))
            (Reset            (next_crit_num))
            (Reset            (next_alt_num))
            (Reset            (next_val_num))
      )
      (@RHS=      (Reset            (clear_problem)))
)
(@METHOD=   IfChange
      (@ATOMID=load_problem;@TYPE=SLOT;)  (@FLAGS=PUBLIC;)
      (@LHS=
            (Assign           (TRUE)            (clear_problem))
            (Execute    ("goto_problem @V(problem_name)")   (@TYPE=EXE;))
            (=          (problem_type)    ("fuzzy","crisp"))
      )
      (@RHS=
            (Reset            (load_problem))
            (Assign           (TRUE)            ('load_'\problem_type\))
      )
      (@EHS=
            (Reset            (load_problem))
            (Assign           (TRUE)            (load_nxp))
      )
)
(@METHOD=   IfChange
      (@ATOMID=load_nxp;@TYPE=SLOT;)      (@FLAGS=PUBLIC;)
      (@LHS=
            (=          (problem_type)    ("nxp"))
            (Retrieve   ("MCDMproblems/@V(problem_name)/fc_criteria.nxp")
                        (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Criteria|;\
                        @PROPS=name,id,pos_id,neg_id,wt_id;\
```

```
                    @FIELDS="name","id","pos_id","neg_id","wt_id";))
        (Retrieve   ("MCDMproblems/@V(problem_name)/fc_alt.nxp")
                    (@TYPE="NXPDB";@FILL=ADD;\
                    @CREATE=|Alternatives|;\
                    @PROPS=name,id;@FIELDS="name","id";))
        (Retrieve   ("MCDMproblems/@V(problem_name)/fc_id.nxp")
                    (@TYPE="NXPDB";@FILL=ADD;\
                    @CREATE=|CriteriaValues|;\
                    @PROPS=crit_id,alt_id,value_id;\
                    @FIELDS="crit_id","alt_id","value_id";))
        (Retrieve   ("MCDMproblems/@V(problem_name)/fc_memb.nxp")
                    (@TYPE="NXPDB";@FILL=ADD;\
                    @CREATE=|Memberships|;\
                    @PROPS=id,x,m;\
                    @FIELDS="id","x","m";))
    )
    (@RHS=
        (SendMessage      ("GetNextCritId") (@TO=<|Criteria|>;))
        (SendMessage      ("GetNextAltId")  (@TO=<|Alternatives|>;))
        (SendMessage      ("GetNextMembId") (@TO=<|Memberships|>;))
        (SendMessage      ("GetNextId")            (@TO=<|Memberships|>;))
        (Reset            (load_nxp))
    )
)
(@METHOD=   GetNextCritId
    (@ATOMID=Criteria;@TYPE=CLASS;)            (@FLAGS=PUBLIC;)
    (@LHS=       (>=          (SELF.id)    (next_crit_num)))
    (@RHS=       (Assign            (SELF.id+1)  (next_crit_num)))
)
(@METHOD=   GetNextAltId
    (@ATOMID=Alternatives;@TYPE=CLASS;) (@FLAGS=PUBLIC;)
    (@LHS=       (>=          (SELF.id)    (next_alt_num)))
    (@RHS=       (Assign            (SELF.id+1)  (next_alt_num)))
)
(@METHOD=   GetNextMembId
    (@ATOMID=Memberships;@TYPE=CLASS;)   (@FLAGS=PUBLIC;)
    (@LHS=       (Execute     ("AtomNameValue")
(@ATOMID=SELF;@STRING="@RETURN=memb_num";))
        (>=          (STR2INT(SUBSTRING(memb_num,4,5)))   (next_memb_num))
    )
    (@RHS=       (Assign            (STR2INT(SUBSTRING(memb_num,4,5))+1)
(next_memb_num)))
)
(@METHOD=   GetNextId
    (@ATOMID=Memberships;@TYPE=CLASS;)   (@FLAGS=PUBLIC;)
    (@LHS=       (>=          (SELF.id)    (next_id_num)))
    (@RHS=       (Assign            (SELF.id+1)  (next_id_num)))
)
(@METHOD=   IfChange
    (@ATOMID=load_fuzzy;@TYPE=SLOT;)    (@FLAGS=PUBLIC;)
    (@LHS=
        (Execute    ("fuzzy2nxp @V(problem_name) @V(file_name)")
(@TYPE=EXE;))
    )
    (@RHS=
        (Assign           (TRUE)            (load_nxp))
        (Reset            (load_fuzzy))
    )
)
(@METHOD=   IfChange
    (@ATOMID=load_crisp;@TYPE=SLOT;)    (@FLAGS=PUBLIC;)
    (@LHS=
        (Execute    ("crisp2nxp @V(problem_name) @V(file_name)")
(@TYPE=EXE;))
```

```
          (Retrieve     ("MCDMproblems/@V(problem_name)/fc_crisp.nxp")
                        (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Crisp|;\
                        @PROPS=class,crit_id,alt_id,val;\
                        @FIELDS="class","crit_id","alt_id","val";))
          (Retrieve     ("MCDMproblems/@V(problem_name)/fc_criteria.nxp")
                        (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Criteria|;\
                        @PROPS=name,id,pos_id,neg_id,wt_id;\
                        @FIELDS="name","id","pos_id","neg_id","wt_id";))
          (Retrieve     ("MCDMproblems/@V(problem_name)/fc_alt.nxp")
                        (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Alternatives|;\
                        @PROPS=name,id;@FIELDS="name","id";))
          (Retrieve     ("MCDMproblems/@V(problem_name)/fc_id.nxp")
                        (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|CriteriaValues|;\
                        @PROPS=crit_id,alt_id,value_id;\
                        @FIELDS="crit_id","alt_id","value_id";))
      )
      (@RHS=
          (SendMessage      ("InitCrisp")           (@TO=<|Crisp|>;))
          (SendMessage      ("GetNextCritId")  (@TO=<|Criteria|>;))
          (SendMessage      ("GetNextAltId")   (@TO=<|Alternatives|>;))
          (SendMessage      ("GetNextId")           (@TO=<|Memberships|>;))
          (Reset            (load_crisp))
      )
)
(@METHOD=    InitCrisp
      (@ATOMID=Crisp;@TYPE=CLASS;)  (@FLAGS=PUBLIC;)
      (@RHS=       (CreateObject     (SELF)       (\SELF.class\)))
)
(@METHOD=    IfChange
      (@ATOMID=load_ranks;@TYPE=SLOT;)    (@FLAGS=PUBLIC;)
      (@LHS=
          (Execute     ("rank2nxp @V(problem_name) rank.out")
(@TYPE=EXE;))
          (Retrieve     ("MCDMproblems/@V(problem_name)/fc_ranks.nxp")
                        (@TYPE="NXPDB";@FILL=ADD;\
                        @CREATE=|Ranks|;\
                        @PROPS=class,rank,alt_id,measure,parameter;\
                        @FIELDS="class","rank","alt","measure","parameter";))
      )
      (@RHS=
          (SendMessage      ("InitRanks")           (@TO=<|Ranks|>;))
          (SendMessage      ("InitMeasures")
(@TO=<|SelectedRankingMeasures|>;))
          (Reset            (load_ranks))
      )
)
(@METHOD=    InitRanks
      (@ATOMID=Ranks;@TYPE=CLASS;)  (@FLAGS=PUBLIC;)
      (@RHS=       (CreateObject     (SELF)       (\SELF.class\)))
)
(@METHOD=    InitMeasures
      (@ATOMID=SelectedRankingMeasures;@TYPE=CLASS;)  (@FLAGS=PUBLIC;)
      (@LHS=       (=            (SELF.class)       ("Acc")))
      (@RHS=       (CreateObject     (SELF)       (|SelectedAccMeasures|)))
      (@EHS=       (CreateObject     (SELF)       (|SelectedWCoGMeasures|)))
)
(@METHOD=    IfChange
      (@ATOMID=import_set;@TYPE=SLOT;)    (@FLAGS=PUBLIC;)
      (@LHS=
```

```
              (Execute    ("xy2nxp @V(problem_name) @V(file_name)
@V(selected.id) @V(next_memb_num)")
                          (@TYPE=EXE;))
              (Retrieve   ("MCDMproblems/@V(problem_name)/memb.nxp")
                          (@TYPE="NXPDB";@FILL=ADD;\
                          @CREATE=|Memberships|;\
                          @PROPS=id,x,m;\
                          @FIELDS="id","x","m";))
       )
       (@RHS=
              (SendMessage      ("GetNextMembId") (@TO=<|Memberships|>;))
              (SendMessage      ("SelectMembers") (@TO=<|Memberships|>;))
              (Reset            (import_set))
       )
)
(@METHOD=   IfChange
       (@ATOMID=save_nxp;@TYPE=SLOT;)       (@FLAGS=PUBLIC;)
       (@LHS=
              (Execute    ("save_problem @V(problem_name)")
(@TYPE=EXE;))
              (Write
("MCDMproblems/@V(problem_name)/fc_criteria.nxp")
                          (@TYPE="NXPDB";@FILL=NEW;\
                          @PROPS=id,pos_id,neg_id,wt_id;\
                          @FIELDS="id(4)","pos_id(7)","neg_id(7)","wt_id(6)";\
                          @ATOMS=<|Criteria|>;))
              (Write            ("MCDMproblems/@V(problem_name)/fc_alt.nxp")
                          (@TYPE="NXPDB";@FILL=NEW;\
                          @PROPS=id;\
                          @FIELDS="id(3)";\
                          @ATOMS=<|Alternatives|>;))
              (Write            ("MCDMproblems/@V(problem_name)/fc_id.nxp")
                          (@TYPE="NXPDB";@FILL=NEW;\
                          @PROPS=crit_id,alt_id,value_id;\
                          @FIELDS="crit_id(8)","alt_id(7)","value_id(9)";\
                          @ATOMS=<|CriteriaValues|>;))
              (Write            ("MCDMproblems/@V(problem_name)/fc_memb.nxp")
                          (@TYPE="NXPDB";@FILL=NEW;\
                          @PROPS=id,x,m;\
                          @FIELDS="id(5)","x(10)","m(10)";\
                          @ATOMS=<|Memberships|>;))
       )
       (@RHS=       (Reset            (save_nxp)))
)
(@METHOD=   IfChange
       (@ATOMID=save_settings;@TYPE=SLOT;) (@FLAGS=PUBLIC;)
       (@LHS=
              (Write
("MCDMproblems/@V(problem_name)/fc_rankparms.nxp")
                          (@TYPE="NXPDB";@FILL=NEW;\
                          @PROPS=parameter;\
                          @FIELDS="parameter(10)";\
                          @ATOMS=<|SelectedRankingMeasures|>;))
              (Write
("MCDMproblems/@V(problem_name)/fc_settings.nxp")
                          (@TYPE="NXP";@FILL=NEW;\
                          @ATOMS=set, Compro, AccGoal;))
       )
       (@RHS=       (Reset            (save_settings)))
)
(@METHOD=   IfChange
       (@ATOMID=run_compro;@TYPE=SLOT;)     (@FLAGS=PUBLIC;)
       (@LHS=
              (Execute    ("nxp2fuzzy @V(problem_name)")      (@TYPE=EXE;))
```

```
          (=          (search_procedure)       ("dynamic"))
     )
     (@RHS=
          (Execute     ("fc")              (@TYPE=EXE;))
          (Reset            (run_compro))
     )
     (@EHS=
          (Execute     ("fc1")             (@TYPE=EXE;))
          (Reset            (run_compro))
     )
)
(@METHOD=   IfChange
     (@ATOMID=run_rank;@TYPE=SLOT;)        (@FLAGS=PUBLIC;)
     (@LHS=
          (Execute     ("nxp2fuzzy @V(problem_name)")        (@TYPE=EXE;))
          (Execute     ("fr")              (@TYPE=EXE;))
     )
     (@RHS=
          (Assign           (TRUE)              (load_ranks))
          (Reset            (run_rank))
     )
)
(@METHOD=   IfChange
     (@ATOMID=plot_metrics;@TYPE=SLOT;)  (@FLAGS=PUBLIC;)
     (@LHS=       (Execute     ("graph compro.out")     (@TYPE=EXE;)))
     (@RHS=       (Reset             (plot_metrics)))
)
(@METHOD=   IfChange
     (@ATOMID=plot_set;@TYPE=SLOT;)        (@FLAGS=PUBLIC;)
     (@LHS=
          (Execute     ("goto_problem @V(problem_name)")    (@TYPE=EXE;))
          (Write            ("MCDMproblems/@V(problem_name)/fc_memb.nxp")
                       (@TYPE="NXPDB";@FILL=NEW;\
                       @PROPS=id,x,m;\
                       @FIELDS="id(5)","x(10)","m(10)";\
                       @ATOMS=<|Memberships|>;))
          (Execute     ("nxp2out @V(problem_name) @V(selected.id)")
(@TYPE=EXE;))
          (Execute     ("graph set.out") (@TYPE=EXE;))
     )
     (@RHS=       (Reset             (plot_set)))
)
(@METHOD=   NewMemberNode
     (@ATOMID=next_memb_num;@TYPE=SLOT;) (@FLAGS=PUBLIC;)
     (@LHS=
          (CreateObject     ('memb'\next_memb_num.Value\) (|Memberships|))
          (CreateObject     ('memb'\next_memb_num.Value\)
(|MembershipValues|))
          (Assign           (selected.id)
('memb'\next_memb_num.Value\.id))
          (Assign           (MemberNode.x)
('memb'\next_memb_num.Value\.x))
          (Assign           (MemberNode.m)
('memb'\next_memb_num.Value\.m))
     )
     (@RHS=
          (Assign           (next_memb_num.Value+1) (next_memb_num.Value))
     )
)
(@METHOD=   NewCriteria
     (@ATOMID=next_crit_num;@TYPE=SLOT;) (@FLAGS=PUBLIC;)
     (@LHS=
          (CreateObject     ('crit'\next_crit_num.Value\) (|Criteria|))
```

```
              (Assign             (next_crit_num.Value)
('crit'\next_crit_num.Value\.id))
              (Assign             (next_id_num.Value)
('crit'\next_crit_num.Value\.wt_id))
              (Assign             (next_id_num.Value+1)
('crit'\next_crit_num.Value\.pos_id))
              (Assign             (next_id_num.Value+2)
('crit'\next_crit_num.Value\.neg_id))
       )
       (@RHS=
              (Assign             (next_crit_num.Value+1) (next_crit_num.Value))
              (Assign             (next_id_num.Value+3)   (next_id_num.Value))
       )
)
(@METHOD=   NewAlternative
       (@ATOMID=next_alt_num;@TYPE=SLOT;)  (@FLAGS=PUBLIC;)
       (@LHS=
              (CreateObject       ('alt'\SELF.Value\)     (|Alternatives|))
              (Assign             (next_alt_num.Value)
('alt'\next_alt_num.Value\.id))
       )
       (@RHS=
              (SendMessage        ("InitNewAlt")          (@TO=<|Criteria|>;))
              (Assign             (next_alt_num.Value+1)  (next_alt_num.Value))
       )
)
(@METHOD=   InitNewAlt
       (@ATOMID=Criteria;@TYPE=CLASS;)     (@FLAGS=PUBLIC;)
       (@RHS=
              (CreateObject       ('id'\next_val_num.Value\)
(|CriteriaValues|))
              (Assign             (SELF.id)
('id'\next_val_num.Value\.crit_id))
              (Assign             (next_alt_num.Value)
('id'\next_val_num.Value\.alt_id))
              (Assign             (next_id_num.Value)
('id'\next_val_num.Value\.value_id))
              (Assign             (next_id_num.Value+1)
(next_id_num.Value))
       )
)
(@METHOD=   ClearMemberships
       (@ATOMID=MembershipValues;@TYPE=CLASS;)   (@FLAGS=PUBLIC;)
       (@RHS=       (DeleteObject     (SELF)
(|MembershipValues|)))
)

(@METHOD=   IfChange
       (@ATOMID=select_pos;@TYPE=SLOT;)    (@FLAGS=PUBLIC;)
       (@LHS=       (SendMessage      ("FindPosId")
(@TO=<|Criteria|>;)))
       (@RHS=
              (SendMessage        ("SelectMembers") (@TO=<|Memberships|>;))
              (Reset              (select_pos))
       )
)
(@METHOD=   FindPosId
       (@ATOMID=Criteria;@TYPE=CLASS;)     (@FLAGS=PUBLIC;)
       (@LHS=       (=             (SELF.id)          (selected.crit_id)))
       (@RHS=       (Assign             (SELF.pos_id)            (selected.id)))
)
(@METHOD=   IfChange
       (@ATOMID=select_neg;@TYPE=SLOT;)    (@FLAGS=PUBLIC;)
```

```
      (@LHS=      (SendMessage      ("FindNegId")
(@TO=<|Criteria|>;)))
      (@RHS=
            (SendMessage      ("SelectMembers") (@TO=<|Memberships|>;))
            (Reset            (select_neg))
      )
)
(@METHOD=   FindNegId
      (@ATOMID=Criteria;@TYPE=CLASS;)     (@FLAGS=PUBLIC;)
      (@LHS=      (=          (SELF.id)         (selected.crit_id)))
      (@RHS=      (Assign          (SELF.neg_id)          (selected.id)))
)
(@METHOD=   IfChange
      (@ATOMID=select_wt;@TYPE=SLOT;)     (@FLAGS=PUBLIC;)
      (@LHS=      (SendMessage      ("FindWtId")
(@TO=<|Criteria|>;)))
      (@RHS=
            (SendMessage      ("SelectMembers") (@TO=<|Memberships|>;))
            (Reset            (select_wt))
      )
)
(@METHOD=   FindWtId
      (@ATOMID=Criteria;@TYPE=CLASS;)     (@FLAGS=PUBLIC;)
      (@LHS=      (=          (SELF.id)         (selected.crit_id)))
      (@RHS=      (Assign          (SELF.wt_id)          (selected.id)))
)
(@METHOD=   IfChange
      (@ATOMID=select_alt_val;@TYPE=SLOT;)      (@FLAGS=PUBLIC;)
      (@LHS=      (SendMessage      ("FindValId")
(@TO=<|CriteriaValues|>;)))
      (@RHS=
            (SendMessage      ("SelectMembers") (@TO=<|Memberships|>;))
            (Reset            (select_alt_val))
      )
)
(@METHOD=   FindValId
      (@ATOMID=CriteriaValues;@TYPE=CLASS;)     (@FLAGS=PUBLIC;)
      (@LHS=
            (=          (SELF.crit_id)          (selected.crit_id))
            (=          (SELF.alt_id)           (selected.alt_id))
      )
      (@RHS=      (Assign          (SELF.value_id)         (selected.id)))
)
(@METHOD=   IfChange
      (@ATOMID=select_p;@TYPE=SLOT;)      (@FLAGS=PUBLIC;)
      (@LHS=      (Assign          (0)               (selected.id)))
      (@RHS=
            (SendMessage      ("SelectMembers") (@TO=<|Memberships|>;))
            (Reset            (select_p))
      )
)
(@METHOD=   IfChange
      (@ATOMID=select_1;@TYPE=SLOT;)      (@FLAGS=PUBLIC;)
      (@LHS=      (Assign          (-1)              (selected.id)))
      (@RHS=
            (SendMessage      ("SelectMembers") (@TO=<|Memberships|>;))
            (Reset            (select_1))
      )
)
(@METHOD=   IfChange
      (@ATOMID=select_1p;@TYPE=SLOT;)     (@FLAGS=PUBLIC;)
      (@LHS=      (Assign          (-2)              (selected.id)))
      (@RHS=
            (SendMessage      ("SelectMembers") (@TO=<|Memberships|>;))
```

```
            (Reset              (select_1p))
      )
)
(@METHOD=   SelectMembers
      (@ATOMID=Memberships;@TYPE=CLASS;)   (@FLAGS=PUBLIC;)
      (@LHS=
            (DeleteObject       (SELF)              (|MembershipValues|))
            (=            (SELF.id)    (selected.id))
      )
      (@RHS=      (CreateObject       (SELF)              (|MembershipValues|)))
)
(@METHOD=   DeleteMembers
      (@ATOMID=Memberships;@TYPE=CLASS;)   (@FLAGS=PUBLIC;)
      (@LHS=      (=            (SELF.id)          (selected.id)))
      (@RHS=      (DeleteObject       (SELF)))
)
(@METHOD=   IfChange
      (@ATOMID=clear_members;@TYPE=SLOT;) (@FLAGS=PUBLIC;)
      (@RHS=
            (SendMessage        ("DeleteMembers") (@TO=<|Memberships|>;))
            (Reset              (clear_members))
      )
)
(@METHOD=   IfChange
      (@ATOMID=md_all;@TYPE=SLOT;)   (@FLAGS=PUBLIC;)
      (@LHS=      (=            (md_all)           (TRUE)))
      (@RHS=      (CreateObject       (<|Memberships|>) (|MembershipData|)))
      (@EHS=      (DeleteObject       (<|Memberships|>) (|MembershipData|)))
)
(@METHOD=   IfChange
      (@ATOMID=delete_criteria;@TYPE=SLOT;)      (@FLAGS=PUBLIC;)
      (@RHS=
            (SendMessage        ("DeleteCriteria")       (@TO=<|Criteria|>;))
            (Reset              (delete_criteria))
      )
)
(@METHOD=   DeleteCriteria
      (@ATOMID=Criteria;@TYPE=CLASS;)      (@FLAGS=PUBLIC;)
      (@LHS=      (=            (SELF.id)          (selected.crit_id)))
      (@RHS=
            (Assign             (SELF.pos_id)            (selected.id))
            (SendMessage        ("DeleteMembers") (@TO=<|Memberships|>;))
            (Assign             (SELF.neg_id)            (selected.id))
            (SendMessage        ("DeleteMembers") (@TO=<|Memberships|>;))
            (Assign             (SELF.wt_id)             (selected.id))
            (SendMessage        ("DeleteMembers") (@TO=<|Memberships|>;))
            (DeleteObject       (SELF))
            (SendMessage        ("DeleteCriteriaValues")
(@TO=<|CriteriaValues|>;))
      )
)
(@METHOD=   DeleteCriteriaValues
      (@ATOMID=CriteriaValues;@TYPE=CLASS;)      (@FLAGS=PUBLIC;)
      (@LHS=      (=            (SELF.crit_id)           (selected.crit_id)))
      (@RHS=
            (Assign             (SELF.value_id)          (selected.id))
            (DeleteObject       (SELF))
            (SendMessage        ("DeleteMembers") (@TO=<|Memberships|>;))
      )
)
(@METHOD=   IfChange
      (@ATOMID=delete_alternative;@TYPE=SLOT;)  (@FLAGS=PUBLIC;)
      (@RHS=
            (SendMessage        ("DeleteAlternative")    (@TO=<|Alternatives|>;))
```

```
		(Reset			(delete_alternative))
	)
)
(@METHOD=	DeleteAlternative
	(@ATOMID=Alternatives;@TYPE=CLASS;) (@FLAGS=PUBLIC;)
	(@LHS=		(=		(SELF.id)		(selected.alt_id)))
	(@RHS=
		(Assign			(SELF.id)		(selected.id))
		(DeleteObject		(SELF))
		(SendMessage		("DeleteAlternativeValues")
(@TO=<|CriteriaValues|>;))
	)
)
(@METHOD=	DeleteAlternativeValues
	(@ATOMID=CriteriaValues;@TYPE=CLASS;)		(@FLAGS=PUBLIC;)
	(@LHS=		(=		(SELF.alt_id)			(selected.alt_id)))
	(@RHS=
		(Assign			(SELF.value_id)		(selected.id))
		(DeleteObject		(SELF))
		(SendMessage		("DeleteMembers") (@TO=<|Memberships|>;))
	)
)
(@METHOD=	IfChange
	(@ATOMID=select_rank_alt;@TYPE=SLOT;)			(@FLAGS=PUBLIC;)
	(@RHS=
		(SendMessage		("FindRankId2")		(@TO=<|Ranks|>;))
		(Reset			(select_rank_alt))
	)
)
(@METHOD=	IfChange
	(@ATOMID=select_rank;@TYPE=SLOT;)	(@FLAGS=PUBLIC;)
	(@RHS=
		(SendMessage		("FindRankId")		(@TO=<|Ranks|>;))
		(Reset			(select_rank))
	)
)
(@METHOD=	FindRankId
	(@ATOMID=Ranks;@TYPE=CLASS;)	(@FLAGS=PUBLIC;)
	(@LHS=
		(DeleteObject		(SELF)				(|CurrentRanks|))
		(=		(SELF.parameter)	(selected_rank_parm))
		(=		(SELF.class)		(avail_ranks))
	)
	(@RHS=		(CreateObject		(SELF)				(|CurrentRanks|)))
)
(@METHOD=	FindRankId2
	(@ATOMID=Ranks;@TYPE=CLASS;)	(@FLAGS=PUBLIC;)
	(@LHS=
		(DeleteObject		(SELF)				(|CurrentRanks|))
		(=		(SELF.alt_id)		(selected_rank_alt))
		(=		(SELF.class)		(avail_ranks))
	)
	(@RHS=		(CreateObject		(SELF)				(|CurrentRanks|)))
)
(@METHOD=	IfChange
	(@ATOMID=load_default;@TYPE=SLOT;)	(@FLAGS=PUBLIC;)
	(@LHS=		(Retrieve	("fc_default.nxp")	(@TYPE="NXP";)))
	(@RHS=		(Reset			(load_default)))
)
(@METHOD=	InitMethods
	(@ATOMID=RankMethods;@TYPE=CLASS;)	(@FLAGS=PUBLIC;)
	(@RHS=		(CreateObject		(SELF)		(\SELF.class\)))
)
```

```
(@RULE=     R1
      @COMMENTS="Load the ranking measure parameter options.";
      (@LHS=
            (Retrieve   ("fc_parameters.nxp")   (@TYPE="NXPDB";@FILL=ADD;\
                                    @CREATE=|RankMethods|;\
                                    @PROPS=class,parameter;\
                                    @FIELDS="class","parameter";))
      )
      (@HYPO=     Initialize)
      (@RHS=
            (SendMessage      ("Init")    (@TO=<|RankMethods|>;))
            (Assign           (TRUE)            (load_default))
      )
)
```

Appendix D. CPSS Scripts.

D.1 update.rules

```
echo Updating the evaluation criteria selection rules from the \"relate.txt\"
file ...

sort relate.txt > relate.tmp
sort objectives.txt > objectives.tmp
sort facts.txt > facts.tmp
join -o 1.2 2.3 relate.tmp objectives.tmp > rules-a.tmp
sort rules-a.tmp > rules-b.tmp
join -o 1.2 2.3 rules-b.tmp facts.tmp > rules-c.tmp
awk '{printf "%-40s\t%-40s\n", $1, $2}' rules-c.tmp > rules.tmp
sort rules.tmp > rules.txt
rm *.tmp

echo The Rulebase description has been dumped to a file called \"rules.txt\"
```

D.2 update.nxp

```
mv relate.nxp relate.nxp%
mv objectives.nxp objectives.nxp%
mv facts.nxp facts.nxp%
mv cat_facts.nxp cat_facts.nxp%
mv cat_obj.nxp cat_obj.nxp%

echo Formatting relationships for an nxpdb file ...
echo " Name|    id_obj|    id_fact|" > relate.nxp
echo "***************************" >> relate.nxp
awk '{printf "%5s|%9s|%9s|\n", "R" NR, $1, $2}' relate.txt >> relate.nxp
echo "***************************" >> relate.nxp

echo Formatting objectives for an nxpdb file ...
echo " Name|   id_obj|   id_class|                                      OF|" >
objectives.nxp
echo "*******************************************************************" >>
objectives.nxp
awk '{printf "%5s|%8s|%10s|%40s|\n", "O" $1, $1, $2, $3}' objectives.txt   >>
objectives.nxp
echo "*******************************************************************" >>
objectives.nxp

echo Formatting facts for an nxpdb file ...
echo " Name|   id_fact|   id_class|                                    fact|" >
facts.nxp
echo "*******************************************************************" >>
facts.nxp
awk '{printf "%5s|%9s|%10s|%40s|\n", "F" $1, $1, $2, $3}' facts.txt        >>
facts.nxp
echo "*******************************************************************" >>
facts.nxp

echo Formatting facts subclasses for an nxpdb file ...
echo " Name| id_class|                                     class|" >
cat_facts.nxp
echo "*********************************************************" >>
cat_facts.nxp
```

```
awk '{printf "%5s|%9s|%40s|\n", "CF" $1, $1, $2}' cat_facts.txt  >>
cat_facts.nxp
echo "**************************************************************" >>
cat_facts.nxp

echo Formatting objectives subclasses for an nxpdb file ...
echo " Name| id_class|                                        class|"  >
cat_obj.nxp
echo "**************************************************************" >>
cat_obj.nxp
awk '{printf "%5s|%9s|%40s|\n", "CO" $1, $1, $2}' cat_obj.txt    >>
cat_obj.nxp
echo "**************************************************************" >>
cat_obj.nxp

echo Formatting available structures for an nxpdb file ...
echo "                  Name| class|     stage|              file_flooded_point|"
 > structures.nxp
echo "*****************************************************************************"
>> structures.nxp
awk '{printf "%20s|%6s|%10s|%30s|\n", $1, $2, $3, $4}' structures.txt
>> structures.nxp
echo "*****************************************************************************"
>> structures.nxp

echo Formatting available alternatives for an nxpdb file ...
echo "                            Name|"          > alternatives.nxp
echo "*******************************"         >> alternatives.nxp
awk '{printf "%30s|\n", $1}' alternatives.txt >> alternatives.nxp
echo "*******************************"         >> alternatives.nxp
```

D.3 digit

```
GIS_LOCK=$$
export GIS_LOCK
v.digit
```

D.4 flood

```
g.remove rast=flood.depth,flood.water
rm volume.db

read structures
echo "$structures " > new

if [ ! -f old_flood_file ]
then
	echo 'not available' > old_flood_file
fi

awk 'NR<=1 { print }' old_flood_file > old

cmp new old > differences
if [ -s differences ]
then
	if [ -f $MAPSET/cell/DEM.mod ]
	then
		g.remove rast=DEM.mod
	fi
	modifyDEM $structures
fi

rm new old differences old_flood_file
cp flood_file old_flood_file

number=1
while read dam el ptfile
do
	echo 'GENERATING A FLOOD FOR DAM ' $number ' ...'
	land=`expr $el + 1`

	echo 'Generating a surface of potential flooding ...'
	echo "1 thru $el = 1 potential flooding
$land thru 300 = 0 land" | r.reclass input=DEM.mod output=flood.b

	echo 'Isolating the reservoir ...'
	r.clump -q input=flood.b output=flood.c
	r.what input=flood.c < $ptfile > temp.1
	sed 's/|/ /g' temp.1 > temp.2
	awk '{print $3 " = 4 flooded area"}' temp.2 | r.reclass input=flood.c
output=flood.r
	rm temp.1 temp.2

	echo 'Generating a map of the flooded area ...'
	r.mapcalc flood.area = flood.r

	echo 'Generating a flooded depth surface...'
	echo '4 = ' $el ' flood elevation' | r.reclass input=flood.r
output=flood.d
	r.mapcalc flood.depth = 'if(flood.d-DEM,flood.d-DEM,1,0)'

	echo 'Calculating added reservoir volume ...'
```

```
      echo "$dam:" > flood.tmp
      r.volume -fq data=flood.depth clump=flood.r >> flood.tmp
      tr '\12' ' ' < flood.tmp > flood.volume
      rm flood.tmp

awk '
      BEGIN {FS=":"}
{
            area = $5 * 30 * 30 / 1000000
            volume = $8 / 1000000000
            print $1 " " area " " volume
} ' flood.volume >> volume.db

      g.remove rast=flood.b,flood.c,flood.d,flood.r
      echo "g.rename rast=flood.area,flood.area.$number" | sh
      echo "g.rename rast=flood.depth,flood.depth.$number" | sh
      nd=$number
      number=`expr $number + 1`
done

echo "Generating a flood map for $nd reservoirs ..."

if [ ! -f $MAPSET/cell/water.only ]
then
      echo '1 2 = 1 water' | r.reclass input=water output=water.only
fi

if [ $nd = "1" ]
then
      g.rename rast=flood.area.1,flood.reser
      g.rename rast=flood.depth.1,flood.depth
elif [ $nd = "2" ]
then
      r.patch -q input=flood.area.1,flood.area.2 output=flood.reser
      r.patch -q input=flood.depth.1,flood.depth.2 output=flood.depth
      g.remove rast=flood.area.1,flood.area.2,flood.depth.1,flood.depth.2
elif [ $nd = "3" ]
then
      r.patch -q input=flood.area.1,flood.area.2,flood.area.3
output=flood.reser
      r.patch -q input=flood.depth.1,flood.depth.2,flood.depth.3
output=flood.depth
      g.remove
rast=flood.area.1,flood.area.2,flood.area.3,flood.depth.1,flood.depth.2,flood
.depth.3
elif [ $nd = "4" ]
then
      r.patch -q input=flood.area.1,flood.area.2,flood.area.3,flood.area.4
output=flood.reser
      r.patch -q input=flood.depth.1,flood.depth.2,flood.depth.3,flood.depth.4
output=flood.depth
      g.remove
rast=flood.area.1,flood.area.2,flood.area.3,flood.area.4,flood.depth.1,flood.
depth.2,flood.depth.3,flood.depth.4
fi

r.patch -q input=water.only,flood.reser,water output=flood.water

awk '
      BEGIN {
            print "                    Name|       res_area|        res_vol|"
            print "*****************************************************"
            }
{ printf "%20s|%15s|%15s|\n", $1, $2, $3 }
```

```
        END     {
                print "******************************************************"
                }
   ' volume.db > volume.nxp

  cat color.depth | r.colors map=flood.depth color=rules
  cat color.flood | r.colors map=flood.water color=rules
  cat color.reser | r.colors map=flood.reser color=rules
```

D.5 gen_fl_file

```
sed 's/|/ /g' flood.nxp > tmp.1
sed 's/*//g' tmp.1 > tmp.2
awk 'NR >= 3 {print $1}' tmp.2 | tr -s '\12' ' ' > tmp.str
echo >> tmp.str
awk '/Dam/ { print $1 " " $2 " " $3 }' tmp.2 > tmp.dams
cat tmp.str tmp.dams > flood_file
rm tmp.*
```

D.6 modifyDEM

```
structures=`expr $#`

echo
echo $structures 'structures to add to the DEM!'

if [ $structures -ne 0 ]
then
      echo
      echo 'Adjusting the DEM ...'
      if [ $structures -eq 1 ]
      then
            r.patch -q input=$1,DEM output=DEM.mod
      elif [ $structures -eq 2 ]
      then
            r.patch -q input=$1,$2,DEM output=DEM.mod
      elif [ $structures -eq 3 ]
      then
            r.patch -q input=$1,$2,$3,DEM output=DEM.mod
      elif [ $structures -eq 4 ]
      then
            r.patch -q input=$1,$2,$3,$4,DEM output=DEM.mod
      elif [ $structures -eq 5 ]
      then
            r.patch -q input=$1,$2,$3,$4,$5,DEM output=DEM.mod
      elif [ $structures -eq 6 ]
      then
            r.patch -q input=$1,$2,$3,$4,$5,$6,DEM output=DEM.mod
      elif [ $structures -eq 7 ]
      then
            r.patch -q input=$1,$2,$3,$4,$5,$6,$7,DEM output=DEM.mod
      elif [ $structures -eq 8 ]
      then
            r.patch -q input=$1,$2,$3,$4,$5,$6,$7,$8,DEM output=DEM.mod
      fi
fi

if [ ! -f $MAPSET/cell/DEM.mod ]
then
        g.copy rast=DEM,DEM.mod
```

```
fi
```

D.7 dummy_script

```
FCSS=/home/ce/u3/mike/CPSS
cd $FCSS/MCDMproblems
if [ ! -d $1 ]
then
      x=$1
else
      echo "This problem already exists.  Renaming to "$1"1"
      x=$1"1"
fi
echo $x
```

D.8 find_problem

```
FCSS=/home/ce/u3/mike/CPSS
echo "                    Name|" > fc_problems.nxp
echo "*********************" >> fc_problems.nxp
ls -l $FCSS/MCDMproblems/ | awk ' /^d/ { printf "%20s|\n", $8 } ' - >>
fc_problems.nxp
echo "*********************" >> fc_problems.nxp
```

D.9 goto_problem

```
FCSS=/home/ce/u3/mike/CPSS

if [ "$#" -lt 1 ]
then
      echo "The goto_problem script demands 1 input:  {problem name}."
      exit 1
fi

if [ ! -d $FCSS/MCDMproblems/$1 ]
then
      echo $1 " does not exist."
      exit 1
fi
```

D.10 load_problem

```
if [ "$#" -lt 1 ]
then
      echo "The load_problem script demands 1 input:  {problem name}."
      exit 1
fi

FCSS=/home/ce/u3/mike/CPSS

if [ ! -d $FCSS/MCDMproblems/$1 ]
then
      echo $1 " does not exist."
      exit 1
fi
```

D.11 save_problem

```
FCSS=/home/ce/u3/mike/CPSS

if [ "$#" -lt 1 ]
then
      echo "The save_problem script demands 1 input:  {problem name}."
      exit 1
fi

if [ ! -d $FCSS/MCDMproblems/$1 ]
then
      mkdir $FCSS/MCDMproblems/$1
fi
```

D.12 crisp2nxp

```
FCSS=/home/ce/u3/mike/CPSS

if [ "$#" -lt 2 ]
then
	echo "The crisp2nxp script demands 2 inputs:  {problem name} {file
name}."
	exit 1
fi

if [ ! -f $FCSS/$2 ]
then
	echo "The input file does not exist."
	exit 1
fi

cd $FCSS/MCDMproblems
if [ ! -d $1 ]
then
	name=$1
else
	echo "This problem already exists.  Renaming to "$1"1"
	name=$1"1"
fi
mkdir $name
cd $name
cp $FCSS/$2 .

echo "  Name| id|" > fc_alt.nxp
echo "***********" >> fc_alt.nxp

echo "   Name| id| pos_id| neg_id| wt_id|" > fc_criteria.nxp
echo "*************************************" >> fc_criteria.nxp

echo "  Name| crit_id| alt_id| value_id|" > fc_id.nxp
echo "************************************" >> fc_id.nxp

echo "  Name|        class| crit_id| alt_id|             val|"  > fc_crisp.nxp
echo "*******************************************************" >> fc_crisp.nxp

awk '
(NR == 1) { nalt=$1 ; ncrit=$2 ; npos=nalt+1 ; nneg=npos+1 ; nwt=nneg+1 }
(NR >= 2) {
++row
for ( i=1 ; i <= ncrit ; i++ ) {
	++id
	if ( row <= nalt ) {
		name = "val" id
		class = "CrispValue"
		alt = row
		}
	if ( row == npos ) {
		name = "pos" id
		class = "CrispPos"
		alt = 0
		}
	if ( row == nneg ) {
		name = "neg" id
		class = "CrispNeg"
		alt = 0
		}
	if ( row == nwt ) {
```

```
            name = "wt" id
            class = "CrispWt"
            alt = 0
            }
      printf "%6s|%12s|%8s|%7s|%15s|\n", name, class, i, alt, $i
      }
}
END {
for ( j=1 ; j <= nalt ; j++ ) {
      name="alt" j
      printf "%6s|%3s|\n", name, j >> "fc_alt.nxp"
      }
for ( i=1 ; i <= ncrit ; i++ ) {
      name="crit" i
      printf "%7s|%3s|%7s|%7s|%6s|\n", name, i, ncrit+i, 2*ncrit+i, i >>
"fc_criteria.nxp"
      }
id=3*ncrit
for ( j=1 ; j <= nalt ; j++ )
      for ( i=1 ; i <= ncrit ; i++ ) {
            name="id" ++name_id
            printf "%6s|%8s|%7s|%9s|\n", name, i, j, ++id >> "fc_id.nxp"
            }
}
' $2 >> fc_crisp.nxp

echo "***********" >> fc_alt.nxp
echo "*************************************" >> fc_criteria.nxp
echo "***********************************" >> fc_id.nxp
echo "****************************************************" >> fc_crisp.nxp
```

D.13 fuzzy2nxp

```
FCSS=/home/ce/u3/mike/CPSS

if [ "$#" -lt 2 ]
then
	echo "The fuzzy2nxp script demands 2 inputs:  {problem name} {file
name}."
	exit 1
fi

if [ ! -f $FCSS/$2 ]
then
	echo "The input file does not exist."
	exit 1
fi

cd $FCSS/MCDMproblems
if [ ! -d $1 ]
then
	name=$1
else
	echo "This problem already exists.  Renaming to "$1"1"
	name=$1"1"
fi
mkdir $name
cd $name
cp $FCSS/$2 .

echo "  Name| id|" > fc_alt.nxp
echo "   Name| id| pos_id| neg_id| wt_id|" > fc_criteria.nxp
echo "  Name| crit_id| alt_id| value_id|" > fc_id.nxp
echo "      Name| id|          x|          m|" > fc_memb.nxp

echo "***********" >> fc_alt.nxp
echo "************************************" >> fc_criteria.nxp
echo "***********************************" >> fc_id.nxp
echo "**************************************" >> fc_memb.nxp

nawk '

BEGIN { nset=0 }

(NR == 1) { res=$1 ; print "\\set\.fc_res\\\=\"" res "\"" > "fc_settings.nxp"
}

(NR == 2) { sen=$1 ; print "\\set\.fc_sen\\\=\"" sen "\"" >> "fc_settings.nxp"
}

(NR == 3) { tol=$1 ; print "\\set\.fc_tol\\\=\"" tol "\"" >> "fc_settings.nxp"
}

(NR == 4) {
con=$1
if ( con == "1" )
	con="TRUE"
else
	con="FALSE"
print "\\set\.fc_con\\\=\"" con "\"" >> "fc_settings.nxp"
}

(NR == 5) {
dil=$1
```

```
if ( dil == "1" )
	dil="TRUE"
else
	dil="FALSE"
print "\\set\.fc_dil\\\=\"" dil "\"" >> "fc_settings.nxp"
}

(NR == 6) {
cintf=$1
if ( cintf == "1" )
	cintf="TRUE"
else
	cintf="FALSE"
print "\\set.fc_cintf\\=\"" cintf "\"" >> "fc_settings.nxp"
}

(NR == 7) {
pinv=$1
if ( pinv == "1" )
	pinv="TRUE"
else
	pinv="FALSE"
print "\\set\.fc_npi\\\=\"" pinv "\"" >> "fc_settings.nxp"
}

(NR >= 10) {
if ( NF == 1 ) {
	n=nset++
	npoints[n]=$1
	record="value"
	}
if (NF >= 2 ) {
	if (record == "membership")
		for ( i=1 ; i<=npoints[n] ; i++ )
			m[n,i]=$i
	else {
		record="membership"
		for ( i=1 ; i<=npoints[n] ; i++ )
			x[n,i]=$i
		}
	}
}

(NR == 8) {
nalt=$1
for ( j=1 ; j <= nalt ; j++ ) {
	name="alt" j
	printf "%6s|%3s|\n", name, j >> "fc_alt.nxp"
	}
}

(NR == 9) {
ncrit=$1
for ( i=1 ; i <= ncrit ; i++ ) {
	name="crit" i
	printf "%7s|%3s|%7s|%7s|%6s|\n", name, i, ncrit+i, 2*ncrit+i, i >>
"fc_criteria.nxp"
	}
id=3*ncrit
for ( j=1 ; j <= nalt ; j++ )
	for ( i=1 ; i <= ncrit ; i++ ) {
		name="id" ++name_id
		printf "%6s|%8s|%7s|%9s|\n", name, i, j, ++id >> "fc_id.nxp"
		}
```

```
}

END {
# print memberships
name_id=1
for ( id in npoints )
        for ( i=1 ; i <= npoints[id] ; i++ ) {
                name="memb" name_id++
                printf "%10s|%4s|%10s|%10s|\n", name, id, x[id,i], m[id,i] >>
"fc_memb.nxp"
                }
}
' $2

echo "***********" >> fc_settings.nxp
echo "***********" >> fc_alt.nxp
echo "***********************************" >> fc_criteria.nxp
echo "**********************************" >> fc_id.nxp
echo "**************************************" >> fc_memb.nxp

cd $FCSS
```

D.14 nxp2fuzzy

```
FCSS=/home/ce/u3/mike/CPSS

if [ "$#" -lt 1 ]
then
      echo "The nxp2fuzzy script demands 1 input:  {problem name}."
      exit 1
fi

if [ ! -d $FCSS/MCDMproblems/$1 ]
then
      echo "This problem does not exist."
      exit 1
fi

cd $FCSS/MCDMproblems/$1
sed -f $FCSS/convert_settings fc_settings.nxp > settings.dat

awk '
/r_res/    { res = $2 }
/r_tol/    { tol = $2 }
/AccGoal\.con/    { con = $2 }
/AccGoal\.dil/    { dil = $2 }
/AccGoal\.cintf/ { cintf = $2 }
/r_acc/    { acc = $2 }
/r_wcog/   { wcog = $2 }
END { print res "\n" tol "\n" con "\n" dil "\n" cintf "\n" acc "\n" wcog }
' settings.dat > rank.dat

sed 's/|/ /g' fc_rankparms.nxp | awk '

/acc/ {
numacc++
acc[numacc] = $2
}

/wcog/ {
numwcog++
wcog[numwcog] = $2
}

END {
if ( numacc >= 1 ) {
      stracc = numacc
      for ( k = 1; k <= numacc; k++ ) stracc = stracc " " acc[k]
      print stracc
      }

if ( numwcog >= 1 ) {
      strwcog = numwcog
      for ( k = 1; k <= numwcog; k++ ) strwcog = strwcog " " wcog[k]
      print strwcog
      }

} ' >> rank.dat

awk ' /goal/ { print "2\n0. " $2 "\n1. 0." } ' settings.dat >> rank.dat

awk '
/c_res/    { res = $2 }
/c_sens/   { sens = $2 }
/c_tol/    { tol = $2 }
```

```
/Compro\.con/    { con = $2 }
/Compro\.dil/    { dil = $2 }
/Compro\.cintf/ { cintf = $2 }
/c_npi/   { npi = $2 }
END { print res "\n" sens "\n" tol "\n" con "\n" dil "\n" cintf "\n" npi }
' settings.dat > compro.dat


sed 's/|/ /g' fc_criteria.nxp | awk ' (NR > 2) && ($1 !~ /\*/) { print $2 " "
$3 " " $4 " " $5 } ' > 0.tmp
sed 's/|/ /g' fc_alt.nxp | awk ' (NR > 2) && ($1 !~ /\*/) { print $2 } ' >
1.tmp
sed 's/|/ /g' fc_id.nxp | awk ' (NR > 2) && ($1 !~ /\*/) { print $2 " " $3 " "
$4 } ' > 2.tmp

sed 's/|/ /g' fc_memb.nxp | nawk '

FILENAME == "settings.dat" {
if ($1 == "c_npi") npi = $2
next
}

FILENAME == "0.tmp" {
numcriteria++
criteria[numcriteria] = $1
pos_id[$1] = $2
neg_id[$1] = $3
wt_id[$1] = $4
next
}

FILENAME == "1.tmp" {
numalt++
alt[numalt] = $1
next
}

FILENAME == "2.tmp" {
numcombos++
value_id[$2,$1] = $3
next
}

(NR > 2) && ($1 !~ /\*/) {
id = $1
set[id] = $2
x[id] = $3
m[id] = $4
if (set[id] == "-1") n1 = "TRUE"
# print id " " set[id]
}

END {
print numalt "\n" numcriteria

# find definition for p
np = 0
for (id in set) {
      if ( set[id] == "0" ) {
            ++np
            xset[np] = x[id]
            mset[np] = m[id]
            }
      }
```

```
sort(xset,mset,np)
xstring = xset[1]
mstring = mset[1]
for ( n = 2; n <= np; n++ ) {
      xstring = xstring " " xset[n]
      mstring = mstring " " mset[n]
      }
print np "\n" xstring "\n" mstring

# accumulate weights
for ( k = 1; k <= numcriteria; k++ ) {
      crit_id = wt_id[k]
      np = 0
      for (id in set) {
            if ( set[id] == crit_id ) {
                  ++np
                  xset[np] = x[id]
                  mset[np] = m[id]
                  }
            }
      sort(xset,mset,np)
      xstring = xset[1]
      mstring = mset[1]
      for ( n = 2; n <= np; n++ ) {
            xstring = xstring " " xset[n]
            mstring = mstring " " mset[n]
            }
      print np "\n" xstring "\n" mstring
      }

# accumulate positive ideals
for ( k = 1; k <= numcriteria; k++ ) {
      crit_id = pos_id[k]
      np = 0
      for (id in set) {
            if ( set[id] == crit_id ) {
                  ++np
                  xset[np] = x[id]
                  mset[np] = m[id]
                  }
            }
      sort(xset,mset,np)
      xstring = xset[1]
      mstring = mset[1]
      for ( n = 2; n <= np; n++ ) {
            xstring = xstring " " xset[n]
            mstring = mstring " " mset[n]
            }
      print np "\n" xstring "\n" mstring
      }

# accumulate negative ideals
for ( k = 1; k <= numcriteria; k++ ) {
      crit_id = neg_id[k]
      np = 0
      for (id in set) {
            if ( set[id] == crit_id ) {
                  ++np
                  xset[np] = x[id]
                  mset[np] = m[id]
                  }
            }
      sort(xset,mset,np)
      xstring = xset[1]
```

```
	mstring = mset[1]
	for ( n = 2; n <= np; n++ ) {
		xstring = xstring " " xset[n]
		mstring = mstring " " mset[n]
		}
	print np "\n" xstring "\n" mstring
	}

# accumulate criteria values
for ( j = 1; j <= numalt; j++ ) {
	for ( k = 1; k <= numcriteria; k++ ) {
		np = 0
		for (id in set) {
			if ( set[id] == value_id[j,k] ) {
				++np
				xset[np] = x[id]
				mset[np] = m[id]
				}
			}
		sort(xset,mset,np)
		xstring = xset[1]
		mstring = mset[1]
		for ( n = 2; n <= np; n++ ) {
			xstring = xstring " " xset[n]
			mstring = mstring " " mset[n]
			}
		print np "\n" xstring "\n" mstring
		}
	}

# find definition for "1"
if ( n1 == "TRUE" ) {
	np = 0
	for (id in set) {
		if ( set[id] == "-1" ) {
			++np
			xset[np] = x[id]
			mset[np] = m[id]
			}
		}
	sort(xset,mset,np)
	xstring = xset[1]
	mstring = mset[1]
	for ( n = 2; n <= np; n++ ) {
		xstring = xstring " " xset[n]
		mstring = mstring " " mset[n]
		}
	print np "\n" xstring"\n" mstring
	}
else {
#	print default fuzzy definition of "1"
	print "3\n0.99 1. 1.01\n0. 1. 0."
	}

# find definition for "1/p"
if ( npi == "1" ) {
	np = 0
	for (id in set) {
		if ( set[id] == "-2" ) {
			++np
			xset[np] = x[id]
			mset[np] = m[id]
			}
		}
```

```
      sort(xset,mset,np)
      xstring = xset[1]
      mstring = mset[1]
      for ( n = 2; n <= np; n++ ) {
            xstring = xstring " " xset[n]
            mstring = mstring " " mset[n]
            }
      print np "\n" xstring "\n" mstring
      }
print "0\n"
}

function sort(ARRAY,DEPENDENT,ELEMENTS,    temp,temp1,i,j) {
      for ( i = 2; i <= ELEMENTS; ++i )
            for ( j = i; ARRAY[j-1] > ARRAY[j]; --j) {
                  temp = ARRAY[j]
                  ARRAY[j] = ARRAY[j-1]
                  ARRAY[j-1] = temp

                  temp1 = DEPENDENT[j]
                  DEPENDENT[j] = DEPENDENT[j-1]
                  DEPENDENT[j-1] = temp1
                  }
return
}

' settings.dat 0.tmp 1.tmp 2.tmp - >> compro.dat

cp compro.dat $FCSS/.
cp rank.dat $FCSS/.
rm *.tmp
```

D.15 nxp2out

```
FCSS=/home/ce/u3/mike/CPSS

if [ "$#" -lt 2 ]
then
	echo "The nxp2out script demands 2 inputs:  {problem name} {membership
id}."
	exit 1
fi

if [ ! -d $FCSS/MCDMproblems/$1 ]
then
	echo "This problem does not exist."
	exit 1
fi
cd $FCSS/MCDMproblems/$1

sed 's/|/ /g' fc_memb.nxp | nawk ' ($2 == id) { print $3, $4 } ' id=$2 | nawk
'

{
++np
x[np] = $1
m[np] = $2
}

END {
# sort nodes for membership set
sort(x,m,np)
for ( k = 1; k <= np; k++ ) print x[k], m[k]
}

function sort(ARRAY,DEPENDENT,ELEMENTS,     temp,temp1,i,j) {
	for ( i = 2; i <= ELEMENTS; ++i )
		for ( j = i; ARRAY[j-1] > ARRAY[j]; --j) {
			temp = ARRAY[j]
			ARRAY[j] = ARRAY[j-1]
			ARRAY[j-1] = temp

			temp1 = DEPENDENT[j]
			DEPENDENT[j] = DEPENDENT[j-1]
			DEPENDENT[j-1] = temp1
			}
return
}

' - > set.out

cp set.out $FCSS/.
```

D.16 rank2nxp

```
FCSS=/home/ce/u3/mike/CPSS

if [ "$#" -lt 2 ]
then
	echo "The rank2nxp script demands 2 inputs:  {problem name} {file
name}."
	exit 1
fi

if [ ! -f $FCSS/$2 ]
then
	echo "The input file does not exist."
	exit 1
fi

cd $FCSS/MCDMproblems
if [ ! -d $1 ]
then
	mkdir $1
fi

cd $1
cp $FCSS/$2 .

echo "   Name|      class| rank|  alt|    measure| parameter|"  > fc_ranks.nxp
echo "******************************************************" >> fc_ranks.nxp

awk '
BEGIN { name="rank" }
/WCoG/ { class="WCoGRanks" }
/Acc/  { class="AccRanks" }
$1 ~ /[0-9]/ {
row++
printf "%7s|%10s|%5s|%5s|%10s|%10s|\n", name row, class, $1, $2, $3, $4
} ' $2 >> fc_ranks.nxp

echo "******************************************************" >> fc_ranks.nxp
```

D.17 xy2nxp

```
FCSS=/home/ce/u3/mike/CPSS

if [ "$#" -lt 4 ]
then
	echo "The xy2nxp script demands 4 inputs:"
	echo "{problem name} {file name} {selected id} {next id number}."
	exit 1
fi

if [ ! -f $FCSS/$2 ]
then
	echo "The input file does not exist."
	exit 1
fi

cd $FCSS/MCDMproblems
if [ ! -d $1 ]
then
	mkdir $1
fi
cd $1
cp $FCSS/$2 .

echo $3 " " $4 > temp

echo "      Name|   id|          x|          m|" > memb.nxp
echo "****************************************" >> memb.nxp

awk '
FILENAME == "temp" { id = $1 ; name_id = $2 ; next }
{ name="memb" name_id++ ; printf "%10s|%4s|%10s|%10s|\n", name, id, $1, $2 }
' temp $2 >> memb.nxp

echo "****************************************" >> memb.nxp

rm temp
```

Appendix E. CPSS FORTRAN Programs.

E.1 f_compro.f (fc)

```
c ******************************************************************
c
c           Fuzzy (COMPROMISE) Alternative Evaluation
c                 (for general [R+] fuzzy sets)
c
c           Problem size is restricted to ...
c                 mn+5n+m+2 < 500
c           ... to change, edit the array sizes
c           in both main and subroutine operation
c
c     r    resolution
c     s    sensitivity
c     t    tolerance
c     x    superset of all sets
c     u    membership values for x
c     m    alternatives
c     n    criteria
c     u*   membership values
c     np*  number of data points
c     p    distance metric exponent (level of compensation)
c     w    weights
c     zp   positive ideal criteria value
c     zn   negative ideal criteria value
c     dz   range of criteria values
c     d    alternative distance metric
c
c     pinv inverse of distance metric exponent
c     wp   weights raised to the power, p
c
c     op   operation
c
c ******************************************************************

        real x(500,0:200),u(500,0:200)

        integer m,n,r,s,con,dil,ci,ns(500),
     +add,subtract,multiply,divide,exponent

        open(1,file='compro.dat')
        open(2,file='compro.out')
      open(3,file='rank.in')

c*******************
c       Input block
c*******************

        read(1,*) r
      read(1,*) s
      read(1,*) t
        read(1,*) con
        read(1,*) dil
        read(1,*) ci
      read(1,*) npi
        read(1,*) m
        read(1,*) n
```

```
      if (m*n+5*n+m+3 .gt. 500) go to 999

      k=1
      read(1,*) ns(k)
        do 10 while (ns(k) .ne. 0)
        ns(k)=ns(k)-1
          read(1,*) (x(k,i),i=0,ns(k))
          read(1,*) (u(k,i),i=0,ns(k))
        k=k+1
        read(1,*) ns(k)
10      continue

      close(1)

      if (r .gt. 100) r=100
      if (r .lt. 10) r=10
      if (s .gt. 10000) s=10000
      if (s .lt. 100) s=100
      if (t .gt. 1.) t=1.
      if (t .lt. 1.E-7) t=1.E-7

      add=1
      subtract=2
      multiply=3
      divide=4
      exponent=5

c***********************************************
c       Compromise distance metric calculations
c             (order of operations)
c***********************************************
c     elements in the file are in the order: p,wi,zpi,zni,zij,1
c     other elements in x,u are calculated
      np=1
      nw=np+1
      nzp=nw+n
      nzn=nzp+n
      nz=nzn+n
      n1=nz+m*n

      npinv=n1+1
      ndz=npinv+1
      nwp=ndz+n
      nd=nwp+n
      nt=nd+m
      nes=nt+1

      write(*,*)
      write(*,*) 'Number of input data sets.',n1
      write(*,*) 'Total number of sets used in computations.',nt
      write(3,*) m
c     1/p
      if (npi .eq. 0) then
        call fuzzy(divide,n1,np,npinv,ns,x,u,r,s,t,con,dil,ci)
      end if
c     calculate dz,wp
      write(*,*) 'Calculating criteria ranges, weight metrics...'
      do 2 i=0,n-1
        call fuzzy(subtract,nzp+i,nzn+i,ndz+i,ns,x,u,r,s,t,con,dil,ci)
        call fuzzy(exponent,nw+i,np,nwp+i,ns,x,u,r,s,t,con,dil,ci)
2     continue
c     main loop to calculate distance metrics for alternatives
      do 3 j=0,m-1
        write(*,*) 'Calculating Lp for alternative ',j+1
```

```
      k=j*n
      kk=nd+j
      write(*,*) ' ... criteria  1'
      call fuzzy(subtract,nzp,nz+k,nes,ns,x,u,r,s,t,con,dil,ci)
      call fuzzy(divide,nes,ndz,nes+1,ns,x,u,r,s,t,con,dil,ci)
      call fuzzy(exponent,nes+1,np,nes+2,ns,x,u,r,s,t,con,dil,ci)
      call fuzzy(multiply,nwp,nes+2,kk,ns,x,u,r,s,t,con,dil,ci)
      do 4 i=1,n-1
        write(*,*) ' ... criteria',i+1
      call fuzzy(subtract,nzp+i,nz+k+i,nes,ns,x,u,r,s,t,con,dil,ci)
      call fuzzy(divide,nes,ndz+i,nes+1,ns,x,u,r,s,t,con,dil,ci)
      call fuzzy(exponent,nes+1,np,nes+2,ns,x,u,r,s,t,con,dil,ci)
      call fuzzy(multiply,nes+2,nwp+i,nes+3,ns,x,u,r,s,t,con,dil,ci)
        call fuzzy(add,kk,nes+3,kk,ns,x,u,r,s,t,con,dil,ci)
4     continue
      call fuzzy(exponent,kk,npinv,kk,ns,x,u,r,s,t,con,dil,ci)
      write(*,*) 'Writing Lp metric for alternative ',j+1
      write(3,*) ns(kk)+1
      write(3,90) (x(kk,i),i=0,ns(kk))
      write(3,90) (u(kk,i),i=0,ns(kk))
      do 5 i=0,ns(kk)
5       write (2,*) x(kk,i), u(kk,i)

3     continue

90    format(200f6.3)

999   stop
      end

c************************************
c       Fuzzy operation computations
c************************************

      subroutine fuzzy(op,na,nb,nc,ns,x,u,res,sen,tol,con,dil,cintf)

      real*8 l12,l13,l23,change

      real x(500,0:200),u(500,0:200),ba,step_ab,
     +a(0:200),ua(0:200),b(0:200),ub(0:200),c(0:200),uc(0:200),
     +temp,tol,u1,u2,step_size,step_min,test1,test2,yy

      integer ns(500),res,sen,na,nb,nc,op,con,dil,cintf,
     +add,subtract,multiply,divide,exponent,y

      zero=1.E-6

      add=1
      subtract=2
      multiply=3
      divide=4
      exponent=5

      npa=ns(na)
      npb=ns(nb)

      do 101 i=0,npa
            a(i)=x(na,i)
            ua(i)=u(na,i)
101   continue
      do 102 i=0,npb
            b(i)=x(nb,i)
            if (b(i) .eq. 0.) b(i)=zero
            ub(i)=u(nb,i)
```

```
102   continue
      if (op .eq. add) then
        c_min=a(0)+b(0)
        c_max=a(npa)+b(npb)
      else if (op .eq. subtract) then
        c_min=a(0)-b(npb)
        c_max=a(npa)-b(0)
      else if (op .eq. multiply) then
        c_min=a(0)*b(0)
        c_max=a(npa)*b(npb)
      else if (op .eq. divide) then
        c_min=a(0)/b(npb)
        c_max=a(npa)/b(0)
      else if (op .eq. exponent) then
        c_min=min(a(0)**b(0),a(npa)**b(npb),a(0)**b(npb),a(npa)**b(0))
        c_max=max(a(0)**b(0),a(npa)**b(npb),a(0)**b(npb),a(npa)**b(0))
      end if

      if (c_min .lt. zero) c_min=zero
      cc=c_max-c_min
      if (cc .lt. zero) cc=zero
      ba=b(npb)-b(0)
      step=cc/REAL(res)
      step_size=step
      step_min=step_size/10.

c*************************************************
c       Loop ... find each pt in the resulting set
c*************************************************
      k=0
      npoints=0
      nend=0
      c(k)=c_min
      do 120 while (c(k) .le. c_max)
        uc(k)=0.

c*****************************************************************************
c       Loop: test to find new point: eg. max[yy->R] min{u[i-1](y),u[i](x-y)}
c*****************************************************************************
        yy=b(0)
        step_ab=ba/REAL(sen+1)
        do 100 y=0,sen
          test2=yy

          if (op .eq. subtract) then
            test1=yy+c(k)
          else if (op .eq. multiply) then
            test1=c(k)/yy
          else if (op .eq. divide) then
            test1=c(k)*yy
          else if (op .eq. exponent) then
            test1=c(k)**(1./yy)
          else
            test1=c(k)-yy
          end if

          u2=0.
          u1=0.
        if ((test1 .lt. a(0)).or.(test1 .gt. a(npa))) go to 130
        l=0
        do 111 while (b(l) .lt. test2)
111     l=l+1
        if (b(l) .eq. test2) then
          u2=ub(l)
```

```
        else
          u2=ub(l-1)+(ub(l)-ub(l-1))*(test2-b(l-1))/(b(l)-b(l-1))
        end if
        if (con .eq. 1) u2=u2*u2
        if (dil .eq. 1) u2=u2**.5
        if (cintf .eq. 1) then
          if (u2 .lt. .5) u2=2.*u2*u2
          if (u2 .ge. .5) u2=1.-2.*(1.-u2)**2.
        end if
        l=0
        do 112 while (a(l) .lt. test1)
112     l=l+1
        if (a(l) .eq. test1) then
          u1=ua(l)
        else
          u1=ua(l-1)+(ua(l)-ua(l-1))*(test1-a(l-1))/(a(l)-a(l-1))
        end if
        if (con .eq. 1) u1=u1*u1
        if (dil .eq. 1) u1=u1**.5
        if (cintf .eq. 1) then
          if (u1 .lt. .5) u1=2.*u1*u1
          if (u1 .ge. .5) u1=1.-2.*(1.-u1)**2.
        end if
        temp=min(u1,u2)
        if (temp .gt. uc(k)) uc(k)=temp
130     yy=yy+step_ab
100   continue

c*************************************************************************
***
c       Test for corners using 3 points, change in length as function for
search
c*************************************************************************
***
      if (k .ge. 2 .and. npoints .eq. 2) then
        l12=((uc(k-1)-uc(k-2))**2.+((c(k-1)-c(k-2))/cc)**2.)**.5
        l23=((uc(k)-uc(k-1))**2.+((c(k)-c(k-1))/cc)**2.)**.5
        l13=((uc(k)-uc(k-2))**2.+((c(k)-c(k-2))/cc)**2.)**.5
        change=100.*(l12+l23-l13)/l13
        if (change .gt. tol .and. step_size .gt. step_min) then
          step_size=step_size/2.
          k=k-2
        else
          c(k-1)=c(k)
          uc(k-1)=uc(k)
          k=k-1
          step_size=step
        end if
        npoints=0
      end if

c**************************************************
c       Prepare for the next point in the added set
c**************************************************
      k=k+1
      npoints=npoints+1
      c(k)=c(k-1)+step_size
      if (c(k) .gt. c_max .and. nend .eq. 0) then
        c(k)=c_max
        nend=1
      end if

120   continue
```

```
c*********************************************************
c     save the operation result in the destination record
c*********************************************************
      k=k-1
      ns(nc)=k
      do 103 i=0,ns(nc)
            x(nc,i)=c(i)
            u(nc,i)=uc(i)
103   continue

      return
      end
```

E.2 f_compro1.f (fc1)

```
c *****************************************************************
c
c           Fuzzy (COMPROMISE) Alternative Evaluation
c                 (for simple [R+] fuzzy sets)
c
c           Problem size is restricted to ...
c                 mn+5n+m+2 < 500
c           ... to change, edit the array sizes
c           in both main and subroutine operation
c
c     r    resolution
c     s    sensitivity
c     x    superset of all sets
c     u    membership values for x
c     m    alternatives
c     n    criteria
c     u*   membership values
c     np*  number of data points
c     p    distance metric exponent (level of compensation)
c     w    weights
c     zp   positive ideal criteria value
c     zn   negative ideal criteria value
c     dz   range of criteria values
c     d    alternative distance metric
c
c     pinv inverse of distance metric exponent
c     wp   weights raised to the power, p
c
c     op   operation
c
c *****************************************************************

        real x(500,0:200),u(500,0:200)

        integer m,n,r,s,ns(500),con,dil,ci,
     +add,subtract,multiply,divide,exponent

        open(1,file='compro.dat')
        open(2,file='compro.out')
      open(3,file='rank.in')

c******************************
c       Input block
c     note that t is not used
c******************************

        read(1,*) r
      read(1,*) s
      read(1,*) t
        read(1,*) con
        read(1,*) dil
        read(1,*) ci
      read(1,*) npi
        read(1,*) m
        read(1,*) n

      k=1
      read(1,*) ns(k)
        do 10 while (ns(k) .ne. 0)
        ns(k)=ns(k)-1
          read(1,*) (x(k,i),i=0,ns(k))
```

```
           read(1,*) (u(k,i),i=0,ns(k))
         k=k+1
         read(1,*) ns(k)
10       continue

       close(1)

       add=1
       subtract=2
       multiply=3
       divide=4
       exponent=5

c**********************************************
c        Compromise distance metric calculations
c              (order of operations)
c**********************************************
c      elements in the file are in the order: p,wi,zpi,zni,zij,1
c      other elements in x,u are calculated
       np=1
       nw=np+1
       nzp=nw+n
       nzn=nzp+n
       nz=nzn+n
       n1=nz+m*n

       npinv=n1+1

       ndz=npinv+1
       nwp=ndz+n
       nd=nwp+n
       nt=nd+m
       ntmp=nt+1
       nes=ntmp+1

       write(*,*)
       write(*,*) 'Number of input data sets.',n1
       write(*,*) 'Total number of sets used in computations.',nt
       write(3,*) m

c      1/p
       if (npi .eq. 0) then
         call fuzzy(divide,n1,np,npinv,ns,x,u,r,s,con,dil,ci)
       end if
c      calculate dz,wp
       write(*,*) 'Calculating criteria ranges, weight metrics...'
       do 2 i=0,n-1
       call fuzzy(subtract,nzp+i,nzn+i,ndz+i,ns,x,u,r,s,con,dil,ci)
       call fuzzy(exponent,nw+i,np,nwp+i,ns,x,u,r,s,con,dil,ci)
2      continue
c      main loop to calculate distance metrics for alternatives
       do 3 j=0,m-1
         write(*,*) 'Calculating Lp for alternative ',j+1
         k=j*n
         kk=nd+j
         write(*,*) ' ... criteria  1'
       call fuzzy(subtract,nzp,nz+k,nes,ns,x,u,r,s,con,dil,ci)
       call fuzzy(divide,nes,ndz,nes+1,ns,x,u,r,s,con,dil,ci)
       call fuzzy(exponent,nes+1,np,nes+2,ns,x,u,r,s,con,dil,ci)
       call fuzzy(multiply,nwp,nes+2,kk,ns,x,u,r,s,con,dil,ci)
         do 4 i=1,n-1
           write(*,*) ' ... criteria',i+1
       call fuzzy(subtract,nzp+i,nz+k+i,nes,ns,x,u,r,s,con,dil,ci)
       call fuzzy(divide,nes,ndz+i,nes+1,ns,x,u,r,s,con,dil,ci)
```

```
      call fuzzy(exponent,nes+1,np,nes+2,ns,x,u,r,s,con,dil,ci)
      call fuzzy(multiply,nes+2,nwp+i,nes+3,ns,x,u,r,s,con,dil,ci)
      call fuzzy(add,kk,nes+3,kk,ns,x,u,r,s,con,dil,ci)
4       continue
      call fuzzy(exponent,kk,npinv,kk,ns,x,u,r,s,con,dil,ci)

        write(*,*) 'Writing Lp metric for alternative ',j+1
        write(3,*) ns(kk)+1
        write(3,90) (x(kk,i),i=0,ns(kk))
        write(3,90) (u(kk,i),i=0,ns(kk))
        do 5 i=0,ns(kk)
5         write (2,*) x(kk,i), u(kk,i)

3     continue

90    format(200f7.3)

999     stop
        end

c************************************
c       Fuzzy operation computations
c************************************

      subroutine fuzzy(op,na,nb,nc,ns,x,u,res,sen,con,dil,ci)

      real x(500,0:200),u(500,0:200),ba,step_ab,
     +a(0:200),ua(0:200),b(0:200),ub(0:200),c(0:200),uc(0:200),
     +temp,u1,u2,step_size,step_min,test1,test2,yy

      integer ns(500),res,sen,na,nb,nc,op,con,dil,ci,
     +add,subtract,multiply,divide,exponent,y

      zero=1.E-6

      add=1
      subtract=2
      multiply=3
      divide=4
      exponent=5

      npa=ns(na)
      npb=ns(nb)

      do 101 i=0,npa
            a(i)=x(na,i)
            ua(i)=u(na,i)
101   continue
      do 102 i=0,npb
            b(i)=x(nb,i)
            if (b(i) .eq. 0.) b(i)=zero
            ub(i)=u(nb,i)
102   continue
      if (op .eq. add) then
        c_min=a(0)+b(0)
        c_max=a(npa)+b(npb)
      else if (op .eq. subtract) then
        c_min=a(0)-b(npb)
        c_max=a(npa)-b(0)
      else if (op .eq. multiply) then
        c_min=a(0)*b(0)
        c_max=a(npa)*b(npb)
      else if (op .eq. divide) then
        c_min=a(0)/b(npb)
```

```
          c_max=a(npa)/b(0)
        else if (op .eq. exponent) then
          c_min=min(a(0)**b(0),a(npa)**b(npb),a(0)**b(npb),a(npa)**b(0))
          c_max=max(a(0)**b(0),a(npa)**b(npb),a(0)**b(npb),a(npa)**b(0))
        end if

        if (c_min .lt. zero) c_min=zero
        cc=c_max-c_min
        if (cc .lt. zero) cc=zero
        ba=b(npb)-b(0)
        step=cc/REAL(res)
        step_size=step
        step_min=step_size/10.

c*************************************************
c       Loop ... find each pt in the resulting set
c*************************************************
        k=0
        nend=0
        c(k)=c_min
        do 120 while (c(k) .le. c_max)
          uc(k)=0.

c**************************************************************************
c       Loop: test to find new point: eg. max[yy->R] min{u[i-1](y),u[i](x-y)}
c**************************************************************************
          yy=b(0)
          step_ab=ba/REAL(sen+1)
          do 100 y=0,sen
            test2=yy

            if (op .eq. subtract) then
              test1=yy+c(k)
            else if (op .eq. multiply) then
              test1=c(k)/yy
            else if (op .eq. divide) then
              test1=c(k)*yy
            else if (op .eq. exponent) then
              test1=c(k)**(1./yy)
            else
              test1=c(k)-yy
            end if

            u2=0.
            u1=0.
          if ((test1 .lt. a(0)).or.(test1 .gt. a(npa))) go to 130
          l=0
          do 111 while (b(l) .lt. test2)
111       l=l+1
          if (b(l) .eq. test2) then
            u2=ub(l)
          else
            u2=ub(l-1)+(ub(l)-ub(l-1))*(test2-b(l-1))/(b(l)-b(l-1))
          end if
          if (con .eq. 1) u2=u2*u2
          if (dil .eq. 1) u2=u2**.5
          if (cintf .eq. 1) then
            if (u2 .lt. .5) u2=2.*u2*u2
            if (u2 .ge. .5) u2=1.-2.*(1.-u2)**2.
          end if
          l=0
          do 112 while (a(l) .lt. test1)
112       l=l+1
          if (a(l) .eq. test1) then
```

```fortran
          u1=ua(l)
        else
          u1=ua(l-1)+(ua(l)-ua(l-1))*(test1-a(l-1))/(a(l)-a(l-1))
        end if
        if (con .eq. 1) u1=u1*u1
        if (dil .eq. 1) u1=u1**.5
        if (cintf .eq. 1) then
          if (u1 .lt. .5) u1=2.*u1*u1
          if (u1 .ge. .5) u1=1.-2.*(1.-u1)**2.
        end if
        temp=min(u1,u2)
        if (temp .gt. uc(k)) uc(k)=temp
130     yy=yy+step_ab
100   continue

c*************************************************
c       Prepare for the next point in the added set
c*************************************************
      k=k+1
      c(k)=c(k-1)+step_size
      if (c(k) .gt. c_max .and. nend .eq. 0) then
        c(k)=c_max
        nend=1
      end if

120   continue

c ********************************************************
c     save the operation result in the destination record
c ********************************************************
      k=k-1

      do 200 i=0,npa
        if (u(na,i) .eq. 1.) ca=x(na,i)
200   continue

      do 210 i=0,npb
        if (u(nb,i) .eq. 1.) cb=x(nb,i)
210   continue

          k=k+1
          if (op .eq. add) then
            c(k)=ca+cb
          else if (op .eq. subtract) then
            c(k)=ca-cb
          else if (op .eq. multiply) then
            c(k)=ca*cb
          else if (op .eq. divide) then
            c(k)=ca/cb
          else if (op .eq. exponent) then
            c(k)=ca**cb
          end if
          uc(k)=1.
      ns(nc)=k

      l=0
      do 311 while (c(l) .lt. c(k))
311     l=l+1
      if (l .lt. k) then
        ct=c(k)
        uct=uc(k)
        do 312 j=k-1,l,-1
          c(j+1)=c(j)
          uc(j+1)=uc(j)
```

```
312       continue
          c(l)=ct
          uc(l)=uct
        end if

c       do 333 j=0,ns(nc)
c         aa=c(j)
c         nalt=uc(j)
c         do 335 i=j-1,0,-1
c           if (c(i) .le. aa) goto 337
c           c(i+1)=c(i)
c           uc(i+1)=uc(i)
c335       continue
c         i=0
c337       c(i+1)=aa
c         uc(i+1)=nalt
c333    continue

        do 103 i=0,ns(nc)
              x(nc,i)=c(i)
              u(nc,i)=uc(i)
103     continue

        return
        end
```

E.3 f_rank.f (fr)

```
c ************************************************************
c
c                        FUZZY RANKING
c                 (for general [R+] fuzzy sets)
c
c           Problem size is restricted to ...
c                 nfs << 98
c           ... to change, edit the array sizes
c           in both main and subroutine operation
c
c     x    superset of all sets
c     u    membership values for x
c     u*   membership values
c     np*  number of data points
c     op   operation
c
c ************************************************************
c 3456789012345678901234567890123456789012345678901234567890123456789012

        real x(100,0:200),u(100,0:200),t,
     +rank(100,10),xbar(100,10),q(10),alpha(10)

        integer r,ns(100),alt1(100,10),alt2(100,10),do_acc,do_cog,
     +con,dil,cintf

        open(1,file='rank.in')
      open(2,file='rank.dat')
      open(3,file='rank.out')

c*******************
c       Input block
c*******************

        read(2,*) r
      read(2,*) t
      read(2,*) con
      read(2,*) dil
      read(2,*) cintf
      read(2,*) do_acc
      read(2,*) do_cog
      if (do_acc .eq. 1) read(2,*) nalpha,(alpha(k),k=1,nalpha)
      if (do_cog .eq. 1) read(2,*) nq,(q(k),k=1,nq)

      read(2,*) ns(1)
            ns(1)=ns(1)-1
        read(2,*) (x(1,i),i=0,ns(1))
        read(2,*) (u(1,i),i=0,ns(1))

      read(1,*) nfs
      nes=nfs+2
        do 10 kk=1,nfs
        k=kk+1
        read(1,*) ns(k)
        ns(k)=ns(k)-1
          read(1,*) (x(k,i),i=0,ns(k))
          read(1,*) (u(k,i),i=0,ns(k))
10      continue

        close(1)
      close(2)
```

```
      if (r .gt. 200) r=200
      if (r .lt. 10) r=10
      if (t .gt. 1.) t=1.
      if (t .lt. 1.E-5) t=1.E-5

      if (do_acc .ne. 0) then
c ****************************************************************
c    acceptability measure
c ****************************************************************

      do 25 k=1,nalpha
        do 30 j=1,nfs
          jj=j+1
          alt1(j,k)=j
          call poss(1,jj,nes,xc,ns,x,u,r,t,con,dil,cintf)
          rank(j,k)=alpha(k)*xc
          call nec(1,jj,nes,xc,ns,x,u,r,t,con,dil,cintf)
          rank(j,k)=rank(j,k)+(1.-alpha(k))*xc
30      continue
25    continue

c ****************************************************************
c    sort rankings (for the acceptability measure)
c ****************************************************************

      do 31 k=1,nalpha
        do 33 j=1,nfs
          a=rank(j,k)
          nalt=alt1(j,k)
          do 35 i=j-1,1,-1
            if (rank(i,k) .ge. a) goto 37
            rank(i+1,k)=rank(i,k)
            alt1(i+1,k)=alt1(i,k)
35        continue
          i=0
37        rank(i+1,k)=a
          alt1(i+1,k)=nalt
33      continue
31    continue

      write(3,*) 'Acc'
      write(3,85)
      write(3,86)
      do 39 k=1,nalpha
        do 38 j=1,nfs
          write(3,90) j, alt1(j,k), rank(j,k), alpha(k)
38      continue
39    continue
c     write(*,*) 'Summary table for Acc measure:'
c     write(*,95) (alpha(k),k=1,nalpha)
c     write(*,*) '________________________________________'
c     do 40 j=1,nfs
c40     write(*,96) j, (alt1(j,k),k=1,nalpha)
      end if

      if (do_cog .ne. 0) then
c ****************************************************************
c    centroid measure
c ****************************************************************

      write(3,*) 'WCoG'
      do 52 k=1,nq
      do 51 jj=1,nfs
        j=jj+1
```

```
        alt2(jj,k)=jj
        xbar(jj,k)=0.
        axsum=0.
        asum=0.
        do 50 i=1,ns(j)
        axsum=axsum+((u(j,i-1)+u(j,i))**q(k))*(x(j,i)**2-x(j,i-1)**2)
        asum=asum+((u(j,i-1)+u(j,i))**q(k))*(x(j,i)-x(j,i-1))
50      continue
        xbar(jj,k)=axsum/(2.*asum)
51    continue
52    continue

c *****************************************************************
c    sort rankings (for the centroid measure)
c *****************************************************************

      do 54 k=1,nq
        do 53 j=1,nfs
          a=xbar(j,k)
          nalt=alt2(j,k)
          do 55 i=j-1,1,-1
            if (xbar(i,k) .le. a) goto 57
            xbar(i+1,k)=xbar(i,k)
            alt2(i+1,k)=alt2(i,k)
55        continue
          i=0
57        xbar(i+1,k)=a
          alt2(i+1,k)=nalt
53      continue
54    continue

      write(3,87)
      write(3,86)
      do 59 k=1,nq
        do 58 j=1,nfs
58        write(3,90) j, alt2(j,k), xbar(j,k), q(k)
59    continue
c     write(*,*) 'Summary table for CoG measure:'
c     write(*,95) (q(k),k=1,nq)
c     write(*,*) '__________________________________________'
c     do 60 j=1,nfs
c60      write(*,96) j, (alt2(j,k),k=1,nq)
      end if

85    format(' Rank Alt Measure Alpha')
86    format('________________________')
87    format(' Rank Alt Measure  q ')
90    format(x,i2,2x,i2,2x,f7.4,f5.1)
94    format(x,i2,2x,f7.4)
95    format(' Rank ',10f5.1)
96    format(x,i2,2x,10i5)

999     stop
        end


c*************************************
c       Fuzzy possibility computations
c*************************************

      subroutine poss(na,nb,nc,xc,ns,x,u,res,tol,con,dil,cintf)

      real*8 l12,l13,l23,change
```

```
      real x(100,0:200),u(100,0:200),
     +a(0:200),ua(0:200),b(0:200),ub(0:200),c(0:200),uc(0:200),
     +tol,xc,test1,test2

      integer ns(100),res,na,nb,nc,con,dil,cintf

      npa=ns(na)
      npb=ns(nb)

      do 101 i=0,npa
            a(i)=x(na,i)
            ua(i)=u(na,i)
101   continue
      do 102 i=0,npb
            b(i)=x(nb,i)
            ub(i)=u(nb,i)
102   continue

      ab_min=min(a(0),b(0))
      if (ab_min .lt. 0.) ab_min=0.
      ab_max=max(a(npa),b(npb))
      ba=ab_max-ab_min

      c_min=ab_min
      c_max=ab_max
      cc=c_max-c_min

      step=cc/REAL(res)
      step_size=step
      step_min=step/10.

c************************************************
c       Cycle through the universe of discourse
c************************************************
      k=0
      npoints=0
      nend=0
      c(k)=c_min

c     write(*,*) ab_max,ab_min,cc,step_size
      do 200 while (c(k) .le. c_max)
c     write(*,*) c(k)
c**************************************************************************
**
c       sup-t composition (t-norm test at each x, sup summary of t-norm tests)
c**************************************************************************
**

        uc(k)=0.
        u1=0.
        u2=0.
        test1=c(k)
        test2=c(k)

        if ( (test2 .lt. b(0)).or.(test2 .gt. b(npb)).or.
     +(test1 .lt. a(0)).or.(test1 .gt. a(npa)) ) go to 325
        l=0
        do 310 while (b(l) .lt. test2)
310       l=l+1
        if (b(l) .eq. test2) then
            u2=ub(l)
        else
            u2=ub(l-1)+(ub(l)-ub(l-1))*(test2-b(l-1))/(b(l)-b(l-1))
        end if
```

```
      l=0
      do 320 while (a(l) .lt. test1)
320     l=l+1
      if (a(l) .eq. test1) then
          u1=ua(l)
      else
          u1=ua(l-1)+(ua(l)-ua(l-1))*(test1-a(l-1))/(a(l)-a(l-1))
      end if
      if (con .eq. 1) u1=u1*u1
      if (dil .eq. 1) u1=u1**.5
      if (cintf .eq. 1) then
        if (u1 .lt. .5) u1=2.*u1*u1
        if (u1 .ge. .5) u1=1.-2.*(1.-u1)**2.
      end if

325   uc(k)=min(u1,u2)

c**************************************************************************
***
c     Test for corners using 3 points, change in length as function for
search
c**************************************************************************
***
c 3456789012345678901234567890123456789012345678901234567890123456789012
     if (k .ge. 2 .and. npoints .eq. 2) then
       l12=((uc(k-1)-uc(k-2))**2.+((c(k-1)-c(k-2))/cc)**2.)**.5
       l23=((uc(k)-uc(k-1))**2.+((c(k)-c(k-1))/cc)**2.)**.5
       l13=((uc(k)-uc(k-2))**2.+((c(k)-c(k-2))/cc)**2.)**.5
       change=100.*(l12+l23-l13)/l13
       if (change .gt. tol .and. step_size .gt. step_min) then
         step_size=step_size/2.
         k=k-2
       else
         c(k-1)=c(k)
         uc(k-1)=uc(k)
         k=k-1
         step_size=step
c        step_size=step_size*2.
c        if (step_size .gt. step) step_size=step
       end if
       npoints=0
     end if

c**************************************************
c     Prepare for the next point in the added set
c**************************************************
     k=k+1
     npoints=npoints+1
     c(k)=c(k-1)+step_size
     if (c(k) .gt. c_max .and. nend .eq. 0) then
          c(k)=c_max
          nend=1
     end if

200  continue

     k=k-1
     ns(nc)=k
     do 103 i=0,ns(nc)
          x(nc,i)=c(i)
          u(nc,i)=uc(i)
103  continue
```

```
      xc=0.
      do 500 i=0,k
500     if (uc(i) .gt. xc) xc=uc(i)

      return
      end

c************************************
c       Fuzzy necessity computations
c************************************

      subroutine nec(na,nb,nc,xc,ns,x,u,res,tol,con,dil,cintf)

      real*8 l12,l13,l23,change

      real x(100,0:200),u(100,0:200),
     +a(0:200),ua(0:200),b(0:200),ub(0:200),c(0:200),uc(0:200),
     +tol,xc,test1,test2

      integer ns(100),res,na,nb,nc,con,dil,cintf

      npa=ns(na)
      npb=ns(nb)

      do 601 i=0,npa
            a(i)=x(na,i)
            ua(i)=u(na,i)
601   continue
      do 602 i=0,npb
            b(i)=x(nb,i)
            ub(i)=u(nb,i)
602   continue

      ab_min=min(a(0),b(0))
      if (ab_min .lt. 0.) ab_min=0.
      ab_max=max(a(npa),b(npb))
      ba=ab_max-ab_min

      c_min=ab_min
      c_max=ab_max
      cc=c_max-c_min

      step=cc/REAL(res)
      step_size=step
      step_min=step/10.

c**************************************************
c       Cycle through the universe of discourse
c**************************************************
      k=0
      npoints=0
      nend=0
      c(k)=c_min

      do 700 while (c(k) .le. c_max)

c*****************************************************************************
***
c       sup-t composition (t-norm test at each x, sup summary of t-norm tests)
c*****************************************************************************
***

        uc(k)=0.
        u2=0.
```

```
      ul=0.
      test1=c(k)
      test2=c(k)

      if ( (test2 .lt. b(0)).or.(test2 .gt. b(npb)).or.
     +(test1 .lt. a(0)).or.(test1 .gt. a(npa)) ) go to 825
      l=0
      do 810 while (b(l) .lt. test2)
810     l=l+1
      if (b(l) .eq. test2) then
          u2=ub(l)
      else
          u2=ub(l-1)+(ub(l)-ub(l-1))*(test2-b(l-1))/(b(l)-b(l-1))
      end if

      l=0
      do 820 while (a(l) .lt. test1)
820     l=l+1
      if (a(l) .eq. test1) then
          ul=ua(l)
      else
          ul=ua(l-1)+(ua(l)-ua(l-1))*(test1-a(l-1))/(a(l)-a(l-1))
      end if
      if (con .eq. 1) ul=ul*ul
      if (dil .eq. 1) ul=ul**.5
      if (cintf .eq. 1) then
        if (ul .lt. .5) ul=2.*ul*ul
        if (ul .ge. .5) ul=1.-2.*(1.-ul)**2.
      end if

825   uc(k)=max(ul,1.-u2)

c***************************************************************************
***
c     Test for corners using 3 points, change in length as function for
search
c***************************************************************************
***
     if (k .ge. 2 .and. npoints .eq. 2) then
      l12=((uc(k-1)-uc(k-2))**2.+((c(k-1)-c(k-2))/cc)**2.)**.5
      l23=((uc(k)-uc(k-1))**2.+((c(k)-c(k-1))/cc)**2.)**.5
      l13=((uc(k)-uc(k-2))**2.+((c(k)-c(k-2))/cc)**2.)**.5
      change=100.*(l12+l23-l13)/l13
      if (change .gt. tol .and. step_size .gt. step_min) then
        step_size=step_size/2.
        k=k-2
      else
        c(k-1)=c(k)
        uc(k-1)=uc(k)
        k=k-1
        step_size=step
c       step_size=step_size*2.
c       if (step_size .gt. step) step_size=step
      end if
      npoints=0
     end if

c**************************************************
c     Prepare for the next point in the added set
c**************************************************
      k=k+1
      npoints=npoints+1
      c(k)=c(k-1)+step_size
      if (c(k) .gt. c_max .and. nend .eq. 0) then
```

```
            c(k)=c_max
            nend=1
          end if

700   continue


      k=k-1
      ns(nc)=k
      do 603 i=0,ns(nc)
            x(nc,i)=c(i)
            u(nc,i)=uc(i)
603   continue

      xc=1.
      do 900 i=0,k
900     if (uc(i) .lt. xc) xc=uc(i)

      return
      end
```

E.4 compro.dat

```
20          ! resolution
1000        ! test sensitivity
.01         ! accuracy tolerance
0           ! concentration toggle (1,0)
0           ! dilation toggle (1,0)
0           ! contrast intensification (1,0)
1           ! definition for 1/p is provided (1,0)
6           ! m number of alternatives
8           ! n number of criteria
2           ! p
1. 2.       ! x values for p
0. 1.       ! u values for p
3           ! w[i]
.4 .5 .9
0. 1. 0.
3
.4 .5 .8
0. 1. 0.
3
.25 .35 .75
0. 1. 0.
3
.2 .35 .7
0. 1. 0.
3
.1 .25 .6
0. 1. 0.
4
.4 .6 .9 1.
0. 1. 1. 0.
3
0. .55 1.
0. 1. 0.
3
.3 .5 .6
0. 1. 0.
2           ! zp[i]
95. 100.
0. 1.
2
8. 10.
0. 1.
2
5. 6.
0. 1.
2
5. 6.
0. 1.
2
5. 6.
0. 1.
2
5. 6.
0. 1.
2
5. 6.
0. 1.
2
5. 6.
0. 1.
2           ! zn[i]
```

```
0. 1.
1. 0.
2
0. .5
1. 0.
2
0. 1.
1. 0.
2
0. 1.
1. 0.
2
0. 1.
1. 0.
2
0. 1.
1. 0.
2
0. 1.
1. 0.
2
0. 1.
1. 0.
3             ! z[i1]
92.3 92.4 92.5
0. 1. 0.
3
.69 .7 .71
0. 1. 0.
6
0. 1. 1.5 2. 2.5 3. 4.
0. 1. 0. 1. 0. 1. 0.
3
1.5 2. 3.5
0. 1. 0.
3
1.5 3. 3.5
0. 1. 0.
3
1.5 3. 3.5
0. 1. 0.
3
3.5 4. 5.5
0. 1. 0.
3
1.5 3. 3.5
0. 1. 0.
3             ! z[i2]
86.4 86.5 86.6
0. 1. 0.
3
2.39 2.4 2.41
0. 1. 0.
3
1.5 3. 3.5
0. 1. 0.
3
1.7 2. 2.3
0. 1. 0.
3
2.5 4. 4.5
0. 1. 0.
3
1.5 3. 3.5
```

```
0. 1. 0.
3
3.5 5. 5.5
0. 1. 0.
3
2.7 3. 3.3
0. 1. 0.
3             ! z[i3]
4. 4.1 4.2
0. 1. 0.
3
5.49 5.5 5.51
0. 1. 0.
3
4.7 5. 5.3
0. 1. 0.
3
3.7 4. 4.3
0. 1. 0.
3
3.5 4. 5.5
0. 1. 0.
3
3.5 5. 5.5
0. 1. 0.
3
2.7 3. 3.3
0. 1. 0.
3
3.5 4. 5.5
0. 1. 0.
3             ! z[i4]
20.9 21. 21.1
0. 1. 0.
3
6.29 6.3 6.31
0. 1. 0.
3
3.5 4. 5.5
0. 1. 0.
3
2.7 3. 3.3
0. 1. 0.
3
3.5 5. 5.5
0. 1. 0.
3
3.7 4. 4.3
0. 1. 0.
3
3.7 4. 4.3
0. 1. 0.
3
1.5 3. 3.5
0. 1. 0.
3             ! z[i5]
28.1 28.2 28.3
0. 1. 0.
3
5.99 6. 6.01
0. 1. 0.
3
3.5 4. 5.5
0. 1. 0.
```

```
3
2.5 4. 4.5
0. 1. 0.
3
3.5 5. 5.5
0. 1. 0.
3
3.5 4. 5.5
0. 1. 0.
3
3.7 4. 4.3
0. 1. 0.
3
1.5 3. 3.5
0. 1. 0.
3            ! z[i6]
6.8 6.9 7.
0. 1. 0.
3
5.29 5.3 5.31
0. 1. 0.
3
3.5 5. 5.5
0. 1. 0.
3
2.5 3. 4.5
0. 1. 0.
3
3.5 4. 5.5
0. 1. 0.
3
3.5 5. 5.5
0. 1. 0.
3
2.7 3. 3.3
0. 1. 0.
3
3.5 4. 5.5
0. 1. 0.
3            ! 1
.99 1. 1.01
0. 1. 0.
11           ! 1/p
.5 .57 .6 .63 .67 .7 .74 .8 .87 .94 1.
1. .76 .66 .57 .5 .42 .36 .24 .15 .07 0.
0
```

E.5 rank.dat

```
60                  ! resolution
.005                      ! accuracy tolerance
1                   ! toggle switch for Acc measure
1                   ! toggle switch for CoG measure
5 .1 .3 .5 .7 .9    ! alpha weight - optimism [0,1]
3 1. 2. 3.          ! q - membership value weighting exponent
2             ! # points for ideal number
0. 4.         ! x values for ideal
1. 0.         ! u values for ideal
```