CIFE CENTER FOR INTEGRATED FACILITY ENGINEERING

MODEL-BASED PLANNING UTILIZING ACTIVITY AGGREGATION BASED ON ZONES

by

Graham Winstanley and Kunito Hoshi

TECHNICAL REPORT
Number 69

July, 1992

Stanford University

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 94305-4020

Summary: TECHNICAL REPORT #69

Title: MODEL-BASED PLANNING UTILIZING ACTIVITY AGGREGATION BASED ON ZONES

Authors: Graham Winstanley and Kunito Hoshi

Date: July 1992

Funding: Graham Winstanley was supported during the academic year 1991/92 by the UK Fellowship of Engineering; Kunito Hoshi was supported by Kumagai Gumi Co., Ltd, Japan.

Abstract

Model-based planning is capable of producing realistic plans on the basis of generalized domain product and process models and the application of specialized domain knowledge. Our research vehicle, OARPLAN, operates by compiling plans at the component detail level subsequent to the input of a detailed and specific product description and consequent instantiation of at least part of the general model. However, problems of complexity and utility result from the inherently bottom-up processes involved, and the aggregation of components and activities into 'virtual components' and 'realistic work packages' is one viable step in the pre-scheduling process. In this research, our objectives were to develop activity aggregation strategies which would be capable of exploiting the model-based paradigm, but which used solid engineering foundations in the reasoning processes involved in determining optimal component clusters, which we have termed zones.

Subject

The work described in this report was carried out over a period of twelve months. Two parallel, but associated themes of research were conducted: activity aggregation as a natural design or immediate post-design step, in which groups of components could be assigned to physical zones on the basis of user choice or system reasoning; the determination of optimal physical zoning (and zone sequence) on the basis of fundamental engineering principals and satisfying the constraints of cost or project duration. The two areas of research come naturally together at the interfaces, i.e. OARPLAN requires component clustering and sequence information, and the zone analysis system a presentation, planning and scheduling system to produce useful output. The key ideas are concerned with strategies of aggregation using model-based planning systems, and the possibility of integrating essentially procedural modules to initially determine optimal component clusters and then to provide taxonomic information for later explanation. The essential message is that model-based planning systems can be designed to exploit fundamental engineering knowledge in the activity aggregation process, which is itself a vital step in the plan-to-schedule process.

Objectives/Benefits

The work has demonstrated the two main aims of the two strands of this research: that activity aggregation can be successfully performed as a further step in the model-based planning process, and that it is possible to define strategies for optimal zoning on the basis of cost or duration. Zoning is an important aspect of construction planning, and this research has demonstrated to utility and potential of integrating external procedural techniques (external to the model-based reasoning system) to carry out zone definitions capable of objective and analytical evaluation. OARPLAN is able to provide information to the zone analyst, which is then able to employ well-founded techniques and reasoning to establish initial component aggregations and suggested zone sequences.

OARPLAN is the able to produce an abstracted plan as a sequence of zones, with the possibility of justifying its actions. The research proves that aggregation is possible in a number of ways, and that the integration of a deep analytical system with a symbolic representation and reasoning system can provide one, potentially powerful, technique for achieving efficient work packages.

Methodology

The research was conducted as two, parallel but associated, themes. The first was to extend the capabilities of the OARPLAN system to cope with the complexity Apparent in scaled-up construction examples (see CIFE Technical Report #66). During our attempts to produce plans for full-scale buildings, and working with an expert in the construction domain, it became apparent that component-level plans, although powerful and expressive, had limited utility to its intended users. Much work was carried out on determining methodologies for aggregation; on the basis of physical zones, site accessibility, trades, etc., and on developing strategies which would provide a range of component clustering techniques, including designer-definition and system inferral. Tests were carried out on two NASA buildings, which had been modeled and planned by OARPLAN, and the results were evaluated by the domain expert employed on the project (CIFE Technical Report #66). One major problem was left to the second theme of this report: the justifiable, intelligent, definition of zones. This work utilized fundamental knowledge and accepted algorithms for zone definition. Its goals were to establish clusters of components in typical reinforced concrete buildings, on the basis of least cost, or optimal duration. Simplified test cases were developed and the system was provided with realistic symbolic and numerical information. Finally, OARPLAN was used to provide plans from the output of the zone analysis system's operation.

Results

The research demonstrates that activity aggregation and abstracted plans can be provided within a model-based planning environment. It is possible, not only to manually associate together groups of components at the design stage, but it also possible to employ AI and hybrid techniques in the automation of the entire process. The zone analysis system has demonstrated the potential benefits of exploiting fundamental knowledge and accepted engineering practices in the vital process of zoning. However, its range of constraints and operating attributes were artificially limited in this study, which should be considered a first attempt to develop and use hybrid techniques in the establishment and planning of aggregated project activities. A paper, based on this report, has been submitted to the Journal of Artificial Intelligence in Engineering Design, Analysis and Manufacture (AI EDAM)

Research Status

This particular project is now completed. However, there is much work required on the use of intelligent reasoning techniques (heuristics) to aggregate on the basis of trades interests, i.e. the clustering of components for each trade involved in the project. This would mean that some components might be shared by several zones, but this approach may provide a more efficient set of relationships for the scheduling process. The integration of a separate analysis and planning system which exploits deep knowledge is an excellent way of providing information on virtual components and suggesting efficient zone sequencing. By developing this aspect of aggregation as a separate process, in which information is bidirectionally communicated via its own interface, its operation can be developed, tested and used in isolation from the model-based planing system. OARPLAN is capable of providing much of the information required by the zone planner (some of which it receives from the integrated CAD system), and is certainly able to exploit the zoning system's output. However, much further work is required on adding more realistic constraints to the system, and applying more engineering knowledge. Fully integrating the zone planner into the OARPLAN system remains a future task.

MODEL-BASED PLANNING UTILIZING ACTIVITY AGGREGATION BASED ON ZONES

Graham Winstanley[1] **and Kunito Hoshi**[2]

Abstract

When model-based planning systems are scaled up to deal with full-sized industrial projects, the resulting complexity in the project-specific model and production plan can create serious problems, not only in dealing with such complexity computationally, but also in user-acceptance. In the model-based planning system described in this paper, activities are dynamically generated, inherently at the detailed level of individual facility components. However, it is possible to intelligently group together collections of components which would be common to 'realistic' work packages, and hence schedule on the basis of virtual components existing within an abstraction hierarchy. This paper describes a technique of project planning within an integrated design/planning system, which exploits fundamental knowledge of engineered systems and provides powerful and flexible planning functionality

1.0 Industrial Planning and the Model-Based Approach

Industrial model-based planning systems are characterized by their inherent knowledge about the domain in which planning is to take place, but more importantly, they are based on a knowledge-rich model of the industry's product or products. In the manufacture of large-scale, high-technology devices, in which each product is based on what might be considered a basic, or general design, but where each product is unique in one way or another, the maintenance of a central generic model is a valuable asset. Such a model could include all possible alternatives in design or production (limited by the industry's constraints on manufacturability, etc.), and could contain enough information and knowledge to very quickly ascertain the implications of design changes, changes in the production process, constraints on resources and other 'external influences'. Concurrent engineering gives us the possibility for rapid and coherent multi-disciplinary design, but the product model represents the 'target' for their efforts.

The Professional Intelligent Project Planning Assistant, PIPPA [Marshall 1987, Winstanley 1990] was one of the first model-based planning systems to be developed and evaluated in an

---

[1] Department of Computing, University of Brighton, UK. C/o The Center for Integrated Facility Engineering, Stanford University, CA 94305-4020

[2] Kumagai Gumi Co., Ltd. Tokyo, Japan. C/o The Center for Integrated Facility Engineering, Stanford University, CA 94305-4020

industrial context. The central concept of the system was a knowledge-rich general model of the company's product, which in this case related to flight simulators. The model itself was based on a comprehensive component hierarchy, each node of which could be associated with an action and one or more resources. Activities were created dynamically by the system, as intersections of relevant objects, actions and resources. Although the definition of an activity as the intersection of objects, actions and resources, was derived in part from the MOLGEN concept of object and action abstraction [Stefik 1981a, 1981b], the implementation of those concepts in the PIPPA system proved to be intuitive, powerful and flexible in the application industry. The PIPPA system was designed to cater for the configuration problems found in the design-to-implementation of large scale, high-technology engineering products, and therefore takes as its input a specification and various user-choices. It is based around what the authors refer to as a central 'information base', which incorporates a generic description of the product and the processes required to produce it (the generic model). The information base also contains a specific model of a project as it is dynamically planned and directed by the system during run-time. Each element in the component breakdown structure, action hierarchy and resource description holds information and fundamental knowledge, which can be applied as required through the invocation of rules [Winstanley 1989]. This implementation strategy results in an object-oriented system behavior, where each component, action, resource and therefore activity is responsible largely for its own existence within the ensuing specific model and manufacturing plan. However, in a configuration system such as this, complexity is gradually added through a progressive and highly-controlled refinement process.

OARPLAN [Darwiche 1989], is one of a family of knowledge-based construction systems that have been developed at CIFE to address problems existing in the construction industry, and belongs to the class of automated planning systems which employ a knowledge-rich domain model in their problem-solving strategy. The model is centered around a generic project description, based on the concepts of product breakdown structure, and similar data structures able to express actions and resources. The system uses declarative knowledge of action/object pairs to identify activities within the plan, and object relationships for activity sequencing. This approach differs from the STRIPS-style 'classical', or state-based planners [Fikes 1971], which are based on preconditions and effects of operators (in this case, activities). The central concept used in these early systems was that an operator, or activity, could only be applied if its stated preconditions can be satisfied, and that the application of the activity would have an explicit set of effects, which may or may not satisfy the preconditions of other activities, etc. In such systems, the planner carries out a state-space search — forwards, backwards, or in both directions — and attempts to generate a sequence of actions that will incrementally change the world from its initial state into the goal state. This form of activity representation provides a general approach to planing and can thus be applied to any planning problem, but it exploits little domain knowledge about a given planning problem. Consequently, planning systems based on this representation of activities must search extensively, even when provided with heuristics to guide the search, must often backtrack to resolve interferences, and are capable of solving only simplistic problems to date. Problems of plan interaction and interference resulted in the planner's inability to find optimal solutions, or any at all. Sophisticated solutions to these problems have been proposed [Sacerdoti 1973 and 1975, Tate 1976, Wilkins 1983, Chapman 1987], but in all these cases the planning systems lack fundamental knowledge of engineered systems, and are unable to reason causally about the inclusion of activities in a plan, and their subsequent sequencing. An excellent review of planners of this type is provide in [Tate 1985].

SIPE [Wilkins 1988], a planner in the STRIPS-style, considered the problem of resources which may be required by more than one activity. It identified such conflicts as arguments existing in two operators, but unlike OARPLAN, the system was not equipped with fundamental knowledge about each object, or the relationships between them. Significant modifications have been made to the original system in SIPEC [Kartam 1990], to incorporate fundamental knowledge of the relationships between objects in a plan . This work also demonstrated the viability of a computational link between a commercial CAD system and an AI planner [Ito et al 1989].

The CONSTRUCTION PLANEX system [Hendrickson 1987] creates activities in *an element activity* hierarchy, based on an extended MASTERFORMAT code. Each component, or *design element* , has associated with it element activities, and aggregation proceeds by grouping several activities of similar nature, again based on its inherent labeling structure. While this approach to aggregation prior to scheduling may provide acceptable solutions in the construction industry, the very fact that it depends on a hierarchical breakdown structure which is predefined as part of its context, renders it somewhat less than flexible. Aggregation should be possible on the basis of a number of interests, including:

- *Physical zones*. These may be whole floors, parts of floors, parts of rooms, etc.
- *Trades*. One skilled workforce may define a completely space as a realistic work package than another trade. Aggregation based on this concept therefore requires consideration of optimal groupings of components, durations for that work package, and any interactions or conflicts that might occur with other 'trade-zones'. This type of aggregation we have termed *functional zoning*.

- *Resource availability.* It may be necessary to create work packages around the availability, and constraints resulting from, major and costly resources, and site constraints.
- *Costs.*Especially labor costs. The main costs of construction are materials, production equipment, and manpower. The cost of material is largely independent of the building system, which places manpower and equipment as the two most important factors in scheduling. Section 4 of this paper discusses a new approach to zoning based on this.
- *Major milestones.* Aggregation is necessary for the benefit of overall project management. Management may only require fairly abstract information on a project plan; the level of abstraction depending on the manager's position within a company's organization, and the control that he/she has on the project. This type of aggregation is more simple than the others to realize, through the abstraction hierarchy inherent in the project model.

This paper concentrates on the problems of activity aggregation as a necessary process in model-based planning systems. OARPLAN is described, as a typical model-based system, and a small example is included of a project planned by the system. Activity aggregation, or zoning, is discussed, both philosophically and through example, and the implementation strategy is expounded. The second part of the paper deals with a technique for approaching the problem of dynamic (and automatic) zone definition, and proposes a system for optimizing labor costs.

The current implementation of the OARPLAN system is on the KEE™ 3.1 platform, on top of Sun™ Common Lisp 4.0™. Previous versions have been implemented as a blackboard system, using BB1 [Hayes-Roth 1987, Darwiche 1989], and on a Macintosh computer in Allegro Common Lisp™. The zone planner runs in the ProKappa™ environment on the same Sun™ computer system

2. OARPLAN: a model-based planning system

OARPLAN has been used to provide an integrated design/construction environment, in which detailed designs are available from the computer-aided design phase, to be used in the production of construction plans. The core concept of this system is a generic model of the domain, existing within the system as component, action and resource hierarchies. Planning knowledge is used in the form of rules, which are applied to establish precedence between construction activities on the basis of inter-component relationships. The application of the OARPLAN methodology on simplified problems in the construction domain has been reported previously [Darwiche 1989], but recently the system has been extended, enhanced and applied in the planning stage of full-scale facilities [Winstanley 1992].

Computer-aided design systems are able to provide descriptions of designs in terms of the project's individual components. Relationships between these components, and groups (or classes) of components, are used in construction to properly sequence and consequently to schedule project activities. By representing objects at the most detailed level, along with information on their specifications, position and relationship to other objects within the design [Ito 1989], it is possible to build a computer-based model of the product.

The OARPLAN methodology states that an activity within a project is a specific action on a specific object using a specific resource (or resources). Activities can share objects and actions with other activities, but in order to produce a plan as an ordered sequence of activities, it is necessary for the system to be equipped with both action and resource models. The action model is constructed in a similar fashion to the component hierarchy, having ACTIONS as its root. The actions construct, install and order are classed as EXISTENCE_ACTIONS and similarly the actions Dispose, Erect, Excavate, Place, Pour, Reinforce, Remove and Weld are represented within the class MODIFICATION_ACTIONS. The hierarchy is further elaborated until 'primitive-level' actions, which are defined as those actions which would be carried out on a component at the lowest level in the object hierarchy, are represented.

By careful design of the action and object hierarchies, it is possible to describe activities at many levels. For example, at the lowest, or primitive level, it is possible to create an activity for the reinforcement of all the spans in a building by linking the object Spans with the action Reinforce. However, this activity can be represented by a number of lower-level ones. For example, by linking the action Reinforce_beam with the object Beam, and by linking the action Reinforce_column with the object Column, etc., a whole sequence of lower-level activities are created. Resources are represented in yet another hierarchy under the standard classes of Equipment, Labor and Materials. Resources are also defined as consumable or non-consumable, and specific resources are attached to the action hierarchy, so that for example, the action *construct_column* would (or might) require the human resources *Carpenter, Rodman and Laborer/Finisher*

By virtue of the inherent botttom-up nature of systems such as OARPLAN, activities are created and sequenced at the lowest possible level of abstraction. Individual components are associated with an action or actions which must be carried out to construct or place that object. Resources are also attached at this level of detail. Abstraction is possible as a direct result of specific objects, actions and resources being members of the hierarchies making up the general model (through instantiation). Components, actions and resources can be grouped into classes, and it is possible to extract project information at various hierarchical levels for project and corporate management purposes [Winstanley 1990]. However, this form of abstraction is insufficient for scheduling purposes. In the construction industry it is necessary to schedule and optimize various skilled work crews in a 3-dimensional and highly-complex spatial system (a building say), taking into account such influences as facility site, resource constraints, costs, project duration, safety, regulations and other, more tacit, human factors. Experience has shown that detailed plans of the OARPLAN variety can be useful in defining job logic, but when applied to full-scale facility planning, the complexity apparent in the non-linear plans produced is only useful as a 'stepping stone' to a realistic activity sequence and eventual schedule. OARPLAN without activity aggregation provides as its output, reasoned information on the correct logic of an eventual plan, i.e. it provides a good specific model of the project, including dependency relationships between the lowest level activities. Activity aggregation is therefore necessary as a further (but not final) step in planning, in which realistic and reasoned work packages are allocated as collections of components on which actions must performed, using certain resources.

2.1 The Process of Model-Based Planning in OARPLAN

The OARPLAN system exists as a central representation and reasoning mechanism, which requires input in the form of a detailed product description, and which requires some form of output presentation medium. The OARPLAN system itself has a facility for manual data input, and through its underlying KEE™ environment, has the means to display its results in a number of formats. In our research we have taken the approach of integrating commercial software with research systems in order to present the intended users with an interface and set of tools which they are used to.

A system specially developed within the *Center for Integrated Facility Engineering (CIFE)* at Stanford University, known as CIFECAD [Ito 1989] has been used to provide a greater level of detail on individual construction designs than conventional systems. Built as an extension of AUTOCAD™, it provides for a description of a facility, not merely as an ordered collection of dimensioned geometric shapes, but also as an interrelated set of components which together describe the whole building. As a result of the inherent 3-dimensional design process within CIFECAD, a textual description is available to systems such as OARPLAN, which includes information on material type, volume of material used and associated components within the design. From this information, it is possible for OARPLAN to extract information relevant for modeling and subsequent planning.

The ability to integrate commercial and research software system in this way is considered by the authors to hold much potential in the future of industrial project planning/management and concurrent design.

The process of project planning using OARPLAN includes the following discrete steps:

Design data input	Input the details of a construction design. This necessarily includes each individual component within that design, described in terms of its component class (Beams, Columns, etc.), a unique identifier (BM123, C123, etc.), its relationship to any other components (Supported_by, Connected_to, etc.) and a list of components which are related in that fashion. This input file can be produced in ASCII form from the CIFECAD system.
Modeling	Produce a specific model for the building as an instantiation of at least part of the generic OARPLAN model. In practice, this results in activities being created for each component in the input file, and by virtue of the OARPLAN methodology, each activity created in this way links actions and resources to each unique activity. Once this process is complete, the specific model exists as an unordered set of activities; their constituents being instantiations of generic objects, actions and resources. it is possible to browse the model and interrogate constraints and construction implications by viewing the cluster of related information now associated with each component.
Elaboration	Rules exist in the system, which are designed to classify activities as either *simple*, which means that they are already at their most detailed level of representation, or *compound.* Compound activities can be elaborated into more detail, for example: *place_concrete*. The activity *place_concrete* can be elaborated into its more primitive activities of *pour_concrete*, *finish_concrete* and *cure_concrete*. Once classification has taken place, more rules are invoked to actually create and assign the more detailed activities identified and labeled by the classification rules. Elaboration can be initiated at various levels of abstraction

(the highest being "build building"), but is more commonly and routinely used at the individual component level.

Planning

In the current OARPLAN system a plan is defined as a sequence of activities ordered by end to start precedence relationships. These relationships are established on the basis of domain-specific knowledge applied to the model, using the inter-component relationships defined in the input file. Activity classification occurs at this time (compound or simple activities), and activity elaboration may optionally be performed.

Zoning

Aggregation can be applied before or after resource allocation, and is invoked via an active button on the OARPLAN system menu. At present the process relies on the specific model created during the modeling phase, and the explicit relationship *includes*. Zones are generated in an object-oriented process of 'remodeling' based on the central OARPLAN concept of *actions*, *objects* and *resources*.

Resource allocation

The generic model within OARPLAN includes a resource hierarchy describing manpower, equipment and materials. In the current implementation, labor resources are associated with actions in the actions hierarchy, in such a way that flexibility is facilitated for varied site conditions, etc. For example, the trades *Carpenter* and *Finisher* are associated with *grade_beams*, which may or may not require the action *form_beam* and/or *finish_beam*, depending on the soil conditions and elevation of the grade_beam. Our strategy has been to model the resources that *might* be needed in any given situation; that situation being described at the start of the OARPLAN modeling-to-schedule process. Resource requirements also depend on material type/volume and position/elevation within the facility. At present these are manually derived, but the CIFECAD system [Ito 1989] is able to provide such information now, and future work will include the development of a query mechanism, which will lead to a fully automatic and dynamic resource allocation system.

Activity timing

The RS Means™ estimating guide has been used as the basis for determining the duration of activities [Winstanley 1992]. This guide calculates the average cost and productivity for most types of construction, from all major US cities. Crew breakdowns are provided as well as geographic adjustment factors. It is one of the most used cost estimating references in the construction industry, and therefore provides a useful and widely-accepted taxonomy for cost and time estimation. At present the system has the ability to access a sub-set of the relevant information represented within the OARPLAN generic model (specifically for structural steel buildings), but future versions of the system will have the added functionality of an automated estimating module.

Plan output This is available as a graphical display on a KEE™ window, or as a text file capable of being used by a commercial project planning system. The graphical display is useful in gaining an immediate appreciation of plan sequence and potential plan concurrency, but the usefulness of this form of display is limited by the size of the plan and the physical limitations of the computer screen.

Scheduling The OARPLAN system is able to produce its output in a form acceptable to the Primavera™ project planning system. Through this medium it is possible to use commercial software in the scheduling and refinement of a given project previously created and planned by OARPLAN. This gives us the potential, not only to interactively create realistic schedules, but also to continually update the central model held within OARPLAN through Primavera's in-built batch file *generation* utility.

3.0 Activity Aggregation Through Physical Zones

Problems associated with model-based planning on a full-scale project are mainly the result of complexity in the plan produced. Although the sequence of activities may be available in a form suitable for verification, the sheer size of the plan produced at the component level granularity causes problems by virtue of the resulting network complexity. Precedence relationships can be displayed to great effect graphically, but with such very large networks, a graphical display of the plan is severely limited in its usefulness. Textual output, and the use of commercial project planning software, can alleviate plan presentation difficulties, but the sheer size and complexity of component-level precedence networks remains a problem. Although the plan may be logically correct, component-level precedences are useful mainly at the level of daily or weekly crew planning by a foreman. However, it is possible to aggregate activities into realistic work 'packages', each with their own set of activities, actions, objects and resources, and capable of being scheduled as such. This approach, which we have termed 'zoning', allows for bottom-up aggregation to any level; that level being specified by the designer and represented directly within the CAD output file. This is similar to the strategy adopted in the PLANEX system [Zozoya 1989], but the OARPLAN implementation is not limited to the 'mechanical' structuring of the product breakdown structure, but rather is dependent on the more flexible grouping of 'virtual components' by the designer, or inferred by the system itself.

In the normal course of design-to-planning using the OARPLAN system, each of the individual components (column_footings, grade_beams, slabs, etc.) would be associated together by one of several relationships. Using zone boundaries, it is possible to partition the problem into smaller clusters of components. It is not necessary, or desirable, to replace or modify any of the relationships exiting between components in two or more zones, but it is possible to provide a extra piece of information on each zone and the components which exist within the pre-drawn zone boundary.

Figure 1 is a schematic view of the the OARPLAN process, from initial plan generation through to schedule production. In this figure (a) represents the total plan produced by the system as a result of specific model production, and planning through the invocation of its planning rules. The plan is typically large and its logic is complex. (b) represents the significant reduction in overall plan complexity through the process of zoning. Each zone (1 through 3) is represented in the system as a zone object and responsible for its own boundary and interface (zone start and end units). (c) is a plan as a sequence of zone activities, without temporal attributes at this time, and (d) represents a schedule as produced by the commercial project planning system.

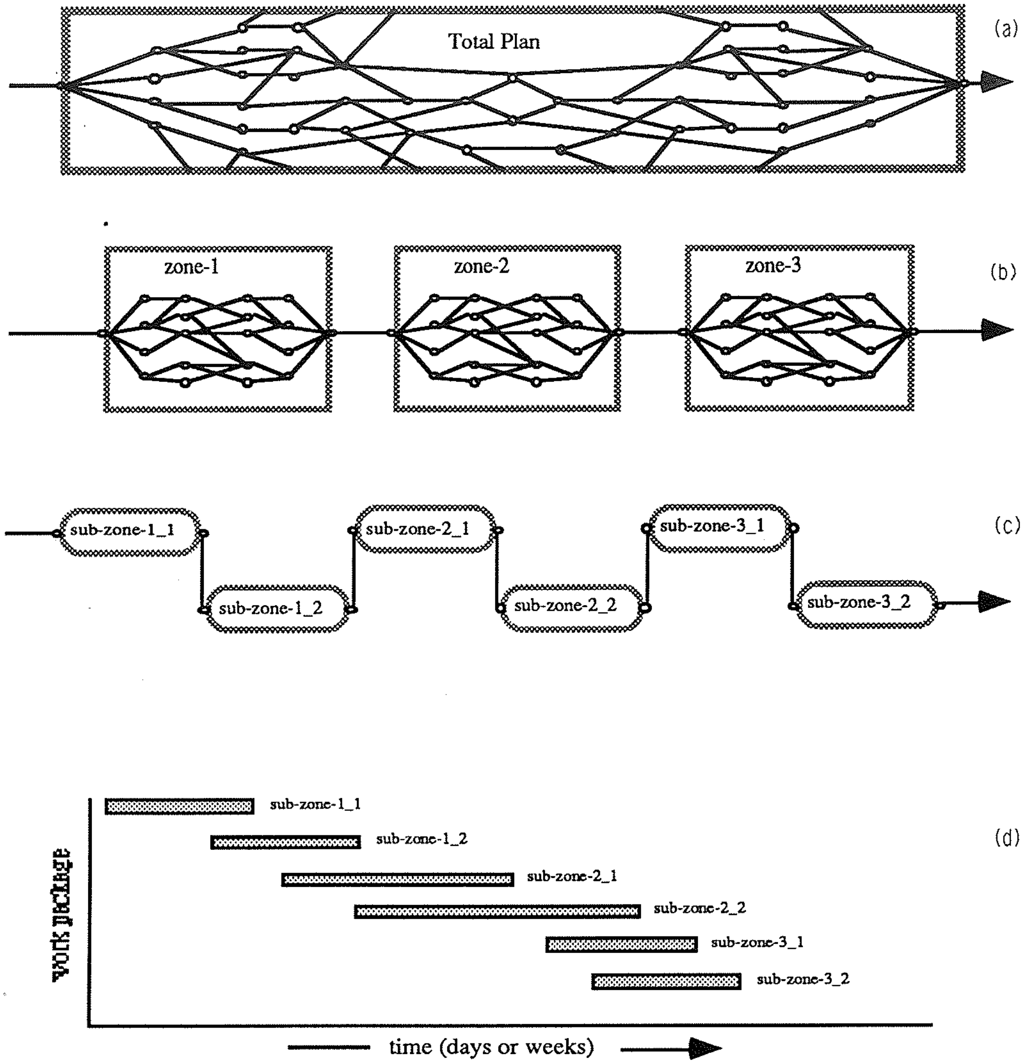


Figure 1. The zoning process within OARPLAN

3.1 Implementation

The core representational concept of object-action-resource in OARPLAN provides a simple, and yet powerful, means to build engineering models and provide reasoning in their application. In order to fully exploit this methodology and to provide for multiple-levels of aggregation, i.e. zones within zones, zones themselves are represented as objects with actions and resources. A zone activity is therefore created as a zone action upon a zone object using a zone resource. Actions and resources are merely collections of included attributes, but in the case of zones, each zone activity has responsibility within the system for the creation of start and end units, and for the actual aggregation (or collection) of appropriate activities from the previous (un-zoned) plan.

The technique involves a message sent to each zone, causing the zone itself to evaluate its included objects and the preceding and succeeding activities assigned to each one during the previous knowledge-based planning process. In the object-oriented zoning process, the zone method adds the new information *zone_prec_actv* and *zone_succ_actv* to each included activity, and when an intersection cannot be found through this matching process, the units:

zone_<zone ID>_start	.... the zone start interface unit, e.g. ZONE_Z1_START
zone_<zone ID>_end	.... the zone end interface unit, e.g. ZONE_Z1_END

are attached through the same *zone_prec_actv* and *zone_succ_actv* relationships. By maintaining two separate lists of sequencing relationship *types*, it is possible to reason separately about zoning, without any danger of modifying the overall (un-zoned) project plan, and it is possible to 'pull out' and display different views of the project. Figure 2 shows how zone boundaries can be imposed on clusters of activities through the definition of intra-zone precedences. This figure also demonstrates the significance of inter-zone precedences, which exist beyond the zone boundary, linking two or more separate zones.

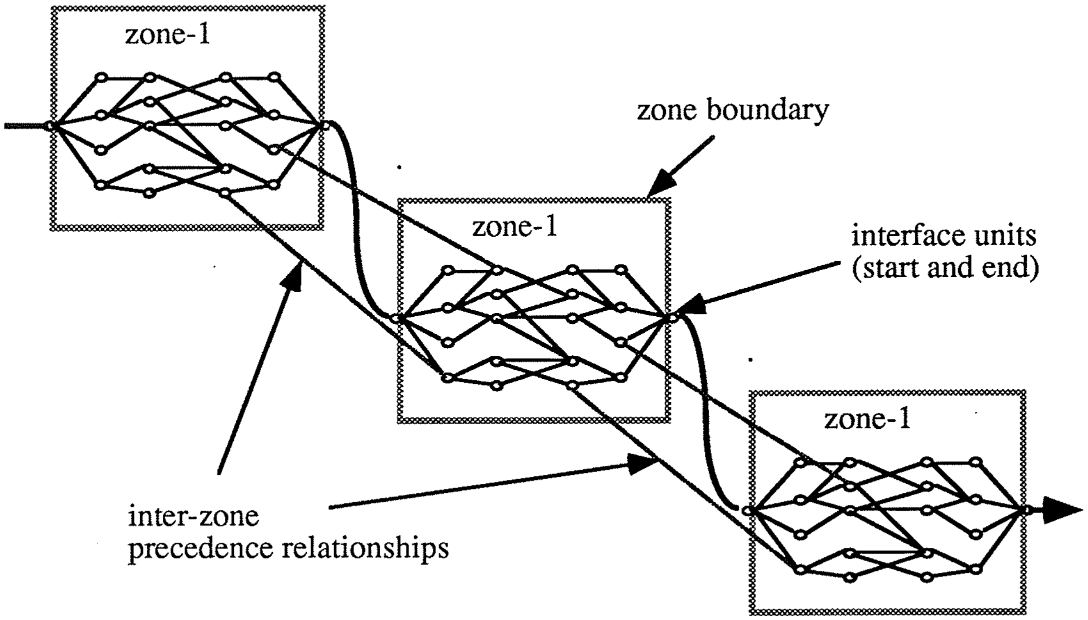


Figure 2. Zones have intra and inter-zone precedences

Functional zoning involves the use of trades, or specialized crews, as the basis for zone definition. It is possible to create functional sub-zones within previously-defined physical zones, merely by re-grouping zoned activities in terms of 'trade interests'. In this way, the virtual components defined by physical component grouping remain valid and unique within one zone. However, it may be possible to infer purely functional zones, related only by the work performed by the various trades employed on the construction. For example, it might be more useful to define a cluster of components which a crew of structural steel workers can work on, and define that as a *structural_steel_zone*. Other trades would, or might, share components with other zones, and the logical dependency of the results of one trades' work on another would provide the necessary sequencing information. Zones defined in this way could be dynamically re-assigned on the basis of crew-size change, resource dependence changes, environmental conditions, etc.

Figure 3 shows a plan, produced by OARPLAN, and presented as a precedence network. This figure represents Zone 4 of a overall plan comprising seven zones, and Figure 4 is the same plan with a KEE™ schematic map superimposed. The schematic map indicates the scale and structure of the knowledge base in question (which is in this case the total plan), and shows clearly the demarcation between activity 'clusters' making up each separate zone. The rectangular box towards the center of the map delineates that part of the plan shown on the 'main' KEE™ screen (in this case Zone 4). The structure of the zoned plan, as represented in the overlaid map in Figure 4 can be compared with the structure of Figure 2.

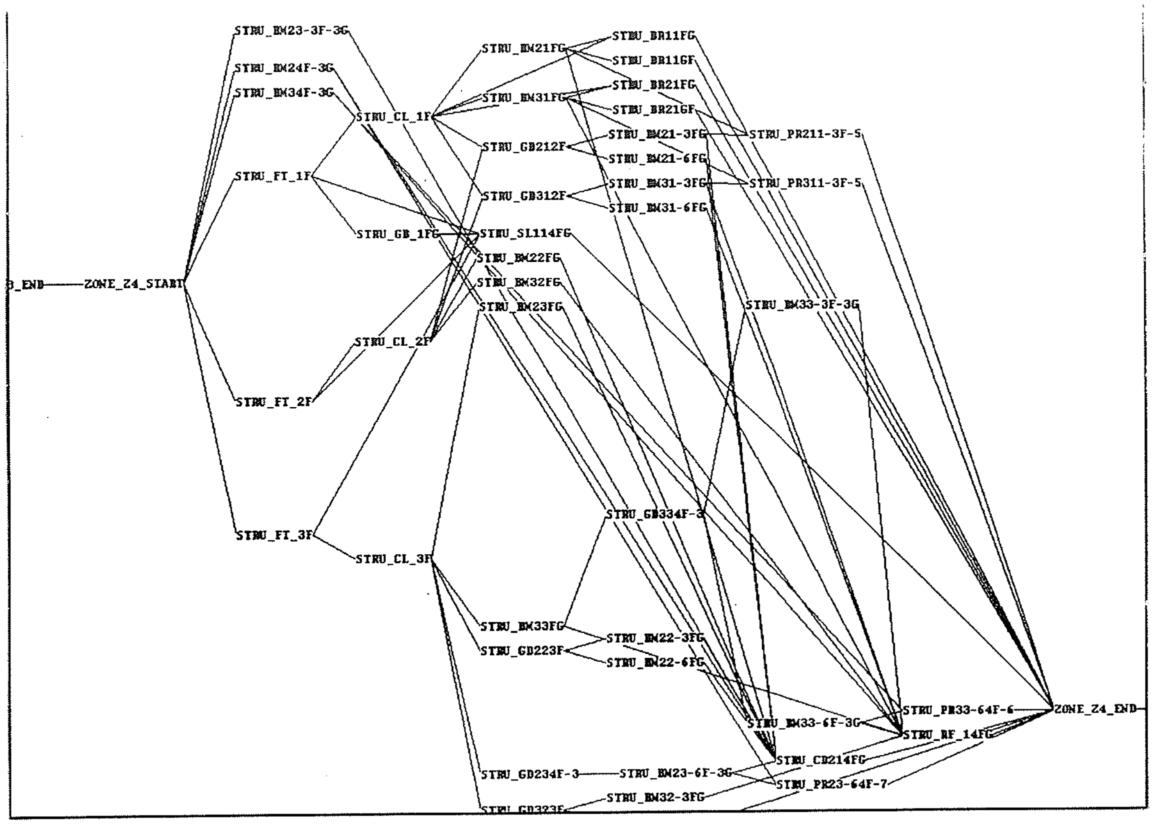


Figure 3. A precedence network plan of Zone 4. The following conventions are used: STRU = structural, FT = footing, CL = column, GB = grade beam, BM = beam, GD = girder, BR = bracing, CD = concrete deck, PR = purlin, RF = roof, SL = slab

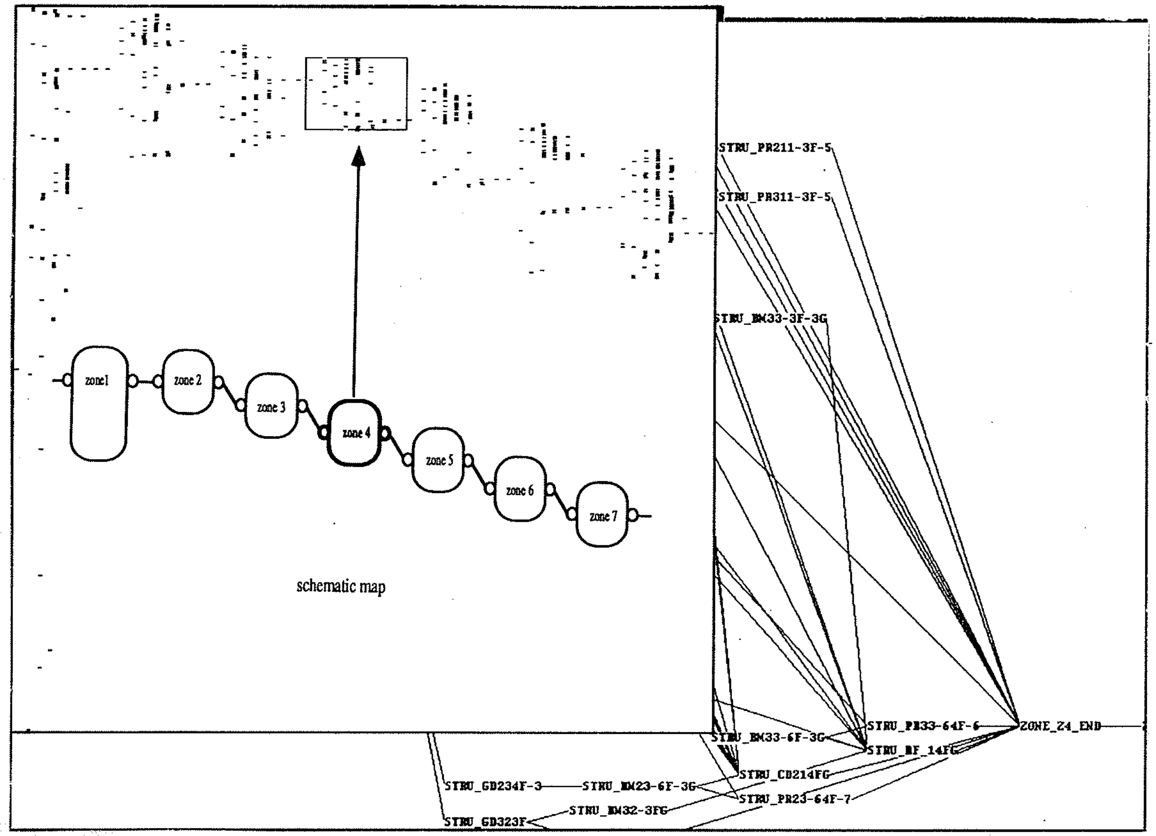


Figure 4. The same plan as in Figure 3, but with a KEE™ schematic map superimposed.

Through the process of attaching aggregated temporal data to zone activities, it is possible to approach the scheduling problem at an acceptable level of activity aggregation. By exploiting the system's inherent bottom-up approach to planning, in which component-level relationships are established and stored, it is possible to use these low-level precedences to provide accurate constraints on the scheduling of aggregated zone activities. For example, by making use of precedence relationships which exist in the overall plan (but not shown on the zone activity plot), it is possible to constrain the level of concurrency between individual zone activities to that permitted by precedences external to the 'zone boundaries'. OARPLAN is therefore capable of not only producing a plan through elaboration, which includes sufficient detail to cater for estimation and control, but is also capable of abstraction through aggregation. The zoning strategy used has the effect of applying extra constraints on each activity, thereby facilitating intelligent zone scheduling and dynamic redefinition of zone boundaries. At this stage, our zoning system is limited to the manual definition of physical zones and the optional creation of functional sub-zones. However, the problem of automatic initial zone definition remains, and a possible approach is through cost or duration optimization.

4.0 Aggregation Through Cost and Time Optimization

The main costs of construction are due to material, labor, and production equipment . Among these, material cost is the least variable for a particular building system, because it is not directly related to the construction system or the plan. Therefore, by optimizing labor and equipment costs, it is possible to approach a cost-optimal plan. An important task for both construction system planners and site engineers alike, is to provide and maintain a constant supply of appropriate labor, and to ensure their efficient deployment. The solution to these problems would result in an improvement in labor productivity and a more consistent and improved product quality. The ideal construction system plan would ensure a zero labor waiting time, based on an appropriate fixed crew size.

Construction, in general, can be classified as an 'order-made' production system by virtue of the 'one-off' speciality nature of each project. In addition to this, generally construction projects are characterized by few repetitive tasks; this being the one major reason for low productivity in this domain. However, by dividing a building into relevant zones, based on efficient labor allocation, the problems of inefficiency inherent in construction can be alleviated. This section describes a prototype system which searches for an optimal zoning plan for labor distribution under several construction constraints. The development of the prototype has been carried out in parallel with the OARPLAN system described in Section 2, and by virtue of the common computer platforms and operating system, is able to share information.

4.1 A Prototype Zoning System based on Labor Distribution

The purpose of this system is to search for an optimal zoning plan for efficient labor distribution, based on labor cost and construction duration. A reinforced concrete building type was used in the experimental work, and a 'test' building was described, with predetermined information such as: building name, construction duration, labor cost, number of stories, structure type, etc. The building can be divided to several blocks; a block in this case being defined as a span, which cannot be divided into others on a structure frame. Each block has the following specific structure data: block name, required material (concrete, form, bar) volume, relationship with other blocks and constructibility (i.e difficulties on construction). A zone consists of a block or several blocks which are attached to each other. Labor and materials are distributed to each zone under construction, and the construction 'behaviors' are executed at the zone one by one.

The zone planner can define construction data for the specific building, for example labor fee, labor's productivity, coefficient of additional labor fee (i.e the coefficient of additional and original labor cost), concrete pump car count at the site, etc. This data is defined with many constraints, including the building specialty (building type, scale, etc.). In the future it should also be possible to use existing information and knowledge about construction methods employed on similar projects.. This definition of the relationships between a building, blocks, zones and construction data is shown in Figure 5.

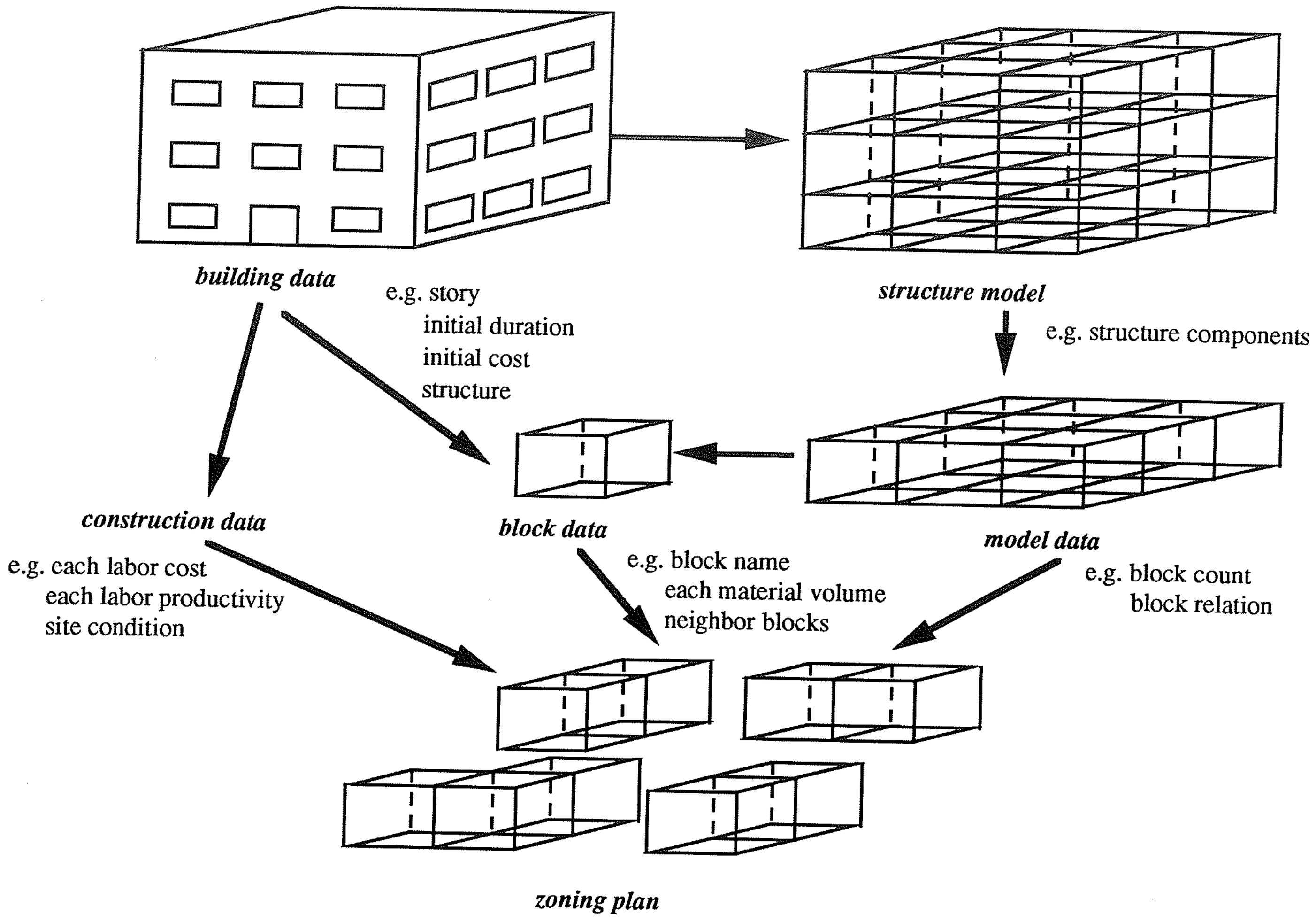


Figure 5. The relationship between building blocks, construction data and zones

4.2 The Reasoning Process

The reasoning process consists of six major constituents as follows (see also Figure 6):

1. Create all possible block's combinations and zone's combinations, called the "case". In creating cases, we assume a model of the required building (combination of blocks) and input block data to each block, i.e. material volume, block neighbors and construction difficulty.
   The system creates an object called "zone", and then the zone searches and includes appropriate blocks until the *capacity count* (variable value) reaches a predetermined limit. When every block eventually 'belongs' to a zone, through this iterative process, the system saves the zoning plan (block distribution layout) as a case. The next candidate for zone organization is selected by the slot values called "neighbor blocks".

2. Check all cases, determining which can and which cannot be executed. The system deletes all those cases which violate the given constraints, and therefore cannot exist in the plan.

For reinforced concrete buildings, the planner considers the concrete volume within each zone, because the manipulation of that material provides a time limitation, i.e. there is a limit on the amount of concrete that can be poured in one day.
The total amount of concrete which is poured in a day depends on the functional capacity of the concrete pump, and the site capacity for such equipment. In our prototype, the system checks each zone's concrete volume, and deletes the inadequate cases for concrete pouring activity.

3. Calculate an optimum labor distribution, taking into account material volume, labor's productivity and construction duration.The optimum labor distribution, performed without recourse to the zoning concept, is calculated using data on required construction duration, material volume, and labor's productivity. The crew size for each trade is adopted as an initial and optimum value for selecting the best zoning plan.

4. Calculate an additional cost, based on labor's 'activity waiting' cost and an additional duration based on activity waiting time in each case, compared with initial cost and duration. Additional cost in each case is calculated by the amount of every kind of labor's waiting time in each zone under the optimum labor crew size.

   Labor's waiting time for each kind of labor is calculated by how much time is required to finish the required activity (volume) in a zone for every trade under the defined labor crew size. The longest duration among them is determined as a duration for the zone. The other trades, which have a lower duration in the zone, are forced to wait until the zone's duration has elapsed. This unproductive time creates additional cost to the project. Additional duration in each case is calculated as the sum of the difference between the amount of the longest duration in each zone which is in in excess of the required construction duration. In Figure 7, the additional duration in each zone is the waiting, or idle time, shown as the shorter of the two labors durations, and the total additional duration is merely the sum of these individual durations.

5. Consider all cases for selection of the most efficient, by evaluating them with both cost and duration attributes. The additional cost and additional duration values for each case are used in selecting the best zoning plan. The experienced user defines the project priorities: cost, duration, or both, through definition of priorities or through maximum value restriction. At this stage in the research, the process is based on a comparison of Euclid distance value among all cases, i.e the minimum distance measured from the origin of an XY graph, with

cost and duration as the axes. This solution uses methods in order to choose the best case. However, in practical use, choosing the best case depends on the user's requests, based on the priority of cost, duration or a balance of them.

6. Create some variation types on the optimum case selected by the above diagnosis. The prototype system currently creates four variations on the optimum case, selected by the above process. This variation is defined as the combination balance of the additional cost and additional duration in the optimum case. The four variations are: *initial construction duration type (option1), minimum construction duration type (option2), reduced cost type (option3),* and *minimum cost type (option4).*

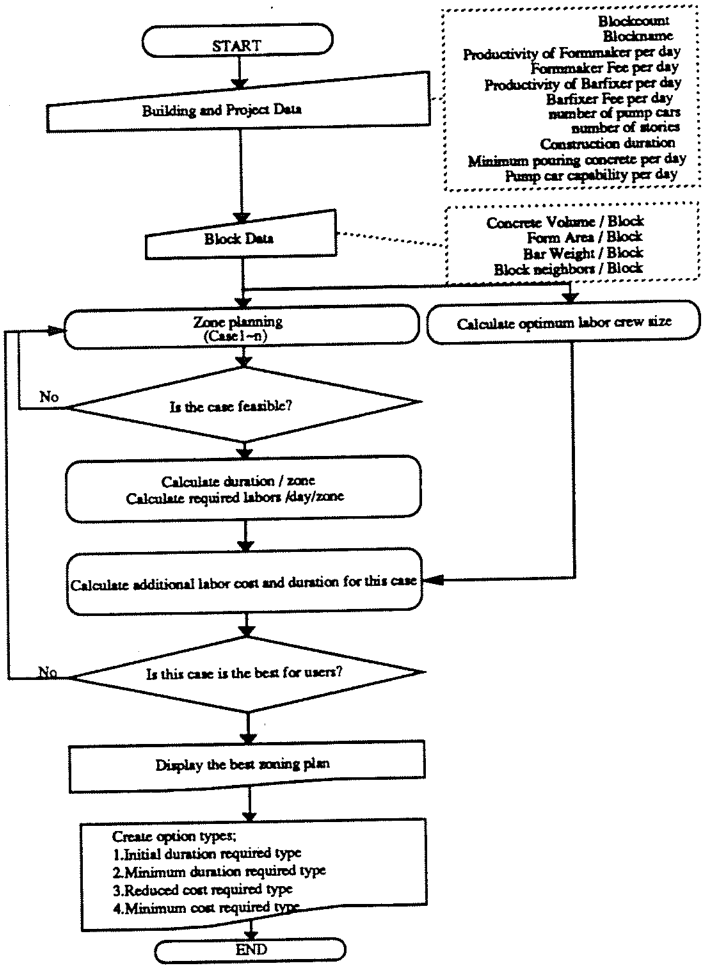


Figure 6. The Zone Planner system flow

Four variation types on the optimum case are follows:

Option 1 (Initial construction duration). Initial duration is defined as the duration required by the client and stated at the specification phase. Additional duration is due to unwanted labor idle time. Option one calculates the additional cost of the optimum zoning plan in the case where the additional duration is reduced to zero, i.e. the highest priority is initial duration. Extra labor (and the associated extra cost) is added to each zone in the trade-off. The initial construction duration can be maintained by increasing one labor type in order to eliminate waiting time. In Figure 7, it can be seen that the duration relating to one labor type has been shortened to eradicate waiting time. The thicker bars on this chart represents an increase in crew size. Minimum labor crew size in every zone is taken to be the optimum labor crew size.

Option 2 (Minimum construction duration - increased cost). Additional cost is defined as the 'extra' cost due to labor idle time *plus* the cost resulting from the process of employing more labor to reduce it. Option two calculates both the additional cost and duration of the optimum zoning plan for the case in which the total construction duration is to be minimized. Extra labor, and extra cost is added to each zone in the process of removing all waiting time. Minimum construction duration results from the replacement of every zone's duration with the shortest duration possible under an optimum crew size in each zone. Minimum labor crew size in every zone is taken to be the optimum labor crew size.

Option 3 (Reduced cost - increased duration). Calculates both the (reduced) additional cost and (increased) duration of the optimum zoning plan for the case in which the cost is to be reduced. In order to remove labor's costly waiting time, the crew size for this case must be reduced. Labor can be reduced for one trade to eliminate waiting time, as seen in Figure 7. However, in practice this must be applied *consistently* throughout the total duration in all zones. Labor relation practices suggest that the initial crews are emploeyd throughout the duration of the project. Addition cost can therefore be reduced, but the overall construction duration increases.

Option 4 (Minimum cost - increased duration). Calculates both the additional cost and duration of the optimum zoning type for the case in which the additional cost is to be minimized. This is accomplished by managing all trades with no waiting time under a reduced crew size. A schematic view of these four variations is shown in Figure 7

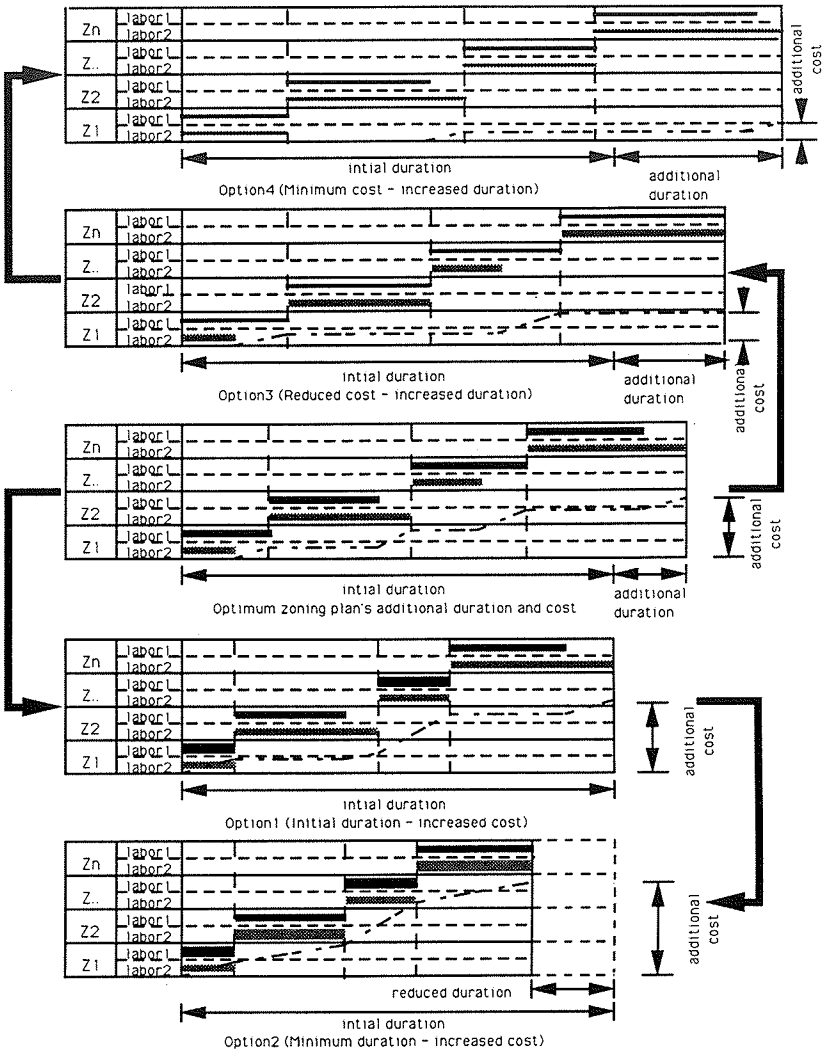


Figure 7. The four variations in the optimum zoning plan (option1, 2, 3 and 4)

As a test case, a small and relatively simple building was devised, having 10 stories and three by three spans. The building and block data are shown in Figure 8. The system comprises 4 major methods in its zone reasoning process. These are:

- *Average volume!* This method calculates the best crew size for bar fixer and form maker (represented in Figure 7 as labor 1 and labor 2) without zone planning, and results in zero waiting time for both trades. This means that costs due to additional labor and waiting time do not exist in this case. We define the labor crew size calculated by this method to be an optimum crew size. It is also used as an initial measure in the comparison with the labor size created by each zoning plan.

- *Find Zones!* This method creates zoning plans for consideration, called "cases". After supplying information on how many blocks a zone may include, and the relevant zoning 'starting block', the method begins to search appropriate neighbor blocks and construct zones.

- *Find best Case!* This method judges each case's executional validity, using concrete volume as the primary determinant. If the case is found to be a valid possibility, the method calculates the additional labor cost and additional duration which inevitably results from the zoning plan. At next step, the method selects the best case in all executable cases, using the additional cost and duration's evaluation, which is set up as a diagnosis. The best Case (zoning plan) is then displayed graphically.

- *Create Options!* as described previously

B1	B4	B7
B2	B5	B8
B3	B6	B9

plan

building name	Build2
story (floor)	10
initial construction duration (days)	300
pump car count	3
productivity of barfixer (ton/day)	0.5
productivity of formmaker (m^2/day)	12
bar fixer fee ($/day)	180
formmaker fee ($/day)	200
pump capability (m^3/day)	240
minimum concrete volume (m^3/day)	100
coefficient of additional and original barfixer fee	1.5
coefficient of additional and original formmaker fee	1.8
total bar weight (ton)	4500
total form area (m^2)	75000

Testcase building data

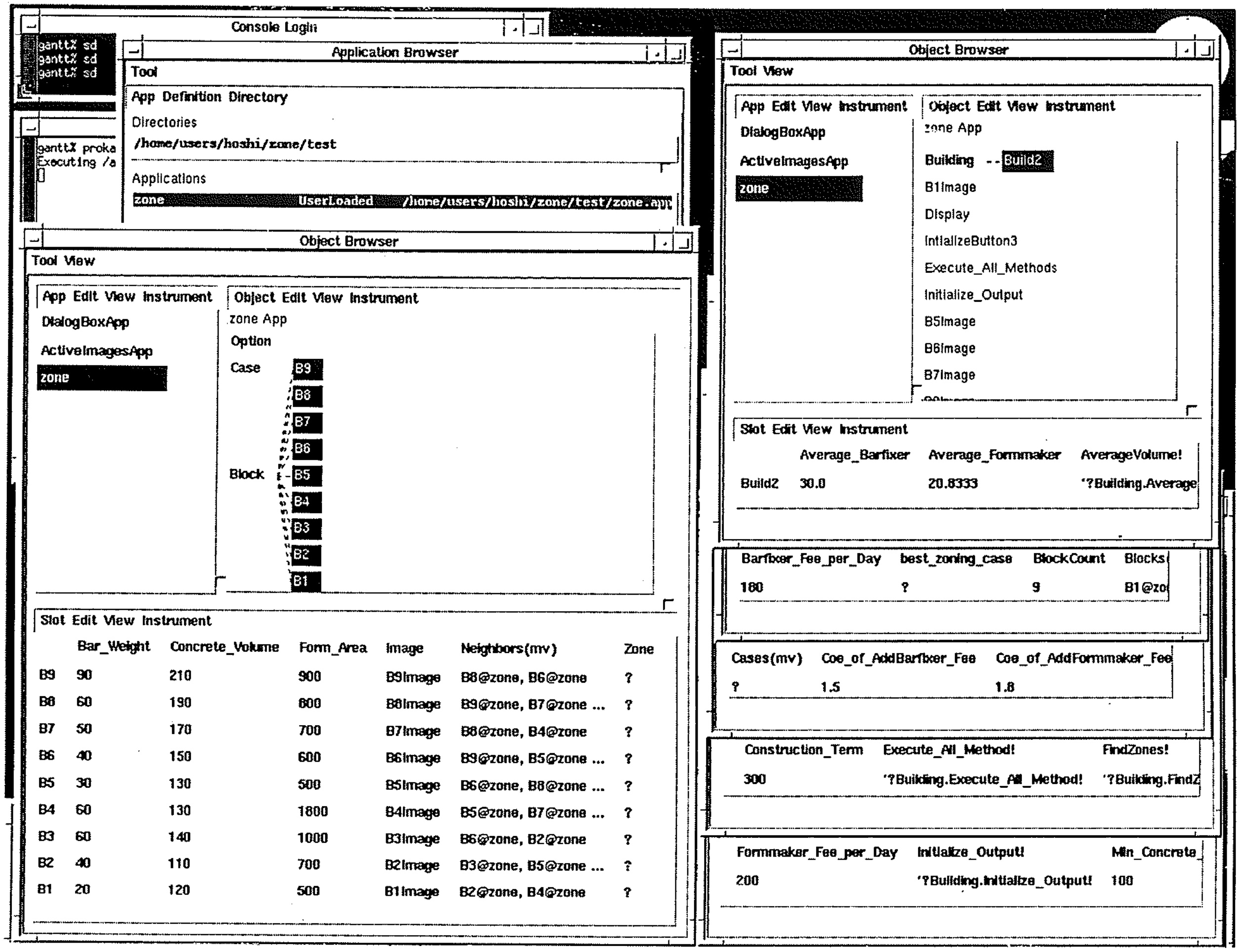
block name	bar weight	form area	concrete volume	neighbor blocks
B1	20	500	120	B2, B4
B2	40	700	110	B3, B5, B1
B3	60	1000	140	B6, B2
B4	60	1800	130	B5, B7, B1
B5	30	500	130	B6, B8, B4, B2
B6	40	600	150	B9, B5, B3
B7	50	700	170	B8, B4
B8	60	800	190	B9, B7, B5
B9	90	900	210	B8, B6
Total	450	7500	1450	

Testcase block data

Figure 8. Building and block data for the test case

4.3 Zoning Plan Output

This testcase outputs two major data as a result. One is the optimum zoning plan, displayed graphically, and the other is a tabular display of the additional cost and duration for each option in the optimum zoning plan. Figure 9 represents the system's input interface, with windows available for the input of crew, cost and block data. The bottom left-hand window shows a summary of the currently-defined blocks by block identifier, bar weight, concrete volume, form area, and block neighbors. Figure 10 is a screen view of the zone planner's output interface. The darkened area of this figure to the extreme bottom left is a schematic view of the current 'world view' of the zoning structure as defined by the system, and shows three zones: *Zone18, Zone19* and *Zone20*. The darkened control panel to the immediate right of the zone structure window has active buttons which can be used to invoke methods, and the window immediately above is a summary of the system operation in terms of the current case and the system's calculations on cost (*Option_cost* - in $), and added duration (*Option_term*).



Slot Edit View Instrument

	Bar_Weight	Concrete_Volume	Form_Area	Image	Neighbors(mv)	Zone
B9	90	210	900	B9Image	B8@zone, B6@zone	?
B8	60	190	800	B8Image	B9@zone, B7@zone ...	?
B7	50	170	700	B7Image	B8@zone, B4@zone	?
B6	40	150	600	B6Image	B9@zone, B5@zone ...	?
B5	30	130	500	B5Image	B6@zone, B8@zone ...	?
B4	60	130	1800	B4Image	B5@zone, B7@zone ...	?
B3	60	140	1000	B3Image	B6@zone, B2@zone	?
B2	40	110	700	B2Image	B3@zone, B5@zone ...	?
B1	20	120	500	B1Image	B2@zone, B4@zone	?

Slot Edit View Instrument

	Average_Barfixer	Average_Formmaker	AverageVolume!
Build2	30.0	20.8333	'?Building.Average

Barfixer_Fee_per_Day	best_zoning_case	BlockCount	Blocks
180	?	9	B1@zo

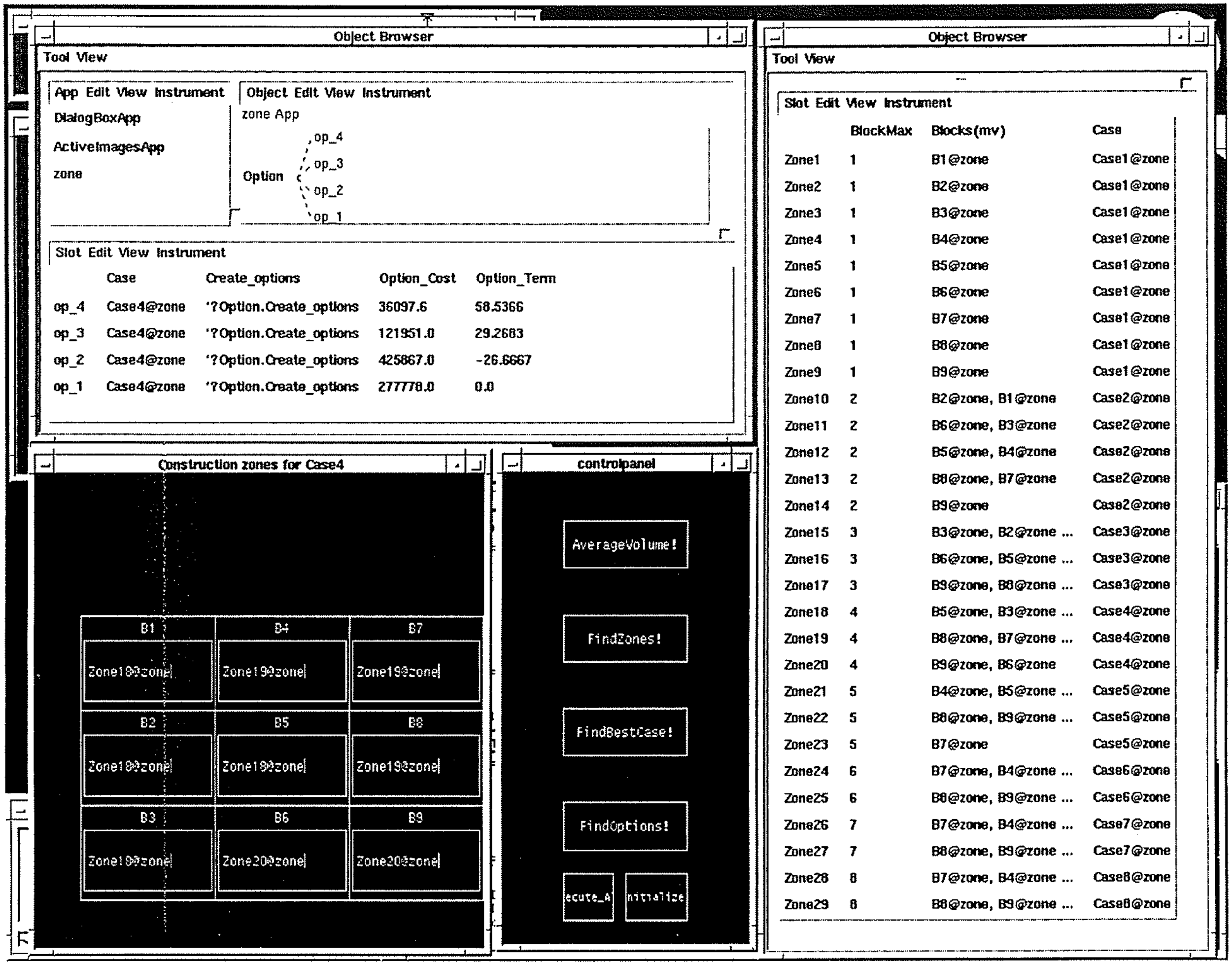
Cases(mv)	Coe_of_AddBarfixer_Fee	Coe_of_AddFormmaker_Fee
?	1.5	1.8

Construction_Term	Execute_All_Method!	FindZones!
300	'?Building.Execute_All_Method!	'?Building.FindZ

Formmaker_Fee_per_Day	Initialize_Output!	Min_Concrete
200	'?Building.Initialize_Output!	100

Figure 9. The Zone Planner's input interface

The zone planner is capable of use as a stand-alone zone definition and evaluation system, with facilities to input appropriate data, create zones on the basis of fundamental engineering knowledge represented within it, and reasoned output via its output interface. However, its true potential may lie in its integration within a design and planning environment, equipped with a central knowledge-rich model-based representation and reasoning system such as OARPLAN. OARPLAN is able to provide the zone planner with relevant data and information required in defining zones. The output can also be read by the OARPLAN system, and its explanation facility can be greatly improved through recourse to the well-founded techniques and protocols at the heart of the zone planner.



	Case	Create_options	Option_Cost	Option_Term
op_4	Case4@zone	'?Option.Create_options	36097.6	58.5366
op_3	Case4@zone	'?Option.Create_options	121951.0	29.2683
op_2	Case4@zone	'?Option.Create_options	425867.0	-26.6667
op_1	Case4@zone	'?Option.Create_options	277778.0	0.0

B1	B4	B7
Zone18@zone	Zone19@zone	Zone19@zone
B2	**B5**	**B8**
Zone18@zone	Zone18@zone	Zone19@zone
B3	**B6**	**B9**
Zone18@zone	Zone20@zone	Zone20@zone

	BlockMax	Blocks(mv)	Case
Zone1	1	B1@zone	Case1@zone
Zone2	1	B2@zone	Case1@zone
Zone3	1	B3@zone	Case1@zone
Zone4	1	B4@zone	Case1@zone
Zone5	1	B5@zone	Case1@zone
Zone6	1	B6@zone	Case1@zone
Zone7	1	B7@zone	Case1@zone
Zone8	1	B8@zone	Case1@zone
Zone9	1	B9@zone	Case1@zone
Zone10	2	B2@zone, B1@zone	Case2@zone
Zone11	2	B6@zone, B3@zone	Case2@zone
Zone12	2	B5@zone, B4@zone	Case2@zone
Zone13	2	B8@zone, B7@zone	Case2@zone
Zone14	2	B9@zone	Case2@zone
Zone15	3	B3@zone, B2@zone ...	Case3@zone
Zone16	3	B6@zone, B5@zone ...	Case3@zone
Zone17	3	B9@zone, B8@zone ...	Case3@zone
Zone18	4	B5@zone, B3@zone ...	Case4@zone
Zone19	4	B8@zone, B7@zone ...	Case4@zone
Zone20	4	B9@zone, B6@zone	Case4@zone
Zone21	5	B4@zone, B5@zone ...	Case5@zone
Zone22	5	B8@zone, B9@zone ...	Case5@zone
Zone23	5	B7@zone	Case5@zone
Zone24	6	B7@zone, B4@zone ...	Case6@zone
Zone25	6	B8@zone, B9@zone ...	Case6@zone
Zone26	7	B7@zone, B4@zone ...	Case7@zone
Zone27	7	B8@zone, B9@zone ...	Case7@zone
Zone28	8	B7@zone, B4@zone ...	Case8@zone
Zone29	8	B8@zone, B9@zone ...	Case8@zone

Figure 10. The Zone Planner's output interface

Figure 11 illustrates the results of the zone planner's operation on the test case. Case 4, which was chosen as the best case by the zone planner, has additional duration (26.6667 days) and cost ($255,111.0) as its attributes. To reduce either of them, the labor trade-off procedure, which is mentioned in section 4.2.6, executes in Case 4. For example, Option 1 maintains the initial duration (300 days) in substitution for increased additional cost ($277,778.0). Option 4 maintains the minimum additional cost ($36,097.6) in substitution for increased additional duration (58.5366 days).

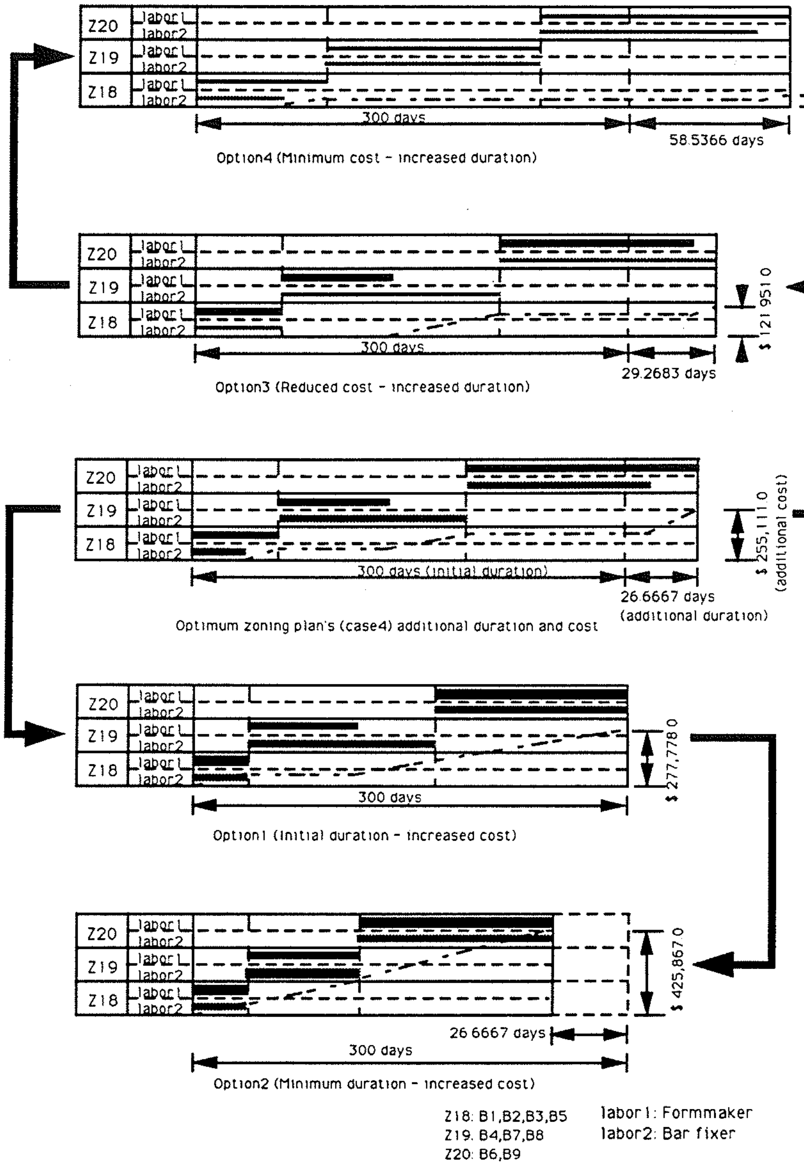


Figure 11. The Zone Planner's results on the test case

4.4 Linking the Two Systems

The OARPLAN and Zone Planner system are able to share data and project information. OARPLAN reasons about components of a facility, and the project activities required to properly sequence activities upon them. As such, it has a representation of objects, actions and resources, as well as more general information on project title, overall budget, location, etc. As an input, OARPLAN expects a list of components, annotated in a grid reference system, the relationships existing between them (e.g. *supported_by*), and the general project information that the CIFECAD system can provide [Fischer 1991]. For zoning purposes, the OARPLAN system requires a list of components *included* in predefined zones, and the required *sequence* of zone construction. The Zone Planner requires information on structure, relationships, cost, required duration, material volume for each component, and information relating to labor cost and productivity. At present, some of this information can be provided via the CIFECAD/OARPLAN environment, and in the future, most of the Zone Planner's required operating information could be provided via an enhanced project support environment incorporating the model-based representation and reasoning 'kernel', a symbolic CAD system such as CIFECAD, which is linked to an intelligent query system [Howard 1989], and a cost estimating facility. However, the current Zone Planning system is at the research stage, and much of its data is closely associated with the research test cases, and implemented directly within the system.

As an example of the utility of the Zone Planner/OARPLAN output link, output of the test case described in Figures 8 and 10 was used as the input to OARPLAN. Figure 12 describes the test case in terms of its structure and component naming convention. Figure 13 is an OARPLAN screen display of the zoned construction plan construction plan produced. In this example, the Zone Planner has provided OARPLAN with a list of components and a list of components aggregated into named construction zones. Although each block defined in the Zone Planner example comprises four columns and four beams, the OARPLAN system uses the simple heuristic that "once a component has been constructed/erected, it cannot be constructed/erected again". The result of the application of this simple rule is the dynamic definition of zone boundaries, depending on the given sequence of zone construction. In the example shown in Figure 12, with a given sequence of Zone18 -> Zone19 -> Zone20, Zone 20 will only include C12, C16, B14, B17, B21,B24, S6 and S9; the beams and columns having already been built.

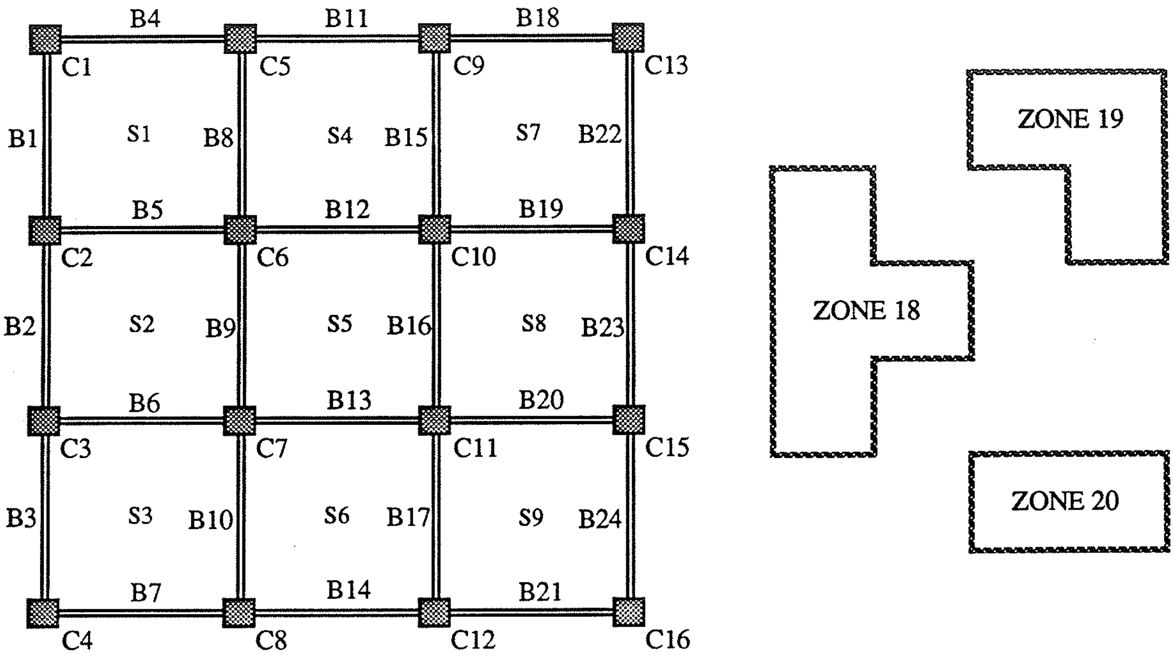
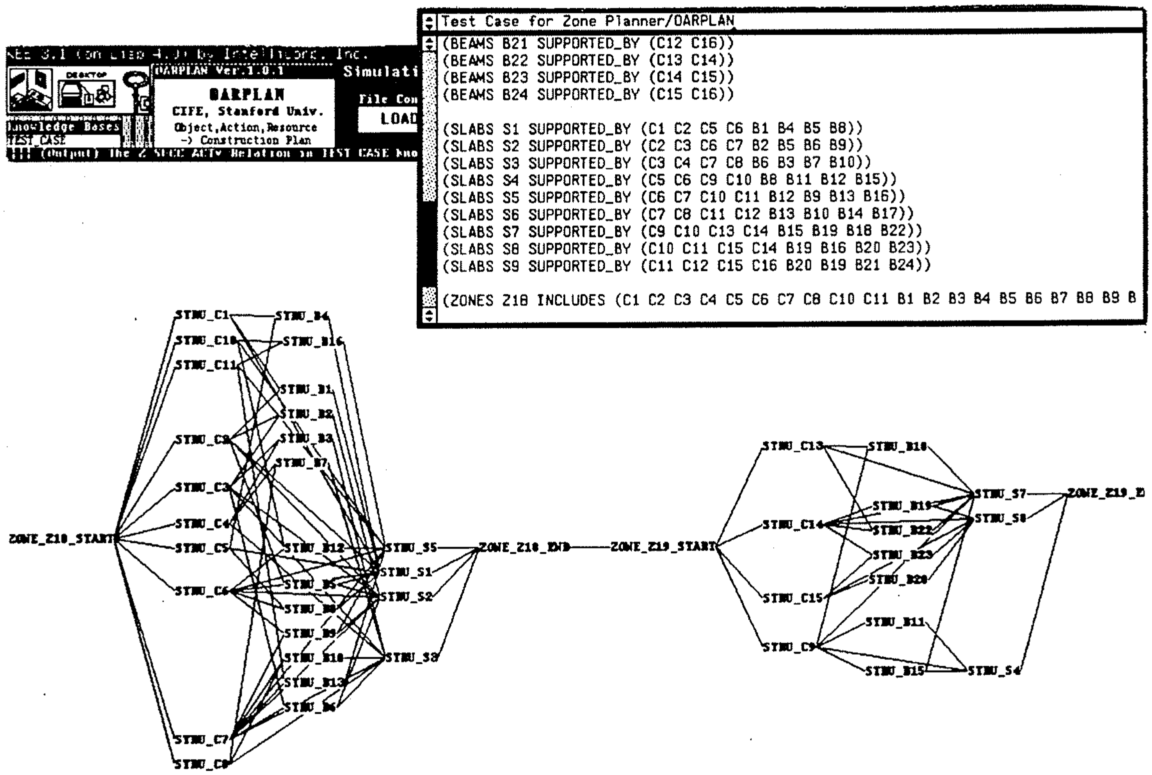


Figure 12. The structure and component layout for the test case



```
(BEAMS B21 SUPPORTED_BY (C12 C16))
(BEAMS B22 SUPPORTED_BY (C13 C14))
(BEAMS B23 SUPPORTED_BY (C14 C15))
(BEAMS B24 SUPPORTED_BY (C15 C16))

(SLABS S1 SUPPORTED_BY (C1 C2 C5 C6 B1 B4 B5 B8))
(SLABS S2 SUPPORTED_BY (C2 C3 C6 C7 B2 B5 B6 B9))
(SLABS S3 SUPPORTED_BY (C3 C4 C7 C8 B6 B3 B7 B10))
(SLABS S4 SUPPORTED_BY (C5 C6 C9 C10 B8 B11 B12 B15))
(SLABS S5 SUPPORTED_BY (C6 C7 C10 C11 B12 B9 B13 B16))
(SLABS S6 SUPPORTED_BY (C7 C8 C11 C12 B13 B10 B14 B17))
(SLABS S7 SUPPORTED_BY (C9 C10 C13 C14 B15 B19 B18 B22))
(SLABS S8 SUPPORTED_BY (C10 C11 C15 C14 B19 B16 B20 B23))
(SLABS S9 SUPPORTED_BY (C11 C12 C15 C16 B20 B19 B21 B24))

(ZONES Z18 INCLUDES (C1 C2 C3 C4 C5 C6 C7 C8 C10 C11 B1 B2 B3 B4 B5 B6 B7 B8 B9 B
```

Figure 13. The OARPLAN sequence for the test case. The inset window to the top right displays part of OARPLAN's input file, which it used to generate this plan

5.0 Conclusions

This paper has demonstrated the potential and problems associated with model-based planning in the construction industry. The OARPLAN system uses constraints associated with physical components as the basis for activity sequencing, and planning knowledge in the form of rules in the constraint satisfaction process. Using the power of knowledge-rich generic models able to capture detailed descriptions of project objects, actions and resources, it is possible to rapidly construct project plans as instantiations of these three concepts. each activity within the sequence is represented in the ensuing specific plan as intersections of the constituents components, actions and resources, but although the plan may be shown to be logically correct, the resulting plan complexity tends to reduce its immediate usefulness. Activity aggregation through physical zoning has the potential to provide realistic and reasoned work packages, which can be defined either by the designer working at the CAD workstation, inferred on the basis of trades requirements, resource availability and site/environmental conditions.

The implementation strategy for zones in the OARPLAN system has been to provide information within each activity, related to total (unzoned) dependency, and those precedences limited to within the automatically-imposed zone boundaries. In this way, aggregation can be initiated in one of a number of ways, and dynamic zone reconfiguration is possible as a result of the system's reasoning. However, the problem of initial zone definition remains. This paper has demonstrated a novel method of component-level aggregation based on accepted practises of cost or project duration optimization. The system developed in this study is capable of exploiting fundamental knowledge about equipment resources and the trades required for a particular project (in this case reinforced concrete construction). The results of aggregation are based on taxonomic calculations, and are thus open to validation in a way that heuristic methods are generally not. The method is readily incorporated into an AI planning system, providing a valuable and verifiable means of zone determination on the basis of given constraints and preferences.

Our work will continue to address the problems of zone determination and maintenance, which will become a major link between the output of model-based planning systems such as OARPLAN, and conventional project management systems. In many ways, aggregation is indeed a part of the scheduling problem, as it includes resources and temporal constraints. In this paper, we have concentrated on labor cost and project duration related to labor idle time in the determination of optimal zoning plans. However, there are other resource costs, constraints and distribution issues of relevance to practical zoning, e.g. production equipment cost and distribution, site condition, temporary stock yard space, activity sequence, etc. Among these, production equipment cost and its optimal distribution are strongly linked (to each other) with additional labor cost caused by labor idle time. In the future, it is planned to add such constraints and the knowledge required to more effectively and completely process zoning plans. The OARPLAN system is already interfaced to the Primavera™ project planning system, and intelligent zoning, able to take account of complex and interrelated resource and time considerations, will have a major impact on AI-levered project planning and management systems.

References

Chapman, D. (1987) "Planning for Conjunctive Goals," *Artificial Intelligence*, 32, 1987.

Darwiche, A.,Levitt, R.E. and Hayes-Roth, B. (1989) "OARPLAN: Generating Project Plans by Reasoning about Objects, Actions and Resources," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 2(3).

Fikes, R.E. and Nilsson,N.J. (1971) "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," *Artificial Intelligence*, 2, 198-208.

Fischer M.A. (1991). "Constructibility Input to Preliminary Design of Reinforced Concrete Structures". CIFE Technical Report #64, Center For Integrated Facility Engineering, Stanford University, CA.

Hayes-Roth, B. and Hewett, M. (1987). "Building Systems in the BB* Environment". In Englemore, R., and Morgan, A. (Eds.) *Blackboard Systems*, Addison-Wesley, London

Hendrickson, C, Zozaya-Gorostiza, C., Rehak, D., Baracco-Miller, E. and Lim, P (1987). "Expert System for Construction Planning". *ASCE J. of Computing in Civil Engineering* Vol. 1, No.4 253-269

Ito, K, Ueno, Y, Levitt, R.E. and Darwiche, A. (1989) "Linking Knowledge-based Systems to CAD Design Data with an Object-oriented Building Project Model". *CIFE Working Paper #7,* Stanford University.

Kartam, N.A. and Levitt , R.E. (1990). "Intelligent Planning of Construction Projects with Repeated Cycles of Operation", *ASCE J. Computing in Civil Engineering.*

Marshall, G., Barber, T.J. and Boardman, J.T. (1987). "A Methodology for Modeling a Project Management Control Environment," *IEEE Proceedings* 134 (Pt. D, 4).

Sacerdoti, E.D. (1973). "Planning in a Hierarchy of Abstraction Spaces," *Proceedings of IJCAI-73*, Palo Alto, CA.

Sacerdoti, E.D. (1975). "The Nonlinear Nature of Plans," *Advance Papers IJCAI-75*, Tbilisi, U.S.S.R.

Stefik, M.J. (1981a) "Planning with Constraints (Molgen: Part 1)," *Artificial Intelligence* 16.

Stefik, M.J. (1981b) "Planning and Meta-Planning (Molgen: Part 2)," *Artificial Intelligence* 16.

Tate, A. (1976). "Project Planning Using a Hierarchical Nonlinear Planner," *Department of Artificial Intelligence Research Rep.* No. 25, University of Edingurgh, Edinburgh, U.K.

Wilkins, D.E. (1988). "Practical Planning: Extending The Classical AI Planning Paradigm", Morgan Kaufmann, Los Altos, CA, 1988.

Winstanley G., Boardman J.T. and Kellett J.M. (1990) "An Intelligent Planning Assistant in the manufacture of Flight Simulators". *Proceedings of the ACME Research Conference held at the University of Birmingham, UK.*, 2nd-5th September.

Winstanley G., Kellett J.M., Best J.T. and Teskey F.N. (1989). "IDEAS - For Expert Systems". In *POP-11 Comes of Age: The advancement of an AI Language (Ed. James Anderson)*, Ellis Horwood Press, 1989.

Winstanley G., Chacon M. and Levitt R.E. (1992). "The Application of Model_based Planning Technology to Full-Scale Projects". CIFE Technical Report #66, Center for Integrated Facility Engineering, University of Stanford, CA.

Zozoya-Gorostiza, C., Hendrickson, C. and Rehak, D. (1989). "Knowledge-based Process Planning for Construction and Manufacturing", Academic Press, London.