CIFE

CENTER FOR INTEGRATED FACILITY ENGINEERING

"Preliminary Investigation of Object-Oriented Programming to Create a Knowledge Environment Simulator for Construction Automation".

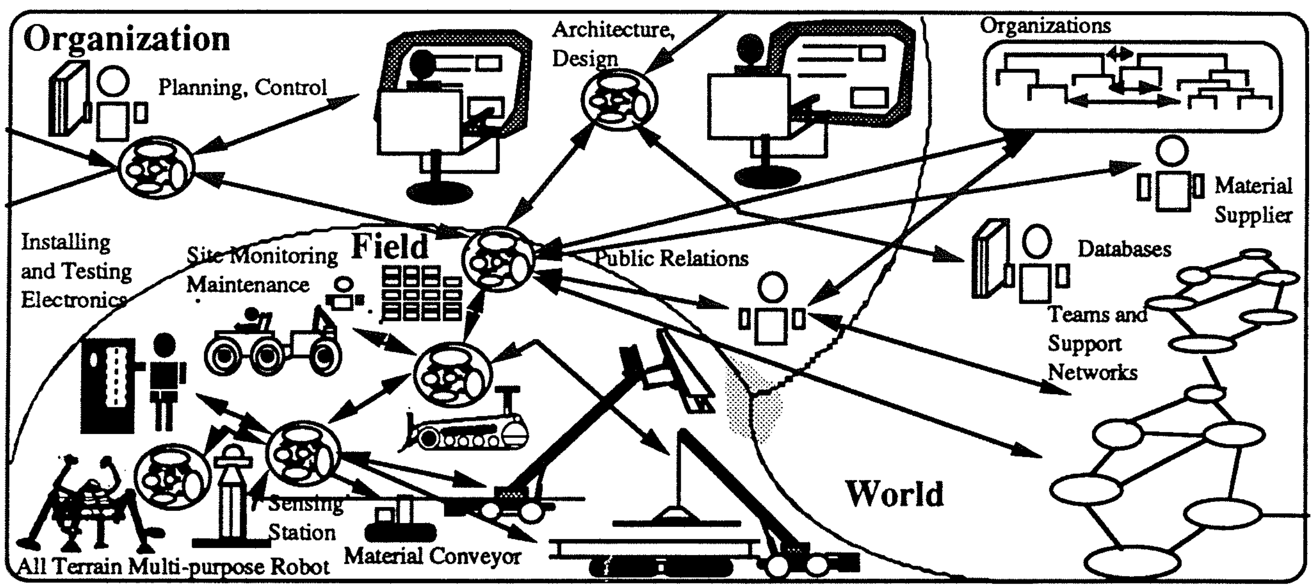
by Boyd Paulson

CIFE WORKING PAPER No.001

STANFORD UNIVERSITY

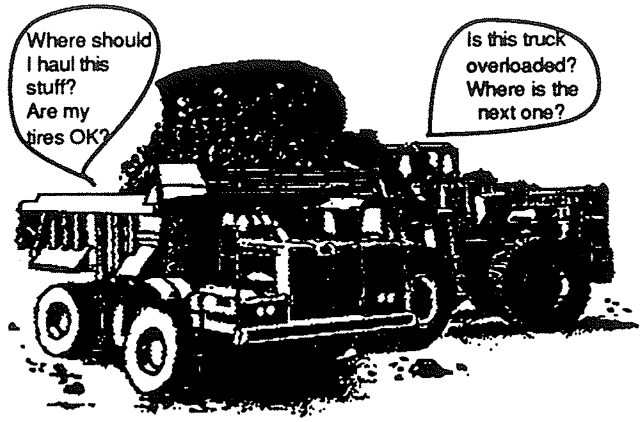
Research in Construction Robotics at Stanford University

In construction, intelligent machines, like their human counterparts, will need to harness considerable knowledge to plan and control autonomous field 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. These are issues of the *knowledge environment.* Here at Stanford, in the Center for Integrated Facilities Engineering, researchers are working toward such a theory in order to support the work of others on more practical applications of robotics in construction. It is a premise of our long-term research that a good base of theory should be established before much effort is spent building isolated robotic devices of limited scope and flexibility. In simpler terms, while others are doing important and highly visible work on the mechanisms that form the "bodies" of robots, and on sensors that enable them to "see", "feel" and "hear," we are focussing on the "brains" of such future machines, and are trying to do so in a fundamental way that can enable these robots to have common modes of "thinking" and communicating so that they can work together—and with humans—in groups.



Simulation of the Knowledge Environment

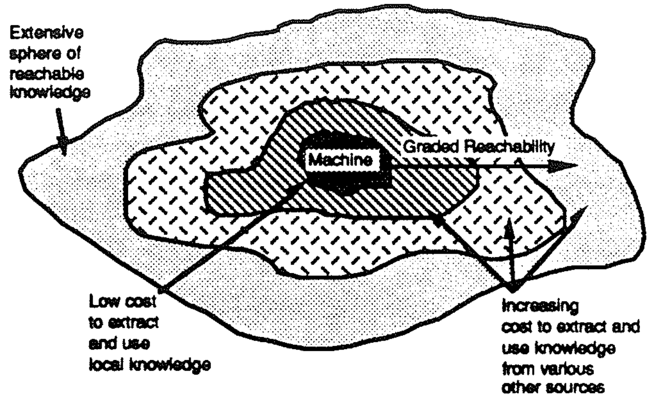
As an intermediate step we are creating *software*-based simulators for testing concepts and methods as they evolve. To establish such an experimental environment with robotic *hardware* is not a feasible alternative at present. Even if we could afford them, robots with sufficient flexibility and computing power for diverse construction operations simply do not yet exist. To create the testbed, we are using object-oriented programming methodologies to develop an experimental simulator of the knowledge environment in which we envision autonomous robots will be working. A broad conceptual view of the environment is shown in in the figure above. Our exploratory research to date indicates that object-oriented programming is among the most promising means to achieve such an objective. While human-oriented extensions such as animated graphical representations may be possible, the initial work on the simulator will focus on the functional simulation of the physical environment and provide an inter-agent communication facility. In the first stage of the project we intend to develop a small system with limited simulation capabilities for a physical and knowledge environment. As an experiment, we hope to be able to simulate some important aspects of construction earthworks. In later stages we will proceed to develop a more sophisticated testbed for the evolving capabilities of intelligent machine agents.



Simulating a Robot Agent

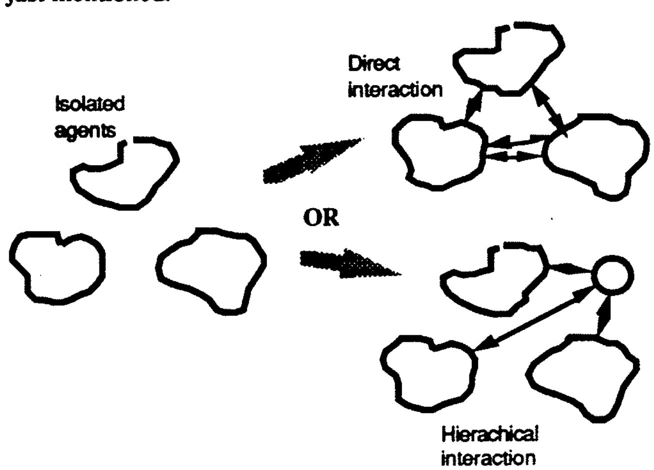
Another path of this research is addressing the problems of modelling *within* robot agent simulators some "understanding" of the knowledge environment, such as knowledge objects and agents, in ways useful for reasoning and also is attempting to deal with key characteristics of the knowledge environment. Typical questions are illustrated in the figure above. Our intention is to enable a machine to reason about objects and about other agents. This research will enhance machines' capabilities to deal with the knowledge environment, and thus lead to improved integration and robustness of automation systems. Such systems will be able to seek and absorb useful knowledge from the knowledge environment beyond their own limited boundaries.

Among other things, we would like to reduce the amount of knowledge that needs to be encoded in machine systems *a priori* but still enable automation systems to reach the vast amount of knowledge available in the knowledge environment. We would also like the automation systems to be able to assemble the knowledge and enlist the other agents that it requires to perform a task. Eventually such machine systems should be able to deal with situations that have not been explicitly pre-programmed in their knowledge, and function effectively in the dynamically changing environment. We are still a long way from achieving this, but when the major portions of the research program are complete answers to the simpler problems should be available. In automated earthmoving, for example, we would like to be able to deal with unexpected situations such as sudden changes in excavated material, haul-road conditions, failure of machine positioning system or incorrect instructions to the machine. In production situations we would like to be able to solve problems of missing or damaged material, improper tools, imprecise instruction, or some new conditions of the site. In the management arena we would like to be able to locate of supplies of material and obtain price and cost information. The next figure gives a conceptual illustration of a machine seeking external knowledge in spheres of increasingly difficult reachability.

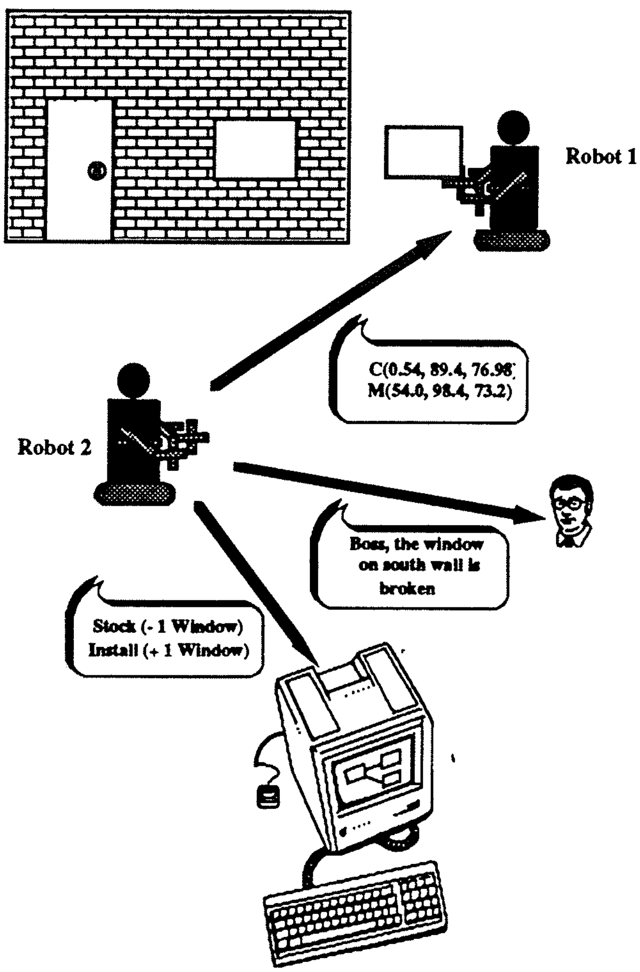


Language and Communication

With the genesis of this new breed of intelligent machines in the making, the issues of language and communication for these machines need to be addressed soon. We need to bring machines from isolated states to cooperative states, either directly or hierarchically, as shown below. Therefore, a third avenue of this research seeks to provide foundations for designing a communication system for autonomous robots operating in a decentralized field knowledge environment. Our initial effort in this project is directed towards the theoretical core of languages: syntax, semantics, and pragmatics. The objectives of this research project are to identify the communication requirements between autonomous robotic agents working in a distributed field knowledge environment, to define the theoretical basis for their language, and to design their communication system. The theoretical foundations of our design, and the evaluation of various programming languages for these needs will be based on the three components of languages just mentioned.



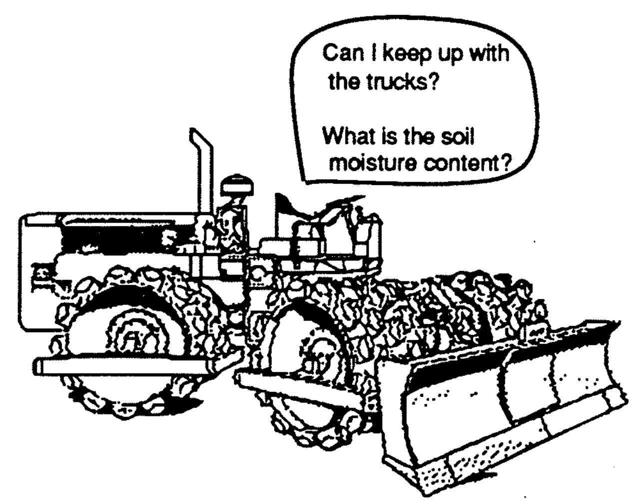
One complexity in language design is that at any given time, different tasks also require different levels of knowledge representation. For example, consider two robots working together in a construction environment (illustrated below) where one of them positions a window frame in a wall while the other secures it. To do this, they will 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 much more efficient in this case. Now assume that one of them wishes to update a database, stored on a central computer, about the window just installed. In this instance, the representational needs are quite different. Finally, assume that the frame is damaged in the process of installation and the agents need to inform the human supervisor about it. They do not need to convey the graphical representation or the precise coordinates, but just a general idea of what has happened. Again, the communication with the human agents is on a different syntactical level, approaching more towards natural language.



Another major issue related to the design of a language is the *level* of implementation. Should the robot be given 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? Consider a situation in which a human tells a robot driving a truck that one of the tires is flat. Should the robot know about flat tires and thus stop the operation, or should it require explicit instructions to stop? The choice of a high-level implementation with considerable background-knowledge may seem obvious, but, considering the dynamic nature of construction sites, providing the robots with even a fraction of this knowledge may be a major undertaking, and supplying all the knowledge is impossible.

Self-Knowledge for Robot Agents

If we are to build autonomous robots that can respond intelligently to unexpected situations in the field, they must fully aware of their own capacity and the limits thereon, even if the response is just seeking help from a human. Maximum *self-knowledge* can help the robot determine if it has the capability to perform various tasks safely and efficiently. Typical questions for a compactor robot are shown below. Self-knowledge is also essential if a robot is to work in conjunction with other machines or its human counterparts. Self-knowledge of a robot should be designed such that it can be accessed and easily interpreted not only by the robot itself, but also by a central knowledge manager. The former will enable the robot to evaluate its capability to perform a certain task, while the latter is important when the robot needs help from the knowledge manager. In this area, our research will focus primarily on designing a knowledge base for self-knowledge which is flexible, efficient, and fully compatible with general language design and communication protocols. The purpose of this part of the research will be the development of specifications, standards, and an overall design of the knowledge base.



Research Program and Benefits

This has been just a brief overview of a complex fabric of interwoven research projects. A summary diagram showing the topics and inter-relationships of the main research projects is shown below. The main benefit of this research will be to serve as a foundation for future projects related to the knowledge environment. It thus might well be considered typical of "ivory-tower" academic research. However, this research will provide a theoretical base for the knowledge environment to sustain automation research for autonomous robots working in real and very challenging field environments. Success in building these simulators, or at least sound prototypes of software foundations upon which they can evolve, should greatly benefit the research of those who follow. The research will provide a testbed for concepts, theories and methods for researchers working toward a theoretical base for the knowledge environment to sustain construction automation. It will further provide computer-based concepts to help integrate the physical construction process into the other phases of the project-life-cycle data and knowledge structures being studied by researchers at Stanford and elsewhere. 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 while retaining the ability to handle unexpected developments, insufficiency of initial knowledge, and a changing environment.

For additional information please contact:

Professor Boyd C. Paulson, Ph.D.
Department of Civil Engineering
Stanford University
Stanford, CA 94305-4020

Phone: (415) 723-2235

E-mail (Internet): bcp@CIVE.STANFORD.EDU

September, 1988