



**INSTRUÇÕES PARA A PREPARAÇÃO E SUBMISSÃO DE
TRABALHOS PARA CONGRESSO BRASILEIRO DE ENSINO DE
ENGENHARIA 2003**

**ALGORITMO EVOLUTIVO PARA A OTIMIZAÇÃO DA
GRADE HORÁRIA DO CURSO DE ENGENHARIA
ELÁTRICA DA FEIS-UNESP**

Ruben Romero - ruben@dee.feis.unesp.br

Edson Buosi – buosi@dee.feis.unesp.br

Anna Diva Lotufo – annadiva@dee.feis.unesp.br

Universidade Estadual Paulista Julio de Mesquita Filho

DEE-FEIS-UNESP

Av. Brasil 56 - Centro

15385-000 – Fone (18) 3743.1150 – Fax (18) 3743 1163

Resumo: *É realizada a formulação adequada do problema de grade horária de um curso de engenharia e a resolução desse problema usando um algoritmo evolutivo. A proposta consiste em minimizar o conflito entre disciplinas correlatas dentro de uma grade horária semanal de forma que permita aos alunos levarem um número maior de créditos em cada semestre. O problema foi formulado levando em conta características específicas do Curso de Engenharia Elétrica mas podem ser facilmente adaptadas para qualquer curso de ensino superior. Os resultados encontrados foram melhores que os elaborados por um especialista de forma manual.*

Palavras-chave: *Condições de ensino, critérios de avaliação, ensino em engenharia elétrica.*



1. INTRODUÇÃO

A elaboração da grade horária de um Curso deve ser elaborada anualmente sob a responsabilidade do Coordenador de Curso. O produto final dessa tarefa é uma proposta de grade horária que especifica o horário de todas as aulas das disciplinas de um curso. Na elaboração da grade horária devem ser levadas em conta critérios genéricos para todos os cursos e critérios específicos para cada curso. Uma grade horária adequadamente elaborada deve permitir que um aluno em fase se matricule em todas as disciplinas do semestre e, eventualmente, tenha possibilidade de adiantar algumas disciplinas de semestres superiores. Também, uma grade deve permitir que alunos defasados tenham a possibilidade de recuperar as disciplinas atrasadas. Para que uma grade permita esse tipo de flexibilidade não devem aparecer conflito de horários entre disciplinas relacionadas.

O problema de otimização de grade horária pertence a uma família de problemas de otimização conhecida na literatura como *timetabling*. Nesse tipo de problema, pretende-se otimizar os recursos distribuídos em uma estrutura de horários fixos de atividades não necessariamente educacionais. O problema mais comum é o problema de otimização de distribuição de recursos educacionais em ensino tipo colegial, conforme VALDES (1996). Nesse tipo de problema, deve-se encontrar uma proposta de programação das aulas das diferentes séries, indicando os horários das disciplinas, as salas de aula que devem ser usadas por cada turma, a programação dos ambientes especiais tais como os laboratórios e auditórios, assim como a programação de aulas dos professores. Esse tipo de problemas assumem cada vez formas mais complexas e não podem ser resolvidas de forma adequada de forma manual por um especialista.

Neste trabalho apresentamos uma proposta de otimização do problema de grade horária para o Curso de Engenharia Elétrica da FEIS-UNESP. O problema é formulado como um problema de minimização de conflitos entre disciplinas correlatas. Para resolver esse problema é usado um algoritmo evolutivo com uma especialização que leva em conta as características específicas do problema. Os resultados encontrados foram melhores que os obtidos de forma manual por um especialista. O algoritmo proposto pode ser usado, com pequenas modificações para otimizar a grade horária de qualquer curso, especialmente a grade horária de cursos com *curriculum* flexível e regime semestral.

2. O PROBLEMA DE OTIMIZAÇÃO DE GRADE HORARIA

O Curso de Engenharia Elétrica da FEIS-UNESP é semestral com período mínimo de conclusão de 10 semestres e tem *curriculum* flexível. O Curso recebe anualmente 80 alunos, 40 em fevereiro e 40 em agosto. Um aspecto importante é que devido ao grau de dificuldade de muitas disciplinas é comum encontrar alunos cursando disciplinas de diferentes semestres. Por outro lado, apesar de que as disciplinas são agrupadas em semestres, algumas disciplinas mantêm relação de pré-requisito em que uma disciplina deve ser concluída para levar a outra disciplina e também relação de co-requisito como disciplinas de teoria e o correspondente laboratório. Existem ainda algumas particularidades no Curso como a possibilidade de levar uma disciplina com o pré-requisito simultaneamente se o aluno já levou o pré-requisito no semestre anterior e reprovou com uma média maior de 3,0 e não teve problemas com índice de frequência nessa disciplina.

No caso do Curso de Engenharia Elétrica da FEIS-UNESP, pretende-se encontrar uma grade ótima sujeito a um conjunto de restrições. Uma proposta de grade ótima é aquela que

minimiza o conflito entre disciplinas correlatas. As restrições devem levar em conta restrições de caráter geral de problemas tipo *timetabling* e restrições específicas ao problema específico que se pretende otimizar. Entre as restrições de tipo geral de um Curso podemos mencionar os seguintes: (1) não deve existir conflito de horário entre disciplinas do mesmo semestre, (2) todas as disciplinas devem ser programadas no horário disponível de 40 horas semanais com 8 horas disponíveis por dia e, (3) as aulas de um professor devem ser programados em horários diferentes e, preferencialmente em horários não consecutivos

Entre as restrições de caráter específico ao Curso de Engenharia Elétrica da FEIS pode-se mencionar o seguintes: (1) Existem disciplinas de teoria e de laboratório, (2) as disciplinas de laboratório tem uma duração de 4 horas quinzenais e são consideradas como disciplinas independentes da teoria, (3) existe a figura do pré-requisito entre disciplinas, (4) o pré-requisito pode ser dispensado desde que um aluno já levou o pré-requisito e reprovou com média maior de 3,0, não teve problemas de frequência, foi levada no semestre anterior e desde que se matricule simultaneamente na disciplina e no correspondente pré-requisito, (5) existem disciplinas relacionadas na forma de co-requisito que significa que essas disciplinas devem ser cursadas simultaneamente, etc.

Existem ainda restrições adicionais que devem ser levadas em conta na elaboração de uma grade horária. Assim, por exemplo, se uma disciplina de teoria que tem 6 horas de aula por semana, deve-se programar apenas uma aula de duas horas de duração por dia e as disciplinas de laboratório devem ter 4 horas de duração e realizadas quinzenalmente. Portanto, o problema de elaboração de grade horária consiste em encontrar uma grade horária que minimize o conflito entre disciplinas correlatas sujeito ao conjunto de restrições anteriormente especificadas. Em outras palavras, uma grade horária bem elaborada deve permitir que um aluno em fase possa se matricular em todas as disciplinas do semestre e algumas disciplinas adicionais de semestres posteriores e que os alunos que não estão em fase tenham a possibilidade de entrar em fase cursando algumas disciplinas reprovadas.

Deve-se observar que é muito difícil formular matematicamente o problema de grade horária apresentada anteriormente. Assim como é difícil formular matematicamente, através de relações algébricas, os problemas da família *timetabling*. Essa característica desse tipo de problemas torna particularmente difícil encontrar a solução ótima desses problemas porque não podem ser utilizadas técnicas de otimização usadas em problemas que apresentam uma modelagem matemática amigável.

Os problemas da família *timetabling* tradicionalmente foram resolvidos usando algoritmos heurísticos que tipicamente usam uma estratégia de dois passos: (1) encontrar uma solução factível para o problema e (2) se possível melhorar a solução factível encontrada. Nos últimos anos estão sendo usadas técnicas chamadas de metaheurísticas para otimizar os problemas da família *timetabling*. Existem muitos algoritmos que podem ser considerados como sendo metaheurísticas e os mais conhecidos são *simulated annealing*, algoritmos genéticos e evolutivos, *tabu search*, GRASP, etc. Metaheurísticas usadas em problemas da família *timetabling* podem ser encontradas em MONFROGLIO (1996), HERTZ (1991) e KRAGELUND (1997).

As metaheurísticas são técnicas de otimização através de busca no espaço de soluções de um problema. Assim, encontra-se uma proposta de solução inicial, define-se uma estratégia de vizinhança dessa proposta de solução e, deve-se realizar uma transição para um dos vizinhos da proposta de solução corrente. O novo vizinho selecionado se transforma na solução

corrente, define-se a nova vizinhança e, deve-se realizar um conjunto de transições determinada pela estratégia de cada tipo de metaheurística. Em determinadas condições se termina o processo de transições e a melhor solução identificada nesse processo de transições é a resposta do processo de otimização. Logicamente, para problemas complexos, dificilmente uma metaheurística encontra a solução ótima do problema. Assim, pode-se dizer que uma metaheurística encontra soluções sub-ótimas de problemas complexos.

3. ALGORITMO EVOLUTIVO PARA OTIMIZAR A GRADE HORARIA

O algoritmo evolutivo é um processo de otimização de problemas complexos e que originalmente foi formulado inspirado em conceitos existentes na seleção natural e na evolução das espécies. Assim, um algoritmo evolutivo simula a seleção natural usando os chamados operadores genéticos. Na estratégia de otimização de um algoritmo evolutivo, deve-se selecionar uma população inicial e aplicar nessa população os operadores genéticos de seleção, recombinação e mutação para encontrar uma nova população ou geração mais evoluída. A teoria sobre algoritmos evolutivos pode ser encontrada em MICHALEWICZ (1999). Um dos problemas críticos na formulação de um algoritmo evolutivo é a forma de codificar ou representar uma proposta de solução. Essa codificação deve permitir identificar um ponto do espaço de busca do problema e também deve permitir aplicar os operadores genéticos nas soluções codificadas. Assim, um dos problemas mais críticos de um algoritmo evolutivo é a escolha adequada da codificação do problema que pretendemos resolver.

Para resolver um problema complexo usando um algoritmo evolutivo, deve-se especificar ou definir cada um dos seguintes aspectos: (1) a proposta de codificação, (2) a forma de montar a população inicial, (3) a forma de encontrar a função objetivo ou um equivalente chamado de função de adaptação, (4) a forma de lidar com infactibilidades, isto é, de propostas de solução que violam alguma das restrições, (5) o tipo de seleção que deve ser implementada, (6) a forma de implementar a recombinação, (7) a forma de implementar a mutação, (8) o critério de parada e (9) a forma de implementar outros operadores que levem em conta as características específicas do problema. A seguir detalhamos a forma em que foram implementados cada um desses tópicos para o problema de otimização de grade horária.

Inicialmente, deve-se observar que como na maioria dos problemas da família *timetabling*, não foi usada uma formulação matemática. Entretanto, deve-se especificar a forma da função objetivo que se pretende otimizar. Assim, a função objetivo considerada foi a minimização de conflitos entre disciplinas. Nesse contexto, uma disciplina se encontra em conflito com outra disciplina se eles compartilham o mesmo horário de uma aula na grade horária. Portanto, levando em conta o tipo de correlação entre as disciplinas foram especificados cinco tipos de conflito com os seguintes valores: (1) 20 unidades de conflito para disciplinas fortemente relacionadas como, por exemplo, disciplinas do mesmo semestre e que não deve existir em uma grade considerada factível, (2) 10 unidades de conflito para disciplinas fortemente relacionadas como, por exemplo, disciplinas relacionadas na forma de pré-requisito, (3) 4 unidades de conflito para disciplinas com algum tipo de relação como, por exemplo, disciplinas de semestres consecutivos, (4) 2 unidades de conflito para disciplinas com baixa relação como, por exemplo, disciplinas de semestres não consecutivos mas com forte relação de conteúdo e (5) zero unidades de conflitos entre disciplinas sem relação apreciável. Assim, uma grade ideal apresentaria apenas conflitos do último tipo (tipo cinco). Entretanto, na proposta consideramos uma grade como sendo factível se não apresenta

nenhum conflito de tipo 1 (20 unidades de conflito) e minimize a somatória total de conflitos entre as disciplinas.

Na formulação do problema, a grade semanal de 40 horas foi separada em 20 unidades de aula sendo que uma aula tem duas horas de duração. Assim, existem 20 pontos de alocação de aulas. Nesse contexto, podemos apresentar a estratégia fundamental do algoritmo evolutivo, isto é, a codificação proposta para o problema.

A codificação escolhida é um arranjo vetorial de tamanho definido em que é alocada a grade horária de cada disciplina. Cada elemento do arranjo vetorial é um número de 1 a 20 que define o horário da aula. Adicionalmente, são usados dois vetores auxiliares para especificar o número que identifica cada disciplina e o tipo de disciplina (teoria ou laboratório). Assim, a proposta de codificação assume a forma mostrada na “Figura 1”.

7	9	10	4	8	5	13	1	2	7	11
1	2	2	3	3	4	4	5	5	80	80
0	1	1	0	0	0	0	1	1	0	0

Figura 1: Codificação no problema de grade horária.

Na “Figura 1”, o primeiro vetor representa a codificação e os outros são vetores auxiliares. O primeiro vetor auxiliar identifica a disciplina e o segundo o tipo de disciplina, 0 para indicar uma disciplina de teoria e 1 para o laboratório. Assim, na codificação apresentada na “Figura 1”, a disciplina No. 1 é de teoria e tem apenas uma aula de duas horas programada na terça de 16:00 a 18:00 horas (7 na codificação); a disciplina No. 2 é de laboratório com 4 horas de duração (duas aulas) programadas na terça de 8:00 a 12:00 horas; a disciplina No. 3 é de teoria com duas aulas (4 horas) sendo que a primeira aula está programada na segunda de 16:00 a 18:00 horas e a outra na terça de 16:00 a 18:00 horas e assim sucessivamente. Analisando o vetor de codificação identificamos o horário de cada disciplina e, portanto, esse vetor representa uma proposta de solução. Deve-se observar que no algoritmo evolutivo se trabalha com uma população de soluções, isto é, deve-se gerar um conjunto de vetores de codificação, cada um representando uma proposta de solução, mas existe apenas dois vetores auxiliares que permite identificar a solução apresentada por cada vetor de codificação. Nos testes apresentados neste trabalho foram usadas 80 disciplinas dos 3 últimos anos do curso de engenharia elétrica da FEIS-UNESP e, nesse contexto, o vetor de codificação tem 157 elementos. Deve-se observar que os elementos do vetor de codificação são números de 1 a 20 sendo que 1, representa uma aula na segunda feira de 8:00 a 10 horas, 2 uma aula na segunda feira de 10:00 a 12:00 horas e assim sucessivamente até chegar a 20 que representa uma aula na sexta feira de 16:00 a 18:00 horas. A codificação apresentada permite identificar facilmente a proposta de grade horária e também permite implementar de forma eficiente os operadores genéticos. Neste caso não foi implementada a chamada codificação binária porque elas são pouco eficientes para problemas que tem a estrutura de problemas tipo *timetabling* como o caso do problema de grade horária, de acordo com MICHALEWICZ (1999).

A população inicial foi montada de forma aleatória mas essa população foi melhorada eliminando propostas inconsistentes antes de passar para a fase iterativa com a implementação dos operadores genéticos. Assim, pode ser montada uma proposta de grade horária simplesmente escolhendo aleatoriamente cada elemento do vetor de codificação gerando números aleatórios de 1 a 20. Assim, cada elemento de uma população que é uma proposta de solução é representado por um vetor codificado na forma mencionada anteriormente e com elementos variando de 1 a 20. Logicamente, essa proposta de grade horária na maioria dos casos deve ser de pobre qualidade e também inactível. O processo de otimização pode ser implementada a partir dessa população. Entretanto, neste trabalho foi implementada uma fase de melhoria dos elementos da população eliminando dois tipos de erros grosseiros: (1) modificando as aulas de disciplinas de teoria que foram programados no mesmo dia e (2) juntando as aulas de laboratório que foram programadas em aulas não consecutivas. Por exemplo, se uma disciplina de teoria com duas aulas estão representados pelos números 2 e 4 significa que essa disciplina está programada na segunda feira de 10:00 a 12:00 e de 16:00 a 18:00 horas. Essa proposta é inaceitável porque as aulas devem ser programadas em dias diferentes. Também, se uma disciplina de laboratório está representado pelos números 5 e 9 significa que está programada uma aula na terça de 8:00 a 10:00 horas e a outra aula na quarta de 8:00 a 10:00 horas. Como as aulas de laboratório do Curso tem 4 horas de duração consecutivas também essa proposta é inaceitável. Deve-se observar que na codificação, as aulas de laboratório adequadamente programadas devem estar representados por dois números consecutivos e sendo o primeiro deles impar. Portanto, a população gerada aleatoriamente é corrigida eliminando esses dois tipos de erros grosseiros na grade horária. Da mesma forma, os dois tipos de erros mencionados anteriormente são evitados na implementação dos operadores genéticos nas seguintes fases do algoritmo evolutivo.

A próxima etapa do algoritmo evolutivo deve encontrar a função de adaptação de cada elemento da população. No problema de grade horária foi definida a função de adaptação como a soma de conflitos entre as disciplinas na grade horária. Como foi mencionado anteriormente, foram definidos cinco tipos de conflito entre as disciplinas sendo que uma delas representa conflito máximo (de valor 20) e no outro extremo temos o conflito mínimo (de nível 0). Assim, no banco de dados do problema, deve-se colocar cada disciplina e o tipo de conflito dessa disciplina com outras disciplinas e não sendo necessário especificar explicitamente o conflito mínimo. Com esse banco de dados é montada a matriz de conflitos que identifica o tipo e valor de conflito de cada disciplina com as outras disciplinas. Com a matriz de conflitos disponível e montada uma única vez é possível calcular a função de adaptação percorrendo o vetor de codificação que representa cada grade horária e com o auxílio da matriz de conflito. Assim, pode-se determinar facilmente a função de adaptação de cada elemento da população, isto é, de cada proposta de grade horária.

O problema de inactibilidades no problema de grade horária é contornada de duas formas diferentes. As inactibilidades chamadas grosseiras neste trabalho e relacionadas com as disciplinas de teoria cujas aulas não podem ser programadas no mesmo dia e das disciplinas de laboratório cujas aulas devem ser programadas em quatro horas consecutivas são sistematicamente eliminadas na formação da população inicial e na implementação dos operadores genéticos. Assim, não existe nenhuma proposta de grade horária com esse tipo de inactibilidade. Por outro lado, as inactibilidades relacionadas com conflito máximo entre disciplinas com elevado nível de correlação são implicitamente eliminadas pelo processo de otimização porque são altamente penalizadas na função de adaptação e também, quando é gerada e armazenada a incumbente (melhor solução encontrada no processo e que no final

representa a resposta do processo de otimização) propostas de grade horária com algum tipo de conflito máximo são rejeitadas.

Na implementação dos operadores genéticos foram escolhidas estratégias padrão amplamente consagradas na literatura especializada. Assim, foi implementada a seleção baseada em torneio. Nesse caso, deve-se implementar a seleção em um processo de np passos para uma população de tamanho np . Em cada passo é realizado um jogo, isto é, escolhe-se aleatoriamente $k = 3$ grades e a grade horária ganhadora é aquela que tem uma função de adaptação de melhor qualidade. Na recombinação foi implementada a recombinação de um ponto. Nesse caso, escolhe-se aleatoriamente um ponto de recombinação e, deve-se implementar a troca das parcelas de duas grades horárias selecionadas para produzir duas novas grades horárias. Nesse caso, para evitar gerar grades horárias com erros grosseiros, deve-se verificar esse tipo de problema e, caso seja necessário, deve-se mudar o ponto de recombinação em uma ou duas posições, evitando separar as aulas de laboratório ou a programação de aulas de teoria no mesmo dia. Esse tipo de problema pode acontecer com a disciplina que está no ponto de recombinação. Finalmente, a mutação é realizada simplesmente mutando o valor de um ou vários elementos de uma grade horária. Assim, uma vez escolhido o ponto de mutação, deve-se aumentar ou diminuir o valor desse elemento no arranjo vetorial em uma unidade. Nesse caso, deve-se levar em conta os problemas de fronteira e também pode ser necessário implementar mudanças adicionais na vizinhança do ponto de mutação para eliminar erros grosseiros relacionados com os horários de disciplina de teoria ou laboratório em torno do ponto de mutação.

No algoritmo evolutivo foi implementado também um processo de otimização local da melhor grade de cada geração. Assim, para a melhor grade de cada geração, identifica-se a existência de conflito máximo e cada um desses conflitos são eliminados transferindo um dos horários em conflito para a posição da grade horária com menor número de ocupações. Esse processo é realizado dinamicamente e leva em conta as particularidade mencionadas anteriormente das disciplinas de teoria e de laboratório para evitar erros grosseiros.

O critério de parada do algoritmo evolutivo foi escolhido usando uma estratégia mista, um critério adaptativo e um critério determinístico. Assim, o algoritmo evolutivo termina o processo de otimização se foi atingido um número máximo de iterações previamente especificado ou se a incumbente não melhora durante um número consecutivo de gerações também previamente especificado.

4. RESULTADOS OBTIDOS

O algoritmo evolutivo encontrou soluções melhores que os obtidos de forma manual por um especialista. Assim, por exemplo, uma das melhores soluções encontradas, na elaboração da grade horária de 80 disciplinas do Curso de Engenharia da FEIS-UNESP (disciplinas do 3º, 4º e 5º ano) encontrou uma grade horária com as seguintes características:

No de conflitos de valor 20 (máximo): zero
No de conflitos de valor 10: zero
No de conflitos com valor 4: 009
No de conflitos com valor 2: 007
No de conflitos com valor 0: 548
Valor da função de adaptação: 50



Assim, foi encontrada uma grade horária em que não existe nenhum conflito nos dois níveis mais elevados, mostrando uma grade horária de excelente qualidade. No processo de otimização foi usada uma população de tamanho 100, uma taxa de recombinação de 0,9, uma taxa de mutação de 0,05 e parando o processo se for atingido um máximo de 4000 gerações ou se a incumbente não muda após 400 gerações consecutivas. A solução mostrada foi encontrada após 3001 gerações. O tempo de processamento foi de 4 a 8 segundos em um computador Pentium III. O algoritmo evolutivo foi implementado computacionalmente em linguagem de programação Fortran.

5. CONCLUSÕES

No presente trabalho foi apresentado um algoritmo evolutivo para o problema de otimização de grade horária de um curso com estrutura semestral e *curriculum* flexível. O algoritmo foi implementado levando em conta as características específicas do Curso de Engenharia Elétrica da FEIS-UNESP mas pode ser facilmente adaptado para qualquer tipo de curso. O algoritmo encontrou soluções de melhor qualidade que os encontrados de forma manual por um especialista. Outra vantagem de usar um *software* baseado em metahuerísticas como o algoritmo evolutivo é que podem ser facilmente geradas muitas propostas de grade horária de qualidade muito parecidas. Assim, o responsável da elaboração e aprovação da grade horária geralmente tem muitas alternativas para escolher a mais adequada levando em conta outras particularidade do Curso.

Agradecimentos

O presente trabalho contou com o apoio da FUNDUNESP

6. REFERÊNCIAS

ALVARES-VALDES, R.; MARTIN, G.; TAMARIT J. M. Constructing Good Solutions for the Spanish School Timetabling Problem. **Journal of the Operational Research Society** n.47, p 1203-1215, 1996 .

HERTZ, A. Tabu Search for Large Scale Timetabling Problems. **European Journal of the Operational Research**. n. 54, p. 39-47. North-Holland, 1991.

KRAGELUND, L.V. Solving a Timetabling Problem Using Hybrid Genetic Algorithms, **Software Practice and Experience**, v.26, n. 3, p. 251-279, 1997.

MICHALEWICZ, Z. **Genetic Algorithms + Data Structures = Evolution Programs**. Springer. 1999.

MONFROGLIO, A. Timetabling Through Constrained Heuristic Search and Genetic Algorithms. **Software Practice and Experience**, v.26, n. 3, p. 251-279, 1996.

Tópico escolhido: Condições de Ensino.



EVOLUTIVE ALGORITHM FOR OPTIMIZING THE TIME TABLE OF THE ELECTRICAL ENGINEERING COURSE OF FEIS-UNESP

Abstract: *It is effectuated an adequate formulation of the time table problem of an engineering course and the solution of this problem using an evolutive algorithm. The proposal consists in minimizing the conflict with correlated disciplines in a week time table, allowing the students to course the maximum of disciplines in each semester. The problem was formulated considering the specific characteristics of an Electrical Engineering Course, but it can easily be adapted to any undergraduate course. The results are better than those elaborated by an expert in a hand way.*

Key words: *Evaluation criteria, teaching conditions, electrical engineering instruction*