CAHIER DU LAMSADE

Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision
(Université de Paris-Dauphine)
Unité Associée au CNRS n° 825

LINEAR PROGRAMMING AND INTERACTIVITY
A MANPOWER SCHEDULING DSS (*)

CAHIER N° 84
janvier 1988

E. JACQUET-LAGREZE
R. MEZIANI

(*) Communication présentée à IFORS '87, 11ème Conférence Triennale de Recherche Opérationnelle, Buenos Aires, Argentine, 10-14 août 1987.

SOMMAIRE

LINEAR PROGRAMMING AND INTERACTIVITY
A MANPOWER SCHEDULING DSS

ABSTRACT

It is now possible to use very efficient micro-computer codes to solve linear programming models. If such a model is included in a more general DSS environment, the user may ask for an optimal solution, and may also progressively modify this solution by taking into account criteria difficult to formalize directly. We illustrate this process on a manpower scheduling problem at AIR-FRANCE for which we present a micro-computer version of the CHEOPS model.

PROGRAMMATION LINEAIRE ET INTERACTIVITE
UN SIAD POUR LA CONSTRUCTION D'HORAIRES

RESUME

Il est possible d'utiliser aujourd'hui des codes de programmation linéaire très puissants sur micro-ordinateurs. On peut construire un modèle interactif permettant à l'utilisateur non seulement de trouver une solution optimale mais aussi de modifier progressivement cette solution en tenant compte de critères difficiles à formaliser. On illustre ce processus sur un problème de construction d'horaires à AIR-FRANCE en présentant une version micro du modèle CHEOPS.

Introduction

In a recent paper Jacquet-Lagrèze et al. (1987) we proposed a multi-objective linear programming method. We thought we had a good opportunity to apply it on a real problem at AIR-FRANCE. In the first section, we present a manpower scheduling problem which is regularly solved by AIR-FRANCE using the CHEOPS model on a mainframe (this initial model is also reviewed in section 1). We tried to implement our multiobjective method on the AIR-FRANCE problem. This attempt is described in Section 2. For reasons we later discuss, applying our model turned out to be too difficult. We therefore designed an interactive monocriterion model, running on a micro-computer. This version of CHEOPS is presented in section 3.

1 - Manpower scheduling - The CHEOPS main frame model

1.1 The problem

The problem is to find an efficient work schedule of different kinds of employees working at an airport using information on the plane schedules, which are known (and designed through other means, taking into consideration workload and competition). The various tasks accomplished by employees are the usual ones : checking-in, catering, aircraft loading, aircraft cleaning, etc.
For each kind of task, we sum up the work demand for each plane, which yields a workload curve for the entire day, having peaks at, for instance, 8 am, 12 noon, 6 pm.

The problem consists in finding either how many employees should be hired for each category of task (problem a) or, given a fixed number of employees, how many should start their work at 6 am, 6.30, am and so on (problem b).
CHEOPS consists of different modules which, for instance:

- Connect to the flight data-base and extracting the pertinent data.
- Compute the day workload curve.
- Compute the different schedules for the day (the scheduling module which we study in detail in the next sub-section).
- Assign employees to each schedule for the whole week.

1.2 The scheduling module

The scheduling module consists of two linear programming models. The first supposes that the workload function is perfectly satisfied, and the second allows violation of some of the constraints in order to reduce the cost (i.e. number of employees required).

The first model, which satisfies the workload curve, is :

$$\text{Min} \quad \sum_{j=1}^{n} c_j \, x_j$$

$$\text{s.t.} \quad \sum_{j=1}^{n} a_{ij} \, x_j \geq b_i \quad \forall_i \ , \ i=1,..,m \qquad (1)$$

$$x_j \in N$$

where

n : number of different teams
m : number of time periods considered to cover 24 hours
x_j : number of employees assigned to team j (j=1,..,n)
c_j : unit "cost" of team j (usually $c_j = 1$)
b_i : workload during period i (i=1,..,m) (i.e. the workload curve)
a_{ij} = 1 if team j covers the period of time i
= 0 otherwise

If c_j=1 for all j , the objective function gives the number of employees necessary to cover the workload curve. The solution (x) gives the number of employees assigned to each team, x_j=0 meaning that the corresponding schedule is not used.

Some additional goals (criteria) are taken into account by introducing some additional constraints. The number of employees cannot exceed a certain amount during a certain period of time (at night especially).

The number of part-time employees must not exceed a given proportion of the total number of employees. All these constraints are linear and do not add any difficulties (see G. Berl 1981).

1.3 Second model: violating in some points the workload curve

In some situations, the minimum given by the first model yields a number of employees higher than the number of available employees (problem b). This means that some of the constraints will have to be violated in order to reduce the cost (i.e. the number of employees). With CHEOPS, this problem is solved by introducing additional slack variables (e_i) as it is done in goal-programming :

$$\sum_{j=1}^{n} a_{ij}\, x_j + e_i \geq b_i \qquad \forall_i \;,\quad i=1,..,m$$

$$e_i \geq 0$$

In order to avoid getting a solution too "distant" from the demand curve, it is possible to constrain the slack variables themselves by adding constraints such as :

$$e_i \leq \alpha\, b_i \qquad i=1,..,m$$

where α is the authorized rate of uncovered workload.
The objective function then gives the priority to minimizing the sum of these slack variables :

$$\text{Min} \quad F = \sum_{i=1}^{m} e_i + \epsilon \sum_{j=1}^{n} c_j\, x_j \qquad (\epsilon \text{ very small})$$

2 - A multicriteria formulation of the problem

2.1 The criteria

As previously mentioned, our aim was to experiment with a MCDM model on this problem.
The goal-programming formulation (second model) shows clearly that there are two conflicting criteria.

(1) - Minimize the total number of employees to be used

$$\text{Min} \sum_{j=1}^{n} c_j\, x_j \qquad (\text{ usually } c_j=1 \text{ }) \;\; j=1,..,n$$

(2) - Minimize the workload not covered

$$\text{Min} \sum_{i=1}^{m} e_i$$

Beside these two main criteria, it is also possible to add other constraints (the proportion of part-time employees,...) considered as objective functions rather than as constraints.

2.2 Outline of the MCDM method

The MCDM method uses three steps (see Jacquet-Lagrèze et al. 1987) :

Step 1: Generating a representative subset (10 to 50) of the set of efficient solutions. This is a technical and non interactive step.

Step 2: Using the PREFCALC method Jacquet-Lagrèze (1985). The Decision Maker (DM) works a with a rather small subset of efficient solutions generated in step 1, using a micro-computer, and the output of this process is an additive piecewise-linear utility function Jacquet-Lagrèze and Shakun (1984). The method is highly interactive and the computer program user friendly, supporting in an efficient way the learning process of the DM.

Step 3: Using the utility function, it is possible to find an optimum for the initial set of alternatives. If each marginal utility function is convex, then a straight-forward LP formulation gives an optimal solution. The main idea of the method is to solve steps 1 and 3 on a mainframe, using an efficient packages of integer programming package (such as the one used for the mainframe CHEOPS model) and to solve step 2 on a micro-computer, since it is the subjective part of the method (assessing a utility function and therefore, implicitly at least, trade-offs among the criteria).

2.3 Presentation of the MCDM formulation to the users and reactions

Although some of the potential users appreciated the MCDM formulation, this approach was rejected for the following reasons :

- As there are 100 to 150 users of CHEOPS, the model has to be easy to use and to understand. The users are not familiar with the MCDM approach and concepts ("efficient solutions" in step 1, and especially "marginal utility functions" used by PREFCALC in step 2).
- AIR-FRANCE wished to have a micro-computer version of CHEOPS. From a practical stand point, step 1 and especially step 3 would have generated LP formulations with too many constraints and variables to be solved on a micro-computer.
- The nature of the problem enables an easy and natural graphic representation of any solution and the way it fits the workload curve (see fig. 2 for example).
- The violation of the workload curve is better presented graphically than by an aggregate computed number $\sum_{i=1}^{m} e_i$.
- We had at the same time developed efficient LP algorithms on micro-computers Jacquet-Lagrèze (1987), so that it seemed reasonable to try and build a simplified version of CHEOPS on a micro-computer.

3 - An interactive micro-computer version of CHEOPS

3.1 Presentation of the micro version of CHEOPS

Considering the remarks mentioned above, we decided to design a micro-computer version of CHEOPS that would be highly interactive, and monocriterion based. We wished to include the following features:

- A tool easily used and understood by users.
- A running time as short as possible so as to facilitate interactivity.
- Graphic representations of the solution (schedules represented by Gantt diagrams), of the workload curve, and of the extent to which the solution satisfies the corresponding constraints.

- The possibility to manually modify any solution, precisely because all the criteria cannot be formalized.

In order to get short running times, we implemented a simplified version (without the integer condition of the x_j). Using the projected gradient method Jacquet-Lagrèze (1987), the running time for three different cases is given in the table 1, using a compatible PC.

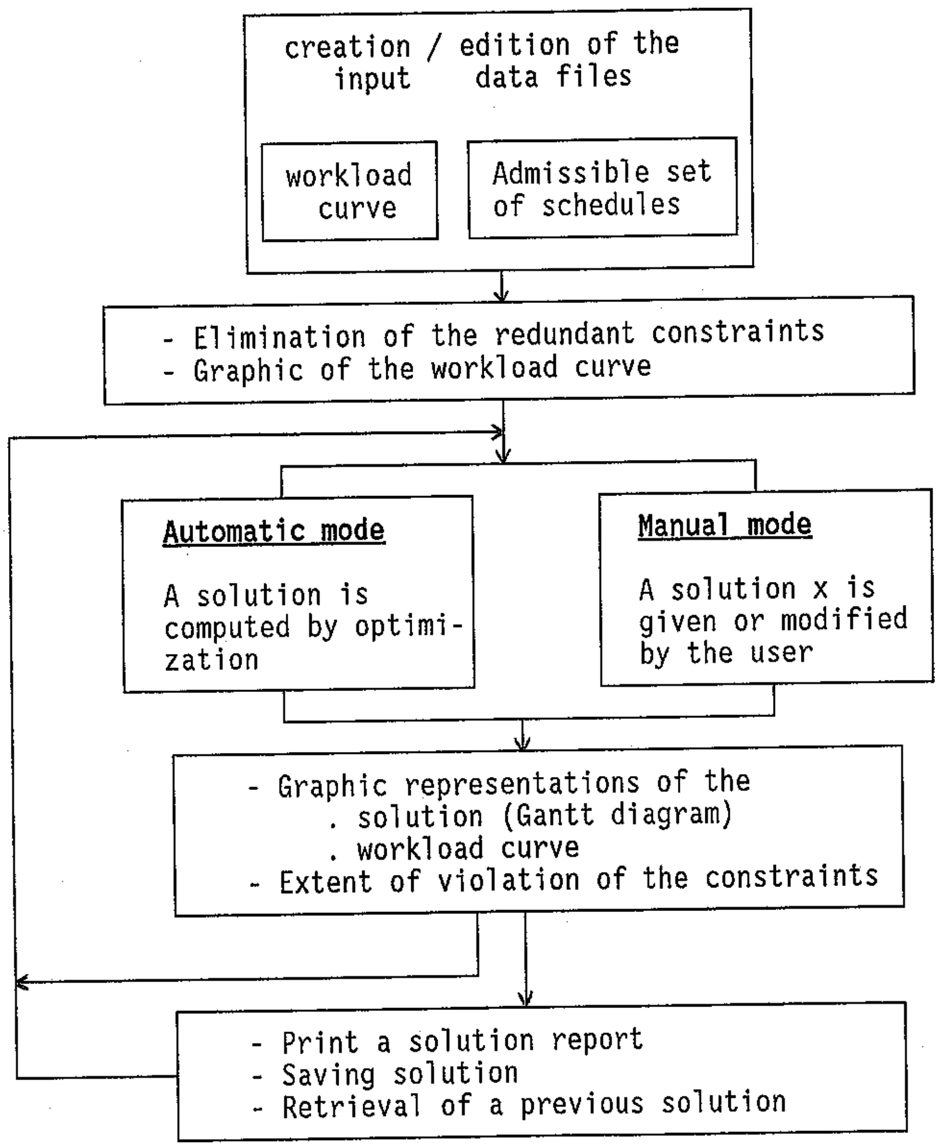
Case	n	m	sec
1	22	22	2
2	105	47	22
3	295	60	180

Table 1 : The running times

Case 3 was the biggest application with which we experimented, dealing with 295 possible teams. The usual period of time considered to express the constraints relative to the workload curve being 10 minutes, 144 constraints are generated. Many of these constraints are redundant, educing to 47 non redundant constraints in case 3. If we add the other constraints as discussed in section 1.3, we get to 60.

Had we explicitly kept the slack variables e_i, it would not have been possible to reduce the amount of constraints, and we would have had to add the 144 slack variables e_i, thus yielding a LP with 295+144=439 variables.

The structure of the micro-computer version of CHEOPS is as follows :



<u>The dialogue management is implemented using the following options</u>

The different options of CHEOPS are the following :

<D> <u>Data</u> : The data management system is not yet fully implemented. Therefore, one has to quit CHEOPS and edit the corresponding ASCII with any text editor.
<O> <u>Optimize</u> : It enables to use the <u>automatic mode</u>. Since the projected gradient method starts from a given solution, this initial solution is either x=0 whenever we start by using <O> first, or x="manual solution" whenever we start from a manually entered (modified) solution. Since there are many optimal solutions in this type of problem , this possibility is useful compared to the simplex method because the optimal solution can be closer to a given manual solution.
<M> <u>Manual</u> : The user can enter or modify any solution, the optimal one for instance. He can decide not to respect the workload curve at a particular point of the day. This option is illustrated in the next paragraph.
<C> <u>Workload curve</u> : Since the ordinary graphic card shows only one colour in high resolution mode, this option enables to show only the workload curve, or the workload curve plus the solution curve (i.e b_i and $\sum_{j=1}^{n} a_{ij}\, x_j$). The difference at each point represents the classical slack :
$\sum_{j=1}^{n} a_{ij}\, x_j - b_i$, and where the demand curve is higher, the difference represents the slack variable introduced in the previous section
$e_i = b_i - \sum_{j=1}^{n} a_{ij}\, x_j$, but not introduced in our formulation.
<S> <u>Save</u> : Results (solution x).
<R> <u>Retrieve</u> : Results (solution x).
<I> <u>Print</u> : Results (detailed report of the solution).
<Q> <u>Quit</u> : CHEOPS.

3.2 <u>Examples of the interactive use of the system</u>

Scenario 1 : <u>Find the "optimal" size of a team</u>

It corresponds to problem a) . We wish to know how many employees should be available at the airport, and prepare decisions such as : hire new employees, move some employees to other jobs and/or airports. The scheduling of the employees is here of little interest, we are more concerned with knowing the number of necessary employees (c_j=1 for all j).

Step 1: <ins><O>ptimize</ins>. As an example we get figure 2, with a cost of 105 employees to satisfy the demand curve.

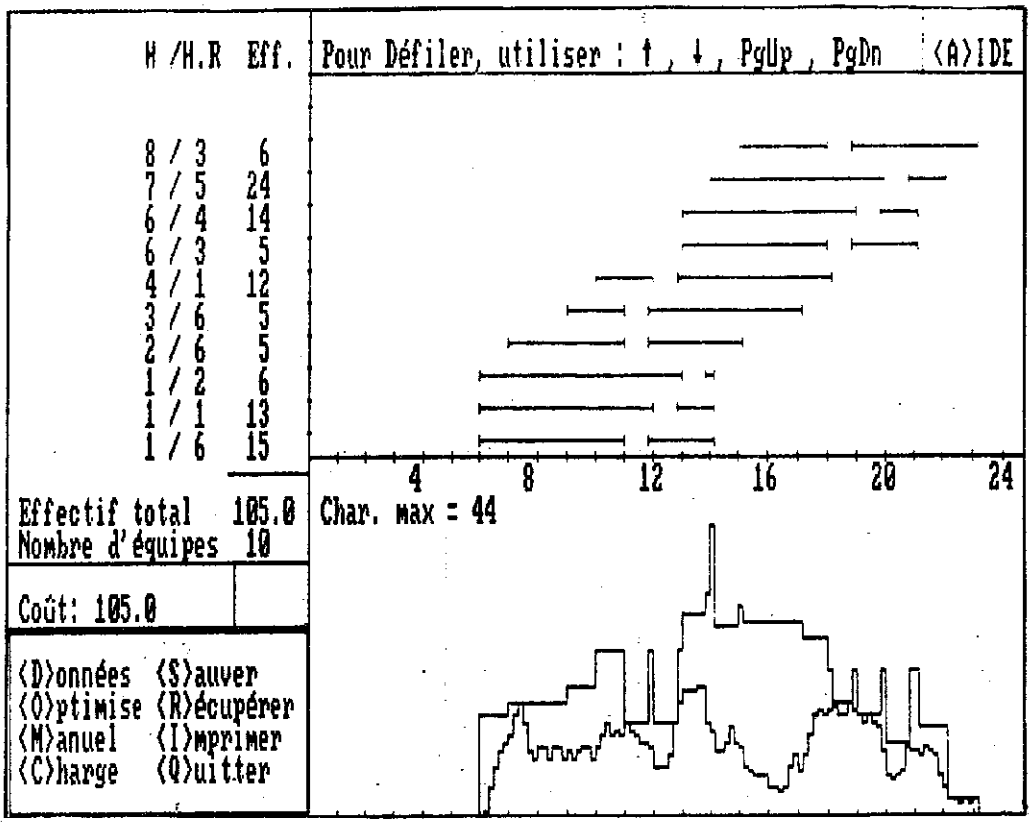


Figure 2 : <ins>Optimization</ins>

Step 2: If we are prepared not to satisfy some of the peaks of the workload curve (at 7.20 am for instance), we can look for one of the teams 1 or 2. We for instance reduce manually (<M> option) the number of employees of team 1/6 from 15 to 12 : 3 employees will be missing for 10 minutes. Similarly for the team 4/1 : (12 --> 9). We get in this way a solution with 99 employees (see figure 3) with a detailed report given by option <I> (see figure 4).

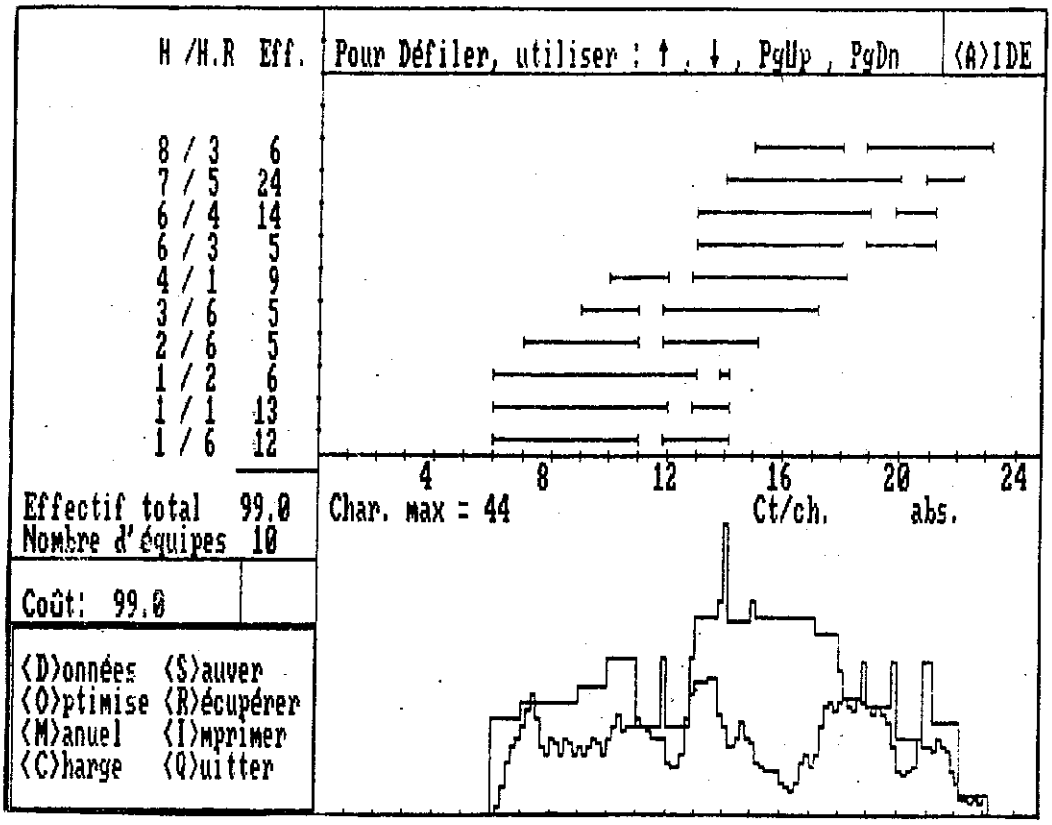


Figure 3 : Manual

C.H.E.O.P.S

PROGRAMME DE DETERMINATION DE LA COUVERTURE HORAIRE

PAS DE L'ETUDE : 10 MINUTES
FICHIER CHARGE : charge12
FICHIER HORAIRE : horaire2

NOMBRE D'HORAIRES RETENUS PAR CHEOPS : 10
COUT DE L'EFFECTIF SELECTIONNE : 99

HORAIRES RETENUS ET NOMBRE D'AGENTS PAR HORAIRE

Numéro Horaire	Numéro Repas	Nombre d'agents	Début	fin	début	fin
1	6	12	600	1100	1145	1409
1	1	13	600	1200	1245	1409
1	2	6	600	1300	1345	1409
2	6	5	700	1100	1145	1509
3	6	5	900	1100	1145	1709
4	1	9	1000	1200	1245	1809
6	3	5	1300	1800	1845	2109
6	4	14	1300	1900	1945	2109
7	5	24	1400	2000	2045	2209
8	3	6	1500	1800	1845	2309

Figure 4

Scenario 2 : Find a schedule with a given number of employees

We assume here that the number of employees is given (90 for instance) and cannot be changed in the short term (problem b).

Step 1: We add a constraint specifying that the total number of employees is equal to 90, and use as an objective function the number of employees present after 10 am (schedule 4/., 5/., 6/., 7/., 8/.). to make this modification, we choose the <D> option in order to modify the data file of the schedule.

- Choosing option <O> gives an optimal solution (figure 5)

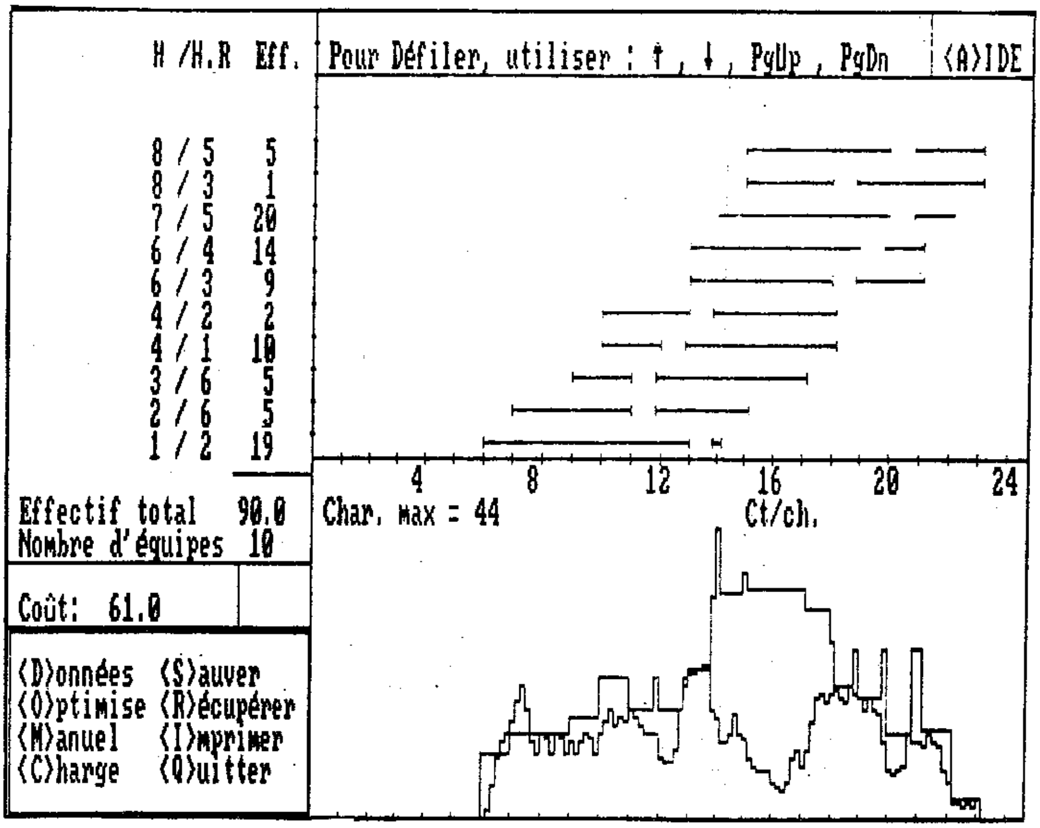


Figure 5 : Optimization

Step 2: We decide to modify manually <M> this initial solution, trying to get another repartition of the 90 employees. In this example we see that the second criterion (Min $\sum_{i=1}^{m} e_i$, see section 2) remains implicit. The advantage is that we can more easily introduce consideration such as "having two $e_i>0$ that are not consecutive is preferable to two consecutive e_i ". The manual modification as in scenario 1 leads to the solution shown in figure 6. The latter consideration relies on a non formalized way to judge a solution by considering only the graphics.

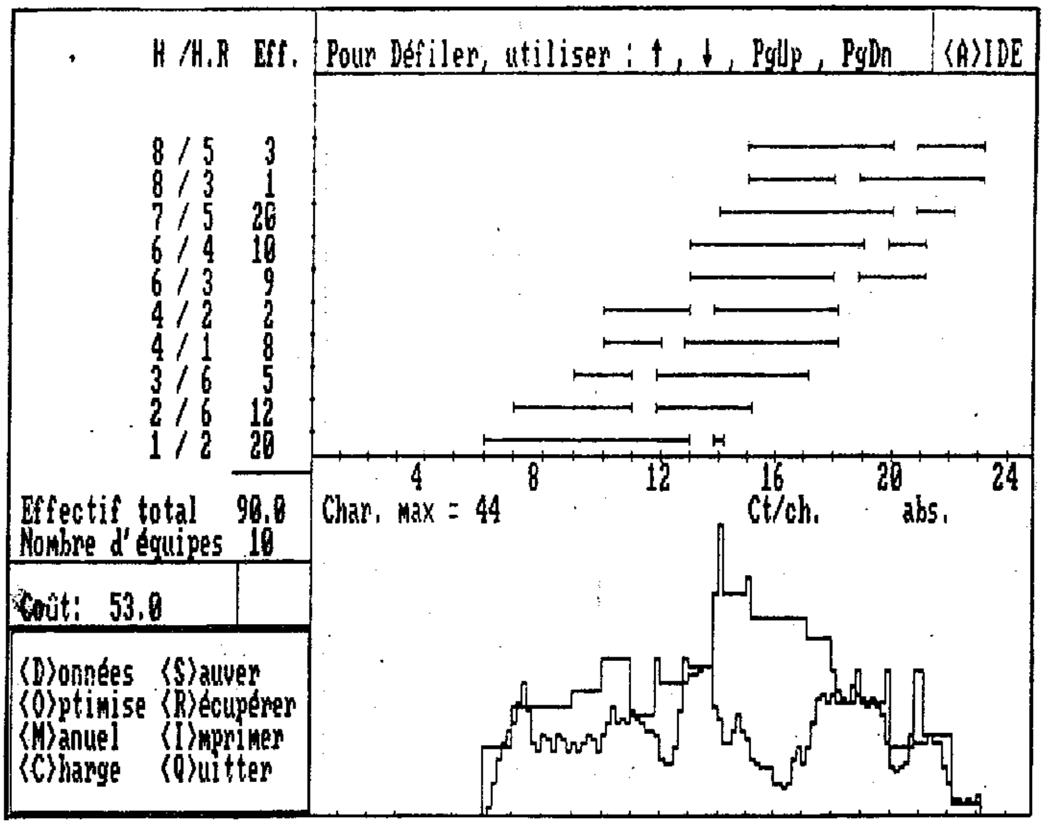


Figure 6 : Manual

Conclusion

This micro-computer version of CHEOPS is fully operational. AIR-FRANCE has already started to use it experimentally for solving such problems. These first experiments will show how the program can be modified in order to support in a more efficient way a great number of users not familiar with highly sophisticated techniques of OR. The question about the use of an explicit Multicriteria LP formulation remains open.

References

G. Berl (AIR-FRANCE), "Airport manpower planning two implemented models " XXIst AGIFORS SYMPOSIUM SANTA-BARBARA, Oct. 1981.

E. Jacquet-Lagrèze, "PREFCALC - Evaluation et Décision Multicritère", Euro-Décision 1983.

E. Jacquet-Lagrèze "A projected gradient method for linear programming" EJOR, cahier du LAMSADE No. 79, juin 1987.

E. Jacquet-Lagrèze, M.Shakun "Decision support systems for semi-structured buying decisions", EJOR, Vol. 16, No. 1 (1984), pp. 48-58.

E. Jacquet-Lagrèze, R. Meziani, and R. Slowinski, "MOLP with an interactive assessment of a piecewise linear utility function", EJOR, Vol. 31, No. 3, Sept 1987, pages 350-357.