Computer-aided configuration of modular robotic systems

by Richard Hooper and Delbert Tesar

University of Texas at Austin

A system has been developed to interactively assemble modular and reconfigurable robotic systems in a computer animation environment. The modular robot is based on a set of one, two and three degree of freedom joint modules and generic links. These modules may be assembled into a large class of robotic systems that include serial, parallel, mobile and hybrid configurations. A model of the configured robotic system is immediately available for use in a wide variety of robotics research areas, including: obstacle avoidance, redundant inverse kinematics, dynamic simulations, mobile platforms and world model databases.

Introduction

Computer animation provides a means of viewing robot motion to aid in human perception and decision making for both design and operation. Current applications of computer animation to robotics have focused on the simulation and programming of existing robots. The most common uses are workcell design, offline programming and the promotion of research programs. Since robots are usually purchased in their final configuration, there has been little demand for computer applications that aid in the design of the robot itself. The development of a modular reconfigurable robotic architecture presents an excellent opportunity to apply computer animation to the early stages of the robot design.

Interactive software packages that generate computer animations have found wide acceptance for programming and simulating industrial robots. Animated workcell design involves graphically placing the robot in its environment, also called the workcell. Machines, tools, parts and any other objects that the robot will interact with are also placed in the workcell. Computer animation is then used to visually simulate these interactions as the robot performs its task.[1] Offline programming uses animation to replace the actual robot while programming the robot's motion and its interactions with its environment. This allows the actual robot to remain in service while motion programs are being developed and thus decreases costly downtime.[2] Research program promotion is also a very important application of computer animation. Computer animation can be used to effectively convey ideas while an impressive graphical simulation of a robot performing a complex manoeuvre will enhance a company's or research program's high-tech image (Fig. 1).

Current robots are purchased in their final configuration from the robot manufacturer. These robots are typically designed to perform a specific class of tasks, and if the application changes significantly the robot is rendered obsolete. The development of a generalised modular robotic architecture using a set of one, two and three degree of freedom joint modules and generic links greatly reduces the threat of obsolescence to the robotic system.[3] A modular architecture also represents an excellent opportunity to use interactive computer animation in the development of robot technology through research enhanced by the use of three-dimensional computer animation as a visualisation tool.

Available graphics workstations can produce very smooth animations of a robot manipulator moving in a complex environment. Dedicated graphics hardware performs the calculations necessary to display solid-surface models on the computer screen. These images can

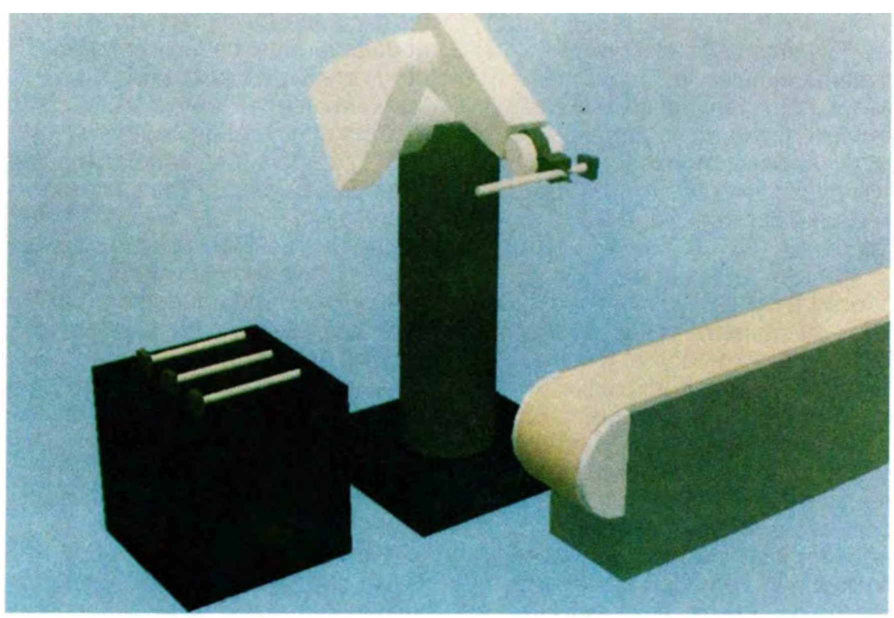

Fig. 1 Simulation of a common industrial robot performing a pick and place operation

attributes	applications
module based user friendly able to interactively assemble an extremely large class of robotic systems requires no computer graphics experience models may be immediately animated configuration model is immediately available	obstacle avoidance redundant inverse kinematics dynamic simulation real-time calculations manual controllers parallel robots mobile robots hybrid systems world model databases

Fig. 2 Attributes and research applications of interactive software package to graphically assemble reconfigurable robotic systems

be displayed in colour, three-dimensions, perspective and with hidden surfaces removed, resulting in very realistic animations. The main drawback to using these graphics workstations is that writing the programs for computer animation is complex and quite time consuming, often requiring thousands of lines of code to produce a single animation. The difficulty and significant time investment in writing the computer animation code has led to the development of application programs that aid in the production of computer animations on graphics workstations.[4]

A software package under consideration in this article has been developed to interactively assemble and animate robotic systems. This package, which is based on a generalised modular and reconfigurable robotic architecture, facilitates the application of computer animation to robotics research, design and operation. The possible uses of computer animation in robotics research are many. They include the development of obstacle avoidance techniques, redundant inverse kinematics algorithms, dynamic simulations, real-time calculations, manual controllers, serial configurations, parallel configurations, mobile robots, hybrid systems, layered structures with several input scales and world model databases (Fig. 2).

Modularity

Modular design is the construction of large systems from smaller discrete units. The benefits to approaching the design of robotic systems from a modular perspective are numerous. From a finite set of one, two and three degrees of freedom joint modules and generic links, a general robotic architecture can be developed. The modular robot is reconfigurable to perform varied tasks and thus reduce obsolescence.[3] The design of the robot is simplified and cost can be reduced by using the same modules in a wide variety of configurations. The integration of technology is facilitated because a new robot does not have to be built in order to test a new idea. The cost of sending payload into space is tremendously expensive, thus making very attractive a robot that could perform many different tasks. Every robot that is sent into a nuclear environment immediately becomes contaminated and must be decontaminated or disposed of. This reality also makes attractive the use of a robot that can be reconfigured to perform many tasks.

There are numerous issues involved in the development of a modular and reconfigurable robotic system. The intelligence and decision making system that will deal with the inverse kinematics of a reconfigurable and possibly redundant system must be developed. A controller that can adapt to reconfiguration of the system under control must also be designed and built.[5] Indeed modularity must be incorporated into every aspect of the robot's design in order to have a truly modular and reconfigurable system.

The creation of an initial computer animation system for modular and reconfigurable robots does not require that all of these issues be resolved. This has allowed the development of the modular and reconfigurable animation system to lead the development of the actual physical system by several years. The computer animation can now be used as a tool in the development of the technology necessary to realise the creation of a truly modular and reconfigurable robotic system.

A modular approach to robot design also provides many direct benefits to the computer animation of robot systems. The modular computer animation package builds graphical robots that are inherently reconfigurable to animate many different robot configurations performing a wide variety of tasks. Based on this package, the creation of each animation is simplified to the extent that no graphics experience at all is necessary to quickly create high-quality animations. The performance of the computer animation is also increased because modularity allows many calculations to be done prior to beginning the actual animation sequence.

Software

A software package has been developed to interactively assemble three-dimensional computer animations of robotic systems. This package was completely designed and implemented by the authors. It is written in the C programming language. One, two and three degrees of freedom joint modules and generic links are used as the basis of this modular and reconfigurable animation package. These modules may be assembled into a large class of robotic systems that include serial, parallel, mobile and hybrid configurations. The animated model is created by specifying which modules are used and how they are connected.

The main menu bar is at the top of the screen and contains the options 'FILE', 'BUILD', 'RUN' and 'UTILITIES'. These menus are of the pull-down variety and are activated by positioning the cursor over the desired option and clicking the mouse button.

The 'BUILD' menu is used to sequentially assemble the modules into the robotic structure. The building procedure relies on the idea of a drawing frame. The frame is translated and rotated to provide a local origin for each module. Each module is added to the robot model relative to this frame and may change the frame before adding the next module.

In general each module uses six arguments that may move the drawing frame before a module is added and six arguments that may move the drawing frame after the module is added. The program calls these the input frames and the output frames, respectively. The modules also need to be scaled. The scaling parameters are the diameter, length, height, width etc.

In the 'BUILD' menu are the choices 'add joint', 'add base', 'add link', 'end effector', 'environment' and 'parallel'. The parallel dialogue box asks for six arguments that specify the position and orientation of a branching chain relative to the current position of the drawing frame. The other dialogue boxes will ask for a more specific choice of which link, joint, base, end effector or object in the environment is to be built with. Modules are added relative to the position and orientation of the drawing frame active when the build option was selected.

The parallel option in the menu tells the program that the current serial chain is branching. The parallel dialogue box asks for six parameters that will specify the position and orientation of the branching frame relative to the current drawing frame. The program will continue to add modules to the branching chain until an end effector is added to the chain. The end effector is a signal to the program to terminate the current chain and return to the drawing frame that was active when the parallel option was invoked. Terminating the first chain with an end effector will cause the initial frame to become active. The parallel option can also be used to create multiple robots operating independently in the same environment.

A mobile robot is created by attaching degrees of freedom to the base frame of the robot. There is a moving reference frame among the choices of bases. This frame takes six parameters and can be used to create a mobile robot that can move to any position and orientation in the environment. The moving reference frame can also be used to create objects that can move around in the environment, for example a pick and place operation or a falling object. Sliders and revolutes can be used to create a mobile robot whose base frame motion has less than six degrees of freedom.

A hybrid robot has both parallel and serial parts. The hybrid robot can be assembled by branching into multiple serial chains and then specifying the proper joint angles to close some of the chains. A hybrid robot can also be created by adding inherently parallel modules, such as the spherical shoulder, to a serial chain. The parallel modules do not require that all of the interior joint angles be specified. The shoulder, for instance, only requires roll, pitch and yaw and then will automatically calculate the proper interior joint angles.

The environment menu is for placing objects in the environment. Each environment module requires six arguments that position and orient the module in relation to the initial reference frame. Objects added from the environment menu are fixed in their position and orientation and do not affect the active drawing frame.

The animation effect is created by showing rapidly in succession the robot scene with the joint angles changed by a small amount in each scene. This gives the illusion that the robot is moving. The models created with this program can be animated by accepting data from the keyboard, from data files or in real-time via shared memory.

The simplest way to animate the robot is to use the keyboard to increment or decrement each joint directly. The joints are numbered in the order that they were added to the model. The specific keys and the joints that they affect are shown in a small box that appears near the top of the screen. Animating from the keyboard is useful for demonstrations and testing.

Running the animation from a data file is an excellent method of viewing the results from inverse kinematics algorithms or dynamic simulations. Each data set in the file represents the state of the robot at a given point in time.

Running the program from shared memory allows the real-time animation of data as it occurs. Selecting 'shared memory' from the 'RUN' menu will cause the program to create a shared memory segment. The program will continuously read from this shared memory and display the robot in the position defined by the joint angles until 'end run' is chosen in the 'RUN' menu.

Example

This is a simple example showing the construction of a serial robot arm mounted on a fixed pedestal. After running the modular animation program, a window appears with the words 'FILE', 'BUILD', 'RUN' and 'UTILITIES' at its top edge. The available bases may be seen by selecting 'add base' from

Fig. 3 Menu of available joint modules

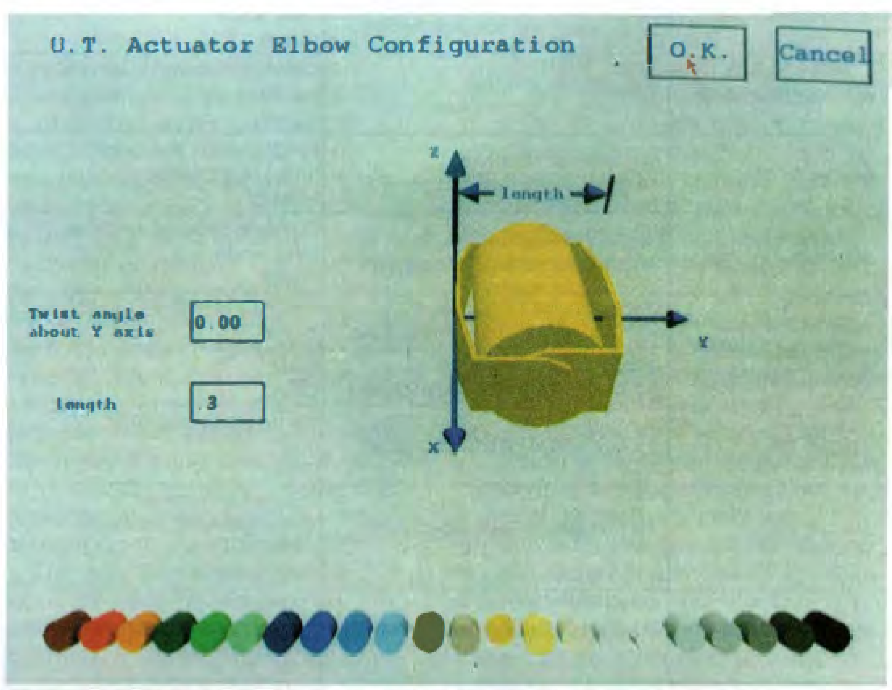

Fig. 4 Scale and select the colour of the joint module

the 'BUILD' menu. After choosing a base for the robot, a dialogue box will appear with a three-dimensional display. A revolute mounting pedestal is used in this example. Either the slanted or the horizontal stand and the colour of the pedestal may be selected by clicking on the appropriate button. The button should be pushed in to verify the choice. Clicking on the 'OK' box adds the base to the robot model.

The elbow actuator is chosen from the 'add joint' option in the 'BUILD' menu (see Fig. 3). The module can be scaled by specifying the length as shown on the display (see Fig. 4). The length is specified by clicking on the box next to the word 'length' and then typing the length using the keyboard. After choosing the colour, the module is added to the robot model.

The inline roll module is next chosen from among the joints in the 'add joint' option under the 'BUILD' menu. After specifying the two lengths shown in the dialogue box and choosing the colour, this module is added to the robot model. The conical link is chosen from among the links in the 'add link' option under the 'BUILD' menu. The length, input diameter, output diameter and colour are specified. The y translation of the output frame is chosen to be the same as the length so that the next module will be added onto the end of the link. Modules can be added until the graphical manipulator is completed (see Fig. 5).

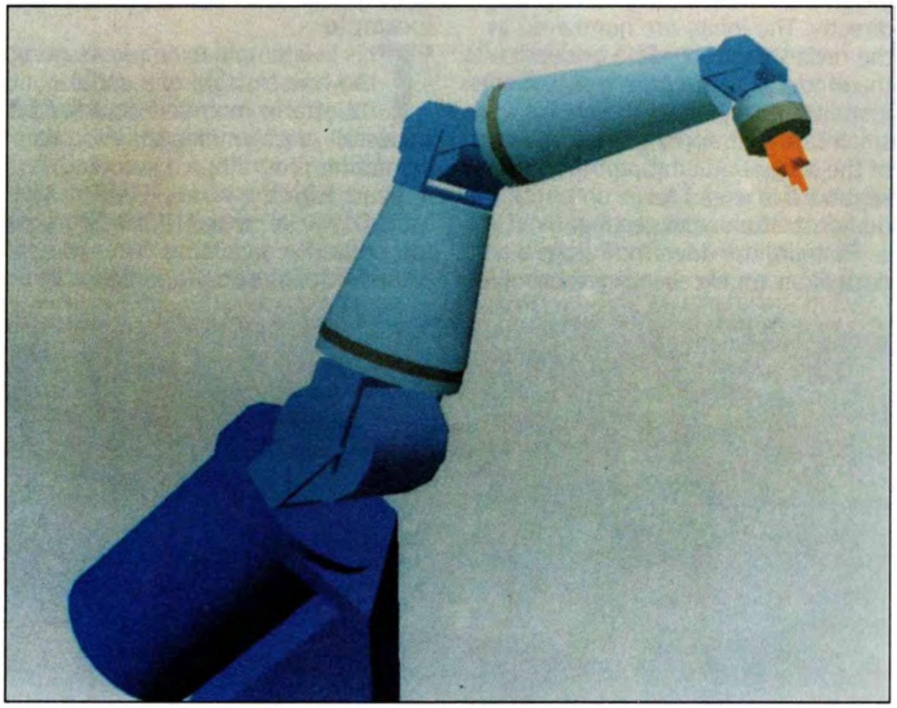

Fig. 5 Finished modular robot graphic

module	inherent features	user-defined features
actuated elbow	one internal rotation 50 polygon surface description	distance between connections twist angle colour
actuated inline roll	one internal rotation 40 polygon surface description	distance between connections radius colour
prismatic joint	one internal rotation 14 polygon surface description	length, width and height axis of translation colour
actuated knuckle	two internal rotations 80 polygon surface description	distance between connections twist angle colour
parallel shoulder	three internal rotations 112 polygon surface description	distance between connections colour
parallel jaw gripper	one internal translation 35 polygon surface description	scale colour
I-beam link	22 polygon surface description	connection geometry scale colour
conical link	16 polygon surface description	connection geometry input radius and output radius length and colour
trapezoidal link	6 polygon surface description	connection geometry input radius and output radius length and colour

Fig. 6 Specific features associated with animating each module

Research applications

The incorporation of modularity and reconfigurability into the computer animation of robotic systems facilitates the application of computer animation to many areas of robotics research. Animations of an extremely large class of robotics systems can be created by simply assembling together modules from a finite set of one, two and three degree of freedom joint modules and generic links. The creation of these animations is simplified to the extent that a model of the Robotics Research dual arm system with torso (see Fig. 8) may be assembled in approximately ten minutes.

Computer animation is an effective method of visually representing kinematic data, such as might be generated by obstacle avoidance algorithms, redundant inverse kinematics routines and as the output from dynamic simulations. Modular and reconfigurable animation can be used in the development of serial, parallel, mobile and hybrid manipulators and is ideally suited to the development of modular and reconfigurable robotic systems (Fig. 6). A modular approach to animation may be used to generate world model databases that include multiple robots and obstacles that may be fixed to a reference frame or may move about in the environment.

Obstacle avoidance refers to moving the robot while avoiding collisions with objects in the environment. The obstacles may include the robot itself, fixed objects in the environment, moving objects in the environment and also other robots that may be operating in the same workspace. The cost of an industrial robot colliding with an obstacle in the environment could be very high, both in terms of repair to the robot and the environment as well as the cost of lost productivity due to downtime. The results of a collision in a more sensitive environment, such as space, nuclear or military applications could be enormous. The cost of such collisions makes obstacle avoidance of fundamental importance when planning the robot's path. Computer animation of modular and

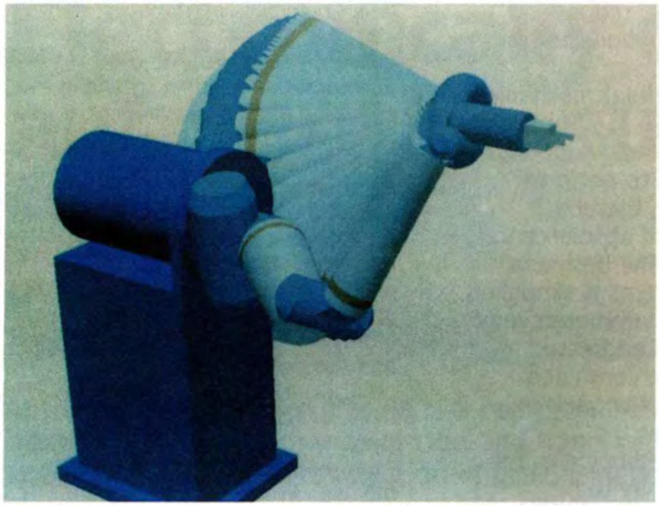

Fig. 7 Self-motions of a redundant manipulator

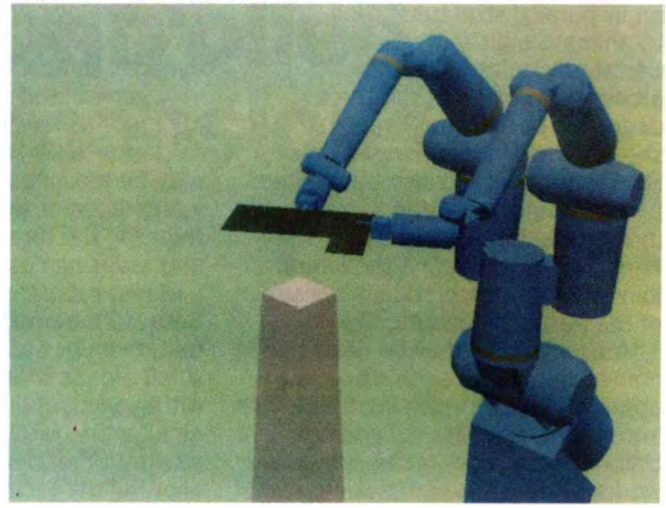

Fig. 8 Seventeen degree of freedom redundant manipulator

reconfigurable robotic systems can be used as a tool in obstacle avoidance research. The same database that is used to generate the modular robot display can also be used as a database for the obstacle avoidance algorithms.

A redundant robot has more degrees of freedom than are necessary to specify the state of the end effector. Any robot with seven or more degrees of freedom is redundant because it only takes six parameters to specify the position and orientation of an object in three-dimensional space. The extra degrees of freedom can be used to improve the performance of the system by allowing the end effector to reach the goal with many different joint paths (see Fig. 7). Different criteria can be used to determine which joint paths are best for a given situation.[6] Walking machines, multifingered grippers and 'snake' manipulators are all examples of redundant robotic systems.

By definition, redundant robots have extra degrees of freedom (Fig. 8). More computer programming time is necessary to generate the forward kinematics and visual simulation for these extra degrees of freedom. This time is significantly reduced by using computer animation based on a generalised modular and reconfigurable mechanical architecture. A robot with extra degrees of freedom simply has more joint modules and generic links. The many criteria that may be used by the inverse kinematics schemes may also be presented graphically by means of computer animation.

Dynamic modelling of robotic systems is an active area of robotics research.[7,8,9] Dynamic modelling incorporates mass, compliance and damping to simulate the system response to joint forces and external loads. Computer animation can be used to display the results from these dynamic simulations.

Dynamic models can be used to evaluate the performance of a robot as it performs a task. The condition number and singular values of the Jacobian matrix may also be incorporated into an evaluation procedure.[10] The state of the inertia matrix has been used as a measure of the robot's dynamic ability.[11] The dynamic model can be incorporated into feedforward control algorithms. Feedforward control incorporates the system dynamics into the control algorithm in order to synthesise the output. The dynamic model for the robotic system may be developed in terms of kinematic influence coefficients.[12] The kinematic influence coefficients employ the geometry of the robot to relate the system's dynamic characteristics as they appear at any input to the system.

An animated robot can exhibit high speed, high precision, no deformation and no backlash. This is, of course, a tremendous simplification. A modular and reconfigurable robotic architecture presents an excellent opportunity to combine dynamic simulation and computer animation. Constructing the graphical robot on the screen defines the modules that make up the robot and the geometry. The dynamic description can be stored as a feature within the modules. As each module is added to the robot, the dynamic equations could be automatically generated using the geometry and the dynamic description of the modules.[13,14]

The joint angles of a parallel robot may not all be specified independently. A four bar planar mechanism, for example, has four joints but only one degree of freedom. By specifiying the angle of one joint the angles of the other joints are constrained. The parallel robot typically has a smaller workspace than the serial robot, but there are also many benefits associated with using a parallel structure. The parallel structure can utilise mechanical advantage to increase its load carrying capacity. The parallel robot allows a choice among the joints as to which ones will be used as the inputs to the system. Several degrees of freedom can be obtained while still allowing direct actuation with the actuators fixed to the robot base.[8] Redundant actuation is also possible with parallel structures. The redundant actuators can provide fault tolerance and antagonistic actuation.[15] Antagonistic actuation shows promise for high-precision tasks that require disturbance rejection.

Computer animation of modular and reconfigurable robotic systems can be used to animate parallel structures. The parallel structure is simply built from open chains. The chain is closed to form a parallel structure by properly specifying each joint angle. This method for creating the animation is of particular benefit for the animation of data that has been generated by kinematic and dynamic analysis that treat the parallel chain as a set of constrained serial chains.[7] The parallel chain is kinematically cut, and the resulting serial chains are constrained to have zero relative position, velocity and acceleration at the cut.

Hybrid systems can be graphically created and animated using the generalised modular mechanical architecture. The modular animation is created by simply specifying which modules are used and how they are connected together. More than one chain can be connected to a link to create a branching effect. The extra chain can then be closed to create an in-parallel structure or left open to animate multi-arm systems. Certain modules within the architecture, such as the three degree of freedom spherical shoulder, are inherently parallel. Hybrid systems will also be created by simply incorporating these modules within a serial chain. The equations which define the interior joint angles are stored as a feature of the parallel modules.

A world model may be described in many different ways. One possible approach involves dividing the world into a three-dimensional grid. Unfortunately a ten metre cubical workspace described in this way with a resolution of one millimetre has 10^{12} elements. This is quite a large data base to be repeatedly faced. Other methods of describing the world model are more object oriented. A list or graph of the type and location of objects in the environment can be created.[16] Modular computer animation may be applied to either grid-based or object-oriented types of world model description.

Conclusions

A system to interactively assemble modular and reconfigurable robotic systems in a simulation environment has been discussed. A finite set of one, two and three degree of freedom joint modules and generic links forms the basis of a general mechanical architecture. These modules may be scaled and assembled to form an extremely large class of robotic systems. Computer animation is currently used in the programming and simulation of existing industrial robots. There are, however, many benefits to be gained by the incorporation of modularity, both in the computer animation of the robotic system and in the design of the actual robot. Finally, the application of modular computer animation to many current robotics research topics has been discussed.

The incorporation of a generalised modular mechanical architecture greatly simplifies the creation of computer animations of robotic systems. The modular computer animation may be quickly constructed with objects that the robotics engineer is familiar with, such as links and joints. A modular approach to computer animation allows increasingly complex systems to be quickly constructed and easily used by the design engineers involved in the development of new robotic systems and technologies. The animation may be used to visually present kinematic data such as might be generated by obstacle avoidance algorithms, redundant inverse kinematics routines and from dynamic simulations.

A generalised modular mechanical architecture has much to offer the field of robotics. Modular robots may be reconfigured to perform many different tasks, thereby reducing the threat of obsolescence and reducing costs. The design of modular robotic systems is simplified because the design parameters may be broken down and addressed in small groups that are contained within each module. A modular architecture facilitates the integration of new technologies by allowing the technology to be incorporated into the design of a single module without requiring that the entire system be redesigned.

In this regard, it now becomes feasible to collect lessons learned from modular robot test beds, from exceptional applications in space, on the ocean floor and from widely used industrial robots in a knowledge base to support expert system development to make modular component selection and system design possible by the non-specialist in robotics. This can include requirements definition from the intended application as well as guidance on criteria to select a 'best' modular design. The best modern example of this modular architectural approach is found in today's personal computers. The use of expert systems to aid in system configuration made it possible for marketing personnel to assemble VAX computers on demand to match customer requirements.[17] This level of success is essential in the field of robotics to reduce their costs, to dramatically broaden their range of applications and to produce high-value-added products with customer input as envisioned by Isaac Asimov in his early writings on robotics.

References

1 BOREN, R. R.: 'Graphics simulation and programming for robotic workcell design', *Robotics Age*, August 1985, pp.30–33

2 KOVACS, W.: 'Previewing robotics motion with computer graphics', *Robotics Age*, August 1985, pp.16–19

3 TESAR, D., and BUTLER, M.: 'A generalized modular architecture for robot structures', *Manufacturing Review*, June 1989, **2**, pp.91–117

4 LEVAS, A. and JAYARAMAN, R.: 'WADE: an object oriented environment for modelling and simulation of workcell applications', *IEEE Trans. on Robotics and Automation*, June 1989, **5**, (3), pp.324–335

5 TESAR, D.: 'A tutorial on modular architecture for robot structures', IEEE Conference on Robotics, April, 1991

6 CLEARY, K., and TESAR, D.: 'Incorporating multiple criteria in the operation of redundant manipulators', IEEE International Conference on Robotics and Automation, 1990, **1**

7 CHO, W., TESAR, D., and FREEMAN, R.: 'The dynamic and stiffness modelling of general robotic manipulator systems with antagonistic actuation', IEEE International Conference on Robotics and Automation, 1989, **3**, pp.1380–1387

8 COX, D. J., and TESAR, D.: 'The dynamic model of a three degree of freedom parallel robotic shoulder module', Proceedings of the 4th International Conference on Advanced Robotics, 1989

9 KANG, H., and FREEMAN, R. A.: 'An interactive software package (MAP) for the dynamic modelling and simulation of parallel robotic systems including redundancy', *ASME Computers in Engineering*, 1990, pp.117–124

10 YOSHIKAWA, T.: 'Manipulability of robotic mechanisms', *International Journal of Robotics Research*, 1985, **4**, (2), pp.3–9

11 KLEIN, C. A.: 'Use of redundancy in the design of robotic systems', 2nd International Symposium of Robotics Research, 1984, pp.58–65

12 THOMAS, M., and TESAR, D.: 'Dynamic modelling of serial manipulator arms', *Transactions of the ASME*, September 1982, **104**, pp.218–228

13 HERNANDEZ, E., and TESAR, D.: 'Efficient algorithms and real-time software for quasi-static manipulator deflection modelling', University of Texas Report to NASA Johnson Space Center, May 1989

14 WANDER, J., and TESAR, D.: 'Pipelined computation of manipulator modeling matrices', *Journal of Robotics and Automation*, 1987, **RA-3**, (6), pp.556–566

15 TESAR, D., PRICE, C., and SREEVIJAYAN, D.: 'Four-level fault tolerance in manipulator design for space operations', Report to NASA Johnson Space Center, June 1990

16 CROWLEY, J. L.: 'World modelling and position estimation for a mobile robot using ultrasonic ranging', IEEE Intrnational Conference on Robotics and Automation, 1989, **3**, pp.674–680

17 McDERMOTT, J.: 'A rule-based configurer of computer systems', Technical Report CMU-CS-80-119, Department of Computer Science, Carnegie-Mellon University, 1980

18 O'NEIL, S. R., and JAMSHIDI, M.: 'ROBOT_S: an interactive robot simulation language', *Robotics*, 1988, **4**, pp.245–256

The authors are with the Mechanical Engineering Department, University of Texas at Austin, BRC/MERB 160, Room 1.206, Mail Code 79925, Austin, Texas 78712-1100, USA.