

UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA
CAMPUS DE ILHA SOLTEIRA

ODILON NOVAES SILVA

PROGRAMAÇÃO DE HORÁRIOS USANDO UM ALGORITMO DE BUSCA EM
VIZINHANÇA VARIÁVEL

Ilha Solteira - SP
2013

ODILON NOVAES SILVA

PROGRAMAÇÃO DE HORÁRIOS USANDO UM ALGORITMO DE BUSCA EM
VIZINHANÇA VARIÁVEL

Dissertação apresentada à Faculdade de Engenharia do Campus de Ilha Solteira - UNESP como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica.

Prof. Dr. Rubén Augusto Romero Lázaro
Orientador

Ilha Solteira - SP
2013

FICHA CATALOGRÁFICA
Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

S586p Silva, Odilon Novaes.
Programação de horários usando um algoritmo de busca em vizinhança variável : programação de horários / Odilon Novaes Silva. -- Ilha Solteira: [s.n.], 2013
132 f.

Dissertação (mestrado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2013

Orientador: Ruben Augusto Romero Lázaro
Inclui bibliografia

1. Algoritmo VNS. 2. Programação de horários. 3. Busca em vizinhança Variável. 4. Algoritmo BVNS. 5. Algoritmo GVNS.



UNIVERSIDADE ESTADUAL PAULISTA
CAMPUS DE ILHA SOLTEIRA
FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA

CERTIFICADO DE APROVAÇÃO

TÍTULO: PROGRAMAÇÃO DE HORÁRIOS USANDO UM ALGORITMO DE BUSCA EM VIZINHANÇA VARIÁVEL

AUTOR: ODILON NOVAES SILVA

ORIENTADOR: Prof. Dr. RUBEN AUGUSTO ROMERO LAZARO

Aprovado como parte das exigências para obtenção do Título de Mestre em Engenharia Elétrica ,
Área: AUTOMAÇÃO, pela Comissão Examinadora:

Prof. Dr. RUBEN AUGUSTO ROMERO LAZARO
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira

Profa. Dra. MARINA LAVORATO DE OLIVEIRA
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira

Prof. Dr. CARLOS ROBERTO MENDONÇA DA ROCHA
Centro de Engenharias e Ciências Exatas / Universidade Estadual do Oeste do Paraná

Data da realização: 20 de dezembro de 2013.

DEDICO

À minha esposa, Cristina, às minhas filhas: Lorena, Milena e Patrícia. E aos meus pais: Odilon e Alvina que me educaram e me possibilitaram mais essa conquista, exemplos de vida fundamentais para minha vida pessoal e profissional.

AGRADECIMENTOS

- *A Deus por ter me dado a saúde e a fé necessários para a conclusão de mais uma jornada.*
- *Aos meus pais, Odilon e Alvina, que durante a passagem abreviada dessa vida sempre me apoiaram em todos os momentos. Agradeço por todo o amor e carinho.*
- *À minha esposa Cristina e minhas filhas Lorena, Milena e Patrícia, pelo incentivo, carinho, paciência e compreensão de sempre, fundamentais para o meu envolvimento nesse trabalho.*
- *À minha sobrinha Suze, símbolo de competência e generosidade. Pelas horas dedicadas à concretização deste trabalho meus sinceros agradecimentos.*
- *Aos meus sobrinhos Beto e Gilmar Jr., pelo espírito de amizade e companheirismo.*
- *Ao compadre Gilmar pelo incentivo e pelo envolvimento na concretização deste sonho, meus sinceros agradecimentos.*
- *A todos os meus irmãos, irmãs, cunhadas, cunhados, sobrinhas e sobrinhos meus sinceros agradecimentos pelo carinho e pelas palavras de incentivo.*
- *Agradeço em especial ao prof. Ruben que abriu as portas e acolheu-me. A este orientador/amigo expresso minha profunda gratidão pelo incentivo, envolvimento, conselhos e críticas, os quais contribuíram para o meu crescimento pessoal e profissional, obrigado por acreditar em mim.*
- *Aos professores Sérgio e Ana Diva, pelas brilhantes considerações e críticas, meus sinceros agradecimentos.*
- *Em nome do prof. Marcelo Mioto meu muito obrigado a todos os professores da UNESP que contribuíram com esta conquista.*
- *Não poderia deixar de lembrar do amigo/irmão Evandro pelas palavras de incentivo.*
- *Aos coordenadores e colegas da UNIC-Universidade de Cuiabá, em especial ao diretor Vitalino Pires.*
- *À Escola Presidente Médici, em especial à diretora Maria Antônia e às coordenadoras Sueli, Kátia, Cilene e Regina.*
- *À amiga Vera Marce expresso minha eterna gratidão.*

- *Meus sinceros agradecimentos aos amigos, em especial aos que acompanharam de perto as dificuldades e realizações nesse tempo: Edgar, Marcos e Ivo. Meus companheiros do grupo “Cuiabá forte em Ilha” e a todos que conquistei no mestrado, vai minha eterna gratidão.*

- *A todos aqueles que com um gesto ou palavra contribuíram para a realização deste trabalho e cujos nomes não aparecem agradeço de igual modo e perdoem-me pelo meu esquecimento.*

Sinceramente,

Odilon Novaes Silva

20 de dezembro de 2013.

“Que os vossos esforços desafiem as
impossibilidades, lembrai-vos de que
as grandes coisas do homem foram
conquistadas do que parecia
impossível.”

Charles Chaplin

RESUMO

Por se tratar de uma tarefa complexa, as instituições passaram a recorrer a diversas metaheurísticas no intuito de resolver um problema árduo e complexo que é a elaboração de grade horária. No Brasil, com o advento do desenvolvimento da microinformática a partir da década de 90, do século XX, esse problema foi tratado com o uso de ferramentas de programação linear e métodos matemáticos baseados em otimização clássica. Posteriormente, passou a ser executado pelas universidades públicas e privadas a partir de propostas computacionais, desenvolvidas para a resolução desse tipo de problema, usando técnicas fundamentadas no uso de metaheurísticas. O presente trabalho visa projetar e implementar computacionalmente um algoritmo tipo VNS (do inglês *Variable Neighborhood Search*) para resolver o problema de programação de horários em ambientes universitários; realizar uma análise teórica e experimental do desempenho do algoritmo VNS e discutir a aplicação desse algoritmo na otimização de outros problemas da família de problemas do tipo *timetabling*. Para isso foi desenvolvido um algoritmo de busca em vizinhança variável para resolver um tipo de problema da família *timetabling* em ambientes universitários.

Palavra-chave – Problema de programação de horários. Algoritmo de busca em vizinhança variável tipo VNS.

ABSTRACT

Building of timetables is a hard work to accomplish due to its complexity, so that institutions started to make use of several Metaheuristics for solving timetabling problems. In Brazil, the development of Computer Science from the Nineties, within the late 20th century, allowed to handle this kind of problem with Linear Programming Tools and Mathematical Methods based upon Classical Optimization Techniques. Afterwards, the same task was carried out by private and public universities using computational proposals based upon Metaheuristics. The aim of this work is to project and implement VNS algorithm computationally to solve timetabling problems in university environments; to provide a theoretical and experimental analysis of the VNS algorithm performance and discuss its application in order to optimize any other type of *timetabling family* problems. Thus, an algorithm of variable neighborhood search was developed for solving problems of *timetabling* family in university environments.

Keywords: Timetable programming problem. VNS algorithm (Variable Neighborhood Search Algorithm).

LISTA DE TABELAS

Tabela 1	- Exemplo ilustrativo de Proposta de solução nº 01(Vetorial)	56
Tabela 2	- Exemplo ilustrativo de Proposta de solução nº 02 (Matricial) - Factível	58
Tabela 3.1	- Proposta de solução com dados de entrada de um exemplo ilustrativo do cap. 1 (Evento x nº de alunos)	59
Tabela 3.2	- Proposta de solução com dados de entrada de um exemplo ilustrativo do cap. 1 (Evento x nº de salas)	59
Tabela 4	- Proposta de solução nº 02 (Matricial) - Infactível	69
Tabela 5	- Dados dos sistemas testados - Instâncias	73
Tabela 6	- Melhores resultados encontrados	74
Tabela 7	- Testes com a instância nº II	74
Tabela 8	- Instância nº 1 do ITC (Competição Internacional de Timetabling)	84
Tabela 9	- Instância nº 2 do ITC (Competição Internacional de Timetabling)	94
Tabela 10	- Instância nº 3 do ITC (Competição Internacional de Timetabling)	104
Tabela 11	- Instância nº 4 do ITC (Competição Internacional de Timetabling)	114

SUMÁRIO

1	INTRODUÇÃO	13
1.1	O PROBLEMA DE PROGRAMAÇÃO DE GRADE HORÁRIA	13
1.2	DETALHAMENTO DAS CARACTERÍSTICAS ESPECÍFICAS DO PROBLEMA DE PROGRAMAÇÃO DE GRADE HORÁRIA	13
1.3	DESENVOLVIMENTO DE UM MODELO MATEMÁTICO	17
1.4	METAHEURÍSTICAS USADAS NO PROBLEMA DA GRADE HORÁRIA	20
1.5	CACTERÍSTICAS DE UM PROBLEMA DE HORÁRIO	21
1.6	MÉTODOS DE PRODUÇÃO AUTOMATIZADA DE HORÁRIOS PARA INSTITUIÇÕES EDUCACIONAIS	24
1.6.1	Busca Local	24
1.6.2	Busca Tabu	24
1.6.3	Simulated Annealing	26
1.6.4	Algoritmos Evolutivos	26
1.6.5	Métodos Baseados em Programação Linear	27
1.6.6	Comparações entre os métodos	28
2	A METAHEURÍSTICA DE BUSCA EM VIZINHANÇA VARIÁVEL	29
2.1	INTRODUÇÃO SOBRE AS HEURÍSTICAS	29
2.1.1	O Algoritmo Heurístico Construtivo	30
2.1.2	O Algoritmo Heurístico de Busca Através de Vizinhança	31
2.1.3	Introdução sobre as Metaheurísticas	32
2.1.4	Simulated Annealing	32
2.1.5	Tabu Search – Busca Tabu	35
2.1.6	O Algoritmo Genético	38
2.1.7	A Metaheurística GRASP	41
2.1.7.1	<i>Teoria Básica de GRASP</i>	<i>41</i>
2.1.7.2	<i>A Fase de Busca Local do GRASP.</i>	<i>44</i>
2.2	O ALGORÍTMO DE BUSCA EM VIZINHANÇA VARIÁVEL	45
2.3	DIFERENTES ESTRUTURAS DE BUSCA EM VIZINHANÇA VARIÁVEL	47
2.3.1	Algoritmo VND	47
2.3.2	Algoritmo RVNS	49

2.3.3	Algoritmo BVNS	51
2.3.4	Algoritmo GVNS	52
3	ALGORITMO VNS APLICADO AO PROBLEMA DE GRADE HORÁRIA	55
3.1	A FORMATAÇÃO DOS DADOS DO PROBLEMA	55
3.2	A REPRESENTAÇÃO DE UMA PROPOSTA DE SOLUÇÃO	55
3.3	A FUNÇÃO OBJETIVO DE UMA PROPOSTA DE SOLUÇÃO	57
3.3.1	Determinação da função objetivo de uma proposta de solução	59
3.3.2	Escolha de matrizes auxiliares	64
3.4	PROPOSTA PARA GERAR UMA SOLUÇÃO INICIAL	66
3.5	CARACTERIZAÇÃO DAS ESTRUTURAS DE VIZINHANÇA	70
3.6	ESTRUTURA GERAL DO ALGORITMO VNS	72
3.7	TESTES E RESULTADOS	73
3.8	CONCLUSÕES PARCIAIS	75
4	CONCLUSÕES E TRABALHOS FUTUROS	76
	REFERÊNCIAS	77
	ANEXO	84

1 INTRODUÇÃO

1.1 O PROBLEMA DE PROGRAMAÇÃO DE GRADE HORÁRIA

O problema de programação ótima de horários em um ambiente universitário é um problema altamente relevante porque se relaciona à programação ótima das atividades nas universidades, assim como o uso adequado dos recursos disponíveis. Além disso, uma solução apresentada para esse tipo de problema também pode ser adaptada a outras tarefas de programação de atividades em empresas e outros centros acadêmicos em razão dos níveis de mobilização que esse problema enseja: a) organizacional; b) pedagógico; c) pessoal, dos quais trataremos detidamente ao longo dos capítulos. De acordo com a literatura específica, esse tipo de problema é conhecido como *timetabling*.

O problema da programação ótima de horários é um problema altamente complexo de resolver para o caso de sistemas reais. A maioria desses tipos de problemas pode ser modelada matematicamente como problemas de programação linear inteiro. Entretanto, para sistemas reais, esse tipo de problema exige tempo de processamento proibitivo. Por esse motivo, na literatura especializada foram apresentadas muitas heurísticas e metaheurísticas para resolver esse tipo de problema.

Nesta dissertação buscamos desenvolver um algoritmo de busca em vizinhança variável (VNS do inglês *Variable Neighborhood Search*) para resolver um tipo de problema da família *timetabling*, isto é, o problema de programação de horários em ambientes universitários.

1.2 DETALHAMENTO DAS CARACTERÍSTICAS ESPECÍFICAS DO PROBLEMA DE PROGRAMAÇÃO DE GRADE HORÁRIA

Nos capítulos seguintes apresentaremos as possíveis soluções para um tipo de problema da família *timetabling*, isto é, trata-se do problema de programação de horários em ambientes universitários.

O problema de criação de uma grade horária válida envolve a alocação de turmas, professores, disciplinas e salas, de tal forma que não sejam alocados mais de uma vez por período.

A seguir, apresentamos matematicamente os dados que se relacionam a esse tipo de problema: uma grade horária disponível (usamos o padrão internacional que separa as atividades em nove horas por dia e cinco dias por semana, o que significa que existem 45 horas disponíveis na grade horária), os eventos programados (que pode representar uma aula de uma disciplina) com as respectivas características de tipo de aula exigida, as salas disponíveis com a informação da capacidade e as características dessas salas e a relação de alunos com a informação dos eventos em que pretende ser matriculado. O resultado final é a informação da programação de cada evento indicando a sala escolhida para o evento e o horário. De forma complementar, existe disponível a grade horária de cada estudante.

Para ilustração do problema consideramos a programação de 4 eventos em apenas um dia, 2 salas disponíveis e 5 alunos. Nesse contexto, os dados foram os seguintes:

- Horas disponíveis para a programação dos eventos (disciplinas). Em problemas reais os eventos são programados em 5 dias e com 9 horas por dia. No exemplo ilustrativo escolhemos apenas 9 horas de um dia para realizar a programação.
- Uma matriz que relaciona os eventos (as disciplinas) com as características necessárias para a realização desse evento. Essa matriz, que chamaremos de matriz A , para o exemplo ilustrativo, assumirá a seguinte forma:

Matriz (eventos X características)

		características			
		1	2	3	
$A =$	1	1	0	0	1
	2	0	1	0	2
	3	0	1	1	3
	4	0	1	0	4

eventos

Na informação **características** aparece informações relacionadas às características da sala necessária à realização do evento (disciplina), a sala de aula, por exemplo, e a disponibilidade de equipamentos de multimídia ou laboratório).

Analisando a matriz A observamos que, por exemplo, o evento 1 precisará de um ambiente que atenda à característica 1 e o evento 3 precisará de um ambiente que atenda às características 2 e 3. Assim, se o valor de $A(i; j) = 1$, significa dizer que o evento i precisará de um ambiente que atenda com a característica j . Caso esse valor seja igual a zero, então o evento i poderá ser programado em um ambiente que não atenda com a característica j .

- Uma matriz que relaciona as salas (ambientes) com as características dessas salas. Essa matriz, que chamaremos de matriz B , para o exemplo ilustrativo, assumirá a seguinte forma:

Matriz (salas de aula X características)

$$B = \begin{array}{c} \begin{array}{ccccc} & \text{características} & & & \\ & 1 & 2 & 3 & \\ \begin{array}{|c|c|c|} \hline 1 & 1 & 0 \\ \hline 0 & 1 & 1 \\ \hline \end{array} & \begin{array}{l} 1 \\ 2 \end{array} & \text{salas de aula} \end{array}$$

Na informação de características aparecerão informações relacionadas com as características da sala, por exemplo, sala de aula, disponibilidade de equipamento de multimídia ou laboratório. Dever-se-á observar que as características exigidas pelo evento devem estar disponíveis na sala escolhida para esse evento. Obviamente, a sala escolhida para um evento poderá ter características adicionais ao mínimo exigido pelo evento. Por exemplo, um evento que exija uma sala de aula simples pode ser alocado em uma sala de aula com multimídia.

Analisando a matriz B poderemos verificar que a sala 2 possui (está equipada) com as características 2 e 3. Assim, se o valor de $B(i; j) = 1$ significa que a sala i possui a característica j . Se o valor for zero, então a sala i não possui a característica j .

- Um vetor que informa sobre a capacidade (número de cadeiras na sala ou ambiente) das salas. Esse vetor, que chamaremos de vetor C , para o exemplo ilustrativo, assumirá a seguinte forma:

Vetor capacidade das salas

$$C = \begin{array}{c} \begin{array}{cc} \text{capacidade das salas} \\ 1 & 2 \\ \begin{array}{|c|c|} \hline 3 & 4 \\ \hline \end{array} \end{array} \text{salas}$$

Analisando o vetor C poderemos concluir que a sala 2 tem capacidade para acomodar 4 alunos. Assim, se o valor de $C(k) = p$ então a sala k terá capacidade para atender p alunos.

- Uma matriz que relaciona os eventos (as disciplinas) com os estudantes. Essa matriz, que chamaremos de matriz D , para o exemplo ilustrativo, assumirá a seguinte forma:

Matriz (eventos X estudantes)

		estudantes						
		1	2	3	4	5		
D =	1	1	0	1	0	0	1	eventos
	2	1	1	0	1	0	2	
	3	0	1	0	1	1	3	
	4	0	1	1	1	1	4	

Analizando a matriz D poderemos concluir, por exemplo, que o estudante 1 pretende participar dos eventos 1 e 2, e o estudante 2 pretende participar dos eventos 2, 3 e 4. Assim, se o valor de $D(i; j) = 1$ significa que o estudante j pretende participar do evento i .

A estrutura de dados mostrada anteriormente pode ser compactada como se mostra no apêndice. Nesse caso aparecem apenas dois arranjos dos dados, isto é, dados que relacionam os eventos com os alunos que pretendem participar desses eventos e dados que relacionam os eventos com as salas em que podem ser ministrados esses eventos.

No problema de programação de horários existem dois tipos de restrições: (i) restrições obrigatórias que devem ser necessariamente cumpridas por todas as propostas de soluções e, (ii) restrições não obrigatórias, mas desejáveis que não estão obrigadas a ser cumpridas por toda proposta de solução. Assim, entre todas as propostas de solução que cumprem com as restrições obrigatórias, a melhor delas é aquela proposta de solução que tiver o menor número de restrições não obrigatórias violadas.

(i) Neste trabalho consideramos as seguintes restrições obrigatórias (restrições rígidas):

- Dois ou mais eventos não podem ser programados na mesma sala e no mesmo horário. A esse tipo de restrição chamaremos de *RO1*.
- A sala programada para um evento deve ter as características requeridas pelo evento e a capacidade suficiente para acomodar os estudantes matriculados no evento (disciplina). A esse tipo de restrição chamaremos de *RO2*.
- Um estudante não poderá ser programado em mais de um evento no mesmo horário. A esse tipo de restrição chamaremos de *RO3*.

(ii) Por sua vez, consideraremos neste trabalho as seguintes restrições não obrigatórias, mas desejáveis (restrições brandas) de serem cumpridas:

- Um estudante não deverá ser programado em mais de dois eventos consecutivos, isto é, não é desejável que um estudante seja programado em 3 ou mais eventos consecutivos. A esse tipo de restrição chamaremos de *RD1*.

- b) Um estudante não deverá ser programado em apenas um evento em um dia. A esse tipo de restrição chamaremos de *RD2*.
- c) Um estudante não deverá ser programado no último horário do dia. A esse tipo de restrição chamaremos de *RD3*.

Neste trabalho estamos implementando um algoritmo VNS, portanto, precisamos formular uma função objetivo para avaliar uma proposta de solução. Assim, no problema de programação de horários a função objetivo deverá ser quantificada como uma medida das infactibilidades obrigatórias e não obrigatórias, mas desejáveis. Logo, a função objetivo assumirá a seguinte forma:

$$\min v = \alpha NRO + \beta NRD$$

onde *NRO* é o número de restrições obrigatórias de uma proposta de solução, *NRD* é o número de restrições não obrigatórias de uma proposta de solução e α e β são parâmetros que deverão ser escolhidos. Neste trabalho consideramos que uma proposta de solução não deverá violar as restrições obrigatórias, isto é, uma proposta de solução que viola uma restrição obrigatória não é aceitável. Adicionalmente, consideramos que entre as propostas de solução que não tem restrições obrigatórias violadas a melhor delas é aquela que tiver o menor número de restrições não obrigatórias (mas desejáveis de serem satisfeitas) violadas. Por esse motivo consideramos que os melhores valores para serem usados na pesquisa são $\alpha = 1000$ e $\beta = 1$.

1.3 DESENVOLVIMENTO DE UM MODELO MATEMÁTICO

Neste trabalho, é proposta uma metaheurística VNS para resolver o problema de horários em ambientes universitários. Entretanto, também apresentamos a seguir uma modelagem matemática, a título ilustrativo, mas não utilizada, para a representação do problema. Deve-se observar que para instâncias disponíveis no problema de grade horária, na competição internacional, Softwares comerciais tais como o CPLEX ainda não são competitivos quando comparados com as metaheurísticas especializadas.

O problema de grade horária pode ser definido como a minimização das restrições brandas, satisfazendo as condições de factibilidade. Dessa forma, no modelo é minimizada a soma das variáveis $f_{a,d}$, $C_{a,d,k}$ e $u_{a,d}$, já que elas representam as restrições brandas, tal como mostrado em (1). As variáveis que definem a programação dos eventos têm três sub-índices: o primeiro para identificar o evento, outro correspondente à sala e o terceiro relacionado com o

bloco de tempo ($x_{e,s,t}$). Um evento deve ser programado somente uma vez, então a soma das variáveis associadas a esse evento deve ser igual a um. Para cada evento existe uma restrição desse tipo (2).

Em uma sala, para um bloco de tempo, não é possível programar mais de um evento, em consequência a soma das variáveis que têm o mesmo índice de sala e bloco de tempo deve ser menor ou igual a um. Existe uma restrição desse tipo para todas as combinações de salas e blocos de tempo (3).

Para garantir que a sala atribuída a um evento cumpre com a capacidade e as características requeridas é usado o parâmetro $ES_{e,s}$. Se a variável $x_{e,s,t}$ tem um valor igual a 1, então deve-se cumprir que a sala s está habilitada para o evento e , portanto, o elemento $ES_{e,s}$ deve ser igual a 1; esta condição é representada em (4). Existe uma restrição desse tipo para todas as variáveis de decisão.

A variável $HA_{a,t}$, calculada usando (5), representa o estado do horário do aluno a no bloco de tempo t . Para calcular o horário dos alunos é usado o parâmetro $EA_{e,a}$ que indica se o aluno a assiste o evento e . Com a variável $HA_{a,t}$ é evitado o cruzamento no horário dos estudantes por meio de (6), que estabelece que o número de eventos que assiste um aluno, numa determinada hora, não pode ser maior que 1. Para cada aluno e cada bloco de tempo existe uma restrição desse tipo.

A restrição (7) é utilizada para determinar se para o aluno a no dia d , existem mais de dois eventos programados de forma consecutiva, nesse caso a variável $C_{a,d,k}$ toma o valor de 1. Essa restrição avalia três horas consecutivas e verifica se o aluno assiste mais de dois eventos, sendo que para cada dia são testadas 7 combinações, representadas usando o índice k .

A variável $f_{a,d}$ indica se o aluno a tem programado um evento na última hora do dia d , como é determinado em (8).

A variável $u_{a,d}$ é usada para representar o caso em que o aluno a tem um único evento no dia d . Essa variável é calculada usando (9) - (12), sendo que a variável contínua $n_{a,d}$ está ativa (variando desde 2 até 9) quando a soma de eventos no dia para o aluno é maior que 2 (determinado por meio da variável binária $\alpha_{a,d}$).

Finalmente, (13) representa a condição binária da variável $x_{e,s,t}$.

Modelo Ilustrativo

$$\min \sum_{\alpha \in \Omega_A} \sum_{d=1}^5 \left(f_{a,d} + \sum_{k=1}^7 c_{a,d,k} + u_{a,d} \right) \quad (1)$$

s.a.

$$\sum_{s \in \Omega_s} \sum_{t \in \Omega_t} x_{e,s,t} = 1 \quad \forall e \in \Omega_E \quad (2)$$

$$\sum_{e \in \Omega_E} x_{e,s,t} \leq 1 \quad \forall s \in \Omega_s, t = 1, \dots, 45 \quad (3)$$

$$ES_{e,s} - x_{e,s,t} \geq 0 \quad \forall e \in \Omega_E, s \in \Omega_s, t = 1, \dots, 45 \quad (4)$$

$$HA_{a,t} = \sum_{e \in \Omega_E} EA_{e,a} \sum_{s \in \Omega_s} x_{e,s,t} \quad \forall a \in \Omega_A, t = 1, \dots, 45 \quad (5)$$

$$HA_{a,t} \leq 1 \quad \forall a \in \Omega_A, t = 1, \dots, 45 \quad (6)$$

$$\sum_{t=9d+k-9}^{9d+k-7} HA_{a,t} \leq 2 + c_{a,d,k} \quad \forall a \in \Omega_A, d = 1, \dots, 5, k = 1, \dots, 7 \quad (7)$$

$$f_{a,d} = HA_{a,9d} \quad \forall a \in \Omega_A, d = 1, \dots, 5 \quad (8)$$

$$\sum_{t=9d-8}^{9d} HA_{a,t} = u_{a,d} + n_{a,d} \quad \forall a \in \Omega_A, d = 1, \dots, 5 \quad (9)$$

$$\sum_{t=9d-8}^{9d} HA_{a,t} \geq 2\alpha_{a,d} \quad \forall a \in \Omega_A, d = 1, \dots, 5 \quad (10)$$

$$2\alpha_{a,d} \leq n_{a,d} \leq 9\alpha_{a,d} \quad \forall a \in \Omega_A, d = 1, \dots, 5 \quad (11)$$

$$\alpha_{a,d} \in \{0, 1\} \quad \forall a \in \Omega_A, d = 1, \dots, 5 \quad (12)$$

$$x_{e,s,t} \in \{0, 1\} \quad \forall e \in \Omega_E, s \in \Omega_s, t = 1, \dots, 45 \quad (13)$$

A denominação das grandezas usadas na modelagem matemática são as seguintes:

$\alpha_{a,d}$ é a variável binária que determina se o aluno a tem mais de dois eventos programados no dia d , Ω_A é o conjunto de alunos, Ω_E é o conjunto de eventos, Ω_s é o conjunto de salas, Ω_t é o conjunto de horas, $C_{a,d,k}$ é a variável que indica se o aluno a tem mais de dois eventos programados consecutivamente, no dia d e na k -ésima combinação, $EA_{e,a}$ é o parâmetro que indica se o aluno a assiste o evento e , $ES_{e,s}$ é o parâmetro que indica se a sala s tem a capacidade e as características requeridas pelo evento e , $f_{a,d}$ é a variável que indica se o aluno a tem programado um evento ao final do dia d , $HA_{a,t}$ é a variável que indica o número de eventos que o aluno a tem programados na hora t , $u_{a,d}$ é a variável que indica se o aluno a tem um único evento programado no dia d , $x_{e,s,t}$ é a variável binária de decisão que indica se o evento e é programado na sala s e na hora t .

1.4 METAHEURÍSTICAS USADAS NO PROBLEMA DA GRADE HORÁRIA

Um problema que toda instituição de ensino se depara ao término de um semestre ou ano letivo e início de outro é o da elaboração manual da grade horária e que, de longe, não é algo fácil de resolver visto que abrange, pelo menos, três dimensões distintas e inter-relacionadas e que concorrem ao bom funcionamento das atividades acadêmicas ao longo do ano/semestre letivo: organizacional, pedagógica e pessoal. Quaisquer ‘erros’ que possam vir a ocorrer, comprometerão o desenvolvimento eficaz e eficiente das atividades previstas pelas instituições implicadas.

Por se tratarem de poderosas simulações inspiradas em modelos do cotidiano humano ou da natureza, as instituições passaram a recorrer a diversas metaheurísticas no intuito de resolver esse problema árduo e complexo de Grade Horária. Também no Brasil, com o advento do desenvolvimento da microinformática a partir da década de 90 do século XX, esse problema que era tratado com o uso de ferramentas de programação linear e métodos matemáticos baseados em otimização clássica, passam a ser executados pelas universidades públicas e privadas a partir de propostas computacionais desenvolvidas para a resolução do

problema, fundamentadas no uso de metaheurísticas com o desenvolvimento de modelos matemáticos específicos às necessidades de cada um.

Inicialmente, a definição que usaremos para metaheurística é a apresentada, adiante, por (GLOVER,1986).Metaheurísticas são técnicas de solução que gerenciam uma interação entre as estratégias de busca local e as estratégias de nível superior para criar um processo de otimização com capacidade de sair de soluções ótimas locais e realizar uma busca robusta através de espaço de busca. Alternativamente, podemos definir uma metaheurística como sendo um processo de otimização representado por uma generalização e/ou integração do algoritmo heurístico construtivo (AHC) de tipo guloso e a heurística de busca através de vizinhança (SDH) de forma que seja possível encontrar soluções de qualidade percorrendo de forma eficiente o espaço de busca. Em outras, palavras, uma metaheurística pode ser interpretada como uma generalização e/ou integração do AHC e da heurística SDH.

Ao adotar a aplicação das metaheurísticas em problemas de otimização, busca-se dentre outros objetivos gerar procedimentos de busca em vizinhança (no espaço de busca) que evitem uma parada prematura em ótimos locais e se encontre soluções melhores.

Os principais objetivos deste trabalho são:

- Projetar e implementar computacionalmente um algoritmo tipo VNS para resolver o problema de programação de horários em ambientes universitários.
- Realizar uma análise teórica e experimental do desempenho do algoritmo VNS e discutir a aplicação desse algoritmo na otimização de outros problemas da família de problemas do tipo *timetabling*.
- Contribuir na formação de profissionais especializados para atuar no setor de programação ótima de atividades.

1.5 CARACTERÍSTICAS DE UM PROBLEMA DE HORÁRIO

Várias características fazem com que Problemas de Horário sejam particularmente difíceis de resolver, como, por exemplo, o tamanho do espaço de busca, a quantidade de restrições, a representação da solução e o critério de avaliação das soluções.

A natureza combinatória de problemas de horário faz com que o tamanho do espaço de busca aumente drasticamente o tamanho do problema, tornando praticamente

impossível explorar explicitamente todas as soluções, com exceção de problemas experimentais muito pequenos.

É frequente a dificuldade de se encontrar uma representação com estruturas de dados associadas que capturam todos os detalhes do problema. Na maioria das vezes, o problema é simplificado para evitar modelagens muito elaboradas para representá-lo.

Devido à natureza combinatória, à medida que o problema cresce a avaliação das soluções tendem a consumir um tempo maior de processamento

Em geral, problemas de horário educacionais possuem um número considerável de requisitos. Esses requisitos limitam os possíveis modos nos quais um horário pode ser construído. Alguns requisitos devem ser satisfeitos para que as soluções possam ser viáveis (requisitos essenciais), enquanto outros são desejáveis (requisitos não-essenciais).

Em geral, existem requisitos básicos que podem ser levados em conta no problema de horário escolar. Entre esses requisitos podem ser mencionados os seguintes:

- (1) Os currículos das classes (conjunto de estudantes que têm um currículo em comum) devem ser respeitados, ou seja, todas as aulas nomeadas a cada classe são completamente marcadas;
- (2) Deve-se assegurar que cada professor ou classe seja nomeado a não mais que uma aula e a uma sala no mesmo período de tempo;
- (3) As aulas podem ter diferentes durações, dadas por um múltiplo do período de tempo;
- (4) Por razões pedagógicas, cada aula é programada para ser dada em não mais que uma vez por dia;
- (5) As salas têm que ser satisfatórias às aulas nomeadas. Cada aula pode requerer um tipo particular de sala, em termos de número de assentos disponíveis ou recursos especiais. Por exemplo, aulas de computação requerem uma sala que contenha computadores e programas específicos;
- (6) Os turnos para cada classe devem ser respeitados. É o caso da programação de aulas para uma classe de trabalhadores, que normalmente é executada durante o período noturno;
- (7) Cada classe e cada professor deverão ter um período de descanso do almoço. Essa restrição impõe um período livre de, pelo menos, um período de tempo a ser usado para a atividade de almoço;
- (8) Deve-se respeitar a indisponibilidade das classes e professores. Na realidade, classes e, muito mais frequentemente, os professores, têm períodos de indisponibilidade, que surgem de atividades extra-classe ou compromissos acadêmicos;

- (9) A ocorrência de períodos ociosos entre as aulas programadas deve ser baixa para classes e professores. Essa condição é desejável para se produzir horários mais compactos (evitar lacunas);
- (10) As preferências do professor por horários e salas devem ser satisfeitas. Em quase toda escola, professores expressam as suas preferências, que deveriam ser conhecidas sempre que possível;
- (11) O número de professores e a troca de classes entre diferentes prédios de aula devem ser minimizados. Para instituições com vários prédios de aula geograficamente distantes, um tempo de troca mínimo deveria ser respeitado, para permitir aos professores e às classes a movimentação entre os diferentes prédios.

Os requisitos (1) e (2) são comuns a todos os problemas de horários escolares. Uma melhor qualidade da solução é assegurada pela satisfação dos requisitos de (1) a (8). Os requisitos não essenciais (9), (10) e (11) representam os aspectos opcionais, ou seja, os desejos que professores e classes gostariam de ter satisfeitos em seus horários, e que devem ser violados o mínimo possível.

(GAMBINI, 2007), enfatizou aspectos sobre os trabalhos existentes na literatura envolvendo métodos computacionais para produção de quadros de horários, com ênfase em trabalhos que consideram problemas iguais ou relacionados ao Problema de Programação de Horários Professor x Turma com Compacidade – PPTC. Vários métodos têm sido utilizados para a produção automatizada de quadros de horários. Entre os métodos heurísticos, em geral, versões sofisticadas são necessárias: mesmo a geração de soluções factíveis para problemas reais de programação de horários é uma tarefa difícil e uma vez que se obtenha uma solução factível o método de melhora preferencialmente deve ser capaz de escapar dos ótimos locais, na maioria das vezes distantes do ótimo global para esse tipo de problema. Metaheurísticas baseadas em busca local como Busca Tabu e Simulated Annealing têm se mostrado técnicas genéricas eficientes para esse tipo de problema, assim como algoritmos evolutivos híbridos. Entre as técnicas exatas destacam-se métodos de Programação Linear Inteira Mista.

A seguir serão apresentados alguns trabalhos que propuseram métodos de produção automatizada de horários para instituições educacionais.

1.6 MÉTODOS DE PRODUÇÃO AUTOMATIZADA DE HORÁRIOS PARA INSTITUIÇÕES EDUCACIONAIS.

1.6.1 Busca Local

O trabalho de (FERLAND E LAVOUE, 1992), apresenta uma modelagem de programação inteira 0-1 para problemas de programação de horários, de modo que o tratamento de problemas como o Problema de Programação de horários Professor x Turma - PPT é possível através da utilização das chamadas *side constraints*. Essa modelagem relativamente genérica permite a modelagem de várias restrições fortes, mas não permite a modelagem de problemas com restrições fracas, como as existentes no Problema de Programação de Horários Professor x Turma com Compacidade – PPTC. Um procedimento heurístico baseado em busca local é apresentado. Nesse procedimento, a busca local primeiramente é aplicada para a diminuição progressiva das infactibilidades. Logo que uma solução factível é obtida, inicia-se a busca por melhores soluções. Três métodos são apresentados: no primeiro utilizam-se trocas simples como busca local; o segundo procedimento realiza uma busca recursiva que apresenta similaridades em relação ao método de busca tabu e, finalmente, um procedimento de relaxação lagrangeana é apresentado.

1.6.2 Busca Tabu

O método de Busca Tabu, proposto independentemente por (GLOVER, 1986) e (HANSEN, 1986), faz uso explícito de estruturas de memória para guiar métodos de descida, de modo que esses continuem a exploração mesmo na ausência de movimentos de melhora.

(HERTZ, 1991), propôs métodos de busca tabu para a geração de horários em universidades, baseados em uma implementação prévia do método para coloração de grafos. Uma peculiaridade dessa proposta é a decisão de colocar a violação de algumas restrições fracas fora do espaço de busca (qualquer solução sempre satisfará essas) e tentar então diminuir a ocorrência de conflitos (restrição forte).

O teste dos métodos em duas instituições mostrou que soluções consideravelmente melhores do que as soluções manuais podem ser obtidas.

Mais especificamente tratando de problemas de programação de horários em escolas, algoritmos baseados em busca tabu foram propostos em [(COSTA, 1994), (SCHAERF, 1999), (ALVAREZ-VALDÉS et al., 1996)]. Nos três trabalhos, instâncias de problemas reais foram consideradas.

Na busca tabu de (COSTA, 1994) a vizinhança é definida pela mudança no período de alocação de uma determinada aula. A memória de curto prazo é implementada através de duas listas tabu: T_1 e T_2 que consideram, respectivamente, a movimentação de uma aula e a movimentação de uma aula para um determinado período, sendo $|T_1| < |T_2|$. O critério de aspiração considera alterações não somente no valor da função objetivo, mas em componentes desta. A diversificação é implementada através da oscilação dos pesos na função objetivo relacionados à ocorrência de conflitos: faz-se uma redução periódica desses pesos permitindo que o processo de busca visite outras regiões do espaço de busca.

No método de (SCHAERF, 1999) os movimentos podem envolver a mudança de período de uma aula na agenda de um professor, mas também a troca de períodos entre duas aulas. A busca tabu de Schaerf é realçada com componentes aleatórios: após certo número de iterações sem melhora é realizada a fase de *RNA (randomized non-ascendent method)*, onde movimentos são sorteados e, caso não gerem conflitos, estes movimentos são realizados. Caso o movimento sorteado gere um conflito, considera-se a reparação desse movimento por meio da exploração (nesse caso, sistemática) de um segundo movimento, gerando os chamados movimentos duplos. O método também considera a utilização de relaxação adaptativa como em (GENDREAU et al., 1994).

O método híbrido baseado em Busca Tabu e GRASP (RESENDE; RIBEIRO, 2003)GBT- II, proposto por [(SOUZA, 2000), (SOUZA et al., 2003)], destacou-se dentre os algoritmos híbridos estudados em (SOUZA, 2000), onde são apresentadas e avaliadas computacionalmente várias propostas de heurísticas para o PPTC, incluindo *Simulated Annealing*, Busca Tabu e Otimização Microcanônica (TORREÃO; ROE, 1995). Os algoritmos propostos agregavam um método de melhoramento baseado em caminhos mínimos (SOUZA et al., 2000) para a pesquisa de movimentos compostos, o qual é utilizado tanto para acelerar a obtenção de soluções factíveis quanto para o melhoramento dessas. A diversificação é implementada através da reinicialização com a fase de construção da metaheurística GRASP.

Nesse trabalho, o método híbrido baseado em Busca Tabu superou consideravelmente os métodos baseados em *Simulated Annealing* e Otimização Microcanônica.

1.6.3 *Simulated Annealing*

O método *Simulated Annealing* foi apresentado no contexto de otimização em (KIRKPATRICK; GELATT; VECCHI, 1983). Baseado em busca local, assim como busca tabu, também considera a realização de movimentos que pioram o valor da função objetivo. Nesse caso, no entanto, um componente aleatório, ao invés da utilização de memória, é considerado para decidir a aceitação do movimento. A probabilidade de aceitação de movimentos desse tipo diminui gradualmente no decorrer da busca. (ABRAMSON, 1991) realiza experimentos com versões sequenciais e paralelas do *Simulated Annealing*. O conjunto de problemas teste consistia em problemas artificiais, gerados de modo que a solução ótima fosse conhecida (sem conflitos, visto que havia sido a única restrição considerada) e um problema real, com algumas restrições adicionais. A versão paralela do método de *Simulated Annealing*, que consistia em uma implementação com granularidade fina (CUNG, 2002): implementada em um sistema paralelo com memória compartilhada, apresentou uma aceleração considerável.

1.6.4 Algoritmos Evolutivos

Vários algoritmos genéticos foram propostos para a resolução de problemas de programação de horários. (ABRAMSON e ABELA, 1992) propuseram um algoritmo genético paralelo para a resolução de problemas de programação de horários em escolas. No problema considerado, somente leva-se em conta a não existência de conflitos. Experimentos com instâncias artificiais são apresentados. O algoritmo genético híbrido de (COLORNI; DORIGO e MANIEZZO, 1998) considera um problema com mais requisições reais. Nesse algoritmo é utilizada uma combinação de operadores genéticos que consideram características do problema e uma busca local em múltiplas vizinhanças.

Comparações com implementações bastante simples de *Simulated Annealing* e *Busca Tabu* são apresentadas, sendo que os melhores resultados foram apresentados pelo algoritmo genético híbrido e pela busca tabu. Um algoritmo genético híbrido também é apresentado em (STEFANO; TETTAMANZI, 2001).

1.6.5 Métodos Baseados em Programação Linear

Técnicas de Programação Linear, incluindo Programação Linear Inteira, também têm sido utilizadas para a geração de quadros de horários. Na maioria dos casos, busca-se a produção de quadros de horários provadamente ótimos. Considerando que a prova da otimalidade pode implicar a resolução de um número computacionalmente intratável de problemas lineares, trabalhos iniciais como (LAWRIE, 1969) tratavam apenas parte do problema hoje conhecido como o problema de programação de horários. Em (LAWRIE, 1969), além de o problema considerar apenas a não ocorrência de conflitos, cabia ao usuário do software informar previamente quais cursos podiam ser oferecidos concomitantemente, através da criação de "arranjos". Em (DREXL; SALESWSKI, 1997) a formulação compacta apresentada com variáveis binárias, permitiu somente a resolução de instâncias artificiais de menor dimensão, considerando o tamanho de problemas reais típicos.

Mais recentemente, progressos na capacidade de resolução de problemas de Programação Linear Inteira Mista (PLIM) trouxeram um renovado interesse na aplicação de PLIM para a resolução de problemas de programação de horários e permitiram que problemas reais com importantes considerações práticas fossem tratados em [(AVELLA; VASILÉV, 2005), (DASKALAKI, 2004), (BIRBAS et al., 1997)]. Uma particularidade dos modelos utilizados em [(AVELLA; VASILÉV, 2005), (BIRBAS et al., 1997)], bastante relacionados com o PPTC, é o fato de que a questão da compactidade de horários para professores não é avaliada na função objetivo. Em (BIRBAS et al., 1997), por exemplo, especifica-se um conjunto de dias de folga para cada professor, já em (AVELLA; VASILÉV, 2005) especifica-se um número máximo de dias em que o professor deve comparecer à instituição de ensino. Essa abordagem, apesar de possibilitar uma considerável diminuição na dimensão do espaço de busca, traz a desvantagem de que na especificação do número de dias de folga/trabalho de cada professor devem ser feitas escolhas sábias: a requisição de horários tão compactos quanto possível, considerando cada professor isoladamente pode resultar em um problema sem solução viável, que somente poderá tornar-se viável através da relaxação desse requisito para alguns professores. A alteração manual do modelo para permitir a factibilidade e oferecer quadros de horários compactos para professores pode excluir soluções ótimas quanto a esse requisito.

Em (PAPOUTSIS et al., 2003) é apresentada uma aplicação de Geração de Colunas (DANTZIG, WOLFE, 1960) para um modelo com um número exponencial de variáveis, para uma variante do PPT encontrada na Grécia.

Nesse modelo, cada variável binária representa um possível quadro de horários semanal de um professor, sendo que a maioria das restrições é considerada na geração das colunas, ou seja, tem-se quase que exclusivamente restrições fortes, de modo que a função objetivo tem poucos (dois) componentes. A determinação de colunas atrativas é realizada através da busca no espaço de soluções factíveis com a estratégia de mais-profundo-primeiro. O procedimento procede iterativamente com a adição de novas colunas, enquanto alguma melhora significativa na função objetivo é observada. Uma vez que a solução resultante do procedimento apresentado geralmente não é inteira, os autores desenvolveram um procedimento específico que tenta factibilizar a integralidade através da fixação de alguns quadros de horários de professores. Deve-se observar que o procedimento por esses autores é heurístico, sem garantia de produção da solução ótima.

1.6.6 Comparações entre os métodos

Apesar da existência de várias publicações que tratam de problemas iguais ou semelhantes, poucos progressos foram realizados no campo de análises comparativas entre as implementações dos diferentes métodos, com eventuais exceções para casos em que os autores implementem as alternativas concorrentes do início [(COLORNI et al., 1998), (SOUZA, 2000)].

No trabalho de (ABRAMSON, 1991), a comparação com os resultados de (GANS, 1981) é realizada utilizando-se instâncias diferentes das do trabalho anterior, visto que o autor não teve acesso aos dados completos que definiam as instâncias. Instâncias de mesma dimensão e conteúdos aleatórios foram utilizadas para os experimentos comparativos. O autor afirma em (ABRAMSON, 1991) que os resultados desse experimento não permitem nenhuma comparação absoluta entre os trabalhos.

(SCHAERF, 1999) argumenta que suas escolhas com relação à representação de solução e definição da vizinhança são mais apropriadas do que as utilizadas em (COLORNI et al., 1998) e (COSTA, 1994), respectivamente, sem apresentar resultados experimentais comparativos que justifiquem essa afirmação. De fato, nenhum dos trabalhos considerou o mesmo conjunto de instâncias, o que torna difícil a elaboração de conclusões sobre quais escolhas foram as mais apropriadas. Nos três trabalhos, a avaliação dos resultados é realizada considerando comparações com soluções manuais, um critério importante mas nem sempre preciso.

2 A METAHEURÍSTICA DE BUSCA EM VIZINHANÇA VARIÁVEL

2.1 INTRODUÇÃO SOBRE AS HEURÍSTICAS

As heurísticas são técnicas de otimização que geralmente encontram soluções de boa qualidade de problemas complexos. Deve-se observar que, entre as décadas de 1960 e 1970, as heurísticas foram as técnicas de otimização matemática mais usadas, especialmente para resolver aqueles problemas não lineares, discretos e não convexos.

A maioria das heurísticas encontram soluções de boa qualidade de problemas altamente complexos em tempos computacionais relativamente rápidos. Adicionalmente, a maioria das heurísticas são simples de entender e também de ser implementadas computacionalmente. Entretanto, as técnicas heurísticas renunciam, pelo menos do ponto de vista teórico, a encontrar a solução ótima global de um problema complexo. Em problemas de grande porte e complexos, as técnicas heurísticas raramente encontram as soluções ótimas. Vamos considerar um problema como sendo complexo quando existe grande dificuldade para encontrar a solução ótima global, devido principalmente a que o problema apresenta a característica da explosão combinatória quando o tamanho do problema cresce e/ou porque o problema apresenta uma modelagem matemática complexa (variáveis inteiras ou discretas, função objetivo não linear e não diferenciável, restrições não lineares, região factível não convexa, etc.).

Uma técnica heurística pode ser muito simples como, por exemplo, o uso de bom senso ou a experiência de um especialista ou pode ser muito sofisticada, geralmente envolvendo a solução de modelos matemáticos relaxados em relação ao modelo original.

Os casos interessantes de se usar técnicas heurísticas de otimização são os seguintes:

- (1) Quando não existe um método exato de otimização para resolver o problema em análise.
- (2) Quando a solução ótima não é muito importante do ponto de vista prático por diferentes motivos, como, por exemplo, a existência de muitas soluções ótimas locais de qualidade muito próxima da solução ótima global.
- (3) Quando os dados usados apresentam incerteza elevada como acontece em muitos problemas relacionados com planejamento.

- (4) Quando existem limitações de tempo de processamento para encontrar a solução procurada e quando existem problemas de memória para armazenamento de dados. Problemas desse tipo podem aparecer em problemas de aplicação em tempo real.
- (5) Quando se pretende encontrar uma boa solução inicial para ser usada como ponto de partida na aplicação de uma técnica de otimização mais sofisticada como, por exemplo, quando pretendemos usar um algoritmo *branch and bound*.

Classificar as técnicas heurísticas de otimização não é uma tarefa das mais simples. Uma proposta de classificar as técnicas heurísticas é a seguinte: algoritmos heurísticos construtivos, algoritmos de decomposição, algoritmos de divisão, algoritmos de redução, algoritmos de manipulação do modelo matemático e algoritmos de busca através de vizinhança (*steepest descent heuristic*). Neste trabalho não pretendemos realizar uma apresentação detalhada de todas as técnicas heurísticas de otimização, assim apresentamos de forma mais detalhada apenas o algoritmo heurístico construtivo (AHC) e o algoritmo heurístico de busca através de vizinhança (SDH).

2.1.1 O Algoritmo Heurístico Construtivo

O algoritmo heurístico construtivo (AHC) é uma das técnicas heurísticas de otimização mais usadas para resolver problemas complexos e, ainda, é muito usado isoladamente ou integrado a metaheurísticas mais sofisticadas. O mais popular dos algoritmos construtivos é o do tipo guloso (*greedy*).

O AHC é uma técnica de otimização que, em um processo passo a passo, gera uma solução geralmente de boa qualidade de um problema complexo. Em cada passo o AHC escolhe um elemento ou componente da solução que está sendo construída e, no último passo, termina de gerar uma solução factível. O elemento ou componente da solução que é escolhido em cada passo do AHC é identificado usando um indicador de sensibilidade que identifica o componente mais interessante a ser incorporado na solução em construção. Assim, a diferença fundamental entre os AHCs usados para resolver um mesmo problema está no indicador de sensibilidade usado.

Um AHC pode assumir a seguinte forma genérica:

- (1) Armazenar os dados do problema e escolher o indicador de sensibilidade a ser usado. Escolher os componentes que podem ser incorporados na solução em construção (geralmente o processo é iniciado sem componentes).

- (2) Verificar se a solução em construção já representa uma solução factível. Caso seja factível então pare o processo porque foi encontrada a solução factível procurada. Em caso contrário ir ao passo 3.
- (3) Usando a solução em construção, resolver o problema que permite identificar o indicador de sensibilidade de todos os componentes do problema que ainda não foram incorporados na solução em construção.
- (4) Usando informação dos indicadores de sensibilidade encontrados no passo anterior identificar a componente que deve ser incorporada na solução em construção. Adicionar a componente identificada na solução em construção e voltar ao passo 2.

2.1.2 O Algoritmo Heurístico de Busca Através de Vizinhaça

O algoritmo heurístico de busca através de vizinhaça (*steepest descent heuristic*) é significativamente diferente do algoritmo heurístico construtivo do tipo guloso. No AHC se gera apenas uma solução factível através de uma sequência de passos e usando um indicador de sensibilidade. No algoritmo heurístico de busca através de vizinhaça, que chamaremos apenas de algoritmo SDH (*Steepest Descent Heurístico*) para problemas de minimização, o processo é geralmente iniciado a partir de uma solução factível e na sequência são encontradas novas soluções factíveis, percorrendo o espaço de busca e passando sempre para a melhor solução vizinha.

A estratégia fundamental da heurística SDH pode ser resumida da seguinte forma:

- (1) O processo de otimização é iniciado através de uma solução inicial (factível ou infactível) que passa a ser chamada de solução corrente.
- (2) Deve-se definir uma estrutura de vizinhaça. Assim, deve existir uma forma de identificar as soluções que são consideradas vizinhas da solução corrente. As soluções vizinhas podem ser factíveis ou infactíveis.
- (3) Na heurística SDH se passa da solução corrente para a melhor solução vizinha.
- (4) O processo termina quando todas as soluções vizinhas são de pior qualidade que a solução corrente.

Deve-se observar que para implementar a heurística SDH não necessariamente estamos obrigados a usar o modelo matemático do problema em análise. Na verdade a heurística SDH pode resolver problemas de otimização que não tem modelagem matemática e essa característica torna a heurística SDH, assim como as metaheurísticas, uma técnica de otimização relativamente distante da lógica de otimização usada na otimização clássica.

2.1.3 Introdução sobre as Metaheurísticas

A definição de uma metaheurística apresentada por Glover é a seguinte: “Metaheurísticas são técnicas de solução que gerenciam uma interação entre as estratégias de busca local e as estratégias de nível superior para criar um processo de otimização com capacidade de sair de soluções ótimas locais e realizar uma busca robusta através de espaço de busca”. Alternativamente, podemos definir uma metaheurística como um processo de otimização representado por uma generalização e/ou integração do algoritmo heurístico construtivo de tipo guloso e a heurística de busca através de vizinhança, de forma que seja possível encontrar soluções de qualidade percorrendo de forma eficiente o espaço de busca. Em outras palavras, uma metaheurística pode ser interpretada como uma generalização e/ou integração do AHC e da heurística SDH mostrados em [2.1.1.] e [2.1.2.].

A seguir apresentaremos de forma resumida as principais metaheurísticas usadas no campo da pesquisa operacional.

2.1.4 Simulated Annealing

Simulated Annealing (SA) é uma das primeiras metaheurísticas que foi usada com muito sucesso na otimização de problemas complexos na pesquisa operacional. SA foi inventado após verificar que existiam muitas semelhanças entre a técnica de construção de cristais perfeitos usando a técnica de *annealing* e a otimização de um problema complexo no campo da pesquisa operacional.

Existem muitas técnicas usadas na construção de cristais perfeitos e uma delas é a técnica de *annealing*. Essa técnica consiste no aquecimento de um material até temperaturas elevadas na qual existe uma movimentação molecular do material esquentado e, portanto, um novo arranjo dos átomos do material. A partir desse estado, se o material for resfriado lentamente, então a movimentação molecular também diminui. Se esse processo de diminuição for adequadamente controlado, preservando o chamado quase equilíbrio termodinâmico na qual a temperatura deve ser diminuída lentamente, então existe grande possibilidade de que o material se transforme em um cristal perfeito. Se esse processo não for realizado de forma adequada, então esse material pode ser transformado em um vidro.

Em resumo, usando o paralelismo ou semelhanças que existem entre a técnica de *annealing* na construção de cristais perfeitos e na otimização de problemas complexos no campo da pesquisa operacional, foi desenvolvido o algoritmo de *Simulated Annealing* que na formulação básica assume a seguinte forma (problema de minimização):

- (1) Passo preliminar: Montar os dados do problema. Escolher uma forma de codificação de uma proposta de solução denominada de p . Identificar uma forma de avaliar a qualidade da função objetivo ou equivalente e denominada $f(p)$. Definir a estrutura de vizinhança a ser usada, o que caracteriza o espaço de busca. Escolher os parâmetros de SA, tais como o parâmetro chamado de temperatura inicial T_o , a temperatura final T_f ou um critério de parada, o número de tentativas de transição no primeiro nível de temperatura N_o , o parâmetro ρ que controla o número de tentativas de transição em cada nível de temperatura e o parâmetro β que controla a diminuição do parâmetro temperatura.
- (2) Encontrar uma solução inicial p_o que se transforma na solução corrente p_c e fazer $N_k = N_o$. $s=0$.
- (3) Identificar e avaliar uma solução vizinha p_v escolhida aleatoriamente.
- (4) Se $f(p_v) < f(p_c)$, então a solução vizinha é melhor que a solução corrente e deve-se realizar a transição, isto é, $p_c = p_v$ e ir ao passo 5. Em caso contrário, gere um número aleatório entre 0 e 1, $P(0,1) = \text{random}[0,1]$ e seja $\Delta f(p) = f(p_v) - f(p_c)$. Assim, se $\exp\left[\frac{-\Delta f(p)}{T_k}\right] > P(0,1)$ então, deve-se realizar a transição e $p_c = p_v$ e, em caso contrário, a solução corrente é preservada. Ir ao passo 5.
- (5) $s = s+1$. Se $s < N_k$ então ir ao passo 3. Em caso contrário, ir ao passo 6.
- (6) Se o critério de parada foi cumprido, então pare. Em caso contrário, fazer $T_{k+1} = \beta T_k$ e $N_{k+1} = \rho N_k$. $k = k + 1$. Voltar ao passo 3.

Deve-se observar também que a técnica de *annealing* é uma técnica mais geral e não é usada apenas para a construção de cristais perfeitos. Essa técnica também é usada, por exemplo, na construção de condutores elétricos para tornar o condutor mais uniforme e maleável. Em relação com o algoritmo SA, deve-se realizar as seguintes observações:

- A relação $\exp\left[\frac{-\Delta f(p)}{T_k}\right]$ sempre é positiva e varia entre 0 e 1. Por esse motivo faz sentido a comparação com um número aleatório $P(0,1)$ que também varia entre 0 e 1. Adicionalmente, se T_k é muito grande ou $\Delta f(p)$ é muito pequeno, então $\exp\left[\frac{-\Delta f(p)}{T_k}\right]$ assume um valor próximo de 1. Por outro lado, se T_k é muito pequeno ou $\Delta f(p)$ é muito grande então $\exp\left[\frac{-\Delta f(p)}{T_k}\right]$ assume um valor próximo de zero.
- Da análise anterior podemos concluir que, se a solução vizinha tem um valor da função objetivo que é de pior qualidade que a solução corrente, mas de valor muito próximo e/ou se o processo se encontra nas fases iniciais (o valor do parâmetro de temperatura elevada), então existe uma probabilidade elevada de aceitar uma solução vizinha de pior qualidade. Por outro lado, se o valor da função objetivo da solução vizinha é de pior qualidade e muito distante do valor da função objetivo da solução corrente e/ou o processo se encontra nas fases finais (valor do parâmetro temperatura baixa) então a probabilidade de aceitar uma solução vizinha de pior qualidade é pequena. Em resumo, nas fases iniciais do processo podem ser aceitas soluções vizinhas de pior qualidade com mais facilidade (priorizando um processo de diversificação) e nas fases finais do processo raramente são aceitas soluções vizinhas de pior qualidade (priorizando o processo de intensificação). Finalmente, em qualquer estágio do processo, soluções vizinhas ligeiramente piores que a solução corrente têm maior probabilidade de serem aceitas, comparadas com soluções vizinhas piores e muito distantes da solução corrente.
- A relação $\exp\left[\frac{-\Delta f(p)}{T_k}\right]$ pode ser substituída por qualquer outra com a única condição de que desempenhe um papel equivalente, isto é, priorize as soluções vizinhas de pior qualidade da forma mostrada nos passos anteriores.

Deve-se observar que o algoritmo SA básico apresentado anteriormente pode ser obtido apenas como uma generalização da heurística SDH. Comparando a heurística SDH e o algoritmo SA podemos verificar as seguintes diferenças fundamentais:

- A heurística SDH avalia todas as soluções vizinhas para identificar o vizinho de melhor qualidade. Assim, o passo 3 da heurística SDH é muito demorado. Por outro lado, SA escolhe aleatoriamente uma solução vizinha e decide se realiza a transição sem conhecer nem avaliar as outras soluções. Nesse contexto, nas fases iniciais SA realiza transições com

maior intensidade que a heurística SDH, mas nas fases finais do processo ambas as técnicas de otimização devem apresentar comportamentos muito próximos.

SA pode sair de um ótimo local ao aceitar de forma probabilística transições para soluções vizinhas de pior qualidade. Essa é a principal diferença entre a heurística SDH e a metaheurística SA.

Obviamente SA apresenta um maior tempo de processamento e deve encontrar soluções de melhor qualidade do que a heurística SDH.

2.1.5 Tabu Search – Busca Tabu

Tabu Search (TS) é uma metaheurística inventada por Fred Glover. Ao contrário da maioria das metaheurísticas que usaram comportamentos ou características existentes em ramos do conhecimento distantes da otimização matemática, esse estudioso usou apenas conhecimentos existentes no campo da otimização matemática. Antes de inventar o TS, Glover já era um pesquisador destacado em otimização clássica, especialmente nas técnicas de otimização de problemas de programação inteira.

A idéia central de Glover é mostrar que é possível inventar uma estratégia de busca inteligente percorrendo o espaço de busca de forma eficiente e seletiva. Nesse processo é fundamental integrar o processo de buscas através das estratégias de intensificação e de diversificação. Nesse contexto, intensificar significa realizar uma busca mais intensa em torno da solução corrente, por exemplo, aumentando o tamanho da vizinhança ou melhorando a qualidade da vizinhança. Por outro lado, diversificar significa sair de uma região do espaço de busca e, de forma proposital, atingir uma região distante para novamente realizar algum processo de intensificação.

A formulação básica de TS, inspirada fundamentalmente no uso da estratégia de intensificação, assume a seguinte forma:

- (1) Passo preliminar: Montar os dados do problema. Escolher uma forma de codificação de uma proposta de solução denominada de p . Identificar uma forma de avaliar a qualidade da função objetivo ou equivalente e denominada $f(p)$. Definir a estrutura de vizinhança a ser usada e que caracteriza o espaço de busca. Identificar os tipos de atributos que devem ser proibidos e o critério de aspiração. Escolher os parâmetros do algoritmo tais como a duração da lista tabu. Escolher o critério de parada.

- (2) Encontrar uma solução inicial p_o que se transforma na solução corrente p_c .
- (3) Identificar e avaliar todas as soluções vizinhas da solução corrente p_c , ordenar essas soluções vizinhas por qualidade sendo que o primeiro da lista é a melhor solução vizinha p^{best} .
- (4) Realizar a transição para a solução vizinha melhor classificada, que não tenha o atributo proibido ou se tiver o atributo proibido, então satisfará o critério de aspiração. Seja p_e a melhor solução vizinha escolhida, então fazer $p_c = p_e$.
- (5) Atualizar a incumbente e a lista de atributos proibidos. Se o critério de parada for satisfeito, então pare. Em caso contrário, voltar ao passo 3.

O algoritmo TS anteriormente apresentado é chamado de algoritmo TS básico pois usa fundamentalmente memória de curto prazo, uma lista de atributos proibidos e um critério de aspiração, isto é, é uma estratégia de otimização que prioriza a intensificação.

Técnicas de otimização tipo TS mais complexas podem ser implementadas onde o TS básico funciona como um módulo de otimização integrado em uma estrutura TS mais complexa.

Essas estruturas mais complexas podem ser idealizadas usando outros operadores existentes em TS, tais como a diversificação, a memória baseada em frequência, a lista de soluções de elite, o *path relinking*, entre outros.

Em relação ao TS básico apresentado anteriormente, deve-se mencionar as seguintes observações:

- A única diferença do algoritmo TS básico em relação à heurística SDH é que TS realiza a transição para a melhor solução vizinha que não tem atributo proibido ou se tem o atributo proibido, então cumpre com o critério de aspiração. Em outras palavras, se todas as soluções vizinhas da solução corrente são piores que a solução corrente, então o algoritmo TS passa para a menos pior, mas montando uma estratégia para tentar evitar visitar uma solução já visitada.
- Como o algoritmo TS pode passar para uma solução de pior qualidade e ao mesmo tempo apresenta uma estratégia gulosa, então deve existir uma estratégia para evitar voltar a visitar uma solução já visitada. Essa estratégia pode ser montada de forma explícita, armazenando todas as soluções já visitadas (ou um conjunto menor representado pelas últimas soluções já visitadas) e verificando se a solução vizinha candidata a ser escolhida já foi visitada anteriormente ou pode ser armazenada de

forma implícita, armazenando apenas uma característica da solução visitada e chamada de atributo.

- O algoritmo TS básico não recomenda o armazenamento de soluções explícitas, devido a problemas de memória para armazenamento da informação e pelo tempo de processamento necessário para verificar se uma solução vizinha já foi visitada anteriormente e, portanto, se encontra armazenada. Entretanto, essa premissa que era muito válida na década de 1980, pode não ter a mesma validade atualmente devido ao incremento elevado de disponibilidade de memória de computadores modernos e também pelo aumento da velocidade de processamento desses computadores. Assim, para determinadas aplicações pode ser oportuno avaliar a possibilidade do uso de memória explícita.
- Na estratégia implícita é usado o conceito de atributo. O atributo relacionado com uma proposta de solução é algum tipo de informação que permite identificar alguma característica da proposta de solução. O tipo de atributo mais usado é o próprio mecanismo usado na transição e que, também, permite a identificação de uma solução vizinha da solução corrente.
- A proibição através de atributos tenta evitar voltar a uma solução já visitada. Entretanto, como essa proibição termina após k_p transições previamente especificadas, então é possível no futuro visitar uma solução já visitada e, nesse caso, acontece o fenômeno chamado de ciclagem. Uma forma de atenuar esse problema é aumentando o valor de k_p . Por outro lado, todas as soluções vizinhas que compartilham um atributo proibido com uma solução já visitada também não poderiam ser visitadas. Nesse contexto, se o valor k_p for muito grande, então podem existir muitas soluções vizinhas que não poderiam ser visitadas por compartilharem os mesmos atributos de soluções já visitadas e que ainda se encontram proibidas. Por esse motivo, o valor de k_p não pode ser muito grande. Adicionalmente, muitas soluções vizinhas que compartilham atributos proibidos com soluções já visitadas podem ser de excelente qualidade e, eventualmente, a própria solução ótima global pode estar proibida. Para contornar esse problema foi inventada a estratégia de critério de aspiração.
- A ideia central do critério de aspiração é eliminar a proibição do atributo de uma solução vizinha de excelente qualidade porque compartilha o mesmo atributo de uma solução já visitada. Assim, se uma solução vizinha satisfaz um critério de qualidade

que permite suspeitar que seja uma solução ainda não visitada, então eliminamos a proibição e realizamos a transição para essa solução vizinha. Esse critério de qualidade está relacionado com o valor da função objetivo. Um critério extremo, por exemplo, consiste em especificar que uma solução vizinha cumpre com o critério de aspiração; se for melhor que a incumbente.

- Nesse caso, existe a certeza absoluta de que essa solução vizinha nunca foi visitada anteriormente, mas apresenta desvantagem de que raras vezes pode ser acionada, o que pode ser quase equivalente a não ter critério de aspiração. Geralmente, pode ser mais oportuno escolher um critério de aspiração menos exigente. Um exemplo pode ser o seguinte: se a solução vizinha é melhor do que as soluções visitadas nas últimas k_{ul} iterações, então cumpre com o critério de aspiração.

2.1.6 O Algoritmo Genético

O algoritmo genético (AG) é uma metaheurística inventada por Holland na década de 70, e que apenas a partir da década de 80 teve sua aplicação de forma intensa para resolver problemas complexos no campo da pesquisa operacional. Para inventar o AG, Holland encontrou semelhanças entre a forma de resolver um problema de otimização matemática e o processo de seleção natural e a evolução das espécies. Na verdade, na natureza, o processo de seleção natural e a evolução das espécies é a consequência de um processo de otimização estocástica que acontece em um determinado ambiente em tempo real.

O processo de seleção natural e a evolução das espécies é um problema muito complexo para que seja imitado de forma adequada por um processo de otimização de um problema complexo do campo da pesquisa operacional. Assim, pode-se afirmar que o AG imita apenas uma parcela dos componentes que fazem parte do processo de seleção natural e de evolução das espécies. Algumas implementações do AG, que não acompanham de forma adequada o processo de seleção natural e a evolução das espécies, são as seguintes:

- Um cromossomo na genética é considerado como equivalente a uma proposta de solução no problema de otimização. Na verdade, uma proposta de solução deveria ser equivalente a um indivíduo. Adicionalmente, na genética, nas espécies mais complexas não existe um cromossomo, mas uma cadeia cromossômica.
- Um cromossomo na genética, na verdade, é um par cromossômico formado por duas cadeias de informação: um herdado do pai e outro herdado da mãe. Um elemento do

cromossomo é um gene que tem dois valores definidos (alelos), um em cada cadeia cromossômica. As duas informações representam uma unidade de informação genética. Na maioria dos casos existem apenas dois tipos de alelos (nessa posição genética e para todos os indivíduos da espécie) na posição correspondente a um gene. Assim, esse tipo de informação pode ser armazenado de forma binária (0 ou 1). Entretanto, foi provado que para uma determinada espécie existem vários tipos de alelos entre os elementos da espécie. Por exemplo, o gene que determina a cor dos olhos nos humanos. Portanto, a codificação binária usada no AG não representa de forma adequada aquilo que acontece na natureza.

- A informação genética presente em um gene define o fenótipo, isto é, a característica externa do indivíduo. Assim, por exemplo, um tipo de alelo (informação genética) define o tipo de sangue humano (Rh+ ou Rh-). Assim, existiria uma relação biunívoca entre tipo de alelo e fenótipo, o que permite montar uma estratégia de codificação (tipo de alelo) e decodificação (fenótipo) no AG. Entretanto, foi provado que em muitos casos apenas um tipo de alelo define várias características fenotípicas (fenômeno conhecido como pleiotropia) e também vários alelos correspondentes a vários genes definem apenas uma característica fenotípica (fenômeno conhecido como poligênia).

Assim, para esses casos não existe uma relação biunívoca entre informação genética e o fenótipo. Adicionalmente, o fenômeno de dominância na genética em que um alelo domina o outro alelo é muito importante na genética e não se encontra representado no AG.

O fenômeno de *crossing over* é fundamental para a existência da diversidade genética entre os indivíduos de uma espécie. A diversidade genética é a chave fundamental na seleção natural e na evolução das espécies. O fenômeno de *crossing over* acontece quando uma célula reprodutiva se duplica e se divide. Assim, uma célula reprodutiva (que é uma unidade celular com duas cadeias cromossômicas) após o *crossing over* gera quatro cadeias cromossômicas independentes. Acontece que na separação uma cadeia cromossômica (por exemplo, um espermatozóide) é formada por parcelas da cadeia cromossômica herdada do pai e da mãe, formada por um processo de recombinação bastante complexo. Em outras palavras, um espermatozóide é formado por pequenas parcelas da cadeia cromossômica do pai e da mãe, misturados por parcelas pequenas de cada um. Sem *crossing over*, um espermatozóide estaria formado apenas pela cadeia cromossômica herdada pelo pai ou pela mãe. Assim, um espermatozóide é uma meia célula que posteriormente pode se juntar a um óvulo para formar uma nova unidade celular (zigoto) e, portanto, um novo indivíduo. Assim, deve-se esclarecer

que o fenômeno de *crossing over* e a reprodução sexuada acontecem em instantes muito diferentes.

No AG o fenômeno de *crossing over* é imitado pelo operador de recombinação. Pode-se verificar que esse processo de imitação não é muito adequado.

O fenômeno da mutação acontece na natureza quando a composição genética de um gene é alterado (no processo de duplicação e separação de uma célula reprodutiva). Geralmente, esse processo de mutação pode gerar um indivíduo de pior qualidade, mas às vezes, pode gerar indivíduos de melhor qualidade. Esse fenômeno é adequadamente imitado no AG.

Em resumo, podemos afirmar que um AG imita de forma não adequada o processo de seleção natural e de evolução das espécies. Esse processo de imitação é mais crítico no fenômeno de *crossing over* e relativamente significativo na codificação binária. Entretanto, apesar de todas as observações realizadas anteriormente, o AG apresenta excelente desempenho na resolução de muitos problemas do campo da pesquisa operacional. Assim, sempre existiu essa dúvida entre os pesquisadores sobre esse aparente paradoxo, o que de alguma maneira explica a busca intensa para encontrar algoritmos inspirados na seleção natural e da evolução das espécies que incorporem de forma adequada o que acontece na natureza.

Essa pode ser uma explicação para a grande diversidade de algoritmos desse tipo que existe na literatura especializada e que podem ser denominados apenas como algoritmos evolutivos.

Um algoritmo genético básico assume a seguinte forma:

- (1) Passo preliminar: Montar os dados do problema. Escolher uma forma de codificação de uma proposta de solução denominada de p . Identificar uma forma de avaliar a qualidade da função objetivo ou equivalente e denominada $f(p)$. Escolher os parâmetros do algoritmo, tais como o tamanho da população n_p , a taxa de recombinação ρ_r , a taxa de mutação ρ_m e o tipo de seleção. Escolher um critério de parada.
- (2) Gerar a população inicial, isto é, gerar um conjunto de n_p propostas de solução que se transforma na população corrente.
- (3) Avaliar a qualidade f_p de todos os elementos da população e atualizar a incumbente, se possível.
- (4) Se o critério de parada for satisfeito, pare. Em caso contrário, ir para o passo 5.
- (5) Implementar o operador de seleção.
- (6) Implementar o operador de recombinação.
- (7) Implementar o operador de mutação, atualizar a população corrente e voltar ao passo 3.

Os primeiros algoritmos genéticos usavam a codificação binária. Entretanto, os algoritmos genéticos modernos seguem a proposta de Michalewicz, sugerindo que a codificação deve ser realizada seguindo a natureza e as características do problema.

2.1.7 A Metaheurística GRASP

O algoritmo GRASP, cujo nome vem do inglês *Greedy Randomized Adaptive Search Procedure*, é uma das metaheurísticas que, juntamente com *tabu search*, foram desenvolvidos usando apenas conceitos existentes no campo da pesquisa operacional. Lembremos que Glover, o inventor das metaheurísticas de *tabu search* e busca dispersa, já era um pesquisador conceituado no desenvolvimento de técnicas de otimização, especialmente no desenvolvimento de técnicas clássicas de otimização.

Os inventores do GRASP (Feo e Resende) também já eram pesquisadores conceituados em temas de otimização no campo da pesquisa operacional. Portanto, tinham pleno domínio da teoria e aplicação das técnicas heurísticas de otimização, assim como das técnicas exatas de otimização (as técnicas clássicas de otimização). Particularmente, Feo e Resende conheciam o algoritmo heurístico construtivo e a heurística de busca através de vizinhança (a heurística SDH, *steepest descent heuristic*). Assim, uma análise detalhada do GRASP permite verificar que esse algoritmo é apenas uma junção e uma generalização do AHC de tipo guloso e da heurística SDH. Na análise das heurísticas tradicionais (que apareceram muito antes das metaheurísticas) verificamos que o AHC de tipo guloso e a heurística SDH eram os mais importantes pelo desempenho que apresentam e porque essas heurísticas podiam ser usadas para inventar metaheurísticas através de um processo de junção e de generalização. Logo, o algoritmo GRASP vai ser apresentado e analisado usando os tópicos conceituais já desenvolvidos para o AHC de tipo guloso e da heurística SDH.

2.1.7.1 Teoria Básica de GRASP

O algoritmo GRASP, para um problema genérico, assume a seguinte forma:

- (1) Passo preliminar: Montar os dados do problema. Escolher uma forma de codificação de uma proposta de solução denominada de p . Identificar uma forma de avaliar a qualidade da função objetivo ou equivalente e denominada $f(p)$. Escolher uma

estratégia heurística construtiva (um algoritmo heurístico de tipo guloso a ser usado na fase construtiva) e uma estratégia de busca local para ser usada na fase de busca local (implica escolher uma heurística tipo SDH ou mesmo outra metaheurística, juntamente com as estruturas de vizinhança e os parâmetros correspondentes).

- (2) Implementar a fase de pré-processamento.
- (3) Realizar a fase de busca construtiva.
- (4) Realizar a fase de pós-processamento de busca local. Atualizar a incumbente, caso seja possível.
- (5) Se o critério de parada não for satisfeito, voltar ao passo 3. Caso contrário, pare. A resposta do algoritmo é a incumbente armazenada.

O pré-processamento tenta identificar determinadas subestruturas, isto é, atributos ou conjunto de atributos, que permitem iniciar o processo de busca construtiva de forma mais eficiente ou diminuir o espaço de busca do problema.

A ideia central na fase de pré-processamento é diminuir o espaço de busca, incorporando componentes do problema na solução em construção porque existe a certeza ou quase certeza de que esses componentes fazem parte da solução e descartando outros componentes porque existe certeza ou quase certeza de que esses componentes não devem fazer parte da solução ótima ou quase ótima. Essa estratégia é muito comum e muito usada na otimização clássica, especialmente nas técnicas de solução de problemas de programação (linear) inteira mista. Nesse caso, existem várias estratégias de pré-processamento que permitem fixar variáveis inteiras ou binárias nos valores ótimos antes de iniciar o processo de otimização (por exemplo combinando restrições em programação binária).

Também existem estratégias de geração de restrições que eliminam uma parcela da região factível, porque existe a certeza de que a solução ótima não se encontra nessa parcela da região factível descartada (por exemplo, as restrições substitutas). Adicionalmente, essa estratégia de fixação de variáveis ou redução do espaço de busca, não somente faz parte de uma fase de pré-processamento. Essa estratégia em alguns casos faz parte da própria estratégia central de uma técnica de otimização. Assim, por exemplo, a estratégia central do algoritmo *branch and bound* consiste em relaxar a região factível do problema original (ao relaxar a integralidade das variáveis inteiras) e depois a ideia central é gerar novas restrições para separar de forma disjuntiva a região factível do problema relaxado e diminuir sistematicamente o tamanho dessas regiões factíveis reduzidas até encontrar a solução ótima do problema original. A ideia central nos algoritmos de planos de cortes (como o algoritmo de

planos de cortes de Gomory) também é muito parecida do ponto de vista conceitual, isto é, inicialmente aumentamos a região factível do problema original relaxando a integralidade das variáveis inteiras ou binárias e depois reduzimos sistematicamente a região factível do problema relaxado ao incorporar novos planos de cortes até que o ótimo do problema relaxado seja inteiro e, portanto, também ótimo do problema original.

Finalmente, outra estratégia de fixação de variáveis na otimização clássica é o Teste de Fixação de Variáveis Binárias de Geoffrion, usado no algoritmo de Balas, especializado na resolução de problemas de programação binária.

A fase construtiva do GRASP é apenas um algoritmo heurístico construtivo (AHC) de tipo guloso generalizado. Portanto, na fase construtiva do GRASP, a idéia central é gerar uma solução de boa qualidade tentando contornar as limitações do AHC de tipo guloso. Adicionalmente, a proposta permite gerar um número elevado de soluções de qualidade e, geralmente, diferentes quando o algoritmo é processado repetidas vezes. Para entender a utilidade da fase construtiva do GRASP, vamos analisar de forma resumida as limitações do AHC de tipo guloso.

Um AHC de tipo guloso gera uma solução, geralmente de boa qualidade, através de uma sequência de passos e, em cada passo, adicionamos uma componente da solução em construção, como mencionamos anteriormente. A componente que é incorporada na solução em construção é identificada usando um indicador de sensibilidade.

Em resumo, a fase construtiva do GRASP tenta contornar as limitações de um AHC e também permite gerar um número elevado de propostas de solução diferentes. A lógica fundamental do GRASP, na fase construtiva, consiste em escolher a próxima componente da solução em construção dentre uma lista reduzida de candidatos, chamada de lista RCL. Um algoritmo heurístico construtivo escolheria o primeiro elemento dessa lista RCL. O número de elementos da lista RCL deve ser variável e deve levar em conta a qualidade dos componentes candidatos à adição.

A fase construtiva do algoritmo GRASP apresenta os seguintes passos:

- (1) Escolher a solução inicial que pode ser vazia, isto é, sem adição de componentes, que se transforma na solução em construção.
- (2) Para a solução em construção (com alguns elementos já adicionados) e usando um indicador de sensibilidade, elaborar uma lista RCL com as k componentes mais atrativas. O valor de k é determinado de forma adaptativa como é mostrado adiante.
- (3) Escolher um elemento (componente) dos k elementos existentes na lista RCL e atualizar a solução em construção corrente com a adição da componente escolhida.

- (4) Se a solução em construção corrente representa uma solução factível ou foi satisfeito o critério de parada (sem encontrar uma solução factível) terminar com a fase construtiva. Caso contrário, voltar ao passo 2.

2.1.7.2 A Fase de Busca Local do GRASP

Na formulação inicial do GRASP, a fase de pós-processamento ou de busca local era realizada por uma heurística de busca através de vizinhança, isto é, por uma heurística parecida com a heurística SDH. A ideia central é tentar encontrar uma solução ótima local na vizinhança da solução encontrada na fase construtiva.

Através do tempo o GRASP foi sofisticando a proposta de otimização na fase de busca local, após a fase construtiva. Assim, nas propostas iniciais a proposta era usar uma heurística de busca local como a heurística SDH ou como o processo de intensificação no algoritmo *tabu search*.

Em geral, a fase de busca local pode ser um processo muito simples ou muito complexo. Assim, no caso mais simples pode ser implementada uma heurística tipo SDH. No caso mais sofisticado, pode ser outra metaheurística, tais como *simulated annealing*, o algoritmo genético, *tabu search*, VNS, etc.

Propostas intermediárias podem incorporar estratégias de otimização que fazem parte de metaheurísticas sofisticadas como, por exemplo, a estratégia de *path relinking*, inventada como parte do *tabu search*.

Em suma, o algoritmo GRASP é uma integração e generalização sofisticada de dois algoritmos que já existiam no campo da pesquisa operacional, isto é, do algoritmo heurístico de tipo guloso e da heurística de busca através de vizinhança clássico, que neste trabalho chamamos de heurística SDH. Também mostramos neste trabalho que metaheurísticas, tais como o algoritmo *simulated annealing*, *tabu search* e o algoritmo genético podem ser idealizados como generalizações da heurística SDH. Ao final, pode-se afirmar que o algoritmo GRASP apresenta as seguintes características fundamentais:

- Uma fase de pré-processamento inspirada em conceitos existentes na pesquisa operacional clássica e cujo objetivo é incorporar ou descartar elementos que formam parte da solução ótima com redução do espaço de busca.
- Uma fase construtiva que na verdade é um algoritmo heurístico construtivo generalizado, onde se tenta contornar a característica gulosa do algoritmo heurístico

construtivo e ter a possibilidade de gerar um conjunto elevado de soluções de qualidade, usando uma mesma heurística construtiva.

- Uma fase de pós-processamento que, na verdade, é uma fase de melhoria da solução encontrada na fase construtiva que, originalmente, era implementado usando uma heurística simples de busca através de vizinhança como a heurística SDH.

Entretanto, essa fase de busca local pode ser implementada usando heurísticas ou metaheurísticas sofisticadas como qualquer outra metaheurística existente na literatura especializada.

Adicionalmente, como a fase construtiva e de busca local pode ser repetida, então existe grande possibilidade de encontrar uma solução ótima global ou soluções quase-ótimas. Finalmente, podemos mencionar que, segundo um dos inventores do GRASP, Mauricio Resende, uma das estratégias mais eficientes que podem ser usadas na fase de busca local é a estratégia de *path relinking*.

2.2 O ALGORITMO DE BUSCA EM VIZINHANÇA VARIÁVEL

O VNS é uma metaheurística recente que explora sistematicamente a ideia de mudança local para encontrar soluções ótimas locais e para sair desses ótimos locais. A mudança sistemática da vizinhança dentro de um algoritmo de busca local produz uma metaheurística simples e eficaz para otimização combinatória denominada Busca em Vizinhança Variável. Nesse aspecto fundamental, VNS é significativamente diferente de outras metaheurísticas.

Uma metaheurística é uma estratégia de busca através do espaço de busca de um problema complexo. A busca é realizada através de transições no espaço de busca, a partir de um ponto inicial ou de um conjunto de pontos iniciais. Nesse contexto, a diferença principal entre as diferentes metaheurísticas é a estratégia usada para realizar as transições no espaço de busca.

A maioria das metaheurísticas aceita a degradação da solução corrente (ou do conjunto de soluções correntes) como uma estratégia para sair de uma solução ótima local. O algoritmo VNS não aceita essa possibilidade. O algoritmo VNS muda a vizinhança como uma forma de sair de soluções ótimas locais. Nesse processo a solução corrente também é a incumbente o que não acontece nas outras metaheurísticas. Assim, podemos afirmar que o

algoritmo VNS realiza um conjunto de transições no espaço de busca de um problema e em cada passo a transição é realizada para a nova incumbente. Se o processo encontra um ótimo local, então o algoritmo VNS muda de vizinhança para sair desse ótimo local e passar para a nova incumbente.

Como uma consequência dessa estratégia, se o algoritmo VNS encontra o ótimo global, então a busca fica estagnada nesse ponto de ótimo global sem possibilidade de sair desse ponto. Esse tipo de comportamento não acontece com as outras metaheurísticas.

A estratégia do algoritmo VNS está inspirada em três fatos importantes:

Fato1: O mínimo local em relação a uma estrutura de vizinhança não é necessariamente também mínimo local para outra estrutura de vizinhança.

Fato2: O mínimo global é um mínimo local com respeito a todas as estruturas de vizinhança.

Fato 3: Para muitos problemas, os mínimos locais para várias estruturas de vizinhança apresentam composição muito parecidas.

O último fato é particularmente importante na formulação de um algoritmo VNS. A observação, de caráter empírica, implica que uma solução ótima local fornece informação importante em relação ao ótimo global, especialmente se a solução ótima local for de excelente qualidade. Existe também a observação empírica de que as soluções ótimas locais geralmente estão concentradas em regiões específicas do espaço de busca. Se as soluções ótimas locais estivessem uniformemente distribuídas no espaço de busca, todas as metaheurísticas se tornariam ineficientes. Portanto, se for encontrado um ótimo local da região em que se encontra o ótimo global, então uma metaheurística tipo VNS tem grande chance de encontrar esse ótimo global. Por outro lado, se o ótimo global se encontra em outra região, então a única possibilidade de encontrar o ótimo global é implementar um processo de diversificação. Por esse motivo, um equilíbrio entre intensificação e diversificação no processo de busca pode ser importante em uma metaheurística.

Existe outro aspecto importante que deve fazer parte da lógica de implementação de um algoritmo VNS, que está relacionado com a qualidade de um ótimo local. Um ótimo local da função objetivo de melhor qualidade, não necessariamente pode ser mais adequado para tentar encontrar um ótimo global. Supor que existem duas soluções ótimas locais x_a e x_b em que $f(x_a) < f(x_b)$ para o problema de minimização. Na análise tradicional, podemos concluir que x_a é um ótimo local de melhor qualidade que x_b . Entretanto, se essas soluções devem ser usadas para iniciar (ou reiniciar) o processo de busca, então podemos afirmar que aquela solução com características internas mais próximas da solução ótima global é a mais

adequada para iniciar (ou reiniciar) a busca e, portanto, não necessariamente a solução x_a deve ser escolhida. Logicamente, em condições normais não conhecemos a solução ótima.

Entretanto, existem problemas onde a solução ótima é conhecida e existem vários algoritmos heurísticos para encontrar soluções ótimas locais para esse problema. Nesse contexto, podemos usar a observação anterior para identificar o algoritmo heurístico que produz soluções ótimas locais de melhor qualidade para iniciar a busca, usando o algoritmo VNS.

2.3 DIFERENTES ESTRUTURAS DE BUSCA EM VIZINHANÇA VARIÁVEL

Existem várias formas de implementar o algoritmo VNS que podem ser usadas de formas independentes ou integradas em estruturas VNS mais complexas.

Neste trabalho apresentamos quatro tipos de algoritmos VNS que são os seguintes: (1) VND, (2) RVNS, (3) BVNS e (4) GVNS. (RIGHETTO, 2010) em sua tese de doutorado, cap. 3, páginas 23 a 28, definiu estes algoritmos da forma apresentada adiante.

2.3.1 Algoritmo VND

A forma mais simples de um algoritmo tipo VNS é o VND (*Variable Neighborhood Descent*). O algoritmo VND está baseado no Fato 1 mencionado anteriormente, isto é, um mínimo local para uma estrutura de vizinhança não é necessariamente mínimo local para outra estrutura de vizinhança. Assim, o ótimo local x' de x na vizinhança $N_1(x)$ não é necessariamente igual ao ótimo local x'' de x para a vizinhança $N_2(x)$. O algoritmo VND assume a seguinte forma:

- (1) Selecione um conjunto de estruturas de vizinhança $N_s, s = 1, \dots, s_{max}$. Encontre uma solução inicial x .
- (2) Faça $s = 1$.
- (3) Encontre a melhor solução vizinha x' de x , ($x' \in N_s(x)$).
- (4) Se x' é melhor que x então faça $x = x'$ e $s = 1$ e volte ao passo 3. Caso contrário vá ao passo 5.
- (5) Fazer $s = s + 1$. Se $s \leq s_{max}$ volte ao passo 3. Caso contrário, pare.

Estrutura básica de um algoritmo VND

Entretanto, esse tipo de algoritmo básico pode ser integrado em uma estrutura mais complexa de algoritmo VNS. Quando o algoritmo VND for usado de forma independente devemos priorizar a busca de soluções de excelente qualidade. Por outro lado, se o algoritmo VND é usado em estruturas mais complexas pode ser mais importante encontrar uma boa solução mais rapidamente.

Em relação ao algoritmo VND devemos fazer as seguintes observações:

- O tamanho da vizinhança é um assunto crítico. Se a vizinhança for muito grande então o algoritmo pode se tornar muito lento. Assim, se o algoritmo VND for usado de forma independente podem ser aceitáveis vizinhanças de cardinalidade elevada (um número elevado de vizinhos). Se o algoritmo VND for usado em uma estrutura VNS mais complexa (ver o algoritmo GVNS) então a vizinhança deve ter cardinalidade não muito grande. Em qualquer contexto pode ser recomendável usar técnicas de redução de vizinhança da mesma forma que no algoritmo TS.
- Em relação ao item anterior, podemos modificar o algoritmo VND para que não seja muito lento. Observamos que no passo 3, todos os vizinhos são avaliados para identificar o melhor. Assim, apenas o melhor vizinho é comparado com a solução corrente x . Esse passo pode ser mudado de forma que os vizinhos sejam ordenados e assim que for identificado um vizinho melhor que a solução corrente então, devemos realizar a transição. Nesse contexto, o tamanho da vizinhança não é crítico.
- Na lógica do algoritmo VND está implícito que as vizinhanças mudam de tamanho de forma crescente, isto é, a vizinhança $N1$ tem menos elementos que a vizinhança $N2$ e assim sucessivamente. Assim, a estrutura do algoritmo VND prioriza a vizinhança de menor tamanho, isto é, sempre que for encontrada uma nova incumbente e após executada a transição o algoritmo VND retorna para a vizinhança $N1$. Uma alternativa pode ser avaliar as vizinhanças em sequência, isto é, em cada passo devemos analisar uma estrutura de vizinhança. Essa proposta pode tornar o algoritmo VND mais lento mas enriquece o processo com maior diversificação.
- As estruturas de vizinhança podem não ser suficientes para encontrar o ótimo global. Observamos que as estruturas de vizinhança crescem em complexidade (cada nova estrutura tem um número muito maior de elementos ou vizinhos). Assim, se as estruturas de vizinhança não estão adequadamente projetadas pode não ser possível

encontrar as soluções ótimas de problemas complexos. Portanto, definir de forma adequada as estruturas de vizinhança é crucial em algoritmos tipo VNS.

2.3.2 Algoritmo RVNS

O segundo tipo de algoritmo VNS é o chamado RVNS (*Reduced Variable Neighborhood Search*). Este tipo de algoritmo está inspirado em dois aspectos fundamentais no processo de busca relacionados com a intensificação e a diversificação. Por um lado temos o Fato 3 que afirma que na região de um ótimo local normalmente existem outras soluções ótimas locais que podem ser encontradas a partir de um ótimo local inicial e, portanto, montamos uma estratégia de intensificação para listar esses ótimos locais. Por outro lado, sair de uma região de ótimo local para encontrar um ótimo local de uma região mais distante exige uma estratégia que implique em uma mudança mais radical na caracterização da vizinhança especialmente em problemas que tem característica multimodal. Assim, uma busca que contemple os aspectos de intensificação e diversificação pode permitir encontrar ótimos locais de uma mesma região e pode permitir ao processo de busca sair para ótimos locais de regiões mais distantes do ponto de ótimo local corrente (incumbente). Esse tipo de trabalho foi incorporado no algoritmo RVNS cuja estrutura assume a seguinte forma:

- (1) Selecione um conjunto de estruturas de vizinhança N_k , $k = 1, \dots, k_{\max}$.
 Encontre uma solução inicial x .
 Escolha um critério de parada.
- (2) Faça $k = 1$.
- (3) Gere uma solução x' aleatoriamente da k -ésima vizinhança de x , ($x' \in N_k(x)$).
- (4) Se x' é melhor que x então fazer $x = x'$ e $k = 1$ e, volte ao passo 3. Caso contrário, vá ao passo 5.
- (5) Faça $k = k + 1$. Se $k \leq k_{\max}$ volte ao passo 3. Caso contrário, vá ao passo 6.
- (6) Se o critério de parada for satisfeito, pare. Caso contrário, volte ao passo 2.

Estrutura básica de um algoritmo RVNS

Em relação ao algoritmo RVNS devemos fazer as seguintes observações:

- No passo 3 é escolhido de forma aleatória um vizinho da solução corrente para uma certa estrutura de vizinhança. Lembremos que a solução corrente geralmente é um

mínimo local. Nesse contexto, uma escolha aleatória pretende encontrar uma solução na vizinhança próxima da solução corrente ou uma solução correspondente a uma outra região de ótimo local distante da solução corrente.

- Outro aspecto crucial é quanto devemos insistir na busca da vizinhança de um ótimo local. Por um lado temos o Fato 2 que estipula que em muitos problemas um ótimo local geralmente se encontra muito próximo de outras soluções ótimas locais e localizado sem pequenas regiões do espaço de busca. Portanto, quando for encontrado um ótimo local então essa solução contém informação implícita das outras soluções ótimas locais e provavelmente da própria solução ótima global. Assim, é natural explorar inicialmente nas proximidades de um ótimo local. Entretanto, se essa região de ótimo local se encontra muito distante de outras regiões de ótimo local a estratégia de busca local pode não ser suficiente para sair dessa região de ótimo local. Portanto, pode ser necessário adicionar uma nova estratégia no processo de busca. Essa possibilidade é introduzida no algoritmo RVNS.
- No algoritmo RVNS escolhemos um conjunto de estruturas de vizinhança N_k , $k=1, \dots, k_{\max}$ caracterizadas em relação com a solução corrente x (que pode ou não ser um ótimo local). Geralmente essas vizinhanças estão aninhadas, isto é, os elementos da vizinhança N_1 também são elementos na vizinhança N_2 e, assim sucessivamente. Nesse contexto, escolhemos aleatoriamente um vizinho na primeira vizinhança. Se esse vizinho escolhido é melhor que a solução corrente então esse vizinho a substitui e reiniciamos o processo de busca a partir da nova incumbente. Em caso contrário, passamos para o seguinte nível de vizinhança. Se for realizada a transição em um nível superior de vizinhança então a busca volta para o primeiro nível. Depois que todos os níveis de vizinhança são avaliados o processo volta para o primeiro nível e o processo de busca deve continuar até cumprir um critério de parada.

Como as vizinhanças são cada vez maiores em tamanho então o algoritmo RVNS explora com maior frequência vizinhanças de níveis elevados já que cada nível de vizinhança não é avaliado inteiramente como acontece com o algoritmo VND. No algoritmo RVNS escolhemos apenas um vizinho e se esse vizinho for de melhor qualidade então voltamos para o primeiro nível de vizinhança e continuamos com o processo de intensificação. Em caso contrário passamos para o seguinte nível. Assim, a vizinhança no algoritmo RVNS pode ser maior em tamanho e complexidade quando comparado com o algoritmo VND.

Devemos observar que o algoritmo RVNS produz uma escolha de vizinhos mais dinâmica, escolhendo vizinhos de todas as estruturas de vizinhança (diversificação) e priorizando a primeira estrutura de vizinhança (intensificação) nas fases iniciais da busca. Entretanto, uma componente importante da estrutura RVNS é a capacidade de encontrar novas regiões promissoras a partir de um ótimo local. O algoritmo RVNS também pode ser usado de forma independente ou pode ser integrado em uma estrutura mais complexa de algoritmo VNS.

2.3.3 Algoritmo BVNS

O terceiro tipo de algoritmo VNS é o chamado BVNS (*Basic Variable Neighborhood Search*). O algoritmo BVNS tenta integrar as características dos algoritmos VND e RVNS. Por um lado o algoritmo VND permite encontrar soluções ótimas locais de boa qualidade e por outro lado, o algoritmo RVNS permite encontrar novas regiões promissoras a partir de um ótimo local. Assim, juntando essas características podem ser formulados dois tipos de algoritmos VNS que geralmente apresentam excelente desempenho. Esses algoritmos são chamados de *Basic Variable Neighborhood Search* (BVNS) e *General Variable Neighborhood Search* (GVNS). A estrutura do algoritmo BVNS assume a seguinte forma:

- (1) Selecione um conjunto de estruturas de vizinhança N_k , $k = 1, \dots, k_{\max}$.
 Encontre uma solução inicial x .
 Escolha um critério de parada.
- (2) Faça $k = 1$.
- (3) Gere uma solução x' aleatoriamente da k -ésima vizinhança de x , ($x' \in N_k(x)$)
- (4) Implemente uma busca local a partir de x' . Seja x'' o ótimo local desse processo de busca.
- (5) Se x'' é melhor que x então faça $x = x''$, $k = 1$ e, volte ao passo 3. Caso contrário, vá ao passo 6.
- (6) Faça $k = k + 1$. Se $k \leq k_{\max}$ volte ao passo 3. Caso contrário, vá ao passo 7.
- (7) Se o critério de parada for satisfeito, pare. Caso contrário, volte ao passo 2.

Estrutura básica de um algoritmo BVNS

Em relação ao algoritmo BVNS devemos fazer as seguintes observações:

O algoritmo BVNS combina a busca local com mudanças sistemáticas da vizinhança em torno do ótimo local encontrado. A lógica de trabalho do algoritmo BVNS é muito interessante.

Inicialmente devemos escolher as k estruturas de vizinhança. O processo de otimização é iniciado de uma solução x e na vizinhança $N_1(x)$. Na sequência escolhemos de forma aleatória um vizinho x' de x em $N_1(x)$. A partir de x' é iniciado um processo de busca local para encontrar um ótimo local x'' .

Nesse contexto podem acontecer 3 casos: (1) se $x''=x$ significa que x já era o ótimo local dessa região e, portanto, devemos mudar para outro nível de vizinhança ($N_2(x)$ neste caso); (2) se x'' é de pior qualidade que x então foi encontrado um ótimo local de pior qualidade que a incumbente x e também devemos mudar de vizinhança; e (3) se x'' é de melhor qualidade que x então significa que foi encontrada uma solução melhor que a incumbente e, portanto, devemos atualizá-la e reiniciar a busca permanecendo na vizinhança $N_1(x)$. Se no processo de otimização atingimos o último nível de vizinhança sem encontrar uma solução melhor que a incumbente então o processo volta a busca no primeiro nível de vizinhança. O processo termina quando se cumpre um critério de parada. O critério de parada pode ser especificado de várias formas tais como: (1) tempo máximo de computação, (2) número máximo de iterações ou (3) um número máximo de iterações sem a mudança da incumbente.

Em qualquer iteração do processo sempre que a busca local encontra uma nova incumbente voltamos para a vizinhança $N_1(x)$ e sempre que a busca local encontra uma solução de igual ou pior qualidade que a incumbente então mudamos para uma vizinhança mais complexa. Essa estratégia e a escolha aleatória do vizinho da incumbente x evita ciclagem e permite encontrar ótimos locais distantes da incumbente corrente.

A estratégia de busca local do passo 4 do algoritmo BVNS pode ser qualquer heurística. Entretanto a busca local também pode usar um algoritmo VNS. Assim, o BVNS pode ser transformado em um algoritmo mais geral chamado GVNS (*General Variable Neighborhood Search*). O GVNS é obtido generalizando o BVNS simplesmente usando um algoritmo VND como estratégia de busca local e usando um algoritmo RVNS para melhorar a solução inicial usada para iniciar a busca.

2.3.4 Algoritmo GVNS

O algoritmo GVNS é uma generalização do BVNS em que a busca local é realizada por um algoritmo VND e a solução inicial é melhorada por um RVNS. Assim, o algoritmo GVNS assume a seguinte forma:

- (1) Selecione um conjunto de estruturas de vizinhanças N_k , $k = 1, \dots, k_{\max}$ que deve ser usado na estrutura principal do algoritmo GVNS.
Escolha um critério de parada para a estrutura principal do algoritmo GVNS.
Selecione um conjunto de estruturas de vizinhança N_p , $p = 1, \dots, p_{\max}$ que deve ser usado na estrutura do algoritmo RVNS.
Escolha um critério de parada para o algoritmo RVNS.
Selecione um conjunto de estruturas de vizinhança N_s , $s = 1, \dots, s_{\max}$ que deve ser usado na estrutura do algoritmo VND.
Encontre uma solução inicial x .
- (2) Faça $p = 1$.
- (3) Gere uma solução x' aleatoriamente da p -ésima vizinhança de x , ($x' \in N_p(x)$).
- (4) Se x' é melhor que x então faça $x = x'$, $p = 1$ e, volte ao passo 3. Caso contrário, vá ao passo 5.
- (5) Faça $p = p + 1$. Se $p \leq p_{\max}$ volte ao passo 3. Caso contrário, vá ao passo 6.
- (6) Se o critério de parada não for satisfeito voltar ao passo 2. Caso contrário, ir ao passo 7 (foi encontrada a melhor solução inicial obtida pelo algoritmo RVNS).
- (7) Faça $k = 1$.
- (8) Gere uma solução x' aleatoriamente da k -ésima vizinhança de x , ($x' \in N_k(x)$).
- (9) Faça $s = 1$.
- (10) Encontre o melhor vizinho x'' de x em $N_s(x)$.
- (11) Se x'' é melhor que x' então faça $x' = x''$, $s = 1$ e, voltar ao passo 10. Caso contrário, vá ao passo 12.
- (12) Faça $s = s + 1$. Se $s \leq s_{\max}$ voltar ao passo 10. Caso contrário, vá ao passo 13 (foi encontrado um ótimo local pelo algoritmo VND).
- (13) Se x' é melhor que x então faça $x = x'$, $k = 1$ e, volte ao passo 8. Caso contrário, vá ao passo 14.
- (14) $k = k + 1$
Se $k \leq k_{\max}$ volte ao passo 8. Caso contrário, vá ao passo 15.
- (15) Se o critério de parada for satisfeito, pare. Caso contrário, volte ao passo 7.

Estrutura básica de um algoritmo GVNS

Todas as observações realizadas para o algoritmo BVNS permanecem válidas no GVNS. Como foi mencionado anteriormente, a mudança fundamental está na fase de melhoria inicial da solução inicial usando um RVNS e na fase de busca local que é realizada usando um algoritmo VND. Em relação ao algoritmo GVNS, podemos fazer as seguintes observações:

As três estruturas de vizinhança presentes no algoritmo GVNS devem ser adequadamente projetadas para que o algoritmo GVNS encontre a solução ótima global ou soluções quase-ótimas. Nesse trabalho, notamos que a ideia fundamental consiste em encontrar as melhores regiões de ótimo local e o mais rapidamente possível. Na verdade a única forma de garantir que o processo não fique estacionado em um ótimo local é quando a união de todas as estruturas de vizinhança reproduzem o espaço de busca completo.

Existe ainda a questão de como proceder se as estruturas de vizinhança geram vizinhos ineficazes. Uma alternativa simples é descartar as soluções vizinhas ineficazes. Outra alternativa, muito usada em outras metaheurísticas, é penalizar a função objetivo incorporando as ineficiências na função objetivo.

3 ALGORITMO VNS APLICADO AO PROBLEMA DE GRADE HORÁRIA

Neste capítulo mostramos o tipo de algoritmo VNS projetado e os testes realizados usando os dados de quatro casos disponíveis no site da competição internacional de grade horária. Como na implementação de toda metaheurística, analisamos em detalhe a forma em que são formatados os dados do problema, a forma de representação de uma proposta de solução, a forma de calcular a função objetivo, a caracterização das estruturas de vizinhança, a forma de encontrar uma solução inicial, a estrutura geral do algoritmo VNS implementado e, finalmente, os testes com as quatro instâncias disponíveis no site da competição internacional de grade horária.

3.1 A FORMATAÇÃO DOS DADOS DO PROBLEMA

Para todos os casos usados nos testes as 3 primeiras instâncias disponíveis na competição internacional conta com número de eventos igual a 400, número de salas disponíveis igual a 10 e número de estudantes igual a 200, com exceção da 4ª instância que contém 350 eventos, 10 salas de aula disponíveis e 350 alunos. A estrutura dos dados usados como dados de entrada para ser resolvido pelo algoritmo VNS assume a forma apresentada no Apêndice (Ver tabelas 7, 8, 9 e 10) da *Competição Internacional Timetabling* (ITC).

Na estrutura de dados apresentada no apêndice aparece, para cada evento, o número de alunos e a identificação dos alunos interessados em cada evento. Adicionalmente, para cada evento, aparece a informação do número de salas em que pode ser programado um evento e a identificação dessas salas em que pode ser programado o evento.

3.2 A REPRESENTAÇÃO DE UMA PROPOSTA DE SOLUÇÃO

A representação de uma proposta de solução é fundamental para a implementação adequada de uma metaheurística. A representação de uma proposta de solução pode estar codificada e, portanto, precisa de uma decodificação para encontrar a solução real ou pode identificar diretamente uma proposta de solução (sem precisar de decodificação).

Neste trabalho apresentamos duas propostas de representação de uma proposta de solução e essas duas formas de representação são usadas em contextos diferentes dentro da estrutura VNS. A primeira forma de representação de uma proposta de solução é através de dois vetores de tamanho igual ao número de eventos (400 eventos nos casos testados). Um vetor armazena a sala em que é programado o evento e o outro vetor armazena a hora em que é programado o evento. Assim, por exemplo, uma proposta de solução, nesse contexto, assume a seguinte forma:

Tabela nº 1:

Hora		Sala		eventos
nho(.)	29	7	1	
	29	8	2	
	30	10	3	
	34	2	4	
	:	:	:	
	26	9	399	
	1	7	400	

Solução factível: $v = 318$ (RD1=150, RD2=15, RD3=153)

A proposta de solução apresentada anteriormente corresponde a uma solução com função objetivo $v = 318$ mostrada adiante (sem restrições obrigatórias violadas, e com valores de RD1= 150, RD2=15 e RD3 = 153). A este tipo de proposta de representação de uma proposta de solução denominamos de representação de uma proposta de solução nº.1.

A outra forma de representação de uma proposta de solução é um arranjo matricial de dimensão $nsal \times ngr$ em que $nsal$ é o número de salas e ngr é o número de horas disponíveis na programação semanal. Nos testes realizados $nsal = 10$ e $ngr = 45$. Em outras palavras, a representação de uma proposta de solução é a própria grade horária. Os elementos desse arranjo são os números de 1 a 400, representando cada um dos eventos, e valores iguais a zero indicando que em uma sala e a uma determinada hora não existe evento programado. Assim, o elemento (i,j) desse arranjo matricial indica o evento programado na sala i e na hora j. Exemplificando, se $(2, 3) = 228$ então o evento 228 foi programado na sala 2 e na hora 3. Por outro lado se $(3, 27) = 0$ então não existe evento programado na sala 3 e na hora 27. Essa proposta corresponde a uma solução com função objetivo $v = 318$ mostrada anteriormente de outra forma (sem restrições obrigatórias violadas, e com valores de RD1= 150, RD2=15 e RD3 = 153).

A esse tipo de proposta de representação de uma proposta de solução denominamos de representação de uma proposta de solução nº 2. (ver tabela nº 2 adiante).

Na tabela nº. 1 apresentamos uma proposta de solução típica onde é apresentado a transposta do arranjo. As duas propostas de solução são usadas na implementação do algoritmo VNS. Deve-se observar que embora as duas propostas representem a mesma solução, essas duas propostas não são equivalentes para uso na implementação de metaheurísticas. Assim, por exemplo, o tipo de vizinhança que pode ser implementada em uma proposta de representação não necessariamente pode ser implementada na outra proposta de representação de uma proposta de solução. Na parte de implementação computacional são mostradas essas diferenças, assim como em outros detalhes de implementação computacional.

3.3 A FUNÇÃO OBJETIVO DE UMA PROPOSTA DE SOLUÇÃO

Como já foi mencionado anteriormente, na implementação de uma metaheurística para resolver o problema de horários a função objetivo deve ser quantificada como uma medida das infactibilidades obrigatórias e não obrigatórias mas desejáveis. Portanto, a função objetivo assume a seguinte forma:

$$\min v = \alpha NRO + \beta NRD$$

onde NRO é o número de restrições obrigatórias de uma proposta de solução, NRD é o número de restrições não obrigatórias mas desejáveis de uma proposta de solução e α e β são parâmetros que devem ser escolhidos. Neste trabalho consideramos que uma proposta de solução não deve ter restrições obrigatórias violadas. Adicionalmente, consideramos que entre as propostas de solução que não têm restrições obrigatórias violadas a melhor delas é aquela que tem o menor número de restrições não obrigatórias (mas desejáveis) violadas. Por esse motivo, na pesquisa foram usados valores de $\alpha = 1000$ e $\beta = 1$.

3.3.1 Determinação da Função Objetivo de uma Proposta de Solução

Uma tarefa fundamental quando pretendemos resolver o problema de programação de horários é a determinação do valor da função objetivo de uma proposta de solução conhecida. Essa tarefa deve ser realizada de forma rápida e eficiente em termos de tempo de processamento. Para ganhar intimidade, encontramos a função objetivo de 3 propostas de solução para o exemplo ilustrativo mostrado no Capítulo 1 e que mostramos novamente nas tabelas nº.3.1 e 3.2 com novo formato.

Tabela nº. 3.1(Relaciona alunos com eventos):

Evento	Nº de Alunos	Identificação dos Alunos
1	2	1 3
2	3	1 2 4
3	3	2 4 5
4	4	2 3 4 5

Tabela nº. 3.2(Relaciona salas com eventos):

Evento	Nº de Salas	Identificação das Salas
1	1	1
2	2	1 2
3	1	2
4	1	2

Adicionalmente, devemos observar que nesse exemplo ilustrativo existe apenas um dia de aula, isto é, $ngr = 9$.

Para resolver os exemplos ilustrativos apresentamos, de forma resumida, a forma em que se determina cada parcela da função objetivo.

- (1) Encontrando as restrições obrigatórias tipo 1: RO1: Para encontrar o número de restrições obrigatórias tipo 1, usando a representação da proposta de solução nº.1, apenas precisamos analisar a proposta de solução armazenada nos vetores $nho(.)$ e $nsa(.)$. Assim, se verificarmos que $nho(i) = nho(k)$ e $nsa(i) = nsa(k)$ então concluímos que os eventos i e k foram programados na mesma sala e na mesma hora e, portanto, existe conflito tipo 1. Por outro lado, para a proposta de

solução nº. 2, se existe apenas um elemento alocado em cada célula do arranjo matricial, então não existe restrição RO1. Adicionalmente, deve-se observar que é muito simples montar um arranjo matricial alocando apenas um evento em cada célula e se as estruturas de vizinhança respeitam essa estrutura, então esse tipo de proposta de solução nunca viola a restrição RO1.

(2) Encontrando as restrições obrigatórias tipo 2: RO2

Para encontrar o número de restrições obrigatórias tipo 2 apenas precisamos analisar o vetor $nsa(.)$ da proposta de solução. Assim, para cada evento devemos verificar se esse evento se encontra programado na sala adequada, o que pode ser verificado percorrendo uma linha do banco de dados relacionados com as salas adequadas para o evento em análise. Entretanto, deve-se observar que para cada elemento do vetor $nsa(.)$ devemos percorrer uma linha da matriz que relaciona os eventos com as salas adequadas para programar esse evento. Por outro lado, quando usamos a representação de uma proposta de solução nº.2, vamos preferir realizar uma alocação de cada evento na sala permitida. Assim, também é relativamente fácil montar estruturas de vizinhança que mudam os eventos dentro do arranjo matricial de forma que sempre um evento seja deslocado para uma sala permitida para esse evento. Assim, nessa forma de representação e quando usamos estruturas de vizinhança adequadas, nunca aparecem violações das restrições RO1 e RO2.

(3) Encontrando as restrições obrigatórias tipo 3: RO3:

Para encontrar o número de restrições obrigatórias tipo 3 apenas precisamos analisar o vetor $nho(.)$ da proposta de solução e a matriz DM. A matriz DM é montada a partir dos dados de entrada e não varia. Essa matriz indica o número de alunos comuns entre dois eventos. Na matriz DM, o elemento $DM(i,j)$ indica o número de alunos comuns entre os eventos i e j para $i \neq j$ e $DM(i,i)$ indica o número de alunos programados no evento i . Portanto, percorrendo o vetor $nho(.)$ se verificamos que $nho(i) = nho(k)$, então os eventos i e k estão programados na mesma hora e, portanto, o valor armazenado em $DM(i,k)$ determina o número de alunos que tem os eventos i e j programados no mesmo horário. Por outro lado, quando usamos a representação de uma proposta de solução nº. 2, analisamos de forma separada cada coluna do arranjo matricial (formado por 10 elementos).

Assim, para cada par de elementos diferentes de zero existentes na coluna em análise (dois eventos) podemos encontrar o número de conflitos simplesmente observando uma posição da matriz DM. Portanto, é possível gerar propostas de solução que violem apenas a restrição RO3 quando usamos a representação de uma proposta de solução nº. 2

(4) Encontrando as restrições não obrigatórias mas desejáveis tipo 1: RD1:

As restrições que não são obrigatórias de serem cumpridas, mas seriam desejáveis de que sejam cumpridas podem ser encontradas simultaneamente. Uma alternativa consiste em montar a matriz auxiliar de horário dos estudantes DH para a proposta de solução como é mostrado adiante. Analisando essa matriz e de forma simultânea, pode-se encontrar o número de restrições violadas não obrigatórias, mas desejáveis. Obviamente, cada tipo de restrição pode ser analisado usando 3 laços Do, diferentes, mas o esforço de processamento pode aumentar de forma significativa. Outra alternativa mais eficiente, consiste em montar a matriz auxiliar DR para cada proposta de solução como é mostrada adiante. Essa matriz é de menor tamanho e, portanto, exige menor tempo de processamento. Deve-se observar que a programação desse módulo, com a matriz DR é mais trabalhosa.

(5) Encontrando as restrições não obrigatórias mas desejáveis tipo 2: RD2:

Podem ser encontradas apenas analisando a matriz auxiliar DH do horário dos estudantes para cada proposta de solução ou também analisando a matriz DR dos estudantes para cada proposta de solução.

(6) Encontrando as restrições não obrigatórias mas desejáveis tipo 3: RD3:

Podem ser encontradas apenas analisando a matriz auxiliar DH do horário dos estudantes para cada proposta de solução ou também analisando a matriz DR dos estudantes para cada proposta de solução que também são usadas nos dois casos anteriores. Caso sejam desenvolvidos módulos diferentes de programação para cada tipo de restrição não obrigatória mas desejável então a restrição RD3: pode ser mais rapidamente encontrada analisando apenas o vetor $nho(.)$. Assim, se para o evento i o valor de $nho(i)$ é múltiplo de 9, então o valor existente em $DM(i,i)$ indica o número de alunos que tem aula no último horário do dia.

Para mostrar a forma em que se encontra a função objetivo mostramos 3 exemplos ilustrativos usando os dados da Tabela nº. 3.1 e 3.2.

Exemplo 1: A proposta de solução assume a seguinte forma:

Usando a proposta de representação nº.1 a solução assume a seguinte forma:

	Hora		Sala		
nho(.) =	3	nsa(.) =	1	1	eventos
	1		2	2	
	6		2	3	
	6		1	4	

Usando a proposta de representação nº.2 assume a seguinte forma:

		Hora								
		1	2	3	4	5	6	7	8	9
Sala	1			1			4			
	2	2					3			

Neste caso a função objetivo assume a seguinte forma:

RO1= 0 RO2=1 RO3=3

RD1= 0 RD2=0 RD3=0

$v = 4000$

A solução proposta apresenta 4 restrições obrigatórias violadas (o evento 4 está programado em sala inadequada e os eventos 3 e 4, programados na mesma hora, têm 3 alunos comuns) e, portanto, representa uma proposta de solução inaceitável para implementação prática.

Exemplo 2: A proposta de solução assume a seguinte forma:

Usando a proposta de representação nº.1 assume a seguinte forma:

	Hora		Sala		
nho(.) =	8	nsa(.) =	1	1	eventos
	9		1	2	
	8		2	3	
	7		2	4	

Usando a proposta de representação n°. 2 assume a seguinte forma:

		Hora								
		1	2	3	4	5	6	7	8	9
Sala	1			1					1	2
	2							4	3	

Neste caso a função objetivo assume a seguinte forma:

$$RO1 = 0 \quad RO2 = 0 \quad RO3 = 0$$

$$RD1 = 0 \quad RD2 = 0 \quad RD3 = 3$$

$$v = 3$$

A solução proposta não apresenta restrições obrigatórias violadas e, portanto, representa uma proposta de solução aceitável para implementação prática. Entretanto, essa proposta de solução apresenta 3 restrições não obrigatórias violadas, mas seria desejável uma proposta sem restrições não obrigatórias violadas. Assim, essa proposta de solução pode ser implementada na prática mas podem existir soluções melhores.

Exemplo 3: Terceira proposta de solução.

Usando a proposta de representação n°. 1 assume a seguinte forma:

		Hora				Sala				
		1	2	3	4	1	2	3	4	
nho(.) =		8				1				1
		5				1				2
		8				2				3
		7				2				4

nsa(.) =

eventos

Usando a proposta de representação n°. 2 assume a seguinte forma:

		Hora								
		1	2	3	4	5	6	7	8	9
Sala	1					2			1	
	2							4	3	

Neste caso a função objetivo assume a seguinte forma:

$$RO1 = 0 \quad RO2 = 0 \quad RO3 = 0$$

$$RD1 = 0 \quad RD2 = 0 \quad RD3 = 0$$

$$v = 0$$

A solução proposta não apresenta restrições obrigatórias violadas e, portanto, representa uma proposta de solução aceitável para implementação prática. Adicionalmente, essa proposta de solução não apresenta restrições não obrigatórias violadas. Portanto, essa proposta de solução é ótima.

3.3.2 Escolha de matrizes auxiliares

Para realizar determinados cálculos de forma mais eficiente e também para recuperar informações implícitas existentes na proposta de solução pode ser oportuno a definição e escolha de matrizes auxiliares. Essas matrizes podem ajudar muito no trabalho de processamento e algumas delas são estáticas (não mudam durante o processamento) e outras são dinâmicas e, portanto, mudam durante o processamento.

Matriz que identifica o número de estudantes comuns a dois eventos:

No processo de otimização existe a necessidade de conhecer o número de alunos existentes em cada evento e o número de alunos em comum que existe entre dois eventos. Essa informação está disponível na matriz D, mas precisa de processamento para obter essa informação. Como esse tipo de informação é constantemente requerida ao avaliar a função objetivo de uma proposta de solução, então montamos a matriz DM a partir da informação armazenada na matriz D. Para o exemplo ilustrativo, mostrado anteriormente, com os dados armazenados na matriz D, a matriz DM assume a seguinte forma:

		eventos					
		1	2	3	4		
DM=	2	1	0	1	1	eventos	1
	1	3	2	2	2		2
	0	2	3	3	3		3
	1	2	3	4	4		4

Na matriz DM, o valor da posição (i; i) informa o número de alunos matriculados no evento i e o valor da posição (i; j) informa o número de alunos comuns matriculados nos eventos i e j. Assim, na matriz DM mostrada, o evento 2 tem 3 alunos matriculados e, os eventos 2 e 3 têm 2 alunos em comum. A matriz DM é estática e, portanto, é montada uma única vez. A matriz DM também pode ser facilmente montada a partir da primeira estrutura de dados mostrada na tabela nº. 2 e também nos dados mostrados nos Anexos.

Matriz de horário dos estudantes:

Para cada proposta de solução podemos precisar também do horário dos estudantes. Essa informação se encontra implícita depois que for encontrada uma proposta de solução identificada pelos vetores nho e nsa . Entretanto, para ter essa informação de forma explícita, devemos cruzar a informação existente na proposta de solução nos vetores nho e nsa com a informação existente na matriz D . Essa informação armazenamos na matriz DH e, para a proposta de solução mostrada no exemplo 2, assume a seguinte forma:

		Horas									
		1	2	3	4	5	6	7	8	9	
DH=	1	0	0	0	0	0	0	0	1	1	1
	2	0	0	0	0	0	0	1	1	1	2
	3	0	0	0	0	0	0	1	1	0	3
	4	0	0	0	0	0	0	1	1	1	4
	5	0	0	0	0	0	0	1	1	0	5

estudantes

Na matriz DH , se o valor da posição $(i; j)$ for igual a 1 então o estudante i tem aula no horário j e se for 0 então o estudante i não tem aula no horário j . A matriz DH pode ser usado para encontrar todas as restrições não obrigatórias mas desejáveis que se encontram violadas. Essa matriz é dinâmica, isto é, existe uma matriz DH para cada proposta de solução.

Matriz de horário dos estudantes modificada:

Para cada proposta de solução podemos precisar também do horário dos estudantes de forma que sejam encontradas as restrições não obrigatórias mas desejáveis que se encontram violadas (RD) de forma mais rápida. Assim, sugerimos montar a matriz de horário dos estudantes modificado, chamada de matriz DR . Essa matriz tem a mesma informação existente na matriz de horário dos estudantes, mas de forma mais compacta.

Para cada proposta de solução, a matriz DR é montada usando a informação existente no vetor $nho(.)$ e na matriz D . Cada linha da matriz DR indica a hora em que um aluno tem aula devido a um evento. Como o número de eventos dos estudantes é variável, então é necessário um vetor auxiliar $ndr(.)$ que indica o número de eventos em que participa cada estudante.

A forma da matriz DR e, para a proposta de solução mostrada no exemplo 2, assume a seguinte forma:

Hora da grade

$$\mathbf{DR} = \begin{array}{|c|c|c|} \hline 8 & 9 & \\ \hline 7 & 8 & 9 \\ \hline 7 & 8 & \\ \hline 7 & 8 & 9 \\ \hline 7 & 8 & \\ \hline \end{array} \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \text{ estudantes}$$

$$\mathbf{ndr} = \begin{array}{|c|} \hline 2 \\ \hline 3 \\ \hline 2 \\ \hline 3 \\ \hline 2 \\ \hline \end{array} \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}$$

Analisando a linha 1 da matriz DR e a posição 1 do vetor $\mathbf{ndr}(\cdot)$ podemos concluir que o aluno 1 está interessado em dois eventos que se encontram programados nos horários 8 e 9 da grade horária para a proposta de solução analisada. Assim, se a posição $(i; j)$ da matriz DR para todo $j \leq \mathbf{ndr}(i)$, assume o valor $\mathbf{DR}(i; j) = p$, então o aluno i tem aula no horário p . Os elementos de uma linha da matriz DR devem ser ordenados de forma crescente para facilitar o cálculo do número de restrições não obrigatórias mais desejáveis que se encontram violadas.

3.4 PROPOSTA PARA GERAR UMA SOLUÇÃO INICIAL

Para iniciar o processo de otimização do problema de grade horária, deve-se ter inicialmente disponível uma solução inicial. Na maioria de problemas de otimização essa solução inicial é geralmente uma solução factível e geralmente de excelente qualidade. Na maioria dos problemas de otimização, como acontece em problemas da engenharia, essa solução inicial é encontrada usando um algoritmo heurístico construtivo de excelente qualidade. Nesse tipo de problemas geralmente é relativamente fácil encontrar uma solução de excelente qualidade. Entretanto, existem problemas altamente restritos e com poucas soluções factíveis. Nesse caso pode ser muito difícil encontrar uma solução factível. Esse é o caso do problema de grade horária. A maneira de ilustração devemos observar nos dados mostrados no exemplo ilustrativo no apêndice onde aparece um problema com 10 salas de aula e com 45 horas de disponibilidade de cada sala. Assim, a disponibilidade total das salas é de 450 horas e, portanto, pode-se programar um máximo de 450 eventos. No mesmo exemplo, pretende-se programar 400 eventos. Isso significa que existe uma folga de apenas 50 horas em disponibilidade de salas. Adicionalmente, se levarmos em conta que os eventos que podem ser programados apenas em alguns tipos de salas, então é fácil verificar que existem grandes

dificuldades em encontrar uma solução factível e, adicionalmente, devemos levar em conta que um aluno não pode ser programado em mais de um evento no mesmo horário.

Na geração de uma proposta de solução inicial, pretende-se apenas gerar uma solução inicial que não viole as restrições obrigatórias ou viole o menor número de restrições desse tipo. Portanto, nesse processo não são levadas em conta as restrições não obrigatórias. Levando em conta as observações anteriores foi montada uma estratégia para encontrar uma solução inicial da seguinte forma:

- (1) Gerar uma proposta de solução inicial que não viole as restrições obrigatórias RO1 e RO2. Geralmente, é fácil gerar uma proposta de solução que não violem as restrições RO1 e RO2, mas que violam um conjunto reduzido de restrições tipo RO3.
- (2) Montar um conjunto de estratégias para eliminar as restrições tipo RO3 na proposta de solução em construção.

A primeira estratégia de gerar uma solução inicial sem violar as restrições RO1 e RO2 e que usa a representação de uma proposta de solução nº. 2 (um arranjo matricial) é implementada da seguinte forma:

- (1) Inicialmente são alocados os eventos que podem ser programados em apenas uma única sala. Devemos observar que, por exemplo, para a instância nº. 2 mostrada no apêndice existem 208 eventos que podem ser programadas em uma única sala. Assim, esse tipo de restrição é altamente restritiva.
- (2) Continuando são alocados no arranjo matricial os eventos que podem ser programados em duas salas e assim sucessivamente até alocar todos os eventos. Obviamente, os últimos eventos alocados no arranjo matricial são aqueles que aceitam ser programados no maior número de salas. Essa estratégia permite que todos os eventos sejam alocados em salas adequadas.
- (3) Quando um evento for escolhido para ser alocado no arranjo matricial devemos ter uma estratégia que permita encontrar muitas soluções iniciais diferentes e ao mesmo tempo gere o menor número de violações das restrições obrigatórias RO3. Assim, neste trabalho foi implementada a seguinte estratégia:
 - Uma vez escolhido o evento a ser alocado, então devemos identificar as salas em que pode ser alocado o evento sem violar a restrição RO2. Caso existam várias salas em que o evento pode ser alocado, então o evento é alocado na sala mais desocupada e na hora em que existe o menor número de conflitos tipo RO3 com os eventos já alocados nessa hora.

- Na estratégia anterior, caso existam várias possibilidades com o mesmo número de conflitos, então a alocação do evento é realizada de forma aleatória entre esse conjunto reduzido de candidatos.
- A estratégia anterior permite programar os eventos sempre naquela sala em que pode ser programada e que apresenta a menor taxa de ocupação entre as salas em que o evento pode ser programado. Essa estratégia permite encontrar uma solução inicial em que todos os eventos são alocados e sem violar as restrições tipo $RO1$ e $RO2$ e que, adicionalmente, viola um número reduzido de restrições tipo $RO3$.
- Como foi mencionado anteriormente, nessa estratégia não são levadas em conta as violações das restrições secundárias, isto é, das restrições $RD1$, $RD2$ e $RD3$.

A maneira de ilustração, na tabela nº. 4 apresentamos uma solução inicial encontrada com a estratégia mostrada anteriormente. Essa proposta não apresenta violações das restrições tipo $RO1$ e $RO2$. Adicionalmente, apresenta apenas 28 conflitos tipo $RO3$. Em outras palavras, essa proposta de solução tem valor objetivo $v = 28769$ com valores de $RO1 = 0$, $RO2 = 0$, $RO3 = 28$, $RD1 = 320$, $RD2 = 62$, $RD3 = 387$.

Adicionalmente, deve-se observar que na proposta de solução mostrada na tabela nº 4, todas as salas apresentam disponibilidade, sendo que a maioria das salas apresentam 5 horas disponíveis (salas 4, 5, 6, 7, 8, 9 e 10) e a menor disponibilidade é de 4 horas (salas 1 e 3). Assim, a estratégia permite gerar uma proposta de solução sem violar as restrições tipo $RO1$ e $RO2$ e, adicionalmente, o número de restrições tipo $RO3$ violadas é pequena. Devemos ressaltar de que não existem restrições tipo $RO1$ violadas porque existe apenas um evento alocado em cada elemento do arranjo matricial (nesse caso não existe a possibilidade de que dois ou mais eventos sejam alocados na mesma sala e na mesma hora). Também não existem violações de restrições tipo $RO2$ porque cada evento foi alocado em uma sala permitida e no final ainda existe folga em todas as salas para alocar eventos.

Tabela nº 4:Solução infactível: $v = 28769$ (RO1 = 0, RO2 = 0, RO3 = 28) (RD1= 320, RD2= 62, RD3= 387)

	Salas									
	1	2	3	4	5	6	7	8	9	10
Horas	1	2	3	4	5	6	7	8	9	10
1	48	0	91	215	60	393	95	330	208	277
2	22	222	181	0	237	217	50	0	294	27
3	275	0	368	377	0	305	123	79	67	200
4	108	165	101	262	41	0	104	116	247	357
5	0	121	366	72	156	392	30	288	355	46
6	81	228	0	376	163	394	214	315	137	190
7	327	207	396	110	185	145	148	0	399	17
8	209	0	370	180	271	391	159	284	0	295
9	98	0	0	122	351	374	188	303	319	112
10	45	138	236	133	164	149	350	63	329	231
11	76	280	160	64	126	118	54	0	38	102
12	184	285	373	175	312	201	223	299	216	259
13	161	43	344	129	383	11	1	2	80	291
14	65	333	66	0	69	289	0	240	55	290
15	9	287	151	331	0	74	85	0	109	144
16	311	4	372	106	0	0	248	297	143	153
17	304	42	326	94	389	0	93	270	263	5
18	321	174	265	238	202	269	253	82	115	194
19	0	272	203	31	179	192	251	354	88	233
20	363	0	378	139	339	52	0	173	292	244
21	197	20	150	167	384	273	6	196	68	0
22	235	0	59	314	15	397	26	334	0	125
23	193	191	279	0	254	7	0	268	322	358
24	36	119	186	0	359	0	204	308	84	152
25	120	19	147	380	0	356	162	335	300	8
26	57	146	371	246	220	111	400	198	142	336
27	0	0	155	266	90	183	87	323	172	239
28	189	307	0	338	387	347	70	127	100	14
29	210	92	86	0	229	252	56	286	0	3
30	157	367	103	131	18	99	10	114	0	49
31	128	234	241	317	135	390	177	302	283	97
32	73	328	255	375	281	348	361	71	0	0
33	346	309	61	205	369	0	342	320	301	278
34	362	365	260	113	382	395	195	267	13	134
35	243	132	136	293	388	226	141	124	398	37
36	62	298	258	364	176	24	35	77	232	341
37	105	47	282	337	40	353	44	51	168	0
38	32	352	332	257	245	158	227	130	345	170
39	33	343	0	58	340	12	21	16	23	29
40	306	219	154	379	0	117	199	140	169	213
41	296	324	211	39	386	230	310	0	53	225
42	250	274	218	96	385	34	212	256	178	83
43	0	318	25	242	182	187	0	313	78	224
44	264	276	381	249	28	349	0	171	89	0
45	166	261	75	206	316	360	221	325	107	0
	4	7	4	5	5	5	5	5	5	5

Horas disponíveis por sala

3.5 CARACTERIZAÇÃO DAS ESTRUTURAS DE VIZINHANÇA

Neste trabalho foram idealizadas e implementadas várias estruturas de vizinhança. Algumas dessas estruturas foram usadas na chamada Fase I que na estratégia VNS corresponde a transformar a proposta de solução inactível encontrada em uma proposta de solução factível. Após encontrar uma proposta de solução factível, sem violar as restrições tipo *RO1*, *RO2* e *RO3*, a estratégia VNS tenta melhorar a qualidade da solução diminuindo as restrições violadas tipo *RD1*, *RD2* e *RD3* e sempre preservando a factibilidade, isto é, sem violar as restrições *RO1*, *RO2* e *RO3*. Em todo o processo de otimização sempre geramos soluções vizinhas que não violam as restrições tipo *RO1* e *RO2*.

Um fato importante na formulação das estruturas de vizinhança na Fase I é que as restrições violadas tipo *RO3* podem ser facilmente identificadas quando usamos a proposta de representação de uma solução nº. 2 como a mostrada na tabela 4. Nesse arranjo matricial podemos identificar todas as restrições violadas tipo *RO3*, percorrendo cada hora do arranjo matricial. Adicionalmente podem ser separadamente identificados os eventos que se encontram em conflito e a hora e sala em que se encontram programados. Assim, em vetores simples são armazenados os eventos em conflito e a linha e coluna em que se encontram no arranjo matricial.

Na fase I foram usadas as seguintes estruturas de vizinhança:

- (1) Deslocar um evento em conflito para um horário vazio da mesma sala. A este tipo de vizinhança chamamos de *N01*.
- (2) Trocar um evento em conflito com outro evento alocado na mesma sala. A este tipo de vizinhança chamamos de *N02*.
- (3) Trocar um evento em conflito para outra sala adequada e trocando de posição com outro evento que pode ser alocado na sala do evento em conflito. Este tipo de vizinhança chamamos de *N03*.
- (4) Escolher o horário de um evento em conflito e outro horário do arranjo matricial. Todos os eventos alocados nesse horário são novamente alocados nesses dois horários seguindo a mesma estratégia usada na geração de uma solução inicial. A este tipo de vizinhança chamamos de *N04*.
- (5) Escolher o horário de um evento em conflito e dois outros horários do arranjo matricial. Todos os eventos alocados nesse horário são novamente alocados nesses três horários seguindo a mesma estratégia usada na geração de uma solução inicial. A este tipo de vizinhança chamamos de *N05*.

Na fase II foram usadas as seguintes estruturas de vizinhança:

- (1) Um evento $E1$ que se encontra na sala $S1$ é deslocado para uma sala $S2$ permitida que se encontra desocupada ou está ocupada por um evento $E2$ que pode ser deslocado para a sala $S1$. Deve-se observar que essa proposta evita violar restrições tipo $RO2$. Deve-se observar também que nesse tipo de vizinhança pode acontecer apenas deslocamento de um evento ou a troca de posição de dois eventos. A este tipo de vizinhança chamamos de $N1$.
- (2) Dois eventos, $E1$ que se encontra na sala $S1$ e $E2$ que se encontra na sala $S2$, são deslocados simultaneamente para posições vazias em salas permitidas ou trocam de posição com os eventos $E3$ que se encontra na sala $S3$ (troca com o evento $E1$) e $E4$ (troca com o evento $E2$) desde que seja possível deslocar os eventos $E3$ e $E4$ para as salas $S1$ e $S2$ respectivamente. Deve-se observar também que nesse tipo de vizinhança pode acontecer apenas deslocamento dos eventos $E1$ e $E2$ para posições vazias, ou pode acontecer um deslocamento e uma troca de eventos (por exemplo, $E1$ é deslocado para $S3$ e $E2$ e $E4$ trocam de posição) ou a troca de posições de quatro eventos. A este tipo de vizinhança chamamos de $N2$.
- (3) k eventos podem ser deslocados ou trocar de posição seguindo a mesma estratégia usada em $N1$ e $N2$. Deve-se enfatizar que no processo de deslocamento de um evento e de troca de posição entre eventos, deve-se eliminar a possibilidade de violar restrições tipo $RO2$. Em outras palavras, se um evento passa para outra posição, então essa nova posição deve representar uma sala adequada para alocar o evento. A este tipo de vizinhança chamamos de Nk .
- (4) Os eventos alocados em uma hora dada e que apresentam os menores valores de conflito $RD3$ são trocados com os elementos que se encontram programados na última hora do dia (as colunas 9, 18, 27, 36 e 45 no arranjo matricial). Esta proposta visa diminuir o número de restrições tipo $RD3$. A este tipo de vizinhança chamamos de $NE1$.
- (5) Duas colunas do arranjo matricial são trocados de posição. Em outras palavras, os eventos programados na hora i passam para a hora j e aqueles programados na hora j passam a ser alocados na hora i . Este tipo de vizinhança visa diminuir o número de restrições tipo $RD1$, $RD2$ e $RD3$. A este tipo de vizinhança chamamos de $NE2$.
- (6) Três colunas do arranjo matricial são trocados de posição. Em outras palavras, os eventos programados na hora i passam para a hora j , os programados na hora j passam para a hora k e os programados na hora k passam para a hora i . Este tipo de vizinhança visa diminuir o número de restrições tipo $RD1$, $RD2$ e $RD3$. A este tipo de vizinhança chamamos de $NE3$.

- (7) k colunas do arranjo matricial são trocadas de posição usando a mesma estratégia usada em $NE2$ e $NE3$. Este tipo de vizinhança visa também diminuir o número de restrições tipo $RD1$, $RD2$ e $RD3$. A este tipo de vizinhança chamamos de NEk

3.6 ESTRUTURA GERAL DO ALGORITMO VNS

O algoritmo VNS desenvolvido neste trabalho apresenta duas fases. Na Fase I, pretende-se encontrar uma solução que não viole as restrições obrigatórias e, portanto, nesse processo não se leva em conta as violações das restrições secundárias. Na Fase II, sem violar as restrições obrigatórias, pretende-se encontrar uma solução com o número mínimo de restrições secundárias violadas. As duas fases são mostradas adiante.

Fase I:

- (1) Montar uma solução inicial em construção alocando os eventos no arranjo matricial de forma que não sejam violadas as restrições obrigatórias $RO1$ e $RO2$ e, adicionalmente, minimizando as restrições obrigatórias $RO3$. Assim, os eventos são alocados nas salas adequadas menos ocupadas e na posição em que apresenta o menor número de conflitos tipo $RO3$. Em caso de empate (que acontece frequentemente) a posição é escolhida de forma aleatória entre as posições que empataram.
- (2) Implementar a Vizinhança NO_1 para cada evento que se encontra em conflito.
- (3) Implementar a Vizinhança NO_2 para cada evento que se encontra em conflito.
- (4) Implementar a Vizinhança NO_3 para cada evento que se encontra em conflito.
- (5) Implementar a Vizinhança NO_4 para cada evento que se encontra em conflito.
- (6) Implementar a Vizinhança NO_5 para cada evento que se encontra em conflito.
- (7) Caso a solução em construção encontre uma solução sem restrições violadas, então passar para a Fase II. Em caso contrário pare o processo porque não foi encontrada uma solução factível.

A Fase II é um algoritmo RVNS que assume a seguinte forma:

Fase II:

- (1) Escolher um evento de forma aleatória e implementar a Vizinhança N_1 . Se a solução vizinha for aceita, então atualizar a solução incumbente e repetir o passo 1.

- (2) Escolher dois eventos de forma aleatória e implementar a Vizinhança N_2 . Se a solução vizinha for aceita, então atualizar a solução incumbente e ir ao passo 1. Em caso contrário ir ao passo 3.
- (3) Escolher três eventos de forma aleatória e implementar a Vizinhança N_3 . Se a solução vizinha for aceita, então atualizar a solução incumbente e ir ao passo 1. Em caso contrário ir ao passo 4.
- (4) Implementar a vizinhança NE_1 e caso seja encontrada uma proposta de solução de melhor qualidade, então atualizar a solução incumbente e ir ao passo 1. Em caso contrário ir ao passo 5.
- (5) Escolher aleatoriamente duas colunas do arranjo matricial e implementar a vizinhança NE_2 e caso seja encontrada uma proposta de solução de melhor qualidade, então atualizar a solução incumbente e ir ao passo 1. Em caso contrário ir ao passo 6.
- (6) Escolher aleatoriamente três colunas do arranjo matricial e implementar a vizinhança NE_3 e caso seja encontrada uma proposta de solução de melhor qualidade, então atualizar a solução incumbente e ir ao passo 1. Em caso contrário ir ao passo 7.
- (7) Se o critério de parada for satisfeito, então pare o processo. Em caso contrário, voltar ao passo 1.

O critério de parada escolhido é o número máximo de avaliações da função objetivo, isto é, de propostas de solução avaliadas.

3.7 TESTES E RESULTADOS

Foram realizados testes com 4 instâncias disponíveis na competição internacional. Três dessas instâncias apresentam 400 eventos, 10 salas de aula disponíveis e 200 alunos. O quarto caso apresenta 350 eventos, 10 salas de aula disponíveis e 350 alunos.

Os dados são apresentados na Tabela 5.

Tabela 5: Dados dos Sistemas Testados - Instâncias

Instância	I	II	III	IV
Eventos	400	400	400	350
Salas	10	10	10	10
Características	10	10	10	5
Alunos	200	200	200	350

Na Tabela 6 são apresentados os melhores resultados encontrados para os quatro casos de testes.

Tabela 6: Melhores Resultados Encontrados

Instância	I	II	III	IV
Nº de restrições secundárias violadas	236	259	249	285
Restrição RD1	131	100	142	136
Restrição RD2	11	8	7	9
Restrição RD3	94	151	100	140

Como o algoritmo VNS desenvolvido apresenta componentes estocásticas, logo o algoritmo pode gerar soluções diferentes para cada número escolhido como semente para gerar a sequência de números pseudo aleatórios. Portanto, espera-se encontrar um conjunto de soluções quando escolhemos sementes diferentes para usar o algoritmo VNS implementado. Para mostrar esse fato, o algoritmo VNS foi processado 10 vezes e os resultados são mostrados na tabela 7 para o caso II do problema de programação de horários.

Tabela 7: Testes com a instância II

Teste	Número aleatório	RD1	RD2	RD3	Nº de restrições secundárias
1	2059	139	4	126	269
2	2079	103	8	163	279
3	2087	131	6	142	279
4	2361	107	5	170	282
5	2396	100	8	151	259
6	2455	128	6	155	289
7	2554	121	12	148	281
8	2600	122	16	150	288
9	2712	115	17	136	268
10	2722	112	7	167	286

Deve-se observar que todas as soluções são diferentes o que mostra a complexidade do tamanho do espaço de busca. Existem duas soluções com função objetivo do mesmo valor, mas essas soluções são diferentes. Entretanto, a qualidade das soluções encontradas são muito próximas, sendo a melhor delas a que apresenta função objetivo $v = 259$.

3.8 CONCLUSÕES PARCIAIS

Os resultados encontrados mostram um desempenho adequado do algoritmo VNS para encontrar propostas de grade horária adequadas para serem implementadas na prática já que encontra com muita frequência soluções válidas e que, portanto, não violam as restrições obrigatórias *RO1*, *RO2* e *RO3*. Entretanto, o número de restrições secundárias violadas é relativamente elevado. Assim, pode-se ainda melhorar as soluções encontradas implementando estruturas de vizinhança mais complexas.

4 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho foi desenvolvido um algoritmo tipo VNS para resolver o problema de grade horária em que a modelagem matemática e a implementação computacional seguem a estrutura de dados da competição internacional de grade horária.

Através de uma análise detalhada e após testes exaustivos foi verificado que esse tipo de problema se trata do tipo de problema altamente restrito, isto é, trata-se de um problema em que é relativamente difícil encontrar soluções factíveis. Assim, não é trivial encontrar soluções em que não se encontrem violadas as restrições tipo $RO1$, $RO2$ e $RO3$. Portanto, deve-se implementar uma estratégia especializada para encontrar uma solução inicial válida. Esse processo especializado foi montada na fase I que pode ser considerada uma estratégia heurística gulosa e complementada com um algoritmo VND.

Uma vez encontrada uma solução factível na Fase I, então na Fase II devemos diminuir o número de restrições violadas tipo $RD1$, $RD2$ e $RD3$. A ideia central desse processo é diminuir os conflitos tipo $RD1$, $RD2$ e $RD3$ sem perder a factibilidade. Nesse contexto, foram projetadas as vizinhanças tipo Nk e NEk . Nessa conjuntura, deve-se observar que quase todas as vizinhanças geradas, especialmente aquelas do tipo Nk são infactíveis. Em outras palavras, a maioria das propostas de solução identificadas são infactíveis, confirmando a observação anterior de que se trata de um problema altamente restrito.

As soluções encontradas podem ser consideradas adequadas. Entretanto, a proposta precisa de melhorias adicionais para que se torne competitiva com as melhores propostas encontradas na literatura especializada. Assim, como em trabalhos futuros podem ser implementadas as seguintes estratégias:

- Implementar computacionalmente vizinhanças tipo Nk e NEk mais complexas já que neste trabalho foram implementadas apenas vizinhanças até $N3$ e $NE3$.
- Evitar a escolha de vizinhos de forma totalmente aleatória. Na verdade pelo menos um dos eventos escolhidos na Fase II deveria ser um evento que viola alguma das restrições tipo $RD1$, $RD2$ e $RD3$ de forma que sejam escolhidas apenas soluções vizinhas com maior probabilidade de que sejam factíveis.
- Idealizar e implementar outras formas de vizinhança.

Referências:

- ABRAMSON, D.* Constructing school timetables using simulated annealing: sequential and parallel algorithms. *Management Science*, v. 37, p. 98–113, 1991.
- ABRAMSON, D.; ABELA, J.* A parallel genetic algorithm for solving the school timetabling problem. In: *Proceedings of the 15th Australian Computer Science Conference*. [S.l.: s.n.], 1992.
- ALVAREZ-VALDÉS, R.; CRESPO, E.; TAMARIT, J.M.* “Design and implementation of a course scheduling system using tabu search”, *European Journal of Operational Research*, Vol. 137, No. 3, pp. 512-523, 2002.
- ALVAREZ-VALDÉS, R.; MARTIN, G.; TAMARIT, M.* Constructing good solutions for the spanish school timetabling problem. *Journal of the Operational Research Society*, v. 47, p. 1203–1215, 1996.
- AL-YAKOOB, S.M.; SHERALI, H.D.* “A mixed integer programming approach to a class timetabling problem: A case study with gender policies and traffic considerations”, *European Journal of Operational Research*, Vol. 180, No. 3, pp. 1028-1044, 2007.
- AL-YAKOOB, S.M.; SHERALI, H.D.* “Mathematical programming models and algorithms for a class faculty assignment problem”, *European Journal of Operational Research*, Vol. 173, No. 2, pp. 488-507, 2006.
- AVELLA, P.; VASILÉV, I.* A computational study of a cutting plane algorithm for university course timetabling. *Journal of Scheduling*, v. 8, p. 497–514, 2005.
- BIRBAS, T.; DASKALAKI, S.; HOUSOS, E.* Timetabling for greek high schools. *Journal of the Operational Research Society*, v. 48, p. 1191–1200, 1997.
- BRAILSFORD, S.C.; POTTS, C.N.; SMITH, B.M.* “Constraint satisfaction problems: Algorithms and applications”, *European Journal of Operational Research*, Vol. 119, No. 3, pp. 557-581, 1999.

BURKE, E.; CARTER, M. “The practice and theory of automated timetabling II: Selected papers from the 2st International Conference on the Practice and Theory of Automated Timetabling”, University of Toronto, 1997, Springer, Lecture Notes in Computer Science, Vol. 1408, 1998.

BURKE, E.; ERBEN, W. “The practice and theory of automated timetabling III: Selected papers from the 3st International Conference on the Practice and Theory of Automated Timetabling”, University of Applied Sciences, Konstanz, 2000, Springer, Lecture Notes in Computer Science, Vol. 2079, 2000.

BURKE, E.; ROSS, P. “The practice and theory of automated timetabling: Selected papers from the 1st International Conference on the Practice and Theory of Automated Timetabling”, Napier University, 1995, Springer, Lecture Notes in Computer Science, Vol. 1153, 1996.

BURKE, E.K. et al. “Automated timetabling: The state of the art”, The Computer Journal, Vol. 40, No. 9, pp 565-571, 1997.

BURKE, E.K. et al. “A graph based hyperheuristic for educational timetabling problems”, European Journal of Operational Research, Vol. 176, No. 1, pp. 177-192, 2007.

BURKE, E.K. et al. “Hybrid variable neighborhood approaches to university exam timetabling”, European Journal of Operational Research, Vol. 206, No. 1, pp. 46-53, 2010.

BURKE, E.K.; PETROVIC, S. “Recent research directions in automated timetabling”, European Journal of Operational Research, Vol. 140, No. 2, pp. 266-280, 2002.

BURKE, E.K.; PETROVIC, S. “Timetabling and rostering”, European Journal of Operational Research, Vol. 153, No. 1, pp. 1-2, 2004.

CARTER, M.W.; LAPORTE, G. “Recent developments in practical examination timetabling”, In *CARTER, M.W. A comprehensive course timetabling and student scheduling system at the University of Waterloo. The practice and theory of automated timetabling III: Selected papers (PATAT 2000).* Lecture Notes in Computer Science 2079. Springer Verlag, pp 3-21, 2001.

CHU, P.; BEASLEY, J.E. “A genetic algorithm for the generalized assignment problem”, *Computers and Operations Research*, Vol. 24, (1), pp. 17-23, 1997.

COLONI, A.; DORIGO, M.; MANIEZZO, V. Metaheuristics for high-school timetabling. *Computational Optimization and Applications*, v. 9, n. 3, p. 277–298, 1998.

COOPER, T.B.; KINGSTON, J.H. “The complexity of timetabling construction problems”, In *CARTER, M.W. A comprehensive course timetabling and student scheduling system at the University of Waterloo. The practice and theory of automated timetabling III: Selected papers (PATAT 2000). Lecture Notes in Computer Science 2079. Springer Verlag*, pp 283-295, 2001.

COSTA, D. “Tabu search algorithm for computing an operational timetable”, *European Journal of Operational Research*, Vol. 76, No. 1, pp. 98-110, 1994.

CUNG, V. Strategies for the parallel implementation of metaheuristics. In: *Essays and Surveys in Metaheuristics*. [S.l.]: Kluwer Academic Publishers, 2002. p. 263–308.

DANTZIG, G. B.; WOLFE, P. Decomposition principle for linear programs. *Operations Research*, v. 8, n. 1, p. 101–111, 1960.

DASKALAKI, S. An integer programming formulation for a case study in university timetabling. *European Journal of Operational Research*, v. 153, p. 117–135, 2004.

DASKALAKI, S.; BIRBAS, T. “An integer programming formulation for a case study in university timetabling”, *European Journal of Operational Research*, Vol. 153, No. 1, pp. 117-135, 2004.

DASKALAKI, S.; BIRBAS, T. “Efficient solutions for a university timetabling problem through integer programming”, *European Journal of Operational Research*, Vol. 160, No. 1, pp. 106-120, 2005.

DE CAUSMAECKER, P.; DEMEESTER, P.; VANDEN BERGHE, G. “A decomposed metaheuristic approach for a real world university timetabling problem”, *European Journal of Operational Research*, Vol. 195, No. 1, pp. 307-318, 2009.

DIMOPOULOU, M.; MILLOTIS, P. “Implementations of a university course and examination timetabling system”, *European Journal of Operational Research*, Vol. 130, No. 1, pp. 202-213, 2001.

DREXL, A.; SALEWSKI, F. Distribution requirements and compactness constraints in school timetabling. *European Journal of Operational Research*, v. 102, n. 1, p. 193–214, out. 1997.

FERLAND, J.; LAVOUE, A. Exchange procedures for timetabling problems. *Discrete Applied Mathematics*, p. 237–253, 1992.

GANS, O. D. A computer timetabling system for secondary schools in the Netherlands. *European Journal of Operational Research*, v. 7, p. 175–182, 1981.

GENDREAU, M.; HERTZ, A.; LAPORTE, G. A tabu search heuristic for the vehicle routing problem. *Management Science*, v. 40, n. 10, p. 1276–1290, out. 1994.

GLOVER, F. Future paths for integer programming and artificial intelligence. *Computers & Operations Research*, v. 13, p. 533–549, 1986.

GLOVER, F.; LAGUNA, M. “Tabu search”, Kluwer Academic Publishers, 1997.

GLOVER, F.; KOCHENBERGER, G. “Handbook of metaheuristics”, Kluwer Academic Publishers, 2003.

HANSEN, P. The steepest ascent mildest descent heuristic for combinatorial programming. In: *Congress on Numerical Methods in Combinatorial Optimization*. Capri, Italy: [s.n.], 1986.

HANSEN, P.; MLADENOVIC, N. “Variable neighborhood search: Principles and applications”, *European Journal of Operations Research*, 130, pp. 449-467, 2001.

HANSEN, P.; MLADENOVIC, N. “Variable neighborhood search”, in F. Glover and G. Kochenberger Eds, *Handbook of metaheuristics*, pp. 145-184, 2003.

HERTZ, A. "Tabu search for large scale timetabling problems", *European Journal of Operational Research*, Vol. 54, No. 1, pp. 39-47, 1991.

HERTZ, A.; WERRA, D. Using tabu search techniques for graph coloring. *Computing*, v. 39, n. 4, p. 345–351, dez. 1987.

INTERNATIONAL TIMETABLING COMPETITION - Organized by Metaheuristics Network and sponsored by PATAT Idsia available www.idsia.ch/Files/ttcomp2002.

KAHAR, M.N.M.; KENDALL, G. "The examination timetabling problem at University Malaysia Pahang: Comparison of a constructive heuristic with an existing software solution", *European Journal of Operational Research*, Vol. 207, No. 2, pp. 557-565, 2010.

KANG, L.; WHITE, G.M. "A logical approach to the resolution of constraints in timetabling", *European Journal of Operational Research*, Vol. 61, No. 3, pp. 306-317, 1997.

KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by simulated annealing. *Science*, v. 220, p. 671–680, 1983.

LAWRIE, N. An integer linear programming model of a school timetabling problem. *Computer Journal*, v. 12, p. 307–316, 1969.

LU, Z.P.; HAO, J.K. "Adaptive tabu search for course timetabling", *European Journal of Operational Research*, Vol. 200, No. 1, pp. 235-244, 2010.

MEISELS, A.; KAPLANSKY, E. "Iterative restart technique for solving timetabling problems", *European Journal of Operational Research*, Vol. 153, No. 1, pp. 41-50, 2004.

NONOBE, K.; IBARAKI, T. "A tabu search approach to the constraints satisfaction problem as a general problem solver", *European Journal of Operational Research*, Vol. 106, No. 2-3, pp. 599-623, 1998.

PAPOUTSIS, K.; VALOUXIS, C.; HOUSOS, E. A column generation approach for the timetabling problem of greek high schools. *Journal of the Operational Research Society*, v. 54, p. 230–238, 2003.

QU, R.; BURKE, E.K.; MCCOLLUM, B. “Adaptive automated construction of hybrid heuristics for exam timetabling and graph coloring problems”, *European Journal of Operational Research*, Vol. 198, No. 2, pp. 392-404, 2009.

RESENDE, M. An introduction to GRASP. In: *SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL*, 39, 2007, Fortaleza. **Anais...** Fortaleza: SBPO, 2007. P. Disponível em: <<http://www.din.uem.br/sbpo/sbpo2007/pdf/arq0296.pdf>>. Acesso em: 10 dez. 2013.

RESENDE, M.; RIBEIRO, C. Greedy randomized adaptive search procedures. In: *KOCHENBERGER, G. (Ed.). Handbook of metaheuristics*. Nowell: Kluwer Academic Publishers, 2003. p. 219–249.

ROMERO, R.; RIDER, M.J.; SILVA, I.J. “A metaheuristic to solve the transmission expansion problem”, *IEEE Transactions on Power Systems*, Vol. 22, No. 4, pp. 2289-2291, 2007.

SANTOS, H. G. Formulações e Algoritmos para o Problema de Programação de Horários em Escolas. Tese (Doutorado) –Universidade Federal Fluminense. Niterói, Brasil, 2007.

SCHAERF, A. “A survey of automated timetabling”, *Artificial Intelligence Review*, Vol. 13, No. 2, pp. 87-127, 1999.

SCHAERF, A. Local search techniques for large high school timetabling problems. *IEEE Transactions on Systems, Man and Cybernetics Part A: systems and Humans*, v. 29, n. 4, p. 368–377, jul. 1999.

SOUZA, M. Programação de Horários em Escolas: Uma Aproximação por Metaheurísticas. Tese (Doutorado) — Programa de Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro - COPPE/UFRJ, Rio de Janeiro, Brasil, 2000.

SOUZA, M.; MACULAN, N. OCHI, L. Melhorando quadros de horários de escolas através de caminhos mínimos. *Tendências em Matemática Aplicada e Computacional*, v. 1, n. 1, p. 515–524, 2000.

SOUZA, M.; OCHI, L.; MACULAN, N. A GRASP-tabu search algorithm for solving school timetabling problems. In: *Metaheuristics: Computer Decision-Making*. [S.l.]: Kluwer Academic Publishers, 2003. p. 659–672.

STEFANO, C. D.; TETTAMANZI, A. An evolutionary algorithm for solving the school time-tabling problem. In: AL., E. B. et (Ed.). *Proceedings of the EvoWorkshop on Applications of Evolutionary Computing*. London, UK: Springer-Verlag, 2001. v. 2037, p. 452–462.

RIGHETTO, E. “Desenvolvimento de técnicas eficientes de programação linear na utilização de metaheurísticas para o problema de planejamento da expansão de sistemas de transmissão” Tese (doutorado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Ilha Solteira : [s.n.], 2010 118 f.

TORREÃO, J. R. A.; ROE, E. Microcanonical optimization applied to visual processing. *Physics Letters*, v. 205, p. 377–382, fev. 1995.

VANDEN BROEK, J.; HURKENS, C.; WOEGINGER, G. “Timetabling problems at the TU Eindhoven”, *European Journal of Operational Research*, Vol. 196, No. 3, pp. 877-885, 2009.

WERRA, D. “An introduction to timetabling”, *European Journal of Operational Research*, Vol. 19, No. 2, pp. 151-162, 1985.

WERRA, D. “The combinatorics of timetabling”, *European Journal of Operational Research*, Vol. 96, No. 3, pp. 504-513, 1997.

ZHANG, D. et al. “A simulated annealing with a new neighborhood structure based algorithm for high school timetabling problem”, *European Journal of Operational Research*, Vol. 203, No. 3, pp. 550-558, 2010.

Apêndice:

Tabela nº 8 – Instância Nº. 01 do ITC

Evento	Nº de salas	Denominação das salas							Nº de alunos	Denominação dos alunos															
1	1	3							9	86	142	143	144	145	146	147	148	199							
2	2	1	5						9	17	99	176	177	178	179	182	183	186							
3	2	5	9						11	98	142	144	146	148	150	152	154	157	164	188					
4	2	4	5						10	127	129	131	133	135	137	139	141	143	145						
5	2	5	7						11	98	130	131	132	133	134	135	136	137	138	142					
6	2	4	5						8	40	41	42	43	44	45	60	64								
7	1	3							8	119	120	121	122	123	124	125	133								
8	2	5	9						0																
9	2	3	6						10	62	63	64	65	66	67	68	69	70	71						
10	1	5							11	5	99	167	169	172	175	178	183	188	195	200					
11	1	6							11	6	7	8	9	10	11	12	13	15	17	99					
12	3	2	4	5					10	85	86	87	88	89	90	91	92	93	94						
13	1	5							10	184	185	186	187	188	189	190	193	195	200						
14	2	3	6						7	9	10	11	12	13	14	100									
15	2	2	5						6	2	7	169	170	193	197										
16	4	4	5	7	10				10	87	88	89	90	91	92	93	94	95	96						
17	6	2	3	5	6	9	10		10	24	26	28	30	32	39	42	45	48	51						
18	1	1							9	46	47	48	52	54	56	61	65	73							
19	3	4	5	7					9	99	100	101	102	103	104	105	109	142							
20	1	3							8	51	52	55	60	63	66	72	75								
21	2	4	5						3	62	69	70													
22	1	3							11	63	64	65	66	67	68	69	70	79	80	81					
23	2	3	6						8	2	4	8	10	18	35	45	200								
24	2	5	7						10	1	2	3	4	5	14	16	18	199	200						
25	3	5	7	8					4	120	121	122	134												
26	4	2	3	5	6				10	67	68	69	70	71	72	73	74	75	76						
27	2	3	6						10	35	36	37	38	43	45	47	49	51	53						
28	2	3	6						8	109	110	111	112	113	114	117	143								
29	1	5							10	19	143	145	147	149	151	153	155	160	165						
30	1	5							9	79	80	81	82	83	84	85	86	87							
31	2	4	5						9	3	4	5	6	7	8	9	13	24							
32	2	4	5						10	136	137	138	139	140	141	142	143	144	180						
33	1	3							9	166	167	168	169	170	171	172	173	185							
34	2	3	6						8	67	68	69	70	71	72	73	74								
35	2	3	6						10	1	2	3	194	195	196	197	198	199	200						
36	2	4	5						10	12	13	14	15	16	17	18	19	20	24						
37	1	1							7	174	175	177	179	181	183	186									
38	1	1							10	115	116	117	118	119	120	122	124	131	142						
39	1	5							10	117	119	121	123	125	127	129	131	133	143						
40	2	3	6						2	135	137														

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas						Nº de alunos	Denominação dos alunos											
41	2	5	10					10	124	125	126	130	132	134	142	145	148	180		
42	1	3						9	164	166	167	168	169	170	172	175	180			
43	1	1						10	146	147	148	149	150	151	152	153	154	187		
44	2	4	5					9	21	23	24	27	30	33	36	39	42			
45	1	3						10	166	167	168	169	170	171	172	173	175	181		
46	2	1	5					10	118	120	122	124	126	128	130	132	134	144		
47	3	4	5	7				6	97	102	103	104	109	142						
48	1	1						8	158	159	160	161	166	168	170	181				
49	2	3	6					10	80	82	86	89	92	95	98	101	120	143		
50	1	5						10	1	147	149	151	153	155	157	159	182	187		
51	4	2	5	8	9			9	129	131	133	135	141	144	147	150	153			
52	2	2	5					10	101	102	103	104	105	106	107	108	110	146		
53	1	10						6	139	144	146	148	150	182						
54	1	10						10	140	141	142	143	144	145	146	147	148	182		
55	1	3						9	25	27	29	31	37	40	43	46	49			
56	2	2	5					10	163	164	165	166	167	168	169	173	175	182		
57	1	1						10	36	37	38	39	40	41	48	50	52	54		
58	2	5	7					8	147	148	150	152	154	156	160	183				
59	2	5	10					7	11	14	190	191	192	193	194					
60	2	1	5					10	5	8	183	184	185	187	189	191	193	195		
61	1	3						10	51	52	53	54	55	56	57	58	59	60		
62	2	5	9					10	73	74	75	76	77	78	79	80	81	82		
63	3	2	3	5				10	103	104	105	106	107	108	109	125	132	143		
64	5	2	4	5	8	9		10	115	116	117	118	119	120	121	122	123	144		
65	1	5						10	7	11	25	179	180	181	185	187	189	191		
66	2	3	6					8	21	22	27	29	31	36	41	46				
67	1	10						10	2	169	170	171	172	173	175	178	183	196		
68	2	5	8					10	61	62	63	64	65	66	67	68	69	72		
69	2	5	7					8	49	52	55	58	67	77	91	144				
70	1	5						9	53	54	55	56	57	58	67	69	72			
71	2	4	5					8	53	54	58	61	64	67	70	76				
72	3	5	7	8				10	24	26	28	30	32	34	36	38	40	42		
73	1	5						10	166	167	168	169	170	171	172	173	176	183		
74	2	2	5					10	24	25	26	27	28	29	30	31	32	33		
75	2	5	7					10	84	85	86	87	88	89	90	91	92	93		
76	1	5						10	48	49	50	51	52	53	54	55	56	57		
77	2	2	5					10	104	105	106	107	108	109	110	111	112	183		
78	2	3	6					6	114	115	118	120	121	122						
79	2	2	5					9	105	106	107	109	112	115	118	121	124			
80	3	1	3	6				10	107	108	109	110	111	112	119	121	123	145		

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas					Nº de alunos	Denominação dos alunos														
81	3	4	5	7			10	3	4	5	9	11	13	15	17	19	21					
82	1	3					9	81	82	84	86	88	90	92	94	98						
83	2	5	7				10	78	79	80	81	82	83	84	85	86	87					
84	2	4	5				7	189	190	191	193	195	197	199								
85	1	1					9	56	57	59	62	65	68	71	74	77						
86	2	1	5				10	163	164	165	166	167	168	171	180	183	186					
87	4	2	5	8	9		9	33	34	35	36	38	41	44	47	50						
88	1	5					10	39	40	41	42	44	46	48	50	52	54					
89	1	1					10	128	130	132	134	136	138	140	142	144	147					
90	2	4	5				8	159	160	162	163	164	165	176	184							
91	3	5	7	9			6	4	5	6	7	8	11									
92	4	2	5	8	9		10	114	115	116	117	118	125	128	131	134	145					
93	1	5					10	75	76	77	78	79	80	81	82	83	84					
94	2	2	5				10	179	180	181	182	183	184	185	193	196	200					
95	3	2	5	9			8	47	49	51	52	53	54	55	56							
96	3	5	7	8			10	116	117	118	119	120	121	122	123	149	185					
97	1	3					9	34	186	187	188	189	190	191	192	193						
98	1	10					10	44	45	46	47	48	49	50	51	55	59					
99	4	2	3	5	6		10	35	170	171	172	173	174	175	176	177	180					
100	2	5	8				10	17	18	19	20	21	22	25	28	31	45					
101	2	5	9				10	1	2	3	4	5	6	7	198	199	200					
102	4	4	5	7	8		8	117	119	121	123	125	128	132	145							
103	3	5	7	8			9	108	109	110	111	112	113	114	115	118						
104	3	4	5	7			7	64	65	66	67	71	82	85								
105	1	5					10	4	80	81	82	83	88	95	98	101	145					
106	1	5					6	63	68	72	80	83	86									
107	2	3	6				10	55	56	57	58	59	60	61	62	66	72					
108	2	4	5				9	43	44	45	49	51	53	55	57	73						
109	2	5	9				10	6	168	169	170	171	172	173	174	175	176					
110	2	4	5				10	6	184	185	186	187	188	189	190	194	196					
111	2	5	8				10	28	29	30	31	32	33	34	35	36	37					
112	2	3	6				9	38	39	40	42	44	48	50	52	54						
113	1	10					10	8	148	149	150	151	152	153	154	155	156					
114	2	4	5				10	67	69	73	76	79	82	85	88	91	91					
115	2	2	5				10	6	131	132	133	134	135	137	139	148	151					
116	1	5					9	53	54	55	56	57	59	61	69	73						
117	3	2	3	5			3	78	96	99												
118	2	5	8				4	103	104	16	110											
119	4	5	7	8	10		8	7	141	142	143	144	145	150	152							
120	1	3					10	63	64	65	69	71	73	75	77	79	81					

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos											
121	2	5 8	6	106	107	118	121	134	145						
122	1	5	5	24	26	28	30	32							
123	1	3	5	21	23	24	25	26							
124	2	4 5	10	7	127	128	129	130	136	138	146	149	152		
125	2	5 9	10	7	166	167	168	169	170	171	172	173	176		
126	2	5 8	10	8	157	158	159	160	161	162	163	164	177		
127	3	2 3 5	10	31	32	33	34	35	36	37	38	39	40		
128	2	5 7	10	2	4	8	21	26	192	194	196	198	200		
129	3	4 5 9	10	9	122	123	124	125	126	127	128	129	132		
130	2	2 5	10	85	86	87	91	93	95	97	113	118	146		
131	4	2 3 5 6	10	88	89	90	91	92	93	94	95	96	97		
132	1	1	9	10	11	12	13	14	15	16	17	18			
133	2	2 5	10	55	56	57	58	59	60	61	62	69	73		
134	1	5	10	27	28	29	30	31	32	33	34	35	55		
135	2	5 9	10	68	69	70	71	80	83	86	89	92	95		
136	1	5	10	7	8	9	10	15	23	26	29	32	46		
137	2	2 5	9	8	174	175	176	177	178	180	181	183			
138	1	5	10	8	88	89	90	92	94	96	110	114	119		
139	1	1	5	126	127	128	129	146							
140	3	5 7 8	10	8	130	131	132	133	134	135	136	137	138		
141	2	2 5	10	9	127	128	129	130	131	132	133	148	151		
142	4	4 5 7 10	10	88	89	90	91	92	93	94	95	97	100		
143	1	5	8	2	14	25	180	184	187	190	193				
144	1	5	8	43	114	116	119	122	131	135	139				
145	3	2 5 9	10	9	110	111	112	113	123	126	129	132	135		
146	2	4 5	6	19	20	28	30	32	47						
147	2	5 8	9	9	160	161	162	163	164	165	166	168			
148	1	3	10	9	102	103	104	105	106	107	108	109	110		
149	2	4 5	10	9	159	160	161	162	163	164	165	174	177		
150	1	5	8	20	21	22	23	24	25	26	29				
151	1	3	10	89	90	91	92	93	94	95	96	97	98		
152	2	5 9	10	9	185	186	187	188	189	190	191	194	196		
153	2	5 7	10	10	133	134	135	136	137	138	139	140	141		
154	2	5 9	10	89	90	91	92	93	94	95	96	97	98		
155	4	4 5 7 9	6	15	195	196	197	198	199						
156	2	2 5	10	10	132	133	134	135	136	137	139	142	145		
157	3	4 5 7	10	10	177	178	179	180	181	182	183	184	185		
158	1	1	10	28	29	30	31	32	33	34	35	36	37		
159	1	3	3	20	158	166									
160	2	5 7	8	44	115	117	120	123	132	136	140				

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos															
161	2	5	7									10	1	2	3	4	5	6	47	198	199	200						
162	3	2	3	5								2	66	77														
163	1	3										10	20	188	189	190	191	192	193	194	195	197						
164	1	10										8	10	153	154	155	156	157	160	163								
165	1	5										10	59	60	61	62	63	64	65	66	70	74						
166	1	5										10	10	113	114	115	116	117	118	120	122	124						
167	2	5	10									10	35	36	37	38	39	40	41	42	43	44						
168	2	4	5									10	10	155	156	157	158	159	160	161	162	163						
169	2	2	5									2	48	152														
170	1	5										10	11	139	140	141	142	143	144	146	149	152						
171	2	3	6									10	83	84	85	86	87	88	89	90	91	94						
172	3	4	5	7								3	175	180	181													
173	1	10										10	33	34	35	36	37	38	39	40	41	45						
174	2	2	5									10	11	119	120	121	122	124	127	130	133	137						
175	1	3										10	1	2	3	4	5	11	197	198	199	200						
176	3	4	5	8								9	11	102	103	104	106	107	108	109	110							
177	2	5	9									8	11	154	155	156	157	158	161	164								
178	1	3										8	58	60	62	70	73	76	82	85								
179	1	10										10	11	152	153	154	155	156	157	158	161	184						
180	3	5	7	8								10	12	135	136	137	138	139	140	143	146	150						
181	1	1										10	12	124	125	126	127	128	129	130	131	142						
182	2	5	7									7	12	164	165	169	171	173	176									
183	2	4	5									10	13	152	153	154	155	156	157	158	161	176						
184	2	2	5									10	18	19	20	25	28	31	34	37	40	43						
185	4	2	5	8	9							8	14	146	149	151	153	155	159	177								
186	2	5	9									9	46	47	48	50	52	54	56	60	74							
187	10	1	2	3	4	5	6	7	8	9	10	10	16	17	18	19	20	21	22	23	24	25						
188	1	10										10	25	27	29	31	33	35	37	39	41	43						
189	1	5										4	12	150	151	152												
190	2	3	6									10	22	140	141	142	143	145	147	149	151	153						
191	1	1										8	74	75	76	77	79	81	84	95								
192	2	4	5									9	21	148	150	152	154	156	158	179	198							
193	1	5										10	13	187	188	189	190	191	192	193	194	197						
194	4	4	5	7	8							8	12	100	102	110	113	116	119	125								
195	3	4	5	7								9	12	102	103	104	105	110	112	114	116							
196	2	5	8									9	52	53	54	55	56	57	58	59	75							
197	2	5	9									10	12	86	87	88	89	90	91	92	107	110						
198	2	2	5									10	2	5	8	12	185	186	190	193	196	199						
199	2	2	5									9	80	83	85	87	89	91	93	97	99							
200	2	3	6									9	19	162	163	164	165	167	169	171	177							

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos															
201	1	1	3	37	39	40													
202	4	1 3 6 10	7	27	98	99	100	101	103	111									
203	1	5	6	21	154	161	164	167	184										
204	4	2 5 8 9	5	173	180	182	184	186											
205	3	2 5 9	6	168	169	178	180	182	185										
206	2	4 5	8	22	50	53	56	59	68	85	92								
207	2	5 7	10	87	88	89	90	91	92	93	94	95	96						
208	2	5 8	10	61	62	63	64	65	66	67	68	70	76						
209	1	10	9	34	35	40	43	46	64	69	74	79							
210	2	4 5	10	71	72	74	77	80	83	86	89	92	95						
211	2	5 8	9	13	165	166	167	168	169	170	171	177							
212	1	10	10	64	65	66	67	68	69	70	71	72	77						
213	3	2 5 9	7	13	125	126	127	128	129	130									
214	3	4 5 7	4	20	186	187	188												
215	2	2 5	10	25	26	27	32	35	38	41	52	56	60						
216	5	1 2 3 5 6	10	83	84	85	86	87	88	89	90	92	96						
217	2	3 6	6	14	125	128	131	134	138										
218	2	5 8	10	28	29	30	33	36	39	42	53	57	61						
219	1	3	10	14	166	167	168	169	170	171	172	182	184						
220	2	5 8	10	5	9	22	28	182	183	184	186	188	190						
221	2	5 9	10	48	153	154	155	156	157	158	159	178	181						
222	2	2 5	10	29	156	157	158	159	160	161	162	163	177						
223	3	4 5 8	9	26	138	139	140	141	146	148	150	154							
224	1	10	10	3	6	9	22	187	188	191	194	197	200						
225	3	2 4 5	7	3	14	184	187	190	194	198									
226	1	5	9	23	70	71	72	73	74	75	76	101							
227	3	4 5 7	9	15	122	123	124	126	129	132	135	139							
228	2	4 5	10	14	126	127	128	129	130	131	140	143	146						
229	3	4 5 10	10	49	95	96	97	98	99	100	101	102	111						
230	1	5	10	16	17	18	19	20	26	28	30	32	34						
231	4	2 5 8 9	8	1	3	5	9	33	38	43	49								
232	2	2 5	9	24	78	79	80	81	82	83	84	111							
233	3	4 5 8	10	14	105	106	107	108	109	110	111	112	121						
234	3	5 7 8	9	29	30	31	32	33	34	35	36	37							
235	1	10	10	16	186	187	188	189	190	191	192	195	197						
236	4	2 5 8 9	9	43	44	45	47	48	49	50	51	52							
237	3	2 3 5	10	15	115	116	118	120	122	124	126	129	135						
238	4	5 7 8 10	9	155	156	157	158	159	160	161	162	178							
239	2	2 5	10	22	134	135	136	137	138	139	140	141	142						
240	1	10	10	15	115	116	117	118	119	120	121	122	125						

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas						Nº de alunos	Denominação dos alunos													
241	2	2	5					10	24	25	26	27	28	29	30	31	32	33				
242	2	2	5					10	7	9	11	13	15	22	25	28	31	46				
243	2	3	6					10	1	2	3	4	6	15	197	198	199	200				
244	2	4	5					10	20	110	111	112	113	114	115	116	117	119				
245	2	2	5					9	89	90	91	92	93	96	99	102	146					
246	2	5	9					7	4	16	185	188	191	195	199							
247	3	4	5	8				10	13	14	15	16	17	18	19	20	21	23				
248	1	1						9	1	158	159	162	164	179	184	189	196					
249	2	5	7					10	15	191	192	193	194	195	196	197	198	199				
250	2	3	6					10	72	73	76	78	81	84	87	90	93	96				
251	1	3						9	2	160	161	163	165	180	185	192	197					
252	3	2	3	5				5	24	25	26	27	28									
253	2	4	5					8	3	168	170	173	181	186	193	198						
254	3	4	5	8				10	25	93	94	95	96	97	98	99	100	112				
255	1	10						9	75	76	77	78	79	80	81	82	83					
256	2	1	10					10	36	37	41	43	45	47	49	51	53	55				
257	1	3						10	1	3	6	10	23	30	193	195	197	199				
258	2	3	6					10	16	133	134	135	136	137	138	139	140	141				
259	2	4	5					10	16	143	144	145	146	147	148	149	150	151				
260	1	5						10	17	190	191	192	193	194	195	196	197	198				
261	1	3						10	17	106	107	108	109	111	113	115	117	126				
262	2	5	9					10	71	72	73	74	75	76	77	78	80	82				
263	2	4	5					9	47	124	125	126	128	129	133	137	141					
264	3	4	5	8				8	17	158	159	160	161	162	163	178						
265	3	5	7	10				9	23	164	165	166	167	168	176	180	199					
266	2	5	7					10	23	24	31	34	37	40	43	54	58	83				
267	2	2	5					9	17	97	98	99	100	101	102	103	107					
268	2	3	6					9	1	10	15	27	196	197	198	199	200					
269	1	5						8	67	68	69	71	74	80	83	86						
270	4	2	3	5	6			9	11	16	28	182	183	185	188	191	194					
271	1	3						9	35	188	189	190	191	192	193	194	195					
272	2	4	5					10	37	132	133	134	135	136	137	138	139	140				
273	3	5	7	8				10	1	2	3	4	5	18	197	198	199	200				
274	2	1	10					10	66	67	68	70	72	74	76	78	80	82				
275	2	5	10					7	38	176	179	181	183	185	187							
276	2	3	6					10	26	87	88	89	91	93	95	97	99	101				
277	2	5	9					10	8	10	12	14	16	23	26	29	32	48				
278	2	5	8					3	16	22	200											
279	2	5	9					10	1	2	3	4	5	6	7	8	9	18				
280	1	10						8	26	170	172	174	175	176	177	178						

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos										
281	2	4	5									10	23	123	124	125	126	127	128	129	130	135	
282	2	2	5									10	36	37	38	39	40	41	42	43	44	50	
283	1	3										10	1	4	7	10	24	184	189	192	195	198	
284	2	3	6									10	52	54	56	58	60	62	65	70	75	100	
285	2	5	9									10	10	11	12	13	14	15	16	17	18	21	
286	2	3	6									8	18	165	166	167	168	171	174	178			
287	2	5	8									9	62	63	64	65	66	67	68	70	77		
288	2	5	8									10	72	73	74	75	76	77	78	79	80	81	
289	2	5	8									10	15	16	17	26	29	32	35	38	41	44	
290	3	2	3	5								10	49	50	51	52	53	54	55	56	57	58	
291	3	4	5	8								10	22	101	102	103	104	105	106	107	108	120	
292	2	3	6									10	18	189	190	191	192	193	194	195	196	197	
293	2	5	9									4	31	102	110	112							
294	3	2	3	5								10	19	98	99	100	104	106	108	111	115	120	
295	1	1										10	19	127	128	129	131	133	141	144	147	151	
296	1	1										9	12	13	14	16	24	27	30	33	51		
297	1	3										9	47	48	49	50	52	60	62	70	78		
298	3	5	7	9								8	19	36	37	41	44	66	71	76			
299	1	10										10	21	177	178	179	180	181	182	183	184	185	
300	2	3	6									9	20	189	190	191	192	193	194	195	196		
301	1	10										10	30	99	100	101	102	103	104	105	107	111	
302	3	4	5	8								7	5	20	186	189	192	196	200				
303	2	5	8									10	21	126	127	128	129	130	131	132	134	147	
304	3	4	5	8								4	22	127	130	138							
305	2	5	9									8	22	153	155	156	157	158	159	162			
306	2	5	9									10	13	14	15	16	17	18	19	20	21	27	
307	2	4	5									3	22	152	155								
308	1	3										10	23	159	160	161	162	163	164	165	174	186	
309	2	2	5									10	6	7	8	10	12	14	16	18	20	23	
310	2	3	6									8	74	76	77	78	79	80	81	82			
311	2	2	5									4	23	73	87	98							
312	2	3	6									10	49	50	51	52	53	54	55	56	57	58	
313	2	5	9									10	25	148	149	150	151	152	153	154	155	156	
314	3	4	5	8								10	39	144	145	146	147	148	149	150	151	152	
315	2	5	9									10	28	29	30	31	32	33	34	35	36	38	
316	2	5	9									8	61	62	63	64	65	66	71	78			
317	2	5	9									10	88	89	90	91	92	93	94	95	96	97	
318	3	1	3	6								10	23	126	127	128	129	130	131	132	133	153	
319	2	3	6									10	26	101	102	103	105	107	109	112	116	121	
320	2	5	9									10	34	35	36	37	38	39	40	41	42	43	

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas					Nº de alunos	Denominação dos alunos														
321	1	10					9	11	12	13	14	15	17	34	44	50						
322	4	2	3	5	6		10	27	151	152	153	157	159	161	163	165	178					
323	1	5					10	7	84	181	182	186	188	190	192	194	196					
324	2	5	9				9	1	2	3	4	5	6	7	8	200						
325	2	3	6				10	50	149	150	151	152	153	154	155	156	157					
326	4	1	2	3	5		1	42														
327	1	10					7	34	162	165	166	167	174	178								
328	2	3	6				9	121	122	123	124	125	138	141	144	147						
329	2	2	5				7	34	159	162	165	168	174	189								
330	1	10					10	38	157	158	159	160	161	162	178	181	184					
331	3	2	5	9			10	39	101	102	103	104	105	106	107	108	120					
332	1	1					9	52	53	54	55	56	57	58	59	60						
333	2	2	5				10	39	154	155	156	158	160	162	164	174	179					
334	1	10					10	40	101	103	108	111	114	117	120	123	178					
335	2	3	6				8	123	124	125	126	127	128	129	139							
336	1	10					10	40	105	106	107	108	109	110	111	112	113					
337	3	2	5	9			8	38	39	42	45	51	67	72	77							
338	2	2	5				10	41	84	85	87	90	93	96	99	118	139					
339	3	2	3	5			4	192	194	196	198											
340	2	3	6				5	25	27	29	31	33										
341	2	5	10				10	68	70	75	78	81	84	87	90	93	96					
342	2	5	7				10	42	43	44	45	46	47	49	51	53	56					
343	2	2	5				10	44	112	113	114	115	116	117	118	119	155					
344	2	4	5				9	5	7	8	9	10	11	12	13	41						
345	4	4	5	7	10		8	140	141	142	144	145	147	150	153							
346	3	2	5	9			8	59	60	61	62	63	64	65	79							
347	1	5					9	85	86	90	92	94	96	98	100	112						
348	2	5	7				9	98	181	182	183	184	185	186	187	188						
349	2	5	7				8	12	45	179	181	186	189	192	195							
350	2	5	9				10	19	20	21	22	23	24	25	26	27	41					
351	1	10					10	79	100	101	102	103	104	105	106	108	112					
352	2	2	5				10	42	100	101	102	103	104	105	106	108	113					
353	2	5	9				9	42	160	161	162	163	174	177	181	200						
354	1	3					9	42	130	131	132	133	134	135	136	137						
355	2	5	9				9	49	50	51	53	55	57	59	63	81						
356	2	4	5				10	42	104	105	106	107	108	109	110	111	113					
357	2	4	5				9	81	83	88	91	94	97	100	119	140						
358	3	5	9	10			10	45	110	111	112	113	114	121	123	126	140					
359	3	5	7	8			9	73	74	75	76	77	78	79	80	81						
360	2	4	5				10	44	96	97	98	99	101	103	105	108	113					

Tabela nº 8 – Instância nº 01 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas						Nº de alunos	Denominação dos alunos									
361	2	5	9					7	44	113	115	116	117	118	140			
362	1	1						9	45	136	137	138	139	140	149	151	179	
363	1	1						8	45	99	100	101	102	103	104	114		
364	1	1						3	1	6	57							
365	1	1						9	43	44	47	49	51	53	55	57	59	
366	3	5	7	10				10	82	172	173	174	175	176	177	179	182	185
367	1	10						10	58	161	162	163	164	165	166	167	179	182
368	2	4	5					8	59	61	72	75	78	81	84	97		
369	1	10						9	45	46	48	50	52	54	56	58	60	
370	1	10						7	191	192	194	196	197	198	199			
371	2	2	5					9	130	132	134	136	142	145	148	151	154	
372	2	4	5					7	46	113	127	130	133	136	141			
373	1	10						10	46	137	138	139	140	143	146	149	152	156
374	1	5						10	1	2	3	4	5	6	7	8	82	200
375	1	5						10	46	113	114	115	116	117	118	119	120	143
376	1	10						9	60	61	62	63	64	65	66	71	83	
377	1	10						8	59	136	137	138	145	147	150	179		
378	2	5	10					9	93	94	95	100	102	104	106	109	114	
379	1	10						10	86	87	88	89	90	91	92	93	94	97
380	2	5	10					10	46	121	122	123	124	125	126	127	131	141
381	1	10						10	60	165	166	167	168	169	170	172	175	179
382	2	2	5					10	4	47	166	171	174	177	182	187	194	199
383	1	1						7	63	64	65	66	68	71	83			
384	1	1						10	23	47	142	143	144	145	147	149	151	153
385	1	1						9	112	113	114	115	116	117	118	119	179	
386	1	5						9	48	154	155	156	158	159	160	162	180	
387	1	1						10	74	75	77	79	82	85	88	91	94	97
388	2	2	5					10	48	136	137	138	139	140	141	156	163	186
389	1	1						10	62	63	64	65	66	67	68	69	71	83
390	1	1						8	84	85	86	87	94	97	100	103		
391	1	5						9	84	115	116	117	118	119	120	121	122	
392	1	1						9	17	18	19	20	21	24	27	30	42	
393	1	1						9	53	55	57	59	61	63	68	73	78	
394	2	2	5					9	56	57	58	59	60	61	62	63	84	
395	1	1						9	169	170	171	172	173	174	175	176	180	
396	1	1						10	84	169	170	171	172	173	174	175	176	187
397	3	1	2	5				7	51	54	57	60	69	88	114			
398	1	1						9	21	23	24	25	27	29	31	33	85	
399	2	2	5					10	43	44	45	46	47	48	49	50	51	85
400	1	5						9	85	94	95	96	97	98	99	107	109	

Tabela nº 9 – Instância Nº. 02 do ITC

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos														
1	3	3	7	8								10	3	142	144	146	148	150	152	160	164	184					
2	1	8										12	34	35	36	39	41	43	45	47	61	64	99	171			
3	1	10										12	91	92	93	94	95	96	97	98	99	100	102	172			
4	2	2	5									2	75	76													
5	8	1	4	5	6	7	8	9	10			11	29	30	31	32	33	34	35	36	37	39	101				
6	2	4	7									11	29	30	31	32	33	34	35	36	37	40	108				
7	2	5	6									10	2	10	14	18	22	109	194	195	198	200					
8	2	5	10									7	28	29	30	31	32	41	110								
9	1	1										11	144	146	148	150	152	154	156	158	162	164	172				
10	3	3	7	8								10	95	96	97	98	99	100	102	104	105	173					
11	1	6										12	88	89	90	91	92	93	94	95	96	97	102	173			
12	1	6										12	41	42	43	44	45	46	47	48	53	63	107	174			
13	2	5	9									10	63	64	65	66	67	68	69	71	73	110					
14	4	4	6	8	10							12	15	17	19	21	23	25	27	29	31	41	105	177			
15	1	5										11	110	111	112	113	116	119	122	125	128	134	148				
16	2	4	8									12	108	109	110	111	112	113	114	115	116	117	118	178			
17	1	10										12	125	126	127	128	129	130	131	132	133	144	148	178			
18	1	5										8	108	112	113	114	115	116	117	118							
19	1	2										2	81	82													
20	1	2										2	156	157													
21	4	3	5	7	9							10	16	18	20	22	24	26	28	30	32	106					
22	2	1	5									7	5	6	7	100	198	199	200								
23	1	9										11	150	151	152	153	154	155	157	159	161	163	165				
24	2	5	6									6	108	167	168	169	170	171									
25	1	3										9	101	102	103	105	106	107	108	109	110						
26	3	3	7	8								10	35	36	37	38	40	42	44	48	53	100					
27	9	1	3	4	5	6	7	8	9	10		10	18	19	20	21	22	23	32	34	42	100					
28	1	5										11	43	44	45	46	47	48	49	50	53	63	100				
29	2	5	10									7	56	62	65	68	72	76	100								
30	2	4	7									11	64	65	66	67	68	69	70	71	72	73	101				
31	1	4										9	132	134	136	138	140	152	156	160	164						
32	1	1										9	43	44	45	46	47	49	53	70	106						
33	5	1	5	7	8	9						11	101	165	166	167	168	169	170	177	179	181	183				
34	2	5	6									11	32	33	34	35	36	37	38	39	40	71	107				
35	3	3	7	8								7	21	23	25	27	29	33	63								
36	1	1										10	87	88	89	90	91	92	93	94	95	96					
37	4	1	5	9	10							7	44	46	48	49	50	64	103								
38	1	9										11	79	80	82	84	86	88	90	92	94	105	111				
39	1	4										11	143	144	145	146	147	148	149	150	151	152	154				
40	1	5										11	41	42	43	44	46	48	50	52	54	74	102				

Tabela nº 9 – Instância nº 02 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas					Nº de alunos	Denominação dos alunos												
81	1	1					11	105	106	107	108	109	110	111	112	113	114	115		
82	2	1	8				10	95	96	97	98	99	100	102	103	104	109			
83	4	1	5	9	10		4	137	139	141	143									
84	1	9					11	7	8	9	10	12	19	22	32	36	66	103		
85	3	3	7	8			10	55	56	57	59	60	61	63	65	67	112			
86	1	3					6	140	142	143	144	145	146							
87	3	5	7	9			11	1	2	3	7	103	195	196	197	198	199	200		
88	1	9					7	94	95	96	97	98	100	103						
89	1	9					11	1	2	3	4	8	107	196	197	198	199	200		
90	1	5					11	11	12	13	14	15	16	17	18	19	25	109		
91	1	3					8	49	51	53	54	55	56	57	58					
92	1	2					2	1	200											
93	3	3	7	8			10	19	20	21	22	23	24	25	26	27	28			
94	1	4					4	4	5	6	7									
95	2	3	7				10	167	168	169	171	172	173	174	175	176	177			
96	1	4					3	126	127	167										
97	4	4	5	6	10		11	2	9	108	185	186	188	190	192	194	196	198		
98	1	1					10	41	43	45	47	49	56	59	62	65	68			
99	2	5	6				11	149	150	151	152	153	154	155	156	157	159	162		
100	2	5	9				11	88	89	90	91	92	93	94	95	96	98	104		
101	1	3					10	17	18	22	24	26	28	30	32	34	67			
102	4	4	5	6	10		11	32	33	34	35	38	39	40	41	42	67	104		
103	1	3					10	126	127	128	129	130	131	132	133	134	135			
104	2	3	7				10	117	118	119	120	122	123	124	143	146	150			
105	3	1	2	5			2	166	167											
106	1	4					11	104	171	172	173	174	175	176	178	180	182	194		
107	2	5	9				11	8	9	10	11	12	13	14	15	16	17	124		
108	2	1	8				11	49	50	51	52	53	54	62	64	66	68	114		
109	2	3	9				5	73	74	76	77	105								
110	1	4					11	87	88	89	90	91	92	93	94	103	105	108		
111	2	5	6				11	104	105	106	107	108	109	110	111	112	113	114		
112	4	4	5	6	10		10	115	116	117	119	121	123	125	127	129	134			
113	1	4					11	38	39	42	44	46	48	51	54	57	70	105		
114	1	8					11	106	178	179	180	181	182	183	184	185	186	187		
115	1	9					8	140	141	142	143	144	148	149	150					
116	1	8					11	73	74	75	76	77	78	79	80	81	82	149		
117	2	5	6				3	125	127	151										
118	2	5	6				10	164	165	166	167	168	169	170	171	172	174			
119	2	2	5				2	27	28											
120	1	1					11	158	159	160	161	162	163	164	165	166	167	168		

Tabela nº 9 – Instância nº 02 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas					Nº de alunos	Denominação dos alunos												
121	1	2					2	192	193											
122	1	4					10	3	11	15	19	26	111	186	187	190	192			
123	2	3	7				7	54	55	59	60	61	62	106						
124	1	8					10	66	67	68	74	76	78	80	82	84	110			
125	4	1	5	9	10		3	172	173	196										
126	1	5					11	45	46	47	48	49	50	51	52	54	68	107		
127	1	8					7	58	59	61	70	72	74	112						
128	1	1					9	146	147	148	149	154	158	162	166	170				
129	2	4	5				5	48	50	54	68	107								
130	2	1	8				11	76	78	80	82	84	86	88	90	92	94	117		
131	2	3	4				10	60	61	63	65	67	75	79	83	91	111			
132	2	2	9				2	136	149											
133	3	3	4	8			8	145	147	149	151	153	155	161	166					
134	4	1	5	9	10		11	131	132	133	134	135	136	137	138	139	140	149		
135	1	5					11	3	10	109	184	187	189	191	193	195	197	199		
136	1	3					10	2	11	190	192	194	195	196	197	198	199			
137	1	9					11	2	4	68	150	188	190	192	194	196	198	200		
138	1	2					2	75	76											
139	1	4					10	43	44	45	46	47	48	49	50	69	111			
140	3	1	5	8			11	111	186	187	188	189	190	191	192	193	194	195		
141	3	3	4	7			7	24	26	27	28	29	90	112						
142	1	9					11	125	126	127	128	129	130	131	132	133	135	150		
143	1	9					11	107	111	114	117	120	123	127	131	135	139	150		
144	4	1	5	9	10		4	101	102	103	104									
145	3	1	5	6			11	164	165	166	167	168	169	170	171	172	173	174		
146	1	2					2	169	170											
147	1	3					8	123	125	131	133	135	137	139	141					
148	3	3	4	7			3	14	16	120										
149	2	5	6				7	24	25	31	33	35	77	116						
150	1	3					6	41	42	43	44	69	77							
151	1	3					7	13	25	28	31	34	37	69						
152	4	4	5	6	10		9	115	179	180	181	182	184	190	192	195				
153	4	4	5	6	10		11	5	6	7	8	9	10	11	12	13	17	115		
154	1	3					10	135	136	139	141	143	145	153	156	159	162			
155	5	3	4	5	7	9	9	79	81	82	83	84	85	86	87	92				
156	1	5					11	74	75	78	79	80	81	83	85	87	92	116		
157	1	1					10	36	37	38	39	40	41	42	43	57	69			
158	2	5	6				5	146	147	148	149	163								
159	2	3	7				10	4	7	11	182	184	186	188	190	192	194			
160	1	3					9	56	57	58	59	60	61	62	63	78				

Tabela nº 9 – Instância nº 02 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas						Nº de alunos	Denominação dos alunos												
161	1	1						11	24	25	26	27	28	29	30	33	38	70	123		
162	3	3	4	7				8	126	127	132	136	138	140	142	151					
163	1	5						8	117	118	119	120	121	122	123	124					
164	1	5						11	156	157	158	159	160	161	162	163	164	167	168		
165	1	2						2	93	94											
166	2	1	8					11	123	141	142	146	149	152	155	158	170	174	178		
167	4	4	5	7	9			11	18	116	167	168	169	170	171	172	173	174	175		
168	1	9						11	134	135	136	139	142	145	148	151	154	157	160		
169	1	9						5	60	63	66	70	119								
170	4	1	5	9	10			11	102	103	104	105	106	124	128	132	136	141	151		
171	1	8						11	104	106	108	114	117	120	123	126	129	135	151		
172	1	9						11	146	147	148	149	150	151	154	157	160	163	165		
173	2	1	8					11	175	176	177	178	179	180	181	182	183	184	185		
174	2	2	5					2	58	59											
175	1	4						11	97	98	99	100	101	102	103	104	105	106	117		
176	1	5						3	134	135	136										
177	3	3	7	8				10	173	174	175	176	177	178	179	180	181	182			
178	1	9						4	153	154	165	168									
179	1	5						11	45	46	47	48	49	50	57	59	61	71	124		
180	1	4						8	34	35	40	49	52	55	58	70					
181	1	3						10	113	114	115	116	117	118	119	120	121	122			
182	1	5						9	58	59	60	61	62	64	65	66	70				
183	3	1	5	6				11	119	171	172	173	174	175	176	177	178	179	180		
184	2	1	5					11	137	138	139	140	141	142	143	145	147	149	151		
185	1	5						10	39	40	41	42	45	53	55	57	71	119			
186	1	3						10	40	41	42	43	44	45	47	49	65	71			
187	3	1	5	6				9	142	143	144	145	151	155	159	163	175				
188	2	4	7					11	31	32	34	35	36	37	38	39	40	85	120		
189	3	1	2	5				2	165	166											
190	2	5	10					10	128	129	130	131	132	133	134	135	136	137			
191	1	2						2	47	48											
192	2	5	6					0													
193	1	1						11	89	90	91	93	95	97	99	101	103	105	120		
194	2	5	10					11	14	15	16	17	18	19	20	21	22	26	121		
195	4	3	5	7	9			10	84	85	86	87	88	89	90	91	92	93			
196	1	8						3	13	14	164										
197	1	1						10	48	49	50	51	52	54	56	58	72	120			
198	2	3	8					10	84	85	86	87	88	89	90	91	92	93			
199	2	7	8					11	40	41	43	45	47	49	52	55	67	71	120		
200	4	4	5	6	10			11	68	69	70	71	72	73	74	75	76	77	121		

Tabela nº 9 – Instância nº 02 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas						Nº de alunos	Denominação dos alunos														
201	4	1	5	6	8			11	68	70	72	77	80	83	86	89	92	96	121				
202	1	5						6	1	2	71	152	199	200									
203	1	3						10	179	180	181	182	183	184	185	186	187	188					
204	1	7						11	124	160	161	162	163	164	165	168	171	175	191				
205	1	4						11	33	34	36	38	40	42	44	50	59	72	121				
206	1	4						11	38	39	40	45	47	49	51	53	59	75	121				
207	1	2						2	58	59													
208	1	9						11	12	121	189	190	191	192	193	194	195	196	197				
209	1	1						11	65	66	73	78	81	84	87	90	93	97	122				
210	3	1	5	9				10	29	122	174	175	176	177	178	179	181	185					
211	1	3						10	105	107	109	115	118	121	124	127	130	136					
212	3	5	7	9				11	145	146	147	148	149	156	158	160	162	164	166				
213	2	5	10					6	20	26	27	28	29	121									
214	1	7						11	22	23	24	25	26	27	28	29	30	31	125				
215	1	4						11	85	86	87	88	89	90	91	92	93	94	152				
216	2	5	9					11	48	49	50	51	52	53	54	55	60	72	122				
217	2	5	6					11	126	127	128	129	130	133	136	139	142	145	152				
218	4	3	4	7	8			8	36	37	41	50	53	56	60	72							
219	2	2	5					2	30	33													
220	4	3	5	7	9			8	51	52	53	54	55	56	57	58							
221	2	7	8					11	21	22	23	24	25	26	27	28	29	72	122				
222	1	2						2	37	38													
223	1	7						11	107	108	109	110	111	112	113	114	115	116	123				
224	2	5	10					11	93	122	175	176	177	178	179	186	188	190	192				
225	2	5	10					10	156	157	158	159	161	162	163	164	166	169					
226	2	5	6					11	137	138	140	142	144	152	155	158	161	164	166				
227	3	5	7	9				11	27	122	176	177	178	179	180	181	182	183	186				
228	2	2	9					2	76	77													
229	1	5						11	52	53	54	55	56	57	58	59	60	78	122				
230	4	1	5	6	8			8	27	28	29	30	31	32	39	123							
231	2	5	10					11	81	82	83	84	86	88	90	92	94	96	123				
232	1	9						11	28	125	144	145	147	150	153	156	159	172	176				
233	2	5	10					11	12	13	14	15	16	17	25	27	29	73	123				
234	2	2	5					0															
235	2	1	5					10	55	56	57	60	61	62	64	66	73	123					
236	1	3						10	14	15	16	17	18	20	23	34	39	73					
237	1	5						10	1	4	28	123	185	187	189	191	193	195					
238	2	4	5					11	29	124	185	186	187	188	189	190	191	192	193				
239	2	5	10					11	95	96	97	98	99	100	101	102	104	106	124				
240	1	8						11	33	34	35	36	37	50	53	56	68	73	124				

Tabela nº 9 – Instância nº 02 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas					Nº de alunos	Denominação dos alunos														
281	1	5					8	167	183	184	185	186	187	188	189							
282	1	3					10	113	114	118	120	122	124	126	128	130	138					
283	2	5	9				10	59	62	64	66	73	77	81	85	89	153					
284	3	1	5	8			5	57	60	63	74	156										
285	1	2					2	153	154													
286	1	8					11	65	66	67	68	69	70	71	72	73	79	154				
287	1	2					2	50	52													
288	3	1	5	8			8	3	7	76	178	180	182	190	199							
289	2	5	6				11	13	14	15	16	17	18	19	20	21	77	154				
290	4	1	5	9	10		11	112	113	114	115	116	117	118	119	120	121	167				
291	4	1	5	9	10		9	76	136	137	138	139	140	141	161	186						
292	1	9					4	137	138	139	141											
293	1	4					11	53	54	55	56	57	58	59	60	62	81	168				
294	1	9					10	61	62	63	64	65	66	67	69	77	155					
295	2	5	10				11	109	112	115	118	121	125	129	133	137	147	155				
296	1	1					11	7	8	9	10	11	12	13	14	23	77	155				
297	1	8					11	157	158	159	160	161	162	163	164	165	166	167				
298	2	2	5				2	36	37													
299	1	8					7	42	58	61	64	67	77	161								
300	1	9					11	79	157	188	189	190	191	192	193	194	195	196				
301	1	9					11	3	4	5	6	7	8	9	10	11	80	168				
302	1	8					3	84	95	155												
303	1	8					4	172	174	176	178											
304	2	1	5				11	108	109	110	111	112	114	116	118	120	122	155				
305	2	5	6				11	23	24	25	26	27	28	29	30	31	78	155				
306	2	1	5				11	82	83	84	85	86	87	88	89	90	94	163				
307	1	2					2	129	131													
308	1	8					11	118	119	120	125	132	135	138	141	144	147	156				
309	1	2					2	85	86													
310	3	3	4	7			7	32	33	35	37	43	65	78								
311	1	1					11	31	32	33	40	42	44	46	51	63	82	169				
312	1	5					11	23	24	25	26	27	28	29	30	31	78	156				
313	1	8					11	4	5	6	11	13	21	31	35	40	86	156				
314	2	3	4				10	8	79	174	175	176	177	178	179	180	181					
315	2	1	8				11	37	38	39	40	41	47	49	51	61	80	161				
316	1	5					11	78	81	83	85	87	89	91	93	95	106	156				
317	1	4					11	27	28	29	30	31	32	33	34	35	80	157				
318	1	2					2	191	193													
319	1	9					11	87	88	92	94	96	98	100	102	104	106	158				
320	2	1	8				11	50	51	52	53	55	58	61	64	67	95	157				

Tabela nº 9 – Instância nº 02 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas								Nº de alunos	Denominação dos alunos															
361	2	7	8							11	125	126	127	128	129	130	131	132	133	153	170					
362	2	1	5							10	1	5	9	13	17	21	163	196	197	199						
363	1	1								9	30	31	32	33	34	35	36	96	170							
364	1	4								11	51	52	53	54	55	56	57	58	62	96	163					
365	2	2	5							1	6															
366	3	2	3	9						2	1	200														
367	2	2	5							2	142	143														
368	1	3								10	6	11	16	98	175	176	182	185	188	191						
369	1	5								11	146	147	148	149	150	151	152	153	154	155	163					
370	1	3								7	99	101	102	104	105	106	107									
371	1	3								9	70	72	73	75	77	79	81	83	96							
372	1	3								10	77	79	81	83	85	87	89	91	93	97						
373	1	3								3	3	6	197													
374	4	1	5	6	8					11	1	2	3	4	5	6	7	9	107	163	200					
375	1	4								11	96	97	98	99	100	101	102	103	104	108	163					
376	1	4								9	97	154	155	156	157	158	159	162	187							
377	3	1	4	8						11	42	43	44	45	46	48	50	52	63	97	175					
378	1	3								5	79	80	83	97	99											
379	1	4								11	1	3	7	12	17	99	164	197	198	199	200					
380	2	3	4							10	48	49	50	51	52	53	54	55	74	97						
381	3	3	4	8						8	23	26	29	32	35	38	97	165								
382	1	5								10	10	11	12	14	16	18	20	22	24	98						
383	1	5								11	2	4	8	13	18	100	166	193	194	195	196					
384	1	5								11	1	2	3	4	5	6	7	8	9	98	166					
385	1	5								6	9	12	13	14	15	16										
386	1	5								9	64	67	68	69	70	71	72	100	173							
387	1	5								9	184	185	186	187	188	189	190	191	193							
388	1	5								4	98	180	182	188												
389	1	5								9	69	70	71	72	74	78	82	90	107							
390	2	5	6							11	15	16	17	18	19	20	21	22	25	101	171					
391	2	5	6							9	8	9	13	15	17	19	21	23	25							
392	2	5	6							9	35	36	37	38	39	46	52	98	169							
393	2	5	6							10	99	179	180	181	182	183	184	185	186	187						
394	2	5	6							11	85	86	87	88	89	90	91	92	93	99	171					
395	3	1	5	6						4	168	173	175	177												
396	9	1	3	4	5	6	7	8	9	10	1	2	109	194	195	196	197	198	199	200						
397	2	5	6							10	99	143	145	147	149	151	153	163	183	198						
398	8	1	4	5	6	7	8	9	10	11	9	14	30	101	170	177	178	183	186	189	192					
399	4	3	5	7	9					7	95	96	97	98	99	100	176									
400	1	7								11	4	8	99	171	192	193	194	195	196	197	200					

Tabela nº 10 – Instância Nº. 03 do ITC

Evento	Nº de salas	Denominação das salas								Nº de alunos	Denominação dos alunos																
1	1	8								9	23	24	25	26	28	30	32	34	134								
2	8	1	3	4	5	6	7	8	10	11	122	124	126	128	130	132	134	136	138	169	198						
3	2	3	7							11	119	120	121	122	123	124	125	127	129	131	198						
4	1	5								7	157	158	159	160	161	164	165										
5	1	2								5	62	78	82	127	199												
6	1	8								11	28	29	30	31	32	38	40	42	44	91	199						
7	8	1	3	4	5	6	7	8	10	11	168	169	170	171	172	173	174	175	176	177	199						
8	4	4	5	7	8					11	15	16	17	18	19	20	21	22	23	24	200						
9	1	8								9	43	45	47	49	51	53	55	89	200								
10	10	1	2	3	4	5	6	7	8	9	10	79	80	81	82	83	84	85	86	87	90						
11	10	1	2	3	4	5	6	7	8	9	10	162	163	164	165	166	167	168	169	170	173						
12	1	5								10	150	151	152	153	154	155	156	157	158	159							
13	3	1	4	5						11	155	156	157	158	159	160	161	162	163	164	200						
14	1	7								11	1	53	54	55	56	58	61	76	81	86	92						
15	1	5								11	1	135	136	138	141	144	147	150	153	156	159						
16	1	8								10	1	4	7	16	187	188	189	192	195	198							
17	1	5								11	1	2	4	193	194	195	196	197	198	199	200						
18	1	9								10	120	121	122	124	128	132	136	140	145	150							
19	2	2	9							9	1	99	100	101	102	104	112	118	127								
20	1	4								10	74	75	76	77	78	79	80	81	82	90							
21	1	4								11	2	88	89	90	91	92	93	94	95	96	98						
22	1	5								10	2	28	29	30	31	32	33	34	35	36							
23	2	3	7							11	2	164	165	166	167	168	169	176	178	180	182						
24	8	1	3	4	5	6	7	8	10	11	3	38	39	40	42	44	46	48	50	52	90						
25	3	1	4	5						10	150	151	152	153	154	156	159	162	165	170							
26	3	3	7	9						10	10	11	12	13	14	15	28	31	34	90							
27	1	2								6	4	22	23	24	25	26											
28	1	7								11	4	10	11	12	13	14	15	16	17	18	20						
29	2	4	5							9	5	96	97	100	102	108	112	120	128								
30	8	1	3	4	5	6	7	8	10	11	5	48	49	50	51	52	53	54	55	72	91						
31	1	9								10	66	67	68	69	70	71	72	73	82	91							
32	10	1	2	3	4	5	6	7	8	9	10	146	147	148	149	150	151	152	153	154	156						
33	1	2								10	145	147	149	151	153	155	157	166	169	172							
34	1	7								10	27	28	29	30	35	37	39	41	43	91							
35	1	9								5	32	33	34	39	91												
36	3	3	7	9						10	6	8	10	12	14	16	18	20	22	24							
37	1	2								5	41	49	51	53	73												
38	1	5								10	158	159	160	161	162	163	164	167	170	173							
39	10	1	2	3	4	5	6	7	8	9	10	119	121	124	127	135	139	143	147	151							
40	1	8								10	1	3	16	187	188	191	193	195	197	199							

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos											
41	3	3 7 9	10	133	134	135	136	137	138	139	140	141	142		
42	1	7	8	20	21	22	23	24	25	40	93				
43	1	4	10	7	10	13	21	195	196	197	198	199	200		
44	1	7	3	49	53	93									
45	10	1 2 3 4 5 6 7 8 9 10	10	114	116	118	125	129	133	137	141	146	160		
46	10	1 2 3 4 5 6 7 8 9 10	10	132	134	139	142	145	148	151	154	157	160		
47	1	5	10	161	162	163	164	165	166	167	168	169	170		
48	1	8	2	55	56										
49	10	1 2 3 4 5 6 7 8 9 10	10	61	62	63	64	65	66	67	68	73	99		
50	10	1 2 3 4 5 6 7 8 9 10	5	2	6	21	183	189							
51	1	7	9	3	7	22	174	175	184	187	190	199			
52	3	1 4 5	10	18	19	20	21	22	29	31	33	40	135		
53	1	9	9	92	93	95	97	99	101	104	118	137			
54	10	1 2 3 4 5 6 7 8 9 10	8	126	127	128	129	130	131	150	162				
55	1	7	10	122	123	124	125	126	127	128	129	130	131		
56	1	5	10	1	3	10	13	16	19	22	195	196	197		
57	1	5	3	154	156	157									
58	1	4	10	53	54	55	56	58	60	63	66	119	128		
59	1	4	9	2	5	8	23	183	190	193	196	199			
60	1	7	10	45	46	47	48	49	61	64	81	120	129		
61	1	7	10	2	4	11	14	17	20	23	198	199	200		
62	10	1 2 3 4 5 6 7 8 9 10	10	33	34	35	36	37	38	39	47	73	94		
63	3	3 7 9	10	168	169	170	171	172	173	174	175	176	177		
64	1	5	7	44	46	48	50	52	54	94					
65	2	4 5	10	43	44	45	46	49	52	55	58	77	94		
66	1	5	8	90	92	96	98	105	110	118	136				
67	1	5	8	3	4	5	6	7	8	9	130				
68	1	2	10	62	63	64	65	66	72	74	77	80	95		
69	3	1 4 5	10	77	78	79	80	81	82	83	84	85	131		
70	9	1 2 3 4 5 6 -7 8 9 10	3	8	23	196									
71	1	9	9	4	8	23	193	195	196	197	198	200			
72	3	3 7 9	10	171	172	173	174	175	176	177	178	179	180		
73	1	7	10	116	118	120	122	124	126	128	130	140	144		
74	1	7	10	91	92	93	97	99	102	105	114	118	163		
75	1	8	10	106	107	108	109	110	111	112	113	118	131		
76	3	3 7 9	10	147	148	149	150	151	152	153	154	155	166		
77	1	4	8	129	130	131	139	141	143	145	147				
78	10	1 2 3 4 5 6 7 8 9 10	10	113	114	115	116	117	118	120	129	132	135		
79	1	9	10	98	99	100	101	102	103	105	111	119	132		
80	4	4 5 7 8	10	49	50	51	52	53	54	55	58	75	97		

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos											
81	10	1 2 3 4 5 6 7 8 9 10	8	3	6	9	24	186	188	190	192				
82	10	1 2 3 4 5 6 7 8 9 10	10	46	48	50	52	59	72	77	82	87	95		
83	10	1 2 3 4 5 6 7 8 9 10	9	18	19	20	21	22	23	24	25	26			
84	3	1 4 5	10	84	85	88	90	92	94	96	98	121	132		
85	1	4	9	125	126	127	128	129	130	131	132	133			
86	10	1 2 3 4 5 6 7 8 9 10	10	130	131	132	133	134	136	140	144	148	152		
87	5	1 4 5 7 8	10	112	113	119	126	130	134	138	142	147	161		
88	3	3 7 9	10	109	110	111	112	113	114	115	116	119	132		
89	1	7	10	17	19	21	23	25	27	29	31	33	97		
90	1	8	10	14	15	16	17	18	19	20	21	22	24		
91	3	3 7 9	9	99	100	101	102	104	106	111	119	141			
92	3	1 4 5	10	82	83	84	85	86	87	88	90	92	96		
93	1	5	9	116	122	125	128	137	141	145	149	157			
94	1	4	10	17	18	19	20	21	37	41	45	50	133		
95	10	1 2 3 4 5 6 7 8 9 10	10	59	60	61	62	63	64	65	67	74	96		
96	10	1 2 3 4 5 6 7 8 9 10	10	100	101	102	103	104	105	106	112	122	133		
97	10	1 2 3 4 5 6 7 8 9 10	10	1	2	3	4	5	196	197	198	199	200		
98	1	7	9	108	109	110	111	112	113	114	119	134			
99	10	1 2 3 4 5 6 7 8 9 10	4	113	115	117	118								
100	1	9	10	74	75	76	77	78	79	80	81	82	97		
101	1	4	10	123	124	125	126	127	128	129	130	131	143		
102	10	1 2 3 4 5 6 7 8 9 10	10	148	149	150	151	152	153	154	155	156	158		
103	1	8	7	59	63	64	71	73	75	148					
104	1	8	5	153	154	163	168	171							
105	1	8	5	64	79	83	119	138							
106	3	1 4 5	4	48	59	76	134								
107	1	2	10	123	125	127	129	131	133	135	137	139	171		
108	3	3 7 9	7	60	61	63	80	84	120	139					
109	1	8	10	106	107	108	109	110	111	114	117	120	144		
110	1	9	3	35	36	159									
111	3	3 7 9	4	54	55	75	164								
112	1	5	10	37	39	41	47	50	53	56	59	78	135		
113	10	1 2 3 4 5 6 7 8 9 10	6	164	165	166	168	169	181						
114	3	1 4 5	9	121	122	129	132	135	139	143	148	160			
115	1	8	10	2	3	4	5	6	7	8	9	41	135		
116	10	1 2 3 4 5 6 7 8 9 10	10	105	106	107	108	109	110	111	113	120	135		
117	3	1 4 5	8	65	67	69	70	72	74	76	149				
118	1	9	10	12	13	14	15	16	17	18	19	20	136		
119	10	1 2 3 4 5 6 7 8 9 10	5	137	138	139	140	141							
120	1	7	10	25	26	27	28	34	38	42	46	51	136		

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos											
121	1	7	10	171	172	173	174	175	176	177	178	179	183		
122	3	1 4 5	9	67	68	69	70	72	84	88	93	136			
123	1	8	10	170	171	172	173	174	175	176	177	178	179		
124	1	2	10	76	77	78	79	80	81	89	91	93	136		
125	1	9	10	102	103	104	105	106	107	110	113	149	161		
126	1	4	6	95	96	97	98	109	136						
127	3	1 4 5	10	137	138	139	140	141	142	143	144	145	161		
128	1	5	10	22	23	24	25	26	27	28	36	41	137		
129	1	9	9	57	59	61	63	65	67	70	75	137			
130	10	1 2 3 4 5 6 7 8 9 10	10	56	58	60	62	64	66	68	71	76	138		
131	3	1 4 5	10	160	161	162	164	166	168	170	172	174	176		
132	3	3 7 9	10	50	51	52	53	54	55	56	57	75	137		
133	1	5	10	75	76	77	78	79	80	82	84	86	162		
134	1	7	6	63	66	73	89	94	137						
135	1	9	10	153	154	155	156	157	158	159	160	161	162		
136	10	1 2 3 4 5 6 7 8 9 10	10	77	78	79	80	81	82	83	84	85	139		
137	1	9	9	141	142	143	144	146	147	148	149	151			
138	3	1 4 5	10	99	100	101	102	103	104	105	113	120	147		
139	1	2	10	174	175	176	177	178	179	180	181	182	183		
140	1	2	10	52	53	54	55	56	62	64	67	76	162		
141	1	9	10	8	9	10	11	12	13	14	15	16	25		
142	1	4	8	140	141	142	143	144	150	152	161				
143	10	1 2 3 4 5 6 7 8 9 10	10	138	139	140	141	142	143	144	145	146	147		
144	1	9	10	142	143	144	145	146	147	148	149	150	162		
145	10	1 2 3 4 5 6 7 8 9 10	8	90	91	92	94	98	114	120	140				
146	1	2	10	170	171	172	173	174	175	176	177	178	179		
147	1	4	10	175	176	177	178	183	185	187	189	191	193		
148	4	4 5 7 8	4	12	13	17	28								
149	1	4	1	142											
150	1	2	10	100	102	104	106	108	110	118	121	124	163		
151	1	4	10	5	6	7	8	9	12	15	18	21	25		
152	3	3 7 9	10	182	183	184	185	186	187	188	189	190	191		
153	1	2	8	16	18	20	22	26	29	32	162				
154	10	1 2 3 4 5 6 7 8 9 10	9	48	49	50	51	52	53	54	55	81			
155	1	9	6	90	91	93	94	98	167						
156	1	4	10	109	110	111	112	113	114	115	116	121	143		
157	1	8	10	42	43	44	45	46	47	48	49	50	144		
158	1	4	10	33	34	35	36	37	39	41	43	45	163		
159	10	1 2 3 4 5 6 7 8 9 10	9	11	16	26	176	177	182	184	186	188			
160	1	7	10	151	152	153	154	155	156	157	158	159	163		

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos													
161	2	2 9	10	71	72	73	74	75	77	79	82	123	144				
162	1	9	10	147	148	149	150	151	152	166	169	172	175				
163	10	1 2 3 4 5 6 7 8 9 10	8	127	130	133	136	140	144	150	164						
164	10	1 2 3 4 5 6 7 8 9 10	10	62	63	64	65	66	67	69	71	94	145				
165	10	1 2 3 4 5 6 7 8 9 10	6	187	188	189	190	191	192								
166	3	1 4 5	4	133	136	140	146										
167	1	8	10	7	9	11	13	15	17	19	21	23	27				
168	1	9	10	10	12	16	19	22	25	34	38	42	165				
169	10	1 2 3 4 5 6 7 8 9 10	10	56	57	58	59	60	61	62	63	82	164				
170	3	1 4 5	10	1	2	3	4	5	6	8	11	14	27				
171	3	1 4 5	10	133	137	140	143	146	149	152	155	158	165				
172	3	1 4 5	10	11	12	13	14	15	16	17	18	19	27				
173	1	4	10	40	41	42	43	44	45	46	56	78	165				
174	1	5	10	3	4	5	6	7	8	9	10	11	165				
175	1	4	10	1	2	3	4	5	6	7	166	199	200				
176	1	8	10	84	85	86	87	88	89	90	115	121	166				
177	1	5	9	147	148	149	150	151	152	153	154	184					
178	10	1 2 3 4 5 6 7 8 9 10	10	74	75	76	77	78	79	80	81	82	166				
179	1	4	4	150	151	152	166										
180	3	1 4 5	9	45	46	47	48	49	50	51	52	53					
181	1	9	9	60	61	62	63	64	65	66	68	167					
182	3	1 4 5	4	55	56	82	171										
183	1	4	10	104	105	106	107	108	109	110	114	121	167				
184	10	1 2 3 4 5 6 7 8 9 10	7	47	48	50	52	54	83	172							
185	1	5	10	99	100	101	102	103	104	106	143	148	167				
186	1	9	10	141	142	143	144	149	157	160	163	166	172				
187	1	7	8	107	108	109	110	111	112	113	114						
188	1	9	10	67	68	69	70	76	78	80	83	124	167				
189	1	4	7	79	80	83	85	92	95	182							
190	1	9	10	29	30	31	32	33	34	35	36	45	168				
191	3	1 4 5	10	3	6	9	174	184	186	191	194	197	200				
192	1	8	10	10	11	12	13	14	15	16	17	19	168				
193	1	8	10	37	38	39	40	41	42	43	44	54	169				
194	1	5	8	16	17	18	19	20	21	28	177						
195	1	4	10	66	67	68	69	70	71	72	73	82	174				
196	1	4	10	5	6	7	8	9	10	11	12	13	183				
197	2	2 9	10	122	123	124	125	126	127	128	130	133	168				
198	1	4	8	115	116	118	119	137	141	145	151						
199	1	5	8	88	89	90	91	93	94	96	168						
200	1	2	7	16	17	18	19	20	21	169							

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos											
201	3	3 7 9	8	135	137	138	140	142	144	146	172				
202	1	9	10	175	176	177	178	179	180	181	182	183	184		
203	10	1 2 3 4 5 6 7 8 9 10	10	70	71	72	73	74	81	83	85	87	170		
204	1	9	9	158	159	163	165	167	171	173	175	178			
205	1	8	6	114	115	116	117	121	122						
206	1	7	10	57	58	59	60	61	62	63	64	83	169		
207	1	8	10	86	87	89	91	93	95	97	99	125	169		
208	1	4	10	3	4	5	6	7	8	9	10	12	185		
209	3	3 7 9	8	178	179	180	181	183	184	185	186				
210	3	3 7 9	6	96	100	108	111	114	152						
211	5	1 4 5 7 8	10	46	47	48	49	50	51	52	53	54	169		
212	10	1 2 3 4 5 6 7 8 9 10	10	1	2	3	4	5	6	7	8	178	200		
213	1	7	10	101	103	105	107	109	111	119	122	125	170		
214	3	3 7 9	9	26	27	28	29	30	31	32	33	34			
215	1	5	8	96	97	98	103	105	114	121	179				
216	4	4 5 7 8	10	24	25	26	27	28	29	30	31	41	170		
217	10	1 2 3 4 5 6 7 8 9 10	10	29	30	31	32	33	34	35	37	43	171		
218	3	3 7 9	10	154	155	156	157	158	159	160	162	165	171		
219	1	8	8	57	59	61	63	65	68	83	180				
220	1	4	10	118	120	123	126	129	138	142	146	150	172		
221	1	5	10	105	106	107	108	109	110	111	114	121	171		
222	1	5	9	13	17	20	23	26	35	39	43	180			
223	1	4	9	44	46	48	50	52	54	56	58	172			
224	1	9	10	65	67	69	71	73	75	77	79	83	172		
225	2	2 5	9	96	98	100	102	104	112	118	122	173			
226	10	1 2 3 4 5 6 7 8 9 10	9	163	164	165	166	168	169	170	171	173			
227	3	3 7 9	10	137	138	139	140	141	142	155	158	161	175		
228	3	3 7 9	10	146	148	150	152	154	156	165	168	171	175		
229	10	1 2 3 4 5 6 7 8 9 10	10	96	97	98	99	100	102	106	115	122	173		
230	10	1 2 3 4 5 6 7 8 9 10	10	32	33	34	35	36	37	38	39	43	173		
231	10	1 2 3 4 5 6 7 8 9 10	3	70	72	173									
232	10	1 2 3 4 5 6 7 8 9 10	10	57	58	59	60	61	63	65	68	83	173		
233	1	2	10	115	117	123	127	131	135	139	144	149	179		
234	3	3 7 9	7	25	26	27	37	39	51	183					
235	1	9	8	10	11	12	13	14	15	29	174				
236	1	9	10	122	123	124	125	126	127	128	129	130	174		
237	1	2	8	145	146	147	151	153	163	166	175				
238	3	1 4 5	10	66	67	68	69	70	71	72	73	83	174		
239	1	2	10	63	64	65	66	67	68	69	70	84	174		
240	1	4	10	26	27	28	32	34	36	38	40	43	175		

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos													
241	1	8	7	104	106	108	110	112	115	176							
242	2	2 9	10	29	30	31	33	35	37	39	41	44	176				
243	10	1 2 3 4 5 6 7 8 9 10	10	88	89	90	91	92	93	94	95	96	181				
244	3	1 4 5	10	125	126	127	129	131	133	135	137	139	180				
245	1	8	9	1	4	7	12	17	192	194	196	198					
246	1	8	10	163	164	165	166	167	168	169	170	171	176				
247	1	2	10	1	186	187	188	189	191	193	195	197	199				
248	3	3 7 9	10	100	101	102	103	104	105	106	115	141	177				
249	1	2	9	35	36	37	38	39	40	41	42	44					
250	3	3 7 9	10	47	49	51	57	60	73	78	83	88	177				
251	10	1 2 3 4 5 6 7 8 9 10	4	147	148	149	151										
252	1	5	10	98	99	100	101	102	103	112	115	140	178				
253	10	1 2 3 4 5 6 7 8 9 10	9	108	109	110	111	112	119	121	131	134					
254	1	4	8	90	91	92	93	94	95	97	98						
255	5	1 4 5 7 8	9	2	5	8	13	20	189	190	191	199					
256	3	1 4 5	10	49	50	51	53	55	57	59	69	84	182				
257	1	2	10	7	8	9	10	11	12	19	37	44	178				
258	1	8	10	2	183	184	185	190	192	194	196	198	200				
259	3	3 7 9	10	69	70	71	72	73	74	75	76	84	182				
260	1	2	9	99	100	102	106	108	110	120	123	179					
261	10	1 2 3 4 5 6 7 8 9 10	10	123	124	128	131	134	138	142	146	153	185				
262	1	4	10	74	75	76	77	78	79	80	81	84	179				
263	10	1 2 3 4 5 6 7 8 9 10	9	106	107	108	109	110	111	113	116	167					
264	10	1 2 3 4 5 6 7 8 9 10	5	75	76	86	90	96									
265	3	3 7 9	9	113	114	115	116	117	118	119	121	180					
266	10	1 2 3 4 5 6 7 8 9 10	9	38	40	42	48	51	57	64	85	179					
267	1	8	5	187	188	189	190	192									
268	1	8	6	101	104	107	137	141	180								
269	1	9	9	1	7	13	180	184	187	189	191	193					
270	3	1 4 5	10	6	7	8	9	10	11	12	13	14	181				
271	1	8	10	152	153	154	155	156	157	158	159	160	180				
272	10	1 2 3 4 5 6 7 8 9 10	9	94	96	98	100	103	112	116	122	194					
273	1	2	8	155	156	159	161	163	167	169	186						
274	1	9	10	63	64	66	68	70	74	79	84	89	181				
275	1	9	9	170	171	172	173	175	176	177	178	181					
276	3	3 7 9	8	2	193	194	195	196	197	198	200						
277	4	4 5 7 8	6	101	109	112	147	154	186								
278	10	1 2 3 4 5 6 7 8 9 10	10	22	23	24	29	35	39	43	47	52	181				
279	10	1 2 3 4 5 6 7 8 9 10	2	180	184												
280	10	1 2 3 4 5 6 7 8 9 10	10	81	82	83	84	85	86	87	88	89	181				

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos											
281	1	5	9	56	57	58	59	60	61	62	64	85			
282	1	5	10	127	128	129	130	131	132	133	134	135	181		
283	1	4	10	114	115	116	117	118	119	120	121	123	185		
284	1	8	10	35	36	37	38	39	40	41	42	44	182		
285	10	1 2 3 4 5 6 7 8 9 10	10	53	56	58	60	62	64	66	77	85	183		
286	1	8	10	145	146	147	148	155	158	161	164	168	181		
287	1	9	10	42	43	44	45	46	47	48	49	86	184		
288	3	3 7 9	4	15	16	17	181								
289	1	7	6	155	156	162	170	173	182						
290	1	2	6	43	44	46	56	65	182						
291	10	1 2 3 4 5 6 7 8 9 10	7	103	107	108	110	116	122	200					
292	1	7	10	100	101	102	103	104	105	106	122	124	186		
293	1	9	9	94	95	96	98	100	103	112	116	123			
294	1	8	9	57	58	59	60	61	68	70	72	95			
295	1	5	7	176	177	178	179	180	181	187					
296	1	7	1	182											
297	1	9	9	66	67	68	69	70	71	72	73	86			
298	1	5	10	170	171	172	173	174	175	177	179	181	184		
299	3	1 4 5	10	18	20	22	24	26	28	30	32	87	185		
300	1	7	9	3	6	9	14	21	193	195	197	200			
301	1	8	10	1	190	191	192	193	194	195	196	197	198		
302	2	2 9	9	36	37	41	43	45	47	49	51	182			
303	10	1 2 3 4 5 6 7 8 9 10	10	62	65	67	69	71	75	80	85	90	183		
304	10	1 2 3 4 5 6 7 8 9 10	10	1	2	3	4	5	6	7	8	9	183		
305	3	1 4 5	10	4	5	6	7	8	9	10	11	14	183		
306	1	4	1	152											
307	1	8	7	2	194	195	197	198	199	200					
308	1	7	5	77	81	85	123	195							
309	1	7	8	98	99	100	101	103	107	119	123				
310	3	3 7 9	9	45	47	49	51	53	55	57	91	97			
311	1	2	10	92	93	94	95	101	104	107	116	123	192		
312	1	5	8	1	5	15	180	181	182	188	191				
313	10	1 2 3 4 5 6 7 8 9 10	8	24	25	26	28	29	30	31	184				
314	10	1 2 3 4 5 6 7 8 9 10	10	35	36	37	38	39	40	41	42	45	193		
315	1	4	5	101	104	107	124	186							
316	10	1 2 3 4 5 6 7 8 9 10	9	17	19	21	25	27	30	33	45	184			
317	2	2 9	10	138	139	140	141	142	143	144	145	146	185		
318	10	1 2 3 4 5 6 7 8 9 10	10	67	68	69	70	71	73	75	78	85	185		
319	1	4	10	3	117	119	121	123	125	127	129	138	142		
320	1	7	10	123	124	128	130	132	134	136	138	140	185		

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos												
321	1	4	9	98	103	105	107	109	111	122	125	193				
322	3	1 4 5	8	60	61	62	63	64	65	85	185					
323	10	1 2 3 4 5 6 7 8 9 10	9	4	61	62	63	64	65	66	67	70				
324	10	1 2 3 4 5 6 7 8 9 10	9	18	19	20	21	22	23	24	26	194				
325	1	4	9	38	39	40	41	42	43	44	45	46				
326	10	1 2 3 4 5 6 7 8 9 10	9	30	31	32	33	34	47	57	74	87				
327	1	5	9	143	144	145	146	151	153	156	159	171				
328	10	1 2 3 4 5 6 7 8 9 10	5	65	71	74	83	87								
329	1	8	9	36	37	38	39	40	43	45	86	186				
330	3	3 7 9	10	9	12	15	186	189	190	191	192	193	194			
331	1	4	6	124	126	134	138	142	186							
332	3	1 4 5	9	107	108	109	110	111	112	113	117	124				
333	3	3 7 9	10	30	31	32	33	36	40	44	48	55	187			
334	1	2	9	74	75	76	77	78	79	80	81	86				
335	1	4	8	10	15	22	178	179	183	185	187					
336	10	1 2 3 4 5 6 7 8 9 10	10	18	19	20	21	22	23	24	25	26	187			
337	1	9	9	46	188	189	190	192	194	196	198	200				
338	1	5	9	157	158	159	160	161	162	163	164	165				
339	1	5	8	32	33	34	35	42	44	46	87					
340	1	7	10	14	15	18	21	24	27	36	40	47	187			
341	3	1 4 5	10	98	99	101	103	105	107	109	117	125	188			
342	1	9	10	7	8	9	10	11	12	13	14	15	194			
343	5	1 4 5 7 8	9	87	88	89	90	93	110	118	126	189				
344	1	7	9	172	173	174	175	176	177	178	179	188				
345	1	9	10	138	139	140	141	142	143	144	145	146	188			
346	1	7	4	42	43	44	55									
347	1	7	6	25	27	29	31	35	37							
348	1	7	10	2	182	183	184	185	186	187	188	189	199			
349	1	8	9	87	88	89	90	91	92	93	94	95				
350	1	2	10	28	29	30	31	32	33	37	41	48	188			
351	1	5	8	2	15	189	190	192	194	196	200					
352	1	8	9	17	18	19	20	21	22	23	24	25				
353	1	8	7	49	52	53	54	69	72	125						
354	1	2	10	7	8	9	10	11	12	13	14	16	189			
355	1	4	4	26	28	30	36									
356	3	1 4 5	9	1	47	186	187	191	193	195	197	199				
357	1	2	10	185	186	187	188	189	190	191	192	193	195			
358	3	1 4 5	10	179	180	181	182	184	186	188	190	192	196			
359	10	1 2 3 4 5 6 7 8 9 10	10	128	129	130	131	132	133	134	135	136	197			
360	1	9	5	36	38	40	55	190								

Tabela nº 10 – Instância nº 03 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas	Nº de alunos	Denominação dos alunos											
361	3	3 7 9	9	154	155	156	157	158	159	160	161	162			
362	10	1 2 3 4 5 6 7 8 9 10	10	160	161	162	163	164	165	166	167	168	189		
363	2	4 5	9	97	99	101	103	105	113	116	119	132			
364	1	5	10	106	107	108	109	110	111	113	117	125	190		
365	1	7	10	153	154	155	156	157	158	159	160	161	198		
366	3	1 4 5	4	107	109	113	117								
367	1	9	10	146	147	148	149	150	151	152	153	155	195		
368	1	7	9	6	91	93	95	97	99	108	117	132			
369	2	4 5	9	159	160	161	163	164	165	166	167	191			
370	1	9	10	127	128	129	130	131	132	135	139	143	191		
371	1	9	10	133	134	135	136	137	138	139	140	141	194		
372	1	2	9	77	78	84	91	94	111	119	127	191			
373	10	1 2 3 4 5 6 7 8 9 10	10	46	47	48	49	50	51	52	53	97	192		
374	1	8	10	28	29	30	31	32	33	34	35	86	192		
375	10	1 2 3 4 5 6 7 8 9 10	8	86	93	95	97	99	117	126	192				
376	10	1 2 3 4 5 6 7 8 9 10	9	112	113	114	115	116	117	120	123	126			
377	1	5	10	5	131	132	133	134	135	136	137	139	143		
378	10	1 2 3 4 5 6 7 8 9 10	9	67	68	69	70	71	72	73	74	86			
379	10	1 2 3 4 5 6 7 8 9 10	10	113	114	115	116	117	118	126	128	130	192		
380	10	1 2 3 4 5 6 7 8 9 10	4	132	133	134	136								
381	10	1 2 3 4 5 6 7 8 9 10	10	3	13	14	15	16	17	18	27	38	87		
382	10	1 2 3 4 5 6 7 8 9 10	10	145	146	147	148	149	150	151	152	156	193		
383	10	1 2 3 4 5 6 7 8 9 10	10	56	57	58	59	60	61	76	79	87	199		
384	1	5	9	31	32	33	36	38	40	42	88	196			
385	10	1 2 3 4 5 6 7 8 9 10	6	6	79	80	81	82	83						
386	10	1 2 3 4 5 6 7 8 9 10	8	54	55	56	57	58	59	89	197				
387	10	1 2 3 4 5 6 7 8 9 10	10	77	78	79	80	81	82	83	84	88	193		
388	10	1 2 3 4 5 6 7 8 9 10	9	124	125	126	127	128	129	130	131	133			
389	3	1 4 5	9	157	158	160	162	166	168	170	182	194			
390	3	1 4 5	10	66	68	70	72	74	76	78	80	88	194		
391	3	1 4 5	10	189	190	191	192	193	194	195	196	197	200		
392	5	1 4 5 7 8	8	132	133	134	135	136	137	161	195				
393	5	1 4 5 7 8	6	92	95	104	108	120	126						
394	1	2	10	190	191	192	193	194	195	196	197	198	199		
395	1	2	9	47	48	52	54	56	58	60	71	89			
396	3	3 7 9	7	22	23	24	25	26	27	196					
397	1	2	9	147	148	149	150	152	154	157	160	172			
398	1	2	8	71	72	73	74	75	76	89	197				
399	10	1 2 3 4 5 6 7 8 9 10	10	50	51	52	57	59	62	65	118	127	197		
400	1	2	10	56	57	58	59	60	61	62	74	89	198		

Tabela nº 11 – Instância nº 04 do ITC

Evento	Nº de salas	Denominação das salas					Nº de alunos	Denominação dos alunos										
1	1	9					43	62	70	72	78	83	88	93	131	132	133	134
								135	136	137	138	139	140	141	142	144	147	150
								153	156	158	162	164	174	178	181	184	187	191
2	1	10					25	195	199	202	207	211	215	268	271	275	278	
								27	30	33	37	54	67	69	71	73	74	303
								304	305	306	307	308	309	310	311	312	313	314
3	2	3	6				14	315	316	317								
								144	145	146	147	168	169	172	175	208	210	212
4	4	1	4	6	9		8	214	216	218								
								166	168	170	176	178	180	184	186			
5	1	9					42	78	84	89	94	132	133	134	136	138	140	142
								144	146	148	150	152	154	156	158	159	160	161
								163	172	174	187	188	189	190	191	195	200	203
6	4	6	7	8	10		7	207	211	215	272	275	278	283	290	295		
								119	120	121	122	123	124	125				
								1	3	5	7	9	11	13	15	17	19	20
7	2	5	10				29	23	307	308	309	310	311	312	313	314	323	325
								327	328	329	330	331	332	333				
								3	6	9	12	25	28	31	289	290	293	310
8	3	3	4	6			14	344	347	350								
								16	20	22	24	26	28	30	32			
9	3	6	7	10			8	205	206	207	208	209	210	235	238	241	244	247
								281	285	289	292	295	298	301	325	329	333	337
10	2	4	10				23	341										
								134	141	145	161	162	163	164	165	166	167	168
11	1	9					39	169	170	171	172	173	174	175	176	177	178	179
								180	181	182	183	184	187	192	195	200	203	208
								211	216	268	272	275	296					
12	3	4	5	10			26	2	4	6	8	10	303	304	305	306	307	308
								309	310	311	312	313	314	315	316	317	318	342
								344	346	348	350							
13	5	2	4	5	6	10	3	342	343	344								
								211	212	213	222	236	239	242	245	278	282	286
14	1	6					11	68	69	70	71	72	73	74	75	76	77	85
								86	87	88	89	90	91	92	93	94	115	117
								119	121	123	125							
15	3	4	5	10			26	277	278	281	283	284	285	286	287	288	289	290
								291	292	293	294	295	296	297	298	299	300	301
								302	303	304	305	306	307	309	311	313		
16	1	10					31	281	283									
17	5	2	4	5	6	10	2											

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos										
18	3	4	9	10								22	147	149	151	153	155	175	178	181	184	188	192
													196	200	208	212	216	268	272	275	279	285	291
19	1	9										31	31	32	33	34	35	36	37	38	39	40	41
													42	43	44	45	46	47	48	49	50	51	62
													63	68	69	70	72	79	84	89	94		
20	2	4	10									23	144	145	146	152	158	160	165	167	169	171	173
													177	183	187	189	191	195	197	199	200	204	209
													212										
21	2	9	10									31	2	4	6	8	9	300	301	302	303	304	305
													306	307	308	318	320	322	324	326	328	330	332
													334	336	338	340	342	344	346	348	350		
22	4	4	6	9	10							15	12	13	14	15	34	37	40	45	344	345	346
													347	348	349	350							
23	4	6	7	8	10							2	148	149									
24	1	3										21	23	24	25	29	35	38	41	44	46	48	50
													52	54	64	66	68	74	99	102	105	108	
25	1	4										26	2	4	6	8	10	12	14	16	18	334	335
													336	337	338	339	340	341	342	343	344	345	346
													347	348	349	350							
26	5	2	4	5	6	10						3	223	225	227								
27	6	4	6	7	8	9	10					6	166	167	169	170	171	172					
28	1	3										24	180	181	183	184	185	186	187	188	189	190	191
													192	193	195	199	202	205	208	211	214	217	220
													233	268									
29	6	1	4	5	6	9	10					14	168	169	170	171	172	173	174	175	176	177	178
													179	180	181								
30	1	6										14	168	169	170	171	172	173	174	175	176	177	178
													180	182	184								
31	1	4										23	119	120	121	122	123	124	125	126	127	128	129
													130	131	132	134	135	136	137	138	139	140	142
													143										
32	1	9										42	90	91	92	93	94	215	216	217	218	219	220
													221	222	223	224	225	226	227	228	229	230	231
													232	233	234	235	236	237	238	240	249	253	261
													262	264	265	269	286	287	88	289	291		
33	2	6	7									9	181	182	183	185	186	187	188	189	190		
34	1	9										43	72	79	84	92	94	104	105	106	107	108	109
													110	111	112	113	124	126	128	130	146	149	159
													162	165	172	175	178	181	184	188	254	258	261
													263	265	267	269	270	272	275	279	286	292	

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos													
35	2	5	10									27	190	192	194	196	198	200	202	204	214	216	218			
													219	220	221	222	223	224	225	226	230	234	255			
													259	262	264	266	268									
36	4	6	7	8	10							7	222	223	224	225	226	232	234							
37	4	6	7	8	10							3	147	148	149											
38	1	10										31	1	3	4	302	303	304	305	306	307	308	309			
													310	311	312	313	314	315	316	317	318	319	320			
													321	322	326	330	334	338	342	346	349					
39	6	1	3	4	6	7	9					10	301	302	303	304	305	306	307	308	309	311				
40	2	5	10									29	264	265	266	267	268	269	270	271	272	273	274			
													275	276	277	278	279	280	281	282	283	284	285			
													286	287	290	294	298	302	310							
41	6	4	6	7	8	9	10					7	104	105	106	107	108	109	110							
42	1	4										19	135	136	137	138	139	140	142	143	147	148	149			
													150	151	152	153	155	156	182	184						
43	5	2	4	5	6	10						3	160	161	162											
44	4	4	5	6	10							13	20	21	22	23	24	25	26	27	28	45	47			
													49	50												
45	2	4	9									26	103	104	105	106	108	110	112	114	116	118	120			
													122	124	126	128	129	130	131	132	133	134	135			
													136	137	138	139										
46	1	10										29	2	4	6	8	10	12	14	15	299	300	301			
													302	303	304	305	306	307	308	309	310	311	312			
													320	322	342	344	346	348	350							
47	5	2	4	5	6	10						3	187	188	189											
48	1	3										27	1	4	7	10	13	16	19	22	25	28	31			
													34	38	45	58	332	333	334	335	336	337	338			
													339	340	342	345	348									
49	1	3										27	1	2	3	4	5	6	7	8	9	11	13			
													15	17	19	21	23	25	27	29	31	33	35			
													37	39	45	47	350									
50	1	3										27	114	115	116	117	118	119	120	121	122	123	125			
													127	129	131	147	150	160	163	169	173	176	179			
													182	185	189	256	260									
51	2	3	6									16	35	36	37	38	39	40	41	42	43	44	45			
													46	47	48	49	50									
52	1	6										16	48	49	50	51	52	53	54	55	56	57	58			
													59	60	61	62	63									
53	4	1	4	6	9							8	70	80	85	89	93	100	103	106						
54	3	4	6	9								16	165	166	167	168	169	170	171	172	173	174	175			

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas						Nº de alunos	Denominação dos alunos											
55	1	10						18	176	177	178	179	180							
									221	249	251	253	255	257	259	261	263	265	267	
56	1	9						33	269	271	273	274	275	276	277					
									58	73	79	84	89	94	259	260	261	262	263	
									264	265	266	267	268	269	270	271	272	273	274	
									275	276	278	280	282	283	287	294	295	297	298	
57	1	9						39	69	70	73	79	206	207	208	209	213	214	215	
									216	217	218	219	220	221	222	223	224	225	226	
									227	228	229	230	231	232	234	241	249	253	257	
									262	264	266	269	275	280						
58	5	4	6	7	8	10		7	78	79	80	81	82	83	84					
59	1	6						16	209	210	211	212	213	214	221	230	233	236	239	
									241	243	245	247	249							
60	1	6						16	270	271	272	273	274	275	276	277	278	279	280	
									281	282	283	284	285							
61	4	6	7	8	10			7	280	281	282	283	284	285	286					
62	3	6	7	10				10	299	300	301	302	303	304	305	306	307	308		
63	4	4	6	7	10			10	249	250	251	252	253	254	255	256	257	258		
64	5	1	3	4	6	9		14	69	70	71	72	73	74	75	76	77	78	79	
									80	81	82									
65	1	3						27	154	155	156	157	158	159	160	161	162	163	164	
									165	179	181	183	185	186	187	188	189	190	191	
									192	193	194	196	198							
66	1	3						27	27	30	33	38	45	64	314	315	316	317	318	
									319	320	321	322	323	324	325	326	327	328	329	
									330	331	332	333	334							
67	5	2	4	5	6	10		3	157	158	159									
68	2	3	4					22	150	151	152	153	154	155	156	157	158	159	160	
									161	162	163	164	173	178	181	185	189	193	196	
69	1	10						31	283	284	285	286	287	288	289	290	291	292	293	
									294	295	296	297	298	299	300	301	302	303	304	
									305	306	307	308	309	310	311	312	350			
70	5	2	4	5	6	10		3	202	203	204									
71	1	6						15	238	239	240	241	243	244	245	246	247	248	249	
									250	251	252	253								
72	1	3						27	222	223	224	225	226	227	228	229	230	231	232	
									233	234	235	236	237	238	239	240	241	242	243	
									244	245	246	247	250							
73	1	9						43	86	87	88	90	95	235	236	237	238	239	240	
									241	242	243	244	245	246	247	248	249	250	251	

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos										
94	3	4	5	10								21	1	3	5	7	9	11	13	330	333	334	335
													336	337	338	339	340	341	343	345	347	349	
95	2	9	10									27	90	95	144	145	146	147	148	149	150	162	164
													178	197	200	203	206	209	212	215	218	221	235
													271	275	283	287	293						
96	5	2	4	5	6	10						1	216										
97	1	9										31	1	7	12	17	18	56	57	58	74	80	85
													90	95	278	279	280	284	285	286	287	288	289
													291	292	293	304	305	306	308	310	312		
98	1	6										16	53	54	55	56	57	58	59	60	61	62	63
													64	65	66	67	68						
99	5	3	6	7	8	10						7	254	255	256	257	258	259	260				
100	1	10										28	71	72	73	74	75	76	77	78	79	81	82
													83	84	86	87	88	89	90	91	92	93	94
													96	97	98	99	100	101					
101	2	4	10									19	205	206	207	208	209	210	211	212	213	214	215
													216	217	218	219	220	221	235	242			
102	1	10										29	91	92	93	94	96	97	98	99	100	101	102
													103	104	105	106	107	108	109	110	111	112	113
													114	116	119	121	123	125	126				
103	4	6	7	8	10							7	257	258	259	260	261	262	263				
104	4	1	4	6	9							14	314	315	317	319	321	323	325	327	329	331	333
													334	335	336								
105	1	10										26	201	202	204	205	209	213	217	219	221	223	225
													227	229	231	233	235	237	239	241	243	245	250
													254	258	290	292							
106	1	4										26	1	5	6	7	8	9	10	11	13	334	335
													336	337	338	339	340	341	342	343	344	345	346
													347	348	349	350							
107	4	4	6	7	9							10	277	278	279	280	281	282	283	284	285	286	
108	1	4										26	175	176	177	178	179	180	181	182	191	193	195
													197	199	201	203	205	207	209	211	213	215	218
													221	230	236	243							
109	1	3										27	318	319	320	321	322	323	324	325	326	327	328
													329	330	331	332	333	334	335	336	337	338	339
													340	341	342	343	345						
110	2	6	7									10	96	97	98	99	100	101	102	103	104	105	
111	2	3	10									25	224	226	228	229	230	231	233	234	237	240	243
													246	279	283	287	290	296	299	323	327	331	335
													339	344	347								

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas						Nº de alunos	Denominação dos alunos											
112	5	2	4	5	6	10		3	210	211	212									
113	1	9						43	3	6	9	12	15	17	19	51	52	53	54	
									55	59	74	80	85	91	95	323	324	325	326	
									327	328	329	330	331	332	333	334	335	336	337	
									338	339	340	341	342	343	344	345	347	350		
114	5	2	4	5	6	10		3	279	280	282									
115	1	9						43	31	32	33	34	35	36	37	38	39	40	41	
									42	43	44	45	46	47	48	49	50	51	52	
									53	54	55	56	57	58	59	60	61	62	63	
									65	67	68	69	70	74	80	86	91	96		
116	3	5	9	10				29	128	129	130	131	132	133	134	135	137	139	141	
									143	145	148	151	153	155	156	157	158	216	219	
									224	231	237	244	250	254	259					
117	1	6						16	181	182	183	184	185	186	187	188	189	190	201	
									203	205	207	210	213							
118	1	6						15	187	188	189	191	192	193	194	195	196	197	198	
									199	201	203	205								
119	1	3						27	27	28	29	30	31	32	33	34	35	36	37	
									38	39	40	41	42	43	44	45	46	47	48	
									49	50	51	52	53							
120	2	5	10					29	1	4	7	10	13	253	254	255	256	257	258	
									259	260	261	262	263	264	289	292	295	297	309	
									311	313	315	317	319	322	348					
121	5	1	3	4	6	9		14	75	76	79	81	83	85	87	89	91	93	95	
									97	99	101									
122	1	9						43	52	56	57	58	85	86	87	88	89	91	96	
									165	166	167	168	169	170	171	179	180	181	182	
									183	184	185	186	187	188	189	190	191	192	193	
									194	195	196	197	198	201	204	209	212	217		
123	2	4	6					15	150	151	153	157	159	164	166	168	170	172	174	
									180	186	194	198								
124	6	1	4	5	6	9	10	14	64	65	66	67	68	69	70	71	72	73	74	
									75	76	77									
125	2	4	9					26	132	133	134	135	136	137	138	139	140	141	142	
									143	144	145	148	151	161	164	170	174	177	180	
									183	186	236	257								
126	1	6						16	42	43	44	45	46	47	48	49	50	51	53	
									55	57	59	61	63							
127	6	4	6	7	8	9	10	7	148	149	150	151	152	153	154					
128	1	6						16	266	267	268	269	270	271	272	273	274	275	276	

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos											
129	1	4										26	288	302	305	308	311							
													2	5	8	11	14	17	20	23	26	29	32	
													35	39	56	324	325	326	327	328	329	330	331	
													341	343	346	349								
130	5	2	4	5	6	10						3	53	54	55									
131	2	6	7									10	104	105	106	107	108	109	110	111	112	113		
132	4	1	4	6	9							14	191	192	193	194	195	196	197	198	199	200	202	
													204	206	208									
133	2	5	10									28	116	117	118	119	120	121	122	130	132	134	136	
													138	140	142	146	148	150	154	156	179	182	185	
													211	215	219	251	254	264						
134	5	1	3	4	6	9						14	110	111	112	113	114	115	116	117	118	119	120	
													132	134	136									
135	1	6										15	234	235	237	242	244	246	248	250	252	263	266	
													270	272	274	276								
136	5	2	4	5	6	10						3	257	258	259									
137	1	9										37	129	130	135	142	146	214	215	216	217	218	219	
													220	221	222	223	224	225	226	227	228	229	230	
													231	232	234	236	237	255	260	262	264	266	269	
													273	276	281	288								
138	1	9										43	20	21	22	23	24	25	26	27	28	38	40	
													42	44	46	48	49	50	51	52	53	54	55	
													56	57	58	59	60	61	62	63	64	65	66	
													67	68	69	70	71	74	81	86	91	96		
139	1	9										43	74	81	86	91	96	227	228	229	230	231	233	
													235	236	237	238	239	240	241	242	243	244	245	
													246	247	248	259	260	261	262	263	264	268	270	
													272	274	275	276	277	278	279	281	288	293		
140	8	2	3	4	5	6	7	8	10			3	249	250	251									
141	4	1	4	6	9							14	204	205	208	210	212	214	216	219	221	223	225	
													227	229	231									
142	4	6	7	8	10							7	62	63	64	65	66	67	68					
143	1	3										27	159	160	161	162	163	164	165	166	167	168	169	
													170	171	172	173	174	175	176	177	178	180	183	
													186	212	216	246	252							
144	1	9										42	59	75	106	107	146	150	151	152	153	154	155	
													156	157	158	159	160	161	162	163	164	165	166	
													167	170	173	176	179	182	185	189	193	196	201	
													204	209	212	219	292	293	294	295	296			
145	2	5	10									29	1	3	5	7	309	310	311	312	313	314	315	

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos											
													316	317	319	321	323	325	327	329	331	333	335	
146	5	3	4	5	6	10						14	337	339	341	343	345	347	349					
													110	111	148	150	152	154	156	176	179	182	185	
147	4	6	7	8	10							7	190	193	201									
148	3	4	9	10								25	301	303	305	307	309	311	313					
													1	4	7	10	13	26	29	32	82	83	84	
													85	87	88	89	90	92	97	312	313	314	315	
													316	317	348									
149	5	2	4	5	6	10						3	142	143	144									
150	2	6	7									10	177	178	179	180	181	182	183	184	185	186		
151	1	10										31	157	158	159	160	161	162	163	164	165	166	167	
													168	169	170	171	172	173	174	175	176	177	179	
													182	185	190	193	197	201	205	210	213			
152	4	1	4	6	9							5	145	146	152	153	158							
153	2	6	7									6	253	254	259	260	261	264						
154	1	3										27	302	304	306	308	310	312	314	315	316	317	318	
													319	320	321	322	323	324	325	326	327	328	329	
													330	332	335	337	339							
155	4	1	4	6	9							14	235	236	237	238	239	240	241	242	243	244	245	
													246	247	248									
156	5	2	4	5	6	10						2	162	165										
157	8	2	3	4	5	6	7	8	10			3	93	94	95									
158	3	3	6	7								0												
159	5	2	4	5	6	10						3	170	171	172									
160	2	3	4									25	196	197	198	199	206	208	210	212	214	216	218	
													220	222	224	226	228	230	232	234	236	238	240	
													242	244	246									
161	3	4	5	10								19	35	38	41	46	48	50	52	310	311	312	314	
													315	316	317	318	320	338	340	342				
162	5	2	4	5	6	10						3	165	166	167									
163	3	6	7	10								8	14	15	16	17	18	19	22	23				
164	5	2	4	5	6	10						3	95	96	97									
165	4	4	6	7	10							10	192	193	194	195	196	200	201	202	203	204		
166	2	9	10									31	65	66	67	68	69	71	73	75	77	79	81	
													83	85	87	89	93	94	95	96	97	112	113	
													137	139	141	143	144	145	146	213	289			
167	4	6	7	8	10							7	30	31	32	33	36	39	42					
168	2	6	7									10	34	35	36	37	38	39	40	41	42	43		
169	4	6	7	8	10							7	151	152	153	154	155	157	165					
170	2	4	6									16	317	318	319	320	321	322	323	324	325	326	327	

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos											
171	2	9	10									31	328	329	330	331	332							
													2	5	8	11	14	16	18	265	267	269	271	
													273	275	277	279	281	283	285	287	290	293	296	
													298	310	312	314	316	318	320	346	349			
172	3	3	4	9								26	180	181	182	183	184	185	186	187	188	189	190	
													191	192	193	194	195	226	228	230	251	254	257	
													260	263	291	295								
173	4	1	4	6	9							14	121	122	123	124	125	126	127	136	138	140	142	
													144	146	149									
174	3	4	6	7								10	291	292	293	294	295	296	297	298	299	300		
175	1	10										21	239	240	241	242	243	244	245	246	247	248	249	
													250	251	252	253	254	255	256	257	258	278		
176	3	5	9	10								29	95	96	97	98	99	100	101	102	103	104	105	
													106	107	108	109	110	111	112	113	114	116	118	
													120	122	124	126	127	128	129					
177	3	3	6	7								9	58	59	60	61	62	63	64	65	66			
178	5	4	6	7	8	10						0												
179	1	6										16	9	10	11	12	13	14	15	16	17	18	19	
													20	21	22	23	24							
180	2	5	10									29	19	24	269	270	212	222	232	24	275	276	277	
													278	279	280	281	282	283	284	285	286	287	288	
													289	290	291	292	293	294	295					
181	1	10										30	2	277	278	279	280	281	282	283	284	285	286	
													287	288	289	290	291	292	293	294	295	296	297	
													298	299	300	338	340	346	347	349				
182	5	2	4	5	6	10						0												
183	4	6	7	8	10							7	104	105	106	107	108	114	116					
184	1	6										14	18	20	21	22	23	24	25	26	27	28	29	
													30	31	32									
185	2	3	10									27	20	21	22	23	24	25	26	27	28	29	30	
													31	32	33	34	35	36	37	38	39	40	41	
													42	43	44	46	48							
186	1	4										26	187	188	189	190	191	192	193	194	195	196	197	
													198	199	200	201	202	203	204	205	206	207	208	
													209	213	217	247								
187	3	4	5	10								21	2	4	6	8	10	12	14	16	18	20	22	
													24	26	28	30	41	124	126	346	348	350		
188	1	3										27	277	278	279	280	281	282	283	284	285	286	287	
													288	289	290	291	292	293	294	295	296	297	298	
													299	300	301	302	303							

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos																
189	4	6	7	8	10							7	183	184	185	186	187	188	189										
190	1	4										22	1	3	5	7	9	11	13	15	17	19	21						
191	1	10										24	158	159	160	161	162	163	164	165	166	167	168						
													169	170	171	172	175	177	179	182	198	202	206						
													210	213															
192	2	3	4									23	226	227	228	229	230	231	232	233	234	236	238						
193	2	4	9									25	240	242	244	246	250	254	256	258	262	264	266						
													268																
													60	61	62	63	64	65	66	67	68	69	70						
													72	73	74	75	76	77	78	79	80	81	82						
													83	84	85														
194	8	2	3	4	5	6	7	8	10			3	343	344	345														
195	4	6	7	8	10							0																	
196	7	3	4	5	6	7	8	10				7	221	222	223	224	225	226	227										
197	2	5	10									28	1	3	5	7	9	11	13	15	17	19	21						
198	4	1	4	6	9							14	23	25	28	31	34	46	65	66	67	335	336						
													337	338	339	340	347	349											
													159	160	161	162	163	164	165	166	167	168	169						
													170	171	172														
199	2	6	10									11	2	5	8	11	24	27	30	33	343	346	349						
200	1	10										31	5	295	296	297	298	299	300	301	302	303	304						
201													305	306	307	308	309	310	311	312	313	314	316						
													320	325	333	335	338	340	346	348	349								
													46	47	48	49	50	51	52	53	54	55							
202	2	4	9									26	1	3	5	7	25	28	31	34	52	55	325						
203	3	3	4	6								14	326	327	328	329	330	331	333	335	337	339	341						
													343	345	347	349													
													161	162	163	165	166	167	168	169	171	173	176						
													179	190	191														
204	1	3										27	219	220	222	224	226	228	230	232	234	236	238						
205													240	242	244	246	247	248	249	250	251	252	253						
													254	255	256	257	258												
													110	111	112														
206	4	1	4	6	9							14	215	216	217	218	219	220	222	231	234	237	240						
207	5	2	4	5	6	10							242	244	246														
													105	106	107														
208	1	6										16	108	109	110	111	112	113	114	115	116	117	118						
209	1	3											130	145	148	151	154												
													1	4	21	24	27	329	330	331	332	333	334						

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos																		
210	1	6											15	335	336	337	338	339	340	341	342	343	344	345							
														346	347	348	349	350													
														61	62	63	64	70	72	76	78	80	82	84							
														86	88	90	92														
211	2	6	7											10	223	224	225	226	227	228	229	232	235	238							
212	5	2	4	5	6	10											3	166	167	168											
213	1	4											23	191	192	193	194	195	196	197	198	199	200	201							
														202	203	204	205	206	208	209	210	212	213	214							
														265																	
214	2	3	6											16	40	41	42	43	44	45	46	47	48	49	50						
															51	52	53	54	55												
215	7	1	2	4	5	6	9	10											3	198	199	200									
216	2	5	10											29	2	5	22	25	28	30	32	290	291	292	293						
															294	295	296	297	298	299	300	301	303	306	309						
															312	314	316	318	320	322	324										
															206	207	209	211	213	215	217										
217	4	6	7	8	10											7	206	207	209	211	213	215	217								
218	3	5	9	10											29	10	12	14	16	18	20	22	24	26	28	30					
																32	34	36	38	43	46	48	49	50	341	342					
																343	344	345	346	347	348	349									
219	1	9											43	52	56	59	75	81	86	92	97	147	148	149							
														150	156	163	166	168	170	172	174	176	178	180							
														211	214	216	218	220	222	224	226	228	231	237							
														247	252	255	260	263	265	267	270	273	276								
220	3	6	7	10											9	11	12	13	14	15	16	17	18	19							
221	3	4	5	10											23	61	68	69	70	71	72	73	74	75	76	77					
																78	79	80	81	82	83	84	85	86	87	88					
																89															
222	1	9											33	33	34	35	36	37	38	39	40	41	52	54							
														56	58	60	62	64	65	68	74	77	80	83							
														86	89	95	127	128	129	130	131	133	140	142							
223	5	2	4	5	6	10											3	208	209	210											
224	5	1	3	4	6	9											14	90	91	92	93	94	95	96	97	98	99	100			
																		101	102	103											
225	1	6											16	108	109	110	111	112	113	114	115	116	117	118							
														119	120	121	122	123													
226	1	9											39	102	103	112	113	114	115	116	117	118	119	120							
														121	122	123	124	125	140	141	142	143	144	145							
														146	147	148	149	195	197	199	205	206	210	214							
														267	270	274	277	282	294												
227	2	3	10											27	14	15	19	21	23	25	27	29	31	33	34						

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas							Nº de alunos	Denominação dos alunos													
228	3	4	6	9				16	35	36	37	38	39	40	41	42	43	44	45				
									46	47	48	49	50										
									190	191	192	193	194	195	196	197	198	199	200				
									201	202	203	205	206										
229	4	4	5	6	10			9	39	40	41	42	43	64	65	66	67						
230	1	6						13	74	77	78	79	86	90	94	98	101	104	107				
231	4	6	7	8	10		6	109	111														
								313	314	316	317	318	319										
								66	91	95	102	105	108	110	112	113	114	115					
								116	117	118	119	120	121	122	123	125	127	128					
232	1	9					33	129	130	131	132	133	134	135	136	137	140	143					
								11	12	13	14	15	16	17	18	19	20	21					
								22	23	24	25	26											
								34	35	36	37	38	39	40	41	42	43	44					
233	1	6					16	45	46	47	48	49											
								251	252	253	254	255	256	257	258	259	260	261					
								262	263	264													
								108	109	110	111	113	114										
236	6	1	4	5	6	9	10	6	9	11	16	17	19	75	81	87	92	97	231				
									232	233	234	235	236	237	238	239	240	241	242				
									243	244	245	246	247	248	249	252	255	258	261				
									288	292	296	299	304	311	313								
238	5	1	3	4	6	9	14	306	307	308	309	310	311	312	313	314	315	316					
								317	318	319													
								226															
								71	72	73													
240	5	2	4	5	6	10	3	38	39	40	41	42	43	44	45	46	47	48					
								49	50	51													
								160	161	162	163	164	165	166	167	168	169						
								201	202	203	204	205	206	208	210	212	214	216					
241	4	1	4	6	9		14	218	220	222													
								186	187	188	189	190	191	192									
								119	120	121	122	123	124	125	126	127	128	129					
								143	146	149	152	155	157	159	163	173	176	180					
242	2	6	7				10	183	186	190	194												
244	4	6	7	8	10		7																
245	2	4	10				26																
246	4	6	7	8	10		0																
247	3	3	5	10			27	283	284	285	286	287	288	289	290	291	292	293					
								294	295	296	297	298	299	300	301	302	303	304					
								305	306	307	308	309											
248	2	4	9					26	2	4	6	8	10	12	14	16	18	20	22				

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos													
249	1	4											26	24	26	29	32	35	44	47	341	342	343	344		
														345	346	348	350									
														227	228	229	230	231	232	233	234	235	236	237		
														238	239	240	241	242	243	244	245	246	247	248		
														249	250	251	252									
250	1	6											16	113	114	115	116	117	118	119	120	121	122	123		
														124	125	126	127	128								
251	4	6	7	8	10						7	333	334	335	336	337	338	339								
252	3	4	6	7						2	78	83														
253	5	2	4	5	6	10						3	89	90	91											
254	1	9											42	78	82	87	92	123	124	125	126	127	128	129		
														131	133	135	137	139	141	143	145	147	149	151		
														155	157	181	184	210	214	218	248	253	256	265		
														266	267	268	270	274	277	282	290	294				
255	1	9											32	33	36	39	44	47	49	51	67	68	69	70		
														71	72	73	74	76	77	82	143	302	303	304		
														305	306	307	308	309	319	337	339	341	343			
256	6	1	4	5	6	9	10						14	53	54	55	56	57	58	59	60	61	62	63		
								64	65	66																
257	4	6	7	8	10						7	74	75	76	77	78	79	80								
258	10	1	2	3	4	5	6	7	8	9	10	3	339	340	341											
259	5	4	6	7	9	10						10	315	316	317	318	319	320	321	322	323	324				
260	7	3	4	5	6	7	8	10						7	326	327	328	329	330	331	332					
261	2	4	6											15	220	222	223	224	225	226	227	228	229	230	231	
															232	233	234	249								
262	1	10											31	8	10	196	197	198	199	200	207	209	211	213		
														215	217	219	221	223	224	225	227	229	250	253		
														256	259	262	289	293	297	301	305	312				
263	5	2	4	5	6	10						3	227	228	229											
264	1	3											27	12	14	15	295	296	297	298	299	300	301	302		
														303	304	305	306	307	308	309	310	311	312	313		
														314	317	320	330	333								
265	4	1	4	6	9						14	32	33	34	35	36	37	38	39	40	41	42				
						43	44	45																		
266	5	3	6	7	8	10						7	341	342	343	344	345	346	347							
267	5	3	6	7	8	10						4	72	83	88	96										
268	1	9											40	25	54	60	133	141	144	147	148	149	150	151		
														152	153	154	155	156	157	158	159	170	171	172		
														173	174	175	176	177	180	183	186	191	194	199		
														202	207	211	214	267	271	274						

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos										
269	2	9	10									28	54	57	61	130	131	132	133	134	135	136	137
													138	139	140	141	142	143	171	174	207	209	211
													213	215	217	219	220	265					
270	1	10										31	3	4	5	6	7	8	9	10	11	12	13
													14	15	16	17	18	19	20	21	22	23	24
													25	26	27	28	29	30	31	33	51		
271	1	6										16	144	145	146	147	148	149	150	151	152	154	156
													158	160	162	164	165						
272	1	10										24	1	2	3	7	9	11	13	15	17	19	21
													23	25	27	29	31	66	69	75	78	81	84
													87	90									
273	10	1	2	3	4	5	6	7	8	9	10	3	161	162	163								
274	1	9										35	51	54	57	71	76	82	87	92	97	98	99
													103	105	107	109	111	113	115	117	119	121	123
													125	127	129	133	141	186	191	194	211	213	267
													271	274									
275	5	3	6	7	8	10						7	110	111	112	113	114	115	116				
276	3	4	6	7								5	275	276	297	298	299						
277	4	1	4	6	9							5	323	324	325	326	327						
278	2	6	7									8	63	64	65	66	67	68	69	70			
279	5	1	3	4	6	9						14	29	30	31	32	33	34	35	36	37	39	41
													43	45	47								
280	2	4	9									26	114	115	116	117	118	119	120	121	122	123	124
													125	126	127	128	129	130	131	132	133	134	135
													136	138	140	142							
281	2	4	10									25	1	3	5	7	9	319	320	321	322	323	324
													325	326	327	328	329	330	331	332	340	341	343
													345	347	349								
282	4	6	7	8	10							7	172	173	174	175	176	177	178				
283	5	4	6	7	8	10						7	55	56	57	58	59	60	61				
284	4	6	7	8	10							6	248	249	250	251	256	266					
285	4	4	6	7	10							10	109	110	111	112	113	115	117	118	119	120	
286	4	6	7	8	10							3	64	73	84								
287	2	9	10									31	77	78	80	82	84	86	88	90	92	94	96
													98	100	102	103	111	112	113	114	115	116	117
													118	119	120	121	122	123	124	125	126		
288	2	6	7									10	52	53	54	55	56	57	58	59	60	61	
289	1	10										30	92	93	94	96	97	98	99	100	101	102	103
													121	122	123	124	125	126	127	128	129	130	131
													132	133	134	135	136	137	138	139			

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas										Nº de alunos	Denominação dos alunos										
290	1	6										15	254	255	256	258	259	260	261	262	263	264	265
													266	267	268	269							
291	3	5	9	10								27	2	252	253	254	264	267	269	271	273	275	276
													277	280	284	288	291	294	297	300	324	328	332
													336	340	345	348	350						
292	6	1	4	5	6	9	10					14	266	268	270	272	274	276	278	280	282	284	286
													288	291	294								
293	4	6	7	8	10							7	223	224	225	226	227	228	229				
294	1	9										35	78	79	80	81	83	84	85	86	88	89	90
													91	93	94	95	96	97	98	99	100	101	102
													103	104	105	106	107	108	109	141	142	143	144
													145	146									
295	3	3	6	7								5	17	328	329	331	332						
296	2	4	9									25	144	145	146	147	148	149	150	151	152	153	154
													155	156	157	158	159	160	164	174	177	180	183
													187	192	209								
297	1	6										16	29	30	31	32	33	34	35	36	37	38	39
													40	41	42	43	44						
298	7	1	2	4	5	6	9	10				3	220	221	222								
299	5	1	3	4	6	9						14	218	221	223	225	227	229	231	233	235	237	239
													241	243	245								
300	2	6	7									10	315	316	317	318	319	320	321	322	324	326	
301	2	6	7									9	14	15	16	17	18	19	20	21	22		
302	2	4	10									25	252	253	254	255	256	257	258	259	260	261	262
													263	264	265	266	267	268	269	273	275	290	292
													294	296	297								
303	6	4	6	7	8	9	10					7	333	334	335	336	337	338	342				
304	3	5	9	10								28	2	4	6	8	26	29	32	36	53	68	70
													72	319	320	321	322	323	324	332	334	336	338
													340	342	344	346	348	350					
305	1	9										41	6	10	16	18	53	57	71	76	83	87	93
													298	299	300	301	302	303	304	305	306	307	308
													309	310	311	312	313	314	315	316	317	318	321
													333	334	336	339	341	347	348	350			
306	5	1	3	4	6	9						14	177	178	179	180	181	182	183	184	185	186	187
													188	189	204								
307	3	3	6	7								10	259	260	261	262	263	265	268	270	272	274	
308	3	5	9	10								27	121	122	123	124	125	126	127	128	129	130	131
													133	135	137	138	139	140	141	142	143	144	145
													146	147	194	252	256						

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas							Nº de alunos	Denominação dos alunos											
309	5	2	4	5	6	10			3	326	327	328									
310	1	4							26	59	60	61	62	63	64	65	66	67	68	69	
										70	71	72	73	74	75	76	77	78	79	80	
										81	82	83	84								
311	4	1	4	6	9				14	98	99	100	101	102	103	104	105	106	107	108	
										109	110	111									
312	2	5	10						28	131	132	133	134	135	142	144	146	148	150	152	
										154	156	158	160	162	164	168	171	174	177	180	
										184	187	266	269	271	273						
313	2	4	10						22	26	27	28	34	37	40	43	45	47	49	51	
										53	63	65	67	69	71	73	75	101	104	107	
314	2	3	10						23	4	5	6	8	10	12	14	16	18	20	22	
										24	26	28	30	32	67	70	76	82	85	88	
										91											
315	3	4	6	9					15	115	116	117	118	130	131	132	133	134	135	136	
										137	138	139	140								
316	1	9							27	16	18	22	25	26	27	28	29	30	52	55	
										58	71	77	83	88	93	296	297	298	299	300	
										301	302	315	318	321							
317	1	4							24	132	133	134	135	136	137	138	139	140	141	142	
										143	173	176	194	201	204	207	210	213	216	219	
										232	238										
318	5	2	4	5	6	10			3	233	236	240									
319	3	4	5	10					20	100	101	102	104	106	108	110	112	114	116	118	
										120	122	124	126	128	188	210	212	214			
320	5	2	4	5	6	10			2	59	60										
321	4	1	4	6	9				13	195	197	198	199	200	201	202	203	204	205	206	
										207	208										
322	5	4	6	7	8	10			7	126	127	128	129	130	131	132					
323	5	3	6	7	8	10			3	19	20	21									
324	2	3	6						14	297	298	307	308	309	310	311	312	313	314	331	
										334	337	339									
325	1	9							43	3	6	9	12	15	18	21	24	27	30	33	
										36	44	57	59	60	61	62	63	64	65	66	
										67	68	69	72	77	83	88	93	98	315	316	
										317	318	319	320	321	322	323	344	347	350		
326	3	4	5	10					18	238	239	241	243	245	247	249	251	262	265	267	
										269	273	277	279	280	281	282					
327	3	4	5	10					22	192	193	194	195	196	197	201	202	203	204	205	
										206	207	208	248	250	251	270	272	276	293	295	

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas								Nº de alunos	Denominação dos alunos											
328	1	10								30	88	89	90	91	92	93	94	95	96	97	98	
											99	100	101	102	103	104	105	106	107	108	109	
											110	111	112	113	114	115	116	117				
329	1	4								26	239	240	241	242	243	244	245	246	247	248	249	
											250	251	252	253	254	255	256	257	258	259	260	
											261	262	263	289								
330	7	1	2	4	5	6	9	10		3	138	139	140									
331	5	1	3	4	6	9				14	8	9	10	11	12	13	14	15	16	17	18	
											19	20	21									
332	5	2	4	5	6	10				3	284	285	286									
333	5	2	4	5	6	10				3	221	222	223									
334	2	6	7							10	313	316	318	320	322	324	326	328	330	332		
335	3	5	9	10						28	34	35	36	37	38	39	40	41	42	43	44	
											45	46	47	48	49	50	51	52	53	54	55	
											56	57	89	117	120	122						
336	1	4								26	58	59	60	61	62	63	64	65	66	67	68	
											69	70	71	72	73	81	82	83	84	85	86	
											87	88	115	118								
337	1	4								26	158	159	160	161	162	164	167	169	171	173	175	
											177	179	209	212	215	217	219	221	223	225	221	
											229	232	238	248								
338	3	4	6	10						0												
339	5	2	4	5	6	10				3	50	51	52									
340	1	9								43	77	83	88	93	98	229	230	231	232	233	234	
											235	236	237	238	239	240	241	242	243	244	245	
											246	247	248	249	250	251	252	253	254	255	256	
											261	263	265	268	271	274	278	282	290	294		
341	2	4	6							12	248	252	254	256	258	260	262	264	266	268	27	
											272											
342	2	3	6							16	198	199	200	201	202	203	218	220	222	224	226	
											228	230	232	233	234							
343	1	3								27	1	2	3	4	5	6	7	321	322	323	324	
											325	326	327	328	329	330	331	332	343	344	345	
											346	347	348	349	350							
344	6	1	4	5	6	9	10			14	136	137	138	139	140	141	142	143	153	155	157	
											159	161	163									
345	1	6								16	102	103	104	105	106	107	108	109	117	118	119	
											120	147	150	152	154							
346	4	1	4	6	9					14	100	101	102	107	109	111	113	115	117	119	121	
											123	125	127									

Tabela nº 11 – Instância nº 04 do ITC (Continuação)

Evento	Nº de salas	Denominação das salas						Nº de alunos	Denominação dos alunos											
347	3	3	5	10				27	264	265	266	261	268	269	270	271	272	273	274	
									275	276	277	278	279	280	281	282	283	284	285	
									286	287	288	290	291							
348	5	1	3	4	6	9		14	7	8	9	10	11	12	13	14	15	16	17	
									18	19	20									
349	1	3						27	183	184	185	186	187	188	189	190	192	194	196	
									198	200	202	204	206	208	210	212	214	217	220	
									225	233	239	248	253							
350	1	3						27	231	232	233	234	235	236	237	238	239	240	241	
									242	243	244	245	246	241	248	249	250	251	252	
									253	270	271	272	273							