An employee timetabling problem in a maintenance service of a software company

Laure-Emmanuelle Drezet[1], Deborah Chesnes[2] and Odile Bellenguez-Morineau[2]

[1] Ecole des Mines de Saint-Etienne, CMP Georges Charpak, Avenue des Anemones - Quartier Saint-Pierre, 13541 Gardanne, France
drezet@emse.fr

[2] Laboratoire d'Informatique, 64 avenue Jean Portalis, 37200 Tours, France
odile.morineau@univ-tours.fr; chesnes@etu.univ-tours.fr

1 Introduction

The considered problem occurs in the maintenance service of a French software company, Delta Informatique. This company sells software that allow to manage bank information systems. The 24 employees of the maintenance service have to ensure the maintenance of these software. They have also to solve dysfunctions that could occur in these products or to give assistance to customers. The employees' timetables have though to satisfy the clients' requests, within a planning horizon of three months.

The help desk is closed only one day per week (from Saturday at 2:00PM to Sunday at 2:00PM). Four shifts are defined for the help desk: MS - Morning shift (from 6:00 AM to 2:00 PM), AS - Afternoon shift (from 2:00 PM to 10:00 PM), NS - Night shift (from 10:00 PM to 6:00 AM), SS - Sunday shift (from 2:00 PM to 0:00 AM). A shift is defined for the employees not assigned to the help desk: DS - Day Shift (from 9:00 AM to 6:00 PM, with a noon break). To respond to clients requests, two employees have to be assigned to MS, AS and to NS, from Monday to Friday. On Saturday, two employees are assigned to MS. On Sunday, two employees are assigned to SS. Employees not assigned to help desk, work from Monday to Friday on a classical shift, DS. These employees develop new functions in software.

In this problem, several hard constraints have to be respected. Obviously, the workforce requirements and the labour legislation must be respected. Each employee must have at least eleven hours between two working days. Each employee must have at least two days-off per week. Moreover, if an employee works on week-ends or on public holiday, some days-off have to be inserted in his/hers timetable.

Another constraint concerns the spread of skills among the present employees. The employees of the maintenance service are divided in five different skill

E. K. Burke, H. Rudová (Eds.): PATAT 2006, pp. 391–393. ISBN 80-210-3726-1.

groups. Each skill group masters a specific part of the software. When an employee is assigned to the help desk, her/his skill group has one unavailable employee. Though the employees assigned to the help desk must belong to different skill groups. Moreover, the customers could call the help desk for different problems. Two employees that are assigned to the same shift in the help desk have to belong to different skill groups, such that the number of mastered skills at the help desk is maximised.

Employees have also different levels: junior or senior. A senior employee has been working for the company for a long time and masters more skills than a junior employee. As only two employees work at night, at least one of them has to be a senior employee.

Several preferences of employees have also to be considered. Each employee can specify a few number of days when she/he can not be assigned to specific shifts. Each employee have also to take a given number of days-off within the three months. A part of these days are fixed by the employee. The remaining days are set by the planner, with respect to hard constraints.

In addition to all these hard constraints, timetables should be as fair as possible. This fairness is evaluated by computing, for each employee, the number of worked nights, week-ends or public holidays and the number of satisfied preferences. Car sharing could also be considered, such that two employees can specify that they prefer to work on the same shifts.

We can observe that this problem is quite close to Nurse Scheduling Problem (as in [1]). As the planner builds timetables by hand, our goal is to propose an automated procedure that allows to find feasible solution quite rapidly.

2 Model

First of all, we have developed an Integer Linear Program. The model is very huge, as all the hard and the soft constraints are considered. The objective function (to minimize) is a weighted sum of all the violated soft constraints and a measure of unfairness. This approach was developed to know the limits of this type of exact approach for this type of problem. Some computational experiments have been carried. As forecasted, with a free solver (GLPK), it takes more than a day to compute a simple timetable for about fifteen employees, for two weeks. As mentioned in [2], it is not surprising. In fact, Integer Linear Programming is not very efficient when global constraints or employees' preferences have to be considered.

Then a priority-rule based heuristic was developed. This greedy algorithm reproduces the human way to build timetables and its first results are encouraging (Timetables are computed in less than five minutes for all the employees

and for three months). This method uses different priority rules and different strategies to assign shifts to employees.

3 Conclusion

The studied timetabling problem is quite different from classical timetabling problems, due to some specific hard constraints and to the huge number of considered soft constraints. Mathematical programming could not be used for solving this problem, due to the size of the model. The greedy method we propose obtain good-quality solutions, in less than five minutes. These results can be improved by using a metaheuristic, such as a genetic algorithm.

References

1. Bellanti, F., Carello, G., Della Croce, F., Tadei, R.: A greedy-based neighborhood search approach to a nurse rostering problem In: European Journal of Operational Research (2004), Vol. 153, 28–40
2. Meisels, A., Gudes, E., Solotorevsky, G.: Combining rules and constraints for employee timetabling In: International Journal of Intelligent System (1997), Vol. 12, 419–439