Development of a VR – based CMM System for Industry Training and CMM Path Planning

by

Long Zhao

A Thesis submitted to the Faculty of Graduate Studies of

The University of Manitoba

in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Mechanical and Manufacturing Engineering

University of Manitoba

Winnipeg

Abstract

This research proposes a VR - based CMM operation system combined with the path planning algorithm to effectively shorten the inspection time. The virtual CMM system is also a good educational tool for industries to train new employees. The realistic interface of virtual CMM environment allows users to practise the CMM operation process before operating the real CMM. The system is also useful to reduce the collision possibility by verifying the collision-free path for products inspection.

This system uses Eon Studio as the simulation tool to build the virtual environment of CMM operations. Collision detection is implemented in Eon studio by using built-in functions and VB script programming. A new algorithm combines the integer programming model. A big penalty M method is proposed to avoid unnecessary collisions. The simulation and optimization results prove that the proposed algorithm can effectively reduce the total probe travelling distance with improved inspection efficiency.

Acknowledgments

I would like to take this opportunity to thank my advisor, Dr. Qingjin Peng. Without his constant support and great patience, I would not able to complete this thesis. I have been confused in my master study and met some family issues in my three years' graduate study. His encouragement is very important for me to complete my degree.

I am grateful to my colleagues in the Virtual Manufacturing Lab: Dr. Xiumei Kang, Tong Zhang, and Tingting Zhao, for their help in my thesis research.

I would also like to thank the general manager of Sunrui Titanium Precision Casting Company, Mrs ShuJuan Bao. Without her understanding and approval, I would not able to come back to university to complete my master program.

Finally, I would like to thank my parents and my wife. They are the most important people in my life. Without their support and understanding, it is impossible for me to obtain this degree.

Contents

List of Tables

List of Figures

List of Abbreviations

CMM	Coordinate Measuring Machine
VR	Virtual Reality
VE	Virtual Environment
Auto IP	Auto Interim Point
MCS	Machine Coordinate System
PCS	Part Coordinate System
VB	Virtual Basic

Chapter 1 Introduction

1.1 Motivation

Product quality, cost, service and delivery date are the four top key issues for a company's reputation. Among these four issues, product quality is the most important one to ensure that a company can survive and receive continuous customer orders. Quality control plays a significant role in the whole manufacturing cycle. Industries have to spend huge time and money to manage and improve their product inspection systems. Therefore increasing the product inspection efficiency and accuracy is necessary and significant for each industry. Any possible development that can be utilized in the current product inspection system is worth to study and analyze.

Coordinate measuring machine (CMM) is a well known device in manufacturing. It can effectively measure physical geometrical characteristics of a product to examine the conformability of the dimensions with original design [1]. CMM supports very high measuring accuracy which is widely used in engineering design and product quality control. For most of products, dimension checking using CMM is an efficient and necessary approach in the product inspection process. However, the dimension checking process which affects the delivery date directly can be very time-consuming if CMM is improperly operated. The CMM path planning turns to be an efficient solution in improving inspection efficiency.

The CMM operation is a complicated processing. For some products with complex geometry, it is difficult even for experienced operators to design an efficient collision free inspection path. The efficiency of measuring process and accuracy of the measuring results cannot be guaranteed in this case. For different operators, different types of errors might occur due to different inspection methods used. It might lead to unreliable results [2]. Therefore a computer controlled measuring system is useful and necessary to verify and maintain the consistency of measuring result. This system should be able to generate an optimal inspection path automatically based on products' geometry characters. To achieve this goal, virtual reality (VR) technology and path planning theory are studied in this research.

Virtual reality (VR) is a technology that allows users to be immersed in a simulated, three-dimensional world which is generated by the computer and other special hardware [3]. Nowadays, VR technology is widely used in many areas around the world. Because of VR's capability to create immersive environments, many VR applications have emerged in the areas of entertainment, education, medicine, military simulation, and engineering [4]. Although VR can be applied in many areas, one of its most common uses is in assisting design engineers in industries [5]. Using VR technology to build a virtual CMM operation system can lead to another advantage for industries. Because of VR's capabilities to create highly virtual 3D environments, many organizations have used this technology to build a VR - based learning framework intended to train employees.

At the beginning, VR training simulator was used when the product was expensive or the operation process was too dangerous to allow trainees to operate the actual equipment in the real world. As the VR technology becomes more and more popular, some companies began to investigate the capability of VR technology as an auxiliary tool to aid distance learning or online training [6]. There are several advantages of using VR as a training tool for distance learning. Firstly, VR provides intuitive impression of operation process and allows interactive learning for trainees [7]. The original printed training documents are plain and non-intuitive. Even though the video record technique can give trainees an image of the real operation process, it does not allow trainees to interact with the operation system. It is difficult for the trainees to understand the training tutorial without practice. In contrast, VR allows trainees to actually manipulate a virtual model and experience the simulation outcomes. Furthermore, if the VR environment provides interactive facilities such as data glove and HMD, the VR-based training framework will be much more realistic and similar to real world operations. Secondly, the VR training framework provides great flexibility of training schedule and location for industries and trainees. Trainees can decide their own training schedule to learn the training tutorial by their own progress [8]. Thirdly, the risk of damaging the equipment can be avoided efficiently. Especially for CMM, the probe sensor system is very sensitive and easily damaged. If trainees can practice the CMM operations in a virtual environment (VE), the maintenance cost can be effectively reduced [9]. According to the benefits mentioned above, the VR-based training framework turns to be a powerful educational tool for industries.

This research aims to build a VR-based CMM training system to improve CMM inspection efficiency. Combining the VR technology and path planning methods, an automatic CMM path planning system could be developed in the virtual environment.

1.2 Research Objectives

The objective of this research is to develop an automatic CMM inspection and path planning system in the VE. It includes two parts: the inspection system and path planning system. The

Virtual CMM system simulates the real CMM operating processes and allows users to experience the outcomes of the measuring process. It should be able to generate an optimal inspection path and examine the path in the VE to improve CMM inspection efficiency.

The optimization algorithm used in the path planning system minimizes the total travelling distance in CMM measuring process. This algorithm will be implemented and verified in a VE-based CMM system which is developed under modeling and simulation software (Eon Studio). The optimization algorithm includes two parts：sequencing and collision avoidance. With the built-in collision detection function in Eon Studio, all unnecessary collisions between the probe and products are detected in the Virtual CMM system. A commercial software Lingo is used in this research to solve an integer programming model.

To simplify the problem, three assumptions are proposed: 1). Fixtures which fix workpieces on the CMM operation table are neglected. 2). Tolerance is not considered in this research. 3). The CMM probe is treated as a spherical object. It simplifies the collision detection problem.

This research develops a VR-based CMM system for industry training and CMM path planning. A virtual CMM environment which allows users to learn and practise the CMM operations is built. A new algorithm which combines an integer programming model and the big penalty M method is proposed to optimize the CMM measuring efficiency.

1.3 Thesis Outline

Chapter 2 reviews the background of virtual CMM and related technologies. The history of virtual reality technology is reviewed to understand the applications of VR technology in manufacturing field. The CMM path planning algorithms are discussed in Chapter 2 to identify the limitations and benefits of each algorithm. Various collision detection methods are evaluated to meet the virtual CMM inspection system need.

Chapter 3 introduces the process of using CAD and VR software to build the virtual CMM environments. The 3D CMM model is built using AutoCAD and imported into the VR simulation software Eon studio. The virtual environment, movement of the CMM probe, and collision detection between probe and workpiece are implemented in Eon studio.

Chapter 4 introduces a path planning algorithm to improve the current CMM inspection path. Integer programming model and the big penalty M method are combined to generate the optimal inspection path. The algorithm is evaluated and validated in this Chapter.

Chapter 5 concludes the thesis and indentifies the contributions of this research. Recommendation and future works are included.

Chapter 2 Literature Review

2.1 Virtual Reality

Virtual Reality (VR) is a technology that allows users to be immersed in a simulated, three-dimensional (3D) world which is generated by the computer and other special hardware [3]. This technique has developed in many ways to become an emerging technology of our time.

VR's history can be traced back to 1960s. In 1960, a cinematographer named Morton Heilig built a single user console called "Sensorama" [10]. Using Sensorama, audiences could experience the impact of visuals, sounds, vibrations and smells simulated by the VR device. Unfortunately, it was not a big commercial success. In 1961, Philco Corporation developed the first head-mounted display (HMD), called Headsight [10]. It had a video screen and tracking system linked to a closed circuit camera. Users could observe a real environment remotely. This device was developed for use in dangerous situations such as helicopter pilots flying at night. In 1966, a computer scientist name Ivan Sutherland designed the first HMD driven by computer graphics [10].

In VR's history, one of the most influential antecedents of virtual reality was the flight simulator. From the mid 1980s, NASA, the Department of Defense and the National Science Foundation invested millions of dollars into VR technology to simulate flying airplanes [10]. Figure 2.1 shows a U.S. Naval soldier using a VR parachute trainer. The VR program tracks the user's motions, particularly his head and eye movements, and correspondingly adjusts the images on the display to reflect the change in perspective. In 1989, Jaron Lanier coined the term "Virtual Reality" [11]. In 1995, a company named Silicon Graphics Inc. (SGI) developed VRML 1.0 (Virtual Reality Modeling Language), which is an advanced tool for the modeling of 3D worlds that are functional and interactive [12].

Figure 2. 1 U.S. navy personnel using a VR parachute trainer [4]

With the rapid development of CAD software and graphic hardware acceleration, virtual reality is widely used in many areas such as new product prototypes testing, manufacturing process simulation, and even interactive computer games [13]. For engineering concern, VR technology is studied as an advanced auxiliary tool to aid and improve the current CMM inspection system. Hu and Yang [1] developed a virtual coordinate measuring machine (VCMM), which simulates CMM's operation and measurement process in a virtual environment. It enables CMM off-line programming and also able to perform error analysis and uncertainty evaluation. Zhou et al [14] designed a 6-DOF parallel-link CMM using the graphics functions in the OpenGL library. They implemented the Aligned Axis Bounding Box (AABB) algorithm to detect objects collisions in the virtual environment. Yang and Chen [15] developed an inspection path generation methodology for haptic virtual coordinate measuring machines (HVCMM). It also enables CMM off-line programming as operators were sitting in front of a real CMM by using haptic device. Teaching programming is made by pointing a cursor at 3D CAD model to generate the inspection path of a part. Collisions can also be checked in the inspection simulation.

2.2 CMM Path planning

2.2.1 Introduction

CMM path planner is a popular research topic to improve the CMM operating efficiency. There are many challenges in building a fully automated inspection planner. An inspection planner should be able to capture the geometry characters of measuring objects, to identify the objects features, and to plan the inspection path. Lin and Murugappan [16] proposed an inspection path planning system that consists of five modules shown in Figure 2.2:

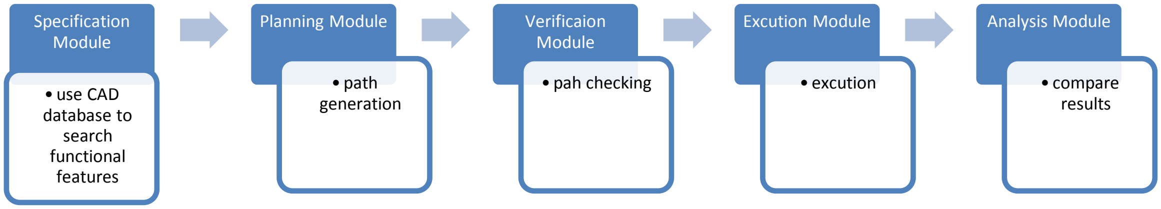


Figure 2. 2 Inspection planner modules

In physics concern, Li et al [17] outlined that the inspection path should be consisted of inspection points, collision avoidance point, back off points, probing vectors, position vectors, and back off vectors. Figure 2.3 shows a way to locate the points and vectors around the workpiece. In Figure 2.3, 1 and 2 are inspection points. B and D are collision avoidance points. A and C are back off points. Usually, the magnitude of the probe vector and back off vector are very small and steady in the measurement. To reduce the total measuring time and improve efficiency, the inspection path and moving velocity of the probe motion from A to B is the key factor.

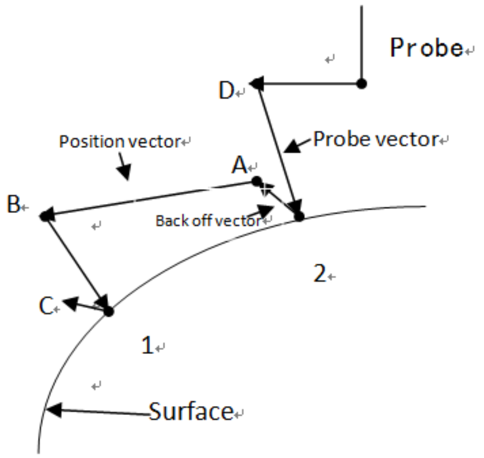


Figure 2. 3 Inspection points and motion vectors [17]

The distribution of inspection points can also affect the inspection result. Zhao [18] et al discussed the relationship between the number of inspection points, the measuring precision and the measuring efficiency. They believed that greater measuring points will lead to better precision. However, it will also increase the cost and time-consuming. The balance of the weight of these elements should be considered in designing an inspection path. They also proposed that the distribution of measuring points should follow the following constraints: 1) uniform distribution, 2) cover the whole surface as much as possible for inspection quality, 3) avoid holes and projectures, 4) avoid edges and boundaries.

CMM path planning is similar to a well-known the traveling salesman problem (TSP). As same as the salesman, the CMM probe has to travel to every inspection points of a product to collect dimension data. Therefore many efficient TSP solutions can be implemented in CMM path planning. Meanwhile, CMM path planning is not only a single TSP problem. The geometry of the product and fixtures can create complicated obstacles that interfere with the movement of the probe. Therefore the path planning problem can be divided into two sub-problems: sequencing and collision avoidance [19]. Sequencing focus on the order of measuring points to minimize the total probe travelling distance which is a typical TSP problem. An optimized and feasible measuring sequence will improve the efficiency of CMM operations and save total production time and cost. Collision avoidance ensures a collision-free path between any two adjacent measuring points in the optimized path. A proved collision free path which avoids unnecessary collisions between the probe and the products will extend the lifespan of probe and sensor system. Based on the benefits mentioned above, an efficient CMM path planning algorithm which helps CMM users to generate a collision-free, near optimal inspection path for CMM operations is very useful and necessary to study.

2.2.2 Collision Detection

Collision detection plays an important role in CMM path planning. The theory is similar to the Cartesian measurement robot problem [20]. The robot system detects the collision object then generates a collision free path to pass the obstacle. However, CMM collision avoidance is more complex. The basic objective of using CMM is to measure the geometry characteristics of a workpiece. The CMM probe needs to move around the workpiece and collect measuring data. Therefore in the virtual environment, the CMM path planning system cannot treat the workpiece as a single object like the Cartesian measurement robot system. All the important features of the workpiece need to be analyzed and measured. Therefore, CMM collision avoidance is more difficult to implement.

Lin and Murugappan [16] used a ray tracing technique to locate the collision of possible paths between the initial point and the target point. They proposed an algorithm to generate an optimum collision-free CMM probe path. In their theory, an imaginary ray is shot from the initial point to the target point (see Figure 2.4 for example). If there is no intersection of the ray with the workpiece geometry, the collision-free path will be a straight line. If there is an intersection, the midpoint of the edge shared by the intersection face and the adjacent face nearest to the target point will be select as the intermediate point. Then the initial point, intermediate point and the target point will be connected as the final path. If the target point is not found on the adjacent surface which means there is still an intersection, taking the endpoint of the first ray as initial point of the new ray and another midpoint of edge is selected as the new intermediate point. The final collision-free path can be found when there is no intersection between the imaginary ray and the workpiece. This algorithm can successfully develop a collision-free inspection path. However, our goal is to generate the optimal path. The midpoint of the edge nearest to the target point is not the optimum point. Besides, this algorithm cannot be applied on any objects with the curved face.

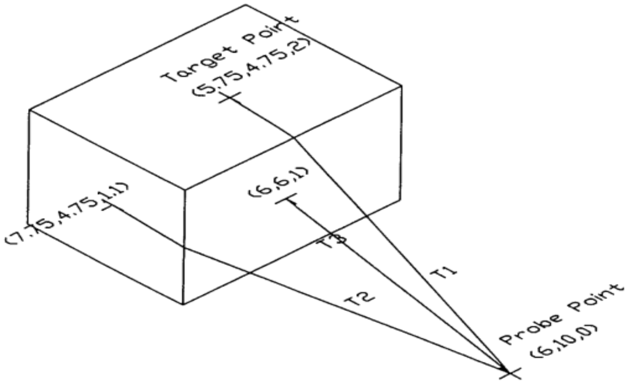


Figure 2. 4 Ray Tracing [16]

Lin and Chen [21] proposed a cut face method to avoid the probe collision. They addressed the minimization of measuring points and the desire for normality of the probe to the surface being inspected. Yang and Chen [15] proposed a very interesting approach by using haptic device to detect collisions. With the proper haptic device, users can experience the force feedback when the collision occurs between the probe and the workpiece.

Boundary box is another widely used method for collision avoidance. Its category falls into Sphere, AABB (Aligned Axis Bounding Box), OBB (Oriented Bounding Box), K-DOPs (Discrete Orientation Polytope) etc. Zhou et al [14] designed a virtual 6-DOF parallel-link CMM using AABB for the probe collision detection. Liu et al [22] proposed an algorithm for collision free path planning by using dual AABBs. Figure 2.5 is an example of AABB. The boundary of AABB is always parallel to the axis. The boundary size is determined by the projection of objects on the corresponding coordinate plane. Two AABB are intersectant if and only if their projections are overlapping on all three–axis. Total six comparisons are needed for the collision detection. AABB is proved to detect collisions between the probe and workpiece effectively.

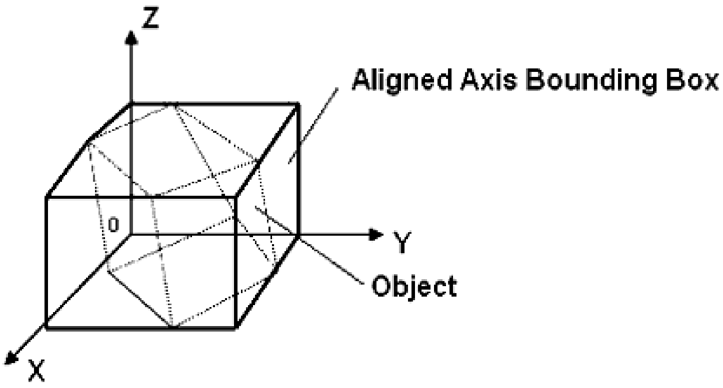


Figure 2. 5 AABB schematic [14]

For the collision detection, Chen et al [23] indicated that the selection of collision avoidance points should also consider the reality factor. A gap shown in Figure 2.6 between the probe body and the workpiece surface is necessary to protect the probe from the collision damage in real CMM operations.

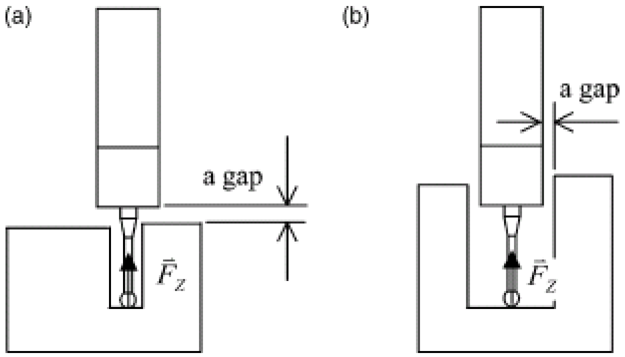


Figure 2. 6 Gaps to avoid collision [23]

2.2.3 Measuring Sequence

Without the collision avoidance concern, CMM measuring sequence is very similar to a TSP problem. TSP is a typical NP-hard optimization problem which has been studied for a very long time. There are many existing algorithms which can be used to solve the TSP problem: such as integer programming, simulate anneal arithmetic, heuristic algorithm, and genetic algorithm.

Lin and Chen [21] used the nearest neighbor method and refinement method for CMM path planning. For n points to be measured, the probe starts from the origin point and move to a nearest point. The moving distance is calculated. The probe keeps moving to the next nearest point until all of points to be measured. A near optimum solution can be achieved by comparing n total distance based on different origin points. The programming formulation is listed below [24].

Let,

$X_{ij} = 1$ if the probe travels directly from point i to point j,

$X_{ij} = 0$ otherwise;

And,

$D_{i,j}$ = the distance from point i to point j,

Then,

P= Minimizes $\sum_{i=1}^{n} \sum_{j=1}^{n} D_{i,j} X_{ij}$,

$\sum_{j=1}^{n} X_{ij} = 1$, i=1,2……n

$\sum_{i=1}^{n} X_{ij} = 1$, j=1,2……n

Limaiem and ElMaraghy [25] also used the nearest neighbor algorithm combined with collision detection routines to build a path planning module of the Computer-Aided Tactile Inspection Planning (CATIP) system. Lin and Chow [26] introduced a near optimal measuring sequence planning system by using a dynamic programming method. They created a measuring points distance matrix to solve the shortest inspection path. Lu et al [27] developed an integer programming model to minimize the probe traveling distance. Dummy points were added in their model to avoid collision. The optimal inspection path can be calculated by using software such as LINDO, LINGO or Matlab. Yau and Menq [28] designed a hierarchical planning system using a heuristic algorithm. Their path planning system deals with the complex parts having sculptured surfaces. Lu and Morton [27] used the genetic algorithm as an optimizer for inspection path planning systems. They compared genetic algorithm with traditional teaching method. The total travelling distance has been improved by 26-33% in their test. They conclude that genetic algorithms is more powerful for multitask inspection problems with large number of measuring points based on the result of optimization.

Measuring sequence is the key element for optimization of the inspection path planning. Generally, it can be divided into local path planning and global path planning. Local path planning focuses only on the measuring sequence on one feature face. The goal of Global path planning is to generate an optimal arrangement and combination of all local paths [26]. For example, to measure the radius of a hole from a product, total six points will be inspected. The measuring sequence of these six points will be determined by using local path planning. However, there are usually more features need to be inspect for one product, such as radius of another hole, diameter of a column, thickness of a plane. CMM only can deal with one feature inspection at each time. Therefore globe path planning is useful here to arrange measuring sequence and connecting points for all feature faces. For this measuring path optimization problem, three following factors must be considered: measuring sequence on a single feature face; measuring sequence of all feature faces; connection points between two faces [29]. These three factors influence each other and make the sequencing not only a Traveling salesman problem (TSP). The problem solving process is much more completed than the process of TSP. Figure 2.7 is a feasible arrangement for globe path planning. In this Figure, local path planning was implemented first. Then the starting and end points of each feature were picked as the connecting points for global path planning. In this research, only local path planning is considered for optimization. The global path planning will be leaved for future work.

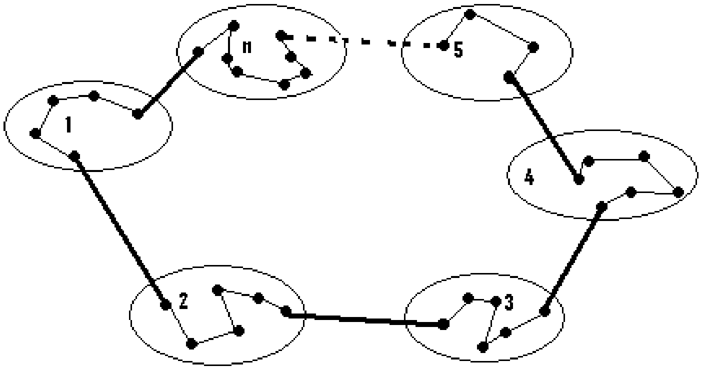


Figure 2. 7 Globe Path Planning

Generally, there are two ways to generate an optimal collision free inspection. One is to optimize the measuring sequence first, and then examines the feasibility of the path by using the collision detection. If a collision is detected, it will return to the sequencing step and add more intermediate points to avoid the collision. Another method is to examine the collision first, then performs the optimization of inspection path. The first method is the most common method right now. Since the second method usually takes more time at the beginning when the geometry of parts and fixtures is complex, the first method is more common to use. No matter which method is applied, both methods require two separate steps to solve the problem. Buchal and Wang [30] proposed a new idea by adding a big penalty value M to the path segment with collisions. This proposal combined two steps into one and might be an effective solution to improve the path planning result.

There are many path planning algorithms that can be applied in virtual CMM systems to improve the CMM measuring efficiency. However, none of these algorithms can guarantee an absolute optimal solution. Implementing the theories into computer program requires lots of further detailed work and programming knowledge. Therefore in this research, the algorithm which fits with the simulation software most, should be considered for the virtual CMM system.

2.3 Chapter Summary

This Chapter reviews two virtual CMM system related technologies: virtual reality and path planning. Virtual reality is a rapid developing technology which is wildly used in engineering design and manufacturing production. The advanced VR technology and VR equipments can successfully satisfy the CMM VE simulation requirement. An appropriate simulation tool should be selected in this research to build the VE for CMM operation. Besides the VR technology, various approaches for path planning are discussed in this Chapter. For measuring sequence optimization, the nearest neighbor, integer programming model, hierarchical planning, and genetic methods are evaluated for CMM inspection optimization. All the existing virtual CMM systems I found have their own limitations. He and Yang's VCMM can simulate the CMM operation process in virtual environment without the collision avoidance concept. Zhou et al successfully implements the collision detection function using AABB method in their program not the path planning. Other systems either only focus on the CMM path planning algorithm without VR simulation or their path planning algorithm does not fit with the collision detection program. Therefore, a virtual CMM which can simulate the CMM operation using advanced VR technique and optimize the inspection path with the collision avoidance function is significant to study.

There are some difficulties found in this research. First, the CMM path planning is not a 2D problem like TSP. To develop an optimization program and drive the CMM probe moving in the 3D environment, many more variables have to be considered. Second, the measuring sequence problem is similar to a TSP problem. However, combined it with the collision avoidance issue, the problem becomes much more complicated. The path planning system needs to identify all the collision points then avoid them in the optimization process. It will add many extra constraints to the optimization problem. The collision signal transition between the collision detection program and path optimization system is also important and difficult to solve.

In this research, Eon studio is selected to simulate the CMM operation in VE. An integer programming model is used for inspection path optimization. The big penalty M method is used to avoid unnecessary collisions during CMM measuring process. Combining virtual reality technology with integer programming model and the big penalty M method, the virtual CMM system should be able to simulate the CMM operation process and to generate optimal collision-frees inspection paths.

Chapter 3 Building a Virtual CMM System

Generally, there are two ways to build a virtual CMM system. One is using the graphic transform function to build the Virtual CMM. Some language tool such as Virtual C++ is required to perform the simulation of VCMM [14]. Another method is using existing modeling software to build the model of CMM and import it into the chosen simulation software. Advanced applications can be achieved by using script language tools.

Comparing these two methods, the second one is more users friendly. With the rapid technology development, the modeling software such as Pro-E, Solidworks and AutoCAD's virtualization effect and compatibility with simulation tools are also improved significantly from yearly updates. Furthermore, nowadays simulation tools are more powerful with some built-in functions. The simulation tools come with some remarkable functions like the collision detection to make the simulation process much easier for designers. This research develops a Virtual CMM system using AutoCAD and Eon studio. Figure 3.1 shows the three major steps to build a virtual CMM system.

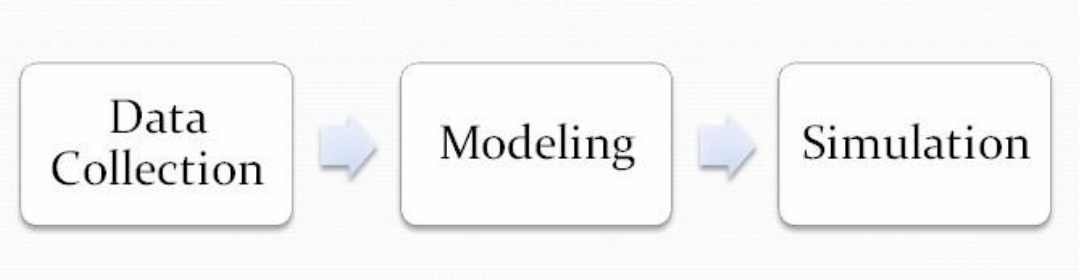


Figure 3. 1 Major Steps

The first stage to build Virtual CMM is to construct a virtual model of CMM. Various types of CAD software can be used to aid industry design nowadays. No matter which software is chosen, the modeling processes should be substantially similar. The intention of building the CAD model is to transform the model into the VR system and then implement the simulation. Therefore the complexity of the CAD model should be carefully considered at the beginning. All the requirements and the constraints of the Virtual CMM should be considered to avoid unnecessary future revision work.

Eon Studio is a VE development tool for building interactive VR applications [31]. It provides program primitives for the virtual object moving in the VE. The 3D model has to be built using other CAD programs then imported into Eon Studio. Eon Studio supports various types of import data formats such as .dxf (AutoCAD), .3ds (3D Studio), .slp (Pro/E) and .wrl (VRML) [31]. The geometric models are managed using a tree structure to define their spatial relations, which enables the models to be manipulated in VE. The simulation routes and controls are programmed using Eon VB Script.

3.1 CMM Modeling

In this research, Checkmaster Model 216-142 DCC is used as the CMM model. Figure 3.2 shows a picture of the CMM which is located in our research lab. All the dimension data of CMM were collected using ruler and caliper. The measurement accuracy is between 0.00015-0.00029 inches in X, Y, Z dimensions. AutoCAD is used building the 3-D model of CMM. All the components are exported in 3ds format in AutoCAD then imported into Eon Studio.

Figure 3. 2 CMM prototype

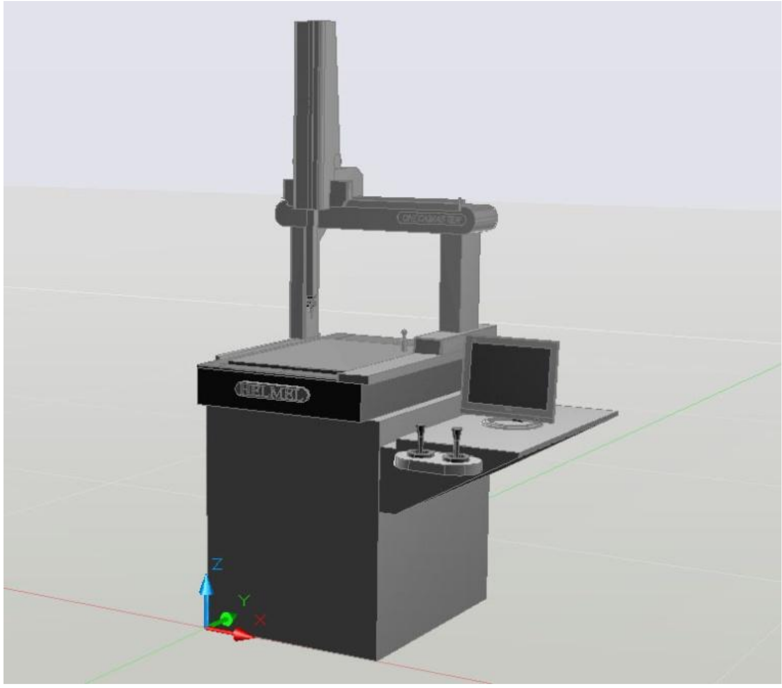


Figure 3. 3 3D - CMM Model

Figure 3.3 shows the 3D model of CMM built in AutoCAD. To avoid any unnecessary revision and save modeling and simulation development time, the CAD model only includes necessary components. Generally CMM consists of three parts: main structure which builds x, y, z axles; probing system and data collection system [32]. In this project, some CMM components such as the wire connection, screws, and the desktop computer, are not included in the 3D model. During the conversion process, Eon studio generates an independent mesh for each component, any unnecessary parts will complicate the design process and slow down the data processing speed [33]. The parts which not related to the probe motion can be treated as one object, such as the measuring table, the monitor. Meanwhile, the model still needs to be constructed as close to the actual CMM as possible. It is very important to develop accurate models to be able to response the operation outcomes of the VR system [34]. Figure 3.4 shows the most important part of CMM: the probe. To ensure the further folding and rotating capability in VE, the probe is built consisting of 11 parts. Total 67 parts are modeled to compose the CMM 3D model.

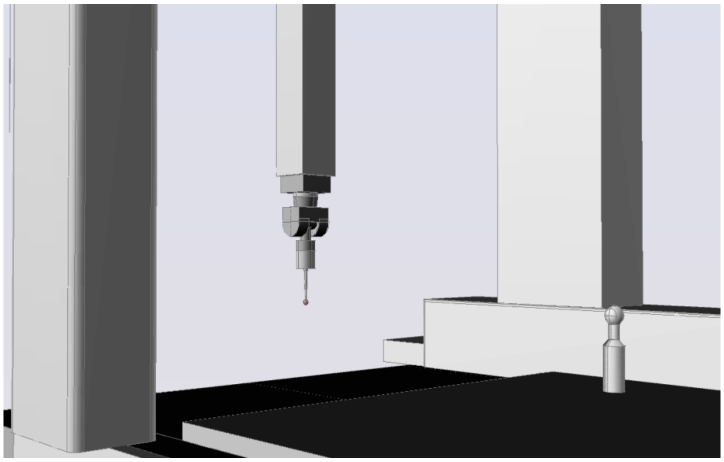


Figure 3. 4 AutoCAD model--- Probe

3.2 CMM Operations Modeling

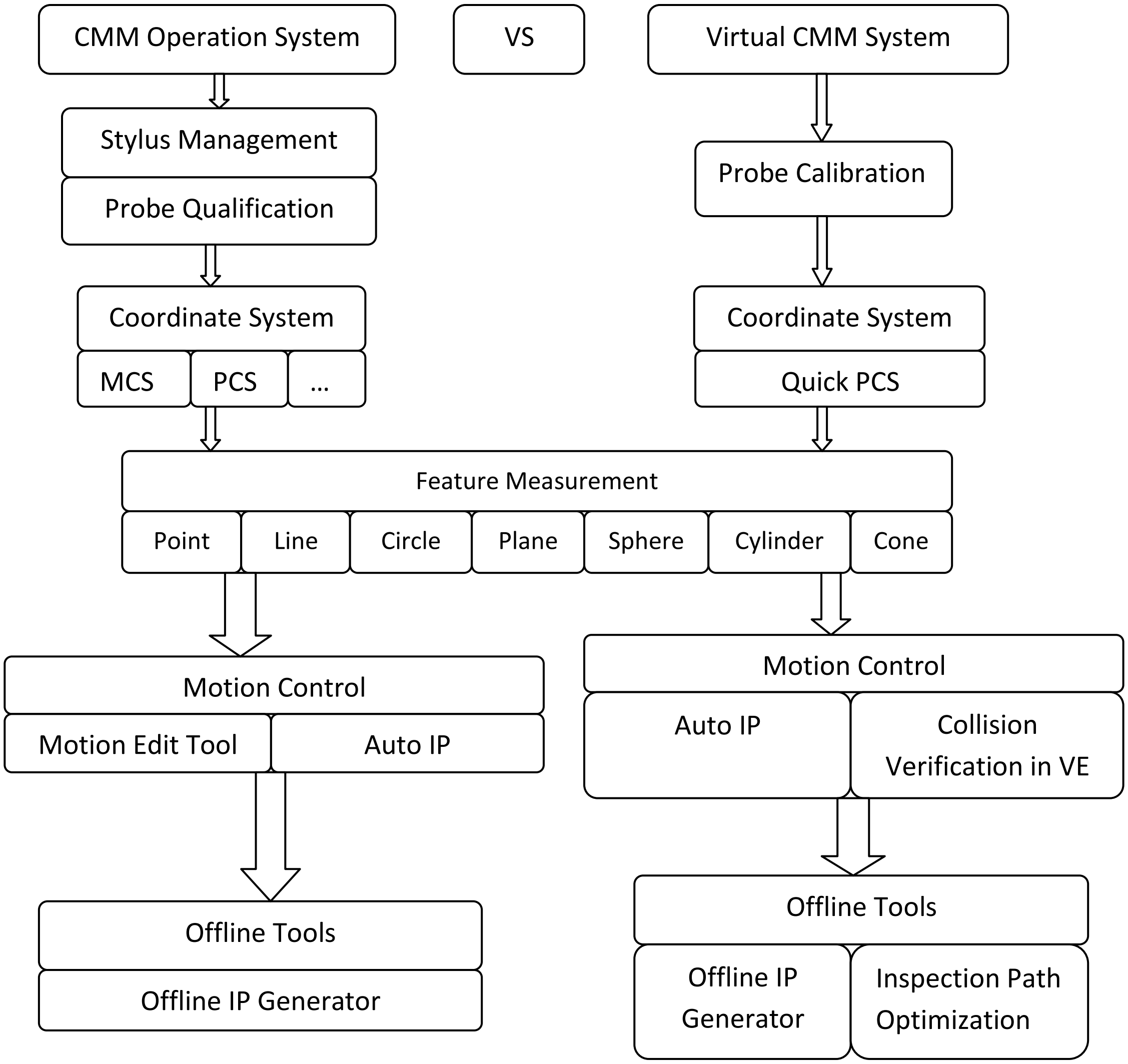
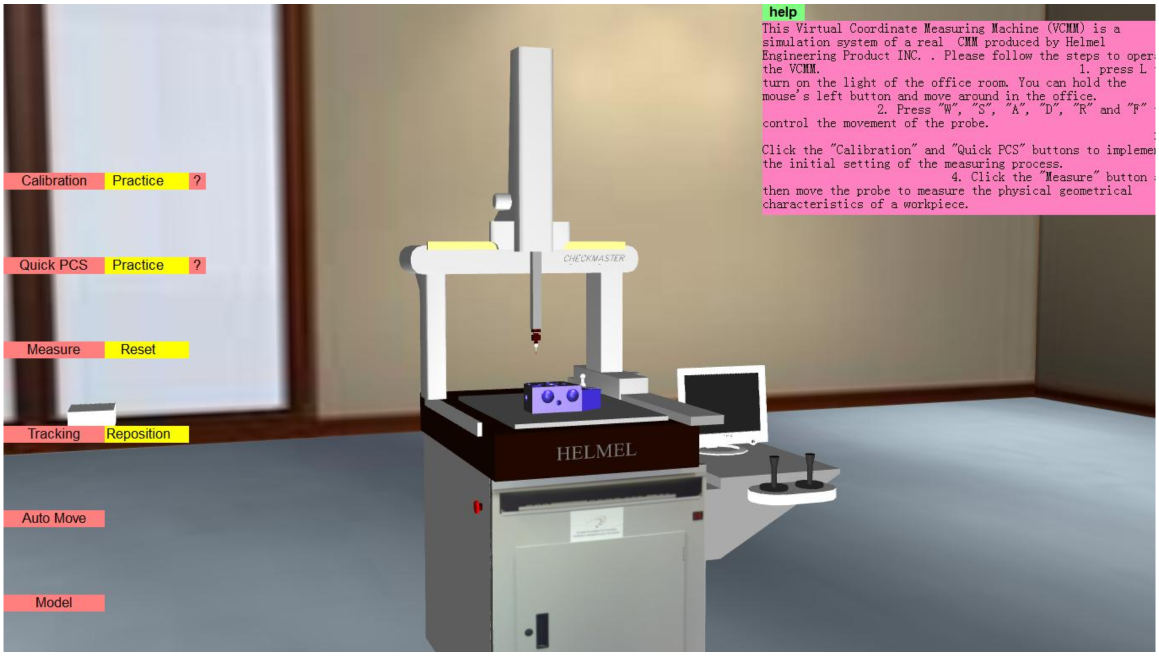


Figure 3. 5 CMM Operation System VS Virtual CMM Operation System

The virtual CMM system simulates most of basic CMM operation functions. Figure 3.5 shows a flow chart of CMM operation system VS virtual CMM operation system. The virtual CMM system simplified some of the functions, such as stylus management and machine coordinate system (MCS). Both systems allow auto interim point (IP) mode for direct computer controlled (DCC) measurement. Two advanced functions are added to the virtual CMM system: offline collision verification and inspection optimization.

Figure 3.6 shows a picture of the simulated CMM model in the virtual environment. The operation menu includes Calibration, Quick PCS, Measure, Teaching program (tracking) and Auto Move. “Calibration” function determines the radius of the probe tip selected in the measurement. “Quick PCS” establishes a part coordinate system on the object. The selection of “Measure” allows probe move around the part and collect measuring data. “Tracking” and “Auto Move” function allows users to record the probe moving path and repeat it on the identical product.



This Virtual Coordinate Measuring Machine (VCMM) is a
simulation system of a real CMM produced by Helmel
Engineering Product INC. . Please follow the steps to oper
the VCMM.   1. press L
turn on the light of the office room. You can hold the
mouse's left button and move around in the office.
2. Press "W", "S", "A", "D", "R" and "F"
control the movement of the probe.

Click the "Calibration" and "Quick PCS" buttons to implemen
the initial setting of the measuring process.
4. Click the "Measure" button
then move the probe to measure the physical geometrical
characteristics of a workpiece.

Figure 3. 6 Virtual CMM Interface

3.2.1 Model Importing and Visualization

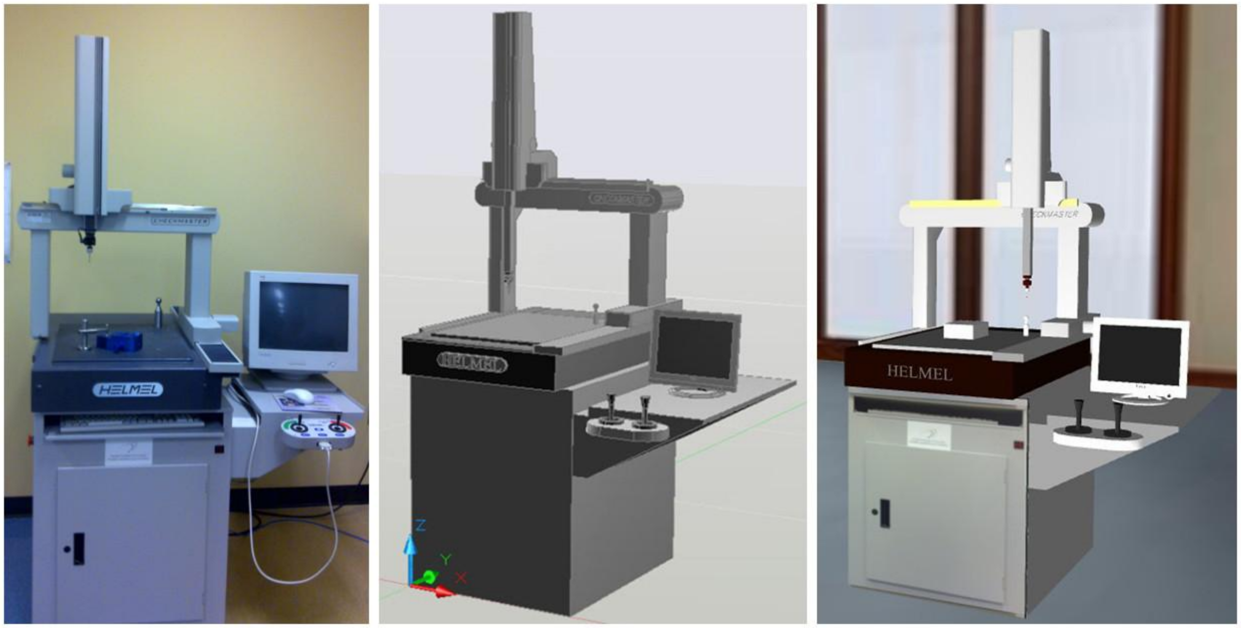


Figure 3. 7 CMM-Model-VE

The first step of simulation is to transform the CAD model into the virtual environment created by Eon Studio. All these 67 parts of CMM are imported into the Eon studio to be assembled together in the VE. Figure 3.7 shows the interface of CMM model in different stages. Several functions program in Eon were used to make the simulation result more realistic. For example, the texture mapping function can effectively improve the visualization effect. As Figure 3.8 shows, a cube plus a picture of the real model can generate a material object in the virtual environment.

Figure 3. 8 Texture Mapping

3.2.2 Probe Motion

To implement all the operation functions of virtual CMM, we must implement the movement of the probe first. When a 3D model of CMM is imported into Eon, all the components that moves in the same direction should be placed under the same folder. For example, all the components which need to be able to move up and down (z direction) are placed under the same frame called Up&Down. However, the probe is not only moving upward and downward but also left and right. Therefore, the Up&Down frame should be a children frame of Left&Right frame. To import all the parts into its corresponding folder correctly, each part belongs to the same motion group should be placed into an individual layer in the AutoCAD program. Then all the parts belong to same motion group can be exported based on the layer name then import into Eon Studio. Figure 3.9 is the simulation tree of the virtual CMM which shows the dependence relations between frames.

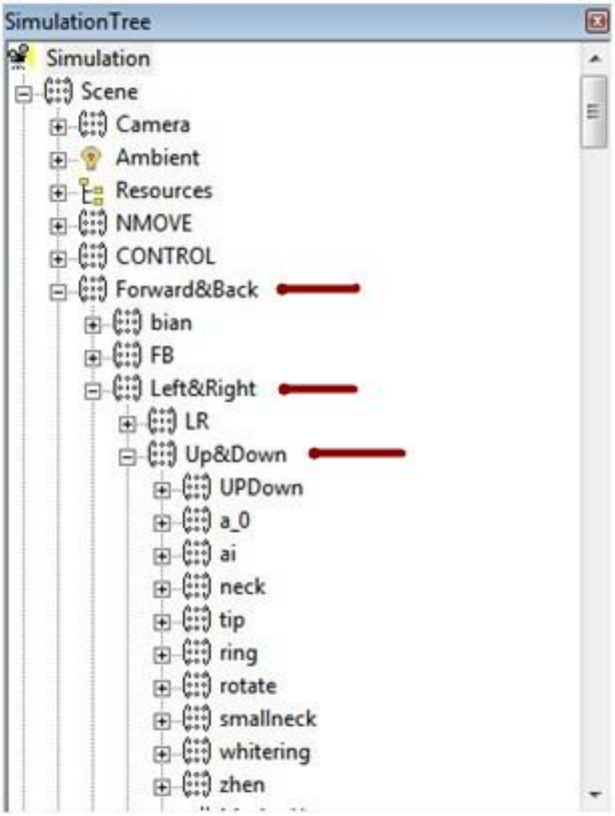
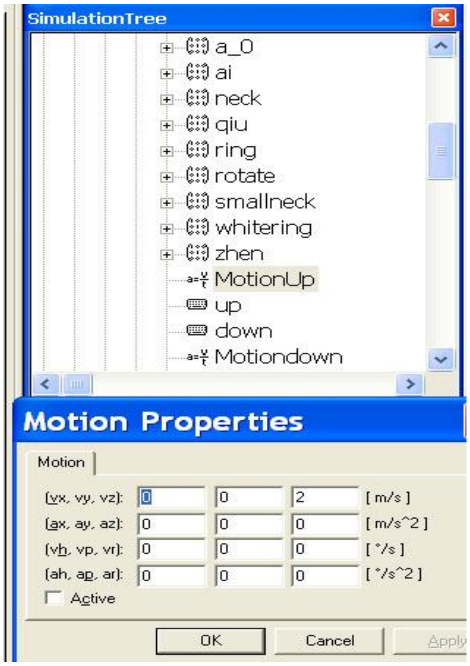


Figure 3. 9 Simulation Tree

Eon Studio provides many powerful programming nodes which allow users to implement the interactive functions and enhance visualization effect. To implement the probe movement, six motion nodes are dragged into the simulation tree under a corresponding folder. The moving speed is set in the property window to active the movement of probe system. Figure 3.10 shows a picture of a motion node under the simulation tree and the motion property window.



(vx, vy, vz): 0 0 2 [m/s]
(ax, ay, az): 0 0 0 [m/s^2]
(vh, vp, vr): 0 0 0 [°/s]
(ah, ap, ar): 0 0 0 [°/s^2]
☐ Active

Figure 3. 10 Motion node

Figure 3.11 shows simulation routes of the Virtual CMM system. The six motion nodes are connected in the routes window with the corresponding keyboard sensor node.

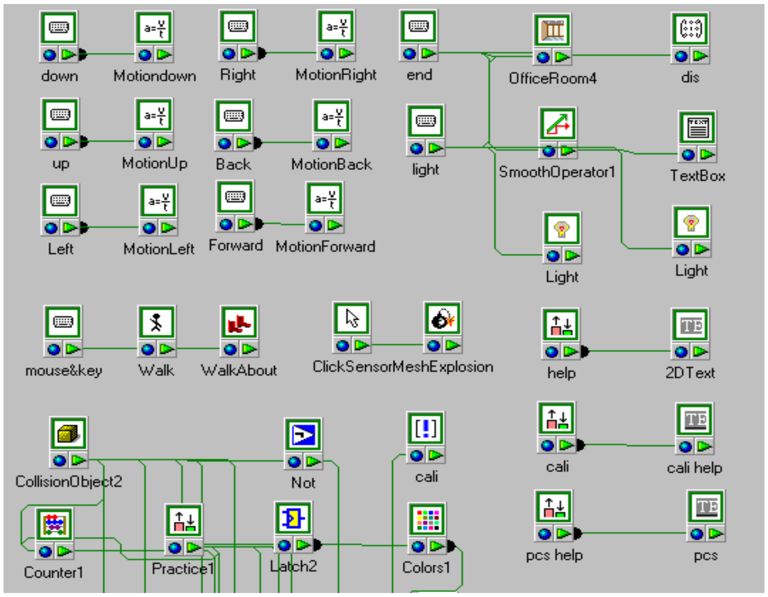


Figure 3. 11 Routes window

Table 1 shows the relations between the keyboard sensor nodes and motion nodes in the routes window. The proper connections ensure that users can control the probe movement by pressing the corresponding key.

Table 1 Routes command

Source node	**Out-field**	**Destination node**	**In-field**
Down	OnKeyDown	MotionDown	SetRun
Up	OnKeyDown	MotionUp	SetRun
Down	OnKeyUp	MotionDown	SetRun_
Up	OnKeyUp	MotionUp	SetRun_

3.2.3 Calibration and Collision Detection

Calibration is the very first step to start CMM measurement. The radius of the probe tip has to be deducted to calculate the final measuring result. To implement the calibration function, the probe has to collide with the positioning sphere five times from different directions. However, without a collision function, when users run the simulation and control two objects passing through each other, an overlap will occurs as shown in Figure 3.12.

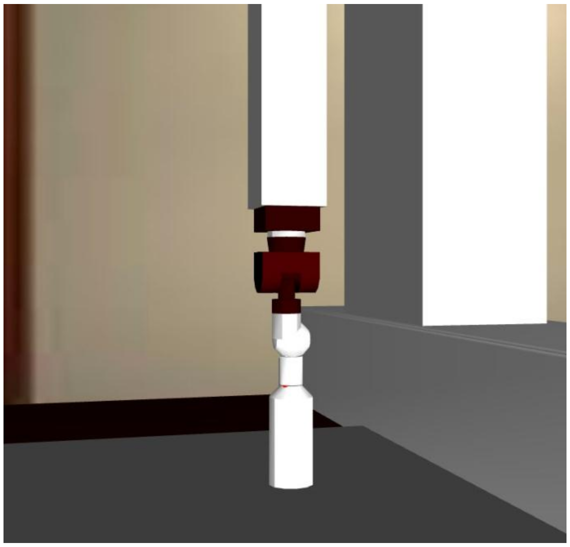


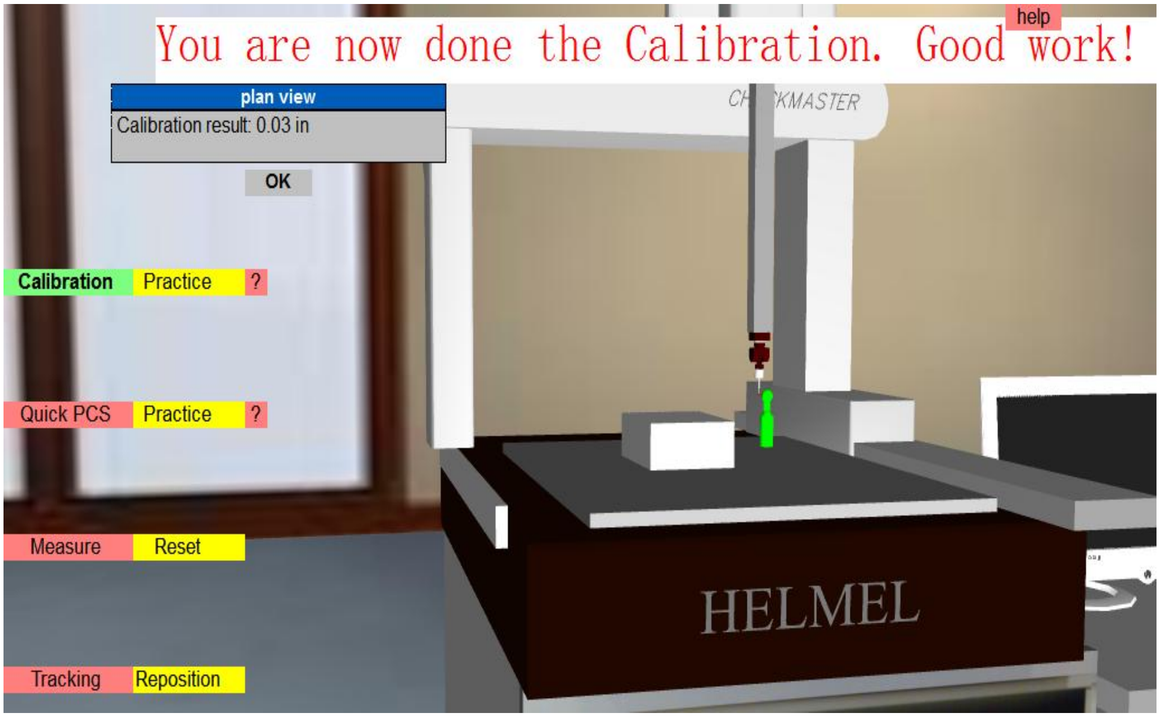
Figure 3. 12 Overlap

Eon’s built-in collision detection function, which allows two objects to physically contact each other, is a very convenient way to solve the overlap problem. Comparing with the ray tracing or AABB boundary method used in the literature, it simplifies the problem and saves developers much programming time

To implement the collision detection function, following steps are taken.

1. Drag a group node into the simulation tree under the scene frame. Then drag a CollisionManager node under the group node.
2. Drag a CollisionObject node into the simulation tree under the Up&Down frame.
3. Drag CollisionObject nodes into the simulation tree under any frames which the probe is going to collide.

The probe will stop wherever other objects block its pathway. A DirectSound node is added to generate a beep when the collision occurs. The surface of the contact object is also set to change color when the collision occurs. The distance can be calculated from the collision positions of the probe during its movement. Figure 3.13 show the interface window of the calibration. Operation instruction and the data result are displayed in the window.



You are now done the Calibration. Good work!

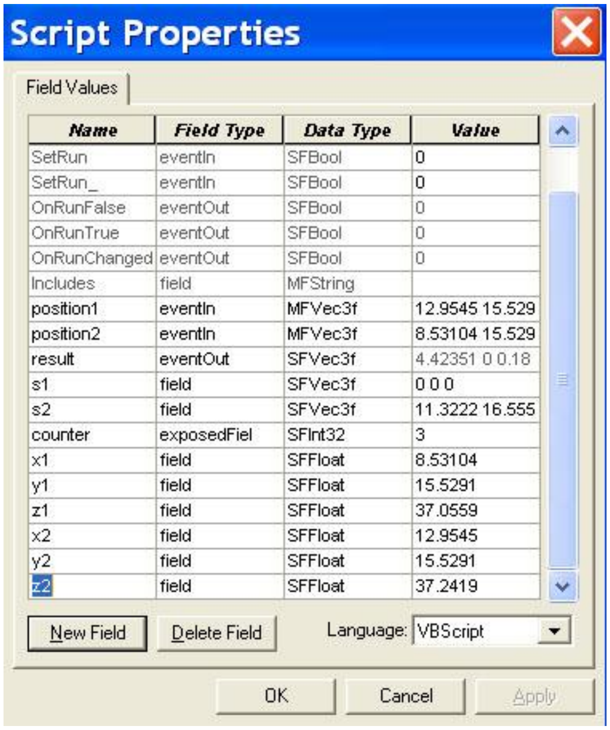
Calibration result: 0.03 in

Figure 3. 13 Calibration

3.2.4 Teaching Method

CMM's teaching method allows users to record the probe moving path and repeatedly measure identical products. There is no existing programming nodes in Eon studio can be used to implement this function in virtual CMM. VB script programming node is used to record the probe position data.

Script is a custom design node which allows users to create specific programming to meet supplementary requirements. Users have to define the “eventIn” and “eventOut” signal types then determine the relationship between script node and other function nodes. The field type and the data type as shown in Figure 3.14

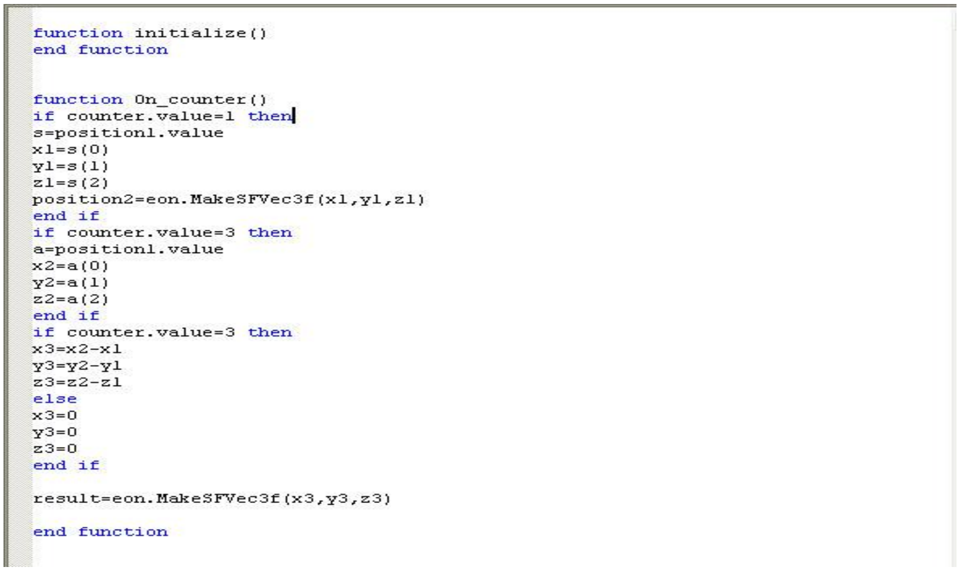


Name	Field Type	Data Type	Value
SetRun	eventIn	SFBool	0
SetRun_	eventIn	SFBool	0
OnRunFalse	eventOut	SFBool	0
OnRunTrue	eventOut	SFBool	0
OnRunChanged	eventOut	SFBool	0
Includes	field	MFString	
position1	eventIn	MFVec3f	12.9545 15.529
position2	eventIn	MFVec3f	8.53104 15.529
result	eventOut	SFVec3f	4.42351 0 0.18
s1	field	SFVec3f	0 0 0
s2	field	SFVec3f	11.3222 16.555
counter	exposedFiel	SFInt32	3
x1	field	SFFloat	8.53104
y1	field	SFFloat	15.5291
z1	field	SFFloat	37.0559
x2	field	SFFloat	12.9545
y2	field	SFFloat	15.5291
z2	field	SFFloat	37.2419

Language: VBScript

Figure 3. 14 Script Node

Figure 3.15 shows the script programming code composed using VB language. When each time key “P” is pressed, the position data of the probe will be recorded in the script node. Figure 3.16 shows the relations between script node and other keyboard nodes and position nodes. Once the Auto Move selection is activated, all the recorded position data will be exported one by one to the position nodes to repeat the saved moving path.



```
function initialize()
end function


function On_counter()
if counter.value=1 then
s=position1.value
x1=s(0)
y1=s(1)
z1=s(2)
position2=eon.MakeSFVec3f(x1,y1,z1)
end if
if counter.value=3 then
a=position1.value
x2=a(0)
y2=a(1)
z2=a(2)
end if
if counter.value=3 then
x3=x2-x1
y3=y2-y1
z3=z2-z1
else
x3=0
y3=0
z3=0
end if

result=eon.MakeSFVec3f(x3,y3,z3)

end function
```

Figure 3. 15 script

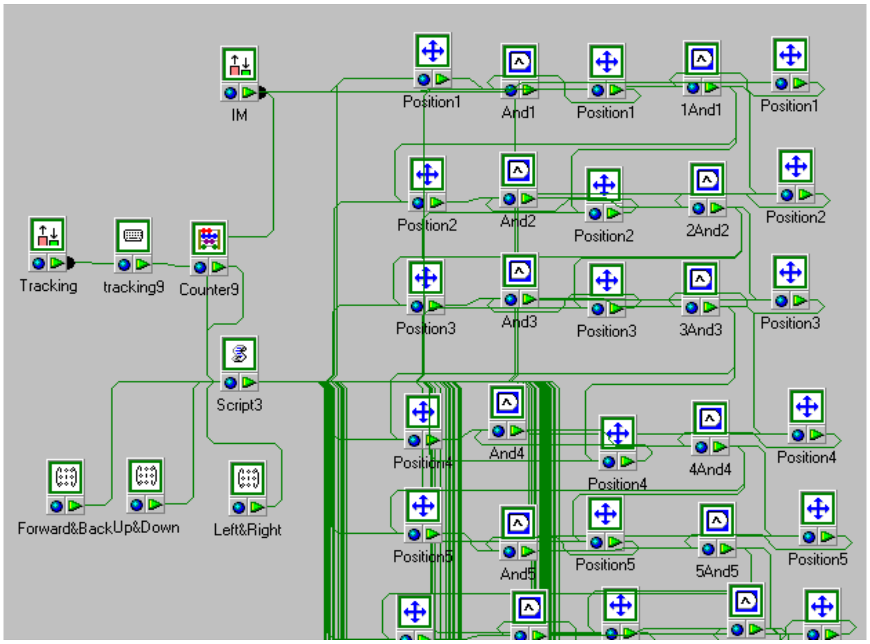


Figure 3. 16 Routes of Tracking

3.3 Chapter Summary

The part of this Chapter was published in proceedings of the ASME 2010 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference. A 3D CAD model of the CMM is built using AutoCAD and the basic structure of a virtual CMM system is built using Eon Studio's built-in functions. The difficulty of this part is to learn a new software: Eon Studio. Collision detection theory and Virtual Basic (VB) language are essential to study to implement the CMM simulation. The virtual CMM system at this stage allows users to perform all the basic CMM operation functions like the real world. There are several advantages for using this virtual CMM as a training tool. First, it can effectively reduce the possibility of unnecessary collisions between the probe and products. It protects the sensor system then reduces the maintenance cost. Secondly, the training and collision free verification can be done offline in the virtual environment. The machine utilization is improved. Thirdly, virtual CMM system provides great flexibility of training schedule and location for industries. Users can adjust their training schedule based on their own progress. Finally, compared with the traditional video and paper training documents, virtual CMM allows users to actually manipulate a virtual model and experience the simulation outcomes. It helps users to better understand the CMM operation concept. In conclusion, the virtual CMM system using advanced VR technology is a qualified training tool for industries. However, to improve the CMM inspection efficiency, the path planning theory should be studied and implemented into the virtual CMM system.

Chapter 4 CMM path planning

4.1 Path Planning Procedure

Based on the literature review, the CMM path planning includes two parts: measuring sequence optimization, and collision avoidance. The optimization algorithm for CMM path planning is not only to solve the TSP integer programming model. Combining with the collision avoidance issue, the path planning problem can be very difficult to solve. The collision detection function configured in Chapter 3 plays a significant role in determining unnecessary collisions. This Chapter will discuss how to avoid these unnecessary collisions and then perform the sequencing optimization. To achieve this objective, many factors are considered, such as collision positions, collision signal, big M penalty, and measuring points distance matrix. Figure 4.1 shows the complete process to generate an inspection path of a product.

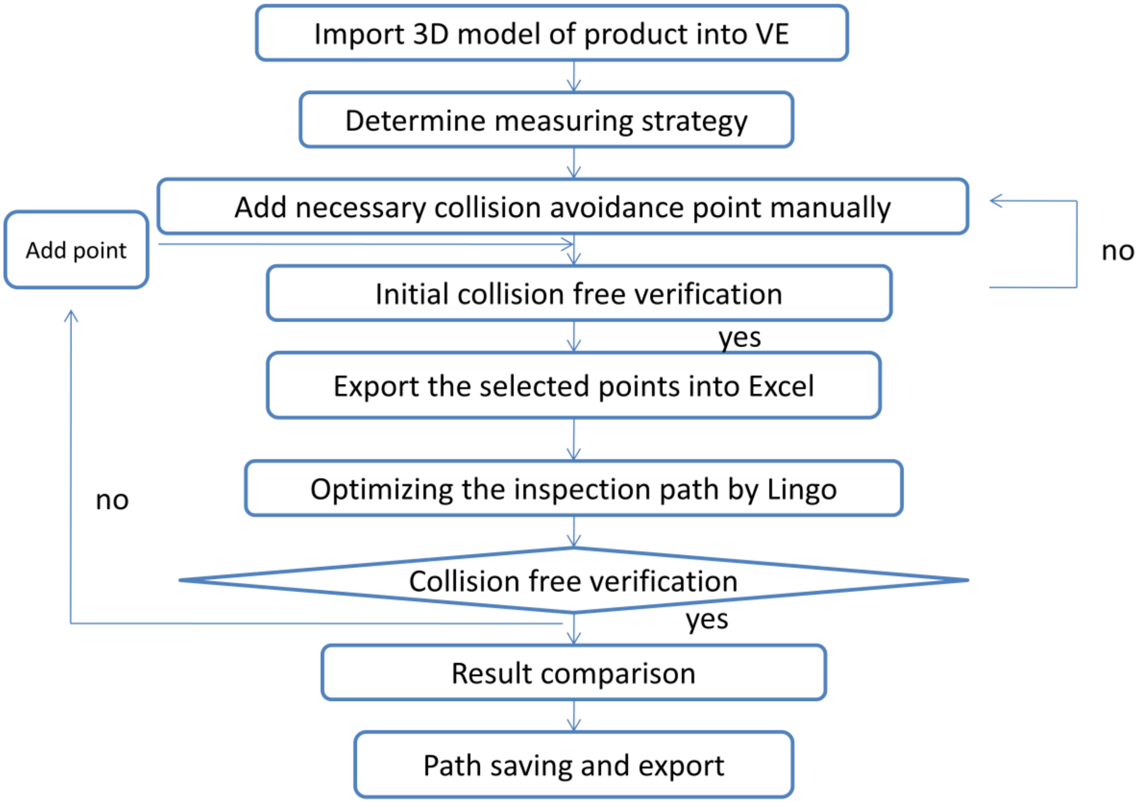


Figure 4. 1 Path Planning Flow Chart

Steps of the path planning:

1) To measure the physical dimensions of a product in the Virtual CMM system, the 3-D model of the product is built in CAD software and imported into VE. The scale and position of the product model are adjusted to fix the workpiece on the virtual CMM operating table. Figure 4.2 shows an example of workpice placed on the operating table.

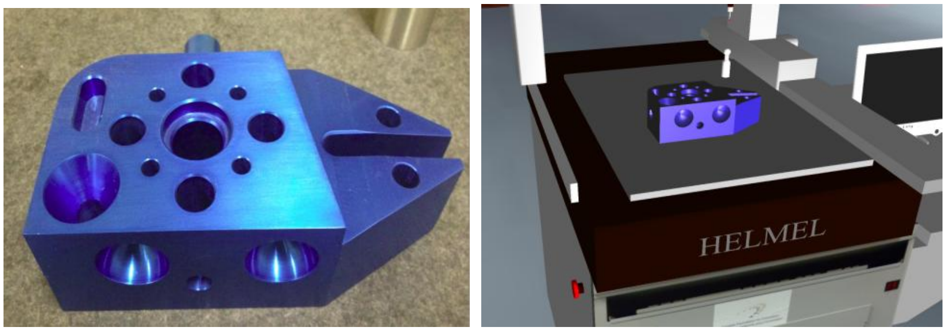
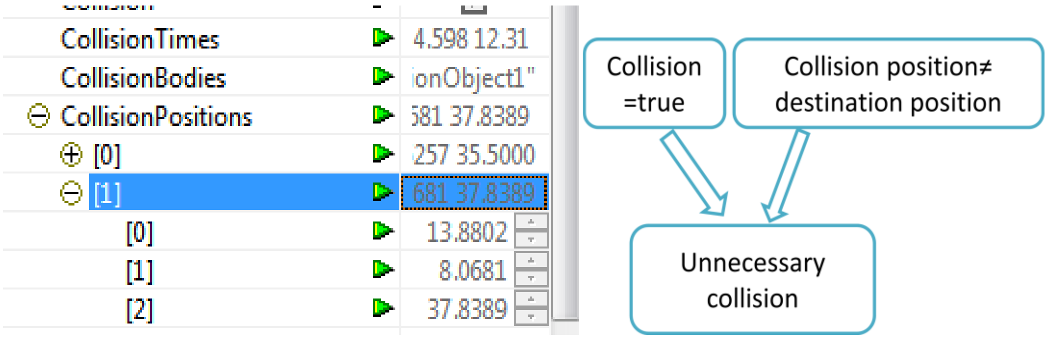


Figure 4. 2 Workpiece and Virtual CMM

2) The measuring strategy is determined according to the shape and applications of the product. For example, to measure the dimension of a sphere, total five inspection points are needed around the sphere. Six inspection points are needed to measure a cylinder, three points at each end. The position data of these inspection points are collected in the Virtual CMM system and exported into a Excel file for further calculation
3) To avoid unnecessary collisions, dummy points are added manually in the Virtual CMM system. As same as the basic inspection points, the position data of these dummy points are exported into the Excel file.
4) The distances between every two inspection points are calculated in Excel. The “n x n” distance matrix is formed for further optimization calculation.
5) Run the collision checking program in the Virtual CMM system. If any collision is detected between a CMM probe and the product. The collision position can be read directly in the system (see Figure 23). In the Eon’s programming code, if the collision Boolean is “TRUE” and at the same time, the collision position does not match with

the destination position. This collision is defined as an unnecessary collision as shown in Figure 4.3.



CollisionTimes 4.598 12.31
CollisionBodies ionObject1"
CollisionPositions 581 37.8389
[0] 257 35.5000
[1] 681 37.8389
[0] 13.8802
[1] 8.0681
[2] 37.8389

Figure 4. 3 Collision detection

According to Buchal and Wang's theory, a big penalty M will be added to the path segment with collisions. In this research, the distance between any two points with the collision are added by 1000 and then exported into Excel.

6) Lingo is selected in this research to optimize the measuring sequence of all inspection points. The model and detail programming code will be introduced later in the algorithm section.

7) The optimized measuring sequence is exported into Excel. The script programming function in Virtual CMM system allows user to read the sequencing data from Excel then examine the inspection path for collision-free verification. If any collision is detected, more dummy points will be selected for collision avoidance. A big penalty M will be added to the path segments with collisions. The optimization process will be repeated until a collision free inspection path is found. The final path will be saved export to real CMM.

4.2 Optimization Algorithm

The algorithm proposed in this research is a combination of the integer programming model and the big M penalty method. The integer programming model which optimizes the measuring sequence is listed below [35].

Objectives: to minimize the total travelling distance P of the CMM probe

$$P=\min\sum_{i=1}^{n}\sum_{j=1}^{n} D_{ij}X_{ij}$$

Constraints: each inspection points can be visited only once.

$$\sum_{j=1}^{n} X_{ij}=1\text{ , i=1, 2……n} \qquad (4.1)$$

$$\sum_{i=1}^{n} X_{ij}=1\text{ , j=1, 2……n} \qquad (4.2)$$

$$u_i-u_j+nX_{ij}\le n-1,\quad 2\le i\ne j\le n \qquad (4.3)$$

Where

n be the number of points to be measured.

i and j be any two points.

$D_{i,j}$ be the distance between i and j.

u_i be the measuring sequence of point i.

$$u_i, u_j \ge 0$$

$$X_{ij}\in\{0,1\},\quad i, j=1, 2\ldots n,\quad i\ne j$$

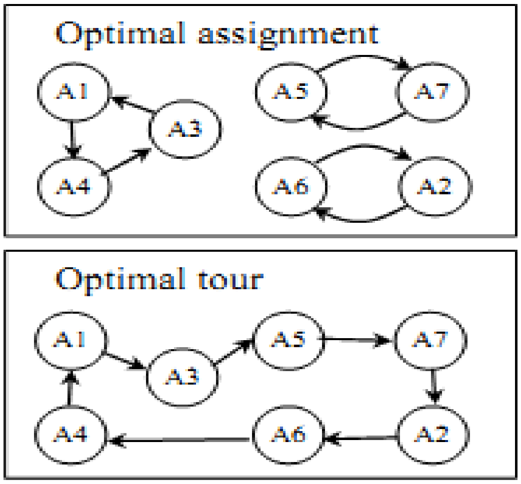


Figure 4. 4 Sub Tour [36]

The sum of the traveling distance between any two points from i to j is minimized in the objective function. Constraints 1 and 2 ensure that each point can be visited only once. Constraint 3 is proposed by Xie and Xue [35] to ensure that there is no sub-tour in the final inspection path. As Figure 4.4 shows on the left, this constraint limits the integer model to generate a signal continuous inspection path.

CMM path planning is not only a TSP problem. Collision avoidance also plays a significant role in this algorithm. There are many existing approaches to detect and avoid collisions such as ray tracing, B-rep. In this research, the big M penalty method which is popular in the operation research study is selected for the collision avoidance.

Big penalty M method is a traditional method using in the minimization problem. If one of the conditions is limited or prohibited, a big value M can be added to this condition then it will not be selected in the following calculation. For example:

$P=\min\sum_{i=1}^{n}\sum_{j=1}^{n} D_{ij}\, X_{ij}$

If we define D_{46} equals to a very big value M by 1000, then the measuring path from point 4 to point 6 will not be selected due to the minimization intention in the objective function search. For each path segment with collisions in the measuring process, a big penalty M is added to its corresponding distance value. The optimization model will neglect those infeasible paths to minimize the total travelling distance P to generate a collision free inspection path. Figure 4.5 shows how the big M method works in the signal transition between programs.

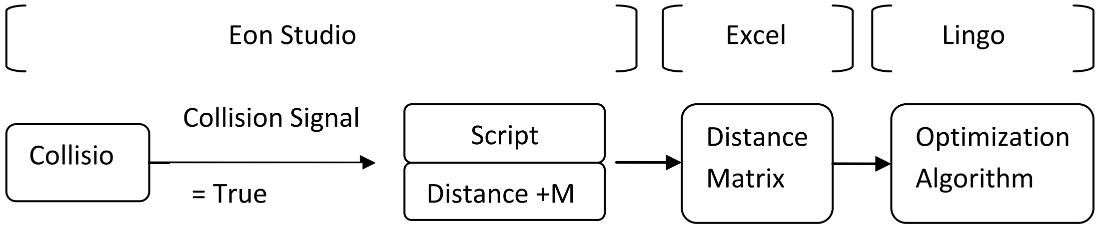


Figure 4. 5 Big M signal Transition

4.3 The four-cubes example

In this research, a 3-D model of 4 cubes is built as an example for dimension checking. Figure 4.6 shows the model located in the Virtual CMM system. Total sixteen basic inspection points are selected to measure in the model.

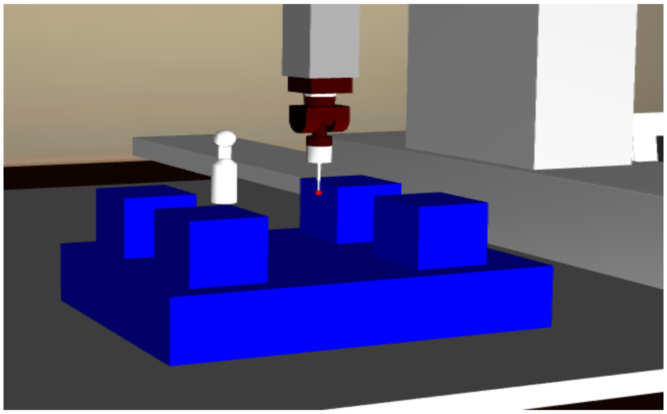


Figure 4. 6 Four-cubes model

To avoid unnecessary collisions between CMM probe and cubes, nine dummy points are added around these cubes. Including the origin point, total twenty-six measuring points are selected to generate an inspection path. Figure 4.7 is a 2D projection of the cubes and the inspection points.

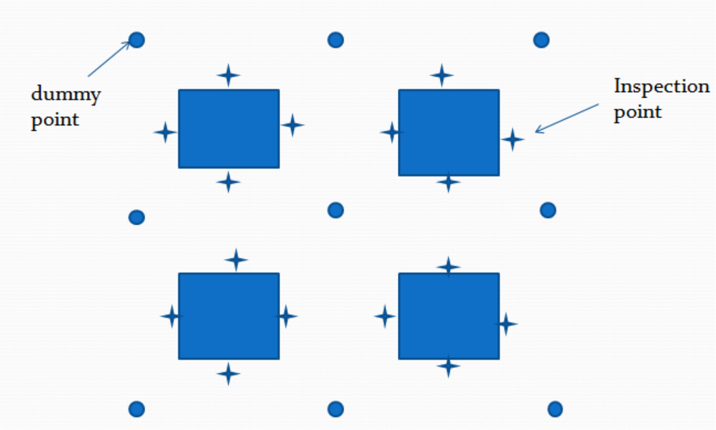
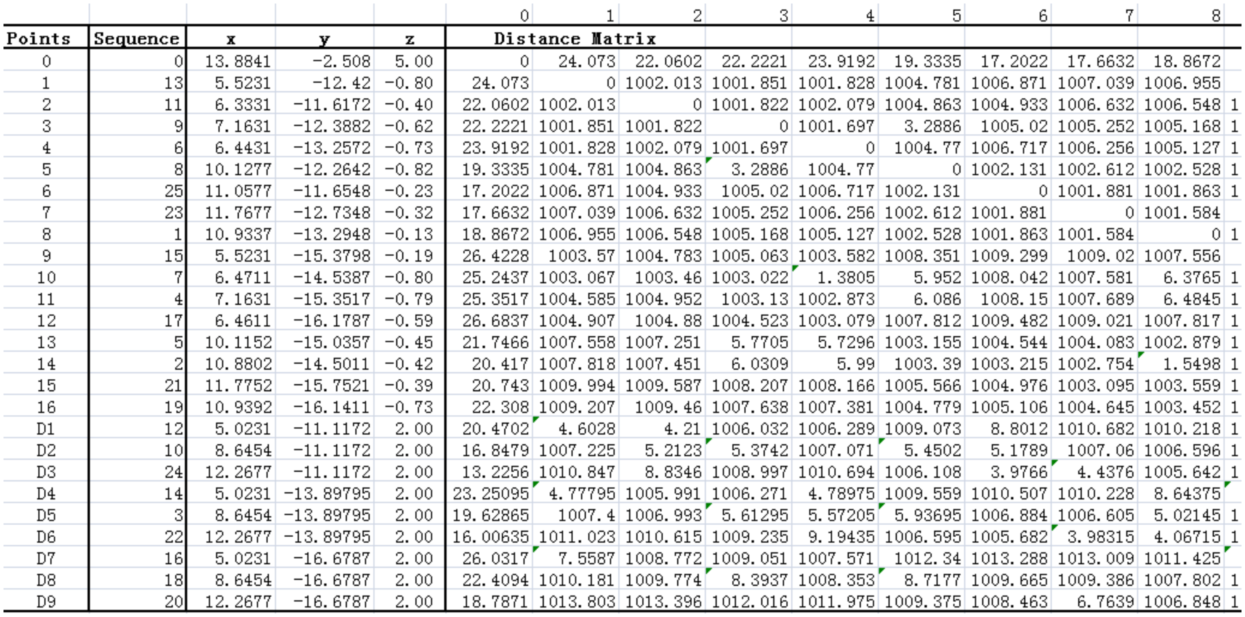


Figure 4. 7 Inspection Points

The 3-axis coordinates of these 26 points are exported into an Excel file (see Figure 4.8). Unlike cities in TSP, CMM inspection points are located in the 3-axis coordinate system. The distance between any two points is calculated by Equation 4.4:

$$Dij=|xi-xj|+|yi-yj|+|zi-zj|. \qquad (4.4)$$

A 26 x 26 distance matrix is formed based on the equation and the big M penalty (1000) is added to each path segment with the collision.



					0	1	2	3	4	5	6	7	8	
Points	Sequence	x	y	z	Distance Matrix									
0	0	13.8841	-2.508	5.00	0	24.073	22.0602	22.2221	23.9192	19.3335	17.2022	17.6632	18.8672	
1	13	5.5231	-12.42	-0.80	24.073	0	1002.013	1001.851	1001.828	1004.781	1006.871	1007.039	1006.955	
2	11	6.3331	-11.6172	-0.40	22.0602	1002.013	0	1001.822	1002.079	1004.863	1004.933	1006.632	1006.548	1
3	9	7.1631	-12.3882	-0.62	22.2221	1001.851	1001.822	0	1001.697	3.2886	1005.02	1005.252	1005.168	1
4	6	6.4431	-13.2572	-0.73	23.9192	1001.828	1002.079	1001.697	0	1004.77	1006.717	1006.256	1005.127	1
5	8	10.1277	-12.2642	-0.82	19.3335	1004.781	1004.863	3.2886	1004.77	0	1002.131	1002.612	1002.528	1
6	25	11.0577	-11.6548	-0.23	17.2022	1006.871	1004.933	1005.02	1006.717	1002.131	0	1001.881	1001.863	1
7	23	11.7677	-12.7348	-0.32	17.6632	1007.039	1006.632	1005.252	1006.256	1002.612	1001.881	0	1001.584	
8	1	10.9337	-13.2948	-0.13	18.8672	1006.955	1006.548	1005.168	1005.127	1002.528	1001.863	1001.584	0	1
9	15	5.5231	-15.3798	-0.19	26.4228	1003.57	1004.783	1005.063	1003.582	1008.351	1009.299	1009.02	1007.556	
10	7	6.4711	-14.5387	-0.80	25.2437	1003.067	1003.46	1003.022	1.3805	5.952	1008.042	1007.581	6.3765	1
11	4	7.1631	-15.3517	-0.79	25.3517	1004.585	1004.952	1003.13	1002.873	6.086	1008.15	1007.689	6.4845	1
12	17	6.4611	-16.1787	-0.59	26.6837	1004.907	1004.88	1004.523	1003.079	1007.812	1009.482	1009.021	1007.817	1
13	5	10.1152	-15.0357	-0.45	21.7466	1007.558	1007.251	5.7705	5.7296	1003.155	1004.544	1004.083	1002.879	1
14	2	10.8802	-14.5011	-0.42	20.417	1007.818	1007.451	6.0309	5.99	1003.39	1003.215	1002.754	1.5498	1
15	21	11.7752	-15.7521	-0.39	20.743	1009.994	1009.587	1008.207	1008.166	1005.566	1004.976	1003.095	1003.559	1
16	19	10.9392	-16.1411	-0.73	22.308	1009.207	1009.46	1007.638	1007.381	1004.779	1005.106	1004.645	1003.452	1
D1	12	5.0231	-11.1172	2.00	20.4702	4.6028	4.21	1006.032	1006.289	1009.073	8.8012	1010.682	1010.218	1
D2	10	8.6454	-11.1172	2.00	16.8479	1007.225	5.2123	5.3742	1007.071	5.4502	5.1789	1007.06	1006.596	1
D3	24	12.2677	-11.1172	2.00	13.2256	1010.847	8.8346	1008.997	1010.694	1006.108	3.9766	4.4376	1005.642	1
D4	14	5.0231	-13.89795	2.00	23.25095	4.77795	1005.991	1006.271	4.78975	1009.559	1010.507	1010.228	8.64375	
D5	3	8.6454	-13.89795	2.00	19.62865	1007.4	1006.993	5.61295	5.57205	5.93695	1006.884	1006.605	5.02145	1
D6	22	12.2677	-13.89795	2.00	16.00635	1011.023	1010.615	1009.235	9.19435	1006.595	1005.682	3.98315	4.06715	1
D7	16	5.0231	-16.6787	2.00	26.0317	7.5587	1008.772	1009.051	1007.571	1012.34	1013.288	1013.009	1011.425	
D8	18	8.6454	-16.6787	2.00	22.4094	1010.181	1009.774	8.3937	1008.353	8.7177	1009.665	1009.386	1007.802	1
D9	20	12.2677	-16.6787	2.00	18.7871	1013.803	1013.396	1012.016	1011.975	1009.375	1008.463	6.7639	1006.848	1

Figure 4. 8 Excel data

As above-mentioned, the integer programming model is calculated using Lingo. The programming code is listed as follows:

```
Model:

sets: point /1..26/:u;

link(point,point):dist,x;

endsets

DATA:

dist= @ole('april20','matrix');

@ole('april20','order')=u;

ENDDATA

n=@SIZE(point); min=@sum(link: dist * x);

@for(point(k): @sum(point(i)|i#NE# k: x(i,k))=1;
```

```
                    @sum(point(j)|j#NE#k:x(k,j))=1;);
@for(point(i): @for(point(j)|j#GT#1 #AND# i#NE#j:
                    u(i)-u(j)+n*x(i,j)<=n-1););
@for(point(i): u(i)<=n-1);
@for(link:@BIN(x));
END
```

The 26 x 26 distance matrix is imported to Lingo using Command @ole. After the optimization process, the measuring sequence array “u” is exported back to Excel. An optimized inspection path is formed based on the value of “u” and examined in the Virtual CMM system for the collision free verification. Figure 4.9 shows the relations between these three programs.

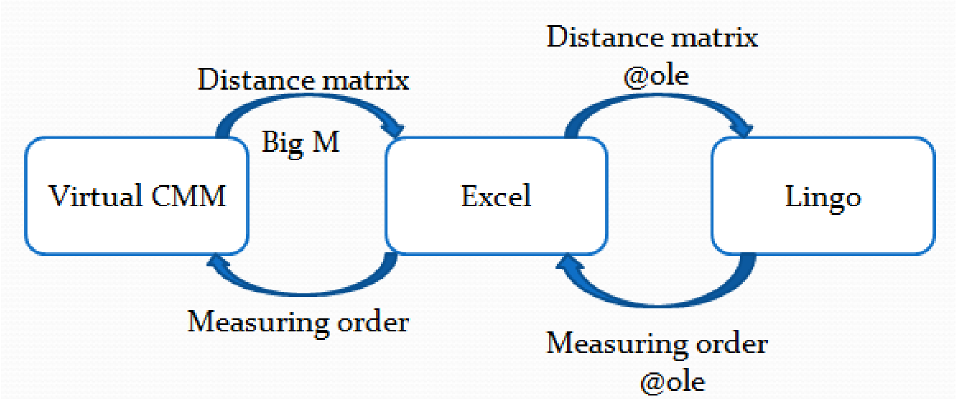
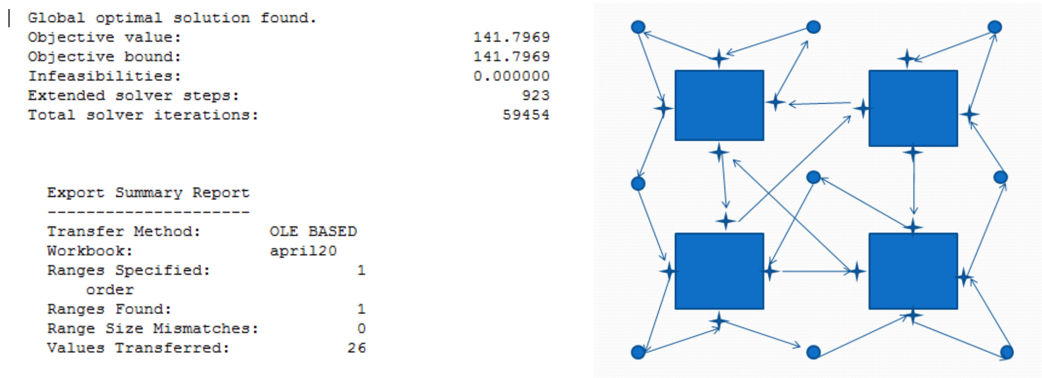


Figure 4. 9 Transition

4.4 Solution Analysis

For the four-cubes example, the final optimized measuring sequence is 0-8-14-D5-11-13-4-10-5-3-D2-2-D1-1-D4-9-D7-12-D8-16-D9-15-D6-7-D3-6. This measuring sequence is verified successfully to form a collision–free inspection path in the Virtual CMM system. Figure 4.10 shows the optimization result from Lingo and the final inspection path. Five random inspection paths are selected for the result comparison. Compared to the optimized result with these five results in Table 2, the total travelling distance of the optimal inspection path is effectively reduced.



```
| Global optimal solution found.
  Objective value:                              141.7969
  Objective bound:                              141.7969
  Infeasibilities:                              0.000000
  Extended solver steps:                             923
  Total solver iterations:                         59454


    Export Summary Report
    ---------------------
    Transfer Method:       OLE BASED
    Workbook:              april20
    Ranges Specified:             1
        order
    Ranges Found:                 1
    Range Size Mismatches:        0
    Values Transferred:          26
```

Figure 4. 10 Optimization Result

Table 2 Results comparison

Number of trial	Total travelling distance	Improvement
Optimized path	141.7969	-
1	173.7471	19%
2	197.9981	29%
3	178.6566	21%
4	204.7137	31%
5	167.9341	16%

To validate this algorithm proposed, the model used in the four cubes example is simplified to a one cube model. Total four inspection points and four dummy points are selected around the cube as shown in Figure 4.11.

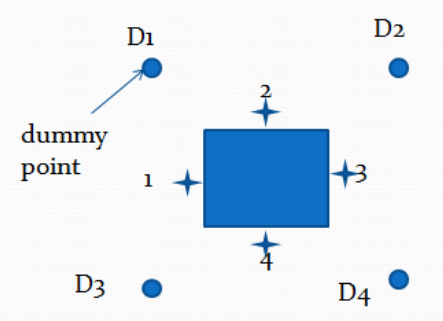
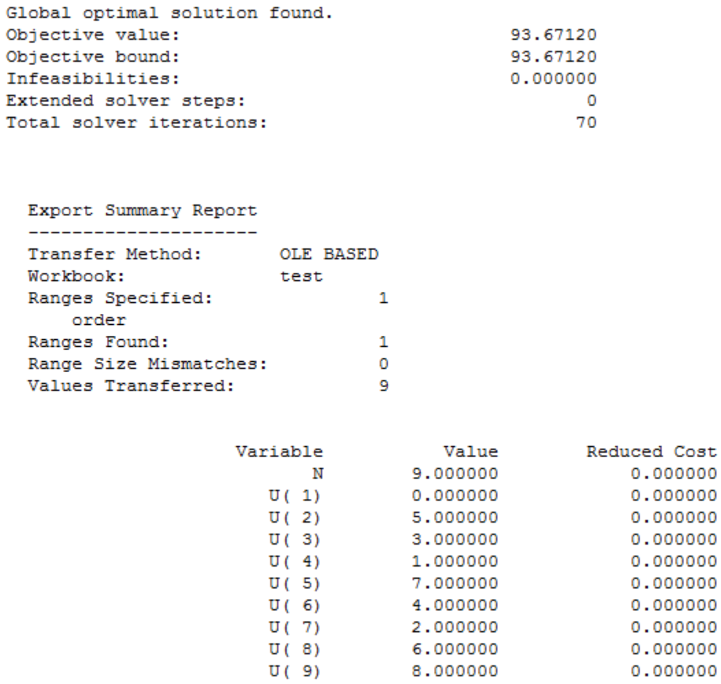


Figure 4. 11 One-Cube model

As same as the four-cubes model, the measuring sequence of these eight points is optimized using the Lingo programming. Figure 4.12 shows the optimization result from Lingo. The optimal objective value is 93.6712. The shortest traveling path is 0-3-D2-2-D1-1-D3-4-D4.



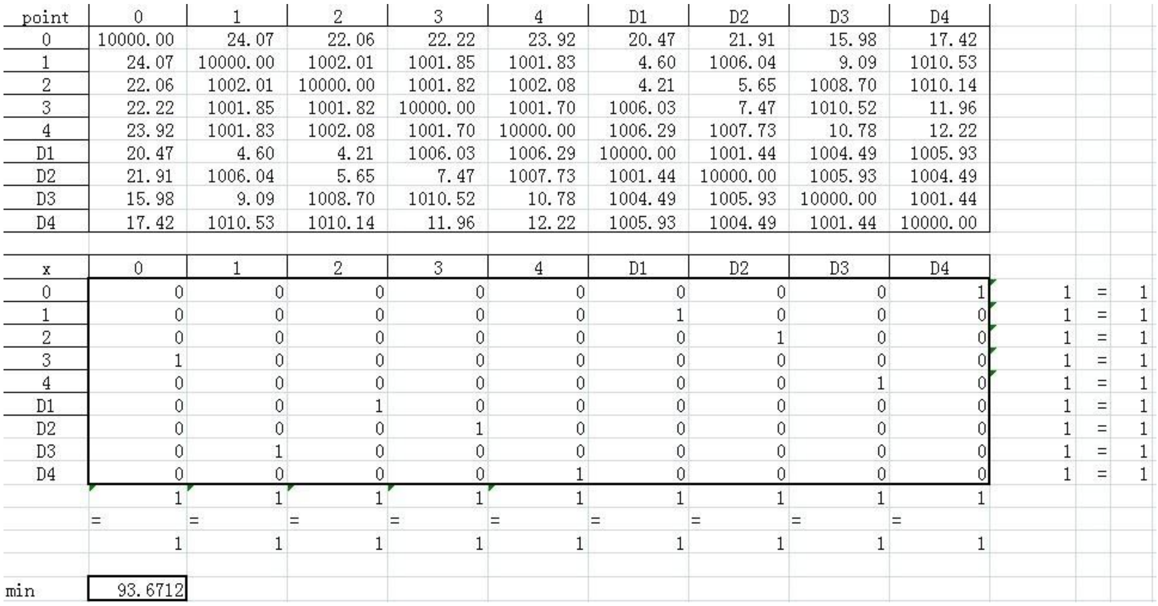
```
Global optimal solution found.
Objective value:                              93.67120
Objective bound:                              93.67120
Infeasibilities:                              0.000000
Extended solver steps:                               0
Total solver iterations:                            70


  Export Summary Report
  ---------------------
  Transfer Method:       OLE BASED
  Workbook:              test
  Ranges Specified:               1
      order
  Ranges Found:                   1
  Range Size Mismatches:          0
  Values Transferred:             9

                     Variable           Value        Reduced Cost
                            N        9.000000            0.000000
                        U( 1)        0.000000            0.000000
                        U( 2)        5.000000            0.000000
                        U( 3)        3.000000            0.000000
                        U( 4)        1.000000            0.000000
                        U( 5)        7.000000            0.000000
                        U( 6)        4.000000            0.000000
                        U( 7)        2.000000            0.000000
                        U( 8)        6.000000            0.000000
                        U( 9)        8.000000            0.000000
```

Figure 4. 12 Lingo Result

To validate the algorithm, the integer programming model is also calculated using Excel solver (see Figure 4.13). The calculated optimal path is 0-D4-4-D3-1-D1-2-D2-3. The result of objective value which equals to 93.6712 is exactly same with the result obtained by Lingo programming. This consequence proves that the algorithm proposed in this research works successfully to generate an optimal collision free inspection path.



point	0	1	2	3	4	D1	D2	D3	D4
0	10000.00	24.07	22.06	22.22	23.92	20.47	21.91	15.98	17.42
1	24.07	10000.00	1002.01	1001.85	1001.83	4.60	1006.04	9.09	1010.53
2	22.06	1002.01	10000.00	1001.82	1002.08	4.21	5.65	1008.70	1010.14
3	22.22	1001.85	1001.82	10000.00	1001.70	1006.03	7.47	1010.52	11.96
4	23.92	1001.83	1002.08	1001.70	10000.00	1006.29	1007.73	10.78	12.22
D1	20.47	4.60	4.21	1006.03	1006.29	10000.00	1001.44	1004.49	1005.93
D2	21.91	1006.04	5.65	7.47	1007.73	1001.44	10000.00	1005.93	1004.49
D3	15.98	9.09	1008.70	1010.52	10.78	1004.49	1005.93	10000.00	1001.44
D4	17.42	1010.53	1010.14	11.96	12.22	1005.93	1004.49	1001.44	10000.00

x	0	1	2	3	4	D1	D2	D3	D4			
0	0	0	0	0	0	0	0	0	1	1	=	1
1	0	0	0	0	0	1	0	0	0	1	=	1
2	0	0	0	0	0	0	1	0	0	1	=	1
3	1	0	0	0	0	0	0	0	0	1	=	1
4	0	0	0	0	0	0	0	1	0	1	=	1
D1	0	0	1	0	0	0	0	0	0	1	=	1
D2	0	0	0	1	0	0	0	0	0	1	=	1
D3	0	1	0	0	0	0	0	0	0	1	=	1
D4	0	0	0	0	1	0	0	0	0	1	=	1
	1	1	1	1	1	1	1	1	1			
	=	=	=	=	=	=	=	=	=			
	1	1	1	1	1	1	1	1	1			
min	93.6712											

Figure 4. 13 Excel Solver

Chapter 5 Conclusion and future work

5.1 Contributions

This research developed a VR-based CMM training and path planning system. This system is operated in a virtual environment for CMM operations. The CMM 3D model and simulation are constructed based on the CMM prototype and operation manual. Major CMM functions can be implemented in the virtual environment. The system provides 3D offline CMM operation environment which let users to manipulate the virtual CMM and gain real life experiences. Manufacturing industry may use this system as an auxiliary education tool to train their new employees. Trainees can understand the operation process before actually manipulating the real CMM. Possible damages to the sensitive probing system can be effectively avoided and maintenance cost can be saved. Experienced CMM operators may use this system to design and examine the inspection path for collision checking. This virtual CMM system is developed using CAD and VR software. 67 parts of CMM were modelled in AutoCAD then exported into Eon Studio for simulation. In Eon Studio, many function nodes such as lighting and texture mapping are used to enhance the visualization effect. Motion and script nodes are used to support CMM probe movement in the virtual environment. Collision manager node combined with the VB language program are used to implement the collision detection function between the CMM probe and workpieces. This virtual CMM system provides a realistic CMM operation interface and allows users to practice CMM operations in VE. It can be used as an effective supplementary training tool to the traditional CMM training system.

To improve the CMM inspection efficiency, many path planning algorithms are studied in this thesis. This research integrates advantages of existing research solutions. A path planning algorithm which combines the big M penalty method and the integer programming model is proposed in this thesis to optimize the CMM path. It allows users to generate an optimal inspection path to save operating time and cost. The CMM path planning system in this research focuses on a part in a given setup using a specific probe in a specific orientation. It forms two objectives: collision avoidance and measurement sequence. A collision-free path is obtained by adding big penalty M to any path with collisions. VB script programming is used to add the M into distance matrix then export into lingo program. Measuring sequence of all inspection points is optimized using Lingo programming. An optimal or near optimal collision free inspection path can be generated at the end for the product inspection. The proposed algorithm is validated by the verification of the collision free path in the Virtual CMM system and the optimization model in Excel using solver. The algorithm proposed is a new attempt by combining two theories together to perform the optimization and collision avoidance at the same time. Although these two methods are traditional and relatively simple, to implement them together in the VE is much more difficult. Three software systems need to be integrated to perform the algorithm. The programming codes and signal transition are carefully revised to enable that the system generates the optimal inspection path in Lingo then transits the measuring sequence back to Eon studio automatically for the result verification. In conclusion, the algorithm proposed in this research might not be the most powerful algorithm for this optimization problem, but it fits with the Virtual CMM system effectively and is able to improve the measuring efficiency for CMM path planning.

5.2 Future Work

In the future research, the globe path planning algorithm should be considered to complete the CMM path planning system. Other optimization algorithms such as genetic algorithm may be studied and considered to compare the optimization results. Additionally, during the path designing process, if a collision is detected in the Virtual CMM system, the extra dummy points have to be added manually. The distance matrix exported into excel also needs to be edited manually to meet the Lingo requirement. The goal of this research is to develop a CMM path planning system which can generate an optimal inspection path automatically. Therefore these problems should be considered in the future research to further improve the path planning efficiency.

References

[1] Hu, Y., Yang, Q. and Wei, P., 2009, "Development of a novel virtual coordinate measuring machine"**,** *2009 IEEE Intrumentation and Measurement Technology Conference*, IEEE, Piscataway, NJ, USA.

[2] Lin, Z.C. and Chen, C.C, 2001, "Collision-Free Path Planning for Coordinate Measurement Machine Probe", *International Journal of Production Research*, **39**:9, pp. 1969–1992.

[3] He, H.W. and Wu, W.M., 2009, "Web-based virtual operating of CNC milling machine tools", *Computers in Industry*, Volume 60, Issue 9, pp. 686-697.

[4] Wikipedia, 2008, *http://en.wikipedia.org/wiki/Virtual_reality*.

[5] Yao, A.W.L. and Hsieh, S.J., 2008, "A Virtual Factory Planning and Estimation System for Engineering and Education", *ASME 2008 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Paper no. DETC2008-49215, pp. 1473-1479.

[6] Liang, J.S, 2008, "A study on virtual reality-based EDM learning framework and effectiveness analysis", *Computer-Aided Design and Applications*, Vol. 5, No. 1-4, pp. 391-400.

[7] Brough, J.E., Schwartz, M., Gupta, S.K., Anand, D.K., Kavetsky, R. and Pettersen, R., 2007, "Towards the development of a virtual environment-based training system for

mechanical assembly operations”, *Virtual Reality*, Vol. 11, No. 4. (October 2007), pp. 189-206.

[8] Yao, L.L., Li X.Q., Chen J.Y., 2009, “Research and implementation of virtual assembly training system”， *IT in Medicine & Education, 2009. ITIME '09. IEEE International Symposium on*, vol.1, no., pp.641-647.

[9] Albuquerque, V.A., Liou, F.W. and Mitchell, O.R., 2000, “Inspection point placement and path planning algorithms for automatic CMM inspection”, *International Journal of Computer Integrated Manufacturing*, Volume 13, pp. 107-120.

[10] Strickland, J., 2008, “Virtual reality history”, *HowStuffWorks*.

[11] Beier, K.P., 2000, “Virtual Reality: A Short Introduction”, *VRL.umich.edu*.

[12] Wikipedia, 2012, *http://en.wikipedia.org/wiki/VRML*.

[14] Zhou, W., Gong, D., Xue, Y. and Liu, C., 2011, “Collision detection for 6-DOF parallel-link CMM”， *Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International Conference on*, vol.1, no., pp. 157-160.

[15] Yang, Z. and Chen, Y., 2005, “Inspection Path Generation in Haptic Virtual CMM”, *Computer-Aided Design & Applications*, Vol. 2(1-4), pp. 273-282.

[16] Lin,Y.J. and Murugappan, P. , 1998 “A new algorithm for CAD-directed CMM dimensional inspection”， *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol.1, pp. 893-898.

[17] Li, X.L., Lu. Y. and Hua, P., 2009, “CMM Measurement Path Generation and Simulation Methods”, *Modular Machine Tool & Automatic Manufacturing Technique*, n6, pp. 71-72.

[18] Zhao, H., Liu, D.X., Dong, Y. and Wang, W.L., 2009, “Development of CAD-based inspection planning system for CMM”, *Chinese Journal of Scientific Instrument*, vol.30, n9., pp. 1846-1853.

[19] Lin, Y.J., Mahabaleshwarkar, R., and Massina, E., 2001, “CAD-based CMM dimensional inspection path planning – a generic algorithm”, *Robotica*, pp. 137-148.

[20] Lu, E., Ni, J. and Wu, S.M., 1994, “An algorithm for the generation of an optimum CMM inspection path”, *Transactions of the ASME. Journal of Dynamic Systems, Measurement and Control*, Vol. 116(3), pp. 396-404.

[21] Lin, Z.C, and Chen, C.C, 1997, “Measuring-sequence planning by the nearest neighbor method and the refinement method”, *International Journal of Advanced Manufacturing Technology*, Vol.13, pp. 271-281.

[22]Liu, D.X., Zhao, H., Dong, Y. and Wang, W.L., 2009, “Collision-free inspection path generation for coordinate measuring machines Source”, *Journal of Computer Aided Design & Computer Graphics*, vol.21, n6., pp. 804-811.

[23] Chen, Y.H., Wang, Y.Z. and Yang, Z.Y., 2004, “Towards a haptic virtual coordinate measuring machine”, *International Journal of Machine Tools and Manufacture,* 44(10), pp. 1009-1017.

[24] Zhao, L and Peng, Q.J., 2010, “Development of a CMM Training System in Virtual Environments”, *ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference,* Paper no. DETC2010-28274, pp. 537-544.

[25] Limaiem, A. and ElMaraghy, H.A., 1998, “Automatic path planning for coordinate measuring machines”, *Proceedings of 1998 IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 887-92.

[26] Lin, Z.C. and Chow, J.J. 2001, “Near optimal measuring sequence planning and collision-free path planning with a dynamic programming method”, *International Journal of Advanced Manufacturing Technology*, Vol. 18(1), pp. 29-43.

[27] Lu,C.G., Morton, D., Wu, M.H. and Myler, P., 1999, “Genetic algorithm modelling and solution of inspection path planning on a coordinate measuring machine (CMM)”, *International Journal of Advanced Manufacturing Technology*, v 15, n 6, pp. 409-416.

[28]Yau,H.T. and Menq,C.T., 1995, “Automated CMM path planning for dimensional inspection of dies and molds having complex surfaces”, *International Journal of Machine Tools and Manufacture*, Volume 35, Issue 6, pp. 861-876.

[29] Spitz, S.N. and Requicha, A.A.G., 2000, “Multiple-goals path planning for coordinate measuring machines”, *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, vol.3, no., pp. 2322-2327.

[30] Buchal, R.O. and Wang, A. 2006, "CMM sequence optimization with collision detection", *International Journal of Computer Applications in Technology*, v 26, n 1-2, pp. 65-74.

[31] EON Reality, 2010, *www.eonreality.com*.

[32] Wikipedia, 2012, *http://en.wikipedia.org/wiki/Coordinate-measuring_machine*.

[33] Martínez, O.D., Castro, C.S, Fernández, H.X. and Martín, R.F., 2007, "Virtual reality system for industrial training", *IEEE International Symposium on Industrial Electronics,* pp. 1715-1720.

[34] Mears, L., Roth, J.T., Djurdjanovic, D., Yang, X. and Kurfess, T., 2009, "Quality and Inspection of Machining Operations: CMM Integration to the Machine Tool", *Journal of Manufacturing Science and Engineering.*

[35] Xie, J.X., and Xue, Y., 2005, "Optimization Modeling and LINDO / LINGO Software", *Tsinghua University Press*.

[36] Jiang, C., 2010, "A Reliable Solver of Euclidean Traveling Salesman Problems with Microsoft Excel Add-in Tools for Small-size Systems", *JSW,* pp. 761-768.