A POLYNOMIAL NETWORK MODELING APPROACH TO A CLASS OF LARGE-SCALE HYDRAULIC SYSTEMS

By

Steven L. Kleinsteuber

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

MASTER OF SCIENCE

Department of Mechanical/Industrial Engineering
University of Manitoba
Winnipeg, Manitoba

ISBN 0-612-13258-7

Name _____

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.

Mechanical Engineering
SUBJECT TERM

0548
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

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION

A POLYNOMIAL NETWORK MODELING APPROACH TO A

CLASS OF LARGE-SCALE HYDRAULIC SYSTEMS

BY

STEVEN L. KLEINSTEUBER

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

MASTER OF SCIENCE

ABSTRACT

This thesis investigates the application of a polynomial abductive network modeling technique to a class of hydraulic actuation systems used in heavy-duty mobile machines. Previous techniques to model and simulate heavy-duty hydraulic functions include using a combination of analytical, numerical and steady-state solutions in a partitioned hierarchical manner [1] or, a transmission line modeling approach to facilitate parallel processing [2]. Using the results from the application of these methods to an excavator-based machine, databases are developed which are then used to construct polynomial abductive network models in order to replace the conventional iterative derivations for fluid flow distribution among the actuators. The networks are generated using a mechanism based on a hybrid learning logic called 'Abductory Induction' [3]. The performance of the polynomial abductive network model is compared to a back-propagation neural network model for a single link case to justify the use of the new technique. The previously established iterative model provides comparison to polynomial network models in a multi-link case. It is shown that the computation time is reduced considerably, yet the simulation results show a sufficient degree of predictive accuracy. Additionally, the new modeling scheme eliminates the need for explicit derivations of the main valve orifice areas, and therefore facilitates the use of experimental observations in developing simulation models.

ACKNOWLEDGEMENTS

There are several people I would like to thank for their guidance and assistance while I was researching and writing this thesis. First, I would like to thank my advisor, Dr. Nariman Sepehri for his patience in guiding this student through the practical as well as administrative pitfalls involved in graduate school. I would also like to give my heartfelt thanks to Ben Urbanietz P.Eng of Westland Steel Products Ltd. who was as flexible and understanding a supervisor anybody could want while I was attending school and working full time simultaneously. Other people I am indebted to for their timely assistance and support are the people at Abtech Corporation who graciously donated their software for my research and the ladies in the mechanical engineering department office who would often allow me to use their laser printer when something needed to be printed yesterday. Finally I would like to thank Dr. D. Strong and Dr. D. Blight, who have volunteered their time to be examine my thesis.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont'd)

LIST OF ILLUSTRATIONS

TABLES

FIGURES

LIST OF ILLUSTRATIONS (cont'd)

LIST OF ILLUSTRATIONS (cont'd)

CHAPTER 1
INTRODUCTION

Hydraulic power has been proven as one of the best tools in use by human beings to magnify their physical power in heavy-duty tasks, and also in hazardous or labour intensive environments such as forest, mine and construction activity sites. These environments utilize many different heavy-duty, hydraulically powered machines. The present control of these machines consists of two two-degree-of-freedom joysticks with a one-to-one mapping between the joystick motions and the links. The application of a new coordinated-mode control technique has recently been under investigation on a prototype machine. The focus is on controlling the endpoint trajectory instead of the individual links. This provides a natural motion interface to the operators. It has been shown that significant productivity increases would be achieved, especially for novice operators, using this new method of control [4].

One issue in this project was to build an accurate and comprehensive computer model of the machine, especially the hydraulic actuation system, with emphasis towards real-time simulations that could be implemented in a manner similar to a 'flight simulator'. The simulator is needed for human factors studies, training, testing and improving control actions before they are actually implemented on the real machine. Also, a fast simulator can be embedded into a fault/hazard diagnosis system which would perform such tasks as preventing the machine from tipping-over [5].

Accurate and computationally efficient simulation of the hydraulic actuation systems in heavy-duty mobile machines is very difficult. The analysis is impeded by the nonlinearity of the fluid flow equations and the physical and numerical interaction between the system components [1,6,7]. This interaction results in a mixture of fast and slow changing state variables which cause the system to be mathematically stiff [8,9,10]. Another problem in modeling hydraulic functions is the occurrence of discontinuities in the model caused by related functional relationships which change suddenly between differing regions of operation. Examples of such discontinuities are stick/slip friction, backlash, or joint limits [11].

The importance of modeling hydraulic systems has been reported in many applications such as automotive [12], marine [13], aerospace [14,15] and construction [16,17], and many modeling techniques have been proposed. The most common technique uses fluid pressure as a state space variable. The stiff equations are then replaced with steady-state equations, wherever possible, to reduce the computation time [1]. Alternatively, the steady-state equations can be combined with the hydraulic lines' equations to form a new set of state space differential equations [9]. Recently, Burton *et al.* [2] suggested a transmission line modeling approach for parallel simulation of hydraulic systems. All these

methods are, however, heavily dependent on a complete knowledge of the component parameters which may not be readily available or easily obtainable in practice.

In this thesis we revisit the problem of modeling and simulation of the hydraulic actuation systems used in heavy-duty machines. We consider a hybrid logic-based learning approach for constructing mathematical models with polynomial abductive networks. The objectives are to: First, reduce the computation time necessary for the simulation in order to achieve real-time performance, and second, to eliminate the explicit derivation of variables (from the simulation model) which are difficult to experimentally obtain from real world systems. An example of these variables is the valve orifice areas.

The physical system upon which all simulations in this thesis are based is the Caterpillar 215B excavator. The actual method for deriving polynomial abductive networks involves using a software package called the 'Abductory Induction Mechanism' (AIM) [18], which uses supervised learning to generate network models to be inserted into the simulation program.

The organization of this thesis is as follows. First, a typical hydraulic system embodied in a heavy-duty excavator is described. Then, an overview of the Abductory Induction Mechanism and Artificial Neural Networks is provided. In chapter 4 we address the modeling of hydraulic functions pertaining to the main valves and pumps' part for a single-actuation case and propose an alternative solution based on polynomial abductive network models. Neural networks are then applied to the problem and the results of the two methods are evaluated with a purpose of justifying the use of polynomial abductive networks. We then proceed to use polynomial abductive networks to model the more general case, that is a multi-actuation system with priority actions in chapter 5. The general results and specific simulation results are presented to validate the methodology of

model development presented in this thesis and computational comparisons are shown in chapter 6. Finally we examine the case where the pump flow is non constant in chapter 7 and the effect it has on the simulations.

CHAPTER 2
HYDRAULIC EXCAVATOR FUNCTION

Figure 2-1 shows the caterpillar 215B excavator. It is a mobile three-degree-of-freedom manipulator with a moveable end-effector namely, the bucket. The first three links are called swing, boom and stick and they are used for positioning the implement. The swing actuator is a hydraulic motor that transfers power through a gear train to rotate the upper structure. All the actuators are activated by pressure and flow through the main valves. Each main valve has its own orifice characteristic with various forms of lapping. Referring to Fig. 2-1(inset), the displacement in the valve spool activates three orifices at a time. Modulation of the oil flow in the main valves is controlled by the pilot oil pressure through the manually operated pilot control valves that are activated through joysticks 1 and 2.

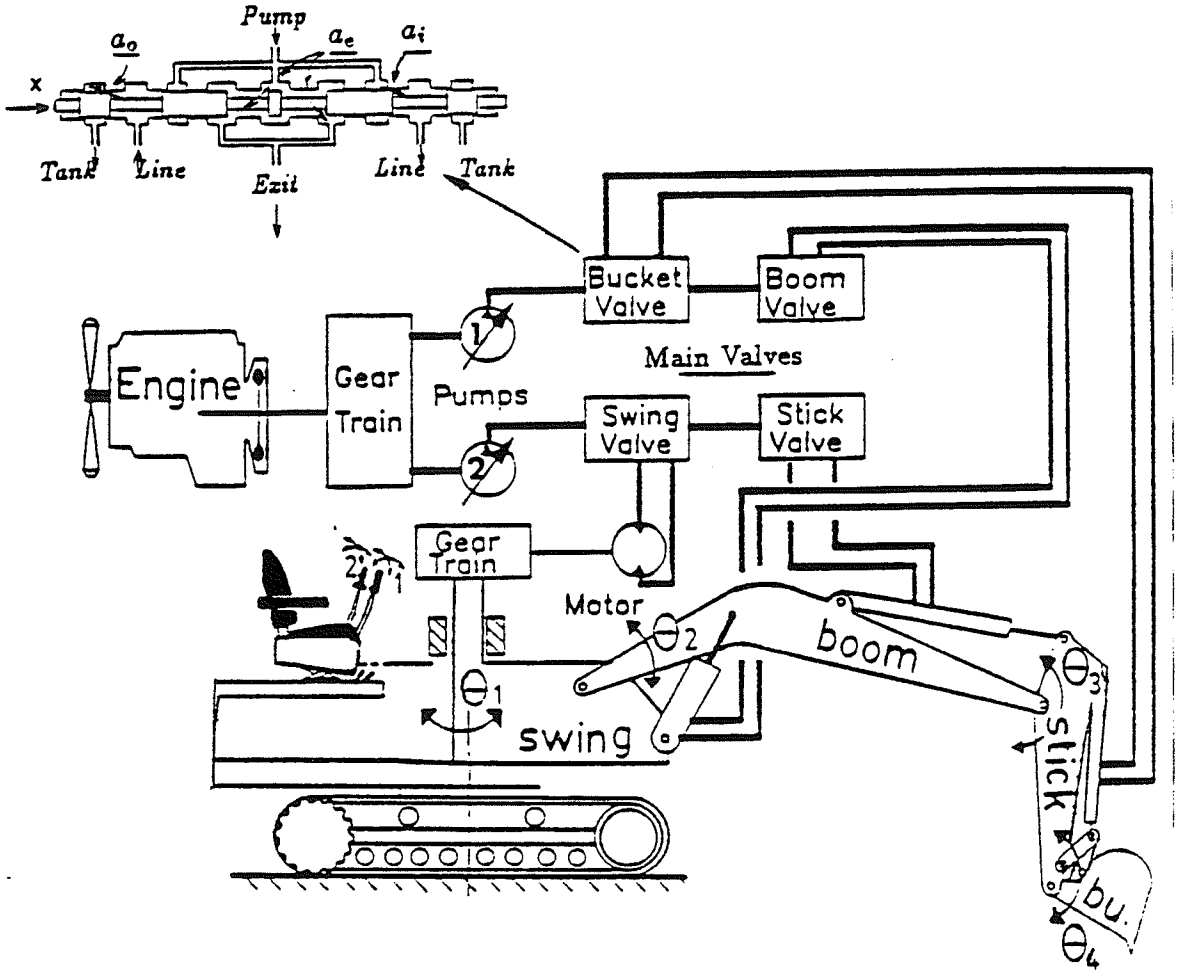


Fig. 2-1. The Caterpillar 215B Excavator; (inset) - Open Center Valve.

Referring to Fig. 2-1, the engine turns two pumps whose outputs are used to operate the actuators. The two pumps deliver oil to the system at a constant flow but with variable pressure. Hydraulic oil from pump 2 passes through both the swing and the stick main valves. Due to the priority action, the stick receives only the oil that passes through the swing valve. Therefore, stick movement is impossible if the swing main valve is fully open. If the former is partly open, the stick can operate but at a lower speed. If neither is open or only partially open then the excess oil bypasses the system and drains into the tank. In some machine models, the motion of the boom and the stick are also coupled via cross-over valves (not shown in the diagram).

The Caterpillar 215B excavator, shown in Fig. 2-1, is a large-scale system made up of several interdependent subsystems. The subsystems have their own particular functions which are hidden from other subsystems, but they share resources and are governed by an overall set of constraints. In previous work [1,10], the excavator was considered to be composed of two main subsystems; the structure and the actuation. The dynamics of the structure were determined using the Lagrangian approach. For the hydraulic part, the main valve system with its connections was considered as one subsystem activated by the pilot subsystem, and was connected to the linkages by means of flexible hoses. Fig. 2-2 shows the subsystems and their input/output arrangement.

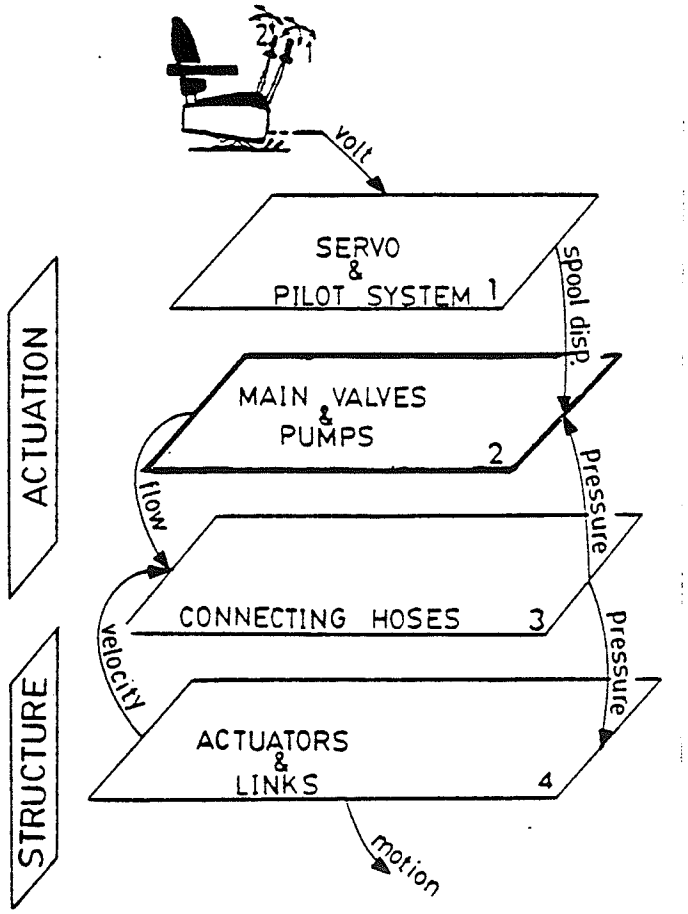


Fig. 2-2 Hydraulic Large Scale System Components and Relations

Referring to Fig. 2-2, the system operation is initiated by moving the joysticks which activate the pilot valves (subsystem 1). The pilot valves then displace the main valve

spools allowing fluid flow from the pump and main valves (subsystem 2), through the connecting hoses (subsystem 3) to the corresponding actuators (subsystem 4). Previous experimental studies showed a first-order relation in subsystem 1 between the input (the applied voltage to each pilot valve) and the output (corresponding main valve spool displacement); this allowed derivation of analytical solutions for this level. The equations describing the main valve dynamics (subsystem 2, Fig. 2-2) consisted of fast-response states. In order to achieve faster computation, the variables in subsystem 2 were redefined and the equations were rearranged to form a set of non-differential equations [1]. Iterative techniques were then used to solve the equations at this level. Equations describing the dynamics of the connecting hoses (subsystem 3) and the linkages (subsystem 4) consisted of slow-response states which determined the time interval for the numerical integration.

The accuracy of the simulation results based on the model described above has been verified and has been reported elsewhere [10]. In this work, the same structure is used with our newly developed network models replacing the main valve orifices and pumps dynamics in subsystem 2 (Fig. 2-2). These models should therefore be capable of being added to the existing simulation program without disturbing the description of other component models.

CHAPTER 3
THE MODELING TOOLS

3.1 Abductory Induction Mechanism Overview

'Abductory Induction' is a hybrid reasoning process based on the theory of abduction introduced by C.S. Peirce [19]. 'Abduction' is the logical process by which a hypothesis is obtained to explain an event or pattern under uncertainty. 'Induction' is reasoning from specific facts to general principles. The hybrid logic of abductory induction is the method of forming a hypothesis under uncertainty (for example, insufficient, noisy, contradictory or unreliable data) and then using the available facts to prove whether the hypothesis is true or false [3]. A logic based overview of abductory induction is included in Appendix A. In this study the software package 'Abductory Induction Mechanism' (AIM) [3,18] is used to create abductive networks of functional nodes. Each functional node is a polynomial and the layered network combines these nodes in a feedforward structure. The polynomial function types used in AIM are as follows:

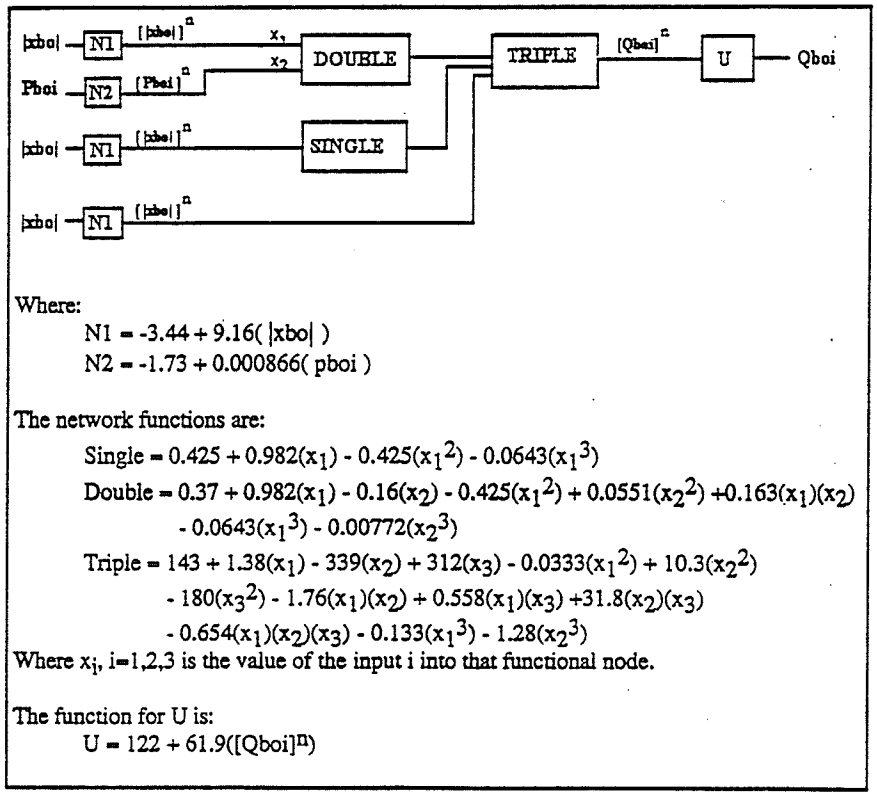
Single: $w_0 + (w_1 \times x_1) + (w_2 \times x_1^2) + (w_3 \times x_1^3)$

Double: $w_0 + (w_1 \times x_1) + (w_2 \times x_2) + (w_3 \times x_1^2) + (w_4 \times x_2^2)$
$+ (w_5 \times x_1 \times x_2) + (w_6 \times x_1^3) + (w_7 \times x_2^3)$

Triple: $w_0 + (w_1 \times x_1) + (w_2 \times x_2) + (w_3 \times x_3) + (w_4 \times x_1^2) +$
$(w_5 \times x_2^2) + (w_6 \times x_3^2) + (w_7 \times x_1 \times x_2) + (w_8 \times x_1 \times x_3) +$
$(w_9 \times x_2 \times x_3) + (w_{10} \times x_1 \times x_2 \times x_3) + (w_{11} \times x_1^3) + (w_{12} \times x_2^3) + (w_{13} \times x_3^3)$

Linear: $(w_1 \times x_1) + (w_2 \times x_2) + (w_3 \times x_3) + ...$

Where $w_0, w_1, w_2, ...$ are the coefficients to be determined by the program, and $x_1, x_2, x_3 ...$ are the input variables into the functional nodes. The above equations are named after the number of input variables into the function. Note that the double and triple functions also have cross terms that provide interaction among the inputs. The linear function consists of the linear weighted sums of all outputs of the previous layer of nodes. Fig. 3-1 shows an example of a network of polynomial functions created by AIM.



Where:

N1 = -3.44 + 9.16(|xbo|)

N2 = -1.73 + 0.000866(pboi)

The network functions are:

Single = $0.425 + 0.982(x_1) - 0.425(x_1^2) - 0.0643(x_1^3)$

Double = $0.37 + 0.982(x_1) - 0.16(x_2) - 0.425(x_1^2) + 0.0551(x_2^2) + 0.163(x_1)(x_2)$
$- 0.0643(x_1^3) - 0.00772(x_2^3)$

Triple = $143 + 1.38(x_1) - 339(x_2) + 312(x_3) - 0.0333(x_1^2) + 10.3(x_2^2)$
$- 180(x_3^2) - 1.76(x_1)(x_2) + 0.558(x_1)(x_3) + 31.8(x_2)(x_3)$
$- 0.654(x_1)(x_2)(x_3) - 0.133(x_1^3) - 1.28(x_2^3)$

Where x_i, i=1,2,3 is the value of the input i into that functional node.

The function for U is:

U = $122 + 61.9([Qboi]^n)$

Fig. 3-1 AIM Model For Boom Actuator Input Flow

This network is actually the final model of the hydraulic oil output flow, Q_{boi}, from the boom main valve into the boom actuator (see Fig. 2-1). The inputs are the absolute value of the main valve spool displacement, $|x_{bo}|$, and the input line pressure, P_{boi}. The relationships among these variables will be discussed in the next chapter, here we only describe how the network calculates the input flow, Q_{boi}, given the inputs $|x_{bo}|$ and P_{boi}.

With reference to Fig. 3-1, to create an AIM network we require a database of the inputs ($|x_{bo}|$ and P_{boi}) and the corresponding output (Q_{boi}). The inputs are first normalized, statistically, into a region with a mean of zero and a variance of one. The normalizing functions of the first layer of nodes, N1 and N2, are used to transform the data using mean sigma normalization [18]. In this example, the mean for $|x_{bo}|$ is $0.376in$ with a variance of 0.10922, the mean for P_{boi} is $2002.0psi$ with a corresponding variance of 1154.43. These transforms create the network variables $[|x_{bo}|]^n$ and $[P_{boi}]^n$. These variables are then fed into the second layer of nodes, which consists of a single and a double polynomial function. Note that inputs to the double node are, $x_1=[|x_{bo}|]^n$ and $x_2=[P_{boi}]^n$. In the triple node of the third layer, x_1 is the output of the double node, x_2 is the output of the single node and x_3 is $[|x_{bo}|]^n$. The final step is for the unitizer function of the output layer to transform the value of $[Q_{boi}]^n$ to a real value of Q_{boi}. The unitizer U performs the reverse of normalization in that it converts the network output which has a mean of zero and a variance of one to a range with a mean of $122.301in^3/sec$ and a variance of 61.90. These values are obtained from the training data statistics for Q_{boi}.

The use of a randomly generated database works best, since it prevents the network from overfitting the batches of 12 training patterns which are fed into the network sequentially to train the network. Using sequential training patterns results in the learning of a specific input sequence instead of the general mapping over all patterns. The normalization and

unitization of data as well as the whole network structure and composition are determined by the AIM program. The user usually decides the complexity and fit of the network through parameters that specify the maximum number of layers and the value of a Complexity Penalty Multiplier (CPM). The CPM is used to determine a Complexity Penalty, KP, as shown below.

$$KP = CPM \times (2K/N)s_p^2$$

Where the values for K, N and s_p^2 are determined from the training database. K is the total number of coefficients in the current network, N is the number of examples in the training database, and s_p^2 is an a-priori estimate of the true unknown model error variance. As N increases or s_p^2 decreases then the value for the complexity penalty goes down and more complex networks are allowed. However as the networks become more complex, K will increase which will raise the value for KP. The criterion used by AIM to evaluate the performance of an abductive network is the Predicted Squared Error, PSE, which is a measure of the forecasted error for data outside the range of the training database. PSE is determined as follows [18]:

$$PSE = FSE + KP$$

where FSE is the fitting squared error for data in the training base. The use of the predicted squared error can be considered as the implementation of induction in the program. When training, the program begins with a simple network, using only normalized inputs to determine an initial PSE. The initial FSE will be large and KP will be zero. As the model synthesis proceeds, the value for FSE will eventually become smaller and the value of KP larger as more complex networks are introduced. This results in the PSE following a parabolic curve as shown in Fig. 3-2. As networks are hypothesized,

they are compared to the previous best performing network (i.e., the network with the lowest PSE so far). If the hypothesized network has a lower PSE than the current network, the hypothesized network will be taken as the 'best' network. The program will continue generating networks until a better PSE is unobtainable, which is determined by networks steadily increasing in PSE. This indicates that the parabolic function for PSE has reached its global minimum.

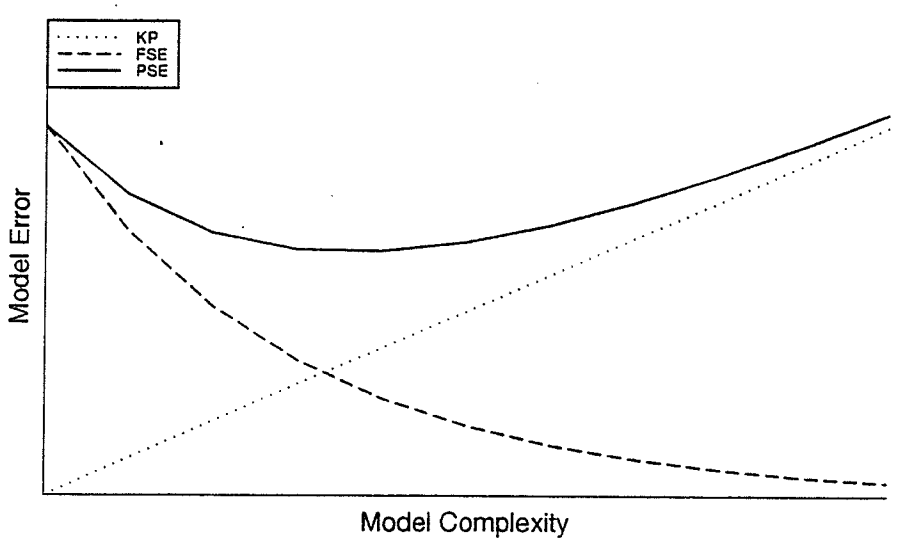


Fig. 3-2 Predicted Squared Error (PSE) Curve

In the example shown in Fig. 3-2, KP has a value of 25.0, FSE has a value of 2.167 and PSE has a value of 27.167. This value for PSE indicates that for data outside the range of the training data, one may expect absolute errors of approximately $\sqrt{27.167} = 5.2 in^3/sec$ or 3.03% of the maximum flow output for Q_{boi}, $171.5 in^3/sec$. Once the network model is finalized, the program generates a C source code subroutine to be embedded into the simulation model.

3.2 BACK PROPAGATION NEURAL NETWORK OVERVIEW

A neural network is a computing architecture that is copied from the structure of the central nervous system. Its operates on the same principle of using simple computational functions, or nodes, that are highly interconnected and make use of parallel processing in order to perform complex tasks[20].

The use of neural networks has grown rapidly in the past decade with the three layer back propagation network being the most predominant. A typical structure for a three layer back propagation network is shown in Fig. 3-3 and a typical neuron structure is shown in Fig. 3-4

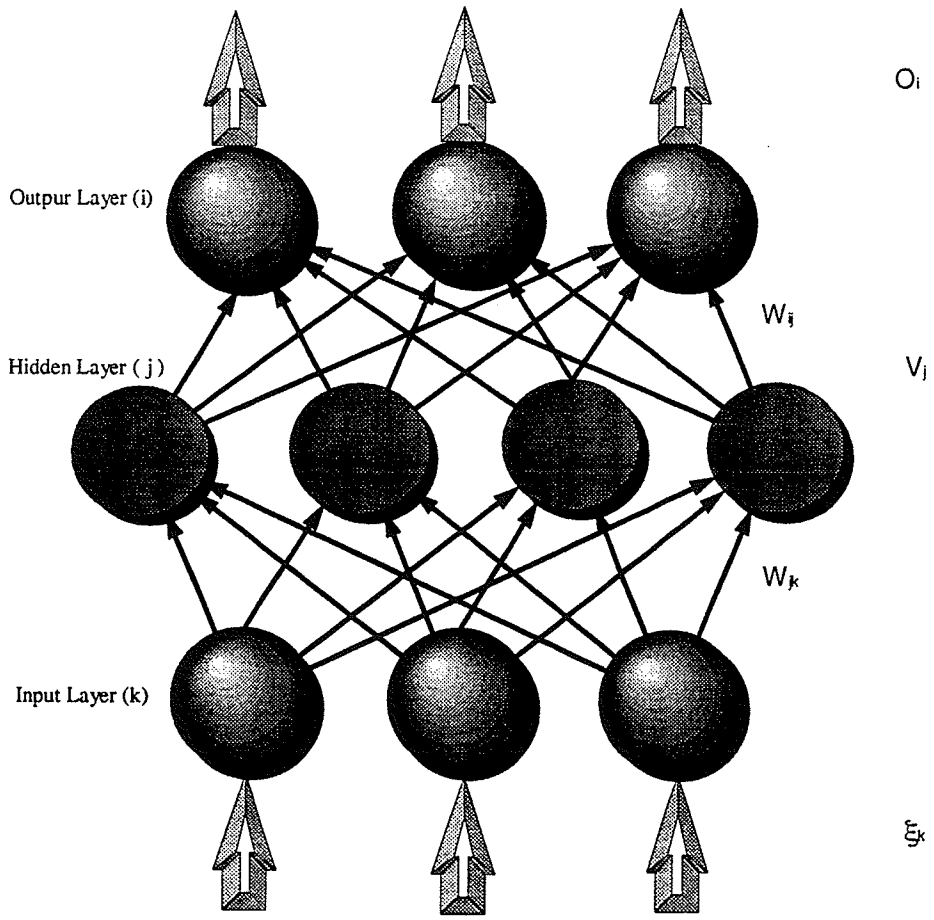


Fig. 3-3 3 Layer Back Propagation Neural Network

A three layer back propagation network is used for supervised learning where all inputs and the desired outputs are known. To train the network the specific architecture (i.e., the number of hidden units and the connections) must be specified by the user. The weights are then randomly set at some initial value for all the connections and the normalized input patterns (normalization using a uniform distribution between 0 and 1) are run through the network sequentially until all patterns have been used. The patterns are read sequentially instead of randomly, as in the AIM training process, because the neural network can process the complete training database before it updates its weights.

The training patterns are processed as follows in a three layer network with the current training pattern being μ , where μ =1, 2, ...,p. First, the hidden unit ,j, will receive from the inputs ,k:

$$h_j^{\mu} = \sum_k W_{jk} \xi_k^{\mu}$$

Where W_{jk} is the weight of the connection from unit k to unit j. The weighted sum for the current input pattern, h_j^{μ}, is used by the sigmoid activation function, $g(h_j^{\mu})$, of hidden unit j to determine the output for that neuron, V_j^{μ}. This process is shown in of Fig. 3-4 and the sigmoid activation function is defined as:

$$g(h_j^{\mu}) = \frac{1}{1+e^{-2\beta h_j^{\mu}}}$$

where β is the steepness parameter used to determine the slope of the sigmoid function.

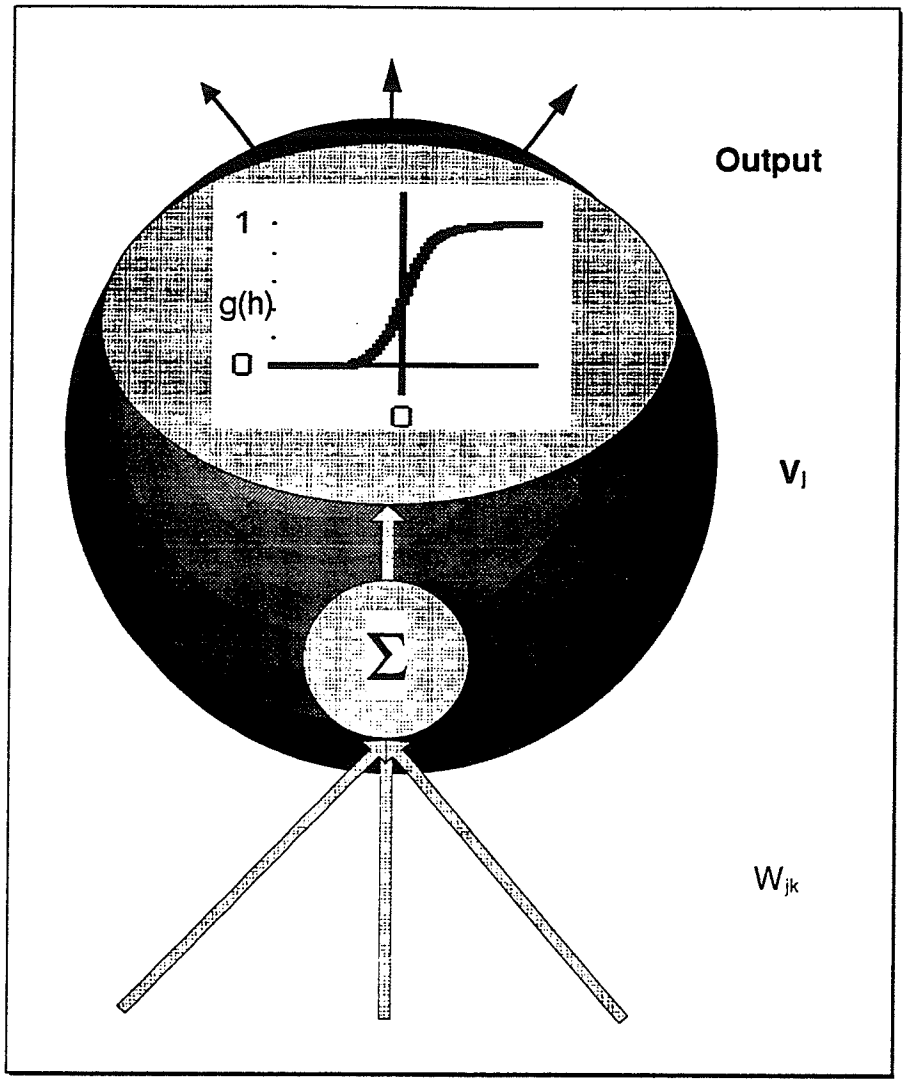


Fig. 3-4. A Schematic Neuron

The hidden neuron is activated and hidden unit j will produce as an output:

$$V_j^{\mu} = g(h_j^{\mu}) = g(\sum_k W_{jk}\xi_k^{\mu})$$

Therefore, a neuron in the output layer i will receive

$$h_i^{\mu} = \sum_j W_{ij} V_j^{\mu} = \sum_j W_{ij} g(\sum_k W_{jk}\xi_k^{\mu})$$

Again the input is processed through a sigmoid function, so the final output of the network will be:

$$O_i^{\mu} = g(h_i^{\mu}) = g(\sum_j W_{ij} V_j^{\mu})$$

Once the network has processed the patterns, the weights must be updated so that its performance will improve, when both input and desired output patterns are known this is referred to as supervised learning. The methodology is to calculate the error between the desired output and the network output and then propagate the error back from the output layer to the input layer and then adjusting the values for the weights. For this to work, the activation functions in the nodes must be differentiable which is the main reason that a continuous monotonic differentiable function like sigmoid is used.

One of the simplest methods of learning is by the gradient descent method [20]. Using gradient descent, the error or cost function for the network is defined as:

$$E[\mathbf{w}] = \frac{1}{2}\sum_{\mu i}[\zeta_i^{\mu} - O_i^{\mu}]^2$$

which is a continuous differentiable function for every weight in the weight space ,$\mathbf{w}=\{W_{ik}\}$. Note that ζ_i^{μ} is the output value for pattern μ from the training database and O_i^{μ} is the output value found by the network.

For the hidden to output connections the gradient descent rule gives a weight change in the connections of:

$$\Delta W_{ij} = -\eta \frac{\partial E}{\partial W_{ij}} = \eta \sum_{\mu}[\zeta_i^{\mu} - O_i^{\mu}]g'(h_i^{\mu})V_j^{\mu}$$

For the input to hidden connections it is necessary to differentiate with respect to the W_{jk}'s so the chain rule is needed. Therefore the weight change in the connections from the input units to the hidden units is:

$$\Delta W_{jk} = -\eta \frac{\partial E}{\partial W_{jk}} = \eta \sum_{\mu} \delta_j^{\mu} \xi_k^{\mu}$$

where the propagated error δ_j^{μ} is defined as:

$$\delta_j^{\mu} = g'(h_j^{\mu}) \sum_i W_{ij} (g'(h_i^{\mu})[\zeta_i^{\mu} - O_i^{\mu}])$$

Once the weights are updated, the training patterns are again processed through the network and the weight values are again changed. This process continues until the error function is less than a predetermined tolerable value or until there is no improvement through successive training iterations.

This chapter introduced the two techniques that will be used in this thesis to model the hydraulic actuation subsystem of an excavating machine. The primary differences between them are that AIM networks use functional nodes consisting of high order polynomials and no weight cost's on the node connections, while the neural network consists of nodes of differentiable functions with a weight cost attached to all connections. When training, the AIM network minimizes the model error by performing a statistically based trade off between model complexity and predicted error. The neural network uses error differentiation propagated through the network to adjust it's weights and no change is made to the unit functions or the structure of the network.

CHAPTER 4
DERIVING THE METHODOLOGY

4.1 GENERATING THE TRAINING DATABASE

To justify using AIM to model the complete excavator's hydraulic pumps and valves, we first attempt to use it to generate a mapping of the valve flow distribution for a simpler single-link actuation system (representing the boom link) shown in Fig. 4-1. The flow distribution and pressures across the lines are all shown on the diagram with the pump flow, Q, being distributed between the actuator and the tank. The displacement in the valve spool, x, activates three orifices, a_i, a_e and a_o, to control the flow distribution. The variation of the orifice areas versus spool displacement is shown in the inset of Fig. 4-1. The valve orifice described by a_i connects to the input line of the actuator, a_o to the output line, and a_e leads to the tank. The flow is controlled by changing these orifice areas.

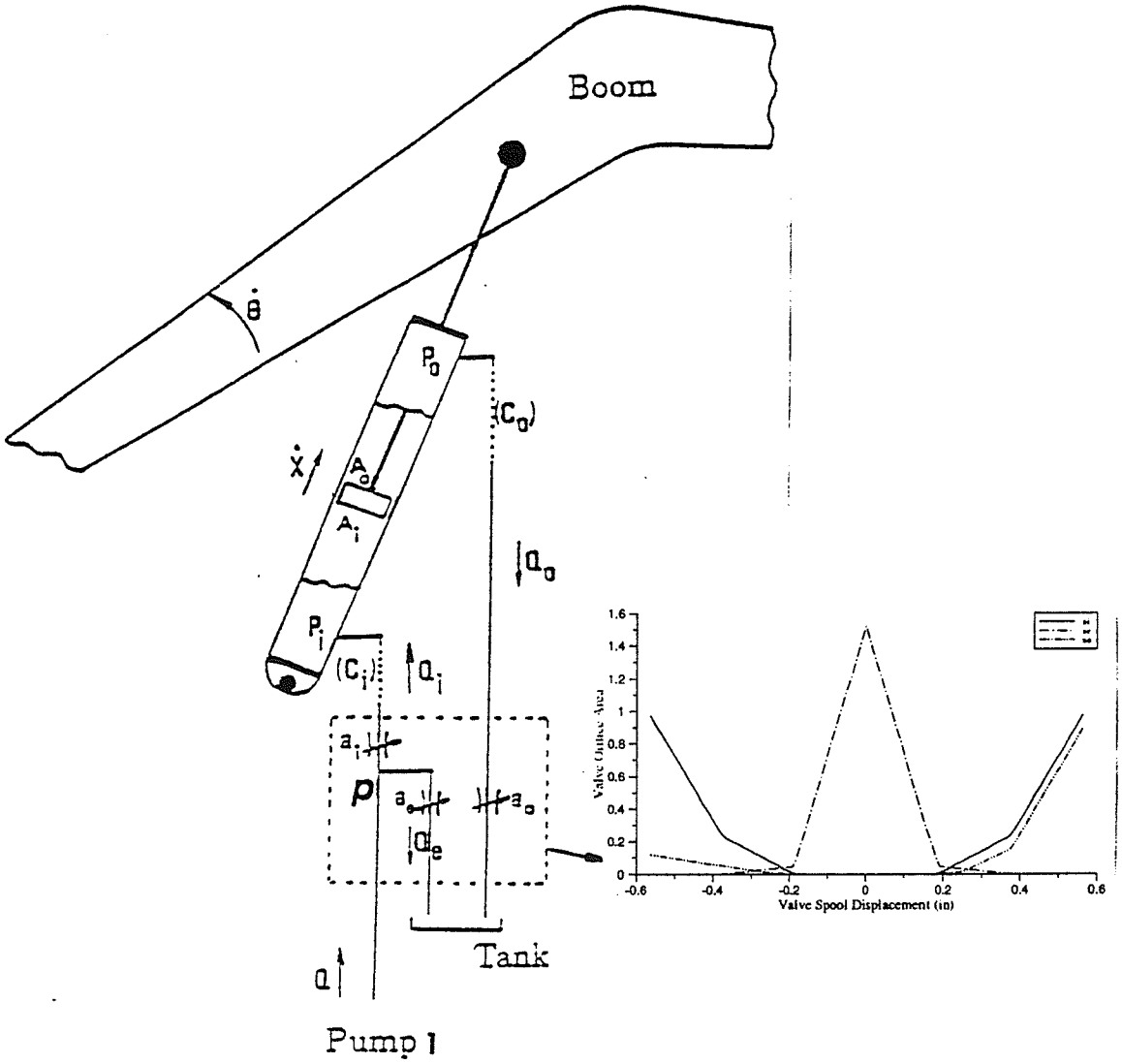


Fig. 4-1. Single Link (Boom) Hydraulic Schematic (inset): Valve Orifice Areas

When the spool is in a neutral position, orifices a_i and a_o are closed and the pump flow, Q, returns to tank through the open orifice a_e. As the spool is displaced, for example to the left, orifices a_i and a_o open and the orifice a_e closes. This results in a rise in the pump pressure P, due to a_e restricting the constant flow from the pump, until it exceeds the cylinder pressure, P_i , allowing fluid flow into the actuator. Note that a_i and a_o start to open after the valve spool moves approximately 0.188*in* (valve deadband); also a_e completely closes once the valve spool travels beyond 0.376*in*.

The general equations describing the hydraulic dynamics are written as follows [9]:

To the actuator:

$$Q_i = Ka_i\sqrt{P - P_i} \tag{1}$$

$$Q_e = Ka_e\sqrt{P - P_e} \tag{2}$$

$$Q - Q_i - Q_e = 0 \tag{3}$$

$$\frac{dP_i}{dt} = \frac{1}{C_i}(Q_i - Q_i') \tag{4}$$

Out of the actuator:

$$Q_o = Ka_o\sqrt{P_o - P_e} \tag{5}$$

$$\frac{dP_o}{dt} = \frac{1}{C_o}(Q_o' - Q_o) \tag{6}$$

where K is the discharge coefficient, $Q_i' = A_i\dot{X}$ and $Q_o' = A_o\dot{X}$ are the flow velocities into and out of the actuator, respectively. C_i and C_o are the values of compliance of the hoses connecting the valve to the actuator at the input and output ports. Note that Eq. (3) was initially written as

$$\frac{dP}{dt} = \frac{1}{C}(Q - Q_i - Q_e) \tag{3'}$$

where the compliance at the pump pipe, C, was very small compared with C_i and C_o. Therefore Eq. (3') was a stiff equation and determined the integration time interval for a

standard simulation. The above equation was then replaced with Eq. (3) by rewriting it in the following form:

$$C\frac{dP}{dt} \approx 0 = (Q - Q_i - Q_e) \qquad (3'')$$

The smallest time constant has, this way, been eliminated which allows a larger integration step size. Also, the number of states to be integrated has been reduced. However, the use of Eq. (3) instead of (3'), requires iterations during a state space simulation. Given the values of line pressure, P_i ,and the orifice areas (determined from the spool displacement, *x*), the pump pressure, P, as well as the flow into the cylinder, Q_i, are iteratively determined from Eqs. (1) to (3).

4.2 AIM APPLIED TO HYDRAULIC ACTUATION SIMULATION

4.2.1 Model Generation

In the initial approach taken to simulate a hydraulic actuation system it was intended to reduce the computation time by replacing the iterative derivation of the pump pressure ,P, (Eq. (1) to (3)) with AIM generated maps that covered the entire range of the pump pressure. Using pump pressures inherent symmetry about the valve spool displacement axis and modularizing the map based on the points of inflection of the valve orifice led to using the following three models to map the entire pressure range. One model was used for each of the following valve spool displacement ranges: $0 \geq |x| \geq 0.188 in$, $0.188 > |x| \geq 0.376 in$, and $0.376 > |x| \geq 0.564 in$. This approach was unsuccessful due to the complexity of the pressure map, shown in Fig. 4-2 (a) which did not allow for a close fitting AIM model, demonstrated in Fig. 4-2 (b). Fig. 4-3 shows the AIM error in calculating P over the complete operating range. Unfortunately this result is unacceptable since error in the pressure calculation is magnified in the flow and consequently the line pressure calculations and quickly destabilizes the complete hydraulic system due to the interaction among the components.

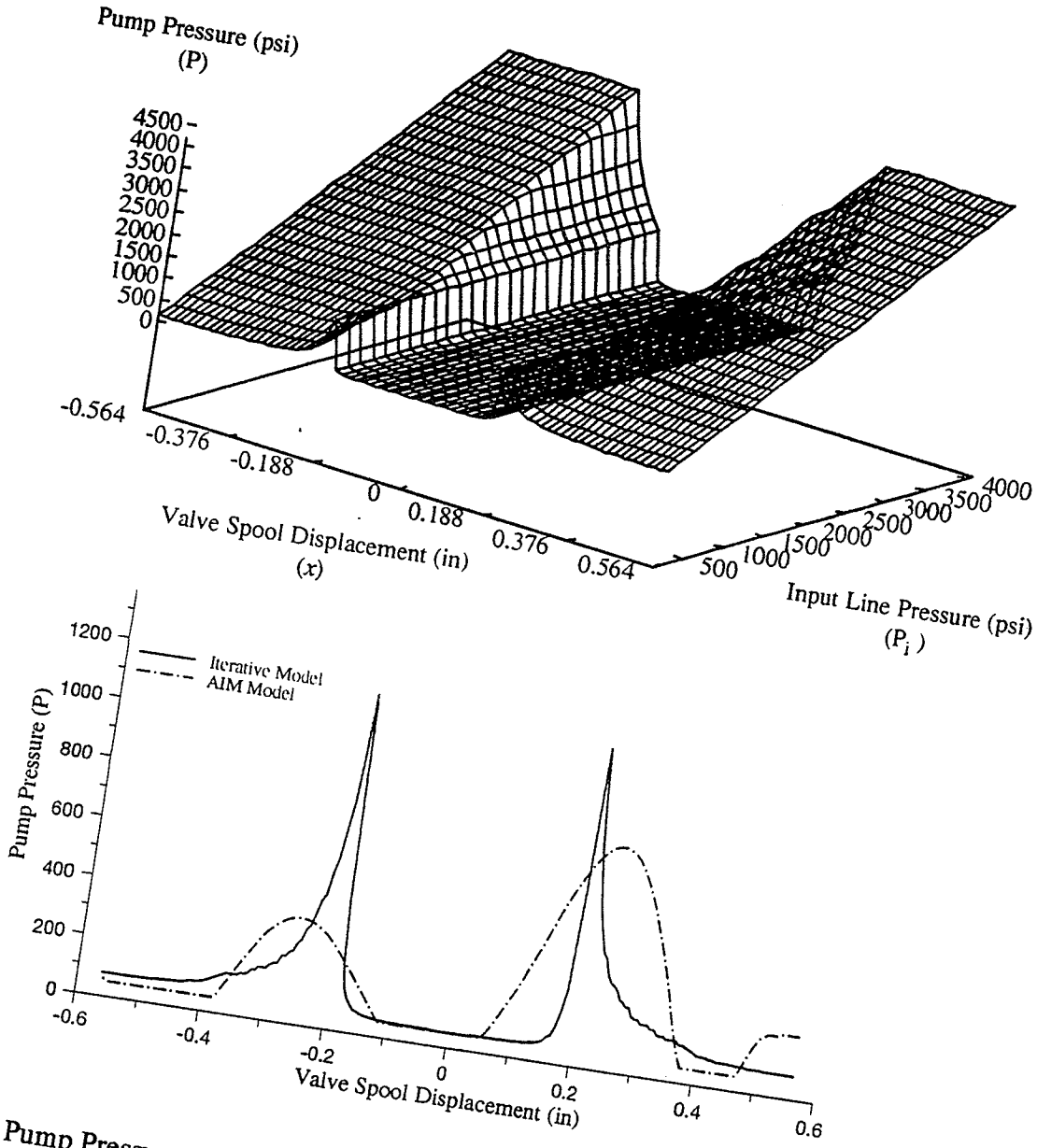


Fig. 4-2 Pump Pressure Map for (a) The Complete Operating Range; (b) Iterative versus AIM Pump Pressure Model with $P_i = 100psi$.

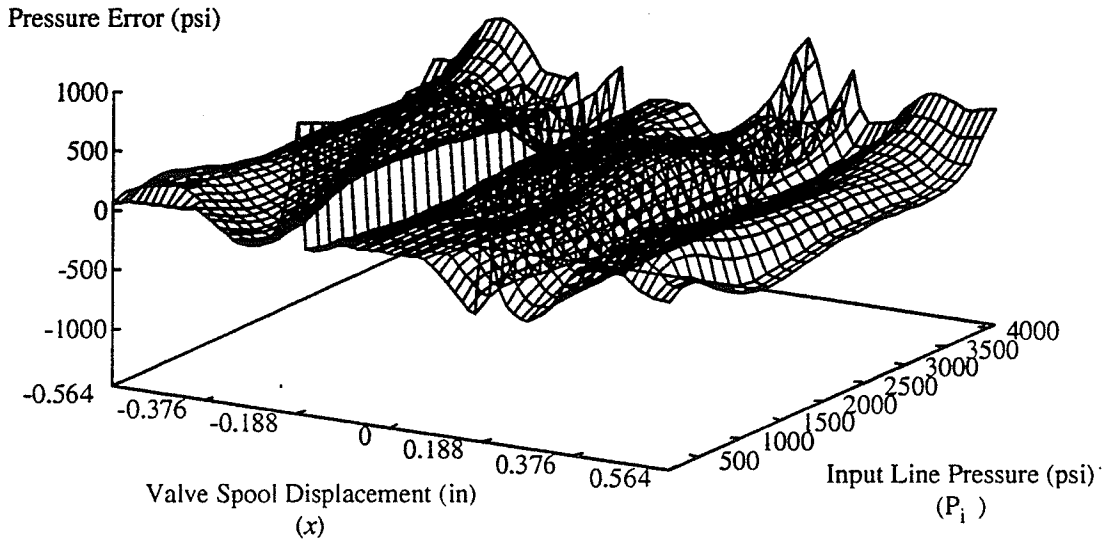


Fig 4-3. Pump Pressure Error Map.

In Fig. 4-4 we show the amount of error in a simulation where the pump pressure was modeled using AIM. A simple sinusoidal input voltage of v = 0.8 sin (1.2566t) was used as input to move the main valve spool shown in Fig. 4-1.

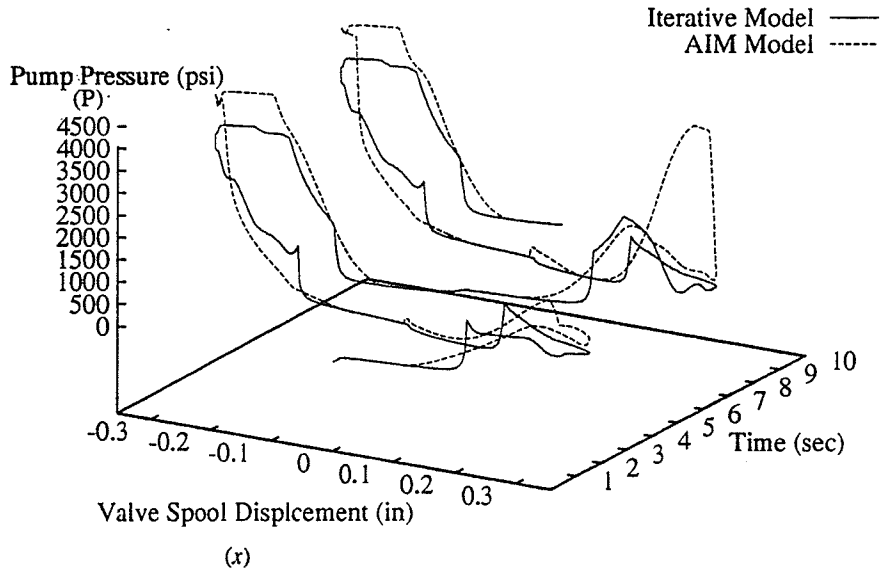


Fig 4-4. Simulation Pressure Error.

Figs. 4-5 and 4-6 show the effect of error in the pump pressure on the actuator flows and the line pressures.

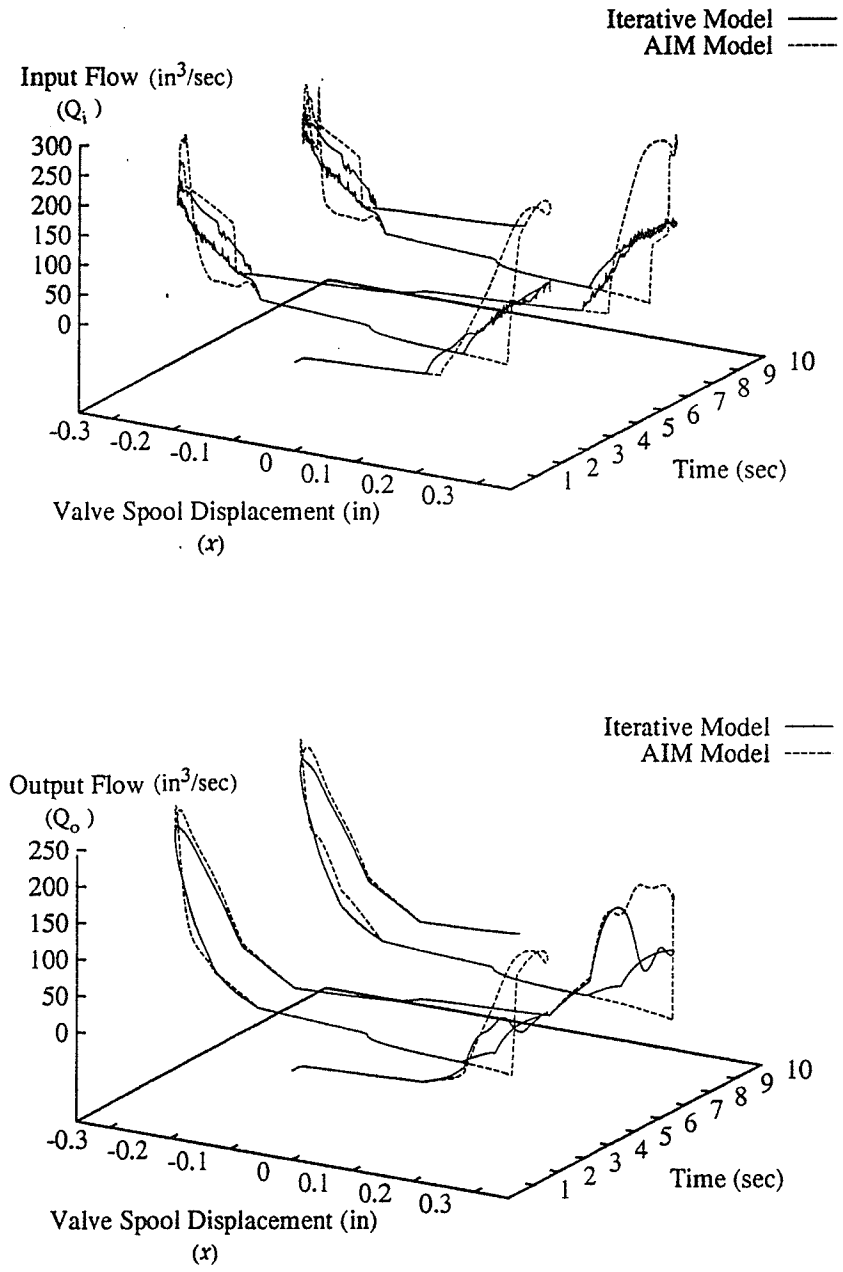


Fig. 4-5 AIM versus Iterative Model for: (a) Input Flow; (b) Output Flow.

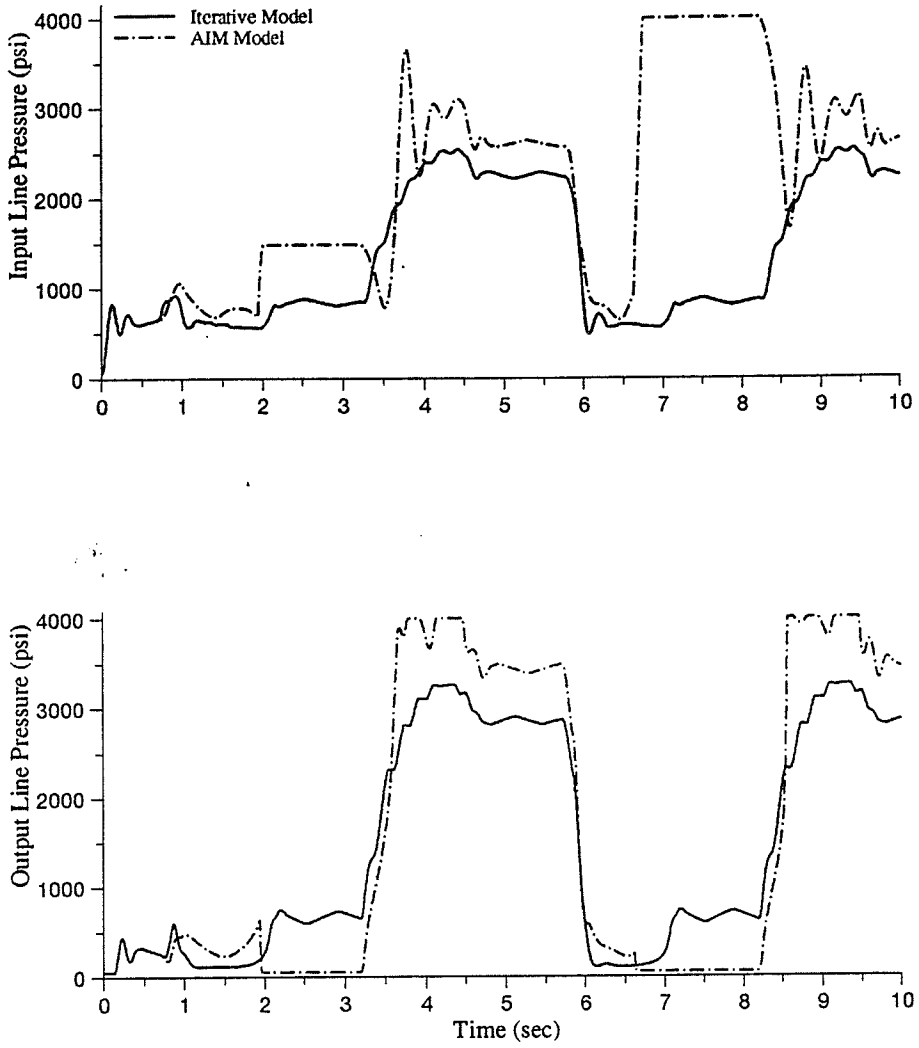


Fig. 4-6 AIM versus Iterative Model for: (a) Input Line Pressure;

(b) Output Line Pressure.

Referring to Fig. 2-2 in Chapter 2 and the equations in section 4.1, it is evident how the pressure error propagates in the following flow and line pressure derivations. First the pump pressure is used to calculate the flows into the actuators using Eqs. (1) and (2). These flows are transmitted from subsystem 2 in our large scale architecture to subsystem 3, where the line pressures are then calculated by integrating Eqs. (4) and (6). The line pressures are then transmitted to subsystem 4 where the error is further propagated in the

physical actuators and links, and in addition, the line pressures are also transmitted back into subsystem 2 where they are used in Eqs. (1) and (5) to calculate both the input and output flows. In Fig. 4-7 the effects of the pump pressure error on the outputs of subsystem 4, the joint displacement and joint velocity of the boom, are shown.

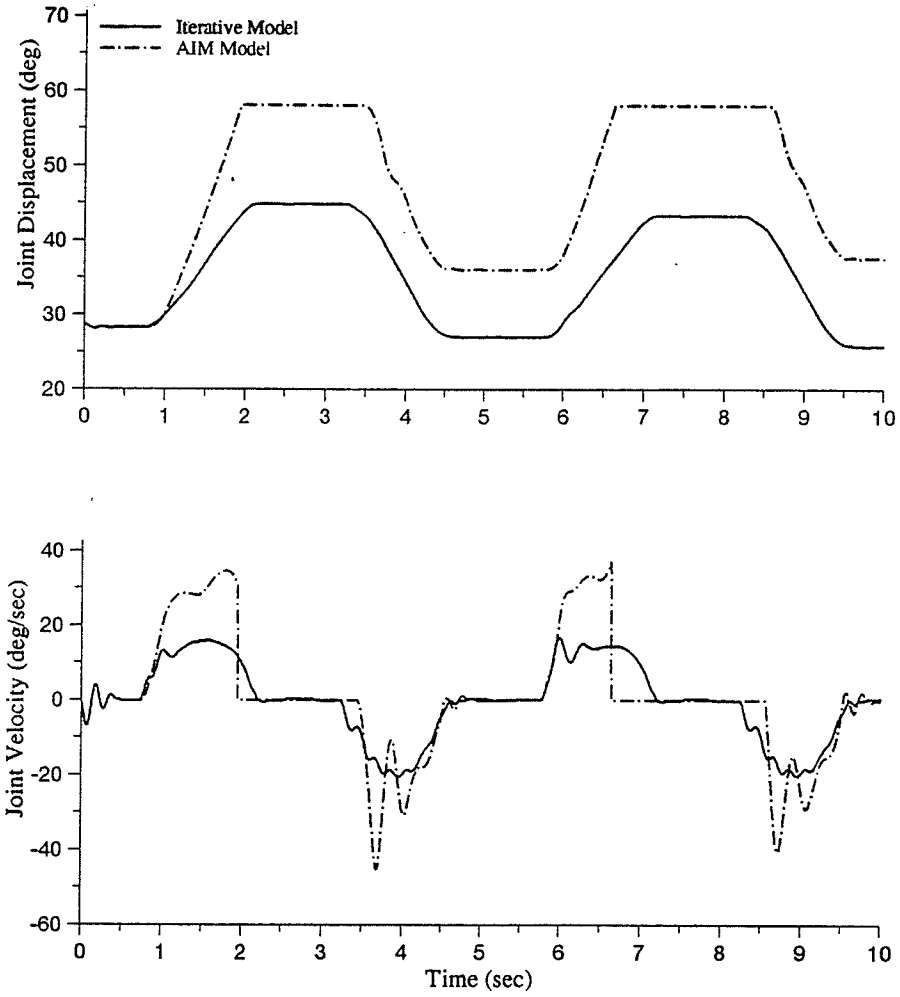


Fig. 4-7. AIM versus Iterative Model for: (a) Joint Displacement; (b) Joint Velocity.

Due to AIM's inability to model the pump pressure to the necessary fit and also due to the error propagation inherent in this system, the next approach bypassed the pump pressure and instead modeled the entire range of the flow into the actuator, Q_i. The advantage to

modeling Q_i is that the map for this variable is relatively straightforward when compared to the map for the pump pressure (i.e. the flow does not 'spike' when the valve spool overcomes its deadband) and although the value for Q_i is dependant upon the input line pressure, the effect of erroneous P_i will not be of the same magnitude as in the pump pressure model due to the removal of this level of computation, which reduces the propagation of error. A complete map for the input flow is shown in Fig. 4-8.

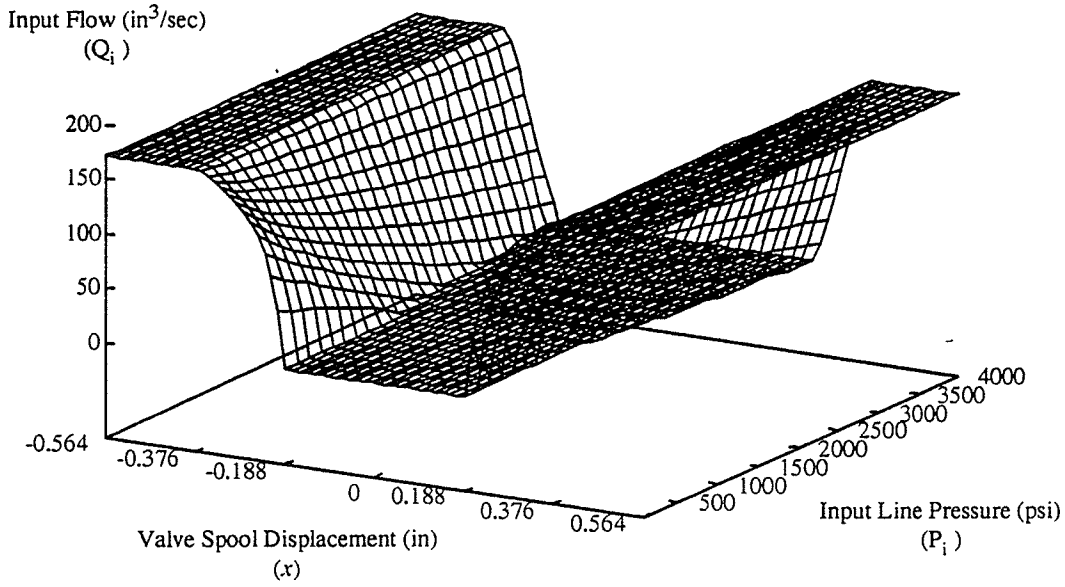


Fig. 4-8 Overall Input Flow Map.

To model the incoming flow it was decided to use an approach similiar to that used for the pump pressure map modeling. Only the positive values of the valve spool displacement were used to train the AIM models since the map is symmetric. In addition, a modular approach to modeling the map was used so that the accuracy of the models was as high as possible. The decision to overfit the curves was desirable since the complete operating domain is known and we are seeking to show that AIM can model these systems without introducing large errors into the joint displacement and velocity.

In the final approach to this problem, the computation time was reduced as much as possible by eliminating the derivation of both the pump pressure and the valve orifice areas. This also presented the additional benefit of eliminating variables in our hydraulic system which cannot be directly measured in a real world system or are difficult to derive. Therefore, the final approach was to eliminate the AIM networks which derived the orifice values a_o , a_e, and a_i and modify the AIM input flow models to eliminate these input variables and thereby attempt a direct mapping of valve spool displacement (x) and input line pressure (P_i) to flow, (Q_i). It was also necessary to replace the derivation of the output flow Q_o with AIM models since it relies on the valve constant a_o. as shown by Eq. (5). Fig. 4-9 shows the mapping of the output flow over its complete range.

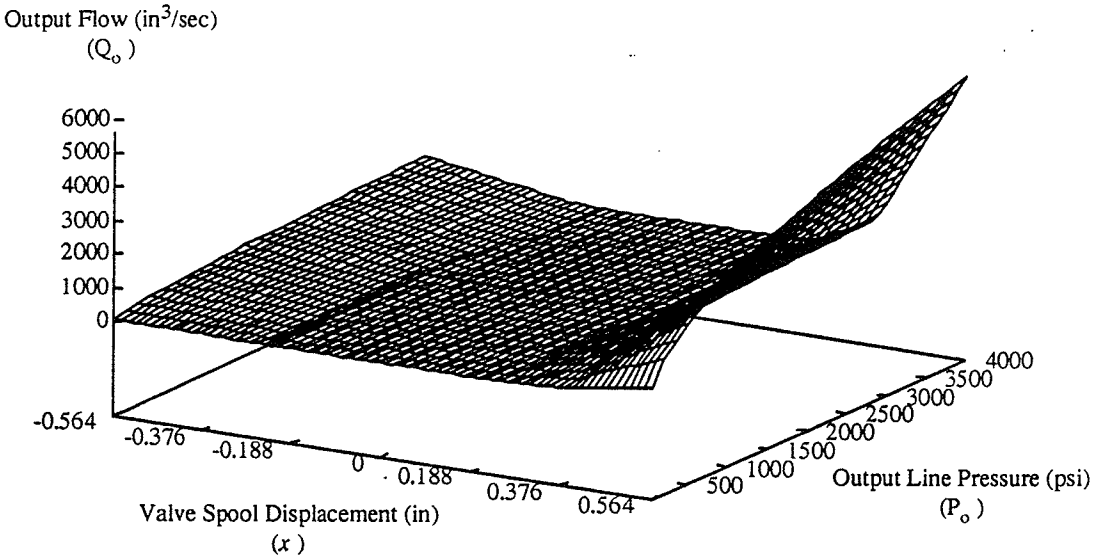


Fig. 4-9 Overall Output Flow Map.

The methodology of modeling Q_o was similar to Q_i in that the 3-D map was broken up at points where the slope of the output flow versus valve spool displacement curve changed. However, since the map for Q_o is not symmetric with respect to positive and negative

valve displacement as shown in Fig. 4-9 models were generated for the complete range of valve spool displacement.

Initially, the values that were used as inputs into the models were the valve spool displacement (x), since it is proportional to a_o,and the output line pressure (P_o). These models were able to map the output flow but the amount of error in the models was unacceptable due to the rapid propagation of errors. Since P_o and x are the only variables which can be used as inputs to the models, see Eq. 5, the training database could not be expanded by adding another variable. Therefore, instead of forcing AIM to learn the $\sqrt{P_o - P_e}$ relationship from P_o, the approach was taken to pre-process the data before network training to simplify the relationships that AIM would be required to learn. To this end the new variable P_{op} was created. This new variable was defined as follows:

Restating Eq. 5:

$$Q_o = Ka_o\sqrt{P_o - P_e}$$

we now model the non-linear square root function by defining a new variable:

$$P_{op} = \sqrt{P_o - P_e}$$

The polynomial abductive network models generated by AIM are now used to replace the previously developed iterative models for the boom actuation part in subsystem 2 of Fig. 2-2. Fig. 4-10 shows the new arrangement. Referring to Fig. 4-10, the inputs to the first network model that determines the input flow, Q_i, are the line pressure, P_i, and the valve spool displacement, x. The second network calculates the output flow, Q_o, and has two input variables - valve spool displacement, x, and the new variable P_{op}.

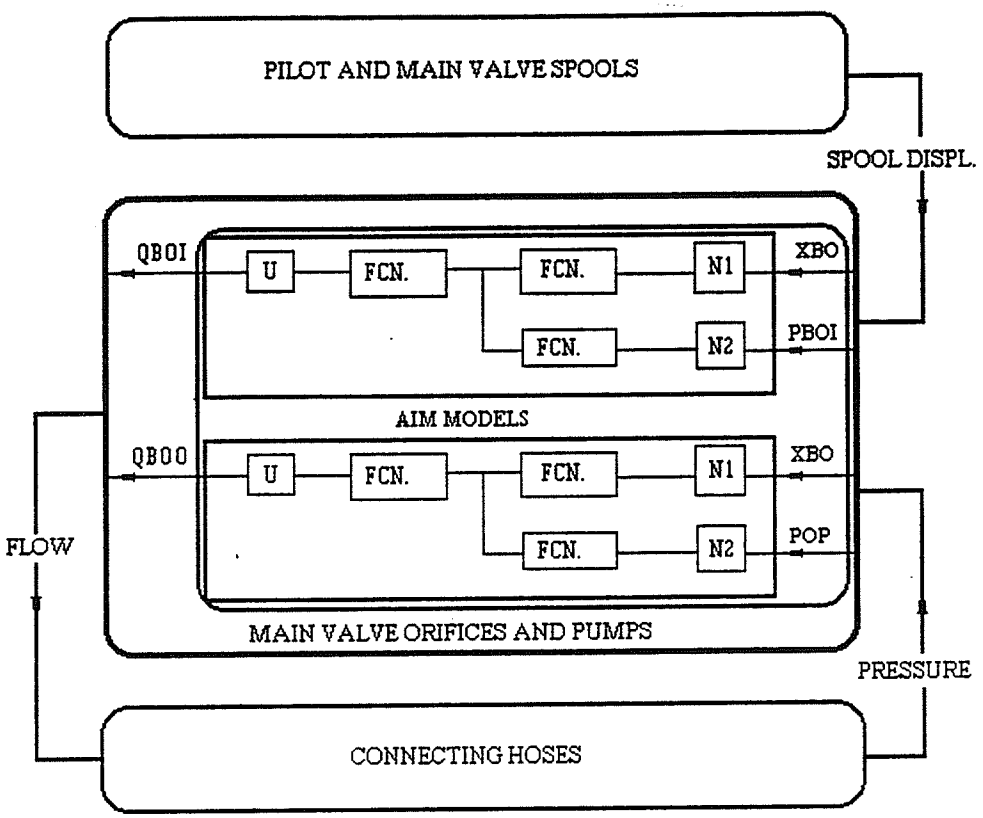


Fig. 4-10 AIM Generated Models and Their Input/Output Arrangement.

4.2.2 Results and Discussion

Figs. 4-11(a) and (b) show the error maps over the complete range of excavator operation for the modeled input and output flows. As with the AIM model for pump pressure the area of greatest error in the input flow model is the area defined by the valve spool displacement range of $0.188 < x \leq 0.376in.$ The maximum error of $-6.85in^3/sec$ equals a 4% error of the maximum flow of $171.5in^3/sec$. The output flow error is concentrated at the extremes of the operating range of the hydraulic system with the maximum error equal to $14.78in^3/sec$ or 0.25% of the maximum output flow of $5580in^3/sec$. This error will have negligible effect on the accuracy of the simulation since a simulation operating at $0.564in.$ of valve spool displacement and at 4000 psi output line pressure is a rare occurance.

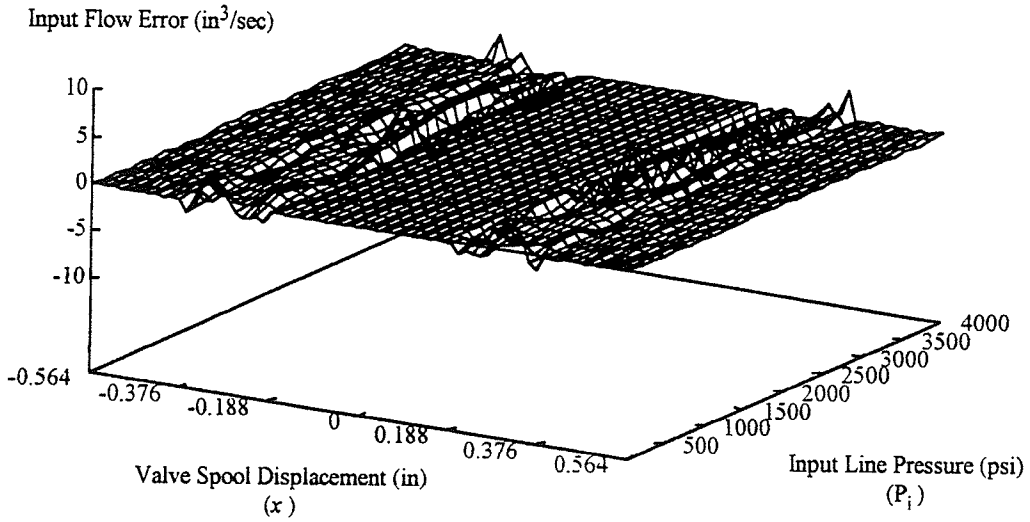


Fig. 4-11 AIM Model Error Maps for: (a) Input Flow; (b) Output Flow.

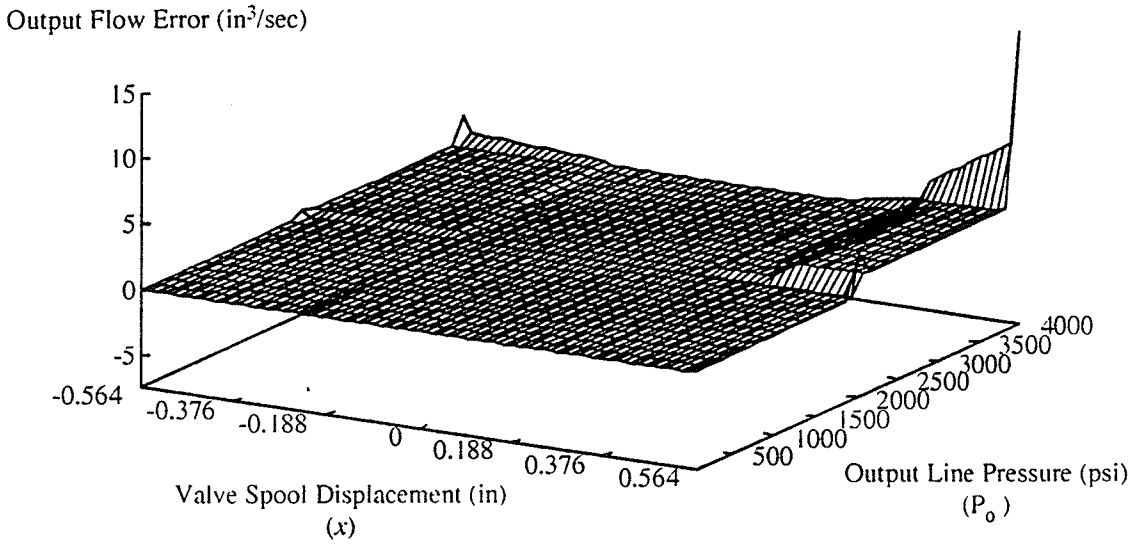


Fig. 4-11 (cont'd).

Figs. 4-12(a) and (b) show the input and output flows generated by a simulation that uses the same input voltage previously used in section 4.2.1 for the pump pressure map model, v = 0.8 sin(1.2566t). Figs. 4-13 and 4-14 show the resulting line pressures and and joint dynamics, respectively.

From the Figs. we see the improvement in the model accuracy by using AIM to model the flows directly instead of modeling the pump pressure and then using the result in the analytic calculation of Q_i (Eq. 1.). The improvement in the AIM modeled input flow (Q_i) shown in Fig. 4-12 results in a much smaller error in the derived physical outputs of the system, the joint displacement and the joint velocity. The error in the joint displacement has been reduced from a maximum of 20*deg* in the pressure simulation, to a maximum error of 0.24 *deg* at time t = 5.0*sec* in Fig. 4-13. The joint velocity error has been reduced from a maximum of 21.0*deg/sec* in Fig. 4-7to 1.51*deg/sec* at time t = 6.53*sec* in Fig. 4-13.

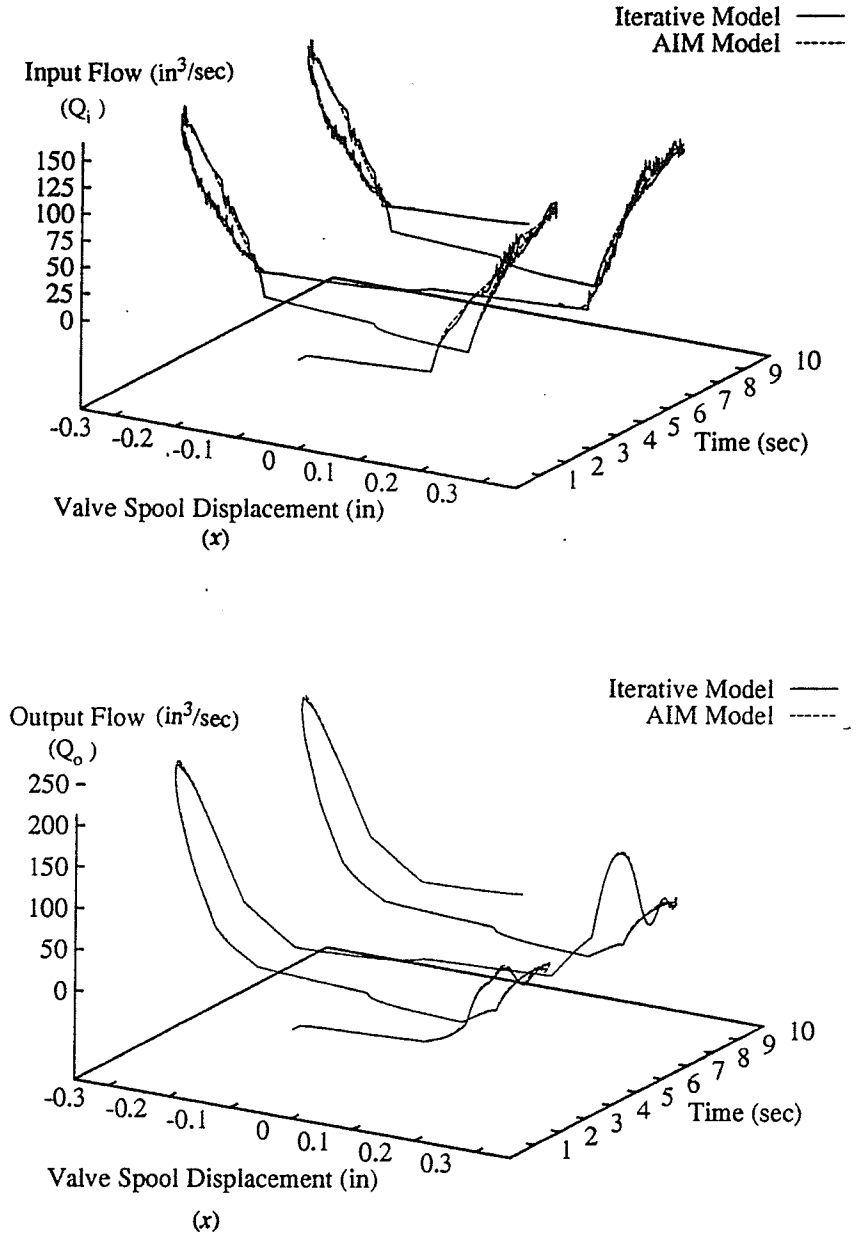


Fig. 4-12 AIM versus Iterative Model for: (a) Input Flow; (b) Output Flow.

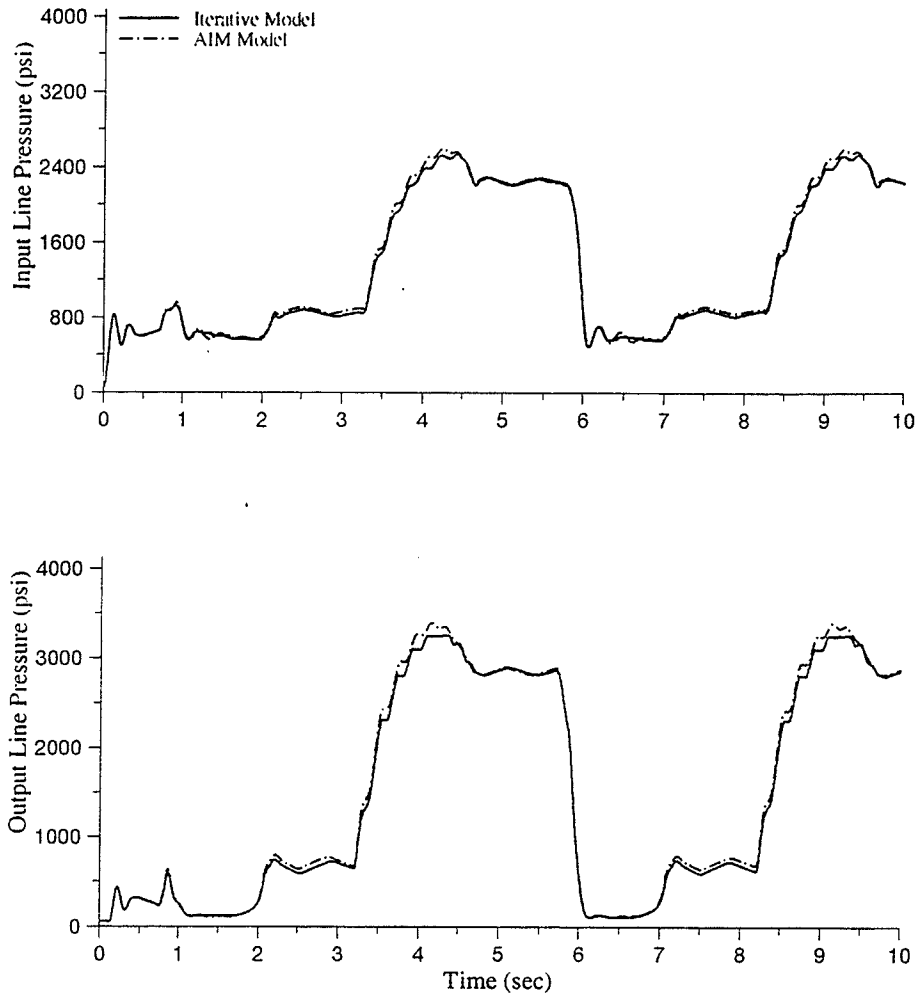


Fig. 4-13 AIM versus Iterative Line Pressures for: (a) Input Flow; (b) Output Flow.

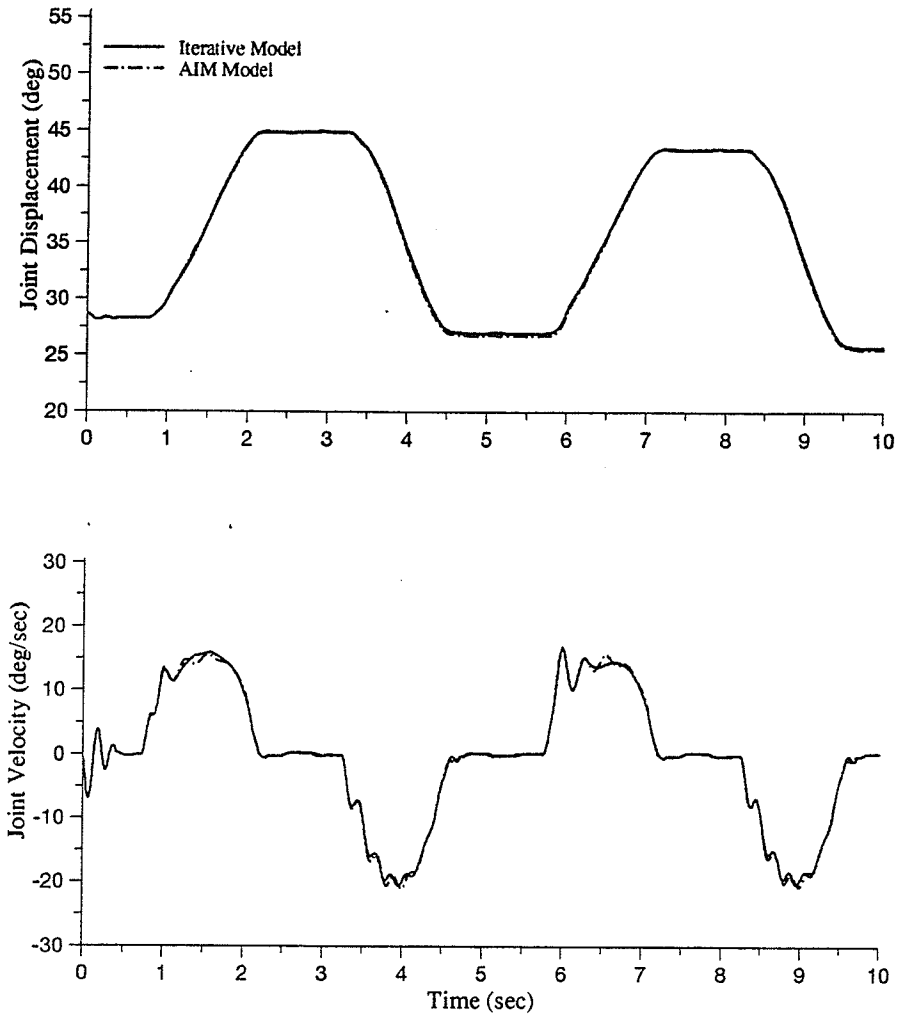


Fig. 4-14 AIM versus Iterative Model for: (a) Joint Displacement; (b) Joint Velocity.

Figs. 4-15(a) and (b) compares the resulting input and output flows generated by a simulation using polynomial abductive networks to those generated by a simulation using the conventional iterative approach. These flows are the result of a sinusoidal control input as shown in the inset of Fig. 4-15(a). The control voltage applied in this test, just opened the spool enough to move the link at a slow speed.

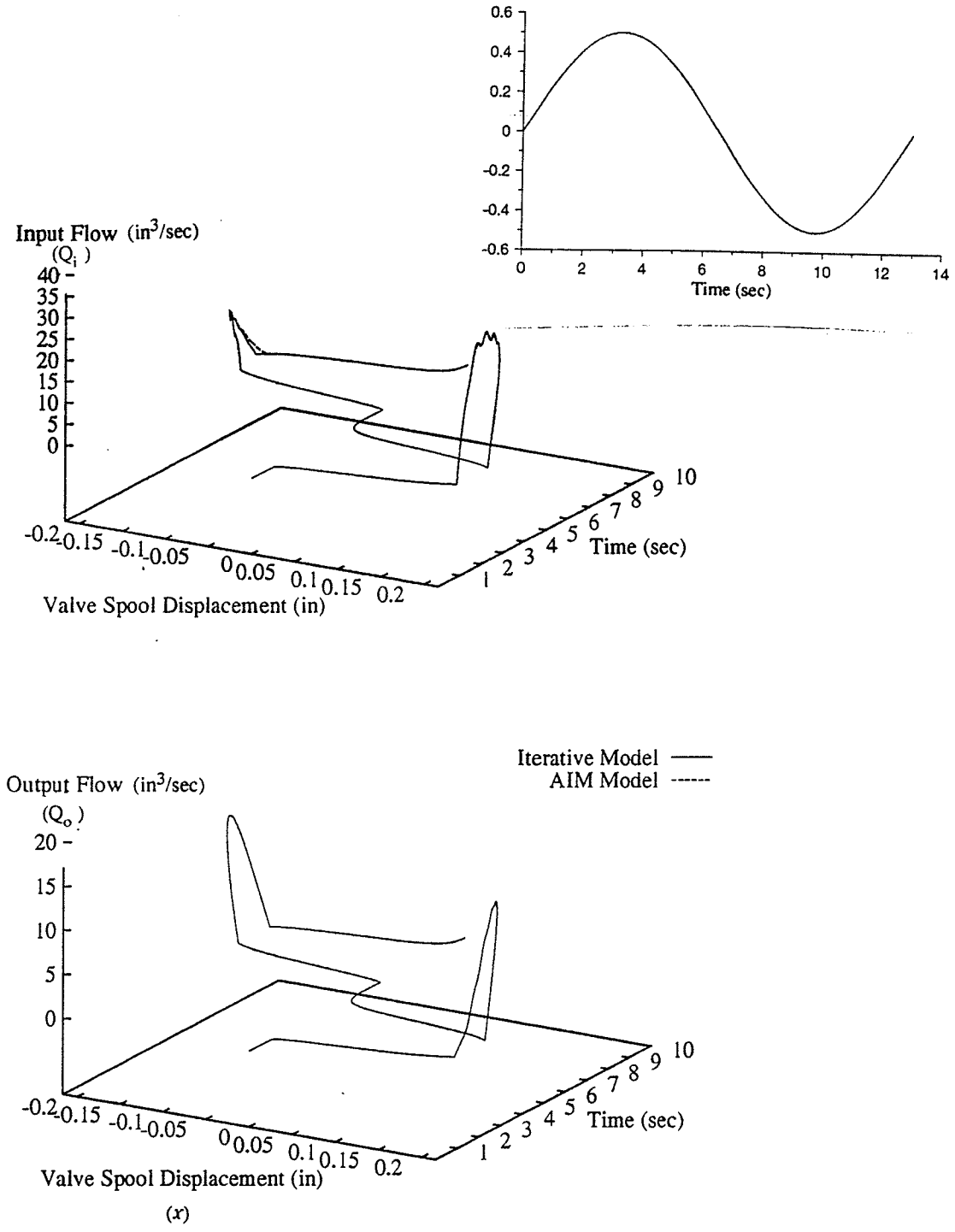


Fig 4-15 AIM versus Iterative Model for: (a) Input Flow ; (b) Output Flow.

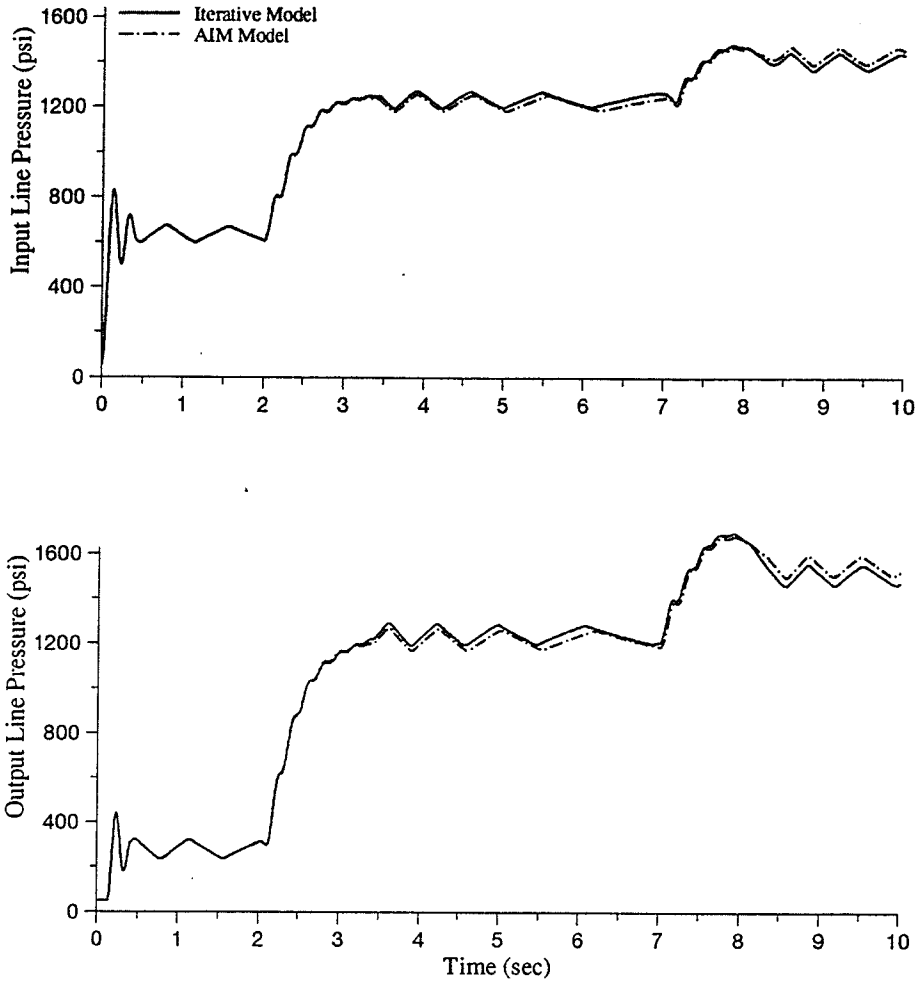


Fig 4-16 AIM versus Iterative Line Pressures for: (a) Input Flow; (b) Output Flow.

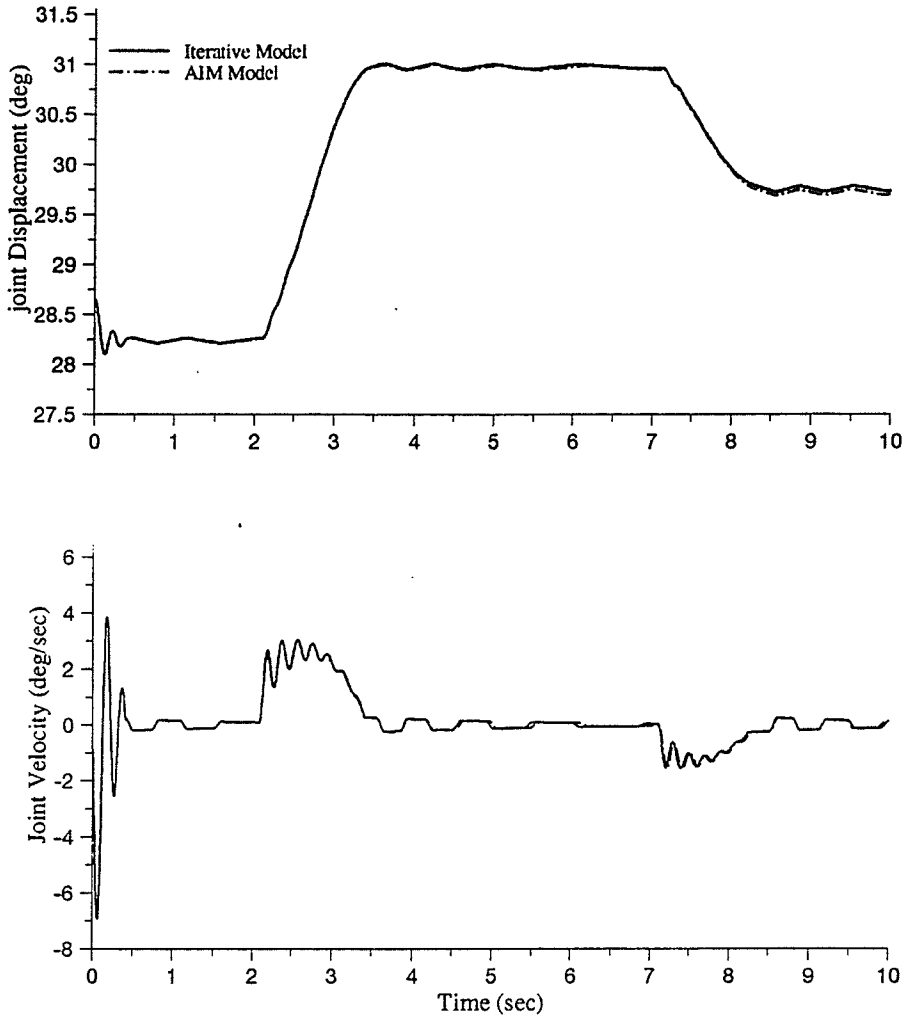


Fig 4-17 AIM versus Iterative Model for: (a) Joint Displacement; (b) Joint Velocity.

Referring to Fig. 4-15, the results for input and output flow for the AIM models is almost indiscernible from the iterative results. The only divergence's in the AIM models occur at time t=3.5*sec.* and t=7.5*sec.* where the AIM model displays a small positive lag during valve spool closure. Figs. 4-16 and 4-17 show that these small lags in flow do not propagate throughout the system as they would have using the AIM models for pump pressure. The reason for this is that one level of computation has been removed from the simulation. Referring to Fig. 4-17(a), the maximum error generated by using the AIM models in the simulation is 0.15*deg* for this specific simulation.

Table 1. shows the computation time required to simulate 10 seconds of motion on a 486/66 MHz personal computer for both the iterative and AIM based simulations.

TABLE 1 Simulation Computation Time Reduction for AIM Model

Iterative Model	AIM Model	Computation Reduction
2.55 sec	0.964 sec	62.19%

The improved computation time is the result of the removal of the iterative routines. The AIM-based approach also removed the need to explicitly determine the values for the orifice areas, a_i, a_e and a_o, Also, since we are bypassing the pump pressure, and therefore Eqs. (1) and (2), any error in the network model will have less accumulated effect on the simulation results. This structure is very feasible in that it provides an avenue to create a completely modeled system from real world experimental data which could be easily obtained.

4.3 NEURAL NETWORKS APPLIED TO HYDRAULIC ACTUATION SIMULATION

4.3.1 Model Generation

Neural networks are now used to provide a comparison to the polynomial network approach using the established modeling methodology described in Chapter 3.2. Therefore, the neural network phase of modeling did not begin until the final AIM models had been constructed. As a result the stages of pressure modeling and valve orifice constant modeling were not pursued and the research focused on using the same variables as in the final polynomial networks.

The neural network simulator that was used is Xerion, a unix based shareware program. The parameters which varied besides the number of hidden units were: The weight cost function, a penalty multiplier for the weights; The zero error radius which set how close a trained output value was to be to the original to be considered correct, and the unit gains which determined the steepness of the sigmoid activation functions.

The initial flow that was modeled using the neural network approach was input flow (Q_i). As shown in Fig. 4-8 the map for Q_i is symmetric about the valve spool displacement axis so only the positive valve spool displacement was modeled. We recognized that the truncated input flow map resembles a sigmoid function extended to a third dimension so our approach was to model the complete map with one network. We approached this problem by creating several 3 layer back propagation networks with the number of hidden units being the difference between them. A training set of 10,000 data points was used for all the networks. Once a close to optimal structure had been found the other parameters were varied to achieve our final network.

When varying the hidden units in each network we determined the best structure by comparing the training error. Fig. 4-18 shows how the training error varies with the different structures, in addition notice that once we have found the optimum number of hidden neurons adding more neurons increases the error.

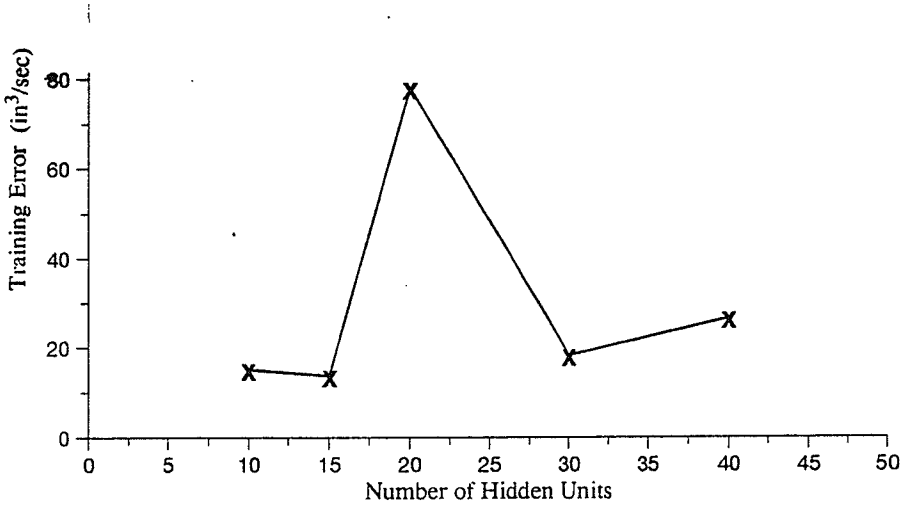


Fig. 4-18 Training Error for Varying Number of Hidden Units

In the figure you will notice that the error values do not form a quadratic type curve although this was expected. Unfortunately, a true error minimization for 20 hidden units was unobtainable due to the slope function becoming unstable at an error value of approximately $78.0 in^3/sec$.

The number of hidden units that are optimal for this problem is 15 with this network structure having an error for Q_i equal to $13.77 in^3/sec$. The final neural network structure for the input flow is as shown in Fig. 4-19.

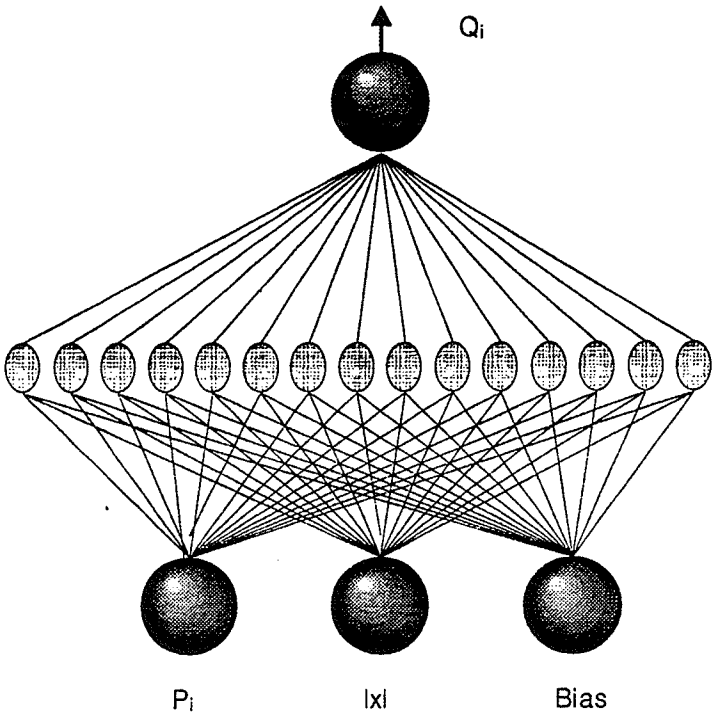


Fig. 4-19 Final Network Structure For Input Flow

To model the output flow, 10,000 data sets were used.. By examining and manipulating Eq. 5 ,as in the section 4.2, the complexity of the relationships to be learned was reduced. However, the method was carried a step further than in section 4.2 by reducing the three dimensional mapping to a two dimensional curve. This is similiar to the solution derived for the AIM output flow, however the AIM approach only substituted for variables and did not modify the map being learned. The way this was accomplished is as follows.

Restating Eq. 5:

$$Q_o = Ka_o\sqrt{P_o - P_e}$$

Since

$$P_{op} = \sqrt{P_o - P_e}$$

we can reduce the curve to two dimensions by substitution and manipulation to get:

$$\frac{Q_o}{P_{op}} = Ka_o = Q_{pop} \tag{7}$$

Therefore, the neural networks for this approach would be based on only one input, the valve spool displacement (x) since it is proportional to a_o, and would have one output, Q_{pop}. Fig. 4-20 shows the new two dimensional curve that was modeled in this approach.

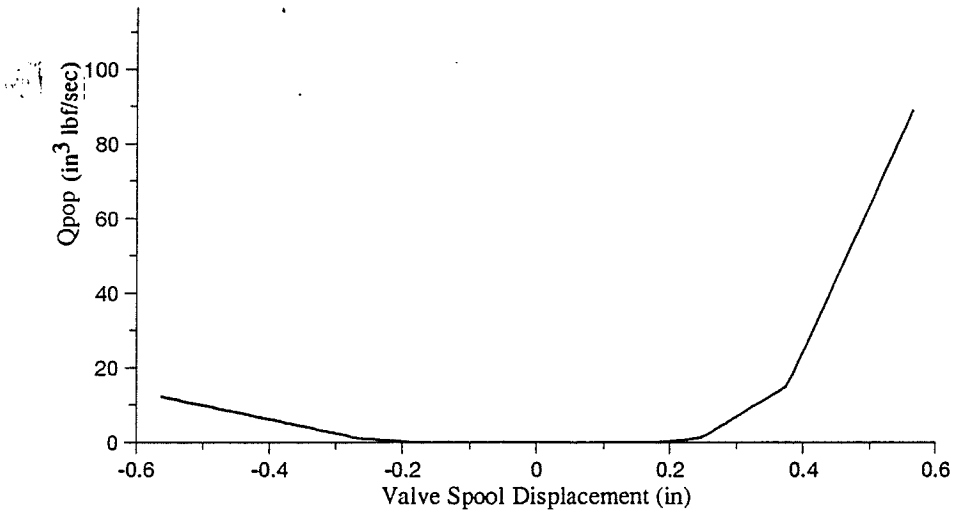


Fig. 4-20 Two Dimensional Map for Output Flow.

It was also decided to break the curve into separate parts based, as in the AIM model. The section of the curve with a zero value was modeled with a direct connection between the input and output layers and large negative weight values were imposed on the connections to ensure an output of zero. The number of hidden units in each network for output flow was seven.

As with the AIM models, the generated neural network models are coded using the C programming language to replace the previously developed iterative models for the boom actuation part in subsystem 2 of Fig. 2-2. The arrangement is similiar to Fig. 4-10 where the network inputs used to determine the input flow, Q_i, are the line pressure, P_i, and the valve spool displacement, x. The second network calculates the output flow, Q_o, and has only one input variable - valve spool displacement, x. Note that in addition to normalizing the data in these networks prior to the inputs being used, we must also post-process the value of Q_{pop} to derive the value of Q_o.

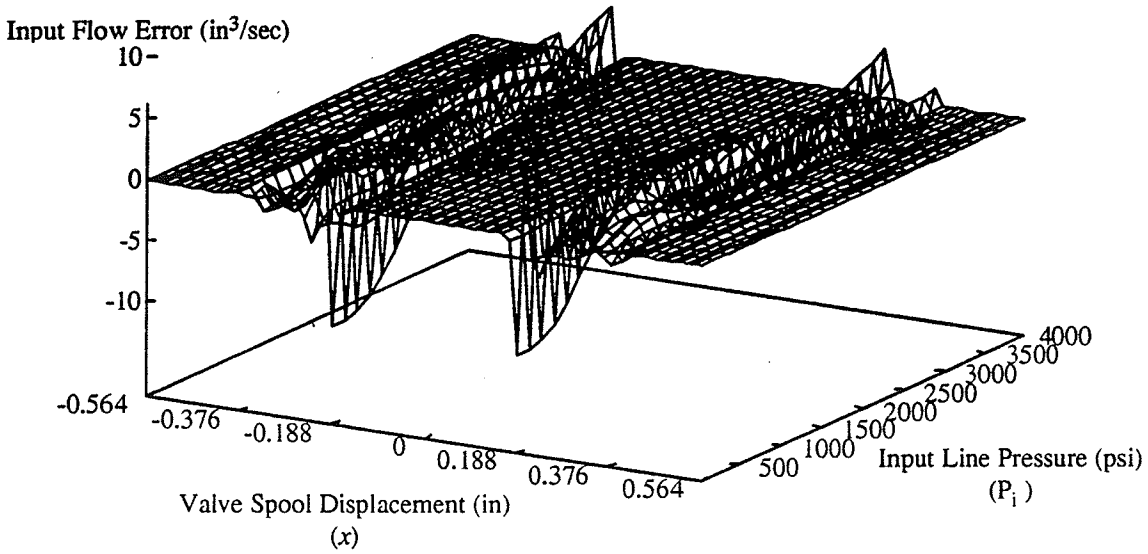
4.3.2 Results and Discussion

Fig. 4-21(a) shows the error map over the complete range of actuator operation for the modeled input flow. As with the AIM model for input flow the area defined by the valve spool displacement range of $0.188 < x \leq 0.376in$ is the area of greatest error. In the neural network model the largest error occurs just after the valve spool has overcome the deadband zone and the error at this point is $\pm 10in^3/sec$, indicating that the neural network model has smoothed the curve at this transition point. In comparison, the AIM model for input flow had a maximum error of $\pm 4.0in^3/sec$ in this region, Fig. 4-11(a). Therefore, the amount of error in the range of $0.188 < x \leq 0.376in$ has essentially doubled by replacing the modular structure of the AIM generated networks with a single neural network. The effect of the larger error is shown in Figs. 4-22 through 4-24 which show the neural network results for a simulation identical to the one used to test the AIM networks

Fig 4-20(b) shows the error in Q_o for the modular neural network structure. As in the AIM model the error is concentrated at the extremes of the operating range of the excavator although error is evident at all the points where the slope of the curve changes. This is due to the neural network reaching saturation in its output neuron at these points so the connection of any two model segments will be slightly smoothed. Note that although the error in Q_{pop} is small throughout the range of valve spool displacement, the error in the output flow, Q_o will increase proportional to the value of the output line pressure (P_o) since:

$$Q_o = Q_{pop} \times P_{op}$$

Therefore, this error will have negligible effect on the accuracy of the simulation at low output line pressures, but causes large errors at higher pressures due to the dependence of the model on output line pressure and the propagation of errors through the simulation.



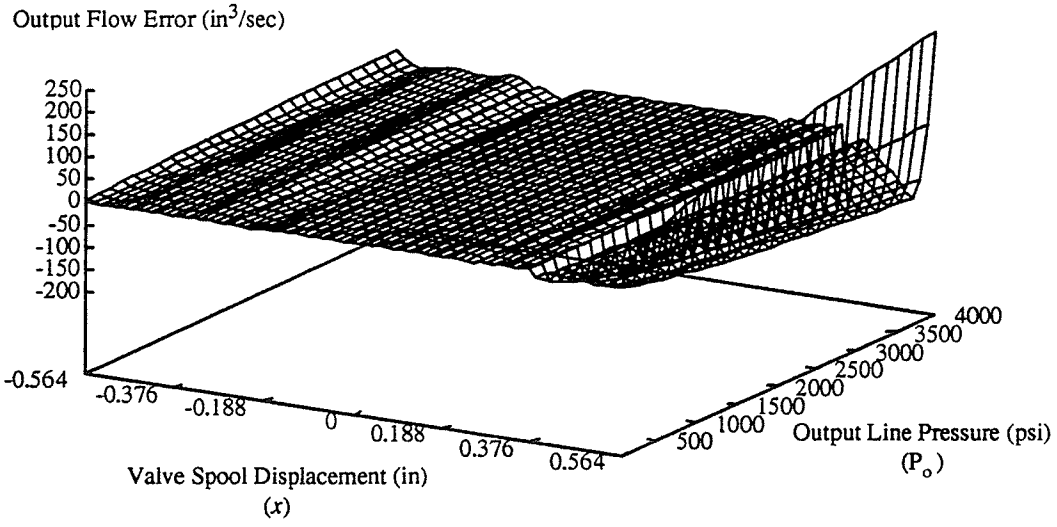
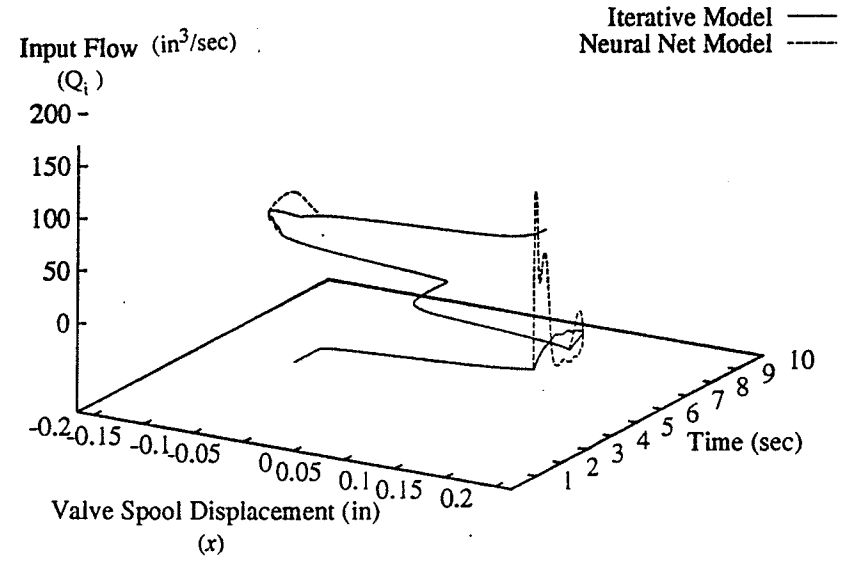


Fig. 4-21 Neural Network Error Maps for: (a) Input Flow; (b) Output Flow.



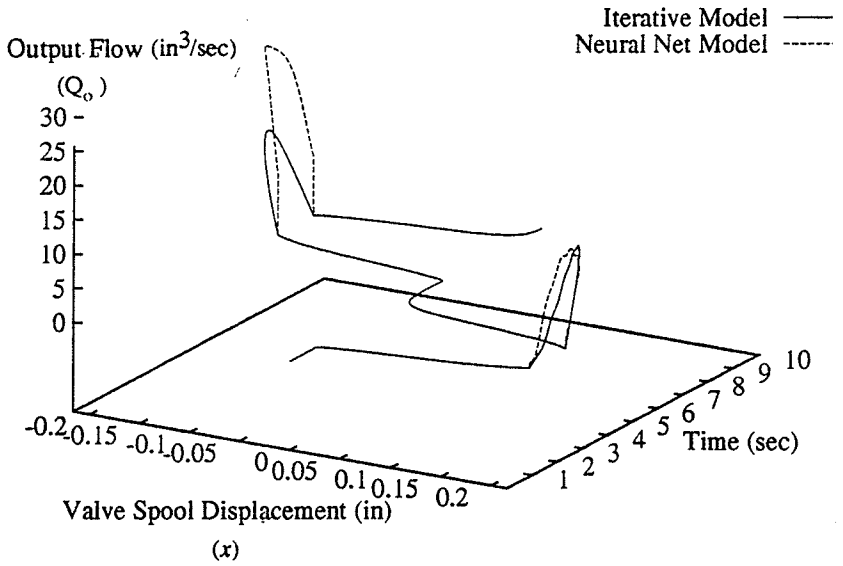


Fig. 4-22 Neural versus Iterative Model for: (a) Input Flow; (b) Output Flow.

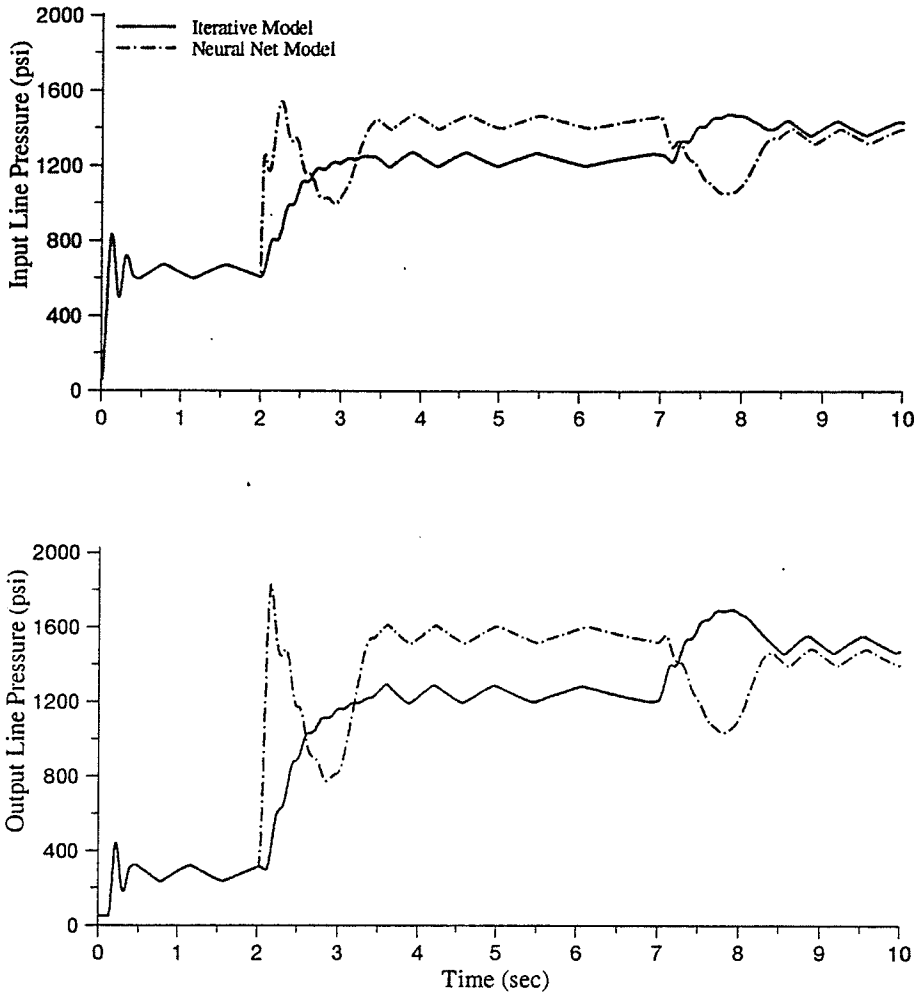


Fig. 4-23 Neural versus Iterative Line Pressure for: (a) Input Flow; (b) Output Flow.

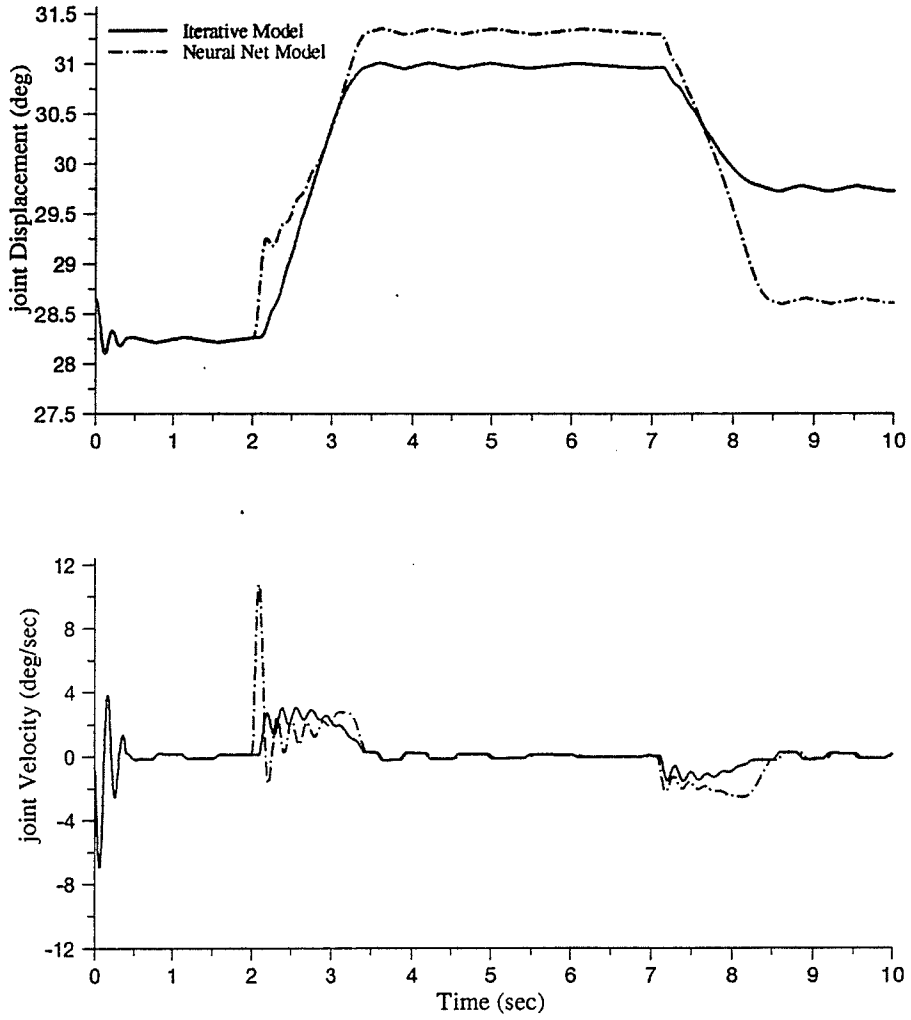


Fig. 4-24 Neural versus Iterative Model for: (a) Joint Displacement; (b) Joint Velocity.

As evidenced by Fig. 4-21 the difference in error between the AIM model and the Neural Network model has a large impact on the simulation at the simulation voltage of v = 0.5sin(0.6283t). The input flow in Fig. 4-22(a) shows a large rise for the neural network model once the valve spool passes 0.188*in.* This is to be expected from the results shown in Fig. 4-21(a), and by the propagation of error which is evidenced by the line pressure and velocity responses at time t=2.0*sec.* in Figs. 4-23 and 4-24. Here we can see the advantage in bypassing the pump pressure in our modeling methodology since the

initial error in flow due to the model did not propagate itself with the input line pressure error.

The simulations in this section and in the AIM modeling section show that with the current model configuration, the AIM models performed better than the neural network models. The neural network approach also performed well, but the evidence of lag and smoothing when the valve opened is more pronounced and due to the propagation of errors in our simulation, resulted in the neural network models not performing as well as the AIM models. It must be noted however that the neural networks used in these models were plain backpropagation neural networks with one hidden layer and the purpose of building the models using neural networks was only to provide a comparison between the AIM models and an established network modeling method

Table 2. shows the computation time required to simulate 10 seconds of motion on a 486/66 MHz personal computer for both the iterative and neural network based simulations.

TABLE 2 Simulation Computation Time Reduction for Neural Network Model

Iterative Model	Neural Network Model	Computation Reduction
2.55 sec	2.08 sec	18.43%

Again improved computation time is demonstrated by the removal of the iterative routines. The neural network approach, however, does not provide as substantial a gain in computation time when compared to the AIM models. Model structure appears to be the cause since for every network many summations and sigmoid function calculations had to be done. In addition, the code was not optimally efficient.

CHAPTER 5
APPLICATION OF THE METHODOLOGY TO THE MULTILINK SYSTEM

5.1 GENERATION OF POLYNOMIAL NETWORKS

Fig. 5-1 shows the schematic of a multi-link actuation hydraulic subsystem which encompasses the swing and stick functions. The two open-centre valves are connected in series with the swing actuator having priority over the stick. The following equations describe the flow distribution from the pump to the actuators:

$$Q = Q_{il} + Q_{el} \tag{8}$$

$$Q_{il} = Ka_{il}\sqrt{P - P_{il}} \tag{9}$$

$$Q_{el} = Ka_{el}\sqrt{P - \overline{P}} \tag{10}$$

$$Q_{e1} = Q_{i2} + Q_{e2} \tag{11}$$

$$Q_{i2} = Ka_{i2}\sqrt{\overline{P} - P_{i2}} \tag{12}$$

$$Q_{e2} = Ka_{e2}\sqrt{\overline{P} - P_{e}} \tag{13}$$

where

$$\frac{dP_{i1}}{dt} = \frac{1}{C_{i1}}(Q_{i1} - Q'_{i1})$$

$$\frac{dP_{i2}}{dt} = \frac{1}{C_{i2}}(Q_{i2} - Q'_{i2})$$

Where Q'_{i1} and Q'_{i2} are related to the swing and stick actuators' velocities, respectively. a_{i1} ,a_{e1} ,a_{i2} , and a_{e2} are determined by the swing and stick main valve spool displacements (x_{sw} and x_{st}).

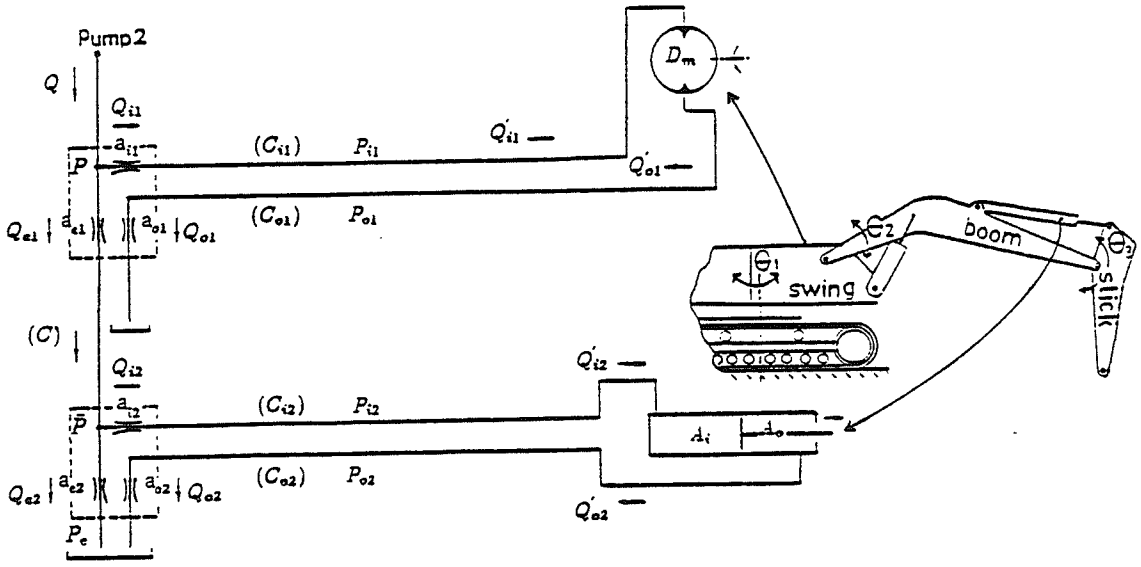


Fig. 5-1 Schematic of the Multi-link Activation Subsystem.

When generating polynomial abductive network models, two outputs are specified for each set of training data - valve flow directed to the stick, Q_{i1}, and valve flow directed to the swing, Q_{i2}. The inputs are the valve spool displacements for swing and stick (i.e., x_{sw} and x_{st}), and the input line pressures, P_{i1} and P_{i2}. By designating two outputs in the training data AIM will create two separate networks. However, since the line pressures and valve spool displacements for both links are included in the training database AIM will possibly find hidden relationships between the swing and stick that are not evident from the analytic equations.

As an extension of the modeling technique used for the single-link, a matrix based on valve spool displacements is used to define the boundaries of the separate network models. This matrix is shown in the following table. For each section an individual polynomial abductive network model is developed.

TABLE 3 AIM Model Matrix for the Multilink System

	$0 \le \lvert x_{sw} \rvert \le 0.188$ (*inch*)	$0.188 < \lvert x_{sw} \rvert \le 0.376$	$0.376 < \lvert x_{sw} \rvert \le 0.564$
$0 \le \lvert x_{st} \rvert \le 0.188$ (*inch*)	***A*** ***both actuators idle***	***C*** ***stick actuator idle***	***E*** ***stick actuator idle***
$0.188 < \lvert x_{st} \rvert \le 0.564$	***B*** ***swing actuator idle***	***D*** ***both actuators active***	

The polynomial abductive network model ***A*** describes a case where both the swing and stick are inactive (flow deadband zones). In the region where model ***B*** is active, the flow into the swing is zero therefore the stick is operating independently. ***C*** represents the case where the flow into the stick actuator is zero but there is flow into the swing. ***D*** is the most general case in which both the swing and the stick are active. Finally, in network model ***E*** there is negligible flow into the stick actuator due to the priority action of the swing. Sections A through E for the swing input flow are shown on Fig. 5-2. The letters on the graph correspond to the relevant section in the model matrix.

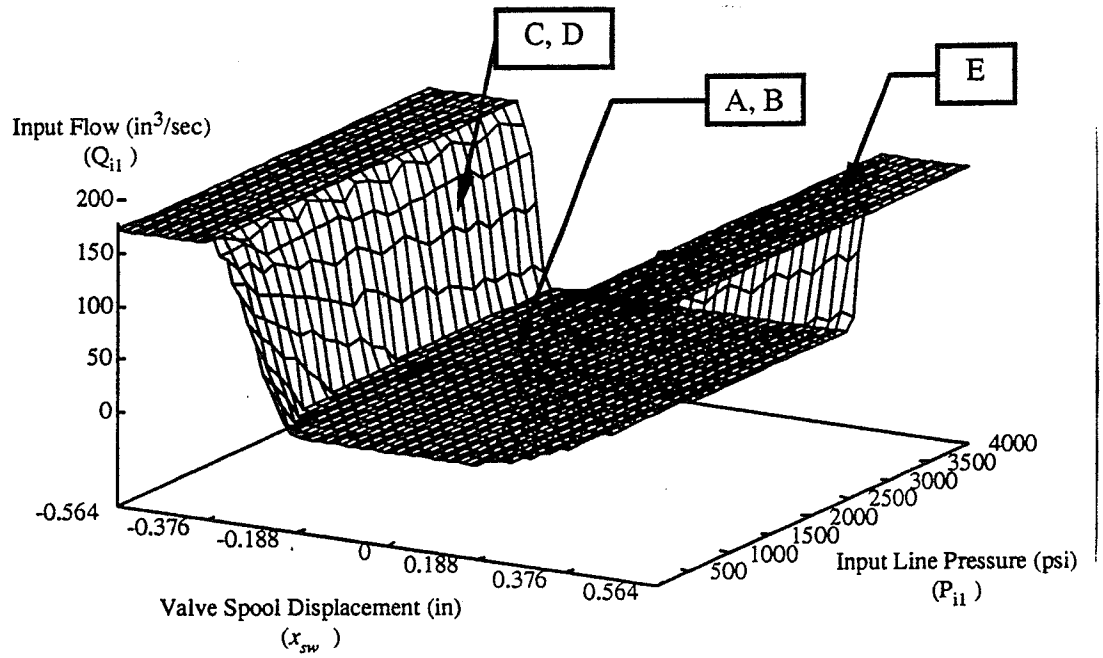


Fig. 5-2 General Map for Swing Input Flow.

Sections **A** through **C** and section **E** for the stick input flow are shown on Fig. 5-3. Fig. 5-3 shows the graph of the stick input flow when swing input flow is zero so all fluid is available for use by this link.

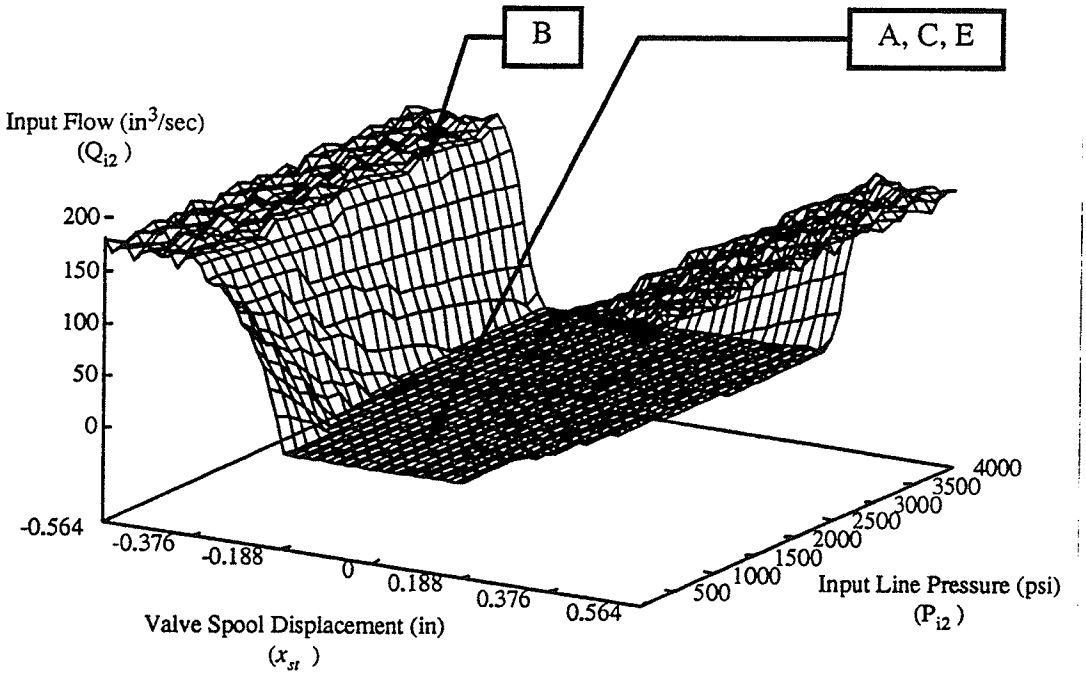


Fig. 5-3 General Map for Stick Input Flow.

Figs. 5-4 (a), (b), and (c) show the resulting stick input flow for section **D**. In this section both the swing and stick are active so not all fluid is available to the stick. The amount of fluid that flows from the swing to the stick (Q_{12}) determines the shape of the general mapping. In Fig. 5-4 (a) the amount of flow sent to the stick is $50 in^3/sec$. In Fig. 5-4 (b) the amount is $100 in^3/sec$ and in Fig. 5-4 (c) the amount is $150 in^3/sec$. This kind of variation in the mapping for stick input flow is expected to increase the amount of error in our AIM models since AIM is modeling not one, but several general maps for stick input flow.

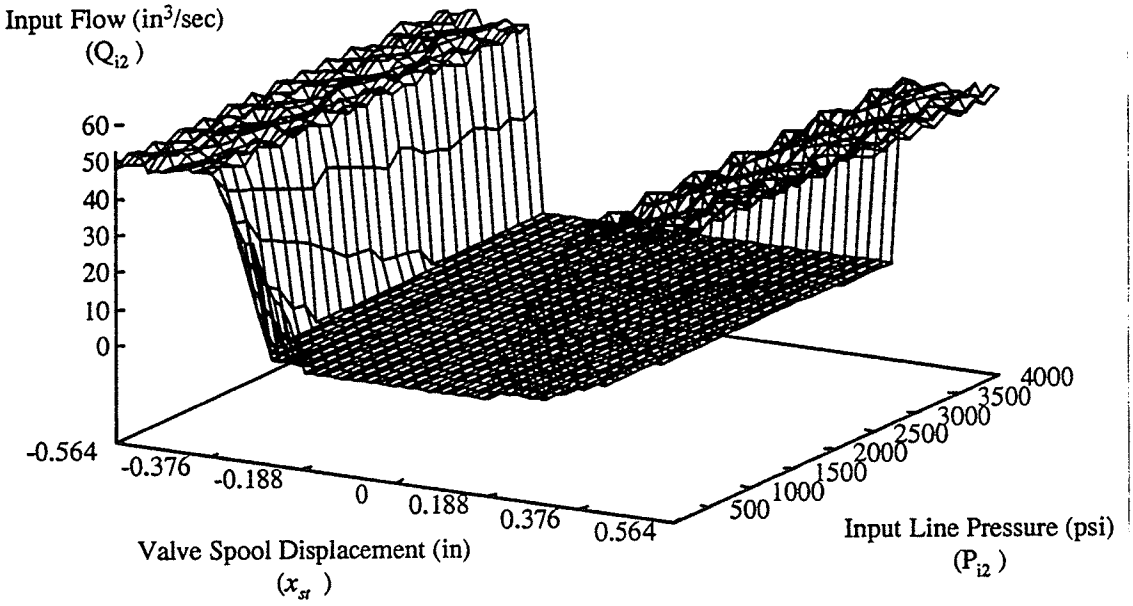
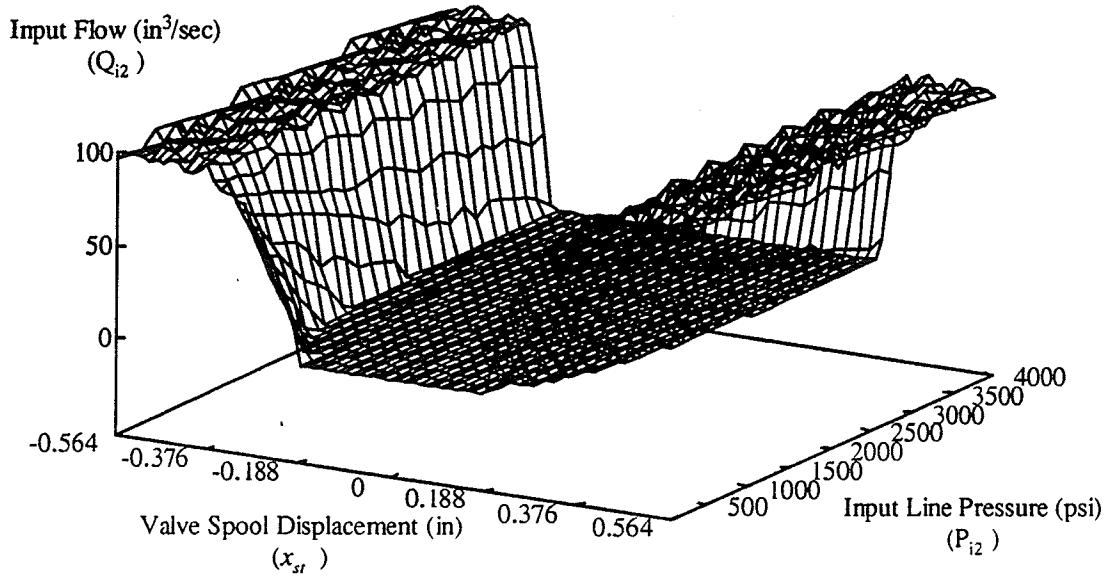


Fig. 5-4 Stick Input Flow Mappings for: (a) Q_{12} = 50 in³/sec; (b) $Q_{12} = 100 in3/sec$; (c) $Q_{12} = 150 in3/sec$.


Input Flow (in³/sec)
(Q_i2)
100
50
0
-0.564
-0.376
-0.188
0
0.188
0.376
0.564
Valve Spool Displacement (in)
(x_st)
500
1000
1500
2000
2500
3000
3500
4000
Input Line Pressure (psi)
(P_i2)

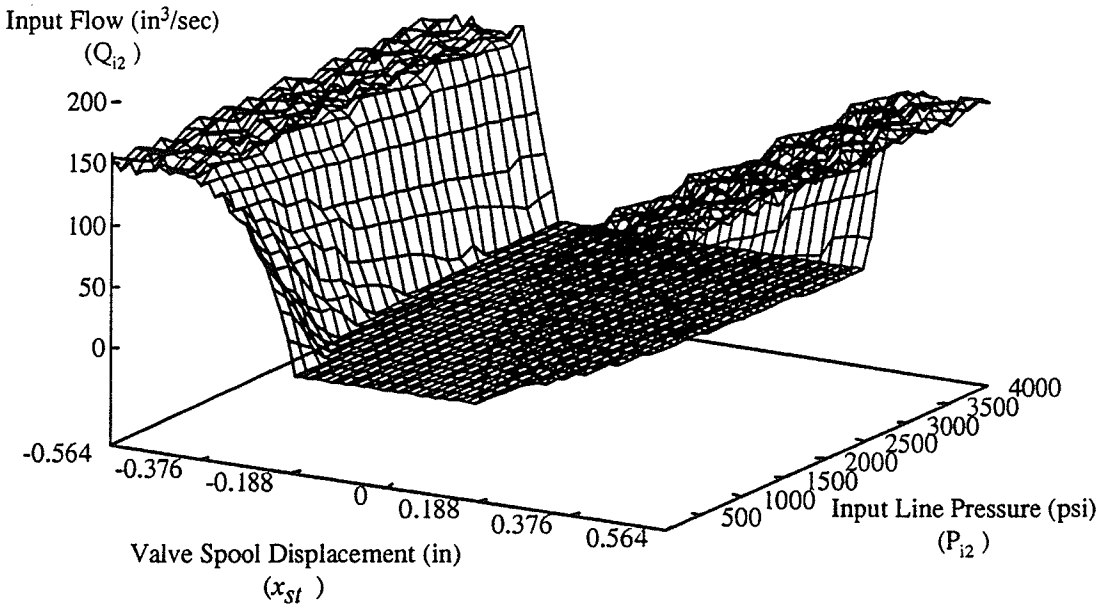

Input Flow (in³/sec)
(Q_i2)
200
150
100
50
0
-0.564
-0.376
-0.188
0
0.188
0.376
0.564
Valve Spool Displacement (in)
(x_st)
500
1000
1500
2000
2500
3000
3500
4000
Input Line Pressure (psi)
(P_i2)

Fig. 5-4 (cont'd).

Output flows back to the valves are not coupled and therefore equations describing them are similar to Eq. (5) in the previous chapter.

$$Q_{o1} = Ka_{o1}\sqrt{P_{o1} - P_e} \tag{14}$$

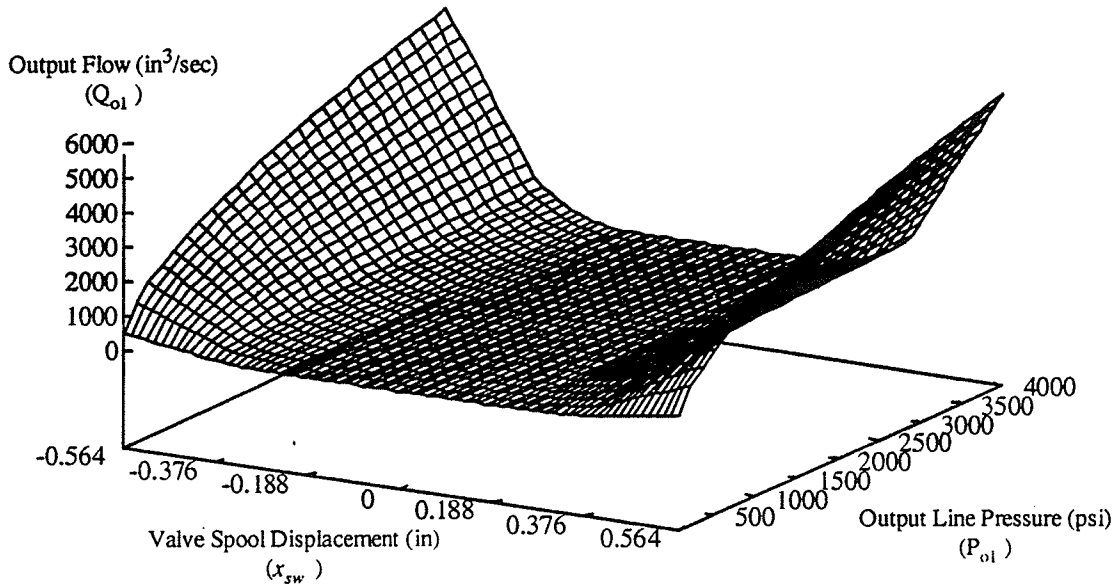
$$Q_{o2} = Ka_{o2}\sqrt{P_{o2} - P_e} \tag{15}$$

where

$$\frac{dP_{o1}}{dt} = \frac{1}{C_{o1}}(Q'_{o1} - Q_{o1})$$

$$\frac{dP_{o2}}{dt} = \frac{1}{C_{o2}}(Q'_{o2} - Q_{o2})$$

The methodology of modeling the output flows is the same as modeling the output flow in the single-link case using neural networks, where the 3-D map is reduced to a 2-D curve. The output flows are independent of each other and the maps are similiar to Fig. 4-10. The output flow maps for the swing and stick are shown in Fig. 5-5.



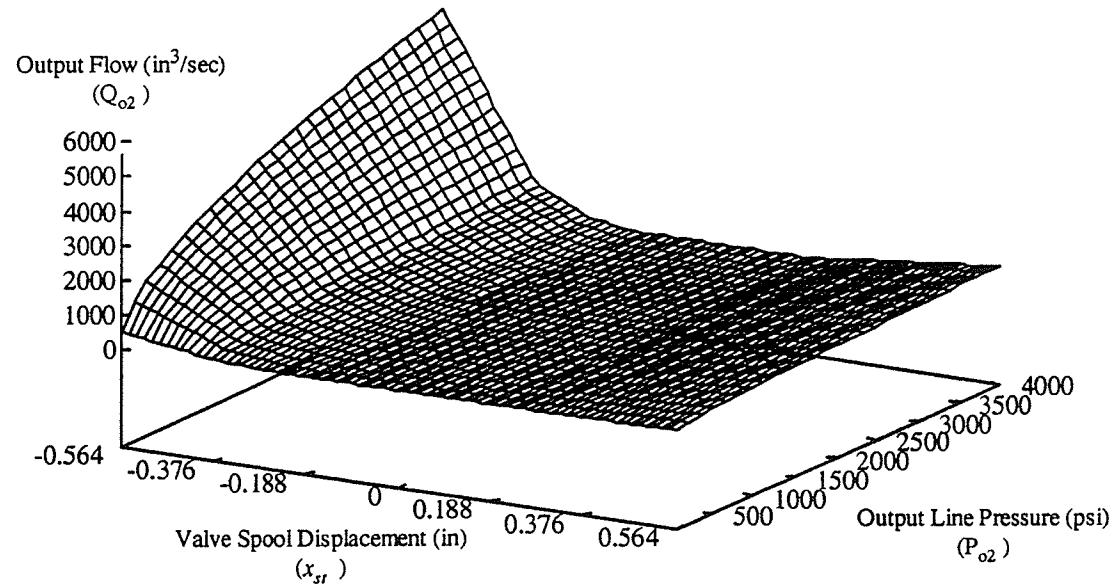


Fig. 5-5 Output Flow Maps for: (a) Swing; (b) Stick.

5.2 GENERAL MODEL RESULTS

Instead of showing the complete error maps for swing and stick and then identifying the areas of error for each model section (**A** through **D**) we will instead show the error graph for swing or stick for that particular model section only. Note that since symmetry was used in the generation of the model we will only show the error graph for the positive valve spool displacement.

The models for section **A** describe the case where both swing and stick are inactive and flow goes directly to the tank. Therefore, the flow into both swing and stick is zero. To model this section AIM simply created a direct link between inputs and outputs, with the only input that wasn't pruned from the model being the respective valve spool displacement. All outputs were set to zero in the model so the resultant error between the AIM model and the iterative model will be zero.

The result for the stick portion of section **B** is shown in Fig. 5-6. The result for the swing is not shown since it is a direct one to one mapping of inputs to outputs with all outputs equal to zero as in section **A**. Fig. 5-6 shows that the primary errors in this model occur around the points where the flow map changes slope, $0.188in$, and $0.376in$. When the valve spool opens at $0.188in$ the AIM model has a higher result than the iterative model as shown by the large negative value. As the valve spool is opened further the model error settles into a range of $\pm 10in^3/sec$ or 6% of maximum flow, with larger fluctuations of $\pm 20in^3/sec$ (12% of maximum flow) occurring about the valve spool displacement of $0.376in$. Clearly, we would expect a large amount of error in any simulation where the stick operates independently and at low input line pressure for any sustained period.

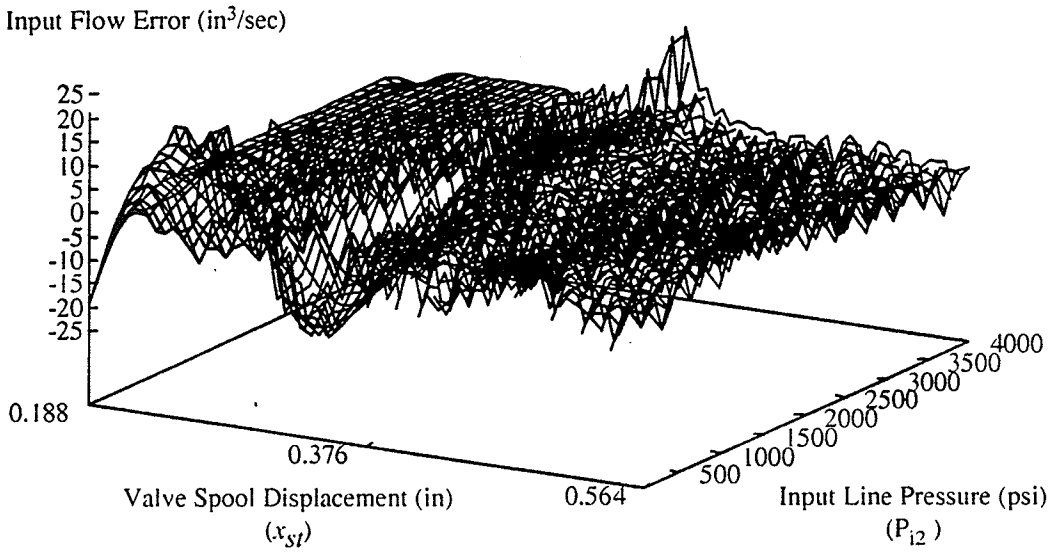
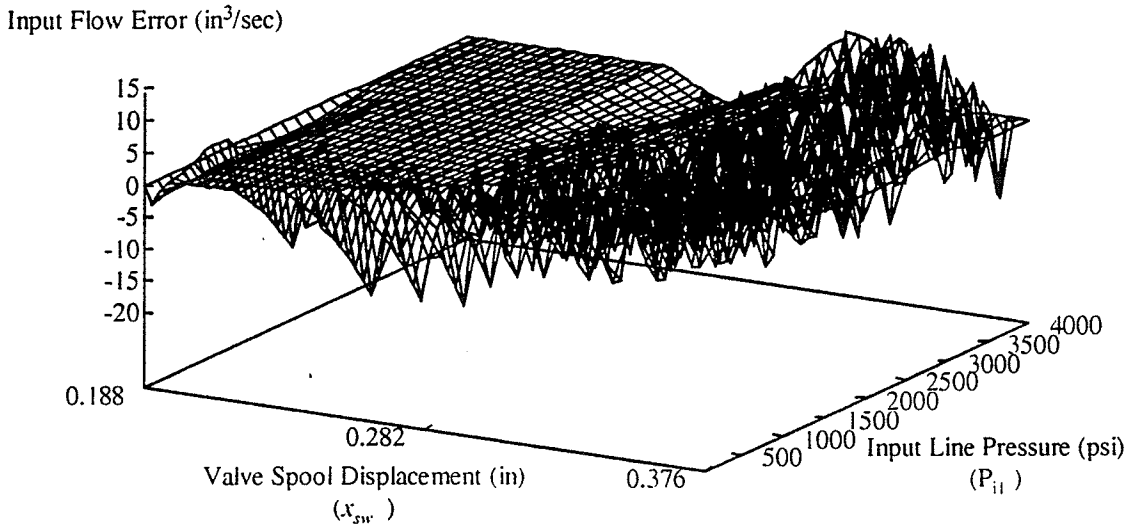


Fig. 5-6 Stick Input Flow Error for Section **B**

The model generated by AIM for section **C** swing input flow used three inputs. The inputs that AIM found useful for this model are the swing valve spool displacement (x_{sw}), the swing input line pressure (P_{i1}), and the stick valve spool displacement (x_{st}). To show the error in the model results for the swing we will graph the swing error for three separate values of stick valve spool displacement. These three responses are shown in Figs. 5-7 (a) (b) and (c) which show the error in the AIM model for stick valve spool displacement values of $0.0in$, $0.94in$ and $0.187in$, respectively.

The results for all three cases are consistent, with the largest AIM model overshoot occurring in the region where the line pressure into the swing motor is just above tank pressure. This is the area where there is the minimal amount of flow into the motor and Fig. 5-2 shows how this region of flow is different from the rest of the map with the flow value increasing at a greater rate at the low line pressure.



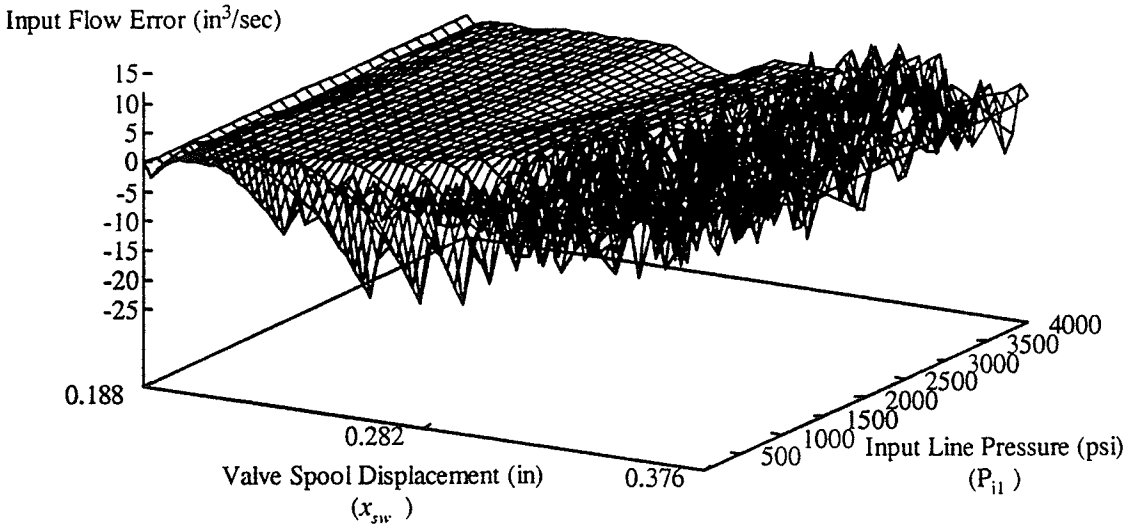


Fig.. 5-7 Swing Input Flow Error for Section C with: (a) x_{sl} =0.0*in*; (b) x_{sl} = 0.94*in*; (c) x_{sl} = 0.187*in*.

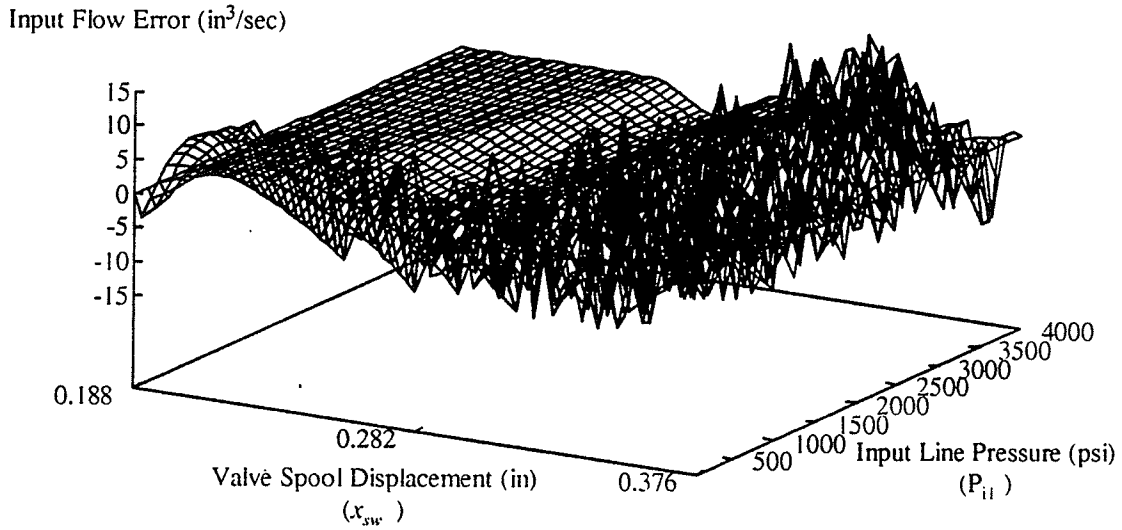


Fig. 5-7 (cont'd).

The graphs are also similar in the region from 0.282*in* to 0.376*in* over the complete range of input flow. Here, the error is not great in magnitude, with the maximum positive error being 15in^3/sec or 8.7% of maximum flow, but the overfitting which we used in designing the AIM models is evident in the random and varying error. Also note that the error graph at the extreme range of stick valve spool displacement, Fig. 5-7 (c), shows the greatest amount of error over the entire range although the amount of AIM model overshoot at low swing input line pressure (P_{il}) is reduced. Thus, the effect of including the stick valve spool displacement (x_{st}) into the model for swing flow in section C is to decrease the value of the AIM generated swing input flow (Q_{il}) for increasing values of x_{st}.

As in the previous sections, the stick input flow model for this section is not shown since the stick is in its deadband zone which results in a direct one to one mapping with all output flows being equal to zero.

The AIM models for section **D** are expected to have the highest amount of error of all the model sections. The reason is that both the swing and stick are active in this region so the complexity of the maps to be learned is much greater. We saw in Fig. 5-4 the effect that the swing has on the stick input flow map and we expect that modeling such a relationship will be difficult at best.

The AIM generated models for both the swing and stick input flows used all four inputs; x_{sw} , x_{st} , P_{i1} , and P_{i2} , to generate the outputs Q_{i1} and Q_{i2}. This is indicative of the complexity and interaction among the swing and stick in this region. Figs. 5-8(a) through (c) shows the error inherent in the AIM model for the swing input flow (Q_{i1}). Note that these graphs were created at specific values of stick valve spool displacement (x_{st}) and that stick input line pressure (P_{i2}), although not shown, is equal to the swing line pressure (P_{i1}).

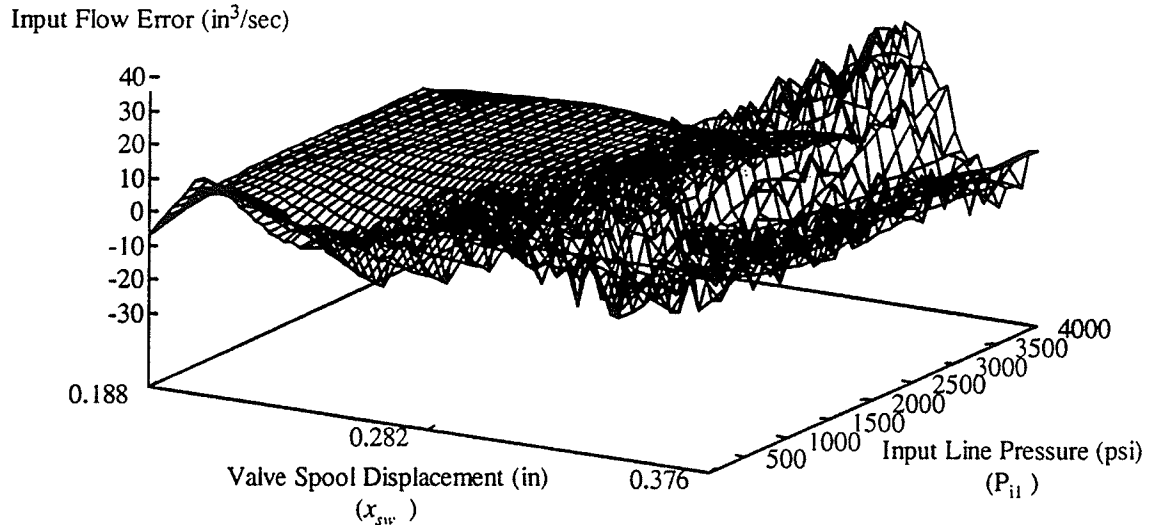
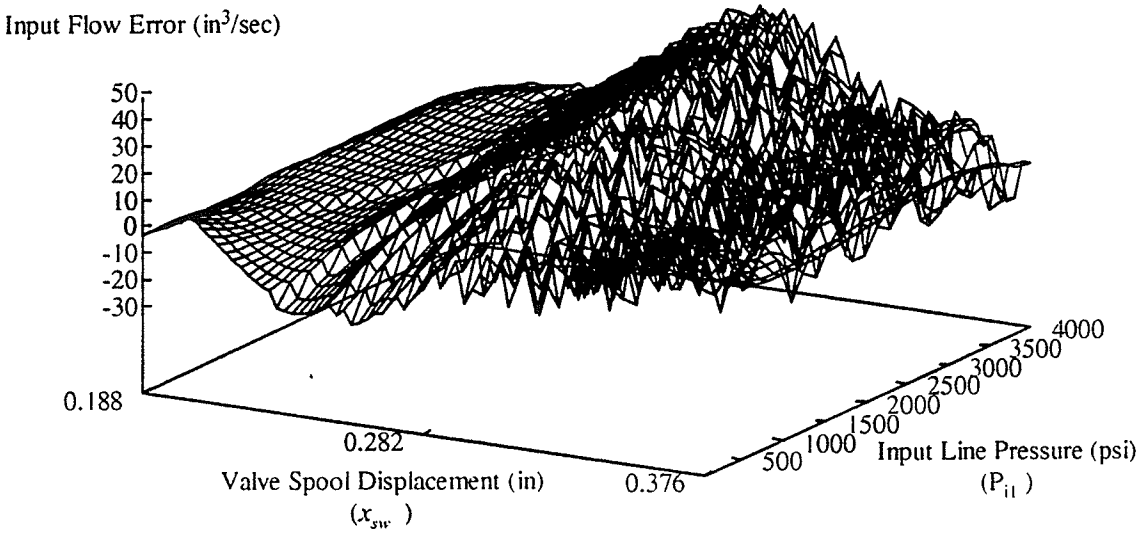


Fig. 5-8 Swing Input Flow Error for Section **D** for: (a) $x_{st} = 0.190in$; (b) $x_{st} = 0.376\ in$; (c) $x_{st} = 0.563in$.


Input Flow Error (in³/sec)
50
40
30
20
10
0
-10
-20
-30
0.188
0.282
0.376
Valve Spool Displacement (in)
(x_{sw})
500
1000
1500
2000
2500
3000
3500
4000
Input Line Pressure (psi)
(P_{il})

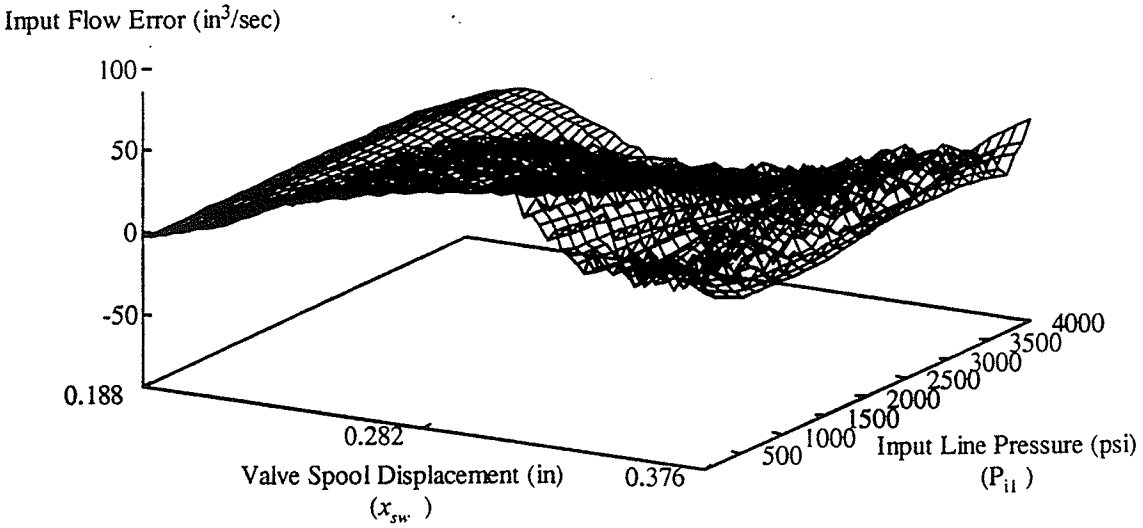
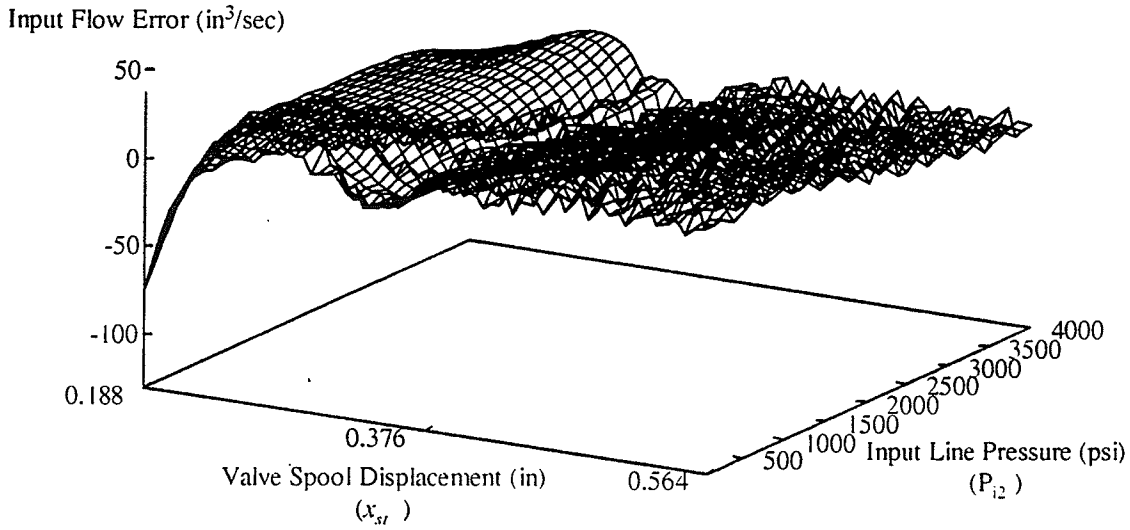

Input Flow Error (in³/sec)
100
50
0
-50
0.188
0.282
0.376
Valve Spool Displacement (in)
(x_{sw})
500
1000
1500
2000
2500
3000
3500
4000
Input Line Pressure (psi)
(P_{il})

Fig. 5-8 (cont'd).

From the graphs in Fig. 5-8 we see two prominent characteristics for error in the swing input flow. First, the maximum error in this model occurs in the region where both the swing input line pressure (P_{i1}) and the stick input line pressure (P_{i2}) are greater than 2000*psi*. Secondly, as the value of the stick valve spool displacement increases so does the magnitude of the error. The reason for this behavior is that as the stick valve spool displacement increases, the AIM model decreases the value of the swing input flow (Q_{i1}). It does this because AIM is attempting to learn a relationship that is not evident in Fig. 5-2. The relation is that as the flow to the stick increases (Q_{i2}) so does the fluid pressure to the stick (shown as $\overline{P}$ in Eq. 12). This results in a lower value for the swing input flow than shown in Fig. 5-2 due to the diversion of fluid and pressure to the stick.

One qualification of the results shown in Fig. 5-8 must be made at this point. In order to generate the values for the iterative maps it was necessary to loosen the restrictions on the convergence test used in the iterative routine to determine the pump pressure (P) amd the pressure of the fluid that is diverted to the stick ($\overline{P}$). The new value of 'eps' that was used was 0.1 and its effect is that the iterative result has a larger range of 'correct' values due to this looser convergence test. This topic will be dealt with more fully in section 5.4.

Fig. 5-9(a) through (c) shows the error maps for the stick input flow (Q_{i2}) for section **D** of the model matrix. As with the swing input flow, all four inputs; x_{sw}, x_{st}, P_{i1}, and P_{i2}, are used to determine the output flow in the AIM model. We hold the swing valve spool displacement constant in each graph and the swing input line pressure (P_{i1}) will be equal to the stick input line pressure (P_{i2}).



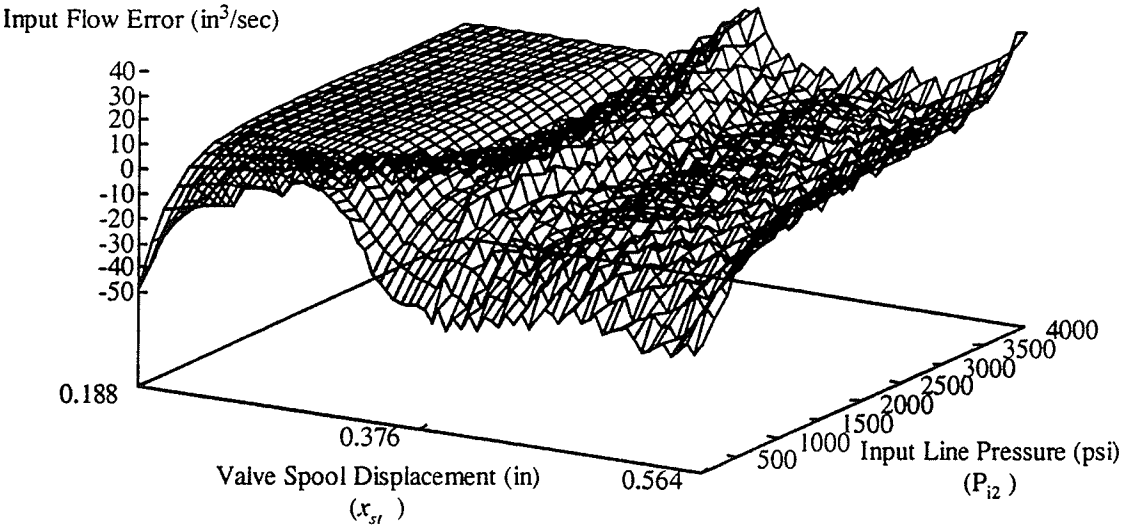


Fig. 5-9 Stick Input Flow Error for Section **D** with: (a) $x_{sw} = 0.190in$; (b) $x_{sw} = 0.270in$ (c) $x_{sw} = 0.375in$.

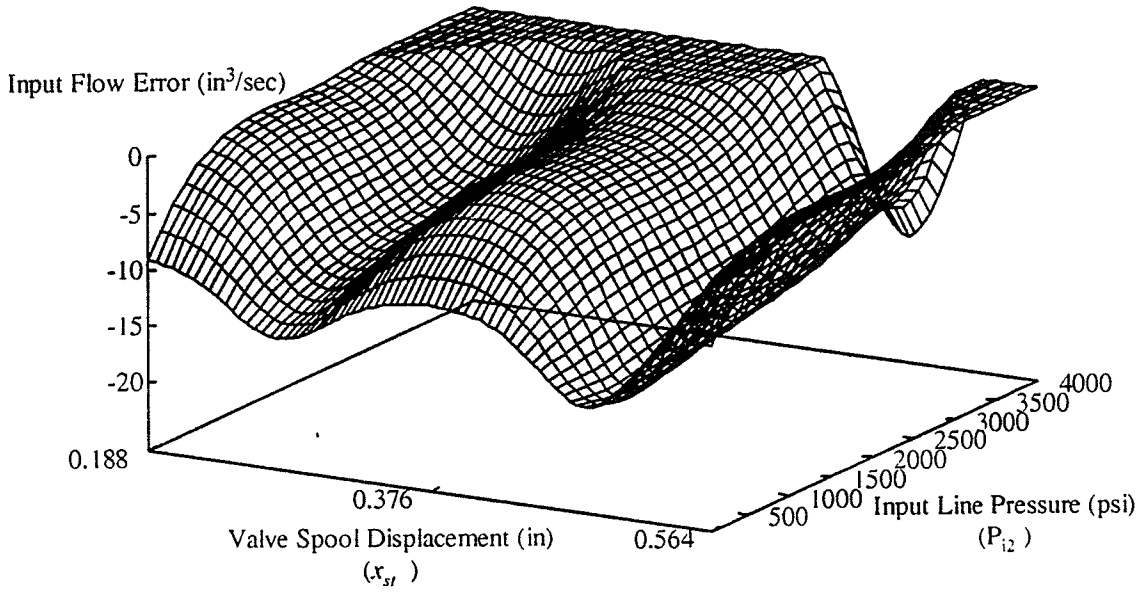


Fig. 5-9 (cont'd).

Fig. 5-9(a) through (c) show several interesting results. First, for all three graphs the AIM model has a prevalent overshoot (shown by a negative error) when the stick valve spool just opens. This is expected due to the fact that the flow is quite non-linear in this region and we know from the results in chapter 4 that this is a region of maximum error. Fig. 5-4 (a) through (c) displayed the effect that increasing values for swing valve spool displacement and input line pressure (i.e. a lower Q_{i2}) had on the stick input flow. Therefore, as we increase the value of x_{sw} and P_{i1} the stick input flow map that we are attempting to model changes from one similar to Fig. 5-3 to the one shown in Fig. 5-4(a). Since AIM must generalize over all the maps in training the model we would expect a large amount of positive error when the values for x_{sw} and P_{i1} are low and a large amount of negative error when they are high. This is exactly the case in Fig. 5-9. In Fig. 5-9(a) we see that when the swing input line pressure (P_{i1}) is low (the right side of the graph) there is a steady positive error. This error is reduced as P_{i1} increases, however the

prevalent type of error in this graph is still error generated by the iterative model having a larger value than the AIM model. Fig. 5-9 (c), by comparison, shows no positive error values whatsoever.

The error graph for the swing portion of section **E** of the model matrix is shown in Fig. 5-10. No stick input flow error graph is shown due to the stick being inactive in this section. The inputs used by the AIM model are the swing input line pressure (P_{il}) and the swing valve spool displacement (x_{sw}).

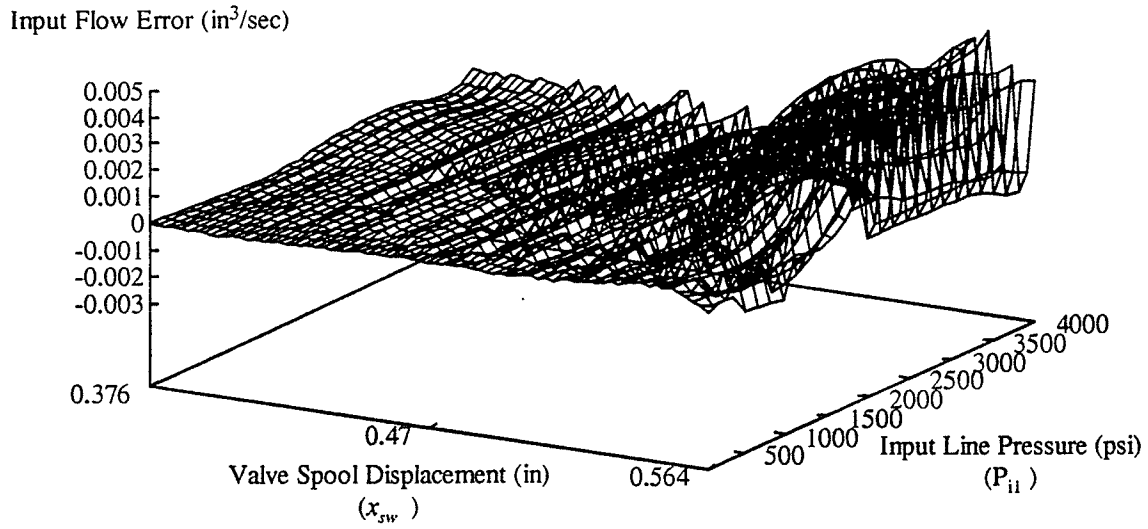


Fig. 5-10 Swing Input Flow Error for Section **E**.

As we would expect with this section, the error in the AIM model is negligible. This is due to the fact that section **E** of the model matrix covers the area of Fig. 5-2 where the swing input flow is essentially constant. In addition, there is no interaction between the swing and the stick so the stick valve spool displacement and line pressure have no effect.

As in chapter 4, it is necessary to model the output flows using AIM in order that all valve orifice areas could be removed from the simulation. The graphs in Fig. 5-11 show the error in the output flow models as compared to the iterative models.

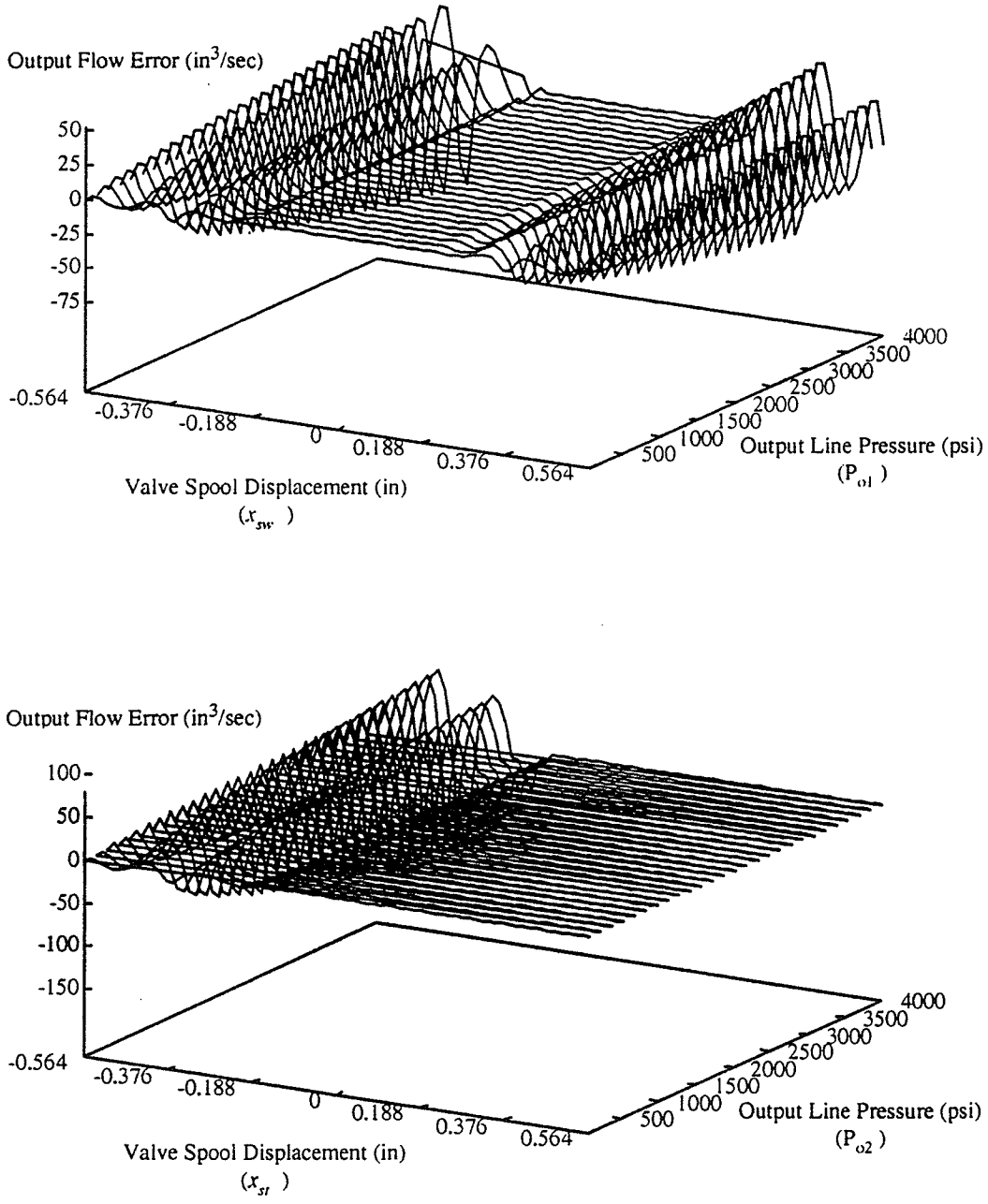


Fig. 5-11 Output Flow Error for: (a) Swing,; (b) Stick.

The error in the values of Q_{pop} have translated to an output flow error proportional to the output line pressure. As Fig. 5-11 shows the error is negligible at low output line pressures but could affect the accuracy of the simulation at high output line pressures.

The maximum error for two dimensional map of Q_{pop} for the swing output flow (Q_{o1})is -1.20*in³ lbf/sec* in the region of ±0.376*in.* This translates to a maximum error in Q_{o1} of -124.99*in³/sec* (2.2% of maximum flow) at the maximum P_{o1} value of 4000*psi.* The rapid changes from positive to negative error and vice versa are due to the overfitting of the AIM model to the iterative values. The result is that instead of a smooth curve generalizing the output flow and causing a constant error , the AIM model overcompensates for errors during learning causing an opposite error as it proceeds through the data set.

The maximum error for the two dimensional map of Q_{pop} for the stick output flow (Q_{o2}) is -1.99*in³ lbf/sec* in the region of - 0.376*in.* This translates to a maximum error in Q_{o2} of -75.65*in3/sec* (1.3% of maximum flow) at the maximum P_{o2} value of 4000*psi.* Fig. 5-5 (b) shows that the flow for this link is active primarily when the stick valve spool is displaced negatively. As a result the error for this map is also concentrated in this area with the largest error occurring where the rate of output flow changes. As with the swing output flow, the model for the stick output flow shows signs of being overfitted.

5.3 SIMULATION RESULTS AND DISCUSSION

In order to properly evaluate the accuracy of the AIM models we will use simulations to compare the AIM models to the iterative case. The analysis of the results will be presented in two parts: First, the behaviour of the individual links will be analysed with an emphasis placed on the analysis of the swing-stick circuit. We will trace the model behaviour from the input/output flows through to the joint displacements and velocities. Secondly we will examine the resulting movement and error of the end effector. This analysis is necessary since the focus of the overall project of which this research is part is orientated to implementing coordinated-mode control.

Four simulations will be used to evaluate the models starting with a trivial case where sinusoidal input voltages are used. This is similiar to the simulations performed in chapter 4.2.2 and 4.2.3 to evaluate the single link case. We will then investigate the accuracy of the AIM model through step voltage and pulse voltage simulations. Finally we will simulate a common task performed by the caterpillar 215B excavator.

5.3.1 SINUSOIDAL VOLTAGES

Figs. 5-12 to 5-15 show the responses for the swing and stick input flows, output flows, input line pressures and output line pressures, respectively, for the sinusoidal input voltages shown in the inset of Fig. 5-12. Note that all input voltages are equal. Figs. 5-16 and 5-17 display the resulting joint displacements and joint velocities for all three links of the excavator.

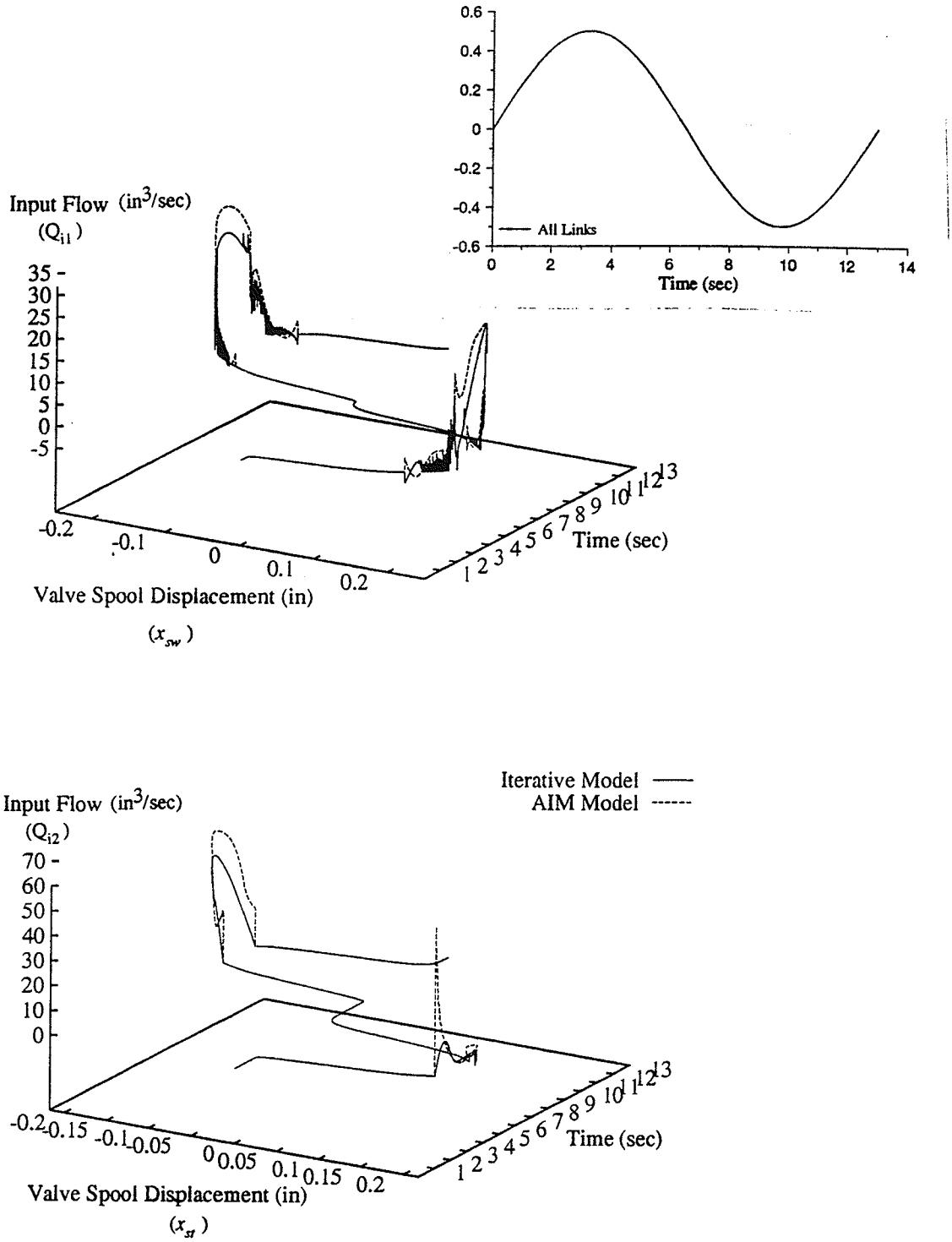
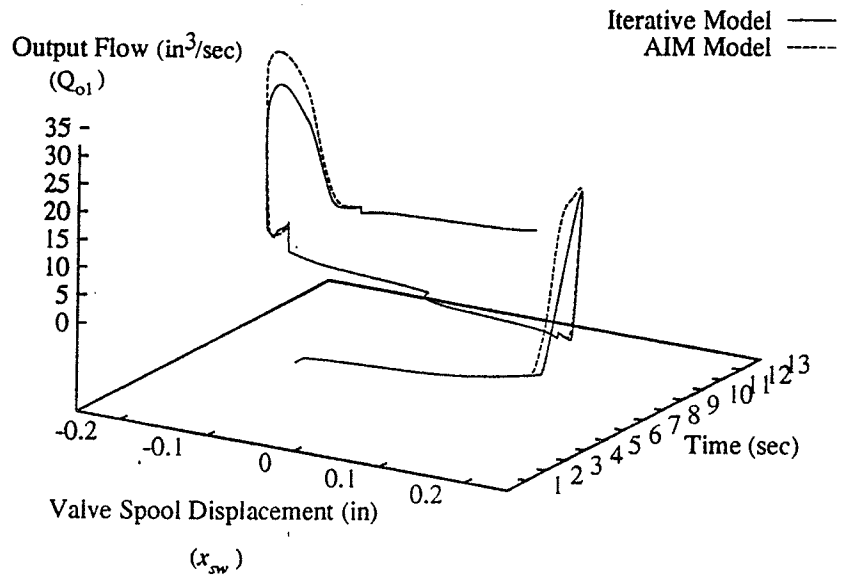


Fig. 5-12 Input Flow Responses to Sinusoidal Inputs for: (a) Swing; (b) Stick.



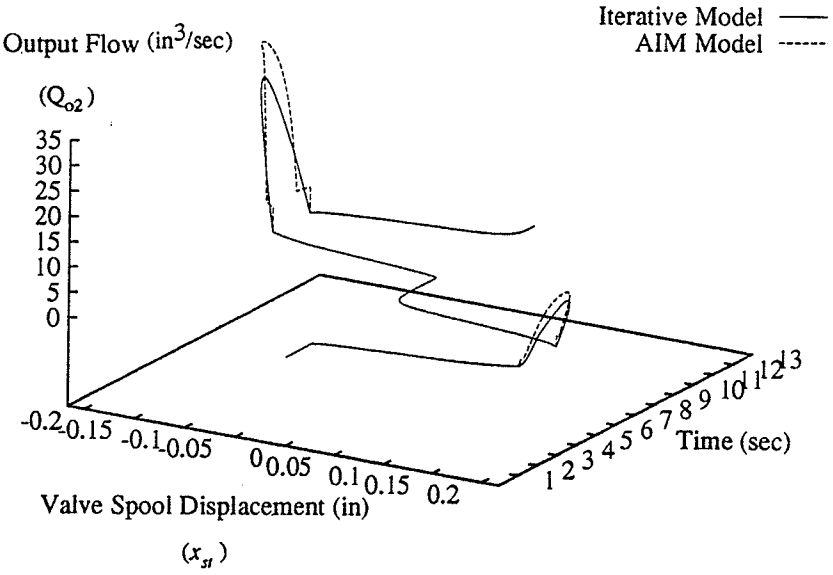


Fig. 5-13 Output Flow Responses to Sinusoidal Inputs for: (a) Swing; (b) Stick.

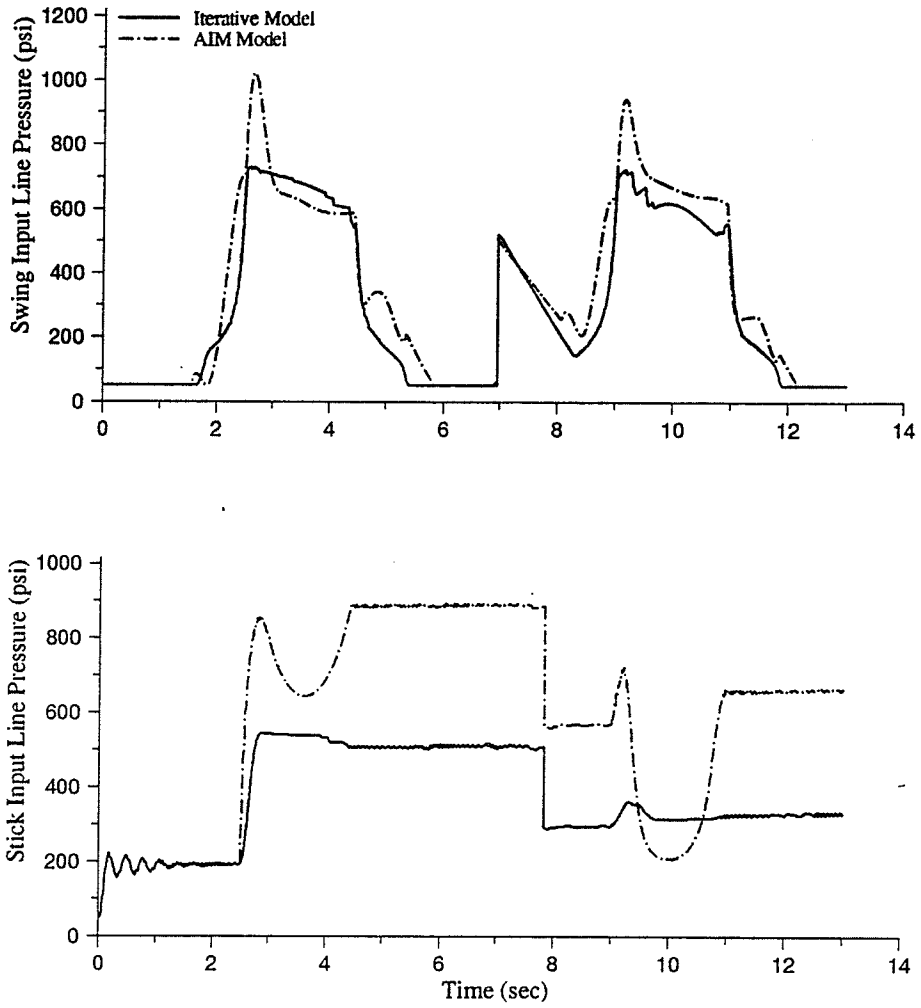


Fig. 5-14 Input Line Pressure Responses to Sinusoidal Inputs for: (a) Swing; (b) Stick.

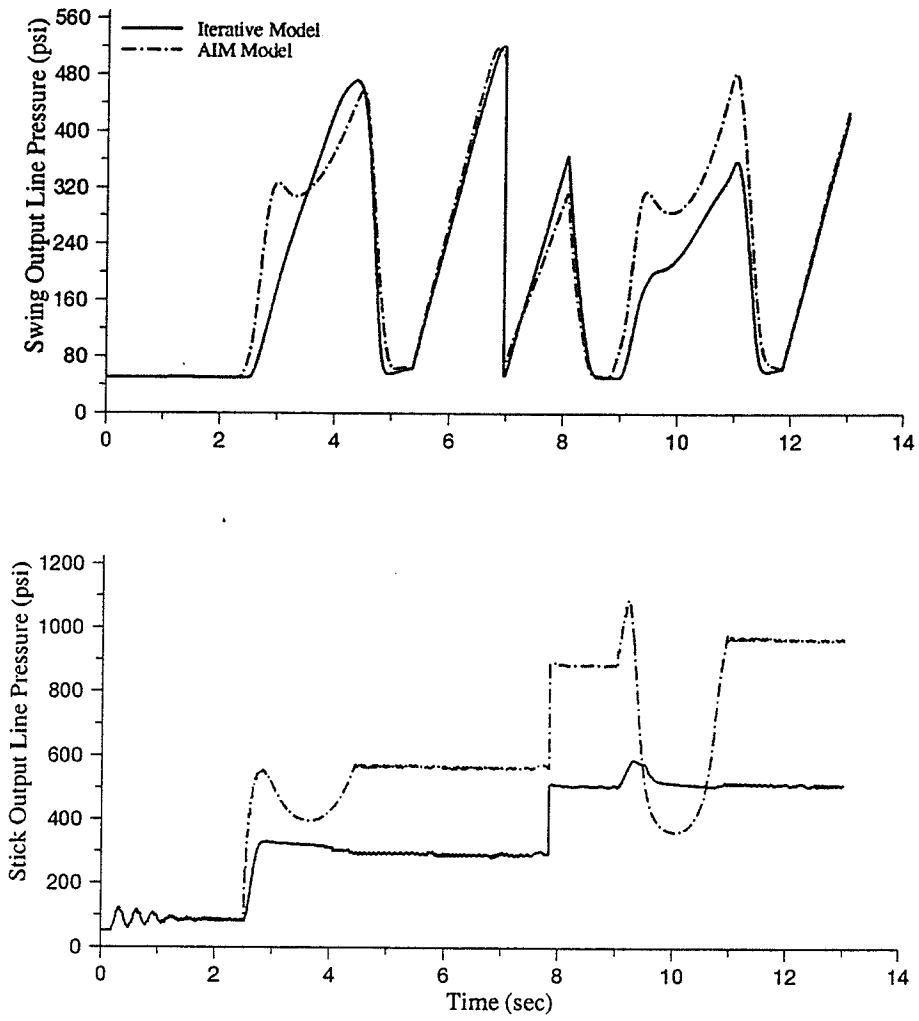


Fig. 5-15 Output Line Pressure Responses to Sinusoidal Inputs for: (a) Swing; (b) Stick.

Figs. 5-12 through 5-15 show the response of the AIM and iterative models of subsystem 2 in our large scale system to a sinusoidal input voltage. The input flows in Fig. 5-12 show that until time t= 2.50*sec* the simulation proceeds with little error. The swing input valve opens first in this simulation and the error introduced is very small. At t = 2.50*sec* the stick valve also opens allowing flow into the stick actuator and the applicable AIM model becomes section **D**. According to Figs. 5-8 and 5-9, we expect that a substantial error will be introduced into the simulation, and the result in Fig. 5-12 indicates this. At this time the value for the swing flow (Q_{i1}) for the AIM model increases to 22.3in^3/sec while the iterative model

only increases to $16.76 in^3/sec$. The result for the stick input flow (Q_{i2}) shows that the AIM model suddenly goes from a value of $0.0 in^3/sec$ to $60.1 in^3/sec$. Inspecting Fig. 5-9(a) at a valve spool displacement (x_{st}) of $0.188 in$ and the input line pressure shown in Fig. 5-14 at time t= $2.5 sec$ we see that this behaviour is expected. The error in the stick input flow quickly decreases, however the error propagates through the system in Figs. 5-13(b), 5-14(b), and 5-15(b). In addition, since the AIM model for both swing and stick in section **D** use all four available inputs, the error will also have an effect on the value for swing input flow in the time steps immediately following the error in the stick input flow.

The simulation proceeds with little error again until time t = $9.005 sec$. At this point the AIM model used again shifts from section **C** to section **D**. Error is again introduced into the simulation for both the swing and stick, however there is no 'spike' for the stick input flow. This is due to the higher input line pressures which, as Figs. 5-8 and 5-9 show, reduce the amount of error in the AIM models.

Fig. 5-13 shows the error in the swing and stick output flows for this simulation. The error in the swing output flow (Q_{oi}) is due to the combination of swing input flow error causing error in the swing line pressures and also to the inherent error in the AIM model which is shown in Fig. 5-11(a). The low output line pressure shown in Fig. 5-15 ensures that the post processed output flow error is low. The error in the stick output flow (Q_{o2}) is entirely due to the error in the input flow model for stick that occurs when the valve is displaced positively. When the valve is displaced negatively, some error is introduced by the stick output flow model error shown in Fig. 5-11(b).

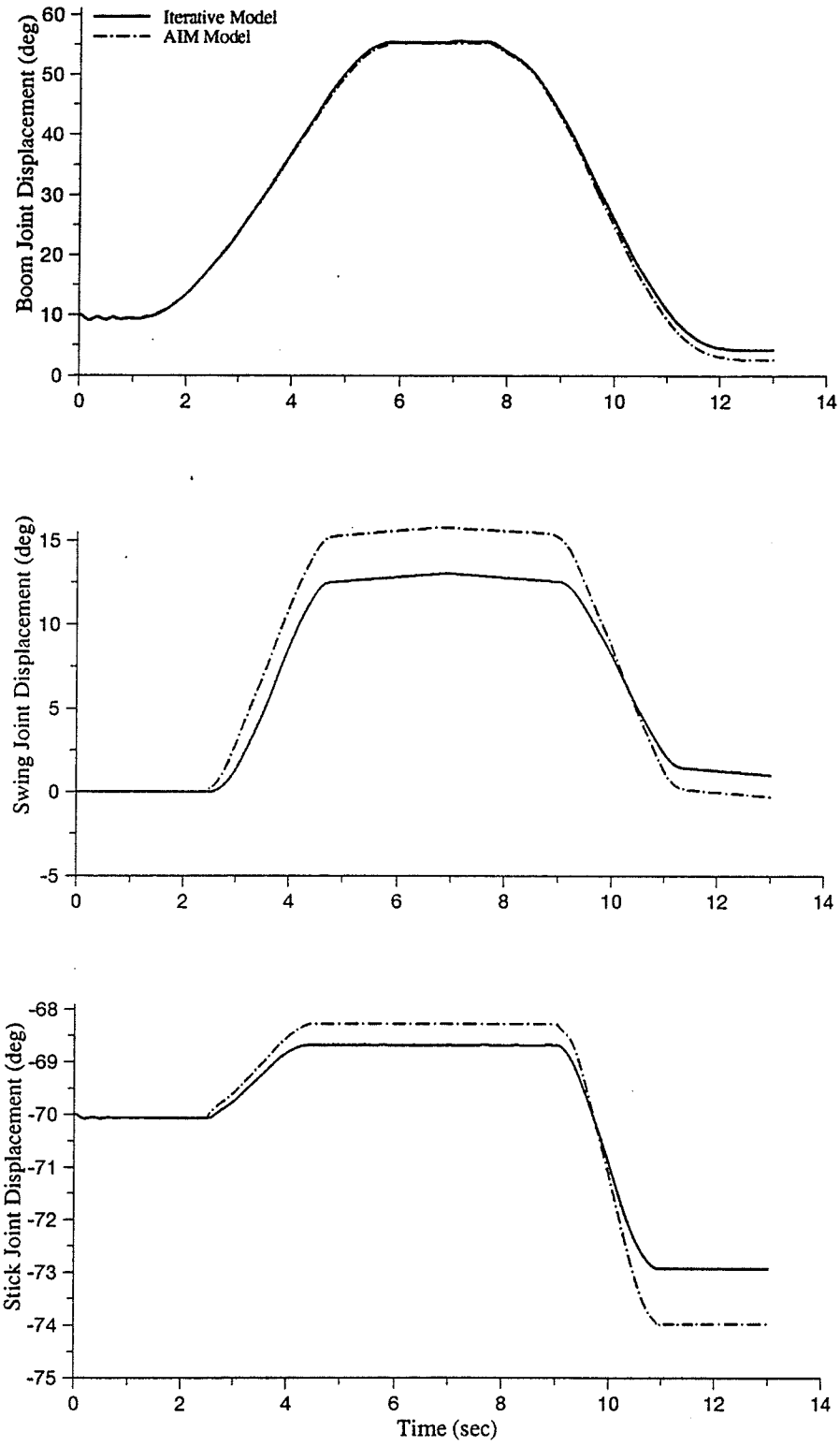


Fig. 5-16 Joint Displacement Responses to Sinusoidal Inputs for: (a) Boom; (b) Swing; (c) Stick

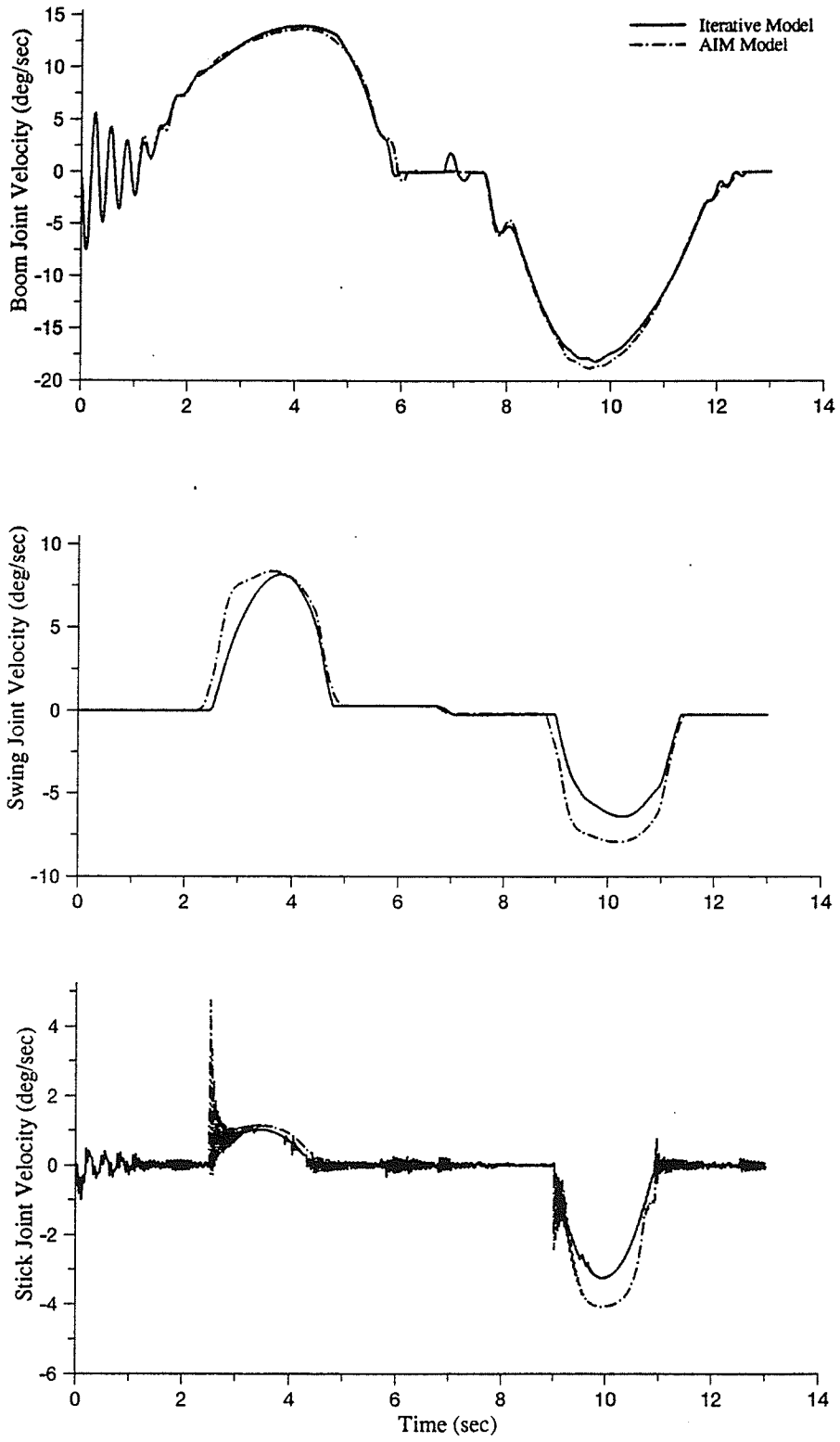


Fig. 5-17 Joint Velocity Responses to Sinusoidal Input for: (a) Boom; (b) Swing; (c) Stick

Figs. 5-16 and 5-17 show the results for the final outputs of the large scale system for this simulation. The boom, joint and stick all suffer from error in their final outputs with the boom error being 1.42*deg* at time t = 13.0*sec*. In the results for the swing and stick we see that when the section **D** AIM model was used, a lag was introduced into the links. Both links show positive AIM overshoot in joint displacement until time t = 10.0*sec* where AIM then undershoots the target value. At time t = 6.5*sec* the AIM error in the swing is 2.76*deg* and the error in the stick is 0.4*deg*. At time t = 13.0*sec* the resultant error is 1.30*deg* for the swing and 1.06*deg* for the stick.

To determine the movement and error in the end effector, forward kinematics is used. Fig. 5-1 which shows the multilink actuation circuit also has an inset of the links of the caterpillar 215B. This inset shows that the relative link joint angles are to be referenced as follows: θ_1 corresponds to the relative joint displacement for the swing, θ_2 corresponds to the relative joint displacement for the boom, and θ_3 corresponds to the relative joint displacement for the stick. Using these values we can determine the end effector location at every time step by solving the following equations. Note that X will determine the X coordinate of the end effector, Y the Y coordinate, and Z the Z coordinate.

The following equations were used to determine the end effector location at each time point.

$$X = (1.8\times\cos(\theta_1)\times\cos(\theta_2+\theta_3)) + (5.19\times\cos(\theta_1)\times\cos(\theta_2)) + (0.81\times\cos(\theta_1)) + (0.12\times\sin(\theta_1))$$

$$Y = (1.8\times\sin(\theta_1)\times\cos(\theta_2+\theta_3)) + (5.19\times\sin(\theta_1)\times\cos(\theta_2)) + (0.81\times\sin(\theta_1)) + (0.12\times\cos(\theta_1))$$

$$Z = (1.8\times\sin(\theta_2+\theta_3)) + (5.19\times\sin(\theta_2))$$

The value 0.81 used in the equations is the X direction offset (in meters) of the boom link origin from the absolute origin, 0.12 is the Y direction offset (in meters). The value 5.190 is the length of the boom link in meters and 1.80 is the length of the stick link in meters.

The equations for X, Y, and Z show that the most prominent links in the derivation of the end effector location is the boom and swing. The joint displacement for the stick, θ_3, is used in all three directions as is the boom (θ_2) and swing (θ_1), however it is only used in conjunction with the boom displacement and therefore plays a secondary role. In addition, the boom is of longer length so it will have a larger effect on the final end effector placement.

The accuracy of the boom model was proven in chapter 4 and the accuracy for the swing models for the various AIM model matrix sections was shown in section 5.2. Therefore, even though the graphs in section 5.2 indicate a poor modelling of the stick in the region where both the swing and stick are active, the effect of this error will be minimized in the end effector movement.

Fig. 5-18 shows the movement of the end effector during the simulation. The end effector error is displayed in Fig. 5-19. Note that the overall end effector error (the distance vector) indicates the magnitude of the error only and not the direction of the error vector. It was calculated as follows:

$$\text{Distance Vector} = \overline{V} = \sqrt{X^2 + Y^2 + Z^2}$$

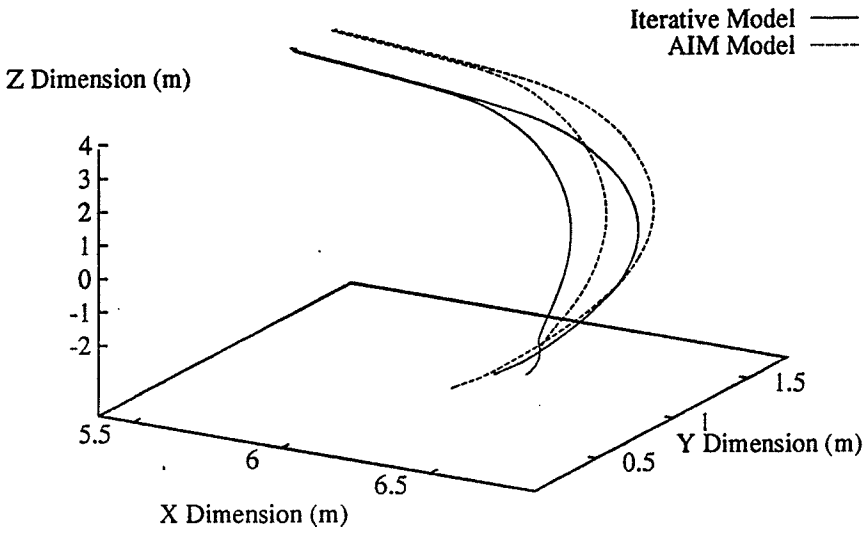


Fig. 5-18 End Effector Response to Sinusoidal Input.

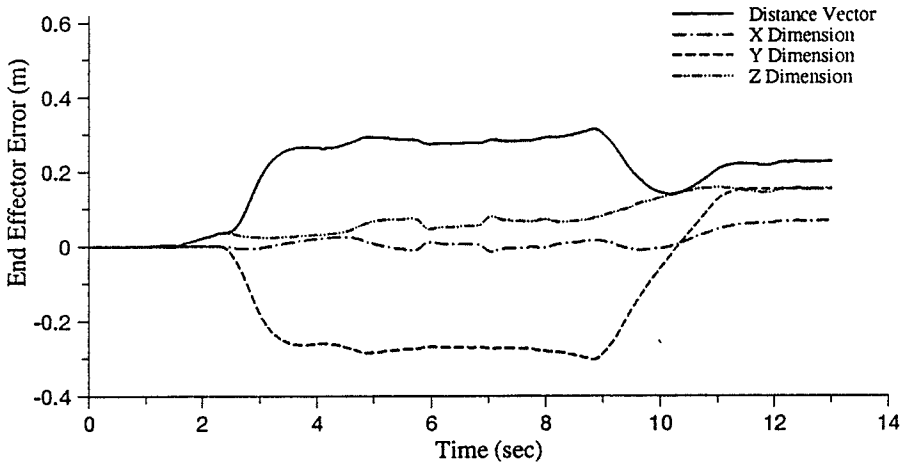


Fig. 5-19 End Effector Error for Sinusoidal Inputs.

As Fig. 5-19 shows, the absolute maximum error for the end effector in this simulation was 0.313m at time t = 8.85*sec*. The majority of the error in this simulation occured in the Y dimension, which is the extension and retraction of the boom-stick linkage. The fact that the boom was stable throughout this simulation kept the error in the X and Z dimensions small,

with both being under 0.1m for the entire simulation. The swing error in model section **D** is the primary cause of the error in the Y dimension.

5.3.2 STEP INPUT

Fig. 5-20 shows the input flows for the swing and stick for both the conventional iterative-based and the AIM-based simulation models. The applied input voltages were 0.3*volt*, 0.4*volt* and 0.7*volt* for the boom, swing and stick, respectively. Fig. 5-21 shows the corresponding output flows while Figs. 5-22 through 5-25 show the input line pressures, output line pressures, joint displacements and joint velocities, respectively.

The primary occurance of interest in this simulation is that the stick hits its physical end stop at -28*deg* at time t = 5.0*sec*. The effect this has on the swing is visible in all the Figs. Since the stick has hit its end stop the swing motion becomes faster, since more fluid is diverted to it. The boom is unaffected by the stick reaching its physical limit since it is not connected to the swing-stick system.

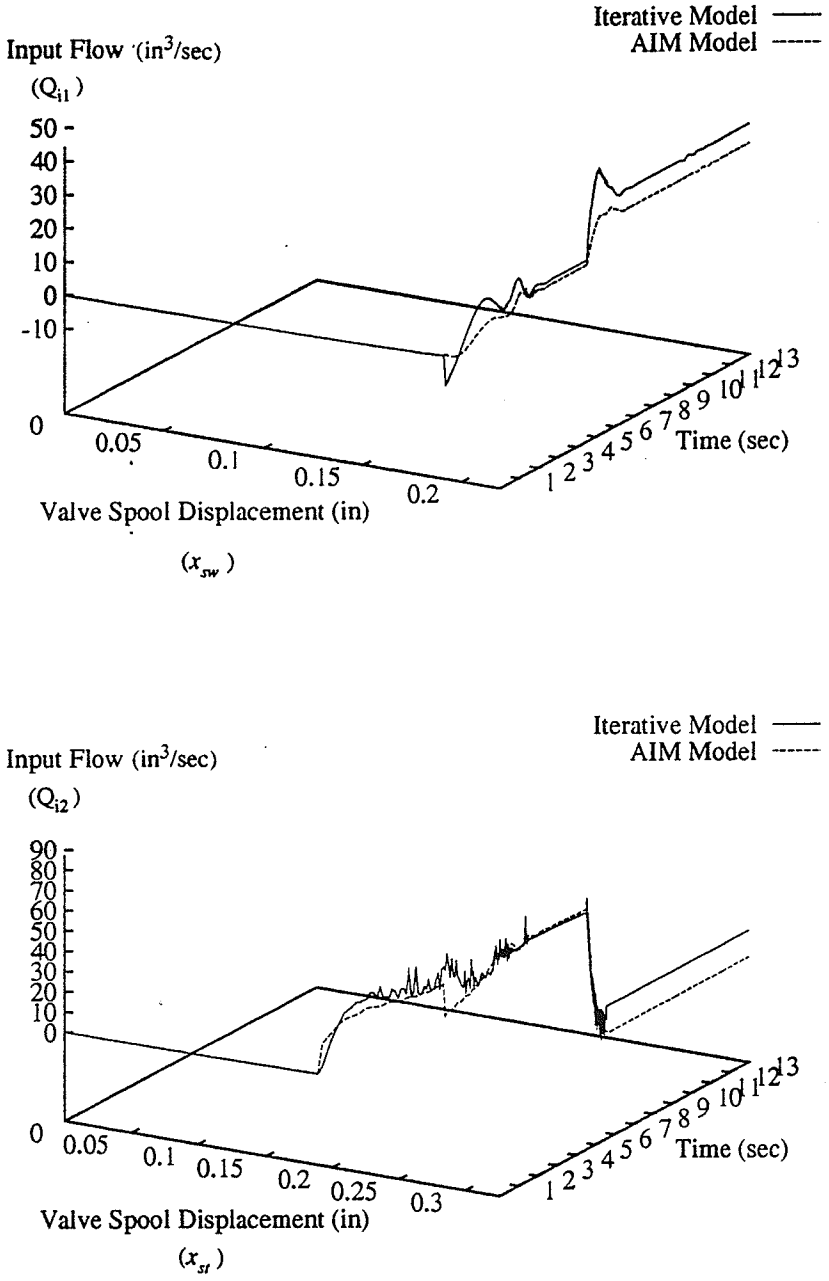
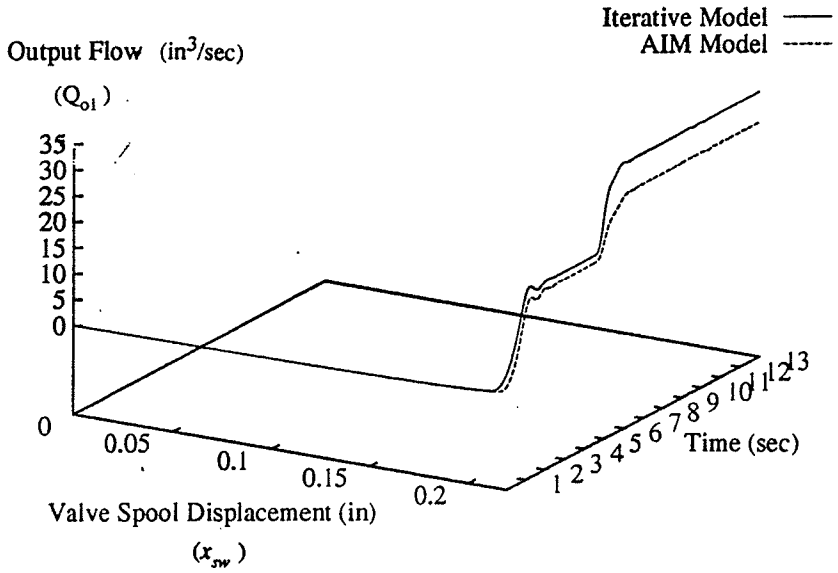


Fig. 5-20 Input Flow Responses to Step Inputs for: (a) Swing; (b) Stick.



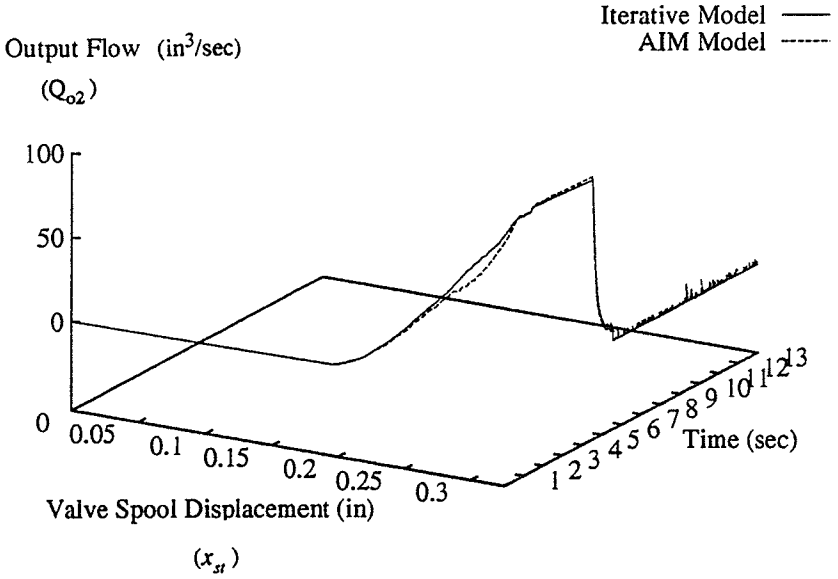


Fig. 5-21 Output Flow Responses to Step Inputs for: (a) Swing; (b) Stick.

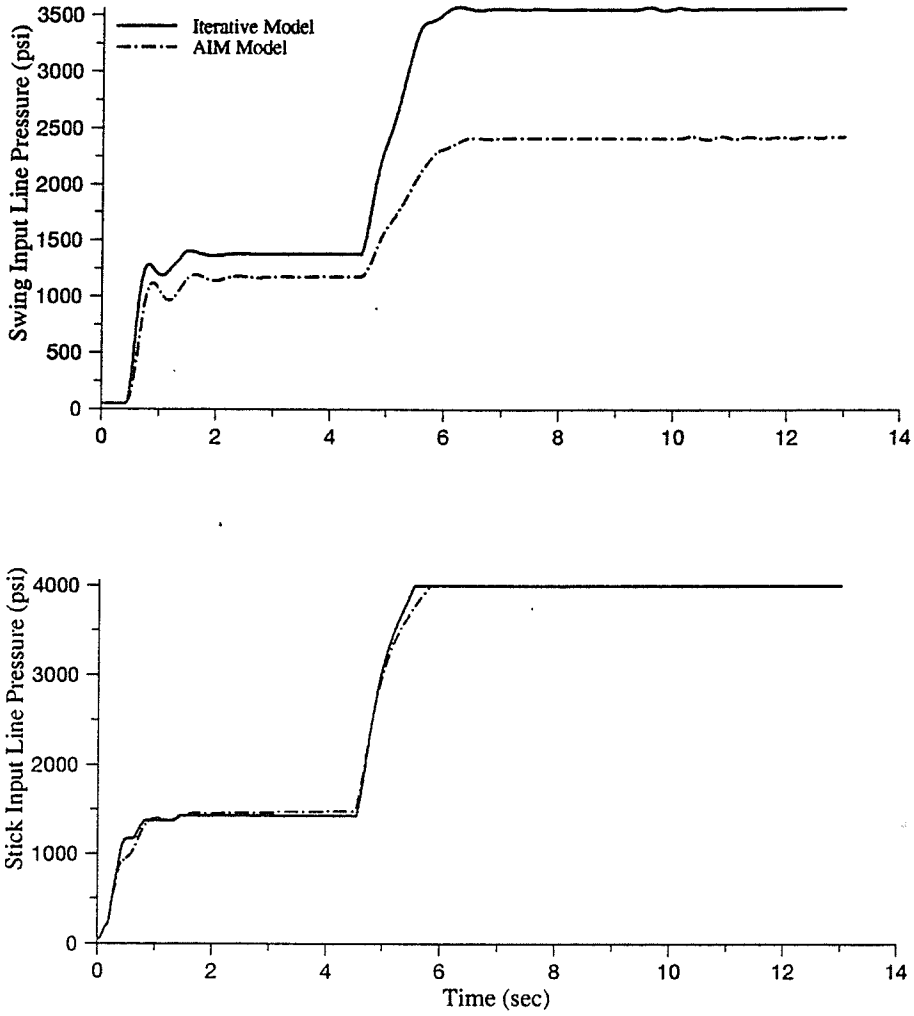


Fig. 5-22 Input Line Pressure Responses to Step Inputs for: (a) Swing; (b) Stick.

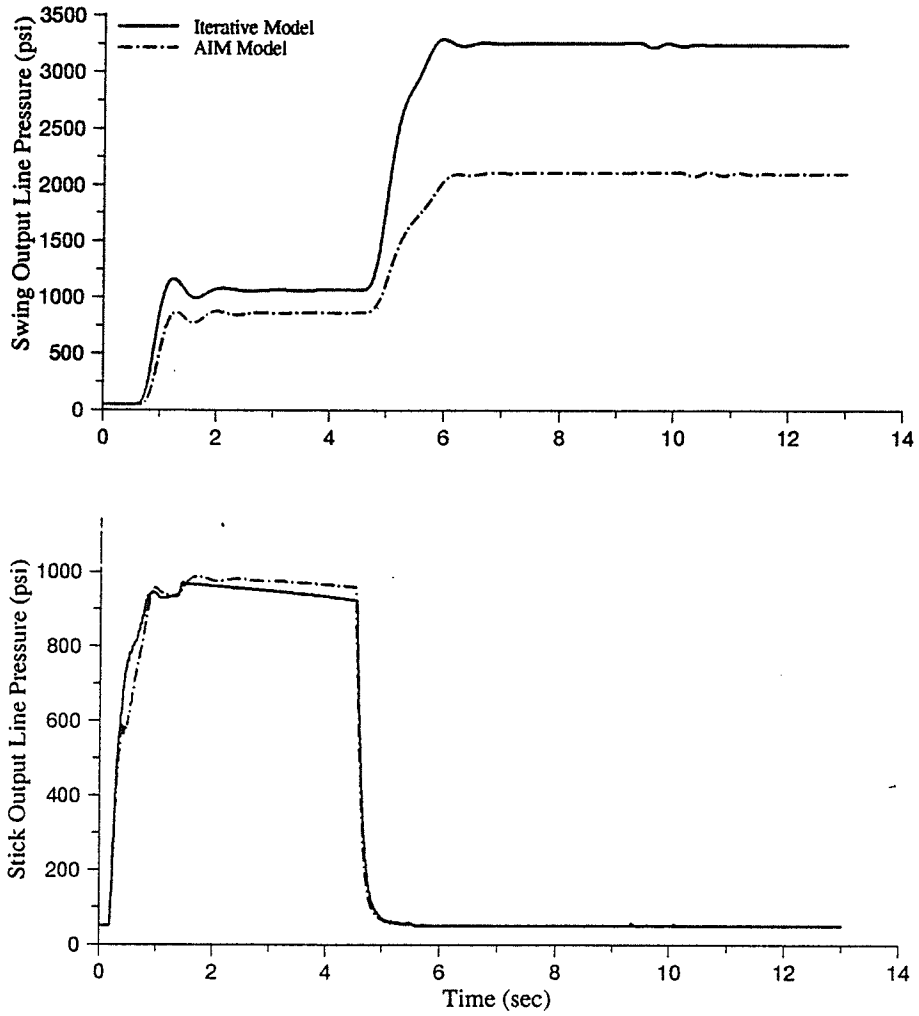


Fig. 5-23 Output Line Pressure Responses to Step Inputs for: (a) Swing; (b) Stick.

In Fig. 5-20 the stick input valve opens first at time t = 0.195*sec*, the stick input line pressure is low, approximately 175*psi*. The error graph shown in Fig. 5-6 for section **B** of the model matrix predicts a large amount of error in this range. However, the swing input valve opens shortly thereafter at time t = 0.395*sec* so there is little time for error to accumulate. When the swing input valve opens the stick valve spool displacement (x_{st}) is 0.279*in* with an input line pressure of 1430*psi*. Fig. 5-8(a) and (b) both show that there will be little swing error at this point, and since the stick was already at 0.279*in* when the swing passed its deadband, there is

minimal error in the stick. The simulation will use model section **D** for the rest of the simulation, however the results are quite good if one discounts the effects of the stick hitting its physical end stop.

The maximum flow errors for both swing and stick occur at the end of the simulation with the error in the swing input flow (Q_{i1}) equal to $5.82 in^3/sec$ and the error for the stick input flow (Q_{i2}) equal to $13.14 in^3/sec$. The major cause of this error in the swing, as stated previously, is the stick hitting its end stop. This increases the amount of flow to the swing in the iterative model but does not in the AIM model since the model was not trained with these types of nonlinearities occuring in the training database.

The effect the stick link reaching its physical end stop has on the joint displacements and velocities is shown in Figs. 5-24 and 5-25. The error in the stick is negligible while there is a substantial cumulative lag in the swing link which reaches its maximum at $13.0 sec$ with an error value of $15.44 deg$. The error in the boom is due to the simulation voltage causing the boom valve spool displacement (x) to remain in the region $0.210 in$ to $0.30 in$ for the entire simulation. As we saw in Fig. 4-11(a), this is the area of maximum error for this AIM model.

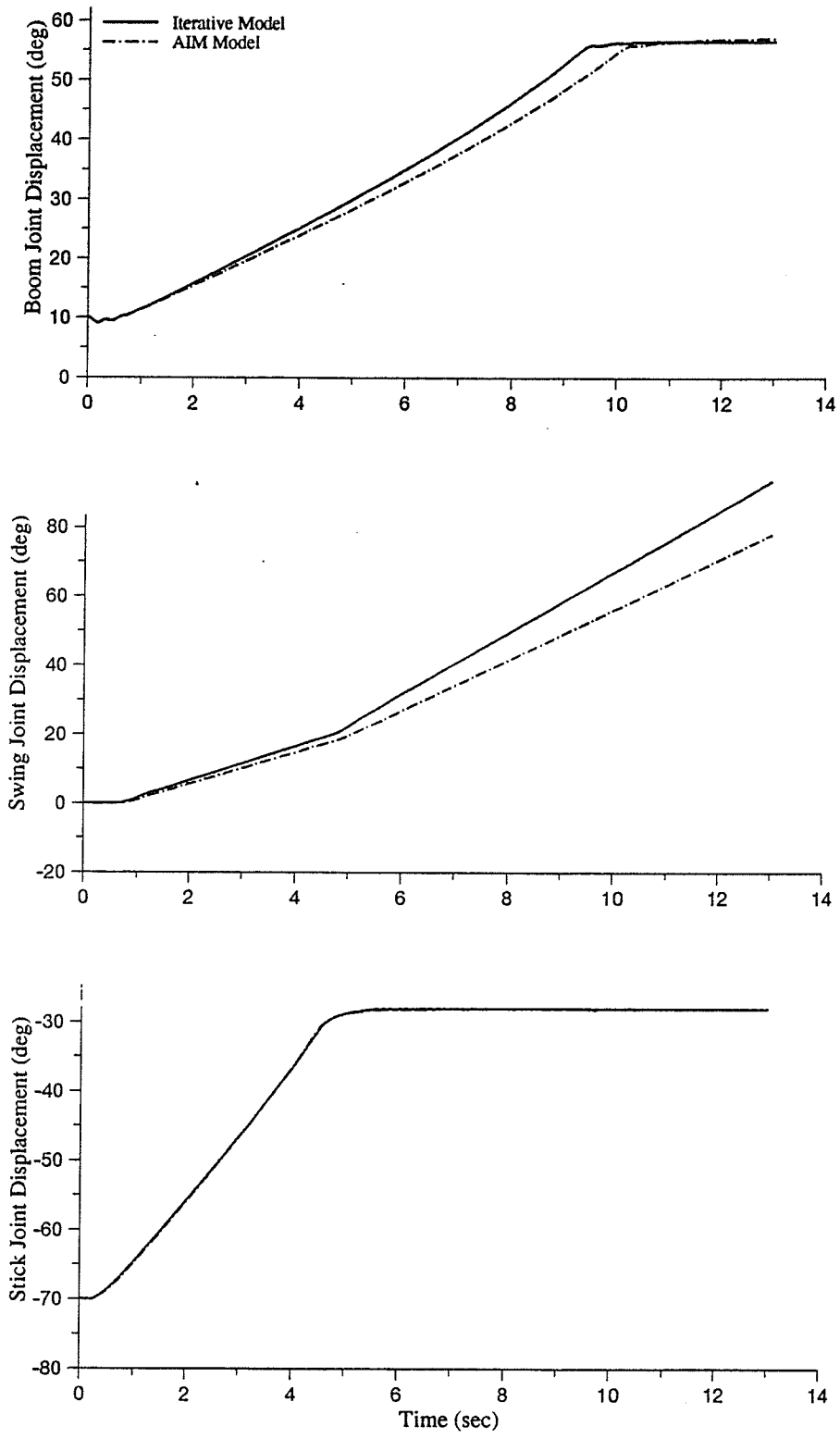


Fig. 5-24 Joint Displacement Response to Step Inputs for: (a) Boom; (b) Swing; (c) Stick.

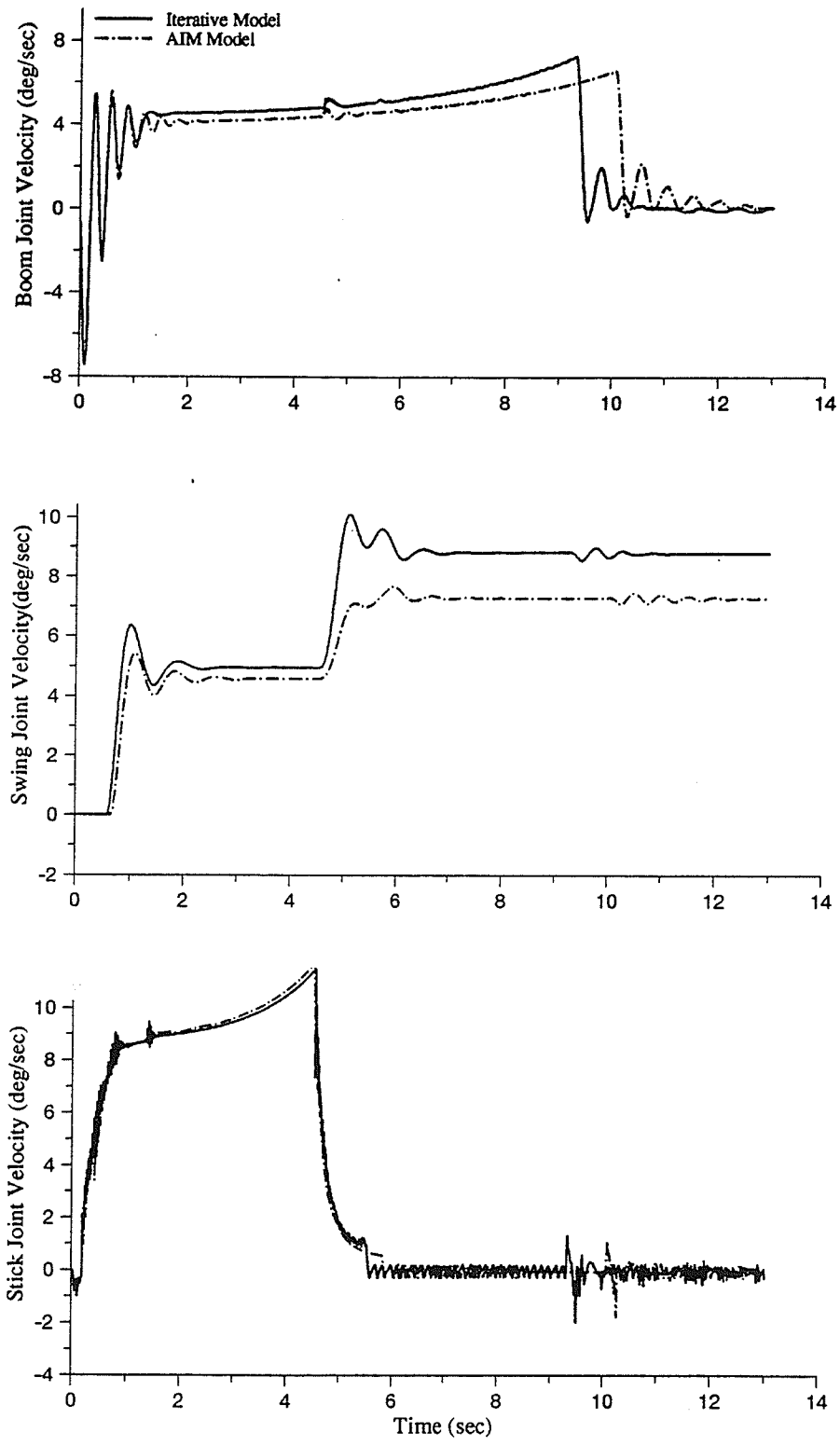


Fig. 5-25 Joint Velocity Responses to Step Inputs for: (a) Boom; (b) Swing; (c) Stick.

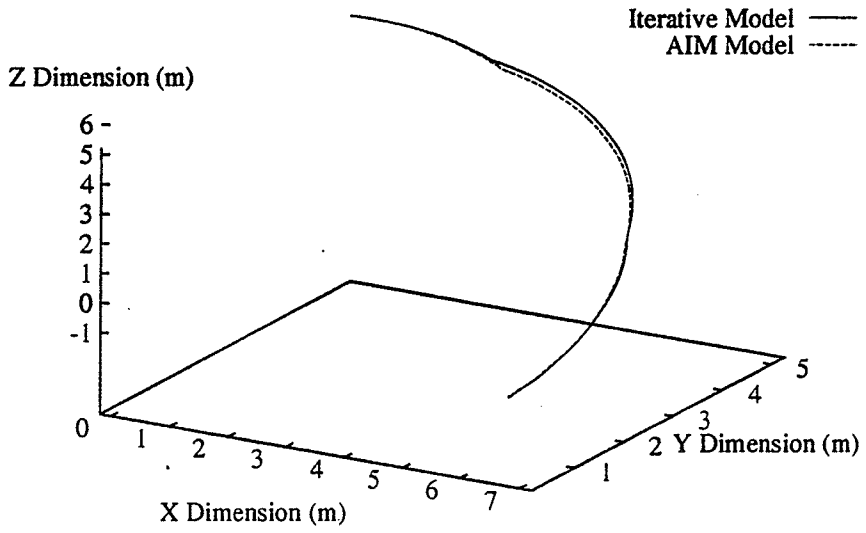


Fig. 5-26 End Effector Response to Step Inputs.

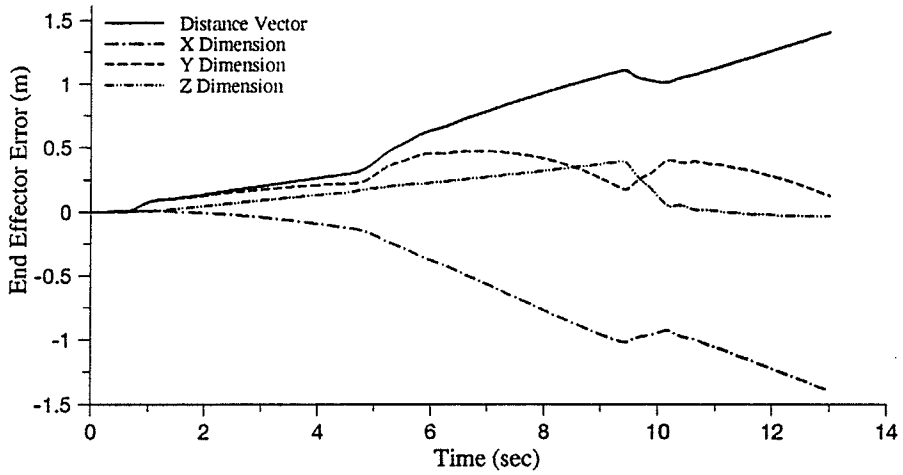


Fig. 5-27 End Effector Error for Step Inputs.

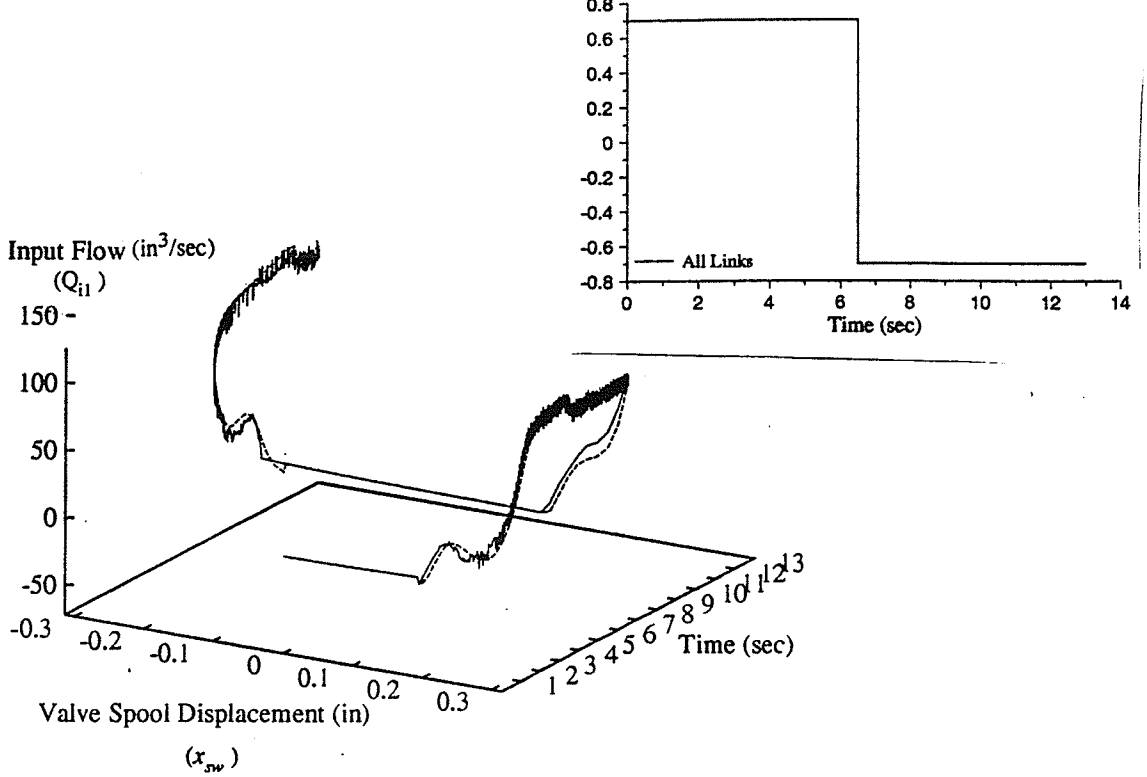
Figs. 5-26 and 5-27 show the end effector movement and the end effector error for this simulation. The absolute maximum error in the end effector is 1.4*m* at t = 13.0*sec*. The majority of the error in the simulation occurs in the X dimension with its error at 13.0*sec* being equal to -1.397*m*. The effect of the swing error as well as the boom error is most

prevalent in the X dimension as we would expect since the swing joint displacement corresponds to θ_1 , and the boom joint displacement corresponds to θ_2.

5.3.3 PULSE VOLTAGE

Figs. 5-28 and 5-29 show the input and output flows for the swing and stick when the pulse voltage shown in the inset of Fig. 5-28 is used for all the links. The resulting input and output line pressures are shown in Figs. 5-30 and 5-31. Figs. 5-32 and 5-33 show the joint displacements and joint velocities for the applied voltages.

This simulation shows the best results of all the simulations so far. Neither the swing nor the stick encounter any mechanical nonlinearities (i.e. reaching end stops) during the simulation. The boom does hit its physical end stop but since it is not coupled to the swing - stick system in any way there is no effect on these links.



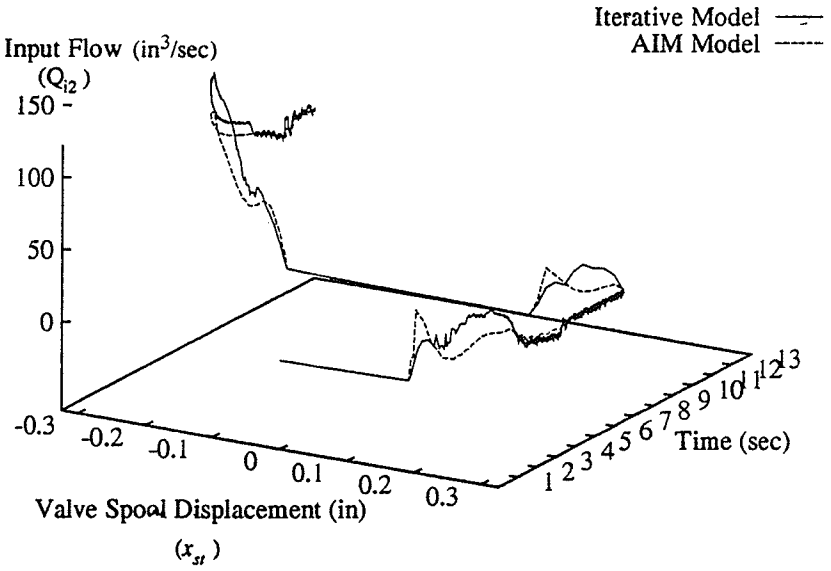
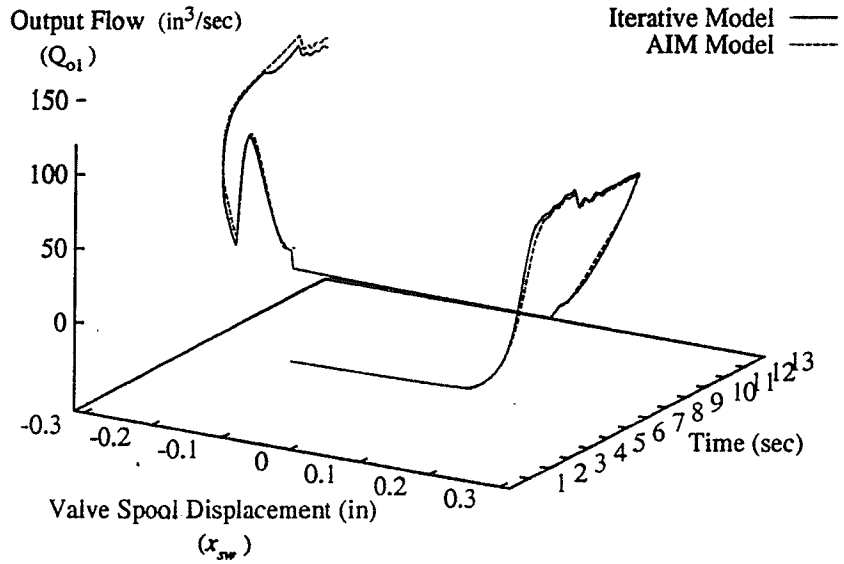


Fig. 5-28 Input Flow Responses to Pulse Inputs for: (a) Swing; (b) Stick.



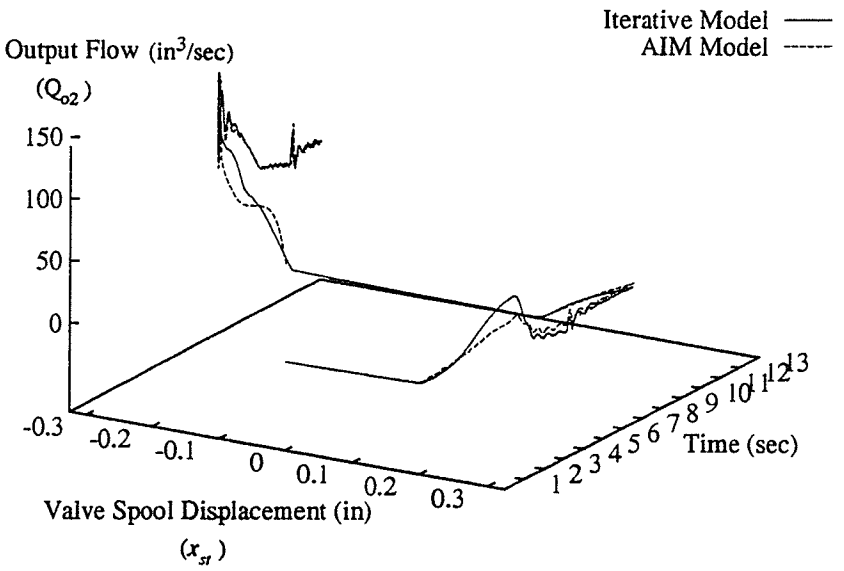


Fig. 5-29 Output Flow Responses to Pulse Inputs for: (a) Swing; (b) Stick.

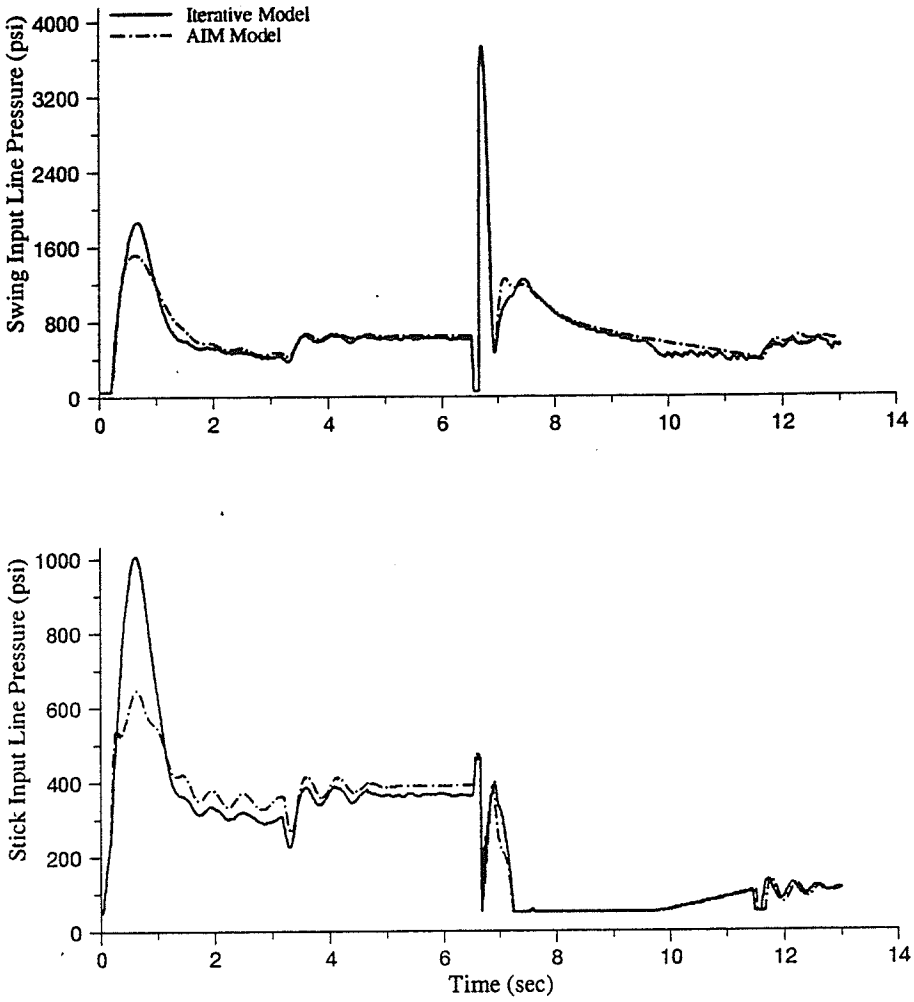


Fig 5-30 Input Line Pressure Responses to Pulse Inputs for: (a) Swing; (b) Stick.

Fig. 5-28 shows the input flows for both the swing and the stick. The stick opens first, at time $t = 0.18sec$ and the swing opens immediately afterward at time $t = 0.19sec$. There is a small spike in the stick AIM model which is predicted in Fig. 5-9 which shows the AIM model error for model section **D**. The AIM model value for the stick input flow at this point is $14.87in^3/sec$, while the iterative model has a value of $3.87in^3/sec$. The simulation proceeds and as the line pressure stabilizes for both the swing and stick we see that the input flow errors become negligible until the valve spool displacements approach the deadband zone.

Figs. 5-29 through 5-31 show the propagation of the input flow error through the line pressures and the output flows.

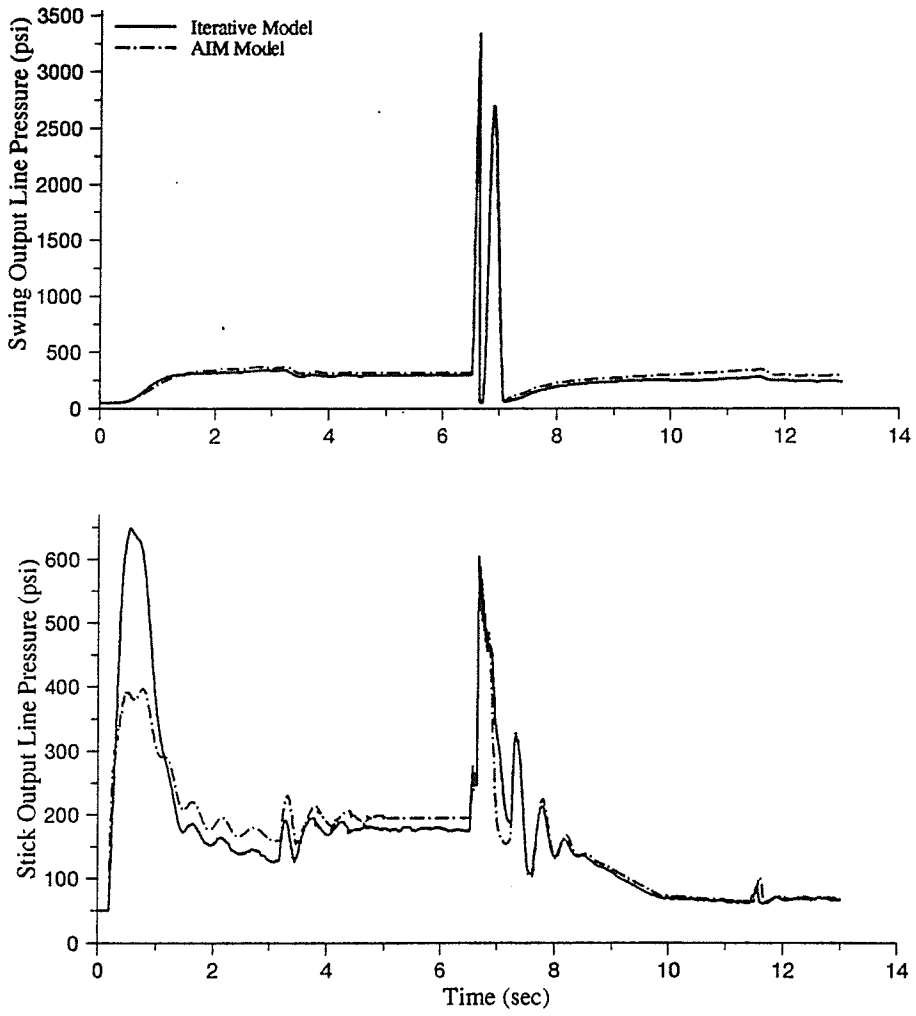


Fig. 5-31 Output Line Pressure Responses to Pulse Inputs for: (a) Swing; (b) Stick.

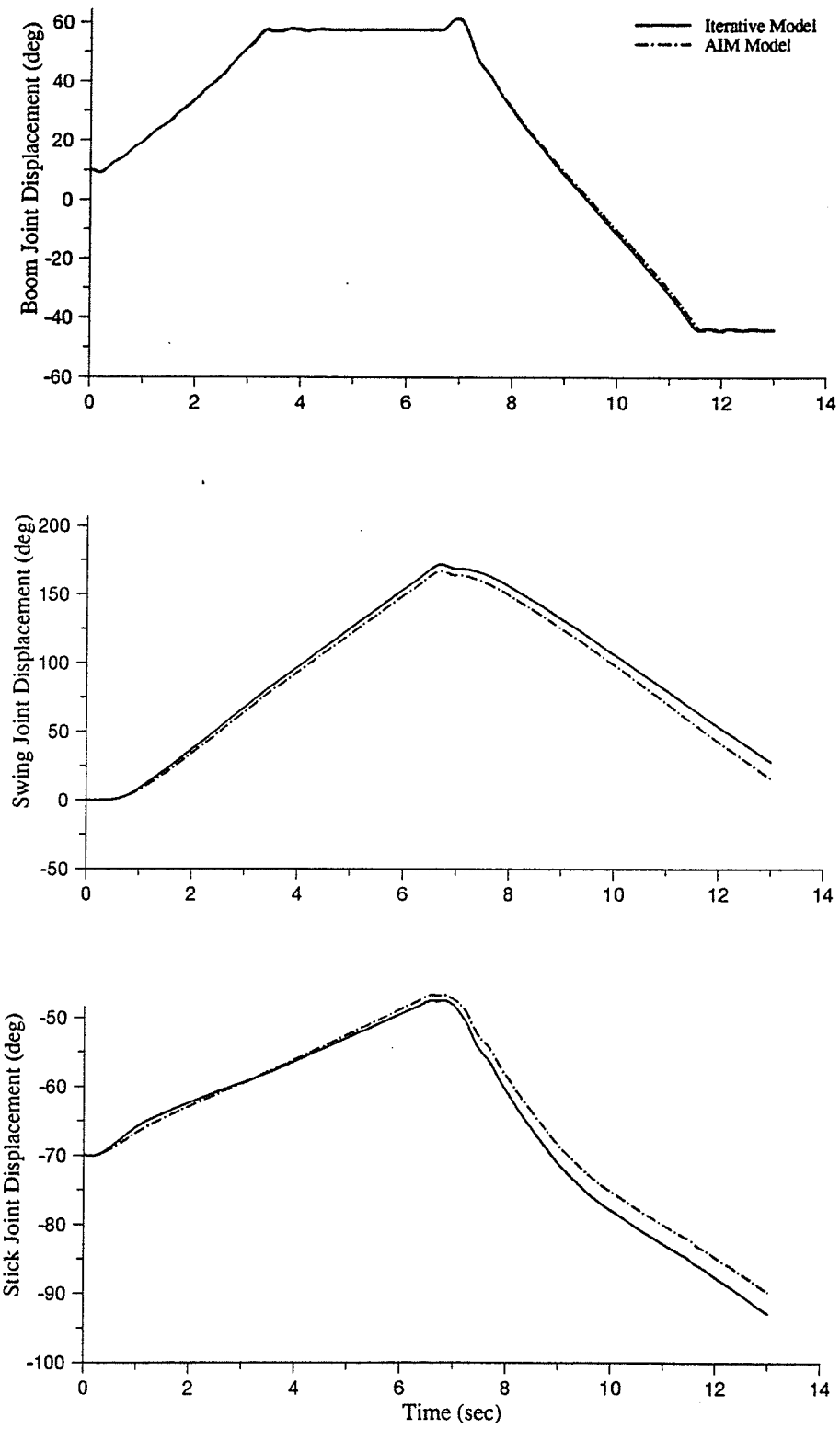


Fig. 5-32 Joint Displacement Responses to Pulse Inputs for: (a) Boom; (b) Swing; (c) Stick.

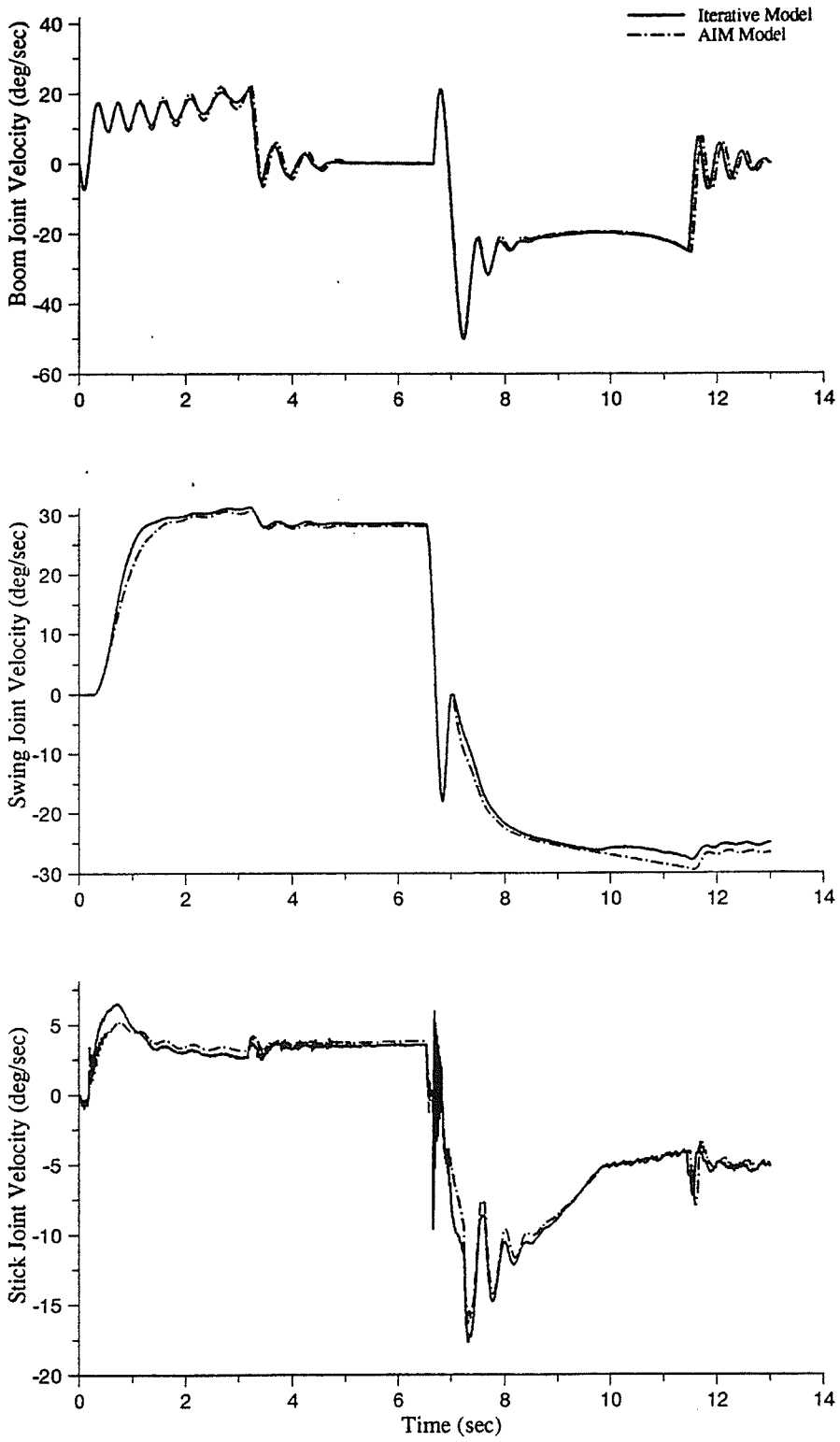


Fig. 5-33 Joint Velocity Responses to Pulse Inputs for: (a) Boom; (b) Swing; (c) Stick.

Figs. 5-32 and 5-33 show the joint displacements and the joint velocities of all the links in the simulation. We see that the amount of error in the links is very small with a culmulative lag occuring in the swing and stick. At time t = 13.0*sec* the error in the swing is 12.3*deg* and the error in the stick is 3.0 *deg*. The error in the boom is negligible.

The effects of these link errors on the placement of the end effector is shown in Figs. 5-34 and 5-35. The absolute maximum end effector error in this simulation occurs at approximately t = 8.0*sec* and is 0.867*m*. The error in the Z dimension is negligible while the error for X increases only after the valve spool displacements have opened in the negative direction. This corresponds to the increase in positve lag in the stick joint displacement shown in Fig 5-33(c). This lag in the stick combined with the small error in the boom joint displacement is also responsible for the movement of Y dimenstion error from negative to positive. The final error in the end effector at the end of the simulation is 0.688*m*.

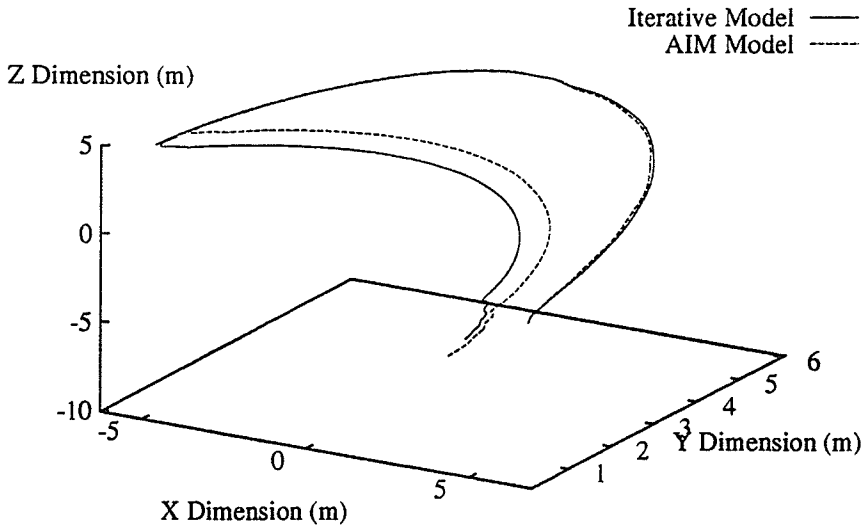


Fig. 5-34 End Effector Response to Pulse Inputs

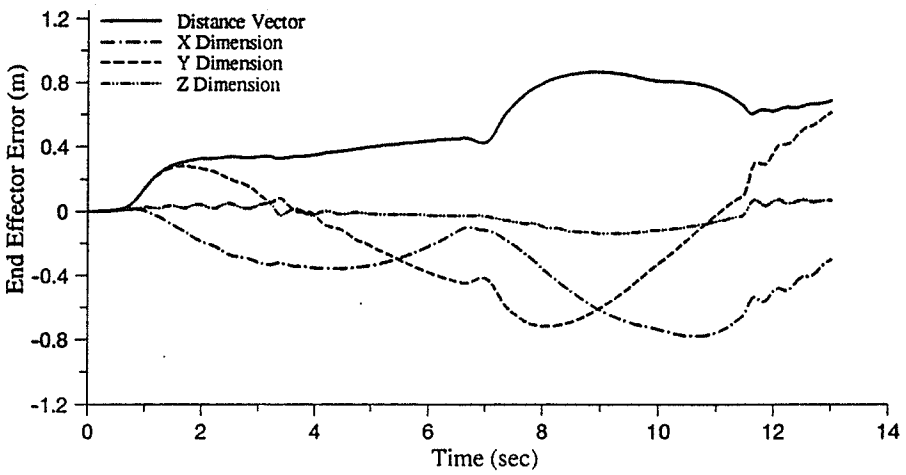


Fig. 5-35 End Effector Error for Pulse Inputs.

5.3.4 SIMULATED PICK/PLACE TASK

In addition to the sinusoidal, step and pulse voltage input simulations, we attempted to model a scenario that would typically occur for such a machine in the real world. To accomplish this we set the voltages to occur as in Fig. 5-36 The ramp voltages mimic an excavator task of picking and placing an object [21]. Referring to Fig. 5-36, the excavator simultaneously swings and lifts the boom for 3.0*sec*, then the boom and stick depress for 4.0*sec* to accomplish the task - digging for example. This is followed by the stick and boom rising for 3.0*sec*. Finally, the last 3.0*sec* has all of the links activated to move the implement to its final destination. Figs. 5-37 through 5-42 again show the input flows, output flows, input line pressures, output line pressures, joint displacements and joint velocities, respectively, for this simulation.

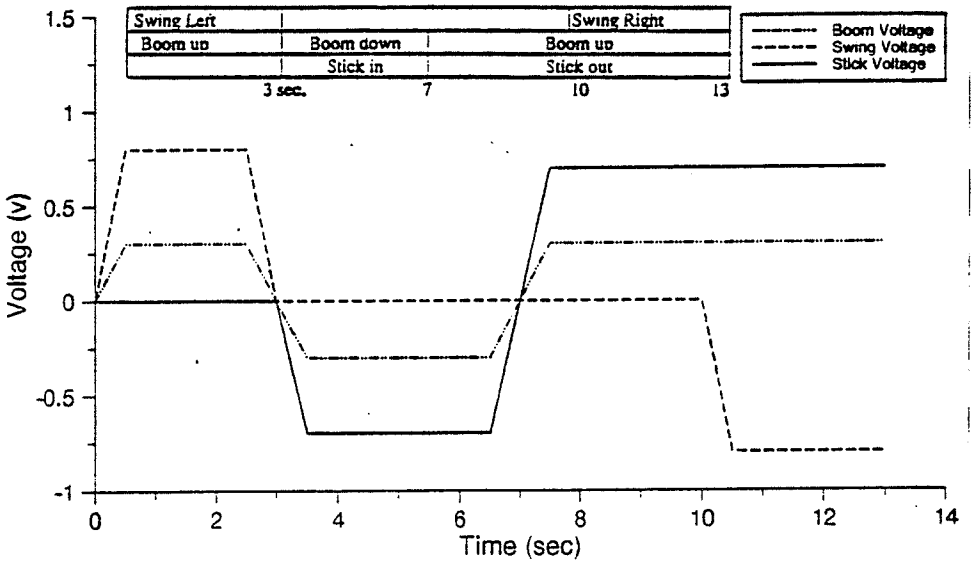


Fig. 5-36 Simulation Voltages for Excavator Task.

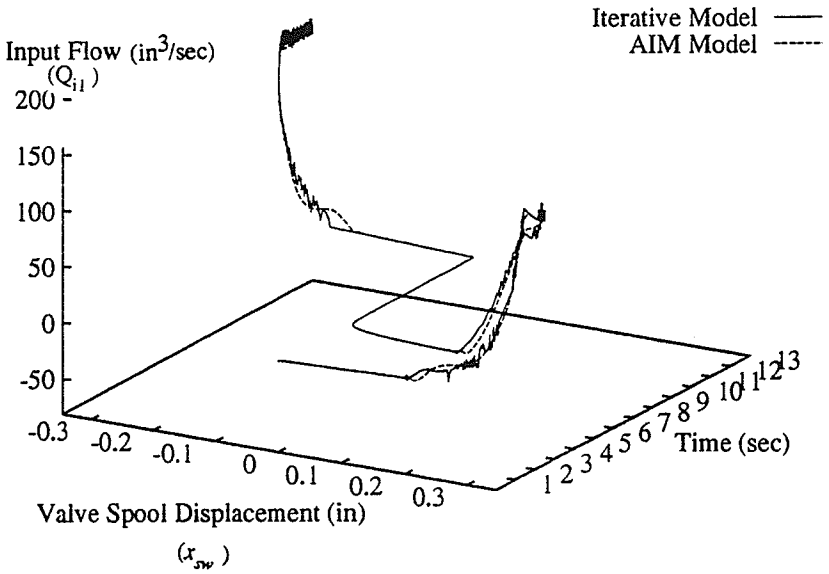


Fig. 5-37 Task Simulation Input Flows for: (a) Swing; (b) Stick.

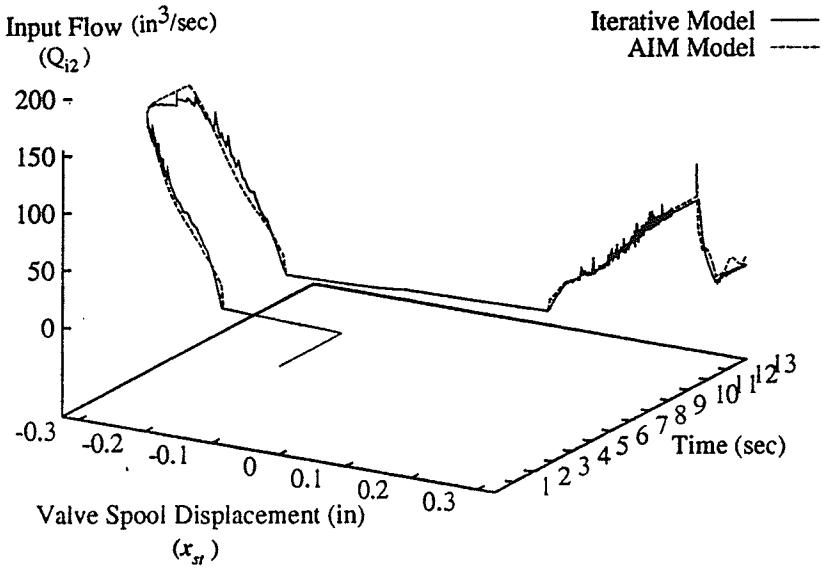


Fig. 5-37 (cont'd).

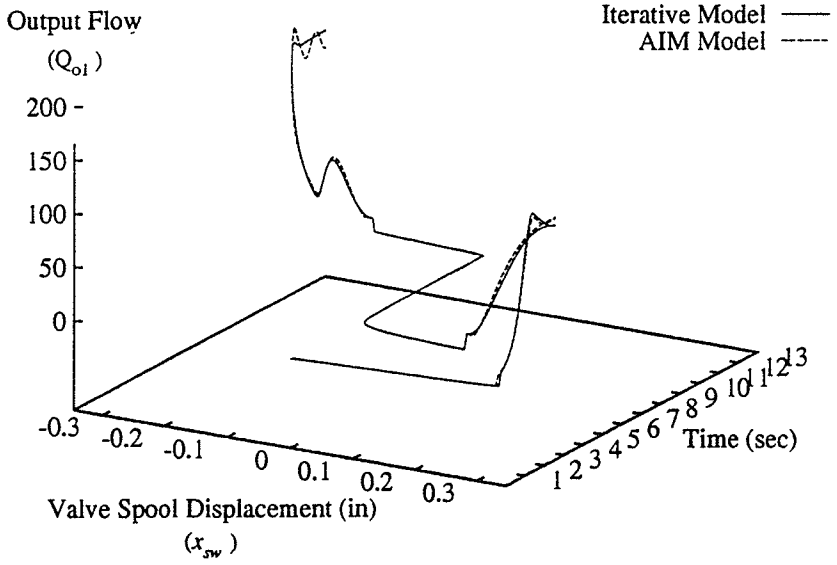


Fig. 5-38 Task Simulation Output Flows for: (a) Swing; (b) Stick.

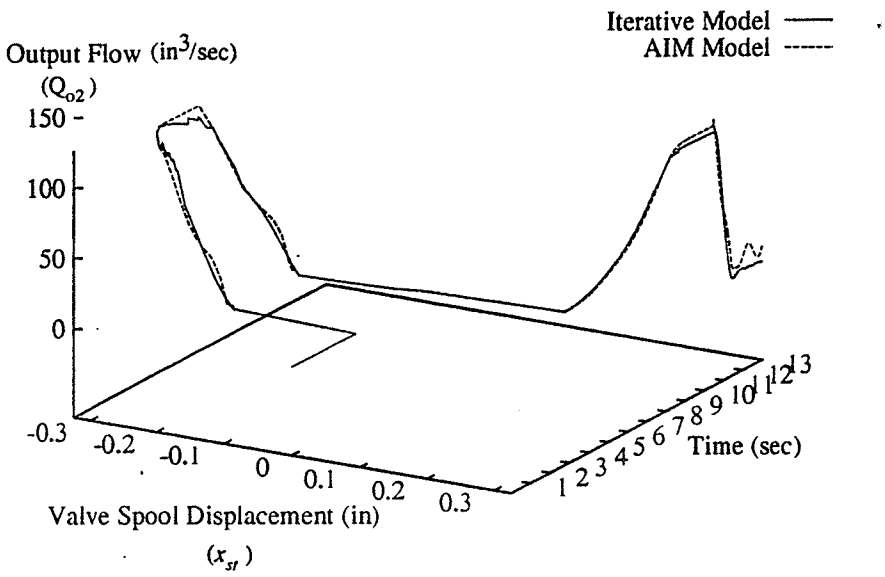


Fig. 5-38 (cont'd).

As Fig. 5-37 shows, the swing valve opens at time t = 0.41*sec* and closes again at time t = 2.95*sec*. The stick is inactive in this time period, so the swing AIM model used is section **C** of our matrix model. As Fig. 5-7(a) shows, there is negligible error for the swing in this range. At time t = 3.515*sec* flow begins for the stick actuator and ceases at time t = 6.81*sec*. The swing is inactive in this period so the AIM model section used is section **B**. We would expect to see a large amount of error in our simulation at this point according to Fig. 5-6, and indeed there is error present. However the stick valve spool displacement quickly moves past this region and into a region of minimal error where AIM model overshoot is $\pm 10 in^3/sec$. At time t = 7.525*sec* the swing again opens and since the input line pressure is high, as shown in Fig. 5-39(b) the amount of error will be negligible. The model for stick will stay in section **B** until time t = 10.41*sec*, at which time the swing valve opens and the applicable AIM model becomes section **D**. Since the input line pressures, shown in Fig. 5-39, are approximately 2000*psi* for both swing and stick, there is minimal error introduced.

The error in the output flows closely mirrors the error in the input flows. This indicates that the output flow errors for both the swing and the stick are due to the propagation of the input flow error through the line pressures and then affects the output flows.

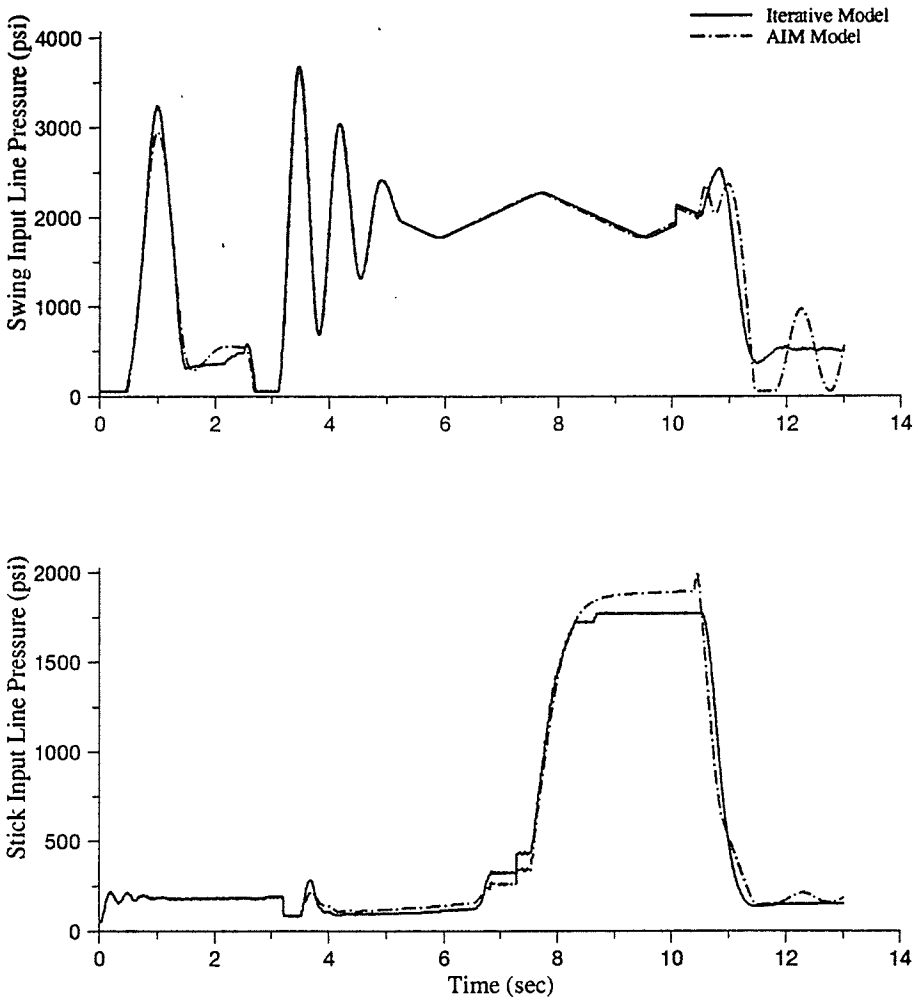


Fig. 5-39 Task Simulation Input Line Pressures for: (a) Swing; (b) Stick.

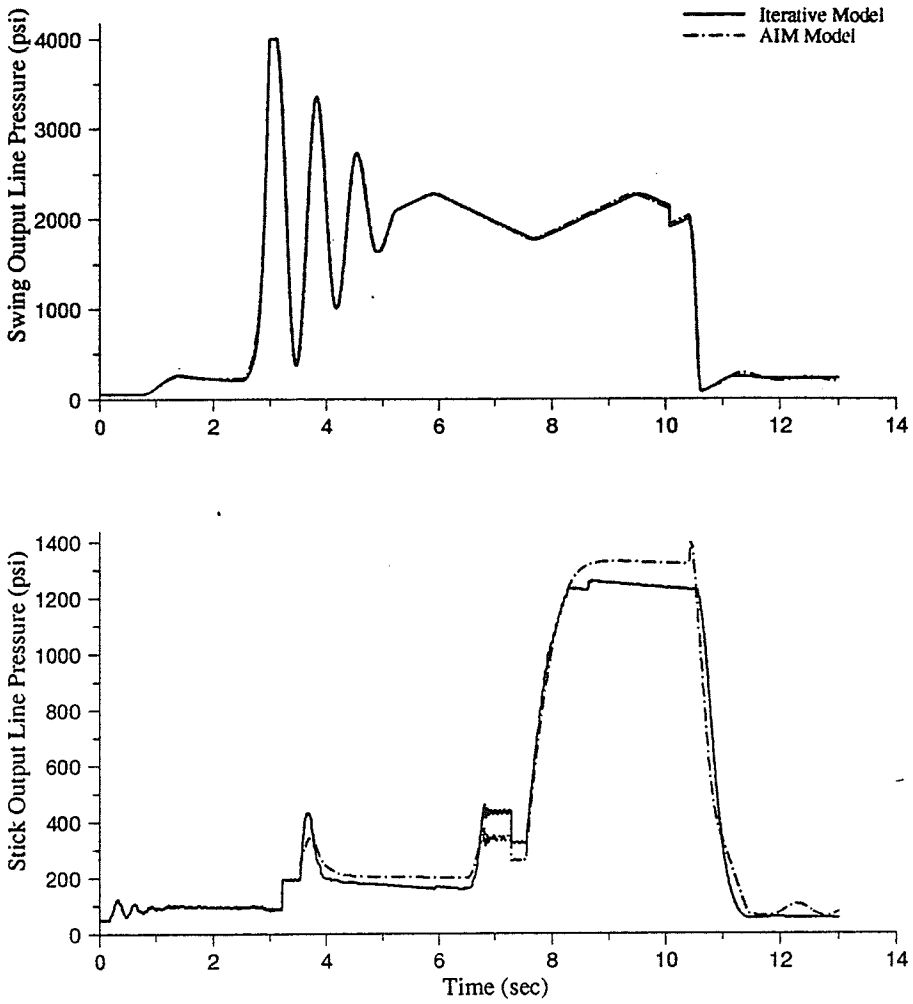


Fig. 5-40 Task Simulation Output Line Pressures for: (a) Swing; (b) Stick.

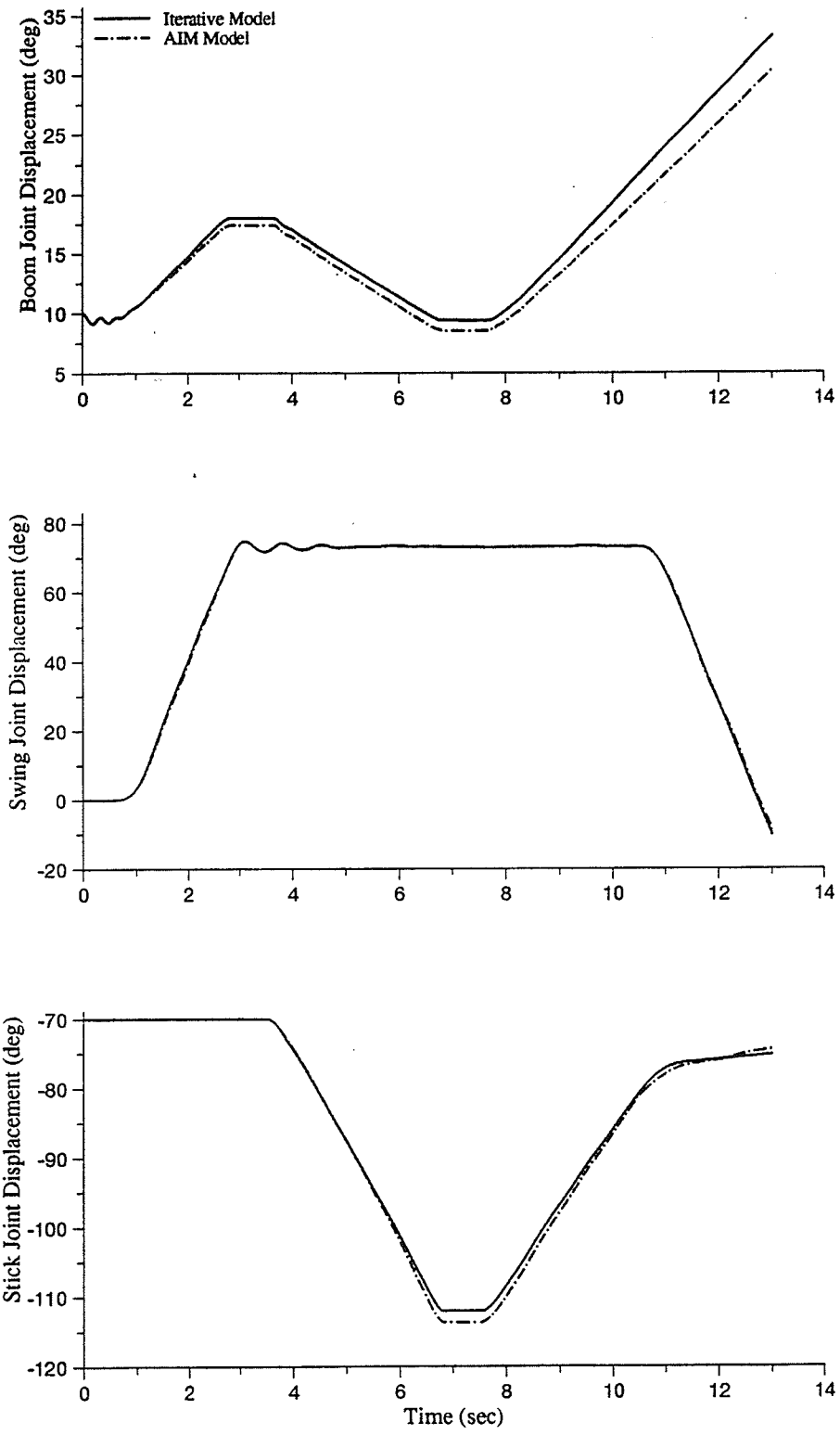


Fig. 5-41 Task Simulation Joint Displacements for: (a) Boom; (b) Swing; (c) Stick.

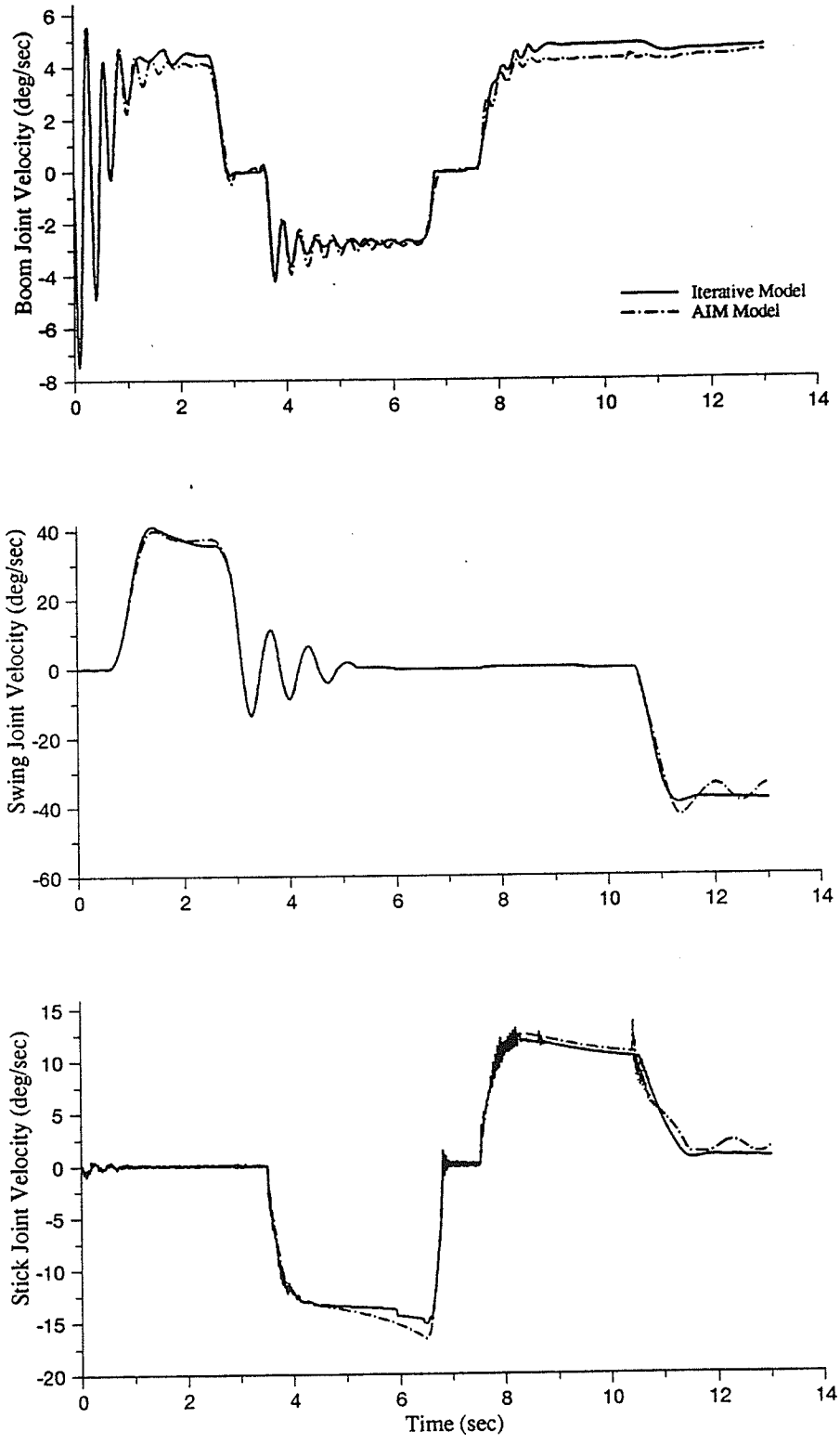


Fig. 5-42 Task Simulation Joint Velocities for: (a) Boom; (b) Swing; (c) Stick.

Figs. 5-41 and 5-42 show that the AIM models produced satisfactory results for this simulation. The maximum error for the boom is 2.67*deg* after 13.0*sec* of active simulation. The error for the swing is negligible and the maximum absolute error for the stick is 1.57*deg* at t=7.0*sec* with a maximum velocity lag occurring at approximately t=6.3*sec*. The error in the boom is caused by the valve spool displacement operating in the range of 0.188*in* to 0.251*in*. Referring to Fig. 4-11, this is the region where largest errors in the flow calculation occur.

Figs. 5-43 and 5-44 show the end effector movement and the end effector error for this simulation. The maximum absolute error for the end effector is 0.347*m* at time t = 13.0*sec* and the cause of the steadily increasing error in all the links is due to the increasing lag in the boom joint displacement. This should be obvious from the error shown in the joint displacements in Fig. 5-41 and the fact that at times before 8.0*sec* the maximum absolute error for this task is below 0.1*m*. As we noted in section 5.3.1 of this chapter, the boom will have a pronounced effect on the end effector due to its prominence in the equations used to derive Y and Z.

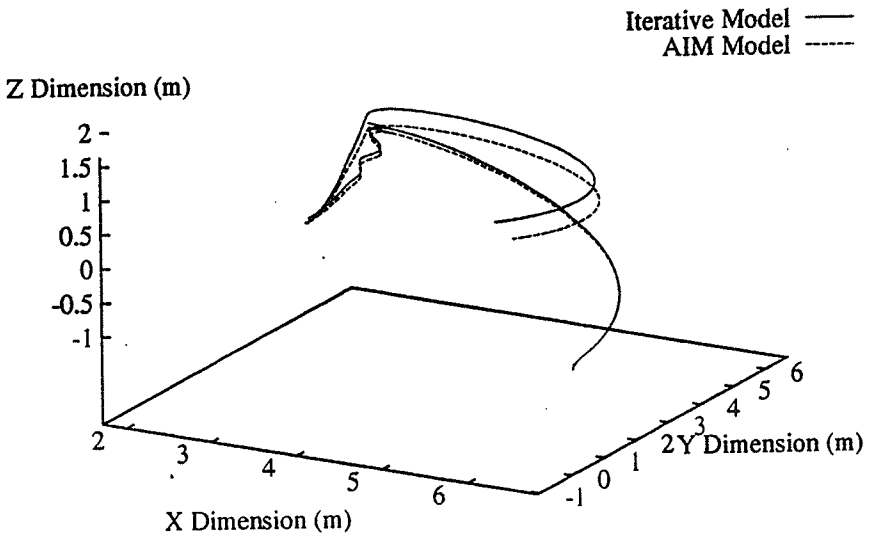


Fig. 5-43 Task Simulation End Effector Response

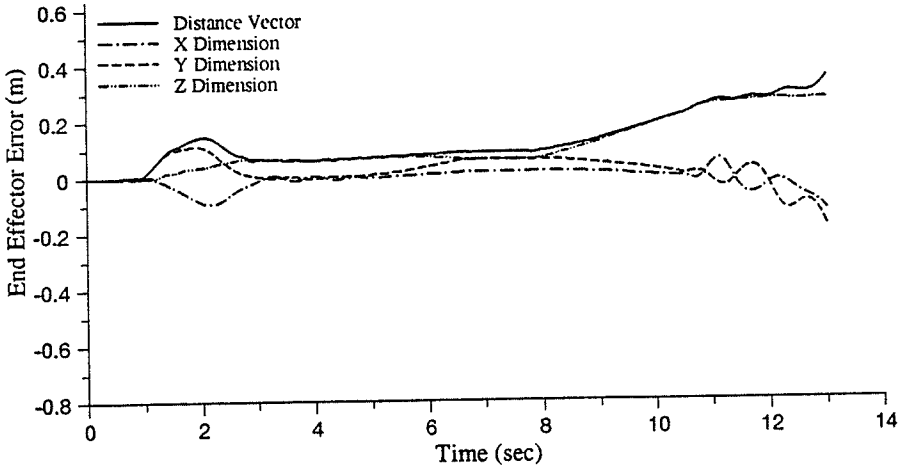


Fig. 5-44 Task Simulation End Effector Error

Note that for all of the simulations presented in this chapter, the polynomial abductive network models were generated from previous simulations derived from the Caterpillar-215B

excavator. This allowed us to formulate the best method of modeling the system without the interference of unknown ranges or variables. It is obvious that to model directly from the real world system, we would not have knowledge of the exact functional changes for the valve orifice area curves and therefore the exact flow maps. However, in a real world system we would be able to obtain data for the flows, pressures and valve spool displacements using available measuring devices. As long as the flow deadbands for the actuators could be determined and modeled separately we would be able to accurately model the system since we have shown how abductive networks can give good results over the entire active ranges of the actuators with the majority of error coming when the swing and stick are both just opening.

5.4 COMPUTATIONAL COMPARISON

As was mentioned in section 5.2, it was necessary to loosen the restrictions on the convergence test for the iterative derivation of pump pressure for the graphs in section **D** when both the swing and stick are operational. The restriction that was loosened is called 'eps' which states the percentage of error allowed in the iterative derivation in order for the pump pressure value to be considered correct.

To illustrate the effect that the percentage of allowed error in the iterative derivation for pump pressure has on the values of the input flows, we will revisit the simulation which corresponds to the excavator performing a task. Figs. 5-45(a) through (c) shows the two dimensional result for the swing input flow for varying values of eps. Note that for all the simulations performed in chapters 4 and 5 the eps value used in the iterative models was 0.05 or 5% error.

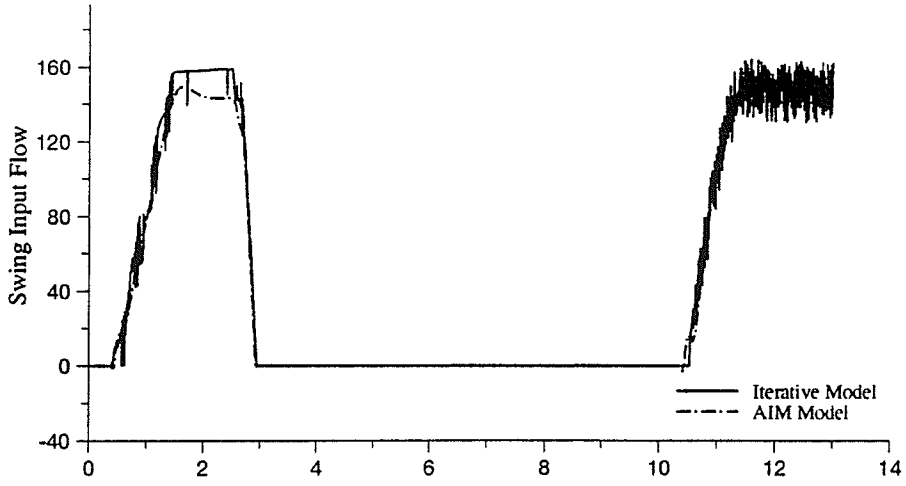


Fig. 5-45 Simulation Results When: (a) eps = 0.1; (b) eps = 0.05; (c) eps = 0.001.

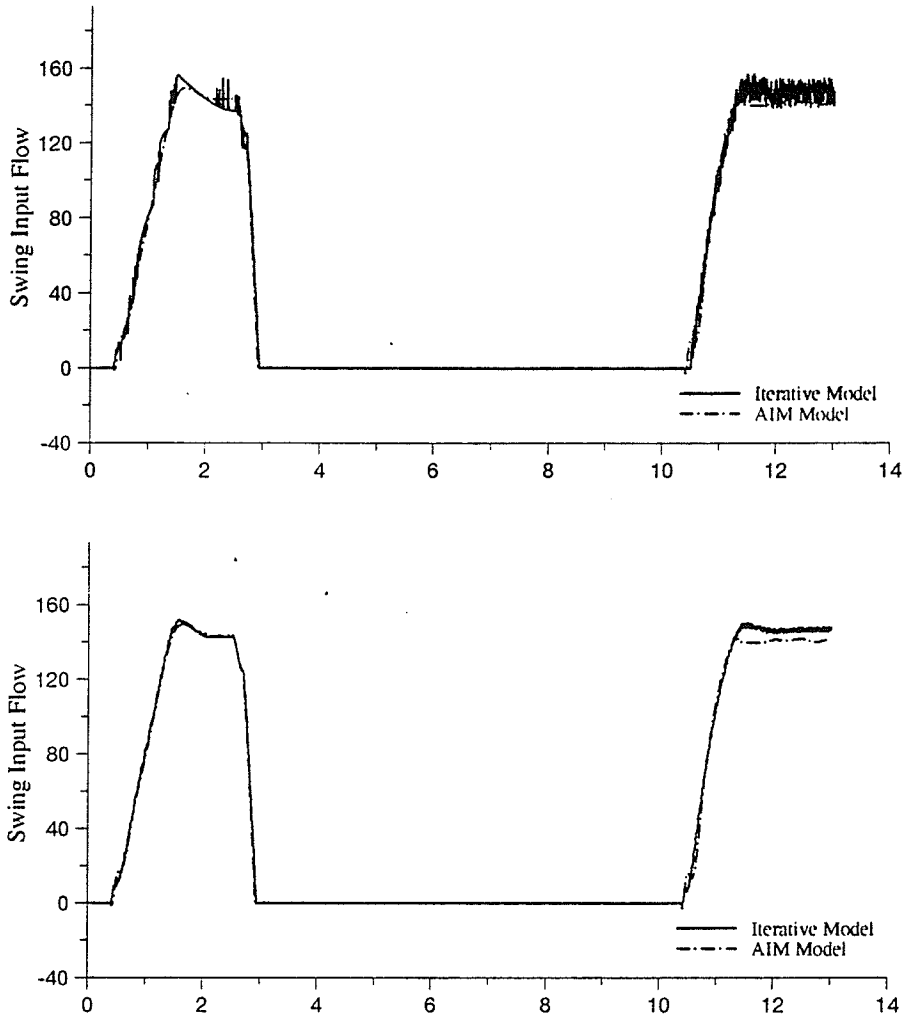


Fig. 5-45 (cont'd).

As we can see in Fig. 5-45, the allowable amount of error does have a considerable effect on the flow response of the swing in the simulation. The allowable range of the values is indicated in these graphs by the consistent value shifts from the high end of the range to the low end which results in the flow values forming a wide bar during the simulation instead of the smooth line as shown in the AIM model results.

When the allowable error is only 0.1% (i.e. eps=0.001), the range of values shown for flow is small with only a variation of $\pm 5 in^3/sec$ shown in the latter part of the simulation.

As the percentage of allowable error is increased the magnitude of the variation in the flow values increases. When the allowable error is 5%, (eps=0.05), the swing flow range is $\pm 20 in^3/sec$ in the latter portion of the simulation and when the allowable error is 10%, the range of the swing flow is $\pm 40 in^3/sec$.

It is obvious then, that the best approach to deriving the iterative models would be to restrict the allowable error as much as possible. Unfortunately a very tight range on the allowable error can lead to another problem, that the iterative model will fail to converge and the program will enter an infinite loop. This is the reason that 5% error was used to generate the iterative models, since allowable errors of 0.1% up to 4% resulted in the program failing to converge in its pump pressure derivation.

Using a wider allowable error range did provide benefits for the iterative model in terms of computation time reduction. Table 4 shows a comparison between the AIM model and the iterative model for a complete 13*sec* simulation with a time step every 0.005*sec*. In addition, computation times are shown for varying values of 'eps' to illustrate the range of times achievable using the iterative model.

TABLE 4 AIM Model versus Iterative Model Computation Times with Varying eps.

eps value	Iterative Model	AIM Model	Computation Reduction
0.001 (0.1%)	16.09 sec	10.63 sec	33.93%
0.05 (5.0%)	14.47 sec	10.63 sec	26.54%
0.1 (10%)	13.98 sec	10.63 sec	23.96%

The AIM model provides for a better than real time simulation meaning that one of the goals of this research has been realized. For each time step of 0.005*sec* the program was able to complete its computations in 0.0041*sec*. If the derivation of pump flow is removed from the iterative simulation the resulting total computation time is 9.90*sec*. This indicates that the amount of computation time needed to use the AIM models in the simulation is a total of 1.73*sec* versus a minimum of 4.08*sec* for the iterative models. The computation reduction is not as pronounced as it was in chapter 4 which considered only the single link case, however the improvement is substantial and there is no danger of the AIM model simulation entering an infinite loop.

The iterative model results vary from 16.09*sec* to 13.98*sec*. The improvement in the iterative models is approximately 2.0*sec* when an eps value of 10% is used over an eps value of 0.1%. Fig. 5-45 shows that the corresponding loss of accuracy in the model makes this a poor trade off however, and since the model still does not provide for a real time simulation the advantage of using the AIM derived models becomes clear.

CHAPTER 6
EFFECT OF NON CONSTANT PUMP FLOW

6.1 Effect on the General Flow Maps

When training the AIM models used in this research and in the simulation comparisons in chapters 4 and 5, we assumed that the type of pumps used in the hydraulic circuits were of varying pressure and constant flow. In the real caterpillar 215B excavator however this is not entirely the case. The excavator's hydraulic circuit uses a technique called 'torque limiting' to prevent the mechanical failure of the components at high pump pressures. This means that at high pump pressures the pump reduces the amount of flow thus preventing a circuit overload.

The inclusion of torque limiting into the models results in a modified general input flow map. As an example the torque limited map for the boom is shown in Fig. 6-1.

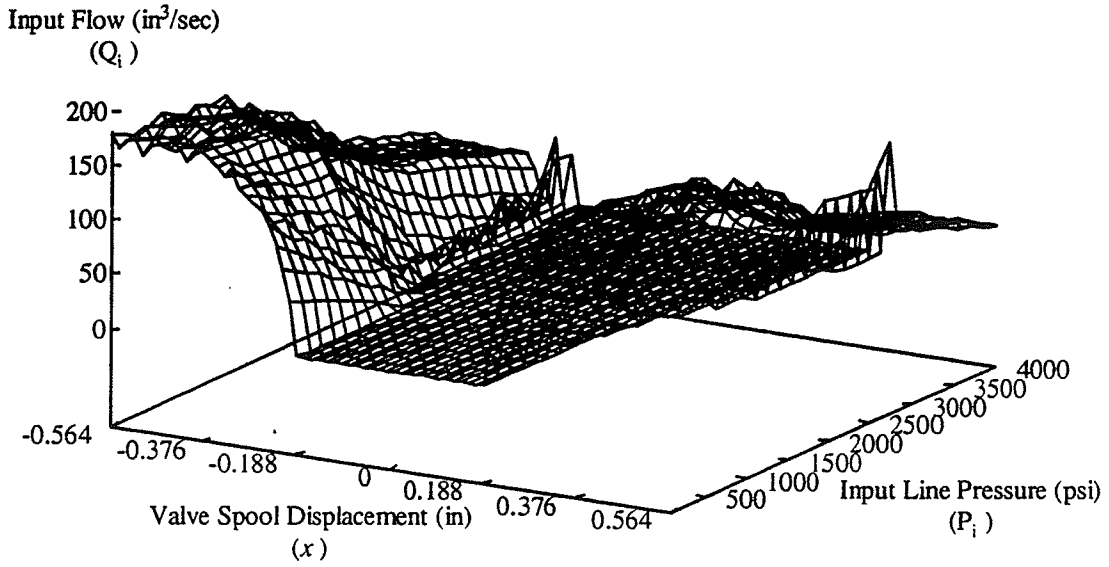


Fig. 6-1 Overall Input Flow Map for Torque Limited Circuit for Boom Link.

We shall investigate the effect that inclusion of torque limiting has on the performance of the AIM modeled simulations for two cases: the sinusoidal example from section 5.3.1, and the excavator task example from section 5.3.4.

6.2 Effect on Sinusoidal Voltage Simulation

Fig. 6-2 shows the resulting input flows for the swing and stick when torque limiting is included in the iterative model. Fig. 6-3 displays the resulting joint displacements for all three links and fig. 6-4 shows the joint velocities. The input voltage for all links is the same as that used in section 5.3.1 for the sinusiodal example.

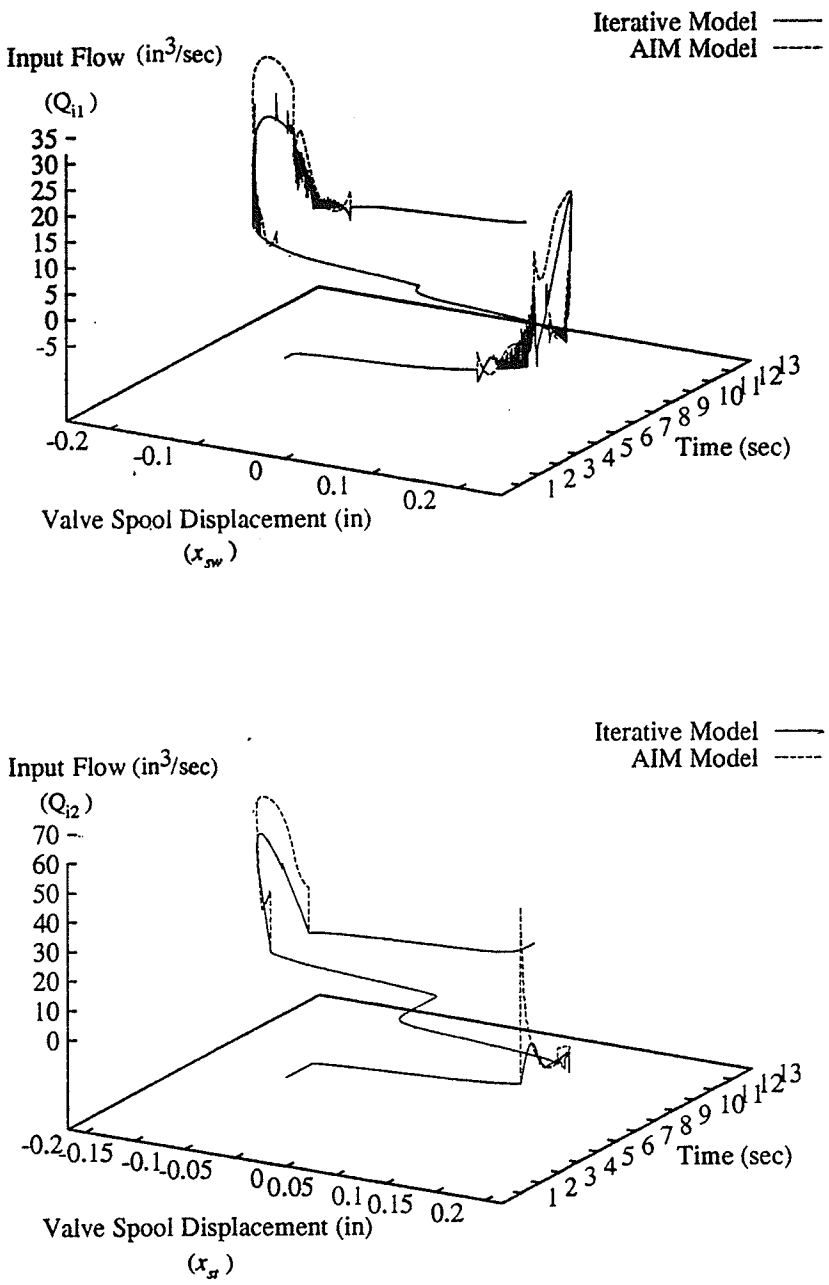


Fig. 6-2 Input Flow Repsonse to Sinusoidal Inputs for: (a) Swing; (b) Stick.

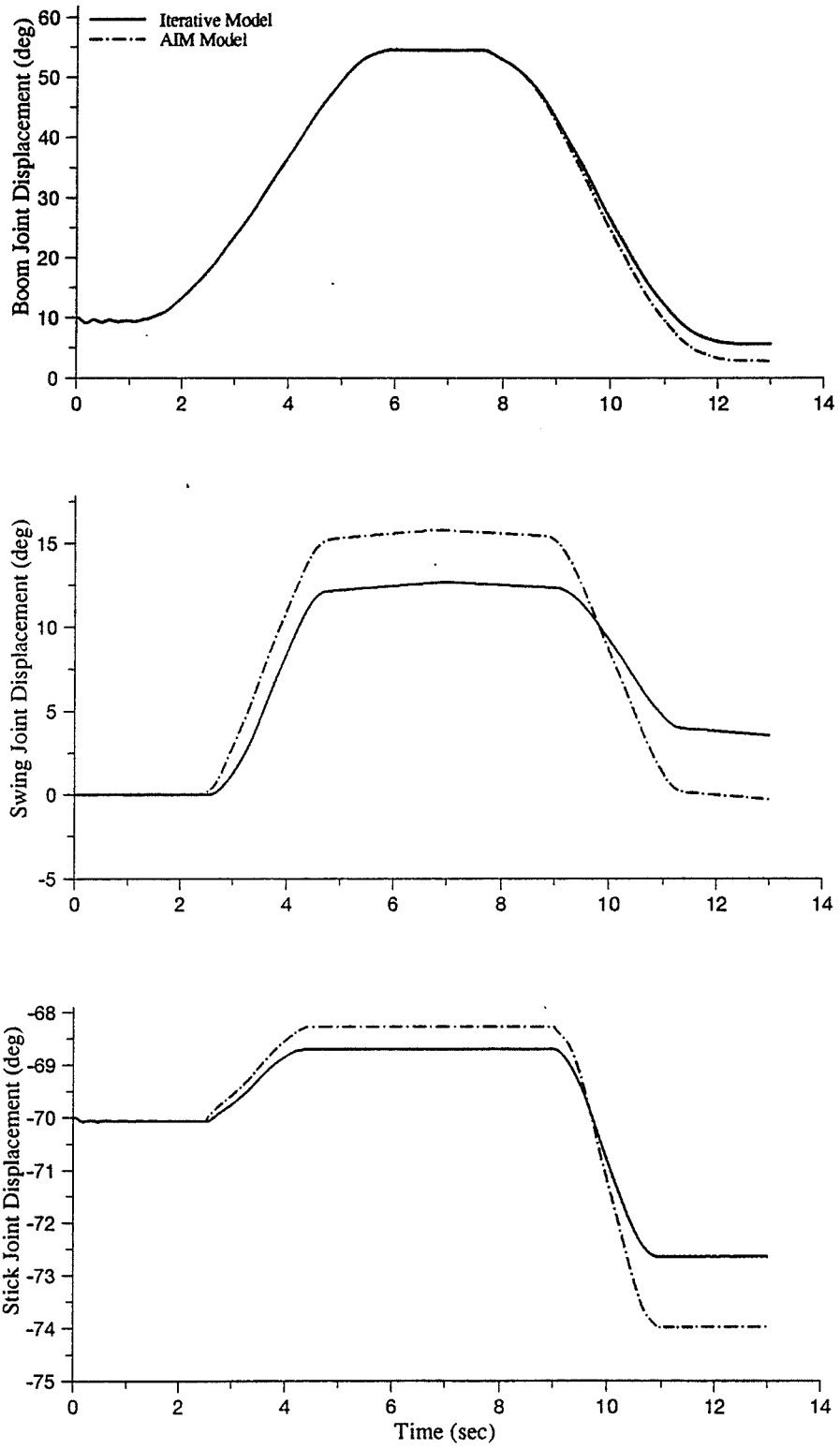


Fig. 6-3 Joint Displacement Responses to Sinusoidal Inputs for: (a) Boom; (b) Swing; (c) Stick.

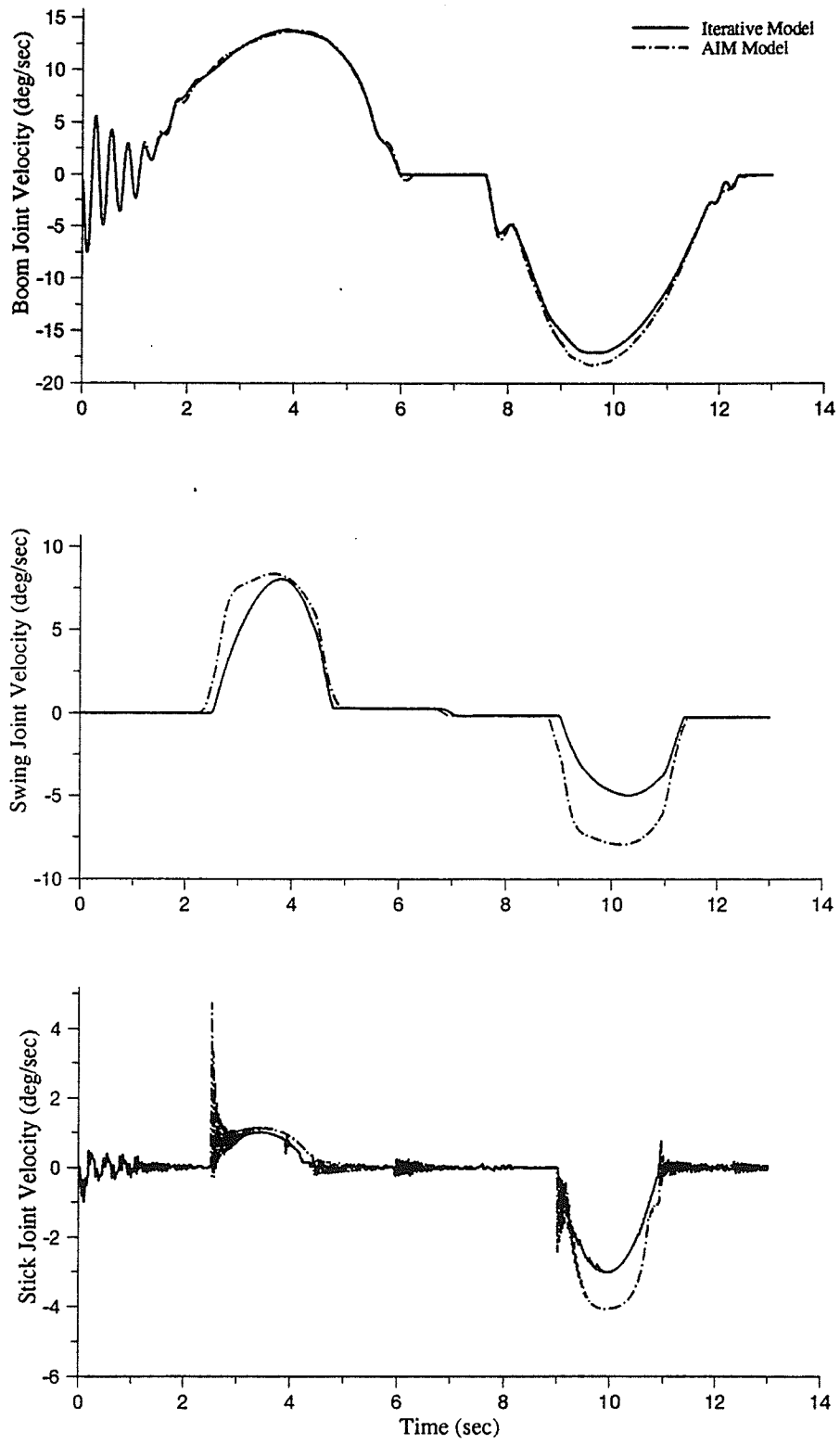


Fig. 6-4 Joint Velocity Responses to Sinusoidal Inputs for: (a) Boom; (b) Swing; (c) Stick.

Fig. 6-2 shows the input flows for the swing and stick when torque limiting is employed. When the valve spools are displaced positively the pressure in the pump is below $2000psi$ so the torque limiting has no effect and the flow values are the same as in section 5.3.1. However, when the valves are displaced negatively, the pump pressure is high enough to invoke torque limiting so we see a jump in the AIM model error. In the previous simulation where no torque limiting is used, the maximum errors for negative displacement for both the swing and stick occured at approximately $10.5sec$. The error for the swing in the previous simualtion at this time was $6.06in^3/sec$. In this simulation the error at time t = $10.5sec$ is $11.51in^3/sec$ which means the error has almost doubled by including torque limiting. For the stick link, the error increased from $9.4in^3/sec$ to $15.9in^3/sec$.

The effect that this magnified flow error has on the joint displacements and the joint velocities of the links is shown in Figs. 6-3 and 6-4. In the boom, we see that the small lag that was present in the previous simulation has become larger. At the end of the simulation the error in the boom joint displacement has grown from $1.42deg$ to $2.86deg$. For the swing and stick we will examine two time points to see the effect that torque limiting had on these joint displacements: time t = $6.5sec$, and time t = $13.0sec$. At time t = $6.5sec$ the error in the swing has increased from $2.76deg$ to $3.14deg$. At time t = $13.0sec$ the value has increased from $1.302deg$ to $3.81deg$. The stick joint displacement error in the swing at time t = $6.5sec$ increased from $0.4deg$ to $0.43deg$, while at t = $13sec$ the error increased from $1.06deg$ to $1.38deg$.

The change in the end effector movement and the error in the end effector is shown in Figs. 6-5 and 6-6, respectively. We see that when the end effector error for this simulation is compared to the previous one, there is a large increase in error for the last $4.0sec$ of the simulation due to the torque limiting induced error. Previously, at time t = $13.0sec$ the error

value was 0.226*m* for absolute error. For this simulation the amount has more than doubled with the error for this simulation being 0.547*m*.

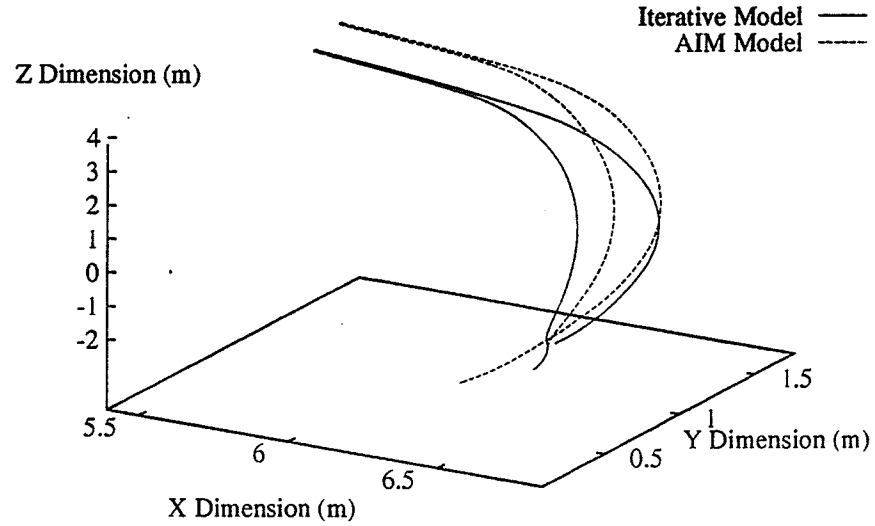


Fig. 7-5 End Effector Response to Sinusoidal Inputs.

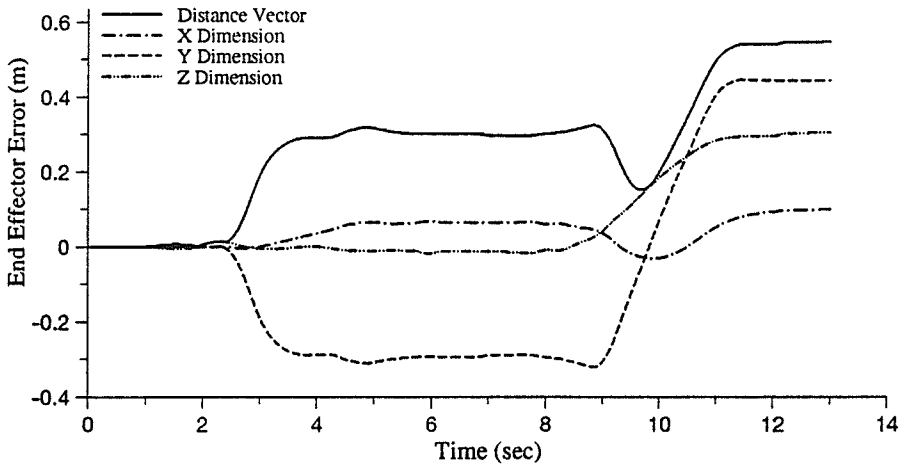


Fig. 7-6 Sinusoidal Simulation End Effector Error.

6.3 Effect on the Simulated Task

Fig. 6-7 shows the swing and stick input flows that result when torque limiting is included into the iterative model for the simulation of the excavator task. Fig.6-8 displays the resulting joint displacements and Fig.6-9 shows the joint velocities.

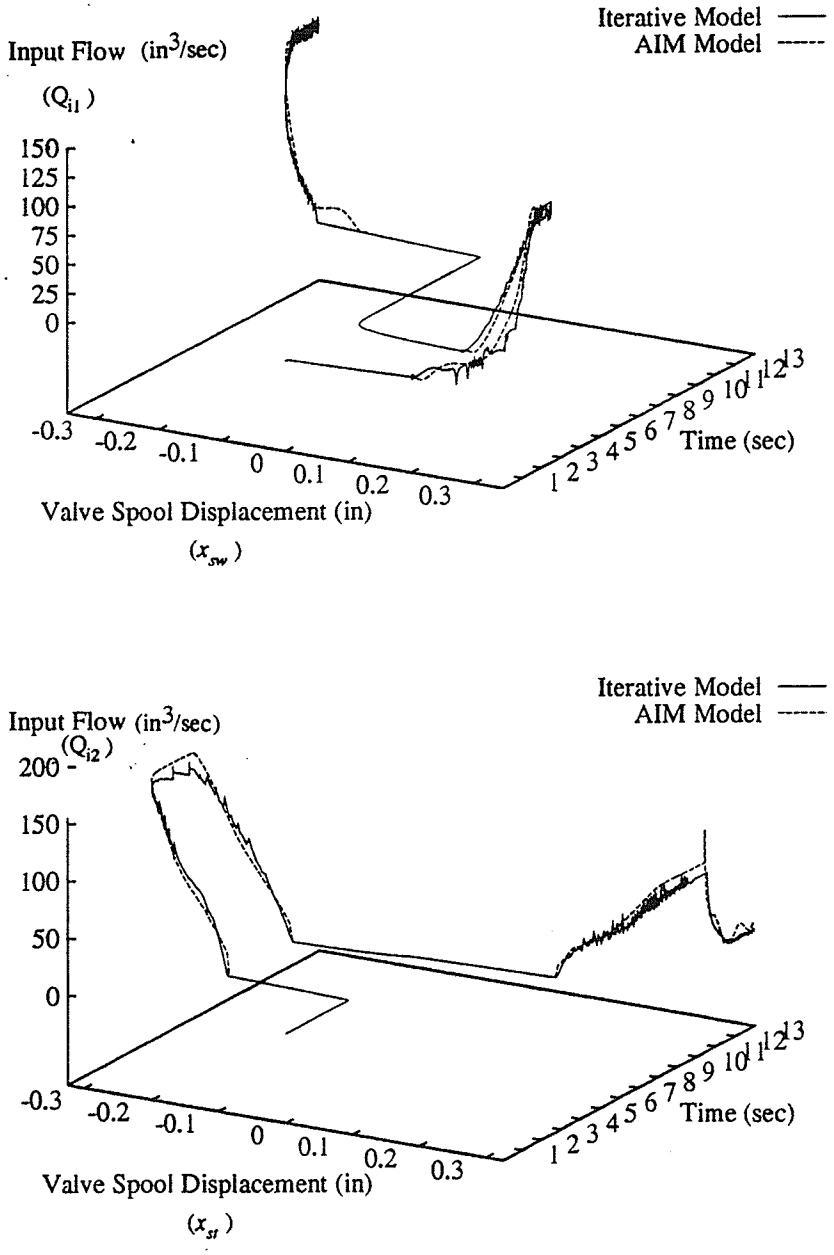


Fig. 6-7 Task Simulation Input Flows for: (a) Swing; (b) Stick.

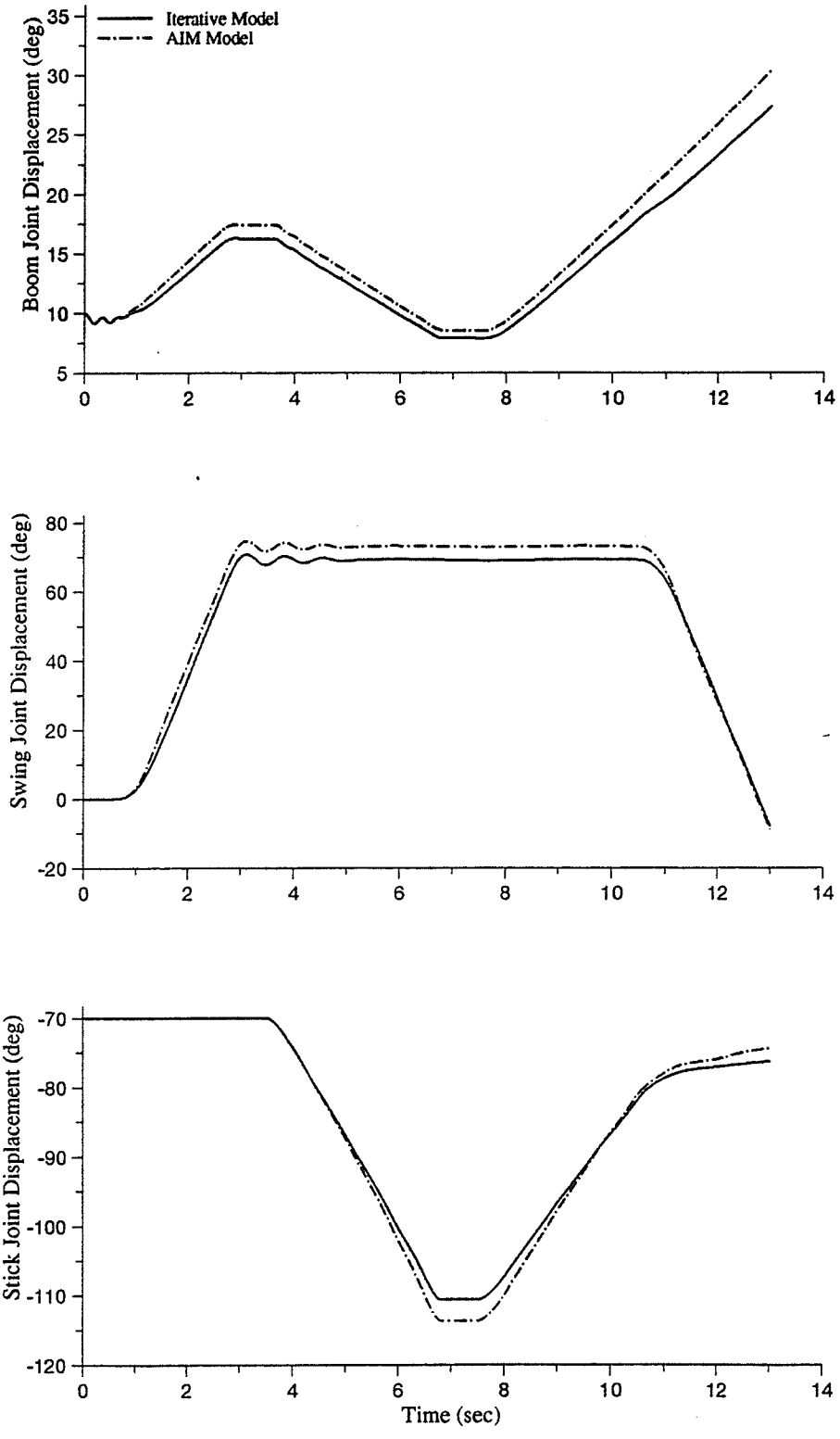


Fig. 6-8 Task Simulation Joint Displacements for: (a) Boom; (b) Swing; (c) Stick.

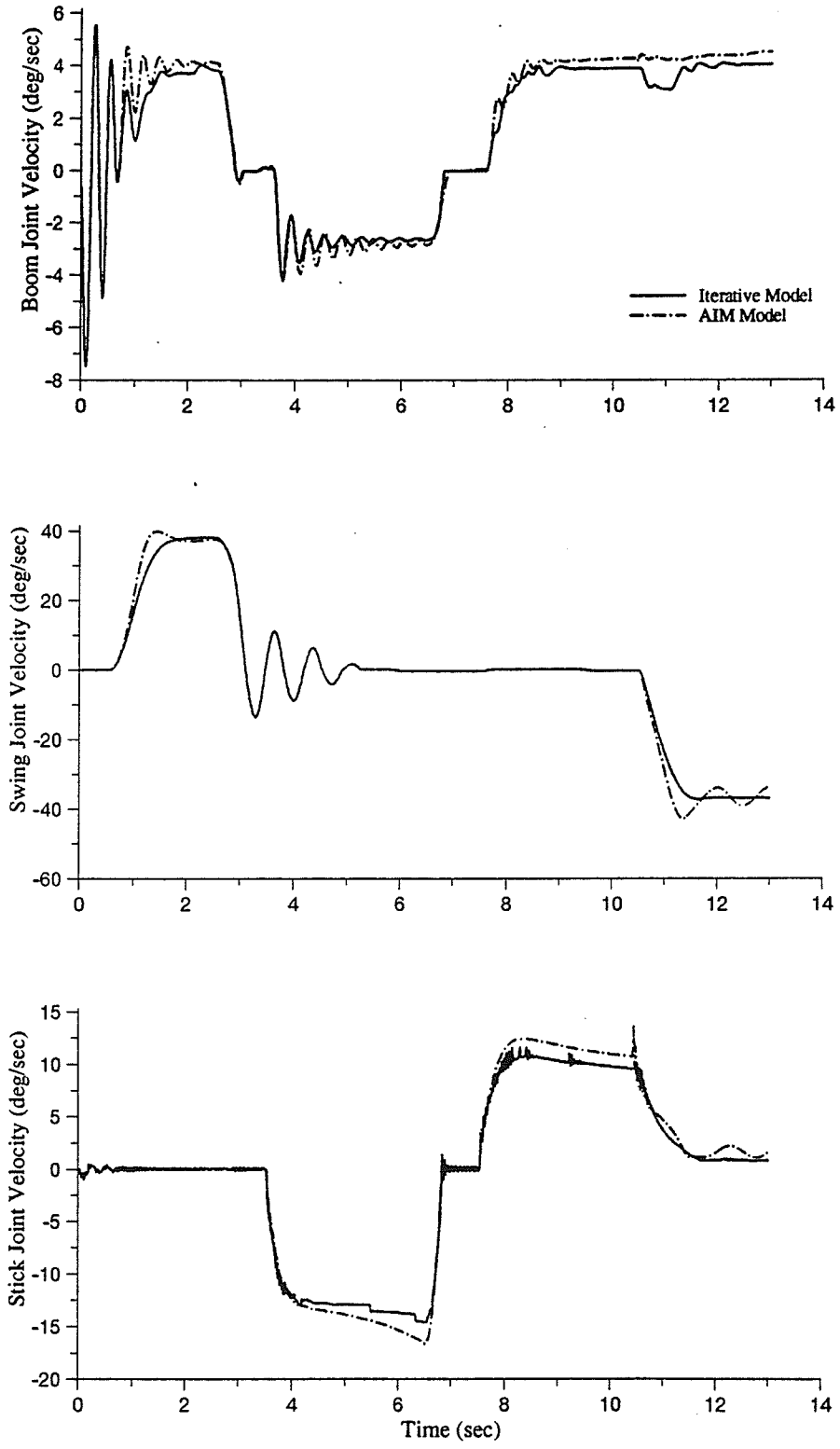


Fig. 6-9 Task Simulation Joint Velocities for: (a) Boom; (b) Swing; (c) Stick.

The effect that the non constant flow had on the input flows to the swing and stick is shown in Fig. 6-7. For the swing, the effect is felt on both the positive and negative sides of the valve spool displacement axis where the difference between the AIM model and the iterative model has increased. The magnitude of the error is not increased substantially, but the amount of time that there is a noticeable lag between the AIM and iterative models has increased. The stick actuator is affected by an increased error in the negative valve spool displacement portion of the simulation and the introduction of a constant lag when the valve spool is displaced postively.

Figs. 6-8 and 6-9 show the resulting joint displacements and joint velocities for the simulation. When we compare them to the results shown in Figs. 5-41 and 5-42 in section 5.3.4 the following differences become evident. First, the amount of boom error has changed dramatically. In the previous simulation the use of the AIM model resulted in an undershoot of $2.84deg$ in the joint displacement at the end of the simulation. With the introduction of non constant flow the AIM model now **overshoots** the iterative result by $3.03deg$ at the end of the simulation. Secondly, In the swing motor the error has increased from being negligible to having an AIM model overshoot of $3.87deg$ at time t=$7.0sec$. Finally, in the stick actuator the error at time t = $7.0sec$ has increased from $1.54deg$ to $3.0deg$ and the error at the end of the simulation has increased from $0.81deg$ to $1.93deg$. In effect, the error has been at least doubled in all the links with the addition of non constant flow.

Figs. 6-10 and 6-11 show the end effector movement and the end effector error for this simulation. There is now a noticeable lag in the end effector movement for a majority of the simulation and Fig. 6-11 shows that the maximum absolute error has increased from $0.347m$ to $0.539m$. In addition, the error now increases rapidly once the valve spools have passed their deadbands and then stay at a relatively constant level for the remainder of the simulation.

Fig. 6-11 also shows that the error in the X and Y dimensions has increased when compared to the results shown in Fig. 5-44, and the Z dimension error has remained equal in magnitude but is now a negative error as opposed to the positive error in Fig. 5-44.

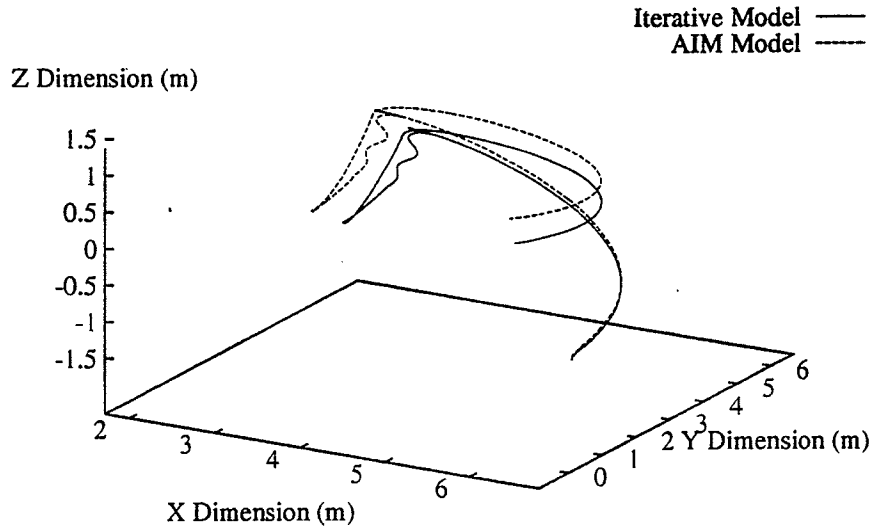


Fig. 6-10 Task Simulation End Effector Movement.

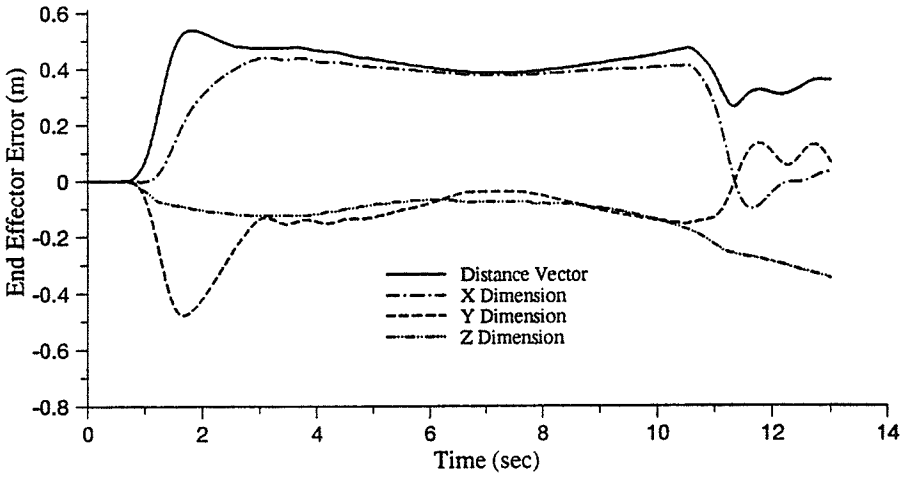


Fig. 6-11 Task Simulation End Effector Error

This chapter has shown that although the AIM networks do well in modeling the excavator when the pump pressure is constant, there is a substantial error introduced when the pump pressure is variable. The two reasons for this are: the training data used to generate the polynomial networks did not account for non constant pressure and, there is no feedback control system to allow the AIM networks to adjust to changes in system parameters.

CHAPTER 7
CONCLUSIONS

A methodology for using AIM-generated polynomial abductive networks to model a class of large-scale hydraulic systems was presented in this thesis. Previous conventional methods of modeling have focused on manipulating the state space variables in a partitioned hierarchical manner, or by applying transmission line methods so that parallel processing can be used. These methods can develop exact, though complex, mathematical models of hydraulic systems. They are, however, computationally inefficient, and rely on a complete knowledge of the hydraulic system parameters and functions.

In this thesis a different approach to the mod ·draulic systems was studied. The approach uses a supervis program called the 'Abductory Induction Mechanism'. The nctional relationships in a heavy-duty hydraulic actuation system, a .der polynomials as nodes in an abductive network model. The feasibility of u nique was validated and it was also shown that polynomial network methods are compa able to the more well known neural

network methods when performing supervised learning without the need to intuitively set several architectural and training parameters. The network models proved to have sufficient accuracy, and were simple and fast enough for real-time simulations. The errors introduced in some simulation results when flow was assumed constant did not detract significantly from the overall performance of the simulations and the errors occurred primarily when the valve spool was just overcoming its deadband.

The significance's of the modeling methodology presented here are: First, the number of variables that would have to be included in the model was reduced. In particular, the need for calculating the valve orifice areas was removed. The valve orifice areas are very difficult to obtain from experimental data taken in a real life situation although it is a fundamental variable in the iterative model used in both the calculation of the input flow and the iterative calculation for pump flow. Therefore, only by using AIM models can the valve orifice areas be removed. Secondly, as a result of the removal of the iterative routines, the required computation time decreased substantially so that a better than real-time simulation for a multi-link system was obtained on a personal computer. Thirdly, by removing the pump pressure derivation from the simulation the propagation of error through the flows and line pressures was reduced.

The outcome of this work has clearly demonstrated the potentials and promises of polynomial abductive network modeling related to complex hydraulic actuation systems. Future work should focus on deriving the model directly from the experimental measurements. This requires a detailed qualitative analysis to relate the degree of incompleteness of the data to the degree of accuracy in the AIM-generated model and would provide a training database that incorporates the effects of non constant flows and mechanical nonlinearities such as stick/slip friction and joint limits.

APPENDIX A
THE LOGIC OF ABDUCTORY INDUCTION

A.1 INTRODUCTION

The deductive logic system is a form of reasoning from general principles and facts and that the inductive logic system is a form of reasoning from facts to general principles. However, it is known that deduction only involves reasoning with certainty and it is based on the law that everything is either true or false with no in-between. Unfortunately, uncertainty is present in almost every problem that we face every day so deductive logic is not a good simulator for the realistic decision process.

A better method of logic is what's known as abduction. Abduction in a practical sense is the process of reasoning from general principles and facts to new facts under uncertainty. This form of reasoning is prominent in statistical analysis, fuzzy logic and various other forms of inference, especially the functioning of the human brain[19].

This appendix will first explore abductive logic in its philosophical sense, then we will discuss a hybrid form of this logic which is applicable to problem solving that is called

abductory induction. Finally this section will explore some practical applications of the abductory induction control process in expert systems.

A.2 ABDUCTION LOGIC

The theory of abduction was first proposed by the philosopher C.S Peirce in the late 1800's [19]. He stated that abduction is the logical process by which new ideas are formed. Therefore, abduction is the first step in scientific inquiry that leads to the forming of hypotheses. He later refined his theory to state that " The process of abduction is concerned with analyzing the reasons for proposing a hypothesis", and he concluded that abduction was a form of inference.

Peirce defined abduction as a form of inference because it is the way in which hypotheses are inferred from a surprising fact. We can deduce that hypotheses are the results of inference because there could be literally trillions of reasons why a certain phenomena was observed. The vast majority of these are discarded without us even consciously thinking about it because we consider them as having a minuscule probability of contributing to the phenomena. What we are left with is a number of theories which could possibly explain the phenomena but we eventually end up with only one hypothesis which we believe (rightly or wrongly) to be the correct cause.

Pierce formulated three considerations to determine the choice of a single hypothesis from the several viable alternatives, and these considerations are:

1) The hypothesis must be such that it will explain the surprising facts around us.
2) The hypothesis must be capable of being subjected to experimental testing.
3) The hypothesis must be considered with regards to economy.

The first consideration is the most obvious one and is the primary mechanism by which we reduce the number of potential hypotheses from several thousand to perhaps a dozen. The second consideration is important because a hypothesis is more likely to be put forward if we

are able to analyze experimental data to determine the validity of our assumption. The third and final point is an important consideration because we are more able to experimentally test our hypothesis if it is cheaper (in terms of money, time, space, etc.) and therefore we will have a preference to choose the hypothesis which is easier on available resources even if it doesn't have the highest probability of being correct. However, even when we have chosen a particular hypothesis we will entertain an altered or new hypothesis if evidence is found contrary to our original guess because we know that there is a certain amount of uncertainty present. Since there is uncertainty, we will naturally accept a certain range of results because our hypothesis is merely a starting point and not a given fact.

As we have seen, abduction is the only logical method for proposing a new idea. Deduction proves that something must be, induction evaluates and shows that something must be, but abduction merely suggests that something may be.

A.3 ABDUCTORY INDUCTION

As was previously discussed, induction is the process of reasoning from facts to general principles. The special process of reasoning from facts to abductive principles, or hypotheses, is termed abductory induction.

This hybrid logic system is used in practical applications under uncertainty. The reason that there is an induction component is that ,as we saw in the last section, abduction is the process whereby hypothesis are formed under uncertainty. Unfortunately, this inference method does not contain a quantitative method for evaluating or working towards that hypothesis. Therefore, it is necessary to amalgamate the inductive inference process with abduction so that a potential hypothesis can be deduced with a quantitative measurement of certainty.

The abductory induction system is a powerful inference control system because it is a process that will work under uncertainty. The process consists of taking facts (which can be

certain or uncertain) and then working in an inductive manner to reach the abductive hypothesis which itself is awarded a degree of certainty or uncertainty [18].

A.4 APPLICATIONS

Abductory induction can be used a control process which can provide the system a greater degree of realism in the manipulation of data and inferring a conclusion. This system allows for the user to work with scenarios in which there are a large number of parameters, missing or uncertain data, unknown relationships between various aspects of the problem, and rapidly changing circumstances.

There are a large number of subject areas in which a system using abductory induction would be very valuable. These areas include scientific research, medicine, simulation, diagnostics, pattern recognition and organizational management.

REFERENCES

[1] N. Sepehri, P.D. Lawrence and F. Sassani, "Partitioned Hierarchical Modeling of Hydraulic Systems," *Proceedings IASTED International Conference on Modeling and Simulation*, Calgary, Canada, pp. 170-174, 1991.

[2] J.D. Burton, K.A. Edge and C.R. Burrows, "Modeling Requirements for the Parallel Simulation of Hydraulic Systems," *ASME J. of Dynamic Systems, Measurement, and Control* **116**, pp. 137-145, 1994.

[3] G.J. Montgomery and K.C. Drake, "Abductive Networks," *Proceedings SPIE - International Society for Optical Engineering* , Orlando, FL, pp. 56-64, 1990.

[4] P.D. Lawrence, F. Sassani, B. Sauder, N. Sepehri, U. Wallersteiner and J. Wilson, "Computer-Assisted Control of Excavator-Based Machines," SAE Technical Paper No. 932486, *International off-Highway & Powerplant Congress & Exposition*, Milwaukee, WI, 1993.

[5] A. Ghasempoor and N. Sepehri, "A Measure of Machine Stability for Moving Base Manipulators," *IEEE International Conference on Robotics and Automation*, Nagoya, Japan, pp. 2249-2254, 1995.

[6] M.R. Sikora, "Hydraulic Control System Analysis Using Network Simulation", *Control Engineering*, pp. 70-72, 1991.

[7] S.K.R. Iyengar, "A Systematic Approach to the Analysis of Complex Fluid Power System," *Proceedings 4th International Fluid Power Symposium*, Sheffield, England, pp. A2.9-A2.34, 1975.

[8] F. Anoush and S. D. Root, "Advanced Computer Simulation and Modelling for Solving Single Phase Hydraulic Problems," *Proceedings International Exhibition and Conference for the Power Generation Industries - Power-Gen*, New Orleans, LA, pp. 1107-1113, 1989.

[9] N. Sepehri, P.D. Lawrence and F. Sassani, "An Alternative Approach for Modeling a Class of Hydraulically Powered Manipulators," *Robotics and Manufacturing* **3** (M. Jamshidi and M. Saif eds.), ASME Press, New York, pp. 57-64, 1990.

[10] N. Sepehri, P.D. Lawrence, F. Sassani and R. Frenette, "Resolved-Mode Teleoperated Control of Heavy-Duty Hydraulic Machines," *ASME J. of Dynamic Systems, Measurement, and Control* **116,** pp. 232-240, 1994.

[11] N. Sepehri, F. Sassani, P.D. Lawrence and A. Ghasempoor, "Simulation and Experimental Studies of Gear Backlash, Stick-Slip Friction and Joint-Limit in Heavy-Duty Hydraulic Machines," submitted to *ASME J. of Dynamic Systems, Measurement, and Control.*

[12] D.Z. Wang, S.M. Rohde and K.P. Oh, "Hydraulic System Modeling and Simulation in the Automotive Industry," ASME Paper No. 90-WA/FPST-4, *Proceedings ASME Winter Annual Meeting*, Dallas, TX, 1990.

[13] P.A. Hogan, C. R. Burrows and K.A. Edge, "Development of a Knowledge Based System for the Diagnosis of Faults in Hydraulic Circuits," ASME Paper No. 90-WA/FPST-2, *Proceedings ASME Winter Annual Meeting*, Dallas, TX, 1990.

[14] R. I. Scoville and M. J. Maxwell, "Concurrent Design/Analysis Tools for Aircraft Hydraulic Systems," SAE Technical Paper No.902004, *Aerospace Technology Conference and Exposition*, Long Beach, CA, 1990.

[15] B. J. Bang and J. G. Draxler, "Dynamic Hydraulic System Simulation.; An integrated Approach," SAE Technical Paper No. 902003, *Aerospace Technology Conference and Exposition*, Long Beach, CA, 1990.

[16] I. Satoshi, H. Keiichi and O. Kouchi, "Dynamic Simulation and Analysis System for Electro-hydraulic circuit," SAE Technical Paper No.921668, *International off-Highway & Powerplant Congress & Exposition*, Milwaukee, WI, 1992.

[17] G.M. Lance and W.C. Prescott, "Modeling and Dynamic simulation of a Hydraulically Driven Backhoe," SAE Technical Paper No. 901640, *International off-Highway & Powerplant Congress & Exposition*, Milwaukee, WI, 1990.

[18] Abtech Corporation, "Abductory Induction Mechanism," *AIM User's Manual*, Charlottesville, Virginia. 1988.

[19] K.T. Fann, "Pierce's Theory of Abduction," Martinusnijhott, The Hague, 1970.

[20] J. Hertz, A. Krogh, R.G. Palmer, "Introduction to The Theory of Neural Computation," Addison Wesley, New York, 1991

[21] R. Henke, "Cycle Profile Provides Key to Circuit Design," *Proceedings National Conference on Fluid Power*, Detroit, Michigan, pp. 255-263, 1986.