Chapter 7: Road Map

This chapter outlines a vision for the future of generalized inverse kinematics and its application to advanced robotic systems. Envisioning the future is by nature a risky activity. Technological changes happen virtually overnight and can render any vision obsolete. This chapter discusses the vision in terms of ten advanced robot applications with the hope that applications will ensure relevance, at least into the near future. Each application possesses a degree of difficulty that will likely require generalized inverse kinematics. Generalized inverse kinematics cannot, however, be applied without the benefit of supporting technology. This chapter therefore discusses a broad spectrum of supporting technologies in terms of their relevance to the application of advanced robotic systems.

7.1 ADVANCED APPLICATIONS

There are several features distinguishing an advanced application. Some of these applications are very demanding kinematically. The robot must operate in very cluttered or constrained environments. These applications call for a redundant robot with at least several extra degrees of freedom. Highly redundant robots require an inverse kinematics method insensitive to singularities. Other advanced applications demand a modular robot. The modular robot is reconfigurable to perform a variety of tasks. Reconfigurability is important in manufacturing operations and in space or nuclear environments. Reconfigurable robots require an inverse kinematics method general with respect to geometry. Applications requiring very precise calibrated robots also require an inverse kinematics method general

with respect to geometry. Fault tolerance is very important in space and nuclear applications. Serial kinematic redundancy is one way of achieving fault tolerance. Achieving fault tolerance through serial kinematic redundancy requires an inverse kinematics method insensitive to singularities and general with respect to geometry.

Table 7.1 A brief description of ten advanced robotic applications which would benefit from generalized inverse kinematics.

Application	**Description**
complex assembly	characterized by the need to operate in cluttered environments
food packaging	relatively simple tasks which must be performed at very high speed
micromanipulation	characterized by very precise operation in a small workspace
in-space assembly and construction	representative tasks include assembling and mating truss sections and solar arrays
satellite retrieval and repair	docking with the faulty satellite, repairing it, and then placing it back into orbit
E.V.A. site preparation	this decreases the amount of time the astronauts must spend outside of the spacecraft
semiconductor wafer handling	moving semiconductor wafers amongst process machines and storage in a clean-room environment
long reach arm	heavy-duty tasks in very large, yet complex work environments for waste-site cleanup and construction
inspection at nuclear facilities	inspection of drums, plumbing, tanks, and pumps for radiation, leaking, temperature increases, and wear in a radioactive environment
decontamination and dismantlement	stabilizing, disassembling, and storing decommissioned nuclear facilities

7.1.1 Complex Assembly

The need to operate in cluttered environments characterizes complex assembly tasks. Perhaps the robot must reach into a "tight" space or reach around an intermediate object. Installing wiring harnesses in an automobile is one example of a complex assembly task. This is not to say that other assembly tasks do not require precision. Complex assembly tasks are simply characterized by the need to operate in very constrained or obstacle-strewn environments.

Assembly tasks of any sort are by nature repetitious. This is what makes them good candidates for robot application. These repetitious tasks are boring for humans and may lead to repetitive motion injuries. Because of increased worker's compensation costs and increased health insurance costs, it is becoming more and more expensive to employ humans to perform these tasks. Also, a robot can typically replace three shifts of workers.

The obstacle avoidance and constrained environments which characterize complex assembly tasks essentially necessitate a redundant robot. The robot must avoid obstacles while still following the specified end-effector path. Also, the end-effector path will likely be quite constrained, at least at the position level. This will make it much more difficult to plan end-effector paths that avoid joint limits and singularities. A complex assembly task will thus require inverse kinematics based on multiple performance criteria as well as a robot with several redundant degrees of freedom. The extra degrees of freedom allow the robot to avoid obstacles and joint limits while still following the specified end-effector path. An inverse

kinematics algorithm insensitive to singularities is desirable when working with a redundant robot.

Representative assembly tasks include: force-fit part mating, controlled stressing of materials, joining parts by impact, and fastening. Performing these tasks in constrained environments adds complexity. These assembly tasks are in the very wide medium-duty range and involve forces of from 1 to 25 pounds. The global accuracy is in the range of from 1 millimeter to 100 micrometers and the required resolution is from .1 to 10 micrometers. Work volumes for these tasks are on the order of from 10 cubic centimeters to 1 cubic meter. This range of tasks recommends the use of a few modules to make a large population of assembly robots for this class of assembly tasks.

7.1.2 Food Packaging

The food industry represents a tremendous opportunity for applying robotic systems. In particular, packaging in the food industry has been identified as a promising application for robots. It is not difficult to imagine the typical packaging tasks in the food industry. One representative task involves putting stacks of wrapped candy bars into multiple bar packs. Another task is to put packages of cookies into their shipping cartons. The prominent feature of these tasks is that they are not very challenging kinematically, yet they must be performed at a relatively high rate of speed.

In the food industry, robots must compete with both hard-automation and humans for packaging applications. Versatility is the main benefit to using robots over hard-automation in these applications. For instance, if the size and shape of

the product changes, the robot can be reprogrammed while the hard-automation may need to be replaced or refitted. This versatility is becoming more and more important as marketing demands for packaging updates and regional variations in packaging increase.

The main benefit to using robots over humans in the food packaging application is associated with the repetitiveness of the tasks. Wrist and back problems are common. Also, these tasks are incredibly boring, involving thousands of packages a day. Finally, these tasks are relatively easy for robots to do.

Since packaging tasks in the food industry are not that challenging, it is reasonable to question the application of advanced robotic systems employing generalized inverse kinematics. The answer is modularity. These are ideal tasks for a modular robot. For instance, many of the tasks are so simple kinematically that a robot with two degrees of freedom will suffice. This robot will almost certainly be lighter and faster than a competing robot with more degrees of freedom performing the same task. A modular robot assembled from standard joint and link modules is cheaper as well. If the task changes significantly, the modules can be rearranged, or more modules can be added to the system. Fulfilling this vision of a truly modular and reconfigurable robot will require an inverse kinematics approach entirely general with respect to the robot's geometry. It will also require virtually all of the supporting technology described later in this chapter.

The food packaging tasks involve light loads, such as a 1 pound box of cookies. Typical forces on the end-effector are from 1 to 10 pounds. These tasks do not require extremely accurate operations. For instance, loading sacks of potato chips into packing boxes requires an accuracy of .5 centimeter. The range of global

accuracy requirements is from 1 to 10 millimeters and the range of required resolutions is from .1 to 1 millimeter. Handling individual pieces of small food products, for instance putting individual pieces of fine chocolates into one or two pound boxes, requires the most accuracy. Food packaging tasks are performed in work volumes of from .1 to 1 cubic meter. The main feature of interest is the very high speed of operation required by these food packaging tasks. Speeds of 1 to 10 cycles per second are required. These high speed requirements suggest that mechanical design will be of the utmost importance for robots involved in food packaging.

7.1.3 Micromanipulation

Micromanipulation tasks are characterized by the need for very precise operation within a very small workspace. Soldering wires from a semiconductor's silicon substrate to the pin-outs is one example of a micromanipulation task. Manipulating individual cells in the biological or animal sciences is another example. Microsurgery is also be a micromanipulation task. In these very precise applications, tolerances during the manufacture of the robot may be important. Manufacturing tolerances cause the actual geometry of the robot to deviate from the specified geometry. Robot metrology measures the actual geometry of the robot after it has been manufactured. The kinematic description of the robot is then modified to include the manufacturing errors. In essence, the robot has been calibrated.

The need for kinematic calibration of a manipulator in turn necessitates an inverse kinematics method general with respect to the robot's geometry. Current

manipulators are designed to have geometries with closed-form inverse kinematic solutions. These closed-form solution methods do not allow for deviations from the ideal geometry. Hence, a solution general with respect to geometry is required for calibrated robots.

Microsurgery is becoming increasingly common. Microsurgery is involved in opthamology, neurosurgery, otolaryngology, and cardiovasculary. The main benefits associated with using robotics in microsurgery involve increased precision during the operation. The change in motion scales is on the order of 100 to 1 and the change in force scales is on the order of 20 to 1. The forces will be very small on the order of from .01 to 1 ounce. This work must be very precise, with a resolution on the order of .1 to 1 micrometers. For instance, suture material for fine operations is about 5 micrometers. The accuracy required for radial keratotomy is 2.5 micrometers. Repairing a blood vessel of 100 micrometers in diameter requires 6 to 10 sutures of 10 micrometers in diameter. The work volume will be approximately 100 cubic centimeters. Clearly, manipulation on this small scale will require an extremely precise robot.

7.1.4 In-Space Assembly and Construction

Assembling the space station is one example of constructing a large space system. Typical tasks could include putting together truss sections and erecting solar arrays. Attaching habitation modules is another example of a typical task during the construction of large space systems. Mating payloads to booster rockets is also an example of in-space assembly. Smaller-scale tasks include screwing, bolting, clamping, winding, coiling, and locking. Safety considerations lead to

preferring robots over humans in space applications. It is extremely dangerous for humans to be in outer space, thus the more tasks performed by robots or other automation the better. Tasks involving large systems would be very difficult for one person to perform. Several astronauts would likely be required, thus magnifying the danger.

This range of tasks suggest the robotic systems will include small, medium, and large scale robots with capabilities ranging from low to very high. The global accuracy may range from 10 cm to 10 μm and the resolution may range from .1 cm to 1 μm. The work volume may range from about 1 cubic meter to 1000 cubic meters. The loads on the robots' end-effectors will vary from about 1 pound to 250 pounds.

The wide range of tasks expected for in-space assembly and construction requires the use of variety of robots each differing in scale. This recommends the use of modular robots that can be assembled from a finite number of joint and link modules. The use of modular robots will reduce costs, increase performance capabilities, and increase availability of the robot systems. The use of modular robots also requires an inverse kinematics algorithm general with respect to the robot's geometry.

Several other aspects of the construction of large space systems call for a very general inverse kinematics solution. Fault tolerance is very important in space applications and general inverse kinematics is essential for third-level fault tolerance. The complexity of the work environment for this application will likely require obstacle avoidance as well. The varied nature of the tasks may well recommend a robot that can be reconfigured to better perform the required tasks.

An inverse kinematics approach general with respect to geometry is required for reconfigurable robots.

7.1.5 Satellite Retrieval and Repair

Satellite retrieval and repair may be divided into three sub-tasks. The first is docking with the faulty satellite. Then the broken satellite needs to be repaired. This might be performed in orbit by astronauts or robots. The satellite could also be stowed for repair on the ground. The final sub task is placing the satellite back into orbit. This subtask will be much easier if the repair is performed in space.

The reasons for wanting to use robots instead of astronauts in the retrieval phase of the satellite repair operation are clear. This phase will require intensive extra-vehicular activity if humans must do all of it. Retrieving satellites using extra-vehicular activity is a very dangerous thing to do with astronauts.

Exactly when a satellite failure will occur, or what the satellite will be doing at the time, is unpredictable. This unpredictability may warrant a robot with several extra degrees of freedom to cope with obstacle avoidance or unforeseen requirements. Fault tolerance is clearly very important in the satellite retrieval application. A modular and reconfigurable robot would also have the advantage that it could be reconfigured to handle a wider range of retrieval operations. There is also another scenario in which a modular and reconfigurable robot could be of use in a satellite retrieval and repair mission. For the retrieval phase, a large number of modules, perhaps including ten or more joints, would be arranged in a serial chain to form a very long arm with a large workspace. This configuration is for the retrieval phase of the operation. For the repair phase. The ten modules could be

divided into two separate arms, perhaps of six and four degrees of freedom. Each of the arms would be mounted to the repair station. The four degree of freedom arm could be used to present the satellite to the six degree of freedom arm. This arrangement would maintain a good amount of dexterity while allowing more precise operation than would a strictly serial robot. Clearly, this type of retrieval and repair scenario is only possible if modularity and reconfigurability are approached from the highest level.

The satellite retrieval and repair operational requirements are similar to the in-space assembly and construction tasks except that after the retrieval phase, the work volume will be smaller. The global accuracy may range from 10 cm to 10 μm and the resolution may range from .1 cm to 1 μm. The work volume may range from about 1 cubic meter to 10 cubic meters. The loads on the robots' end-effectors will vary from about 1 pound to 250 pounds.

7.1.6 EVA Site Preparation

EVA is an acronym for extra vehicular activity; essentially, an astronaut leaves the space craft. Preparing the site of the task is desirable to minimize the amount of time that the astronaut will spend outside of the space craft. Common EVA site preparation tasks involve unstowing tools and bringing them to the work site. A robot could unload the toolbox, bring it to the work site, and then open it. Installing foot restraints and power fixtures are other EVA site preparation tasks. These implements must also be replaced in storage after the EVA task is complete.

The EVA site preparation task is a natural function for robots to perform. The large workspace and variations in the tasks recommend a serial robot. Also,

performing the tasks will likely only place modest demands on the manipulator. This should allow the robots to operate autonomously, or under ground control, to save the astronauts intra-vehicular time as well.

As with any application in space, fault tolerance is an important issue in EVA site preparation. Serial kinematic redundancy is one way of achieving fault tolerance. An inverse kinematics approach general with respect to the robot's geometry is required to achieve this level of fault tolerance. Also, some obstacle avoidance capabilities would likely be desirable for EVA site preparation. Finally, a robot capable of very dexterous motion may relax some design constraints on systems the robot must interact with. Meeting these dexterity and obstacle avoidance goals may require a robot with several redundant degrees of freedom.

The operational requirements on the EVA site preparation task are very modest. The precision will be on the order of 1 centimeter and the resolution does not have to be significantly better than that. The loads will be very low, on the order of from 1 to 5 pounds. The workspace will be large, about 100 cubic meters.

7.1.7 Semiconductor Wafer Handling

Numerous separate operations are required during the semiconductor manufacturing process. Between almost every operation the wafers must be moved among process machines. Complex integrated circuits, such as microprocessors, currently require about one hundred separate operations as the circuit is formed on the silicon substrate. This represents a significant amount of handling operations as the wafers are being processed. Logistics typically dictate that the wafers must also spend some time in storage as they are being processed. The movement into and

out of storage increases the handling load. The bulk of the wafer processing takes place in a clean room environment. The purity of this environment is very important. To reduce contamination of the clean room, the humans who are currently performing these handling operations must always wear special clothing.

The clean room environment is the key factor recommending robots rather than humans for these handling operations. Because the robots do not have to leave the clean room, they do not introduce particulate matter carried in from outside. Also, the robots can operate in a near-vacuum environment, which has been suggested as a method of reducing particulate contamination. Another factor to consider is the effect working in a clean room environment has on a human. The special clothing that must be worn is restricting and uncomfortable. Masks must be worn to cover the face. The handling task itself is very repetitive. All these factors recommend using robots for semiconductor handling tasks.

The semiconductor wafer handling application imposes fault tolerance requirements on a robotic system. This is because each robot must fulfill the handling function for several process machines. While the robot is inoperable, the capabilities of several process machines will be lost. This could interrupt the production of several hundred wafers. Fault tolerant robots, which can continue to perform their tasks even under fault conditions, are required so that the manufacturing process will not be interrupted until servicing is arranged.

The actual semiconductor manufacturing processes (masking, lithography, CVD, PVD, etc.) require extremely high tolerances. These submicron tolerances are far beyond the capabilities of any serial robot arm. Transferring wafers between specialized process machines is the main application for robots in the semiconductor

manufacturing process. This transfer process requires a precision of .1 to .01 centimeters for loading the wafers into susceptor ports in the machines. There is essentially no difference in the requirements for precision and resolution. Each wafer may require 100 or more separate processes during its manufacture. To accommodate this many process machines, the total workspace is large, on the order of 1000 cubic meters. Though each wafer is very light, the robot may be asked to carry a cassette of multiple wafers which weighs on the order of 10 pounds.

7.1.8 Long Reach Arm

This long reach arm concept is in response to the DOE plan to develop key technologies for efficient, timely, and cost-effective clean-up of existing nuclear waste sites and the decommissioning of various nuclear fuel facilities. This long reach arm concept is part of a larger national effort structured at DOE headquarters through ORNL and Sandia National Lab to create a special class of robot systems to work in the hazardous environment of nuclear waste sites with minimum human intervention to bring down the level of ORE that would be experienced by direct operator involvement. This concept effort expands on technology base activities at The University of Texas (approaching $1,800,000/year of funding by DOE and other federal and state agencies) in order to produce prototype technology which would otherwise not be achievable because of limits in resources. Specifically, strong partnerships with industrial development groups (Lockheed Austin Division), government labs (ORNL, Sandia) and other universities (Michigan, Tennessee, Florida) are being integrated in this team to respond to tasks associated with

inspection, sampling, transport, etc. in clean-up tasks associated with restricted spaces in nuclear waste tanks, low level waste barrel storage, etc.

The prototype long reach arm will be based on a modular structure of very lightweight (but stiff) tetrahedra some of which will be actuated by 7 specially designed motor spindle drives which are unusually lightweight, stiff, compact, and nearly frictionless. There will be 3 rotary actuators used (based on an existing fault tolerant prototype) at the base, at the wrist, and at the end-effector. The design will be highly modular such that the failure of any one actuator will not prevent continued operation (until the system can be taken off-line for maintenance) and it can be reconfigured (by interchanging links, actuators, tools, etc.) and up-dated without major changes to the operating systems (SFW and HDW). This level of technology will dramatically accelerate the implementation of long reach arm technology ultimately reducing its costs, making modular repairs feasible, and reducing the threat of obsolescence by allowing rapid "tech mods" to the system.

The actuator that drives the joints is the fundamental component of a robot and the long reach arm will put special demands on the actuator. The actuators in this robot will be loaded with extremely large forces primarily in tension. Because of these very large forces, the actuators for this robot must be almost frictionless. Also, the length of this robot will produce tremendous amplification of small errors at the actuators to very large errors at the end-effector. This amplification demands the use of a very precise actuator. To meet these demands, the University of Texas will build on its experience with advanced robot actuators to specialize a revolutionary design initially developed at the German Aerospace Establishment (DLR), with which the University of Texas Robotics Research group has a long

standing and very good relationship. This design uses screw planets and a spindle to provide an extremely compact, lightweight, and powerful linear actuator. The resulting actuator will be precise, fault-tolerant, and capable of developing the extremely large forces required by the long reach arm. Some specific characteristics include: high reduction (600:1), very fine thread pitch (0.2 mm/revolution), and 96% improvement in efficiency over standard spindle-nut. The long reach arm will work in a volume of about 10000 cubic meters. The arm will have capabilities on the order of a course industrial robot. The loads will be from 1 to 250 pounds. The required global accuracy is about 10 cm and the resolution 1 cm.

7.1.9 Inspection at Nuclear Facilities

Inspection operations are very common at modern nuclear facilities. Drums, plumbing, tanks, valves, and pumps all must be routinely inspected. The sensing part of the inspection process may be fairly complex and involve checking for many different signs of problems. Radiation tests will clearly be common. Certain equipment will be checked for cracks or other signs of excess stresses. Stored waste may need to be checked for excess heat generation or gas leakage. The actual path that the sensors must follow during the inspection process is quite complex. Many of the sensors need to be moved directly over the entire exterior surface of the equipment being tested.

The dangers associated with human exposure to radiation is the main reason robots are desirable in nuclear environments. There are federally mandated limits on the amount of radiation to which a worker may be exposed. As the inspection tasks are routine and must be performed regularly, the total exposure of workers to

radiation continues to accumulate over time. Some of the inspections, especially those of the waste, will have to be repeated for a long time into the future. The cumulative exposure factor also recommends robots for inspection tasks at nuclear facilities.

There are several reasons only a very advanced robot will be capable of performing many of the inspection tasks at nuclear facilities. Just following the required sensor path requires a very dexterous arm. The complex and cluttered work environment necessitates advanced obstacle avoidance capabilities. Reaching through port holes or passageways may also be required as part of some inspection tasks. Fault tolerance issues will be important in almost all applications of robots in nuclear facilities. Finally, the inspection tasks will likely be so varied that no one robot geometry will suffice. A spectrum of robots with varied scale, yet based on a standard set of joint and link modules, could be assembled to perform this wide range of inspection tasks. This wide range of demanding requirements will only be satisfied by a very advanced system of robots, of which a general inverse kinematics approach will certainly be a part. The inspection process requires the robot to carry a suite of sensors. The combined weight of these sensors will represent a load of from 10 to 25 pounds. The inspection process is low precision, requiring a global accuracy of 1 cm and a resolution of .1 cm. The work volume may be as large as the volume of the plant, which is approximately 100,000 cubic meters.

7.1.10 Decontamination and Dismantlement

Decontamination and Dismantlement at nuclear facilities is part of the environmental restoration and waste minimization (ER&WM) process. The

physical tasks associated with ER&WM will range from simple inspection and transport to complex disassembly, cutting, welding, rigging, packaging, repair of failed equipment, etc. Some of the tasks will be underground (piping), in large volumes (tanks), in cluttered environments (reactors), etc., and will require a diverse class of manipulator systems. Some of these systems can be quite simple (3-DOF planar devices) to multiple arms of extra DOF (8 to 10 DOF) capable of precision tasks under varying forces in a cluttered environment, all operating remotely sometimes with human intervention through a manual controller or through supervisory software taught to perform repetitive tasks by human input. The work environment for the slave may contain high levels of moisture, corrosive chemicals, radiation, dust, etc. Hence, decontamination, maintenance, replacement of failed parts, etc., must be planned as part of the manipulator technology in its foundation architecture.

The ER&WM tasks addressed here involve the deployment of robots for the selective removal of equipment. This part of ER&WM presents many challenges due to the complex environment and demanding tasks. ORNL developed the robotic Dual Arm Work Module as a vehicle for developing the technology necessary for robots to perform this task. The DAWM has two arms, each with multiple redundant degrees of freedom. This robot has the ability to perform a task in a variety of ways while simultaneously obeying hardware and environmental constraints. Due to the complexity of the application there are a variety of important criteria that impact the robot's ability to perform its task. This list of potentially important criteria includes: joint travel and torque limits, load-carrying capabilities, precision operation, obstacle avoidance, and dexterity issues. These

multiple criteria provide a more balanced basis for decision making that ultimately improves computational reliability and task performance capabilities.

The most challenging task for the robot in the selective removal of equipment is dismantling the equipment. This dismantling may involve forces as high as 250 pounds. The task is moderately precise, requiring a global accuracy of 1 mm and a resolution of .1 mm. Though the work may be performed throughout the nuclear facility, the work volume for each dismantling task will be approximately 10 cubic meters.

Table 7.2 This table presents a summary of the requirements for the ten advanced applications.

Application	**Load (lbf)**	**Precision (m)**	**Resolution (m)**	**Work Space (m^3)**
complex assembly	1 - 25	$1 - .1 * 10^{-3}$	$.1 - .01 * 10^{-3}$	.1 - 1
food packaging	1 - 10	.01 - .001	$1 - .1 * 10^{-3}$	.1 - 1
micromanipulation	.001 - .1	$1 - .1 * 10^{-6}$	$.1 - .01 * 10^{-6}$	$1 * 10^{-4}$
in-space assembly and construction	1 - 250	.1 - .0001	1 - .00001	1 - 1000
satellite retrieval and repair	1 - 250	.1 - .0001	1 - .00001	1 - 100
E.V.A. site preparation	1 - 5	.01	.01	100
semiconductor wafer handling	1 - 10	$1 - .1 * 10^{-3}$	$1 - .1 * 10^{-3}$	1000
long reach arm	1 - 250	.1	.01	10000
inspection at nuclear facilities	10 - 25	.01	.001	100000
decontamination and dismantlement	1 - 250	$1 * 10^{-3}$	$.1 * 10^{-3}$	10

7.2 SUPPORTING TECHNOLOGIES

Table 7.3 A brief description of ten supporting technologies required for the application of generalized inverse kinematics.

Technology	Description
computer hardware	the computing machinery on which the inverse kinematics algorithms will be developed and executed
computer architecture	the structure of the computer, both hardware and software
computer software	the language used to communicate the inverse kinematics algorithms to the computer
performance criteria development	continued development of performance criteria primarily in the area of physical interpretation
sensors	sensors extract information about the robot and its environment
data communication	information exchange amongst the sensors, the controller, and the actuators
control	the software and hardware which must operate and interface with the actual robot
mechanical architecture	the structure and granularity of the robot, primarily issues of modularity
mechanical design	the conception and detail of the robot's physical structure
robot actuators	structural units supplying the mechanical power at the robot's joints

The application of generalized inverse kinematics depends on the support of these ten technologies, which fall into four basic groups. The computer technologies include: the hardware, software, and architecture required for the actual execution of the inverse kinematics algorithms. The electronic technologies include the robot's sensors, controllers, and communications systems. The mechanical technologies include the robot's mechanical design, actuators, and

construction. Intellectual development is the final technology group. This group includes computer algorithms and performance criteria. This discussion of the technologies includes: the performance required for the application of generalized inverse kinematics, the performance available, and the changes in performance levels over the past ten years. The discussion also includes features of the technologies particularly important to their support of generalized inverse kinematics.

7.2.1 Computer Hardware

Any algorithm for solving the generalized inverse kinematics problem will ultimately execute on some sort of computer hardware. Today's most common architecture for computer hardware consists of three major subsystems. These three major subsystems are the central processing unit, the long-term storage device, and the input/output pathways. The "brains" of the computer is the central processing unit, which is commonly called the CPU. This discussion considers RAM (short-term memory used by the CPU) part of the CPU. A long-term storage device, such as a hard disk, is required to save programs and data for future use. The final subsystem performs what is called input and output. Input and output, or I/O, is where information and instructions are exchanged with the computer. A user will typically instruct the computer through the keyboard, a pointing device, or voice input. The computer responds through a video display or audio feedback. The visual display of information through computer graphics is the major I/O function this section describes. Each of these three subsystems will be discussed in terms of the performance required by a generalized inverse kinematics implementation, the

performance which is currently available, and the changes in performance levels which have occurred over the last several years.

Choosing some reasonable numbers for the system's parameters allows the calculation of the CPU performance required for the real-time application of generalized inverse kinematics. Thirty Hertz is commonly cited as the required frequency for solving the inverse kinematics problem in real-time (Hernandez and Tesar, 1989), and thus will be used as the definition of "real-time" for this discussion. A geometry with seven degrees of freedom will be used in these calculations for two reasons. The first is that a geometry with seven degrees of freedom is a common example found in the literature. The second is that criteria calculation performance benchmarks are available for the Robotics Research Corporation's kb1207 manipulator, which has seven degrees of freedom. The following calculations are made for a direct search inverse kinematics algorithm general with respect to the robot's geometry. Geometric simplification, such as that provided by a spherical wrist or shoulder, would significantly reduce the demands on the CPU. The inverse kinematics algorithm used in this example incorporates a factorial perturbation pattern and steepest descent decision making.

The direct search method of solving the generalized inverse kinematics problem may be broken into two major parts, the local exploration and the decision making. Generating the joint perturbations during the exploration is simply a short series of addition and subtraction. The decision making, in contrast, involves calculating the performance criteria values for each of the trial solutions and represents far more computational overhead. The computational demand depends upon which performance criteria the solution incorporates. Van Doren and Tesar

(1992) present a set of benchmarks that separate the criteria into several categories. The benchmarks are for the Robotics Research kb1207 robot and the Silicon Graphics 25TG personal workstation. The amount of time it took for one calculation of an entire criteria category is presented. For instance, the set of Jacobian-based criteria took .00822 seconds to calculate. The set of Hessian-based criteria took .01614 seconds to calculate and the entire suite of twenty-nine criteria took .1594 seconds to calculate (Van Doren and Tesar, 1992). The 25TG personal workstation is rated at 1.6 MFLOPS, so these timing figures may be written as seconds per MFLOP as shown in Table 7.3.

Table 7.4 The amount of time per MFLOP for each calculation of the values associated with three criteria sets

Criteria Set	**Benchmarks**
Jacobian-based	.00514 seconds/MFLOP
Hessian-based	.01009 seconds/MFLOP
29 criteria	.09963 seconds/MFLOP

For a robot with seven degrees of freedom, the factorial perturbation pattern generates 128 trial solutions. Each of these trial solutions must be addressed at least thirty times a second for real-time performance. The demand, in MFLOPS, on the CPU may now be calculated. For Jacobian-based criteria only, the required CPU performance is

$$\text{MFLOPS} = (128*30)\frac{\text{calculations}}{\text{second}} * .00514\frac{\text{seconds}}{\text{calculation}} = 19.7$$

For Hessian-based criteria only, the required CPU performance is

$$\text{MFLOPS} = (128*30)\frac{\text{calculations}}{\text{second}} * .01009 \frac{\text{seconds}}{\text{calculation}} = 38.7$$

For all twenty-nine criteria, the required CPU performance is

$$\text{MFLOPS} = (128*30)\frac{\text{calculations}}{\text{second}} * .09963 \frac{\text{seconds}}{\text{calculation}} = 382.6$$

These calculations assumed that all of the criteria must be calculated at each point along the path. Thus, in a sense these numbers represent a path planning argument. Nonetheless, these calculations do give an impression of the magnitude of CPU performance required to address the generalized inverse kinematics problem in real-time. In a larger sense, this type of calculation is useful for determining the performance requirements for addressing any multiple criteria decision making problem. For instance, if one-hundred choices must be faced at a rate of thirty Hertz and each choice takes .1 seconds per MFLOP to evaluate, then the required CPU performance is

$$\text{MFLOPS} = (100*30)\frac{\text{evaluations}}{\text{second}} * .1 \frac{\text{seconds}}{\text{evaluation}} = 300$$

The CPU performance available on the market today may be gleaned from available manufacturer's data. The MIPS corporation's 100 MHz R4400 CPU is rated at 22 MFLOPS (Gorey, 1992). The DEC corporation's 200 MHz Alpha CPU is rated at 200 MFLOPS and the Intel corporation's 50MHz i860 CPU is rated at 100 MFLOPS (Comerford, 1992). There are many other CPUs available from a variety of manufacturers, but these three give an impression of the range of CPU performance typically available to today's engineer. Each of these CPUs is found in computers costing from $5,000.00 to $100,000.00. The main difference in the cost

is not due to the CPU, but rather due to other computer hardware, primarily for graphics display.

Comparing the available CPU performance with the performance required for solving the generalized inverse kinematics problem shows that today's CPU's are capable of incorporating some criteria into real-time solutions.

Though incorporating a full-complement of performance criteria in a general solution will take more CPU performance than is currently available, the recent pace of CPU advancements is truly amazing. For example, in May of 1990, the top-of-the-line MIPS CPU was the 25MHz R2000 which was rated at 3.3 MFLOPS (Gorey, 1990). This CPU was only available in computers which cost at least $50,000.00. Today, the 100MHz R4400 is rated at 22 MFLOPS and is available in computers costing as little as $5000.00. The price-to-performance improvement in only three years is almost one hundred times! Even if this rate of improvement decreases dramatically, and there are no indications that it will, incorporating a full-complement of performance criteria into the real-time solution of a multicriteria decision making problem should be possible within the next few years.

An alternative to repeatedly calculating performance criteria values for each trial solution is to calculate the values once, over the entire workspace, and then save the values in a permanent storage device. The main concern here is the amount of data that will need to be stored. The amount of data is dependent upon the number of degrees of freedom, the range of motion at each joint, the resolution over the range, and the precision with which each of the data points is represented. The total amount of storage space required is then

$$\text{storage space} = \left(\text{range} * \frac{\text{precision}}{\text{resolution}} \right)^{\text{DOF}} \qquad (7.1)$$

Substituting reasonable numbers gives an impression of the storage space requirements. For instance, a robot with seven rotational joints each having a range of 180 degrees, is representative of current research robots. A precision of four bytes allows each data point to be represented to seven decimal digits (assuming the common digital representation of one sign bit, seven exponent bits, and a twenty-four bit mantissa). With these choices (DOF = 7, range = 180 degrees, and precision = 4 bytes) the storage space requirements for three different resolutions will be calculated. For a resolution of five degrees, the storage space required is

$$\text{storage space} = \left(180 * \frac{4}{5} \right)^7 = 1.28 * 10^{15} \text{ bytes}$$

For a resolution of one degrees the storage space required is

$$\text{storage space} = \left(180 * \frac{4}{1} \right)^7 = 1.00 * 10^{20} \text{ bytes}$$

For a resolution of .001 degrees the storage space required is

$$\text{storage space} = \left(180 * \frac{4}{.001} \right)^7 = 1.00 * 10^{41} \text{ bytes}$$

Of these examples, only the resolution of .001 degrees is sufficient for directly controlling a robot. The other two resolutions, five and one degrees, would require interpolation.

Unfortunately, these storage space requirements are far beyond what is available with the current state-of-the-art. Recently, The IBM corporation announced that they had successfully demonstrated optical storage densities of 387

million bits per square centimeter (Zorpette, 1993). This density will allow approximately 6.5 gigabytes of information to be stored on one 5.25 inch disc. This is still many orders of magnitude less than what is required to store criteria values generated over a complete workspace, even at very coarse resolutions.

It is, however, interesting to note the recent advancements in data storage techniques. During the mid-1980's, the standard storage medium was a 5.25 inch magnetic disc that could store about 500 kilobytes of data. Assuming that the 6.5 gigabytes of storage on a 5.25 inch optical disc becomes available in the mid-1990's, the storage capacity will have increased by four orders of magnitude in just ten years. The ability to store the amount of data represented by Equation (7.1), even at course resolutions, in the near future is unlikely.

Computer graphics is an essential part of a complete generalized inverse kinematics system. As an interface, computer graphics queries for information and provides visual feedback. The most demanding form of computer graphics is computer animation. For a smooth animated display, specialized graphics hardware is currently required. The capabilities of the graphics hardware dictate the amount of information that can be displayed.

A graphical user-interface is an efficient and intuitive means of querying for information. Even the most basic personal computer is capable of supporting a graphical user interface. One method of providing visual feedback shows an animation of the robot while simultaneously displaying criteria values in the form of a bar graph, line graph, or numerical display. This type of display requires graphics performance of approximately $30*10^3$ polygons/second (Hooper and Tesar, 1990). Another type of display shows parameter values by varying the shading along the

surface of the animated robot. For instance, stress levels along the robot can be shown with artificial coloring of the animated robot's surface. This type of display might also show how kinetic or compliance energy is circulating throughout the system. For an animated robot, parameter display by surface shading is estimated to require graphics performance of $30*10^6$ polygons/second (Hooper and Tesar, 1990).

Computers built by the Silicon Graphics Corporation provide a benchmark for available graphics hardware. The Silicon Graphics company is the currently acknowledged leader in computer graphics hardware. The highest performance graphics hardware available from Silicon Graphics is the Reality Engine, which is capable of $1.1*10^6$ polygons/second (Gorey, 1992). This graphics performance is easily capable of displaying an animated robot and parameter values via a graphical user interface. More advanced display of criteria values through surface shading requires further hardware development.

Unfortunately, the state of the art in graphics performance, at least as measured by polygons per second, has recently been stagnant. For instance, in May of 1990, $1.0*10^6$ polygons per second was available from Silicon Graphics (Gorey, 1990). Today, the performance has only increased to $1.1*10^6$ polygons/second (Gorey, 1992). Even the cost of these machines has not changed much in the last several years. In 1990, the minimum cost for $1.0*10^6$ polygons/second was $94,900.00 (Gorey, 1990) Today, the minimum cost for $1.1*10^6$ polygons per second is $42,900.00 (Gorey, 1992). It may be several years before the real-time display of criteria values by surface shading, estimated to require $30*10^6$ polygons/second, is possible.

7.2.2 Computer Architecture

Any algorithm for solving the generalized inverse kinematics problem will ultimately execute on a computer. This section addresses the question of whether there is a specific computer architecture especially suited for solving the generalized inverse kinematics problem in the context of multicriteria decision making.

Computer architecture includes such things as: the layout of the bus, digital logic, multiple processing units, and system input/output. Computer architecture includes both hardware and software issues. The computer scientist is interested in all of these subjects. The subject of multiple processing units has the greatest impact on the speed of solving the multicriteria decision making problem. Typically, multiple processing units are arranged in either serial or parallel architectures. The serial architecture is often called a pipeline. A computer pipeline is roughly analogous to an automobile assembly line. Many automobiles are assembled at the same time, but different tasks are performed at each workstation. The automobile is finished when it comes off the end of the of the assembly line. A computer pipeline is suited for problems where many different operations must be performed on each data point; with each operation depending upon the result from the previous operation. Computer pipelines are often used as graphics subsystems because displaying each pixel requires several sequential operations, such as: scaling, clipping, and shading. On the other hand, a parallel architecture is suited for problems that may be decomposed into a set of smaller subproblems that do not depend on results from each other. The parallel architecture is best for multicriteria decision making problems.

The parallelizable percentage of the problem is one indicator of the improvement expected from a parallel architecture. The percentage is measured relative to the amount of computation time spent on each part of the problem. For instance, if parallelization makes half the problem twice as fast, the overall improvement in speed is twenty-five percent. The performance increase as a function of the number of parallel processors is the other major indicator of improvement possibility. In the ideal case, there is a one-to-one correspondence. Two processors solve the problem twice as fast and three processors solve the problem three times as fast, etc. Judging by both of these indicators, the multiple performance criteria decision making problem is ideally suited to a parallel computer architecture. This is because the decision making strategies spend the vast majority of computation time calculating criteria values for each of the choices. The calculations in Section 7.2.1 assume that the algorithms spend all of their time calculating criteria values.

Both of the above observations bode very well for using parallel computers to solve multicriteria decision making problems. The first result is important because parallel computers with several processors are beginning to become available to the average engineer. For example, the Silicon Graphics company recently introduced the Onyx line of parallel computers that have from two to twenty-four parallel processors. The Onyx computers are engineering workstations with state-of-the-art microprocessors. These computers cost from $100,000 to $750,000. The second result is important because massively parallel computer architectures are currently dominating the supercomputing industry. Both the Cray corporation and Thinking Machines Inc. are currently developing parallel computers

with hundreds of microprocessors. A multicriteria decision making problem which must face a very large number of choices seems ideally suited to this type of computer architecture. Though these massively parallel computers are currently only found in the supercomputer domain, the recent trend has been that supercomputer performance comes to an engineer's desktop in only a few years. For instance, the Cray X/MP with one vector processor has a peak performance of 240 MFLOPS. This was state-of-the-art in the 1980's. Today, a single ALPHA CPU has virtually this same peak performance, 200 MFLOPS. As has been discussed, the alpha CPU is found in engineering workstations costing less than $10,000.00.

7.2.3 Computer Software

The inverse kinematics algorithms and multicriteria decision making strategies are coded into computer software for simulation and application. Developing software is by nature time-consuming. Thus it is important to get the most benefit from any software development effort. Certain measures correspond to the relative benefit expected from a given software development effort. These measures include the software's generality, portability, and efficiency.

The ability to expand and upgrade the software to include new performance criteria and new decision making strategies measures the generality of the software. In the sense that developing software is both time-consuming and expensive, software development may be compared to robot development. A modular robot architecture has been proposed as a method of decreasing the time and expense associated with developing a robot. A modular approach to software development

has similar benefits. Essentially, the modular approach allows the incorporation of new software modules without redesigning the entire software package. Keeping the software modules separate instills a degree of reconfigurability. For instance, one module might calculate the Jacobian based criteria values while another module could calculate the factorial perturbation pattern. Different criteria modules could be arranged with different decision making modules to best configure the software given a specific task requirement. The software can incorporate new performance criteria and decision making strategies as they are developed. Clearly, the interfaces to the software modules are of the utmost importance. The inputs and outputs of each module must be carefully designed and documented. Reduced obsolescence of the software package is the reward for this effort. Maintaining a separation of various software packages also increases generality. The various software packages can communicate with one another through data files, shared memory, or by some other means. Several examples of the benefits associated with maintaining a separation of software packages exist in the University of Texas Robotics Research group. The robo_cad software, for instance, was originally designed as an interactive configuration environment for modular robots. Because the software incorporates the ability to communicate through datafiles and shared memory, it has since been used for real-time animation, design optimization, obstacle avoidance research, path-planning, and inverse kinematics development. Another example is the NDOF dynamic modeling software. Since this software is general, with the specific robot and path characteristics communicated through data files, it remains useful after almost ten years.

Software is portable if it runs on different computing platforms without modification. Due to the rapid pace of computer hardware improvements, software portability is currently of the utmost importance. Portable software takes advantage of improvements in hardware automatically. Portable software may sacrifice some execution speed at first, but before long, portable software will likely run much faster than any software bound to an obsolete computing platform. Of the current computer languages, "C" is probably the most portable. There are "C" compilers for personal computers, specialized controllers, engineering workstations and supercomputers. It is also possible to take advantage of specialized computer hardware while still maintaining portable software. The Code 2.0 graphical programming language allows software to take advantage of parallel architectures while still maintaining portability.

The efficiency of the software relates to execution speed without sacrificing generality or portability. Efficiency may be loosely divided into algorithmic efficiency and execution efficiency. Algorithmic efficiency is associated with the formulation and coding of the algorithm itself. For instance direct search can achieve some performance increase by not calculating criteria values for trial solutions that have already been explored. Also, many different performance criteria depend on the singular value decomposition of a matrix. Significant performance improvement is achieved if the singular value decomposition is performed once and the results distributed to the various criteria calculation modules. It is probably unwise to seek these performance improvements if they come at the expense of generality or portability. Careful formulation of the computer code and the underlying numerical representation often improves

execution efficiency. Integer performance, for instance, is often faster than floating point performance. Formulating an algorithm using integers, when possible, is a way to increase performance without sacrificing generality or portability.

As another example of improving computational speed without sacrificing portability, consider Table 7.5. This table shows that the rapidly changing criteria need to be sampled the most frequently, but slower changing criteria need to be sampled less frequently. The order of the criteria directly influences the speed at which the criteria are changing. The higher order criteria change more rapidly and must be sampled more frequently. This must be balanced against the relative importance of the criteria to the application as determined by the robot's operator. Taking into account the order of the criteria to determine the sampling rate will improve computational speed without sacrificing portability.

Table 7.5 The relative sampling frequency for the performance criteria as determined by their order.

Criteria Basis	**Sampling Rate**	**Order of Criteria**
position	low	zero order
g	modest	first order
h	medium	second order
kinetic energy	medium	second order
gh product	high	third order

7.2.4 Performance Criteria Development

Clearly, performance criteria are a fundamental part of generalized inverse kinematics. The University of Texas Robotics Research Group has made a

considerable investment in criteria development. The results of this effort are over thirty geometric, inertial, kinetic, and compliance criteria. There are two avenues for continued performance criteria development. The first is to generate more criteria and the second is to improve the criteria that have already been identified. Both of these avenues are worthy of exploration. Of these, the most important is developing multiple physical, yet independent, meanings. As for generating more criteria, the largest void is the lack of criteria associated with obstacle avoidance. This is not for lack of trying. Two Masters theses within the University of Texas Robotics Research Group have been dedicated wholly to obstacle avoidance and two more are in progress. Currently, the major results from this work is that the obstacle avoidance issue is very difficult to formulate mathematically and very difficult to solve numerically. This has led to the realization that progress on the obstacle avoidance issue can only be made by building on a solid foundation of fundamental principals. As far as improving the existing criteria, a physical interpretation of the criteria's meaning is still missing.

The importance of obstacle avoidance criteria to the application of advanced robotics cannot be overemphasized. Most of the applications discussed in this chapter require reaching around objects or through small passageways. These applications require obstacle avoidance by nature. The use of hyper-redundant robots to achieve fault-tolerance also requires obstacle avoidance. This is because the portion of the robot between the base and the end-effector is relatively long and could easily experience collisions even though the end-effector follows a collision-free path. Controlling both the end-effector of the robot as well as the intermediate joints and links represents a tremendous demand during teleoperation. The human

should control the end-effector while collisions involving the intermediate portion of the robot are avoided automatically. If the costs of the robot's actuators and their controllers decrease, then the benefit/cost factor will increase and redundant robots will become much more realistic for applications. Further advances should decrease the weight and volume of the actuators as well. If the robot is slimmer and lighter, then obstacle avoidance becomes much more feasible.

Assessing the need for obstacle avoidance criteria is not nearly as difficult as formulating a course of action for developing useful criteria. Developing useful obstacle avoidance criteria will involve both analytic and abstract reasoning. Analysis is required to express the obstacle avoidance criteria in terms of fundamental properties. First, second, and even higher order geometric properties will likely be required. Incremental improvements towards the realization of real-time obstacle avoidance can only be achieved by building on a sound foundation of fundamental principles. Developing obstacle avoidance criteria also requires a degree of abstract reasoning. For instance, when should an obstacle in the workspace even influence the decisions? If there is no collision, the obstacle does not create a problem. How close should the manipulator be allowed to come to the obstacle? Current efforts address these questions by simply operating the robot at very low speeds and decreased work volumes (NASA, 1992). This approach limits the robot's performance to extremely modest tasks. In some cases, it might be desirable that the manipulator even push against an obstacle. The issue of reaching through an access port is also important. For instance, if the manipulator is not allowed close to an obstacle, a wall with an access port in it may look the same as a

wall with no port. These are just a few of the issues that must be addressed on an abstract as well as an analytic level.

Incorporating performance criteria will only result in enhanced performance if one or more physical interpretations are attached to each of the criteria. These interpretations are the main area yet to be addressed in performance criteria development. A physical interpretation allows the operator to intelligently choose and rank the performance criteria for a given application. For instance, kinetic energy partition values presented in purely geometric terms have several important physical interpretations that can guide the operator. Rapidly changing kinetic energy partition values indicate rapid energy movement from one part of the system to another. This destructive movement implies: large internal forces, high torque demands, high force-induced deflections, and poor system responsiveness. Table 7.6 lists some more important physical interpretations that may be associated with the performance criteria.

Table 7.6 Physical interpretations that may be associated with the criteria.
Deflection Energy Partition Values
Backlash, Hysteresis, and Frictional Distribution
Sensitivity to Temperature Variations
Internal Force Magnification
Criteria Affecting Minimum Step
Criteria Affecting Resolution
Criteria Affecting Accuracy
Criteria Affecting Control (Saturation, Overshoot, Oscillations, etc.)

Achieving a valid physical interpretation of the performance criteria requires a focused effort. One result of this effort will be the use of lessons-learned from experts as part of a supporting expert system. Ultimately, the physical interpretation requires experimental validation. This experimental validation involves an extensive instrumentation and metrology effort.

7.2.5 Sensors

Accurate sensor feedback is crucial for the application of advanced redundant robots. Sensors extract information about physical realities and typically use a transduction process. For instance, a sensor may generate a voltage level corresponding to a displacement. The voltage is interpreted by a computer. Because sensors provide information about the physical environment, sensor inaccuracies may result in an incorrect response. This section further discusses the fundamental importance of sensors to the application of generalized inverse kinematics. Some basics concerning sensor design, sensor placement, and the interpretation of sensor data are also discussed.

Integrating advanced sensor technology is very much intertwined with applying generalized inverse kinematics. Thus far, the inverse kinematics problem has been discussed in an open loop context. That is, a given set of joint displacements results in a known displacement of the end-effector. Unfortunately, this is not the case in an actual application. Errors due to manufacturing tolerances, temperature variations, and unmodeled phenomena (such as backlash) cause a deviation between the predicted and the actual response of the robot. Since errors accumulate in a serial structure, a redundant robot tends to accumulate more errors.

A hyper-redundant robot accumulates even more errors. Sensors will identify these deviations from the predicted response. A course of action may then nullify the errors. The generality of the inverse kinematics algorithms developed in this dissertation allows the incorporation of a wide variety of sensor data. For instance, temperature sensors along the links of the robot can identify changes in the geometry of the robot due to variations in temperature. The sensors may then communicate these changes to the inverse kinematics algorithm that will incorporate this information while finding the required set of joint displacements.

There are many factors to consider when choosing a sensor. Resolution, range, precision, stability, and noise immunity are common measures of a sensor's performance. The resolution is the smallest reliably detected change in input. The range is the difference between the maximum and minimum input states. The precision is given by the equation

$$\text{resolution} * \text{range} = \text{precision} \tag{7.2}$$

Equation 7.2 can be used to choose a sensor when the resolution and range are determined by the application. For instance the range for one full revolution is 360°. Assuming the application requires positioning the joint to within .001 degrees, the required precision is

$$1000 \frac{\text{counts}}{\text{degree}} * 360 \text{ degrees} = 360{,}000 \text{ counts} \tag{7.3}$$

Thus, the required precision is 360,000 decimal digits or 19 binary bits. Though in actual operation the precision at the robot's end-effector is the important concern, calculating the precision at the end-effector relating to the precision at a joint is specific to both the robot and the robot's position within its workspace. The sensor

must be stable and not drift during normal operation. Finally, the sensor should be immune to noise interference. This is because a system is just as sensitive to noise coupled through the sensor as it is to the controlling input variable. Figure 7.1 shows the block diagram of a simple control loop with noise in the feedback.

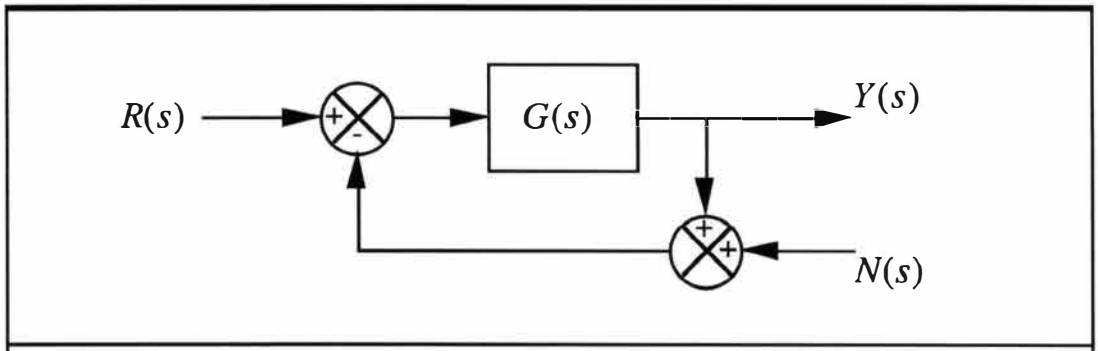

Figure 7.1 A simple control system with noise coupled into the feedback loop.

The transfer function between the controlling input, $R(s)$, and the output, $Y(s)$, is

$$\frac{Y(s)}{R(s)} = \frac{G(s)}{1+G(s)} \tag{7.4}$$

The transfer function between the noise, $N(s)$, and the output, $Y(s)$, is

$$\frac{Y(s)}{N(s)} = \frac{G(s)}{1+G(s)} \tag{7.5}$$

The transfer function is the same for the noise as it is for the controlling input variable. Thus the noise has effectively the same control over the system as the input. The noise can be removed with filters if it is in a frequency band different than the frequencies of interest. Preventing the noise from entering the system is a better approach. This may be accomplished by choosing sensors, such as optical sensors, which are by nature immune to noise interference during the transduction

process. Careful shielding and grounding then minimizes the noise corrupting the signal after transduction.

The layout of the sensors within the system is also important. Essentially, the sensors are placed so as to measure the parameters of interest. Typical sensors at the joints measure positions, velocities, forces, and torques. Sensors at the end-effector are commonly of the tactile, proximity, and force/torque variety. External sensors, such as ultrasound or laser, measure the position and orientation of the end-effector in workspace coordinates. Sometimes there may be more than one sensor of the same variety in almost the same location. For instance, angular sensors at the joints might be placed on both sides of the gear train. The sensor on the motor-side of the gear train provides stability in the control loop (backlash in the gears adversely affects stability), while the sensor on the output side measures the actual angular output of the joint.

Finally, the issue of sensor fusion must be addressed. The use of multiple sensors often provides redundant information. For example, position sensors at the joints and external end-effector sensors both provide information about the position and orientation of the end-effector. The most accurate sensor data should be chosen. Choices based on intuition may be acceptable when there are only a few redundant sensors. Statistical techniques are common when there are many redundant sensors.

7.2.6 Data Communication

Data communication within a robot having many joints and sensors is clearly an issue. The signal carriers must be routed over a demanding physical

pathway that includes turning joints and many tight spaces. Reconfigurable robots compound the problem by requiring connections at each of the module interfaces. The total number of pathways required to carry the information must be addressed. Also, the noise immunity of the pathways and their ability to withstand the demands of the robotic environment are important considerations. Data is typically carried by radio waves, conducting wires, and fiber optics.

The total number of conductors devoted to carrying information in even a simple industrial robot is amazingly large. The Cincinnati Milacron T3-776, for example, has over seventy-five conductors devoted exclusively to carrying data. This robot has six joints and the only sensors are one tachometer and one course/fine resolver. An advanced redundant robot with a wide variety of sensors could easily require hundreds of data pathways. Routing a single conductor for each of these data pathways would be absurd. The solution is to multiplex the information and transmit it over fewer physical carriers. The bandwidth of the carrier determines the amount of information placed on each line. Redundant lines may be required for fault tolerance.

Radio waves are one method of carrying data from place to place. The elimination of routing and connection problems is the main benefit associated with using radio waves. There is also no requirement for multiple lines for fault tolerance, just multiple transmitters and receivers. Because there is essentially only one transmission medium, only a limited amount of data can be carried over radio waves. There also may be problems associated with interference.

Electrical conductors are the most common method of carrying data through robots. Because a tremendous variety of machines use conductors, there has been

considerable development of conductor technology. This includes very flexible conductors capable of withstanding the twisting and turning of revolving joints. Because conductors usually require connectors, there has also been considerable connector development. Multiplexing techniques allow conductors to carry many signals and extra conductors can be used if there is more information than will fit on a single conductor. Carefully shielded conductors have good noise immunity.

A single fiber optic can carry very many signals. This is because fiber optics operate at frequencies in the light spectrum. The exact number of signals depends on the signals' bandwidth, but for signals with a bandwidth to 10 KHz a single fiber optic can carry about 100 individual signals. Fiber optics also have a very good noise immunity. The difficulty with using fiber optics is associated with the stress on the fiber due to flexing. Movements of the robot's joints cause flexing. The problems with flexing require very careful routing of the fiber through physical pathways designed specifically for fiber optics. Making fiber optic connections is also difficult. Imperfect connections distort the optical signal. These connections typically use a special fiber optic glue. Reconfigurable robots require connectors that can be repeatedly connected and disconnected (no glue).

7.2.7 Control

An advanced robot places tremendous demands on its controller. Almost every characteristic that makes a robot advanced corresponds to increased demands on the controller. These characteristics include: fault tolerance, kinematic redundancy, modularity, and multiple sensing capabilities.

Fault tolerance requires an extremely intelligent controller. The controller must first identify the fault. Once a fault is detected, the controller must respond appropriately. This requires a course of action determined by both the location and the type of fault. For instance, consider a short-circuit in one actuator of a redundantly actuated joint. The drive power to the faulty actuator must be interrupted, since supplying power into a short-circuit is a dangerous activity. The controller must then disengage the failed actuator from the joint with a clutch. Finally, the controller must notify other components in the system of the fault so that they can respond appropriately. Upon being notified of the reduced force capabilities in the faulty joint, the inverse kinematics algorithm can redistribute the torque demands among the joints. Thus, a controller for a fault-tolerant robot must identify a fault, initiate a course of action, and communicate news of the fault to other components in the system.

The addition of kinematic redundancy increases the demands on the controller. Clearly, the controller must be more powerful to drive the extra joints. Since errors accumulate in a serial robot, a robot with more joints accumulates more errors. A very accurate controller is necessary to minimize the errors. There also may be increased demands on the stability of the controller, especially in the case of a hyper-redundant manipulator, due to dynamic coupling among the joints.

Modular and reconfigurable robots place a unique set of demands on the controller. The controller must automatically adjust to a reconfiguration of the system. For this, the controller will act in concert with the automatic plant description system described in Chapter 5. The controller also must control a wide

variety of modules. Finally, the controller should be easily expandable if more modules are added to the robot.

A robot with a wide variety of sensors also increases the demand on the controller. The controller must interface and communicate with all of the sensors. Once the sensor data has been accumulated, it must be interpreted and the appropriate response undertaken.

7.2.8 Mechanical Architecture

A modular mechanical architecture will almost certainly form the foundation of any truly advanced robotic system. Some advanced applications for robotic systems explicitly call for modular robots. Other advanced applications implicitly require modularity. Numerous journal articles, master's theses, and doctoral dissertations detail the benefits of modular robots. This discussion focuses on modularity from an applications perspective.

A truly modular and reconfigurable robot requires a tremendous amount of supporting technology. Virtually every component of the system must be designed with modularity in mind. The interface requirements alone are staggering. Both mechanical and electrical connections must be established quickly and easily as the modules are assembled to form the robot. Fiber optic, pneumatic, and hydraulic carriers may be connected as well. The operational software and system controller must also be easily reconfigurable. Without these, and other, supporting technologies, the modular robot will not be truly reconfigurable.

Other advanced applications require a very precise and dependable robot with multiple sensing capabilities and kinematic redundancy. The question is how

to develop these robots, because they aren't available now. One avenue follows the example set by past robot development efforts. Unfortunately, this path would be expensive, slow, and likely unsuccessful. For example, the Cincinnati Milacron T3 series of manipulators required seven million dollars of development expense and took seven years to bring to market (Tesar and Butler, 1989a). The NASA space shuttle manipulator was delivered at a total cost of one hundred million dollars (Tesar and Butler, 1989b). The FTS robot, which was supposed to be the next-generation of advanced space robot, was canceled after almost four hundred million dollars in development costs (NASA, 1992). Clearly these are not the models to emulate.

The personal computer is a much more successful model. The personal computer was introduced about ten years ago. The architecture of this personal computer has become very familiar. It is a microprocessor and RAM connected to other system components, such as the hard-drive and video display, through a bus. Because these fundamental components were kept separate, progress was made on each one of them independently. These separate components were improved incrementally without the tremendous expense associated with redesigning the entire system. This modular approach made possible today's personal computers; which have very much the same architecture as those from ten years ago, yet are hundreds of times better by almost any measure. During this same ten years, the mechanical design of robots remained stagnant. Even though not all of the advanced applications discussed in this chapter explicitly call for reconfigurability, the requirements they place on a robot will only be met by adopting a modular mechanical architecture.

7.2.9 Mechanical Design

The truly advanced robot's mechanical design is another hurdle remaining. This advanced robot will certainly incorporate a wide-variety of sensors. Fault-tolerance capabilities may call for clutches to disengage faulty actuators and, like common industrial robots, the advanced robot requires brakes, motors, and gears. A kinematically redundant robot also requires the set of components associated with each additional joint. These components can all be purchased "off-the-shelf" and incorporated into the mechanical design of the robot, or they can be designed specifically as integral elements. There are advantages and disadvantages associated with either approach.

Using standard off-the-shelf components is certainly easier. The monetary cost of these components is less because their manufacturer spreads the design expense over a wider variety of applications. Finally, using standard components that are based on proven and tested designs involves less risk. The advantages associated with ease of use, lower expense, and lower risk come with some serious disadvantages. The main disadvantage is that standard components include their own enclosures and mounting hardware. This makes them both heavy and bulky. Given that an advanced robot incorporates a large number of sensors and other components, this added bulk and weight is likely unacceptable. In any case, added bulk and weight certainly reduce the performance capabilities of the robot.

The alternative to using standard components is integrating the functions they perform as part of the robot's design. This requires a large concerted effort encompassing several engineering disciplines. A well-planned and systematic

design effort is essential. The initial expense of this effort will likely be greater than designing a robot using standard components. There is also a significant amount of risk involved since the failure of even one of the components will jeopardize the success of the entire effort. If, however, the designs are successful and perform as they are intended, the benefits will be tremendous. The robot will be much lighter and more compact. The savings in weight and bulk come from eliminating the enclosures and mounting hardware associated with off-the-shelf components. The robot should also be stiffer and more precise since intermediate bearings and shafts are eliminated. Finally, the sensors should gather data of a higher fidelity since they can be located to more precisely measure the physical phenomena of interest.

7.2.10 Robot Actuators

Robot actuators are the final supporting technology this chapter discusses. The actuators supply the power that moves the robot's joints. A robot actuator has two main parts. The first is the prime-mover, which actually develops the power. The second part is the drive elements which are between the prime mover and the motion output of the actuator. By nature, kinematically redundant robots have more actuators than their non-redundant counterparts. Unfortunately, actuator technology is probably the largest factor limiting robots. The actuators are typically the heaviest and most expensive part of the robot.

A comparison between robot's and human's prime-movers gives an impression of why actuators are currently limiting the performance of robots. The actuators in the human biomechanical system are the individual muscles. The biceps muscle of the arm is a representative muscle that has been extensively

studied. The biceps muscle weighs, on average, about .27 pounds and can generate 14.75 foot pounds of torque (Winters and Stark, 1988). This gives the biceps muscle a torque-to-weight ratio of 55.66 foot-pounds per pound of actuator. For comparison the N9M4 electric motor from PMI Corporation, which is made with current state-of-the-art rare-earth magnets, weighs 3.1 pounds and generates .36 foot pounds of torque (PMI, 1991). This gives the N9M4 electric motor a torque-to-weight ratio .116 foot-pounds per pound of actuator. Using the torque-to-weight ratio as the sole measure of performance, the human muscle is about 480 times better than the current state-of-the-art in robot prime movers.

Basing the above comparison solely on torque-to-weight ratio is not entirely fair. Gear trains and other drives will improve an actuator's torque-to-weight ratio. Also, humans have a limited ability to produce a controlled force and cannot endure heavy loads for very long. Finally, mechanical and electromechanical actuators can develop much higher total forces than can any human. For instance, a large robot can carry and manipulate an engine block from a car. Most human's cannot carry this heavy of a load with their muscles.

The drive elements in robot actuators, which are very poor, compound the deficiencies of the prime-mover. The drive elements connect the prime mover to the motion output of the actuator, typically with a gear drive or other speed reducer. The prime movers and speed reducers are almost without exception in separate cases with separate shafts mounted on separate bearings. These separate cases, shafts, and bearings all contribute excess weight. There is compliance, and the associated imprecision, contributed by the extra shafts. The speed reducers also contribute compliance and, especially if they are gears, backlash. Backlash is a

major source of imprecision and makes the actuator's controller less stable. In keeping with the biomechanics comparison, the muscles are connected to their output at the joints via tendons with no backlash and very little compliance.

Clearly, the actuators in robots are not very good, especially when compared to the actuators in humans. What can be done to improve robot actuators? Basically, the torque-to-weight ratio and the precision must be improved. Designing the prime-movers to generate more power, perhaps with forced cooling or superconducting technology, will improve the torque capabilities. Eliminating the excess cases, shafts, and bearings will decrease the weight. Ultimately, the actuator should be designed as a single modular unit.

7.3 PRIORITIZATION OF THE COMPONENT TECHNOLOGIES

This prioritization focuses on generalized inverse kinematics and the ten advanced applications discussed in this chapter. The estimation of these numbers took into account the previous discussion of the ten advanced applications and supporting technologies.

The features that distinguish an application as being advanced also influence its most important supporting technologies. Those applications dependent upon generalized inverse kinematics in turn are especially dependent upon computer hardware and computer software for executing the algorithms. Performance criteria are clearly an indispensable element of generalized inverse kinematics. Robots with the extra degrees of freedom necessary to make the most use of generalized inverse kinematics place extremely high demands on their actuators. Obstacle avoidance is dependent upon generalized inverse kinematics, but also includes requirements for

sensors to identify obstacles and data communication to support these sensors. Applications requiring fault tolerance place the most demands on the supporting technologies. Fault tolerance places high demands on computer software, sensors, data communication, control, mechanical architecture, and actuators. Applications for modular robots demand generalized inverse kinematics, mechanical architecture, mechanical design, and actuators. Tables 7.7 prioritizes and presents an estimate of the possible improvements in the ten supporting technologies over the next five years. Table 7.8 presents some numerical support for the above observations relating the supporting technologies to the advanced applications.

Table 7.7 The prioritization and short-term improvement possible for the ten component technologies.

Technology	**Relative Importance**	**Possible Improvement Over 5 Years**
computer hardware	8.0	100+
computer architecture	6.5	8
computer software	5.5	3
performance criteria development	9.0	2
sensors	5.0	7
data communication	9.0	10
control	5.0	2
mechanical architecture	7.5	4
mechanical design	7.0	2
robot actuators	10.0	100

Table 7.8 Relative demands placed on each of the supporting technologies by each of the advanced applications.

	computer hardware	computer architecture	computer software	criteria development	sensors	data communication	control	mechanical architecture	mechanical design	robot actuators	
complex assembly	7	4	6	8	6	5	7	5	5	10	6.3
food packaging	5	3	5	2	2	3	9	9	5	7	5.0
micromanipulation	3	3	3	7	2	2	6	2	9	9	4.6
in-space assembly and construction	8	8	9	7	8	8	7	8	8	9	8.0
satellite retrieval and repair	8	8	10	8	9	10	8	9	9	10	8.9
E.V.A. site preparation	2	2	2	2	3	4	5	6	5	5	3.6
semiconductor wafer handling	3	3	6	5	4	4	2	7	6	5	4.5
long reach arm	6	6	7	8	8	5	6	8	10	10	7.4
inspection at nuclear facilities	7	7	8	6	9	5	5	8	6	6	6.7
decontamination and dismantlement	8	7	8	8	9	8	8	9	9	9	8.3