colloquium exactarum

APLICAÇÃO DE ALGORITMOS GENÉTICOS NA CONSTRUÇÃO DE GRADES DE HORÁRIOS

APPLICATION OF GENETIC ALGORITHMS IN CONSTRUCTION OF TIMETABLES

Mário Ferreira Marques Júnior, Danillo Roberto Pereira, Francisco Assis da Silva

Universidade do Oeste Paulista – UNOESTE, Faculdade de Informática, Presidente Prudente, SP. e-mail: chico@unoeste.br

RESUMO - Este trabalho apresenta uma solução para a construção de grades de horários de Turma-Professor por meio da aplicação de algoritmos genéticos, levando em consideração as restrições encontradas na elaboração das grades de horários semestrais. O *software* que gera as grades de horários foi desenvolvido em C# e exibe combinações de horários plausíveis que atenda as restrições de horários de cursos, professores e disciplinas cadastradas. Por se tratar de um problema NP-Completo, nem sempre combinações válidas são encontradas pelo algoritmo, neste caso o programa sinaliza os horários das disciplinas que estão em conflito.
Palavras-chave: Algoritmo genético; Grade de horários; Problema da turma-professor; Algoritmo evolutivo.

ABSTRACT - This work describe a method to finding a solution for the construction of timetables. The approach uses the application of genetic algorithms to obtain the solution. The implementation took into account the constraints presents in the preparation of grids college. The software displays plausible combinations of schedules such that satisfies the schedules constraints of undergraduation, teachers and courses registered in the system. Since the problem is a NP-Complete, not always valid combinations are found by the algorithm, in this case the program highlights the course that are in conflict.
Keywords: Genetic Algorithm; Timetables; Problem Class-Teacher; Evolutionary Algorithm.

Recebido em: 05/10/2014
Revisado em: 20/11/2014
Aprovado em: 08/12/2014

1. INTRODUÇÃO

A construção de uma grade de horários também conhecido como Problema da Turma-Professor (PTP), segundo Sousa, Moretti e Podesta (2008); consiste em fixar uma sequência de agendamentos de aulas envolvendo professores e turmas (que possuem um mesmo currículo de disciplinas) em um período pré-determinado (tipicamente um período semanal). Esta sequência de agendamentos deve satisfazer um conjunto de requisitos didáticos, físicos e organizacionais. Há uma infinidade de variações do PTP. Isto se deve ao fato de que em países, regiões de um país, ou até mesmo, instituições de ensino de uma mesma região adotam critérios educacionais diferenciados entre si. Devido a esta característica, o PTP é um problema de difícil generalização. Resolvê-lo manualmente é uma tarefa difícil e que podem durar dias ou até semanas para sua conclusão, podendo gerar resultados insatisfatórios com relação a diversos requisitos. Muitas vezes, professores exigem um mínimo de aulas duplas para algumas disciplinas e, principalmente, por uma grade de horário com menos períodos vagos entre uma aula e outra. Dependendo do número de grupos de estudantes e professores envolvidos no problema, ele se torna manualmente intratável. Por estes motivos, uma atenção considerável tem sido investida à solução automática do PTP ao longo dos últimos anos.

Um número considerável de abordagens pode ser utilizado para a resolução desse problema (SCHMIDT; STRÖHLEIN, 1980) e modificações de sua abordagem tem sido utilizada já há algum tempo (BRITTAN; FARLEY, 1971). A grande quantidade de restrições envolvidas faz do PTP um candidato potencialmente adequado à resolução por meio de algoritmos genéticos (SOUSA; MORETTI; PODESTA, 2008).

Otimização é o processo de adaptação das entradas ou características de um dispositivo, processo matemático ou experimento para encontrar um resultado mínimo ou máximo. (HAUPT; HAUPT, 2004).

Algoritmos Evolucionários (AEs) são algoritmos de otimização ou que executam tarefas de aprendizagem com a capacidade de evoluir. Eles possuem três características principais:

- São baseados em população: AEs mantem um grupo de soluções, chamado de população, para otimizar ou aprender o problema de forma paralela. A população é um princípio básico do processo evolucionário.
- *Fitness*-orientado. Cada solução em uma população é chamada de indivíduo. Cada indivíduo tem sua representação genética, chamada de código, e avaliação de desempenho, chamado de valor de "*fitness*".

AEs dão preferência a indivíduos mais aptos, que é o fundamento da otimização e convergência dos algoritmos.

• Guiados por variações: Os indivíduos são submetidos a uma série de operações de variação para simular alterações genéticas, que é fundamental para a busca do espaço de soluções. (Yu; Gen, 2010).

Algoritmos com essas características têm sido propostos desde a década de 60. O termo Algoritmo evolucionário engloba, entre outras técnicas, os Algoritmos Genéticos.

Um algoritmo genético (GA) é uma técnica de otimização e busca baseada nos princípios da genética e seleção natural. O GA permite que uma população composta por muitos indivíduos evolua segundo as regras de seleção definidas para um estado que maximiza ou minimiza a "aptidão" (ou seja, minimiza ou maximiza a função custo) (HAUPT; HAUPT, 2004).

O método foi proposto por John Holland e popularizado por David Goldberg, que foi capaz de resolver um problema considerado de difícil solução (HOLLAND, 1975; GOLDBERG, 1989;).

2. TRABALHOS RELACIONADOS

A aplicabilidade de GAs é muito ampla, principalmente em problemas de busca ou otimização (LINDEN, 2012). Lembrando que GAs devem ser limitados apenas a problemas em que não há um algoritmo exato capaz de resolvê-los em tempo razoável, uma vez que algoritmos genéticos tendem a resolver qualquer tipo de problema porem em um tempo relativamente maior que um algoritmo especifico.

Ao usar um algoritmo genético para um problema deve-se atentar principalmente a escolha da representação cromossomial adequada, os operadores genéticos e seus comportamentos durante as gerações do algoritmo, a função de avaliação e o critério de parada do algoritmo.

Vários trabalhos e soluções envolvendo escalonamento de horário são apresentados, um fator comum na resolução destes problemas é que eles lidam com algum tipo de recurso (salas de aula e horários, por exemplo) que deve ser compartilhado, ou distribuído de forma a atingir o melhor efeito possível.

No trabalho de Corloni, Dorigo e Maniezzo (1990), o problema é representado como uma matriz, em que cada coluna representa um dia e cada linha representa um professor. Cada entrada rij da matriz consiste em uma sequência de cinco caracteres, em que cada posição pode assumir um valor especifico que representa de forma cromossomial as restrições do problema, com isso o objetivo de cada solução é obter o menor valor de uma função

de avaliação f, que leva em consideração fatores como a violação de restrições impostas pelo problema, como: dias de folga do professor, conflitos de horários de professores e turmas, ausência de professor para uma matéria, etc. O operador de *crossover* usa como blocos básicos as linhas das matrizes (o horário completo de um professor). Assim dois pais geram filhos cedendo algumas linhas para compor o horário completo do filho. Para tentar gerar filhos melhores, a avaliação parcial de cada linha é calculada e são criados filhos com as k melhores linhas (baseadas na sua avaliação) de um pai com as n-k linhas restantes do outro pai, onde n é o número total de linhas. O operador de mutação usado opera também em nível de linha, invertendo dois dias de um determinado professor. Outro operador usado é uma busca local, que inverte dias de horários aos pares, procurando o melhor individuo existente na vizinhança do cromossomo corrente. Com isso horários inaceitáveis são gerados, o que leva a necessidade de uma função de reparado usada nesta solução.

Rossi-Doria e Paechter (2004), assim como o trabalho de Corloni, Dorigo e Maniezzo (1990), descreve uma abordagem baseada em algoritmo memético juntamente com os GAs. O cromossomo usado neste trabalho é uma matriz em que as colunas representam os horários e as linhas representam as salas. Em cada posição da lista será colocada uma entrada representando a matéria que ocupara aquela sala naquele determinado horário, com o valor especial de -1 representando que aquela sala naquele horário está livre. Esta representação é interessante, pois elimina uma ampla gama de soluções inviáveis, como aquelas que representam duas matérias ocupando uma mesma sala no mesmo horário. As restrições neste problema são tratadas como *hard* (tornam um cromossomo inviável) e *soft* (devem ser atendidas, se possível). O operador de *crossover* é uma versão simples do operador de *crossover* uniforme, onde para cada horário é feito um sorteio de um numero 0 ou 1, se for 0 o filho herda a matéria do primeiro pai e se for selecionado o numero 1 ele herda do segundo pai, com isso apenas um filho é gerado de cada vez e um modulo de população do tipo steady-state onde o pior elemento da população é substituído pelo novo filho gerado.

No trabalho de (KANOH; SAKAMOTO, 2004), um gene consiste em uma atribuição Disciplina-Horário, onde Horário é uma combinação dada por (período, dia da semana, horário do dia). Cada cromossomo é uma alocação completa de todas as disciplinas existentes na faculdade, sendo avaliado de acordo com uma serie de restrições hard e soft, que são pontuadas de

acordo com a sua importância. Este trabalho usa um operador de mutação de inversão de posições, um operador de crossover uniforme e um operador de mapeamento parcial (PMX).

Além de trabalhos acadêmicos, existem diversas aplicações comerciais utilizadas atualmente, que utilizam diversas técnicas em conjunto para a resolução deste problema da melhor forma possível, entre elas o uso dos algoritmos genéticos.

Uma das mais conhecidas é o *software* aSc TimeTables (*http://www.asctimetables.com/)* que fornece uma grande quantidade de restrições e configurações que podem ser inseridas de acordo com o problema do usuário. Além disto, possui uma característica importante que ao não encontrar uma solução ideal para o problema, é oferecido ao usuário uma solução aproximada e a possibilidade de uma adequação manual.

Diversos outros *softwares* conhecidos para montar horários escolares podem ser citados, como o Urânia (*http://www.horario.com.br/*) um *software* pago de código fechado que deve ser adquirido na forma de licença de uso, mas existem alguns programas gratuitos e de código aberto (*open source*) como é o caso do FET (*Free Timetabling Software*) que se encontra disponibilizado em *http://www.lalescu.ro/liviu/fet/* na sua ultima versão FET-5.19.0.

3 DESENVOLVIMENTO DO PROJETO

3.1 Visão geral

O objetivo geral deste projeto foi desenvolver um *software* que gere grades de horários para os termos dos cursos cadastrados no sistema, baseando se em um conjunto de restrições mais comuns na Faculdade de Informática de Presidente Prudente. Para a resolução deste problema foram estudados e implementados algoritmos genéticos adaptados a esta aplicação especifica.

Com isso um estudo aprofundado a respeito de algoritmos genéticos foi realizado, desde seu conceito e funcionamento básico, passando por sua representação cromossomial (forma de representar o problema computacionalmente), escolha da população inicial, função de avaliação, modo de seleção dos pais, operadores genéticos, modulo da população, critério de parada até a implementação, execução e teste do algoritmo, após isso foram estudadas técnicas para refinar o algoritmo, adaptando seu funcionamento para o problema da grade de horário.

A interface desenvolvida teve como função principal fornecer dados para a execução do algoritmo, apresentando seus

resultados para o usuário de uma forma básica. O usuário inicialmente alimentará o sistema, cadastrando dados como cursos, disciplinas e professores e suas restrições.

No cadastro do curso é necessário especificar seu nome e quantidade de termos, para cada termo devem ser inseridas todas suas disciplinas, para que essas disciplinas sejam encaixadas na grade de horário do curso. Deve-se também cadastrar os horários disponíveis para o curso (noturno, diurno, integral ou sábados).

Para cadastrar a disciplina é necessário seu nome, a qual curso ela esta vinculada, o professor que a ministra (e um professor auxiliar opcional). Também deverão ser configuradas como serão suas aulas semanais, onde cada aula possui um tempo fixo de 50 minutos e poderão ser do tipo uma aula dupla, pares de aulas duplas, uma aula dupla e outra tripla, entre outras combinações.

No cadastro de professores é necessário seu nome e seus horários disponíveis em cada dia da semana para dar aula.

Após a inserção destes dados o algoritmo gerará as grades de horário dos cursos, especificando previamente quais termos serão gerados, por exemplo, em cursos onde turmas são anuais pode haver somente termos pares ou impares em um semestre.

Nem sempre uma solução ótima será encontrada pelo algoritmo, oferecendo uma solução aproximada do problema, devendo ao usuário avalia-la e tentar modificar de maneira manual através de conciliações no horário, como em caso de professores que dão aula em outros cursos ou universidades não cadastradas no sistema.

3.2 Metodologia

Inicialmente foram levantadas as restrições e necessidades envolvidas na construção de uma grade de horários juntamente com um estudo aprofundado dos diversos tipos de algoritmos genéticos. A aplicabilidade, restrições e técnicas computacionais exigidas foram elencadas para que fosse escolhida uma abordagem mais adequada ao problema proposto.

A seguir foram realizados alguns experimentos, implementando alguns tipos de algoritmos genéticos para a resolução de partes mais simples do problema, por exemplo, com número reduzido de restrições. Neste passo, foram verificadas a adequação e eficiência das técnicas propostas, bem como suas potenciais limitações.

Por fim, o algoritmo foi elaborado e implementado para a situação real da construção de uma grade de horários da FIPP.

3.3 Implementação

O algoritmo utiliza o funcionamento básico dos algoritmos genéticos, onde inicialmente é formada uma população inicial, neste caso um vetor de população de grades de horários.

Cada grade de horários pertence a um curso e termo específico e são representadas por matrizes, estas são tratadas como o cromossomo propriamente dito, possuindo sua função de avaliação e operadores genéticos. Cada horário disponível na matriz (cromossomo) é tratado como um gene, que pode possuir o valor de um código de uma disciplina agregado a um professor e professor auxiliar opcional. Os horários que estão vagos ou não podem ser preenchidos por limitações de período do curso são representadas por valores -1.

A população inicial gerada pode apresentar cromossomos válidos ou inválidos, decorrente das restrições cadastradas no problema. O próximo passo é avaliar todos esses cromossomos através de uma função custo, que retorna um valor representando a nota (*fitness*) daquele cromossomo.

A função de avaliação que determina o *fitness* de cada cromossomo usa um sistema de pontuação que leva em consideração fatores como: o professor da disciplina não possui conflitos de horários com outras disciplinas de outros cursos ou termos, isso também vale para o professor auxiliar caso existir. Todas as disciplinas cadastradas estão presentes no cromossomo e de forma válida, por exemplo, em disciplinas onde existem mais de um dia de aula na semana, estas não podem estar alocadas no mesmo dia. Com estas regras o cromossomo aumenta ou diminui seu *fitness*.

Após todos os cromossomos terem sua função de avaliação calculada, estes valores são armazenados e são utilizados posteriormente pela função que implementa o Método da Roleta Viciada. Neste método é como se existisse uma circunferência representando uma roleta, e cada cromossomo possui um pedaço dessa circunferência proporcionalmente ao seu *fitness* (quanto maior o *fitness* maior o pedaço na roleta), com isso é sorteado uma posição aleatória da roleta e o cromossomo que está naquela posição é escolhido. Consequentemente quanto maior o valor *fitness* do cromossomo maior será a probabilidade dele ser escolhido, assim cromossomos com menor valor *fitness* tendem a serem excluídos das próximas populações e não passam seus genes adiante.

O método da roleta viciada é utilizado para que sejam selecionados pares de pais a cada repetição da geração dos filhos. Este método realiza a parte de seleção dos pais do algoritmo, onde serão aplicados os

operadores genéticos para que gerem um novo filho.

Os operadores genéticos escolhidos consistem de uma variação do *crossover* de um ponto, funcionando de duas maneiras. A primeira utiliza um ponto de corte na grade de horário separando-a em relação aos dias (segunda,terça...), visualmente na matriz como uma divisão vertical conforme a Figura 1.

MANHÃ

	Horários		2ª Feira		3ª Feira		4ª Feira		5ª Feira		6ª Feira		Sábado	
	Início	Fim	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T
2ª	08:20	09:10	EMPREEND I Local LAB.107	A			LFA Local 11.3	A	LFA Local 11.3	A1	PROB.EST. Local 11.3	A1		
3ª	09:30	10:20	EMPREEND I Local LAB.107	A			LFA Local 11.3	A	LFA Local 11.3	A1	PROB.EST. Local 11.3	A1		
4ª	10:20	11:10	P.G.PROJ. Local LAB.101	A	REDES I Local REDES	A	E.SOF. III Local LAB.101	A	PROB.EST. Local 11.3	A	GRAFOS Local LAB.103	A		
5ª	11:10	12:00	P.G.PROJ. Local LAB.101	A1	REDES I Local REDES	A1	E.SOF. III Local LAB.101	A	PROB.EST. Local 11.3	A	GRAFOS Local LAB.103	A		

TARDE

	Horários		2ª Feira		3ª Feira		4ª Feira		5ª Feira		6ª Feira		Sábado	
	Início	Fim	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T
1ª	13:30	14:20			COMP.GRAF. Local LAB.103	A	T.T.COMP. Local LAB.104	A1	E.SOF. III Local LAB.101	A1				
2ª	14:20	15:10			COMP.GRAF. Local LAB.103	A	T.T.COMP. Local LAB.104	A1	E.SOF. III Local LAB.101	A1				
3ª	15:30	16:20			GRAFOS Local LAB.103	A	T.T.COMP. Local LAB.104	A1	COMP.GRAF. Local LAB.103	A1				
4ª	16:20	17:10			GRAFOS Local LAB.103	A			COMP.GRAF. Local LAB.103	A1				

NOITE

	Horários		2ª Feira		3ª Feira		4ª Feira		5ª Feira		6ª Feira		Sábado	
	Início	Fim	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T
1ª	19:00	19:50												
2ª	19:50	20:40												
3ª	20:50	21:40												
4ª	21:40	22:30												

Figura 1. Representação do *crossover* entre os dias.
Fonte: (Autor).

O outro operador de *crossover* separa a grade de horários em relação aos períodos de aula (manhã, tarde, noite) e também internamente a cada 50min de aula, como um ponto de corte horizontal na matriz conforme a Figura 2.

MANHÃ

	Horários		2ª Feira		3ª Feira		4ª Feira		5ª Feira		6ª Feira		Sábado	
	Início	Fim	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T
2ª	08:20	09:10	EMPREEND I	A Local LAB.107			LFA	A Local 11.3	LFA	A1 Local 11.3	PROB.EST.	A1 Local 11.3		
3ª	09:30	10:20	EMPREEND I	A Local LAB.107			LFA	A Local 11.3	LFA	A1 Local 11.3	PROB.EST.	A1 Local 11.3		
4ª	10:20	11:10	P.G.PROJ.	A Local LAB.101	REDES I	A Local REDES	E.SOF. III	A Local LAB.101	PROB.EST.	A Local 11.3	GRAFOS	A Local LAB.103		
5ª	11:10	12:00	P.G.PROJ.	A1 Local LAB.101	REDES I	A1 Local REDES	E.SOF. III	A Local LAB.101	PROB.EST.	A Local 11.3	GRAFOS	A Local LAB.103		

TARDE

	Horários		2ª Feira		3ª Feira		4ª Feira		5ª Feira		6ª Feira		Sábado	
	Início	Fim	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T
1ª	13:30	14:20			COMP.GRAF.	A Local LAB.103	T.T.COMP.	A1 Local LAB.104	E.SOF. III	A1 Local LAB.101				
2ª	14:20	15:10			COMP.GRAF.	A Local LAB.103	T.T.COMP.	A1 Local LAB.104	E.SOF. III	A1 Local LAB.101				
3ª	15:30	16:20			GRAFOS	A Local LAB.103	T.T.COMP.	A1 Local LAB.104	COMP.GRAF.	A1 Local LAB.103				
4ª	16:20	17:10			GRAFOS	A Local LAB.103			COMP.GRAF.	A1 Local LAB.103				

NOITE

	Horários		2ª Feira		3ª Feira		4ª Feira		5ª Feira		6ª Feira		Sábado	
	Início	Fim	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T	Abrev.	T
1ª	19:00	19:50												
2ª	19:50	20:40												
3ª	20:50	21:40												
4ª	21:40	22:30												

Figura 2. Representação do *crossover* entre as aulas.

Fonte: (Autor).

Em cada geração de um filho uma das duas formas do *crossover* ocorre, com a probabilidade de 50% de ocorrência de cada. Após isso um novo filho é gerado obtendo as partes de cada pai, e nele pode ser aplicado ou não a mutação, que acontece na ordem de 1%. Esta mutação seleciona uma disciplina aleatória na grade e a troca por uma disciplina que não está presente naquele cromossomo.

Este procedimento de geração de filhos é repetido "n" vezes, onde n seria a quantidade de elementos presentes na população anterior, com isso uma nova população é gerada com a mesma quantidade de elementos, geralmente o valor de avaliação dos cromossomos aumenta.

Este processo pode ser repetido caso a solução não tenha sido encontrada (a função de avaliação não seja máxima, ou seja, todas as disciplinas estão presentes na grade e não possuem conflitos de horários). Neste caso o algoritmo irá parar ao atingir um número máximo de gerações/iterações. Com isso o melhor cromossomo será apresentado como uma possível solução e suas disciplinas em conflito serão destacadas em vermelho para o usuário.

4 RESULTADOS

Foram testados 10 cenários diferentes, onde cada cenário consiste em uma combinação diferente de professores, disciplinas, termos e cursos cadastrados no

sistema para o cálculo das grades de horários. Para cada cenário o programa foi executado 10 vezes. Por se tratar de um problema combinatorial onde não há somente uma solução correta, alguns cenários apresentaram soluções diferentes ou iguais as já encontradas. Os resultados destes testes encontram-se na Tabela 1. Cada teste possui um percentual de convergência total que representa o percentual de soluções válidas encontradas. É exibido também o percentual de disciplinas em conflito que representa os testes que não encontraram solução (não convergiram).

Tabela 1. Resultado dos testes para 10 cenários diferentes.

Cenário	Média de iterações	Convergência Total	Disciplinas em conflito
1	718	90%	14%
2	395	100%	0%
3	270	100%	0%
4	1300	80%	22,5%
5	680	90%	10%
6	200	100%	0%
7	180	100%	0%
8	2300	70%	35%
9	425	100%	0%
10	915	80%	17,5%

Fonte: (Autor).

5 CONSIDERAÇÕES FINAIS

De modo geral, o problema conseguiu ser resolvido em tempo hábil, variando de 1 a 10 segundos em média. Vale realçar que quanto maior o numero de restrições envolvidas, maior é o custo computacional para encontrar a solução. Outros fatores como quantidade de cursos, termos, disciplinas e professores também influenciam diretamente em seu tempo de processamento. Por se tratar de um problema NP-Completo nem sempre uma solução é encontrada, com isso é sinalizado ao usuário (destacado em vermelho) quais disciplinas permaneceram em conflito na melhor solução encontrada. Em trabalhos futuros iremos envolver no problema a alocação de salas de aula da FIPP que também é um fator importante.

REFERÊNCIAS

BRITTAN, J. N. G.; FARLEY, F. J. M. College timetable construction by computer. **The Computer Journal**, v. 14, n. 4, 1971. doi:10.1093/comjnl/14.4.361. http://dx.doi.org/10.1093/comjnl/14.4.361

CORLONI A.; DORIGO, M.; MANIEZZO, V. **Genetic algorithms and highly constrained problems:** the time-table case. INTERNATIONAL WORKSHOP ON PARALLEL PROBLEM SOLVING FROM NATURE, 1., 1990. **Proceedings**... Dortmund, Alemanha

Springer-Verlag, 1990. P. 55-59. Lecture Notes in Computer Science 496.

GOLDBERG, David E. **Genetic algorithms in search, optimization and machine learning.** Boston, MA: Addison-Wesley Longman Publishing, 1989.

HAUPT, Randy L.; HAUPT, Sue Ellen. **Practical genetic algorithms. 2. ed.** Nova Jersey: John Wiley & Sons, 2004.

HOLLAND, J. **Adaptation in natural and artificial systems.** [S.l]: The Univ. of Michigan Press, 1975.

KANOH, H.; SAKAMOTO, Y. Interactive timetabling system using knowledge-based genetic algorithms. **IEEE International Conf. on Sys., Man and Cybernetics**, v. 6, 2004. http://dx.doi.org/10.1109/icsmc.2004.1401129

LINDEN, Ricardo. **Algoritmos genéticos. 3. ed**. Rio de Janeiro: Ciência Moderna, 2012.

ROSSI-DORIA, O.; PAECHTER, B. **A memetic algorithm for University Course Timetabling**: relatório técnico. Edimburgo, Escócia: Napier University, School of Computing, 2004.

SCHMIDT, G.; STRÖHLEIN, T. Timetable construction: an annotated bibliography. **Computer Journal**, v. 23, n. 04, p. 307-316, 1980. http://dx.doi.org/10.1093/comjnl/23.4.307

SOUSA, Vânia Nobre de; MORETTI, Antônio Carlos; PODESTA, Valéria Abrão de. **Programação da grade de horário em escolas de ensino fundamental e médio**. Revista Pesquisa Operacional (ISSN 1678-5142), vol. 28, no. 3, Rio de Janeiro, Dec. 2008. Disponível em http://www.bibliotecadigital.unicamp.br/document/?down=vtls000382790; 8 abr 2013. doi: 10.1590/S0101-74382008000300002.

YU, Xinjie; GEN, Mitsuo. **Introduction to evolutionary algorithms.** Londres: Springer-Verlag, 2010.