Min Conflicts Based Heuristics for Rotating Workforce Scheduling Problem

Nysret Musliu*

*Vienna University of Technology
Favoritenstrasse 9-11, A-1040 Wien, Austria
musliu@dbai.tuwien.ac.at

1 Introduction

Workforce scheduling problems are typical constraint satisfaction problems in which different constraints must be fulfilled, like needed workforce per day, forbidden sequences of shifts, length of work and days-off blocks etc. There are two main variants of workforce schedules: rotating (or cyclic) workforce schedules and non-cyclic workforce schedules. In a rotating workforce schedule all employees have the same basic schedule but start with different offsets. For this problem we have given (for formal definition see [10]): number of employees: n, set of m shifts, length of schedule, temporal requirements, constraints about the sequences of shifts permitted to be assigned to employees (the complement of inadmissible sequences), maximum and minimum length of periods of consecutive shifts and maximum and minimum length of blocks of work days and days off. The aim is to find a cyclic schedule (assignment of shifts to employees) that satisfies the requirement matrix and fulfill all given hard constraints.

For solving the problems in workforce scheduling, different approaches have been used in the literature. For recent survey of algorithms used for different workforce scheduling problems the reader is refereed to [1]. Balakrishnan and Wong [2] solved a problem of rotating workforce scheduling by modeling it as a network flow problem. Laporte [6] considered developing the rotating workforce schedules by hand and showed how the constraints can be relaxed to get acceptable schedules. Several other algorithms for generation of workforce schedules have been proposed in the literature [5, 9]. Musliu et al [10] proposed and implemented a new method for the generation of rotating workforce schedules, which is based on pruning of search space, by involving the decision maker during the generation of partial solutions. The algorithms have been included in a commercial product called First Class Scheduler (FCS) which is part of a shift scheduling package called Shift-Plan-Assistant (SPA) of XIMES[1] Corp. This system has been used by several companies in Europe since 2000. Although this product has been shown to work well in practice for solving of a broad range of real life problems, for very large instances solutions cannot always be guaranteed, because of the size of the search space. In this paper we propose methods with aim of improving of a system First Class Scheduler.

[1] http://www.ximes.com/

Of particular interests was to develop techniques for this problem which can solve problem instances which can not be solved by FCS in a reasonable amount of time.

In this paper we propose a method which includes so called tabu mechanism in min conflicts based heuristic for solving rotating workforce scheduling problem. The min conflicts based heuristics is adopted according to specifics of this problem. Further we consider introducing of random noise and random walk in adopted random restart min conflicts heuristics for this problem. The proposed methods are evaluated empirically in set of real life test problems and benchmark examples form literature. Empirically we show that the methods proposed in this paper give better results for real life problems compare to commercial system First Class Scheduler, which to our best knowledge is the state of art system for generation of rotating workforce schedules.

1.1 Min conflicts based heuristic for rotating workforce scheduling

The min conflicts heuristic ([7]) has been used successfully for solving constraint satisfaction problems. In this technique during each iteration randomly a conflicted variable is selected. For the selected variable the new value is picked such that the number of conflicts is minimized. Pure min conflicts technique can also get stuck in a local optimum. To escape the local optimum the algorithm can be restarted or noise strategies such as random walk ([11]) can be introduced to min conflicts heuristic. In this paper we adopt all these variants of pure min conflicts heuristics for the rotating workforce scheduling problem. Due to the specifics of problem the techniques proposed here have some differences compare to pure min conflicts heuristic.

Further we describe the adopted random restart min conflicts heuristic for the problem we consider in this paper. A cyclic schedule is represented by an $n \times w$ matrix S. Each element $s_{i,j}$ of matrix S corresponds to one shift. Element $s_{i,j}$ shows which shift employee i works during day j or whether the employee has time off. The initial solution is generated so that it fulfills all workforce requirements. However, the position of shifts inside of column is selected randomly. Considering generation of neighborhood for the current solution in the simplest case the neighborhood is generated by swapping of content of cell which is randomly selected from cells which appear in some violated constraint with the content of other cells appearing in same column. The swap is limited between cells which appear in same column, because with this limitation is assured that solutions fulfill always the workforce requirements. We experimented also with swapping of block of cells.

The adapted min conflicts based heuristic for rotating workforce scheduling has these differences compared to pure random restart min conflicts heuristic due to the specifics of this problem: In our case not only one variable in conflict changes the value. Here is possible that more than one variable changes their content. In case of simple neighborhood two variables will change their contents, whereas in case of swap of block of length 4, their content will change 8 variables. In our procedure also a worse solution can be accepted for the next iteration, whereas in pure min conflicts not. Considering minimizing of conflicts the basic idea of minimizing of number of conflicts remains, but we evaluate solution not only based on number of conflicts, but also based on the degree of constraint violation ([8]). To calculate the fitness of the solution, for each violation of constraint a penalty is given, based on the constraint and the degree of the constraint violation. The fitness represents the sum of all penalties caused from the violation of constraints. For exact penalties and calculation of fitness for this problem the reader is refereed to [8]. Similar approach for calculating of fitness based on penalty functions is proposed by Galinier and Hao [3].

Further we considered combination of min conflicts with random walk and random noise. The random walk based strategy in our case picks with probability p, first randomly one cell which appears in some violated constraint (conflict). The content of this cell is swapped with the content of other cell which is selected randomly in the same column. In method which combines min conflicts with random walk during each iteration with probability p the random walk procedure is applied, whereas with probability $1-p$ the min conflicts based procedure is followed.

The procedure in which random noise is introduced in min conflicts is the same as when the random walk is introduced, except that in this case with probability p both cells to be swapped are selected randomly, i.e. the column and the rows of cells to be swapped are selected randomly. The probability p has an important impact in the search process and we determined it by experimenting with these values: 2, 5, 10, 15.

1.2 Introducing tabu mechanism in min conflicts based heuristic

We propose an extension of min conflicts based heuristic described above by introducing the so called tabu mechanism ([4]) in the min conflicts heuristic. The basic idea is to store information about the swap of cells during each iteration. Information about the history of swaps is then used in the selection process of the best solution from neighborhood to be accepted for next iteration. We proceed as follows. Each swap used for obtaining the solution for next iteration is stored in tabu list. Additionally, also the inverse swap is stored in this list. For example if the solution accepted for the next iteration is obtained with swapping of contents of cells 4 and 5 in the first column (move $swap(1,4,5)$) of schedule, in the tabu list are stored these moves: $swap(1,4,5)$ and $swap(1,5,4)$. The information for swaps is kept in the list only for determined number of next iterations. Note that during the swap of block of cells also the information for the length of blocks is stored in history of swaps. During the selection of solution from the neighborhood for the next iteration the solution obtained from the swaps stored in the tabu list are not taken in considerations for selection. An exception is done if the solution has best evaluation so far (aspiration criteria [4]).

The main difference compare to min conflicts based heuristics is that in this procedure not always the solution which is the best considering minimizing of number of conflicts is selected for the next iteration. The solution will be accepted for the next iteration only if it is obtained with moves which were not stored in a tabu list during the past determined number of iterations. This should help avoiding cycles or repetitions of same solutions during the search. The only case in which a solution which is obtained with the forbidden swaps is accepted is the case when the solution has the best fitness so far.

Considering length of tabu list we use such tabu list, such that the length of tabu list is dependent from the size of problem. We experimented with these lengths of tabu list: $0.5*n, 1*n, 4*n, 7*n, 10*n$, where n represents number of employees.

2 Computational results

In this section we report on computational results obtained with the current implementation of methods described in this paper. We give the results for 20 test problems. For comparison we take into consideration the time for which the solution is found. The experiments were performed in a machine Pentum 4, 1,8GHZ, 512 MB RAM. For each problem 10 independent runs were executed. The maximal number of evaluations for each run is 10 mil. evaluations.

We give computational results for three test problems which appeared previously in literature and for other real life examples which we collected from a broad range of organizations, like airport, factory, health care organization, etc. The collection of all these problems can be found on http://www.dbai.tuwien.ac.at/staff/musliu/benchmarks/. The problems for which we give the results in this paper contains from 7 to 64 groups. Two test problems (19, and 20) are very large instances and contains respectively 120 and 163 groups. Each group can contain one or more employees. The first three examples are benchmark examples we could find in literature with some modification according the definition of problems we solve in this paper. The description of these three examples is given in [2] and [10]. Other 17 examples appeared in real life situations in different organizations.

2.1 Results

Further we give the best results that could be obtained for each technique described in this paper. In the Table 1 a summary of results for all techniques is given. In the second column of table the size of problem is shown. Results for random restart min conflicts based heuristic are presented in column three and four. The column named 'Sol.' represents number of solutions found in 10 runs. Considering random restart, for each run 10 random restarts are perfomed. Maximal number of evaluations per random restart is 1 mil. evaluations (for each run total 10 mil. evaluations). Further, in next columns of Table 1 results for min conflicts based strategy and random walk (MC-RW) and for min conflicts based strategy in which tabu mechanism (MC-T) is included are given. The last column represents the results obtained with commercial software First Class Scheduler. We also experimented to additionally introduce random noise and random walk in MC-T. However, that did not improved results obtained only by MC-T and these results are not presented here. For each problem the average time needed (in seconds) for generation of solution is presented. For calculation of averages only the successful runs are considered.

From Table 1 we can conclude that the ingredients given to random restart min conflicts improves this technique for the rotating workforce scheduling problem. Introduction of random walk makes possible to solve all problems in each run. Finally results show that introducing of tabu mechanism in min conflicts heuristics makes this technique more powerful for the problem we consider here. The results obtained by MC-T are slightly better considering the average time for solving of problems compare to min conflicts in which the random walk strategy is introduced.

The last column in Table 1 represents time needed for generation of first solution with a system First Class Scheduler ([10]). To our best knowledge FCS is the state of art commercial systems for generation of rotating workforce schedules. In [10], a comparison of methods

Table 1: Results for three techniques

		MC		MC and RW		MC-T		FCS
Ex.	Groups	t(sec)	Sol	t(sec)	Sol	t(sec)	Sol	t(sec)
1	9	4.77	10	0.07	10	0.07	10	0.9
2	9	1.48	10	0.11	10	0.07	10	0,4
3	17	69.36	10	0.68	10	0.42	10	1.9
4	13	0.12	10	0.13	10	0.11	10	1.7
5	11	15.78	10	0.48	10	0.43	10	3.5
6	7	2.89	10	0.06	10	0.08	10	2
7	29	62.51	10	51.68	10	52.79	10	16.1
8	16	32.52	10	0.63	10	0.74	10	124
9	47	84.17	5	11.48	10	15.96	10	>1000(?)
10	27	11.40	10	0.94	10	0.60	10	9.5
11	30	254.82	1	10.02	10	13.15	10	367
12	20	74.26	10	2.17	10	1.17	10	>1000(?)
13	24	68.32	10	1.74	10	0.87	10	>1000(?)
14	13	8.77	10	0.79	10	0.76	10	0.54
15	64	331.11	7	328.52	10	159.04	10	>1000(?)
16	29	14.48	10	0.90	10	0.54	10	2.44
17	33	54.79	10	1.31	10	2.16	10	>1000(?)
18	53	60.58	10	7.22	10	6.83	10	2.57
19	120	577.96	7	44.18	10	75.83	10	>1000(?)
20	163	183.82	10	78.36	10	71.38	10	> 1000(?)

implemented in FCS with other approaches for benchmark examples in literature is given.

From the Table we can conclude that the methods proposed in this paper outperform FCS almost in all instances. Considering larger instances FCS solves faster instance 7 and 18. For instances in which in column FCS '>1000 (?)' is written, FCS could not find solution in 1000 seconds and it is not clear if FCS can find solutions for these problems. The methods proposed in this paper give to our best knowledge the best results in literature (see [2] and [10]) for three benchmark problems we could find in literature (first three test problems in test instances).

3 Conclusions

In this paper we proposed a method which includes tabu mechanism in min conflicts based heuristic for solving rotating workforce scheduling problem. The min conflicts based heuristics is adopted according to specifics of this problem and we considered introducing of random noise and random walk in adopted random restart min conflicts heuristic. Empirically we showed that the given ingredients to min conflicts based heuristics improve this method for solving rotating workforce scheduling problem. Furthermore, empirical results show that the methods proposed in this paper improve the performance of commercial workforce scheduling system First Class Scheduler.

References

[1] H. K. Alfares. Survey, categorization, and comparison of recent tour scheduling literature. *Annals of Operations Research*, 127(1-4):145–175(31), 2004.

[2] Nagraj Balakrishnan and Richard T. Wong. A network model for the rotating workforce scheduling problem. *Networks*, 20:25–42, 1990.

[3] P. Galinier and J.K. Hao. A general approach for constraint solving by local search. *Journal of Mathematical Modelling and Algorithms*, 3(1):73–88, 2004.

[4] Fred Glover and Manuel Laguna. *Tabu search.* Kluwer Academic Publishers, 1997.

[5] R. Hung. A multiple-shift workforce scheduling model under 4-day workweek with weekday and weekend labour demands. *J Opl Res Soc*, (45):1088–1092, 1994.

[6] G. Laporte. The art and science of designing rotating schedules. *Journal of the Operational Research Society*, 50:1011–1017, 1999.

[7] Steven Minton, Mark D. Johnston, Andrew B. Philips, and Philip Laird. Minimizing conflicts: a heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence*, 58:161–205, 1992.

[8] Nysret Musliu. Applying tabu search to the rotating workforce scheduling problem. In *The 5th Metaheuristics International Conference (MIC'03), Kyoto, Japan*, 2003.

[9] Nysret Musliu. Local search strategies for rotating workforce scheduling. In *Paper presented at the Workshop on Design and Evaluation of Advanced Hybrid Meta-Heuristics, Nottingham, UK*, 2004.

[10] Nysret Musliu, Johannes Gärtner, and Wolfgang Slany. Efficient generation of rotating workforce schedules. *Discrete Applied Mathematics*, 118(1-2):85–98, 2002.

[11] Richard J Wallace and Eugene C Freuder. Heuristic methods for over-constrained constraint satisfaction problems. In *Overconstrained Systems, Springer 1106*, 1996.