Journal of Engineering and Applied Sciences 8 (2): 58-63, 2013
ISSN: 1816-949X

Simulated Annealing with Dynamic Initial Temperatures for University Course Timetable Problem

[1]Hassan Y. Tarawneh, [1]Masri Ayob and [2]Zulkifli Ahmad
[1]Data Mining and Optimization Research Group (DMO), Center of Artificial Intelligence Technology, Faculty of Information Science and Technology,
[2]School of Linguistics and Language Studies, Faculty of Social Sciences and Humanities, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia

Abstract: This research proposes a dynamic initial temperature for Simulated Annealing (SA) to solve a problem of curriculum-based course timetabling. Initial temperature setting is an important factor that affects the performance of the SA where very high initial temperature will lead SA to accept any solution whilst the lower value leads SA to quickly trap in local optima which behaves as a descent heuristic. Unfortunately, different initial temperature is required for each instance to ensure that SA can perform well. Therefore, researchers propose a dynamic mechanism to initialize the initial temperatures according to some solutions for each instance. Given the feasible initial solution, the SA starts several iterations and calculates the deviations average where the deviation equals the difference between the current objective value and the new one. Using this average, the mechanism will decide the initial moderate temperature according to the SA acceptance criterion ratio that we examine in the extermination. A computational result shows the effectiveness of the proposed mechanism to dynamically initialize the initial temperature compared with the fixed initial temperatures.

Key words: Simulated annealing, local search, initial temperature, course timetabling problem, meta-heuristics

INTRODUCTION

Timetabling is defined as the allocation, subject to constraints of given resources to objects being placed in space-time in such a way as to satisfy as nearly as possible a set of desirable objectives (Wren, 1996). Many researchers have focused on solving this problem using many algorithms, such as ant colony optimization (Eley, 2007), evolutionary search (Beligiannis *et al.*, 2008) and Simulated Annealing (SA) (Aycan and Ayav, 2009). This research focuses on solving university course timetable problem that involves scheduling a set of courses within a given number of rooms and time periods using Simulated Annealing algorithm (SA).

SA is one of the most popular meta-heuristic algorithm that have been used to solve many kinds of computational optimization problem including job scheduling (Kolonko, 1999), course timetabling (Pongcharoen *et al.*, 2008), examination timetabling (Azimi, 2005), communication systems (Salcedo-Sanz *et al.*, 2004) and travelling salesman (Cerny, 1985).

Simulated annealing algorithm has a strategy to escape from local minima by accepting worse solution using probability acceptance criteria. However, SA could be trapped into local optimum which may consume longer to find good solutions (Xinchao, 2011).

To avoid all these drawbacks, many researchers have attempted to improve the simulated annealing performance by using for example, Adaptive Simulated Annealing (ASA) (Ingber, 1996) or hybridize SA with another heuristics, such as genetic algorithm (Cordon *et al.*, 2002).

The intention of this research is to improve the SA performance by choosing good initial temperature. Many researchers presented ideas to solve the problem of satisfying initial temperature. For example, Poupaert and Deville (2000) proposed a rule of choosing initial temperature based on the initial acceptance ratio χ_0 which is defined as the number of the bad transition that accepted divided by the number of attempted bad transitions and on the average increase in objective function value. On the other hand, Zhang *et al.* (2010) determined the initial temperature by starting with very high temperature and then tried to derive the real start temperature by using the functional dependence between

Corresponding Author: Masri Ayob, Data Mining and Optimization Research Group (DMO), Center of Artificial Intelligence Technology, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia

Table 1: The differences between the dynamic initial temperature with others from the literature

Techniques	First initial temperature	Average (Δ)	Applying part	Advantages	Disadvantages
The technique	Dynamic	$\lvert\bar{\Delta}\rvert$	First LL iterations	Avoid the first high temperature	Need good chosen cooling schedule
Simulated annealing with estimated temperature (Poupaert and Deville, 2000)	Infinite	$\lvert\bar{\Delta}\rvert$	All (for each solution neighbors)	Estimate the temperature for all iterations, the temperature is not control parameter but the acceptance probability	Could trap in zigzag walk during the temperature estimation
Solving the course scheduling problem using simulate annealing (Aycan and Ayav, 2009)	10000	$\lvert\bar{\Delta}\rvert$	Before the algorithm start (one time)	Try to choose a good first initial temperature	The initial temperature still high

the starting acceptance probability χ_0 (0.95 or 0.9) and the temperature T_0, using Eq. 1 and 2 (Poupaert and Deville, 2000):

$$\chi_0 = \chi\left(\left\{\delta_1, \delta_n, \delta_{(n+1)}, \ldots, \delta_m\right\}\right), T_{(0)} \tag{1}$$

$$\chi_0 = \frac{1}{m}\sum_{i=1}^{n}\exp\left(-\frac{\delta_i}{T_0}\right) + \frac{m-n}{m} \tag{2}$$

Where:

χ_0 = The starting acceptance probability between (60-80%)
T_0 = The starting temperature
δ_i = $f(s_i)$-$f(s_0)$
s_0 = The initial solution
s_i = The new or neighbor solution of s_0
f = The objective function for each solution
m = The neighbor solution space size

The idea of previous equations which proposed and presented by Poupaert and Deville (2000) and Zhang *et al*. (2010) still can initialize high temperatures because it creates the initial temperature from the first iteration using the first Δ.

This mechanism try to avoid the problem by choosing good suitable initial moderate for each instance independently, by applying several iterations and calculate the differences between the current solution and the new one even it worse or not and calculate the average to decide the initial temperature according to the acceptance criteria ratio. Table 1 shows a comparison between the initial temperature mechanism and others in literature.

CURRICULUM-BASED COURSE TIMETABLING

The curriculum-based course timetabling problem for the ITC-2007 consists of scheduling all lectures of a set of courses into a weekly timetable where each lecture of a course must be assigned a period and a room in accordance to a given set of constraints which satisfies the hard constraints and minimizes the soft constraints. The four hard constraints (H1-H4) and four soft constraints (S1-S4) are defined as follows (Gaspero *et al*., 2007):

- H1: All lectures of a course must be scheduled to a distinct periods
- H2: Two lectures cannot be assigned in the same period and the same room
- H3: Lectures of courses in the same curriculum or taught by the same teacher cannot be scheduled in the same period
- H4: If the teacher of a course is not available at a given period then no lectures of the course can be assigned to that period
- S1: For each lecture, the number of students attending the course should not be greater than the capacity of the room hosting the lecture. Each student above the capacity counts as 1 point of penalty
- S2: All lectures of a course should be scheduled at the same room. If this is impossible, the number of occupied rooms should be as few as possible. Each distinct room used for the lectures of a course but the first, counts as 1 point of penalty
- S3: The lectures of a course should be spread into the given minimum number of days. Each day below the minimum counts as 5 points of penalty
- S4: For a given curriculum a violation is counted if there is one lecture not adjacent to any other lecture belonging to the same curriculum within the same day which means the agenda of students should be as compact as possible. Each isolated lecture in a curriculum counts as 2 points of penalty

The hard constraints (H1-H4) must be satisfied to obtain a feasible solution. However, soft constraint can be violated if necessary. The quality of the timetable (penalty cost) is calculated by summing all violations of soft constraints (S1-S4).

THE SIMULATED ANNEALING

Initial solution phase: This research starts by generating the initial solution using sequential greedy heuristic as by Lu and Hao (2010). There is no proof that this greedy heuristic guarantees to find feasible solution (Lu and Hao, 2010). So, researchers will use steepest decent heuristic to rectify the solution until they get the feasible solution.

Table 2: Percentages of SA acceptance criterion among different temperatures and penalty costs (best result out of 10 runs for each range)

	The initial temperatures									
$\|\Delta f\|$	10^{10} (%)	10^{5} (%)	10000	5000 (%)	2000	1000 (%)	500	100 (%)	10 (%)	1
150	100	100	99.0	95.8	94.3	85.5	71.5 %	27.4	0.0	0.0
50	100	100	99.0	98.9	97.6	95.5	88.3	28.9	0.0	0.0
5	100	100	99.9	99.9	99.9	99.9	97.5	94.9	70.5	0.0

Table 3: The initial temperature selection

Deviation average (γ)	Initial temperature (T_0)
≥50	Random (1000, 2000)
5-49	Random (500, 1000)
1-4	Random (100, 500)

Simulated annealing phase: Simulated annealing optimizes the given solution using probability accepting criterion as a mechanism to escape from local optimum by accepting worse solutions. SA algorithm mechanism accepts neighbor solution, s* when it penalty cost is lower than or equal to the current one s. There is a possibility to accept neighbor solution that has higher penalty cost using probability acceptance criterion:

$$p(x) = e^{-\frac{\Delta f}{T_i}}$$

Where, Δf = f(s*)-f(s) with f(s) and f(s*) is the penalty costs of solution s and s*, respectively and T_i the current temperature. T_i is reduced according to a cooling schedule with given cooling rate, α for each iteration or level until this temperature reaches final minimum temperature closed to zero T_{min}.

The SA starts with initial feasible solution generated by the constructive part. Then, the neighborhood structures (simple move and simple swap) will be applied to generate several feasible neighbors to the current solution. The best neighbor solution s* among them will be accepted if the cost is better than or equal to the current one. Otherwise, s* will be accepted based on the probability p(x) or will be accepted if:

$$p(x) < e^{-\frac{\Delta f}{T_i}}$$

Where, p(x) is a random number between 0 and 1. The next part explains the main contribution to answer the following question: How to choose the best initial temperature?

The dynamic initial temperature: Normally when the temperature is very high, the range of accepting worse solutions is very high as well. Let say, researchers have a current solution (s) with penalty cost f(s) equals 2000 and neighbor solutions (s*) with f(s*) equals 2150, 2050 and 2005. The question is: What percentage will SA accept those worse solutions when the temperature is very high, very low and medium?

The high temperatures will let SA to accept any worse solutions. Therefore, there is no point using simulated annealing when the temperature is very high. Therefore, researchers propose a new technique (a dynamic initial temperature SA (SA-D)) to avoid this problem.

In order to estimate the suitable range of accepting worse solution at early stage, researchers perform a preliminary experiment (Table 2 and 3). In this research for several iterations, SA decides the initial temperature amount according to the penalty values deviation average γ (Eq. 3 and 4):

$$Av = \sum_{i=1}^{n} |\Delta f| \tag{3}$$

Where, Δf = f(s*)-f(s) and n is the total current iterations:

$$\gamma = \frac{Av}{n} \tag{4}$$

Using this technique, researchers will avoid the wasting computation time by choosing good initial temperature.

Let initial solution ($s_0 1$); initial cost f($s_0 1$); set best solution ($s_0 1^* \leftarrow s_0 1$); the current temperature (T_k); minimum temperature (T_{min}); dynamic initial temperature (T_0); initial dynamic temperature iterations (k). The SA with dynamic initial temperature pseudo-code is presented in Fig. 1.

The SA algorithm that is applied involves: Neighborhood structure, temperature, cooling schedule and aspiration criteria.

SA parameters conclude; temperature, plateau length (L_i) where the temperature will decrease using cooling schedule ($T_i = T_{i-1} * \alpha$) where α = 0.99 and stopping criterion. In this research, the algorithm stops in three cases or at least reach one of them when the minimum temperature (T_{min}) closed to zero (frozen stage) (T_{min} = 0.0001), number of iterations or CB-CTT problem ITC 2007 timeout condition. For the neighborhood structures, researchers apply simple move and simple swap.

```
Set i = 1;
Do while (termination criteria does not met)
        Apply neighborhood structures (N1, N2) for the Sol and calculate the best costs function among
  them (NSol)
      Δf = f(NSol) - f(Sol)
          T_Δf = |Δf| + T_Δf
          γ = T_Δf / i
   if i<k then
     Select T_0 according to Table 3
    end if
   If Δf ≤ 0
          Sol = NSol
          Sol* = NSol
  Else
     Create e random number called RAND between [0, 1];
    if (e^(-Δf/T) > RAND then
       Sol = NSol
      end if
   end if
   Apply the cooling schedule to update the temperature;
```

Fig. 1: A pseudo-code for SA with dynamic initial temperature (SA-D)

Simple move: Move one lecture of course c from the current period to another free position period; the size of this neighborhood is five.

Simple swap: Swap one lecture chosen randomly by another one random lecturer that belongs to two different courses, periods and rooms, the size of this neighborhood is five.

EXPERIMENT AND RESULT

In this study, researchers test their research in 21 competition instances from The Second International Timetabling Competition track 3: Curriculum-based course timetabling (<http://www.cs.qub.ac.uk/itc2007/>).

The main purpose for this experimentation is to compare the proposed mechanism to initialize the initial temperature with the fixed one. Researchers compare the temperature with initial temperatures applied by previous researchers (Table 4). For this comparison, the cooling rate (α = 0.99), the number of iterations equal 40,000 iterations for each run and for each temperature will use different plateau length (L_k) where the temperature decreases, in order to arrange the decrement among them.

Table 5 summarize the comparison results to the proposed dynamic initial temperatures and others fixed initial temperatures (Aycan and Ayav, 2009; Goffe *et al*., 1994). The first column indicates the instances, columns 2-6, 7-11 and 12-16 reports the best solution, mean, median, standard deviation and the best solution time in seconds over 30 runs to each instance. For these experimentations, researchers apply the standard simulated annealing (Eglese, 1990).

Table 4: List of initial temperatures used by other researchers

Researchers	Intial temperature
Aycan and Ayav (2009)	10000
Goffe *et al*. (1994)	10^7

In Table 5 that the high temperature takes more time to achieve the solution than the dynamic initial temperature achieved before. The average solutions costs over 21 runs showed that the temperature leads the search to reasonable results and outperforms the other temperatures.

For instances in Comp 1 and 11, the optimal solution is found by using the dynamic initial temperature (run under the ITC 2007 timeout condition) where the other results are comparable to other approaches (perhaps it needs more computation time).

Table 6 summarizes the comparison results between the proposed research with the best known results under ITC 2007 timeout condition. The first column indicates the instances, columns 2-3 report the penalty cost of this research and the best known. Therefore, one can observe that the SA-D best results are quite comparable to the best known results.

Table 6 showed that this research reached the best optimal results in Comp 1 and 11 while the other instances results is found to be quite far from the best foundsolution. However, the aim is just to show that the proposed dynamic initial temperature is better than the static initial temperature.

Table 5: Comparison of the best, average results and average runs time of sa performance using the dynamic initial temperature and the initial temperature value proposed by Aycan and Ayav (2009) and Goffe *et al.* (1994)

	Goffe *et al.* (1994)					Aycan and Ayav (2009)					The dynamic initial temperature				
Instance	Best	Mean	Median	STD	Time (sec)	Best	Mean	Median	STD	Time (sec)	Best	Mean	Median	STD	Time (sec)
Comp 1	7	9.00	8	2.03	282	5	8.32	8	2.06	280	5	5.0000	5.0	0.00	230
Comp 2	172	185.37	188	7.04	415	171	180.89	182	16.61	413	160	169.6600	167.5	7.79	400
Comp 3	119	133.00	134	9.71	436	119	134.21	136	9.08	434	112	128.2500	129.0	8.23	423
Comp 4	72	83.47	84	6.76	445	73	81.74	85	18.85	448	67	79.0000	79.5	8.00	443
Comp 5	334	347.84	349	7.20	413	330	349.11	351	8.03	412	318	344.2500	346.0	14.78	403
Comp 6	96	115.68	120	11.32	405	97	113.59	110	11.47	398	87	113.8460	116.0	16.29	378
Comp 7	56	77.58	83	13.57	431	53	72.00	73	11.94	427	42	55.5960	55.5	7.37	421
Comp 8	67	80.68	80	8.46	353	65	75.95	75	7.71	343	65	78.5660	78.0	6.73	323
Comp 9	152	174.89	179	10.20	465	153	164.16	164	8.08	460	150	165.6600	166.0	8.77	456
Comp 10	48	62.11	57	10.88	454	46	58.79	58	7.15	454	34	47.1566	47.5	6.58	448
Comp 11	1	1.26	1	1.28	230	0	1.32	1	0.95	226	0	0.0000	0.0	0.00	213
Comp 12	484	494.84	492	8.72	458	488	493.69	493	6.58	459	460	473.8100	474.0	8.42	454
Comp 13	107	127.58	132	9.48	441	108	125.26	126	8.05	440	100	122.4400	125.5	11.55	421
Comp 14	90	119.53	124	15.26	368	89	106.32	103	10.12	368	89	106.1600	106.5	9.29	366
Comp 15	133	146.37	149	9.07	444	132	141.68	142	5.72	442	121	131.1300	131.0	5.96	432
Comp 16	76	85.84	89	6.57	402	77	86.16	86	5.54	378	69	77.4100	77.5	6.13	367
Comp 17	112	124.05	124	8.22	380	112	123.74	126	8.19	375	137	124.9100	127.5	10.90	366
Comp 18	118	130.47	132	9.04	464	117	136.63	138	10.68	463	152	131.3100	132.5	12.78	450
Comp 19	109	127.21	127	11.11	468	109	129.84	132	10.73	464	110	113.1600	114.5	7.86	450
Comp 20	67	80.95	79	9.37	460	69	80.05	79	10.33	466	82	79.8800	81.0	6.16	454
Comp 21	139	148.47	146	6.64	404	132	147.16	148	7.71	370	152	142.8700	142.5	8.40	367

All the values are the penalty cost

Table 6: Best results of SA with dynamic initial temperature (SA-D) compared to best known results on curriculum-based course timetabling ITC 2007 Track 3

Instance	SA-D	Results	Methods
Comp 1	5	5	Tabu search
Comp 2	160	24	SAT-Modulo-Theory
Comp 3	112	66	Local search
Comp 4	67	35	Local search
Comp 5	318	290	Simulated annealing
Comp 6	87	27	SAT-Modulo-Theory
Comp 7	42	6	SAT-Modulo-Theory
Comp 8	65	37	Other
Comp 9	150	96	Tabu search
Comp 10	34	4	SAT-Modulo-Theory
Comp 11	0	0	Tabu search
Comp 12	460	300	Simulated annealing
Comp 13	100	59	Tabu search
Comp 14	89	51	Mathematical programming
Comp 15	121	66	Tabu search
Comp 16	69	18	SAT-Modulo-Theory
Comp 17	137	56	SAT-Modulo-Theory
Comp 18	152	62	Hybrid method
Comp 19	110	57	Local search
Comp 20	82	4	SAT-Modulo-Theory
Comp 21	152	75	Simulated annealing

Figure 2 and 3 show the behavior of SA-D compared to SA that use a static initial temperature (for Comp 1 and 4 instances).

Figure 2 and 3 showed that using high temperatures causes the search to accept worse solutions which are very far from the current solutions which will consume longer processing time. Although, the proposed mechanism still leads the search to accept worse solutions but it is not very far from the current solutions. Therefore, the processing time will be shorter and faster.

Based on the observed results in the comparison, researchers can conclude that the amount of the initial temperatures must be chosen carefully. The temperature must be moderate and within good range according to the search space improvement for each instance (problem) independently.

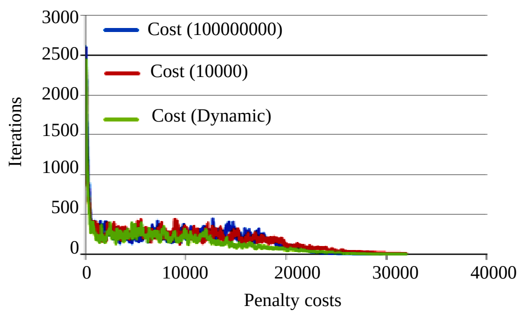

Fig. 2: Behavior of SA-D compared to SA that use a static initial temperature for Comp 1

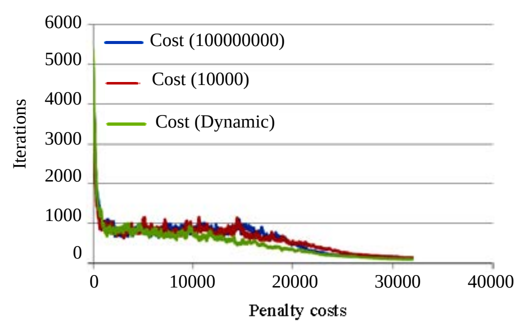

Fig. 3: Behavior of SA-D compared to SA that use a static initial temperature for Comp 4

CONCLUSION

This study proposed a dynamic initial temperature to simulated annealing algorithm for curriculum-based course timetabling problem; from the experimentation results reported in this study, researchers can attempt to draw several conclusions.

The proposed dynamic initial temperature can provide less computational time with good solution quality comparing with high and low fixed initial temperature with some enhancement to the cooling schedule the solutions can be better.

The search space for each instance is different, so the initial temperatures and cooling schedules should be chosen carefully. Using high temperatures in the small instance dataset will waste the computational time. On the other hand, low temperature to large instance will lead the search to get trapped very fast in local optimum.

RECOMMENDATIONS

In future research, researchers suggest applying median cooling schedule with the proposed dynamic initial temperature, in order to prevent the search from getting trapped in local optimum which may cause longer computational time. Therefore, good cooling schedule will make the temperature more worthy rather than using high or low temperatures with slow or fast cooling schedule.

ACKNOWLEDGEMENTS

The researchers wish to thank Ministry of Higher Education for supporting this research under the FRGS Research Grant Scheme (FRGS/1/2012/SG05/UKM/02/11).

REFERENCES

Aycan, E. and T. Ayav, 2009. Solving the course scheduling problem using simulate annealing. Proceedings of the IEEE International Advance Computing Conference, March 6-7, 2009, Patiala, pp: 462-466.

Azimi, Z.N., 2005. Hybrid heuristics for examination timetabling problem. Appl. Math. Comput., 163: 705-733.

Beligiannis, G.N., C.N. Moschopoulos, G.P. Kaperonis and S.D. Likothanassia, 2008. Applying evolutionary computation to the school timetabling problem: The Greek case. Comput. Oper. Res., 35: 1265-1280.

Cerny, V., 1985. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. J. Optimiz. Theo. Appli., 45: 41-51.

Cordon, O.F. Moya and C. Zarco, 2002. A new evolutionary algorithm combining simulated annealing and genetic programming for relevance feedback in fuzzy information retrieval systems. Soft Comput., 6: 308-319.

Eglese, R.W., 1990. Simulated annealing: A tool for operational research. Eur. J. Oper. Res., 46: 271-281.

Eley, E., 2007. Ant algorithms for the exam timetabling problem. Proceedings of the 6th International Conference on Practice and Theory of Automated Timetabling, August 30-September 1, 2006, Brno, Czech Republic, pp: 364-382.

Gaspero, L.D., B. McCollum and A. Schaerf, 2007. The second international timetabling competition (ITC-2007): Curriculum-based course timetabling (Track 3). Technical Report. http://www.cs.qub.ac.uk /itc2007/curriculmcourse/report/curriculumtechreport.pdf.

Goffe, W.L., G.D. Ferrier and J. Rogers, 1994. Global optimization of statistical functions with simulated annealing. J. Econ., 60: 65-99.

Ingber, L., 1996. Adaptive Simulated Annealing (ASA): Lessons learned. J. Control Cybernet., 25: 33-54.

Kolonko, M., 1999. Some new results on simulated annealing applied to the job shop scheduling problem. Eur. J. Operat. Res., 113: 123-136.

Lu, Z. and J.K. Hao, 2010. Adaptive tabu search for course timetabling. Eur. J. Oper. Res., 200: 235-244.

Pongcharoen, P., W. Promtet, P. Yenradee and C. Hicks, 2008. Stochastic optimisation timetabling tool for university course scheduling. J. Int. J Prod. Econ., 112: 903-918.

Poupaert, E. and Y. Deville, 2000. Simulated annealing with estimated temperature. AI Commun., 13: 19-26.

Salcedo-Sanz, S., R. Santiago-Mozos and C. Bousono-Calzon, 2004. A hybrid hopfield network-simulated annealing approach for frequency assignment in satellite communications systems. IEEE Trans. Syst. Man Cybernet. Part B: Cybernet., 34: 1108-1116.

Wren, A., 1996. Scheduling, Timetabling and rostering: A special relationship, the practice and theory of automated timetabling. Pract. Theor. Autom. Timetabling, 1153: 46-75.

Xinchao, Z., 2011. Simulated annealing algorithm with adaptive neighborhood. Applied Soft Comput., 11: 1827-1836.

Zhang, D., Y. Liu, R.M. Hallah and S.C.H. Leung, 2010. A simulated annealing with a new neighborhood structure based algorithm for high school timetabling problems. Eur. J. Operational Res., 203: 550-558.