Pergamon

Computers ind. Engng Vol. 35, Nos 3–4, pp. 507–510, 1998
© 1998 Elsevier Science Ltd. All rights reserved
Printed in Great Britain
PII: S0360-8352(98)00145-4 0360-8352/98 $19.00 + 0.00

$(DP)^2SM$: A Dynamic Approach to Resouce Constraint Project Scheduling

HOMAYOUN KHAMOOSHI
Nene University College Northampton, UK

The purpose of this paper is twofold, first to introduce and evaluate a dynamic priority scheduling model developed in this research for solving the resource constraint project scheduling problem, second to introduce an improvement made upon the first model by cross breeding Dynamic Programming with the Dynamic Priority Scheduling Method (DPSM). The second model called Dynamic Priority Dynamic Programming Scheduling Method [$(DP)^2SM$] aims at optimising the staged resource allocation decisions in DPSM. DPSM divides a project into phases (cycles) the length of which depend on the duration of the project and the period of clock cycle selected. The scheduling process starts by allocating resources to the first phase/cycle using a variety of policies, then the best schedule is selected based on an objective function. The process continues till all the activities are scheduled. In DPSM the interaction between phases is ignored while the decisions of each phase or cycle will affect all the remaining phases. Using $(DP)^2$ SM it may be possible to improve the quality of a schedule and reduce the duration of a project by optimising the overall project schedule. © 1998 Elsevier Science Ltd. All rights reserved.

Key words: project management, resource, schedule, quality, flexibility,dynamic programming.

1. INTRODUCTION TO DYNAMIC PRIORITY SCHEDULING METHOD (DPSM)

When a project is to be scheduled the recommended procedure of most researchers (Davis 1973) is to schedule the project by different priority rules and choose the one with minimum project delay (percentage increase with respect to original critical path length), then during project life cycle or monitoring, schedule again and choose the best one. If for example in the planning phase " minimum total float " is selected as the best priority rule, then during monitoring " minimum late start " might be preferred scheduling policy. So it could be concluded that there is no tendency or preference toward keeping the policy uniform throughout the project life cycle. Let us look at the problem from another point of view. Consider an engineering project from the very beginning to the end ; it could be divided into few subnets(smaller networks) each including hundreds of activities. fig. 1 shows an engineering project divided into three subnets.

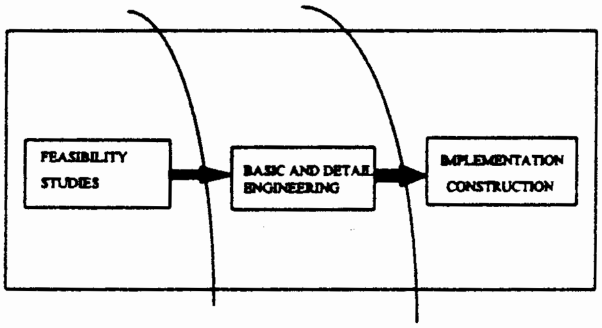

Fig. 1: Project breakdown into phases/subnets

If the project in fig. 1 is to be scheduled and a heuristic priority rule is to be applied all the way through the project, we might be sub-optimising the problem, while better solutions could be possible. The reason for that lies behind the fact that not necessarily the same scheduling policy would be the best for all subnets. If this logic is acceptable then why should not we divide a project into parts/phases and decide on each part applying selected priority rules? This clue led the author to a new approach to heuristic-based scheduling problem named "Dynamic Priority Scheduling Method" or in short DPSM. In DPSM the project will be divided into sections; this could be division based on different subnets if they run serially that is one follows the other or a more general approach is to use a time section or so called clock cycle to divide a project into parts, sections or phases. For each section or piece a search for the best schedule is carried out looking for the desired objective function (not necessarily minimum duration). The procedure starts with the first section then continues to the second section and so on till all the activities are scheduled. The objective function could be defined as desired . In this study four different options were considered: 1-MAXIMUM NUMBER OF ACTIVITIES SCHEDULED PER CLOCK CYCLE, 2-MAXIMUM RESOURCE UTILISATION PER CLOCK CYCLE, 3-MINIMUM FLUCTUATION PER CLOCK CYCLE 4-MULTI OBJECTIVE SCHEDULING.

Considering the case of unconstrained resource project scheduling, the objective is not to minimise duration but to produce a good quality schedule incorporating better use of resources, better distribution of float (flexibility) etc.. DPSM could be used to search for and fulfil such objectives.

2. DYNAMIC PRIORITY DYNAMIC PROGRAMMING SCHEDULING METHOD $(DP)^2SM$

DPSM divides a project into phases (cycles) which depend on the duration of the project (length of critical path) and the clock cycle selected. The scheduling process starts by allocating resources to the first section or cycle using a variety of policies, then the best schedule is selected based on an objective function. The process continues over the second phase assuming the first phase decisions are made. The same algorithm applies to the third, fourth and the rest of the cycles till all the project is covered. The point to remember is that in DPSM the interaction between phases is ignored; in other words it does not aim at overall optimisation of the system. If the priority rule selected for the first phase is "late finish", the setting or schedule produced might be different from a schedule produced by "total float" if this was used as the priority policy. Thus, the succeeding activities are affected by the decisions made at the first phase or section (clock cycle). In general the decisions of each phase or cycle will affect all the remaining phases. It may be possible to improve the quality and duration of a schedule by trying to optimise the overall project schedule. Having that in mind, it seems that one can find a similarity between scheduling a project through phases using a number of priorities and the classical dynamic programming approach to stagewise decision making problems.

Dynamic programming is an approach that permits decomposing one large stagewise mathematical model into a number of smaller problems. Once all the smaller problems have been solved, we are left with one optimal solution to the large problem. So the technique can be applied to a decision problem that is multistage in nature. Though in project scheduling there is no evidence of stagewise decision making one can not deny that resource allocation decisions made at each phase will affect the following decisions. If we consider each decision as a stage, we will end up with numerous stages. In DPSM the project is divided into phases (stages) which incorporate a number of resource allocation decisions within each cycle. In this proposed approach the stages are assumed to be the same as clock cycles/phases in DPSM which could exist naturally or be generated artificially. The most important feature of the dynamic programming approach is that the smaller problems or stages are not considered to be independent and the impact of decisions made in each stage on the remaining stages is taken into consideration. Let us consider the DPSM model and see how we can apply dynamic programming techniques to improve the solutions (schedules) produced. If we use a clock cycle to divide the project into three sections (equivalent to three stages in dynamic programming) and make use of three different priority rules, the schematic presentation of the model will look like fig. 2.

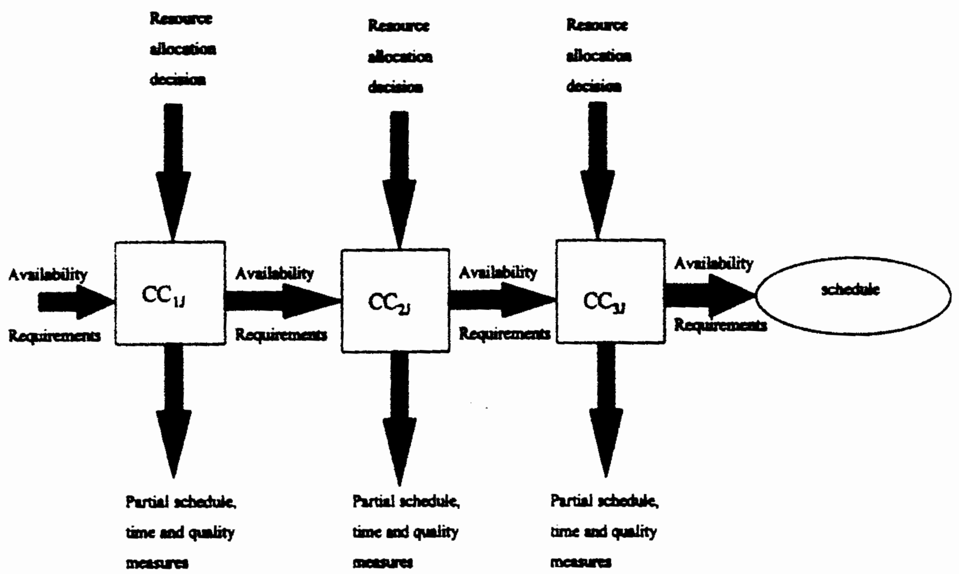

Fig. 2: DPSM analogy with dynamic programming stages

CCij shows the clock cycle i with priority policy j. It is important to note that, contrary to the classical dynamic programming approach for solving large stagewise mathematical models, in applying dynamic programming to project scheduling the solutions produced for each stage are not based on an optimal approach, but partial schedules are produced by heuristic scheduling policies and as such the overall solution should not be expected to be an optimal one. However by developing a dynamic programming model for project scheduling the schedules may be improved.
For demonstrating this model further, a case of three priority rules is used in the pictorial presentation of fig. 3 to show the compatibility of the problem with dynamic programming model. Also a clock cycle is selected so that the project is divided into three stages as the smaller the number of phases the easier to schedule the project.

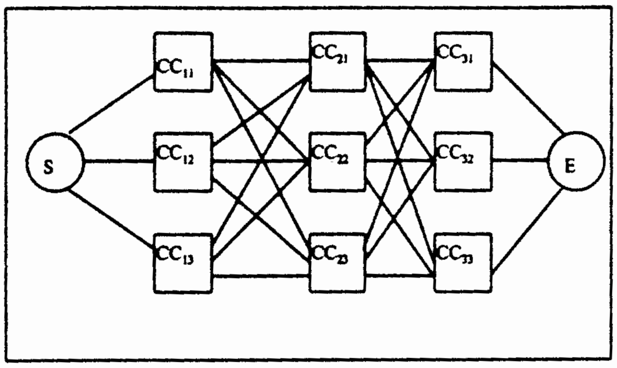

Fig. 3: $(DP)^2SM$, a case of 3 priorities with 3 clock cycles analogue to travel salesman problem

In developing a dynamic programming model for resource constrained scheduling problems there may be some complications in terms of the number of priority rules to be used and also the duration of the clock cycles. The conclusions drawn from the author's research make it clear that there is no point in incorporating a large number of priorities; including a maximum of five good priority rules should suffice.

3. EXPERIMENTAL DESIGN AND DATA COLLECTION

The general policy of the experiment was to study the practical requirements of project planning and scheduling. Therefore, although artificial networks are also used in the experiments the intention has been to use large projects with muti-resource requirements. In this research a comparative study of performance of different heuristic rules is carried out while the measure of performance or efficiency of a scheduling rule has not only been project duration or project delay, but also other measures are introduced (see appendix 1 for detail discussion on performance measures). Also it is important to note that the objective of the comparative study was not to compare traditional priority rules but to study the performance of newly developed models. Data collection was done in three phases: the collection of data through generation of artificial networks, selection of a number of test problems through the survey of literature and collection of some real life project. In all, 30 test problems were collected using a program generating networks. A number of networks used in the experiment were collected through literature survey. In this section the problems and projects presented by different authors and researchers in the books and papers were collected. These problems were about 30 in number. Quite a large number of real life projects were collected through the "applied project planning and scheduling courses" conducted by the author in a variety of industries. So these projects are the ones which could be counted on much more than the artificial and theoretical networks. The number of real life projects was 40. The number of activities in a project varied from ten to 300 and the number of resources from one to fifteen.
Selecting the priority rules for the experiment was a difficult task. Diversity of point of views of different researches made it more complicated to decide upon the priority rules to be used in the experiment .Therefore two approaches were made . First a survey of literature and second a study of project management packages available to the author (on the market). Then the best of all were selected. The priority rules selected to be used in the experiment were among the best reported in the literature .The priorities used include: 1- Total Float 2- Early Start 3- Late Finish 4- Dependent Activities- Critical Resource Requirement- Total Resource Demand 7- Late Start 8- Remaining Duration 9- MaximumResource Demand .

4. CONCLUSION AND SUMMARY OF RESULTS

The newly developed heuristic approach (DPSM) was applied to one hundred resource-constrained project scheduling cases through a number of experiments and many encouraging results were obtained.
Though this research did not incorporate the use of $(DP)^2SM$ in the experiments conducted, it goes without saying that by optimising DPSM schedules the best possible combination of priority rules will be implemented , thus the schedule is bound to improve at least remain the same.

The diversity of heuristic-based resource constrained project scheduling approaches and large number of priority rules used in the literature, also the significant difference between duration and characteristics of the schedules produced led the way to evolution of an approach named Dynamic Priority Scheduling Method: DPSM & $(DP)^2SM$. They not only incorporate all well performing priority rules but also there is a higher chance of producing better schedules than any other priority rule.
Using DPSM one may be more certain that the schedule produced for a project is better in terms of time and other characteristics. The approaches available for selecting an appropriate policy for scheduling a project are neither practical nor reliable. DPSM is based on the notion that a project can be partitioned into a number of clock cycles and there is no obligation to use a single policy all the way through all the cycles over project duration.
DPSM can produce good schedules without the need for checking any other priority policy. If one has the time and resources to spare on a search approach for a better schedule (shorter duration, better indices) , it is recommended to use DPSM with a variety of clock cycles instead of going for traditional priority policies. The schedules produced by DPSM are generally better, at least as good as the best which can be produced by a variety of scheduling policies. In spite of being simple, attractive and to some extent promising, $(DP)^2SM$ needs more elaboration, consideration and analysis before any final conclusion can be reached. An important point to mention is that with increasing speed, capacity and capability of micro computers, there is no doubt that computer time and memory is not going to be a barrier to applying DPSM and $(DP)^2SM$ to real life projects.

REFERENCES

Crouston ,Wallace B. S. Decision CPM: Network Reduction and Solution. Operational Research Quarterly, Vol. 4, pp 26-36 (1970).
Dar-el E.M. & Tur Yigal . A Multi-Resource Project Scheduling Algorithm. AIIE Transactions Vol.9, No.1, Mar., Pages 44-52 (1977).
Davis E.W. Project Scheduling Under Resource Constraints-Historical Review and Categorisation of Procedures. AIIE Dec. pp 297-311 (1973).
Hollowy, Nelson & Surphongschai. Comparison of a Multi-Pass Heuristic Decomposition Procedures With Other Resource-Constraint Project Scheduling procedures. Management Science Vol.25, No.9, Sep. , Pages 862-872 (1979).
Kelly ,J.E. Critical Path Planning and Scheduling. Eastern Joint Computer Conference, Boston, Dec. (1959).
Pascoe,T.L. An Experimental Comparison of Heuristic Methods for Allocating Resources. Ph.D. Thesis, Eng. Dept. Cambridge University (1965).
Talbot F.Brian. Resource-Constrained Project Scheduling with Time- Resource Trade Offs: The Nonpreemptive Case. Management Science Vol.28 ,No.10 ,Oct., Pages1197-1210 (1982).
Wiest JD and Levy FK. A Management Guide to PERT/CPM. 2nd edn, Prentice Hall Englewood Cliffs, NJ (1977).
Willis R. J. Critical Path Analysis and Resource Constrained Project Scheduling- Theory and Practice. EJOR 21 , pp 149-155 (1985).
Haralakis G. Objective Evaluation of Nonstorable Resource-Scheduling Profiles. PM, Nov.,Vol 3 No.4 pp 199-207 (1985).

FURTHER READINGS

Battersby,A. Network Analysis for Planning and Scheduling, Macmillan. New York (1978).
Davis Edward W. Project Network Summary Measures Consrained-Resource Scheduling. AIIE Transactions Vol. 7, No.2 June , Pages 132-142 (1975).
Elmaghraby S. E. Activity Network :Project Planning and Control by Network Models. John Wiley, Chichester , Pages 168-226 (1977).
Grool M.C. et al. Project Management in Progress Tools and Strategies for the 90s . Elsevier. (1986).
Harrison F L. Advanced Project Management. Gower, London , Pages 192-202 (1981).
IBM Corporation. Project Analysis and Control System (PROJACS) Program Reference Manual. IBM (1988).
Kurtulus,I.S. & Narula, S.C.). Multi-Poject Scheduling: Analysis of Project Performance. IIE Transactions Vol.17,No.1 ,March (1985).