CONTROL SYSTEMS FOR COMPUTER INTEGRATED CONSTRUCTION

Leonhard E. Bernold, Assistant Professor
Dulcy M. Abraham, Graduate Assistant

Department of Civil Engineering, University of Maryland
College Park, MD 20742

ABSTRACT

The concept of Computer Integrated Construction (CIC) creates many challenges in developing an integrated system which links vertically and horizontally all the different participants on a construction project. In addition to the sharing of information, CIC provides new opportunities for construction management which include integrated planing and control. Researchers at the Construction Automation Research Laboratory at the University of Maryland are investigating aspects of automated control in construction and have used decomposition techniques and hierarchical control principles to develop vertically integrated control systems. This paper introduces the basic principles of hierarchical control which is based on state machine concepts and will discuss the use of a Rule Based Control System Emulator (RCSE) which was developed for emulating different control structures.

1. INTRODUCTION

The advent of computer integrated construction makes it necessary to develop models which will enable interfacing of construction system elements of varying levels of sophistication. The authors of this paper have concentrated on studying hierarchical control models for controlling fixed type construction operations and on emulating and simulating production systems such as fabrication of wood trusses.

In the following sections, the paper will first briefly describe the nature and the functions of control systems. Further, the decomposition method and the principles of hierarchical control will be discussed. Finally, a model developed for automated truss fabrication will be presented.

2. INTEGRATED CONTROL FOR CONSTRUCTION

With the marked increase in the use of automation in many areas in construction (e.g., excavation, fireproof spraying, etc.), and the influx of automated methods for performing traditionally manual or mechanical processes (e.g., rebar bending, truss manufacture), different types of equipment with varying degrees of automation, controlled by different methods, have to work cooperatively. This has resulted in a need to have integrated control to ensure the quality of information flow.

Personal computers which until quite recently, had been used at the levels of middle or upper management, mainly for cost control and estimating, now have the capability of processing commands from and feedback to high level controllers, and can be employed for developing and for modifying the programs that are executed by the construction equipment's (e.g., a fireproof spraying robot) on-board computer. For optimal utilization of the capabilities of these microcomputers, they should also be integrated into the overall control system.

3. DECOMPOSITION FOR DESIGN OPTIMIZATION AND CONTROL

The idea of using decomposition for solving large-scale nonlinear systems was first proposed by Kron (3) who proposed the solution of systems with a very large number of variables by breaking the systems into large number of small subdivisions. Dantzig-Wolfe (1) initiated the extensive use of decomposition in mathematical programming, initially for the decomposition of linear programming problems, whose coefficient matrices have an "angular structure." In this method, the original program is decomposed into several linear subprograms and a "master" (coordinating) program. At each iteration, the subprograms receive a set of parameters (simplex prices) from the master program. The subprograms then send their current solutions to the master program which in turn obtains a new set of prices to be sent back to the subprograms. The iterations continue until an optimal solution is obtained.

Most of the decomposition methods reported in the literature are essentially combinations of two different approaches which are called the model coordination method and goal coordination method (3). In the model coordination method, the decomposition is made possible by adding constraints to the mathematical model of the problem in the form of fixing some variables in order to coordinate the activities of the subproblems. The model coordination method is also known as the feasible decomposition method due to feasibility of the intermediate values of variables. This method is particularly attractive from an engineering design point of view, since the iteration process may be terminated whenever it is desirable, with a feasible, even though nonoptimal, result.

In the goal coordination method, the decomposition is made possible by modification of the objective (goal) of the subproblems, while cutting the variables' links between subproblems. The goal coordination method is also known as the dual method, since the upper-level problem is usually the dual of lower-level subproblems.

The decomposition approach has been applied also to construction in the development of the Work Breakdown structure (WBS). Control system designs for flexible production system have also adapted the concept of decomposition. Through modularization and partitioning the complexity of the overall control system is structured in such a way that individual elements of the systems are able to interact according to specific rules. A concept which has been successfully applied for integrated control in a manufacturing environment is hierarchical control.

4. PRINCIPLES OF HIERARCHICAL CONTROL

Hierarchical control systems are constructed using the philosophy of levels of control. Superior/subordinate relationships are created between levels with command data flowing downward in the hierarchy towards machines and processing stations at the lowest levels, and sensory data flowing upward in the hierarchy toward the manufacturing level. Figure 1 shows two levels of control linked via commands and feedback.

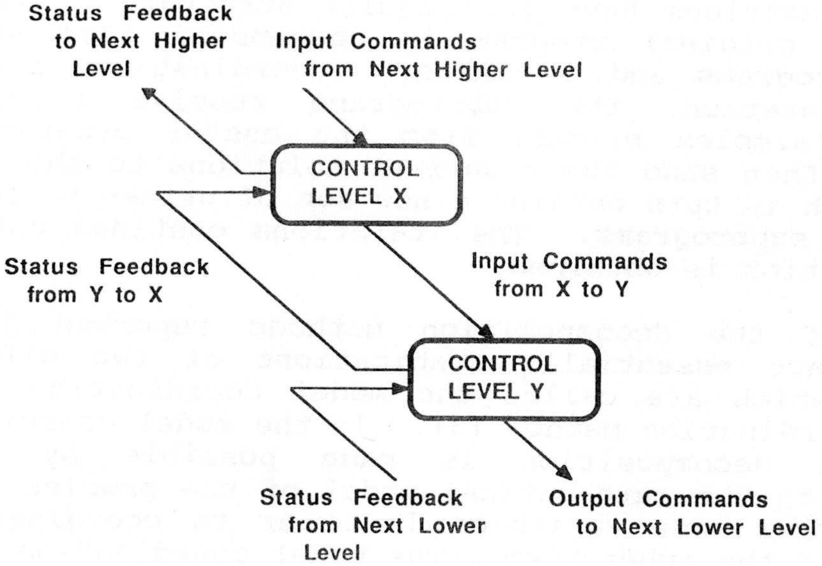

Fig. 1 Control System Hierarchy

Control hierarchies can be grouped into two categories - open and closed. Open hierarchies can have any number of levels, depending on the need of the specific application. The number is not specified in the conceptual framework. As a result, the model cannot bind unique functions to specific levels. Instead they emphasize generic functions that every level must provide. This permits the development of a common software framework for every level, reducing the programming effort needed to implement a control hierarchy. Closed Hierarchies have a fixed number of levels, usually with specific bounds (functions) to each level. They are a hybrid of the control and organizational views.

5. THE STATE TABLE CONCEPT

Commands in a hierarchical system are decomposed into simpler subcommands or subgoals by the task decomposition modules which have the general nature of finite state machines. For each possible set of inputs (goal, sensory data and status), they generate the list of outputs (subgoal, status, and sensor request) that are appropriate.

Hierarchical control systems also contain a world model which is primarily a database which contains both apriori information about the world (e.g., CAD data) and a model of the world based on experience. It is responsible for providing information to the sensory processing module. The world model at each control level must contain information about parts that is relevant to the decision-making at that level. The internal world model is a data model (maintained internally) that reflects the state of the controller and each of its subordinates. The supervisor writes action fields which tell the subordinate how or when to perform the work and reads status fields written by the subordinate. A subordinate reads the action fields and writes status fields which report its progress in carrying out the order. Control messages (commands and status) carry pointers to work order records maintained in the database. A procedure is used by a controller to decompose work order records. All these pieces of information constitute a state table. A state table therefore, consists of condition/action sections. Fig. 2 shows the structure of the state table.

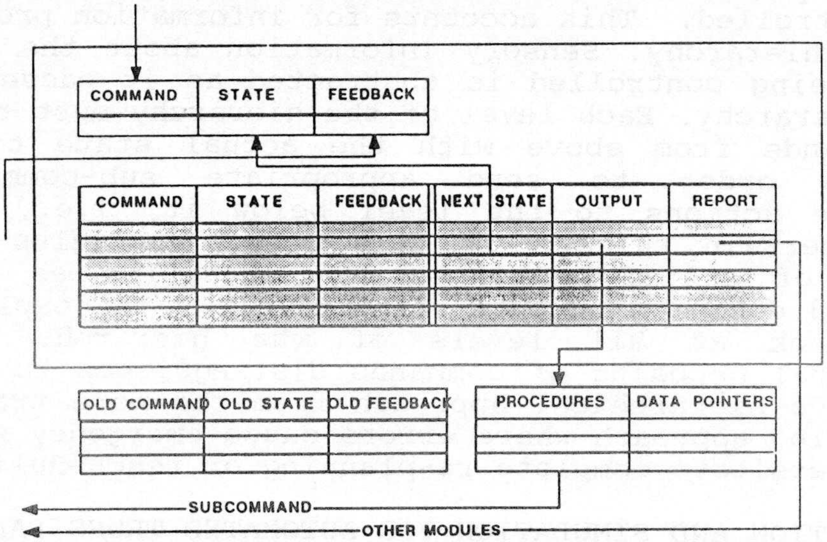

Fig. 2 The State Table Concept

Conditional tests are based on such information as current system state, supervisor's command, subordinate feedback and world model checks. Actions that are defined for each state include: updates to the internal world model, commands to subordinates, feedback to the system's supervisor, and the setting of the state variables. The basic elements used in modeling hierarchical control systems are commands, state variables, and sensory feedback, which are linked as shown in Fig 2.

6. USE OF EMULATION AND SIMULATION IN HIERARCHICAL CONTROL

To assess the possible effects of discrepancies between the design model of the system and the actual one, simulation can be employed. It averts serious errors that can arise from these discrepancies because the control system design can be corrected without physical damage to the actual facility. However, it is not sufficient to merely simulate the input/output relations of the control system because simulation by itself is not sufficient to study the impact of the errors in the designer's conceptual model of the system's logical relationship.

In order to represent the internal logical structure of the control system and its implementation features, a one-for-one representation of the control system logic is desired. This one-for-one simulation is termed emulation (2).

When a command is entered at the top of the hierarchy, it is successively decoded into more detailed instructions at the lower levels, until the lowest levels of the hierarchy provide an interface with the physical processes being controlled. This accounts for information propagation down the hierarchy. Sensory information about the physical process being controlled is abstracted as it passes upward in the hierarchy. Each level of the hierarchy must reconcile its commands from above with the actual state of events below in order to send appropriate sub-commands or corrective actions to the level below it, i.e., it must satisfy certain rules. In this case, the rules and the outcomes of satisfying these are specified as decision inputs and decision outputs respectively. The explicit use of feedback at all levels of the hierarchy and the hierarchical decoding of commands distinguishes the sensory interactive hierarchical approach from the more traditional pre-planning approach where errors cause emergency shutdowns which necessitate complete re-planning or rescheduling (2).

7. EMULATION AND SIMULATION FOR AUTOMATED TRUSS FABRICATION

A Rule Based Control System Emulator (RCSE) was developed at the Construction Automation Research Laboratory at the University of Maryland for testing the hierarchical control principles. RCSE runs in the PC environment. Production of simple roof truss (shown in Fig. 3) has been used as a testbed for demonstration purposes.

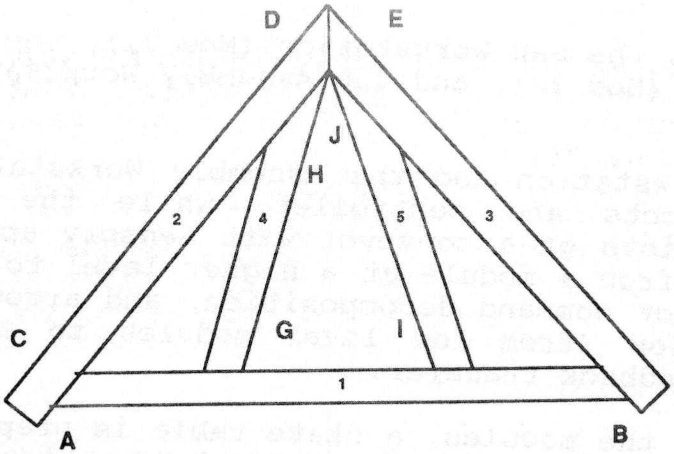

Fig. 3 Simple Roof Truss

The overall structure consists of the three major parts: (a) the process planner which provides the sequence of work, (b) the control hierarchy, and (c) the execution environment which can be simulated. Figure 4 shows the different modules and the links between them.

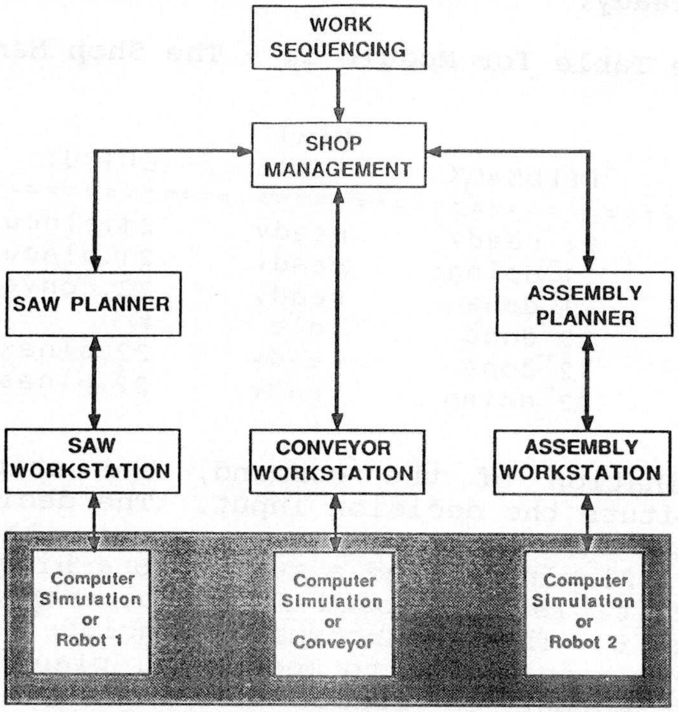

Fig. 4 Hierarchical Control Structure for Automated Truss Manufacture

At the first level (the highest level) in control hierarchy, there is a single module - the Shop Manager (Mod 11) which allocates the tasks of cutting, conveying and assembling to the other modules in the hierarchy . Two modules - the Saw Planner (Mod 21) and the Assembly Planner (Mod 22), constitute level two. The lowest level, consists of three modules, the Saw Workstation (Mod 31), the Conveyor Workstation Unit (Mod 32), and the Assembly Workstation (Mod 33).

The Saw Workstation and the Assembly Workstation each consists of robots and controller, while the Conveyor Workstation consists of a conveyor with sensory appliances. Arrows pointing from a module at a higher level to one at a lower level depict command decomposition, and arrows in the opposite direction (from low level modules to superiors) correspond to feedback transfer.

For each of the modules, a state table is prepared. Any line in a state table contains all information that is necessary for the transformation of the decision input to the decision output. Table 1 shows the state table for mod 11. The first line reads, "mk3_t1, idle, 21_ready, ready, 21.plncut, 11.n/a. "mk3_t1" stands for make 3 trusses of type 1 which is a command. The module is idle and the feedback it has received from a lower module (in this case module 21) is ready.

Table 1. State Table for Module 11 - The Shop Manager

ADDRESS : 11

INDEX	COMMAND	STATE	FEEDBACK	NEXT STATE	OUTPUT	REPORT
1	mk3_t1	idle	21_ready	ready	21.plncut	11.n/a
2	mk3_t1	ready	21_doing	ready	21.plncut	11.n/a
3	mk3_t1	ready	21_done	ready	32.convey	11.n/a
4	mk3_t1	ready	22_done	idle	#	11.n/a
5	mk3_t1	ready	32_done	ready	22.plnass	11.n/a
6	mk3_t1	ready	22_doing	ready	22.plnass	11.n/a

The combination of the command, the state and the feedback constitute the decision input. The decision output consists of next state (in this case, ready), the output to a lower module (21.plncut) and a report to a higher module. The command mk3_t1 is decomposed into subcommands as can be seen in Table 6. These subcommands include - plan for cutting (plncut) - an output to module 21, plan for assembly (plnass) - an output to module 22 and begin to convey (convey) - an output to module 32.

The report is a feedback to a higher level. Since module 11 is at the highest level, reports have no further implication - hence the appearance of n/a for all decision outputs. However, for lower level modules, the report is the path through which the change in feedback (which is part of the decision input) is initiated .

All the state tables are created by the system analyst who has to design the entire operation in detail. Once these tables are created, the initial states and feedbacks of the modules are added and the process durations (Table 2) for each of the work tasks (in this case, for each of the actual physical movements such as making the cuts 1A through 5J, the conveying operation, and the tasks of assembling each of the elements in the right manner to form the truss).

Table 2. Durations of Work Tasks

INDEX	COMMAND	STATE	FEEDBACK	DURATION
1	cut1A	idle	n/a	10
2	cut1B	ready	n/a	10
3	cut2C	ready	n/a	10
4	cut2D	ready	n/a	10
5	cut3E	ready	n/a	10
6	cut3F	ready	n/a	10
7	cut4G	ready	n/a	10
8	cut4H	ready	n/a	10
9	cut5I	ready	n/a	10
10	cut5J	ready	n/a	10
11	ass1	idle	n/a	5
12	ass2	ready	n/a	5
13	ass3	ready	n/a	5
14	ass4	ready	n/a	5
15	ass5	ready	n/a	5
16	convey	idle	n/a	20

On completion of creation of the different modules, and the tables for initial states and durations, the emulator is ready to be run. Commands are either entered manually or selected from a list, and the time at which the execution is to be terminated is entered. A match is then sought between the combination of first command and the current state and the current feedback.

Table 3. Eventlist for Command mk3_t1

Event No.	Event Name	Start Time	End Time
1	cut1A	0	10
2	cut1B	10	20
3	cut2C	20	30
4	cut2D	30	40
5	cut3E	40	50
6	cut3F	50	60
7	cut4G	60	70
8	cut4H	70	80
9	cut5I	80	90
10	cut5J	90	100
11	convey	100	110
12	ass1	110	120
13	ass2	120	130
14	ass3	130	140
15	ass4	140	150
16	ass5	150	160

When a match is found, the decision output can be executed. The process of command decomposition and feedback is repeated until the command is executed or until the run time, specified earlier, expires. Table 3 shows an eventlist of tasks that are executed along with their start times and end times. At each step in the emulation, the decomposition and feedback transfer are explicitly displayed to enable the experimenter to test his/her line of reasoning regarding each production step.

8. CONCLUSION

Computer Integrated Construction (CIC) offers many opportunities for integrated planning and control. The unique features of construction presents many problems that have to be considered before the adaptation of existing control systems or eventually, the development of new approaches. This paper has presented one aspect of control systems namely the hierarchical decomposition through commands and feedback and the decision making process in a rule-based control environment. The use of a microcomputer based system for studying and testing control systems has been discussed by its application for automation truss manufacture. An in-depth understanding of goal formulation, goal decomposition and modeling was necessary for this task. Future efforts will be directed at the integration of the control system with actual robots at the lowest level of control.

9. REFERENCES

1. Dantzig, G.B., and Wolfe, P., "Decomposition Principle for Linear Programs," Operations Research 8, pp. 101-111, 1960.

2 Furlani, C., Hierarchical Control System Emulation, User's Manual, National Institute of Standards and Technology (NIST), January 1985.

3. Kron, G., "A Method for Solving a Very Large Physical System in Easy Stages," Proceedings of the Inst. of Radio Engineers, Vol. 42, No. 4, pp. 680-686, 1954.