



UMA HEURÍSTICA GRASP+VND PARA O PROBLEMA DE PROGRAMAÇÃO DE HORÁRIO ESCOLAR

Janio Gloria de Oliveira^a, Dalessandro Soares Vianna^{a,b}, Marcilene de Fátima Dianin Vianna^c

^a Mestrado em Pesquisa Operacional e Inteligência Computacional, Universidade Candido Mendes – Campos (UCAM-Campos), Campos dos Goytacazes, RJ, Brasil

^b Departamento de Ciência e Tecnologia, Universidade Federal Fluminense – Pólo Universitário de Rio das Ostras (UFF-PURO), Rio das Ostras, RJ, Brasil

^c Departamento de Ciências Naturais, Universidade Estadual Norte Fluminense (UENF), Campos dos Goytacazes, RJ, Brasil

Resumo

Este trabalho propõe uma heurística GRASP que utiliza, na etapa de busca local, a técnica VND para solucionar o Problema de Horário Escolar. Consideraram-se, para programação de dois turnos do ensino médio, os dados da Escola Agrotécnica Federal de Alegre (ES, Brasil) nos anos de 2006 e 2007 e outras instâncias geradas neste trabalho para testes. O método proposto tem como objetivo satisfazer as restrições de formação do horário da instituição, tais como: satisfazer aos professores quanto à quantidade de dias em que suas aulas são distribuídas e respeitar seus horários de indisponibilidade; minimizar o número de aulas consecutivas da mesma disciplina; evitar a ocorrência de aulas vagas; evitar a sobreposição de aulas de um mesmo professor; minimizar a ocorrência de “janelas” no horário do professor; evitar que uma turma esteja com aula no mesmo horário com mais de um professor e outras restrições que trazem qualidade ao horário definido. Os resultados computacionais obtidos demonstraram a viabilidade de sua utilização gerando horários que atendem a todas as restrições de viabilidade e um percentual acima de 85% para as restrições de qualidade da solução.

Palavras-chave: Meta-heurística, GRASP, VND, Problema de Horário Escolar

1. INTRODUÇÃO

Segundo Schaefer (1999), o problema de programação de horários em escolas diz respeito à alocação das aulas de uma escola a um conjunto restrito de horários, satisfazendo diversas restrições. Para Bardadym (1996), a solução manual deste problema é uma tarefa penosa e complexa e normalmente requer vários dias de trabalho. Além do mais, a solução obtida pode ser insatisfatória com relação a vários aspectos. Por exemplo, em função da programação feita, pode haver aulas de uma mesma matéria sendo ministradas em dias consecutivos, com prejuízo na sedimentação da aprendizagem.

Em função de situações como essa, uma atenção especial vem sendo dada à automação do problema de programação de horários em escolas. Sendo o mesmo NP-difícil, segundo Even, Itai e Shamir (1976), ele é comumente abordado através de técnicas heurísticas, dentre as quais se destacam as chamadas meta-heurísticas. Essas técnicas, ao contrário das heurísticas convencionais, têm caráter geral e são

providas de mecanismos para escapar de ótimos locais. Como exemplos de aplicações bem sucedidas de meta-heurísticas a problemas de programação de horários, podem-se citar: *Simulated Annealing* em Abramson (1991); Busca Tabu em Costa (1994) e Burke *et al.* (2001); Programação Genética em Ueda *et al.* (2001), Carrasco & Pato (2001), Colorni, Dorigo e Maniezzo (1998) e Erben & Keppler (1996); e GRASP em Moura & Scaraficci (2010).

Em geral, no entanto, as meta-heurísticas sofrem grande influência das soluções iniciais, isto é, uma solução inicial de boa qualidade induz a um processo de busca mais rápida, com produção de soluções finais melhores. A fase de construção da meta-heurística GRASP, por Feo & Resende (1995), é um dos instrumentos atualmente utilizados em várias aplicações, segundo Souza (2000), para alcançar este objetivo.

Este artigo propõe uma heurística GRASP para o problema de programação de horário escolar, a qual utiliza em sua etapa de busca local a técnica VND. Este trabalho aborda o problema de programação de horários da Escola Agrotécnica Federal de Alegre (ES, Brasil).



O presente trabalho encontra-se organizado da seguinte maneira: na próxima seção, o problema do horário escolar abordado será descrito com maiores detalhes. Na Seção 3, é detalhada a heurística **GRASP+VND** proposta. Na Seção 4, os resultados computacionais são apresentados. Na Seção 5, são apresentadas as conclusões e, por fim, são realizados os agradecimentos e apresentadas as referências bibliográficas.

2. REVISÃO DA LITERATURA

Problemas de programação de horários são, há algumas décadas, foco de trabalhos de pesquisa. Isso ocorre devido à dificuldade de obtenção de soluções ótimas em tempos computacionais aceitáveis e também devido ao fato de instituições educacionais do mundo todo enfrentarem a difícil tarefa de gerarem boas grades de horários considerando os inúmeros requisitos pedagógicos, pessoais e institucionais envolvidos.

Na literatura existem vários trabalhos apontando possíveis soluções para o problema de programação de horários, dentre os quais, tem-se Abramson (1991); Costa (1994); Burke *et al.*(2001); Ueda *et al.*(2001); Carrasco & Pato (2001); Colorni, Dorigo e Maniezzo (1998); Erben & Keppler (1996); Borges (2003); Cowling, Kendall e Han (2010); e Moura & Scaraficci (2010).

A maioria dos trabalhos citados envolve o Ensino Superior, em que a grade horária semanal pode não ser completa, resultando em uma flexibilidade não permitida aos Colégios de Ensino Médio ou às Escolas de Ensino Fundamental. Sendo assim, para estas instituições, o problema torna-se mais complexo, pois as turmas têm aulas em todos os horários e a mudança de turma de um professor implicará na mudança de outro.

Os trabalhos que existem na literatura para o problema de programação de horário para escolas de ensino médio e fundamental, em geral, se propõem a atender um problema específico, sendo difícil adaptá-lo em outra instituição. Dentre os trabalhos que já foram feitos para geração automática de grade escolar, pode-se citar o de Gonçalves (2010), que visa a criação de um software para esse problema. O principal objetivo desse trabalho é criar um software que atenda ao sistema escolar gaúcho de ensino fundamental e médio. Além disso, ele visa ser uma alternativa gratuita para outros softwares que já foram desenvolvidos em que é necessário pagar para utilizar. E, por fim, ele tenta servir como base para outros sistemas escolares que tenham características diferentes.

Outro trabalho que também pode ser citado é o de Otimização na Alocação de Professores na Construção de uma Grade Horária Escolar, de Kotsko, Machado e Santos (2005). Foi desenvolvido um sistema em LINGO para alocação otimizada de horários das aulas de professores de uma escola da rede pública de ensino do estado do Paraná,

atendendo às suas preferências e necessidades e também às exigências administrativas e pedagógicas da escola.

Martins (2004) desenvolveu um software para facilitar, para o Cefet Campos (hoje denominado Instituto Federal de Educação, Ciência e Tecnologia Fluminense – IFF), a criação da grade escolar. Ele busca o melhor horário via emprego de algoritmos genéticos, respeitando as condicionantes.

Ciscon *et al.* (2005) prioriza a eliminação de janelas e aulas isoladas, buscando horários com mais aceitação dos professores. Para possibilitar uma solução de qualidade em tempo aceitável, ele propôs um algoritmo genético. Ele utiliza duas escolas como estudo de caso para comprovar a importância de priorizar essas restrições.

Goés (2005) desenvolveu um protótipo utilizando três algoritmos (exato, heurístico e misto) para encontrar uma solução que satisfaça a preferência de cada professor, as exigências pedagógicas e operacionais. O protótipo utilizou dados passados pelos professores da rede Municipal de Ensino de Auracária-PR.

Goés, Costa e Steiner (2010) apresentam três técnicas (modelo matemático, abordagem heurística e método misto) com o intuito de encontrar uma solução que satisfaça as exigências pedagógicas e operacionais das instituições, assim como as preferências de cada professor com relação ao dia e horários de aulas. Com o objetivo de comparar o desempenho das técnicas, foi desenvolvido um protótipo para a construção da grade de uma escola municipal da cidade de Araucária – PR.

Ferreira *et al.* (2010) desenvolveram um trabalho para automatizar e otimizar a distribuição de encargos didáticos em instituições de ensino. O modelo matemático foi baseado no caso específico do Departamento de Matemática Federal do Paraná – UFPR, só que eles esperam que possa ser adaptado para outros departamentos e instituições. Eles criaram um modelo matemático de programação linear inteira binária, que foi desenvolvido com ajuda do software LINGO. Eles também fizeram uma interface gráfica através do Visual Basic. O trabalho deles baseia-se nas restrições de turmas, da carga horária dos professores e também em atribuições de peso do professor para assumir ou não uma determinada turma.

Birbas, Daskalaki e Housos (2009) utilizam a programação inteira para o problema da geração da grade escolar. O trabalho aborda o problema em duas fases. Na primeira fase, os turnos de trabalho são atribuídos aos professores com base nas suas preferências, na política da escola sobre cursos básicos e na necessidade do currículo. Na segunda fase, é realmente tratado o problema de geração da grade escolar.



2. DESCRIÇÃO DO PROBLEMA

Para análise, considerou-se o problema de programação de horários da Escola Agrotécnica Federal de Alegre (ES, Brasil). Esta instituição oferece cursos de nível médio, técnico e superior e funciona em dois turnos, sendo que o ensino médio é cursado em concomitância com o ensino técnico e a carga horária é distribuída durante toda a semana. Considerou-se, neste trabalho, a formação de horários para o ensino médio.

De acordo com pesquisa realizada na instituição no ano de 2008, no processo de elaboração dos horários dos professores, além das restrições comumente presentes em problema de programação de horários, tais como: não ocorrer sobreposição de professores ou turmas; e fazer com que a carga-horária semanal de cada disciplina seja cumprida, vários outros requisitos devem ser observados para a melhoria do quadro de horários, tais como os descritos abaixo:

- todas as aulas devem ser simples - todas as aulas da instituição são de 60 minutos e, por decisão da área pedagógica, aulas consecutivas devem ser evitadas ao máximo;
- o número máximo de aulas diárias de uma mesma disciplina para uma mesma turma deve ser respeitado;
- a disponibilidade dos professores deve ser respeitada;
- o número de dias em que cada professor ministra aulas deve ser minimizado;
- evitar ao máximo aulas vagas, ou seja, aula no meio do turno sem disciplina alocada; e
- evitar “janelas”, ou seja, intervalos entre aulas menor ou igual a dois tempos.

3. METODOLOGIA

Para a resolução do problema de alocação de horários da Escola Agrotécnica Federal de Alegre é proposta, neste trabalho, uma heurística GRASP (*Greedy Randomized Adaptive Search Procedure*). O GRASP é um método iterativo, proposto por Feo & Resende (1995) que consiste de duas fases:

- uma fase de construção, na qual uma solução é gerada, elemento a elemento;
- uma fase de busca local, na qual um ótimo local na vizinhança da solução construída é pesquisado.

A melhor solução encontrada, ao longo de todas as iterações GRASP realizadas, é retornada como resultado.

Na heurística GRASP proposta, é utilizada, na etapa de busca local, a técnica VND (*Variable Neighborhood Descent*) proposta por Mladenovic e Hansen (1999). O VND é um método de busca local que consiste em explorar o espaço de soluções através de trocas sistemáticas de estruturas de vizinhança, aceitando somente soluções de melhora da solução corrente e retornando à primeira estrutura quando uma solução melhor é encontrada.

O pseudocódigo da heurística **GRASP+VND** proposta é descrito na Figura 1, o qual recebe como parâmetros de entrada o valor α responsável pelo grau de aleatoriedade da etapa de construção e o número total de iterações, *GRASPmax*. Como saída, o algoritmo retorna a melhor solução encontrada. Durante *GRASPmax* iterações uma solução é construída pelo algoritmo **Construção**, descrito em detalhes na Subseção 3.3, e refinada pelo algoritmo **VND**, detalhado na Subseção 3.4.

Algoritmo GRASP+VND (α , *GRASPmax*)

Entrada

α – valor de aleatoriedade;
GRASPmax – número de iterações;

Saída

s – a melhor solução encontrada;

Início

```

 $f^* \leftarrow \infty$ ;
Para  $iter \leftarrow 1$  to GRASPmax faça
     $s \leftarrow$  Construção ( $\alpha$ );
     $s \leftarrow$  VND ( $s$ );
    Se  $f(s) < f^*$  então
         $s' \leftarrow s$ ;
         $f^* \leftarrow f(s)$ ;
    fim-se;
fim-para;
 $s \leftarrow s'$ ;
Retorne  $s$ ;
Fim-GRASP+VND

```

Figura 1 – Heurística GRASP+VND proposta. Fonte própria.

Antes de apresentar os métodos de construção e de refinamento de soluções, serão apresentadas, respectivamente nas Subseções 3.1 e 3.2, a forma adotada de representação de uma solução e a função de avaliação de uma solução (função objetivo).

3.1 Representação

Uma solução (alocação) do problema é representada por uma matriz $S = (S_{ij})_{n \times m}$, em que m representa o número de horários reservados para a realização das aulas e n o número de professores. Em cada célula S_{ij} é colocado o número da turma t alocada no horário j ao professor i . Uma célula com valor zero indica que o professor j está disponível no horário i , mas não tem turma alocada neste horário. Uma célula com valor -1 indica que o professor está indisponível



PROF	SEMANA LETIVA																			
	SEGUNDA				TERÇA				QUARTA				QUINTA				SEXTA			
	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16	H17	H18	H19	H20
1	-1	-1	-1	-1	7	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	3	4	0	8	4	0	0	0	-1	-1	-1	-1	0	0	0	0	0	0	0
3	0	0	3	6	5	2	0	0	0	0	0	0	0	8	0	0	0	0	0	0
.	0	8	5	3	2	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	0	0
.	0	6	0	0	4	-1	0	0	0	0	-1	-1	-1	-1	0	0	0	0	0	0

Figura 2 – Exemplo de uma alocação

Fonte: própria

para este horário. Um exemplo desta representação é dado pela Figura 2.

3.2. Função de avaliação

Em problemas reais, para determinar a qualidade de um quadro de horários, faz-se necessário considerar diferentes objetivos que, geralmente, estão em conflito entre si. Por exemplo, deseja-se minimizar o número de dias que um professor dedica-se à sala de aula e, ao mesmo tempo, maximizar a quantidade de aulas simples (01 aula de 60 minutos). No entanto, observa-se na literatura que a maioria dos problemas de programação de horários aborda o uso de uma única função que minimiza a soma dos pesos de penalidades para avaliar as soluções e só um horário, com a mais baixa penalidade total, é retornado como resultado.

Segundo Souza (2000), para avaliar um quadro de horários, os requisitos do problema devem ser separados em subconjuntos, cada qual representando uma função objetivo linear de penalidades. É assegurada uma maior importância aos requisitos essenciais. Os requisitos essenciais são aqueles que, se não forem satisfeitos, geram um quadro de horários inviável. Consideram-se, neste trabalho, como requisitos essenciais: evitar que professores e classes estejam programados para mais de um encontro em um mesmo horário; respeitar o limite de aulas de uma disciplina para uma determinada classe; e respeito as indisponibilidades dos professores.

Os requisitos não essenciais são aqueles que buscam melhorar a qualidade do quadro de horários. No entanto, se não forem satisfeitos, não geram um quadro de horários inviável. Os requisitos não essenciais, por sua vez, são divididos em dois grupos:

- Não essenciais mais prioritários – são aqueles requisitos que, se atendidos, melhoram muito a qualidade da solução. Estão neste grupo os seguintes requisitos: evitar aulas vagas e respeitar os horários de disponibilidade dos professores.

- Não essenciais menos prioritários – são aqueles que, se atendidos, melhoram também a qualidade da solução, mas em magnitude menor do que os do grupo anterior. Estão neste grupo os seguintes requisitos: o número de dias em que cada professor ministra aulas deve ser minimizado; todas as aulas devem ser simples; e evitar “janelas”.

Neste trabalho, uma solução s é avaliada com base na seguinte função de avaliação:

$$f(s) = (\omega \times f_1(s)) + (\delta \times f_2(s)) + (\rho \times f_3(s)),$$

em que a primeira componente ($\omega \times f_1(s)$) mensura o grau de viabilidade do quadro de horários s . As duas outras mensuram a qualidade de s , em que a segunda componente ($\delta \times f_2(s)$) representa as restrições que devem ser respeitadas ao máximo (não essenciais mais prioritárias); e a terceira ($\rho \times f_3(s)$), as outras restrições (não essenciais menos prioritárias).

A função $f_1()$ é representada pelo número de vezes que um professor ministra aulas para mais de uma classe em um mesmo horário, somado ao número de vezes que uma turma tem aula com mais de um professor no mesmo horário.

A função $f_2()$ representa o número de vezes em que uma turma está sem atividades (aula vaga), somado ao número de vezes que o horário de indisponibilidade de um professor não é respeitado.

A função $f_3()$ representa o número de vezes que é violado o número máximo diário de uma disciplina para uma mesma classe, somado ao número de “janelas” e ao número de dias em que cada professor ministra aulas.

Portanto, na formação da função objetivo, foram atribuídos valores determinados pelo grau de importância de cada um dos fatores da função objetivo, sendo atribuído



um peso maior para os casos que nunca podem ocorrer, um peso médio para os casos a serem evitados ao máximo e um peso baixo para os casos a serem evitados se possível. Os valores utilizados neste trabalho, arbitrados empiricamente, são mostrados na Tabela 1.

Tabela 1 – Pesos da função objetivo.

Fator	Peso
ω	500
δ	200
ρ	1

Fonte própria.

3.3. Geração da solução inicial

Uma solução inicial para o problema de alocação de aulas abordado é gerada por um procedimento construtivo guloso aleatorizado que segue o modelo do método GRASP descrito por Feo & Resende (1995).

A ideia do algoritmo construtivo desenvolvido neste trabalho é priorizar a alocação dos professores com menor folga – neste trabalho, entende-se por folga do professor p a diferença entre o total de horários disponibilizado pelo professor p e o número de horários deste professor a serem preenchidos por disciplinas alocadas a ele. A intenção aqui é evitar ao máximo alocar um professor em um horário que não esteja na sua disponibilidade. É criada então uma lista de professores, LP , ordenada crescentemente pela folga destes. A lista restrita de candidatos (LRC) é formada pelos $a\%$ primeiros elementos de LP . A lista LRC é importantíssima para uma heurística GRASP, pois ela permite que diferentes soluções iniciais sejam geradas a cada iteração. O próximo professor a ser alocado é escolhido aleatoriamente de LRC , o qual recebe uma turma. Esse procedimento é repetido até que todas as aulas sejam alocadas. Na Figura 3, é apresentado o pseudocódigo do procedimento de construção, o qual recebe como parâmetro de entrada o percentual de aleatoriedade a e gera como saída uma solução s construída. Este algoritmo não garante a construção de uma solução viável; o que ele tenta é minimizar o número de violações das restrições impostas para a formação do horário.

Algoritmo Construção (α)

Entrada
 α – percentual de aleatoriedade;

Saída
 s – solução construída;

Início

1. Criar uma lista de professores (LP) ordenada crescentemente pela folga.
2. Definir a lista restrita de candidato (LRC) como sendo os $\alpha\%$ primeiros elementos de LP ; Escolher aleatoriamente um professor p em LRC ;
3. Alocar aleatoriamente uma turma ao professor p , respeitando as restrições do problema;
4. Quando não for possível satisfazer as restrições, alocar turmas aos professores com horário disponível;
5. Repetir o processo até preencher todo o horário;

Fim-Construção

Figura 3 - Algoritmo Construção
 Fonte própria

3.4. Método VND proposto

O Método VND proposto utiliza três estruturas de vizinhança: “Troca Horários do Professor no Mesmo Dia”; “Troca Horários do Professor em Dias Diferentes”; e “Troca de Turmas Entre Professores”, as quais serão detalhadas, respectivamente, nas Subseções 3.4.1, 3.4.2 e 3.4.3. A ordem de análise das vizinhanças foi definida empiricamente. O pseudocódigo do método VND proposto é apresentado na Figura 4, o qual recebe como parâmetros de entrada a solução s_0 a ser refinada e as três estruturas de vizinhança. Como saída, o algoritmo retorna a solução s (solução obtida após o refinamento).

3.4.1 Vizinhança “Troca Horários do Professor no Mesmo Dia”

Este tipo de estrutura realiza, em cada linha i da matriz (solução) s , a troca de dois valores distintos e não negativos no mesmo dia. Esse movimento é identificado pela tripla $\langle i, k_1, k_2 \rangle$, na qual k_1 e k_2 representam as turmas que têm aula com o professor i nos horários Hk_1 e Hk_2 , que serão permutados respeitando a condição de trocas no mesmo dia. Um exemplo de uma vizinhança é dado na Figura 5, gerada a partir da tabela da Figura 2. Neste exemplo, as turmas presentes nos horários H5 e H6, destacados na Figura 5, do professor 1 são trocadas.

Algoritmo VND (S_0, N_1, N_2, N_3)

Entrada
 s_0 – solução inicial;
 N_1 – “Troca Horários do Professor no Mesmo Dia”;
 N_2 – “Troca Horários do Professor em Dias Diferentes”;
 N_3 – “Troca de Turmas Entre Professores”.

Saída
 s – solução refinada;

Início
 $s \leftarrow s_0$; {Solução corrente}
 $k \leftarrow 1$; {Tipo de estrutura de vizinhança}

Enquanto ($k \leq 3$) faça
 Encontre o melhor vizinho $s' \in N^{(k)}(s)$
Se $f(s') < f(s)$ então
 $s \leftarrow s'$;
 $k \leftarrow 1$;
Senão
 $k \leftarrow k+1$;
fim-se;
fim-enquanto;

Fim-VND

Figura 4 - Algoritmo VND
 Fonte: própria

	SEGUNDA				TERÇA				QUARTA			
PROF	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12
1	-1	-1	-1	-1	0	7	0	0	0	0	0	0

Figura 5 – Movimento de troca entre H5 e H6.
 Fonte própria.



3.4.2 Vizinhança “Troca Horários do Professor em Dias Diferentes”

Essa estrutura realiza, em cada linha i da matriz s , a troca de dois valores distintos e não negativos durante toda a semana. Esse movimento é identificado pela tripla $\langle i, k_1, k_2 \rangle$, na qual k_1 e k_2 representam as turmas que têm aula com o professor i nos horários Hk_1 e Hk_2 que serão permutadas. Um exemplo é dado pela Figura 6, gerada a partir da tabela da Figura 2. Neste exemplo, as turmas presentes nos horários H5 e H13, destacados na Figura 6, do professor 1 são trocadas.

3.4.3 Vizinhança “Troca Turmas Entre Professores”

Esta vizinhança é alcançada por trocas de turmas entre professores que ministram aulas nas mesmas turmas. A Figura 7 mostra um movimento nesta vizinhança em relação à Figura 2. Neste exemplo os professores 1 e 3, destacados na Figura 7, trocarão de horário na turma 8.

4. RESULTADOS COMPUTACIONAIS

O algoritmo foi implementado na linguagem C usando o compilador Dev-C++ 4.9.9.2 e testado em um microcomputador PC PENTIUM IV, 3.0 GHz com 512 MB de memória RAM, sob o sistema operacional Windows XP SP2.

Para testar o algoritmo, foram utilizados dados da Escola Agrotécnica Federal de Alegre (ES, Brasil) relativos aos anos de 2006 e 2007 e outras instâncias geradas neste trabalho para comprovar a eficiência do algoritmo. A partir de definições dadas pela coordenação de ensino, os horários foram gerados pela heurística **GRASP+VND** e os resultados foram comparados com os horários gerados manualmente, aplicados em cada período mencionado. Na Figura 8, encontram-se as instâncias utilizadas, sendo que a instância “50% 2007” e “100% 2007” foram geradas aleatoriamente, aumentando-se o número de professores e de turmas em cinquenta e cem por cento em relação ao horário de 2007.

	SEGUNDA				TERÇA				QUARTA				QUINTA				SEXTA			
PROF	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16	H17	H18	H19	H20
1	-1	-1	-1	-1	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0

Figura 6 – Movimento de troca entre H5 e H13

Fonte própria

	SEGUNDA				TERÇA				QUARTA				QUINTA				SEXTA			
PROF	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16	H17	H18	H19	H20
1	-1	-1	-1	-1	7	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0
2	0	3	4	0	8	4	0	0	0	-1	-1	-1	-1	0	0	0	0	0	0	0
3	0	0	3	6	5	2	8	0	0	0	0	0	0	0	0	0	0	0	0	0

Figura 7 – Movimento de troca entre Prof. 1 e 3 nos horários H7 e H14

Fonte própria

Instância \ Composição	2006	2007	50% 2007	100% 2007
Professores	19	19	27	38
Disciplinas	32	32	32	32
Turmas	12	12	18	24

Figura 8 – Instâncias utilizadas

Fonte própria



Algoritmo Instância	GRASP1		GRASP2		GRASP3		GRASP+VND	
	f^*	Tempo(s)	f^*	Tempo(s)	f^*	Tempo(s)	f^*	Tempo(s)
2006	3099	23	22	1261	3654	4	20	1289
2007	2787	21	25	1265	3080	4	23	1342
50% 2007	6060	33	50	2133	4744	3	40	2170
100% 2007	8560	36	59	2845	6695	3	47	3013

Figura 9 – Resultados Computacionais

Fonte própria

Algoritmo Instância	GRASP1	GRASP2	GRASP3	GRASP+VND
	f^*	f^*	f^*	f^*
2006	1573	24	1274	20
2007	1876	24	2027	23
50% 2007	3870	44	3836	39
100% 2007	5117	67	4933	49

Figura 10- Testes realizados com execução de 30 minutos

Fonte própria

Foram realizados testes com quatro configurações do algoritmo, os quais utilizam o algoritmo **Construção**, descrito na Subseção 3.3, e se diferenciam pelo método de busca local utilizado:

- **GRASP1** – investiga na etapa de busca local a vizinhança “Troca Horários do Professor no Mesmo Dia”;
- **GRASP2** – investiga na etapa de busca local a vizinhança “Troca Horários do Professor em Dias Diferentes”;
- **GRASP3** – investiga na etapa de busca local a vizinhança “Troca Turmas Entre Professores”; e
- **GRASP+VND** – utiliza o algoritmo **VND**, descrito na Subseção 3.4, na etapa de busca local.

Para verificação dos resultados computacionais, executaram-se dois conjuntos de testes e os resultados encontram-se nas Figuras 9 e 10. A Figura 9 contém os dados obtidos (valor da função objetivo, f^* , e o tempo gasto em segundos) a partir de um valor fixo de iterações (20 iterações) e os dados da Figura 10 foram obtidos com a execução de todos os algoritmos por um tempo fixo de 30 minutos. É importante destacar que este tempo foi definido, pois, na prática, esperar 30 minutos por uma solução de qualidade é adequado para este problema. No entanto, foi verificado nos experimentos realizados que boas soluções podem ser obtidas em tempos menores.

Apresenta-se nas Figuras 11 e 12 a discriminação das quantidades das restrições não atendidas em relação ao

valor da função objetivo de cada teste realizado com o algoritmo **GRASP+VND** proposto.

Instâncias f^*	2006	2007	50% de 2007	100% de 2007
	f^*	20	23	40
Excesso Aulas	17	18	35	38
Janelas	3	5	5	9

Figura 11 - Quantidade Restrições não Atendidas – Teste 20

Iterações GRASP+VND. Fonte própria.

Instâncias f^*	2006	2007	50% de 2007	100% de 2007
	f^*	20	23	39
Excesso Aulas	16	19	32	39
Janelas	4	4	7	10

Figura 12 - Quantidade Restrições não Atendidas – Teste 30 Minutos

GRASP+VND. Fonte própria.

Em todas as instâncias utilizadas, observa-se que o algoritmo **GRASP+VND** apresentou os melhores resultados com relação ao valor da função objetivo (f^*). Todos os valores de f^* , neste caso, são de não atendimento à condição excesso de aulas e “janelas”.

Chegou-se sempre a horários viáveis com percentuais de qualidade superiores a 85%, calculando-se a relação



entre o total de restrições não atendidas e o total de aulas programadas. Esse valor foi considerado, pela coordenação pedagógica da instituição, bastante satisfatório quando comparado aos horários gerados manualmente.

5. CONCLUSÕES

De acordo com os testes realizados, a heurística **GRASP+VND** revelou-se uma boa solução para o problema de programação do horário escolar da Escola Agrotécnica Federal de Alegre (ES, Brasil) pela qualidade dos resultados obtidos quando comparados aos horários aplicados nos períodos considerados para validação da solução. De acordo com a coordenação pedagógica da instituição, os resultados obtidos pela heurística **GRASP+VND** são de 15 a 20% melhores do que os alcançados manualmente (após dias de trabalho).

Em todos os testes, o horário gerado não atendeu a apenas dois tipos de restrições não essenciais e de baixa prioridade. Oitenta e quatro por cento dessas restrições referem-se a excesso de aulas, ou seja, mais de uma aula do professor da mesma disciplina no mesmo dia e os dezesseis por cento restantes são referentes a intervalos entre aulas menor ou igual a dois tempos (“janelas”).

6. AGRADECIMENTOS

Este trabalho foi financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), pela Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ), pelo Parque de Alta Tecnologia do Norte Fluminense (TECNORTE) e pela Fundação Estadual do Norte Fluminense (FENORTE).

7. REFERÊNCIAS

ABRAMSON, D. Constructing School Timetables Using Simulated Annealing: Sequential and Parallel Algorithms. **Management Science**, v. 37, pp. 98-113, 1991.

BARDADYM, V. A. Computer-Aided School and University Timetabling: The New Wave. **Lecture Notes in Computer Science**, v. 1153, pp. 22-45, 1996.

BIRBAS, T., DASKALAKI, S., e HOUSOS, E. D. School timetabling for quality student and teacher schedules. **Journal of Scheduling**, v. 12, n. 2, Abril 2009.

BORGES, S. K. **Resolução de timetabling utilizando algoritmos genéticos e evolução cooperativa**. Curitiba, 2003. 79 f. Dissertação (Mestrado em Ciência da Computação) – Setor de Ciências Exatas, Universidade Federal do Paraná.

CARRASCO, M. P. e PATO, M. V. A Comparison of Discrete and Continuous neural Network Approaches to Solve the Class/Teacher Timetabling Problem. **European Journal of Operational Research**, v. 153, n. 1, pp. 65-79, 2001.

CISCON, L. A. et al. O Problema de Geração de Horários: um Foco na Eliminação de Janelas e Aulas Isoladas. **XXXVIII Simpósio Brasileiro de Pesquisa Operacional**. Gramado - RS, 2005.

COLORNI, A., DORIGO M., e MANIEZZO V. Methaheuristics for high school timetabling. **Computational Optimization and Applications**, v. 9, n. 3, pp.275-298,1998.

COSTA, D. A tabu search algorithm for computing an operational timetable. **European Journal of Operational Research**, v. 76, pp.98-110, 1994.

COWLING, P., KENDALL, G., e HAN, L. **An investigation of a hyperheuristic genetic algorithm applied to a trainer scheduling problem**. Disponível em: <<http://www.cs.nott.ac.uk/~gkx/papers/cec2002lxh.pdf>> Acessado em: 22 de dez. 2010.

ERBEN, W. e KEPPLER, J. A genetic Algorithm Solving a Weekly Course-Timetabling Problem. **Lecture Notes in Computer Science**, v. 1408, pp.37-52, 1996.

EVEN, S., ITAI, A. e SHAMIR, A. On the complexity of timetabling and multicommodity flow problems. **SIAM Journal of Computation**, v. 5, pp.691-703, 1976.

FEO, T. A. e RESENDE, M. G. C. Greedy randomized adaptive search procedures. **Journal of global optimization**, v. 6, pp.109-133, 1995.

FERREIRA, P. S., RIBEIRO, A. A., KARAS, E. W., e SILVA, A. L. Aplicação de Programação Inteira na Distribuição de Encargos Didáticos em Instituições de Ensino. **XXIII Congresso Nacional de Matemática Aplicada e Computacional**. Águas de Lindóia – SP, 2010.

GOÉS, A. R. T. **Otimização na Distribuição da Carga Horária de Professores – Método Exato, Método Heurístico, Método Misto e Interface**. 2005. Dissertação (Programa de Pós-Graduação em Métodos Numéricos em Engenharia – Programação Matemática, Setores de Tecnologia e de Ciências Exatas) - Universidade Federal do Paraná. Curitiba, 2005.

GOÉS, A. R. T., COSTA, D. M. B., e STEINER, M. T. A. Otimização na Programação de Horários de Professores/Turmas: Modelo Matemático, Abordagem Heurística e Método Misto. **Revista Eletrônica Sistemas & Gestão**, v. 5, n. 1, pp. 50-66, 2010.

GONÇALVES, T. S. **Software para Organização de Horários Escolares**. Anteprojeto de Dissertação (Sistemas de Informação) – Instituto de Ciências Exatas e Tecnológicas - Universidade FEEVALE, 2010.

KOTSKO, E. G. S., MACHADO, A. L. F., e SANTOS, E. M. Otimização na Alocação de Professores na Construção de uma Grade Horária Escolar. **Revista do Centro de Ciências Agrárias e Ambientais**, v. 1, n. 1, Jan/Jun. 2005.



MARTINS, S. V. Melhor Horário para o Sistema Cefet Campos. **Vértices**, v. 6, n. 1, jan./abr, 2004.

MLADENOVIC, N. e HANSEN, P. Variable Neighborhood Search. **Computers and Operations Research**, v. 24, pp. 1097-1100, 1999.

MOURA, A. V.; SCARAFICCI, R. A. A GRASP strategy for a more constrained school timetabling problem. **International Journal of Operational Research**, v. 7, pp. 152-170, 2010.

SCHAEFER, A. A survey of automated timetabling. **Artificial Intelligence Review**, 13:87-127, 1999.

SOUZA, M.J.F. **Programação de Horários em Escolas: Uma aproximação por Metaheurísticas**. Tese de Doutorado. Universidade Federal do Rio de Janeiro. Rio de Janeiro, Brasil, 2000.

UEDA, H., OUCHI, D., TAKAHASHI, K.; MIYAHARA, T. A Co-evolving Timeslot/Room Assignment Genetic Algorithm Technique for Universities Timetabling. **Lecture Notes in Computer Science**, v. 2079, pp. 48-63, 2001.



A GRASP+VND HEURISTIC FOR THE SCHOOL TIMETABLING PROBLEM

Abstract

This work proposes an hybrid GRASP metaheuristic using, in the local search phase, the VND technique to solve the timetabling problem. It is considered to review data of Escola Agrotécnica Federal of Alegre-ES-Brazil in the years of 2006 and 2007 and other instances generated at this work for testing, for scheduling two rounds of education high school. The method proposed intends to satisfy the restrictions of timetable of the institution, such as: to meet teachers on the amount of days that their classes are distributed within hours of its availability; to minimize the number of consecutive classes in the same discipline; to prevent the occurrence of teaching vacancies; to minimize the occurrence of “windows” at the timetable of the teacher; to avoid overlapping of classes from the same teacher; to avoid that exists a classroom lecture with the same time with more than one teacher and other restrictions that bring quality to the timetable. The computational results demonstrated the feasibility of its use generates timetable that serve all restrictions of feasibility and a percentage above 85% for restrictions of quality of the solution.

Keywords: Metaheuristics, GRASP, VND, School timetabling problem
