

**Simulating Construction Robot Agents
and Their Knowledge Environment**

Boyd Paulson, Jr., Nadeem Babar,
Lai-Heng Chua and Thomas Froese

**CIFE Working Paper
Number 3**

February 1990

Stanford University

Copyright © 1990 by
Center for Integrated Facility Engineering

If you would like to contact the authors please write to:

*c/o CIFE, Civil Engineering,
Stanford University,
Terman Engineering Center
Mail Code: 4020
Stanford, CA 95305-4020*

SIMULATING CONSTRUCTION ROBOT AGENTS AND THEIR KNOWLEDGE ENVIRONMENT

By Boyd C. Paulson, Jr.¹, M.ASCE, Nadeem Babar²,
Lai-Heng Chua³ and Thomas Froese⁴

Abstract

In future construction environments, intelligent machines must harness considerable knowledge to plan and control autonomous tasks in spite of the fact that they will be limited in their own knowledge. However, no unifying theory and few guidelines exist for defining and communicating knowledge about designs and field operations in a way that can effectively be utilized by such machines. This paper first describes research needed to discover and formulate a general core of theory and software for such machines and the knowledge in their environment. For researchers interested in the cognitive aspects of construction robotics, it next surveys current research to simulate characteristics of robot agents themselves, simulate their knowledge environment, and develop means of communication. Finally, it suggests avenues researchers might pursue to better integrate research in construction automation. The main benefit of this research will be to provide a theoretical base for the knowledge environment to sustain automation research for autonomous robots working in real and very challenging field conditions.

Introduction and Background

In future distributed field production environments such as construction, intelligent machines, like their human counterparts, will need to harness considerable knowledge to plan and control autonomous tasks in spite of the fact that they will be limited in their own knowledge and abilities. However, no unifying theory and few guidelines exist for defining and communicating knowledge about designs and field operations in a way that can effectively be utilized by such machines. A good base of theory should be established before much effort is spent building isolated robots of limited scope and flexibility. At Stanford and elsewhere, researchers are working toward such a theory in order to support the work of others on more practical applications of robotics in field conditions. Their research is focusing on cognitive aspects of future machines, thus to enable these robots to have common modes of “thinking” and communicating—in effect a common *culture*—so that they can work together and with humans in groups.

The broad goals of this paper are (1) to define the scope of research needed to support a higher level of robotics integration in construction; (2) to review current research that supports more intelligent robotics; (3) to suggest avenues for research that might be undertaken in

¹Professor, Dept. of Civil Engineering, Stanford University, Stanford, CA 94305-4020.

²Research Assistant, Dept. of Civil Engineering, Stanford University, Stanford, CA 94305-4020.

³Research Assistant, Dept. of Civil Engineering, Stanford University, Stanford, CA 94305-4020.

⁴Research Assistant, Dept. of Civil Engineering, Stanford University, Stanford, CA 94305-4020.

construction automation; and (4) to challenge construction robotics researchers to look at the broad theoretical and software needs of the field to better integrate the current hardware-oriented robotics research activities that are already making some progress in construction.

Research to Support Construction Robotics

The scope of the research needed to build the theories and software to support construction robotics is vast. In general, researchers need to develop machine agents having enhanced abilities to work well in construction. Each step in this research should lead to developing theoretical principles, algorithms or software to be woven into an architecture handling the knowledge an agent needs to function productively in a knowledge environment. The resulting software could then be embedded in robot agents specialized to handle particular areas of expertise, whether in managing other machines or in doing specific physical tasks, while retaining the essence of their ability to deal with unexpected situations and a changing environment. Initially, research is needed to build a knowledge environment simulator in which to test evolving agent software; to simulate characteristics of robot agents themselves; and to develop means of communication. These components would provide a platform for more rapid development of integrated robotic systems.

Simulation of the Knowledge Environment. As an intermediate step we need to create *software*-based simulators of the construction knowledge environment to test concepts and methods as they evolve. Figure 1 shows a broad conceptual view of the environment. It illustrates the organizational context in which the robots might be working, the interfaces to computer-aided design (CAD) databases and reasoning, interactions with other field agents—both human and machine—and interfaces to knowledge sources in the world beyond the field. To establish such an environment with robotic *hardware* is not feasible now. Even if one could afford them, robots with sufficient flexibility and computing power for diverse operations do not yet exist.

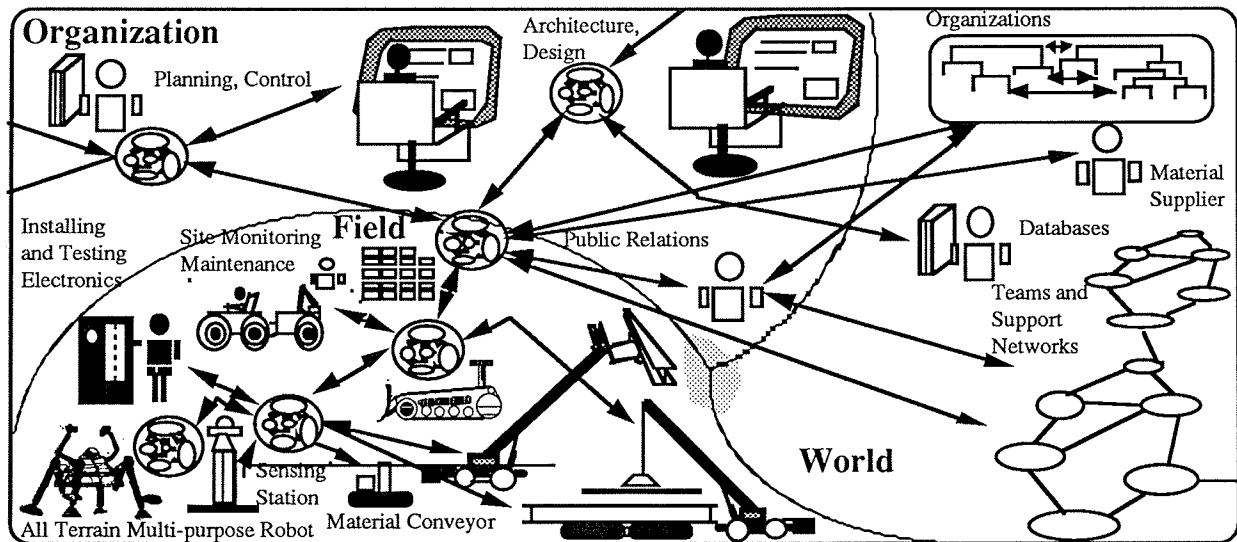


Figure 1—Conceptualization of the Knowledge Environment for Construction Automation

To create prototype testbeds, researchers are using advanced programming methodologies to develop experimental simulators. Some work focuses on the representation of the physical environment and provision of inter-agent communication facilities for multiple agents working in the environment. Early stages of the research also seek to model three-dimensional space in a way that is compatible with the three-dimensional CAD systems now becoming the basis for design communications. Simulators can also implement basic physical mechanics to monitor agent motion and location, provide minimal representations to convert design specifications into incremental task objectives for robots, and implement a type of organizational hierarchy (e.g., a central supervisor agent to which task agents can turn for help). Later stages of research can proceed toward more sophisticated testbeds for the evolving capabilities of intelligent machine agents.

Simulating Robot Agents. Other avenues of research should address problems of modelling *within* robot agent simulators some “understanding” of the knowledge environment, such as key characteristics of objects and other agents, in ways useful for reasoning. Figure 2 illustrates typical questions asked by agents either to other agents or to databases in the knowledge environment. Research will enhance such machines’ capabilities to deal with the knowledge environment, and thus lead to improved integration and robustness of automation systems.

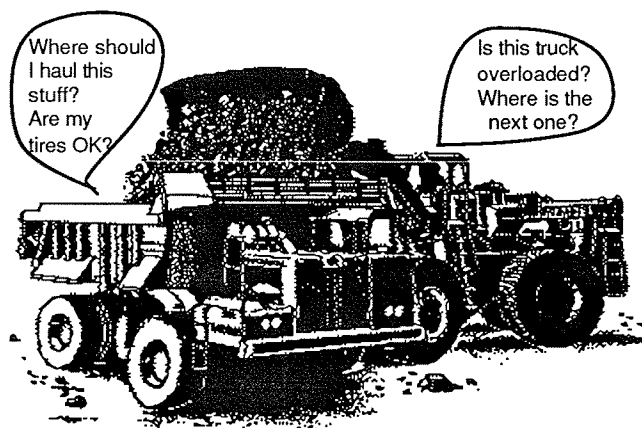


Figure 2—Typical Questions Posed by Machines in the Knowledge Environment

Among other things, researchers should seek to reduce the knowledge that needs to be encoded in machine systems *a priori* by enabling them to tap the vast knowledge sources in their environment when needed. Automata should be able to assemble knowledge and enlist other agents needed to perform a task and respond dynamically to change. For example, robot reasoning and control software should deal with unexpected obstacles, road conditions, failure of a machine positioning system, damaged material, improper tools, or imprecise instructions.

Figure 3 illustrates a machine seeking external knowledge in spheres of increasingly difficult reachability. The nucleus of the figure shows the knowledge and data that might be pre-programmed within the robot itself and thus be immediately accessible to it. The next layer out could be the knowledge and data available nearby from supervisors or other agents, or from computer databases or expert systems on-site. The next layer might still be within the robot’s organization—perhaps an expert at the home office. Beyond that could be the world at large—material suppliers, equipment manufacturers, consultants and public databases. The concept of

graded reachability in the figure implies that it becomes more and more difficult and time-consuming to tap data and knowledge further and further from the robot agent's knowledge core, but the agent still should have enough general knowledge and communications abilities to (a) recognize the need for knowledge beyond its own sphere, and (b) to know at least where to begin, whom to ask, or how to obtain the knowledge it needs to perform or continue its tasks.

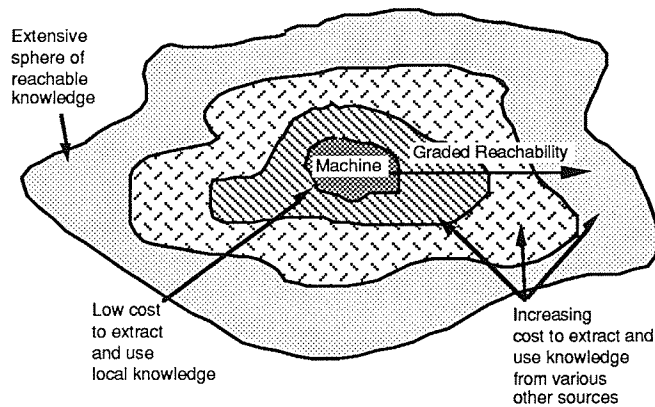


Figure 3—Accessibility of Knowledge in the Environment for Construction Automation

Robots that can respond intelligently to unexpected situations in the field must be aware of their own capacity and the limits thereon, even if only to seek help from a human. *Self-knowledge* can help a robot determine its capability to perform tasks safely and efficiently [Fagin and Halpern, 85, 86]. For example, typical questions for a compactor robot are shown in Figure 4. Self-knowledge is also essential if a robot works together with other machines or human counterparts. Robot self-knowledge should be accessible and interpretable not only by the robot itself, but also by a central knowledge manager. The former will enable the robot to evaluate its own capability, while the latter is important when the robot needs help from the manager. Research should focus on designing a knowledge base for self-knowledge which is flexible, efficient and compatible with the general language design and communication protocols presented in the next section.

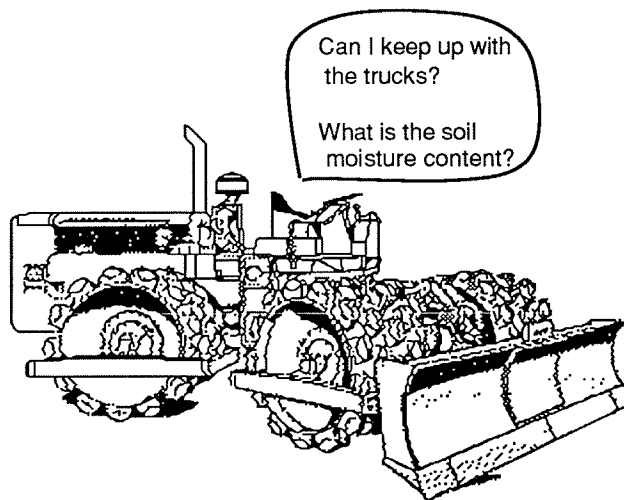


Figure 4—Questions of Self-Knowledge

Language and Communication. With this new breed of intelligent machines in the making, the issues of language and communication need to be addressed soon. Machines must evolve from isolated states to cooperative states, either directly or hierarchically, as shown in Figure 5. Initial efforts should be directed towards the theoretical core of languages: syntax, semantics, and pragmatics. The objectives will be to identify the communication requirements between autonomous robotic agents working in a construction knowledge environment, to define the theoretical basis for their language, and to design their communication systems.

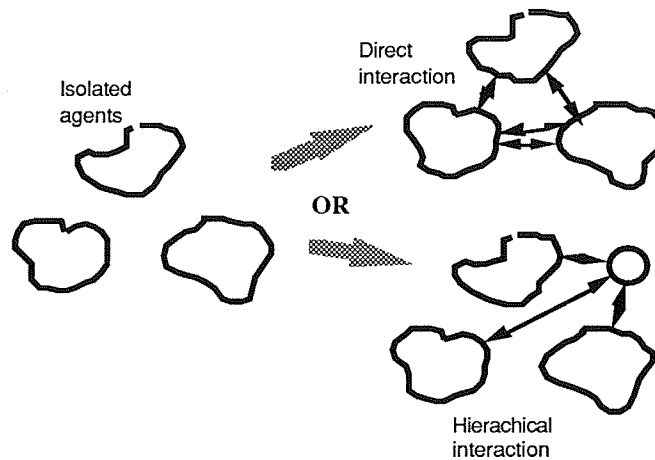


Figure 5—Organizations for Communications and Coordination

One complexity in language design is that at any given time different tasks require different levels of knowledge representation. As an example, consider two robots working together in the construction environment of Figure 6, where one positions a window frame in a wall while the other secures it with bolts. For this task, they need to communicate with each other about the geometry and the exact coordinates of the frame. A syntactic representation with built-in words for geometric shapes and different types of motion will be more efficient in this case. Now assume that Robot 2 wishes to update a database, stored on a central computer, regarding the window just installed. In this instance, a conventional functional language with descriptive codes and numeric quantities may be enough. Finally, assume that the frame is damaged during installation and the agents need to inform their human supervisor. They do not need to convey the graphical representation or the precise coordinates, just a general idea of what has happened, perhaps in a form more like natural language.

Another issue related to the design of a language is the *level* of implementation. Should a robot have enough background knowledge to infer missing parts of a discourse commonly left out in human conversation? Or should it be given complete and detailed instruction at every step? For example, if a human tells a robot driving a truck that one of the tires is flat, should the robot know about flat tires and thus know to stop, or should it require explicit instructions? The choice of a high-level implementation with background-knowledge may seem obvious, but, considering the dynamic nature of field project sites, providing the robots with even a fraction of this knowledge would be a major undertaking, and supplying all the knowledge is impossible.

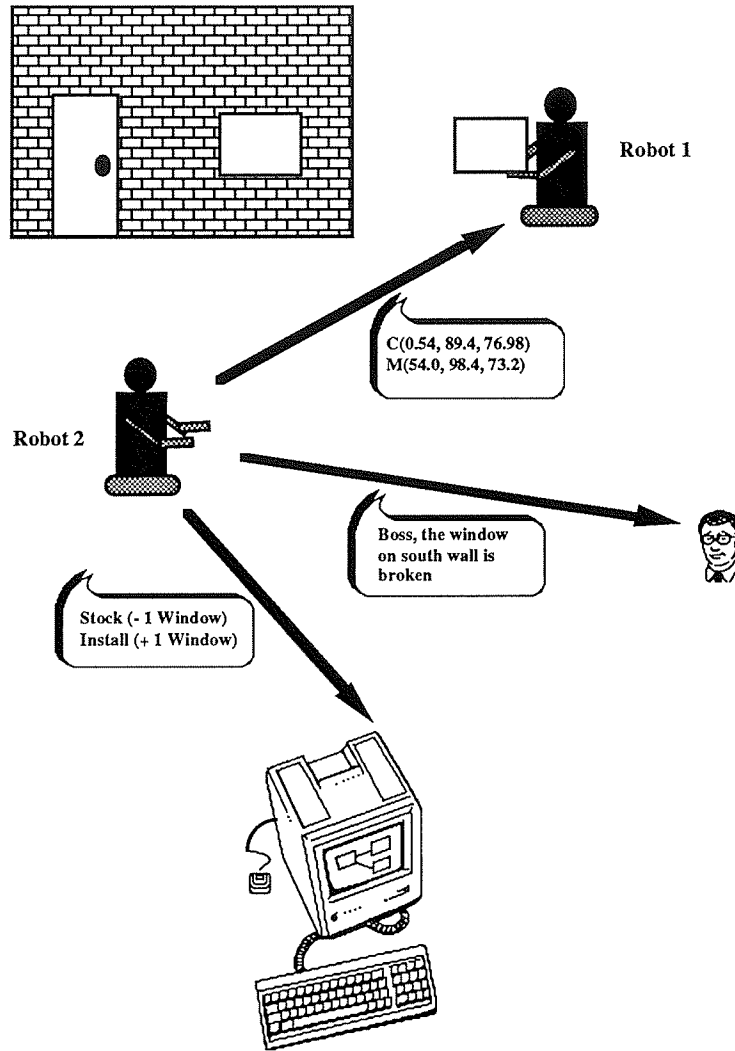


Figure 6—The Different Representational Needs of Different Audiences

Related and Supporting Research

Related research occurs in several main categories: (1) field-based automation and robotics; (2) general theories for simulating the knowledge environment and knowledge agents; (3) implementation methodologies; (4) distributed processing systems and concurrent programming; and (5) artificial robot languages. This survey will enable readers interested in construction robotics to better understand what others are doing.

Field Production Automation. Some equipment manufacturers and contractors are exploring and beginning to implement programmed sensing and control for their earthmovers, cranes, tunneling machines and other equipment. The Japanese are particularly active, but so are some Americans and Europeans. The best source of information about construction robotics is the annual proceedings of the International Symposiums on Automation and Robotics in Construction

(held at Carnegie-Mellon University in 1984 and 1985, Marseilles, France in 1986, Haifa, Israel in 1987, Tokyo, Japan in 1988 and Burlingame, California in 1989). Typical of research applications in the mining industries are [Hagenbuch 88, and White and Zoschke 87]. For the most part we view the current situation in industry as one where developers are following the lines of low resistance, exploiting the small portions of technology that improve current methods without addressing the need to better integrate the automation of field processes.

More advanced prototype applications are also being explored in academia, particularly at Carnegie-Mellon University [Fenves et al 85, Skibniewski and Hendrickson 88, Whittaker 86, Whittaker and Bandari 86], and at MIT [Slocum 86a, 86b], Maryland [Abraham and Bernold 88], Texas [O'Connor and Fisher, 88], Purdue [Skibniewski 88], Stanford [Chan and Paulson 87, 88, Paulson 84, 85, Paulson and Sotoodeh-Khoo 87], and elsewhere [Warzawski and Sangrey 85]. Research on automated vehicles has been supported by DARPA [Thorpe et al 88, Turk 88]. It is not clear yet what this type of automation means for the design of the tasks for such machines.

The research most closely related to that described here is that by [Keirouz et al 86, 88] at Carnegie-Mellon University and by [Paulson and Chua 88a, 88b] at Stanford University. Both are using Smalltalk-based object-oriented modeling of the temporal, geometric and functional evolution of constructed facilities. As reported, the Carnegie-Mellon researchers have focused mainly on the constructed facilities themselves—implying that the models might also be embedded in or accessed by robots. The work at Stanford is in many ways complementary because it focuses on the robot agents themselves, including the means for such agents to recognize and access domain data and knowledge such as that modeled by the Carnegie-Mellon researchers.

Theories for Simulating Robot Agents and their Environment. The scope of this inter-disciplinary research is vast, and only selected topics will be highlighted here, mainly in logic theory and methodologies in artificial intelligence and closely allied fields.

[Genesereth 83] proposed a method of using *first-order predicate calculus* to reason about statements in first-order logic and the reasoning mechanisms invoked. His method can describe the beliefs of other agents and their reasoning as well. [Konolige 80, 85 and Nilsson 83] studied first-order models of belief in which an agent can reason rather than resort to modal operators. These approaches model the *knowledge* of an agent rather than the agent per se.

Relevant experimental artificial intelligence areas include *blackboards* [Hayes-Roth 85, Nii 86a, 86b], *general problem solving architecture* [Laird et al 87] and *natural language processing* [Grosz 86, Hendrix et al 86, Hewitt 71], especially that of *discourse* [Brady and Berwick 83]. They are relevant to construction robotics research because language is a medium of encoding and communicating knowledge. The blackboard model started as a model of knowledge encoding for opportunistic problem solving [Erman et al 81].

[Lenat 75] proposed a model for *problem solving* by a number of “experts” and continued general research on heuristics [Lenat 80,83]. [Minsky 86] dealt in more depth on the issues of problem solving by a number of simple agents grouped into increasingly complex agencies. He also dealt with issues of knowledge representation and use, including frames and K-lines—a knowledge architecture of interest to construction robotics.

[Kellogg 83, Kogan 84 and Koo 87] proposed construction of *intelligent assistants* for information management. [Euzenat et al 85] looked at intelligent database access—that is,

providing an intelligent interface for databases rather than just a natural language one. In line with this has been work on *deductive databases* [Gallaire et al 85]. [Ingwersen 86, Lebowitz 86, and Shaw and Gaines 86] took it further towards an *intelligent information system*. [Schoppers 87] considered it an abuse to regard symbols as representing aspects of an agent's environment.

Supporting Development Methodologies. Smalltalk-80 [Goldberg and Robson 83] has been the main object-oriented programming language. Since inception it has been used for a wide range of applications from AI type systems to ordinary algorithmic programming. MacApp is an object-oriented applications Pascal extension for the Macintosh [Schmucker 86]. A framework with Smalltalk classes has been prepared for use as a front-end to MacApp for application development. Other important languages or extensions in this area include C++ [Stroustrup 86] and Objective C (The Stepstone Corporation). The latter is central to the new NeXT computer [Thompson and Baran 88, Tucker 88] which has been much touted as a workstation suited for object-oriented programming.

Several other developments contain ideas related to this research. ThingLab [Borning 79,81; Farrah and Borning 86], a constraint-based language developed at the Xerox Palo Alto Research Center, can be used to build simulations of real-world systems. Alternate Reality Kit [Smith 86] is a simulation kit for interactive physics. Another simulator written in Smalltalk, called INSIST [Meulen 87], is also applicable. ACTRA [LaLonde et al 86] is a distributed object-oriented computer system based on Smalltalk and targeted for industrial applications such as flexible manufacturing, simulation, control, CAD/CAM and project management.

MACE [Gasser et al 87], an acronym for *Multi-Agent Computing Environment*, is a simulation environment designed to serve as an *instrumented* testbed for building experimental distributed artificial intelligence systems even at different levels of granularity. MACE has powerful constructs for describing agents, their organizational structure and interaction and could provide a useful testbed for construction robotics research.

Distributed Processing Systems and Concurrent Programming. *Distributed systems models* such as ACTORS [Agha 86, Hewitt et al 75] and distributed databases [Howard and Rehak 87, Rehak and Howard 85] relate to this research. There has been significant work concerned with distributed problem solving [Durfee and Lesser 87, Georgeff 82, Konolige and Nilsson 80]. [Davis and Smith 83, Lesser and Corkill 81, 83, and Koo 87] studied cooperative planning and distributed problem solving. [Corkill and Lesser 83] used the monitoring of distributed vehicles as a testbed for studying distributed problem solving. These researchers are more concerned about *global properties or behavior* of the system design while robot agents mainly need the *knowledge* to do their work and get around in their existing knowledge environment. [Shin and Epstein 87] discuss an integrated multi-robot system (IMRS) and its use of concurrent programming to achieve the communication and synchronization needs. Depending on the nature of the process, the communication needs, the control structure and the language will be different. Construction automation researchers must examine typical field production processes to identify particular communication needs and their effect on the design of the language.

Robot Programming Languages. Almost all current robot programming languages are oriented towards manufacturing automation. A comparative study of these languages by [Bonner and Shin 82] categorizes them into four groups. The lowest level is the point-to-point languages like IBM's *Funky* [Grossman 77] and *T3* [Tarvin 80], in which every movement has to be

explicitly programmed. The second level is simple languages like *VAL* [Unimation Inc. 80] and *RCL* [Shin et al 82] in which motion can be specified using joint coordinates. Third is structured languages like *AL* [Mujtaba 79] and *PAL* [Takasa et al 81] which permit coordinate transformations and use state variables to represent certain physical quantities. Highest is the task-oriented languages like IBM's *AUTOPASS* [Lieberman and Wesley 77] which have limited capability for describing some manufacturing primitives like, "Pick up a bolt, insert it into a hole."

The working group of the NATO workshop on Robot Programming Languages expressed the dissatisfaction of programmers about current robot programming languages [Volz 88]. Three major shortcomings mentioned are: low level of abstraction, non-repeatable results from executing a robot program because of external factors, and the lack of many language features which are standard in modern computer languages, such as primitive and abstract data types, control structures, information hiding, separate compilation and improved debugging.

More general developmental software includes object-oriented languages (Smalltalk, C++), procedural and functional languages (Pascal, C, Lisp), and logic languages (Prolog). Using more than one language for the overall system should be given serious consideration.

Since natural language is full of ambiguities and anomalies, the design of a systematic high-level computer language is extremely difficult [Winograd 83]. Some researchers suggest that since syntactic processing must be done to understand the meaning of a sentence, it may be more efficient to also use what may be called semantic parsing [Schank and Childers 84]. Using "scripts" or "plans" associated with key words can speed up processing and reduce ambiguity, but may become extremely cumbersome. To reduce the ambiguity in processing, [Pustejovsky 87] suggests that in designing the structure of a new language, a model of lexical semantics can help constrain word types, just as grammar constrains sentence types. Using such a model, which would be determined by the particular knowledge domain, we eventually might approach natural language constructs without compromising much clarity and efficiency.

Avenues for Research

By this point a reviewer who is familiar with current research in artificial intelligence and robotics will appreciate that the objectives described earlier in this paper represent a most ambitious program of research. Furthermore, there has been slow progress to date in supporting research fields. We are quite well aware of the obstacles and difficulties. Working on various parts of the research with microcomputers such as the Macintosh II, and even with excellent development environments like the ParcPlace Smalltalk-80 system, the Macintosh Programmers Workbench and its object-oriented extensions, and some good implementations of Prolog and Lisp, we indeed have found our progress to be frustrating at times. Nonetheless we have seen enough to appreciate not only the needs but the possibilities for such research. Based on this experience we would like to suggest a few places where interested researchers might get a foothold. Some are areas in which we have been actively working, but even here we have barely scratched the surface. Others are just ideas that we have had too little time to pursue. We wish to encourage concurrent research efforts by others toward better integrating automation and robotics into construction.

Knowledge Environment Simulator. Research is needed to explore and test concepts and programming methods to simulate and help formalize parts of the construction

knowledge environment in machine-usable representations. As a first step in this process, we ourselves have focused on two object-oriented programming environments, Smalltalk-80 and MacApp, and also Prolog to model elements of metaknowledge and communications associated with hierarchical and distributed field robotic environments. In general, research in this area must help to identify the possible system components and interfaces to the knowledge environment of construction automation that desirably should be reflected in a simulator. Priorities must be developed so that simulators can be built in stages and modules.

To focus our own effort, implementation of several *specific scenarios* is being undertaken. We will analyze the knowledge content, the dynamics and structure of each scenario's elements, the participants and the surrounding conditions in which these are embedded. Each scenario will contain the basic guidelines for developing a simulation. What results from this investigation will be language and graphical constructs and simulation algorithms suitable for programming.

To illustrate a possible scenario, Figure 7 embeds instantiations of a software agent into a simulation environment built to test it. This situation has a flat floor except for a 3-inch step discontinuity. Agent 1 is unable to move the object to its destination without the help of Agents 2 and 3 or by some other means. What kinds of knowledge does Agent 1 need to have, what reasoning does it have to perform and what actions does it need to take to achieve its task? What kinds of behavior should Agents 2 and 3 exhibit to enable cooperation to occur? A simulation environment based on such scenarios is one that could create situations with many kinds of difficulties and also provide many opportunities for their solution. One could then explore how well an agent could work in this environment even though it is only a simplified model of the real world. Seeking human help should not be excluded from a machine's strategy, but rather should very much be part of the reason for modelling knowledgeable agents in a machine.

These scenarios actually fall into a class of problems requiring a common problem solving *behavior*. There are strategies that a machine can apply over and over again to solve a wide variety of problems. It is a challenge, however, to represent and be able to use this kind of generic knowledge. Rather than concentrate on single scenarios which one can tailor-make an agent to solve, we plan to develop the environment simulator where it can create unexpected scenarios for the agents to be built. We do not expect our agents to be able to succeed over all scenarios, but we hope that they will function sufficiently well over a fairly broad range of them.

An important part of such a knowledge environment will be the communication facilities for the agents. For a start it could be a simple message-passing facility, where a message is basically a list of parameter names and values or a string, but extensions to bit maps and other forms of data should be possible. Clearly even a small simulator is a major undertaking if one deals with a three-dimensional world. One must develop a means for describing the geometry and for representing objects and agents within it. For the operational principles of this world, one could use a constraint-based approach to implement Newtonian laws of physics, conservation laws, spatial interaction laws, and friction. The simulator should further maintain as far as possible the interaction advantages, such as incremental description, like those provided by Smalltalk.

Robot Agent Simulator. Research in this area involves the modelling of knowledge objects and agents and developing means to give agents access to the knowledge environment. Part of the agenda must be to study how machines can break out of their closed knowledge world to the much richer environment beyond. There are both direct and indirect methods of reaching

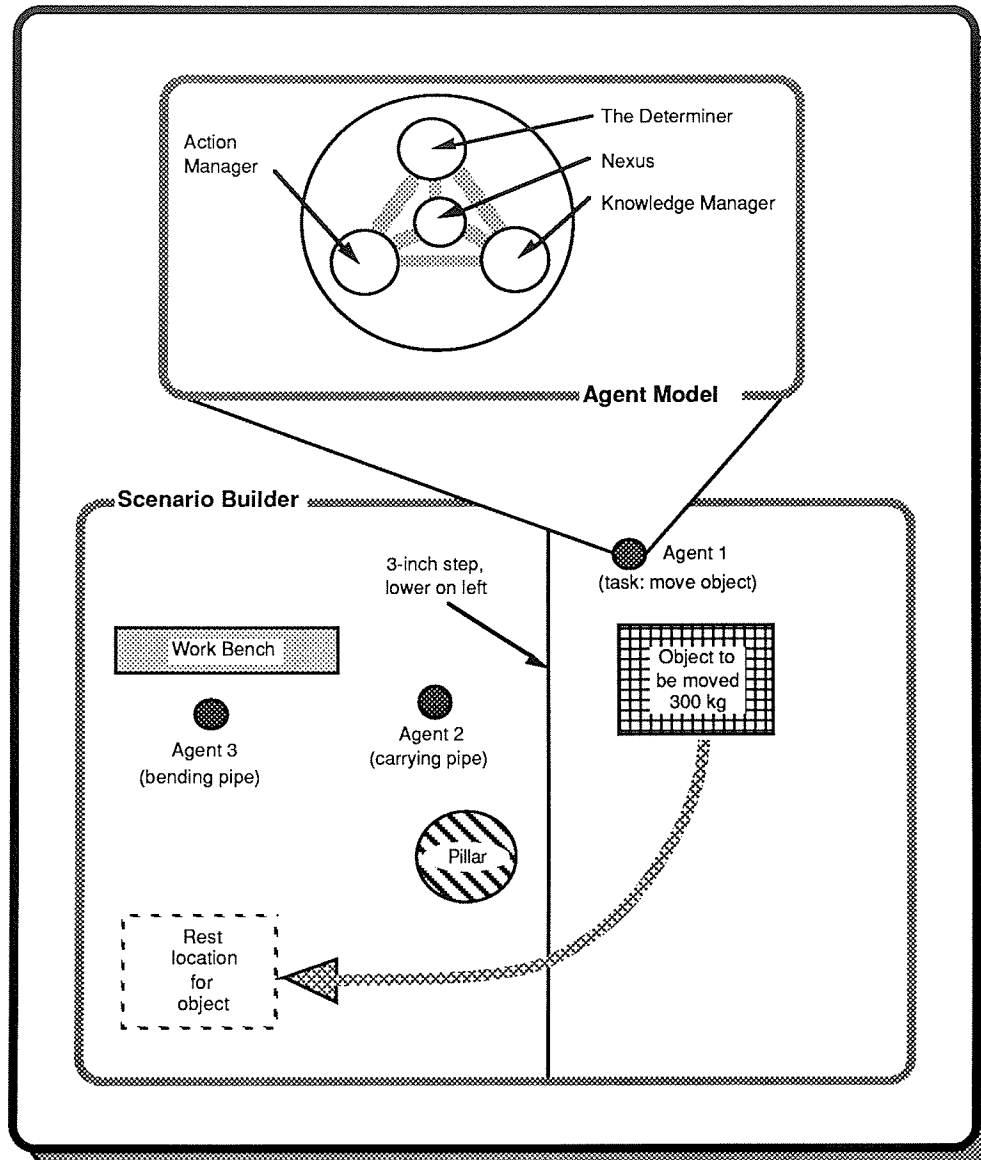


Figure 7—Example Scenario Generated by Knowledge Environment Simulator

and using data and knowledge. They can be obtained directly through fixed sensors or transparent lines to databases. The direct methods can be hard-coded into the system if the environment is stable. More challenging are the abilities to reach data and knowledge indirectly and use the knowledge of other agents. The latter capabilities are important to machines that are to support and organize the work of other machines.

While one normally thinks of a robotic machine as having local knowledge in the form of programmed instructions, machines that deal with complex environments must also have *metaknowledge* (knowledge about the location, structure and use of knowledge) that enables them to recognize and exploit a very much larger sphere of knowledge. Research is needed to learn how to deal with graded reachability concept illustrated in Figure 3.

In the knowledge environment, various knowledge objects can provide an agent with information, assuming the agent knows how to access them and interpret their content. A phone book, for instance, is a useful knowledge object. It is its content, intent and structure which we understand that enables us as agents to obtain the information we desire. Almost all information processing tools are knowledge objects. There are many similarities among knowledge objects and it is the general principles about these objects that are of interest in this research.

Figure 8 describes the problem of a robot's need to find a wrench to tighten a special large octagonal nut. Another agent is nearby that might help if needed. One might write a special rule-based system to tackle this "unexpected" situation, but doing so will only result in a brittle system; the knowledge in the rules could not be effectively used in a slightly different situation.

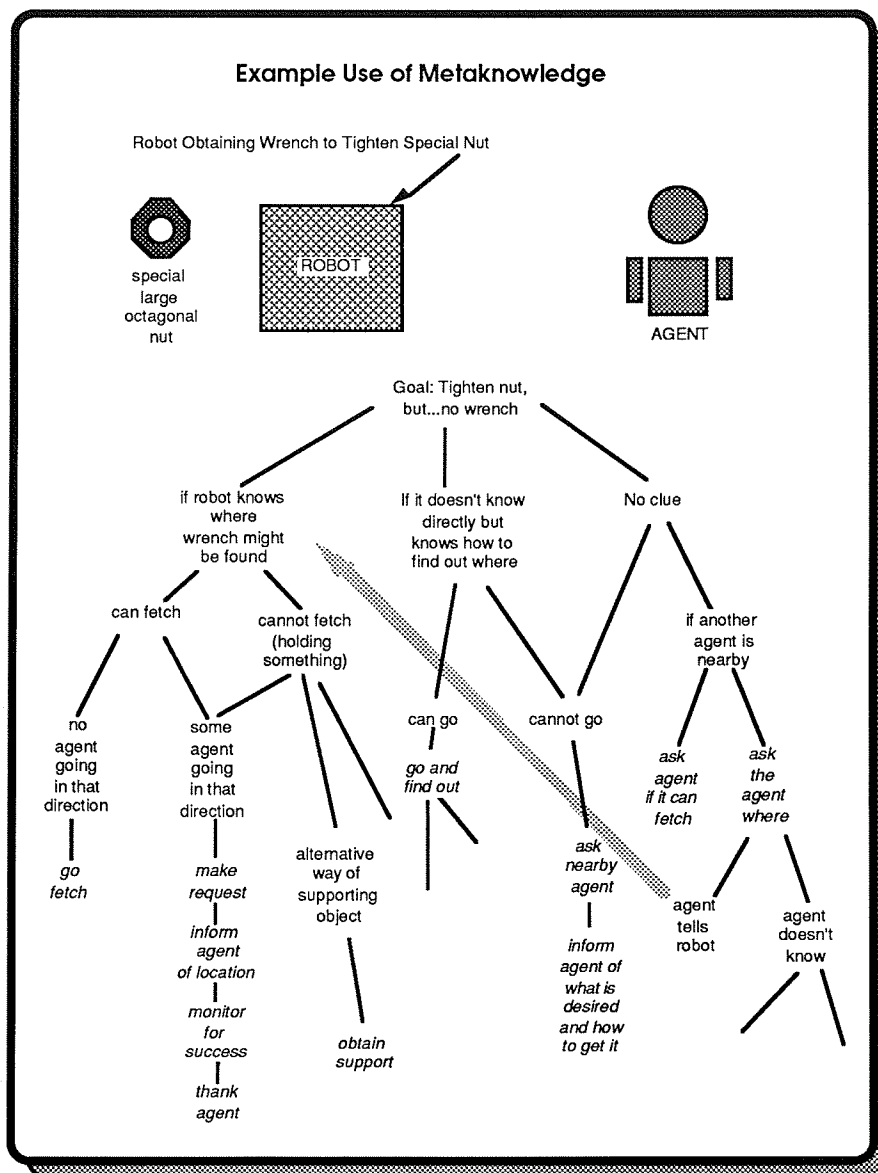


Figure 8—Using Metaknowledge in the Knowledge Environment

Additional research is needed to define the self-knowledge needs and the environmental-knowledge interpretations of the robotic agents. The method of investigation for the environment could involve formulation of knowledge about the tasks and processes in general and suitable tasks in particular. The self-knowledge might be divided into two parts: machine-specific self-knowledge like self-diagnostics and error-recovery; and environment-specific self-knowledge like machine weight, physical dimensions, lifting capacity and available power. The next step would be to match suitable language constructs to the representational needs.

The knowledge environment in which these agents are to be embedded is potentially vast. Any aspect of the environment can be described with great detail. The environment is changing so there may not be accurate correspondence of an agent's knowledge about the environment to its real state at any time. Artificially circumscribing a portion of the environment should take account of the great likelihood that the portion admits a flow of knowledge and agents into and out of it. Researchers must understand in greater depth these and other characteristics of the knowledge environment and how to model and reason about it and its components in a machine agent.

Language and Communications. Three modules, namely study of natural language, knowledge representation and communication requirements, provide input for the overall integrated design of a communications language for construction robots. The starting point will consist of studying the theoretical bases of language and identifying the elements of a given language. Researchers can start by using existing theory in this field. A comparative study of computer languages could identify usable constructs in light of the foundations of robot languages.

Another research thrust should explore communication requirements of autonomous robots for a multi-agent system in general, and the simulated knowledge environment in particular. The strategy might involve identifying the types of communications needed for simulated tasks, given both self and environment knowledge. To identify communication needs, one might consider some typical scenarios. Researchers could classify various types of communication rather than detail specific ideas to be communicated. This grouping would better handle unexpected events in an intelligent way rather than depend on pre-programming for possible contingencies.

Conclusion

This has been a brief overview of a complex fabric of interwoven research needs. Machine agents in complex field environments will have to deal with many difficulties. The only tractable approach in the short term is to limit the use of machines to environments that are—or can be—carefully structured, or to leave most of the control and sensing with human agents. Researchers must look further to develop the underpinnings for more capable machines. We must discover and formulate the general computer-based reasoning and communications core for machines that would provide them with the ability to deal with unexpected events to a greater extent and exploit the opportunities provided by the knowledge environment. In particular, the kind of machines envisaged will be able to deal with uncertainty, adapt to a dynamically changing environment, be able to seek knowledge beyond their spheres, and work in teams to perform complex tasks. The research should also provide for a more robust basis for integration of machines with human organizations, with other machines and with a multitude of tools of production.

Success in developing theory and in building simulators of the knowledge environment and robot agents that work within it, or at least sound prototypes of software foundations upon which they can evolve, should greatly benefit the research of those who follow. The simulators will provide a testbed for concepts, theories and methods for researchers working toward a theoretical base for the knowledge environment to sustain distributed automation. They will further provide computer-based concepts to help integrate construction field processes into the other phases of the project life-cycle, such as planning, design and operations. The ultimate objective is to design and develop the general software core for machine agents — the rudimentary “brains” of the beasts — which can then be specialized for particular tasks. They should have the basic construction automation *cultural* knowledge to enable them to communicate and work together with other agents—both human and machine—in their knowledge environment.

Acknowledgement

For this research we are fortunate to be part of the Center for Integrated Facility Engineering (CIFE) at Stanford, which is operated jointly by the Departments of Civil Engineering and Computer Science and has corporate members from the design, construction and computer industries and representatives from owners and government agencies. Computer-Science researchers Barbara Hayes-Roth, Jean-Claude Latombe and Yoav Shoam have contributed directly or indirectly to our understanding for these research efforts. In addition to the CIFE support, we are indebted to The National Science Foundation, which, through a series of grants (including MSM 83-12350, MSM 84-06852 and MSM 86-14238) has provided the intellectual freedom to explore the basic research that led to the formulation of the research agenda described herein.

Appendix.—References

- Abraham, D. M., and Bernold, L. E., 1988, “Intelligent Simulation and Emulation for Computer Integrated Construction,” *Proceedings of the 5th Conference on Computing in Civil Engineering*, Alexandria, VA (March 29-31).
- Agha, G. A., 1986. *Actors*, MIT Press, Cambridge, Massachusetts.
- Bonner, S., and Shin, K. G., 1982. A Comparative Study of Robot Languages, *Computer*, (December), pp. 82-96.
- Borning, A., 1981. “The Programming Language Aspects of ThingLab, A Constraint-Oriented Simulation Laboratory,” *ACM Transactions on Programming Languages and Systems*, Oct.
- Brady, M., and Berwick, R. C., 1983. *Computational Models of Discourse*, MIT Press, Cambridge, Massachusetts.
- Chan, Weng-Tat, and Boyd C. Paulson, Jr., 1987. “Exploratory Design Using Constraints,” *Journal of Artificial Intelligence in Engineering Design and Management*, Vol. 1, No. 1, December, pp. 59-71.
- Chan, Weng-Tat, and Boyd C. Paulson, Jr., 1988. “An Integrated Software Environment for Building Design and Construction,” *Proceedings of the Symposium on Microcomputer Knowledge-Based Expert Systems in Civil Engineering*, Hojjat Adeli, Ed., 1988 Spring Convention, ASCE, May 9-11, pp. 188-202.

- Corkill, D. D., and Lesser, V. R., 1983. "The Use of Meta-level Control for Coordination in a Distributed Problem Solving Network," *Proc. 8th International Joint Conference on Artificial Intelligence*, Vol. 2, pp. 748-756.
- Davis, R., and Smith, R. G., 1983. "Negotiation as a Metaphor for Distributed Problem Solving," *Artificial Intelligence Journal*, Vol. 20, pp. 63-109.
- Durfee, E. H. and Lesser, V. R., 1987. "Using Partial Global Plans to Coordinate Distributed Problem Solvers," *Proc. 10th International Joint Conference on Artificial Intelligence*, Vol. 2, pp. 875-883.
- Erman, L. D., London, P. E., and Fickas, S. F., 1981. "The Design and an Example Use of HEARSAY," *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, Los Altos, CA, pp. 409-415.
- Euzenat, B., Normier, B., and Ogonowski, A., 1985. "SAPHIR+RESEDA, A New Approach to Intelligent Data Base Access," *Proc. 9th International Joint Conference on Artificial Intelligence*, Vol. 2, pp. 855-857.
- Fagin, R., and Halpern, J. Y., 1985. "Belief, Awareness, and Limited Reasoning: Preliminary Report," *Proc. 9th International Joint Conference on Artificial Intelligence*, Vol. 1, pp. 491-501.
- Fagin, R., and Halpern, J. Y., 1986. "What Can Machines Know? On the Epistemic Properties of Machines," *AAAI 1986*, pp. 428-434.
- Farrah, T., and Borning, A., 1986. *A User Interface for ThingLab Based on Direct Manipulation of Physical Objects*, Technical Report 86-09-02, Department of Computer Science, University of Washington, Sep.
- Fenves, S. J., Baker, N., and Balash J., 1985. "Expert Systems for Planning Robotic Excavation," *Proceedings of the Second Conference on Robotics in Construction*, Carnegie-Mellon University, Pittsburgh, PA, (Jun. 24-26), pp. 1-30.
- Gallaire, H., Minker, J., and Nicolas, J. M., 1985. "Logic and Databases: A Deductive Approach," *ACM Computing Surveys*, Vol. 16, No. 2, (Jun.), pp. 153-185.
- Gasser, L., Braganza, C. and Herman, N., 1987. "MACE: A Flexible Testbed for Distributed AI Research,"
- Genesereth, M. R., 1983. "An Overview of Meta-level Architecture," *Proceedings of the National Conference on Artificial Intelligence*, (Aug.), pp. 119-124.
- Georgeff, M., 1982. "A Theory of Action for MultiAgent Planning," *Proc. 2nd National Conference on Artificial Intelligence*, AAAI-82, pp. 121-125.
- Goldberg, A., and Robson, D., 1983. *Smalltalk-80: The Language and its Implementation*, Addison-Wesley Publishing Company, Menlo Park.
- Grossman, D. D., 1977. "Programming of a Computer Controlled Industrial Manipulator by Guiding through the Motions," *IBM Research Report RC 6393*, IBM T. J. Watson Research Center, Yorktown Heights, NY
- Grosz, B., 1986. "The Representation and Use of Focus in a System for Understanding Dialogs," *Readings in Natural Language Processing*, Grosz, B. J., Jones, K. S., Webber, B. L., Eds, Morgan Kaufmann Publishers Inc, Los-Altos, CA.

- Hagenbuch, LeRoy, 1988. "Computer Integrated Mobile Equipment Operation (Cimeo) and the Future Implementation of Such in the Earthmoving and Mining Industries," Society of Automotive Engineers, Technical Paper 880798, *Proceedings of the Earthmoving Industry Conference*, Peoria, IL (April 12-14).
- Hayes-Roth, B. et al. 1985. "Blackboard Architecture for Control," *Journal of Artificial Intelligence*, Vol. 26, pp. 251-321.
- Hendrix, G., Sacerdoti, E., Sagalowicz, D., and Slocum, J., 1986. "Developing a Natural Language Interface to Complex Data," *Readings in Natural Language Processing*, Grosz, B. J., Jones, K. S., Webber, B. L., Eds, Morgan Kaufmann Publishers Inc., Los-Altos, California.
- Hewitt, C., 1971. "Procedural semantics - models of procedures and the teaching of procedures," *Natural Language Processing*, Courant Science Symposium, Rustin, Ed., pp. 331-350.
- Hewitt, C., Bishop, P., and Steiger, R., 1975. "A Universal Modular ACTOR-Formalism for Artificial Intelligence," *Proc. 3rd International Joint Conference on Artificial Intelligence*, Vol. 1, pp. 235-245.
- Howard, H. C., and Rehak, D. R., 1987. "KADBASE -- A Prototype Expert System-Database Interface for Integrated CAE Environments," *Proceedings, AAAI-87, Sixth National Conference on Artificial Intelligence*, Seattle, Washington, (July).
- Ingwersen, P., 1986. "Cognitive Analysis and the Role of the Intermediary in Information Retrieval," *Intelligent Information Systems*, progress and prospects, Roy Davis, Ed., Ellis Horwood Limited, pp. 206-237.
- Keirouz, W., Rehak, D., and Oppenheim, I., 1986. "Object-Oriented Domain Modelling of Constructed Facilities for Robotic Operations," *First International Conference on Applications of Artificial Intelligence to Engineering Problems*, Southampton University, U.K., (Apr.).
- Keirouz, W., Rehak, D., and Oppenheim, I., 1988. "Issues in Domain Modelling of Constructed Facilities," *Fifth International Symposium on Robotics in Construction*, Tokyo, Japan, (June 6-8), pp. 341-350.
- Kellogg, C. H., 1983. "Intelligent Assistants for Knowledge and Information Resources Management," *Proc. 8th International Joint Conference on Artificial Intelligence*, Vol. 1, pp. 170-172.
- Kogan, D., 1984. "The Manager's Assistant - An Application of Knowledge Management," *Proceedings of the International Conference on Data Engineering*, Institute of Electrical and Electronics Engineering, IEEE Computer Society, Los Angeles, CA, (Apr.), pp. 592-595.
- Konolige, K., 1980. "A First-Order Formalization of Knowledge and Action for a Multi-Agent Planning System," *Technical Note 232*, SRI International, Menlo Park, California.
- Konolige, K., 1985. "A Computational Theory of Belief Introspection," *Proc. 9th International Joint Conference on Artificial Intelligence*, Vol. 1, pp. 502-508.
- Konolige, K. and Nilsson, N. J., 1980. "Multiple-Agent Planning Systems," *Proc. 1st National Conference on Artificial Intelligence*, AAAI-80, pp. 138-141.
- Koo, Charles C., 1987. *Synchronizing Plans among Intelligent Agents via Communication*, thesis submitted to Stanford University in partial fulfillment of the requirements for the degree of Ph.D., August.

- Laird, J. E., Newell, A., and Rosenbloom, P. S., 1987. "SOAR: An Architecture for General Intelligence," *Artificial Intelligence Journal*, in press.
- LaLonde, W. R., Thomas, D. A. and Johnson, K. 1987, "Smalltalk as a Programming Language for Robotics?" *IEEE International Conference on Robotics and Automation*, Raleigh, North Carolina, Mar.
- Lebowitz, M., 1986. "An Experiment in Intelligent Information Systems: RESEARCHER," *Intelligent Information Systems*, progress and prospects, Roy Davis, Ed., Ellis Horwood Limited, Chichester, West Sussex, England, pp. 127-150.
- Lenat, D. B., 1975. "BEINGS: Knowledge as interacting experts," *Proceedings of the Fourth International Joint Conference on Artificial Intelligence*, pp. 246-255.
- Lenat, D. B., 1980. "The Nature of Heuristics," *Artificial Intelligence Journal*, Vol. 19, pp. 189-249.
- Lenat, D. B., 1983. "EURISKO: A Program That Learns New Heuristics and Domain Concepts, The Nature of Heuristics III: Program Design and Results," *Artificial Intelligence Journal*, Vol. 23, pp. 61-98.
- Lesser, V. R., and Corkill, D. D., 1981. "Functionally Accurate, Cooperative Distributed Systems," *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-11, Vol. 1, (Jan.), pp. 81-96.
- Lesser, V. R., and Corkill, D. D., 1983. "The Distributed Vehicle Monitoring Testbed: A Tool for Investigating Distributed Problem Solving Networks," *AI Magazine*, Vol. 4., pp. 15-33.
- Lieberman, L. I. and Wesley, M. A., 1977. "AUTOPASS: An Automatic Programming System for Computer Controlled Mechanical Assembly," *IBM J. Research and Development*, Vol 21, No. 4, Yorktown Heights, pp. 321 - 333.
- Meulen, P. S. 1987, "INSIST: Interactive Simulation in Smalltalk," *Proceedings of the 1987 OOPSLA Conference*, Orlando, Florida, Oct.
- Minsky, M., 1986. *Society of Mind*, Simon and Schuster, New York.
- Mujtaba, S. M., 1979. "Current Status of the AL Manipulator Programming System," *Proceedings of the 10th International Symposium on Industrial Robots*, Milan, pp. 118 - 127.
- Nii, H. P., 1986a. "Blackboard Systems Part One: The Blackboard Model of Problem Solving and The Evolution of Blackboard Architectures," *AI Magazine*, Vol. 7, No. 2, (Jun.).
- Nii, H. P., 1986b. "Blackboard Systems Part Two: Blackboard Application Systems Conclusion," *AI Magazine*, Vol. 7, No. 3, (Aug.), pp. 82-106.
- Nilsson, M., 1983. "A Logical Model of Knowledge," *Proc. 8th International Joint Conference on Artificial Intelligence*, Vol. 1, pp. 374-376.
- O'Connor, J. T., and Deborah J. Fisher, 1988. "Constructability for Semi-Automated Piping Construction: Research Progress," *Fifth International Symposium on Robotics in Construction*, Tokyo, Japan, (June 6-8), pp. 185-194.
- Paulson, Boyd C., Jr., 1984. "The Potential for Robotics in Construction," Proceedings Paper MS84-1050, *Robot West Conference*, Anaheim, Calif., (Nov. 28-30), Society of Manufacturing Engineers, Dearborn, Michigan.
- Paulson, Boyd C., Jr., 1985. "Automated Control and Robotics for Construction," *Journal of Construction Engineering and Management*, ASCE, Vol. 111, No. 3, (Sep.), pp. 190-207.

- Paulson, Boyd C., Jr., and Lai-Heng Chua, 1988a. "Preliminary Investigation of Object-Oriented Programming to Create a Simulator for the Knowledge Environment of Construction Automation," Stanford University Proposal No. CE 56-88(N), (July).
- Paulson, Boyd C., Jr., and Lai-Heng Chua, 1988b. "Modeling Aspects of the Knowledge Environment in Machine Agents," Stanford University Proposal No. CE 68-88(N), (August).
- Paulson, Boyd C., Jr., and Sotoodeh-Khoo, Hooman, 1987. "Expert Systems in Real-Time Construction Operations," *Managing Construction Worldwide, Volume 1: Systems for Managing Construction*, Proceedings of the 5th International Symposium on The Organization and Management of Construction, CIB W-65, September 7-10, London, pp. 554-565.
- Pustejovsky, J. and Berlgler, S., 1987. "Acquisition of Conceptual Structure for the Lexicon," *Proceedings of the Sixth National Conference on Artificial Intelligence*, (July 13-17), Seattle.
- Rehak, D. R., and Howard, H. C., 1985. "Interfacing Expert Systems with Design Databases in Integrated CAD Systems," *Computer-Aided Design*, (Nov.)
- Schank, R. C. and Childers, P. G., 1984. *The Cognitive Computer*, Addison-Wesley, Menlo Park.
- Schmucker, Kurt J., 1986, *Object-Oriented Programming for the Macintosh*, Hayden Book Co., Hasbrouck Heights, N.J..
- Schoppers, M. J., 1987. "Universal Plans for Reactive Robots in Unpredictable Environments," *Proc. 10th International Joint Conference on Artificial Intelligence*, Vol 2., pp. 1039-1046.
- Shaw, M. L. G., and Gaines, B. R., 1986. "A Cognitive Model for Intelligent Information Systems," *Intelligent Information Systems*, progress and prospects, Roy Davies, Ed., Ellis Horwood Limited, Chichester, West Sussex, England, pp. 238-258.
- Shin, K. G., and Epstein, M., 1987. "Intertask Communications in an Integrated Multirobot System," *IEEE Journal of Robotics and Automation*, Vol. RA-3, no. 2, (April), pp. 90-99.
- Shin, K. G., et, at., 1982, "Development of an Explicit Robot Control Language," *Proceedings of Compsac*, Chicago.
- Skibniewski, M. J., 1988. "Framework for Decision-Making on Implementing Robotics in Construction," *Journal of Computing in Civil Engineering*, ASCE, Vol 2, No. 2, April, pp. 188-201.
- Skibniewski, M. J., and Hendrickson, C. T., 1988. "Analysis of Robotic Surface Finishing Work on Construction Site," *Journal of Construction Engineering and Management*, ASCE, Vol. 114, No. 1, March, pp. 53-68.
- Slocum, A. H., 1986a. "Robot Research," *Civil Engineering at MIT*, Vol. 1, No. 1, Massachusetts Institute of Technology, Cambridge, MA., Autumn.
- Slocum, A. H., 1986b. "Development of an Integrated Construction Automation Methodology," *CAD and Robotics in Architecture and Construction*, Proceedings of the Joint International Conference, Marseilles, France (June 25-27), pp. 133-149.
- Smith, R. B., 1986, "The Alternate Reality Kit: An Animated Environment for Creating Interactive Simulations," *Proceedings of 1986 IEEE Computer Society Workshop of Visual Languages*, Dallas, Texas, Jun.
- Stroustrup, B., 1986. *The C++ Programming Language*, Addison-Wesley, Menlo Park, CA.

- Takasa, K., et. al., 1981. "A Structural Approach to Robot Programming and Teaching," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-11, No. 4, April, pp. 274 - 289.
- Tarvin, R. L., 1980. "Considerations for Off-Line Programming of a Heavy Duty Industrial Robot," *Proceedings of the 10th International Symposium on Industrial Robots*, Washington, D. C., pp. 297 - 307
- Thompson, T. and N. Baran, 1988. "The NeXT Computer," *Byte*, Vol. 13, No. 12, pp. 158-175.
- Thorpe, C., et al, 1988. "Vision and Navigation for the Carnegie-Mellon Navlab," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 3, May, pp. 362-373.
- Tucker, M. J. 1988, "Ready for What's Next?," *Unix World*, (November).
- Turk, M. A., et al, 1988. VITS—A Vision System for Autonomous Land Vehicle Navigation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 3, May, pp. 342-361.
- Unimation Inc., 1980. *Unimate PUMA Robot Manual*, Text 398H1A, Vol. 1
- Volz, R., 1988. "Report of the robot programming language working group: NATO workshop on robot programming languages," *IEEE Journal of Robotics and Automation*, Vol. 4, no. 1, (February) pp 86-91.
- Warzawski, A., and Sangrey, D., 1985. "Robotics in Building Construction," *Journal of Construction Engineering and Management*, ASCE, Vol. 111, No. 3, (Sep.).
- White, J. W., and Zoschke, L. T., 1987. "The Development of a Computerized Truck Dispatching System," *Mining Magazine*, Vol. 157, No. 6, (Dec.).
- Whittaker, W. L., 1986. "Construction Robotics: A Perspective," *Proceedings, International Joint Conference of CAD and Robotics in Architecture and Construction*, Marseilles, France, (Apr.).
- Whittaker, W. L., and Bandari, E., 1986. "A Framework for Integrating Multiple Construction Robots," *Proceedings, International Joint Conference of CAD and Robotics in Architecture and Construction*, Marseilles, France, (Apr.).
- Winograd, T., 1983. *Language as a Cognitive Process*, Addison-Wesley, Menlo Park, California.