On Solving Network Fixed Logic Problem — Resource-Driven CPM Scheduling

T. C. Chang and *C. William Ibbs*
Department of Civil Engineering, University of California, Berkeley, CA 94720

ABSTRACT: Logic relationships of activities in a CPM network can be categorized into highly dependent and largely independent. The current available CPM scheduling techniques, however, equally treat the logic relationships and, thus, is considered as one of its major deficiencies. This paper proposes an approach that starts from a preliminary network in which highly dependent activities are sequenced according to their physical constraints while largely independent activities are initially treated as possibly being parallel. Then, a resource allocation algorithm is suggested to modify the preliminary network and to set the "sequence" of the "parallel" activities when they compete using same resources that are limited. In addition, the impacts from both network related and project related factors are included in the algorithm.

1. INTRODUCTION

Fixed logic relationship in CPM network analysis has been considered one of its major deficiencies. The performances of certain construction activities are highly dependent due to their physical characteristics. Therefore, their logic relationships in a CPM network can be properly considered as "fixed". That a second floor slab cannot be constructed before the first floor columns are erected is an obvious example of this type of activities. Some activities, however, are largely independent; they are much more parallel or interchangeable in nature. The erection sequence of first floor columns (assume there are many columns), for example, is largely independent. They can be installed in parallel or sequentially. In case they are sequentially installed, the order probably will be interchangeable. In theory, activities of this kind have no constraints among themselves and can be simultaneously performed. In reality, they may be subject to some degree of external constraint due to insufficient resources or cost considerations. But when the resource and cost conditions are changed, the constraints should be changed, too. Unfortunately, most of the current available CPM software tools treat such constraints as fixed logic relationships which cannot reflect the constraint changes. This fixed logic approach deprives the largely independent activities of their parallel or interchangeable characteristics, and is a drawback in the CPM schedule planning and control algorithms.

In this paper, a new approach is proposed to solve this problem. This approach starts from a preliminary network in which highly dependent activities are sequenced according to their logic dependencies while largely independent activities are initially treated as possibly being parallel. External constraints, such as project duration, milestone deadlines, resource availability, and cost impacts, will then be considered in the resource allocation and levelling process. A network is considered complete only after this process is done. This process requires the input of resource profiles and will focus on the allocation of resources to achieve a practical and feasible resource-driven schedule both for planning and project controls purposes.

In our approach, schedule, resources and cost are considered an integrated "world". The impact of resource availability is evaluated when determining schedule. In future research, this impact will be reflected on project cost. In evaluating the impacts, fuzzy set and possibility theories are used; the information for the evaluation is organized using knowledge-based system concepts. The entire process is being implemented under an object-oriented programming environment to facilitate dynamic improvements and future extensions.

2. LOGIC RELATIONSHIP

Fixed logic relationship for largely independent activities is not proper, although it is proper for highly dependent activities. Figure 1 shows the fixed logic for six activities of column erection, assuming these columns are on the same floor and only two of them can be erected at a same time due to the constraint of concrete pouring crew and equipment (resource constraint). Using the current CPM technique to meet this constraint, the logic should be identified and fixed as those shown in Figure 1 before a schedule can be determined. The major disadvantage in this approach

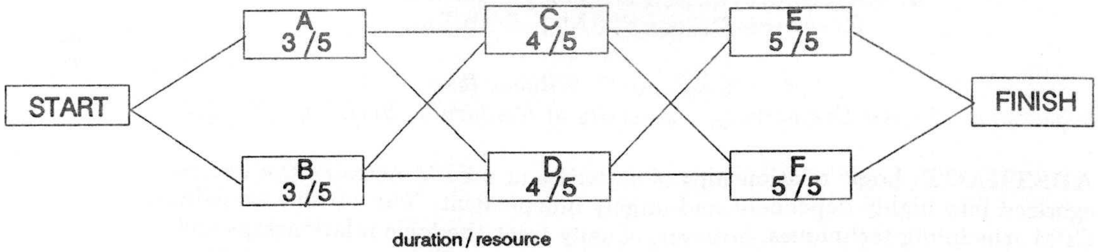

Figure 1: Fixed Logic Network

is that when the constraint changes or disappears the fixed logic relation can not automatically respond to it.

To improve this, we suggest that no logic dependency be assigned among the largely independent activities in a network. Thus, all largely independent activities are initially assumed to be performed simultaneously. This network, however, is only preliminary and a schedule based on this network will be considered as a preliminary schedule. Modification or "second pass scheduling" on the preliminary schedule must be imposed according to the impacts of external constraints such as resource availability and weather conditions. Figure 2 shows this second pass scheduling, where the precedence diagram indicates the preliminary schedule while the bar chart format shows the modified schedule by considering the resource constraint.

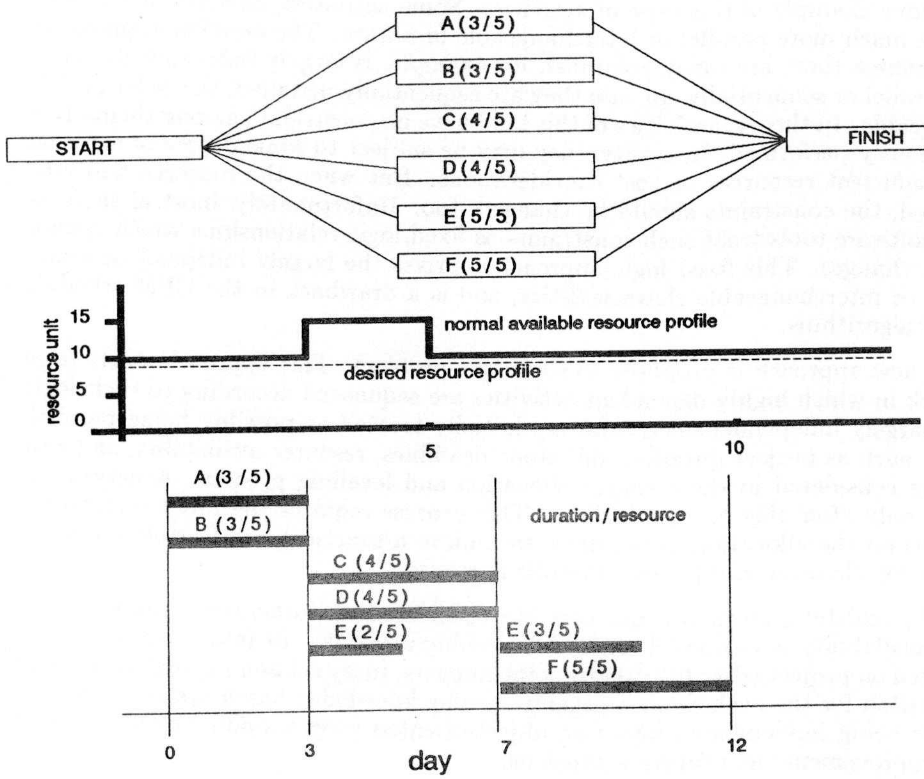

Figure 2: No Logic Dependency on Largely Independent Activities

This approach requires the input of resource profiles as shown in Figure 2, which will be discussed in detail later. Using this approach, any changes in the resource profiles will trigger a "second pass" and a new schedule will be obtained without touching the logic relation at all. Furthermore, the project team will know they can begin their work at any of the six activities by looking at the precedence diagram. The bar chart schedule is a suggested sequence. This indicates that the order of the six activities is actually interchangeable.

3. RESOURCE-DRIVEN SCHEDULING

The resource-driven approach proposed in this paper is based on the assumption that a preliminary schedule mentioned above exists. This is a reasonable assumption since the preliminary schedule can be established by a scheduling engineer with the idea that no logic dependency be assigned among the largely independent activities. It is also possible to use automatic schedule generators such as GHOST [6], CONSTRUCTION PLANEX [9], and other research projects currently being conducted in University of Illinois and Stanford University, for example. Given this preliminary schedule or network, this section shows how to impose external constraints to complete a final and feasible construction schedule.

3.1 Basic concept of resource allocation

The basic concept of resource allocation [5] can be easily explained with the help of the sketch shown in Figure 3, where P = the candidate set of activities which have all their logical precedence constraints met, and, therefore, are candidates for resource assignment and initiation of physical work; S = the set of activities scheduled during a time interval; and C = the set of completed activities.

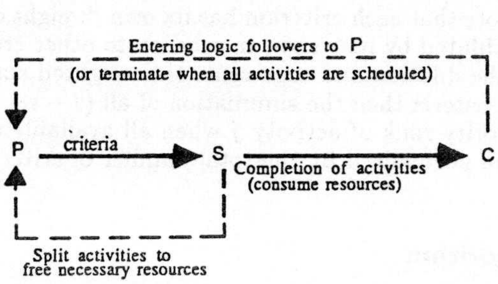

Figure 3: Basic Concept of Resource Allocation

Assume that a list of criteria has been established for allocating limited resources in a specified time frame. Then, the arrow between P and S indicates that each "candidate" (activity) in P is checked against the list of criteria to see if the criteria are met. If the criteria are met, it is scheduled according to its network constraints. If not, a check is made of the next candidate in set P. In the case where two or more candidates meet the criteria and they require the same resources that are insufficient to start all of them at the same time, priorities must be set before any one of them can be started. Each candidate in P will be scheduled according to the priorities. Determining these priorities thus can be seen to be a key issue in the process of resource allocation.

The criteria can be categorized into two groups: (1) external criteria (project related) such as weather conditions, resource availabilities, changes, and changed conditions; and (2) internal criteria (schedule related) such as total float, impact of losing float and the impact of downstream resource requirements. This section focuses on the external criteria; the internals will be discussed later in section 3.2.

Assume that n external criteria exist. Let $C_i (i = 1, 2, ..., n)$ denote them. The importance of these criteria will depend on time, project location, project characteristics, project manager's preferences, and other similar factors. The importance does not simply mean that C_i is either important or not important (what is known as a "crisp set" concept). Rather, it can be better treated as a "linguistic variable", or as the "degree" of importance or the "weight" of importance (fuzzy set concept). Symbolically, let w_i denote the measure of the "weight" for the importance of criterion C_i. For example, if there exists three criteria C_1, C_2 and C_3, they may be described such as criterion C_1 is *very* important (w_1 = very high), C_2 is *more or less* important (w_2 = more or less high), and C_3 is *less* important (w_3 = low). In [2], a method based the ***possibility theory*** [7] has been developed to evaluate the w_i.

From the basic concept of resource allocation, it has been shown that each candidate in P should be checked against the list of criteria to see if the criteria are met. More specifically, this checking determines the ***susceptibility*** of each candidate to each criterion. Let s_{ij} denote the ***susceptibility*** of activity j to criterion C_i. And, let s_{ij} range between 0 and 1 for convenience. So when $s_{ij} = 0$ it means that activity j is not susceptible to C_i, and $s_{ij} = 1$ means that activity j is "absolutely" or "totally" susceptible to C_i. When s_{ij} = any other intermediate value, the activity j has some "degree" of susceptibility to C_i. The ***susceptibility*** (s_{ij}) can be determined using ***fuzzy logic*** concept [1, 8]. A fuzzy reasoning system has been developed for this purpose in [4].

Knowing w_i and s_{ij}, the priority rank (P_j) for activity j in the candidate set P can be defined as follows:

$$P_j = \frac{\sum_{i=1}^{n}(1 - s_{ij} \cdot w_i)}{n} \tag{1}$$

The term $(1 - s_{ij} \cdot w_i)$ can be interpreted as a measure of activity j's priority against criterion C_i. For example, assume that a criterion C_i called "severity of rain" has been established, and an activity j called "placing concrete" will be done in an exposed area where no protection from rain is provided. Therefore, the activity is highly susceptible to rain (s_{ij} is large or close to 1). The priority to schedule the activity under the probability of impact by rain should consequently be low. The term $(1 - s_{ij} \cdot w_i)$ reflects this situation and is defined as a relative value to measure the priority against each criterion. Note that each criterion has its own "weight of existence" (w_i). So, the susceptibility (s_{ij}) would be diluted by its weight in relation to other criteria. The product of s_{ij} and w_i gives the measure of the dilution and represents the weighted susceptibility of activity j to criterion C_i. If there exist n criteria then the summation of all $(1 - s_{ij} \cdot w_i)$ for $i = 1, 2, \cdots, n$ yields a relative value for the priority rank of activity j when all available criteria are taken into account. The priority rank is then normalized by the total number of criteria (n) to yield a value that is in the range from 0 to 1.

3.2 Resource Driven Algorithm

Once the priority ranking P_j in Eq. (1) has been resolved, limited resources can be allocated to the activities in candidate set P to initiate their physical work or, in a planning phase, to schedule their starting and finishing dates. In other words, the schedule for the activities can be prepared according to the priority ranking, the preliminary network, and the limited resource pool. Such a schedule, however, may not be completely satisfactory because the ***internal factors*** such as an activity's criticality, total float, downstream resource effect and other such issues are not considered. (Ignoring such consideration may cause the project duration to be extended beyond that desired.) In addition, the resulting resource profiles may not be desirable because of excessive variation. To correct these two potential deficiencies it is necessary to include the internal factors as allocation criteria and to combine both allocation and levelling techniques in the scheduling process. This section proposes an algorithm which considers both internal and external factors in the allocation process and allows user control in the levelling mechanism.

3.2.1 Factors under User Control

Desired Resource Profile — Crandall [5] proposed a list of factors that can be controlled by a project manager during the scheduling process. A series of equations are proposed to evaluate these factors in [1]. This section briefly reviews these factors and related terminologies that will be used in the algorithm explained in section 3.2.2.

Figure 4 shows five different resource profiles that are used to organize the resources available for scheduling construction activities. The *desired resource profile* (DRP) is subjectively defined by the project team to show the efficient buildup of resources and their utilization. It is conditioned only by the team's judgement and not necessarily by the network logic constraints. Users can establish any shape for the profile depending on their particular needs and experience. However, two constraints must be followed: (1) the area under the DRP must be at least equal to the resource demand under the *early start profile* (ESP), and (2) the DRP is not allowed to exceed the *maximum available resource pool* (MAP). The ESP is the resource accumulation profile based on the *early start* (ES) of the activities that are scheduled by the logic constraints in the preliminary network without considering the limited resource problem. The MAP is the maximum limit of resource which is available in a specified time frame. Beyond this limit, no resource can be obtained. So the premium cost can be considered as infinite when resource requirements exceed this limit. The premium cost resource is the resource that exceeds the *normally available pool* (NAP).

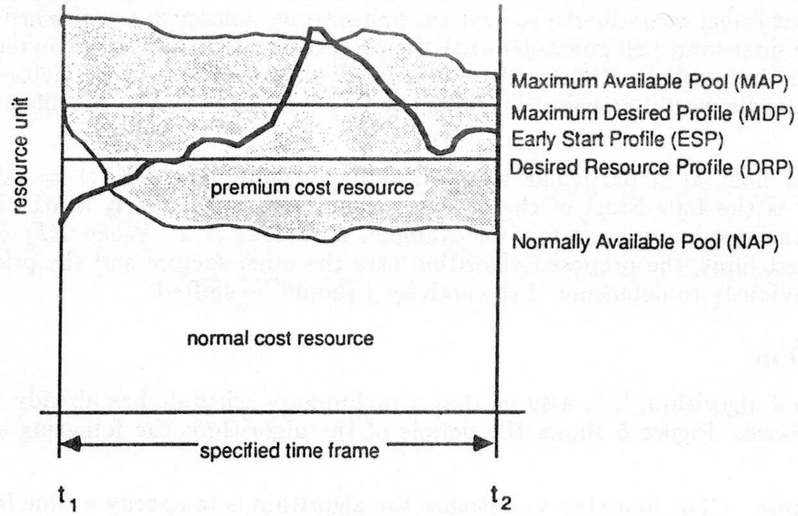

Figure 4: Concept of Resource Profiles

The *maximum desired profile* (MDP) can be determined by analysis of actual resource availability given a specific value of maximum premium cost. Alternatively, once the DRP has been established by the users, the MDP can be defined as $MDP = DRP \cdot (1 + max\%)$, where max indicates the maximum percentage which the users are willing to pay for an extra resource even at a premium cost. High max values will lower the impact of exceeding the MDP and, therefore, reduce the delay of activities.

Effect of Deviation — Any deviation from the DRP is undesirable though sometimes inevitable. Thus, the deviation should be minimized whenever possible. To minimize the deviation, it is first necessary to measure its effect. Let $E_{d_j}(t)$ denote the effect of deviation from the DRP when considering the additional resource required by an activity j at particular time t, then the following equation will give a practical measurement [1]:

$$E_{d_j}(t) = K_d \frac{MD_j}{(max\%) \cdot DRP} \tag{2}$$

where MD_j is maximum deviation (above the DRP) when assigning the required resource to the activity j. In other words, let $D_j(t)$ denote the deviation ***above*** the DRP when assigning the required resource to activity j at time t, then MD_j can be computed as $max\{D_j(t') \mid t' = t \text{ to } t + \text{duration of } j\}$. The $D_j(t)$ can be computed using the following equation:

$$D_j(t) = ARP(t) + R_j(t) - DRP(t) \tag{3}$$

where $ARP(t)$ = Actual Resource Profile at time t prior to consider the activity j; and $R_j(t)$ = Resource required by the activity j at time t; and $DRP(t)$ = the Desired Resource Profile value at time t. For facilitating the explanation of the resource driven algorithm (section 3.2.2), the term $ARP(t) + R_j(t)$ will be referred to as $RRP(t)$ to stand for the *resultant resource profile*, i.e., $RRP(t) = ARP(t) + R_j(t)$.

The K_d in Eq. (3) is a weight which is controlled by users and is their subjective judgement. This weight helps the users to adjust the RRP and may affect the final schedule.

Effects of Losing Float and Downstream Resource Usage — When an activity j is delayed, it affects not only its own "path float" but also floats on the "downstream" paths of followers on the logical network. In addition, the downstream resource usage is also affected. These effects will be referred to as $E_{f_j}(t)$ and $E_{r_j}(t)$, respectively, in the resource driven algorithm (section 3.2.2). Methods to evaluate these two factors are available in [1].

Float Limit — Float limit is a critical "threshold" of an activity or work package. At a given time t, an activity j with a remaining total float ($RF_j(t)$) equal to or less than this limit will be considered as critical and should not be delayed. In the proposed algorithm (3.2.2), the criticality (as represented by the float limit) is considered to have the first priority among all possible criteria. Note that a non-negative float limit will guarantee that the project duration will not be extended because the activities on the critical path(s) will be scheduled regardless any other factors. A negative float limit can also be used if a project is allowed to be extended beyond a specified time frame.

The remaining total float at a particular time t can be computed as $RF_j(t) = LS_j - max\{t, ES_j\}$ where LS_j is the Late Start of the activity j; and ES_j is the Early Start of the activity j; and max is the maximum operator, for example, $max(1,2) = 2$. When RF_j of an activity j exceeds the float limit, the proposed algorithm uses the other factors and the priority ranking P_j described previously to determine if the activity j should be shifted.

3.2.2 The Algorithm

To start the proposed algorithm, it is assume that a preliminary schedule has already been available as explained above. Figure 5 shows the details of the algorithm; the following steps explain it.

A. Specify Time Frame — The first step to initialize the algorithm is to specify a time frame within which the resource allocation and levelling will be considered. From the preliminary schedule, all activities logically available for scheduling in the period will be examined, and priorities will be assigned to them according to a list of criteria.

B. Generate Activity Set — When a time frame is specified, the next step is to organize all the activities in this time frame into different groups or sets for allocating resources. Three different activity sets used in Figure 5 are defined as follows: (1) P_0 is a set which contains all activities from the preliminary schedule that are scheduled to be done within the specified time frame $[t_{k-1}, t_k]$. (2) P_t is a candidate set at time t which is generated from the P_0. In other words, P_t contains all activities that have logical precedence constraints met at time t, and, therefore, are candidates for resource assignment. (3) P_t^{nc} is a candidate set for time t in which all activities are *non-critical*, and a subset of P_t. An activity is *critical* when its remaining total float is equal to or less than the float limit.

C. Find Priority Ranking — Once the set P_0 has been generated, the next step is to determine the priority rank P_j (Eq. (1)) for all activity j in P_0. An expert system called *Priority Ranking* has been developed in [3] to help evaluating the priority rank P_j. The P_j is then used to sort the P_t^{nc} as shown in Figure 5.

D. Allocation and Levelling Procedure — The allocation and levelling procedure can be started once the P_j's for all j in P_0 have been determined. The steps shown in Figure 5 are self-explanatory.

3.2.3 Impact on Activity Duration

It is important to consider the impact on activity duration due to external forces such as weather conditions, material delays and changed conditions. Although this is not shown in the steps of the above algorithm, it is not difficult to include this type of impact in the actual programming of the algorithm. To do this, a "penalty for inefficiency" should be considered when the algorithm schedules an activity with low priority value. This is necessary because the low priority value comes from a high susceptibility (s_{ij}) value. High susceptibility can have significant impact on productivity when conducting the activity. For example, assume the impacts of rain and temperature are considered as external criteria. And assume that an activity j is very susceptible to rain and temperature (e.g., it is an outdoor activity). In addition, assume that the weather conditions are heavy rain and low temperature during the performance period of the activity j. Then if resources are allocated to activity j and j is performed under heavy rain and low temperature the efficiency (or productivity) of performing activity j will likely be low. One possible way to measure this impact is to extend the duration of activity j (D_j) by a *penalty factor* $1/(1 - K(w_i \cdot s_{ij}))$:

$$\text{New } D_j = \frac{\text{original} \quad D_j}{\prod_{i=1}^{n}(1 - K \cdot w_i \cdot s_{ij})} \tag{4}$$

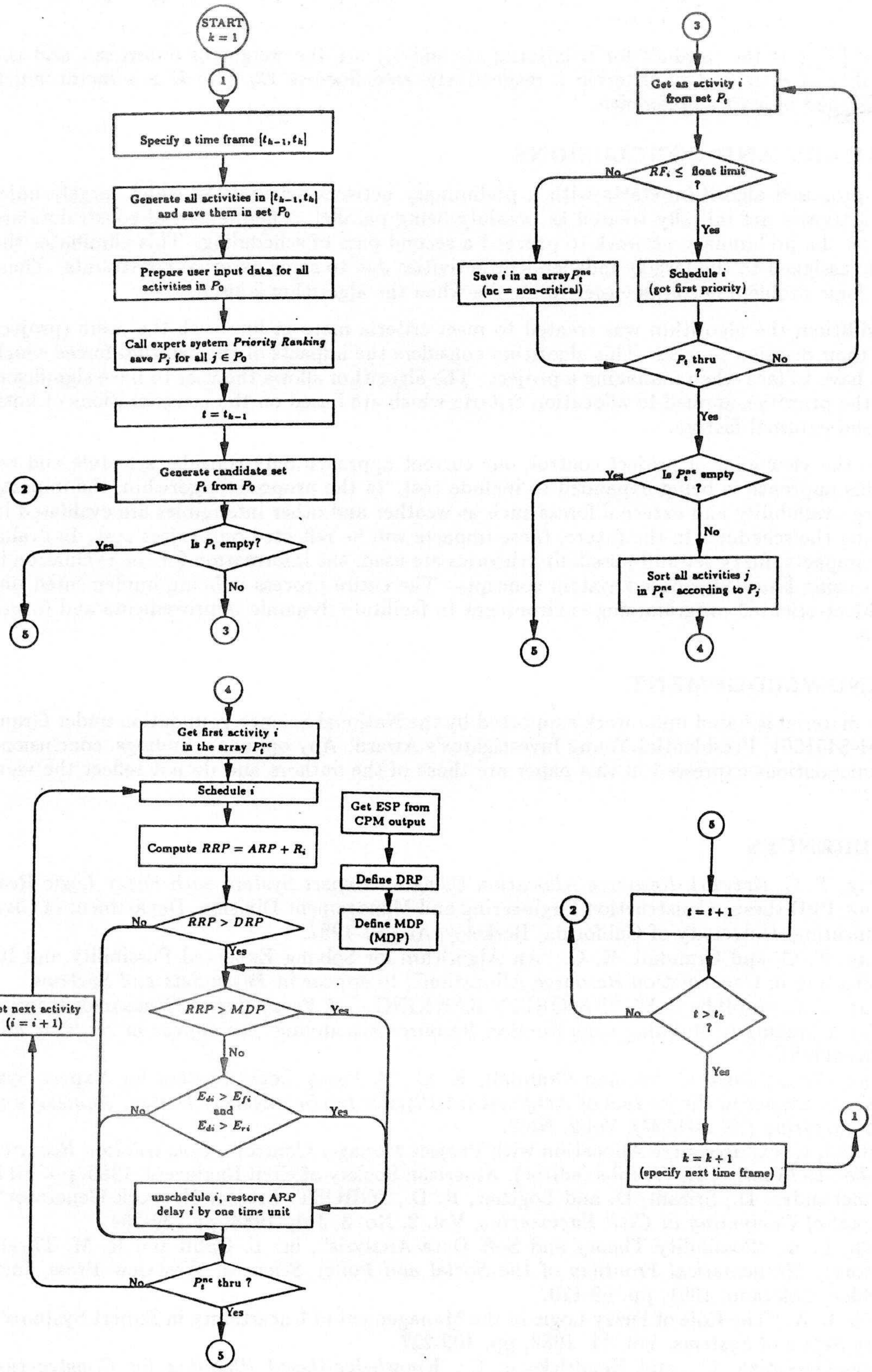

Figure 4: An Algorithm for Resource Allocation and Levelling

where the $\prod_{i=1}^{n}$ is the product for n criteria; w_i and s_{ij} are the weight of criterion i and the susceptibility of activity j to criterion i, respectively, as defined in Eq. (1); K is a factor which can be assigned by user's preference.

4. SUMMARY AND CONCLUSIONS

The proposed algorithm starts with a preliminary network schedule in which largely independent activities are initially treated as possibly being parallel. Then, external constraints are imposed on the preliminary network to proceed a second pass of scheduling. This eliminates the fixed logic assigned to the largely independent activities due to some external constraints. Thus, the fixed logic problem becomes easier to handle when the algorithm is used.

In addition, the algorithm was created to meet criteria more in line with the users (project team) in their decision process. This algorithm considers the impacts of the external forces which the users have to face when managing a project. The algorithm allows the user to have significant input to the priorities applied to allocation criteria which are based on the considerations of both internal and external factors.

From the viewpoint of project control, our current approach only consider schedule and resource; this approach is being expanded to include cost. In the proposed algorithm, the impacts of resource availability and external forces such as weather and other intangibles are evaluated in determining the schedule. In the future, these impacts will be reflected on project cost. In evaluating the impacts, fuzzy set and possibility theories are used; the information for the evaluation is organized using knowledge-based system concepts. The entire process is being implemented under an object-oriented programming environment to facilitate dynamic improvements and future extensions.

5. ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under Grant No. MSM-8451561, Presidential Young Investigator's Award. Any opinions, findings, conclusions or recommendations expressed in this paper are those of the authors and do not reflect the view of NSF.

6. REFERENCES

1. Chang, T. C. *Network Resource Allocation Using an Expert System with Fuzzy Logic Reasoning*, PhD thesis, Construction Engineering and Management Division, Department of Civil Engineering, University of California, Berkeley, August 1987.
2. Chang, T. C. and Crandall, K. C. "An Algorithm for Solving Expected Possibility and Its Application in Construction Resource Allocation", to appear in *Fuzzy Sets and Systems.*
3. Chang, T. C. and Ibbs C. W. "PRIORITY RANKING — A Fuzzy Expert System for Priority Decision Making in Building Construction Resource Scheduling", to appear in *Building and Environment.*
4. Chang, T. C., Ibbs, C. W. and Crandall, K. C. "A Fuzzy Logic System for Expert Systems", to appear in the journal of *Artificial Intelligence for Engineering Design, Analysis and Manufacturing (AI EDAM)*, Vol.2, No.2.
5. Crandall, K. C. "Resource Allocation with Project Manager Control", *Construction Research Applied to Practice*, C. W. Ibbs (editor), American Society of Civil Engineers, 1985, pp. 1-17.
6. Navinchandra, D., Sriram, D. and Logcher, R. D., "GHOST: Project Network Generator", *Journal of Computing in Civil Engineering*, Vol. 2, No. 3, July 1988, pp.239-254.
7. Zadeh, L. A. "Possibility Theory and Soft Data Analysis", in: L. Cobb and R. M. Thrall, (editors), *Mathematical Frontiers of the Social and Policy Sciences*, Westview Press, Inc., Boulder, Colorado, 1981, pp. 69-129.
8. Zadeh, L. A. "The Role of Fuzzy Logic in the Management of Uncertainty in Expert Systems", *Fuzzy Sets and Systems*, Vol. 11, 1983, pp. 199-227.
9. Zozaya-Gorostiza, C., and Hendrickson, C., *Knowledge-Based Planning for Construction Projects*, PhD thesis, Department of Civil Engineering, Carnegie-Mellon University, (Report No. R-88-170), April 1988.