VRSPATIAL: DESIGNING SPATIAL MECHANISMS USING VIRTUAL REALITY

Judy M. Vance[1]
Department of Mechanical Engineering
Virtual Reality Applications Center
Iowa State University
Ames, Iowa 50011
jmvance@vrac.iastate.edu

Pierre Larochelle
Department of Mechanical Engineering
Florida Institute of Technology
Melbourne, Florida 32901
pierrel@fit.edu

Denis Dorozhkin
Department of Mechanical Engineering
Virtual Reality Applications Center
Iowa State University
Ames, Iowa 50011
dorodv@vrac.iastate.edu

ABSTRACT

The traditional computer interface provides a two-dimensional (2D) interface to design. The monitor is a two-dimensional visual display and the desktop mouse is a two-dimensional input device. These tools are sufficient for designing 2D products such as planar linkages. When these tools are used to design three-dimensional (3D) products however, the designer must learn how to use the 2D tools to interact with objects defined in three-dimensional space. Virtual reality technology provides truly three-dimensional tools, with position tracked visual displays and input devices, that allow interaction with 3D digital products. Over the last few years we have developed a virtual environment for spatial mechanism design [1-4]. This paper describes the differences between designing a spatial mechanism using traditional computer interfaces and designing using the VRSpatial application.

Keywords: virtual reality, immersive environments, user interface, spatial mechanisms.

INTRODUCTION

Planar mechanisms can be easily designed using the traditional two-dimensional computer interface such as the monitor, keyboard and mouse. For motion generation tasks, locations can be placed in the plane of the monitor and the mouse can move around in the 2D design space. The final mechanism can be displayed and animated to verify that it meets the design objectives. Designing spatial mechanisms presents different challenges. The design input is fully three-dimensional and the resulting mechanism moves in three-dimensional space. Therefore, to best design spatial mechanisms, a three-dimensional interface is needed.

Virtual reality (VR) technology removes the traditional computer interface of the keyboard, monitor and desktop mouse and replaces it with a position tracked stereo visual display and a position tracked input device (Fig. 1). This interface allows the user to interact with the design by visualizing and moving around in 3D space. Two of the more common VR visual displays are the head mounted display and the CAVE environment. The head mounted display is a helmet that is worn by the user. Separate left eye and right eye images are displayed on small CRT or LCD lenses in the helmet. A device is attached to the helmet to provide for position tracking. As the user moves around in space, the views displayed on the lenses shift to match the user's viewpoint. The CAVE environment supports visual display by using large projection screens that form the side, bottom and/or top surfaces of a room. Left eye and right eye images are projected on the screens and are synchronized with LCD shutter glasses so that the left eye only receives the left eye image and the right eye

---

[1] Corresponding Author

only receives the right eye image. For either visual display, several different input devices are available. Gloves that can sense finger orientation, gloves that are used to gesture for input, and multi-button mice are common. Each of these input devices is also attached to a position sensor so that the position of the input device can be used in the virtual reality program.

This paper outlines some of the differences in using a traditional computer interface to a virtual reality design environment when designing spatial mechanisms. The VRSpatial application has been developed as a design environment for the creation of spatial 4C mechanisms.

DESIGNING SPATIAL MECHANISMS

The VRSpatial software is written in C^{++} and uses the VRJuggler software library. The speech recognition is performed using IBM ViaVoice™ and IBM Speech for Java along with the OmniORB2 implementation of the CORBA standard. Currently VRSpatial is implemented only for SGI® computers but will soon be converted to a PC-based application. The current version of the software does not impose excessive computational load on the machines; therefore, no significant reduction in the performance is anticipated after conversion. This section will describe the differences between VRSpatial and traditional interfaces for each of the steps in the spatial mechanism design process.

The synthesis of spatial mechanisms for four location motion generation can be divided into the following steps:
1. Specify the position and orientation of the desired location of the movable object
2. Generate all possible solutions
3. Select a candidate solution
4. Evaluate the candidate solution
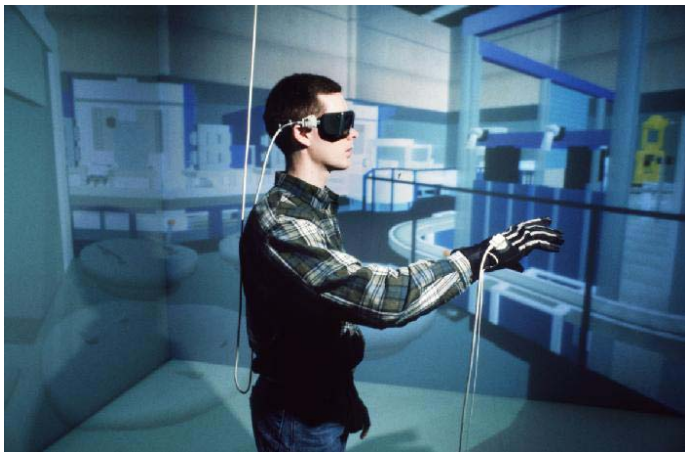5. Iterate to the best design

Figure 1. Virtual environment showing position-tracked visual and input devices.

Specify the position and orientation of the desired location of the movable object

In designing spatial mechanisms, there are two common ways to specify the position and orientation of the desired object: explicitly define the 3D coordinates of the location or manipulate the movable object in space into the correct location. In practical terms, most design project requirements consist of "move this from here to there" where "here" and "there" are related in space to locations of other objects in the area. For example, a pick and place task would be defined by specifying that the object move from the conveyor off-load station to the work-in-process conveyor. The start location would be a part sitting on the off-load station. The end location would be the part sitting on the work-in-process conveyor. Additional constraints might be to avoid other objects in the environment while moving.

In the traditional interface, multiple 3D locations can be input by entering the 3D coordinates and angles or moving objects in the space using the monitor, mouse, and keyboard. Determining 3D coordinate values for positions and orientations is not always easy. Measurements can be taken from the surrounding geometry in the real environment or from a CAD drawing of the area. Placing the parts in the work area using the monitor, mouse and keyboard is also difficult. Either multiple views are needed or multiple reorientations of the work area have to be performed to get the 3D placement just right. Adjustments in locations are awkward as well.

When implemented in a virtual environment, placing objects in the desired locations relative to other parts in the area is straightforward (Fig. 2). Following the 'design in context' approach geometric models of the objects in the work area can be loaded into the design environment. This provides the user with the ability to investigate interaction of the mechanism with other objects in the work environment and to design mechanisms within the physical constraints imposed by the work environment. While looking through the stereo viewing device, the user moves to the desired location and uses the position tracked input device to release a digital representation of the part in the correct location. Adjustments are made by picking up the part and moving it until the user is satisfied with the position and orientation of the design locations. A digital readout of the part location and orientation can be shown in the virtual environment and adjusted to get to the exact location if the 3D coordinates are known.

Figure 2. Placing locations in the VRSpatial application.

Generate possible solutions

The input to the solution-generation algorithm consists of the positions and orientations of the specified locations. This information can be taken from the locations of the movable parts in either the traditional interface application or the virtual reality application.

Pick a solution

A dedicated set of C++ routines has been developed in order to provide the user with the ability to interact with the generated solution space. There are two methods that are used to pick a solution: selection from the type map or selection from the fixed and moving congruences. The 4C spatial mechanism is constructed from two dyads. The type map is a two dimensional graph of all possible solutions with the driving fixed axes plotted along the vertical axis of the graph and the driven fixed axes plotted along the horizontal axis of the plot (Fig. 3). Points at the intersection of a driven axis and a driving axis create a unique mechanism and are classified according to mechanism type so the user can pick solutions based on mechanism type. This method of selection is especially tailored to be used with a traditional computer interface. This method plots the data on a 2D graph, making selection using traditional input devices of the mouse and keyboard very easy. However, there is no information on the geometry of the mechanism that is carried with the type map. The only information that the designer has is the type of mechanism that is selected.

Congruences on the other hand are spatial entities. They are infinite planes of lines that represent all possible solutions. Once again, two selections are needed to uniquely specify a mechanism. Here, however, selecting one line from the fixed congruence automatically selects the corresponding line from the moving congruence, and similarly, selecting one line from the moving congruence automatically selects the corresponding fixed line. Because these congruences are infinite planes in 3D space, it is very difficult to select from this set using the
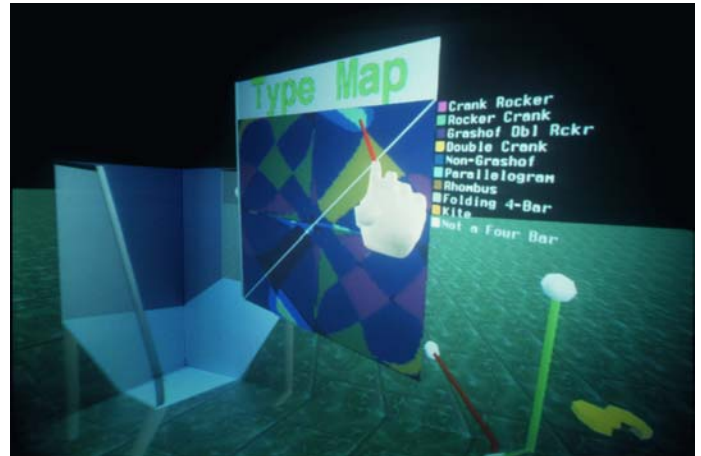

Figure 3. Type map selection.

traditional monitor, mouse and keyboard. In a virtual environment these planes are placed in their spatial orientation with respect to the mechanism design locations (Fig. 4). A designer simply moves around the space and selects from various congruences until an acceptable design has been achieved.

In VRSpatial, the type map selection procedure and the congruences have been linked. When selecting from the type map, the corresponding planes on the congruences are highlighted. When selecting lines from the congruences, the corresponding point on the type map is highlighted.
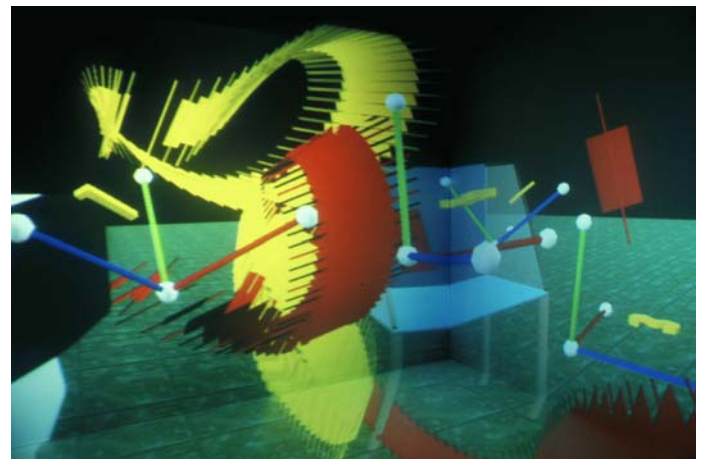

Figure 4. Congruence planes.

In this manner, designers can get a feel for the 3D design space using the congruences and also be able to pick a solution using mechanism types as a selection criterion.

Verify the mechanism motion

Using traditional interfaces, the mechanism can be animated in 3D space but must still be displayed on a 2D monitor. In order to verify the mechanism motion, the

3

environment must be rotated within the 2D monitor. In the virtual environment, the mechanism can also be animated. Here, however, the designer has the ability to walk around the space and examine the mechanism as it moves throughout the work area.

Iterate to the best design

Once a mechanism is designed, small adjustments might be needed to size the mechanism to fit within the space available. The same issues of placement of design locations and selection of possible designs are present in the iteration process.

MAJOR LESSONS LEARNED

Here we summarize what we have learned about using virtual reality techniques for the design of spatial 4C mechanisms. Most, if not all, of these lessons learned apply equally well to the design of all 3D mechanical systems.

The common practice of using coordinate frames to define the locations of bodies in traditional graphic representations is also desirable in a virtual environment. Though the natural instinct in a virtual environment is to represent all bodies with solid geometric models, the designer benefits from having the ability to interact with either the geometric model itself or a coordinate frame. This may be due to the current features present in the user interface technologies available in virtual environments. Future high bandwidth haptic interfaces with force feedback may provide the designer with sufficient "feel" for the solid models that coordinate frames would no longer be useful.

The evaluation of a candidate design requires a careful examination of the entire motion cycle of the mechanism. For example, when evaluating a candidate solution for a planar four-bar mechanism designed to achieve four finite positions, the designer knows that the motion of all parts of the mechanism is confined to the plane and that all of the links at all times will remain inside a bounded rectangular region of the plane (this region has no side longer than the sum of the two longest link lengths). This is not the case with spatial mechanisms. When evaluating a candidate solution for a spherical four-bar mechanism designed to achieve four finite orientations, the designer knows that the motion of all parts is confined to the surface of a sphere and that the radius of the sphere is a free design variable. However, for the design of spatial mechanisms, the design space is truly spatial and in nature with no obvious bounds. For example, consider a non-Grashof spatial 4C mechanism with all link lengths equal to unity. In four different configurations one or more C joint translations will approach infinity [5]. Moreover, the motion is not confined to a plane or a sphere. The moving body can and does move in non-intuitive fashions and this necessitates a careful evaluation of the entire motion cycle. The virtual reality representation of the mechanism facilitates this evaluation.

The use of a well-developed virtual environment greatly facilitates the evaluation of candidate designs to secondary design criteria. These criteria often include: interference avoidance, collision avoidance, aesthetics, and ergonomics. The ability to walk around the solution space and "peer into" constrained spaces gives the designer a three-dimensional view of the mechanism and how the motion relates to other objects in the design space.

It is important to avoid providing the designer with too much visual information. It is easy to create a virtual environment that provides a very interactive and detailed virtual environment in which to work with highly texture mapped surroundings. However, our experience has been that focused presentation of information, as requested by the designer, is the most efficient. We have found that by letting the designer request detailed information when it is needed the designer feels most comfortable working in a virtual representation of the design workspace. Utilization of these concepts in the fully-immersive environment of a VR system has provided mechanism designers with a powerful design tool that is easy to use and yet effective.

SUMMARY

In this paper we have described our efforts to develop a virtual environment for spatial mechanism design entitled VRSpatial. Our focus has been upon discussing the differences between using a traditional computer interface and using the VRSpatial application. In addition, we provide a summary of lessons learned from our efforts that we believe are applicable to the creation of virtual design environments for all 3D mechanical systems. It is our hope that this and future efforts will further demonstrate the effectiveness of virtual reality based design environments.

REFERENCES

[1] Furlong, T., Vance, J., and Larochelle, P., 1999, "Spherical Mechanism Synthesis in Virtual Reality", ASME Journal of Mechanical Design, **121**, No.4, pp. 515-520.

[2] Kihonge, J., Vance, J. and Larochelle, P., 2001, "Spatial Mechanism Design in Virtual Reality with Networking", *Proc., ASME 2001 International Design Engineering Technical Conferences,* Pittsburgh, Pennsylvania, September 9-12, 2001. ASME Press.

[3] Larochelle, P. and Vance, J., "Interactive Visualization of the Line Congruences Associated with Four Finite Spatial Positions", 2000, *Proc., Symposium Commemorating the Legacy, Works, and Life of Sir Robert Stawell Ball upon the 100th anniversary of the publication of his seminal work "A Treatise on the Theory of Screws".* University of Cambridge, Trinity College, England, July 9-11, 2000.

[4] Vance, J. and Larochelle, P., 2001, "Creating a Virtual Environment for Spatial Mechanism Design", *Proc., 2001 NSF Design, Service and Manufacturing Grantees and Research Conference,* Tampa, Florida, January 7-10, 2001.

[5] Larochelle, P., "Circuit and Branch Rectification of the Spatial 4C Mechanism", *Proc., ASME 2000 International Design Engineering Technical Conferences*, Baltimore, Maryland, September 10-13 2000. ASME Press.