

## REPORT DOCUMENTATION PAGE

REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS		
SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION / AVAILABILITY OF REPORT  Unlimited Distribution		
DECLASSIFICATION / DOWNGRADING SCHEDULE				
PERFORMING ORGANIZATION REPORT NUMBER(S)  RSD-TR-23-87		5. MONITORING ORGANIZATION REPORT NUMBER(S)		
NAME OF PERFORMING ORGANIZATION College of Engineering The University of Michigan	6b. OFFICE SYMBOL <i>(If applicable)</i>	7a. NAME OF MONITORING ORGANIZATION USAF/AFSC Aeronautical Sys Div/PMRRC		
ADDRESS (City, State, and ZIP Code)  Ann Arbor, Michigan 48109-2110		7b. ADDRESS (City, State, and ZIP Code)  Wright-Patterson AFB, OH 45433-6503		
NAME OF FUNDING / SPONSORING ORGANIZATION	8b. OFFICE SYMBOL <i>(If applicable)</i>	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER		
ADDRESS (City, State, and ZIP Code)		10. SOURCE OF FUNDING NUMBERS		
		PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.
				WORK UNIT ACCESSION NO.

TITLE (Include Security Classification)  
Flexible Automatic Discrete Parts Assembly

PERSONAL AUTHOR(S)  
D.E. Atkins, R.A. Volz

1. TYPE OF REPORT Annual	13b. TIME COVERED FROM 9/1/86 TO 8/31/87	14. DATE OF REPORT (Year, Month, Day) December 1987	15. PAGE COUNT
-----------------------------	---	--	----------------

SUPPLEMENTARY NOTATION

COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number) Manufacturing Science, Robotics, Artificial Intelligence, Sensors, Machine Vision, Special Purpose Processors, Motion Planning, knowledge Based Systems, Requirements Analysis.
FIELD	GROUP	SUB-GROUP	

ABSTRACT (Continue on reverse if necessary and identify by block number)

This project addresses the problem of flexible automated assembly. The long term goal is to develop technologies that can be applied to many different aspects of assembly automation and that can be evolved to successively higher levels of abstraction, i.e., that will eventually allow high level commands, such as "assemble product X", to be given. We are addressing three aspects of the problem, vision, planning/programming, and system integration. This report describes the second year's effort toward these goals. Specific achievements during this reporting period are:

- New matching techniques based on k-d trees have been developed to greatly speed up object recognition.

DISTRIBUTION / AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS		21. ABSTRACT SECURITY CLASSIFICATION	
3. NAME OF RESPONSIBLE INDIVIDUAL		22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL

19. (continued)

- We have developed replanning strategies for dealing with failures in assembly. We have applied the approach to peg-in-the-hole tasks, and we have developed a program to test the method through simulation.
- We have developed a simple query language named ESQL tailored specifically to engineering databases. Specifically, it can be used for engineering project management.
- The workstation-based rapid prototyping "vision workbench" developed in previous years has been enhanced to support video windows.
- A series of algorithms for segmenting range images has been successfully ported to our hypercube parallel computer a 64 processor NCUBE/6. Significant performance improvements have been observed. In the case of low-level vision algorithms close to linear speedup has been achieved.
- The basic dimensions and issues involved in distributing the execution of Ada programs have been identified. A prototype translator to allow distributed execution has been developed and is being tested.
- Object recognition from partial views has been extended so that the viewer need not be at a known distance from the object.
- Surface classification and region generation algorithms based on bivariate polynomials have been developed. Initial experiments suggest these methods can tolerate partial occlusion of the surfaces.
- Prior work on Ego-motion Complex Log Mapping (ECLM) was shown to maintain projection invariance for arbitrary translational motion. We have used it to recover qualitative depth information, in particular relative orderings of objects in a scene.
- Our work on knowledge based techniques for finding events in sequences of images continues, and is now being applied to extract depth from stereo.
- Our work in dynamic scene analysis has been successfully applied to autonomous vehicle "road following."
- We have begun work on scene analysis using Moise interferometry techniques.
- A simple probabilistic measure of "workspace" has been developed that accounts for various sizes and shapes of obstacles. It will form the basis of our path planning algorithms.
- Two general high level planning strategies have been developed. They are subgoal ordering and goal augmentation. Tests have shown that they significantly improve heuristic planning.
- The approach to assembly proposed in previous work has been modified. the new technique is based on constraint diagrams, and a set of operations that can be applied not only in forward and backward search strategies but also in "opportunistic" strategies.

FLEXIBLE AUTOMATIC  
DISCRETE PARTS ASSEMBLY

ANNUAL REPORT

September 01, 1986 ~ August 31, 1987

Principal Investigator

R.A. Volz

Contract F33615-85-C-5105



# EXECUTIVE SUMMARY

## FLEXIBLE AUTOMATIC DISCRETE PARTS ASSEMBLY

### Motivation and Goals

There is great emphasis today on the development of fully intergated manufacturing, including both the automation of individual manufacturing operations and the integration of all phases of manufacturing, from design through planning through production. The associated problems, however, are extremely complex and will require a large research effort over and extended period of time. This project is focused on an important category of flexible automation problems, in itself quite large, for which there are yet very few operational systems, flexible automated assembly. Assembly is the most highly labor intensive manufacturing process in the production of durable goods. There is thus great potential for direct cost saving through the development of flexible automated systems for assembly, as well as the indirect benefits mentioned above.

The long term goal of this work is to develop technologies that can be applied to many different aspects of assembly automation and that can be evolved to successively higher levels of abstraction, i.e., that will eventually allow high level commands, such as “assemble product X”, to be given. We are initially addressing three aspects of the problem, vision, planning/programming, and system integration. This report describes the second year’s effort toward these goals.

### Context of Research

Figure 0.1 portrays a taxonomy of an automated assembly system into which our work fits. The boxes represent capabilities and operations that must be developed while inputs and outputs are shown in ovals. The vertical sequence of operations and outputs proceeding downward in the middle of the diagram illustrates the sequence of operations that must be performed in order to automatically assemble a product. The other boxes indicate essential capabilities which must be available in order to achieve the operation shown in the center. Few of the components shown are adequately developed at present. Moreover, many of the different constituents in an automatic assembly system will be implemented on different computers and, possibly, in different languages. Integration of these constituents into a working whole is an extremely important aspect of the overall problem.

The distinction between off-line and on-line activities is somewhat variable among different systems, but as more sophisticated and complex planning algorithms are developed will tend to move downward toward the position shown in the figure. Our work addresses aspects

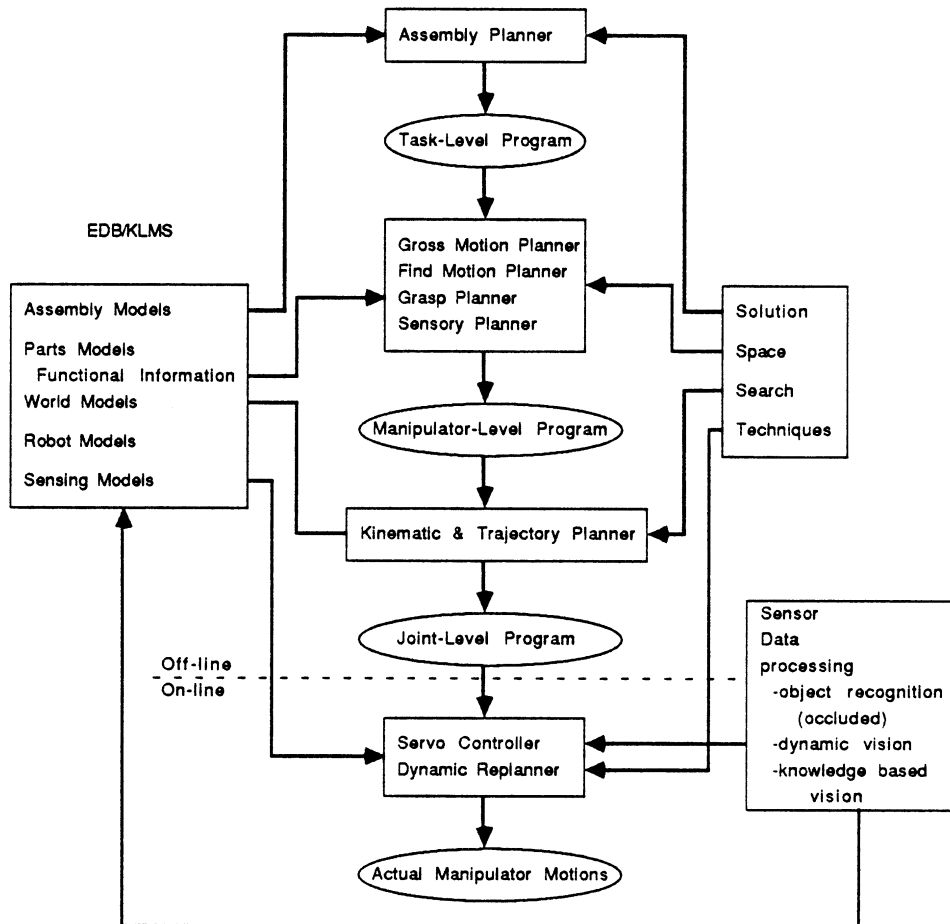


Figure 1: Interrelation between components in an automated assembly system

of most of the off-line operations shown, the sensor data processing portion, primarily vision, of the on-line processing and the systems integration techniques by which such a system can be made operational. Our approach is to use existing technology, where available, coordinate with other research projects in the Robotics Research Laboratory and focus the research on this project toward key underlying technologies.

## Highlights of Research Accomplishments

While the activities being reported cover the second year of this contract, they also, to a certain extent represent a transition from the previous contract. We thus report some activities which were completed during the year, work continuing from the previous year, and new activities initiated during the second contract year. The following paragraphs briefly overview the principal accomplishments in each of the major research areas. It is important to recognize that in each of these areas there is significant additional work being done under the auspices of other contracts acquired, in part, due to the recognition and base of support provided by the Air Force.

### *Computer Vision*

A flexible assembly system must be able to perceive its environment in order to take appropriate actions. Computer vision is being increasingly used to help machines achieve this perception. We are addressing several aspects of vision particularly important to manufacturing, including the following:

- Object recognition from partial views,
- Object recognition using range images,
- Dynamic scene analysis,
- Motion stereo,
- Qualitative vision, and
- Knowledge based vision techniques.

Specific achievements during this reporting period are:

- New matching techniques based on k-d trees have been developed to greatly speed up object recognition.
- Object recognition from partial views has been extended so that the viewer need not be at a known distance from the object.

- Surface classification and region generation algorithms based on bivariate polynomials have been developed. Initial experiments suggest these methods can tolerate partial occlusion of the surfaces.
- Prior work on Ego-motion Complex Log Mapping (ECLM) was shown to maintain projection invariance for arbitrary translational motion. We have used it to recover qualitative depth information, in particular relative orderings of objects in a scene.
- Our work on knowledge based techniques for finding events in sequences of images continues, and is now being applied to extract depth from stereo.
- Our work in dynamic scene analysis has been successfully applied to autonomous vehicle “road following.”
- We have begun work on scene analysis using Moise interferometry techniques.

### *Planning and Programming*

As is evident from Figure 1 there are several levels of planning and support tools that are necessary to achieve automated assembly. The areas investigated during the past year include:

- Gross motion path planning,
- Assembly sequence planning,
- Planning how to use sensor information to overcome uncertainties in geometries (tolerances), sensor errors, and control errors,
- Tolerancing,
- Graphical robot programming,
- Heuristic problem solving, and
- Development of an Engineering Database/Knowledge Management System.

Accomplishments during the past year include:

- A simple probabilistic measure of “workspace” has been developed that accounts for various sizes and shapes of obstacles. It will form the basis of our path planning algorithms.



- Two general high level planning strategies have been developed. They are subgoal ordering and goal augmentation. Tests have shown that they significantly improve heuristic planning.
- The approach to assembly proposed in previous work has been modified. The new technique is based on constraint diagrams, and a set of operations that can be applied not only in forward and backward search strategies but also in “opportunistic” strategies.
- We have developed replanning strategies for dealing with failures in assembly. We have applied the approach to peg-in-the-hole tasks, and we have developed a program to test the method through simulation.
- We have developed a simple query language named ESQL tailored specifically to engineering databases. In particular, it can be used for engineering project management.

### *Systems Integration and Architecture Techniques*

Systems integration is essential for any robotics or manufacturing system, but is today done with little more than ad hoc tools, and often starting from scratch each time a system is built. Moreover, there is little experience with the use of newer forms of parallel architectures for robotics and manufacturing applications. In this area we are investigating:

- A robotics rapid prototyping system,
- The mapping of (vision) algorithms unto parallel architectures, and
- Distributed Language techniques, which we believe to be fundamental to developing a systematic way of accomplishing systems integration.

Specific accomplishments during the past year include:

- The workstation-based rapid prototyping “vision workbench” developed in previous years has been enhanced to support video windows.
- A series of algorithms for segmenting range images has been successfully ported to our hypercube parallel computer, a 64 processor NCUBE/6. Significant performance improvements have been observed. In the case of low-level vision algorithms close to linear speedup has been achieved.
- The basic dimensions and issues involved in distributing the execution of Ada programs have been identified. A prototype translator to allow distributed execution has been developed and is being tested.

## Organizational Considerations

The second year has seen a continued excellent performance in terms of student participation and technical publication. Summary statistics directly supported by Air Force funding are:

- 13 papers presented at major conferences,
- 14 papers appeared in reviewed journals,
- 8 papers are currently under review or have been accepted for journal publication or conference presentation.
- 16 graduate students participated in the program,
- 1 student participating in the program received the Ph.D.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Motivation and Goals . . . . .	3
1.2	Overview of Automatic Assembly . . . . .	4
1.3	Project Organization and Activities . . . . .	6
<b>2</b>	<b>Research</b>	<b>10</b>
2.1	Computer Vision . . . . .	10
2.1.1	Range Image Understanding . . . . .	10
2.1.2	Dynamic Scene Analysis . . . . .	13
2.1.3	Knowledge Based Computer Vision . . . . .	15
2.2	Planning and Programming . . . . .	18
2.2.1	Real-Time Collision-Free Path Planning Using Probability . . . . .	18
2.2.2	Automated Robot Task Planning . . . . .	24
2.2.3	Automatic Assembly Planning . . . . .	28
2.2.4	Replanning . . . . .	31
2.2.5	Automatic Determination of Retrieving Paths in a Semantic Data Model	38
2.3	System Integration Techniques . . . . .	43
2.3.1	The Robotics Prototyping System . . . . .	43
2.3.2	Research into Vision Algorithms . . . . .	48
2.3.3	Distributed Systems Integration Techniques . . . . .	53
<b>3</b>	<b>Publications</b>	<b>63</b>
3.1	Journal Publications . . . . .	63
3.2	Conference Presentations and Papers . . . . .	65
3.3	Under Review . . . . .	67
<b>4</b>	<b>Personnel</b>	<b>68</b>
4.1	Faculty . . . . .	68

4.2	Students . . . . .	69
4.3	Degrees Awarded . . . . .	70
4.4	Graduate Student Placement . . . . .	70
4.5	Permanent Staff . . . . .	71
<b>5</b>	<b>Coupling Activities</b>	<b>72</b>
5.1	Intra Project Interactions . . . . .	72
5.2	University Interactions . . . . .	72
5.3	Interactions with Industry . . . . .	73
5.4	Government Interaction . . . . .	75
<b>6</b>	<b>Distribution List</b>	<b>76</b>

# FLEXIBLE AUTOMATIC DISCRETE PARTS ASSEMBLY

## 1 Introduction

### 1.1 Motivation and Goals

There is great emphasis today on the development of fully integrated manufacturing. Indeed, the development of integrated manufacturing techniques is critical to the survival of many segments of United States' industry. Integrated manufacturing includes both the automation of individual manufacturing operations and the integration of all phases of manufacturing, from design through planning through production. The motivations for integrated manufacturing, ultimately, are economic, though there are numerous intermediate reasons such as improved quality, reduced lead time on introducing new products, reduced inventories (through the ability to quickly produce new parts when needed) and the ability to rapidly prototype new designs, in addition to the traditional reason of reduced direct production cost.

This project is focused on an important category of flexible automation problems for which there are very few operational systems, flexible automated assembly. Assembly is the most highly labor intensive manufacturing process in the production of durable goods [1]. There is thus great potential for direct cost saving through the development of flexible automated systems for assembly, as well as the indirect benefits mentioned above.

There are many levels of assembly automation one can seek. Ultimately one would like to be able to give a very high level command such as "Build 30 of product X by deadline," and have the system automatically schedule the assembly process, order the components and tools needed, schedule their delivery to the work area, determine the assembly sequence and fixturing required, generate the robot programs required, and manage the flow of all material and information to accomplish the assembly. The achievement of this level of automatic operation is a long way off. A slightly lower, but also very difficult level is the task level, in which one would like to be able to give commands such as "assemble product X," "insert bolt A in hole B," or "grasp bracket C" under the assumption that fixturing, materials, and tooling are all available inputs; even within this level there are obviously many sublevels.

The long term goal of our work is to develop technologies that can be applied to any of several levels of assembly automation and can be evolved to higher levels of abstraction. In particular we are addressing three aspects of the problem, vision, planning/programming, and system integration.

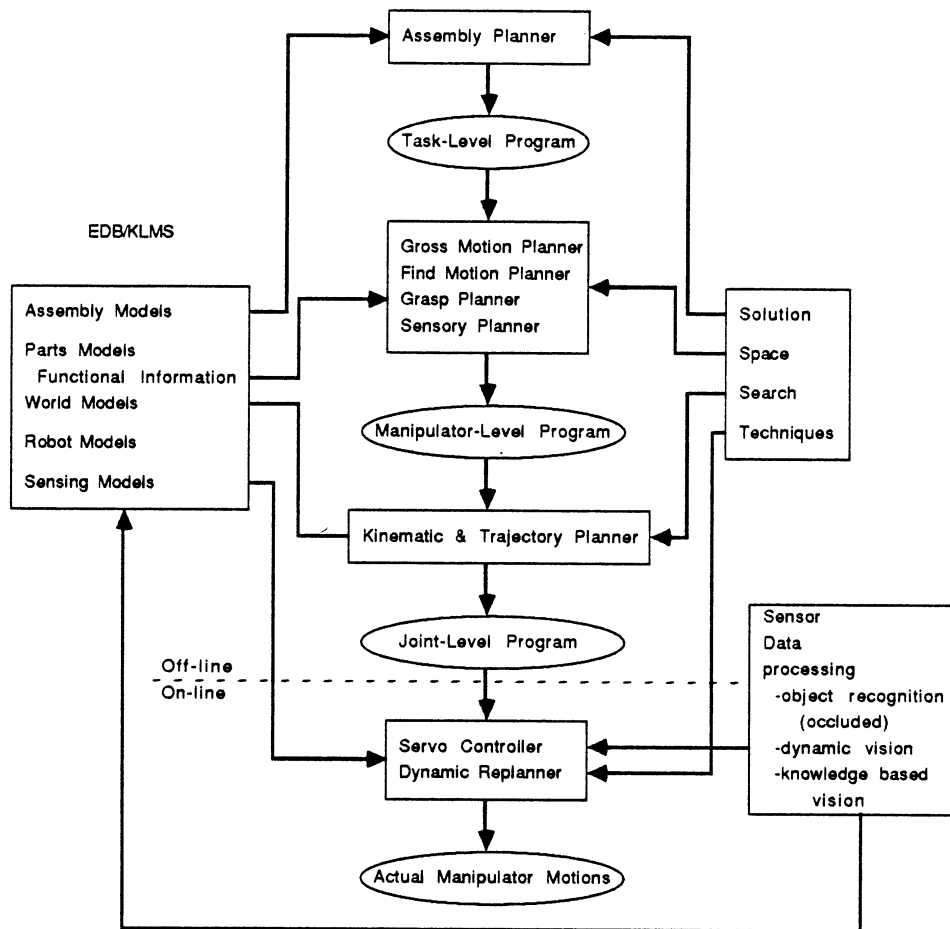


Figure 2:

## 1.2 Overview of Automatic Assembly

Figure 2 portrays the major components which must be part of an automated assembly system at the task level (i.e., the “assembly product X” level), and the relationships between them. The boxes represent capabilities and operations while inputs and outputs are shown in the ovals. The vertical sequence of operations and outputs proceeding downward in the middle of the diagram illustrates the sequence of operation which must be performed in order to automatically assembly a product. The other boxes indicate essential capabilities which must be available in order to achieve the operation shown in the center. Few of the components shown are adequately developed at present. Indeed, even the appropriate contents of the database are not fully known at this time.

We assume as inputs to the process a wide variety of design, work cell model and sensor information. In order to manage this information, an Engineering Database/Knowledge Management System (EDB/KMS) is necessary that combines the data storage/retrieval capabilities normally present in database management system with semantics extraction and inferential mechanisms normally associated with knowledge management systems. Further, the EDB/KMS must support access at various levels of abstraction with corresponding levels of access times.

The assembly planner produces the sequence in which the components to the product are to be assembled, and the associated fixturing and orientation information for each step in the sequence, called the (a subtask level program) in Figure 1. Each step at the subtask level consists of a single operation such as placing a part in a fixture, changing a tool on a robot end effector, or mating two parts.

Typically there is a robot motion associated with each subtask step resulting from the assembly sequence planner. Each such subtask requires several further stages of planning. First, the motion from the initial position of the robot to a position in free space near its final position must be determined which avoids collision and minimizes an appropriate cost function, e.g., time or energy (gross motion planning). Next motions involving contact among the parts being mated must be planned (fine motion planning). These two planning stages are separated because some gross motion planners [2] do not permit contact (though in some other cases the two might be performed jointly [3]). Also, if a component is to be picked up by a robot, the point at which it is to be grasped must be chosen so that no interference occurs among the hand, the component and other parts of the assembly, and also so that the component will not slip in the grasp of the robot. Finally, at this stage of planning, sensor plans must be developed which can use the robot's sensors to determine whether or not a subtask step was successfully completed.

From the perspective of the robots, the output of subtask planning stage is a sequence of specific motion, gripper and sensor commands in a world coordinate system of the workspace. The kinematic and trajectory planner converts these motions into joint motions of the robots involved and determines the planned time history of each joint along the desired path.

The servo level, of course, directs the movement of each joint of the robots. It will be important for assembly tasks that the control system be capable of compliant motion (e.g., move in a given direction while maintaining a prescribed contact force). Guarded motions, i.e., motions that continue until some sensed force, torque, or position condition is satisfied, will also be important and, indeed, are closely related to the sensor planning for subtask verification. Essentially, as each guard condition is satisfied and a motion stopped, the sensor readings are compared with planned values to determine whether or not the desired motion actually occurred. Unfortunately, uncertainties in part geometries, actual gripping position, sensor errors, control errors, etc., fairly frequently result in a motion being stopped due to

excessive force or torque. When this is detected, as a result of the sensor verification plan, a dynamic fine motion replanning is necessary. Since presumably the robot is close to the desired position, this planner can be much simpler, and hence done in real-time, than the original planner.

Several of the planning stages required searching extensive solution spaces. Efficient techniques are necessary if automatic planning is to be realistically achieved.

It is obvious that a multitude of sensing is required for successful operation of an assembly system. Vision, force/torques, tactile and range sensors are the primary candidates, though if the assembly involves are welding, temperature sensing (of the weld puddle) may prove useful. Many of the actual motion target positions will be determined from sensor information, some servoing of the motions may be driven by higher level sensors (e.g., vision) and certainly sensing is required for verification.

Finally, many of the different constituents in an automatic assembly system will be implemented on different computers and, possibly, in different languages. Integration of these constituents into a working whole is an extremely important aspect of the overall problem.

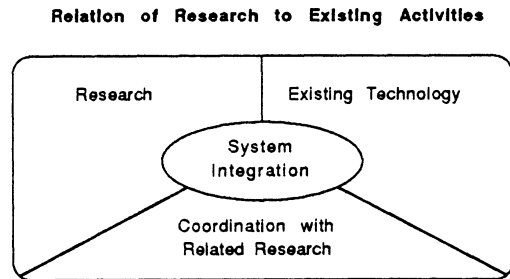
### 1.3 Project Organization and Activities

Fully automated discrete parts assembly is a very complex problem and will require a large research effort over an extended period of time. Our approach is to use existing technology, where available, coordinate with other research projects in the Robotics Research Laboratory and direct the research on this project toward key underlying technologies. This concept is illustrated in Figure 3. In particular, our research emphasis may be grouped in three important categories, machine vision and sensing, programming and planning, and systems integration techniques. Figure 4 shows the organization of the project in greater detail, and identifies the major subprojects.

The machine vision research is exploring a variety of techniques ultimately useful for object recognition and pose determination. This is being studied via single image analysis, the analysis of a sequence of images, and the use of knowledge based approaches and with both grey level and range images as inputs. Results obtained for occluded part recognition have been shown to outperform previously known techniques, the fundamental basis for differential geometric descriptions and segmentation of surfaces in range images has been established, and promising work has been started on the mapping of vision algorithms onto parallel processors of a hypercube architecture.

Work in the planning and programming area has produced significant results in path planning, grasping and heuristic problem solving techniques (which are continuations of





**Air Force supported research**

- Vision and sensing
- Programming and automatic planning
- Systems integration and rapid prototyping technology

**Coordination with related research**

- Development of distributed systems integration language
- Relation of (vision) algorithms and architectures
- Automatic robot program modification to accommodate component redesign
- Development of tactile sensor

**Use of Existing technologies**

- Robots, and effectors and controls
- Sensors

Figure 3:

previous work) and begun important new activity in assembly and sensor planning. The extension of parallel finger grasping to include nonuniform grasping pressure, general loading forces and accurate computation of torques resisting slippage has been more fully tested and completed. In the area of general problem solving, the techniques obtained have been shown to be very robust, exhibiting performance equal to the best individual domain specific solution obtained across a broad range of problems.

To date there has been almost no work on automatic planning of assembly sequences. We are initially approaching the problem by considering a restricted class of assembly tasks involving primarily axis oriented operations, i.e., placing parts (e.g. brackets, washers, nuts) on a shaft of some type. In the area of sensor planning we are introducing two important new principals: 1) establishing relations between assembly design constraints and the use of sensors to guide the assembly operation in the present of uncertainties, and 2) the reduction of the problem dimensionality through the use of *contact formations* as the basis for verification and replanning rather than six dimensional configurations. This effort is only in its early stages, but initial results are encouraging and some design constraints which allow certain formations to be distinguished, even in the presence of uncertainties, have been established.

The systems integrations work has had two aspects: 1) building a rapid prototyping robot/vision system which can be used as the basis for future experimentation and research,

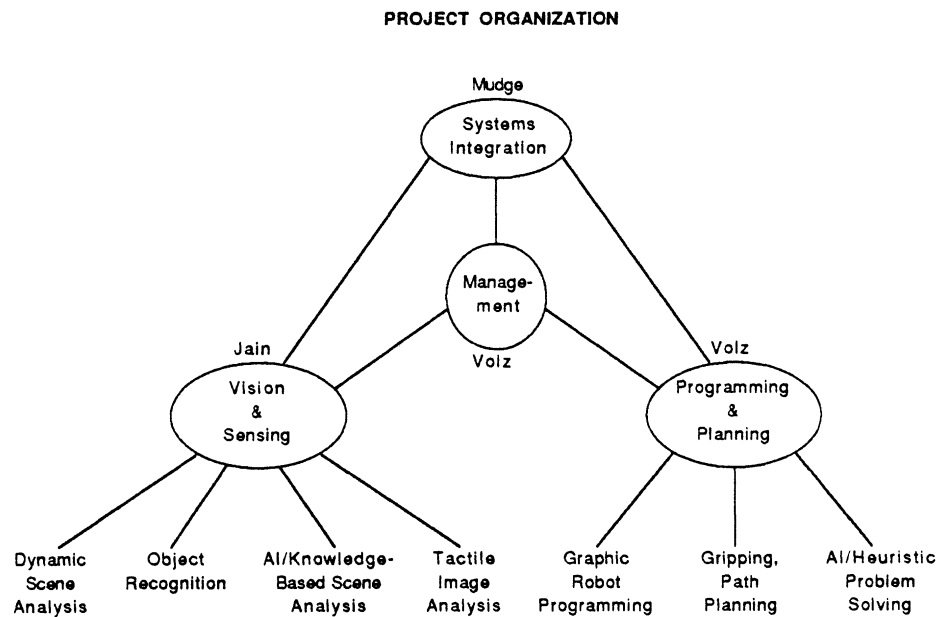


Figure 4:

and 2) serving as a focussing agent for systems integration techniques based around distributed languages (largely supported by other contracts). An initial version of the rapid prototyping system is operational. Image acquisition and edge detection essentially proceeds in real-time, making the complete real-time implementation of vision algorithms a real possibility in our future work. In view of the importance of and need for systems integration techniques for distributed systems and the role of this project as a focussing agent for our work in this area, our vision of distributed manufacturing software will be described and related activities described briefly.

## References

- [1] A.G. Boothroyd. "Use of Robots in Assembly Automation," *Annals of the CRP*, Vol. 3, No. 2, 1984.
- [2] E.G. Gilbert and D.W. Johnson. "Distance Functions and their Application to Robot Path Planning," *IEEE Journal of Robotics and Automation*, Vol. RA-1, No. 1, pp. 21-30, March 1985.

- [3] Tomas Lozano-Perez, and M.T. Mason and R.H. Taylor. "Automatic Synthesis of Fine Motion Strategies for Robots," *The Inter. Jour. of Robotics Research*, Vol. 3, No. 1, pp 3-24, 1984.

## 2 Research

### 2.1 Computer Vision

A flexible assembly system should be able to perceive the environment to take appropriate actions. Computer vision is being increasingly used to help machines perceive their environment. In our project, we are addressing several relevant aspects of computer vision. We are addressing the problem of object recognition from their partial views. We have started addressing the object recognition problem using range images. We continue to address problems in dynamic scene analysis and have made significant progress in our implementation of motion stereo and our understanding of qualitative vision. We are exploring application of knowledge based techniques to vision systems for assembly. Finally, we have just started work on 3-dimensional scene analysis using moire interferometry and expert system techniques (see Appendix A).

#### 2.1.1 Range Image Understanding

Investigator: R. Jain

##### Research objective

In most applications, a 3-dimensional object must be recognized from its 2-dimensional projections. In a scene, there may be several objects that may be only partially visible. Global features cannot be used to recognize objects from their partial views. Local features have been proposed for recognizing objects. The success of these has been limited due to the difficulties in their determination and matching.

A symbolic surface descriptor will capture the intrinsic nature of the surface as a global feature. Such a descriptor will be very useful in object recognition if it can be computed from partial views of object surfaces. Commonly used global features in object recognition lack this desirable characteristic. We are trying to develop the concept of symbolic surface descriptors for object recognition in range images.

##### Approach

A vision system must be able to recognize a 3-dimensional object from its 2-dimensional projections. It has been shown that recognition of objects by exhaustive matching is a computationally impossible task. Therefore, an object must be recognized by using features,

entities on an image which capture the surface shape structure. Conventional global features, such as areas, moments and Fourier descriptors, cannot be used to recognize objects from only partial views. Local features like corners and edges have been used with limited success due to the sensitivity of extraction methods and the difficulties in matching. Partial occlusions among objects may make local features disappear entirely. It is desirable to discover kinds of features that capture the global intrinsic nature of the surface and, at the same time, can be computed from partial views of surface. We propose to use symbolic surface descriptors extracted from range images as features for object recognition.

The shape structure of a surface is explicitly represented by the collection of depth values in a range image. We developed an approach for segmenting a range image into a set of smooth surface primitives [1]. Each surface is represented in explicit polynomial form,

$$z = a + bx + cy + dx^2 + exy + fy^2 + gx^3 + hx^2y + ixy^2 + jy^3 + \dots$$

By approximating surfaces with this equation up to quartic terms, it is possible to describe very complex surfaces. The coefficients of this bivariate polynomial function together with its domain of definition on the image are referred to as symbolic surface descriptor. The intrinsic nature of a surface is abstracted into these coefficients. Clearly, a symbolic surface descriptor is a global representation of a surface. It can be also computed from partial views of a surface due to minor occlusion by using well established approximation techniques.

The non-zero coefficients of a symbolic surface descriptor and relationships among them characterize the surface. It is possible to describe the shape of surface by looking at which terms are present in the polynomial representation. Based on the non-zero coefficients and relationships among them, we may categorize surfaces into several classes. Two surfaces with similar shapes should have symbolic descriptors which can be classified into the same "type". Each class of symbolic surface descriptors should be significantly different from each other in a way that enables us to tell the existence of a shape difference. Depending on the viewpoint, this explicit polynomial representation will change. However, the class of symbolic descriptor remains identical under small change of viewpoint; this intrinsic nature of the surface is not sensitive to small viewpoint variation. It is also important to know how the symbolic surface descriptor of a surface changes from one class to another under drastic changes of viewpoint.

For surfaces modeled by the biquadratic polynomial function

$$z = a + bx + cy + dx^2 + exy + fy^2 ,$$

we can classify surfaces into five different types: convex and concave elliptic paraboloid, convex and concave cylindrical surface and hyperbolic paraboloid [2]. The classification scheme is based on the second order terms in the equation which characterize the surface shape. The biquadratic polynomial equation is considered as an implicit representation of

a surface by  $g(x, y, z) = 0$  which give a quadric surface equation. The classical method to determine type of quadric surface is then applied to obtain the class of symbolic surface descriptor.

### Status

Many experiments were performed to test the robustness of the classification scheme for biquadratic surface function. Factors in consideration are the size of surface patch, amount of noise and the magnitudes of curvature values. Experimental results show that the classification scheme is not very sensitive to the above factors. For high quality images, it is possible to reliably identify a surface class from the symbolic surface descriptor. For noisy image, the classification is prone to error. Since our recognition strategy does not depend on the presence or absence of features, subjective probabilities for features will be the output of surface classification for hypothesizing the presence of an object.

### Future Research

We are currently investigating the classification of higher order surfaces. Since higher order polynomial functions can model a very complex shape surface, classification becomes more complicated. There is no mathematics literature that studies any possible classifications based on global shape nature. Thus, we are forced to pursue more heuristic approaches. Experiments and learning algorithms are set up to observe the classes of bi-quartic polynomial functions which may manifest themselves from experimental data.

## References

- [1] P.J. Besl and R. Jain, "Segmentation Through Symbolic Surface Descriptions", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp. 77-85, June 1986.
- [2] R. Jain, T. Sripradisvarakul, and N. O'Brien, "Symbolic Surface Descriptors for 3-D Object Recognition", *Proc. SPIE*, Jan. 1987.

## 2.1.2 Dynamic Scene Analysis

Investigator: R. Jain

### Research Objective

This project addresses the problem of recovering depth information in dynamic scenes using a sequence of images acquired using one camera. Stereo information can be obtained using a single translating camera. Depth of moving objects may also be obtained if the camera is stationary and the objects move. We are addressing both cases. In the case of stationary camera, we are developing a qualitative approach. The analysis of a sequence of images obtained using a translating camera configuration may be facilitated by using a mapping similar to the retino-striate mapping in mammals, which we call the Complex Logarithmic Mapping (CLM). The mathematical properties of this mapping make it ideal for object recognition, depth determination, and dynamic scene segmentation.

### Status of Project

There is widespread interest in the Military, Space, and Industrial communities in algorithms for analyzing images from a moving camera. Applications include autonomous vehicle navigation, space station construction and maintenance, and robot arm control for manufacture and inspection. In all of these cases, at least rough ego-motion information is available. Using this information to choose the origin of the mapping allows us to use important characteristics of optical flow without the need to calculate it.

Massone, et al. [6] and others have shown the usefulness of the mapping for object recognition. A comparison of mapping methods [3] has shown that different mapping algorithms give essentially similar results. Jain et al. [2,4,5] have shown that depth information can be obtained using CLM. They have also explored the sensitivity of the method to camera movement and position of the object in the image, showing that depths can be determined more accurately for large camera movements and for objects whose feature points are not near the mapping origin.

Qualitative representations of problem domains attempt to capture the fundamental nature of the system, avoiding those characterizations which are brittle. Much of the work done in qualitative physics involves determining appropriate states and symbols and obtaining an understanding of the nature of state change. An important qualitative reasoning ability is the simulation process, allowing for a grasp on causality. We developed an approach for recovering depth using qualitative information. That approach represents object relationships in a graph structure that is updated as more information is acquired in a dynamic environment [1]. It computes relative depths using very general rules. The depths calculated are qualitative in the sense that the only information obtained is which object is in front of other objects. The motion is qualitative in the sense that the only required motion data is whether objects are

moving toward or away from the camera. Reasoning, which takes into account the temporal character of the data and the scene, is qualitative. This approach to dynamic scene analysis can tolerate imprecise data because in dynamic scenes the data are redundant.

### Future Research

Since the mapping is performed on a digitized image, a direct implementation of the CLM results in an image that contains less information than the original image. Work is being done on interpolation techniques that will maximize the information in the mapped image. Computer vision techniques for static scene analysis rely mainly on point, line, and region techniques. We are exploring how these algorithms perform in CLM space and how they can be used for dynamic scene analysis. The qualitative approach has been tested with synthetic images. Experiments will be performed with movie sequences. Our goal is to develop a robust approach for finding relative depths of moving objects in a dynamic scene.

## References

- [1] S.M. Haynes and R. Jain, "A Qualitative Approach for Recovering Relative Depths in Dynamic Scenes", Tech. Report 11, Cognitive Science and Machine Intelligence Laboratory, University of Michigan, June 1987.
- [2] Jain, R., "Complex Logarithmic Mapping and the Focus of Expansion", *Proc. SIGGRAPH/SIGART Workshop on MOTION: Representation and Perception*, pp. 42-49, Toronto, April 1983.
- [3] Jain, R., S.L. Bartlett, and N. O'Brien, "Some Experiments in Ego-Motion Complex Logarithmic Mapping", to appear in *Advances in Computer Vision and Image Processing, Volume III*, Thomas Huang, ed., JAI Press, Greenwich, 1987.
- [4] Jain, R. and N. O'Brien, "Ego-Motion Complex Logarithmic Mapping" *SPIE Proc. Intelligent Robots and Computer Vision*, vol. 521, pp. 16-23, Nov. 1984.
- [5] Jain, R., S.L. Bartlett, and N. O'Brien, "Motion Stereo Using Ego-Motion Complex Logarithmic Mapping", *IEEE Trans. on Pattern Analysis and Mach. Intell.*, vol. PAMI-9, pp. 356-369, vol. 9, May 1987.
- [6] Massone, L., G. Sandini and V. Tagliasco, "'Form-Invariant' Topological Mapping Strategy for 2D Shape Recognition", *Computer Vision, Graphics, and Image Processing*, vol. 30, pp. 169-188, 1985.



### 2.1.3 Knowledge Based Computer Vision

Investigator: T. Weymouth

#### Problem Description & Research Objectives

The general problem being addressed by this project is the recovery of three-dimensional structure from the images of a scene. The primary thrust of our research in knowledge-based vision is to develop methods of dealing with the uncertainty that arises in the various stages of scene interpretation. Information about the scene is presented as a stream of images; each image contains a noisy sample of the environment. To overcome that noise and to reduce the data, abstractions of the image information are made. In the abstraction process, further errors are introduced, primarily because assumptions must be made for abstractions to be made. We are investigating the flexible use of two types of knowledge to reduce the uncertainty introduced by this abstraction process.

The first type of knowledge is model based descriptions of the environment. When the types of objects that are expected can be classified as to their geometric properties and can be described in a way useful to the data abstraction procedures, then this knowledge can be used to control the creation of appropriate descriptions of the scene. For example, if objects with straight-line edges are expected then the abstraction process for linking edges into straight lines and for finding sharp corners should be applied in the sequence of images arising from the scene.

The second type of knowledge is of objects in the scene. As the description of the scene evolves it should be possible to apply rather specific abstraction processes. In a sequence of images, all from the same scene, following each other by short intervals, objects will not have moved that much. The recognition of object features (by abstraction) in one frame should be used to closely guide the recognition of those same features in subsequent frames.

A flexible control framework is needed so the selection of processes can be made effectively. In addition, during the process of experimental design and development, a flexible framework is desirable to accommodate the frequent changes of design typical of experimental development.

#### Approach

Each of the projects described below fits into a general architectural framework. A computer vision system is assumed to consist of one or more cameras, each directly controlled by the computer, that deliver a stream of images to the system. The images are selectively processed to abstract features. These features may be grouped into further features through several levels of abstraction. Selected abstractions are integrated into a scene description, maintained by the system, that reflects the goal-specific interpretation of the scene.

The selection of features, the grouping of features, and the integration of those groups into a description are controlled, in part by the past description of the scene. This is symbolic feedback. Current projects are concentrating on symbolic feedback as a means of reducing the error and uncertainty inherent in the abstraction processes.

Specifically:

- Geometric models can be used to control the types of edges that are abstracted from the image sequence. Further, as a scene description evolves, the current expectations of object position will lead to a prediction of where edges in the image should occur and exactly what type they will be. This allows limiting the bulk of the image processing to verification of expected feature values. We are experimenting with knowledge-guided interpretation of moving objects of known geometric characteristics.
- The computation of shape from optic flow has been an attractive and elusive goal. Our approach is to use symbolic feedback. In addition to investigating the constraints needed to derive shape from optic flow, we are investigating how predictions of shape can be used to facilitate the computation of optic flow.
- We are also investigating means of computing depth from stereo. Rather than use the pair of images from a single frame, however, we are developing methods for the integration of stereo disparity over several frames into an overall depth map. Assuming that the camera motion is known (e.g. under computer control) it is possible to predict where disparity points from one frame will lie in another. Using methods to reinforce consistent points and inhibit inconsistent ones, over time, we can build up a depth map. This project is new and has been given additional funding by General Dynamics.

To review: the three active projects are the knowledge-based interpretation of geometric forms, the computation of motion parameters from optic flow, and the computation of depth values from dynamic stereo. The knowledge-based interpretation project is the next step in the project for knowledge-based camera control; the computation of motion parameters and the computation of depth values are mechanisms that will be useful in any system for perception.

### Status

The computational framework into which each of these projects is embedded is a black-board architecture. This architecture allows for the flexible development of alternative approaches and the experimental development of program modules. In addition, it allows flexible design of control strategies within each project. We have imported a standard black-board shell (the GBB system from University of Massachusetts) and have begun to adapt it to our use.

The work on the knowledge-based interpretation of geometric forms has feature extraction and grouping routines embedded in the blackboard framework and some preliminary experiments in grouping have been conducted.

The experiments in computation for motion have resulted in the development of a new method for the computation of motion parameters which is now being written up. Further experiments are being performed to understand the limits and robustness of this particular algorithm. We are also in the process of designing a more general system, in which the computation of motion parameters (of a single object) interacts with a motion based segmentation for recognizing which portions of the scene belong to a single object. Our hope is to combine these two methods to produce an system for computing the motion parameters of multiple objects.

In the final project, we are currently examining various methods for computing disparity from stereo images. In addition we are developing a prototype system and testing it on a stereo image sequence.

### **Future Directions**

Although funding within the project for these particular goals has been diminished, research will continue in the following directions:

- Interpretation of moving objects ultimately rests on an understanding of object geometry and geometric relations. We continue to investigate methods for applying geometric constraints.
- Understanding methods for the extraction of motion features is the bases for interpretation of motion images. We are working on algorithms for the extraction of motion parameters from optic flow and the segmentation of optic flow.
- The principle source of information for the reduction of error is the redundancy of nature. We will be working on methods for the integration of predictions (feedback) into the processing of incoming data. This research is in the area of active stereo vision.

### **Industrial Funding**

A grant has been made by General Dynamics for a period of four months to cover initial study and assistance in technology transfer in a study of active stereo vision, as described above. The grant covers basic research in the development of a stereo algorithm based on the concepts of successive approximation and incremental recovery. Additional funding with General Dynamics is being pursued.

## 2.2 Planning and Programming

During the past year a continuation of previous work has been carried out in three areas: (1) Path Planning, (2) Graphic Programming of Robot, and (3) Heuristic Problem Solving. Work in the first two of these areas has been completed and work is being initiated in four new areas:

- Assembly Planning,
- Sensor Planning,
- Tolerancing, and
- Development of an Engineering Database/knowledge Management System.

In the following sections we will describe the status of the work in each of these areas. In the new areas, the emphasis will be on the approach being taken and the general nature of the results expected.

### 2.2.1 Real-Time Collision-Free Path Planning Using Probability

Investigators: K. G. Shin and S. Jun

#### Problem Description and Research Objectives

The goal of this research is to develop a real-time collision-free path planner. Most collision-free path planners [1] [2] [3] [4] focus on the optimization of traveling distance of the manipulator and/or safety of the path. Another important factor to be considered for the design of a path planner is the time required to find the path, i.e., search time. The search time is less critical in database systems where a path is generated *a priori* and accessed frequently. These algorithms are based on the assumption that the workspace, and the origin and destination of every path are completely known and do not change after computing the path.

A slight change in the workspace will require a new path to be generated. Furthermore, there are many circumstances where the origin and destination of a path will not be known beforehand. In such applications, the search time is as much important as the other two factors. Those algorithms developed for such dynamically changing environment [5] are known to suffer from encountering *local maxima* during the search. This is due to the algorithm's inability to differentiate the various sizes, shapes, and orientations of obstacles. In general, *concave* obstacles are more difficult to avoid than *convex* obstacles while bigger obstacles are more difficult to go around than smaller obstacles. These differences in obstacle

types are subjective and almost impossible to classify. Instead of trying to classify obstacles, we have developed a single measure that takes all the relevant factors into consideration.

Most search algorithms use the *select* and *expand* paradigm to reach the goal node from its initial node. When a node is selected, the algorithm checks whether or not it is the goal node. If it is the goal, the algorithm stops and reports the result. Otherwise, it will expand the chosen node to its children. Among the nodes already expanded, the algorithm selects the next node to be expanded. This procedure continues until the goal node is reached. Depending on the search strategy, it may or may not select one of the nodes that have been generated most recently. The search will be most efficient if it only generates the nodes that will lead to the goal, which is usually impossible. Whenever the search selects a node that was not generated most recently, it may have wasted the previous expansion. Sometimes, a selected node may not generate any attractive child and leave the search to choose a previously expanded node. This situation is called a *backtrack*, and the node that forces the search to backtrack as a *dead-end*. The search strategy used in our algorithm is to select a node that will least likely to become a dead-end.

## Approach

We assume that the workspace is divided into  $m \times n$  points. Without loss of generality, we can assume that a robot is shrunk to a point while obstacles are expanded. The following assumptions are made throughout this report:

1. A robot moves from its origin to the destination by advancing one point at a time. The goal of the path planner is to find a sequence of neighboring points from the origin to the destination while minimizing certain costs associated with the path.
2. The origins and destinations are uniformly distributed throughout the workspace.

Following are the symbols and their definitions used in this report:

$V$	The set of all points in the workspace.
$V_{ij}$	A point in the workspace.
$Nei(v_{ij})$	The set of points that are neighbors of $v_{ij}$ ; $\{v_{kl} : \max( i - k ,  j - l ) \leq 1, v_{kl} \neq v_{ij}\}$ .
$V_{obs}$	The set of points that are occupied by obstacles.
$\hat{V}_{ij}$	The set of destinations that will lead $v_{ij}$ to be a dead-end.
$D_{ij}$	The event that search meets a dead-end when path includes $v_{ij}$ .
$R_{ij}^{kl}$	The event that path contains $v_{ij}$ and $v_{kl}$ after $v_{ij}$ .
$p_{ij}$	The probability that $v_{ij}$ becomes a dead-end.
$q_{ij}$	The probability that the search will meet a dead-end when the search path passes through $v_{ij}$ .

Only those points that are adjacent to an obstacle can be dead-ends. Since destinations are assumed to be uniformly distributed throughout the workspace, the probability that a

point becomes a dead-end is:

$$p_{ij} = \frac{|\hat{V}_{ij}|}{|V|}$$

where  $|V|$  is the cardinality of the set  $V$ . Figure 1 shows various examples of obtaining  $\hat{V}_{ij}$ . The probability of meeting a dead-end can be obtained as follows:

$$\begin{aligned} P[D_{ij}] &= P[\cup_{v_{kl} \in V} (R_{ij}^{kl} \cap D_{kl})] \\ &= P[\cup_{v_{kl} \in V} ((\cup_{v_{mn} \in Nei(v_{ij})} R_{ij}^{mn} \cap R_{mn}^{kl}) \cap D_{kl})] \\ &= P[\cup_{v_{mn} \in Nei(v_{ij})} ((\cup_{v_{kl} \in V} R_{ij}^{mn} \cap R_{mn}^{kl}) \cap D_{kl})] \\ &= P[\cup_{v_{mn} \in Nei(v_{ij})} R_{ij}^{mn} \cap D_{mn}]. \end{aligned}$$

Using the above formula, we can obtain the probability of a point meeting a dead-end recursively for the entire workspace.

### Status

The probabilities in the previous section vary as the cost function of a path changes. Figures 2.2 and 2.3 show the probabilities of the entire map of the workspace using the obstacle data given in Figure 2.1 with the cost of a path being the total number of points in the path. Currently, we are implementing these data for the various search methods known so far. Initial experiments have indicated that our path planner yields favorable results over other known planners.

### Future Direction

We have formulated simple measure of the workspace status with various sizes, shapes, and orientations of obstacles using probability. We plan to use these probabilities in various environments with:

- Different cost functions associated with path.
- Different search strategies.
- Multi-robot environment.

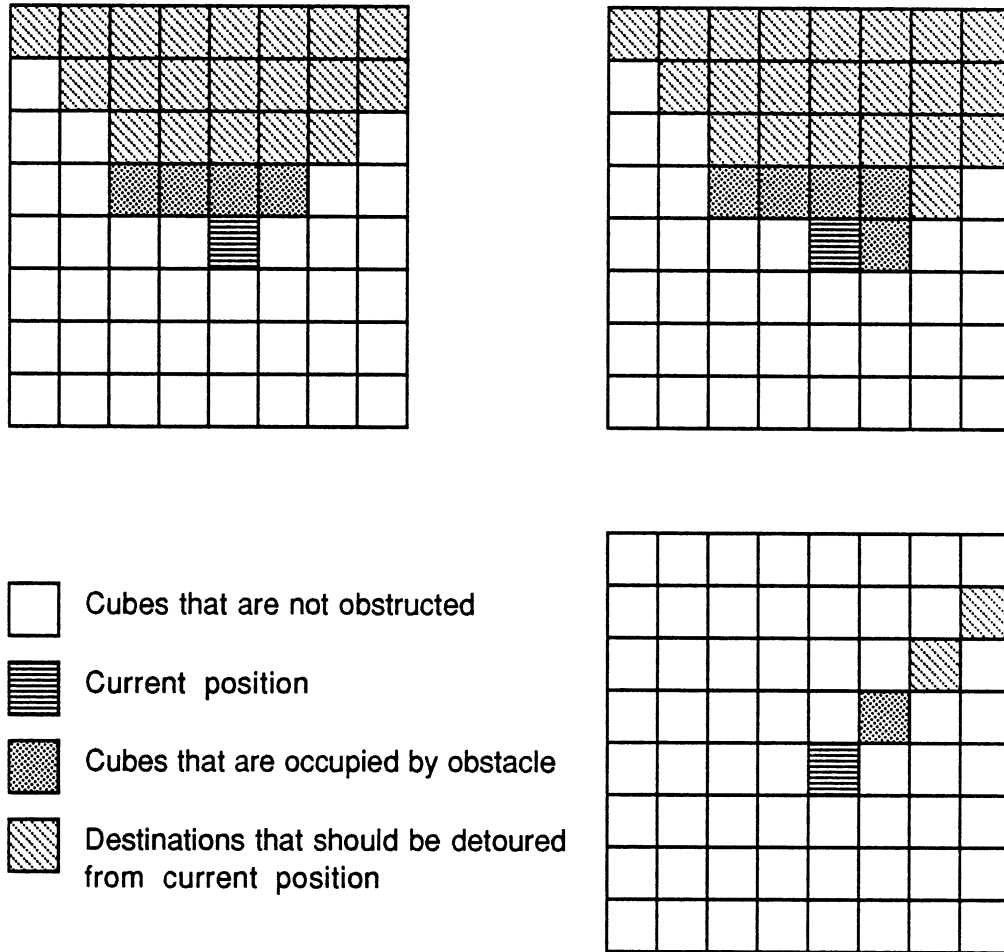


Figure 5: Examples of obtaining  $\hat{V}_{ij}$

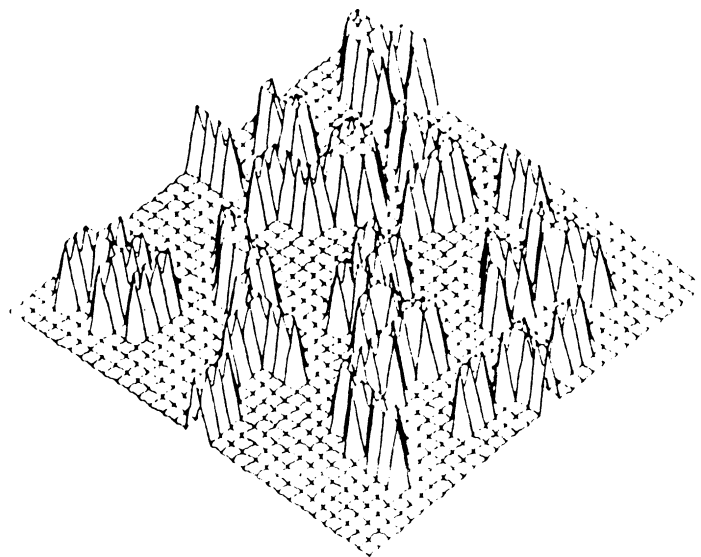


Figure 6: The map of the workspace

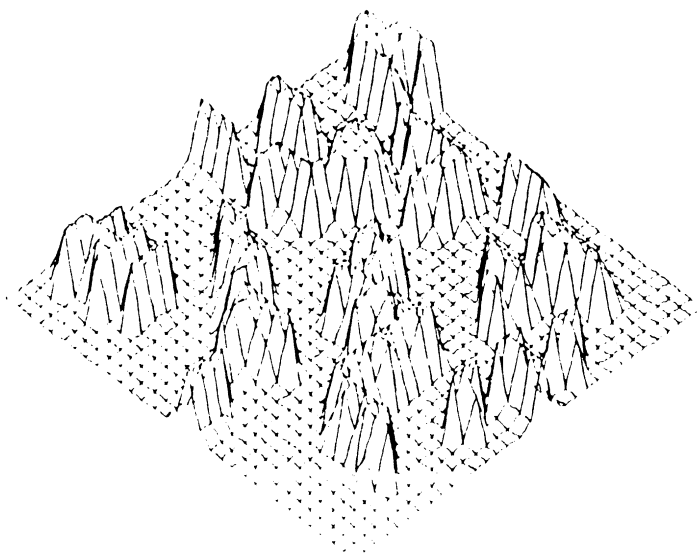


Figure 7:



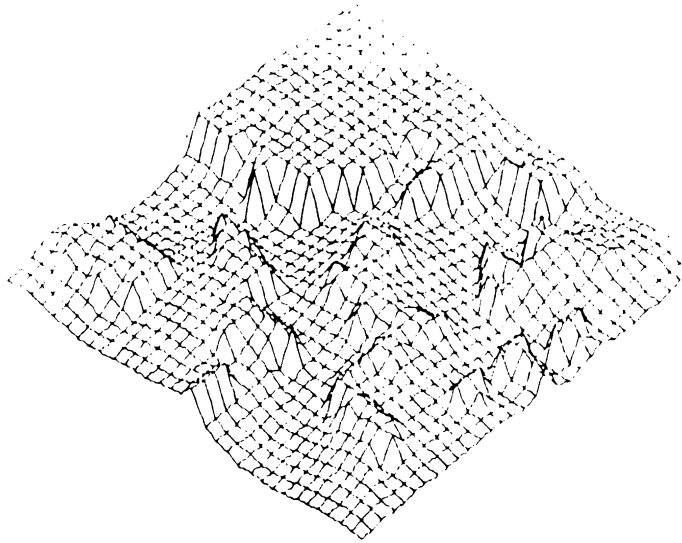


Figure 8:

## References

- [1] T. Lozano-Perez. "Spatial Planning: A Configuration Space Approach." *IEEE Trans. Comp*, February 1983.
- [2] C. O'Dunlaing and C. K. Yap. "The Voronoi Diagram method of motion-planning: I. The case of a disc.," *J. Algorithm*, vol. 6, pp. 104-111, 1985.
- [3] D. Johnson and E. Gilbert. "Minimum Time Robot Path Planning in the Presence of Obstacles." *Proc. of 24th Conf. on Decision and Control*. pp. 1748-1754 December 1985.
- [4] E. Gilbert and D. Johnson. "Distance Functions: Their Application to Robot Path Planning in the Presence of Obstacles," *IEEE Journal Robot. Automat.*, Vol. RA-1, pp. 21-30, March 1985.
- [5] O. Khatib, "Real-time Obstacle Avoidance for Manipulators and Mobile Robots," *IEEE Int. Conf. on Robotics and Automation*, pp. 500-505, March 1985.

## 2.2.2 Automated Robot Task Planning

Investigator: K.B. Irani

### Introduction

One very important but difficult goal of our research on automated robot task planning is to derive good plans efficiently. The past year was devoted to exploring and developing effective means to accomplish this goal.

The focus of our research is on search control which is a key factor in determining both the planning efficiency and efficacy. There are two levels involved in the search control. The higher level is responsible for ordering the problem subgoals and augmentation of problem goals. The lower level is responsible for selection of the next operator to apply at any state in a planning process. Our research is directed to provide a set of effective strategies for both the levels of search control. The eventual goal of our work in this line of research is to establish a formal framework for control knowledge acquisition for heuristic problem solving.

We pursue the above-mentioned goal via two approaches: (1) to develop a set of procedures which can systematically reason using a problem formalism to deduce necessary knowledge for subgoal ordering and goal augmentation, (2) to develop a set of learning mechanisms which can automatically induce or learn planning control knowledge. The knowledge acquired is then incorporated into the general planning methodology to produce effective search control strategies.

In the last year, we have achieved two general high level planning strategies, namely, subgoal ordering and goal augmentation. These strategies have been formally formulated, justified and also tested on several robot task planning problems. The testing results show the performance improvements of the heuristic problem solving brought by the incorporation of these two strategies with the search procedures.

Besides the development of the subgoal ordering and goal augmentation strategies, we have also investigated two learning approaches for acquiring search control knowledge, namely, a strong ( or knowledge-intensive) learning method for the induction of high level search control knowledge, and a weak learning method for the induction of low level search control knowledge. The initial results are very encouraging, because the learning caused significant improvement of performance on solving several problem sets. These results warrant further investigation of the approaches.

### Approaches and Current Status

#### *Subgoal Ordering and Goal Augmentation*

Subgoal ordering and goal augmentation are both high level planning control strategies which transform problem goals into a form which enables tighter heuristic estimation in the best-first heuristic search.

The subgoal ordering approach we propose is based on systematic derivation of subgoal ordering constraints from basic problem specifications. Two algorithms are developed. The first algorithm, SOC (Subgoal Ordering Constraints), can construct a relation over a set of problem subgoals. This relation reasons using problem operator schemes, and reveals an important type of inter-subgoal constraint, namely, the constraint that “one subgoal must precede another in any problem solution”. The second algorithm, called GOAL\_ORDER, takes a set of subgoals and this relation as its input, and assigns each subgoal a rank level indicating how early this subgoal can be achieved relative to the others. Based on the ranking result, the original problem goal is then transformed into a partial order of subgoals. This approach can ensure that the ordering is correct, and therefore, no re-ordering is necessary in the planning process.

The goal augmentation is a means by which an incompletely specified goal can be transformed into a relatively complete goal specification. This augmentation again reasons upon problem operator schemes to discover the missing goal information. The augmentation of goals greatly improves the accuracy of the automatically generated problem search heuristic. An algorithm has been developed to implement this approach and several properties of the approach have been proved.

Currently, both the subgoal ordering and the goal augmentation strategies have been incorporated with the automatic generation of search heuristic, which makes our heuristic problem solving methodology more effective and more efficient.

#### *Learning for Search Control Knowledge*

The subgoal ordering and the goal augmentation strategies we reported above actually work in a similar way – both strategies derive search control knowledge by reasoning using the problem model. However, not all search control knowledge exists in a simple problem formulation. We therefore propose the development of machine learning techniques with the aim of mechanically acquiring search control knowledge not available in a problem model. There are two sources of such knowledge, one is the problem solving experience and another is the additional domain knowledge supplied explicitly in the form of a domain theory. To exploit these two types of knowledge sources, we are adopting two approaches to learning control:

- Development of a weak method for learning control. It is based on monitoring the system’s performance during task execution and requires no special knowledge of the domain. It mainly acquires the knowledge as to which rule to apply out of a set of candidate rules at any intermediate problem solving state.

- Development of a strong method for learning control. It utilizes additional domain knowledge supplied in the form of a domain theory and employs explanation-based generalization. The method acquires the knowledge of useful operator sequences in association with any problem goal.

#### *Weak Learning Method*

The weak method is intended for use when little knowledge of the domain is available. In this method, the history of problem solving in a same category is encoded into “connections” which are established between rules. A connection strength is associated with each connection. The connection strength from Rule-1 to Rule-2 is a measure of the success of firing Rule-2 after Rule-1 during problem solving. It is increased whenever the two appear in the right sequence on a solution path. It is decreased if Rule-2 is fired after Rule-1, but only Rule-1 is on the solution path. Thus a high connection strength between Rule-1 and Rule-2 corresponds roughly to the following heuristic assertion:

Whenever Rule-1 fires, then based on experience, the most promising candidate rule to be fired which can eventually lead to success is Rule-2.

The rule connection strength can be used as a basis for the rule selection at any state during problem solving. Moreover, it can also be used to determine whether or not a composition of several individual rules into a “new rule” is warranted. Only rules with strong connections between them are composed, because a high connection strength is a strong indication of the successfulness of the rules being used in the sequence.

This method has been tested in the Euclidean geometric proof problem domain and encouraging results have been obtained.

#### *Strong Learning Method*

The second learning method we propose, namely, the strong method, is intended for problem domains for which rich domain knowledge is available but is unoperational – not usable in controlling the search directly. The essence of this method is to reformulate this domain knowledge into operational search control strategies through the observation and “explanation” of the solution paths of problem solving instances.

In this method, the learned knowledge is organized into a structure called “schemata” which consists of a goal specification, a precondition list and a sequence of problem subgoals arranged in a best order for achieving the specified goal.

One goal of this method is to selectively learn problem-solving schemata as described above. Whenever multiple operators are to be investigated for a particular goal, it is useful to learn which operator is the right one to be used in a given situation. This would reduce the amount of search needed during subsequent problem solving.

Another goal of this method is to extract meaningful segments out of a learned schema. A schema may consist of a sequence of meaningful segments, and the operator sequence for each segment could vary from situation to situation. If a schema is represented as a sequence of subgoals each of which corresponds to a segment, a different schema need not be added each time the specific operator sequence within a segment changes. Since meaningful segments are likely to be used frequently in other plans, we can take additional advantage of using them in representing other schemata.

This learning method can avoid major problems encountered by previous control knowledge learning strategies, such as the MACROPS learning strategy for STRIPS. These problems are over-specificity, rapid increase of stored macros, the combinatorial explosion, and lack of ability to extract meaningful segments of sequences.

Currently, the first goal of the method has been attained and the second goal has been partly attained. The method has been shown to be effective in our initial test on some robot planning problems.

### **Future Work**

#### *Subgoal Ordering*

1. Develop procedures for detecting other types of inter-subgoal constraints so that the subgoal ordering approach can be made more powerful for the control and reduction of search.
2. Extend the two strategies so that they can work with indirect effects of problem operators, which would greatly improve the applicability of the strategies.

#### *Weak Learning Method*

1. Extend this method to address the issues of interference and lack of goal-directedness. Emphasis on efficiency and low overhead will be carried through in the extension work.
2. Develop a formal model of the method along with some general evaluation criteria to be applied to robot planning domains. The aim is to develop an evaluation scheme to determine the expected usefulness of this method for the robot planning domain.
3. Study the effects of varying the learning parameters used in increasing and decreasing connection strength and the connection strength threshold for rule composition.

#### *Strong Learning Method*

1. Develop an algorithm for assimilating newly learned schemata into existing ones.

2. Develop a method for learning top level problem solving plans in the form of sequences of subgoals by extending the schemata to cover the essential aspects of the whole operator sequences given by a problem solver.
3. Develop a method for refining learned plans in case of failure during subsequent problem solving.

### 2.2.3 Automatic Assembly Planning

Investigators: J. Wolter, A.C. Woo, R.A. Volz

#### Problem Definition

During the last three decades considerable advances have been made in the development of artificial intelligence systems to generate plans to construct assemblies. These robot-planning systems have typically solved problems in which they are confronted with workspace containing several parts which must be rearranged to achieve a given goal position in a minimum number of steps.

While methods of solving this type of problem are of great interest, they are not generally very applicable to manufacturing problems. When designing an assembly line or workcell for the production of a new product, the engineer normally uses whatever fixtures and feeders are appropriate to the problem. Thus, the workspace geometry and the initial part placement will be outputs from the planning process rather than inputs.

To address this type of assembly problem, we are working on the design of a planner which takes as its input a geometric model of the assembly to be built, and produce the following: (1) a high-level description of the fixtures that will be needed. (2) for each fixture, a sequence of parts or subassemblies to be inserted. (3) for each part insertion, a trajectory that intersects no previously inserted parts. The fixture specification would consist of a list of parts that the fixture must hold, and, for each, a description of the forces against which the part must be held. Each subassembly will be built in a fixture, and then used as a part in either another subassembly, or the final subassembly.

For such a planner to be in practical, careful consideration must be given to the criteria used to select plans. The criterion used in traditional robot planning is "minimum number of operations," but in assembly planning there are normally a very large number of plans with the same number of operations. Furthermore this fails to capture such important considerations as the difficulty of implementing the specified fixture and performing the specified operations.

In this research project, we have been working to develop a practical approach to the automatic solution of assembly-planning problems with meaningful criteria.

## Approach

Traditional robot planning systems usually pay very little attention to the problem of selecting trajectories with which to insert parts. Normally they are simply inserted from above. In assembly planning there is no predefined orientation in which the assembly will be built, thus "above" is not a meaningful concept. A more thorough treatment of trajectories is needed.

Until the plan is complete, we have no way of knowing which parts may already be in place when we try to insert a part. Rather than attempting to run a path planning algorithm for every possible combination of obstacles, we solve the problem in two stages. (1) Propose a list of likely insertion trajectories for each part. (2) Generate a plan using one trajectory selected from each part's list.

Insertion trajectories will be proposed by recognizing certain common structures in assemblies. For example, when a series of parts is found on a shaft, trajectories parallel to the shaft will be proposed. When two parts are threaded together, a spiraling trajectory parallel to the screw shaft will be proposed. Vectors normal to contact surfaces are also likely candidates.

Once we have proposed an insertion trajectory for a part, we test it to see which other parts would block it if they were already in their final positions. The results of these tests for the assembly shown in Figure 1 are diagrammed in Figure 2. Here each part is represented by a large circle, and each trajectory which has been proposed for that part is represented by a small circle. For plate C there are two trajectories. It may be inserted either from above (assuming the orientation shown in figure 1) or from below. If we choose to use the trajectory from above, then both screw A and screw B must be inserted after C, so as not to block that trajectory. In figure 2, this is represented by the arrows pointing from the down trajectory of node C to nodes A and B.

This type of diagram fully represents all geometric constraints in the assembly problem. It is the basic data structure to be used by our planner. Three operations can be used to develop this problem representation into an assembly plan: (1) Discard a possible trajectory from a part having more than one trajectory. (2) Add a constraint arrow from one part trajectory to another part. (3) Form two or more parts into a subassembly, as in figure 3, where A and C are broken off as a subassembly. If the result is a diagram in which each part has only one trajectory and all the parts in each subassembly are constrained into a strict linear sequence, then it is a valid assembly plan.

Several criteria have been developed to guide the application of these three operations to the constraint diagram. Our research will initially focus on the following three: (1) Monodirectionality – As much as possible parts within a subassembly should be inserted from the same general direction. This tends to reduce both the complexity of the fixture and the number of degrees of freedom needed for the robot. (2) Manipu-

lability – The more complex operations should be done with the smaller parts. That is, when screwing a bolt onto an automobile, we turn the bolt, not the automobile. (3) Fixture Complexity – The number of degrees of fixture which the fixture must hold things against should be minimized. It is easier hold a shaft and insert a series of parts onto it than it is to hold all the parts in line, and insert the shaft through them.

Several other criteria may be considered in greater detail later. These include, among others: (1) Uniformity – If there are similar parts or sets of parts, they should be built in similar ways. This reduces the number of different operations the robots must be able to preform. In some cases, it may be able to use the same fixture for several subassemblies. (2) Parallelism – When there are multiple robots, it may be advantageous if several operations can be preformed at once.

Clearly the relative importance of these different criteria would be different in different applications. In a single-robot workcell monodirectionality is more important than parallelism. On an assembly line, the contrary may be true. The relative weights of criteria would thus be supplied by the operator.

### Status

In the last year, our original approach, as described in the previous report, has been completely revised. The method described above is much more general in that it is easily extended to many types of assembly problems, whereas our original method was confined many to assemblies which were screwed together.

The constraint diagrams provide a general, unified representation of both the assembly problems and partial or complete solutions to such problems. A set of operations have been developed by which any possible solution can be found. The operations can be applied in any order, so that not only conventional backward and forward search strategies, but also opportunistic search strategies can be applied. In opportunistic strategies we make the decisions which seem most critical first, instead of always starting with the first or last operation.

The criteria have been formalized and study has begun on how to apply them to making planning decisions. Particular emphasis is being given to the problem of deciding when to form subassemblies. This appears to be much more challenging than simply making decisions about the sequencing of parts within an assembly.

### Future

During the next year, we expect to implement a test system to solve assembly problems using the new approach described here. To achieve this goal, a more detailed understanding of the criteria must be developed so that a planner can be designed which will be able to effectively chose good directions in which to search for a plan.



## 2.2.4 Replanning

Investigator: R.A. Volz, J. Xiao

### Problem Description and Research Objectives

The manual development of robot programs for assembly is a laborious, error prone task. It would be highly desirable to be able to derive them automatically from design information. In order to do so, several important problems must be solved: 1) automatic determination of an assembly plan (sequence of assembly operations), 2) automatic determination of parts presentation and fixturing, 3) gross motion planning, 4) fine motion planning, and 5) automatic generation of error recovery routines[7]. This report addresses the last of these problems.

It has often been stated that the largest part of any robot assembly program is made up of fix ups to handle things that don't quite work as planned. Algorithms for generating these fix ups are crucial to successful automatic program generation. Problems in the execution of nominal robot programs arise for the following reasons:

- *mechanical and control errors*: robots and all other mechanical devices are only accurate to within certain bounds;
- *sensor errors*: all sensors have limited sensitivity and accuracy;
- *manufacturing tolerances*: different instances of the same model are not exactly identical because of manufacturing limitations.

Often such errors can lead to the failure of a nominal program that would theoretically accomplish a task such as part mating. The automatic generation of robot programs for assembly cannot be successful until some means of either avoiding or correcting these errors is found.

Several approaches towards handling these uncertainties have appeared in the literature. Taylor[11] introduced an error-propagation method to estimate compound errors from the uncertainty bound or tolerance of each individual part involved in a task (including the robot hand). Brooks[1] extended Taylor's method by making the process backward, so that the constraints on some compound errors to make the task succeed can lead to constraints on individual parts. This method, however, suffers from high computational complexity. And it does not provide means to reduce errors dynamically. The inductive learning approach by Dufay and Latombe[3] corrects run-time errors by adding rules into the system as a corrective plan. In this approach, error-handling is not fully automatic, since rules must be provided by human users.

The pre-image approach introduced by Lozano-Pérez, Mason, and Taylor[6], and simplified by Erdmann [4] incorporates the effect of uncertainty directly within one planning phase. The goal is to create motion plans that will avoid errors. However, this

approach has unsolved theoretical problems and its applicability to practical cases is questionable. A more practical force control method was developed by Whitney[12]. His remote center compliance device can correct small insertion errors for a peg-in-hole task by applying correct forces to the peg. But it only works when the peg is in or partly in the hole.

All the previous approaches contribute to solve the problem in some ways, but none fully solves it. The problem is so complex that new solutions are still in need.

### Approach

The approach we present is based upon three hypotheses:

1. The problem cannot be solved in general. Design constraints relating nominal parameters, tolerances and error parameters must be satisfied if success is to be guaranteed.
2. A two phase planning process, an off-line nominal planner and an on-line replanner to correct for run-time errors, simplifies the overall system.
3. Replanning can be based upon knowledge of which surfaces of the parts involved are in contact with which other surfaces.

In our previous report, we have introduced the notion of *contact formations* [2] to describe different topological contacts among the parts being assembled, and has described a technique for, under certain design constraints, uniquely determining the contact formation in which the parts lie in spite of sensor and control errors. When an error occurs in a nominal plan, an unintended contact occurs among surface elements of the parts involved. While there can, in general, be a large number of contact formations, there are typically only a modest number that are achievable with small deviations from the nominal plan. Thus, one can consider devising replanning strategies based upon the contact formation in which the system completes an attempted move (we presume that there are force/moment guards that stop motions that make unintended contacts). Further, since the actual positions may be presumed to be close to the nominal trajectory, simple replanning strategies, such as straight-line sliding motions or single axis rotations may be considered without having to worry about obstacle avoidance. Thus, our approach is to adopt such simple replanning strategies, and by exploring the relationships between tolerances for successfully accomplishing a task and bounds on uncertainties, to set up appropriate design and motion constraints, which if satisfied, can be used to *guarantee* the success of the replanning strategy.

We restrict the replanning to be defined on *motion in contact* only[5], for errors are more crucial and more easily to be identified when a collision (i.e. a contact) occurs. Further, the desired goal of most part-mating operations involves contacts among

the parts involved. The underlying assumption for that restriction is that compliant motions[8][9][10] are applied wherever necessary.

The process of realizing our approach is as follows:

- establish an appropriate system error model (which include the geometric, sensory, mechanical and control errors),
- develop a replanning motion strategy,
- obtain appropriate constraints on the error model which reflect requirements to the design of the assembly and the sensors in order to guarantee the success of the replanning, i.e. design constraints,
- obtain appropriate constraints on the replanning motion (under which the success of the replanning can be guaranteed), i.e. motion constraints,
- implement the replanning motion strategy under all design and motion constraints to handle uncertainties.

### Status

We have applied our approach to peg-in-hole tasks by introducing a simple replanning strategy, and developing proper design and motion constraints necessary to the success of the strategy.

For a peg-in-hole task, it is most possible that the peg is not put properly above the hole before the insertion, so that the peg hits the entry surface of the hole and the task fails. Although the phenomenon is most commonly studied in the area of uncertainty handling, previous research either does not guarantee the success of the task or is too complex to be practical. Thus, the replanning that is of interest starts from that point. The objective is to move the peg above the surface, with error within the tolerance, so that the insertion can be done.

Some restrictions or assumptions of the study are stated below:

- the replanning motion is compliant, and the Compliant surface (C-surface) is the entry surface of the hole;
- the entry surface and the bottom surface of the peg which contacts the entry surface are planar;
- the bottom surface of the peg fully contacts the entry surface of the hole, so that the task space has only two translational and one rotational degrees of freedom(Fig. 9);
- there are no other objects (i.e. obstacles) except for the peg and the hole.

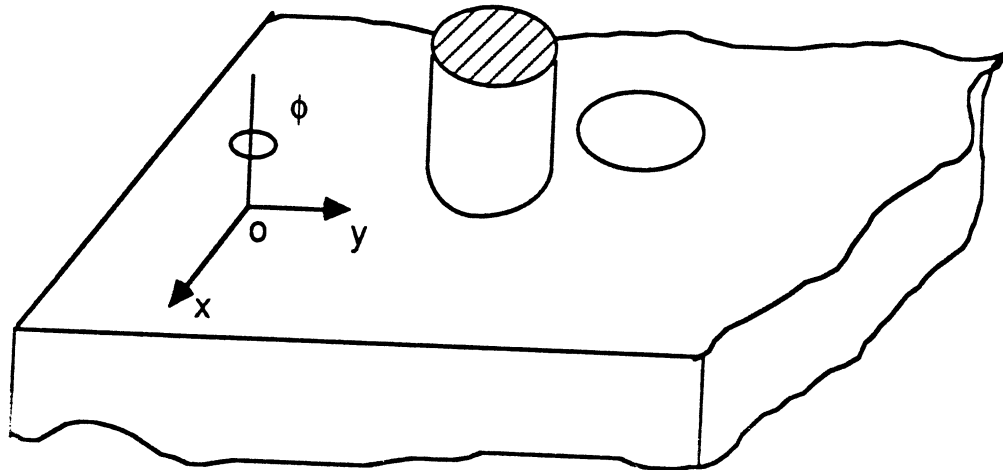


Figure 9: Peg-in-hole task space

### *Error Model*

We suppose that only position/orientation and force/torque sensors are available in the system. Thus, three kinds of errors have been modelled: position /orientation sensor uncertainty  $\epsilon_p$  and  $\epsilon_o$ , force/moment sensor uncertainty  $\epsilon_f$  and  $\epsilon_m$ , and robot motion uncertainty  $\epsilon_v$ ,  $\epsilon_\omega$ ,  $v_r$ , and  $\omega_t$ [13].

### *Replanning Strategy*

Let  $(\mathbf{P}_a^p, \phi_a^p)$  and  $(\mathbf{P}_a^h, \phi_a^h)$  be the locations of the peg and the hole respectively. Let  $\mathbf{d}_a = \mathbf{P}_a^h - \mathbf{P}_a^p$ , with  $d_a = \|\mathbf{d}_a\|$  be the actual distance. Let  $\phi_a = \phi_a^h - \phi_a^p$ .

Ideally, since it is assumed that there is no obstacle around, the motion for replanning can be simply a (compliant) translation of the peg in the direction of  $\mathbf{d}_a$  with moving length  $l = d_a$ , and then a rotation of the peg to make sure that a proper  $\phi_a^p$  is obtained for the peg to enter the hole (Fig. 10). However in a real case, because

- only a *sensed*  $\mathbf{d}_s$  for  $\mathbf{d}_a$  and a sensed  $\phi_s$  for  $\phi_a$  can be possibly obtained,
- $\mathbf{d}_s$  and  $\phi_s$  are different from  $\mathbf{d}_a$  and  $\phi_a$  respectively due to sensory errors,
- there are also motion errors,

that ideal strategy with only *one* translation and *one* rotation can not guarantee the success of replanning. A modified, more realistic strategy must be set up.

In essence, the proposed strategy consists of more than one motion step. Each *desired* motion step is formed by a (compliant) translation along the direction of the sensed  $\mathbf{d}_s$ , and a rotation to correct the orientation error caused by the translation (if necessary). The distance  $l$  of the translation is constrained with respect to both the sensor error in

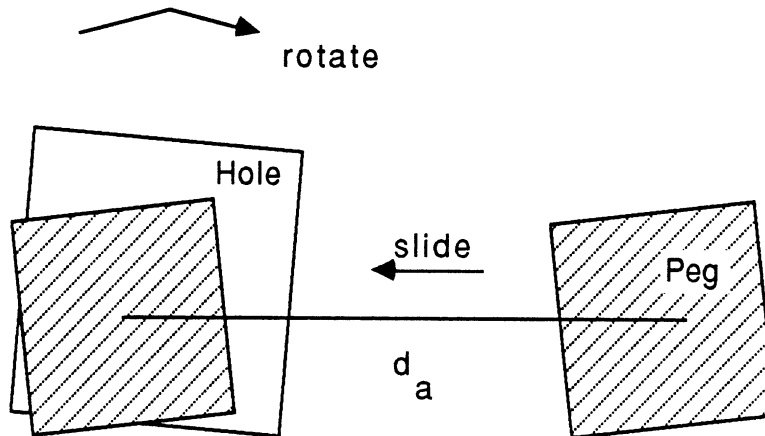


Figure 10: Simple replanning strategy in ideal case

$d_a$ , and the motion error in actual moving, so that after each motion step,  $d_a$  always becomes smaller. In this way, the positional replanning can be convergent. After the success of the positional replanning (i.e. after  $d_a$  is less than the positional tolerance of the task), the peg is rotated to correct  $\phi_a$ . Note that since the motion error contributing to  $\phi_a$  is corrected in each motion step,  $\phi_a \leq \epsilon_o$ , where  $\epsilon_o$  is the uncertainty bound in orientation sensing. If a rotation step can correct  $\phi_a$ , while not destroying the success in position, the success of the whole replanning can be guaranteed. For realizing the strategy, developing proper design constraints and motion constraints (on  $l$  of each motion step) is fundamental.

#### *Design and Motion Constraints*

We have derived design and motion constraints for successful replanning in two cases:

1. using position/orientation sensing only,
2. using force/torque sensing to compensate position/orientation sensing.

The first case is relatively easier. For convenience, yet without losing generality, we restrict our study to those peg-and-holes with *regular polygon* bases.

Let  $r_h$  be the *nominal* radius of the inner tangent circle of a hole; and  $r_h^a$  be an actual radius. Let  $r_p$  be the *nominal* radius of the inner tangent circle of the corresponding peg, and  $r_p^a$  be an actual radius. Defining  $\delta_h$  as the *tolerance* of the hole, and  $\delta_p$  as the *tolerance* of the peg, we require  $r_h^a$  and  $r_p^a$  to satisfy  $\|r_h - r_h^a\| \leq \delta_h$  and  $\|r_p - r_p^a\| \leq \delta_p$  respectively. Then, we defined  $\delta = r_h - r_p - \delta_h - \delta_p$  as the tolerance of the task. From the position sensor uncertainty  $\epsilon_p$ , we derived a *position sensing uncertain area*  $U_p$ ,

which is a circular area about  $P_a^h$  with radius  $d_p$ , such that when  $P_a^p$  is inside  $U_p$ , position sensor can not determine the next replanning motion[13]. Clearly, we needed  $d_p \leq \delta$  as a design constraint to guarantee the success of the replanning *with position sensor only*. In general, all design constraints are derived as inequality relations among the uncertainty parameters, the task tolerance  $\delta$ , the nominal linear speed  $v_d$ , and the nominal angular speed  $\omega_d$  of the end-effector, while the motion constraint is expressed as a bound put on the distance  $l$  of each translation step[13].

The fact that a force/torque sensor can possibly sense a moment when the peg is partially overlapping the hole suggests that even if the peg reaches the position uncertain area, the replanning process may not have to be terminated, since an ineffective position sensing might be compensated by an effective moment sensing to continue guiding the replanning. Believing in this, we studied the second case. Since it is very difficult to obtain constraints for this case *generally* under which the replanning strategy can always be successful, we have studied a special group: cylindrical peg-in-hole (CPH) tasks[13].

Similar to the first case, we defined the tolerance of the task  $\delta$ , where the only difference is that  $r_h$  and  $r_p$  here are the radii of the hole and the peg respectively. We then derived a *moment-sensible area*  $R_m$ , as a circular area about  $P_a^h$  with radius  $d_m$ , such that when  $P_a^p$  is inside  $R_m$ , the moment sensor can determine the next replanning motion[13]. Clearly, instead of the constraint  $d_p \leq \delta$  of the first case, now the constraint becomes  $d_p \leq d_m$ , under which the moment sensing can compensate the position sensing to guarantee the success of the replanning. Since the task tolerance  $\delta$  is usually much smaller than  $d_m$ , the constraint  $d_p \leq d_m$  is thus much weaker than  $d_p \leq \delta$  of the first case. This shows that with moment sensing to compensate position sensing, the robot can perform a peg-in-hole task with much tighter tolerance than it can do with position sensing only.

### *Simulation Results*

To test the constraints and the replanning strategy, we designed computer programs to simulate the replanning process for both the regular-polygon-based peg-in-hole (RPH) cases and the CPH cases. The computer simulation we have done shows that the design constraints obtained are correct and practically reasonable, and that under the design constraints, the motion constraint is *necessary* for guaranteeing the success and efficiency of the strategy. In particular, the simulation results show empirically that the theoretical constraints can be relaxed somewhat with excellent results still obtained[13].

### **Future Direction**

Our next step is to establish a testbed on the puma 560 robot, to further explore the applicability of the strategy.

## References

- [1] R. A. Brooks, 1982. Symbolic error analysis and robot planning. *Int. J. Robotics Res.* 1(4):29.
- [2] R. S. Desai, and R. A. Volz, 1986(Sept.). Contact formations — a new approach to local motion in mechanical assembly. Robot Systems Division, Center for Research on Integrated Manufacturing, University of Michigan.
- [3] B. Dufay, and J. C. Latombe, 1983(Aug.). An approach to automatic robot programming based on inductive learning.
- [4] M. Erdmann, 1986. Using backprojections for fine motion planning with uncertainty. *Int. J. Robotics Res.* 5(1):19.
- [5] J. Hopcroft, and G. Wilfong, 1986. Motion of objects in contact *Int. J. Robotics Res.* 4(4):32.
- [6] T. Lozano-Pérez, M. T. Mason, and R. H. Taylor, 1984. Automatic synthesis of fine-motion strategies for robot. *Int. J. Robotics Res.* 3(1 ):3.
- [7] T. Lozano-Pérez, 1983. Task planning. *Robot Motion: Planning and Control*, M.Brady et al. Eds.
- [8] M. T. Mason, 1981. Compliance and force control for computer controlled manipulators. *Robot Motion: Planning and Control*, M.Brady et al. Eds.
- [9] M. T. Mason, 1982. Compliant motion *Robot Motion: Planning and Control*, M.Brady et al. Eds.
- [10] M. H. Raibert, and J. J. Graig, 1981. Hybrid position/force control of manipulators. *Robot Motion: Planning and Control*, M.Brady et al. Eds.
- [11] R. H. Taylor, 1976. The synthesis of manipulator control programs from task-level specifications. *AIM-282*. Stanford Univ.
- [12] D. E. Whitney, 1977. Force feedback control of manipulator fine motions *J. Dynamics Sys. Measurement, and Control.* 98:91.
- [13] J. Xiao, and R. A. Volz, 1987. Design and motion constraints of part-mating planning in the presence of uncertainty handling. Robot Systems Division, Center for Research on Integrated Manufacturing, The University of Michigan.

### 2.2.5 Automatic Determination of Retrieving Paths in a Semantic Data Model

Investigators: Y.C. Lee, M.A. Doctor, C.H. Cho

#### Problem Descriptions and Research Objectives

In a manufacturing environment it is very likely that a great deal of useful information will be stored in many existing databases which have different structures and/or use different query languages. It is also possible that most of these databases are still being retrieved and updated by their own users. On the other hand, as information management techniques have progressed, more new knowledge and existing experience has been organized in the form of rules and high level operators that are quite different from the traditional formatted data types.

Semantic models for databases [KiMc85,Su86] that result in more meaningful databases and provide meta-knowledge about the data seem to be good candidates for the integration of various systems. In fact, in the PDES (Product Data Exchange Specification) documents recently released, any conceptual schema proposed for the ongoing PDES project is required to be an integrated semantic data model represented using the IDEF1X modeling technique developed by the U. S. Air Force. However, methods to exploit this increased information so that the task of the user is simplified have been few in comparison to the number of semantic models in recent years. Accordingly, while being highly expressive, semantic data models also present the user a much more complicated view of data to deal with.

Assuming that there is a semantic data model for project management as indicated in the figure below. Each project (PJ) can be either a development project (DP) or a scientific project (SP). A DP is always led by a senior engineer (SE) while a DP must be led by a research fellow (RF), who is also an engineer (ENGR). An SP can be joined (JOIN) by any kind of engineer, including the third type, technical staff (TS). A DP is however joined by technical staff only. To further complicate the data model, the name as well as the office information of an engineer is kept with the employee (EMPL) whereas the secretary is the other type of employee.

In order to be more expressive, more evolvable, and more disciplined, the semantic data model uses a number of high-level constructs to capture the semantics. Nodes in the model are classified into four types: *interaction*, *generalization*, *aggregation*, and *member*. Each *interaction* node combines two participating nodes to represent the relationship between them. Each *generalization* node combines a number of individual nodes, each being a special subtype of the generalization node. Each *aggregation* node groups together several nodes and associates them as properties with a node. Each *member* node stands for a property which can be printed or compared to a constant value.



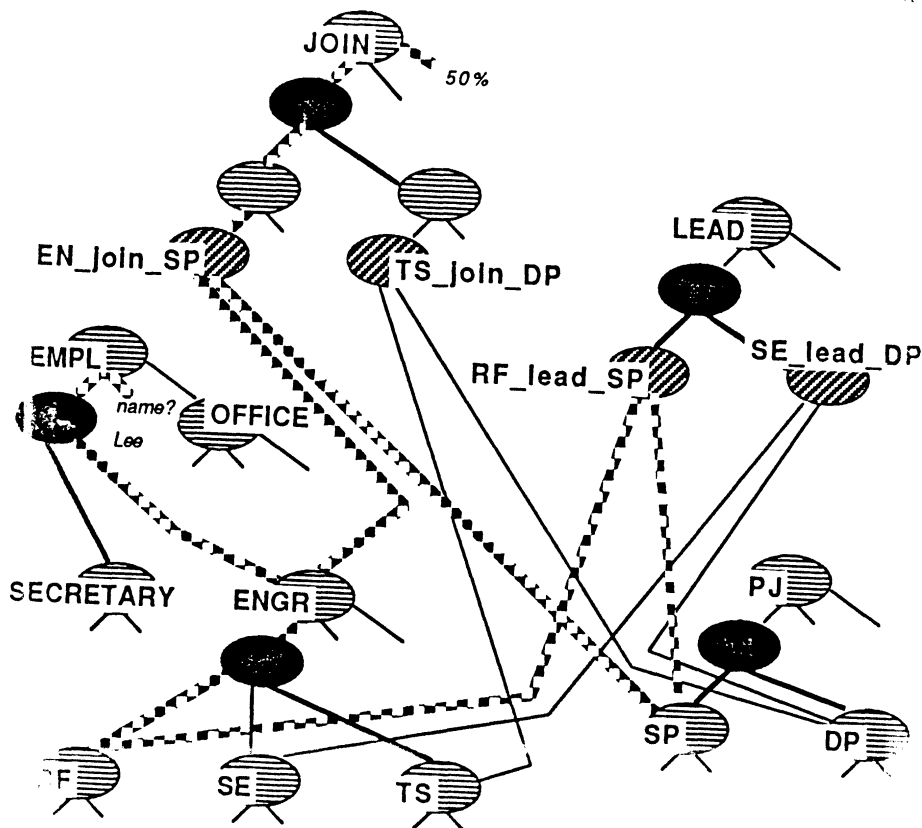


Figure 11: An Example of Semantic Data Model

As more semantic nodes are defined, the model requires more relations in the relation scheme through which a user specifies queries. Consider the query “Find the name for all engineers who join the project led by an employee Lee with at least 50% appointment.” As far as the user is concerned, a simple formulation such as the one below would be just natural and adequate for this query.

```

FIND      ENGR.name
WHERE     JOIN(ENGR,PJ).pct_of_work >= 50%
AND       LEAD(EMPL,PJ)
AND       EMPL.name = "Lee"
    
```

Unfortunately, existing relational query languages are not intelligent enough to support such a formulation. Instead, a much more complicated formulation such as the SQL statement below would be required.

```

FIND      ENGR.name
FROM      SP, DP, RF, SE, TS, LEAD, JOIN,
          EMPL EMPL1 EMPL2, ENGR ENGR1 ENGR2,
          RF_lead_SP, SE_lead_DP, EN_join_SP, TS_join_DP
    
```

```

WHERE      (      EMPL1.# = ENGR1.#
            AND    ENGR1.# = EN_join_SP.engr
            AND    EN_join_SP.# = JOIN.#
            AND    JOIN.pct_of_work >= 50%
            AND    EN_join_SP.sp = SP.#
            AND    SP.# = RF_lead_SP.sp
            AND    RF_lead_SP.rf = RF.#
            AND    RF.# = ENGR2.#
            AND    ENGR2.# = EMPL2.#
            AND    EMPL2.name = "Lee"      )
OR         (      ...
            /* similar statements regarding DP
            ... )

```

Note carefully that, in formulating the above statement, the user is in fact navigating the model to identify all relations that are involved and the associations between them. Although most query languages are nonprocedural (i.e., it is not required to specify *how* the data would be retrieved), it is still needed for the user to inform the system of *where* the data resides and *what* interrelations exist between data items. As the number of relations increases, query specifications become lengthy and complicated, as indicated by this example.

The objective of this research is therefore to equip the semantic data model with inferential capabilities so that the requirement of *where* specification can be eliminated and the *what* specifications be simplified. We believe that, with the meta-knowledge available through data semantics, an intelligent query processor would be able to evaluate queries that are in a simplified form such as the first one suggested above.

### Approach

In general, each algebraic operation performed on the database yields some derived data which is used in another operation with some other data set from the database to obtain data which is in a sense closer to the data required by the query. Thus, from the starting point till the final stage when the answer is available, a number of relations need to be examined in a specific order and manipulated. The collection of relations can be referred to as a path. The process of automatically finding the path and answering the query is called the path determination problem.

Our first approach tries to find the complete path, i.e., the solution tree, from the graph which represents the semantic data model[LeDo87]. It turns out that the requirement of the solution tree residing completely in the graph is very time consuming and some portions of the path can actually be bypassed by some logical paths. In order to have a

concrete domain of possible query types so that possible improvements can be made, we have looked into queries that are basically in the form of SQL statements[Cham76]. The study has led to a significant revision from the original approach.

### Status of Research Effort

During the past year, the study of possible query types has resulted in the development of an enhanced but simple query language named ESQL. The design of the syntax is mainly based on the notion of the expressions *object.property* and *event(object,object).property*. The object can be a node of any type except the member type. The event can be an interaction node or a generalization node with interaction nodes as subtypes. The property can be the name of any node that is a descendant of the object node. This notion, if supported, would relieve the user of the burden in identifying necessary associations within each generalization hierarchy.

We have developed most of the algorithms required to support this notion. Key components are the procedures for backchaining and forward chaining which locate all the needed physical paths. Logical paths that link all physical paths to form a query tree are generated by the ESQL compiler. In the following, we will use the example described earlier to introduce the basic ideas of backchaining and forward chaining. Details of the language syntax and the chaining procedures are available in [LeDA88]

In implementing the semantic data model by relations, each node except those of member type, is augmented with a column of surrogates. A *surrogate* is a permanent system assigned pseudo-attribute that functions as a key for the node. The added advantage of having surrogates for each node is that the inclusion of the node into its parent node can be done by including the surrogate as an attribute in the parent node with the attribute name being the name of the node.

To evaluate the query which is specified as

```

FIND      ENGR.name
WHERE     JOIN(ENGR,PJ).pct_of_work >= 50%
AND       LEAD(EMPL,PJ)
AND       EMPL.name = "Lee"

```

the query processor first identifies all the expressions involved in the *WHERE* clause. They are *JOIN(ENGR,PJ).pct\_of\_work*, *LEAD(EMPL,PJ)*, and *EMPL.name*. Each expression will be evaluated to generate a table (i.e., an intermediate relation) with attributes that appear in the expression. If all the tables can be generated, they will be combined by operations such as join and the answer *ENGR.name* will be found by a similar procedure. So the problem is how these tables can be generated.

To evaluate each expression, the process must identify the path between the object node and the property node. In general, the backchaining procedure will be used to

link nodes following backward pointers. However, in more complicated cases where an event is specified, e.g., *JOIN(ENGR,PJ).pct\_of\_work*, paths that link all four nodes must be identified and the procedure of forward chaining is needed.

To explain the mixed use of backchaining and forward chaining, let us consider the expression *JOIN(ENGR,PJ).pct\_of\_work*. A backchaining would immediately link the nodes *JOIN* and *pct\_of\_work*, and a table with two columns, *JOIN*'s surrogates and *pct\_of\_work*, can be generated by a projection of the relation corresponding to *JOIN*. If the property node *pct\_of\_work* were not a direct subnode of *JOIN* but instead a node which shared with *JOIN* a lowest common ancestor which is not *JOIN* itself, the backchaining would have followed the backward pointer from it up until the lowest common ancestor where the chain meets the chain from *JOIN*.

To link the event *JOIN* and the two participating objects, *ENGR* and *PJ*, a forward chaining must be performed first. As *JOIN* is not an interaction node but instead a generalization node of subtypes *EN\_join\_SP* and *TS\_join\_DP*, a forward chaining from *JOIN* will generate a path to *EN\_join\_SP* and another one to *TS\_join\_DP*. It thus breaks the expression *JOIN(ENGR,PJ)* into two expressions, *EN\_join\_SP(ENGR,PJ)* and *TS\_join\_DP(ENGR,PJ)*, whose corresponding tables will later be unioned.

The backchaining for the three nodes, *EN\_join\_SP*, *ENGR*, and *PJ*, in the expression *EN\_join\_SP(ENGR,PJ)* requires an additional treatment for the interaction node *EN\_join\_SP*. While all nodes are backchained, the *EN\_join\_SP* is also forward chained, but only to reach its two direct subnodes. The two direct subnodes will then be backchained as normal. Without this one-step of forward chaining from an interaction node, the path will never be constructed.

### Future Work

While it is the complexity of the semantic data model that discourages casual users from using it, it is also the structural information embedded in the semantic model that enables the system to reason. Aiming at providing a more intelligent and easy to use query language for semantic databases, this research has addressed the problem of automatic path determination. The approach used is a combination of backchaining and forward chaining of semantic nodes. This work has resulted in an enhanced but simple query language through which query specifications are much more simplified. During the past year, the syntax of the query language and the algorithms for path determination have been designed. We are currently developing the prototype language.

Meanwhile, we are applying the semantic data modeling techniques to engineering applications. With the prototype query language as well as a number of application models, we will then be able to evaluate the total success of the approach proposed in this research.

An expressive semantic data model with inferential capabilities is clearly a desirable tool for the integration of various databases since a new user will be able to access all databases through the semantic association model using simple and unified queries while the access methods for each individual database remain unchanged. The capabilities of such a system, however, should not be so limited. It would be great if the inferential power could be enhanced to deal with not only conventional data types but also geometric data types, rules, high level operators and so on.

## References

- [Cham76] D.D. Chamberlin *et. al.*, "SEQUEL 2: A Unified Approach to Data Definition, Manipulation, and Control," *IBM J. Research and Development*, Nov. 1976, pp.560-575
- [KiMc85] King, R. and McLeod, D. J., "Semantic data models," in *Principles of database design, Vol. 1, Logical organizations*, ed. by S. B. Yao, Prentice-Hall 1985
- [LeDo87] Lee, Y. C. and Doctor M., "Automatic Determination of Retrieving Paths in A Semantic Data Model," Proc. IEEE Office Automation Symposium, Gaithersburg, MD, April 27-29, 1987, pp89-95
- [LeDA88] Lee, Y. C., Doctor M., and Armstrong E., "ESQL: An Enhanced but Simple Query Language for A Semantic Data Model," submitted to the fourth IEEE Data Engineering Conference, Los Angeles, CA, Feb 2-4, 1988
- [Su86] Su, S. Y. W., "Modeling integrated manufacturing data with SAM\*," *Computer*, January 1986, pp.34-49

## 2.3 System Integration Techniques

### 2.3.1 The Robotics Prototyping System

Investigator: T.N. Mudge

#### Problem Description and Research Objectives

We are developing a Robotic Prototyping System (RPS) that will allow researchers to experiment with ideas quickly. However, the intent of this project is broader than simply assembling a state-of-the-art resource for experimental robotics. The long term goal is to establish a research environment in which new and more fruitful modes of research will emerge. Three points that we believe are key ingredients in achieving

this are the ability to represent data graphically (although how to represent non-visual data still remains an open question), the capability to acquire sensory data quickly from experiments, and the ability to command the effectors from the same programs that acquire and process the sensory data.

### Approach

We have chosen the technologies of networked graphics workstations, high performance array processors and reusable software components as the basis for RPS. Networked graphic workstations support the software development environment on which RPS is based. Networking allows sharing of a limited number of sensors with out the processing power limitations of a single processor. They also provide the graphic display of CAD model and sensory data. High performance array processors are used to accelerate the time consuming low-level image processing on which all vision algorithms are based. Array processors that support floating point calculations may also be used to compute robot kinematics. Reusable Software components provide a researcher with a library of standard functions which have been well characterized. This will allow the rapid assembly of tested modules and will drastically reduce the time and cost spent in software development.

### Status

Figure 12 shows the current hardware configuration of the RPS. The system integrates cameras, a Mercury Zip array processor, a tactile sensor, two PUMA 600 robots with servo grippers, an X-Y table, a wrist-force sensor and a wrist force sensor. As can be seen from the figure, the system integration is achieved through a network of workstations (we are using the Apollo domain). It is the recent availability of commercial high performance workstations that can be networked and that are capable of displaying high quality graphics that makes the RPS feasible. The network allows the prototyping hardware to be shared among a group of users; eventually it will allow software in the form of reusable software components to be shared. The graphics capability of the workstations allows pictorial representations of the sensor data that, we believe, significantly improves the users ability to conceptualize the results of various experimental steps.

In the case of images for robot vision, it is fairly obvious that the images captured with the cameras are their own best representation, and can be displayed directly on the workstation. Similarly, images that have been processed as part of vision algorithm can also be represented on the workstation directly. So, for instance, the following may occur during the development of an algorithm to extract depth information using stereo views of an object:

The two images of the object captured by the cameras are displayed initially.  
Then the processed versions of the images showing noise removal, edge

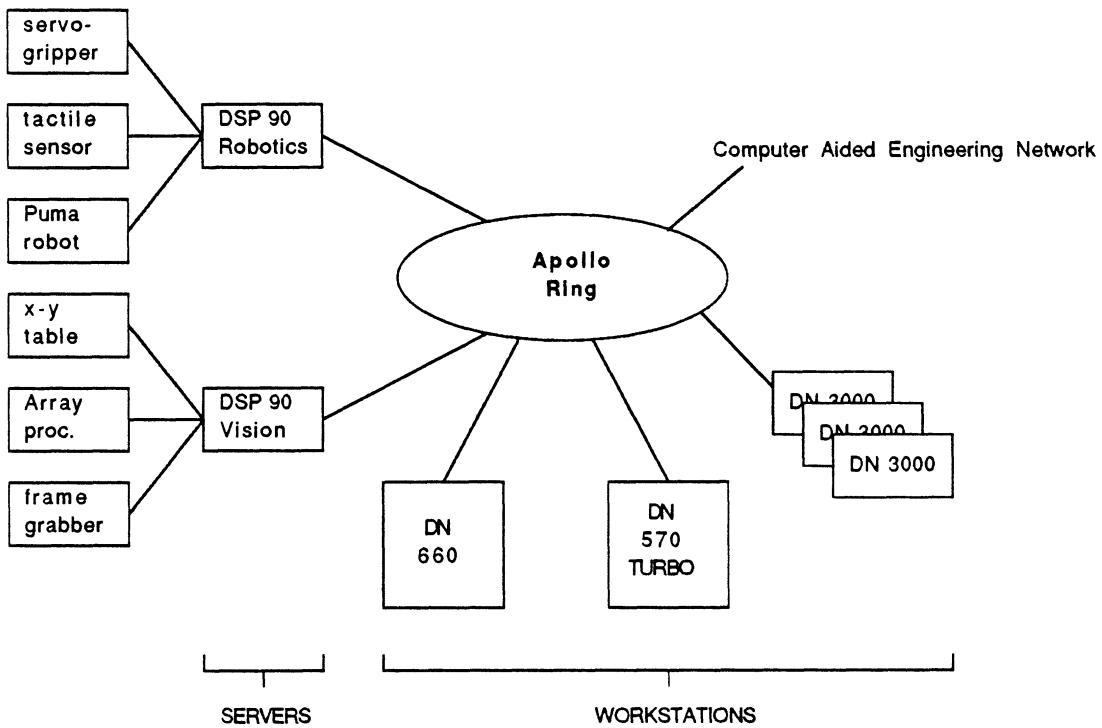


Figure 12: Robot prototyping system.

detection, feature extraction, and correlation operations between features in both of the images. This might be followed by annotated images that depict the effects of trying to make correspondences between the features in the two images.

At each phase the windowing capability can be used to display multiple images in various stages of processing in addition to the code that operates on them. This instant access to a representation of the sensory data and the code associated with extracting the "meaning" from that data will, we believe, result in the new modes of research mentioned above. In the case of non-visual sensory data, representing it graphically is an important issue if the above research paradigm is to succeed.

The RPS is currently capable of performing the above stereo vision scenario by the use of a set of high performance image processing boards from Mercury Computing Systems. The Mercury Zip array processor has been fully integrated into the Apollo Domain so that it is transparently accessible from any host on the network. As a compute server the array processor receives data and instructions from a host and sends back processed data to the host using the paradigm of a remote procedure call. The processed data can be displayed in a window or saved for later use. The bit-sliced technology used in the array processor makes possible a  $3 \times 3$  kernel convolution with a  $512 \times 512$  image in under 300 msec. A library of routines have been developed to allow researchers to exploit this speed to perform low level image processing tasks (eg. filtering, histograming). A variety of edge detection routines have also been developed. The most significant of these is Canny edge detection, which is an optimal edge detector that produces well connected pixel wide edges. The array processor implementation of Canny's algorithm takes 2 seconds to execute on a  $512 \times 512$  image, where as the Apollo high-performance graphics workstation DN-570-T executes the same algorithm in 120 seconds. The two order of magnitude increase in speed greatly increases a researcher's efficiency thus allowing the exploration of more algorithmic alternatives. Figure 13 shows a workstation display with a window that contains an original image and the results of Canny edge detection.

As an example of sensory and motor integration an application package has been developed that automatically determines the transformation between camera coordinates and robot coordinates and is then able to pick up a cylindrical object placed within the robots grasp. The use of previously developed packages to control the robot and to acquire images from the camera greatly facilitated the construction of this application.

Work so far has focussed on vision applications. A more challenging problem is to invent ways to represent non-visual data. Figure 14 shows an example of our initial attempts at this with the Lord tactile sensor. The sensor is organized as a  $10 \times 16$  array of elements. The sensor is capable of indicating, among other things, 4 bits of



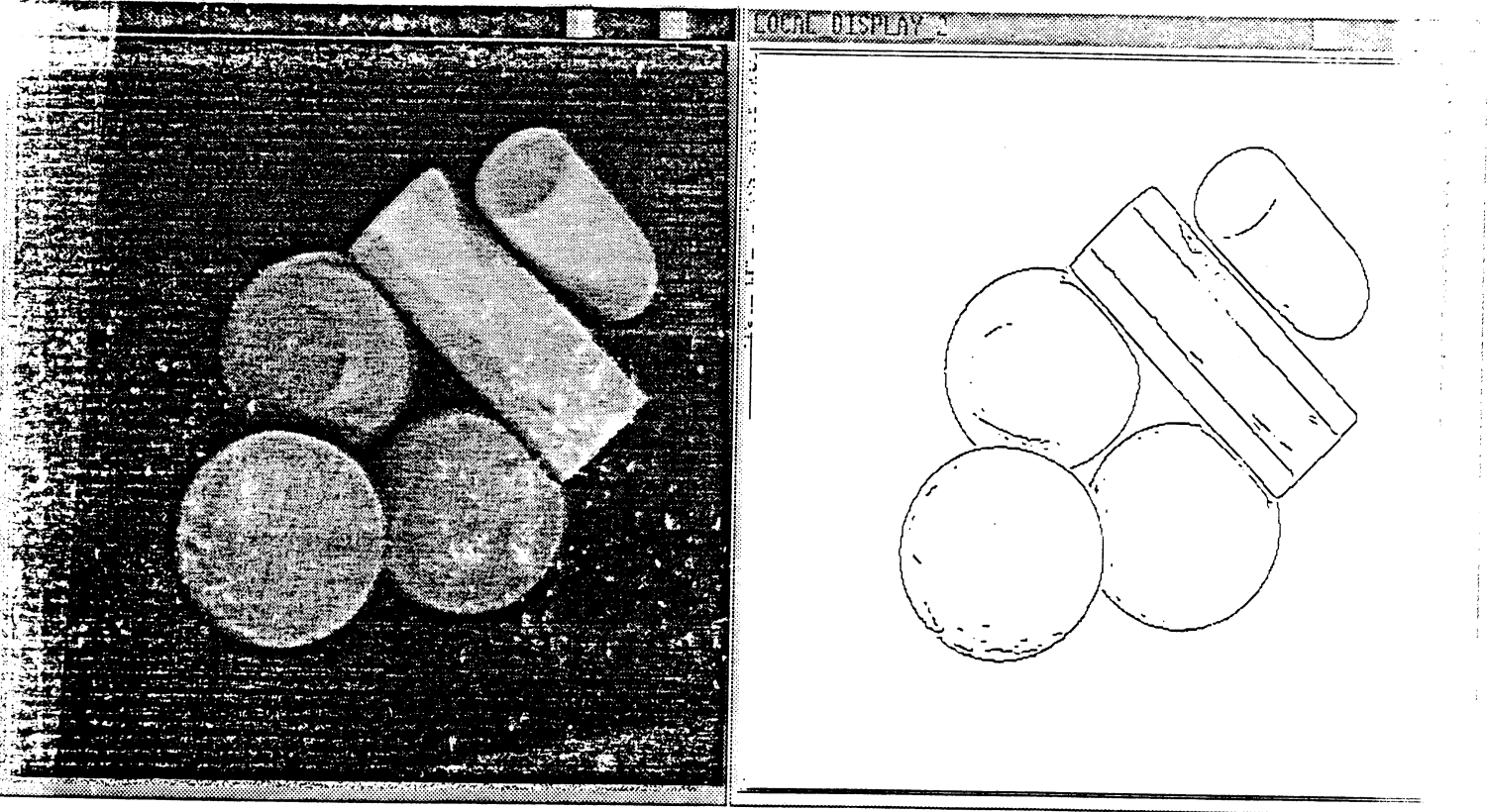


Figure 13: Canny edge detection

Surface

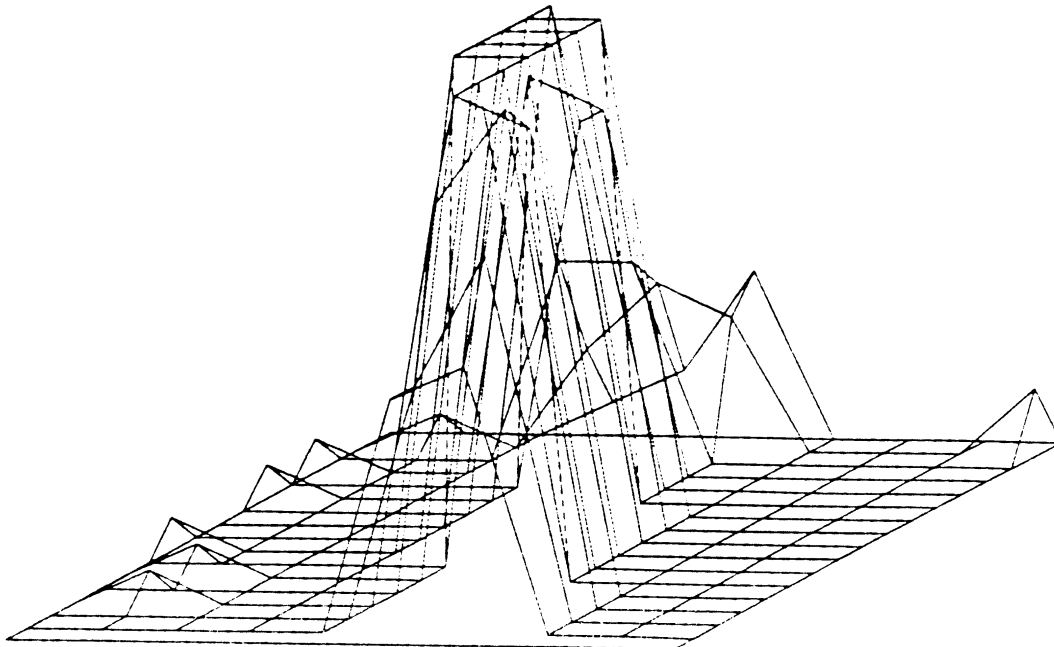


Figure 14: Visual representation of tactile data.

deflection at each element. Figure 13 shows a plot of the deflection when the robot arm with the sensor grips an object. Clearly there is a strong similarity between the gray levels at the photo-sites of a camera and the deflection levels at each element of the tactile sensor. We are currently using this analogy as a guide to exploring representations for the tactile sensor. For example, edges in a visual scene are loci of high gray level gradient; by analogy edges in a “deflection image” are loci of high deflection gradient as may occur when the gripper grasps a block by its corners.

### Future Directions

The work we plan for the next year is as follows:

- Integration of GEOMOD, a solid modeling program, into the system. This will allow the creation of synthetic images to test out ideas when real scenes are not appropriate, as for example if a segmentation algorithm is to be tested on an “impossible” figure (such as an Escher staircase), or for example when a noise free image is required. Preliminary work has begun and GEOMOD has been used to test out some ideas in 3D vision.
- The addition of a floating point array processor. For most problems in vision a fixed point array processor is sufficient however, robotic kinematics is another computationally intensive application that require floating point arithmetic and could also benefit from an array processor.
- Evaluate a suitable range sensor to integrate into the RPS. Such a sensor could be used to generate GEOMOD models.
- Develop more algorithms for the array processor.
- Interface our hypercube multiprocessor (see net section) to the RPS network to provide a massively parallel processor resource to the users.
- Continue the collection and development of a coherent set of reusable software components for the RPS. The user community is an important source of software components. We are in the process of defining interface standards so that users can share their software.

### 2.3.2 Research into Vision Algorithms

Investigator: T.N. Mudge

**Problem Description and Research Objectives.** A characteristic of much of the work in robotics is the need to process large sets of data. This is particularly true of vision work where single images typically contain  $\frac{1}{4}$  megabytes of data from a standard camera with  $512 \times 512$  photosites. If multiple views are required (as in our

stereo example above) this can grow further, and if video frame rate processing is required as in the case of dynamic scene analysis (c.f. Ramesh Jain's work) as much as 10 megabytes of raw image data needs to be processed per second. Furthermore, near term developments will give us color images (a factor of 3 increase in data) and finer pixel resolution from the cameras ( $4096 \times 4096$  instead of  $256 \times 256$ ), all of which are going to lead to continued increases in the data sizes.

An attractive solution to deal with the enormous computation requirements of robot vision is parallel processing. This is particularly attractive for low-level robot vision tasks like edge detection, since many of these low-level tasks are simply repeated neighborhood transformations applied to the two dimensional array of image data. Such operations are inherently parallel. Another development that contributes to the attractiveness of parallel processing is the emergence of very high performance 32-bit microcomputers with the functionality and performance of supermini class computers. Assembling large numbers of these to form massively parallel machines is now a possibility. The objective of this research is to investigate the uses of massively parallel processors for vision applications.

**Approach.** Our work in this area has focussed on developing algorithms for the NCUBE massively parallel hypercube multiprocessor [6]. An  $n$ -dimensional hypercube or  $n$ -cube computer is a multiprocessor that consists of  $N = 2^n$  processors interconnected as an  $n$ -dimensional binary cube. Each processor  $P$  forms a node of the cube with its own cpu and local main memory.  $P$  has direct communication paths to  $n$  other processors (its neighbors), which correspond to the edges of the cube that are connected directly to  $P$ . Figure 15 illustrates the hypercube topology for  $n = 4$ , or a 4-cube. Hypercubes have a number of features which make them attractive for vision applications as well as many other applications [4], [5], [6].

We have performed our experiments on the NCUBE/six hypercube multiprocessor which has 64 processors interconnected as a 6-dimensional cube. We have benchmarked a number of low-level and mid-level computer vision algorithms and the results have been reported in [4], [5], [7]. The results have been encouraging and current work is now focussed on high-level algorithms. The first step in this work is to characterize high-level algorithms for robotics. Our initial attempts are examining generalized branch and bound (B&B) algorithms. A large class of algorithms such as graph theoretic algorithms, backtracking, searching, AND-OR search, and rewriting systems can be viewed as B&B, and these are the kinds of algorithms that one associates with "reasoning" about a scene, particularly at the level where multisensor integration is being performed.

The B&B algorithms finds a solution to a problem by repeatedly decomposing the solution space of that problem into subspaces of decreasing sizes examining each subspace until a solution is found. To bound the number of subspaces examined, if a subspace

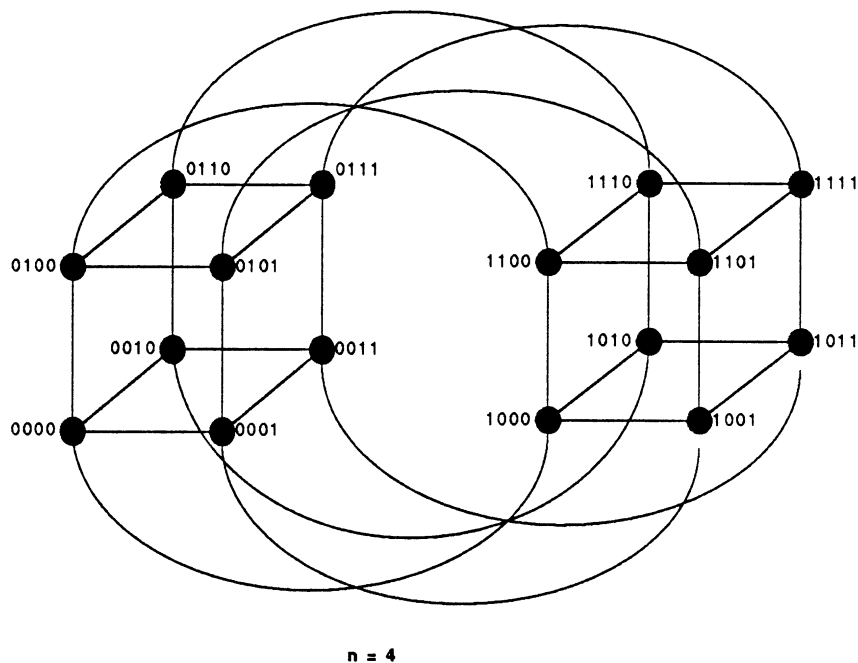


Figure 15: A 4-cube.

is shown not to contain a solution it is eliminated from further decompositions by the algorithm. The repeated decomposition and elimination of solution subspaces forms the “branching” and “bounding” of the B&B algorithm. The decomposition process is represented as a rooted tree over the solution space. The root of the tree represents the complete solution space. Children nodes represent partial solution spaces of the original space, or equivalently *subproblems* of the original problem. The algorithm selects a subproblem that is most likely to lead to a solution based on some heuristic information from the problem domain and decomposes it to smaller subproblems, hence, extending the B&B tree. An example of a B&B tree is shown in figure 16.

**Status.** We have implemented a generic B&B algorithm that consists of two components: a controller and a user library. The controller implements the control strategy component of the algorithm and does not require any knowledge of problem. The user library implements the problem itself. This approach allows us to experiment freely with a number of problems. We have tested the algorithm with integer programming problems and traveling salesman problems. We have selected these problems because the correctness of their solutions can be easily validated while still capturing the features of the heuristic search.

A B&B algorithm requires global information for the selection and deletion of subproblems during the search process. In the case of distributed memory multiprocessors

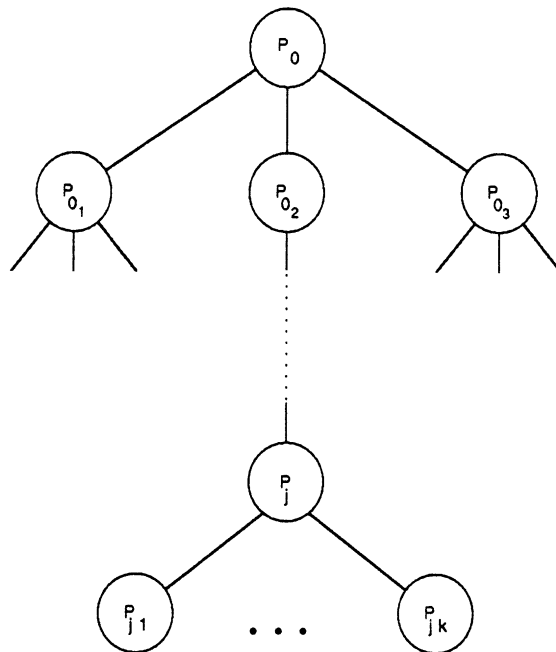


Figure 16: A B&amp;B tree.

such as the hypercube, enforcing this requirement can result in severe performance degradation due to the high communication overhead among the processors. An interesting property of parallel B&B algorithms is that such global information may be only partially maintained at the expense of more search steps to find a solution.

We have considered this possible tradeoff between the communication overhead and the number of solution steps in our experiments. Figure 17 shows the speedup of two implementations which illustrate the tradeoff [8]. The first algorithm, referred to as the centralized algorithm (CENT), maintains all global information in the host memory. In this case, all the information is available to the processors, but rather at a high communication overhead. The second algorithm referred to as the distributed algorithm (DIST), distributes the global information across the processors. Consequently, not all of the information is accessible to all the processors in the system. However, the communication overhead is low. The results indicate that distributing the information can lead to better results. This result may not be true in all cases. We are currently looking into ways to characterize B&B algorithms, in particular high level vision applications, to decide on the best implementation.

#### Future Direction.

- Investigating and evaluating other implementations of parallel B&B algorithms on hypercubes that demonstrate the various tradeoffs involved.

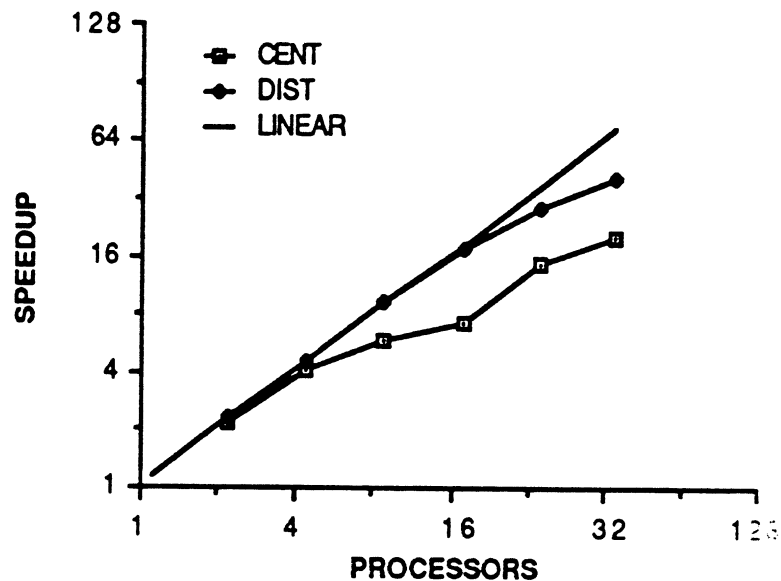


Figure 17: Results.

- Relating generic B&B algorithms already implemented to actual high-level vision algorithms.
- Integrating the low and mid-levels of vision with the high-level on the NCUBE system.
- Making the NCUBE a server on the RPS network, providing a high speed parallel processor interface to the user.

## References

- [1] J.L. Turney, T.N. Mudge and R.A. Volz. "Recognizing Partially Hidden Objects," *Proc Soc. of Photo-Optical Engineering*, Vol. 521, pp. 108-113, November 1984.
- [2] J.L. Turney, T.N. Mudge and R.A. Volz. "Recognizing Partially Hidden Objects," *IEEE. Trans. PAMI*, Vol. PAMI-7, No. 4, pp. 410-421, July 1985.
- [3] J.L. Turney, T.N. Mudge and R.A. Volz. "Automatic Generation of Salient Features for the Recognition of Partially Occluded Parts," to appear *Robotica*, 1987.
- [4] T.N. Mudge and T. S. Abdel-Rahman. "Vision Algorithms for Hypercube Machines," *Journal of Parallel and Distributed Computing*, 4, 1987.
- [5] T.N. Mudge. "The Next Generation of Hypercube Computers," *ARO Workshop on Future Directions in Computer Architecture and Software*, May 1986.

- [6] J.P. Hayes, T.N. Mudge, Q.F. Stout, S. Colley and J. Palmer. "Architecture of a Hypercube Supercomputer," *1986 International Conference on Parallel Processing*, August 1986.
- [7] T. N. Mudge and T. S. Abdel-Rahman. "Architectures for Robot Vision," in *Specialized Computer Architectures for Robotics and Automation*, J. Graham, ed., Gordon & Breach, New York, 1987.
- [8] T. S. Abdel-Rahman and T. N. Mudge. "Parallel Branch-and-Bound Algorithms on Hypercube Multiprocessors," to be presented at *Third SIAM Conference on Parallel Processing for Scientific Computing*, December, 1987.

### 2.3.3 Distributed Systems Integration Techniques

Investigator: R.A. Volz

Due to the importance of related work on distributed systems integration techniques to this project, we briefly describe that work. Essentially, this work revolves around distributed computing techniques.

#### Problem Discussion

Distributed computing is an essential part of most large systems today. Until now, however, the principal focus on distributed computing systems has been on their architecture (particularly interconnection mechanisms) and their gross capabilities (usually calculated in some simplistic sense such as multiplying the capabilities of a single processor by the number of processors in the system). One of the most critical problems for the future is the set of programming tools available for such systems.

Software tools for distributed systems must deal with both diverse hardware and the use of existing software written in a wide variety of languages. They must also incorporate the best techniques developed in software engineering over the past two decades and extend these concepts effectively for use in distributed systems<sup>1</sup> It is our hypothesis that extending these concepts to permit distributed execution is a critical step in addressing the distributed computing software problem. Two obvious benefits of so doing are the extension of compile time error checking across machine boundaries and allowing the programmer to use normal language mechanisms for expressing parallel (concurrent) operation without having to invent new application level communication protocols. (Note that computer communication networks do *not* adequately address the applications level protocol issue.) These two advantages alone should greatly improve the problem of developing software for distributed systems.

---

<sup>1</sup>Among the key concepts are: 1) data encapsulation and hiding, 2) abstract data types, 3) modularization of programs, 4) separate compilation (of both modules and specifications), 5) concurrency mechanisms at the language level, and 6) extensive compile time error checking.

### Previous Work and Research Status

The implementation of a system to support distributed program execution for real-time applications requires the solution to a substantial number of problems. Previously, we have discussed the following basic issues:

- Problem analysis—the basic issues in distributed program execution, [1],
- Timing mechanisms—basic software timing mechanisms amongst distributed tasks, [2], and new instruction level mechanisms for simplifying timing implementation, [3],
- Performance evaluation—specifically oriented toward the real-time performance of emitted code, [4], and
- Experimental implementation techniques, [5].

During this past year, we have concentrated on two things, the development of techniques for our second generation translator, and the application of these ideas to manufacturing software, the latter work heavily involving Prof. A. W. Naylor and being partially funded by General Motors.

#### *Distributed Translation Techniques*

Our distributed Ada system allows us to distribute library packages and library subprograms statically among a set of homogeneous processors.

At the present time, the distributed translation system is operational for distributed packages with remote access available to:

- both simple and complex data objects. Record and array objects can be nested arbitrarily deeply within one another and include pointers to remote objects.
- subprograms
- declared task objects (no timed or conditional calls across processor boundaries).

In addition, the handling of task types is nearly complete. There is also a Unix make-file like capability to simplify use of the translation system. Tests have been successfully completed with up to three VAX processors cooperating on the execution of a single program. We write a single program and use a `pragma` (essentially a compiler directive) called `SITE` to specify the location on which each library unit is to execute. For example, if the simple transport system mentioned earlier were controlled by computer number 2 and the cell control using it were on computer 1, a sample of relevant code might look as follows:

```
pragma SITE (2);
```



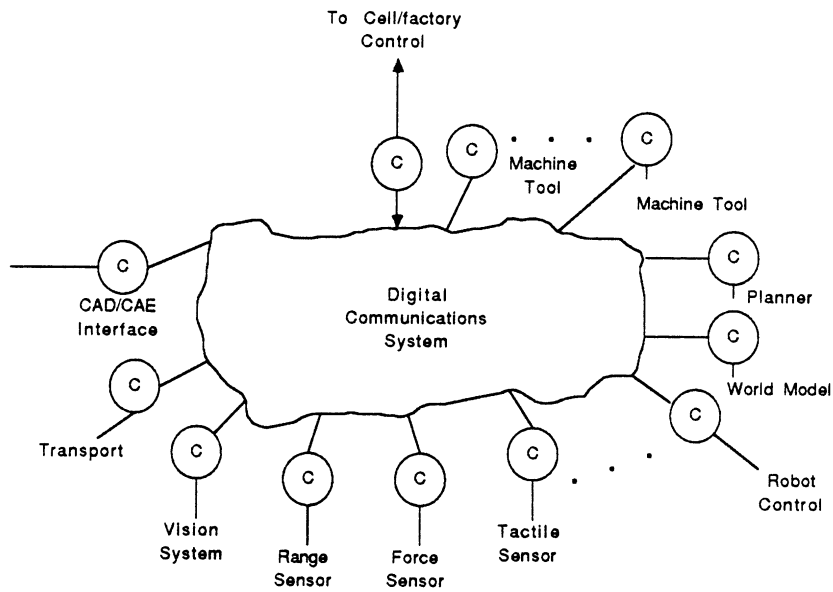


Figure 18:

```

package VEHICLE is
    procedure MOVE_FORWARD;
    :
end VEHICLE;
:
pragma SITE(1);
with VEHICLE;
procedure CONTROL is
    :
begin
    :
    VEHICLE.MOVE_FORWARD;
    :
end;

```

Our translation system would replace the local call to the procedure `VEHICLE.MOVE_FORWARD` with the appropriate remote call. Similarly any references in `CONTROL` to data objects defined in package `VEHICLE` would be translated into appropriate remote references as would task entry calls.

The operation of our translator is illustrated in Figures 18, 19 and 20. We presume that

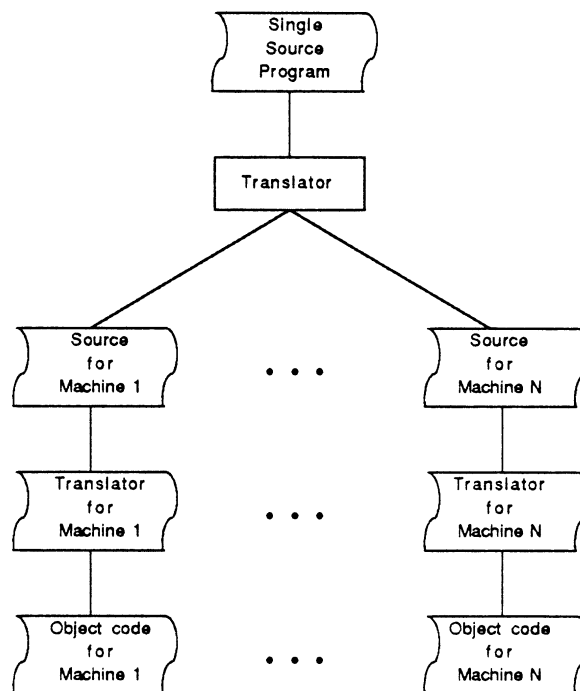


Figure 19:

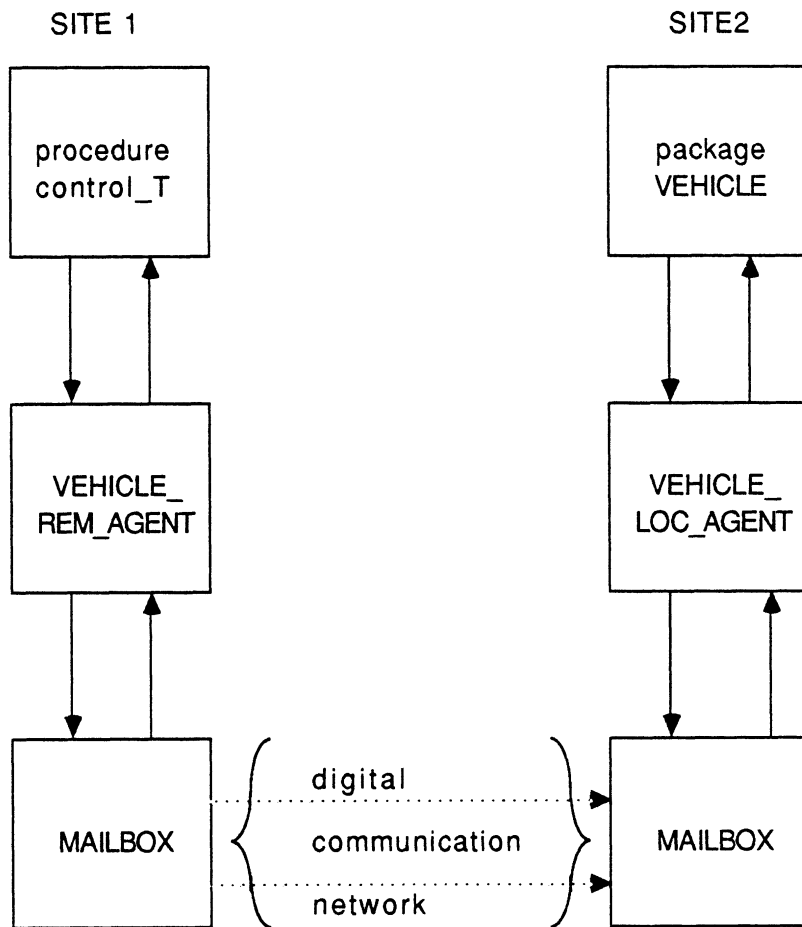


Figure 20:

our computers are interconnected by a communication network, as shown in Figure 18. Figure 19 illustrates the operation of our translator. It translates our single Ada program into a set of independent Ada programs which include library modules of our design to effect communication among processes. Each of the individual programs can thus be compiled by an existing Ada compiler. This approach simplifies the translation process considerably since our pre-translator is much less complex than a full Ada compiler.

The communication is managed by a set of *agents* [6] created for units that can be remotely referenced and an underlying process-to-process mailbox routine. Roughly speaking, the relevant entities and flow of data are as shown in Figure 20. CONTROL\_T is the translation of CONTROL with the references to objects in VEHICLE replaced with calls into VEHICLE\_REM\_AGENT.

The operation of this system can be modeled in terms of the run time overhead associated with various kinds of remote references. From the tests performed in [4] we know that task rendezvous times exceed procedure call times by one and a half to two orders of magnitude. We can also reasonable expect the network communication times to be sizable. For example message end-to-end times for repeat MAP are on the order of 100ms, [7], for the Intel hypercube, a few milliseconds, and for the NCUBE hypercube, several hundred microseconds to a millisecond. Thus, we neglect all local procedure and function call times, and model our overhead in terms of the number of messages and local rendezvous required. Let  $t_m$  and  $t_r$  be the times to complete a message transfer and local rendezvous, respectively, and let  $n_m^o$  and  $n_r^o$  be the number of messages and local rendezvous required for a remote operation of type  $o$ . Then, the time to complete a remote operation is

$$n_m^o \cdot T_m + n_r^o \cdot t_r$$

For task rendezvous we have  $n_m^r = 2$ ,  $n_r^r = 6$  and the remote rendezvous time is  $2 \cdot t_m + 6 \cdot t_r$ . For example, if we have a slow network in which  $t_m = 100\text{ms}$  and an Ada compiler that produces 0.500 ms rendezvous times (several such compilers now exist for Motorola 680x class processors), we would have a remote rendezvous time of 203 ms, obviously, heavily dominated by the message passing time. For a fast network with, say, a  $t_m$  of 500  $\mu$  sec, we would have a rendezvous time of 4 ms., which is close to single machine rendezvous times of a year ago [4].

Clearly, within the message passing times achievable with today's technology, a reasonable distributed language capability is feasible. For a more complete description of the distributed Ada translator system, see [6].

#### *Manufacturing software*

Our conceptual framework for manufacturing software is based upon the following key concepts which are illustrated in Figure 21:

- recursively defined *hardware/software components* encompassing both the logical and physical aspects of the entity being defined,
- *formal semantic models* of the components and their assemblages, and of process plans.
- *generic* components that can be instantiated into real components by supplying appropriate parameters (either statically at the time a system is built, or during an initialization period with parameters supplied by physical devices attached to the system),
- *libraries* of components, and
- a common *distributed language* environment.

Our view of hardware/software components is both recursive and distributed. That is, components may be built up out of other components, and need not all reside on the same computer. In fact, one component may itself span several computers. Each physical device on the factory floor is encapsulated inside a software/hardware component, and can be thought of as having two faces. One is made up of its interactions on the factory floor. The other is the software interface connected to the higher levels of the manufacturing software system. Our components must encompass both; indeed, the purpose of the software is to control the interactions on the factory floor through the software interface.

The formal modeling part of our framework is expressed in a simple extension to first-order logic [8] that can be used to represent the process plans that the cell is to implement as well as the actions of the components. The uniform modeling of process plans and software/hardware components simplifies the software structure and allows one to view the process plan as just another component in the system.

The generic components are abstractions of real components designed to allow reusability of component software. By *instantiating* them with actual parameters, real components are created and can be used as part of a real factory control system. The top half of Figure 21 illustrates the relationship among the models, generic components and actual parameters. The instantiation process shown here is a generalization of the simple generic instantiation process of languages like Ada<sup>2</sup>. Actual components of the manufacturing system are created from the generic components by supplying parameters that complete the component description.

One component may include *models* of other components. The models may then used in a predictive simulation manner to examine the likely outcome of a possible control strategy before it is actually applied. And, the formal models of process plans can be

---

<sup>2</sup>Ada is a registered trademark of the Ada Joint Program Office.

converted to actual components that drive the operation of a system. At present the translation from the formal models to actual software is performed manually, but conceptually (at present, and in the future actually) they could be converted automatically. These instantiated components are then used in the control software being developed, as shown in the bottom half of Figure 1.

The bottom half of Figure 21 illustrates not only a manufacturing software system, but the environments in which it was created and operates. While the underlying digital communications network is shown in this figure, it is logically contained within the distributed language environment. That is, the software for controlling the manufacturing system is written using normal *interprocess* communication mechanisms within the language, without any explicit references to *interprocessor* communication. There is a mapping that assigns program parts to the different processors in the system. The language translation system ensures that interprocess communication is appropriately translated to interprocessor communication when necessary. Thus, the programmer does not have to be concerned with network protocols or inventing application level inter-program protocols.

### Future Directions

Both areas of work described above will continue during the coming year. The next areas to be investigated and implemented in the translation system will be distributed task types and generics. Both of these involve some particularly difficult problems, since it can be shown that one cannot know at the normal compilation time whether or not certain references will be local or remote.

The application of distributed languages and software components ideas to manufacturing software will take a direction of investigating design methodologies and examining the role of the component approach in these.

## References

- [1] R.A. Volz, T.N. Mudge, G.D. Buzzard, and P. Krishnan. Translation and execution of distributed Ada programs: is it still Ada? *IEEE Transactions on Software, Special Issue on Ada*, to appear 1987.
- [2] R.A. Volz and T.N. Mudge. Timing issues in the distributed execution of Ada programs. *IEEE Transactions on Computers, Special issue on Parallel and Distributed Processing*, C-36(4):449-459, April 1987.

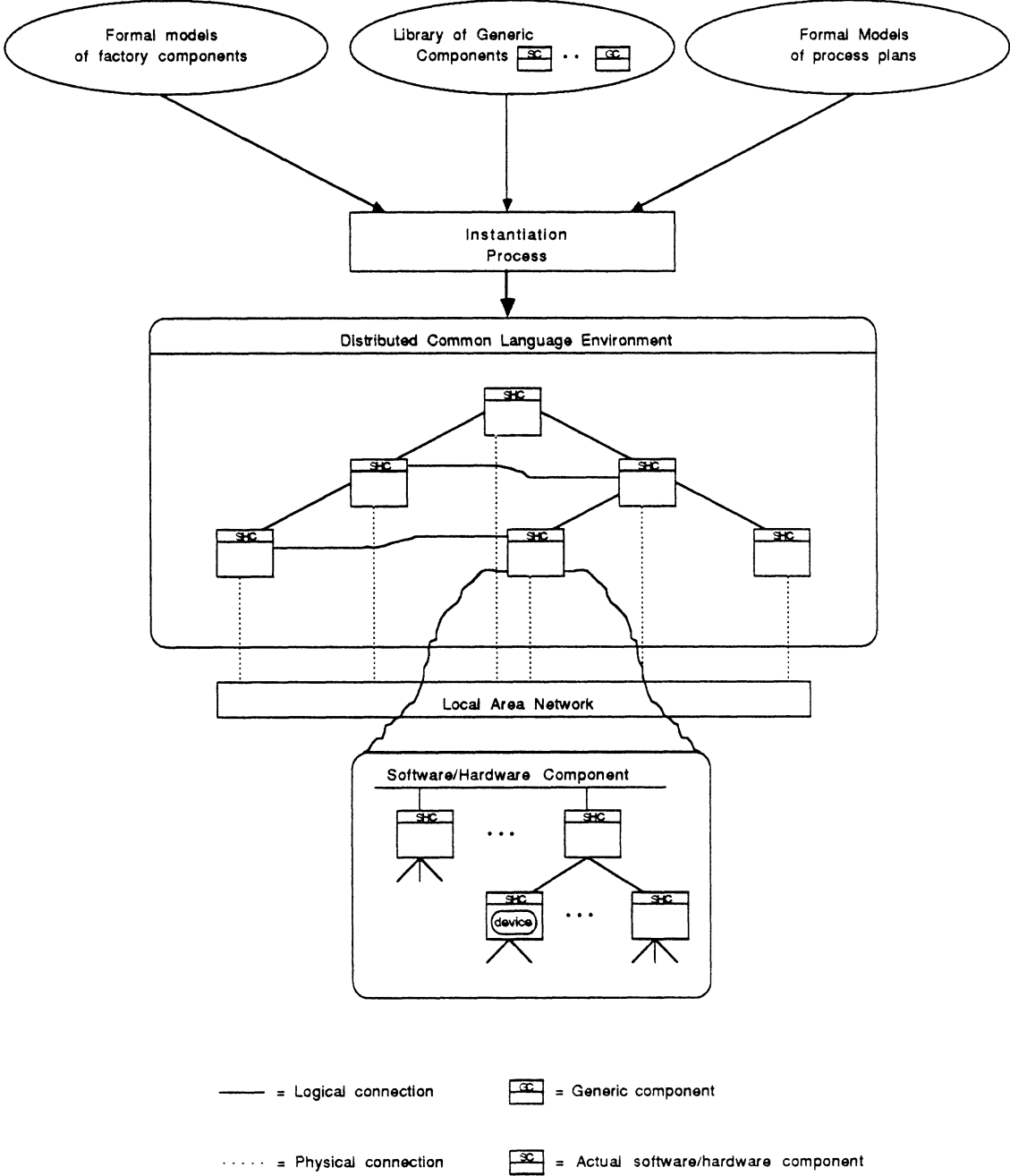


Figure 21:

- [3] R.A. Volz and T.N. Mudge. Instruction level timing mechanisms for accurate real-time task scheduling. *IEEE Transactions on Computers, Special issue on Real-Time Systems*, C-36(8):988-993, August 1987.
  
- [4] R.M. Clapp, L. Duchesneau, R.A. Volz, T.N. Mudge, and T. Schultze. Toward real-time performance benchmarks for Ada. *Communication of the ACM*, 29(8):760-778, August 1986.
  
- [5] R.A. Volz, T.N. Mudge, A.W. Naylor, and J.H. Mayer. Some problems in distributing real-time Ada programs across machines. In J.G.P. Barnes and G.A. Fischer, editors, *Ada in use, Proc. 1985 International Ada Conf.*, pages 72-84, Cambridge University Press, May 1985.
  
- [6] R.A. Volz, P. Krishnan, and R. Theriault. An approach to distributed execution of Ada programs. In *NASA Workshop on Telerobotics*, to appear 1987.
  
- [7] C. Green. Theorem proving resolution as a basis for question-answering system. *Machine Intelligence*, 1969.
  
- [8] H.B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, 1972.



## 3 Publications

### 3.1 Journal Publications

#### Motion Stereo

1. Ramesh Jain, Sandra L. Bartlett, Nancy O'Brien, "Motion Stereo Using Ego-Motion Complex Logarithmic Mapping," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 3, May 1987.

#### Object Recognition

1. Trevor N. Mudge, Jerry L. Turney, Richard A. Volz, "Automatic Generation of Salient Features for the Recognition of Partially Occluded Parts," *Robotica*, vol. 5, part 2, April-June 1987, pp. 117-127.
2. Paul J. Besl, Ramesh C. Jain, "Invariant Surface Characteristics for 3D Object Recognition in Range Images," *Computer Vision, Graphics, and Image Processing* 33, pp. 33-80, 1986.
3. Paul J. Besl, Ramesh C. Jain, "Segmentation Through Symbolic Surface Descriptions," Hypotheses," *IEEE Trans. on PAMI*, in press.

#### Real-Time Task Scheduling

1. Richard Volz, Trevor Mudge, "Instruction Level Timing Mechanisms for Accurate Real-Time Task Scheduling," *Special Issue IEEE Transactions on Computer*, Aug. 1987.

#### Vision for Robots

1. Shih-Ping Liou, Ramesh C. Jain, "Road Following Using Vanishing Points," *IEEE Computer Vision, Graphics, and Image Processing* 39, pp. 116-130, 1987.

#### Integrated Manufacturing System

1. Kang G. Shin, "Port Manipulator for the Distributed Realization of an Integrated Manufacturing System," *Journal of Computer Systems, Science and Engineering*, 1987.
2. A.W. Naylor, R.A. Volz, "Design of Integrated Manufacturing System Control Software," *IEEE Systems, Man and Cybernetics*, July 1987.
3. K.G. Shin and M.E. Epstein, "Intertask Communications in an Integrated Multi-robot System," *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 2, pp. 90-100, April 1987.

### Planning and Programming

1. Kang G. Shin, Neil D. McKay, "Robust Trajectory Planning for Robotic Manipulators Under Payload Uncertainties," *IEEE Transactions on Automatic Control*, vol. AC-32, no. 12, December 1987 (to appear).
2. Richard A. Volz, *et al*, "Automatic Evaluation of Two-Fingered Grips," *IEEE Transactions of Robots and Automation*, vol. 3, no. 4, August 1987.

### Geometric Modeling

1. Y.C. Lee, K.S. Fu, "Machine Understanding of CSG: Extraction and Unification of Manufacturing Features," *IEEE Computer Graphics, and Applications*, vol. 7, no. 1, January 1987, pp. 20-32.

### Problem Solving

1. Irani, K.B. and Yoo, S.I., "A Methodology for Solving Problems: Problem Modeling and Automatic Heuristic Generation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1987.

### Machine Learning

1. Usama M. Fayyad, Kristina E. Vanvoorhis, Mark D. Wiesmeyer, "Learning Control Information in Rule-Based Systems," *4th IEEE Conference on AI Applications*, San Diego, CA, March 1988.

## 3.2 Conference Presentations and Papers

### Object Recognition

1. T.N. Mudge, J.L. Turney, P.G. Gottschalk, "Two-Dimensional Partially Visible Object Recognition Using Efficient Multidimensional Range Queries," *Proceedings of the 1987 International Conference on Robotics and Automation*, April 1987, pp. 380-386.
2. Ramesh Jain, Thawach Sripradisvarakul, Nancy O'Brien, "Symbolic Surface Descriptors for 3-Dimensional Object Recognition," *Proceedings of the SPIE*, January 1987.
3. Thomas F. Knoll, Ramesh C. Jain, "Learning to Recognize Objects Using Feature Indexed Hypotheses," *International Conference on Computer Vision*, London, U.K., June 1987.

### Robot Control

1. Theodore A. Nesse, Kang G. Shin, "High-Level Design of a Spray Finishing Robot Controller," *ROBOTS 11/17TH ISIR Conference*, Chicago, Illinois, April 27-30, 1987.

### Planning and Programming

1. Y.C. Lee, M.A. Doctor, "Automatic Determination of Retrieving Paths in a Semantic Data Model," *Proceedings of the IEEE Computer Society Symposium on Office Automation*, Gaithersburg, MD, April 27-29, 1987, pp. 89-95.
2. Richard A. Volz, Lejun Shao, "PROGRESS - A Graphical Robot Programming System," *1987 IEEE Conference on Robotics and Automation*.
3. K.B. Irani and Jie Cheng, "Subgoal Ordering and Goal Augmentation for Heuristic Problem Solving," *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, Milan, Italy, August 23-28, 1987, pp. 1018-1024.

### Geometric Modeling

1. Y.C. Lee, K.-F. Jea, "PAR: A Representation Scheme for Rotational Parts," *Proceedings of the IEEE International Conference on Robotics and Automation*, Raleigh, NC, March 30 - April 03, 1987, pp. 973-978.

### System Integration

1. T.N. Mudge, O.A. Olukotun, "A Preliminary Investigation into Parallel Routing on a Hypercube Computer," *Proceedings of the 24-th Design Automation Conference*, Miami Beach, FL, June 1987, pp. 814-820.
2. Richard Volz, "Distributed Ada Execution: A Definitional Void," *International Workshop on Real-Time Ada Issues*, May 1987.

### Integrated Manufacturing System

1. Richard Volz, Arch Naylor, "Software for Integrated Manufacturing Systems," *SOAR '87 Conference*, Houston, TX, August 1987, pp. 397-403.
2. Arch W. Naylor, Richard A. Volz, "Integration and Flexibility of Software for Integrated Manufacturing Systems," *National Academy of Engineers Talk*, February 1987.

### 3.3 Under Review

#### Journals

1. Y.C. Lee, K.-F. Jea, "PAR: A CSG-based Unique Representation Scheme for Rotational Parts," to appear in *IEEE Transactions on Systems, Man, and Cybernetics*.

#### Conferences

1. Paul G. Gottschalk, Trevor N. Mudge, Richard A. Volz, "Testing Object Recognition Techniques: A Standard Set of Objects and Some Reporting Guidelines," *1988 Robotics and Automation Conference*, September 1987.
2. Jing Xiao, Richard A. Volz, "Design and Motion Constraints for Part-Mating Planning in the Presence of Uncertainties," *accepted for 1988 International Robotics and Automation Conference*, September 1987.
3. Y.C. Lee, M.A. Doctor, E. Armstrong, "ESQL: An Enhanced but Simple Query Language for a Semantic Data Model," submitted to the *Fourth IEEE International Conference on Data Engineering*, Los Angeles, CA, February 1988.
4. Y.C. Lee, C.H. Cho, "Logical Extension of Relational Databases," submitted to the *Fourth IEEE International Conference on Data Engineering*, Los Angeles, CA, February 1988.

## 4 Personnel

### 4.1 Faculty

There are many ways in which a faculty member or student may be supported by a contract of this type. The names listed on the form DD1473 include only those faculty who received direct salary support from the contract. However, in addition, some faculty received support indirectly through support of their graduate or post doctoral students. This section lists faculty who have either directly received support for a portion of their salaries or received support for one or more of their graduate students.

#### Professors

Keki B. Irani	Electrical Engineering and Computer Science
Ramesh Jain	Electrical Engineering and Computer Science
Kang G. Shin	Electrical Engineering and Computer Science
Richard A. Volz	Electrical Engineering and Computer Science

#### Associate Professors

Trevor N. Mudge	Electrical Engineering and Computer Science
-----------------	---

#### Assistant Professors

Y.-C. Lee	Electrical Engineering and Computer Science
-----------	---

## 4.2 Students

The number of students who have received support during the second year of this contract exceeds the number of “standard” student appointments indicated in the proposal. There are two primary reasons for this. First, and most importantly, we were able to leverage the student funding to increase the student involvement. In a number of cases we were able to obtain partial fellowship support for the student from internal funds, thus reducing the cost per student and increasing the number of students we could involve in the project. Similarly some students had partial outside fellowship support and only needed supplementary support to be able to participate in the project. The involvement of students with fellowship support in the project is deemed an acceptable activity since the work being performed is generally part of the student’s dissertation work. Secondly, through natural attrition, graduation, and staff changeover, some replacement of the students supported occurred during the year. The students listed below, then, include all of the students who have received support during the contract year.

Tarek Abdel-Rahman	EECS <sup>3</sup> Ph.D. expected 1987-88
Sandra Bartlett	CSE <sup>4</sup> Ph.D. expected 1987-88
Fan Jiang	ECE <sup>5</sup> Ph.D. expected 1988-89
Sung Taeg Jun	EECS Ph.D. expected 1988-89
Oyekunle Olukotun	EECS Ph.D. expected 1988-89
Zhaogang Qian	CICE <sup>6</sup> Ph.D. expected 1988-89
Kwang Ryel-Ryu	CICE Ph.D. expected 1989-89
Thawach Sripradisvarakul	CICE Ph.D. expected 1987-88
Jing Xiao	CICE Ph.D. expected 1988-89
Jie Cheng	CICE Ph.D. expected 1988-89
C.-H. Cho	CSE Ph.D. expected 1989-90
Jan Wolter	CICE Ph.D. expected 1987-88
Usama Fayyad	CSE Ph.D. expected 1989-90
Susan Haynes	EECS Ph.D. expected 1986-87
Lejun Shao	CICE Ph.D. expected 1988-89
Woo-Jong Lee	IOE <sup>7</sup> Ph.D. expected 1989-90

---

<sup>3</sup>Electrical Engineering and Computer Science

<sup>4</sup>Computer Science and Engineering

<sup>5</sup>Electrical and Computer Engineering

<sup>6</sup>Computer, Information, and Control Engineering

<sup>7</sup>Industrial Operations Engineering

### 4.3 Degrees Awarded

Dan Johnson Ph.D. in AE<sup>8</sup>

### 4.4 Graduate Student Placement

#### *1982-1983*

Gary Swanson, Ph.D.	Lincoln Labs
Henry Chu, M.S.	U of M Ph.D. Program
Paul Eichel, M.S.	U of M Ph.D. Program
Stacy Leon, M.S.	U of M Ph.D. Program
Mike Steigerwald, M.S.	U of M Ph.D. Program
Jan Wolter, M.S.	U of M Ph.D. Program
Brian Dent, M.S.	ADAPCO, Melville, New York
Myrung Chung, Ph.D.	Korean Institute of Technology
Dean Askounis, M.S.	FMC Robots, Troy, Michigan

#### *1983-1984*

Thomas Wielenga, Ph.D.	M.D.I., Ann Arbor, Michigan
Joseph Pincu, M.S.	U of M Ph.D. Program
Daniel Barash, M.S.	Harris Corporation, Washington D.C.
Paul Firehammer, M.S.	unknown
Nirwan Ansari, M.S.	U of M Ph.D. Program
David Gal, M.S.	R.O.L.M. in California
Mark Epstein, M.S.	IBM, Boca Raton, Florida
Neil Nathason, M.S.	IBM, Toledo, Ohio
Chi-Chan Tsang, M.S.	Computerized Office Systems, Ann Arbor
Glen Healy, B.S.	Stanford Graduate Program
Elesh Shah, M.S.	unknown
Jerry Turney, M.S.	U of M Ph.D. Program

#### *1984-85*

D. Beard, Ph.D.	University of North Carolina
B. Lee, Ph.D.	Purdue University
N. McKay, Ph.D.	General Motors Research Laboratory

---

<sup>8</sup>Aerospace Engineering



J. Ha, Ph.D.	General Motors Research Laboratory	
P. Eichel, Ph.D.	Sandia Laboratory	
J. Lah, Ph.D.	Intel Corporation	
S. Yoo, Ph.D.	Seoul National University	1985-86
D. Shin, Ph.D.	University of Connecticut	
P. Bixel, M.S.	General Electric	
K. Lloyd, M.S.	Bell Laboratories	
R. Rubinfeld, B.S.	University of California, Berkeley	

Paul Besl, Ph.D.	General Motors Tech Center
Nabil Chalhoub, Ph.D.	University of Reno
Il Hyun Choi, Ph.D.	IBM - Virginia
Kukjin Chun, Ph.D.	University of Washington
H.P. Huang, Ph.D.	National Taiwan University
Brian Schipper, M.S.E.	unknown
Jerry Turney, Ph.D.	KMS Fusion Inc.

1986-87

Dan Johnson, Ph.D. Martin Marietta, Baltimore

## 4.5 Permanent Staff

### Research Engineers

Al Dobryden

### Systems Programmers

Brent Harper

Ron Theriault

### Technician

Rob Giles

### Administrative Assistant

Janice Ransom

### Secretaries

Dolores Bolsenga

Marianne Moylan

Elizabeth Olsen

## 5 Coupling Activities

An important aspect of our program is the development of strong coupling activities both with and without the program. Four categories of interaction are being pursued:

- interaction among people participating in the project
- interactions with other university scientific groups
- interactions with industry
- interactions with the Air Force

The following sections describe the coupling activities which have taken place during the past year.

### 5.1 Intra Project Interactions

There are regular technical discussion meetings among members of several of the research groups occurring on a weekly basis. Membership in more than one group is common, particularly among the faculty concerned. Also, most of the faculty are members of several doctoral committees on topics outside of their own immediate research activities. This has proven to be a useful form of information diffusion.

### 5.2 University Interactions

Interactions with people from other universities have taken place primarily through seminars, both those given at the University of Michigan, and those given by Michigan people at other universities.

#### Outside Seminar Speakers

- W.J. Book, Georgia Institute of Technology  
“Automation with Lighter, Faster Machines”
- Professor Lui Sha, Carnegie-Mellon University  
“Real-Time Systems Engineering Closing the Gap Between Theory and Practice”
- Professor Leonard Uhr, University of Wisconsin  
“Massively Parallel Hierarchical Perceptual Recognition”
- Professor Sirius Chitsaz, North Carolina State University  
“The CSPC Research Agenda and Interactions with Industry”

- Christopher LeMaistre, Rensselaer Polytechnic Institute  
“CIM, Composites and the Center for Industrial Innovations”
- Professor John Hopcroft, Cornell University  
“The Promise of Electronic Prototyping”
- Professor John White, Georgia Institute of Technology  
“A Systems View of Manufacturing: A Material Handling Perspective”
- Professor Roger Brockett, Harvard University  
“Modeling and Control of the Process of Grasping”
- Professor Moshe Barash, Purdue University  
“Study in Improving Machine Tool Accuracy”

#### **Seminars at other Universities given by Michigan faculty**

- Professor K.G. Shin - Ohio State University  
“Design and Analysis of Distributed Real-Time Systems”
- Professor K.G. Shin - University of Texas - Austin  
“Combined Performance and Reliability of Real-Time Systems”
- Professor R.A. Volz - Cornell University  
“An Approach to Future Robot and Manufacturing Software”

### **5.3 Interactions with Industry**

#### **Seminars and Briefings for Industry**

- Professor K.B. Irani - General Motors Research Labs.
- Professor R. Jain - General Dynamics
- Professor R.A. Volz - Fairchild, NASA/Goddard, Tau Corporation, General Motors, General Dynamics, Grumman Aerospace
- Professor K.G. Shin - General Motors and Ford

#### **Seminars at Michigan by Industrial People**

- Gerhard Roth and Colin Archibold - National Research Council of Canada  
“Sensor Based Robotics at the NRCC Laboratory for Intelligent Systems”

- Harold Dewhurst, Owens-Corning Fiberglas  
“Electronic Communications, Technology in Industry/University Relations”
- Jack Russell, Technology Deployment Services  
“The Technology Deployment Service of the State of Michigan”
- George H. Kuper, National Academy of Sciences  
“The Proposed National Center for Manufacturing Sciences”
- Professor Ranga Komanduri, National Science Foundation  
“On Tool Materials for Machining”
- Professor Harris Burte, Wright-Patterson Air Force Base  
“Unified Life Cycle Engineering
- Professor Susan Finger, National Science Foundation  
“Current Theories of Design - Summary and Issues”

#### **Private technical discussions with industry**

- Professor R. Jain has had regular discussions with ERIM, Ford, Hitachi, Dana
- Professor T.N. Mudge - Vicom, Applied Intelligent Systems, HBW Management, Synthetic Vision Systems, KMS Fusion, Astronautics of America, and Applied Dynamics International
- Professor R.A. Volz - General Motors, General Dynamics, Grumman Aerospace, Fairchild, Alsys, Tau Corporation

#### **Faculty and Students Giving Seminars within the University**

- Professor K.B. Irani - AI-Seminar  
“Research Interest - Robot Planning and Problem Solving”
- Professor N.H. McClamroch - “Time-Optimal Control for a Robotic Contour Following Problem”
- Mark Jakiela - “An Intelligent Design Modeller for the Conceptual Design Phase”
- Professor E.G. Gilbert - “Distance Measures for Objects in Three Space and Their Application to Robot Collision Problems”
- D.W. Johnson - “A Fast Procedure for Computing the Distance Between Complex Objects in Three Space”
- Robert D. Throne - “Robot Path Planning Using Geodesic and Straight Line Segments with Voronoi Diagrams”
- Professor A.C. Woo - “Linear Dissassembly in the Plane”

- Professor M. Walker - "Manipulator Kinematic and the Epsilon Algebra"
- Suk-Hwan Suh - "Planning of Minimum-Time Trajectories Under Practical Considerations"
- Professor J.L. Stein - "The Robotics/Human Interfaces: The Design and Control of Prosthetic Limbs"
- Suk-Hwan Suh - "A Variational Dynamic Programming Approach to Robot Path Planning with Distance-Safety Criterion"

#### **5.4 Government Interaction**

- Professor T.N. Mudge - "Issues in Computer Architecture", U.S. Army Electronics Research Strategy Planning Workshop, Quail Roost, North Carolina
- Professor K.G. Shin - ONR workshop on real-time systems research, November 1986
- Professor R.A. Volz - "Distributed Execution and Performance of Ada Programs," June 1987
- Professor R.A. Volz - "New Concepts in Tele-Autonomous Systems," June 1987

- Jack Russell, Technology Deployment Services  
“The Technology Deployment Service of the State of Michigan”
- George H. Kuper, National Academy of Sciences  
“The Proposed National Center for Manufacturing Sciences”
- Professor Ranga Komanduri, National Science Foundation  
“On Tool Materials for Machining”
- Professor Harris Burte, Wright-Patterson Air Force Base  
“Unified Life Cycle Engineering
- Professor Susan Finger, National Science Foundation  
“Current Theories of Design - Summary and Issues”

#### **Private technical discussions with industry**

- Professor R. Jain has had regular discussions with ERIM, Ford, Hitachi, Dana
- Professor T.N. Mudge - Vicom, Applied Intelligent Systems, HBW Management, Synthetic Vision Systems, KMS Fusion, Astronautics of America, and Applied Dynamics International
- Professor R.A. Volz - General Motors, General Dynamics, Grumman Aerospace, Fairchild, Alsys, Tau Corporation

#### **Faculty and Students Giving Seminars within the University**

- Professor K.B. Irani - AI-Seminar  
“Research Interest - Robot Planning and Problem Solving”
- Professor N.H. McClamroch - “Time-Optimal Control for a Robotic Contour Following Problem”
- Mark Jakiela - “An Intelligent Design Modeller for the Conceptual Design Phase”
- Professor E.G. Gilbert - “Distance Measures for Objects in Three Space and Their Application to Robot Collision Problems”
- D.W. Johnson - “A Fast Procedure for Computing the Distance Between Complex Objects in Three Space”
- Robert D. Throne - “Robot Path Planning Using Geodesic and Straight Line Segments with Voronoi Diagrams”
- Professor A.C. Woo - “Linear Dissassembly in the Plane”
- Professor M. Walker - “Manipulator Kinematic and the Epsilon Algebra”

## 6 Distribution List

### University and Research Institutes

Professor Thomas Binford  
Computer Science Dept.  
Stanford University  
Stanford, CA 94305

Professor Robert McGhee  
Dept. of Electrical Engineering  
Ohio State University  
2015 Neil Avenue  
Columbus, OH 43210

Professor Raj Reddy  
Computer Science Dept. & Robotics  
Carnegie Mellon University  
Schenley Park  
Pittsburgh, PA 15213

Professor Michael Brady  
Cambridge  
England

Professor Richard Paul  
Dept. of Computer Information  
and Sciences  
University of Pennsylvania  
N60 Town Bldg.  
Philadelphia, PA 19104

Professor Roger Nagel  
Lehigh University  
Bethlehem, PA 18015

Professor Georges Saridis  
ECSE  
Rensselaer Polytechnical Institute  
Troy, NY 12181

Professor Delbert Tesar  
Dept. of Mechanical Engineering  
University of Texas  
Austin, Texas

Dr. David Whitney  
Draper Laboratories  
555 Technology Square  
Massachusetts Institute of Technology  
Cambridge, MA 02139

Dr. David Nitzen  
Director of Robotics Research  
Stanford Research Institute  
333 Ravenswood Avenue  
Menlo Park, CA 94025

Dr. Robert Bolles  
Stanford Research Institute  
333 Ravenswood Avenue  
Menlo Park, CA 94025

Dr. William Brown  
ERIM  
P.O. Box 8618  
Ann Arbor, MI 48107

Dr. Azriel Rosenfeld  
 Research - Room 4115C  
 Computer & Space Sciences Bldg.  
 University of Maryland  
 College Park, MD 20742

Dr. Geoffrey Boothroyd  
 Mechanical Engineering  
 University of Massachusetts  
 Amhurst, MA 01003

Dr. Don Falkenburg  
 Industrial Technology Institute  
 Beal Avenue  
 Ann Arbor, MI 48109

#### Government

Dr. James Albus  
 Room A127 Bldg 220  
 National Bureau of Standards  
 Washington D.C. 20234

Dr. Mike Wozny  
 Program Director  
 Automation and System Integration  
 Division of Design, Manufacturing  
 and Computer Engineering  
 National Science Foundation  
 1800 G. Street N.W.  
 Washington D.C. 20550

Dr. Howard Moraff  
 ERC Program  
 National Science Foundation  
 1800 G. Street N.W.  
 Washington D.C. 20550

Mr. Bernie Chern  
 Division of Electrical, Computer &  
 System Engineering  
 Room 1101  
 NSF  
 Washington D.C. 20550

Captain Robert Delyser  
 DFEE  
 U.S. Air Force Academy  
 Colorado 80840

Dr. Charles Holland  
 Head  
 Dept. of Navy  
 Information Sciences Division  
 Office of Naval Research  
 800 N. Quincy  
 Arlington, VA 22217

Dr. Vincent Russo  
 Wright Patterson Air Force Base  
 AFWAL/MLTC  
 Wright Patterson, OH 45433

Mr. Thomas Lagnese  
 Project Director  
 Computer Integrated  
 Manufacturing Branch  
 Department of Air Force  
 Air Force Wright Aeronautical Lab  
 Wright Patterson Air Force Base,  
 Ohio 45433



Mr. Michael Hitchcock  
Computer Integrated  
Manufacturing Branch  
Department of Air Force  
Air Force Wright Aeronautical Lab  
Wright Patterson Air Force Base,  
Ohio 45433

Mr. William M. Spurgeon  
University of Michigan at Dearborn  
Dearborn, Michigan

Mr. G. Ronald Green  
AIRMICS  
115 O'Keefe Building  
Georgia Institute of Technology  
Atlanta, GA 30332-0800

Mr. Ted J. Brandewie  
Computer Integrated  
Manufacturing Branch  
Department of Air Force  
Air Force Wright Aeronautical Lab  
Wright Patterson Air Force Base,  
Ohio 45433

Mr. Daniel Herman  
NASA Headquarters  
Code S  
Washington DC, 20546

Major James Crowley  
Directorate of Mathematics  
and Information Sciences  
AFOSR  
Bolling Air Force Base  
Washington, D.C. 20332-6448

### Industry

Mr. Mark Hornick  
ASEA  
Industrial Robot Division  
16250 W. Glendale  
New Berlin, WI 53151

Mr. Otto Renius  
Chief Scientist  
General Dynamics  
Land Systems Division  
P.O. Box 2074  
Warren, MI 48090

Mr. Don Dresselhouse  
General Dynamics  
Land Systems Division  
P.O. Box 1902  
Warren, MI 48092

Dr. Steve Holland  
Mr. Frank DiPietro  
Mr. Gerald Elson  
Mr. Gabe Tiberio  
Mr. Richard Beecher  
General Motors  
General Motors Research Laboratories  
Warren, MI 48090

Mr. Pete Matheson  
Intel Corporation  
5200 N.E. Elam Young Park Way  
Hillsboro, OR 97123

Mr. Emil Sarpa  
College Relations Manager  
Intel Corporation  
2565 Walsh Avenue  
Santa Clara, CA 95051

Mr. Anthony Anderson  
Intel Corporation  
7071 Orchard Lake Road  
Suite 100  
West Bloomfield, MI 48033

Mr. J.L. Escover  
Mr. Steve DeBrock  
Lockheed  
Missiles and Space Company, Inc.  
P.O. Box 1700  
Austin, TX 78760

Mr. Dwight Carlson  
Perceptron  
23920 Freeway Park Drive  
Farmington Hills, MI 48024

Mr. Clifford T. Anglewicz  
Volvo of America Corporation  
Volvo Automated Systems Division  
40712 Brentwood Drive  
Sterling Heights, MI 48078

Dr. Gale Cutler  
Whirlpool  
Monte Road  
Benton Harbor, MI 49022

Dr. John Klein  
IBM  
1798 NW 40th Street  
Boca Raton, FL 33432

Mr. Bruce Haupt  
IBM  
951 N.W. 51st.  
Boca Raton, FL 33432

Dr. Michael Wesley  
IBM  
T.J. Watson Research Center  
P.O. Box 218  
Yorktown Heights, NY 10598

Mr. William Hollenback  
IBM  
Neighborhood Rd.  
Kingston, NY 12401

Mr. George Leeman  
IBM  
Thomas J. Watson Research Ctr.  
P.O. Box 704  
Yorktown Hts, NY 10598

Mr. James Lowrie  
Advanced Automation  
Technology Section  
Martin Marietta Aerospace  
P.O. Box 179  
Denver, CO 80209

**RSD-TR-23-87**

Mr. Donald E. Waters  
Program Manager  
Technical Strategy Center  
Honeywell, Inc.  
1700 W. Highway 36  
St. Paul, MN 55113

## Appendix A

### Robotics Industrial Affiliates Members List

**ASEA Robotics, Inc.**

Industrial Robot Division

16250 W. Glendale

New Berlin, Wisconsin 53151

**Company Contact:**

Mark Hornick

Telephone: (414) 785-3482

**Deneb Robotics**

1120 East Long Lake Road, Suite 200

Troy, MI 48098

**Company Contact**

Dr. Rakesh Mahajan

Director Sales and Marketing

(313) 689-7763

**Digital Equipment Corp.**

APO-1/G2 100 Minuteman Road

Andover, MA 01810

**Company Contact:**

Mr. Bob Williams

Manager, NASA

GSG Federal Government

Accounts Group

8181 Professional Pl.

Landover, Maryland 20785

(301) 731-9287

Mr. Al Jones, Technology Exchange

Program Manager

HL02-3/K11

77 Reed Road

Hudson, MA 01749

**RSD-TR-23-87**

**Ford Aerospace and Communications Corp**  
Western Development Laboratories Division  
3939 Fabian Way M/S G13  
Palo Alto, CA 94303

**Company Contact:**

Mr. Lawrence P. Seidman  
Manager  
Adv. Systems Space Systems Operation  
(415) 852-4771

**General Dynamics**  
Land Systems Division  
P.O. Box 2074  
Warren, Michigan 48090

**Company Contact:**

Mr. Otto Renius  
Chief Scientist  
Land Systems Division  
(313) 362-8096

**General Motors**  
General Motors Research Laboratories  
Warren, MI 48090-9057

**Company Contact**

Steven W. Holland, Asst. Head  
Computer Science Department  
(313) 986-1510

**Grumman Aerospace Corporation**  
Aircraft Systems Division  
Mail Station B18-002  
Bethpage, New York 11714

**Company Contact**

Mr. Ronald J. Heitzmann  
Executive Staff Assistant  
(516) 575-7122

**Intel Corporation**

7071 Orchard Lake Road  
Suite 100  
West Bloomfield, Michigan 48033

**Company Contact:**

Mr. Keith Hooper  
Telephone: (313) 851-8096  
Ms. Suzzane DeStross  
Manager Corporate College  
Relations and Recruiting  
5000 W. Chandler Blvd. WF-104  
Chandler, AZ 85226  
(602) 961-8051

**Lockheed**

Missiles and Space Company, Inc.  
Department T3-35  
Building 30-D  
2124 East St. Elmo Street  
Austin, Texas 78744  
Mail to: P.O. Box 1700  
Austin, Texas 78760

**Company Contact:**

J. L. Escover  
Mfg. Engineering Manager  
Telephone: (512) 362-7307

**Lockheed**

Missiles and Space Company, Inc.  
Dept. 5301  
Bldg. 580  
1111 Lockheed Way  
Sunnyvale, California 94086

**Company Contact:**

Mr. S. C. DeBrock  
(University Liason)  
Telephone: (408) 743-1571

**RSD-TR-23-87**

**Lockheed**

Lockheed Aeronautical Systems

Dept. 41-11

Bldg. 1

Zone 102

86 South Cobb Dr.

Marietta, GA 30063

**Contact Person:**

Mr. Walt Plumley

(404) 424-3794

**Perceptron**

23855 Research Drive

Farmington Hills, MI 48024

**Contact Person:**

Mr. Jeremy Salinger

(313) 478-7710

**Westinghouse Corporation**

Artificial Intelligence

335 E. Expo-Mart

105 Mall Blvd.

Monroeville, PA 15146

**Company Contact**

Edward P. Liscio

Telephone: (412) 374-7266





**Appendix B**  
**Industrial Technology Institute Report**





10 September 1987

Kathryn R. Warner  
Contracts Administrator  
University of Michigan  
Office of Contract Administration  
341 West Engineering Building  
Ann Arbor, Michigan 48109

Re: Report on Account 080012  
Subcontract on Air Force Contract No. F33615-85-C-5105

Dear Ms. Warner:

The Industrial Technology Institute respectfully submits bimonthly report number 1, per the cited contract effort above, for the period covering 7/1/87 to 8/31/87, as called out in the statement of work on this effort. This report covers the technical status of this effort during this period, schedule information, and projected work for the next period. Costing information is submitted separately.

Sincerely,

A handwritten signature in cursive script that reads 'RJBieringer'.

Dr. Robert J. Bieringer, Manager  
Automated Inspection & Monitoring

RJB/sj

Enclosure



## Industrial Technology Institute

P.O. Box 1485  
Ann Arbor, MI 48106

Bimonthly report number 1  
for 7/87 - 9/87  
on subcontract to  
University of Michigan  
Air Force Contract NO. F33615-85-C-5105

### Program Objective

The objective of this program is to develop a moire interferometer and expert system technique for 3-dimensional scene analysis. The purpose of this program is to develop this nondestructive evaluation technique such as to be capable of full-field range measurement and 3-dimensional scene analysis of randomly ordered and partially occluded parts. The application of scene understanding is directed toward flexible assembly, inspection and as an aid to total work cell automation.

The concept demonstration phase of this program being conducted at this time has the objective,

- to establish the potential of optical data manipulation through moire techniques;
- to identify the requirements and limits of the application of this approach to 3-D machine vision; and
- to develop AI techniques to use this data for scene analysis.

A primary emphasis of the moire development is the establishment of techniques to provide absolute moire measurement, including providing information on the sign of the slope of surfaces, and the absolute position of surfaces as determined by a point in space fixed with respect to the sensor.

### Schedule

The program schedule is shown in the form on the following page.

### Technical Progress

During this reporting period, work was conducted to better establish the approach and

PROGRAM SCHEDULE

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
MOIRE MACHINE VISION														
Phase I tasks														
A. Absolute moire development														
1. Frequency shifting theory							====							
2. White light color separation experiments							=====							
3. Laser light color separation experiments							=====							
4. Frequency shifting moire experiments							=====							
5. Data analysis development							=====							
6. Order equipment							====							
B. Scene analysis														
1. Access current technology							=====							
2. Select approach							=====							
3. Develop AI sys prototype							=====							
4. Exercise & evaluate prototype AI system							=====							
C. Reporting														
											*			final

subtasks to be pursued for both the moire development, and the scene analysis. Two primary issues to be addressed in using moire for 3-D measurements is determining the sign of the slope of surfaces, and determining the absolute position of such surfaces. The former problem has been addressed in the past by using bias fringes in the moire pattern, or by recording three sets of moire interference fringes, each shifted in phase by a small amount. Both of these techniques permit the distinguishing of peaks from valleys, information not readily available in simple interferometry techniques. However, the measurements are still insensitive to a rigid body translation of the object, i.e., the data is only relative to other connected points on the object, and in the case of multi-phase or phase shifting, time is taken to capture three serial images. Such insensitivity to rigid body translations can be useful in some situations, but is insufficient when looking at a step change in the surface geometry, or determining the location of the surface to be related to the rest of the world, such as to a robot's coordinate system.

The approach being investigated to overcome these issues is to use simultaneous viewing of two or three sets of moire patterns which are separated by means of color or other optical separating scheme. The three images may be viewed by a single camera capable of recording three color images (RGB camera), or by three separate cameras which view three images separated by some form of color or other filtering scheme. The first consideration will be to apply this approach to obtaining three moire fringe patterns of different phase as described above. This will allow us to analyse the effectiveness of the separation scheme using techniques a possibly machine vision hardware already developed for analysing phase shifted interference fringe patterns. The second consideration will be to apply the image encoding and separation method to three moire fringe patterns in which the frequency of the moire fringe pattern has been changed rather than the phase. This latter consideration of frequency shifting is believed to have the capability to provide the absolute moire measurement data desired, relative to a fixed and locatable point in space. The theoretical development will be conducted in parallel with the initial image separation experiments.

There are three sets of variables which will be investigated to identify the best combination to provide three simultaneously taken images, which can be separated by the optics or electronics.

- **Projection means** - a *white light* projection means would incorporate a white light source, and either a colored grating, or a black and white grating which is imaged to the subject in question. A white light source of this type would be inexpensive, but may be light inefficient and limited in the type of

objects that can be viewed. A *Laser source* can produce a projected grating pattern by passing the light through a prism, such as a Wollaston prism, which splits the laser beam into two beams at a small angle to each other. The two beams can then be made to interfere to create a coherent interference pattern to be used as the master grating. The grating pattern made by means of the laser has the advantages that:

- the grating pattern is a sine wave pattern as desired to create a clean moire pattern
- the grating pattern exists over a much larger depth-of-field than can be obtained with a comparable white light system,
- the colors of the laser grating pattern are pure colors which do not overlap (as most white light colors do), and as such should be easy to separate)
- the interference pattern created by the laser light passing through the Wollaston prism will already change in period or frequency for different colors of the laser light, in a very predictable manner.

The primary disadvantage of the laser is the cost and possible bulk of a sufficient light source. Although the intent is to use an argon laser in initial experiments for ease of evaluation, new diode lasers with many colors of output light have recently started to become available. These diode lasers are very compact and robust, but generally put out light in the near infrared region, where video cameras can see, but people can not.

- **Grating character** - either color or black and white in conjunction with optical filters. To perform moire measurements, we must project one grating pattern onto the part to be contoured, and view that grating pattern through a second submaster grating, the difference between these two gratings being the beat fringe which constitutes the moire fringe pattern. If a color grating is used on both the input and output of the system, then the resulting moire pattern will consist of up to 6 colored sets of fringes produced by the beats between both like colors, and unlike colors. It may be difficult to separate these many patterns, or it may provide additional information which can be used. Using a color grating on either the input or output and a black and white grating on the other will produce only three colors of fringes (if all of the gratings are of the same frequency). Each color of fringe pattern would appear in black and white to be formed in the same manner that a pair of black and white gratings would create fringe patterns. The final alternative is to use three entirely separate sets of black and white gratings (with one common input black and white grating) which are viewed by three separate black and white cameras. This last option is the most brute force of the methods, and would be the most difficult to properly align and maintain (since all components in the viewing systems are independent).



- **Camera sensor** - can be either an RGB color camera, or three separate black and white cameras. A single RGB color camera would afford a fixed relationship of the three images, as they would all come from the same physical camera. The primary issue in question is whether the separation of the three colors in an RGB camera is sufficient to present three separate images. The ability to separate the images will be a factor of both the RGB camera parameters, and the colors presented to the camera. The colors from a white light source seen through a color grating will likely be broad in nature, covering a range of wavelengths of light which may overlap. The colors from a laser will not be broad and will not overlap. Using three black and white cameras, narrow color filters or dichroic mirrors (which only reflect one color) may be more effective or better matched to separating the colors than the filters in the RGB color camera. The three camera arrangement, however, will not have the same inherent alignment and compactness that a single camera would provide.

These variables are summarized in the tables below.

To date on the moire development effort, we have identified a ready source of color gratings or high quality in the form of color polaroid film. This film, when exposed to white light, is made up of many fine sets of red, green and blue lines. We have been able to balance the intensity of these colors to appear uniform to a black and white camera. We are currently considering the projected quality, color separation, and mechanical stability of this film product. As an option, we may make our own gratings by contact printing black and white gratings, illuminated with different colors, onto Ektachrome slide film. The separability of these colors in the gratings may dictate which combination of source and gratings will be most appropriate or feasible for the rest of the development. Equipment for the color and three-camera experiments has been ordered, including the RGB camera system, three black and white cameras, and monitors. In addition, basic mounting equipment, optics, and supplies have been ordered. This equipment is expected to be received within the next month, at which time we can begin the RGB versus black and white camera options comparisons.

The AI work on this project has concentrated on identifying two basic research problems within the overall architecture of a moire-based scene analysis system. 1. Our literature survey shows that the highest level attained by most vision research is that of geometric primitives, such as cylinders and planes, while manufacturing applications need to reason in terms of functional features, such as ribs, fillets, and flanges. Representation of and reasoning with such features poses a number of interesting problems, since they involve functions in the application domain of manufacturing and

mechanical engineering as well as pure geometric features. 2. The most credible candidates for representation both of pre-processed image information and of descriptions of expected items are semantic network constructs. In this context, the recognition problem becomes one of subgraph homomorphism, which is computationally prohibitive. We are exploring parallel relaxation heuristics for addressing this problem.

Table 1. Color sorting variable options.

<u>INPUT SOURCE</u>	<u>OUTPUT GRATING</u>	<u>CAMERA</u>
white/color grating	color	RGB
white/color grating	b&w	RGB
white/color grating	color	3 mono & dichroic
white/color grating	b&w	3 mono & dichroic
laser/prism	b&w	RGB
laser/prism	color	RGB
laser/prism	b&w	3 mono
laser/prism	color	3 mono

Table 2. Variable advantages and disadvantages.

<u>VARIABLE</u>	<u>ADVANTAGES</u>	<u>DISADVANTAGES</u>
White light	inexpensive, compact	energy inefficient, shallow depth range
Laser Source	pure, separate colors large depth range	expensive, bulky
Colored grating	inexpensive fixed image relations	color separation unknown
RGB Camera	fixed image relations compact	color separation unknown
B&W camera/ grating	controlled color separation	bulky, image alignment not fixed

### Projected Work

During the next reporting period, we will complete analysis of the white light, and color grating projection option, and perform initial tests with the RGB camera. Work will continue on the theoretical derivation of the frequency shifted moire interpretation, as well as the image interpretation definition.

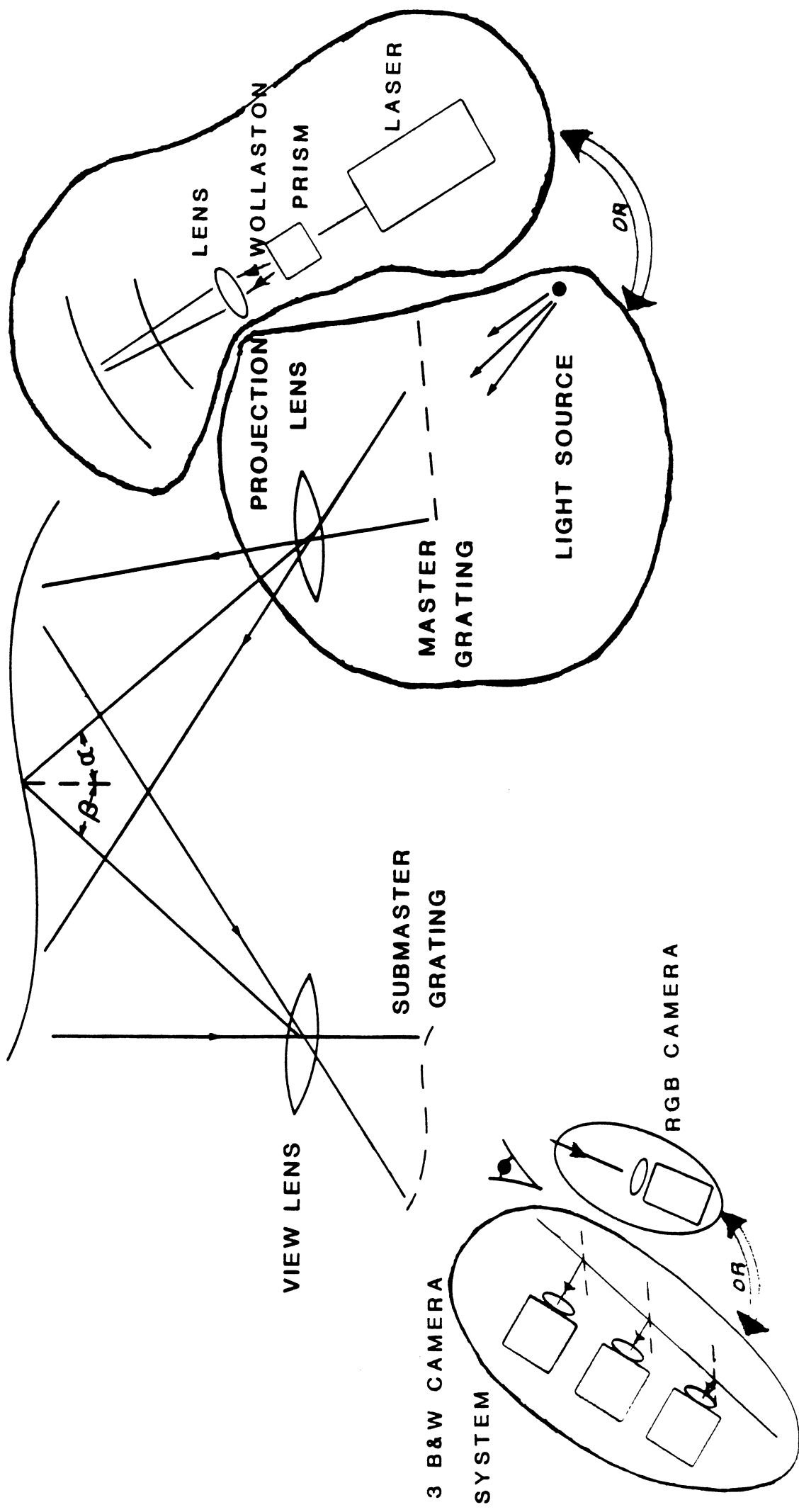


Figure 1. Schematic diagram of moiré configuration options for color encoding.

