MPS – Decision Support System for Multiobjective Project Scheduling

Roman Słowiński, Bogdan Soniewicki,
Jan Węglarz

CP-91-007
June 1991

IIASA International Institute for Applied Systems Analysis □ A-2361 Laxenburg □ Austria
Telephone: +43 2236 715210 □ Telex: 079 137 iiasa a □ Telefax: +43 2236 71313

Foreword

This Collaborative Paper is one of a series which presents the different software packages designed and implemented for interactive decision support. These packages constitute the outcome of the contracted study agreement between the System and Decision Sciences Program at IIASA and several Polish scientific institutions. The theoretical part of these results is presented in the IIASA Collaborative Paper CP-90-008 entitled *Contributions to Methodology and Techniques of Decision Analysis (First Stage)*, edited by Andrzej Ruszczyński, Tadeusz Rogowski and Andrzej P. Wierzbicki.

The distributable versions of the software are usually tailored for the illustration of methodology and possible applications. However, for most of these software packages there exists a version made for a specific application and it is possible to modify each software package for a specific real-life application (if the corresponding mathematical programming model is of the type for which a particular package has been designed).

All software developed within the scientific cooperation mentioned above is available either at distribution cost or free of charge for scientific non-commercial usage by institutions and individuals from the countries which are members of IIASA. Inquiries about more detailed information and requests for the software should be addressed to the Leader of the MDA Project.

This volume contains the theoretical and methodological backgrounds as well as the User's Guide for multiobjective project scheduling under multiple-category resource constraints. It handles quite a general class of nonpreemptive scheduling problems with renewable, nonrenewable and doubly-constrained resources, multiple performing modes of activities, precedence constraints in the form of an activity network and multiple project performance criteria of time and cost type.

Alexander B. Kurzhanski
Chairman
System and Decision Sciences Program

Abstract

The report presents a decision support system (DSS) for multiobjective project scheduling under multiple-category resource constraints. It handles quite a general class of non-preemptive scheduling problems with renewable, nonrenewable and doubly-constrained resources, multiple performing modes of activities, precedence constraints in the form of an activity network and multiple project performance criteria of time and cost type. The DSS has been implemented on a microcomputer compatible with IBM PC, and called MPS. It is based on three kinds of heuristics: parallel priority rules, simulated annealing and branch-and-bound. The last algorithm can even yield exact solutions when sufficient processing time is available. Some parts of the MPS are interactive, in particular, the search for a best compromise schedule. Graphical facilities enable a thorough evaluation of feasible schedules. The report starts with a methodological guide presenting the problem formulation and the three heuristics. Then, the general scheme of the MPS is given together with an executive guide. An expanding menu and all its options are described and illustrated with a simple example. The last part presents a real problem solving consisting in scheduling 40 farm activities.

Contents

MPS – Decision Support System for Multiobjective Project Scheduling

*Roman Słowiński**, *Bogdan Soniewicki***
*Jan Węglarz**

1 INTRODUCTION

Simply stated, the problem adressed is how to schedule precedence and resource-constrained activities of a project in order to accomplish a given managerial objective. Over the past twenty five years, a number of techniques have been developed to help project managers answer this question, the applicability of each technique being a function of project characteristics and the managerial objective (see, for example, [1, 2, 3, 4]).

In [5], a scheduling technique was proposed that is capable of heuristically or optimally solving most of the nonpreemptive forms of project scheduling problems previously examined in the literature. This includes time-based, time-cost trade-off, time-resource trade-off, and resource constrained projects. In addition, the proposed algorithm permits the scheduling of activities where activity performance can increase as well as decrease with the availability of resources such as cash, and where resource-duration interactions exist. The algorithm is a branch-and-bound procedure of the backtracking variety.

In practice, however, project scheduling problems usually should involve multiple objectives. It follows mainly from joint consideration of resources submitted to two different kinds of availability constraints: (i) on the amount available at every moment of project duration, and (ii) on the total consumption over a given time period. Depending on what constraint is imposed, we have resources of different categories: renewable (e.g. machines, manpower, equipment) if only (i) is imposed, nonrenewable (e.g. money, fuel, raw materials) if only (ii) is imposed, and doubly constrained (e.g. power, rate of investment, fuel flow) if both (i) and (ii) are imposed. The utilization of resources submitted to constraint (i) is usually concerned with time criteria (e.g. minimization of machine idle time is equivalent to minimization of project duration) while the consumption of resources submitted to constraint (ii) is measured by cost criteria. Thus, joint consideration of multiple-category resources involves time and cost criteria which are in general conflicting.

The above facts have motivated a construction of a DSS for multiobjective project scheduling, called MPS. Its general scheme was introduced in [6, 7, 13]. In the next section, we present a methodological guide of MPS, with a general problem formulation, the structure of the system and the algorithms used in the calculation phase. Then, in section 3, an executive guide of MPS is given with a detailed description of all the options from an expanding menu. The final section presents the steps of solving a real scheduling problem consisting of 40 agricultural activities.

*Institute of Computing Science, Technical University of Poznań, 60-965 Poznań, Poland.
**Center for Education and Development in Agriculture, 61-659 Poznań, Poland.

2 METHODOLOGICAL GUIDE

2.1 Problem formulation

It is assumed that all time characteristics of the project are integer and that the time horizon T_h is divided into periods of unit length, $t = 1, ..., T_h$.

The project is characterized by four components: set of resources R, set of activities A, precedence constraints in set A, and set Q of project performance measures (objectives, criteria).

Set R is composed of:

- p types of renewable resources $R_1^r, ..., R_p^r$ with the usage limited to N_{kt}^r units in every period $t (k = 1, ..., p; t = 1, ..., T_h)$,
- v types of nonrenewable resources $R_1^n, ..., R_v^n$ with the consumption limited to B_k^n units $(k = 1, ..., v)$,
- u types of doubly constrained resources $R_1^d, ..., R_u^d$ with consumption limited to B_k^d units and usage to N_{kt}^d units $(k = 1, ..., u; t = 1, ..., T_h)$.

Set A is composed of n activities which have discrete resource requirements. For each activity $A_j \in A$ we have in general w_j performing modes, i.e. feasible assignments of resource amounts to this activity. In order to simplify problem formulation we will admit that $w_j = w$ $(j = 1, ..., n)$. Performing mode m of activity A_j is defined by the vector $r_{mj} = [r_{mj1}^r ... r_{mjp}^r \;\; r_{mj1}^n ... r_{mjv}^n \;\; r_{mj1}^d ... r_{mju}^d]$ whose elements determine the usage of renewable and doubly constrained resources and consumption per period (constant for every activity) of nonrenewable resources $(m = 1, ..., w; j = 1, ..., n)$. For each mode m the duration of A_j is known: D_{mj} $(m = 1, ..., w; j = 1, ..., n)$. Ready time a_j and due date d_j are also specified for each A_j $(j = 1, ..., n)$.

The precedence constraints are represented by a directed acyclic graph using activity–on–arc convention. A unique ending dummy activity A_{n+1} with zero duration and resource requirement is appended to the graph.

The set Q of project performance measures is composed of following time and cost criteria:

- project completion time: CT,
- smoothness of the resource profile: S_k^r $(k = 1, ..., p)$ and S_k^d $(k = 1, ..., u)$,
- mean weighted lateness: MWL,
- total number of tardy activities: TNTA,
- mean weighted flow time: MWFT,
- total resource consumption: TC_k^n $(k = 1, ..., v)$ and TC_k^d $(k = 1, ..., u)$,
- weighted resource consumption: WC,
- net present value: NPV.

Scheduling of the project consists in such an allocation (considered in time) of resources from set R to activities from set A that all activities are completed, the constraints are satisfied and the best compromise between criteria from set Q is reached.

The above model of multicriteria project scheduling has been considered first for the preemptive case [8, 9]. The nonpreemptive case has been investigated in [10, 11] and [12, 13] where an interactive procedure for multiobjective project scheduling has been proposed. It is using the implicit enumeration algorithm by Patterson at al. [5] in the calculation phase. The mathematical programming formulation of this problem has been given in [13]. It is based on the following definition of the zero-one decision variable:

$$x_{jmt} = \begin{cases} 1 & \textit{if activity } A_j \textit{ performed under mode } m \\ & \textit{is completed in period } t, \\ 0 & \textit{otherwise} \end{cases}$$

$(j = 1, ..., n; m = 1, ..., w; t = 1, ..., T_h)$. Associated with each activity A_j are its critical-path determined early finish completion period e_j and late finish completion period l_j. Both e_j and l_j are calculated in the usual way but using the set of minimum duration modes for all activities. When determining late finish completion periods, l_{n+1} is set equal to T_h. Of course, $x_{jmt} \equiv 0$ for $t < e_j$ and $t > l_j$. Let P_j be the set of all immediate predecessors of activity A_j $(j = 1, ..., n)$. Now, the multiobjective project scheduling problem can be formulated as the following zero-one MOLP problem:

$$\min CT = \sum_{m=1}^{w} \sum_{t=e_{n+1}}^{l_{n+1}} t x_{(n+1)mt} \tag{1}$$

$$\min S_k^r = \max_{1 \le t \le T_h} \left\{ \sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{q=t}^{t+D_{mj}-1} r_{mjk}^r x_{jmq} \right\} \quad k = 1, \ldots, p \tag{2}$$

$$\min S_k^d = \max_{1 \le t \le T_h} \left\{ \sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{q=t}^{t+D_{mj}-1} r_{mjk}^d x_{jmq} \right\} \quad k = 1, \ldots, p \tag{3}$$

$$\min MWL = (1/n) \sum_{j=1}^{n} v_j \sum_{m=1}^{w} \sum_{t=d_j+1}^{l_j} (t - d_j) x_{jmt} \tag{4}$$

$$\min TNTA = \sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{t=d_j+1}^{l_j} x_{jmt} \tag{5}$$

$$\min MWFT = (1/n) \sum_{j=1}^{n} v_j \sum_{m=1}^{w} \sum_{t=e_j}^{l_j} (t x_{jmt} - a_j) \tag{6}$$

$$\min TC_k^n = \sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{t=e_j}^{l_j} r_{mjk}^n \, x_{jmt} \quad k = 1, \ldots, v \tag{7}$$

$$\min TC_k^d = \sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{t=e_j}^{l_j} r_{mjk}^d \, D_{jm} \, x_{jmt} \quad k = 1, \ldots, u \tag{8}$$

$$\min WC = \sum_{k=1}^{v} c_k^n \, TC_k^n + \sum_{k=1}^{u} c_k^d \, TC_k^d \tag{9}$$

where c_k^n and c_k^d are unitary costs of R_k^n and R_k^d, respectively,

$$\max NPV = \sum_{t=1}^{T_h} \alpha_t \left\{ \sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{q=t}^{t+D_{mj}-1} x_{jmq} r_{mjk}^n / D_{mj} - \sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{q=t-1}^{t+D_{mj}-2} x_{jmq} r_{mjk}^n / D_{mj} \right\} \qquad (10)$$

where R_k^n is cash, $\alpha_t = (1-\delta)^{1-t}$ is a discount factor and δ is a discount rate, *subject to the constraints:*

- on the performance of each activity using one mode only

$$\sum_{m=1}^{w} \sum_{t=e_j}^{l_j} x_{jmt} = 1 \quad j = 1, \ldots, n \qquad (11)$$

- on the precedence

$$\sum_{m=1}^{w} \sum_{t=e_j}^{l_j} (t - D_{mj}) x_{jmt} - \sum_{m=1}^{w} \sum_{t=e_f}^{l_f} t x_{fmt} \geq 0 \quad j = 1, \ldots, n+1, \quad \bigvee f \qquad (12)$$

- on resource availability
  - renewable

$$\sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{q=t}^{t+D_{mj}-1} r_{mjk}^r x_{jmq} \leq N_{kt}^r \quad k = 1 \ldots, p; \quad t = 1, \ldots, T_h \qquad (13)$$

  - nonrenewable

$$\sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{t=e_j}^{l_j} r_{mjk}^n x_{jmt} \leq B_k^n \quad k = 1, \ldots, v \qquad (14)$$

  - doubly constrained

$$\sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{q=t}^{t+D_{mj}-1} r_{mjk}^d x_{jmq} \leq N_{kt}^d \quad k = 1 \ldots, u; \quad t = 1, \ldots, T_h \qquad (15)$$

$$\sum_{j=1}^{n} \sum_{m=1}^{w} \sum_{t=e_j}^{l_j} r_{mjk}^d D_{mj} x_{jmt} \leq B_k^d \quad k = 1, \ldots, u \qquad (16)$$

Even for a single criterion, the above problems are NP-hard and need implicit enumeration algorithms which are rather not efficient. The interactive procedure [13] proposed for solving the multiobjective problem makes an aggregation of objectives in a scalarizing function of the augmented weighted Chebyshev form. Generation of compromise solutions in successive iterations needs solving as many complex optimization problems. If one would like to solve them optimally, the waiting time for a next proposal would be very long and perhaps unacceptable for the decision maker (DM). For this reason, when constructing the DSS for the most general case of project scheduling, we decided to use heuristic algorithms.

2.2 Structure of the MPS

The MPS is composed of four modules:

- model editor,
- heuristic algorithms for single-objective project scheduling,
- interactive procedure for multiobjective project scheduling,
- display of results.

Model editor permits to enter and modify the data required to set up the model of a particular project scheduling problem. Its functional description will be made in the next section.

Three kinds of heuristic algorithms for single-objective project scheduling are used:

- parallel priority heuristics,
- simulated annealing,
- branch-and-bound type heuristics.

2.3 Priority heuristics

The idea of the parallel priority heuristics can be summarized in the following steps (cf.[14]):

Step 1. Using a priority function create a list of activities. Set the current time equal to zero.

Step 2. Choose the first activity from the list which is ready to be performed, i.e. such that all its predecessors are completed at the current time. If the amounts of available resources are sufficient to cover the requirements of the first performing mode [1] of the chosen activity, then allot the required amount to this activity. Otherwise try next performing modes and if it also fails, increase the current time by one unit and retry the operation. Remove the scheduled activity from the list.

Step 3. Augment the current time to the earliest period from among the completion period of the scheduled activities without successors in the partial schedule.

Step 4. If the list is empty then stop, otherwise go to step 2.

In MPS, twelve priority functions take into account time or/and resource characteristics of the project:

1. $p_j = 1/LFT_{1j}$, where LFT_{1j} is the latest completion time of activity A_j, following from the critical path analysis for the first performing modes of activities.
2. $p_j = 1/LST_{1j}$, where LST_{1j} is the latest starting time of A_j.
3. $p_j = 1/EST_{1j}$, where EST_{1j} is the earliest starting time of A_j.

[1]The activity performing modes are ordered according to increasing duration when time criteria are considered, and according to increasing cost when cost criteria are considered.

4. $p_j = 1/(EST_{1j} + d_j)$, where d_j is the due date of A_j .

5. $p_j = 1/d_j$.

6. $p_j = 1/(a_j + d_j)$, where a_j is the ready time of A_j .

7. $p_j = 1/a_j$.

8. $p_j = D_{1j}$, where D_{1j} is the duration of A_j performed under mode 1.

9. $p_j = D_{1j} + \sum_{i \in S_j} D_{1i}$, where S_j is the set of successors of A_j .

10. $p_j = r^r_{1j1}$, where r^r_{1j1} is the requirement for renewable resource R^r_1 of A_j performed under mode 1.

11. $p_j = \sum_{k=1}^{v} r^n_{1jk}$, where r^n_{1jk} is the requirement for nonrenewable resource R^n_k of A_j performed under mode 1 (k=1,...,v).

12. $p_j = R_j + \sum_{i \in S_j} R_i$,
where activities start in their EST_{1j}, $R_j = \sum_{k=1}^{p}(r^r_{1jk}/N^r_k)(V_k/V_{max})D_{1j}$, $V_k = U_k + E_k N^r_k$, $V_{max} = \max_k(V)$, U_k is the latest time period in which there is a non-zero requirement for resource R^r_k, E_k is the total excess requirement for resource R^r_k over the availability N^r_k.

Moreover, in order to obtain a greater variety of feasible schedules, for each priority heuristic 5 mutations are generated for a priority list corresponding to a particular priority function. The mutations are generated in such a way that the next activity to be scheduled is taken either from the top of the original list, or is randomly chosen from first 2, or 3, or 4, or 5 activities. In this way, we can obtain up to 60 different feasible schedules evaluated from the viewpoint of the chosen criteria.

Additionally, we use a heuristic procedure specialized in smoothing resource profiles. It is based on the algorithm proposed by Harris [15] for the case of a single renewable resource and one performing mode per activity. For a given project completion time, it tends to minimize a "turning moment" of the resource profile. It works in the following steps:

Step 1. Assume mode 1 for each activity and calculate their EST_{1j} (j=1,...,n) ignoring all resource constraints.

Step 2. Create a subset of activities without predecessors in the current set.

Step 3. For each activity from the subset, calculate a coefficient of improvement after delaying it by all possible (integer) lapses of time under all possible performing modes. Select an activity with the best positive coefficient, update its starting time and performing mode, and remove it from the subset and the current set. If there is no activity with a positive coefficient then remove all the activities.

Step 4. Repeat steps 2 and 3 until the current set becomes empty.

Step 5. If no improvement was made in the last iteration then stop, otherwise return to step 2 for pushing activities in an opposite direction (predecessors have to be replaced by successors and vice versa).

2.4 Simulated annealing

The simulated annealing procedure starts the search from the best solution obtained using parallel priority heuristics. Instead of a simple local search algorithm, simulated annealing attempts to avoid becoming trapped in a local optimum by sometimes accepting a neighbourhood move which increases the value of the objective z to be minimized (cf.[16, 17]. The acceptance or rejection of an uphill move is determined by a sequence of random numbers, but with a controlled probability. The probability of accepting a move which causes an increase d of z is called the acceptance function and is normally set to $exp(-\delta/T)$ where T is a control parameter which corresponds to temperature in the analogy with physical annealing. This acceptance function implies that small increases in z are more likely to be accepted than large increases, and that when T is high most moves will be accepted, but as T approaches zero most uphill moves will be rejected. So in simulated annealing, the algorithm is started with a relatively high value of T, to avoid being prematurely trapped in a local optimum. The algorithm proceeds by attemping a certain number of neighbourhood moves at each temperature, while the temperature parameter is gradually dropped.

In order to apply the simulated annealing one has to define the neighbourhood of any solution and an efficient method of moving from one solution to its neighbourhood solution with a simultaneous calculation of a new value of the objective. In the general case of project scheduling we are dealing with, we propose to define the neighbourhood in the following way. Two activities from among those which are not precedence-related are randomly chosen. Then, those two activities are permuted on the priority list corresponding to the current solution. The new list is used by the parallel heuristic to construct the neighbourhood solution.

The simulated annealing for project scheduling is organized in the following steps:

Step 1. Take a starting solution, i.e. the best solution obtained using parallel priority heuristics.

Step 2. Set the value of the control parameter (temperature):

$$T = \Delta/ln(x_0^{-1})$$

where Δ is an acceptable deterioration of the score on considered objective z in relation to the starting solution (e.g.20%), x_0 is an acceptance coefficient close to one (e.g.0.9).

Step 3. Construct a neighbourhood solution to the starting one.

Step 4. If the value of z has improved, then the probability of accepting the neighbourhood solution as a new starting solution is equal to one; otherwise calculate the probability as $P = exp(-\delta/T)$, where δ is an actual deterioration of the score on z.

Step 5. Generate a random number from the interval [0,1] according to the uniform distribution. If this number is less than or equal to P then accept the solution constructed in step 3 as a new starting point; otherwise don't change the starting solution.

Step 6. Repeat steps 3, 4 and 5 until the number of new starting solutions attains a given constant which corresponds to an approximate equilibrium.

Step 7. Set the control parameter $T = 0.9 \times T$. Repeat steps 3 to 6 until there is no improvement of z for three consecutive values of T. Then STOP.

2.5 Branch and bound algorithm

The branch and bound algorithm uses the strategy of the depth-first-search and starts backtracking from the best solution obtained using parallel priority heuristics. The search tree consists of all precedence-feasible permutations of activities combined with their performing modes. The details of the algorithm for time and cost criteria are presented in [5]. The results of a computational experiment with the algorithm are presented in [18]. The search process can be arbitrarily interrupted (then resumed if desired) and a current best schedule can be displayed.

2.6 Multicriteria scheduling

In the case of multicriteria scheduling, the interactive search for the best compromise solution is organized on the set of nondominated schedules obtained using parallel priority heuristics or simulated annealing, possibly improved using the branch and bound algorithm.

The interactive procedure is organized in three steps repeated iteratively (cf.[7, 13]):

Step 1. (Starting step). Construct the $k \times k$ pay-off table Z on the set of nondominated schedules. Elements z_{ij} of Z are values of criterion i for the best schedule from the viewpoint of criterion j. The diagonal of Z defines an "ideal" solution which is unfeasible in general. The vector composed of the worst scores z_{ij} on particular criteria is a "nadir" solution.

Step 2. (Calculation step). Find the nondominated schedule being the closest to the "ideal" one in the sense of the scalarizing function (augmented weighted Chebyshev norm).

Step 3. (Decision step). Present the schedule found in step 2 to the decision maker (DM). If he finds it satisfactory on all the criteria then STOP; otherwise ask the DM to specify a satisfactory criterion to be relaxed and amount of the relaxation in order to gain on other criteria. The relaxation is translated into a penalty function appended to the scalarizing function and the algorithm returns to step 2.

The scalarizing function used in the calculation step has the following form:

$$s(\underline{z}, \underline{\pi}) = \max_{1 \le i \le k}(\pi_i(z_i - u_i)) + \epsilon \sum_{i=1}^{k} \pi_i(z_i - u_i)$$

where

$$\pi_i = \Phi / \sum_{i=1}^{k} \Phi_i, \quad \Phi_i = (U_i - u_i)/U_i, \quad i = 1, \ldots, k,$$

$$\epsilon = 0.001, \quad u_i = \min_{\underline{z} \in N}(z_i), \quad U_i = \max_{\underline{z} \in N}(z_i), \quad i = 1, \ldots, k,$$

N is the set of nondominated schedules.

Relaxation Δz_i determined in the decision step is translated to the penalty function appended to the scalarizing function:

$S(\underline{z}) = s(\underline{z}, \underline{\pi}) + b_j(\Delta \underline{z}, \underline{\pi})$,

$b(\Delta \underline{z}, \underline{\pi}) = 1/2 \sum_{i=1}^{k} (\max(0, \pi_i(z_i - z_i^{j-1} - \Delta z_i)))^2$

where j is the iteration index and $\underline{z}^{j-1} \in N$ is the schedule found in previous iteration. Morever, the coefficient π_i of the relaxed criterion is reduced to zero in iteration j.

In MPS, the DM has yet two other possibilities of scanning the set of nondominated schedules when looking for the best compromise one. The first consists in moving from one schedule to the next one from the list of nondominated schedules (option: "another solution"). The second possibility consists in moving from one schedule to the next improved schedule from the viewpoint of an indicated criterion (option: "next improved solution").

3 USER'S MANUAL

3.1 Installing the MPS

MPS runs on a PC compatible with IBM having the following minimal configuration :

- RAM 640 kb,
- MS-DOS ver. 3.10 or higher,
- one floppy disk drive or, preferably, a hard disk.

The MPS diskette or the MPS directory on the hard disk must contain the following files :

- MPS.EXE a compiled system
- 4x6.fon fonts for CGA card
- 8x8.fon fonts for EGA or VGA card
- 14x9.fon fonts for HERCULES card

MPS has been written in TURBO-PASCAL 5.0 with Borland Graphics Toolbox 4.0. Program MPS must be compiled for a graphic card available on a user's computer.

3.2 Menu of the system

MPS starts when you write on the computer monitor its name and press the key ENTER. Then, the invitation screen appears for few seconds (cf. Fig. 2 in the Appendix). At the top of the menu you can choose one of three options :

- DATA HANDLING,
- HEURISTICS,
- EXIT.

To choose an option use vertical arrows which move a highlighted field. In order to accept a highlighted option press key ENTER. Then you go to the next step of the menu.

After choosing DATA HANDLING option, three next options appear:

- NEW PROJECT,
- MODIFICATIONS,

- EXIT.

Fig. 3. shows a copy of the screen after choosing the DATA HANDLING option. Option HEURISTICS offers the following possibilities in the next step of the menu :

- SINGLE OBJECTIVE (optimisation),
- MULTIOBJECTIVE (optimisation),
- EXIT.

Next step of the menu gives the choice of the following approximation algorithms to be used in calculations :

- PRIORITY PROCEDURES,
- SIMULATED ANNEALING,
- BAB TYPE PROCEDURE,
- EXIT.

The last step of this expanding menu permits to choose one criterion in the case of single objective optimisation or two to seven criteria in the case of multiobjective optimisation. The seven criteria are the following :

- COMPLETION TIME,
- RESOURCE SMOOTHING,
- MEAN WEIGHTED ACTIVITY LATENESS,
- TOTAL NUMBER OF TARDY ACTIVITIES,
- MEAN WEIGHTED FLOW TIME,
- NET PRESENT VALUE,
- WEIGHTED RESOURCRE CONSUMPTION,
- EXIT.

Fig. 4. presents the full MPS menu with chosen options being highlighted. To accept his choice, the user must press the key ENTER. The system then goes automatically to the calculation phase.

3.3 Data operating

3.3.1 The new project building

In order to introduce a new project scheduling problem, the user must choose the option DATA HANDLING and next NEW PROJECT.

First, MPS reads precedence constraints among activities, i.e. relation of partial ordering in the set of project activities. It is represented in a form of lists of immediate predecessors of particular activities. The following conditions have to be fulfilled :

1. Project cannot have more than 100 activities.

2. The name of the activity must be unique. If not, MPS will write a message.

3. The name of a predecessor of the activity must be previously defined as a name of the activity. Activities without predecessors must be defined first.

4. Because of an "activity on arc" convention for a graphical representation of the precedence constraints, activities going out from a vertex must have the same list of predecessors. If the user wants a dummy activity, he must add it himself.

The name of the activity must be written in the highlighted field preceded by the text "Name of the activity". This name has not be longer then 20 letters. Then, one can press a vertical arrow key and move to the bottom where the list of predecessors is composed. In order to pass to the next activity one must press key F2, and in order to go back to the previous activity one must press F3. Fig. 5. displays an example of the screen in the stage of reading precedence constraints for a project. When the last activity has been entered, press F4 to go to read global data of the project. The global data are composed of :

1. Due date for the project (in time units).

2. How many types of renewable resources are in the project (maximally 4).

3. Name of a renewable resource. The length of this name should have no more then 10 characters. The user can use only small letters.

4. Number of available units of ... (here name of a renewable resource).

5. How many types of nonrenewable resources are in the project (maximally 3).

6. Name of a nonrenewable resource. The length of this name should have no more then 10 characters (small letters only).

7. Number of available units of ... (here name of a nonrenewable resource).

8. Inflow(+) or outflow(-) of ... (name of nonrenewable resource) at the end of the project.

9. Interest rate (in percents). This rate refers in principle to money as a nonrenewable resource.

It should be specified that MPS treats a doublyconstrained resource as a simultaneously renewable and nonrenewable. For example, if money would be a doubly constrained resource, then from the viewpoint of its total consumption it is considered as a nonrenewable resource, while from the viewpoint of the intensity of its consumption in particular time periods, it is considered as a renewable resource.

The user must insert the data in the highlighted fields which can be moved by pressing the vertical arrow keys. After pressing ENTER, MPS goes to the next phase - phase of reading local data for each particular project activity. There are then three further possibilities :

1. After pressing keys Shift and E, option DATA HANDLING is terminated and data are saved in a file.

2. After pressing keys Shift and Q, option DATA HANDLIND is terminated without changing data in a file.
3. After pressing keys Shift and N the program goes to the activity with a specified number.

In Fig. 6, one can see a copy of the screen in the phase of reading global data.

In the phase of reading local data, specific for each activity, MPS reads the following parameters :

1. Release time.
2. Due date.
3. Number of the parralell activity before which the currently displayed activity cannot start - (Name of the current activity) cannot start before activity ...
4. Weighting factor.
5. Number of the mode (maximally 3).
6. Activity duration for mode (number).
7. Requirement of renewable resource (name) per unit duration, for mode (number).
8. Requirement of nonrenewable resource (name) per unit duration, for mode (number).
9. Inflow(+) or outflow(-) of (name of a nonrenewable resource) per unit duration, for mode (number).

After entering data for a particular activity, the user must press ENTER to go to the next activity. In order to go to the activity with another number, the user must press simultaneously Shift and N , and then give its number. The stage DATA HANDLING terminates after pressing Shift and E (the data file is saved) or pressing Shift and Q (the data file is not saved).

If the user arrives to the last activity and accepts the inserted data by pressing the key ENTER, all data are saved automatically in a file.

3.3.2 Modification of data

MODIFICATIONS of data are similiar to data entering for a new project. At the beginning, the user must choose the name of the file with the data to be modified. When going through three phases described in 3.3.1., the user can modify all the parameters of the project.

In the first phase he can :

- add an activity at the end of the list of activities,
- delete the activity by inserting spaces instead of its name,
- delete one of predecessors of the activity by inserting spaces instead of its name.
- add a predecessor to an activity.

In the two next phases, the user can modify project and activity parameters. It can be done in the same manner as in the case of building a new project. However, all modifications can be saved only after pressing Shift and E.

3.4 Calculation

The project scheduling problem characterized in an activated data file can be solved using one of the three algorithms described in section 2 :

- priority heuristics,
- simulated annealing,
- branch and bound (BAB).

The same selection of algorithms is offered for single and multiobjective optimization. The priority heuristics give up to 60+p feasible schedules (p is the number of renewable resource types). In the single objective case, the best schedule from the viewpoint of a chosen criterion is adopted, while in the multiobjective case, a subset of nondominated schedules is identified and an interactive search over this subset is organized by the system.

Simulated annealing is activated for each selected criterion (cf. Fig. 8). In the multiobjective case, each generated schedule is tested for dominance and it is saved when nondominated (cf. Fig. 12). This algorithm is especially worthwhile when the user is interested in getting a large variety of nondominated schedules for the interactive phase.

Branch and bound algorithm (BAB) improves nondominated schedules obtained using priority heuristics. At its output, one gets either the best improved solution from the viewpoint of a single criterion or a set of nondominated schedules from the viewpoint of multiple criteria. The user can break every procedure by pressing any key during its run. In the case of BAB, MPS displays after the break the current best schedule and asks the user about the continuation of the calculation. After pressing key "Y", MPS continues calculation with BAB; key "N" terminates the procedure.

In the multiobjective case, the number of nondominated schedules generated and saved in the calculation phase is displayed (NNDS=...) in the upper right corner of the screen (cf. Figs. 12-15).

3.5 Display of results

The feasible schedules submitted for evaluation of the user are displayed in a graphic and table form. An axample of such a display is shown in Figs. 10, 16. Gantt charts and resource profiles are shown in four windows of the screen. A highlighted window is an active one and can be either enlarged on the whole screen (cf. Fig. 9) or used to display another picture from an offered variety, or a table schedule specifying start and end times of activities, their modes and resource requirements in time. The user can change the active window by pressing arrow keys. The screen can be copied on the printer by pressing key H (HARDCOPY). When pressing key O, the user gets a new menu offering the possibility of changing the contents of the active window. Key Q is used to abort the current stage and go to the main menu.

In the multiobjective case, there are three possibilities of scanning the set of nondominated schedules (cf. p. 2.6) :

1. To see the next nondominated schedule from the generated set (key A).
2. To see a nondominated schedule with the minimal improvement on the criterion indicated by the cursor (key N).
3. To relax one criterion indicated by the cursor and search for an improved schedule. A value of the relaxation is showed in a lower line.

Table 1. List of activities of the agricultural project

1.	harrowing wheat	21.	tedding grassland
2.	harrowing rape	22.	raking grassland
3.	sowing rape	23.	croopping grassland
4.	croopping harv lupin	24.	desication potato
5.	drying lupine	25.	croopping sugar beet
6.	croopping of strorer	26.	fertilisation lupine
7.	cutting down lucerne	27.	croopping fodder bee
8.	cutting down grass	28.	fertilization wheat
9.	tedding grassland	29.	ploughing
10.	raking grassland	30.	skimming lupine
11.	croopping grassland	31.	harrowing lupine
12.	desication potato	32.	harrowing wheat
13.	harrowing wheat	33.	sowing wheat
14.	harrowing rape	34.	croopping sugar beet
15.	sowing rape	35.	croopping potato
16.	croopping harv lupin	36.	harrowing potato
17.	drying lupine	37.	ploughing
18.	croopping of strorer	38.	sowing corn
19.	cutting down lucerne	39.	fertilization fodder
20.	cutting down grass	40.	ploughing corn

An error of filling the cursor field can be erased after pressing key D. Position of the cursor can be moved using the keys of horizontal arrows. The user can view the current schedule after pressing key V. In Figs. 13, 14 and 15, examples of displays in the interactive phase of the multicriteria optimisation are shown.

3.6 Example of scheduling

In this section, we shall illustrate the functioning of MPS on an example of scheduling 40 farm operations subject to precedence and resource constraints (cf.[6]). The list of operations is given in Table 1.

There are two types of renewable resources: manpower and tractors, available in 200 and 150 units, respectively. The only nonrenewable resource, money, is available in 20000 units at the beginning of the project. For majority of project activities there are specified three performing modes differing by resource requirements and duration. Ready times and due dates of activities follow from an agricultural calendar. Those data are presented in Table 2. A graphical representation of precedence constraints in the set of activities is given in Fig.1.

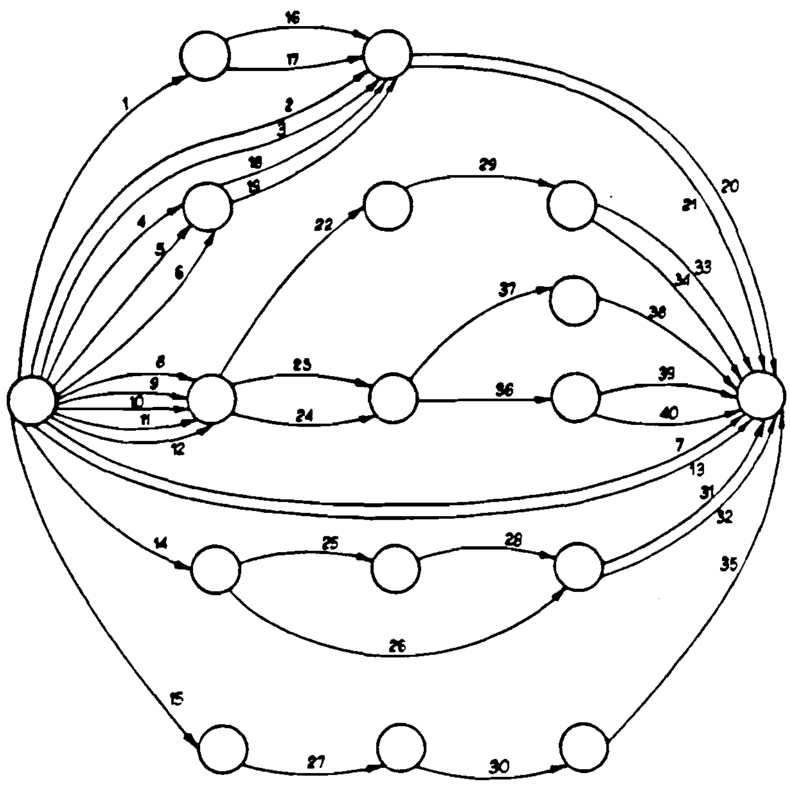


Fig 1. Precedence constraints in the set of agricultural activities

In Table 2, the following notation has been used: d_{i1}, d_{i2}, d_{i3} - duration of activity i for the first, the second and the third performing mode, respectively, α_i - ready time of activity i, δ_i - due date of activity i, R^r_{ijk} - requirement of activity i for the j-th renewable resource under k-th performing mode, R^n_{ijk} - requirement of activity i for the j-th nonrenewable resource under k-th performing mode.

Screen copies shown in the Appendix summarize the process of single and multiobjective calculations for the agricultural project.

Table 2. Characteristics of agricultural activities

Lp	d_{i1}	d_{i2}	d_{i3}	α_i	δ_i	R^r_{i11} / R^r_{i21}	R^r_{i12} / R^r_{i22}	R^r_{i13} / R^r_{i23}	R^n_{i11}	R^n_{i12}	R^n_{i13}
1	8	10	16	0	28	4	3	2	80	50	30
						4	3	2			
2	8	10	16	0	28	13	11	7	150	140	110
						13	11	7			
3	9	12	18	0	48	20	15	10	200	170	150
						11	9	6			
4	11	15	22	0	28	25	20	13	230	200	170
						10	8	5			
5	10	13	20	0	28	6	5	3	500	400	350
6	16	25	32	0	28	33	25	17	200	180	150
						12	9	6			
7	8	12	16	0	11	10	16	8	160	130	100
						10	16	8			
8	11	16	22	0	23	10	8	5	160	120	90
						10	8	5			
9	20	30	40	0	23	23	18	12	200	150	110
						23	18	12			
10	10	15	20	0	23	16	12	8	150	120	80
						16	12	8			
11	17	25	34	0	23	26	20	13	80	75	70
						8	6	4			
12	12	18	24	10	22	12	9	6	160	150	140
						12	9	6			
13	20	30	40	0	28	56	42	28	400	370	350
						50	37	25			
14	12	18	24	46	81	14	11	7	220	200	190
						10	8	5			
15	40	60	80	42	93	65	50	33	470	450	400
						9	7	5			
16	6	9	12	8	48	20	15	10	240	220	200
						12	9	6			
17	20	30	40	8	48	12	9	6	220	200	180
						12	9	6			
18	6	9	12	16	48	23	18	12	150	100	90
						23	18	12			
19	4	6	8	16	48	10	8	5	150	100	80
						10	8	5			
20	6	9	12	28	58	4	3	2	80	60	50
						4	3	2			

Table 2. cont.

Lp	d_{i1}	d_{i2}	d_{i3}	α_i	δ_i	R^r_{i11} / R^r_{i21}	R^r_{i12} / R^r_{i22}	R^r_{i13} / R^r_{i23}	R^n_{i11}	R^n_{i12}	R^n_{i13}
21	20	30	40	28	58	12	9	6	250	180	150
						7	5	4			
22	29	44	57	23	80	46	35	23	800	700	600
						42	32	21			
23	40	60	80	22	83	36	27	18	500	400	300
						20	15	10			
24	24	36	48	22	83	106	80	53	700	600	550
						6	4	3			
25	12	18	24	58	98	19	15	10	200	170	150
						19	15	10			
26	12	18	24	58	93	14	11	7	190	160	140
						10	8	5			
27	4	6	8	84	88	6	5	3	120	100	70
						5	4	2			
28	19	29	38	70	117	15	11	8	260	240	200
						15	11	8			
29	13	20	26	80	93	14	11	7	360	350	320
						10	8	5			
30	8	12	16	88	105	20	15	10	240	210	200
						16	12	8			
31	23	34	40	70	117	30	23	25	250	220	200
						17	14	9			
32	24	36	48	70	117	44	33	22	700	600	500
						44	33	22			
33	24	36	48	93	117	48	36	24	580	500	420
						38	29	19			
34	24	36	48	93	117	13	10	7	250	240	230
						13	10	7			
35	10	5	20	96	117	6	5	3	70	60	50
						6	5	3			
36	6	9	12	62	97	19	15	11	130	120	110
						15	12	8			
37	12	18	24	62	95	56	42	28	350	320	300
						6	5	3			
38	13	19	26	74	108	6	5	3	80	70	60
						6	5	3			
39	20	30	40	68	117	58	45	29	400	350	340
						48	36	24			
40	18	27	36	68	117	20	15	10	250	220	200
						20	15	10			

References

[1] E.W. Davis and J.H. Patterson. *A comparison of heuristic andoptimal solutions in resource-constrained project scheduling.* Management Science 21 (8), 1975, pp 944-955.

[2] S.E. Elmaghraby. *Activity networks: project planning and control by network models.* John Wiley & Sons, New York, 1977.

[3] J.H. Patterson. *A comparison of exact approaches for solving the multiple constrained resource, project scheduling problem.* Management Science 30 (7), 1984, pp 854-867.

[4] R. R.Słowiński, J.Węglarz (eds.). *Advances in Project Scheduling.* Elsevier, Amsterdam, 1989.

[5] J.H. Patterson,R.Słowiński, F.B.Talbot, J. Węglarz. *An algorithm for a general class of precedence and resource constrained scheduling problems.* in: R.Słowiński, J. Węglarz (eds.), Advances in Project Scheduling, Elsevier, Amsterdam, 1989, pp 3-28.

[6] R.Słowiński, B. Soniewicki, J. Węglarz. *MPS - komputerowy system wspomagania decyzji dla wielokryterialnego rozdzialu zasobow.* Archiwum Automatyki i Telemechaniki (to appear).

[7] R.Słowiński, B. Soniewicki, J. Węglarz. *System wspomagania decyzji dla wielokryterialnego rozdzialu zasobow, Postepy Cybernetyki.* (to appear).

[8] R.Słowiński. *Multiobjective network scheduling with efficient use of renewable and nonrenewable resources.* Europ. J. Opl. Res., 7 (3), 1981, pp 265-273.

[9] R.Słowiński. *Problemes d'allocation de moyens limites en presence de criteres multiples, Cahier de LAMSADE, no.36, Universitw de Paris-Dauphine.* Paris, 1981.

[10] R.Słowiński. *Modelling and solving multicriteria project scheduling problems.* in A.Straszak (Ed.), Large Scale Systemst-t Theory and Applications, Pergamon Press, Oxford, 1983, pp 469-474.

[11] J. Błażewicz, W. Cellary, R.Słowiński, J. Węglarz. *Scheduling Under Resource Constraints - Deterministic Models.* J.C.Baltzer AG, Annals of Operations Research, vol.7, 1986, Basel.

[12] R.Słowiński, J. Węglarz. *An interactive algorithm for multiobjective precedence and resource constrained scheduling problems.* in: Vriethoff, J.Visser, H.H.Boerma (Eds.), Proc. 8th INTERNET Congress, Elsevier/North-Holland, Amsterdam, 1985, pp 866-873.

[13] R.Słowiński. *Multiobjective project scheduling under multiple-category resource constraints.* in: R.Słowiński, J. Węglarz (eds.), Advances in Project Scheduling, Elsevier, Amsterdam, 1989, pp 151-167.

[14] J.E. Kelley Jr. *The critical-path method: resource planning and scheduling.* in: J.F.Muth and G.L.Thompson (eds.), Industrial Scheduling, Prentice Hall, Engl. Cliffs, New Jersey, 1963.

[15] R.B. Harris. *Precedence and Arrow Networking Techniques for Construction.* Wiley, New York, 1978.

[16] S. Kirkpatrick, C.D. Gellat Jr., M.P. Vecchi. *Optimization by simulated annealing.* Science 220, 1983, pp 671-680.

[17] V. Cerny. *The termodynamical approach to the travelling salesman problem: An efficient simulation algorithm.* J. Opt. Theory Appl. 45, 1985, pp 41-51.

[18] J.H. Patterson, R.Slowiński, F.B. Talbot, J. Węglarz. *Computational experience with a backtracking algorithm for solving a general class of precedence and resource constrained scheduling problems.* Europ. J. Opl. Res. 49, 1990, pp 68-79.

Appendix

Screen copies of the MPS

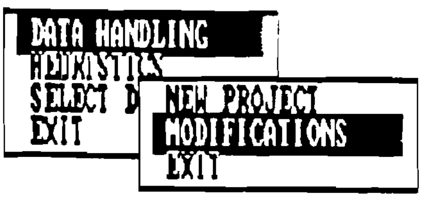
DECISION SUPPORT SYSTEM FOR
MULTIOBJECTIVE RESOURCE-CONSTRAINED PROJECT SCHEDULING
v. 2.02 DEMONSTRATION VERSION 1990

Roman Slowinski, Jan Weglarz,
Institute of Computing Science,
Technical University of Poznan, 60-965 Poznan, Poland
Boleslaw Soniewicki,
Center for Education and Development
in Agriculture, 61-659 Poznan, Poland

This software has been developed within the scientific
cooperation with the System and Decision Sciences Program
of the International Institute for Applied Systems Analysis
A-2361 Laxenburg, Austria
This copy has been licensed free of charge to:
##
##
to be used only for nonprofitable scientific purposes.
Any other use of this software requires written
permision from the authors.

Fig. 2

Multiobjective Resource-Constrained Project Scheduling



The Name of the Active File :

Select item with ↑ or ↓ , then hit RETURN

Fig. 3

Multiobjective Resource-Constrained Project Scheduling

DATA HANDLING
HEURISTICS
SELECT D
EXIT

SINGLE OBJECTIVE
MULTIOBJECTIVE
EXIT

PRIORITY PROCEDURES
SIMULATED ANNEALING
BAB TYPE PROC
EXIT

COMPLETION TIME
RESOURCE SMOOTHING
MEAN WEIGHTED ACTIVITIES LATENESS
TOTAL NUMBER OF TARDY ACTIVITIES
MEAN FLOW TIME
NET PRESENT VALUE
WEIGHTED RESOURCE CONSUMPTION
EXIT

The Name of the Active File : RADZIMRA

Select item with ↑ or ↓ , then hit RETURN

Fig. 4

Reading Precedence Relation

Name of the activity **cutting down grass b**

List of immediate predecessors
harrowing rape a
sowing rape a
croopping harv lup b
drying lupine b
croopping of stror b
cutting down lucer b

F1 NEXT FIELD F2 NEXT ACTIVITY F3 PREVIOUS ACTIVITY F4 EXIT

Fig. 5

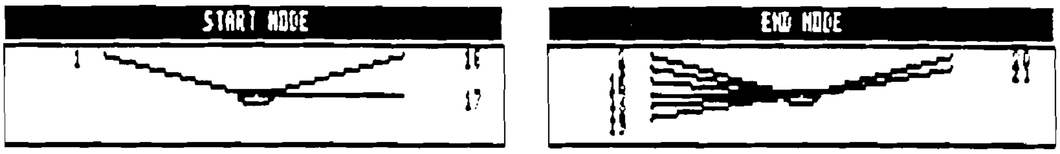
Data for the Project

Due date for the project (in time units)	150
How many types of renewable resources ? (max 4)	2
Name of renewable resource 1	workers
Name of renewable resource 2	tractors
Number of available units of "workers "	200
Number of available units of "tractors "	150
How many types of nonrenewable resources ? (max 3)	1
Name of nonrenewable resource 1	cash
Number of available units "cash "	20000
Inflow(+) or outflow(-) of "cash " at the end of the project	1000
Interest rate (percent)	5

↑↓ NEXT FIELD ENTER FIRST ACTIVITY E END Q QUIT N NUMBER OF ACTIVITY

Fig. 6

Editing activity no. 17 Name of activity drying lupine b



Release time	8
Due date	48
"drying lupine " cannot start before parallel activity no.	16
Weighting factor	1
Number of mode (max 3)	1
Activity duration for mode 1	20
Requir. of renew. resource "workers " per unit duration for mode 1	12
Requir. of renew. resource "tractors " per unit duration for mode 1	12
Requir. of nonrenew. res. " cash " per unit duration for mode 1	11
Inflow(+) or outflow(-) of "cash "at the end of the act mode 1	0

↑↓ NEXT FIELD ENTER NEXT ACTIVITY E SAVE Q QUIT N NO. OF ACTIVITY.

Fig. 7

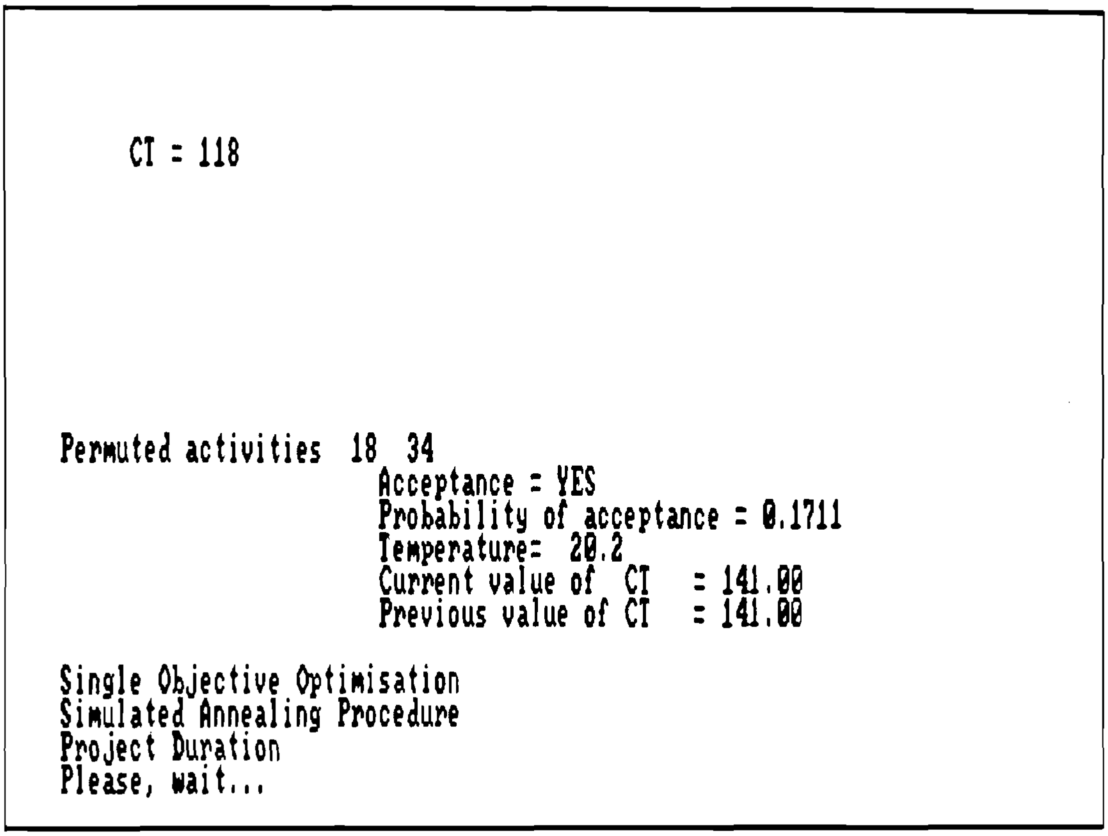

CT = 118
Permuted activities 18 34
Acceptance = YES
Probability of acceptance = 0.1711
Temperature= 20.2
Current value of CT = 141.00
Previous value of CT = 141.00
Single Objective Optimisation
Simulated Annealing Procedure
Project Duration
Please, wait...

Fig. 8

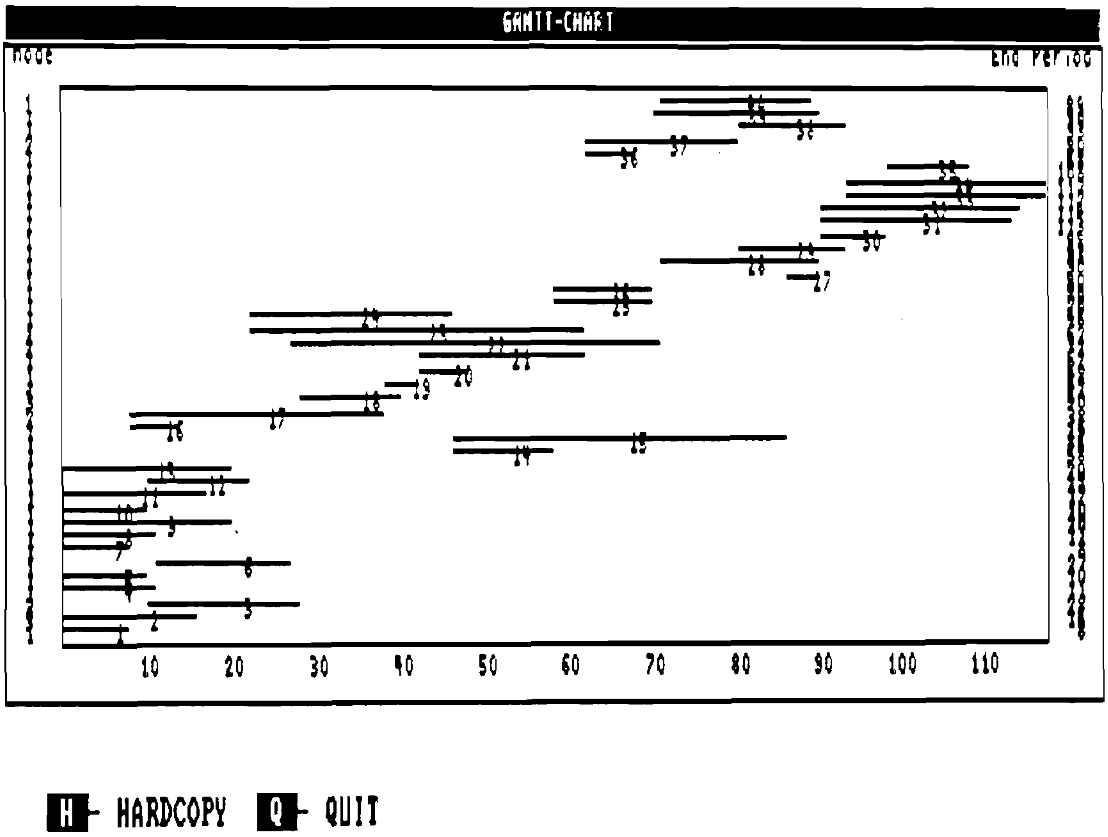

GANTT-CHART
10
20
30
40
50
60
70
80
90
100
110
H- HARDCOPY
Q- QUIT

Fig. 9

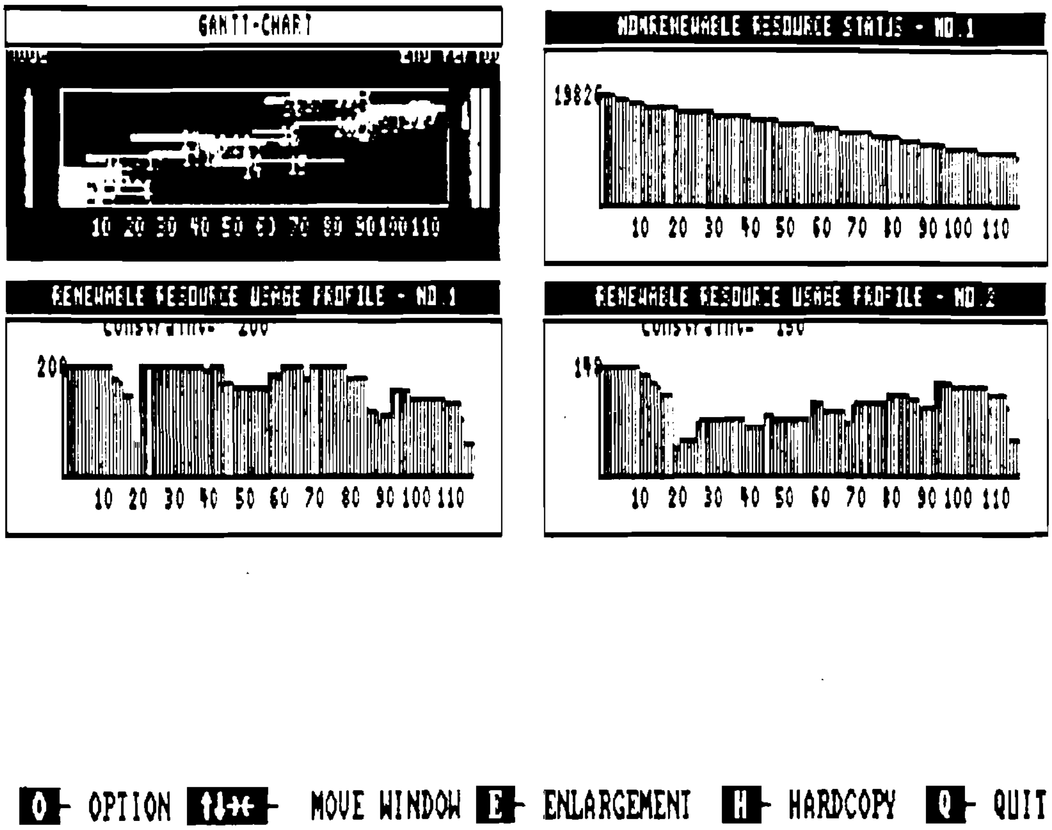

GANTT-CHART
NONRENEWABLE RESOURCE STATUS - NO.1
RENEWABLE RESOURCE USAGE PROFILE - NO.1
RENEWABLE RESOURCE USAGE PROFILE - NO.2
10 20 30 40 50 60 70 80 90 100 110
O - OPTION
MOVE WINDOW
E - ENLARGEMENT
H - HARDCOPY
Q - QUIT

Fig.10

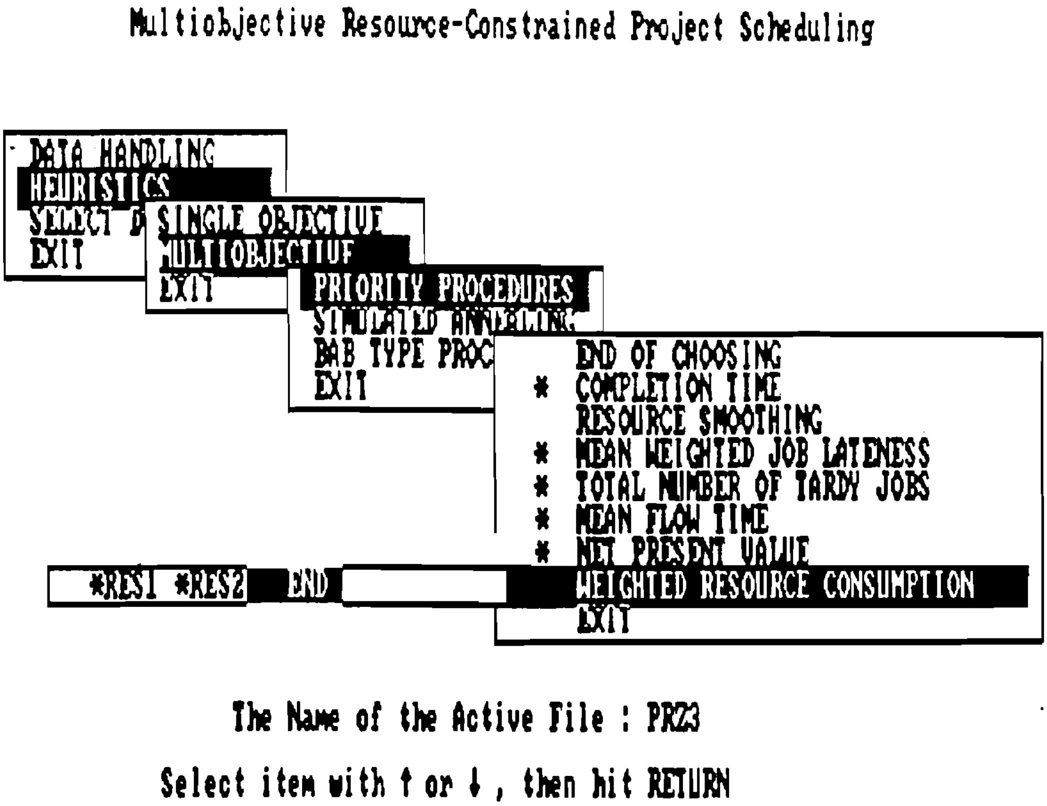

Multiobjective Resource-Constrained Project Scheduling
DATA HANDLING
HEURISTICS
EXIT
SINGLE OBJECTIVE
MULTIOBJECTIVE
EXIT
PRIORITY PROCEDURES
BAB TYPE PROC
EXIT
END OF CHOOSING
* COMPLETION TIME
RESOURCE SMOOTHING
* MEAN WEIGHTED JOB LATENESS
* TOTAL NUMBER OF TARDY JOBS
* MEAN FLOW TIME
WEIGHTED RESOURCE CONSUMPTION
EXIT
*RES1 *RES2
END
The Name of the Active File : PRZ3
Select item with ↑ or ↓ , then hit RETURN

Fig.11

```
                                              NNDS= 25
   Current point              Current ideal point
    CT   =   131.00            CT   =   118.00
   MWJL  =    53.00           MWJL  =    23.00
   TNTJ  =     8.00           TNTJ  =     3.00
   MWFT  =   902.00           MWFT  =   788.00
   NPV1  = 17415.05           NPV1  = 17807.78
   WRC1  = 10576.00           WRC1  = 10131.00


Permuted activities   2  22
                       Acceptance = YES
                       Probability of acceptance = 1.0000
                       Temperature=4500.0
                       Current value of  NPV1  = 17556.05
                       Previous value of NPV1  = 17536.36


Multiobjective Optimisation
Please, wait...
```

Fig. 12

Comparison of the obtained solution with the previous one
and the reference point

NNDS= 24

Objectives	CT	MWJL	TNTJ	MWFT	NPV1	WRC1
Ideal point	118	23	3	788	17808	10131
Nadir point	161	303	16	1300	17313	10829
Previous but one	0	0	0	0	0	0
Previous	0	0	0	0	0	0
Obtained	122	34	7	811	17369	10465
Relax level						
Relaxed level						

N ND SOLUTION WITH IMPROUVED VALUE OF A CHOSEN CRITERION
[illegible] ANOTHER ND SOLUTION
←→ CHOOSE THE FIELD ENTER ACCEPT D DELETE CHANGING V VIEW

If you relaxed at least one field, you continue the calculation

Fig. 13

Comparison of the obtained solution with the previous one
and the reference point

NNDS= 24

Objectives	CT	MWJL	TNTJ	MWFT	NPV1	WRC1
Ideal point	118	23	3	788	17808	10131
Nadir point	161	303	16	1300	17313	10829
Previous but one	128	47	7	887	17548	10373
Previous	122	34	7	811	17369	10465
Obtained	122	59	7	788	17313	10307
Relax level						
Relaxed level		60				

N ND SOLUTION WITH IMPROUVED VALUE OF A CHOSEN CRITERION
: ANOTHER ND SOLUTION
→← CHOOSE THE FIELD ENTER ACCEPT D DELETE CHANGING V VIEW

If you relaxed at least one field, you continue the calculation

Fig. 14

Comparison of the obtained solution with the previous one
and the reference point

NNDS= 24

Objectives	CT	MWJL	TNTJ	MWFT	NPV1	WRC1
Ideal point	118	23	3	788	17808	10131
Nadir point	161	303	16	1300	17313	10829
Previous but one	122	59	7	788	17313	10307
Previous	122	59	7	788	17313	10307
Obtained	122	45	7	826	17498	10282
Relax level						
Relaxed level		60		850		

N ND SOLUTION WITH IMPROUVED VALUE OF A CHOSEN CRITERION
: ANOTHER ND SOLUTION
→← CHOOSE THE FIELD ENTER ACCEPT D DELETE CHANGING V VIEW

If you relaxed at least one field, you continue the calculation

Fig. 15

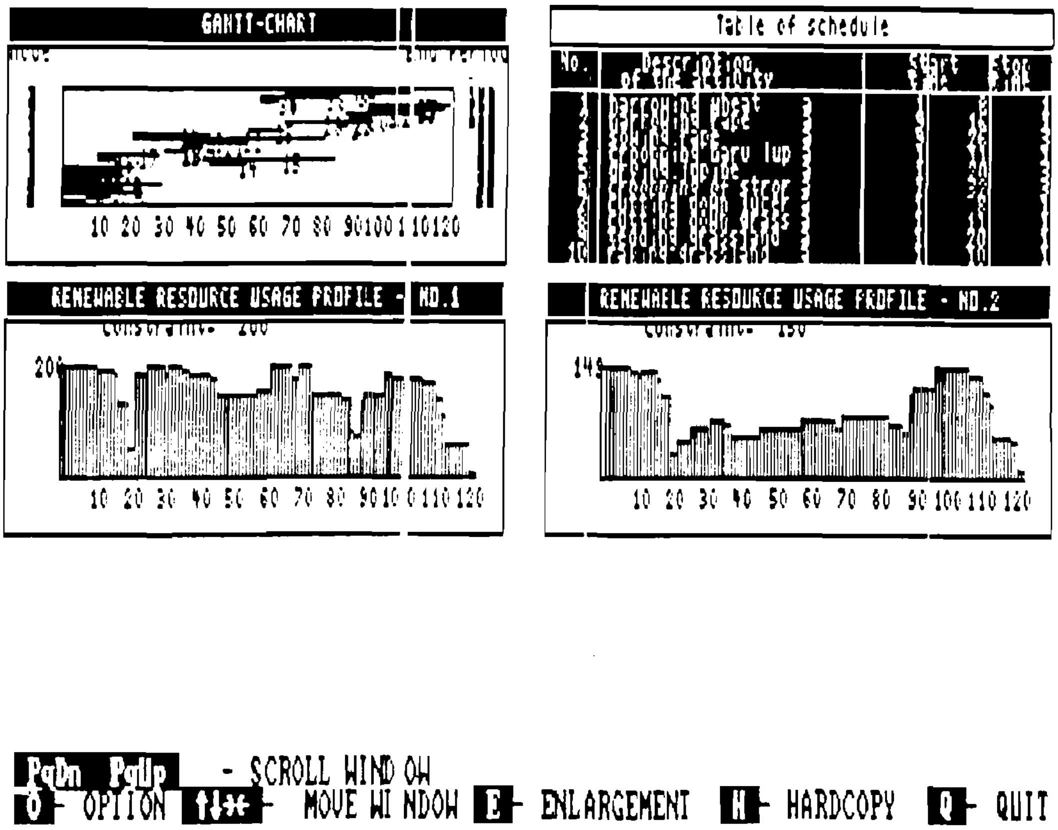

GANTT-CHART
10 20 30 40 50 60 70 80 90 100 110 120
Table of schedule
No.
RENEWABLE RESOURCE USAGE PROFILE - NO.1
10 20 30 40 50 60 70 80 90 100 110 120
RENEWABLE RESOURCE USAGE PROFILE - NO.2
10 20 30 40 50 60 70 80 90 100 110 120
PgDn PgUp - SCROLL WINDOW
O - OPTION - MOVE WINDOW E - ENLARGEMENT H - HARDCOPY Q - QUIT

Fig. 16