Proceedings of the 2001 Winter Simulation Conference
B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, eds.

A VIRTUAL ENVIRONMENT FOR SIMULATING MANUFACTURING OPERATIONS IN 3D

Ravi Chawla
Amarnath Banerjee

Department of Industrial Engineering
3131 TAMU
Texas A&M University
College Station, TX 77843-3131, U.S.A.

ABSTRACT

This paper presents a method for simulating basic manufacturing operations (unload, load, process, move, and store) in a 3D virtual environment. The virtual environment provides a framework for representing a facility layout in 3D, which encapsulates the static and the dynamic behavior of the manufacturing system. The 3D manufacturing objects in the facility are mapped with the nodes in the framework. The framework, a modified scenegraph structure, is a tree structure, which can be manipulated by updating the parent-child relationships and the transformation matrix to simulate the basic manufacturing operations. The method can be easily extended to represent more specific manufacturing operations.

1 INTRODUCTION

The advances in virtual reality (VR) technology in the last decade has provided the impetus for applying VR to different engineering applications. VR hardware has seen significant improvements in terms of robustness, usability, portability, and interface with computer systems. The driving software has increased in performance, robustness, support of different hardware configurations, graphics, displays, and use of multiprocessing. Virtual reality provides an environment for immersing users in the environment as well as the ability to interact with the objects in the environment. The virtual environment provides support for using 3D graphics.

The above features make VR an ideal environment for use in a simulation environment to synthesize object interaction. Object interaction depends on the nature of the operation and the application being simulated. In this paper, we demonstrate the use of VR as an environment to simulate basic discrete manufacturing operations, such as load, unload, move, process, and store. These operations can be broken down to a series of elementary operations in the virtual environment.

2 BACKGROUND

A brief description about the current 3D data representation techniques is presented here, followed by a description on the manufacturing operations that are being used in the simulation model. These are the two building blocks of the virtual manufacturing simulator.

2.1 3D Data Structure

The data structure used to represent 3D objects is known as a *scenegraph*. The scenegraph structure is a directed acyclic graph (DAG) with the nodes depicting the objects and its properties, and the edges depicting the relationship between the nodes. A DAG is represented by the notation – G (N, E) which consists of a nonempty set of nodes N and set of edges E. In a directed graph, edges E are ordered pairs (v, w) of nodes. Relationship between nodes is of the form of parent-child linkage and object property inheritance. The nodes at various levels in the hierarchy represent different levels of abstraction. As illustrated in Figure 1, the nodes at the bottom level represent individual objects in the environment. The nodes that are a level higher are used to group the individual nodes. The common properties of the individual nodes can be grouped together at the next higher level.

The scenegraph structure automatically encapsulates some of the physical properties of objects that are represented as nodes. These properties include position and orientation in 3D, material properties and texture information. Depending on the implementation medium of the scenegraph, several other properties are also encapsulated. Examples of these are spatial sound, illumination properties, and elementary sensory data. These properties are found in most of the popular scenegraph implementations, such as VRML (Hartman and Wernecke 1996), Java3D (Sowizral *et al.* 1998), IRIS Performer (Rohlf and Helman 1994; Performer 2001), and WorldToolKit and WorldUp (Sense8 2001). Several enhancements have been made to the scenegraph structure with the addition of important behavioral characteristics, such as precedence relationships and event control lists (Banerjee *et al*. 2000). These enhancements have made it possible to use the scenegraph structure for virtual manufacturing operations.

An interesting observation about the scenegraph structure is the one-to-one relationship between the layers of a facility layout and the levels of the DAG. A layout can be clustered into logical areas based on functional or operational grouping, commonly referred to as manufacturing cell (Hassan *et al*. 1998; Wang *et al*. 1998), which is a group of similar workstations. The workstations are themselves a logical group of different types of equipment primarily based upon interaction. The lowest level in the scenegraph hierarchy corresponds to the equipment within the layout, which are the physical objects in the facility. The level above the lowest level in the scenegraph corresponds to the formation of workstations, while the next higher level corresponds to the manufacturing facility. Figure 1 illustrates the relationship between a facility layout and a scenegraph. The number of levels in the scenegraph is dictated by the number of levels of logical grouping in the facility model. Each logical group possesses specific behavioral attributes. These can be encapsulated in the corresponding level of the scenegraph structure.

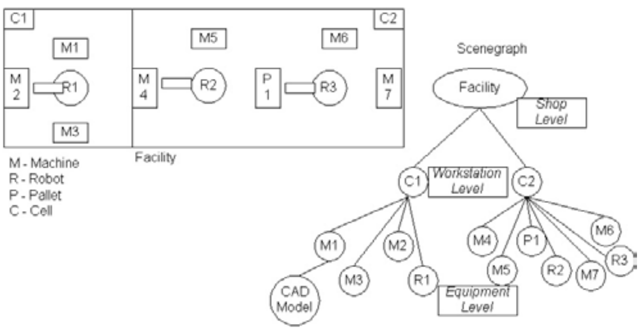

Figure 1: A sample scenegraph hierarchy based on a 2D block layout

2.2 Basic Manufacturing Operations

A discrete part manufacturing process can be characterized by a combination of a fixed set of basic manufacturing operations – *load, unload, move, store, process* (Wysk *et al*. 1995). A sequence of instances of the basic manufacturing operations is used to create a specific part, which depends on the part type, attributes and the sequence of operations. The primary requirements to simulate the manufacturing process within a facility are these basic operations. These operations are discussed from the perspective of the scenegraph and its manipulation.

Wysk *et al*. (1995) differentiate shop floor equipment into three categories on the basis of functional specifications. *Material processors* (MP) are responsible for creation of parts, and primarily perform *process* operation. *Material transporters* (MT) are responsible for moving parts between locations, which is the *move* operation. *Material handlers* (MH) pick parts from material transporters and place them in material processors, and vice versa. The operations include *load*, *unload*, and *store*, and require synchronization between material processors and material transporters.

3 3D OBJECTS AND ITS CHARACTERISTICS

The geometry, material and texture properties of objects (hereby referred to as physical characteristics) are useful in displaying a spatial static 3D facility, which is good for performing architectural walkthrough. It provides a limited set of information comprising of cell formations, facility appearance and dimensions. This information is arranged in a hierarchical format in the scenegraph structure. There are various methods of creating a scenegraph structure from available object data. One such method involving space filling curves (SFC) is described in (Chawla and Banerjee 2001). The scenegraph acts as a facilitator in displaying the solution in 3D.

With suitable modifications, a scenegraph structure is capable of encapsulating a diverse set of information (hereby referred to as operational characteristics) in addition to the physical characteristics. This serves as the base for the integrated framework for display, simulation and analysis. The simulation of a basic manufacturing activity within the virtual environment can be represented by suitable object manipulation within the scenegraph.

Each 3D object has a classification attribute denoting the usage of the object. This has been done to expand the 3D object structure to different types of applications. In the present circumstances, this attribute is set to 'manufacturing' to denote *manufacturing objects*. There are two distinct types of manufacturing objects in terms of their ability to move within the facility – static manufacturing objects and dynamic manufacturing objects. During the synthesis of a manufacturing process, there is interaction between the static and dynamic objects, and the manufacturing objects undergo a series of modifications representing the discrete intermediate steps in the process.

3.1 Static and Dynamic Manufacturing Objects

Static objects are those that remain in a fixed position within the facility. The object as a whole remains stationary during facility operations. The static objects might possess moving components that move during operations, but the movement is confined within the operational bounds or physical bounds of the object. The *operational bound* of an object is defined as the region surrounding an object where one or more components of the object can reach during an operation. Examples of static objects include storage areas, stationary robots and equipment such as lathes, milling machines and machining centers.

Dynamic objects are the class of objects that can move during operations. The entire dynamic object can move or be moved during an operation or it can have one or more components that move during an operation. Examples of the former type include raw and finished parts, forklift trucks and AGVs, while a conveyor is an instance of the latter type. The task remains the same for both types of dynamic objects – move parts from one location to another. As a result, the entire set of material handling equipments or material transporters belong to the dynamic objects class. The movement path can be fixed as in the case of conveyors and AGVs, or can consist of a fixed set of possible paths as in the case of aisles for forklift trucks. The movement of dynamic objects can be modeled by updating the transformation matrix and object ownership within the scenegraph hierarchy structure. They are treated separately as compared to rest of the objects, which have a static hierarchical relationship in the structure..

3.2 Manufacturing Object Interaction

As described earlier, material handlers pick parts from material transporters and place them in material processors, and vice versa. The operations include *load*, *unload*, and *store*. The interaction between manufacturing objects is performed with the material handler synchronously interfacing with the material processor and material transporter.

The interaction is performed only at a finite number of fixed locations, referred to as *interaction points*. The interaction points lie in the intersection of the operational bounds of material handlers and material processors or material transporters. The interaction points are represented as nodes in the scenegraph and are owned by the manufacturing objects. The parts that are being processed in the manufacturing facility are attached to the scenegraph as children of the interaction points. Figure 2 illustrates the operational bounds and interaction points of some of the equipments in TAMCAM. The ownership of the parts is updated during a manufacturing operation.

3.2.1 Load, Unload and Move Operations

A load operation can be characterized by a sequence of steps where a part is initially owned by a material transporter. In order for the part to be moved to a material processor, a series of steps are performed. The material transporter reaches a destination within the cells where the part needs to be processed. The destination is an interaction point between the material transporter and a material handler. The material handler takes over the part ownership at the interaction point, which signifies the picking up of the part by the material handler. The material handler then moves to the interaction point between the material processor and the material handler. At the interaction point, the material handler transfers the ownership of the part to the material processor. This is the step where the part is loaded into the material processor.

The reverse sequence of steps is performed during the unload operation. The part ownership is transferred from the material processor to the material handler at the interaction point. The material handler moves to the interaction point between the material handler and the material transporter. Here the ownership of the part is transferred to the material transporter.

The part ownership is transferred from an instance of one interaction point owned by a shop floor equipment to another shop floor equipment. Due to the hierarchical nature of the scenegraph structure, a number of physical characteristics (mainly the transformation information) of the equipment is inherited by the part during the time it is owned. This is useful while moving the part; it moves as part of the moving equipment that owns it.

The load, unload and move operations are illustrated in Figure 3. TAMCAM facility has been utilized to depict the details. An instance of a process plan is taken into consideration where a MP (Sabre) has completed processing *part1*. *Part1* is *unloaded* from MP (Sabre) interaction point by MH (Puma) and *loaded* onto the interaction point of the MT (Conveyor of cell 2). Interaction points are depicted as children of the MP or MT in the scenegraph of the 3-D facility. An *unload* operation can be characterized by a sequence of steps where the ownership of a part is initially owned by a MP (Sabre). The MH (Puma) takes over the part ownership at the interaction point where it is picked up and *loaded* to the MT (Conveyor). The MT takes over the ownership of the part when the MH reaches the interaction point.

3.2.2 Process Operation

A material processor is capable of processing a part. Processing involves giving a new shape to the part by performing a series of machining operations whereby material is removed from the part. The resulting part may or may not have similar geometry as the original part, depending on the type of processing. For example, a grinding operation will remove minimal material thus retaining the shape. On the other hand, a turning operation may give a completely new geometry to the part.

The depiction of change in shape is dependent on the amount of processing data that is available as well as the available computational resources. Often, while performing factory operations simulation, it is not very essential to depict the part processing. The initial and final shapes are of importance.

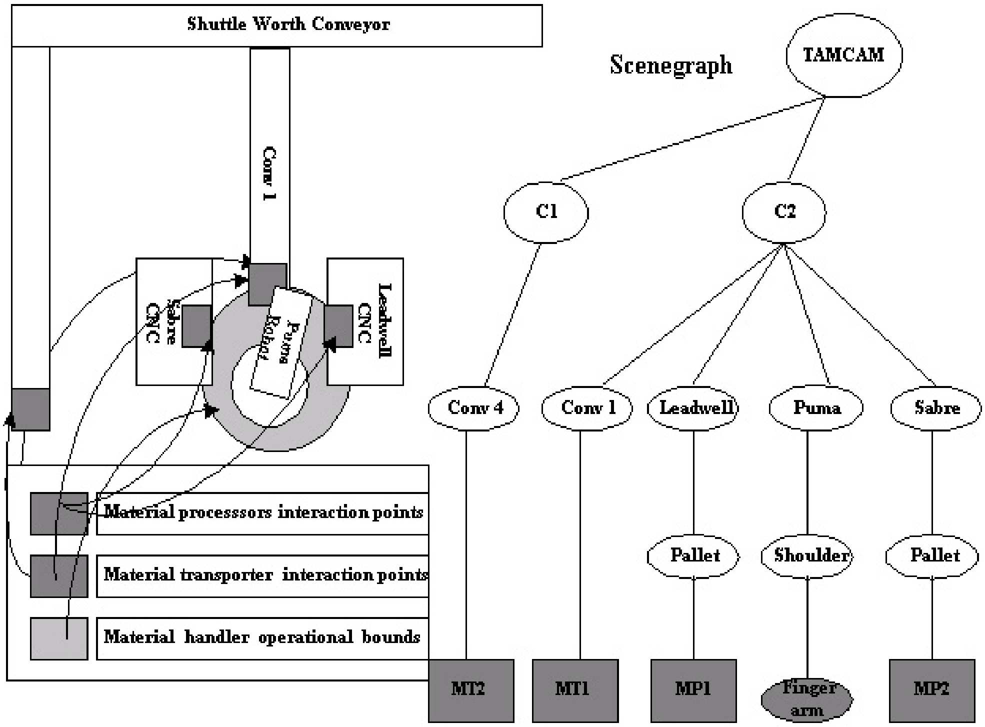

Figure 2: Operational bound and Interaction point in some of TAMCAM equipment

The final shape of the part can be substituted for the original part at the completion of the processing step. The process operation can be represented at the aggregate or at the detailed level.

At the aggregate level, the intermediate processing steps are not represented in the scenegraph. The initial and the final steps are represented in the scenegraph by substituting the unfinished part geometry by the finished part geometry. The aggregate level is time saving and can be used when it is not essential to simulate the part being processed.

At the detailed level, the NC file is used to generate the intermediate geometry by a combination of computational solid geometry (CSG) primitive operations (union, intersection, difference). The NC file is parsed and the machining instructions are used to determine the operation to be performed and the parameter values. The detailed level is computationally expensive, but provides realistic representation of the actual manufacturing operations.

4 IMPLEMENTATION DETAILS

An object-oriented approach is being used to develop the virtual environment. Figure 4 illustrates the 3D manufacturing object data structure that is being used for the implementation. The development is being done on an SGI Octane workstation running IRIX 6.5 operating system. SGI's IRIS Performer is used for development and manipulation of the scenegraph structure. The Octane drives the Immersadesk™, which is the virtual reality hardware that is being used. The CAVELIB (Cavelib 2001) is used to develop the interface with the Immersadesk. Figure 5 provides a snapshot of the factory model.

The model can be used in a non immersive environment such as a PC desktop with relative ease. The most suitable form would be using the VRML file format. VRML uses a text based scenegraph similar to the scenegraph structure used in IRIS Performer. The object oriented nature of the implementation makes it easy to develop a VRML scenegraph structure by creating a one-to-one map of VRML nodes with the Performer scenegraph nodes. The VRML models can be viewed with one of the available VRML plugins for internet browser. The models can be interacted with using keyboard and mouse.

5 CONCLUSIONS AND FUTURE WORK

The 3D manufacturing simulation model provides an easy and convenient method to synthesize basic manufacturing operations in a virtual environment.

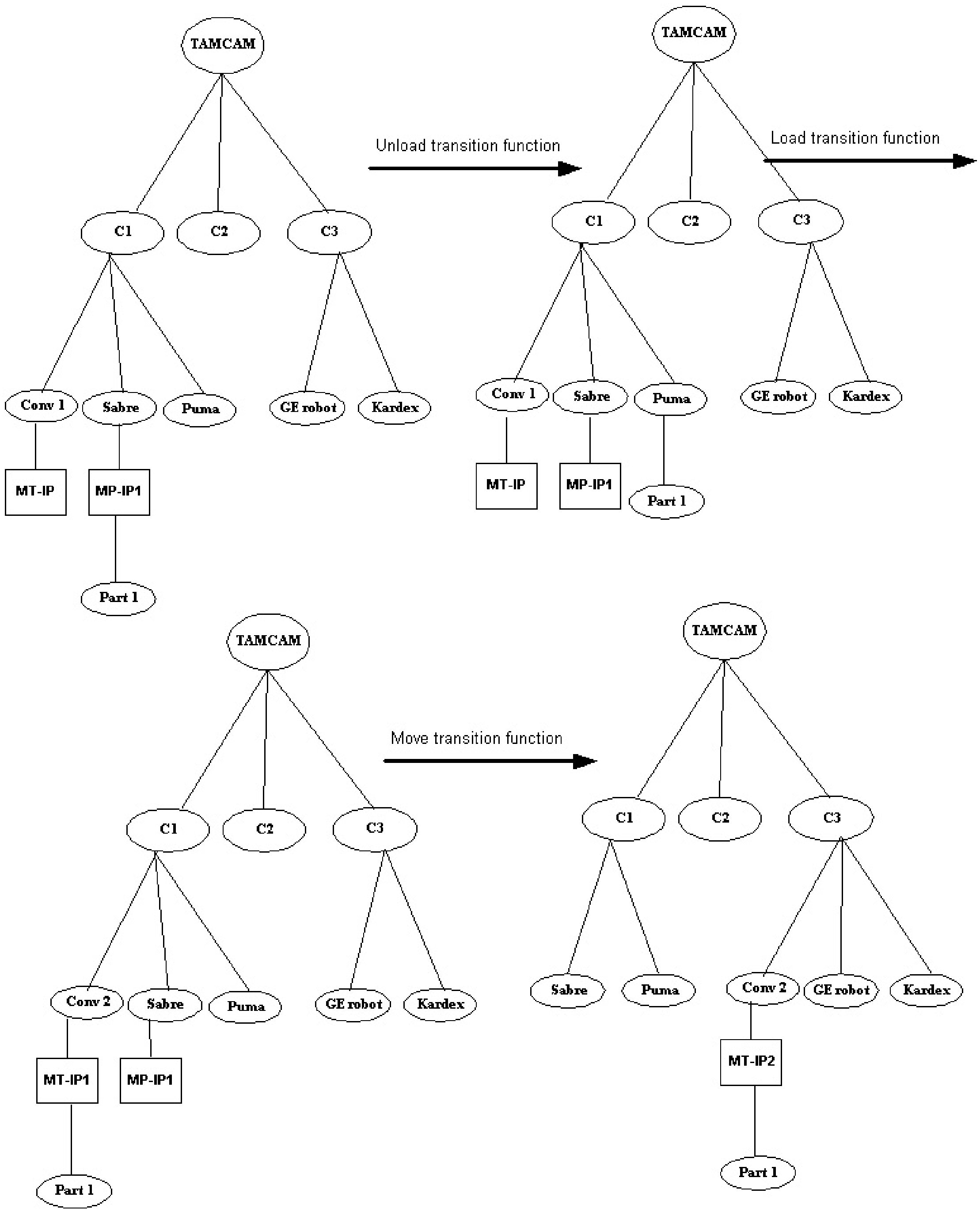

Figure 3: Basic manufacturing operation representation

The method can be extended to model specific manufacturing operations using these operations as the base. The method can be also extended to model non manufacturing scenarios. Future work include integrating with existing simulation software, and molding the 3D model as a decision making tool.

ACKNOWLEDGMENTS

The material is based upon work supported by the Texas Advanced Research (Technology) Program under Grant No. 00512-0387-1999.

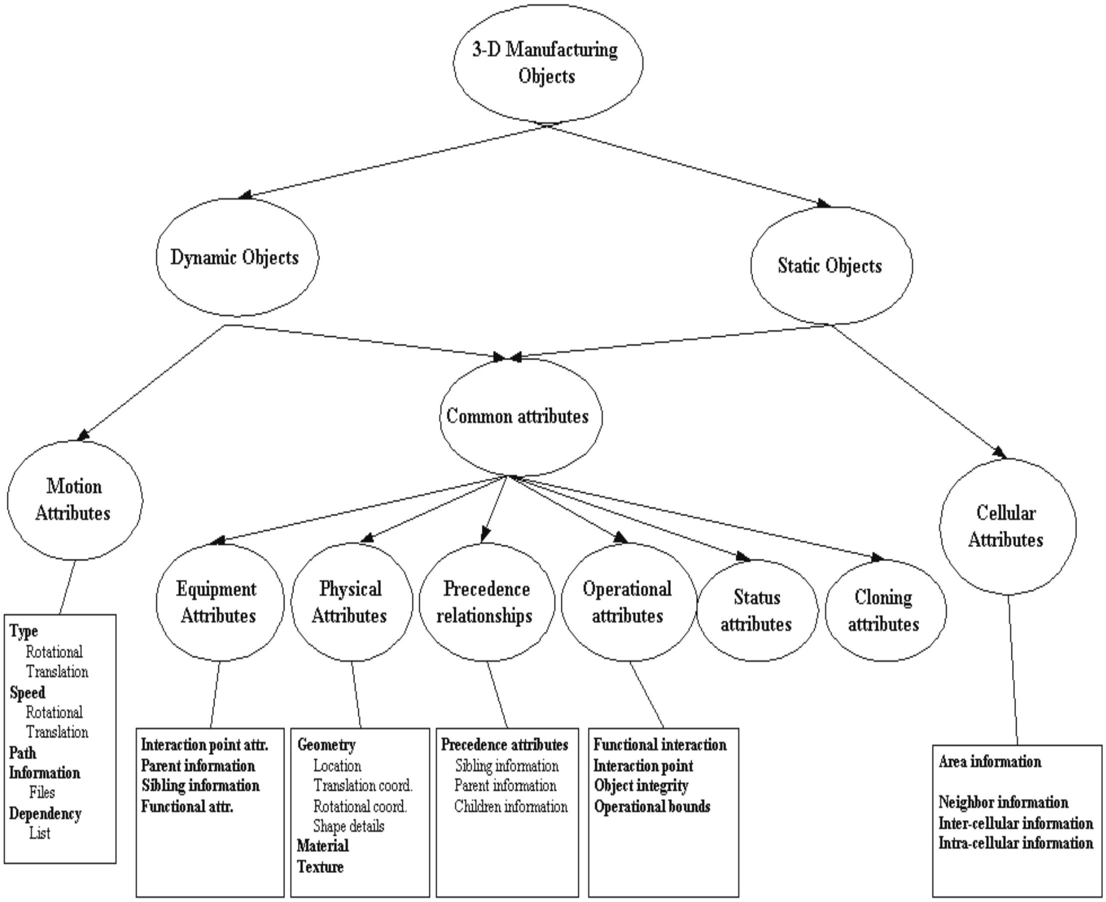

Figure 4: 3D Manufacturing Object Data Structure

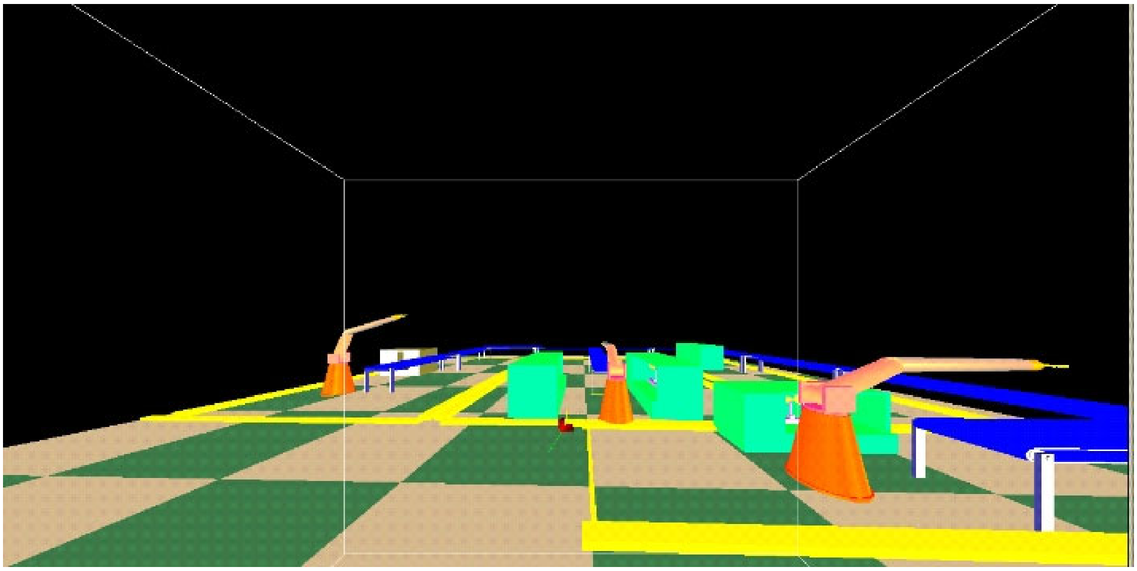

Figure 5: Snapshot of the factory model in a virtual environment

REFERENCES

Banerjee, A., P. Banerjee, T. DeFanti, A. Hudson, B. Dodds and J. Curtis. 2000. A Behavioral Layer Architecture for Telecollaborative Virtual Manufacturing Operations. *IEEE Transactions on Robotics and Automation* 16, 218--227.

Cavelib. 2001. [online]. Available online via `http://www.vrco.com/products/cavelib_main.html` [accessed August 8, 2001].

Chawla, R. and A. Banerjee. 2001. An Automated 3-D Facilities Planning and Operations Model Generator. *submitted to Journal of Intelligent Manufacturing*.

Hartman, J. and J. Wernecke. 1996. *The VRML 2.0 Handbook – Building Moving Worlds on the Web*. Reading, MA: Addison-Wesley.

Hassan, M. S., R. G. Askin and A. J. Vakharia. 1998. Cell Formatiom in Group technology: Review evaluation and Directions for Future Research. *Computers and Industrial Engineering* 34 *,* 3--20.

Performer. 2001. [online]. Available online via `http://www.sgi.com/software/performer/index.html` [accessed August 8, 2001].

Rohlf, J. and J. Helman. 1994. A High Performance Multiprocessing Toolkit for real-time 3D Graphics. *Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH)*, 381–394.

Sense8. 2001. [online]. Available online via `http://www.sense8.com` [accessed August 8, 2001].

Sowizral, H. A., D. R. Nadeau, M. J. Bailey and M. F. Deering. 1998. *Introduction to Programming with Java3D* [online]. SIGGRAPH Tutorial session. Available online via `http://java.sun.com/products/java-media/3D/collateral/class_notes/java3d.htm` [accessed August 8, 2001].

Wang, T., H. Lin and K. Wu. 1998. An Improved Simulated Annealing For facility Layout Problems in Cellular Manufacturing Systems. *Computers and Industrial Engineering* 34 , 309--319.

Wysk, R. A., B. A. Peters and J. S. Smith. 1995. A Formal Process Planning Schema for Shop Floor Control. *Engineering Design and Automation* 1, 3--19.

AUTHOR BIOGRAPHIES

AMARNATH BANERJEE is an Assistant Professor of Industrial Engineering at Texas A&M University. He received his Ph.D. from University of Illinois at Chicago in 1999. His research interests are in virtual reality, virtual manufacturing, telecollaboration, remote decision making, facility design, and digital video processing. He is a member of IIE. His email address is `<banerjee@tamu.edu>`.

RAVI CHAWLA is a consultant with J.D. Edwards. He received his MS in Industrial Engineering from Texas A&M University in 2001. His research interests are in virtual reality, facility design, databases, and web based application design. His email address is `<cravi@tamu.edu>`.