

AJUSTE DE PARÂMETROS DO SIMULATED ANNEALING APLICADO AO PROBLEMA DE CONSTRUÇÃO DE HORÁRIO EM ESCOLA

Leonardo Laraniaga Martins (PIBIC/CNPq/UEM), Ademir Aparecido Constantino (Orientador), e-mail: ademir.uem@gmail.com.

Universidade Estadual de Maringá / Departamento de Informática.

Ciências Exatas e da Terra, Ciência da Computação

Palavras-chave: Meta-heurística, Simulated Annealing, PHE.

Resumo:

Este trabalho consiste na investigação do Problema em Horários em Escolas (PHE), que se refere à construção de horário para os professores de escola atendendo a um conjunto de restrições operacionais. Este é um problema de otimização combinatória difícil do ponto de vista computacional. Foi utilizado o algoritmo trabalhado por (Pedro, 2016) baseado na meta-heurística *Simulated Annealing*, utilizando os operadores Torque e Double Move para tentar resolver o problema PHE. O algoritmo passou por um grande número de testes e com diferentes parâmetros sobre uma base de dados com instâncias para o problema. Os resultados obtidos foram comparados com resultados de trabalhos anteriores.

Introdução

O problema de horários de escolas é um problema NP-difícil pelo ponto de vista da teoria da complexidade computacional, logo, não se conhece algoritmo de complexidade polinomial capaz de gerar uma solução ótima para o problema. Como os algoritmos capazes de fornecer soluções ótimas, para estes casos, possuem complexidade exponencial, uma alternativa é o desenvolvimento de algoritmos heurísticos o problema. Tais métodos não oferecem uma solução ótima para o problema, mas podem oferecer boas soluções em um tempo computacional aceitável.

Esses algoritmos que fazem uso da técnica de busca local utilizando o conceito de vizinhança onde permite o algoritmo explorar um espaço de soluções do problema, e seleciona uma solução vizinha para substituir a atual. Dentre eles, os mais utilizados na literatura para o PHE são: Algoritmo Genético (AG), *Greedy Randomized Adaptive Search Procedure* (GRASP), Busca Tabu (BT), *Iterated Local Search* (ILS), *Variable Neighborhood Search* (VNS), *Simulated Annealing* e outros híbridos, como: GRASP + Tabu (Costa, 2003), ILS + Torque (TQ) (Saviniec, 2013), *Algoritmo Memético + Reconexão* (Coelho; Souza, 2006). Para o sucesso de um algoritmo deve-se usar um operador de vizinhança eficiente.

Neste trabalho foram realizados vários testes com o *Simulated Annealing* utilizando os operadores de vizinhança *Double Move* (DM), e Torque (TQ), que é o melhoramento de um operador clássico, desenvolvido por (Saviniec, 2013) e apresentou resultados superiores quando comparados com outras técnicas da literatura.

Materiais e métodos

Foi utilizado nos testes o método heurístico *Simulated Annealing* desenvolvido por (Pedro, 2016) em que, dada uma alta temperatura, pode ser permitida alterações ruins pois está longe de um ótimo local, com isso a temperatura vai reduzindo através de uma função de redução, onde as possibilidades de reduções ruins vão se reduzindo quanto mais próximo de um ótimo global.

O algoritmo *Simulated Annealing* foi compilado em C++ usado o compilador GCC. Os testes foram feitos num servidor com processador Westmere E56xx/L56xx/X56xx (Nehalem-C) com 8 núcleos, 2,26Ghz e 16GB de RAM, rodando em Windows Server 2012, 64 bits.

Foram explorados testes com combinações dos seguintes parâmetros para o *Simulated Annealing*:

Solução inicial: *Double Move*, Torque.

Temperatura inicial (T_0): 8, 9, 10, 11, 12, 20, 50, 100, 500, 1000.

Parâmetro de redução (α): 0,75; 0,78; 0,85; 0,90.

Número de iterações (Niter): 250, 1000.

Em todos os testes, a busca local foi realizada com o operador TQ.

Os experimentos foram realizados com instâncias reais (Saviniec, 2013), e cada uma delas foi processada 30 vezes em cada um dos testes.

Resultados e Discussão

Na tabela 1 serão exibidos resultados obtidos. Como critério de comparação, serão mostradas as melhores soluções entre as 30 execuções de cada instância para cada teste realizado. Os resultados são comparados com os das melhores soluções obtidas até então para estas instâncias do problema por (Saviniec, 2013), (Mamoru, 2014) e (Pedro, 2016).

Os testes possuem nomenclatura DMTQ ou TQTQ, dependendo se a solução inicial é obtida utilizando o operador DM ou TQ, respectivamente.

Pode-se perceber que os testes DMTQ foram inferiores na grande maioria das instâncias exceto em 22, 23 e 24. Em contrapartida, os testes TQTQ apresentaram resultados mais próximos ou até melhores do que o grupo de melhores soluções.

Tabela 1: Comparação dos melhores resultados obtidos.

Instância	Melhores soluções	Melhores soluções com DMTQ	TQTQ $\alpha = 0.78$, $T_o = 8$, 250 it.	TQTQ $\alpha = 0.90$, $T_o = 10$, 1000 it.	TQTQ $\alpha = 0.90$, $T_o = 1000$, 250 it.
1	290	350	340	310	310
2	290	330	320	280	300
3	310	370	360	360	340
4	270	370	330	320	310
5	280	340	320	310	320
6	320	450	410	370	390
7	620	640	580	610	600
8	300	320	290	280	290
9	360	390	350	340	340
10	100	160	120	110	120
11	650	590	540	510	510
12	520	550	500	490	510
13	70	90	70	70	70
14	400	410	390	380	390
15	90	110	100	90	90
16	160	200	180	170	170
17	130	170	160	140	150
18	140	180	160	140	150
19	90	120	90	90	90
20	170	210	170	170	170
21	590	590	620	630	620
22	450	420	560	520	530
23	1310	1130	1300	1260	1260
24	1390	1250	1400	1360	1360
25	620	660	650	600	640
26	220	270	280	260	260
27	210	250	280	240	250
28	590	690	670	650	660
29	610	680	650	630	630
30	720	720	720	700	700
31	660	730	720	710	700
32	500	620	570	570	560
33	530	610	590	580	590
34	500	620	590	570	570

OBS: Quanto menor o número, melhor é a solução.

Destaca-se o teste acima com $T_o = 8$, que pelos seus parâmetros baixos, apresentou resultados em baixo tempo de execução e que foram satisfatórios. Destaca-se, ainda, o teste acima de 1000 iterações por ter, no geral, apresentado os melhores resultados de todos os testes, ainda que possua um tempo de execução maior.

Por fim, destaca-se, negativamente, o teste acima de $T_0 = 1000$, por ter resultados muito semelhantes aos outros testes (ou até piores), e possuir um tempo de execução muito maior. O mesmo vale para todos os outros testes realizados com temperaturas iniciais a partir de 20.

Conclusões

Foi possível observar a partir dos resultados apresentados que a meta-heurística *Simulated Annealing*, combinada com o operador Torque conseguiu resultados muito próximos aos melhores resultados atingidos por estudos anteriores na maioria das instâncias, sendo que, ainda, foram descobertas novas melhores soluções durante a bateria de testes.

Pode-se constatar, com base nos experimentos, que o operador Torque foi o que mais contribuiu com a melhoria dos resultados obtidos até então em trabalhos anteriores. Já em relação aos outros parâmetros, recomenda-se utilizar altos parâmetros de redução e temperatura final (não muito maior que 12 para esta) para obter soluções mais próximas da ótima desde que o alto tempo de execução não seja um problema.

No geral, devido aos resultados de todos os algoritmos TQTQ terem sido muito próximos uns dos outros constatou-se que o operador Torque, combinado com a meta-heurística *Simulated Annealing* apresenta uma abordagem competitiva, pelo menos na faixa de valores investigada neste projeto

Agradecimentos

Meus agradecimentos ao CNPq e à Fundação Araucária pela bolsa PIBIC, ao Dr. Ademir Constantino por ter me orientado, e ao Pedro Moraes pela orientação na utilização do algoritmo.

Referências

MASSAGO, M. **Algoritmos heurísticos baseados em Busca Tabu para o problema de horários em escolas**. 2014. Relatório (Apresentação PIBIC/CNPq) – Universidade Estadual de Maringá, Maringá.

MORAES, P. **Simulated Annealing Combinado com Métodos Exatos para o Problema de Horários em Escolas**. 2016. Relatório (Apresentação PIBIC/CNPq) – Universidade Estadual de Maringá, Maringá.

SAVINIEC, L. **Operadores de Vizinhança eficientes para algoritmos de busca local aplicado ao problema de horário de escolas**. Dissertação de Mestrado em Ciência da Computação, Universidade Estadual de Maringá, p. 1-90, 2013.