Proceedings of the International MultiConference of Engineers and Computer Scientists 2014 Vol II,
IMECS 2014, March 12 - 14, 2014, Hong Kong

Scheduling a Professional Sports League using the PEAST Algorithm

Kimmo Nurmi, Jari Kyngäs, Dries Goossens, Nico Kyngäs

***Abstract*— Generating a schedule for a professional sports league is an extremely demanding task. Good schedules have many benefits for the league, such as higher incomes, lower costs and more interesting and fairer seasons. This paper presents the format played in the Finnish major ice hockey league in the 2013-2014 season. The format is very complicated requiring computational intelligence to generate an acceptable schedule. We have used the PEAST algorithm to schedule the league since the 2008-2009 season. We report our computational results especially for the 2013-2014 season.**

***Index Terms*— sports scheduling; real-world scheduling; PEAST algorithm**

I. Introduction

In the past decades professional sports leagues have become big businesses; at the same time the quality of the schedules have become increasingly important. This is not surprising, since the schedule directly impacts the revenue of all involved parties. For instance, the number of spectators in the stadiums, and the traveling costs for the teams are influenced by the schedule. TV networks that pay for broadcasting rights want the most attractive games to be scheduled at commercially interesting times in return. Furthermore, a good schedule can make a tournament more interesting for the media and the fans, and fairer for the teams. Nurmi et al. [1] report a growing number of cases where academic researchers have been able to close a scheduling contract with a professional sports league owner. Excellent overviews of sports scheduling can be found in [2]-[5]. An extensive bibliography can be found in [6] and an annotated bibliography in [7].

In a sports tournament, n teams play against each other over a period of time according to a given timetable. The teams belong to a *league*, which organizes games between the teams. Each game consists of an ordered pair of teams, denoted *(i, j)* or *i-j*, where team *i* plays *at home* - that is, uses its own *venue* (stadium) for a game - and team *j* plays *away*. Games are scheduled in *rounds*, which are played on given *days*. A *schedule* consists of games assigned to rounds. A schedule is *compact* if it uses the minimum number of rounds required to schedule all the games; otherwise it is *relaxed*. If a team plays two home or two away games in two consecutive rounds, it is said to have a *break*. In general, for reasons of fairness, breaks are to be avoided. However, a team can prefer to have two or more consecutive away games if its stadium is located far from the opponent's venues, and the venues of these opponents are close to each other. A series of consecutive away games is called an *away tour*.

In a *round robin tournament* each team plays against each other team a fixed number of times. Most sports leagues play a double round robin tournament (*2RR*), where the teams meet twice (once at home, once away), but quadruple round robin tournaments (*4RR*) are also quite common. A *mirrored* double round robin tournament (*M2RR*) is a tournament where every team plays against every other team once in the first $n - 1$ rounds, followed by the same games with reversed venues in the last $n - 1$ rounds.

TABLE I. A DOUBLE ROUND ROBIN TOURNAMENT WITH SIX TEAMS

R1	**R2**	**R3**	**R4**	**R5**
1 – 6 2 – 5 4 – 3	3 – 1 5 – 4 6 – 2	1 – 5 2 – 4 3 – 6	2 – 1 5 – 3 6 – 4	1 – 4 3 – 2 6 – 5
R6	**R7**	**R8**	**R9**	**R10**
4 – 2 5 – 1 6 – 3	1 – 2 3 – 5 4 – 6	3 – 4 5 – 2 6 – 1	1 – 3 2 – 6 4 – 5	2 – 3 4 – 1 5 – 6

Table I shows an example of a compact double round robin tournament with six teams. The schedule has no breaks for team 1, three-in-a-row home games for team 6 and a four-game away tour for team 4.

Sports scheduling involves three main problems. First, the problem of finding a schedule with the *minimum number of breaks* is the easiest one. De Werra [8] has presented an efficient algorithm to compute a minimum break schedule for a *1RR*. If n is even, it is always possible to construct a schedule with $n - 2$ breaks. For an *M2RR*, it is always possible to construct a schedule with exactly $3n - 6$ breaks.

Second, the problem of finding a schedule that *minimizes the travel distances* is called the Traveling Tournament Problem (TTP) [9]. In TTP the teams do not return home after each away game but instead travel from one away game to the next. However, excessively long away trips as well as home stands should be avoided. The TTP is recently shown to be strongly NP-complete [10].

Third, most professional sports leagues introduce many additional requirements in addition to minimizing breaks and travel distances. We call the problem of finding a

Manuscript received October 12, 2013.

Kimmo Nurmi, Jari Kyngäs and Nico Kyngäs are with the Satakunta University of Applied Sciences, Pori, Finland (phone: +358 44 710 3371, e-mail: cimmo.nurmi@samk.fi).

Dries Goossens is with the Faculty of Economics and Business Administration, Ghent University, Belgium (e-mail: dries.goossens@ugent.be).

schedule which *satisfies given constraints* [1] the Constrained Sports Scheduling Problem (CSSP). The goal is to find a feasible solution that is the most acceptable for the sports league owner - that is, a solution that has no hard constraint violations and that minimizes the weighted sum of the soft constraint violations.

Scheduling the Finnish major ice hockey league is an example of a CSSP. It is very important to minimize the number of breaks. The fans do not like long periods without home games, consecutive home games reduce gate receipts and long sequences of home or away games might influence the team's current position in the tournament. It is also very important to minimize the travel distances. Some of the teams do not return home after each away game but instead travel from one away game to the next. There are also around a dozen more other criteria that must be optimized.

Section II presents the format played in the Finnish major ice hockey league in the 2013-2014 season. The section also introduces the requirements, requests and other constraints that the format implies. In Section III we describe the PEAST algorithm which has been used since the 2008-2009 season to schedule the league. Section IV reports some statistical findings and our computational results especially for the 2013-2014 season.

We are not aware of any sports scheduling papers dealing with such a broad class of constraints that arise in the Finnish major ice hockey league. We believe that the model and the solution method help sports scheduling researchers to evaluate, compare and exchange their equivalent ideas.

II. The Finnish major ice hockey league format and the constraint model

Ice hockey is the biggest sport in Finland, both in terms of revenue and the number of spectators. The spectator average per game for the current season (2012-2013) is about 5200. In the Saturday rounds one percent of the Finnish population (age 15-70) attended the games in the ice hockey arenas.

The Finnish major ice hockey league has 14 teams (see Table II). Seven of the teams in the league are located in big cities (over 100,000 citizens) and the rest in smaller cities. One team is quite a long way up north, two are located in the east and the rest in the south (see Figure 1).

TABLE II. The fourteen teams in the Finnish major ice hockey league and their number of titles

#1	Jokerit	6	#8	Ilves	16
#2	HIFK	7	#9	HPK	1
#3	Blues	0	#10	JYP	2
#4	TPS	11	#11	Pelicans	0
#5	Ässät	2	#12	SaiPa	0
#6	Lukko	1	#13	KalPa	0
#7	Tappara	15	#14	Kärpät	5

The format played in the league since the 2012-2013 season is somewhat eccentric. The competition starts with a regular season in September and ends with the playoffs from mid-March to mid-April. The league fixes the dates on which the games can be played. The last team of the regular season plays best out of seven elimination games against the best team of the Finnish 1st division ice hockey league. The six best teams of the regular season proceed directly to quarter-finals. Teams placing between 7th and 10th play preliminary playoffs best out of three. The two winners take the last two quarter-final slots. Teams are paired up for each playoff round according to the regular season standings, so that the highest-ranking team plays against the lowest-ranking, and so on. The playoffs are played best out of seven. The winner of the playoffs receives the Canada Bowl, the championship trophy of the League.

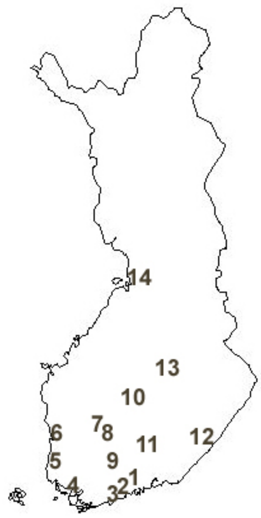

Fig. 1. The fourteen teams on the map of Finland

The basis of the regular season is a quadruple round robin tournament resulting in 52 games for each team. In addition, the teams are divided into two groups of seven teams in order to get a few more games to play. The teams in the groups are selected based on fairness, i.e. the strengths of the teams are most likely to be equal. These teams play a single round robin tournament resulting in 6 games. The home teams of the games are decided so that in two consecutive seasons each team has exactly one home game against every other team in the group.

Finally, the so-called "January leveling" adds two extra games for each team. In January, in the middle of the season, the last team on the current standings selects an opponent against which it plays once at home and once away on two consecutive days on Friday and on Saturday. The opponent selects the day for its home game. Then, the second last team (or the third last if the second last was selected by the last team) selects its opponent from the rest of teams and so on. The teams can choose to select their opponents either by maximizing the winning possibilities or by maximizing the ticket sales.

The quadruple round robin, six group games and two extra games per team total to 60 games for each team and 420 games overall. The format includes several other interesting features than those mentioned earlier. First, the standard game days are Tuesday, Friday and Saturday. The schedule should maximize the number of games on Fridays and on Saturdays in order to maximize the revenue. For the

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

same reason consecutive home games are not allowed on Fridays and Saturdays. Furthermore, due to the travel distances between some venues, certain combinations of a Friday home team playing a Saturday away game against the given team are not allowed. Table III shows the forbidden pairs. The games that cannot be scheduled on Fridays due to these restrictions, are played on Thursdays.

TABLE III. THE FORBIDDEN COMBINATIONS OF A FRIDAY HOME TEAM PLAYING A SATURDAY AWAY GAME AGAINST THE GIVEN TEAM

Ässät	KalPa, Kärpät, SaiPa
Blues	KalPa
HIFK	KalPa
HPK	KalPa
Ilves	Kärpät
Jokerit	KalPa
JYP	Kärpät, TPS
KalPa	Ässät Blues HIFK HPK Jokerit Lukko
Kärpät	Ässät Ilves Lukko SaiPa Tappara TPS
Lukko	KalPa Kärpät SaiPa
Pelicans	
SaiPa	Ässät Kärpät Lukko TPS
Tappara	Kärpät
TPS	KalPa Kärpät SaiPa

Second, every team should play in the two rounds before and the two rounds after New Year's Eve and adhere to the traveling rules given in the Table III. Third, the schedule should include a weekend when seven pairs of teams play against each other on consecutive rounds on Friday and on Saturday. These match-up games are called "back-to-back games". Fourth, local rivals (see Table IV) should play as many games as possible against each other in the first two rounds.

TABLE IV. THE LOCAL RIVALS

Jokerit	HIFK	Blues	
TPS	Ässät	Lukko	
Tappara	Ilves	HPK	Pelicans
JYP	SaiPa	KalPa	Kärpät

Fifth, the number of Friday and Saturday games between some local rivals (see Table V) should be maximized. Sixth, the Ässät, HPK, JYP, KalPa, Kärpät, Lukko, Pelicans, SaiPa and TPS teams should play at least one Friday or Saturday home game against the Jokerit and HIFK teams. These two teams guarantee the best revenue for the home team.

TABLE V. THE NUMBER OF GAMES THAT SHOULD BE PLAYED ON FRIDAYS AND SATURDAYS

Jokerit	HIFK	4	JYP	KalPa	4
Jokerit	Blues	3	Kärpät	KalPa	4
HIFK	Blues	4	SaiPa	Pelicans	5
Tappara	Ilves	5	HPK	Pelicans	4
Ässät	Lukko	5			

Seventh, the traveling distances between some of the venues require some teams to make away tours. That is, they should play away either on Tuesday and on Thursday or on Thursday and on Saturday. The teams that make away tours, their possible opponents and the minimum number of tours required are given in Table VI.

TABLE VI. THE TEAMS THAT MAKE AWAY TOURS, THEIR POSSIBLE OPPONENTS AND THE MINIMUM NUMBER OF TOURS REQUIRED

Kärpät	HPK+Ilves, TPS+Ässät, TPS+Ilves, Pelicans+Tappara, SaiPa+HIFK	6
KalPa	HPK+Blues, Ässät+Lukko, HIFK+Pelicans	4
SaiPa	Ässät+Lukko	2
Ässät	SaiPa+KalPa	2
Lukko	SaiPa+KalPa	2

Eighth, the Tappara and Ilves teams cannot play at home on the same day because they share a venue. Also the Jokerit and HIFK teams cannot play at home on the same day because they share the same (businessmen) spectators. Ninth, some of the teams cannot play at home in certain days because their venues are in use for some other event. A total of 69 home game restrictions existed in the 2012-2013 season. Finally, in the last two rounds each team should play exactly one home game.

Next, we present the constraint model of the Finnish major ice hockey league scheduling problem for the 2013-2014 season. Nurmi et al. [1] present a collection of typical constraints that are representative of many scheduling scenarios in sports scheduling. In the following problem description, we refer to this constraint classification. The detailed problem file will be found on the sports scheduling web site [11]. The objective is to find a feasible solution that is the most acceptable for the sports league owner. That is, a solution that has no hard constraint violations and that minimizes the weighted sum of the soft constraint violations.

The hard constraints are the following:

- C01. There are at most *90* rounds available for the tournament
- C04. Team t cannot play at home in round r (43 cases)
- C07. The *Tappara and Ilves* teams and the *Jokerit and HIFK* teams cannot play at home in the same round
- C08. A team cannot play at home on two consecutive calendar days
- C12. A break cannot occur in the *second* and *last* round
- C41. The schedule should include *one* weekend where *seven* pairs of teams play against each other on consecutive rounds on *Friday* and on *Saturday*.

The soft constraints are the following:

- C09. Team t wants to play at least m_1 away tours (see Table VI)
- C13. Teams cannot have more than *two* consecutive home games
- C14. Teams cannot have more than *two* consecutive away games

C15. The total number of breaks must not be larger than *140*

C19. There must be at least *five* rounds between two games with the same opponents

C22. Two teams cannot play against each other in series of HHAA, AAHH, HAAAH or AHHHA

C23. Team t wishes to play at least m_1 and at most m_2 home games on $weekday_1$, m_3-m_4 on $weekday_2$ and so on (see [11])

C26. The difference between the number of played home and away games for each team must not be larger than *two* in any stage of the tournament

C27. The difference in the number of played home games between the teams must not be larger than *two* in any stage of the tournament

C37. A team t_1 cannot play away against team t_2 if it played at home against team t_2 on the previous round, and the two rounds are on consecutive calendar days (see Table III). Note that this constraint is also used for the two rounds before and two rounds after New Year's Eve.

C38. Teams in the *first two* groups should play *two* games against each other and teams in the *last two* groups should play *four* games against each other between rounds *one* and *two* (see Table IV).

C39. At least m games between teams t_1 and t_2 should be played on *Fridays* and *Saturdays* (see Table V) (10)

C40. The schedule should include at least *200* games played either on *Friday* or on *Saturday* (10)

Some of the given soft constraints are actually goals. These goals are presented as exact numbers in the constraint model:

– minimize the number of breaks (C15)
– the defined local rivals should play as many games as possible in the first two rounds (C38)
– the number of Friday and Saturday games between some local rivals should be maximized (C39)
– the schedule should maximize the number of games on Fridays and on Saturdays (C40).

III. The PEAST algorithm

This section describes the PEAST algorithm which is used to schedule the league. The usefulness of an algorithm depends on several criteria. The two most important ones are the quality of the generated solutions and the algorithmic power of the algorithm. Other important criteria include flexibility, extensibility and learning capabilities. We can steadily note that the PEAST algorithm realizes these criteria. It has been used to solve several real-world scheduling problems (see eg. [12-15] and it is in industrial use. In this section we present the components of the algorithm.

The PEAST algorithm is a population-based local search method. The heart of the algorithm is the local search operator called GHCM (greedy hill-climbing mutation). The GHCM operator is used to explore promising areas in the search space to find local optimum solutions. Another important feature of the algorithm is the use of shuffling operators. They assist in escaping from local optima in a systematic way. Furthermore, simulated annealing and tabu search are used to avoid staying stuck in promising search areas too long. We next discuss these and other important characteristics briefly. For the detailed discussion we refer to [16]. The pseudo-code of the algorithm is given in Figure 2.

```
Set the iteration limit t, cloning interval c, shuffling interval s, ADAGEN
update interval a and the population size n
Generate a random initial population of schedules S_i for 1 <= i <= n
Set best_sol = null, round = 1
WHILE round ≤ t
  index = 1
  WHILE index++ <= n
    Apply GHCM to schedule S_index to get a new schedule
    IF Cost(S_index) < Cost(best_sol) THEN Set best_sol = S_index
  END REPEAT
  Update simulated annealing framework
  IF round ≡ 0 (mod a) THEN Update the ADAGEN framework
  IF round ≡ 0 (mod s) THEN Apply shuffling operators
  IF round ≡ 0 (mod c) THEN Replace the worst schedule with the
                            best one
  Set round = round + 1
END WHILE
Output best_sol
```

Fig. 2. The pseudo-code of the PEAST algorithm.

The GHCM operator is based on similar ideas to the Lin-Kernighan procedures [17] and ejection chains [18]. The basic hill-climbing step is extended to generate a sequence of moves in one step, leading from one solution candidate to another. The GHCM operator moves an object, o_1, from its old position, p_1, to a new position, p_2, and then moves another object, o_2, from position p_2 to a new position, p_3, and so on, ending up with a sequence of moves.

Picture the positions as cells as shown in Figure 3. The initial object is selected by tournament selection with $k = 7$. In the (deterministic) tournament selection we randomly pick k objects and then we choose the best one. The cell that receives the object is selected by considering all the possible cells and selecting the one that causes the least increase in the objective function when only considering the relocation cost. Then, another object from that cell is selected by considering all the objects in that cell and picking the one for which the removal causes the biggest decrease in the objective function when only considering the removal cost. Next, a new cell for that object is selected, and so on. The sequence of moves stops if the last move causes an increase in the objective function value and if the value is larger than that of the previous non-improving move, or if the maximum number of moves is reached. Then, a new sequence of moves is started. The maximum number of moves in the sequence is 10.

It may sound surprising that the best way to select the new cell for the object is to consider all possible cells and select the best one. Moreover, the best way to select a new object from that cell is again to consider all the objects in that period. Very often a not so greedy strategy ends up with better results.

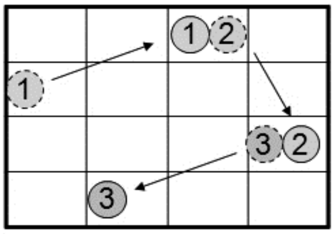

Fig. 3. A sequence of moves in the GHCM heuristic.

We improve the GHCM operator by introducing a tabu list which prevents reverse order moves in the same sequence of moves. i.e. if we move an object *o* from position p_1 to position p_2, we do not allow *o* to be moved back to position p_1 before a new sequence of moves begins.

We use a simulated annealing refinement to decide whether or not to commit to a sequence of moves in the GHCM operator. This refinement is different from the standard simulated annealing. It is used in a three-fold manner. Firstly, when choosing an object to be moved from a cell, a random object is chosen with probability $\exp(-1/T_k)$ instead of choosing the least fit object. Secondly, when choosing the cell where to move the object, a random cell is chosen with probability $\exp(-1/T_k)$ instead of choosing the fittest cell. Lastly, when the sequence of moves is cut short (i.e. a worsening move is made, and it worsens the solution more than the previous worsening move did), the whole sequence will still be committed with probability $\exp(-\text{costDiff}/T_k)$ instead of rolling back to the best position (i.e. the position at which the objective function value is the lowest) of the sequence. The cooling scheme T_k can be found in [16].

For most PEAST applications we introduce a number of shuffling operators – simple heuristics used to perturb a solution into a potentially worse solution in order to escape from local optima – that are called upon according to some rule. The idea of shuffling is the same as in hyperheuristics [19] but the other way around. A hyperheuristic is a mechanism that chooses a heuristic from a set of simple heuristics, applies it to the current solution to get a better solution, then chooses another heuristic and applies it, and continues this iterative cycle until the termination criterion is satisfied. We introduce a number of simple heuristics that are used to worsen the current solution instead of improving it.

In sports league scheduling we use five shuffling operations:

1. Select a random game and move it to a random round, and do this k_1 times.
2. Swap two random games, and do this k_2 times.
3. Select a random round and move k_3 random games from that round to random rounds.
4. Swap all the games in two random rounds.
5. Select a random game A-B and swap it with the game B-A, and do this k_4 times.

The best results have been obtained using the values $k_1 = 3$, $k_2 = 2$, $k_3 = 3$ and $k_4 = 2$.

No crossover operators are applied to the population of schedules. Every *c* iterations the least fit individual is replaced with a clone of the fittest individual. This operation is completely irrespective of the globally fittest schedule (*best_sol* in Fig. 1) found. The PEAST algorithm uses ADAGEN, the adaptive genetic penalty method introduced in [20]. A traditional penalty method assigns positive weights (penalties) to the soft constraints and sums the violation scores to the hard constraint values to get a single value to be optimized. The ADAGEN method assigns dynamic weights to the hard constraints based on the constant weights assigned to the soft constraints. The soft constraints are assigned fixed weights according to their significance. The hard constraint weights are updated every *k*th generation using the method given in [11].

Random initial solutions work best in all the real-world cases where we have applied PEAST algorithm (see eg. [12-15]). We have found no evidence that a sophisticated initial solution improves results. On the contrary, random initial solutions seem to yield superior or at least as good results.

IV. Computational Results

We believe that scheduling the Finnish major ice hockey league is one of the most difficult sports scheduling problems because it combines break minimization and traveling distance restrictions with dozens of constraints that must be satisfied. We have used the PEAST algorithm and its predecessors to schedule the league since the 2008-2009 season. This section reports our computational results for the 2013-2014 season. We start with some interesting statistical findings from the earlier seasons.

In addition to scheduling the league we also contribute to the process of improving the league format. For example the "January leveling" and the "back-to-back games" have been introduced to the format based on our ideas. Table VII shows the increase in the number of spectators in the last four (five) seasons.

TABLE VII. The Number of Spectators in the Finnish Major Ice Hockey League in the Last Seven Seasons

2005-2006	1 958 843
2006-2007	1 943 312
2007-2008	1 964 626
2008-2009	1 997 114
2009-2010	2 015 080
2010-2011	2 036 915
2011-2012	2 145 462 (avg 5108)
2012-2013	2 189 350 (avg 5213)

The standard game days used to be Tuesday, Thursday and Saturday. From the 2011-2012 season the league decided to change Thursdays to Fridays to get more spectators. Friday games have had about 10% more spectators.

However, playing at home both on Fridays and on

Saturdays is not allowed. Due to this, the games that cannot be scheduled on Fridays are played on Thursdays. This on the other hand means that some teams play two consecutive games and some teams have a rest day before the Saturday game. In the last ten seasons the probability for a home team to defeat an away team that has had a rest day is 10% smaller. Likewise, the probability for an away team to win a home team that has had a rest day is even 85% smaller.

Minimizing the number of breaks is very important because it is likely that having two consecutive home games on Thursday and on Saturday decreases the number of spectators. In the last ten seasons the number of spectators on Thursday has decreased by 3.5% and on Saturday by 1.9%.

Some teams desire away tours because of the traveling distances between their venue and some of the opponents' venues (see Table VI). For example, the Kärpät team wants to make at least six away tours. In the last ten seasons the probability for the team to win its second away game is 30% smaller than to win any away game.

The process of scheduling the league takes about two months. First, we discuss the possible improvements to the format with the league's competition manager. Then, the format is accepted by the team CEOs. Next, all the restrictions, requirements and requests by the teams are gathered. Finally, the importance (penalty value) of the constraints is decided. We ran the algorithm for one week and choose the best solution. The algorithm was run on an Intel Xeon X5690 3.47GHz with 24GB of RAM running Windows 7 Professional.

TABLE VIII. THE BEST SOLUTION FOUND (ACCEPTED BY THE LEAGUE)

C01	Hard	There are at most 90 rounds available for the tournament	0
C04	Hard	A team cannot play at home in the given round (43 cases)	0
C07	Hard	Two pairs of teams cannot play at home in the same round (2 cases)	0
C08	Hard	A team cannot play at home on two consecutive calendar days	0
C12	Hard	A break cannot occur in the second and last rounds	0
C41	Hard	"Back-to-back games"	0
C09	Soft 10	Number of away tours not scheduled	0
C13	Soft 10	One violation for each case when a team has more than two consecutive home games	0
C14	Soft 3	One violation for each case when a team has more than two consecutive away games	0
C15	Soft 5	One violation for each break more than 140	0
C19	Soft 5	One violation for each round less than five	1
C22	Soft 1	One violation for each case when two teams meet in series of HHAA, AAHH, HAAAH or AHHHA	4
C23	Soft 1	One violation for each home game less or more than the requested number on given weekday	2
C26	Soft 1	One violation for each case when the difference is more than two	1
C27	Soft 3	One violation for each case when the difference is more than two	0
C37	Soft 4	One violation for each forbidden combination	0
C37	Soft 4	One violation for each forbidden combination (the compact rounds around New Year's Eve)	0
C38	Soft 10	One violation for each game less than 12	0
C39	Soft 10	One violation for each game not played on Fridays or on Saturdays	0
C40	Soft 10	One violation for each game less than 200	0

Recall that the objective is to find a feasible solution that is the most acceptable for the sports league owner. That is, a solution that has no hard constraint violations and that minimizes the weighted sum of the soft constraint violations. Table VIII shows the constraints used for the 2013-2014 season, whether they are decided to be hard or soft constraints, the importance (penalty value) of the soft constraints and how the constraint violations are calculated.

Table VIII also shows the best solution found. The solution has no hard constraint violations and the penalty value for the soft constraint violations is 12. The schedule has 89 rounds (C01), 139 breaks (C15) and 220 games played either on Friday or on Saturday (C40). This was by far the most difficult schedule to generate compared to the earlier seasons. The league accepted the schedule and it will be used in the 2013-2014 season.

It should be noted that for the 2000-2009 seasons the average number of 3-breaks at home (C13) was 14 and the average number of cases when there were less than five rounds between two games with the same opponents (C19) was 10. Furthermore, in these seasons the maximum difference between the number of played home and away games (C26) was 5 and the difference in the number of played home games between the teams was 4.

V. CONCLUSIONS

We presented the format played in the Finnish major ice hockey league in the 2013-2014 season. The format is very complicated requiring computational intelligence to generate an acceptable schedule. We presented computational results that show that our PEAST algorithm generated a good-quality schedule for the 2013-2014 season. The league owner accepted the schedule.

We note that during the last five years we have experienced and compared the performance of the PEAST algorithm to several other algorithms such as tabu search, simulated annealing, ant algorithms, genetic algorithms, hyper heuristics and variable neighborhood search. In our test runs the PEAST algorithm has clearly outperformed the other algorithms. Furthermore, it has shown its value in business use, especially in sports league scheduling and workforce scheduling. We have also compared the performance of the PEAST algorithm to CPLEX [21]. The instance used represented the first half of the 2011-2012 season (double round robin). CPLEX was not able to produce a feasible solution to the problem even though we used a very simplified version of the problem. We strongly believe that the PEAST algorithm clearly outperforms MIP solvers in solving highly constrained real-world sports scheduling problems

REFERENCES

[1] K. Nurmi, D. Goossens, T. Bartsch, F. Bonomo, D. Briskorn, G. Duran, J. Kyngäs, J. Marenco, CC. Ribeiro, FCR. Spieksma, S. Urrutia and R. Wolf-Yadlin, "A Framework for Scheduling Professional Sports Leagues", in Ao, Sio-Iong (ed.): IAENG Transactions on Engineering Technologies Volume 5, Springer, USA, 2010.

[2] K. Easton, G. Nemhauser and M. Trick, “Sports scheduling”, in Handbook of Scheduling, edited by Leung, Florida, USA: CRC Press, pp 52.1-52.19, 2004.

[3] J.H. Dinitz, D. Froncek, E.R. Lamken and W.D. Wallis, “Scheduling a tournament”, in Handbook of Combinatorial Designs, edited by Colbourn and Dinitz, Florida, USA: CRC Press, pp. 591-606, 2006

[4] A. Drexl and S. Knust, “Sports league scheduling: graph- and resource-based models”, Omega 35, pp. 465-471, 2007.

[5] P. Rasmussen and M. Trick, “Round robin scheduling - A survey”, European Journal of Operational Research 188, pp. 617-636, 2008.

[6] S. Knust, “Sports Scheduling Bibliography” [Online], Available: http://www.inf.uos.de/knust/sportssched/sportlit_class/, (Last update 26.11.2012).

[7] G. Kendall, S. Knust, C.C. Ribeiro and S. Urrutia, “Scheduling in Sports: An annotated bibliography”, Computers and Operations Research 37, pp. 1-19, 2010.

[8] D. de Werra, “Scheduling in sports”, in Studies on graphs and discrete programming, edited by Amsterdam and Hansen, pp. 381-395, 1981.

[9] K. Easton, G. Nemhauser, and M. Trick “The traveling tournament problem: description and benchmarks” in Proc of the 7th. International Conference on Principles and Practice of Constraint Programming, Paphos, pp. 580–584, 2001.

[10] C. Thielen and S. Westphal, "Complexity of the traveling tournament problem", Theoretical Computer Science 412 (4-5), pp. 345-351, 2011.

[11] K. Nurmi et. al., “Sports Scheduling Problem” [Online], Available: http://www.samk.fi/ssp, (Last update 7.3.2013).

[12] N. Kyngäs, K. Nurmi and J. Kyngäs, “Solving the person-based multitask shift generation problem with breaks”, in Proc. of the 5th International Conference On Modeling, Simulation And Applied Optimization (ICMSAO), Hammamet, Tunis, 2013.

[13] N. Kyngäs, K. Nurmi, E.I. Ásgeirsson and J. Kyngäs, ”Using the PEAST Algorithm to Roster Nurses in an Intensive-Care Unit in a Finnish Hospital”, in Proc. of the 9th Conference on the Practice and Theory of Automated Timetabling (PATAT), Son, Norway, 2012.

[14] N. Kyngäs, K. Nurmi and J. Kyngäs, “Optimizing Large-Scale Staff Rostering Instances”, Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists, Hong Kong, 2012.

[15] K. Nurmi and J. Kyngäs, ”A Conversion Scheme for Turning a Curriculum-based Timetabling Problem into a School Timetabling Problem” in Proc of the 7th Conference on the Practice and Theory of Automated Timetabling (PATAT), Montreal, Canada, 2008.

[16] N. Kyngäs, K. Nurmi and J. Kyngäs, “Crucial Components of the PEAST Algorithm in Solving Real-World Scheduling Problems”, in Proc of the 2nd International Conference on Software and Computer Applications, Paris, France, 2013.

[17] S. Lin and B.W. Kernighan, “An effective heuristic for the traveling salesman problem”, Operations Research 21, pp. 498–516, 1973.

[18] F. Glover, “New ejection chain and alternating path methods for traveling salesman problems”, in Computer Science and Operations Research: New Developments in Their Interfaces, edited by Sharda, Balci and Zenios, Elsevier, pp. 449–509, 1992.

[19] P.Cowling, G. Kendall and E. Soubeiga, “A hyperheuristic Approach to Scheduling a Sales Summit”, in Proc. of the 3rd International Conference on the Practice and Theory of Automated Timetabling (PATAT), pp. 176-190, 2000.

[20] K. Nurmi, “Genetic Algorithms for Timetabling and Traveling Salesman Problems”, Ph.D. dissertation, Dept. of Applied Math., University of Turku, Finland, 1998. Available: http://www.bit.spt.fi/cimmo.nurmi/

[21] N. Kyngäs, D. Goossens, K. Nurmi and J. Kyngäs, ”PEAST vs. CPLEX”, in Proc of the 2nd International Conference on the Theory and Practice of Natural Computing, Madrid, Spain, 2013.

ISBN: 978-988-19253-3-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)