

Thiago Augusto de Oliveira Silva

**Otimização Determinística e Estocástica de
Sequenciamento e Alocação de Recursos
em Cirurgias**

Belo Horizonte,

Abril (2015)

Thiago Augusto de Oliveira Silva

Otimização Determinística e Estocástica de Sequenciamento e Alocação de Recursos em Cirurgias

Tese apresentada ao Programa de Pós-Graduação em Engenharia de Produção da Escola de Engenharia da Universidade Federal de Minas Gerais para fins de obtenção do título de doutor em Engenharia de Produção.

Universidade Federal de Minas Gerais

Escola de Engenharia

Orientador: Maurício Cardoso de Souza, Dr.

Coorientador: Rodney Rezende Saldanha, Dr.

Belo Horizonte,

Abril (2015)

S586o	Silva, Thiago Augusto de Oliveira. Otimização determinística e estocástica de sequenciamento e alocação de recursos em cirurgias [manuscrito] / Thiago Augusto de Oliveira Silva. – 2015. 150 f., enc.: il. Orientador: Maurício Cardoso de Souza. Coorientador: Rodney Rezende Saldanha. Tese (doutorado) - Universidade Federal de Minas Gerais, Escola de Engenharia. Bibliografia: f. 145-150. 1. Engenharia de produção - Teses. 2. Cirurgia - Teses. 3. Programação inteira - Teses. 4. Programação dinâmica - Teses. I. Souza, Maurício Cardoso de. II. Saldanha, Rodney Rezende. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV. Título. CDU: 658.5(043)
-------	---



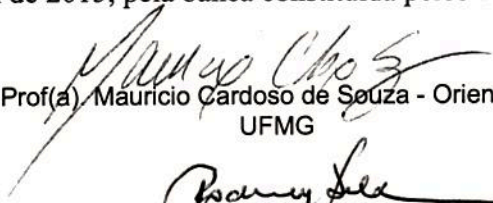
FOLHA DE APROVAÇÃO

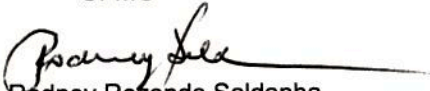
Otimização Determinística e Estocástica do Sequenciamento e Alocação de Recursos em Cirurgias

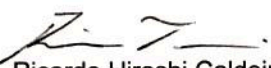
THIAGO AUGUSTO DE OLIVEIRA SILVA

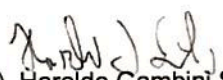
Tese submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em ENGENHARIA DE PRODUÇÃO, como requisito para obtenção do grau de Doutor em ENGENHARIA DE PRODUÇÃO, área de concentração PESQUISA OPERACIONAL E ENGENHARIA DE MANUFATURA, linha de pesquisa Modelos e

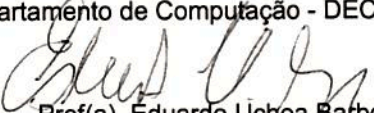
Aprovada em 24 de abril de 2015, pela banca constituída pelos membros:


Prof(a). Mauricio Cardoso de Souza - Orientador
UFMG


Prof(a). Rodney Rezende Saldanha
UFMG


Prof(a). Ricardo Hiroshi Caldeira Takahashi
UFMG


Prof(a). Haroldo Gambini Santos
Departamento de Computação - DECOM/UFOP


Prof(a). Eduardo Uchoa Barboza
Departamento de Engenharia de Produção - UFF


Prof(a). Alexandre Salles da Cunha
UFMG

Belo Horizonte, 24 de abril de 2015.

Aos meus queridos pais, Francisco e Sônia.

À minha doce irmã Viviane.

Às minhas amadas crianças Sofia e João.

À minha companheira Flávia.

Agradecimentos

Gostaria de agradecer aos meus familiares e amigos que tanto me apoiaram durante esta jornada. Aos Professores Dr. Maurício de Souza e Dr. Rodney Saldanha pela dedicação a este projeto. À Escola de Engenharia da UFMG pelos 14 anos de aprendizado.

À direção do Hospital das Clínicas pela abertura e especialmente aos servidores Heloísa Sampaio e Professor Dr. Antônio Ribeiro, diretor da gestão 2010-2014. À direção do Hospital da Mulher e Maternidade Santa Fé pela oportunidade de realização do estudo, em especial, aos servidores Ana Maria Ferreira e Marcia Reis e aos médicos Leo Miller e Clóvis Bacha.

Ao Professor Edmund Burke, Phd. pela recepção na Universidade de Stirling durante a realização do doutorado sanduíche e à CAPES pela oportunidade concedida.

Aos membros da banca pela disponibilidade e pelas contribuições que farão ao trabalho.

Resumo

De uma maneira geral, entende-se por planejamento a atividade de tentar conciliar os recursos disponíveis para atender da melhor maneira possível a demanda por serviços ou produtos em um ambiente de produção. Este trabalho tem como foco o sequenciamento de cirurgias que se refere à parte específica do planejamento responsável pelo agendamento e alocação de recursos para a realização do procedimento operatório em si. O objetivo é descrever e propor solução para o problema de sequenciamento diário de cirurgias. É apresentada uma proposta de modelo determinístico para o sequenciamento diário de cirurgias capaz de lidar com as especificidades no tratamento de recursos humanos especializados. Foram propostos métodos de solução para o problema determinístico e estes foram testados em instâncias baseadas na situação real encontrada em um hospital de grande porte. Um modelo de programação dinâmica estocástica foi proposto para obter uma abordagem que fosse capaz de considerar as incertezas intrínsecas ao processo e de capturar a natureza dinâmica do problema. Métodos de soluções para o problema dinâmico estocástico foram testados e comparados em instâncias baseadas em uma situação real de um hospital de médio porte. Tanto para o caso determinístico, quanto para o caso dinâmico estocástico, os resultados demonstraram que a aplicação das abordagens propostas podem gerar ganhos em situações reais.

Palavras-chaves: Sequenciamento de Cirurgias, Programação Inteira, Programação Dinâmica.

Sumário

	Sumário	7
1	INTRODUÇÃO	12
2	REVISÃO DE LITERATURA	17
2.1	Abordagens determinísticas	23
2.1.1	Métodos Exatos	24
2.1.2	Heurísticas e meta-heurísticas	26
2.2	Abordagens estocásticas	27
2.2.1	Solução via Simulação	28
2.2.2	Solução através de Modelos Estáticos	30
2.2.3	Solução através de Modelos Dinâmicos	34
2.3	Oportunidades no tema	36
3	MODELO DETERMINÍSTICO DE PROGRAMAÇÃO DE CI- RURGIAS	38
3.1	Descrição do problema	38
3.2	Modelo Proposto	40
3.3	Heurísticas baseadas em programação inteira	44
3.3.1	Heurística <i>Relax-and-fix</i>	45
3.3.2	Heurística baseada em PL	47
4	EXPERIMENTOS COMPUTACIONAIS PARA O PDSARC .	48
4.1	Um exemplo real	48
4.2	Características das instâncias de teste	50
4.2.1	Cenário Base	50
4.2.2	Cenários de variação dos recursos especializados	51
4.2.3	Cenários de variação da janela de tempo	52
4.2.4	Cenários combinados de variação da janela de tempo e dos recursos especializados	52

4.3	Parametrização dos experimentos computacionais	53
4.4	Resultados obtidos	53
5	PROGRAMAÇÃO DINÂMICA ESTOCÁSTICA	68
5.1	Elementos de um problema de Programação Dinâmica Es-	
	tocástica	69
5.1.1	Estado e Estágio	69
5.1.2	Ação	70
5.1.3	Política	70
5.1.4	Informações Exógenas	71
5.1.5	Transição	71
5.2	Objetivo	72
5.3	Métodos para solução	73
5.4	Maldição da Dimensão	79
5.5	Aproximação de Programação Dinâmica	81
6	MODELO DINÂMICO ESTOCÁSTICO PARA O SEQUEN-	
	CIAMENTO E ALOCAÇÃO DE RECURSOS EM CIRURGIAS	84
6.1	Modelo proposto	84
6.1.1	Estado	86
6.1.2	Incertezas	89
6.1.3	Decisão	90
6.1.3.1	Transição	91
6.1.3.2	Custos	94
6.1.4	Problema de Programação Dinâmica Estocástica	97
6.2	Abordagens para a solução do problema	100
6.2.1	Discussão e Complexidade	100
6.2.2	Procedimento para geração de uma Política Míope	101
6.2.3	Procedimento para geração de uma Política do tipo <i>Lookahead</i> . . .	103
6.2.4	Algoritmo de aproximação de programação dinâmica para definição	
	de uma política baseada na aproximação do valor da função $J^{(t)}()$.	104
6.2.4.1	Funções indicadoras	106
6.2.4.2	Atualização dos coeficientes da aproximação linear	109

6.2.4.3	<i>Approximate Policy Iteration Algorithm</i>	111
7	EXPERIMENTOS COMPUTACIONAIS PARA O PDEDSARC	115
7.1	Características das instâncias	115
7.2	Parametrização dos experimentos computacionais	118
7.3	Parametrização do PIA	120
7.4	Resultados obtidos	127
7.4.1	Comparação dos métodos de atualização dos coeficientes da regressão	127
7.4.2	Comparação dos algoritmos desenvolvidos para a solução do problema	129
7.5	Considerações sobre a aplicação prática do método PIA . . .	136
8	CONCLUSÕES	138
	Referências	145

Lista de ilustrações

Figura 1	– Gráfico de Gantt para o anestesista $A_2 \in A_e$	50
Figura 2	– Percentual de duração das cirurgias agendadas em relação à duração total das cirurgias	59
Figura 3	– Janela de tempo (T_s) e janela de agendamento ($T_s^{(t),a}$) de uma cirurgia s	87
Figura 4	– Espaço de estados para o <i>status</i> de uma cirurgia	93
Figura 5	– Possíveis transições de uma cirurgia causadas por decisões . . .	100
Figura 6	– Distribuição percentual esperada das chegadas ao longo do dia .	118
Figura 7	– Valores de $\theta_{td}^{(t),\pi_{n-1}}$ e $\theta_{td}^{(t),\pi_n,m}$ por iteração dos períodos $t = 1$ (a) $t = 12$ (b) e $t = 23$ (c) para a aplicação do <i>PIAMQR</i> na instância s18e6r6p9ch	122
Figura 8	– Comparação do custo esperado em $t_0 \times \pi_n$ para diferentes valores M do <i>PIAMQR</i>	123
Figura 9	– Custo esperado em $t_0 \times \pi_n$ para 40 iterações do <i>PIAMQR</i> . . .	124
Figura 10	– Custo esperado em $t_0 \times \pi_n$ para instâncias do grupo s18	125

Figura 11 – Custo esperado em $t_0 \times \pi_n$ para instâncias do grupo s24	126
Figura 12 – Comparação dos valores de custo médio total obtidos pelos métodos MQR e SE	128
Figura 13 – Comparação dos tempos de execução dos métodos MQR e SE .	129
Figura 14 – <i>Scatterplot</i> de V_0^{LKD} x V_0^{PIA} para instâncias do grupo s18 . . .	133
Figura 15 – <i>Scatterplot</i> de V_0^{LKD} x V_0^{PIA} para instâncias do grupo s24 . . .	134

Lista de tabelas

Tabela 1 – Resumos dos dados necessários para a execução do PPLIM . . .	42
Tabela 2 – Instâncias HC, similares ao caso real. UB - limite superior, LB - limite inferior, $g(\%)$ - <i>gap</i> percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.	60
Tabela 3 – Instâncias HB, compatibilidade apenas entre salas adjacentes. UB - limite superior, LB - limite inferior, $g(\%)$ - <i>gap</i> percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.	61
Tabela 4 – Instâncias HA, 40% menos indivíduos no conjunto de recursos especializados. UB - limite superior, LB - limite inferior, $g(\%)$ - <i>gap</i> percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.	62
Tabela 5 – Instâncias HE, com dois tipos de recursos especializados, i.e., $ E = 2$. UB - limite superior, LB - limite inferior, $g(\%)$ - <i>gap</i> percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.	63
Tabela 6 – Instâncias HL, com janela de tempo somente para recursos especializados. UB - limite superior, LB - limite inferior, $g(\%)$ - <i>gap</i> percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.	64
Tabela 7 – Instâncias HS, sem janelas de tempo. UB - limite superior, LB - limite inferior, $g(\%)$ - <i>gap</i> percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.	65

Tabela 8 – Instâncias HLA, com 40% indivíduos em A e com janela de tempo apenas para os recursos especializados. UB - limite superior, LB - limite inferior, $g(\%)$ - <i>gap</i> percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.	66
Tabela 9 – Instâncias HLB, compatibilidade apenas entre salas adjacentes e com janela de tempo somente para os recursos especializados. UB - limite superior, LB - limite inferior, $g(\%)$ - <i>gap</i> percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.	67
Tabela 10 – Visão geral do desempenho dos métodos.	67
Tabela 11 – Maldição da dimensão	80
Tabela 12 – Usos da variável $y_{rsk}^{(t)}$	91
Tabela 13 – Dinâmica de transição em um exemplo ilustrativo	94
Tabela 14 – Possíveis transições de uma cirurgia causadas por decisões . . .	100
Tabela 15 – Visão geral dos procedimentos desenvolvidos	106
Tabela 16 – resultados para as instâncias do grupo s18	131
Tabela 17 – resultados para as instâncias do grupo s24	132
Tabela 18 – Resumo dos resultados	135

1 Introdução

É indiscutível a importância dos serviços de saúde na sociedade em que vivemos. Independente se o sistema é público ou privado, a necessidade de se oferecer serviços com melhor qualidade a custos cada vez menores é uma realidade que emerge como um grande desafio para a engenharia atual.

Cardoen et al. (2010) observaram que hospitais necessitam reduzir os custos e melhorar seu faturamento, mas por outro lado é necessário também maximizar o nível de satisfação do paciente.

Neste contexto, segundo Rais e Viana (2011), diversos países tem encarado o desafio de recorrer à aplicação do planejamento em sistemas de saúde com o objetivo de prover serviços adequados. Os autores destacam ainda que a ferramenta otimização tem sido muito utilizada para solução de problemas de controle de custos na gestão hospitalar e ressaltam que, dentre os principais problemas trabalhados, estão a estimação da demanda, a seleção de localização para unidades hospitalares e o desenvolvimento de unidades de emergência para manuseio eficiente de pacientes.

Inserido no contexto de planejamento hospitalar, o presente trabalho se propõe a discutir o processo de sequenciamento diário de cirurgias, suas respectivas restrições e o tratamento adequado das incertezas inerentes à realização destas atividades.

De uma maneira geral, entende-se por planejamento a atividade de tentar conciliar os recursos disponíveis para atender da melhor maneira possível a demanda por serviços ou produtos em um ambiente de produção. No caso do planejamento de cirurgias, o ambiente em questão é o centro (ou bloco) cirúrgico composto por unidades pré-operatórias, pelas salas de operação e pelas unidades de tratamento pós-operatórias (Unidades de tratamento pós-anestésicas (UTPS) e Unidades de Tratamento Intensivo (UTI)), ver, por exemplo, Guerriero e Guido (2011).

Este trabalho tem como foco a programação (sequenciamento) de cirurgias que se

refere à parte específica do planejamento responsável pelo agendamento e alocação de recursos para a realização do procedimento operatório em si. O sequenciamento de pacientes e o sequenciamento de recursos são os problemas mais explorados na literatura de pesquisa operacional em serviços de saúde.

Dentre estes, o sequenciamento de enfermeiros tem tido atenção especial. Embora de grande aplicabilidade, o problema de sequenciamento de cirurgias tem sido pouco explorado quando comparado ao de enfermeiros (RAIS; VIANA, 2011).

Os recursos envolvidos são tanto recursos físicos, tais como salas e equipamentos específicos; quanto, principalmente, recursos humanos, como cirurgiões, anestesistas, instrumentadores e enfermeiros. O serviço em questão é a realização de cirurgias em pacientes, respeitando suas respectivas necessidades individuais e o(s) critério(s) de decisão(ões) geralmente está(ão) relacionado(s) com indicadores de utilização de recursos ou custos, mas pode(m) variar de uma situação para outra.

Conforme observado por Przasnyski (1986), o custo de construção por pés-quadrados dos blocos cirúrgicos é mais alto nas salas de operação do que em qualquer outra dependência hospitalar. Além disto, diversos autores (GUERRIERO; GUIDO, 2011; MIN; YIH, 2010; CARDOEN et al., 2010; PHAM; KLINKERT, 2008) alegam que os custos operacionais do bloco cirúrgico representam entre 30 a 40% dos custos totais de hospital.

Cardoen et al. (2010) destacam o fato de que as atividades desenvolvidas neste setor possuem grande impacto no desempenho do hospital como um todo. Além disto, cerca de 60 a 70% das entradas em hospitais são devidas às intervenções cirúrgicas e o centro cirúrgico figura como a maior fonte de faturamento hospitalar (CARDOEN et al., 2010; GUERRIERO; GUIDO, 2011; MIN; YIH, 2010).

Os fatos relatados demonstram a importância do setor para a gestão hospitalar. A pesquisa operacional fornece ferramentas capazes de auxiliar na gestão hospitalar indicando meios para um melhor aproveitamento dos recursos. Frente a isto, (RAIS; VIANA, 2011) destacam que o crescente número de artigos publicados recentemente é certamente um indicativo da importância da pesquisa operacional em serviços de saúde.

Nesta tese trabalhou-se com o problema de sequenciamento de cirurgias em um

horizonte de planejamento diário para definir o horário de início do atendimento a um paciente. Os modelos foram construídos para serem capazes de atender situações reais e os dados gerados para os experimentos computacionais foram baseadas nas situações práticas.

Foram desenvolvidos dois modelos, um para uma abordagem determinística e outro para uma abordagem dinâmica estocástica. No caso do problema determinístico, a principal contribuição vem do tratamento de recursos humanos especializados e a alocação simultânea destes. Foram propostos três procedimentos heurísticos baseados na formulação como técnicas de solução.

Para a abordagem estocástica foram comparados dois procedimentos para a solução do problema, um algoritmo de aproximação de programação dinâmica baseado no *Policy Iteration Algorithm* com a utilização de regressão linear múltipla e um algoritmo de referência que consiste na reaplicação de um modelo determinístico a cada estágio do problema (abordagem estática *online*).

Este trabalho está organizado da seguinte maneira: após a apresentação, ainda neste capítulo, dos objetivos gerais, específicos e da metodologia, o capítulo 2 traz uma revisão de literatura acerca do problema de sequenciamento de cirurgias. São apresentadas taxonomias para o problema e são descritos alguns trabalhos relevantes que modelam o problema tanto de maneira determinística quanto de maneira estocástica.

Posteriormente, no capítulo 3, é proposta uma modelagem determinística para um Problema Diário de Sequenciamento e Alocação de Recursos em Cirurgias (PDSARC) encontrado em uma situação real de um hospital de grande porte. Foram incorporados neste problema restrições pouco exploradas na literatura e três diferentes métodos heurísticos baseados na formulação do problema foram desenvolvidos.

No capítulo seguinte, 4, são descritas com base no caso real 144 instâncias para o problema. Este volume de instâncias foi obtido a partir da construção de 3 diferentes replicações para 48 possíveis situações geradas através da combinação de 8 diferentes cenários de parâmetros com 6 diferentes dimensões do problema. Os experimentos computacionais realizados são, então, discutidos.

No capítulo 6, um modelo capaz de lidar com as incertezas e com a natureza dinâmica do PDSARC é proposto e discutido. Duas abordagens para a solução do problema dinâmico estocástico diário de sequenciamento e alocação de recursos em cirurgias (PDEDSARC) são desenvolvidas e descritas. A seguir, capítulo 7, são descritas 16 situações que combinadas com 3 diferentes cenários de chegada geram 48 instâncias para o problema. Os experimentos computacionais, incluindo a parametrização dos métodos e a comparação de resultados, são apresentados.

Por fim, no capítulo 8, as contribuições são discutidas, as conclusões apresentadas e caminhos alternativos para a continuidade da pesquisa são propostos.

Objetivos Gerais

Descrever e propor soluções para o problema de sequenciamento de cirurgias através de abordagens que sejam capazes de considerar as especificidades no tratamento de recursos humanos especializados, as incertezas intrínsecas ao processo e a natureza dinâmica do problema.

Objetivos Específicos

1. Compreender aspectos relevantes do problema de programação de cirurgia.
2. Identificar lacunas na literatura a serem exploradas como problema de pesquisa.
3. Formular e descrever o problema de maneira estática e determinística.
4. Propor e testar abordagens para a solução do problema determinístico.
5. Formular um problema de programação dinâmica estocástica capaz de descrever o problema identificado.
6. Desenvolver e testar abordagens para a solução do problema dinâmico estocástico em tempo computacional viável.

Metodologia

De uma maneira geral, dentro da pesquisa operacional, pode-se dizer que sempre existiu uma distância entre a teoria e as aplicações em gestão de operações. Este cenário, no entanto, vem sendo rapidamente modificado nas últimas décadas e o conhecimento teórico gerado pela pesquisa operacional tem tido grande impacto no desenvolvimento da gestão de operações. A sofisticação das formulações e modelos estudados tem crescido consideravelmente, fazendo com que estes se tornem mais próximos da realidade (BERTRAND; FRANSOO, 2002).

Neste trabalho utiliza-se uma metodologia de pesquisa baseada em modelagem quantitativa para abordar o problema. O objetivo é apresentar modelos que representem adequadamente as variantes determinísticas e estocásticas do PDSARC a partir dos quais serão desenvolvidas estratégias de solução que sejam capazes de auxiliar a tomada de decisão. De acordo com Bertrand e Fransoo (2002) essa abordagem metodológica é classificada como pesquisa axiomática normativa baseada em modelagem quantitativa.

Segundo estes autores uma pesquisa baseada em modelos quantitativos pode ser descrita como a geração racional de conhecimento através de modelos que são capazes de explicar total ou parcialmente o comportamento do processo, além, é claro, da essência da tomada de decisão. Uma pesquisa axiomática formaliza o comportamento do processo gerando conhecimento a respeito da interação entre variáveis e parâmetros. Esta pesquisa pode ser descritiva, quando o objetivo é o entendimento do processo ou normativa, quando visa desenvolver políticas ou estratégias para melhoria da solução corrente.

2 Revisão de Literatura

Margerlien e Martin (1978), em seu trabalho de revisão, remetem aos primeiros estudos no tema programação de cirurgias datados da década de 60 do século passado. Após esta data, muitos trabalhos foram realizados acerca do tema, mas a grande maioria foi publicada após o ano 2000 (CARDOEN et al., 2010).

Encontramos na literatura bons exames relativos ao tema (BLAKE; CARTER, 1997; CARDOEN et al., 2010; CAYIRLI; VERAL, 2003; MARGERLIEN; MARTIN, 1978; GUERRIERO; GUIDO, 2011; RAIS; VIANA, 2011; PRZASNYSKI, 1986). Dentre estes, destacamos dois trabalhos recentes (CARDOEN et al., 2010; GUERRIERO; GUIDO, 2011) pela abrangência e complementaridade. Além da cobertura de alguns trabalhos diferentes, os autores estruturam sua revisão através de classificações distintas.

A classificação mais comum dos problemas de planejamento de cirurgias é a utilizada por Guerriero e Guido (2011), que divide os problemas em níveis de decisão, a saber - estratégico, tático e operacional.

No nível estratégico estaria o problema da definição dos tipos de cirurgias - *Hospital case mix selection problem*, que consiste em definir as prioridades de tipos de cirurgias realizadas no hospital de forma a obter melhor desempenho em algum critério como, por exemplo, balanceamento de *mix*, lucratividade ou aumento da margem de contribuição. As restrições mais comuns são a disponibilidade de recursos, capacidade de atendimento e limitação da quantidade de pacientes. Estes problemas, no geral, são tratados como problemas de programação linear (PPL). Pode-se exemplificar este nível através dos trabalhos de Hughes e Soliman (1985), Robbins e Tuntiwongpiboom (1989) e Blake e Carter (2002).

O nível tático de decisão consiste, de acordo com Guerriero e Guido (2011), na divisão do tempo de sala de operação disponível entre as especialidades (clínicas) médicas. A resposta para este problema define o que é conhecido como Plano Mestre de Cirurgias (PMC). Esta definição pode ser resultado de negociações entre as próprias clínicas, resultado de uma divisão através de técnicas de otimização ou

não existir, como nos casos de sistemas abertos que definiremos a seguir.

Quando resolvido através de modelagem matemática, os objetivos frequentemente são o balanceamento da capacidade entre as clínicas, a maximização da utilização das salas ou a minimização de custos. São consideradas como restrições as disponibilidades de equipamentos e de recursos pós-operatórios em um horizonte de planejamento que envolve de 1 a 4 semanas. Os problemas são formulados através de programação linear inteira mista (PLIM) e têm sido resolvidos tanto de maneira exata como através de métodos heurísticos. Pode-se exemplificar este nível através dos trabalhos de Beliën (2007), Santibáñez et al. (2007), Blake et al. (2002), Marcon et al. (2003) e Beliën e Demeulemeester (2008).

O nível operacional consiste no processo de sequenciamento das cirurgias. O sequenciamento, segundo Margerlien e Martin (1978), pode ser dividido em duas etapas distintas: (i) *Advanced Scheduling* - que determina o agendamento de pacientes para um determinado dia; e (ii) *Allocation Scheduling* - que consiste na resposta integrada para as questões “quais recursos serão alocados para as cirurgias?” e “Qual será a ordem de execução das cirurgias dentre os recursos alocados?” (PHAM; KLINKERT, 2008).

Há ainda, segundo Margerlien e Martin (1978) e Pham e Klinkert (2008), duas abordagens para o problema de *Allocation Scheduling*: sistemas fechados e sistemas abertos. A primeira é baseada em blocos de horários previamente definidos pela solução do problema tático (*Block Systems*, ou sistemas fechados) e a segunda resolve de maneira integrada o problema tático-operacional (*Non Blocking Systems*, ou sistemas abertos).

Os estudos que envolvem abordagens abertas para a solução do problema de sequenciamento foram classificados por Guerriero e Guido (2011) como problemas de níveis mistos, justamente por atuar em mais de um nível de decisão.

Guerriero e Guido (2011) constataram que a maioria dos estudos de sequenciamento foram feitos para uma base diária (*Allocation Scheduling*), após a definição do PMC (*Block Systems*). No entanto, a discussão sobre qual abordagem utilizar, é antiga. Margerlien e Martin (1978) apresentam os resultados de uma pesquisa realizada por Rinde e Blakely (1971) (*apud* (MARGERLIEN; MARTIN, 1978))

onde a grande maioria dos hospitais de Chicago, utilizava sistemas abertos. Já Pham e Klinkert (2008) destacam que, embora os sistemas abertos sejam mais flexíveis, raramente são utilizados devido à preferências dos cirurgiões.

Margerlien e Martin (1978) e Przasnyski (1986) argumentaram que, no entanto, sistemas abertos geram altas taxas de cancelamento de cirurgias devido, principalmente, à necessidade de horas extras causadas pelo elevado número de agendamentos e a disparidade da utilização do bloco cirúrgico entre as clínicas. Przasnyski (1986) destacou a redução da competitividade entre os médicos como um benefício dos sistemas fechados. Apesar disso, os autores reconhecem que sistemas fechados podem gerar subutilização dos blocos devido à irregularidades nas demandas por cirurgias e a dificuldade de agendar cirurgias de última hora.

Do ponto de vista de otimização, resolver o problema aberto (isto é, de maneira integrada) apontará para resultados melhores ou, em um limite, iguais à melhor solução encontrada para o problema decomposto. No entanto, a dificuldade de tratar computacionalmente o problema integrado é muito maior, o que justificaria uma concentração de estudos no uso da abordagem fechada. Ademais, como destacado por Przasnyski (1986), a existência de problemas políticos e comportamentais é comum neste tipo de ambiente e não devemos subestimá-las ao desenvolver técnicas para o uso no planejamento cirúrgico.

Além dos três níveis convencionais, Cardoen et al. (2010) propõem um quarto nível de decisão, que divide os trabalhos em abordagens *offline* e *online*. O que diferencia estes dois tipos de formulações é o instante e a forma da tomada de decisão referente à realização da operação. Nas abordagens *offline* a decisão é tomada uma única vez antes do início do planejamento ao passo que em métodos com o enfoque *online* a decisão é tomada inicialmente e revisada durante a execução do processo. As definições de Vredeveld (2012) e Powell (2011) corroboram com esta.

Devemos diferenciar a classificação entre *online/offline* da divisão entre abordagens estáticas/dinâmicas. Um método estático não considera o impacto de decisões futuras na decisão corrente, enquanto o enfoque dinâmico procura incorporar em sua modelagem o gerenciamento ativo ao longo da ocorrência do processo.

A abordagem *online*, quando é estática, significa resolver de maneira sequencial o problema com novos parâmetros. Segundo Powell (2011) problemas dinâmicos também podem ser *online* ou *offline*. Os problemas *online* remetem a situações onde se tenta identificar uma política ótima enquanto o problema está acontecendo. Em muitos casos, entretanto, é possível estimar os parâmetros da programação dinâmica via simulação antes da execução (abordagem *offline*). Esta abordagem tem como vantagem não correr o risco de tomar decisões ruins durante o processo de aprendizado.

Independente de serem estáticos ou dinâmicos, *offline* ou *online*, todo método que considera a incerteza dos parâmetros em sua formulação é dito estocástico, caso contrário é classificado como determinístico.

A grande maioria dos problemas operacionais tratados na literatura são estáticos (GUERRIERO; GUIDO, 2011). Os poucos tratados de maneira dinâmica, pelo menos os que foram encontrados durante o processo de revisão de literatura desta tese, se restringem a tratar a etapa de *Advanced Scheduling*.

Pham e Klinkert (2008) sugerem utilizar o método de re-sequenciamento para trabalhar os problemas de *Allocation Scheduling* de forma *online*. Esta abordagem foi utilizada, por exemplo, no trabalho de Gröflin e Klinkert (2007). É importante ressaltar que, uma vez que o sequenciamento é feito de forma estática, o resultado desta abordagem é, do ponto de vista dinâmico, subótimo se comparado ao resultado obtido através da abordagem dinâmica. Isto ocorre justamente por não considerar, em sua modelagem, o impacto de decisões futuras ao tomar decisões correntes.

Diversas metodologias de solução foram utilizadas para resolver os problemas operacionais, mas segundo Guerriero e Guido (2011) e Cardoen et al. (2010) os métodos heurísticos são mais frequentes.

De maneira alternativa, Cardoen et al. (2010) estruturaram sua revisão através dos seguintes tópicos:

- Características dos pacientes

As características dos pacientes se referem à diferenciação entre cirurgias com

demandas pré-agendadas, denominadas como cirurgias eletivas, e cirurgias críticas, que não foram agendadas previamente, mas que devem ser realizadas dentro de um intervalo estreito de tempo - não eletivas. As cirurgias não eletivas podem ser divididas em dois grupos dependendo do grau de criticidade: urgências ou emergências (mais críticas).

Segundo Cardoen et al. (2010) a grande maioria dos trabalhos consideram apenas os pacientes eletivos. Muitos destes trabalhos ignoram os efeitos provocados pelas cirurgias não eletivas. Em alguns estudos os autores argumentam haver, no local estudado, salas reservadas para cirurgias não eletivas.

- Medidas de desempenho

Esta característica se refere à medida de desempenho utilizada como critério(s) a ser(em) otimizado(s). O problema em questão é reconhecidamente multi-objetivo e os principais critérios são: Tempo de espera do paciente no sistema, utilização das salas, *makespan*, medidas financeiras de faturamento, lucro ou custo, balanceamento do uso de recursos e minimização do número de cancelamentos.

- Tipo de decisão

A classificação do trabalho de Cardoen et al. (2010) evita estruturar em níveis estratégicos por considerar que há uma confusão na literatura acerca destes níveis. A classificação é feita baseando-se em quem guia a decisão e no tipo de decisão que é tomada. Os autores definem como guia a entidade para quem a decisão está sendo tomada e as dividem em três categorias: clínicas, cirurgiões, tipo de pacientes. O tipo de decisão classifica o trabalho com base no que será definido após a solução do problema, isto é, podem ser classificadas em três níveis: alocação de salas, agendamento de datas e definição de hora ou divisão de capacidade.

- Metodologia de pesquisa

Esta característica está relacionada a duas dimensões, a primeira sobre o tipo de análise empregada, por exemplo, métodos exatos, heurísticos ou análise

de cenários. É uma segunda dimensão que especifica a técnica empregada para solução, por exemplo, programação linear, programação quadrática, programação por metas, programação inteira mista, programação dinâmica, geração de colunas, *branch and price*, simulação de eventos discretos, entre outras.

- Presença de incerteza

Este critério de classificação divide os trabalhos entre os que tratam a existência de incertezas nos parâmetros e os que não tratam. Muitos trabalhos reconhecem a existência de incerteza, principalmente referente à chegada de cirurgias não eletivas. No entanto, segundo os autores, a maioria dos trabalhos utilizam uma abordagem determinística para o problema.

- Aplicações

Segundo os autores, a grande maioria das pesquisas são baseadas em dados reais, mas as ferramentas desenvolvidas raramente passam a ser utilizadas na prática. Neste contexto, os autores identificam como oportunidade a publicação de trabalhos que discutam o processo de implantação destas ferramentas no dia a dia dos hospitais.

Como já comentado, nesta tese tratou-se duas versões do problema de sequenciamento de cirurgias: uma abordagem determinística e uma abordagem dinâmica estocástica. Nos parágrafos a seguir é feita a classificação destas versões com base na estrutura utilizada por (CARDOEN et al., 2010).

A primeira abordagem considera apenas os pacientes eletivos; utiliza como medida de desempenho a ocupação das salas; é guiada para os pacientes eletivos e refere-se à alocação de salas e recursos (materiais e humanos especializados); utiliza métodos heurísticos baseados em programação inteira para encontrar a solução; é definido a partir de uma situação real e testado em instâncias baseadas na realidade do sistema.

A segunda abordagem considera pacientes eletivos e não eletivos; utiliza como critério de desempenho a minimização de custos; é guiada para os pacientes (eletivos e não eletivos) e refere-se à alocação de salas e recursos materiais; utiliza

algoritmos para aproximação de programação dinâmica para encontrar a solução; é definido a partir de uma situação real e testado em instâncias baseadas na realidade do sistema.

Um vez que o foco desta tese é o problema operacional, optou-se por apresentar os trabalhos correlatos através de uma estrutura diferente da classificação usual em níveis de decisão. Nas seções a seguir é utilizado o critério do tratamento de incertezas para diferenciar os trabalhos. Estes foram separados entre os que utilizam abordagem determinística e os que utilizam abordagem estocástica. As categorias foram subdivididas de acordo com a técnica de solução empregada.

2.1 Abordagens determinísticas

Conforme pode ser visto no trabalho de Cardoen et al. (2010), a maior parte dos trabalhos que tratam o sequenciamento de cirurgias de maneira determinística o formulam como um problema de programação linear inteira mista (PLIM). Encontramos basicamente dois tipos de modelos: (i) os modelos baseados na modelagem clássica de problemas de sequenciamento (ver, por exemplo, Pham e Klinkert (2008)), onde o tempo é uma variável de decisão contínua; e (ii) os modelos baseados na indexação temporal de variáveis binárias (ver, por exemplo, Riise e Burke (2011)).

Em ambos casos, a estrutura básica do problema são modelos de sequenciamento já existentes na literatura. No entanto, a grande maioria dos trabalhos desenvolvidos modelam situações específicas que se ajustam a estudos de casos reais. Por este motivo, não há uniformidade em termos de modelagem na literatura, o que dificulta a comparação de modelos e técnicas de solução.

Nos tópicos a seguir, discorreremos sobre alguns importantes trabalhos, destacando suas semelhanças e diferenças. Os trabalhos serão separados por técnica utilizada para a solução dos problemas. A proposta é apenas apresentar alguns exemplos de bons trabalhos na literatura e não de exaurir a discussão acerca de abordagens determinísticas para o problema. Neste ponto, reitera-se que os interessados podem recorrer a revisões recentes e completas disponíveis na literatura. Para revisões mais completas ver Cardoen et al. (2010), Guerriero e Guido (2011) e Blake e

Carter (1997).

2.1.1 Métodos Exatos

Pham e Klinkert (2008) modelaram o problema como um *Multi Mode Blocking Job Shop*, onde uma cirurgia é considerada pelos autores um *Job* com três etapas: pré-operação, operação e pós-operação. A alocação de um conjunto de recursos para a realização de uma etapa da cirurgia é considerada como um modo de processamento. O objetivo do problema modelado é minimizar o *makespan*. O problema é resolvido através da PLIM e testado através de instâncias adaptadas da literatura.

Jebali et al. (2006) abordaram o problema de sequenciamento de cirurgias em duas etapas: primeiramente as cirurgias foram alocadas em salas de operação e em um segundo momento, a ordenação foi realizada com o objetivo de maximizar a utilização das salas considerando as restrições de alocação de recursos. Dentro do método desenvolvido, os autores compararam duas abordagens - possibilidade ou não de troca de sala na segunda etapa. Os algoritmos foram testados em instâncias geradas aleatoriamente.

Cardoen et al. (2009a), trabalharam o problema de forma multi-objetivo e desenvolveram um modelo de PLIM para o sequenciamento diário a partir de um PMC já estabelecido. Foi incorporado ao modelo características específicas do ambulatório de um hospital universitário de Leuven (Bélgica). Os autores utilizaram a plataforma CPLEX para a solução e buscaram focar na geração de soluções viáveis. Foram testadas 224 instâncias construídas a partir dos dados fornecidos pelo hospital. Os autores testaram também procedimentos heurísticos para solução sempre com o foco na geração de uma ou mais soluções viáveis.

Como continuidade deste trabalho, Cardoen et al. (2009b) desenvolveram uma abordagem exata baseada no método de *Branch and Price* para resolver o problema multiobjetivo. Para uma melhor adequação à abordagem, os autores apresentaram uma formulação alternativa à apresentada no trabalho anterior e utilizaram duas estratégias para resolver o subproblema de *pricing*. O método foi testado nas mesmas instâncias utilizadas no trabalho anterior.

Fei et al. (2008) Também desenvolveram uma abordagem baseada no método de

Branch and Price para a solução do problema de sequenciamento de cirurgias eletivas em salas de operação equipadas de maneira multi-funcional com o objetivo de minimizar os custos operacionais. O problema modelado é ligeiramente diferente do trabalhado por Cardoen et al. (2009b), o que dificulta a comparação dos resultados obtidos. Mas assim como no caso citado anteriormente, a técnica de *Branch and Price* se demonstrou eficiente para este tipo de problema.

Marques et al. (2012) propuseram um modelo para o sequenciamento de cirurgias eletivas em um horizonte de tempo semanal com o objetivo de maximizar a ocupação das salas. Foram considerados diferentes níveis de prioridade das cirurgias, janela de tempo e tempo limite para a realização de cirurgias pelos cirurgiões. Para encontrar a solução, os autores decompueram o problema em duas fases. Cada subproblema foi resolvido utilizando o CPLEX com tempo limite para a solução. Após a execução do CPLEX uma heurística de refinamento foi aplicada em cada fase. O problema foi testado em instâncias reais de um hospital universitário de Lisboa.

Ghazalbash et al. (2012) modelaram o problema de sequenciamento de cirurgias em um hospital universitário. O objetivo do problema foi a minimização do tempo ocioso e buscou distribuir as cirurgias de forma a assegurar que os residentes teriam iguais oportunidades de adquirir experiência durante a residência. O problema foi resolvido através uma abordagem de decomposição do PPLIM usando o CPLEX em 30 instâncias reais de um hospital universitário do Irã.

Jeang e Chiang (2012) propuseram um modelo de programação quadrática com o objetivo de minimizar o desvio entre a disponibilidade de tempo das salas e o tempo efetivamente gastos com cirurgias. O modelo proposto foi testado em uma instância real de um hospital universitário e foi resolvido utilizando o software GAMS.

Meskens et al. (2013) propuseram uma abordagem de programação por restrições para lidar com disponibilidade de recursos, preferência do corpo técnico e afinidade entre seus membros. O modelo apresentado era multi-objetivo e visava minimizar o *makespan*, as horas extras e também maximizar a afinidade entre os membros da equipe cirúrgica.

2.1.2 Heurísticas e meta-heurísticas

Hsu et al. (2003) formularam o problema como uma nova variante do *process shop scheduling* em dois estágios e sem espera. O problema foi resolvido através da implementação de uma Busca Tabu que se demonstrou bastante eficiente para encontrar boas soluções em instâncias reais de um centro cirúrgico de um hospital universitário.

Dentro um PMC já definido, Fei et al. (2010) trabalharam o problema de sequenciamento diário de cirurgias com dois objetivos - maximização da utilização e minimização de custos. Os autores decompuseram o problema em duas fases. Primeiramente a cirurgia era agendada para uma determinada data (*advanced scheduling*) e, em um segundo momento, era realizada a alocação de recursos e ordenação (*allocation scheduling*). Para ambas fases foram desenvolvidos procedimentos heurísticos. A primeira fase foi modelada como problema de particionamento e resolvido através de uma heurística baseada no método de geração de colunas. A segunda fase foi modelada como um *flow shop* híbrido de dois estágios e solucionado através de algoritmos genéticos. Ao comparar os resultados obtidos pela abordagem com os dados de um hospital belga, os autores constataram que os métodos eram capazes de gerar melhores resultados em todas as dimensões colocadas como objetivo.

Roland et al. (2010) propuseram um modelo para lidar com diferentes aspectos de recursos humanos e materiais (renováveis e não renováveis). As preferências dos cirurgiões eram expressas através de suas disponibilidades e alguns recursos poderiam cobrir mais de uma sala de cirurgia, uma vez que não participavam durante toda operação.

Os autores desenvolveram uma heurística baseada em algoritmos genéticos para solucionar o problema e compararam o desempenho desta com a solução do PPLIM obtida pelo CPLEX em quatro instâncias reais replicadas em dois cenários de restrição do usos das salas.

Riise e Burke (2011) trabalharam o problema integrado de sequenciamento e alocação de recursos em cirurgias (referenciado como problema de admissão de cirurgias). Os autores destacam como uma contribuição a adaptação do problema

para o caso onde se faz necessário definir um sequenciamento para os cirurgias, além das salas. Como método de solução, os autores construíram procedimentos heurísticos de busca baseados na Busca Local Iterada e no *Variable Neighborhood Descend*.

Vijayakumar et al. (2013) buscaram maximizar a soma ponderada do número de cirurgias realizadas em um horizonte diário e semanal. Os autores modelaram o problema como *multi-dimensional dual bin-packing problem*. Os autores desenvolveram um procedimento heurístico baseado no *First Fit Decreasing algorithm* e testaram em 22 instâncias geradas aleatoriamente.

2.2 Abordagens estocásticas

Comparado ao volume de trabalhos que tratam o sequenciamento de cirurgias de forma determinística, segundo (CARDOEN et al., 2010) a literatura apresenta poucos trabalhos que discutem sobre a incerteza neste tipo de problema. No entanto, assim como salientado por Guerriero e Guido (2011), a chave para o sequenciamento (e planejamento) de cirurgias é a coordenação de múltiplas atividades em um ambiente de incertezas.

Possíveis fontes de incerteza nestes ambientes são, entre outras, as seguintes:

- Incerteza na duração das cirurgias - O tempo de duração de duas cirurgias, mesmo que do mesmo tipo, pode variar muito de paciente para paciente devido, por exemplo, a complicações ao longo do processo.
- Incerteza quanto ao número e tipo de cirurgias - Além das cirurgias eletivas, existem também as cirurgias não eletivas que aparecem de última hora e possuem urgência em sua realização. Não é possível prever quantas cirurgias não eletivas surgirão e muito menos o tipo destas cirurgias. Segundo Lamiri et al. (2009), dentre uma grande variedade de estudos propostos na literatura, apenas alguns poucos estudos consideram a situação em que os recursos devam ser divididos entre cirurgias eletivas e não eletivas.

- Incerteza quanto à realização de cirurgias (*No-Show*) - Embora agendadas previamente, a realização de cirurgias eletivas também pode ser fonte de incerteza uma vez que a cirurgia pode não ocorrer por indisponibilidade do cirurgião ou por complicações no estado de saúde do paciente.
- Incerteza quanto à disponibilidade de recursos - Devido ao intenso compartilhamento de recursos de diversas naturezas entre as cirurgias e o carácter aleatório da duração destas, alguns recursos essenciais alocados para uma determinada cirurgia podem não estar disponíveis para o seu início.

2.2.1 Solução via Simulação

A grande maioria dos trabalhos presentes na literatura que consideram algum tipo de incerteza em sua modelagem são analisados através da técnica de simulação (CARDOEN et al., 2010; GUERRIERO; GUIDO, 2011). Abaixo descreveremos alguns trabalhos que utilizaram esta metodologia para a solução do problema.

Kim e Horowitz (2002), complementando seu trabalho anteriormente desenvolvido (KIM et al., 1999) analisaram via Simulação de Eventos Discretos (SED) uma política de reservas de leitos em UTI para cirurgias eletivas. O objetivo foi avaliar o impacto desta política no cancelamento de cirurgias e no planejamento do processo de admissão de pacientes que possuem necessidade deste recursos (*Advanced Scheduling*). A pesquisa foi realizada com base em 6 meses de dados de um hospital universitário de Hong Kong.

Os autores consideraram a existência de incerteza nos processos de chegada de pacientes que demandam tratamento na UTI e também na duração das estadias nos leitos. Para os pacientes provenientes de cirurgias eletivas, o processo de admissão foi controlado através de políticas de alocação / reserva de sala. Os autores concluíram que a melhor política testada, em termos de redução no cancelamento de cirurgias, combinava o sistema de alocação flexível de leitos com o sistema de quotas variáveis.

Dexter e Traub (2002) compararam regras de inserção de novas cirurgias eletivas dentro de uma sequência já estabelecida de cirurgias via SED. No caso estudado, a admissão das cirurgias é realizada em comum acordo com o cirurgião e o paciente.

Em seguida os outros recursos são ajustados para atender a demanda de forma a maximizar a utilização das salas. As regras comparadas foram a data mais cedo de início e a data mais tarde de início. Além disto, os autores buscaram analisar o impacto do erro de estimativa do tempo de cirurgia na utilização das salas. Por fim, concluiu-se que os resultados dependem muito da qualidade das estimativas, mas que, no geral, o critério data mais cedo de início provê melhores resultados.

Sciomachen et al. (2005) estudaram o bloco cirúrgico de um grande hospital em Genova (Itália) através do uso de SED. O objetivo era comparar o desempenho dos principais indicadores do centro cirúrgico entre o processo de agendamento corrente e o agendamento obtido através do método proposto. Este método consiste na combinação de um plano mestre de cirurgia, obtido via PPLI com o uso de critérios heurísticos de sequenciamento, a saber - Maior Tempo de Espera (LWT, sigla em inglês), Menor Tempo de Processamento (SPT, sigla em inglês), Maior tempo de processamento (LPT, sigla em inglês).

Os autores consideraram tanto pacientes eletivos, quanto não eletivos e assumiram incerteza no processo de chegada e de duração das cirurgias. Como resultados, verificou-se melhor desempenho do modelo de programação inteira sobre o modo atual de agendamento e constatou-se que o critério LPT provê melhor utilização das salas, enquanto o critério SPT gera sequências com menores usos de hora-extra para o ambiente estudado.

Em uma análise semelhante à de Kim e Horowitz (2002), através da comparação de cenários e utilizando SED, Wullink et al. (2007) compararam duas políticas para lidar com cirurgias não eletivas: (i) o uso de uma sala dedicada a elas e (ii) reserva de tempo em salas compartilhadas com as cirurgias eletivas. Os autores consideraram a incerteza na duração das cirurgias, além da probabilidade de chegada de uma cirurgia não eletiva. Concluiu-se que a política de reserva de tempos em salas compartilhadas sempre se sobrepunha ao uso de salas dedicadas em termos de desempenho.

2.2.2 Solução através de Modelos Estáticos

Além dos trabalhos baseados na abordagem metodológica de comparação de cenários via simulação, apresentados no tópico anterior, outros autores consideraram a incerteza em seus trabalhos ao tentar abordar o problema sob o ponto de vista da otimização estática.

Apesar dos trabalhos apresentados neste tópico utilizarem simulação como parte de seus procedimentos de solução, estes se diferem dos apresentados como trabalhos que buscaram solução via simulação devido à preocupação metodológica de se encontrar a solução ótima, ou uma boa solução, através de seus algoritmos. Alguns destes trabalhos serão apresentados nos próximos parágrafos.

Denton et al. (2007) estudaram o problema de *Allocation Scheduling* sob incerteza da duração das atividades. O objetivo é determinar a sequência diária de cirurgias que minimize o custo total esperado referente à soma dos tempos de espera dos pacientes, ociosidade das salas e uso de horas extras.

O problema foi representado por um modelo de programação estocástica (PE) em dois estágios. Os autores utilizaram técnicas de programação estocástica para encontrar solução para o modelo proposto. Foram testados também, três procedimentos heurísticos. Os resultados foram comparados utilizando dados gerados com base na situação real encontrada em um hospital universitário de Nova Iorque.

Três critérios míopes construtivos foram implementados para determinar uma solução inicial a saber - ordem crescente de duração média, ordem crescente de variabilidade da duração e ordem crescente do coeficiente de variação. As heurísticas implementadas consistem na combinação dos critérios construtivos com uma heurística de refinamento que troca a ordem de duas cirurgias escolhidas aleatoriamente. Como resultado, foi percebido pelos autores que o critério de alocação de cirurgias por ordem crescente de variabilidade da duração apresentava, para este conjunto de dados, um melhor desempenho.

A abordagem de Hans et al. (2008) se difere um pouco das demais uma vez que o foco é alocar, além das cirurgias, tempo reservado para cirurgias urgentes para as salas de operação durante os dias. O objetivo é maximizar a utilização e minimizar o risco de cancelamento e/ou necessidade de uso de horas extras. O estudo

foi realizado em um hospital universitário na Holanda que disponibilizou dados históricos para os experimentos.

A proposta dos autores foi o desenvolvimento de heurísticas construtivas refinadas por buscas locais que exploravam a informação estatística da duração das cirurgias. Como resultado, foi possível demonstrar que a abordagem favorecia a utilização total das salas quando comparada com a abordagem anteriormente empregada.

Lamiri et al. (2008) estudaram o problema de *Advanced Scheduling* em um horizonte finito e uma quantidade pré-definida de pacientes eletivos. O problema determina quais pacientes eletivos serão alocados em cada dia e considera que as salas são igualmente equipadas. Os tempos de operação são considerados como variáveis aleatórias, assim como a chegada de pacientes de emergência (não eletivos) que devem ser operados mesmo que seja necessário o cancelamento de cirurgias eletivas ou o uso de horas-extras.

O objetivo do problema é a minimização do valor esperado da soma dos custos de espera das cirurgias eletivas e dos custos totais provenientes do uso de horas-extras. O problema foi modelado como um PPE.

Os autores provaram que o problema em questão é fortemente NP-difícil e aproximaram o problema através da técnica de aproximação por média amostral (SAA, sigla em inglês). Foi demonstrado que o resultado obtido via SAA, neste caso, converge quase certamente para o verdadeiro ótimo. Como resultado, foi possível demonstrar que o tratamento determinístico do problema aumenta, em média, os custos totais em cerca de 4%.

Lamiri et al. (2009) continuaram estudando os métodos de solução para o problema definido por Lamiri et al. (2008). Inicialmente os autores provaram que a convergência da técnica SAA ocorre de maneira exponencialmente rápida e mostraram que esta taxa de convergência pode ser maximizada caso seja combinada com a técnica de redução de variância *Common Random Numbers* (CRN).

Além disto, Lamiri et al. (2009) propuseram e compararam procedimentos heurísticos alternativos à solução ótima obtida pela técnica SAA. Um procedimento genérico heurístico construtivo míope baseado na ordenação das cirurgias eletivas foi construído. A partir deste procedimento, foram propostas três heurísticas para

definição de soluções iniciais que diferiam entre si pelo critério de ordenação das cirurgias. Combinados a estes procedimentos, foram utilizadas três heurísticas de refinamento para melhorar a solução inicial.

Foram também implementadas duas meta-heurísticas - *Simulated Annealing* e Busca Tabu. Os algoritmos foram testados em instâncias geradas aleatoriamente e a Busca Tabu demonstrou-se o melhor método dentre as heurísticas sempre possuindo valores próximos ao método “exato”SAA (e até mesmo melhores quando o tempo de execução não foi suficiente para a convergência) mas com tempo de processamento significativamente menor.

Os autores apontaram como sugestão para trabalhos futuros a combinação de técnicas de relaxação Lagrangeana e/ou geração de colunas dentro do SAA para melhorar o desempenho computacional da técnica. Do ponto de vista de modelagem do problema, Lamiri et al. (2008) sugeriram considerar o problema de *Allocation Scheduling* e outras fontes de incerteza.

Denton et al. (2010) modelaram o problema de alocação de cirurgias em salas como uma variação do *Bin-packing problem*. Os autores consideraram como variáveis de decisão a abertura ou não de uma sala e a alocação de cirurgias entre as salas abertas. O objetivo era minimizar a soma do custo fixo de abertura de salas com esperança do custo de hora extra, em uma dada sala. Foram consideradas apenas cirurgias eletivas com incerteza em sua duração. Embora seja um problema de alocação de recursos (sala), a sequência das cirurgias não foi definida.

Os autores desenvolveram um modelo de PE em dois estágios, para o qual foram estabelecidos limites para a solução. Além disto, foi desenvolvido um modelo robusto e uma heurística para o problema. A heurística implementada com base no modelo robusto se demonstrou extremamente eficiente para encontrar a solução ótima.

Min e Yih (2010) consideraram o problema de sequenciamento de cirurgias eletivas mediante a incerteza do tempo e de capacidade de recursos pós-operatórios (UTI). O objetivo consistiu em minimizar o custo total esperado composto pelo custo de espera dos pacientes e o custo de horas-extra. Os autores trabalharam o problema de *Advanced Scheduling* sobre um PMC conhecido a priori.

Foi considerada pelos autores a incerteza na duração das cirurgias e no tempo de utilização de um leito no UTI. Além disto, a capacidade inicial de cada bloco do PMC foi considerada incerta uma vez que os autores descontaram um tempo aleatório que representava a alocação de capacidade de um bloco à execução de cirurgias não eletivas.

O problema foi modelado como um PPE e utilizaram a técnica SAA para encontrar as soluções. Por fim, os autores recomendaram que o problema fosse modelado através da programação dinâmica estocástica (PDE) em trabalhos futuros. O objetivo seria poder considerar, por exemplo, a aleatoriedade nos processos de chegada, isto é, considerar a existência de cirurgias não eletivas.

Begen e Queyranne (2011) estudaram o problema estático de sequenciamento estocástico de cirurgias eletivas em tempo discreto e estabeleceram propriedades de convexidade para a função objetivo. Os autores modelaram o problema como sequenciamento de uma única máquina sem tempo de preparação e com função de custo do tipo L-convexa. Para este caso, foi provado que é possível encontrar a solução ótima em tempo polinomial. Os autores estenderam a aplicabilidade deste modelo para inclusão da incerteza de chegada (cirurgias não eletivas) e não comparecimento (*No-shows*).

Saremi et al. (2013) combinaram as técnicas de PLI, Busca Tabu e SED, para resolver de maneira heurística a definição da sequência das cirurgias (*Appointment Scheduling*) de maneira estática. O problema modelado considera a disponibilidade de recursos como restrição, mas não faz alocação destes entre as cirurgias. Apenas a incerteza na duração das cirurgias eletivas foi considerada na construção do modelo e o objetivo era reduzir o tempo total de execução (*makespan*).

Três procedimentos heurísticos foram desenvolvidos através da combinação das técnicas já citadas e o desempenho destes foram comparados aos de regras heurísticas mais simples em conjuntos de dados gerados aleatoriamente para o problema. Os autores concluíram que os procedimentos que utilizam programação inteira possuíam melhor desempenho especialmente nos casos onde há restrição de janela de tempo para a realização de cirurgias. Os procedimentos também foram testados em instâncias reais de um bloco cirúrgico em um hospital do Canadá.

Bruni et al. (2015) propuseram um modelo de programação estocástica para resolver o problema de *Advanced Scheduling* em um horizonte semanal com múltiplos blocos cirúrgicos.

Os autores criaram um procedimento heurístico com duas variantes, uma baseada em programação inteira e outra baseada em uma busca iterativa. As abordagens foram testadas em 72 instâncias geradas aleatoriamente e comparadas com a solução do problema determinístico correlato tendo apresentado bons resultados.

2.2.3 Solução através de Modelos Dinâmicos

Embora a elaboração de modelos estocásticos para o sequenciamento dinâmico de cirurgias tenha aparecido primeiramente em trabalhos da década de 70 do século passado, os primeiros trabalhos que implementaram técnicas de solução só apareceram recentemente.

Kolesar (1970) propôs o uso do processo de decisão de Markov (PDM) em hospitais, ele formulou modelos simplificados para a tomada de decisão em *Advanced Scheduling* de pacientes em hospitais. Já Collart e Haurie (1976) propuseram um modelo semi-markoviano para pacientes eletivos e de não eletivos e formulou um problema de controle estocástico ótimo, no entanto, nenhum dos dois trabalhos, apresentou técnicas de solução.

Mais recentemente, Gerchak et al. (1996) criaram um modelo capaz de determinar a quantidade ótima de pacientes eletivos frente à demanda de emergência em um centro cirúrgico. Eles mostraram que a política ótima não era apenas limitar o agendamento de pacientes para um determinado dia, mas agendar proporcionalmente ao tempo de espera.

O trabalho de Patrick et al. (2008) está focado no desenvolvimento de um método mais sistematizado para agendamento de pacientes com diferentes níveis de prioridade frente às incertezas de chegada. Os autores observaram que as decisões de agendamento impactam diretamente a disponibilidade para demandas futuras. Se pacientes com baixa prioridade forem agendados muito cedo, então haverá capacidade insuficiente para pacientes futuros com alta prioridade. No caso de agendamento tardio, há o risco da subutilização de recursos. O problema foi tratado

para o uso de um recurso hospitalar genérico, não sendo focado, especificamente, no agendamento de cirurgias.

Os autores formularam o problema de *Advanced Scheduling* como um PDM de horizonte infinito descontado e o transformaram em um problema de programação linear (PPL), que caso fosse possível de se resolver, retornaria a resposta ótima para o MDP. No entanto, devido ao tamanho do espaço de estados, nem o PDM nem o PPL são tratáveis. Os autores usaram a técnica de aproximação de programação dinâmica (APD) para produzir uma aproximação por programação linear tratável, isto é, um problema linear aproximado (PLA) que fosse tratável.

O PLA foi resolvido através do método de geração de colunas o que possibilitou a construção de políticas ótimas aproximadas (POA) que não necessitam ser resolvidas através do PLA. Os resultados foram testados através de simulação entre as políticas aproximadas e a estratégia atual validando, assim, o bom desempenho da POA.

Liu et al. (2010) também trabalharam o problema de agendamento de pacientes, o trabalho se difere do desenvolvido por Patrick et al. (2008) por não considerar prioridades entre os pacientes e por considerar a possibilidade de cancelamento ou não comparecimento do paciente no dia da agendado para a realização do procedimento médico. A modelagem do problema considera o fato de que o cancelamento ou não comparecimento está relacionado com o tempo de espera entre o contato do paciente e o agendamento do procedimento.

O problema foi modelado como um PDM. Foram consideradas as incertezas das demandas diárias por agendamento e dos cancelamentos ou não comparecimentos. As variáveis de decisão são as datas de agendamento dos paciente. O sequenciamento de cirurgias dentro de uma data é assumido como a ordem dos agendamentos.

Para solucionar o problema os autores propuseram e compararam políticas heurísticas via simulação. Através da heurística de melhoramento de políticas (PIH, sigla em inglês, ver Tijms (1994)), os autores partiram de uma boa solução inicial (estática) e desenvolveram quatro políticas heurísticas dinâmicas (HDP, sigla em inglês). Os resultados foram comparados com três critérios heurísticos que reproduzem métodos comumente utilizados na prática. Os resultados da comparação

via simulação foram capazes de demonstrar que os procedimentos desenvolvidos possuem desempenho claramente melhor do que os métodos utilizados na prática.

Zonderland et al. (2010) trabalharam o problema de reserva ótima de tempo em salas para cirurgias não-eletivas semi-urgentes (que poderiam ser realizadas em um prazo de até uma semana) inicialmente como um problema de filas e posteriormente como um PDM. A ação decidida a cada período consiste na reserva de tempo adicional para este tipo de cirurgia por meio do cancelamento de cirurgias eletivas. Uma vez que o estado e as ações possuem dimensões pequenas e a função de custo esperado futuro é conhecida, os autores utilizaram o *policy iteration algorithm* para encontrar a política ótima para o problema definido.

O método desenvolvido foi testado em dados reais do departamento de neurocirurgia situado em um hospital universitário da Holanda. Os autores indicaram como trabalho futuro a extensão do modelo para o sequenciamento individual de tais cirurgias dentro deste processo dinâmico para aproveitar melhor o tempo alocado para cada tipo de cirurgia.

Ceschia e Schaerf (2014) abordaram o problema de *Advanced Scheduling* em um horizonte de planejamento variável considerando a incerteza na chegada de pacientes não eletivos e na duração da estadia dos pacientes.

Os autores propuseram uma heurística baseada em busca local e testaram em 30 intâncias geradas aleatoriamente para o problema.

2.3 Oportunidades no tema

Rais e Viana (2011) relataram a importância do uso de métodos de pesquisa operacional dentro de sistemas de saúde e enfatizaram que há campo tanto para modelagem de novos problemas, quanto para o desenvolvimento de novos métodos de solução.

Uma questão pouco explorada na literatura é a integração entre diversos recursos e a consideração de características individuais destes recursos, tais como não uniformidade de equipamentos em salas, ou preferências por tipo de procedimentos pelos anestesistas, ou disponibilidade, por exemplo.

Há uma clara lacuna na literatura referente ao desenvolvimento de modelos dinâmicos para o problema de *Allocation Scheduling*. A dificuldade apontada com frequência é o desenvolvimento de métodos eficientes de solução para modelos estocásticos e, especialmente, para os dinâmicos.

O foco desta tese é o problema operacional, especificamente a etapa de *Allocation Scheduling*, de sequenciamento e alocação de recursos para cirurgia em um sistema aberto, isto é que não possui um PMC. A modelagem via programação inteira visa explorar as especificidades do uso de recursos humanos especializados em contexto multibloco. A modelagem via programação dinâmica estocástica busca explorar a lacuna identificada na literatura para o tratamento do problema de *Allocation Scheduling*.

3 Modelo determinístico de programação de cirurgias

Este capítulo descreve o modelo de programação linear inteira mista para o problema diário de sequenciamento e alocação de recursos em cirurgias eletivas (PD-SARC). O problema tratado neste capítulo está situado no nível operacional. Precisamente, a modelagem se propõe a abordar as decisões classificadas como *Allocation Scheduling*.

3.1 Descrição do problema

O desenvolvimento deste modelo foi guiado por uma situação real encontrada na prática de um hospital de grande porte da cidade de Belo Horizonte. O problema tem como características particulares a possibilidade do emprego simultâneo de recursos humanos especializados e a individualização destes recursos. No escopo deste trabalho, considera-se um recurso especializado aquele que possui especialidades próprias. Porém, mesmo dentro de um mesmo grupo, não possuem características homogêneas. Os anestesiologistas são um exemplo destes recursos já que estes profissionais são identificados por suas especialidades, experiência, treinamento, preferências pessoais e disponibilidade.

As cirurgias eletivas são conhecidas a priori e podem começar em qualquer período da janela de tempo estabelecida para seu início. As decisões a serem tomadas envolvem fazer ou não a cirurgia, o horário de início, a sala de realização e os recursos alocados a ela. Para tomar a decisão é necessário conhecer o conjunto de cirurgias, suas características e o conjunto de recursos utilizados, assim como suas respectivas características e disponibilidades.

Cada cirurgião possui um conjunto de pacientes, portanto, cada operação deve ser realizada por um cirurgião específico, previamente definido. Por se tratar de um sistema aberto, isto é, por não haver um PMC, a alocação de cirurgias para as

salas de operação faz parte da decisão. É necessário definir o horário de início que deve respeitar a janela de tempo tanto do cirurgião quanto do caso cirúrgico (paciente/cirurgia).

Janelas de tempo podem capturar prioridades em termos de horário de início. Por exemplo, crianças devem ser operadas no início da manhã (RIISE; BURKE, 2011) ou cirurgias para paciente de fora da cidade devem ser agendadas para o período da tarde (VIJAYAKUMAR et al., 2013). Há também preferências da equipe médica que podem ser vistas como necessárias para melhoria do processo cirúrgico (ROLAND et al., 2010).

O problema é restringido por limitações na disponibilidade de recursos a serem compartilhados. Um procedimento pode requerer recursos renováveis ou não renováveis, características já observadas em outros trabalhos (ROLAND et al., 2010).

Recursos renováveis (humanos ou materiais) são aqueles alocados para o procedimento estritamente durante sua duração como, por exemplo, enfermeiros, instrumentadores ou equipamentos médicos. Por outro lado, recursos não renováveis são aqueles que, uma vez alocados à um procedimento, não podem mais ser utilizados durante o horizonte de planejamento, por exemplo, vagas na unidade de tratamento intensivo (UTI).

Cada cirurgia está associada a subconjunto de recursos especializados capazes de realizar a intervenção, recursos estes que são renováveis. Conforme dito anteriormente, recursos especializados são recursos individualizados, isto é, que possuem características próprias. Observe que a individualização não os tornam necessariamente insubstituíveis. Os anestesistas são exemplos de recursos especializados. Os cirurgiões são outro exemplo de recurso especializado, no entanto, no modelo proposto, estes são definidos a priori.

Na situação encontrada na prática existe a possibilidade do emprego simultâneo de recursos especializados em diferentes intervenções cirúrgicas. Esta simultaneidade depende da compatibilidade das salas onde as operações estão ocorrendo. Isto leva a uma outra dimensão que o processo decisório deve considerar para poder usufruir do compartilhamento dos recursos especializados.

O objetivo é maximizar o uso das salas. Recentes exames de literatura apontam

este objetivo como um dos frequentemente empregados em outros trabalhos (CARDOEN et al., 2010; GUERRIERO; GUIDO, 2011). Embora este trabalho não seja um estudo de caso, visitas técnicas foram feitas para entendimento das nuances dos processos de agendamento de cirurgias (ver capítulo 4) em hospitais e pudemos constatar que, nos locais visitados, este objetivo é aderente com as necessidades práticas.

3.2 Modelo Proposto

A seguir os elementos da modelagem são descritos e a Tabela 1 fornece uma visão geral dos parâmetros necessários para descrever o modelo. O horizonte de tempo de decisão (dia) foi repartido em pequenas unidades discretas pertencentes ao conjunto de períodos T . O conjunto de cirurgiões é representado por C e $T_c \subseteq T$ é o subconjunto de T em que um cirurgião pode iniciar uma cirurgia, isto é, sua janela de tempo.

O conjunto de cirurgias é representado por S e $S_c \subseteq S$ é o subconjunto de cirurgias que devem ser realizadas pelo cirurgião $c \in C$. O subconjunto $T_s \subseteq T_c$ representa a janela de tempo de início da cirurgia $s \in S_c$.

O conjunto R é o conjunto de todas as salas de operação. Uma vez que as salas não são homogêneas, o subconjunto $R_s \subseteq R$ contém as salas onde o procedimento $s \in S$ pode ocorrer. As salas podem diferir entre si, por exemplo, por apresentarem tamanhos diferentes ou serem equipadas com diferentes recursos. Uma sala é utilizada por uma cirurgia s durante d_s períodos de tempo, que inclui o período de limpeza da sala. Os demais recursos são alocados apenas durante o procedimento em si e são liberados ao término deste. O parâmetro $\bar{d}_s$ contabiliza apenas a duração do procedimento.

Os recursos são agrupados em dois conjuntos K_1 e K_2 referentes aos renováveis e aos não renováveis, respectivamente. Para um dado $k \in K_1$ (resp. $k \in K_2$), g_{sk}^1 (resp. g_{sk}^2) indica se o recurso renovável (não renovável) k é necessário para realizar a cirurgia s .

A quantidade disponível do recurso não renovável $k \in K_2$ é q_k^2 unidades. Um

recurso renovável $k \in K_1$ possui uma disponibilidade de q_k^1 unidades móveis e também pode já estar instalado em uma sala do subconjunto $R_k \subseteq R$.

Uma cirurgia que necessita do recurso k designada para operar em uma sala $r \in R_k$ não consome nenhuma unidade do recurso móvel. Por outro lado, o recurso instalado em uma sala do conjunto R_k é fixo e não pode ser transportado para outra sala.

O conjunto de tipos de recursos especializados é representado por E , e $E_s \subseteq E$ representa o subconjunto necessário para a realização da cirurgia $s \in S$. Como exemplo ilustrativo, suponha que o conjunto $E = \{\alpha, \beta\}$, ou seja o hospital conta com os tipo de recursos α e β . Seja $\dot{s}, \hat{s} \in S$ duas cirurgias para as quais $E_{\dot{s}} = \{\alpha\}$ e $E_{\hat{s}} = \{\alpha, \beta\}$. Isto quer dizer que a cirurgia $\dot{s}$ necessita de um recurso especializado do tipo α e a cirurgia $\hat{s}$ necessita de um recurso de cada tipo disponível.

Dado $e \in E$, A_e é o conjunto de indivíduos que compõem o quadro da especialidade $e \in E$. Por exemplo, $A_\alpha = \{\alpha_1, \alpha_2, \alpha_3\}$ e $A_\beta = \{\beta_1, \beta_2\}$. Para uma determinada cirurgia $s \in S$, dado os elementos $e \in E_s$, nem todos os indivíduos que compõem o quadro A_e estão aptos para realizar a cirurgia s , apenas os elementos do subconjunto $A_e^s \in A_e$. No exemplo acima, suponha que $A_\alpha^{\dot{s}} = \{\alpha_1, \alpha_2\}$, $A_\alpha^{\hat{s}} = \{\alpha_2, \alpha_3\}$, $A_\beta^{\hat{s}} = \{\beta_1\}$.

Um indivíduo $a \in A_e$ pode realizar q_a cirurgias simultaneamente, desde que a distância entre as sala r e $\bar{r}$ onde as cirurgias estiverem ocorrendo permita. O parâmetro $b_{r\bar{r}}$ indica a compatibilidade das salas r e $\bar{r}$, isto é, tem o valor igual a 1 caso seja possível utilizar simultaneamente um recurso alocado para cirurgias que ocorrem nestas salas. O problema tratado é, portanto, multibloco, i.e., considera o compartilhamento de recursos entre diferentes instalações.

Cada indivíduo $a \in A_e, e \in E$ possui sua própria janela de tempo para início de uma cirurgia. Portanto, o subconjunto $T_e^a \subseteq T$ contém os períodos em que o indivíduo a pode iniciar uma cirurgia.

A Tabela 1 apresenta, de maneira resumida, os conjuntos e parâmetros utilizados no modelo. Foram definidos dois conjuntos de variáveis binárias. No primeiro conjunto, a variável y_{st}^r , possui valor 1 se a cirurgia $s \in S$ for alocada para a sala $r \in R_s$ e iniciar no período $t \in T_s$, e 0 caso contrário. No segundo, w_{st}^a , possui

Conjuntos	Descrição
S	cirurgias
C	cirurgiões
$S_c \subseteq S$	cirurgias alocadas para o cirurgião $c \in C$
R	salas de operação
$R_s \subseteq R$	salas onde $s \in S$ pode ser realizada
T	períodos
$T_c \subseteq T$	janela de tempo do cirurgião $c \in C$ para início de uma cirurgia
$T_s \subseteq T_c$	janela de tempo para início de uma cirurgia $s \in S_c$
K_1	recursos renováveis
$R_k \subseteq R$	salas que possuem o recursos renovável $k \in K_1$ instalado
K_2	recursos não renováveis
E	tipos de recursos renováveis especializados
$E_s \subseteq E$	tipos de recursos renováveis especializados necessários a realização de $s \in S$
A_e	indivíduos que compõem o quadro de recursos do tipo $e \in E$
$A_e^s \subseteq A_e$	indivíduos do recurso do tipo $e \in E_s$ aptos a realizar a cirurgia $s \in S$
$T_e^a \subseteq T$	janela de tempo do indivíduo $a \in A_e$ para início de uma cirurgia
Parâmetros	Descrição
d_s	duração da cirurgia $s \in S$ com tempo de limpeza
$\bar{d}_s$	duração apenas do período de intervenção da cirurgia $s \in S$
g_{sk}^1	indicador da necessidade de uso do recurso $k \in K_1$ para $s \in S$
q_k^1	unidades móveis de $k \in K_1$ disponíveis
g_{sk}^2	quantidade de recurso $k \in K_2$ necessária para realizar $s \in S$
q_k^2	disponibilidade diária de $k \in K_2$
q_a	número de cirurgias que o indivíduo $a \in A_e (e \in E_s)$ pode realizar simultaneamente
$b_{r\bar{r}}$	indicador de compatibilidade das salas $r, \bar{r} \in R$

Tabela 1 – Resumos dos dados necessários para a execução do PPLIM

valor 1 se para a cirurgia $s \in S$ for alocado o indivíduo $a \in A_e^s$ do tipo de recursos especializados $e \in E_s$ com início no período $t \in T_e^a \cap T_s$, e 0 caso contrário. O modelo pode ser escrito através das expressões (3.1) - (3.12).

$$\max \sum_{s \in S} \sum_{r \in R_s} \sum_{t \in T_s} d_s y_{st}^r \quad (3.1)$$

$$\sum_{r \in R_s} \sum_{t \in T_s} y_{st}^r \leq 1 \quad \forall s \in S \quad (3.2)$$

$$\sum_{s \in S_c} \sum_{r \in R_s} \sum_{l=t-\bar{d}_s+1}^t y_{sl}^r \leq 1 \quad \forall c \in C, \forall t \in T_c \quad (3.3)$$

$$\sum_{s \in S | r \in R_s} \sum_{l=t-\bar{d}_s+1}^t y_{sl}^r \leq 1 \quad \forall r \in R, \forall t \in T \quad (3.4)$$

$$\sum_{s \in S} \sum_{r \in R_s \setminus R_k} \sum_{l=t-\bar{d}_s+1}^t g_{sk}^1 y_{sl}^r \leq q_k^1 \quad \forall k \in K_1, \forall t \in T \quad (3.5)$$

$$\sum_{s \in S} \sum_{r \in R_s} \sum_{t \in T_s} g_{sk}^2 y_{st}^r \leq q_k^2 \quad \forall k \in K_2 \quad (3.6)$$

$$\sum_{r \in R_s} y_{st}^r - \sum_{a \in A_e^s | t \in T_e^a} w_{st}^a \leq 0 \quad \forall s \in S, \forall e \in E_s, \forall t \in T_s \quad (3.7)$$

$$\sum_{t \in T_s} \sum_{a \in A_e^s | t \in T_e^a} w_{st}^a \leq 1 \quad \forall s \in S, \forall e \in E_s \quad (3.8)$$

$$\sum_{s \in S | e \in E_s} \sum_{l=t-\bar{d}_s+1}^t w_{sl}^a \leq q_a \quad \forall e \in E, \forall a \in A_e, \forall t \in T_e^a \quad (3.9)$$

$$y_{st}^r + w_{st}^a + \sum_{l=t-\bar{d}_s+1}^t (y_{sl}^{\bar{r}} + w_{sl}^a) \leq 3 \quad \forall s, \bar{s} \in S, \forall e \in E_s \cap E_{\bar{s}}, \forall a \in A_e^s \cap A_{\bar{e}}^{\bar{s}},$$

$$\forall r \in R_s, \forall \bar{r} \in R_{\bar{s}}, b_{r\bar{r}} = 0, \forall t \in T_e^a \cap T_s \quad (3.10)$$

$$y_{st}^r \in \{0, 1\} \quad \forall s \in S, \forall r \in R_s, \forall t \in T_s \quad (3.11)$$

$$w_{st}^a \in \{0, 1\} \quad \forall e \in E, \forall a \in A_e, \forall s \in S,$$

$$\forall r \in R_s, \forall t \in T_e^a \cap T_s \quad (3.12)$$

O objetivo do problema é maximizar a utilização das salas de operação representada, neste problema, como a soma do tempo total gasto com procedimentos cirúrgicos.

As restrições (3.2) asseguram que uma cirurgia iniciará uma única vez. O principal problema quando se maximiza o uso das salas em uma base diária é que uma cirurgia pode ser postergada indefinidamente sem nunca ser agendada. No entanto, ao usar um modelo na base diária, pode ser útil substituir a desigualdade por uma igualdade e introduzir uma variável de folga na restrição (3.2). Isto possibilitaria impor o agendamento de uma determinada cirurgia ao forçar o valor da variável de folga a ser zero.

A realização simultânea de cirurgias é proibida pelas restrições (3.3) e (3.4). A primeira é relativa ao cirurgião, enquanto a segunda é relativa ao uso da salas. Note que um cirurgião pode iniciar outra cirurgia tão logo a duração do procedimento cirúrgico terminar ($\bar{d}_s$), enquanto a sala só é liberada após d_s períodos.

As restrições (3.5) e (3.6) são restrições de capacidade para os recursos renováveis e não renováveis, respectivamente. Sempre que uma cirurgia s é agendada para iniciar no período $t \in T_s$, um indivíduo $a \in A_e^s$ de cada tipo de recurso especializado $e \in E_s$ deve ser alocado (restrições 3.8). A alocação de recursos renováveis especializados deve respeitar a janela de tempo T_e^a do indivíduo $a \in A_e$, $e \in E_s$ (restrições 3.9).

As restrições (3.9) limitam a quantidade de cirurgias que o indivíduo a pode realizar simultaneamente. Como mencionado anteriormente, a distância entre duas salas interfere na possibilidade de um indivíduo realizar duas cirurgias simultâneas. Se o $b_{r\bar{r}} = 0$, a restrição (3.10) proíbe que um indivíduo seja designado para realizar as cirurgias s e $\bar{s}$ simultaneamente nas salas r e $\bar{r}$, respectivamente. Por fim, as restrições (3.11) e (3.12) definem os domínios das variáveis.

3.3 Heurísticas baseadas em programação inteira

Em muitas aplicações práticas, formulações de programação inteira têm sido utilizadas para gerar boas, mas não necessariamente ótimas soluções como assinalado, por exemplo, por Ball (2011) e Wolsey (1998).

Em testes preliminares, a solução do modelo apresentado obtida com o CPLEX demonstrou-se dispendiosa computacionalmente. Uma vez que os testes realizados também demonstraram que a relaxação linear do modelo proposto pode gerar bons limites superiores para a solução ótima, foram desenvolvidos heurísticas baseadas em programação inteira para obter boas soluções a custos computacionais menores.

Por esta razão, as relaxações foram utilizadas para propor heurísticas baseadas em programação inteira. Estes procedimentos foram capazes de fornecer soluções de boa qualidade. A partir da relaxação, como primeira etapa, as heurísticas selecionam os candidatos à alocação de cirurgias para salas e períodos. Na segunda

etapa, um modelo reduzido é utilizado para obter a viabilidade da solução.

3.3.1 Heurística *Relax-and-fix*

Uma heurística do tipo *Relax-and-fix*, na qual apenas o conjunto de variáveis w é relaxado, foi proposta. No primeiro passo o problema (3.1) a (3.11) é resolvido através da relaxação linear de (3.12) para produzir um limite superior e definir um conjunto de designações de cirurgias candidatas.

Seja $\bar{y}$ o vetor ótimo (inteiro) encontrado a partir da relaxação de w . O conjunto Υ é composto pelas cirurgias designadas pelo modelo relaxado, isto é, contém cada cirurgia s para qual exista uma sala η_s e um período τ_s para os quais $\bar{y}_{s\tau_s}^{\eta_s} = 1$.

As designações de cirurgias encontradas na primeira etapa são viáveis em termos de sobreposição de recursos renováveis e não renováveis, isto é, respeitam as restrições (3.2) a (3.6). Porém, não pode haver alocação viável de recursos especializados que atendam as restrições (3.7) a (3.10).

Na segunda etapa, um modelo reduzido é resolvido para obter viabilidade. A ideia geral é tentar alocar um recurso especializado para cada uma das cirurgias $s \in \Upsilon$, ou cancelar a designação caso não seja possível realizar a alocação.

O conjunto de variáveis binárias w_s^a recebem valor igual a 1 se para $s \in \Upsilon$ for designado um indivíduo $a \in A_e^s$ do conjunto de recursos especializados $e \in E_s$, e 0 caso contrário. A variável w teve sua dimensão reduzida em relação ao modelo original uma vez que a cirurgia s tem início já definido para τ_s caso esta seja viabilizada. A variável contínua z_s naturalmente recebe 0 ou 1, 1 se a cirurgia $s \in \Upsilon$ não for realizada e 0 caso contrário.

O conjunto Λ_s , $s \in \Upsilon$, representa um conjunto das cirurgias com potencial conflito com s , ou seja, é formado por toda cirurgia $s' \in \Upsilon - \{s\}$ que está agendada para $\tau_{s'}$ e não terá sido finalizada em τ_s . Portanto, $\Lambda_s = \{s' \in \Upsilon - \{s\} | \tau_s - d_{s'} + 1 \leq \tau_{s'} \leq \tau_s\}$. O conjunto $\bar{\Lambda}_s$ é definido de maneira análoga, exceto que somente a duração do procedimento cirúrgico $\bar{d}_s$ é considerada. O modelo reduzido é descrito por (3.13) - (3.18).

$$\min \sum_{s \in \Upsilon} d_s z_s \quad (3.13)$$

$$w_s^a + \sum_{s' \in \bar{\Lambda}_s | a \in A_{s'}^e, \tau_{s'} \in T_e^a} w_{s'}^a \leq q_a \quad \forall s \in \Upsilon, \forall e \in E_s, \forall a \in A_s^e, \tau_s \in T_e^a \quad (3.14)$$

$$w_s^a + w_{s'}^a \leq 1 \quad \forall s \in \Upsilon, \forall s' \in \bar{\Lambda}_s, b_{\eta_s \eta_{s'}} = 0, \forall e \in E_s \cap E_{s'}, \\ \forall a \in A_s^e \cap A_{s'}^e, \tau_s \wedge \tau_{s'} \in T_e^a, q_a > 1 \quad (3.15)$$

$$\sum_{a \in A_s^e | \tau_s \in T_e^a} w_s^a + z_s = 1 \quad \forall s \in \Upsilon, \forall e \in E_s \quad (3.16)$$

$$0 \leq z_s \leq 1 \quad \forall s \in \Upsilon \quad (3.17)$$

$$w_s^a \in \{0, 1\} \quad \forall s \in \Upsilon, \forall e \in E_s, \forall a \in A_s^e, \tau_s \in T_e^a \quad (3.18)$$

A função objetivo (3.13) minimiza a perda de ocupação gerada pela não confirmação das designações das cirurgias. O conjunto de restrições (3.14) limita o número de cirurgias que um indivíduo de um determinado tipo de recursos especializado $e \in E$ pode realizar simultaneamente.

O conjunto de restrições (3.15) evita a situação onde um indivíduo de um determinado tipo de recurso especializado $e \in E$ é alocado para realizar duas cirurgias em salas não compatíveis.

O conjunto de restrições (3.16) garante que uma designação candidata só será confirmada se houver os recursos especializados alocados à cirurgia. O domínio das variáveis é definido por (3.17) e (3.18). Uma vez que w é binária, (3.16) garante a integralidade de z .

Ao final da segunda etapa a heurística retorna uma solução viável com uma ocupação de $\sum_{s \in \Upsilon} d_s(1 - \bar{z}_s)$, onde $\bar{z}$ é o vetor z ótimo para a formulação (3.13) - (3.18).

A formulação (3.1) - (3.11) não precisa ser resolvida à otimalidade para se obter boas soluções viáveis para o problema original. Foi identificado que é possível se obter soluções de boa qualidade resolvendo a primeira etapa com um gap de otimalidade, por exemplo, de 5%.

3.3.2 Heurística baseada em PL

Tentou-se obter uma lista maior de designações de cirurgias candidatas através da relaxação de (3.1) - (3.12). Seja $\bar{y}$ a solução ótima (não necessariamente inteira) para as variáveis y da relaxação linear. Dado uma cirurgia $s \in S$, seja η_s a sala e τ_s o período para o qual o maior valor de $\bar{y}$ é obtido, i.e., $(\eta_s, \tau_s) = \arg \max_{(r,t) \in R_s \times T_s} \{\bar{y}_{st}^r\}$.

O conjunto $\Upsilon = \{s \in S \mid \bar{y}_{s\tau_s}^{\eta_s} \geq \rho\}$ agora possui cada cirurgia s para a qual a variável de designação para η_s e τ_s é maior ou igual ao valor de ρ . O conjunto Γ_r , $r \in R$, contém as cirurgias candidatas alocadas para a sala r , isto é, $\Gamma_r = \{s \in \Upsilon \mid \eta_s = r\}$. Adicionalmente às restrições (3.7) - (3.10), o conjunto de designações de cirurgias candidatas não é, necessariamente, viável em relação às restrições (3.3) e (3.4), e às restrições de recursos (3.5) e (3.6).

Na segunda etapa, o modelo reduzido dado por (3.13) e (3.18) acrescido das restrições (3.19) - (3.22) é resolvido.

$$z_s + z_{s'} \geq 1 \quad \forall c \in C, \forall s \in S_c \cap \Upsilon, \forall s' \in S_c \cap \bar{\Lambda}_s \quad (3.19)$$

$$z_s + z_{s'} \geq 1 \quad \forall r \in R, \forall s \in \Gamma_r, \forall s' \in \Gamma_r \cap \Lambda_s \quad (3.20)$$

$$g_{sk}^1(1 - z_s) + \sum_{s' \in \bar{\Lambda}_s \mid \eta_{s'} \notin R_k} g_{s'k}^1(1 - z_{s'}) \leq q_k^1 \quad \forall k \in K_1, \forall s \in \Upsilon, \eta_s \notin R_k \quad (3.21)$$

$$\sum_{s \in \Upsilon} g_{sk}^2(1 - z_s) \leq q_k^2 \quad \forall k \in K_2 \quad (3.22)$$

As restrições (3.19) e (3.20) obrigam uma cirurgia a não ser realizada para todo par de duas cirurgias conflitantes que possuem o mesmo cirurgião ou que foram alocadas para a mesma sala. A primeira impede que um cirurgião inicie uma cirurgia sem ter finalizado a anterior. A segunda garante que uma sala não será alocada simultaneamente para duas cirurgias. As restrições (3.21) e (3.22) impedem que a capacidade dos recursos renováveis e não renováveis, respectivamente, sejam excedidas.

4 Experimentos computacionais para o PDSARC

Este capítulo apresenta os experimentos computacionais realizados para testar os procedimentos desenvolvidos para a solução do PDSARC proposto no capítulo 3. Inicialmente, a solução de um exemplo real baseado no sequenciamento de cirurgias do hospital que motivou o estudo é apresentada. Posteriormente, a comparação do desempenho das heurísticas desenvolvidas em 144 instâncias geradas com base no caso real é apresentada.

4.1 Um exemplo real

Esta seção descreve um exemplo real para aplicação do modelo para o PDSARC. Atualmente, no referido hospital são realizadas cerca de 30 procedimentos cirúrgicos por dia. A ocupação média dos blocos é de 45 a 50%.

Boa parte dos recursos e operações necessários para o bom funcionamento do bloco são externos a este (e.g. raio-x, exames de sangue, vagas no CTI, alocação de equipes de enfermagem). O planejamento é realizado para um dia e definido com 48 horas de antecedência para inibir qualquer início de preparação pré-operatória desnecessária. As demandas pelo uso da sala são enviadas com 72 horas de antecedência.

A assessoria de planejamento decide sobre o local, horário e recursos alocados para um determinado procedimento. Atualmente o hospital aloca as cirurgias às salas sem a realização de um PMC. Entre duas cirurgias deve haver um tempo de desinfecção. A duração da preparação da sala (*setup*) é de, em média, 30 minutos.

O caso estudado possui 64 cirurgias que significa, na prática, dois dias de sequenciamento. A soma da duração de todas as cirurgias é 367 períodos e a atividade de limpeza da sala demanda 1 período após cada cirurgia. Cada período é relativo a 30 minutos e o expediente (T) é formado por 24 desses períodos.

No total, 43 cirurgiões estão envolvidos com este conjunto de cirurgias. Há dois recursos renováveis, o microscópio e o intensificador de imagens, com 3 e 5 unidades móveis, respectivamente. O intensificador de imagens também possui uma unidade fixa em uma das salas. O recurso não renovável considerado é a unidade de tratamento intensivo com disponibilidade de 7 leitos.

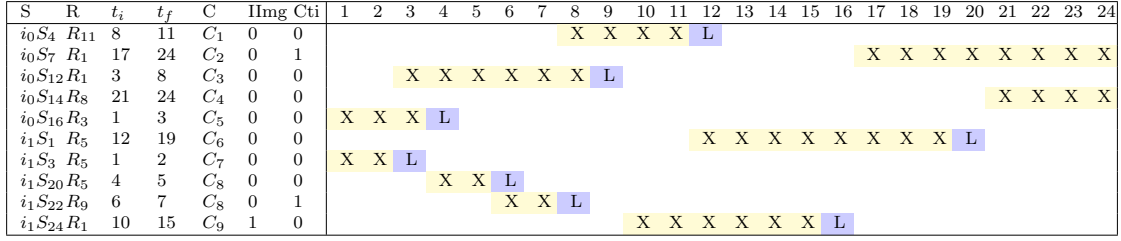
No hospital há 14 salas divididas em dois blocos para cirurgias eletivas. O bloco principal possui 11 salas e o ambulatório possui 3 salas. A maioria das cirurgias, no caso 57 do total de 64, deve ser realizada no bloco principal. Cirurgias cardíacas devem ser realizadas em uma sala específica e a mesma restrição se aplica para as cirurgias neurológicas que devem ser realizadas em uma outra sala específica. As salas específicas não são exclusivas para cirurgias cardíacas e/ou neurológicas podendo ser utilizadas por todos os outros tipos de cirurgias.

Os anestesistas foram considerados como recursos especializados. Dos 12 indivíduos existentes, 6 estão disponíveis durante todo o dia. Dentre os restantes, 2 trabalham somente de manhã e outros 4 apenas na parte da tarde.

O conjunto de anestesistas que estão aptos à realização de uma determinada cirurgia foi definido de acordo com as preferências e as habilidades destes profissionais. Eles podem realizar até duas cirurgias simultaneamente, desde que ocorram no mesmo bloco cirúrgico.

O modelo de programação inteira foi capaz de resolver o caso relatado em 416 segundos utilizando um computador pessoal com processador Intel Core i3, 2.6 GHz com 4 GB de RAM.

A Figura 1 apresenta parte do sequenciamento obtido para o anestesista A_2 . Na figura, é identificada a cirurgia (S), a sala para qual esta foi agendada (R), o período de início (ti) e de término (tf), o cirurgião (c), a necessidade do recurso renovável intensificador de imagens (IImagem) e do recurso não renovável CTI (CTI). Para as cirurgias realizadas por este anestesista não houve necessidade do uso de recurso renovável microscópio. São marcados com 'X' os períodos em que a cirurgia ocorrerá e com 'L' o período de limpeza subsequente à realização das cirurgias. Percebe-se o uso intenso deste recurso especializado e que o mesmo realiza duas cirurgias simultâneas em quase todo o expediente.

Figura 1 – Gráfico de Gantt para o anestesista $A_2 \in A_e$

Na solução ótima do problema, 49 cirurgias foram agendadas utilizando 292 períodos. Isto equivale a uma utilização de 87%. Na prática, o agendamento de cirurgias é feito de maneira manual e as taxas de utilização giram em torno de 45% a 50%. A meta de utilização do bloco é de 60%.

4.2 Características das instâncias de teste

Um total de 144 instâncias foram geradas. O número de salas e a quantidade de recursos foram utilizados de acordo com o observado na prática, ou seja, 14 salas, 2 recursos renováveis e 1 recurso não renovável. O expediente é dividido em 24 períodos de 30 minutos, $|T| = 24$. Foram considerados 6 cirurgiões. Os outros parâmetros do modelo foram gerados aleatoriamente mas seguindo as dimensões observadas na prática e detalhadas no exemplo real da seção 4.1.

O número de cirurgias variou de 40 a 90 com incremento de 10 cirurgias. Foram definidos 8 cenários, descritos nas subseções a seguir. Para cada uma das 48 combinações (número de cirurgias $\times$ cenários) foram geradas 3 instâncias, totalizando 144.

4.2.1 Cenário Base

Os valores dos parâmetros utilizados por este grupo de instâncias, denominado HC, foram gerados a partir de orientações fornecidas pelo corpo técnico responsável pela programação de cirurgias do hospital que motivou o estudo.

As salas são divididas em dois blocos, um com 3 salas e outro com 11 salas.

Dentro de cada bloco, é permitido o uso simultâneo de recursos especializados entre duas salas pelas cirurgias, i.e., $b_{r\bar{r}} = 1$ se r e $\bar{r}$ são do mesmo bloco e 0 no caso contrário. Há apenas um tipo de recurso humano especializado, os anestesiistas, com 10 indivíduos, isto é, $|E| = 1$ e $|A| = 10$.

Para cada instância, entre 30% a 50% dos indivíduos pertencentes a A são capazes de realizar duas cirurgias simultaneamente ($q_a = 2$), os demais podem realizar apenas uma cirurgia por vez ($q_a = 1$).

As durações das cirurgias foram geradas a partir de uma distribuição uniforme discreta com no mínimo 1 e no máximo 11 períodos. As cirurgias foram alocadas aos cirurgiões de maneira aleatória. Cada cirurgia possui uma probabilidade de 90% de necessitar do recurso humano especializado. Em alguns casos, cirurgias com anestesia local não necessitam do acompanhamento de um anestesiista. A composição dos conjuntos A^s foi feita aleatoriamente, assim como a definição do conjunto R_s .

A necessidade de uso de um recurso renovável é definida aleatoriamente utilizando uma probabilidade entre 10% e 20% para cada cirurgia. O mesmo ocorre para os recursos não renováveis, mas com probabilidade de uso distribuída entre 0% e 15%. A disponibilidade de recursos foi definida a partir de uma distribuição uniforme discreta com mínimo de 1 e máximo de 6 para os renováveis e mínimo de 3 e máximo de 6 para o não renovável.

Uma sala possui probabilidade de 2% de possuir um recurso renovável instalado. A quantidade necessária de um recurso para realizar uma cirurgia é de 1 quando a cirurgia necessita do recurso. Isto é, g_{sk}^1 e g_s^2 possuem valores iguais a 0 ou a 1. Os cirurgiões e os recursos especializados, podem estar disponíveis no período da manhã (períodos de 1 a 12), da tarde (períodos de 13 a 24) ou durante todo o expediente (períodos de 1 a 24).

4.2.2 Cenários de variação dos recursos especializados

Foram gerados três diferentes cenários de variação das características dos recursos humanos especializados. Esses conjuntos de instâncias são denominados HB, HA e HE foram criados a partir do conjunto HC. Foram feitas alterações na compo-

tibilidade entre salas, na cardinalidade de A , e na cardinalidade do conjunto E , respectivamente.

Nas instâncias do tipo HB, as salas passaram a pertencer a um único bloco, mas o parâmetro $b_{r\bar{r}}$ assume o valor 1 apenas para salas adjacentes. Nas instâncias do tipo HA o número de indivíduos do conjunto A foi reduzido em 40%, passando de 10 para 6. No grupo HE foram considerados dois tipos de recursos especializados, o tipo 1 já presente nas instâncias HC e o tipo 2 com $|A| = 7$.

Para cada instância, entre 30% e 50% dos indivíduos pertencentes ao conjunto A_2 são capazes de realizar 2 cirurgias simultaneamente e cada cirurgia possui uma probabilidade de 90% de necessitar deste tipo de recurso. Assim como nas instâncias do tipo HC, a composição dos conjuntos A_2^s foi definida aleatoriamente. Uma cirurgia pode necessitar dos dois tipos de recursos (81%), apenas um (18,9%), ou nenhum deles (0,1%).

4.2.3 Cenários de variação da janela de tempo

Dois grupos de cenários de variação das janelas de tempo dos recursos foram gerados a partir de HC. No grupo HL, as janelas de tempo foram relaxadas para os cirurgiões e para as cirurgias, isto é, $T_c = T_s = T$. As janelas de tempo dos recursos especializados permaneceram inalteradas, $T_e^a \subseteq T$. No grupo das instâncias HS todas as janelas de tempo foram relaxadas.

4.2.4 Cenários combinados de variação da janela de tempo e dos recursos especializados

A partir da combinação do conjunto de instâncias do tipo HL com os conjuntos HB e HA, foram gerados os conjuntos HLA e HLB que tornam mais crítico o uso de recursos especializados.

Nestes cenários as janelas de tempo dos cirurgiões e das cirurgias foram relaxadas, portanto, além de apenas os recursos especializados estarem sujeitos a restrições de tempo, as instâncias tipo HLA possuem 40% menos indivíduos e as instâncias tipo HLB permitem o uso simultâneo destes recursos apenas em salas adjacentes.

4.3 Parametrização dos experimentos computacionais

Todos os experimentos foram executados em um computador Intel Xeon X5690 @ 3.47GHz com 24 processadores e 132 GB de RAM com sistema operacional Linux. Todas as instâncias foram executadas através do modelo de PLI, da heurística baseada em programação linear e da heurística *Relax-and-Fix*. A heurística *Relax-and-Fix* foi executada resolvendo a primeira etapa na otimalidade e também resolvendo a primeira etapa para um gap de 5% (denominada *relax-and-fix 5%*).

Todos os métodos propostos foram codificados em AMPL e resolvidos utilizando CPLEX versão 12.4. Para executar o CPLEX, utilizamos os parâmetros `presolve = 1` que habilita a execução do pré-solver, `parallelmode = 1`, para garantir uma busca paralela em modo determinístico, e `timelimit = 21600` segundos que limita o tempo de execução. A diretriz `clocktype = 1` foi utilizada e significa que a contagem de tempo considera o tempo gasto por todos os processadores, isto é, a soma do tempo gasto em cada processador para resolver o problema. Por exemplo, caso o tempo decorrido desde o início da execução do modelo tenha sido 1 hora e durante todo este tempo foram utilizados 2 processadores, então o tempo informado para solução do modelo será de 2 horas.

As restrições (3.10) foram tratadas com o parâmetro `lazy = 1` quando o modelo inteiro foi executado e também na primeira etapa das heurísticas *relax-and-fix*. O parâmetro `mipemphasis` foi definido como 1 para a primeira etapa da heurística *relax-and-fix 5%* para priorizar a busca por soluções viáveis sobre a prova de otimalidade.

4.4 Resultados obtidos

Os resultados obtidos são apresentados nas tabelas (2) a (9). Para melhor apresentação do texto, as tabelas se encontram ao final do capítulo.

Todas as tabelas possuem a mesma estrutura. As duas primeiras colunas representam a identificação e o número de cirurgias em cada instância ($|S|$). As três próximas colunas apresentam os resultados para a execução do modelo de PLI ((3.1) - (3.12)). São apresentados o valor da função objetivo do maior limite infe-

rior (LB), isto é, o número de períodos de ocupação das salas da melhor solução encontrada, o *gap* relativo de otimalidade desta solução ($gap(\%)$), retornado pelo CPLEX, e o tempo de execução (t), em segundos, para encontrar a referida solução. O tempo informado é a soma dos valores `_solve_user_time` e `_ampl_user_time` gerados pelo AMPL.

Entre a 6^a e a 11^a colunas são apresentados os resultados para a heurística baseada em programação linear (Seção 3.3.2). São apresentados o limite superior (UB) obtido na primeira etapa, o tempo necessário para se obter o limite superior (t) `_solve_user_time` retornado pelo AMPL, o limite inferior obtido após a etapa de viabilização (LB) (solução do modelo reduzido (3.13) - (3.22)), tempo adicional necessário para a viabilização (t) `_solve_user_time` retornado pelo AMPL, o *gap* percentual em relação ao limite superior obtido na primeira etapa, isto é, $\frac{UB-LB}{LB}$ ($gap(\%)$) e o tempo total de execução (t) é representado em segundos demandados para encontrar a solução, isto é, a soma dos valores `_solve_user_time` (das duas etapas) e `_ampl_user_time` gerados pelo AMPL.

Entre a 12^a e a 17^a colunas são apresentados os resultados obtidos pela heurística *Relax-and-Fix* e as colunas são apresentadas de maneira análoga às colunas de 6 a 11. O limite superior obtido é dado pela execução do modelo de PLIM com a relaxação das restrições (3.12). E a solução viável é obtida pela execução do modelo reduzido (3.13) - (3.18).

Nas ultimas três colunas são apresentados os resultados para a heurística *Relax-and-Fix* 5%. São apresentados o limite inferior obtido após a etapa de viabilização (LB) (3.13) - (3.22), o *gap* percentual relativo ao limite superior obtido na primeira etapa ($g(\%)$) - que não é necessariamente igual ao limite superior obtido pela primeira etapa da heurística *Relax-and-Fix*), e o tempo total de execução (t) em segundos para encontrar a solução, isto é, a soma dos valores `_solve_user_time` (das duas etapas) e `_ampl_user_time` gerados pelo AMPL.

Observe que os valores das colunas $g(\%)$ são calculados exclusivamente com a informação sobre o respectivo método e não sobre o melhor limite superior obtido. Desta forma, cada coluna $g(\%)$ representa a garantia, a posteriori, de desempenho de cada método. O *gap* médio em relação à solução ótima ou à melhor solução

conhecida é apresentado para cada método na Tabela 10.

Na heurística baseada em programação linear, o limiar ρ pode ser utilizado como parâmetro de *tunning* e define o tamanho do problema que será resolvido na segunda etapa do algoritmo. Resultados preliminares mostraram que o modelo reduzido (3.13) - (3.22) é resolvido rapidamente. Por esse motivo, foi utilizado $\rho = 0.01$ para considerar quase todas as cirurgias com variável y ativa na relaxação linear do modelo inteiro (3.1) - (3.12) na construção do conjunto Υ . Desta forma, uma quantidade maior de possibilidades são fornecidas para a segunda etapa da heurística sem resultar em aumento significativo do custo computacional.

A Tabela 2 contém os resultados obtidos para as instâncias do grupo HC. Foi possível resolver na otimalidade todas as instâncias deste grupo, mas o tempo computacional necessário foi consideravelmente alto para algumas das instâncias. Pela coluna UB da heurística baseada em programação linear, pode-se perceber que o modelo gera bons limites de relaxação, isto é, a formulação é forte. No entanto, os valores da relaxação linear não foram úteis para encontrar bons limites inferiores na segunda etapa do algoritmo. A heurística baseada em programação linear não foi capaz de gerar bons resultados, gerando *gaps* maiores que 10% para a maioria das instâncias.

As heurísticas *Relax-and-Fix* foram bem eficientes para encontrar soluções próximas da ótima neste conjunto de instâncias. Os tempos computacionais foram moderados mas, exceto pela instância HC5, em todos os casos foi possível encontrar a solução ótima. No caso da instância HC5, embora o UB obtido pela primeira etapa do método tenha sido equivalente ao valor ótimo, não foi possível viabilizar a alocação de recursos para o conjunto de cirurgias selecionado. O modelo reduzido (3.13) - (3.18) rodou extremamente rápido. A *Relax-and-Fix* 5% se demonstrou uma boa alternativa para conseguir bons resultados de maneira rápida. O método encontrou 5 soluções ótimas e não gerou *gaps* maiores do que 5.4% para um tempo aproximado de um minuto.

As Tabelas 3, 4 e 5 apresentam os resultados para as instâncias HB, HA e HE, respectivamente. O parâmetro de compatibilidade entre salas e a cardinalidade de E tiveram impacto no desempenho do modelo de PLI para resolver o problema.

Para quatro instâncias o modelo atingiu o tempo limite sem garantir a otimalidade das soluções encontradas.

A heurística *Relax-and-Fix* teve bom desempenho nestes grupos de instâncias. Foi encontrada a solução ótima para todas as instâncias de HB e de HE, com exceção de HE5. Todos os tempos computacionais foram razoáveis, exceto para a instância HE8 que demorou mais de 10 minutos. Da mesma forma, a *Relax-and-Fix* 5% também teve bom desempenho apresentando *gaps* pequenos e tempos computacionais razoáveis. A solução ótima foi encontrada em 11 instâncias de HB e 8 de HE, sendo que em 5 dessas instâncias a otimalidade é provada pelo próprio método *Relax-and-Fix* 5%.

Por outro lado, o conjunto de instâncias HA foi mais fácil de ser resolvido pelo modelo de PLI devido ao número reduzido de variáveis. No entanto, as heurísticas *Relax-and-Fix* apresentaram *gaps* maiores do que para os conjuntos discutidos anteriormente, devido a redução das possibilidades de viabilização para a segunda etapa.

Em termos de relaxação linear, o modelo ainda é capaz de fornecer limites estreitos, mas a heurística baseada em programação linear teve um desempenho fraco, como pode ser visto pelos elevados *gaps*.

Uma observação importante é que, para algumas instâncias, a relaxação linear executada na primeira etapa da heurística baseada na programação linear demorou mais do que a solução da primeira etapa das heurísticas *Relax-and-Fix*. Isto pode ser explicado pelo fato de se utilizar o parâmetro `lazy = 1` para o conjunto de restrições (3.10). Foi observado através de experimentos que o tempo computacional necessário para resolver a relaxação linear do modelo PLI sem as restrições (3.10) é significativamente menor que para o modelo completo.

As Tabelas 6 e 7 apresentam os resultados das instâncias HL e HS, respectivamente. Estas instâncias são mais difíceis de se resolver devido à existência de um número maior de variáveis após a relaxação da janela de tempo. Não foi possível garantir a otimalidade da solução para 10 instâncias de cada um dos cenários e sequer foi possível encontrar uma solução para as instâncias HL17.

Por outro lado, as heurísticas *Relax-and-Fix* geraram soluções com pequenos *gaps*

para a maioria das instâncias. No entanto, estes resultados foram obtidos a elevados custos computacionais e cinco instâncias continuam abertas: HL13, HL17, HL18, HS5, HS17.

A heurística *Relax-and-Fix* 5% se apresentou como uma boa alternativa para estas instâncias mais difíceis. Com tempos computacionais razoáveis, foi capaz de garantir *gaps* menores que 5% (incluindo 5 ótimas para HL e 6 para HS), exceto para a instância HL17 que teve um custo computacional maior do que dos outros casos.

As Tabelas 8 e 9 apresentam os resultados para as instâncias HLA e HLB, respectivamente. Estes dois grupos possuem as instâncias mais difíceis dentre os experimentos.

Para a maioria das instâncias, não foi possível garantir a otimalidade usando o modelo de PLIM ou a heurística *Relax-and-Fix*. A heurística *Relax-and-Fix* 5% demonstrou ser a melhor alternativa para lidar com esses grupos de instâncias. Exceto para instâncias HLB17, o método garantiu *gaps* menores do que 7%. Para maioria dos casos a solução foi gerada com tempos computacionais razoáveis.

A Tabela 10 apresenta uma visão geral do desempenho dos métodos em termos de qualidade da solução e tempo computacional. A primeira coluna identifica o cenário, em seguida, para cada método, é apresentado o valor máximo e o valor médio dos *gaps* percentuais em relação à melhor solução encontrada, o número de vezes que o método obtém a melhor solução do experimento e o tempo computacional médio, em segundos.

A última coluna relata a quantidade de instâncias em cada cenário que permanecem abertas após a aplicação dos testes. O *gap* é calculado como $\frac{(BLB-LB)}{LB}$, onde BLB é o valor da função objetivo para a solução ótima ou para a de melhor limite inferior encontrada e LB é o melhor limite inferior obtido pelo método.

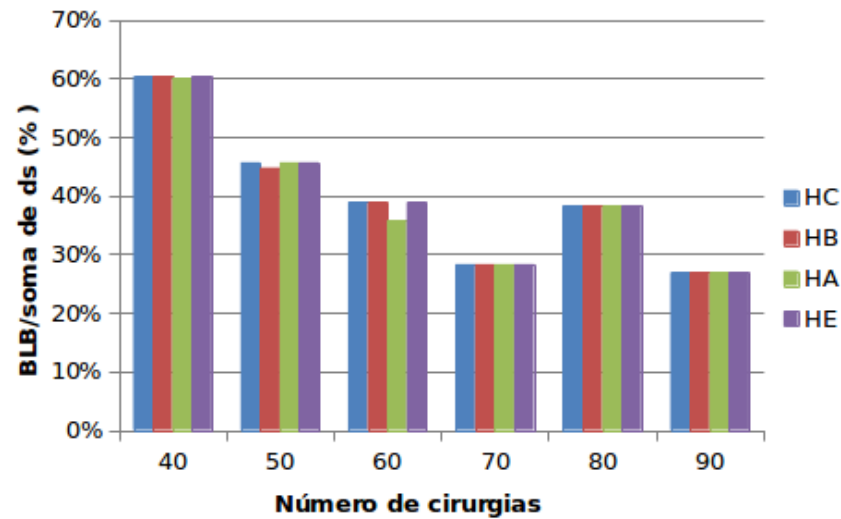
O *gap* não é definido quando o modelo de PLI não foi capaz de encontrar solução viável para uma determinada instância dentro do tempo limite de 21600 segundos. Isto foi observado para instâncias com a relaxação da janela de tempo, que são mais difíceis.

Para cada cenário dentre HL, HS, HLA, e HLB, o modelo inteiro necessitou de elevados tempos computacionais. Frequentemente o limite de 21600 segundos foi atingido sem a prova de otimalidade. Como já foi observado, a heurística baseada em programação linear não foi capaz de gerar boas soluções. A estratégia *Relax-and-Fix* foi capaz de obter valores ótimos e próximos do ótimo para todos os cenários. A heurística *Relax-and-Fix* obteve os melhores resultados embora tenha necessitado de um tempo médio de 3 horas para os conjuntos de instâncias mais difíceis, HLAe HLB.

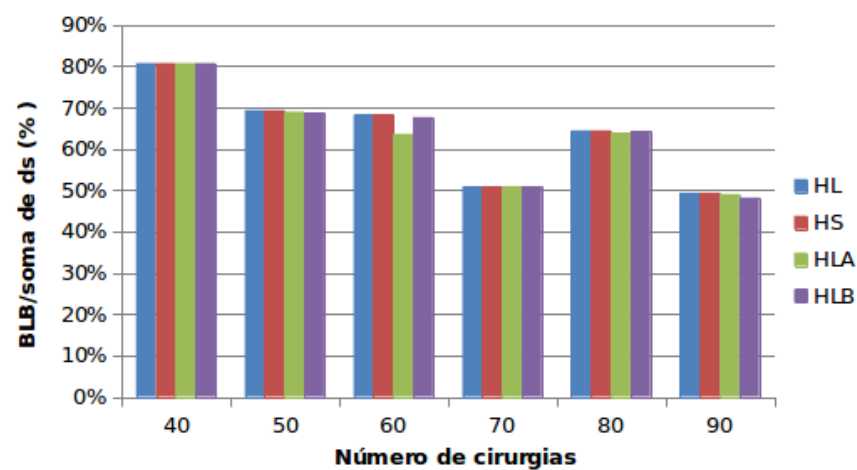
A heurística *Relax-and-Fix* 5% demonstrou-se uma alternativa efetiva para obter pequenos gaps de otimalidade em tempos computacionais mais razoáveis (menor que 1 hora).

A Figura 2 apresenta, em termos percentuais, a proporção entre o valor da melhor solução encontrada e o limitante superior trivial definido pela soma das durações de todas as cirurgias, isto é, $\sum_{s \in S} d_s$. O gráfico (a) é referente às instâncias com maior restrição da janela de tempo (HC, HB, HA e HE), enquanto o gráfico (b) refere-se aos demais conjuntos de instâncias.

A barra representa o percentual médio para cada uma das três instâncias de cada cenário de quantidade de cirurgias $|S|$. Pode-se perceber que as instâncias não são folgadas uma vez que a proporção percentual varia entre 25% e 60% para as instâncias mais restritivas e entre 50% e 80% para instâncias menos restritivas.



(a) Instâncias com janelas de tempo mais restritivas



(b) Instâncias com janelas de tempo menos restritivas

Figura 2 – Percentual de duração das cirurgias agendadas em relação à duração total das cirurgias

I	S	PI		Heurística PL				Relax-and-Fix				Relax-and-Fix 5%	
		LB	g(%)	t	UB	t	LB	t	LB	t	g(%)	LB	t
HC1	40	76	0.0	12	77	1	72	0	76	57	0	76	58
HC2	40	115	0.0	146	117	12	91	0	115	5	0	114	22
HC3	40	82	0.0	28	83	3	74	0	82	3	0	80	17
HC4	50	98	0.0	17932	102	6	83	0	98	90	0	98	109
HC5	50	117	0.0	4113	120	17	95	0	117	34	0	111	63
HC6	50	77	0.0	34	79	2	70	2	77	2	0	74	20
HC7	60	95	0.0	62	97	6	87	0	95	2	0	95	27
HC8	60	107	0.0	14090	110	10	94	0	107	282	0	105	314
HC9	60	89	0.0	34	90	3	83	0	89	2	0	86	18
HC10	70	81	0.0	60	81	7	72	0	81	3	0	79	36
HC11	70	99	0.0	65	100	5	86	0	99	3	0	96	49
HC12	70	100	0.0	49	103	5	89	0	100	10	0	97	39
HC13	80	101	0.0	118	103	12	92	0	101	4	0	101	67
HC14	80	121	0.0	96	123	15	99	0	121	8	0	120	58
HC15	80	112	0.0	260	113	24	90	0	112	6	0	109	74
HC16	90	102	0.0	138	102	14	97	0	102	5	0	99	74
HC17	90	100	0.0	98	101	7	82	0	100	3	0	100	53
HC18	90	101	0.0	128	102	7	93	0	101	3	0	98	59

Tabela 2 – Instâncias HC, similares ao caso real. UB - limite superior, LB - limite inferior, g(%) - gap percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.

I	S	PI		Heurística PL				Relax-and-Fix				Relax-and-Fix 5%							
		LB	g(%)	t	UB	t	LB	t	g(%)	t	UB	t	LB	t	g(%)	t			
HB1	40	76	0.0	31	77	2	71	0	8.5	14	76	1	76	0	0.0	14	75	1.3	16
HB2	40	115	0.0	398	117	15	103	0	13.6	59	115	82	115	0	0.0	122	113	1.8	81
HB3	40	82	0.0	122	83	7	79	0	5.1	33	82	6	82	0	0.0	32	82	1.2	33
HB4	50	98	2.0	21816	102	15	82	0	24.4	54	98	90	98	0	0.0	126	97	3.1	55
HB5	50	112	4.5	22140	120	25	92	0	30.4	89	117	170	117	0	0.0	229	113	3.5	392
HB6	50	77	0.0	82	79	7	66	0	19.7	45	77	3	77	0	0.0	48	77	0.0	48
HB7	60	95	0.0	922	97	13	87	0	11.5	58	95	4	95	0	0.0	59	94	2.1	58
HB8	60	107	0.0	4654	110	29	94	0	17.0	82	107	6	107	0	0.0	69	105	1.9	67
HB9	60	89	0.0	64	90	6	74	0	21.6	30	89	30	89	0	0.0	61	88	1.1	52
HB10	70	81	0.0	150	81	17	71	0	14.1	77	81	4	81	0	0.0	74	81	0.0	72
HB11	70	99	0.0	142	100	16	95	0	5.3	96	99	10	99	0	0.0	103	99	0.0	102
HB12	70	100	0.0	146	103	11	88	0	17.0	62	100	5	100	0	0.0	63	100	0.0	74
HB13	80	101	0.0	309	103	66	91	0	13.2	183	101	237	101	0	0.0	370	101	2.0	148
HB14	80	121	0.0	198	123	40	94	0	30.9	129	121	190	121	0	0.0	291	121	1.7	119
HB15	80	112	0.0	1331	113	33	94	0	20.2	182	112	22	112	0	0.0	163	112	0.9	173
HB16	90	102	0.0	324	102	27	94	0	8.5	164	102	8	102	0	0.0	162	102	0.0	151
HB17	90	100	0.0	180	101	19	79	0	27.8	115	100	6	100	0	0.0	118	100	0.0	111
HB18	90	101	1.0	21749	102	18	94	0	8.5	133	101	156	101	0	0.0	285	101	1.0	161

Tabela 3 – Instâncias HB, compatibilidade apenas entre salas adjacentes. UB - limite superior, LB - limite inferior, g(%) - gap percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.

I	S	PI		Heurística PL				Relax-and-Fix				Relax-and-Fix 5%			
		LB	g(%)	t	UB	t	LB	t	g(%)	t	LB	t	LB	g(%)	t
HE1	40	76	0.0	49	77	5	67	0	14.9	27	76	0	74	4.1	6
HE2	40	115	0.0	1121	117	27	99	0	18.2	78	115	0	115	0.9	44
HE3	40	82	0.0	72	83	8	73	0	13.7	39	82	0	81	2.5	29
HE4	50	98	0.0	347	102	720	81	0	25.9	763	98	0	95	4.2	44
HE5	50	117	0.0	1457	120	2134	93	0	29.0	2206	117	0	111	5.4	255
HE6	50	77	0.0	79	79	7	70	0	12.9	48	77	0	75	2.7	43
HE7	60	95	0.0	212	97	1545	83	0	16.9	1600	95	0	95	1.1	112
HE8	60	107	0.9	21765	110	553	93	0	18.3	626	107	0	105	3.8	71
HE9	60	89	0.0	209	90	12	73	0	23.3	49	89	0	84	6.0	46
HE10	70	81	0.0	163	81	28	66	0	22.7	98	81	0	81	0.0	80
HE11	70	99	0.0	546	100	20	84	0	19.0	105	99	0	98	1.0	89
HE12	70	100	0.0	724	103	41	80	0	28.8	118	100	0	98	3.1	231
HE13	80	101	0.0	368	103	2708	94	0	9.6	2822	101	0	101	0.0	133
HE14	80	121	0.0	432	123	80	104	0	18.3	194	121	0	121	0.8	160
HE15	80	112	0.0	2052	113	2491	87	0	29.9	2610	112	0	110	2.7	350
HE16	90	102	0.0	297	102	1070	89	0	14.6	1190	102	0	102	0.0	141
HE17	90	100	0.0	193	101	375	82	0	23.2	474	100	0	100	1.0	118
HE18	90	101	0.0	359	102	55	87	0	17.2	182	101	0	101	0.0	155

Tabela 5 – Instâncias HE, com dois tipos de recursos especializados, i.e., $|E| = 2$. UB - limite superior, LB - limite inferior, g(%) - gap percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.

I	S	PI			Heurística PL				Relax-and-Fix				Relax-and-Fix 5%						
		LB	g(%)	t	UB	t	LB	t	g(%)	t	UB	t	LB	t	g(%)	t			
HL1	40	120	0.0	423	122	21	101	0	20.8	63	120	8	120	0	0.0	56	117	2.6	60
HL2	40	132	0.0	1284	134	26	117	0	14.5	77	132	415	132	0	0.0	476	130	3.1	68
HL3	40	114	0.0	136	114	11	112	0	1.8	63	114	10	114	0	0.0	76	110	3.6	70
HL4	50	165	1.2	21825	168	97	138	0	21.7	185	166	9488	166	0	0.0	9596	165	1.8	263
HL5	50	153	0.7	21792	154	63	122	0	26.2	129	153	5347	153	0	0.0	5414	153	0.7	295
HL6	50	132	0.0	1276	134	41	126	0	6.3	141	132	3317	132	0	0.0	3445	130	3.1	191
HL7	60	173	0.6	21860	174	95	155	0	12.3	169	173	1435	173	0	0.0	1530	169	3.0	105
HL8	60	163	9.8	21744	181	3607	137	0	32.1	3698	178	2160	178	0	0.0	2257	178	1.7	545
HL9	60	156	1.9	21761	161	91	126	0	27.8	170	159	15466	159	0	0.0	15569	159	0.6	714
HL10	70	153	0.7	22366	156	95	117	0	33.3	179	154	2566	154	0	0.0	2663	149	4.7	301
HL11	70	177	0.0	6703	179	139	137	0	30.7	308	179	21611	177	0	1.1	21823	177	1.1	374
HL12	70	173	1.7	21778	177	111	121	0	46.3	194	174	3987	174	0	0.0	4079	170	4.1	254
HL13	80	191	1.0	21799	193	5802	148	0	30.4	5990	193	21616	191	0	1.0	21822	187	3.2	2521
HL14	80	186	0.0	18997	187	148	157	0	19.1	267	186	2355	186	0	0.0	2508	185	1.1	835
HL15	80	184	0.0	18944	186	297	141	0	31.9	492	184	3873	184	0	0.0	4086	183	1.6	445
HL16	90	180	0.0	8212	181	327	148	0	22.3	543	180	4194	180	0	0.0	4470	180	0.6	830
HL17	90	-	-	21768	192	3551	138	0	39.1	3710	192	21607	188	0	2.1	21782	188	2.1	11669
HL18	90	189	0.5	21835	190	318	139	0	36.7	536	190	21610	189	0	0.5	21845	188	1.1	450

Tabela 6 – Instâncias HL, com janela de tempo somente para recursos especializados. UB - limite superior, LB - limite inferior, g(%) - gap percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.

I	S	PI		Heurística PL				Relax-and-Fix				Relax-and-Fix 5%			
		LB	g(%)	UB	t	LB	t	g(%)	t	UB	t	LB	t	g(%)	t
HS1	40	120	0.0	122	22	104	0	17.3	73	120	550	120	0	0.0	610
HS2	40	132	0.0	134	14	116	0	15.5	71	132	234	132	0	0.0	291
HS3	40	114	0.0	114	11	110	0	3.6	70	114	6	114	0	0.0	77
HS4	50	165	1.2	168	67	144	0	16.7	165	167	21605	166	0	0.6	21720
HS5	50	153	0.7	154	82	141	1	9.2	177	153	365	153	0	0.0	475
HS6	50	132	0.0	134	41	126	0	6.3	152	132	3378	132	0	0.0	3485
HS7	60	173	0.0	174	82	158	0	10.1	170	173	1406	173	0	0.0	1515
HS8	60	177	0.6	181	129	131	0	38.2	243	178	1846	178	0	0.0	1979
HS9	60	160	0.6	163	94	124	0	31.5	228	160	991	160	0	0.0	1125
HS10	70	154	1.3	156	160	120	0	30.0	335	154	905	154	0	0.0	1080
HS11	70	177	0.0	179	181	137	0	30.7	353	177	1218	177	0	0.0	1396
HS12	70	174	0.6	177	105	136	0	30.1	215	174	5598	174	0	0.0	5710
HS13	80	189	2.1	193	285	146	0	32.2	474	191	3778	191	0	0.0	3987
HS14	80	186	0.0	187	92	149	0	25.5	244	186	77	186	0	0.0	271
HS15	80	184	0.5	186	281	152	0	22.4	529	184	5320	184	0	0.0	5610
HS16	90	180	0.0	181	246	148	0	22.3	544	180	3630	180	0	0.0	3940
HS17	90	185	3.2	192	197	136	0	41.2	411	191	21609	188	0	1.6	21877
HS18	90	188	1.1	190	308	149	0	27.5	564	189	1880	189	0	0.0	2186
															533

Tabela 7 – Instâncias HS, sem janelas de tempo. UB - limite superior, LB - limite inferior, g(%) - gap percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.

I	S	PI			Heurística PL			Relax-and-Fix			Relax-and-Fix 5%		
		LB	g(%)	UB	t	LB	t	g(%)	UB	t	LB	t	g(%)
HLB1	40	120	0.0	9069	37	94	0	29.8	119	120	120	0	0.0
HLB2	40	131	1.5	21969	39	122	0	9.8	137	132	132	0	0.0
HLB3	40	114	0.0	485	57	102	0	11.8	163	114	114	0	0.0
HLB4	50	164	1.8	21802	2858	134	0	25.4	2999	167	21612	166	0
HLB5	50	74	108.1	21763	96	128	0	20.3	230	153	16308	149	0
HLB6	50	132	0.0	4405	126	127	0	5.5	351	132	337	132	0
HLB7	60	173	0.0	4715	105	154	0	13.0	263	173	1708	173	0
HLB8	60	108	67.6	21817	2729	131	0	38.2	2925	178	7572	175	0
HLB9	60	153	3.9	21792	197	126	0	27.8	342	160	21608	154	0
HLB10	70	150	4.0	21950	217	115	2	35.6	406	154	2895	154	0
HLB11	70	177	0.6	22071	287	157	0	14.0	682	179	21619	177	0
HLB12	70	173	1.2	21870	219	118	0	50.0	398	174	13475	174	0
HLB13	80	189	2.1	22123	1651	159	0	21.4	2082	193	21716	191	0
HLB14	80	185	0.5	21968	790	154	0	21.4	1026	186	650	184	0
HLB15	80	184	0.0	21038	13682	158	0	17.7	14201	186	21626	184	0
HLB16	90	180	0.6	22217	9412	129	0	40.3	9831	181	21687	180	0
HLB17	90	-	-	22887	21659	142	0	35.2	22061	191	21624	175	0
HLB18	90	185	2.7	21819	6847	143	0	32.9	7383	189	21694	188	0

Tabela 9 – Instâncias HLB, compatibilidade apenas entre salas adjacentes e com janela de tempo somente para os recursos especializados.

UB - limite superior, LB - limite inferior, g(%) - gap percentual dado por $\frac{UB-LB}{LB}$, t - tempo computacional em segundos.

	PI			heurística PL			Relax-and-Fix			Relax-and-Fix 5%		
	gap(%)	max	av	best	av-t	best	gap(%)	max	av	best	gap(%)	max
HC	0.0	0.0	18	2081	16	0	2.6	0.1	17	64	5.4	2.0
HB	4.5	0.2	15	4153	89	0	0.0	0.0	18	133	3.5	0.7
HA	0.0	0.0	18	962	26	0	13.2	1.3	16	594	15.4	3.5
HE	0.0	0.0	18	1691	735	0	4.5	0.2	17	249	6.0	1.6
HL	inf	-	12	15250	940	0	0.0	0.0	18	7972	3.6	1.2
HS	1.6	0.2	13	13810	279	0	0.0	0.0	18	4219	2.7	0.9
HLA	inf	-	7	18061	996	0	0.0	0.0	18	11452	4.9	1.8
HLB	inf	-	8	18098	3644	0	1.7	0.1	16	12149	15.9	2.2

Tabela 10 – Visão geral do desempenho dos métodos.

5 Programação dinâmica estocástica

O objetivo deste capítulo é realizar uma breve revisão sobre o método de modelagem e solução de problemas dinâmicos conhecido como Programação Dinâmica. Esta ferramenta é especialmente útil em problemas onde é necessário tomar decisões sequenciais de natureza correlata.

A idéia é fornecer uma base para que o leitor deste texto acompanhe, com mais facilidade, o desenvolvimento dos capítulos seguintes. Para um maior aprofundamento no tema recomenda-se, no entanto, que o leitor procure bibliografias já consagradas na área como, por exemplo, Bertsekas (2005), Bertsekas (2012) e Powell (2011).

A programação dinâmica lida com problemas onde as decisões são feitas em estágios. Um aspecto chave destas situações é que a decisão não pode ser vista de maneira isolada e, portanto, deve-se balancear a minimização dos custos presentes com a minimização dos custos futuros (BERTSEKAS, 2005).

A ferramenta possui grande flexibilidade de modelagem, característica que a torna extremamente poderosa em termos de aplicação, mas ao mesmo tempo exige um elevado grau de especialização do modelador, fato este que muitas vezes limita a popularização desta técnica.

Outra grande vantagem desta técnica é que ela é capaz de lidar bem tanto frente a problemas determinísticos quanto com decisões mediante incerteza. No entanto, na maioria dos casos, o tamanho de um problema modelado através de Programação Dinâmica cresce muito rapidamente, o que dificulta a solução de instâncias reais do problema. Este crescimento é conhecido como Maldição da Dimensão (*curses of dimensionality*).

Do ponto de vista gerencial, ser capaz de estimar o desempenho do sistema nos diversos cenários de realização da incerteza e o uso de um gerenciamento ativo, isto é, a reação frente a mudanças nos parâmetros iniciais do problema, são formas de contornar o efeito causado pela incerteza.

Considerar a existência do gerenciamento ativo significa realizar um planejamento inicial sabendo que no futuro este será alterado de maneira ótima para lidar com os novos cenários de informação.

Segundo Bertsekas (2005), o resultado de uma decisão tomada no presente pode não ser totalmente previsível e, neste caso, a obtenção da solução ótima deve considerar a ocorrência dos custos presentes e a esperança dos custos futuros, assumindo que decisões ótimas serão tomadas em estágios futuros tendo em vista as informações disponíveis a cada momento.

Em outras palavras, o tratamento do problema como um problema de programação dinâmica estocástica permite o gerenciamento ativo e uma representação mais adequada da situação frente a cenários de incerteza. A existência do gerenciamento ativo é o que difere a programação dinâmica estocástica de uma programação estocástica estática. Apesar de admitir incerteza nos parâmetros, a programação estocástica estática não considera a reavaliação da decisão a partir da realização, no futuro, da incerteza. Tal reavaliação é o que define o gerenciamento ativo.

5.1 Elementos de um problema de Programação Dinâmica Estocástica

Embora, como já argumentado, a Programação Dinâmica possua grande flexibilidade em termos de modelagem, os elementos que serão descritos a seguir são parte de *framework* no qual se baseia a técnica. Mesmo com diversos formatos, tais elementos estão presentes em todos os problemas de Programação Dinâmica Estocástica.

5.1.1 Estado e Estágio

Os problemas de programação dinâmica sempre tratam de sistemas dinâmicos de eventos discretos ou a tempo contínuo. No caso discreto, os eventos podem ser interpretados como instantes discretos de tempo, estágios de decisão ou ocorrência de um evento de interesse.

O estágio é definido como um instante de tempo onde o sistema pode ser avaliado. Em todos os estágios, as informações do sistema necessárias para descrevê-lo e, portanto, avaliá-lo definem o estado do sistema no instante corrente. O estado, representado neste capítulo por $x^{(t)}$, deve conter tão somente o que for necessário para caracterizar o sistema.

Em problema clássico de gestão de estoques, o estado pode, por exemplo, ser o nível de estoque de um determinado item. No caso do sequenciamento de cirurgias, o estado deverá conter, entre outras, informações sobre o agendamento de cirurgias e sobre a disponibilidade dos recursos.

5.1.2 Ação

O conjunto de ações definem as possíveis decisões tomada em cada um dos estágios do processo. Através destas ações, é possível controlar parcialmente a dinâmica do sistema. São as variáveis de decisão do problema e são representadas neste capítulo por $u^{(t)}$.

No caso do sequenciamento de cirurgias, a ação que representa o agendamento de uma cirurgia s para uma sala r iniciando em um período t pode ser modelada através da variável binária y_{srt} , por exemplo.

5.1.3 Política

Em programação dinâmica, o termo política refere-se à uma regra (ou função) que determina a decisão a partir da informação disponível no estado em que o sistema se encontra (POWELL, 2011). Uma política é dita ótima se for capaz de encontrar a melhor solução possível para o problema em qualquer estado do sistema.

Powell (2011) agrupa, inicialmente, as políticas em quatro categorias, descritas a seguir.

- Políticas míopes - Otimiza o objetivo no estágio atual sem estimar e/ou considerar qualquer informação ou ação futura.

- Políticas *Lookahead* - Otimiza o objetivo em um horizonte de tempo combinando estimativas de informações futuras com ações futuras.
- Políticas baseadas em aproximação de regras (*Policy function approximations*) - é a aplicação direta de uma regra para obter uma ação a partir de um estado sem qualquer tipo de procedimento de otimização e sem utilizar estimativas de informações futuras.
- Políticas baseadas em aproximação do valor da função (*Value function approximations*) - Considera o impacto das decisões em um estado futuro através da aproximação do valor de se estar neste estado no futuro.

As características destes quatro grupos elementares podem ser combinadas para obter políticas mais elaboradas. Tanto as políticas do tipo *Lookahead* quanto as políticas baseadas em aproximação do valor da função buscam incorporar elementos futuros na decisão presente. A diferença é que a última busca especificamente aproximar o valor de se estar em um estado para poder aproveitar a estrutura da equação de Bellman.

Destaca-se a diferença entre os termos míope e guloso. Uma política é dita míope se não considera a informação e o impacto da decisão atual nos estágios futuros e é dita gulosa se realiza a otimização com base em uma representação incompleta do objetivo do problema. As políticas baseadas em aproximação do valor da função são gulosas, mas não míopes.

5.1.4 Informações Exógenas

São informações que não são conhecidas à priori e representam as incertezas do processo. Neste capítulo serão representadas por $w^{(t)}$. Podem ser, por exemplo, a duração incerta de uma cirurgia.

5.1.5 Transição

A função de transição determina como o sistema evolui ao longo dos estágios de decisão. Esta função determina a dinâmica do processo. Sendo $x^{(t)}$ o estado presente, sua configuração depende do estado do sistema no estágio anterior ($x^{(t-1)}$),

da decisão tomada (ou controle aplicado) no último estágio($u^{(t-1)}$) e também da influência de um parâmetro aleatório($w^{(t)}$).

Desta forma a dinâmica descrita pela equação abaixo representa a evolução do sistema após aplicação do controle ($u^{(t)}$) no estágio t para um outro estado no estágio $t + 1$. Na equação 5.1, $f(x^{(t)}, u^{(t)}, w^{(t+1)})$ representa uma função recursiva para o estado no período $t + 1$ com base nas informações passadas.

$$x^{(t+1)} = f(x^{(t)}, u^{(t)}, w^{(t+1)}) \quad (5.1)$$

5.2 Objetivo

O objetivo do problema é minimizar uma função de custo com base em decisões tomadas ao longo dos estágios ($u^{(t)}$). O custo de cada estágio é dado em função do estado em que o sistema se encontra e também das decisões tomadas.

Definiremos a função $g(x^{(t)}, u^{(t)})$, como o custo de aplicar o controle u , no estágio t , estando no estado x mediante a ocorrência de uma w .

No estágio final (T_{max}) nenhuma decisão será tomada e um custo terminal $g_{T_{max}}(x^{(T_{max})})$ será gerado, este custo pode representar um prêmio ou penalidade imposto pelo modelo de se chegar ao estágio final no estado $x^{(T_{max})}$. Sendo assim, o custo total $G(x^{(0)})$ é representado por:

$$G(x^{(0)}) = g_{T_{max}} + \sum_{t=0}^{T_{max}-1} g_t(x^{(t)}, u^{(t)}) \quad (5.2)$$

O objetivo do problema é minimizar o custo total, no entanto, devido à presença dos parâmetros aleatórios ($w^{(t)}$) não é possível garantir esta minimização. Em contextos onde a incerteza está presente, a decisão deve ser tomada com a informação disponível e por este motivo, o objetivo do problema passa a ser a minimização da função que representa o custo esperado do sistema.

Seja $\nu = \{u_1, u_2, \dots, u_{T_{max}}\}$ como um conjunto de decisões aplicadas nos estágios $t = 1, \dots, T_{max}$ e também $J_\nu^{(0)}(x^{(0)})$ como o custo mínimo esperado de se aplicar o

conjunto de decisões ν partindo do estado inicial $x^{(0)}$. O Objetivo é dado por 5.3, onde $E_W[\cdot]$ representa a esperança matemática para as incertezas W .

$$\text{Objetivo: } J_\nu^{(0)}(x^{(0)}) = \min_\nu \left\{ E_W \left[g_{T_{max}} + \sum_{t=0}^{T_{max}-1} g_t(x^{(t)}, u^{(t)}) \right] \right\} \quad (5.3)$$

Desta maneira, o objetivo pode ser interpretado como encontrar a sequência de decisões ν que minimize o valor de $J_\nu^{(0)}(x^{(0)})$ para o problema e a função objetivo descrita conforme 5.3.

Em problemas estocásticos é fácil perceber que não é possível definir a priori o conjunto ν^* , isto ocorre porque a melhor decisão em qualquer estágio t depende do valor de $x^{(t)}$ que é uma variável aleatória que terá valor conhecido apenas no instante t .

Uma política π será definida como uma sequência de funções $\mu_t(x^{(t)})$ que selecionam os valores de $u^{(t)}(x^{(t)})$ para um dado estado $x^{(t)}$ em cada estágio. Representaremos uma política qualquer $\pi = \{\mu_1(x_1), \mu_2(x_2), \dots, \mu_{T_{max}}(x^{(T_{max})})\}$ e a política ótima será dada pelo valor de π^* , expressão 5.4.

$$\pi^* = \arg \min_{\pi} \{ J_\pi^{(0)}(x^{(0)}) \} \quad (5.4)$$

5.3 Métodos para solução

Para alguns problemas, a modelagem via programação dinâmica pode, a partir de procedimentos analíticos, gerar como resposta políticas ótimas para um problema específico sem que seja necessário a execução de algoritmos para encontrar a política ótima a cada nova instância do problema ((BERTSEKAS, 2005)). Em outras palavras, o uso da programação dinâmica é, em alguns casos, capaz de extrair do problema resultados gerais que servirão como regras básicas para a tomada de decisão.

Para a maioria dos problemas, no entanto, é necessário que seja construído um algoritmo para ser executado a cada instância do problema. Antes de apresentar algoritmos para a solução de problemas de programação dinâmica, é necessário

que seja definido o *Princípio da Otimalidade de Bellman*, que é baseado em um conceito intuitivo e de fácil entendimento apresentado a seguir:

Princípio da Otimalidade de Bellman

Seja $\pi^* = \{\mu_1^*, \mu_2^*, \dots, \mu_{T_{max}}^*\}$ para um determinado problema e $x^{(t)}$ um estado possível de ocorrer utilizando a política π^* , o subproblema envolvendo os estágios $k = t, \dots, T_{max}$ que tem como estágio inicial $x^{(t)}$ terá como política ótima a política parcial $\pi_{t:T_{max}}^* = \{\mu_t^*, \mu_{t+1}^*, \dots, \mu_{T_{max}}^*\}$

A consequência direta deste princípio é o fato de que a solução de um problema de programação dinâmica pode ser encontrada resolvendo de forma recursiva os subproblemas para os estágios que se situem a jusante do estado corrente e utilizando o resultado como entrada para a solução do presente.

Formalmente, imagine um subproblema que se inicia no estágio t ($[t : T_{max}]$), o custo total de ir de t até T_{max} ($G(x^{(t)})$) é dado pela equação 5.2 adaptada ao horizonte do subproblema, equação 5.5.

$$G(x^{(t)}) = g_{T_{max}} + \sum_{t=t}^{T_{max}-1} g_t(x^{(t)}, u^{(t)}) \quad (5.5)$$

Utilizando a relação 5.1, o custo mínimo deste subproblema, iniciando no estado $(x^{(t)})$ é dado por 5.6.

$$\begin{aligned} J_{\nu^{(t:T_{max})}}^{(t)}(x^{(t)}) &= \min_{\nu^{(t:T_{max})}} \mathbb{E}_{W^{(t+1:T_{max})}} \left[g^{(t)}(x^{(t)}, u^{(t)}) + \sum_{k \in \{t+1:T_{max}\}} g_k(x^{(k)}, u^{(k)}) \right] \\ &= \min_{\nu^{(t:T_{max})}} \left\{ g^{(t)}(x^{(t)}) + \mathbb{E}_{W^{(t+1:T_{max})}} \left[\sum_{k \in \{t+1:T_{max}\}} g_k(x^{(k)}, u^{(k)}) \right] \right\} \\ &= \min_{u^{(t)}} \left\{ g^{(t)}(x^{(t)}, u^{(t)}) + \mathbb{E}_{W^{(t+1:T_{max})}} \left[J_{\nu^{(t+1:T_{max})}}^{(t+1)}(x^{(t+1)}) | x^{(t)}, u^{(t)} \right] \right\}, \\ &\quad \forall t \in \{0\} \cup T \end{aligned} \quad (5.6)$$

E a política ótima para o subproblema $[t : T_{max}]$ é a união do controle ótimo para o estado t (μ_t^*) com a política ótima do estágio $t + 1$, expressão 5.7.

$$\pi_{[t:T_{max}]}^* = \{\mu_t^*\} \cup \pi_{[t+1:T_{max}]}^* \quad (5.7)$$

O custo $J_{\pi^*}^{(t)}(x^{(t)})$ representa o menor custo possível para ir do estágio t ao T_{max} e é conhecido como *cost-to-go*.

Um problema de programação dinâmica em horizonte finito é um problema definido para um número finito de estágios de decisão com valor terminal conhecido para cada possível estado terminal ($g_{T_{max}}$). Para estes casos, o princípio de Bellman nos leva diretamente para ao método de solução chamado *Backward Dynamic Programming Algorithm*.

O método é um algoritmo recursivo que consiste em realizar a avaliação dos estados a partir do estágio terminal até o estágio inicial, em um movimento “para trás” no horizonte de tempo. Os passos deste método estão especificados no Algoritmo 1. Em problemas determinísticos podemos encontrar a solução ótima também através de uma função *Forward* que soluciona o problema a partir da solução de subproblemas iniciais (BERTSEKAS, 2005).

Em alguns problemas determinísticos de natureza combinatória, a modelagem via programação dinâmica pode auxiliar a construção de um algoritmo eficiente para encontrar a solução ótima do problema, como é o caso de Wagner e Whitin (1958), Kumar e Muthuraman (2004) e Li et al. (2004) ou uma heurística para o problema, como fez Bollapragada e Morton (1999).

```

1 início
2   | inicialize  $J^{(T_{max})}(x) = g_{T_{max}}(x) : \forall x \in X_{T_{max}};$ 
   fim
3 para todo  $t \in \{T_{max} - 1..0\}$  faça
4   | para todo  $x \in X_t$  faça
5     |  $J^{(t)}(x) \leftarrow \min_u \left\{ g(x, u) + \sum_{x' \in X_{t+1}} \mathbb{P}(x'|x, u) J^{(t+1)}(x') \right\};$ 
     fim
   fim
6 retorna  $J^{(0)}(x^{(0)})$  e  $u^{(0)*}$ ;

```

Algoritmo 1: *Backward Dynamic Programming Algorithm*

Uma vez que o valor de todos os estados no estágio terminal $J^{(T_{max})}$ é conhecido, é possível calcular o valor de todos os possíveis estados no estágio anterior $t = T_{max} - 1$ através da equação do passo 5. Esta lógica é utilizada até que o algoritmo atinja o estágio 0. A otimalidade deste método é garantida pelo princípio de Bellman, no entanto, o método enumera todos os possíveis estados a cada estágio.

Quando o número de estágios é infinito, o problema só é solucionável quando se pode comparar o valor de um estado no futuro com o valor do estado no presente através da aplicação de um fator de desconto $\gamma \in [0, 1)$. Isto garante a convergência da busca e implica que custos e valores de decisões presentes possuem importância maior do que custos e valores de decisões futuras. Neste caso, a relação apresentada pela equação 5.6 passa a ser descrita pela equação 5.8.

$$J^{(t)}(x^{(t)}) = \min_{u^{(t)}} \left\{ g^{(t)}(x^{(t)}, u^{(t)}) + \gamma \mathbb{E}_{W^{(t+1:T_{max})}} \left[J^{(t+1)}(x^{(t+1)}) | x^{(t)}, u^{(t)} \right] \right\}, \quad \forall t \quad (5.8)$$

Outras características comumente encontradas em problemas de horizonte infinito são a estacionariedade ou a ciclicidade dos parâmetros. A estacionariedade implica que os parâmetros do problema não se alteram com os estágios, e a ciclicidade significa que a alteração se dá em ciclos. Para problemas estacionários, a relação da equação 5.8 é descrita por 5.9.

$$J(x) = \min_u \{g(x, u) + \gamma \mathbb{E}_W [J(x') | x, u]\} \quad (5.9)$$

Mesmo um problema de programação dinâmica estacionário/cíclico definido para um horizonte infinito descontado não é possível utilizar o Algoritmo 1 por se tratar de um método enumerativo. Os algoritmos *Value Iteration Algorithm* (2) e *Policy Iteration Algorithm* (3) são os métodos mais utilizados para resolver problemas de horizonte infinitos descontados.

```

1 início

2   | inicialize  $J_0(x) = 0 : \forall x \in X$ ;
3   | inicialize  $n = 1$ ;
   fim
4 enquanto  $\|J_n - J_{n-1}\| \geq \frac{\epsilon(1-\gamma)}{2\gamma}$  faça

5   |  $n = n + 1$ ;
6   | para todo  $x \in X$  faça
7     |  $J_n(x) \leftarrow \min_u \{g(x, u) + \gamma \sum_{x' \in X} \mathbb{P}(x'|x, u) J_{n-1}(x')\}$  ;
     fim
   fim
8 Seja  $\pi^\epsilon$  a política resultante do passo 7 e  $J_\epsilon$  o vetor de valores para cada
   estado;
9 retorna  $\{J_\epsilon, \pi^\epsilon\}$  ;

```

Algoritmo 2: *Value iteration Algorithm*

No *Value Iteration Algorithm* (VIA - Algoritmo 2), ϵ é uma constante de tolerância ($\epsilon \sim 0$), γ é o fator de desconto do problema e o operador $\|v\|$ é a norma máxima do vetor v expressa pela equação 5.10. A versão do método descrito é a apresentada por Powell (2011)(pág. 69). A ideia do algoritmo é estimar o valor de estar em um estágio x iterativamente, até que o maior desvio encontrado seja menor do que o critério de convergência $\frac{\epsilon(1-\gamma)}{2\gamma}$. Powell (2011) apresenta a prova de convergência para o algoritmo e variações do método.

É importante observar que antes que o critério de convergência seja atingido, os valores J_n estão sendo aproximados, isto é, não são os valores exatos. Os erros provenientes desta aproximação leva à geração de políticas subótimas ao longo das iterações até que o critério de convergência seja alcançado.

$$||v|| = \max_{x \in X} |v(x)| \quad (5.10)$$

A seguir, Algoritmo 3 apresenta um método alternativo ao VIA. Trata-se do *Policy Iteration Algorithm* (PIA) e a versão descrita é a apresentada por Powell (2011)(pág. 74). A política π é definida, neste algoritmo, como a aplicação de uma decisão $u(x)$ para cada um dos possíveis estados do sistema $x \in X$. P^π e c^π são respectivamente a matriz estacionária da transição de estados e o e o vetor de custos de aplicação da política π . A operação realizada no passo 6, refere-se ao cálculo do valor de se estar em um determinado estado a partir de uma política e é detalhada por Powell (2011) na página 68.

```

1 início
2   | selecione uma política  $\pi^0$ ;
3   | inicializar  $n = 1$ ;
   fim
4 encontre a matrix de transição  $P^{\pi^{n-1}}$  ;
5 encontre o vetor de custo  $c^{\pi^{n-1}}$  associado ao uso da política  $\pi^{n-1}$ ;
6 resolva  $(I - \gamma P^{\pi})J_n = c^{\pi^{n-1}}$  para  $J_n$  ;
7 encontre a política  $\pi^n$  definida de forma que
    $u^n(x) = \arg \min_u (G(u, x) + \gamma P^{\pi} J_n)$ ;
8 se  $\pi^n = \pi^{n-1}$  então
9   | retorna  $\{\pi^n\}$  ;
   fim
10 senão
11  | volte ao passo 4;
   fim

```

Algoritmo 3: *Policy Iteration Algorithm*

No PIA, uma política é escolhida e então o valor de estar em cada um dos estados $x \in X$ é calculado. A partir do cálculo destes valores (J_n), uma nova política é encontrada, passo 7. Em contraste com o VIA, o PIA parte de uma política arbitrária e encontra os valores exatos de J a cada iteração e a partir destes, atualiza a política utilizada. Powell (2011) argumenta que o PIA é mais fácil de ser implementado e que, considerando o número de iterações, converge mais rapidamente. No entanto, além do passo 7, o passo 6 possui elevado custo computacional o que pode inviabilizar o uso do método quando o número de estados é grande.

5.4 Maldição da Dimensão

O principal problema da programação dinâmica estocástica é a maldição da dimensão (*curses of dimensionality*). O rápido crescimento do problema se deve à três fontes: (i) dimensão da variável que representa o estado do sistema; (ii) di-

mensão do conjunto de ações realizáveis e (iii) dimensão do conjunto de caminhos aleatórios que modelam as incertezas do processo.

Em um problema com um estado de dimensão n onde cada variável de estado tenha, por exemplo, x valores possíveis há, inicialmente, x^n estados diferentes a cada estágio de decisão. Suponha também que cada estado possua m variáveis de decisão associada com y valores cada e que uma vez aplicada a ação o resultado dependa de l fontes de incerteza representadas por variáveis aleatórias com z possíveis saídas. A expansão da árvore de possíveis caminhos possuirá, a cada estágio, $x^n y^m z^l$ ramos saindo de cada nó de estado.

Para um problema de horizonte finito com T estágios, o algoritmo *Backward* deverá avaliar $\frac{(x^n y^m z^l)^T - 1}{x^n y^m z^l - 1}$ estados. Considerando um problema extremamente pequeno com $T = 5$, $n = m = l = 2$ e $x = y = z = 2$ (Problema 1) sendo executado em um computador com um único processador de $1.0GHz$ e supondo seja necessário apenas uma operação para avaliar cada nó da rede, seriam necessários apenas 0,02 segundos para executar o algoritmo *Backward*. No entanto, em um problema ligeiramente maior (Problema 2), com $x = y = z = 4$ seriam necessários 3,26 dias e em um problema mais realista (Problema 3), ainda assim pequeno, com $x = y = z = 10$, 32 milhões de anos!

O Algoritmo *Backward* faz uma busca em profundidade, sendo necessário armazenar apenas uma pequena parte da árvore para a realização dos cálculos. No entanto, caso quiséssemos armazenar toda a árvore de estados necessitaríamos de 68,2 MB; 1,27PB e 4YB para os problemas 1, 2 e 3, respectivamente, considerando que a codificação do estado fosse feita utilizando um número inteiro por dimensão ao custo de 4 Bytes por inteiro. A Tabela 11 sumariza os valores apresentados.

Problema	$x = y = z = ?$	Tempo computacional (<i>Backward</i>)	Necessidade de memória para representação de toda a árvore de estados
1	2	0,02 s	68,2 MB
2	4	3,26 dias	1,27 PB
3	10	32 milhões de anos	4 YB

Tabela 11 – Maldição da dimensão

No caso de problemas estacionários com o horizonte infinito descontado, o uso do método exato VIA demandaria a solução de um problema de otimização (passo 7) para cada estado a cada iteração até a convergência do algoritmo. Considerando os exemplos de problemas citados, seriam $x^n y^m z^l$ problemas equivalente à 64, 4096 e a 10^6 problemas por iteração, respectivamente. A utilização do método PIA implicaria, adicionalmente, na inversão de uma matriz quadrada de dimensão x^n (64, 4096 e a 10^6) a cada iteração do método.

5.5 Aproximação de Programação Dinâmica

A maldição da dimensão gerou a necessidade do desenvolvimento de novos métodos para solucionar problemas reais de programação dinâmica estocástica. Tanto Powell (2011) quanto Bertsekas (2012) consideram que o fundamento básico da aproximação de programação dinâmica consiste em tentar resolver o problema do início para o fim, isto é, utilizando o Algoritmo *Forward Dynamic Programming*.

Para encontrar o valor de estar no estágio $x^{(t)}$ ($J^{(t)}(x^{(t)})$) deve-se resolver a equação de Bellman para $x^{(t)}$ (5.6) que pode ser escrita, de maneira mais geral, como 5.11.

$$J^{(t)}(x^{(t)}) = \min_{u^{(t)} \in \Xi^{(t)}} \left\{ g^{(t)}(x^{(t)}, u^{(t)}) + \gamma \sum_{x' \in X^{(t+1)}} \mathbb{P}(x'|x^{(t)}, u^{(t)}) J^{(t+1)}(x') \right\}, \quad \forall t \in \{0\} \cup T \quad (5.11)$$

Em 5.11, $\Xi^{(t)}$ é o conjunto das ações possíveis em t , $X^{(t+1)}$ o conjunto de todos os estados possíveis para $t + 1$ e γ o fator de desconto. Uma vez que, no geral, não é possível percorrer todos os estados do sistema, a Aproximação de Programação Dinâmica (APD) busca estimar o valor da aplicação de uma ação através da simulação de caminhos aleatórios para o futuro a partir de $x^{(t)}$ e $u^{(t)}$. Além disto, o valor de $J^{(t+1)}(x')$ não é conhecido de maneira exata e, por este motivo, necessita de ser aproximado por $\overline{J^{(t+1)}}(x')$ a partir de informações coletadas ao longo das iterações do algoritmo desenvolvido.

Powell (2011) (pág. 147) apresenta um *framework* genérico para a construção de um algoritmo de Aproximação de Programação Dinâmica. O Algoritmo 4 apre-

senta esta estrutura. O autor argumenta que os métodos de APD podem ser vistos como “**Simuladores Otimizantes**” e podem ser utilizados tanto para resolver problemas de programação dinâmica quanto para otimizar processos simulados de maneira inteligente.

```

1 início

2   | inicialize um estado inicial  $x^{(0),1}$ ;
3   | inicialize uma função de aproximação
   |  $\overline{J_0^{(t)}}(x^{(t)}), \forall x^{(t)} \in X^{(t)} \forall t \in \{0\} \cup T$ ;
4   | inicialize  $n = 1$ ;
   | fim
5 enquanto não atingir o critério de parada faça

6   | gere um caminho aleatório  $\omega^n$  ;
7   | para todo  $t \in \{0\} \cup T$  faça

8     | Otimização: encontre a decisão ótima  $(u^{(t),n})^*$ ;
9     | Simulação: utilize  $\omega^n$  e  $(u^{(t),n})^*$  para encontrar  $x^{(t+1),n}$ ;
     | fim
10  | atualize  $\overline{J_n^{(t)}}(x^{(t)}), \forall t \in \{0\} \cup T$ ;
     | fim
11 retorna  $\overline{J_n^{(t)}}(x^{(t)}), \forall x^{(t)} \in X^{(t)} \forall t \in \{0\} \cup T$  ;

```

Algoritmo 4: Estrutura geral de um algoritmo de aproximação de programação dinâmica

Os principais desafios decorrentes da construção de métodos para resolver problemas de programação dinâmica estocástica consistem no desenvolvimento do modelo de simulação (passos 6 e 9), o procedimento para solução dos subproblemas de otimização (passo 8) e na definição do método de aproximação de $J^{(t)}$ (passo 3). Usualmente, o método de aproximação de $J^{(t)}$ é feito a partir de regressão dos valores obtidos a cada iteração tendo como predecessores indicadores que represente parte das informações contidas no estado avaliado. Por exemplo, em agendamento de cirurgias, um indicador que pode ser utilizado é a ocupação

das salas nos próximos períodos.

Outro grande desafio é a solução do *trade-off* entre diversificação e intensificação (*Exploration vs Exploitation*) durante o processo de aproximação da função $J^{(t)}$. A diversificação permite refinar a função de aproximação através do conhecimento do espaço de estado e soluções. Por outro lado, o processo de intensificação busca uma boa solução a partir da aproximação corrente.

No Capítulo 6, um modelo para problema dinâmico estocástico diário de sequenciamento e alocação de recursos em cirurgias (PDEDSARC) será apresentado, assim como métodos para sua solução. Os métodos desenvolvidos se baseiam no *Policy Iteration Algorithm* (PIA) Aproximado por Regressão Linear Múltipla. Os resultados dos testes computacionais são discutidos no Capítulo 7.

6 Modelo dinâmico estocástico para o sequenciamento e alocação de recursos em cirurgias

6.1 Modelo proposto

A abordagem determinística utilizada no Capítulo 3 não é capaz de considerar incertezas como a duração dos procedimentos e a chegada de cirurgias urgentes. É necessário observar, no entanto, que os recursos disponíveis para a realização de cirurgias eletivas devem ser compartilhados com cirurgias urgentes. Variações no tempo previsto para realização dos procedimentos podem inviabilizar o planejamento inicial ou torná-lo subótimo no decorrer do expediente. É necessário o desenvolvimento de métodos de solução que sejam capazes de lidar com estas características.

A partir de uma simplificação do modelo para o PDSARC determinístico, apresentado no Capítulo 3, o problema foi reformulado para introduzir incertezas e considerar a dinamicidade do processo chegando assim, a um modelo para o problema dinâmico estocástico diário de sequenciamento e alocação de recursos em cirurgias (PDEDSARC), proposto neste capítulo. As simplificações foram necessárias para tornar tratável o subproblema de otimização avaliado a cada estágio e possibilitar, com isso, o uso da abordagem dinâmica estocástica. Não foi considerada a existência dos recursos humanos especializados, bem como todas as características do problema modelado no Capítulo 3 relacionadas a estes elementos.

No modelo desenvolvido, as cirurgias podem ser realizadas durante todo o dia, mas devem inicialmente ser planejadas para serem executadas dentro da janela de tempo representada pelo conjunto $T = \{1..T_{max}\}$. Os elementos $t \in T$ representam o início de cada uma das T_{max} frações de hora pertencentes ao expediente do dia.

Por exemplo, o expediente pode ter início às 7 horas da manhã e término às 19

horas. Dividindo o tempo em fração de 30 minutos, temos o expediente representado pelo conjunto $T = \{1..24\}$. Às 7h da manhã tem-se $t = 1$, 7h30 $t = 2$, 12h $t = 11$ e às 18h30 inicia-se o último período, $t = 24$. Ao início de cada período, após a realização da incerteza, uma decisão é tomada.

Os períodos de discretização do expediente diário $t \in T$ coincidem com os instantes de tempo onde o sistema será avaliado e o planejador poderá tomar decisões. Desta forma, os períodos de discretização são também os estágios de decisão do problema. Cabe ressaltar que o estágio $t = 0$ representa o período anterior ao início do expediente diário e que, embora exista uma decisão a ser tomada nesse instante de tempo, o intervalo $[0, 1)$ não faz parte do expediente, isto é, não pode ser utilizado para o agendamento de cirurgias. O objetivo do problema é planejar e controlar o sequenciamento de cirurgias e a alocação de recursos em um bloco cirúrgico de forma a minimizar o custo total durante um dia.

Por se tratar de uma abordagem dinâmica, assumimos que ao início de cada período o analista poderá decidir alterar o planejamento inicial frente ao novo cenário. Isto porque, ao início de cada período, novas informações são reveladas a partir da realização de incertezas. Podem chegar cirurgias urgentes ou uma cirurgia pode terminar antes do previsto ou demorar mais que o esperado.

A cada período o analista poderá **remarcar** uma cirurgia previamente agendada, mas que ainda não se iniciou, isto é, alterar o horário de início ou a alocação de algum recurso; poderá também **cancelar** uma cirurgia agendada e não iniciada. Nesse caso, não será possível realizar uma remarcação posterior; e também **agendar** uma cirurgia que tenha acabado de chegar. Caso uma cirurgia não seja agendada no período em que chega, consideramos que a mesma não será realizada.

Como exemplo pode-se descrever uma situação onde uma cirurgia urgente chega às 10 horas da manhã, possui duração estimada de 4 horas e uma janela de tempo para seu início de 3 horas. Suponha que esta cirurgia necessite de um recurso renovável que está agendado para ser utilizado em outra cirurgia que se iniciará às 11h e possui duração prevista de 5 horas. Frente a este problema o tomador de decisão pode optar por remarcar a cirurgia previamente agendada para outro horário, ou pode cancelá-la caso não seja possível realizar a remarcação. Há também a

possibilidade de não aceitar a cirurgia urgente e, neste caso, ela não seria agendada.

6.1.1 Estado

O conjunto de cirurgias, $S^{(t)}$, é definido para cada estágio de decisão t pois evolui com o passar do tempo. Este conjunto é particionado nos subconjuntos $S_D^{(t)}$, $S_M^{(t)}$ e $S_I^{(t)}$, que representam respectivamente os subconjuntos de **cirurgias disponíveis** para agendamento, o conjunto de **cirurgias agendadas** e o conjunto de cirurgias que estão ocorrendo em um determinado tempo t , **cirurgias iniciadas**.

Em $t = 0$, o conjunto $S_D^{(0)}$ é composto pelas cirurgias eletivas e $S_M^{(0)} = S_I^{(0)} = \emptyset$. A partir de $t = 1$, $S_D^{(t)}$ é formado pelas cirurgias urgentes que chegaram depois da tomada de decisão do período anterior, ou seja, no intervalo de tempo $[t - 1, t)$. No conjunto $S_M^{(t)}$ encontram-se cirurgias que foram marcadas em períodos anteriores para iniciar em um período k , $t \leq k \leq T_{max}$. Já o conjunto $S_I^{(t)}$ é formado pelas cirurgias que iniciaram nos períodos anteriores e não terminaram até t , ou seja que ainda estarão ocorrendo no intervalo $[t, t + 1)$.

Para melhor compreensão, será antecipada parte da lógica de transição de um estágio para outro. A transição do problema será apresentada de forma detalhada na seção 6.1.3.1. Durante a transição, uma cirurgia pode mudar de um conjunto para o outro.

Uma cirurgia que está em $S_D^{(t)}$ pode passar para $S_M^{(t+1)}$ ou $S_I^{(t+1)}$. Passará para $S_M^{(t+1)}$ se for agendada para um período k , $t + 1 \leq k \leq T_{max}$; ou passará para $S_I^{(t+1)}$ se tiver início agendado para o tempo t , isto é, assim que chegou. Uma cirurgia de $S_M^{(t)}$ pode passar para $S_I^{(t+1)}$ se tiver início no período t . As cirurgias podem, também, deixar o sistema. Isto ocorre quando $s \in S_D^{(t)}$ não é agendada e, neste caso, não passa nem para $S_M^{(t+1)}$ e nem para $S_I^{(t+1)}$; ou quando $s \in S_M^{(t)}$ é cancelada e desta forma não passa para $S_M^{(t+1)}$ e nem para $S_I^{(t+1)}$; ou quando $s \in S_I^{(t)}$ é finalizada entre $[t, t + 1)$. As cirurgias que continuam agendadas, seja com ou sem remarcação, passam para $S_M^{(t+1)}$.

O conjunto de cirurgias também pode ser particionado com base no conjunto de clínicas C e no conjunto de preceptores médicos MP . Desta forma, $S_c^{(t)}$ representa o conjunto de cirurgias $s \in S^{(t)}$ que pertencem à uma determinada clínica C e $S_p^{(t)}$

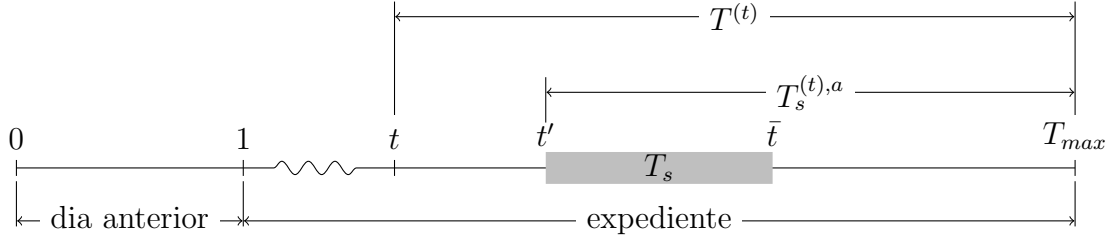


Figura 3 – Janela de tempo (T_s) e janela de agendamento ($T_s^{(t),a}$) de uma cirurgia s

o conjunto de cirurgias $s \in S^{(t)}$ atribuídas a um preceptor $p \in MP$.

Considera-se a existência de um conjunto de salas R o qual pode ser particionado de forma a associar, para cada clínica $c \in C$, um conjunto $R_c \subseteq R$ de salas em que o procedimento cirúrgico s da clínica c ($s \in S_c$) pode ser realizado. Além das salas, outros recursos renováveis são considerados ($q \in Q$) os quais podem ou não ser necessários para a realização de uma cirurgia. O parâmetro h_q indica a quantidade total de recursos renováveis do tipo $q \in Q$ disponível no início do dia.

Uma cirurgia é uma entidade que possui características associadas a ela. Para cada cirurgia é definida uma janela de tempo $T_s \subseteq T$ para seu início, um conjunto R_s de salas onde a mesma pode ser realizada (definido a partir de R_c), um parâmetro binário a_{sq} que indica a necessidade do recurso renovável q e uma distribuição de probabilidade discreta $P_s(\delta)$ para a definição de sua duração Δ_s .

As cirurgias podem iniciar até o último período do expediente, o que permite que sejam realizadas parcialmente fora do expediente. No entanto, é possível contabilizar custos para esta situação dimensionando de maneira adequada a janela de tempo de início da cirurgia. Seja $T_s = \{t', \dots, \bar{t}\}$, as cirurgias não podem iniciar antes do início t' da janela de tempo T_s , mas podem iniciar após seu término ($\bar{t}$). Neste caso, define-se $T_s^a = T_s \cup \{k \in T : \bar{t} + 1 \leq k \leq T_{max}\}$ como o conjunto de períodos para os quais pode-se realizar o agendamento de s . Conforme será discutido na seção 6.1.3.2, no caso de início fora da janela de tempo, um custo é cobrado como penalização.

Em cada estágio de decisão, isto é, ao início de cada período, o planejador decidirá

acerca do sequenciamento das cirurgias existentes no sistema para os períodos futuros $k \in T^{(t)}$, onde $T^{(t)} = \{t, t+1, \dots, T_{max}\}$ representa o conjunto de períodos futuros. Para cada cirurgia $s \in S^{(t)}$ é associada uma janela de tempo $T_s^{(t)} = \{k \in T_s \cap T^{(t)}\}$ e o conjunto de períodos de agendamento $T_s^{(t),a}$. A Figura 3 representa os subconjuntos de períodos que formam as referidas janelas de tempo. O indicador L_p^t possui o valor 1 se o preceptor $p \in MP$ estiver realizando uma cirurgia no instante t .

O parâmetro indicador $L_r^{t,k}$ é igual a 1 se no instante t houver a informação de que a sala $r \in R$ estará, certamente, ocupada no período $k \in \{t, t+1\}$. Deve-se notar que se o indicador $L_r^{t,t+1} = 0$ não implica que a sala não será utilizada em $k = t+1$, significa apenas que não há certeza acerca do uso. É possível determinar o valor de $L_r^{t,t+1}$ devido à existência de um período de desinfecção pós-cirúrgica. Certamente, uma sala alocada para uma cirurgia que está em andamento no período atual ($s \in S_I^{(t)}$) estará indisponível em $t+1$. Por exemplo, uma cirurgia (s_1) está ocorrendo em uma sala (r_1) no período atual (t), e uma vez que existe um período de limpeza pós-cirúrgica, r_1 estará indisponível em $t+1$ e mesmo que s_1 termine no intervalo de tempo corrente $[t, t+1)$. Portanto, certamente a sala estará indisponível em $[t+1, t+2)$. Neste caso, $L_{r_1}^{t,1} = 1$ e $L_{r_1}^{t,2} = 1$.

A informação referente a $L_r^{t,k}$ complementa a informação contida no conjunto $S_I^{(t)}$ no sentido de definir o horizonte de ocupação confirmada do recurso sala. Este parâmetro é utilizado na construção da região de viabilidade do modelo de programação inteira para a decisão em um determinado período t .

Em um estágio t , para toda cirurgia agendada ($s \in S_M^{(t)}$) ou iniciada ($s \in S_I^{(t)}$), o parâmetro $\tau_s \in T_s$ representa o período para o qual o início da cirurgia está agendado ($s \in S_M^{(t)}$) ou o período no qual a cirurgia foi iniciada ($s \in S_I^{(t)}$). De maneira análoga, $\eta_s \in R_s$ representa a sala alocada a s .

O estado é composto por toda informação necessária para tomada de decisão em um determinado estágio, isto é, o conjunto de todos os parâmetros acima descritos. Com o objetivo de simplificar a notação denotaremos por $x^{(t)}$ o conjunto de todas as variáveis de estado do estágio $t \in \{0\} \cup T \cup \{T_{max} + 1\}$. O período $T_{max} + 1$ e o estado associado ao mesmo são definidos apenas para auxiliar na contagem dos

custos. O conjunto $X = \bigcup_{t \in \{0\} \cup T \cup \{T_{max}+1\}} x^{(t)}$ é o caminho de transição dos estados e $\mathcal{X}^{(t)}$ é o conjunto de todas as possibilidades de valores para as variáveis de estado $x^{(t)}$.

6.1.2 Incertezas

Como mencionado anteriormente, três fontes de incerteza foram consideradas: (i) duração das cirurgias; (ii) chegada de cirurgias urgentes; (iii) disponibilidade de recursos. A duração de cada cirurgia δ_s foi descrita através de uma variável aleatória discreta e limitada (Δ_s) . O parâmetro $\bar{\delta}_s$ é utilizado para representar a esperança da duração da cirurgia s . Além das cirurgias eletivas (quantidade conhecida a priori) o modelo considera também a possibilidade de chegada de cirurgias de urgência através do processo estocástico $P_s(t)$ que modela as probabilidades de chegada de cirurgias urgentes do tipo s no período t .

As chegadas das cirurgias de urgência obedecem a processos estocásticos discretos e indexados nos estágios de decisão $k \in T$. Cada tipo de procedimento cirúrgico possui um processo estocástico específico a ele associado. Uma cirurgia chega no sistema com o preceptor já definido e a definição desse preceptor se dá a partir de uma distribuição discreta de probabilidade. Esta distribuição de probabilidade está associada aos cirurgiões pertencentes à especialidade (clínica) da qual o tipo de procedimento cirúrgico faz parte. Após a chegada de uma cirurgia de urgência, tem-se uma janela de tempo T_s para que esta seja iniciada.

A incerteza na disponibilidade de recursos se dá pela combinação dos atrasos na liberação destes em decorrência das incertezas de chegada e/ou de duração das cirurgias. Denota-se como $\omega^{(t)}$ o conjunto de todas as variáveis aleatórias do estágio $t \in T$. Em um processo estocástico indexado em períodos discretos, um caminho aleatório é uma sequência de possíveis valores para este processo. Assim, $\omega = \bigcup_{t \in T} \omega^{(t)}$ o caminho aleatório em T , $\Omega^{(t)}$ é o espaço amostral das variáveis aleatórias $\omega^{(t)}$ e Ω o espaço amostral dos caminhos aleatórios ω .

No estágio de decisão t tem-se a realização de variáveis aleatórias. Portanto, $\omega^{(t)}$ indica as cirurgias urgentes que chegaram no intervalo $[t - 1, t)$ e também

as cirurgias de $S_I^{(t-1)}$ que terminaram. Por exemplo, $\omega^{(3)}$ pode indicar a chegada de uma cirurgia cardíaca urgente de um paciente do cirurgião José e de uma cirurgia pediátrica urgente associada à cirurgiã Maria. Em $\omega^{(3)}$ pode vir também a informação de que a cirurgia de $S_I^{(2)}$ que ocorria na sala r_1 , com o cirurgião Paulo terminou no intervalo $[2, 3)$. Por outro lado, $\Omega^{(3)}$ representa todas as possibilidades de combinações de chegadas de cirurgias atreladas aos cirurgiões e também todas as possibilidades de término (ou não) das cirurgias de $S_I^{(2)}$.

6.1.3 Decisão

A variável de decisão $y_{rsk}^{(t)}$, no estágio t , determina o agendamento, a remarcação ou o cancelamento de uma cirurgia $s \in S^{(t)}$. A decisão, no estágio t , de iniciar no período t uma cirurgia $s \in S_D^{(t)}$ ou $s \in S_M^{(t)}$ corresponde ao agendamento para $\tau_s = t$ em alguma sala $r \in R_s$. A decisão de rejeitar uma cirurgia $s \in S_D^{(t)}$ corresponde ao não agendamento da mesma. Para $s \in S_D^{(t)}$, $y_{rsk}^{(t)} = 1$ indica o agendamento de s para a sala $r \in R_s$ e no tempo $k \in T_s^{(t),a}$. É importante observar que se uma cirurgia de $S_D^{(t)}$ não for agendada, isto é, todas as variáveis $y_{rsk}^{(t)} = 0$, a mesma é rejeitada. Para $s \in S_M^{(t)}$, $y_{rsk}^{(t)} = 1$ pode indicar a desmarcação de s que estava previamente agendada para (r, k) ou a remarcação para a sala $r \in R_s$ no período $k \in T_s^{(t),a}$. O cancelamento de uma da cirurgia ocorre para cirurgias de $S_M^{(t)}$ que forem desmarcadas e não remarcadas. A variável $d_s^{(t)}$ indica o cancelamento de uma cirurgia $s \in S_M^{(t)}$ previamente agendada.

A Tabela 12 resume o comportamento da variável $y_{rsk}^{(t)}$. No caso da remarcação, em algum estágio t' anterior a t , $1 \leq t' \leq t - 1$, a cirurgia s havia sido agendada para iniciar em um período $\tau_s \geq t$ na sala $\eta_s \in R_s$ e a decisão atualiza os parâmetros $\tau_s e \eta_s$. No caso de cancelamento, a cirurgia sai do sistema e os parâmetros $\tau_s e \eta_s$ não são mais definidos. Uma cirurgia $s \in S_M^{(t)}$ se inicia quando $\tau_s = t$ e não há cancelamento ($d_s^{(t)} = 0$) ou quando estava agendada para $\tau_s > t$ é remarcada para iniciar em t , i.e., $\exists r \in R_s : y_{rst} = 1$. Uma cirurgia $s \in S_M^{(t)}$ mantém seu agendamento inalterado quando $y_{rsk}^{(t)} = 0, \forall r \in R_s, \forall k \in T_s^{(t),a}$ e $d_s^{(t)} = 0$. Uma cirurgia $s \in S_D^{(t)}$ se inicia logo que chega, isto é, no próprio período t , se $y_{rst}^{(t)} = 1$ para algum $r \in R_s$. Uma cirurgia $s \in S_D^{(t)}$ é rejeitada (não agendada) se $y_{rsk}^{(t)} = 0, \forall r \in R_s, \forall k \in T_s^{(t),a}$.

Agendamento	Remarcação	Cancelamento
$s \in S_D^{(t)}$	$s \in S_M^{(t)}$	$s \in S_M^{(t)}$
$y_{rsk}^{(t)} = 1$	$y_{\eta_s s \tau_s}^{(t)} = 1$	$y_{\eta_s s \tau_s}^{(t)} = 1$
$k \geq t; r \in R_s$	$y_{rsk}^{(t)} = 1$	$d_s^{(t)} = 1$
	$r \neq \eta_s \text{ e/ou } k \neq \tau_s$	

Tabela 12 – Usos da variável $y_{rsk}^{(t)}$

Em programação dinâmica o conjunto de todas as ações realizadas em cada um dos estágios de decisão representa uma política. Na seção 6.2.1 o termo política é definido e discutido. Denotaremos como $u^{(t)}$ o conjunto de todas as variáveis de decisão do estágio $t \in \{0\} \cup T$. A união de todas as ações realizadas, $\pi = \bigcup_{t \in \{0\} \cup T} u^{(t)}$, define a política implementada. O conjunto $\Xi^{(t)}$ contém os valores viáveis das variáveis de decisão de $u^{(t)}$.

A seguir formalizaremos o processo decisório através de um modelo de otimização para cada um dos estágios $t \in \{0\} \cup T$.

6.1.3.1 Transição

A dinâmica do processo é descrita pelo ciclo realização da incerteza ($\omega^{(t)} \rightarrow$ obtenção da informação ($x^{(t)} \rightarrow$ Decisão ($u^{(t)} \rightarrow$ realização da incerteza ($\omega^{(t+1)}$) e assim por diante. O processo se inicia com a obtenção da informação em $t = 0$, isto é, o conhecimento do estado inicial do sistema $x^{(0)}$. O estado inicial é composto pelas cirurgias eletivas e possui todos os recursos e salas disponíveis em todos os períodos.

O estado do sistema sofre alteração de um estágio para o outro. A transição do sistema de t para $t+1$ se dá através da intervenção por meio da decisão ($u^{(t)}$) e da realização de incertezas ($\omega^{(t+1)}$). Utilizaremos a notação $x^{(t+1)} = f_t(x^{(t)}, u^{(t)}, \omega^{(t+1)})$ para representar a transição do sistema, mas cada um dos parâmetros do estado possui uma regra própria para a transição.

Os conjuntos R , Q , C , MP e R_c não sofrem alteração ao longo dos períodos, isto é, são estáticos. Com o passar do tempo, todos os conjuntos de períodos (janelas de tempo) são atualizados para excluir o período imediatamente anterior.

Os parâmetros indicadores $L_r^{(t,k)}$ são atualizados para os períodos de informação conhecida, isto é, $\{t+1, t+2\}$ de forma que seja igual a 1 para todas as salas $r \in R$ que estarão certamente em uso nos períodos $t+1$ e $t+2$ ($\exists s \in S_I^{(t+1)} : \eta_s = r$).

De maneira semelhante, os indicadores $L_p^{(t+1)}$ são atualizados de forma que seja igual a 1 para todas os preceptores $p \in MP$ que estarão certamente atuando no período $t+1$ ($\exists s \in S_I^{(t+1)} \cap S_p^{(t)}$).

A Figura 4 apresenta o espaço de estados de uma cirurgia no sistema. Os estados D , M , I são representados no modelo pelos conjuntos $S_D^{(t)}$, $S_M^{(t)}$ e $S_I^{(t)}$, respectivamente. Além destes existe também o estado Fn que representa o ambiente externo ao sistema. O conjunto de cirurgias disponíveis para o agendamento no estágio t , $S_D^{(t)}$ depende apenas da realização da incerteza referente à chegada de novas cirurgias. Uma vez que a cirurgia chega, esta é encaminhada para o estado D . Essa transição está representada na Figura 4 pelo arco p_1 . As cirurgias pertencentes $S_D^{(t)}$ que não tiverem sido agendadas saem do sistema (p_9) enquanto as agendadas podem passar a integrar o conjunto $S_M^{(t+1)}$ (p_2) ou $S_I^{(t+1)}$ (p_3). Além das cirurgias que chegaram e que tiverem sido agendadas para iniciar em $k \geq t+1$ (p_2), as cirurgias que tiverem sido agendadas em períodos anteriores (incluindo as que tiverem sido remarcadas) e que não tiverem sido canceladas ou iniciadas em t ($\{s \in S_M^{(t)} : d_s^{(t)} = 0 \ \& \ \tau_s \neq t\}$) (p_4) compõem também o conjunto $S_M^{(t+1)}$. O conjunto $S_I^{(t+1)}$ é formado pelas cirurgias que haviam sido iniciadas em períodos anteriores, mas que não tiverem sido finalizadas ao final do período t ($\{s \in S_I^{(t)} : \tau_s + \delta_s - 1 > t\}$, arco p_7), pelas cirurgias de $S_M^{(t)}$ que tiverem sido iniciadas no período t ($\{s \in S_M^{(t)} : \tau_s = t \ \& \ d_s^{(t)} = 0\}$, p_5) e pela cirurgias que chegaram e logo se iniciaram em t ($\{s \in S_D^{(t)} : \sum_{r \in Rs} y^{(t)}rst = 1\}$, p_3). As cirurgias que tiverem sido canceladas ($\{s \in S_M^{(t)} : d_s^{(t)} = 1\}$) saem do sistema (p_6) bem como as cirurgias que tiverem sido finalizadas ($\{s \in S_I^{(t)} : \tau_s + d_s - 1 = t\}$, arco p_8).

A Tabela 13 ilustra lógica da dinâmica de transição em um exemplo simples. O exemplo se inicia no período t_2 quando há 5 cirurgias no sistema ($S^{(2)} = \{SE_1, SE_2, SE_3, SNE_1, SNE_2\}$), das quais três são eletivas (SE_1, SE_2, SE_3) e duas não eletivas (SNE_1, SNE_2). A cirurgia SE_1 , de responsabilidade do cirurgião c_1 , já iniciou no período t_1 na sala r_2 e possui duração esperada de 2 períodos. A cirurgia SNE_2 é uma cirurgia urgente que chegou ao sistema no intervalo $[1, 2)$ e é

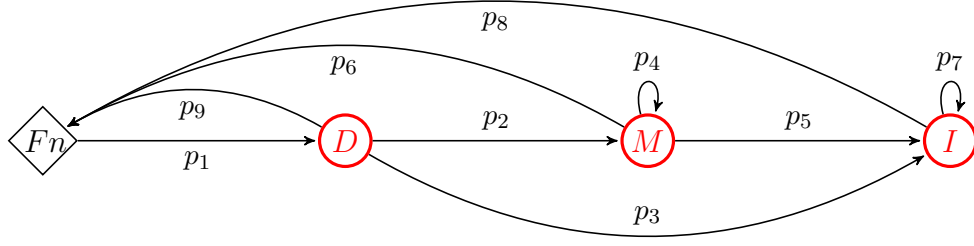


Figura 4 – Espaço de estados para o *status* de uma cirurgia

de responsabilidade do cirurgião c_3 . Neste exemplo, as cirurgias urgentes possuem uma janela de tempo de 3 períodos após sua chegada para iniciar. No conjunto de cirurgias agendadas estão três cirurgias dentre as quais 2 são eletivas e 1 é não eletiva, $S_M^{(2)} = \{SE_2, SE_3, SNE_1\}$. Entre parênteses, ao lado da identificação das cirurgias, está informação a respeito do cirurgião, da sala de agendamento e do período programado para o início, (c, η_s, τ_s) .

A tabela interna à primeira coluna da Tabela 13 representa o conjunto $S^{(2)}$ e no quadro ao lado é apresentado o sequenciamento das cirurgias. Em amarelo está representado a duração das cirurgias e em azul o período de limpeza (rótulo 'L'). A informação da tabela e do quadro definem o estado $x^{(2)}$.

A decisão tomada em t_2 é descrita por $u^{(2)}$ e determina o agendamento da cirurgia SNE_2 para a sala r_1 no período t_2 , ou seja, iniciar a cirurgia logo que chegou. Para viabilizar este agendamento, a cirurgia SE_2 foi remarcada para t_5 na mesma sala r_1 , e a cirurgia SE_3 , do cirurgião c_3 , foi remarcada t_5 na sala r_2 . A cirurgia SNE_1 começa em t_2 porque seu agendamento foi mantido.

A incerteza referente ao período t_3 é representada por $\omega^{(3)}$ e ao se realizar trouxe a informação da chegada da cirurgia não eletiva SNE_3 de responsabilidade do cirurgião c_2 e indicou que a cirurgia SE_1 , prevista para terminar ao final de t_2 , não terminou. Na tabela, destaca-se o período adicional com a cor rosa e o rótulo 'A'. O novo estado é apresentado em seguida, onde o fundo cinza representa os períodos que já passaram e atividades já realizadas.

A decisão $u^{(3)}$ determina o não agendamento da cirurgia SNE_3 , isto é, rejeita a

cirurgia. Os outros agendamentos são mantidos. A realização de $\omega^{(4)}$ ao se realizar trouxe a informação da chegada da cirurgia não eletiva SNE_4 de responsabilidade do cirurgião c_1 e o fim das cirurgias SE_1 e SNE_2 ao final do período t_3 .

Frente ao novo cenário (estado $x^{(4)}$), $u^{(4)}$ define o cancelamento da cirurgia SE_2 e o agendamento da cirurgia SNE_4 para a sala r_1 com início em t_5 . A realização de $\omega^{(5)}$ informa o fim da cirurgia SNE_1 ao final de t_4 . Por fim, o estado $x^{(5)}$ é apresentado.

$S_D^{(2)}$ $SNE_2(c_3)$	$S_M^{(2)}$ $SE_2(c_1, r_1, t_3)$ $SE_3(c_3, r_2, t_4)$ $SNE_1(c_2, r_3, t_2)$	$S_I^{(2)}$ $SE_1(c_1, r_2, t_1)$	r/t	1	2	3	4	5	6	7	8
r1							SE_2		L		
r2		SE_1	L					SE_3		L	
r3			SNE_1					L			

$u^{(2)}$: $y_{r_1, SNE_2, t_2} = 1$; $y_{r_1, SE_2, t_3} = 1$; $y_{r_1, SE_2, t_5} = 1$; $y_{r_2, SE_3, t_4} = 1$; $y_{r_2, SE_3, t_5} = 1$;
 $\omega^{(3)}$: Novas: $SNE_3(c_2)$ Fim: –

$S_D^{(3)}$ $SNE_3(c_2)$	$S_M^{(3)}$ $SE_2(c_1, r_1, t_5)$ $SE_3(c_3, r_2, t_5)$	$S_I^{(3)}$ $SE_1(c_1, r_2, t_1)$ $SNE_1(c_2, r_3, t_2)$ $SNE_2(c_3, r_1, t_2)$	r/t	1	2	3	4	5	6	7	8
r1					SNE_2	L			SE_2		L
r2		SE_1	A	L				SE_3			L
r3			SNE_1					L			

$u^{(3)}$: $y_{r, SNE_2, t} = 0, \forall r, \forall t$;
 $\omega^{(4)}$: Novas: $SNE_4(c_1)$ Fim: SE_1 ; SNE_2

$S_D^{(4)}$ $SNE_4(c_1)$	$S_M^{(4)}$ $SE_2(c_1, r_1, t_5)$ $SE_3(c_3, r_2, t_5)$	$S_I^{(4)}$ $SNE_1(c_2, r_3, t_2)$	r/t	1	2	3	4	5	6	7	8
r1					SNE_2	L			SE_2		L
r2		SE_1		L					SE_3		L
r3			SNE_1					L			

$u^{(4)}$: $y_{r_1, SE_2, t_5} = 1$; $d_{SE_2} = 1$; $y_{r_1, SNE_4, t_5} = 1$;
 $\omega^{(5)}$: Novas: – Fim: SNE_1

$S_D^{(5)}$ $SE_3(c_3, r_2, t_5)$ $SNE_4(c_1, r_1, t_5)$	$S_M^{(5)}$	$S_I^{(5)}$	r/t	1	2	3	4	5	6	7	8
r1					SNE_2	L		SNE_4	L		
r2		SE_1		L				SE_3		L	
r3			SNE_1					L			

Tabela 13 – Dinâmica de transição em um exemplo ilustrativo

6.1.3.2 Custos

O objetivo do problema modelado é a minimização de custos. Entendemos que uma vez que há um custo de não realização de cirurgias e que os custos, não

modelados, de contratação de recursos são fixos, a maximização da utilização está correlacionada com este objetivo.

Os custos gerados em um determinado estágio de decisão dependem do estado em que o sistema se encontra e da decisão tomada. A definição da ação a ser executada em um determinado estágio depende, no entanto, não só da minimização do custos gerados no estágio corrente, mas também da esperança dos custos que serão gerados em estágios futuros. Foram modelados três tipos de custos que incorrem durante a dinâmica do sistema:

- Custo de cancelamento (c_s^C) - custo fixo de cancelar o agendamento de uma cirurgia previamente agendada, $s \in S_M^{(t)}$. Incorre sempre que ação de cancelamento for realizada.
- Custo de rejeitar (c_s^{NR})- custo fixo de não realização de uma cirurgia $s \in S_D^{(t)}$. Incorre sempre que uma cirurgia não é agendada no período em que chega no sistema.
- Custo de início fora da janela de tempo ($c_{t,s}^{IF}$) - custo fixo de se inicializar uma cirurgia s em um período t fora de sua janela de tempo.

As equações ((6.1) - (6.3)) definem a forma de mensurar os custos gerados em um determinado estágio.

$$c_C^t(x, u) = \sum_{s \in S_M^{(t)}} c_s^C d_s^{(t)}, \quad \forall t \in T \quad (6.1)$$

$$c_{NR}^t(x, u) = \sum_{s \in S_D^{(t)}} \left(1 - \sum_{r \in R_s} \sum_{k \in T_s^{(t),a}} y_{rsk}^{(t)} \right) c_s^{NR}, \quad \forall t \in \{0\} \cup T \quad (6.2)$$

$$c_{IF}^t(x, u) = \sum_{\substack{s \in S_M^{(t)}: \\ t \notin T_s^{(t)}}} \left[(1 - y_{\eta_s t}^{(t)}) 1_{[\tau_s=t]} + y_{\eta_s t}^{(t)} 1_{[\tau_s \geq t+1]} + \sum_{\substack{r \in R_s: \\ r \neq \eta_s}} y_{rst}^{(t)} \right] c_{t,s}^{IF}, \quad \forall t \in T \quad (6.3)$$

Na equação (6.1) o custo de cancelamento é definido como sendo a soma do custo fixo de cancelamento para todas as cirurgias de $S_M^{(t)}$ que tiverem sido canceladas, isto é, $\{s \in S_M^{(t)} | d_s = 1\}$. A equação (6.2) soma para todas as cirurgias disponíveis o custo de rejeição para as cirurgias não agendadas, isto é, $\{s \in S_D^{(t)} | \sum_{r \in R_s} \sum_{k \in T_s^{(t),a}} y_{rsk}^{(t)} = 0\}$. Em (6.3) o custo de iniciar uma cirurgia após o término de sua janela de tempo é computado para todas as cirurgias agendadas para iniciar em t que não tiverem sido canceladas e quando t estiver fora da janela de tempo. A função $1_{\square}$ é uma função que recebe o valor 1 sempre que a condição entre colchetes é satisfeita e zero em situação contrária. É necessário observar que $c_{IF}^t(x, u)$ é cobrado somente se a cirurgia for iniciada em t . Caso uma cirurgia $s \in S_M^{(t)}$ com $t \notin T_s^{(t)}$ já esteja agendada para iniciar em t , o primeiro termo garante a cobrança do custo quando o agendamento não é alterado e o terceiro termo garante a cobrança do custo em situações onde somente a sala é alterada. Se a cirurgia $s \in S_M^{(t)}$ com $t \notin T_s^{(t)}$ estiver agendada para iniciar em $k \geq t + 1$, a soma do segundo com o terceiro termo garante a cobrança do custo para uma eventual remarcação da cirurgia para início em t .

Uma vez que os custos dependem da decisão tomada, em todo o estágio $t \in \{1..T_{max}\}$, as notações $c_C^t(x, u)$, $c_{NR}^t(x, u)$ e $c_{IF}^t(x, u)$ referem-se aos custos gerados pela decisão u estando no estado x . O custo total de uma decisão u no estágio t e

é expresso pela equação (6.4). No estágio 0, temos $g_0(x, u) = c_{NR}^0(x, u)$, uma vez que $S_D^{(0)}$ é o conjunto de cirurgias eletivas e os demais conjuntos são vazios. No estágio terminal, $T_{max} + 1$, o custo residual é nulo $g_{T_{max}+1}(x, u) = 0$.

$$g_t(x, u) = C_C^t(x, u) + C_{NR}^t(x, u) + C_{IF}^t(x, u), \quad \forall t \in T \quad (6.4)$$

6.1.4 Problema de Programação Dinâmica Estocástica

Com base nas definições anteriores, a expressão (6.5) representa o custo total gerado, por uma política π durante todo o horizonte de planejamento. O objetivo do problema é minimizar a esperança deste custo, (6.6), onde Ξ representa a região de viabilidade das variáveis de decisão do conjunto π .

$$CT(\pi) = \sum_{t \in \{0\} \cup T} g_t(x^{(t)}, u^{(t)}) \quad (6.5)$$

$$\text{Objetivo: } \min_{\pi \in \Xi} E_{\Omega} [CT(\pi) | x^{(0)}] \quad (6.6)$$

Por se tratar de um problema dinâmico, são definidos, para cada estágio $t \in \{0\} \cup T$, os valores das funções $J^{(t)}(x^{(t)})$, expressos pelo conjunto de equações (6.7), que representam o custo mínimo esperado que ocorra até o fim do horizonte de planejamento T_{max} ao estar no estado $x^{(t)}$ no estágio t . Constantemente, o valor da função $J^{(t)}$ aplicado ao estado $x^{(t)}$ é referido somente por **valor da função**. No estágio terminal, $T_{max} + 1$, $J^{(T_{max}+1)}(x^{(T_{max}+1)}) = g^{(T_{max}+1)}(x^{(T_{max}+1)}) = 0$. O objetivo do problema passa, portanto, a ser encontrar o valor de $J^{(0)}(x^{(0)})$.

$$\begin{aligned} J^{(t)}(x^{(t)}) &= \min_{\pi(t:T_{max}) \in \Xi(t:T_{max})} E_{\omega^{(t+1)}} \left[g^{(t)}(x^{(t)}, u^{(t)}) + \sum_{k \in \{t+1:T_{max}\}} g_k(x^{(k)}, u^{(k)}) \right] \\ &= \min_{\pi(t:T_{max}) \in \Xi(t:T_{max})} \left\{ g^{(t)}(x^{(t)}, u^{(t)}) + E_{\omega^{(t+1)}} \left[\sum_{k \in \{t+1:T_{max}\}} g_k(x^{(k)}, u^{(k)}) \right] \right\} \\ &= \min_{u^{(t)} \in \Xi^{(t)}} \left\{ g^{(t)}(x^{(t)}, u^{(t)}) + E_{\omega^{(t+1)}} [J^{(t+1)}(x^{(t+1)}) | x^{(t)}, u^{(t)}] \right\}, \\ &\quad \forall t \in \{0\} \cup T \end{aligned} \quad (6.7)$$

A expressão (6.7) é conhecida como **equação de Bellman**, onde $\Xi^{(t)}$ é o conjunto que representa a região de viabilidade para $u^{(t)}$ definido pelos conjuntos de inequações (6.8) a (6.14), que foram geradas a partir das limitações do sistema real.

$$\sum_{k \in T_s^{(t),a}} \sum_{\substack{r \in R_s : \\ L_{rk}^{(t)} = 0}} y_{rsk}^{(t)} \leq 1 \quad \forall s \in S_D^{(t)} \quad (6.8)$$

$$y_{\eta_s s \tau_s}^{(t)} - \sum_{k \in T_s^{(t),a}} \sum_{\substack{r \in R_s : \\ L_{rk}^{(t)} = 0, \\ (r,k) \neq (\eta_s, \tau_s)}} y_{rsk}^{(t)} = d_s^{(t)} \quad \forall s \in S_M^{(t)} \quad (6.9)$$

$$\begin{aligned} & \sum_{\substack{s \in S_M^{(t)} : \\ \tau_s \leq k, \\ \tau_s + \bar{\delta}_s - 1 \geq k}} (1 - y_{\eta_s s \tau_s}^{(t)}) a_{sq} + \sum_{s \in S_M^{(t)}} \sum_{\substack{h \leq k, \\ h + \bar{\delta}_s - 1 \geq k}} \sum_{\substack{r \in R_s : \\ L_{rh}^{(t)} = 0, \\ (r,h) \neq (\eta_s, \tau_s)}} y_{rsh}^{(t)} a_{sq} + \\ & \sum_{\substack{s \in S_D^{(t)} : \\ k \in T_s^{(t),a}}} \sum_{\substack{h \leq k, \\ h + \bar{\delta}_s - 1 \geq k}} \sum_{\substack{r \in R_s : \\ L_{rh}^{(t)} = 0}} y_{rsh}^{(t)} a_{sq} + \sum_{\substack{s \in S_I^{(t)} : \\ \tau_s + \bar{\delta}_s - 1 \geq k}} a_{sq} \leq H_q \quad \forall q \in Q, \\ & \quad \forall k \geq t \end{aligned} \quad (6.10)$$

$$\begin{aligned} & \sum_{\substack{s \in S_M^{(t)} \cap S_p^{(t)} : \\ \tau_s \leq k, \tau_s + \bar{\delta}_s - 1 \geq k}} (1 - y_{\eta_s s \tau_s}^{(t)}) + \sum_{s \in S_M^{(t)} \cap S_p^{(t)}} \sum_{\substack{h \leq k, \\ h + \bar{\delta}_s - 1 \geq k}} \sum_{\substack{r \in R_s : \\ L_{rh}^{(t)} = 0, \\ (r,h) \neq (\eta_s, \tau_s)}} y_{rsh}^{(t)} + \\ & \sum_{\substack{s \in S_D^{(t)} \cap S_p^{(t)} : \\ k \in T_s^{(t),a}}} \sum_{\substack{h \leq k, \\ h + \bar{\delta}_s - 1 \geq k}} \sum_{\substack{r \in R_s : \\ L_{rh}^{(t)} = 0, \\ r = \eta_s, h \neq \tau_s}} y_{rsh}^{(t)} + \sum_{s \in S_I^{(t)} \cap S_p^{(t)}} 1_{[\tau_s + \bar{\delta}_s - 1 \geq k]} \leq 1 \quad \forall p \in MP, \\ & \quad \forall k \geq t \end{aligned} \quad (6.11)$$

$$\begin{aligned} & \sum_{\substack{s \in S_M^{(t)} : \\ \eta_s = r, \tau_s \leq k, \\ \tau_s + \bar{\delta}_s - 1 \geq k}} (1 - y_{rs\tau_s}^{(t)}) + \sum_{\substack{h \leq k, \\ h + \bar{\delta}_s - 1 \geq k}} \sum_{\substack{s \in S_M^{(t)} : \\ (r,h) \neq (\eta_s, \tau_s), r \in R_s}} y_{rsh}^{(t)} + \\ & \sum_{\substack{h \leq k, \\ h + \bar{\delta}_s - 1 \geq k}} \sum_{\substack{s \in S_D^{(t)} : \\ r \in R_s \\ h \in T_s^{(t),a}}} y_{rsh}^{(t)} + \sum_{s \in S_I^{(t)}} 1_{[\tau_s + \bar{\delta}_s \geq k]} \leq 1 \quad \forall r \in R, \\ & \quad \forall k \geq t \end{aligned} \quad (6.12)$$

$$\begin{aligned} & y_{rsk}^{(t)} \in \{0, 1\} \quad \forall s \in S^{(t)}, \\ & \quad \forall r \in R_s, \quad \forall k \in T_s^{(t),a} \end{aligned} \quad (6.13)$$

$$d_s^{(t)} \in \{0, 1\} \quad \forall s \in S^{(t)} \quad (6.14)$$

As inequações (6.8) definem que o início de uma cirurgia disponível poderá ser agendada no máximo uma vez. As igualdades (6.9) coordenam o processo de remarcação ou cancelamento de uma cirurgia já agendada. As inequações (6.10) impedem a sobreposição no uso de recursos durante todo o horizonte de planejamento. O conjunto de inequações (6.11) garante, para todo período de planejamento, que duas cirurgias pertencentes ao mesmo preceptor não serão executadas simultaneamente. Já o conjunto de restrições (6.12) garante a não sobreposição do uso das salas. Em (6.11) e (6.12), a função $1_{\square}$ assume o valor 1 se a condição informada entre colchetes for satisfeita e 0 caso contrário. As restrições (6.13) e (6.14) definem o domínio das variáveis.

A priori não se conhece garantia de otimalidade da política de se proibir a sobreposição de recursos durante todo o horizonte de planejamento. Supõe-se que possa haver situações onde o *overbooking* gere um maior aproveitamento dos recursos dependendo da realização das incertezas.

No entanto, o modelo (6.8)-(6.14) não permite que seja feito o *overbooking* de recursos/salas em períodos futuros. A apresentação do modelo desta maneira se deve ao fato de que testes preliminares apontaram que os resultados obtidos sem *overbooking* foram muito superiores aos que permitiam *overbooking*. Na seção 6.2.2 é apresentada uma versão do modelo que permite a sobreposição de recursos.

Conforme já explicado na seção 6.1.3 e sumarizado na Tabela 12, é importante observar que as variáveis binárias $\{y_{rsk}^{(t)}\}$ podem ter dois diferentes significados. Quando $s \in S_D^{(t)}$, a atribuição do valor 1 para a variável significa o agendamento da cirurgia s para a sala r iniciando no período k . Para $s \in S_M^{(t)}$, $y_{\eta_s s \tau_s}^{(t)}$ assume 1 se há remarcação ou cancelamento da cirurgia s . No primeiro caso, $y_{rsk}^{(t)}$ assume 1 para algum $r \in R_s$ e $k \geq t$ tal que $r \neq \eta_s$ e/ou $k \neq \tau_s$. No segundo caso, $d_s^{(t)}$ assume 1. Note que se para uma cirurgia $s \in S_M^{(t)}$ é mantido um agendamento previamente realizado para o período τ_s na sala η_s , então $\{y_{rsk}^{(t)}\} = 0$, $\forall r \in R_s$ e $\forall t \in T_s^{(t),a}$. A Figura 5 e a Tabela 6.1.4 descrevem as possíveis transições de estado de uma cirurgia definido pelas variáveis de decisão. A ocorrência das transições representadas pelos arcos p_1 , p_7 e p_8 (apresentados anteriormente na Figura 4) não dependem da decisão e sim da realização das incertezas e, portanto, estes arcos não foram representados na Figura 5.

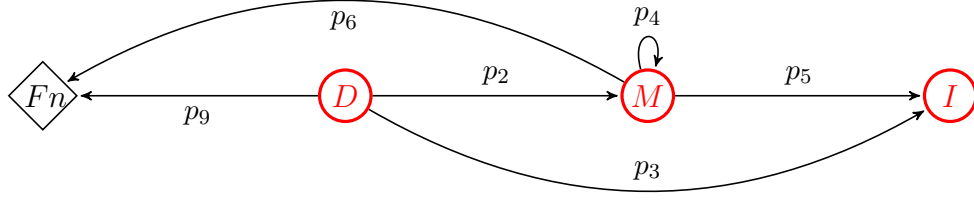


Figura 5 – Possíveis transições de uma cirurgia causadas por decisões

Arco de transição	Condição
p_2	$\sum_{r \in R_s} \sum_{k \in T_s^{(t),a}} y_{rsk}^{(t)} = 1$
p_3	$\sum_{r \in R_s} y_{rst}^{(t)} = 1$
p_4	$d_s^{(t)} = 0$
p_5	$(1 - y_{\eta_s st}^{(t)})1_{[\tau_s=t]} + y_{\eta_s st}^{(t)}1_{[\tau_s \geq t+1]} + \sum_{r \in R_s: r \neq \eta_s} y_{rst}^{(t)} = 1$
p_6	$d_s^{(t)} = 1$
p_9	$y_{rsk}^{(t)} = 0, \forall k \in T_s^{(t),a}, \forall r \in R_s$

Tabela 14 – Possíveis transições de uma cirurgia causadas por decisões

6.2 Abordagens para a solução do problema

6.2.1 Discussão e Complexidade

A programação dinâmica estocástica é conhecida por sofrer da maldição da dimensão (*curses of dimensionality*), o que dificulta a avaliação por métodos exatos (no caso, através do algoritmo *Backward*). No caso específico do problema tratado, para cada cirurgia s disponível para o agendamento, teríamos um espaço de solução da ordem $Dim(s \in S_D) = |R_s| \times |T_s^a| + 1$ que representa as possibilidades de agendamento acrescidas da possibilidade de não agendar. Para cada cirurgia já agendada, teríamos um espaço de solução da ordem de $Dim(s \in S_M) = |R_s| \times |T_s^a|$ que consiste na possibilidade de cancelar acrescida da possibilidade de remarcar em uma configuração alternativa. A cada estágio, o número de cirurgias disponíveis para o agendamento dependeria da chegada, ou não, de cada um dos n diferentes tipos de cirurgias, totalizando 2^n diferentes possibilidades, simplificando que apenas uma cirurgia de cada tipo poderia chegar. Para cada cirurgia iniciada, teríamos

um número de caminhos igual a $|\omega_{d_s}|$, os quais deveriam ser combinados em cada ramo de diferentes configurações de chegada, com cada ramo de possibilidade de decisão de $s \in S_D$ e de $s \in S_M$. Para se obter a quantidade final de caminhos na árvore, devemos considerar a multiplicação das respectivas quantidades em cada estágio de decisão. Isto torna inviável, por exemplo, a avaliação através de método exato para uma instância que possua dois tipos de cirurgias, duas salas idênticas e quatro períodos de tempo.

Neste trabalho, buscou-se desenvolver procedimentos heurísticos aproximados para a obtenção de políticas para a solução do problema. A importância da definição formal do modelo de programação dinâmica estocástica se dá para poder comparar os diversos resultados dentro de um mesmo *framework*.

Foram desenvolvidos três procedimentos para obtenção destas políticas: um procedimento que define uma política míope baseada no custo do estado presente; um procedimento que define uma política do tipo *Lookahead* e um procedimento que gera uma política híbrida do tipo *Lookahead* e baseada na aproximação do valor da função.

O procedimento tipo *Lookahead* desenvolvido é uma abordagem que se baseia na execução sucessiva da versão estática e determinística do problema diário de sequenciamento de cirurgias. No procedimento baseado na aproximação do valor da função, a função $J^{(t)}$ é aproximada por um modelo de regressão linear. Os coeficientes desta regressão são estimados a partir de uma simulação. Uma vez que os coeficientes foram estimados, um modelo de programação inteira que possui a aproximação de $J^{(t)}$ como parte da função objetivo é executado para encontrar a solução. Os procedimentos desenvolvidos estão descritos nas seções a seguir.

6.2.2 Procedimento para geração de uma Política Míope

O procedimento consiste em, a cada período, decidir de forma a minimizar a soma dos custos de cancelamento, de não realização de cirurgias e de início fora da janela de tempo (6.15) sujeito às restrições impostas pelo conjunto de inequações (6.8), (6.9), (6.13) e (6.14), acrescidas das inequações (6.16) - (6.18), região $\Xi_M^{(t)}$. O

período t é o estágio corrente em que é tomada a decisão.

$$\begin{aligned} \text{Objetivo: } \min_{u \in \Xi_M^{(t)}} g_t^M(u) = & C_C^t(x^{(t)}, u) + C_{NR}^t(x^{(t)}, u) + \\ & C_{IF}^t(x^{(t)}, u), \quad \forall t \in \{0\} \cup T \end{aligned} \quad (6.15)$$

$$\begin{aligned} \sum_{\substack{s \in S_M^{(t)} : \\ t = \tau_s}} (1 - y_{\eta_s st}^{(t)}) a_{sq} + \sum_{s \in S_M^{(t)}} \sum_{\substack{r \in R_s : \\ L_{rt}^{(t)} = 0, \\ (r, t) \neq (\eta_s, \tau_s)}} y_{rst}^{(t)} a_{sq} + \\ \sum_{\substack{s \in S_D^{(t)} : \\ t \in T_s^{(t),a}}} \sum_{\substack{r \in R_s : \\ L_{rt}^{(t)} = 0}} y_{rst}^{(t)} a_{sq} \leq H_q^{(t)} \quad \forall q \in Q \end{aligned} \quad (6.16)$$

$$\begin{aligned} \sum_{\substack{s \in S_M^{(t)} \cap S_p^{(t)} : \\ t = \tau_s}} (1 - y_{\eta_s st}^{(t)}) + \sum_{s \in S_M^{(t)} \cap S_p^{(t)}} \sum_{\substack{r \in R_s : \\ L_{rt}^{(t)} = 0, \\ (r, t) \neq (\eta_s, \tau_s)}} y_{rst}^{(t)} + \\ \sum_{\substack{s \in S_D^{(t)} \cap S_p^{(t)} : \\ t \in T_s^{(t),a}}} \sum_{\substack{r \in R_s : \\ L_{rt}^{(t)} = 0}} y_{rst}^{(t)} \leq 1 - L_p^{(t)} \quad \forall p \in MP \end{aligned} \quad (6.17)$$

$$\begin{aligned} \sum_{\substack{s \in S_M^{(t)} \cap S_p^{(t)} : \\ k = \tau_s, r = \eta_s}} (1 - y_{rsk}^{(t)}) + \sum_{\substack{s \in S_M^{(t)} : \\ k \neq \tau_s, r = \eta_s}} y_{rsk}^{(t)} + \\ \sum_{\substack{s \in S_D^{(t)} : \\ r \in R_s : \\ k \in T_s^{(t),a}}} y_{rsk}^{(t)} \leq 1 - L_{rk}^{(t)} \quad \forall r \in R \\ \forall k \in \{t, t+1\} \end{aligned} \quad (6.18)$$

$$(6.19)$$

Para o período corrente não poderá haver sobreposição no uso de recursos (6.16). O conjunto de inequações (6.17) garantem, para o período atual, que duas cirurgias pertencentes ao mesmo preceptor não serão executadas simultaneamente. O conjunto de restrições (6.18) garante a não sobreposição do uso das salas. No período de informação conhecida, t e $t+1$ devido à limpeza de salas, o método não impede que cirurgias sejam agendadas compartilhando recursos, isto é, ela permite o *overbooking*.

O método é míope uma vez que não considera o impacto da decisão nos estágios futuros, visto que a função objetivo considera apenas os custos incorridos no estágio

corrente de decisão, e é também guloso por ser guiado por uma representação incompleta do objetivo real do problema. Para a execução do método, as informações futuras são aproximadas. No caso da duração das cirurgias, utiliza-se o valor esperado e, no processo de chegada, considera-se que não chegarão cirurgias não-eletivas no futuro.

6.2.3 Procedimento para geração de uma Política do tipo *Lookahead*

O procedimento consiste em, a cada período, decidir de forma a minimizar a soma dos custos de cancelamento, de não realização de cirurgias e de início fora da janela de tempo (6.20) em todo o horizonte de planejamento, isto é, a cada estágio executa-se um novo problema estático com a informação atualizada. Assim como no caso do procedimento Míope, as informações futuras são aproximadas. No caso da duração das cirurgias, utiliza-se o valor esperado de sua duração. O procedimento ignora a chegada de cirurgias não eletivas no futuro. O objetivo é a minimização do custos aplicados sobre os valores esperados a partir da aproximação das informações futuras para todo o horizonte de tempo (6.20).

O método é guloso por ser guiado por uma representação incompleta do objetivo do problema já que não considera as incertezas e a possibilidade do gerenciamento dinâmico. A região de viabilidade em um estágio t é dada pelos conjuntos de inequações (6.8) - (6.14), região $\Xi_L^{(t)}$. Assim, o método não permite que cirurgias sejam pré-agendadas compartilhando recursos, isto é, ela impede o *overbooking*.

$$\text{Objetivo: } \min_{u \in \Xi_L^{(t)}} g_t^L(x^{(t)}, u) = \sum_{k \in T^t} C_C^t(x^{(t)}, u) + C_{NR}^t(x^{(t)}, u) + C_{IF}^t(x^{(t)}, u), \quad \forall t \in \{0\} \cup T \quad (6.20)$$

Este procedimento define a política de reagir às incertezas resolvendo, a cada período, o problema estático e determinístico para os períodos subsequentes.

6.2.4 Algoritmo de aproximação de programação dinâmica para definição de uma política baseada na aproximação do valor da função $J^{(t)}()$

A ideia básica dos procedimentos de aproximação de programação dinâmica é desenvolver um algoritmo *Foward* que gera uma política subótima a partir da solução de subproblemas iniciais. No caso da construção de procedimentos capazes de definir políticas aproximadas baseadas na aproximação do valor da função $J^{(t)}()$, um dos desafios é definir uma forma de estimar $J^{(t+1)}()$.

Os procedimentos desenvolvidos são versões do *Policy Iteration Algorithm* (PIA) **aproximado** e a lógica destes algoritmos é avaliar uma dada política e utilizar os valores observados durante tal avaliação para aproximar $J^{(t)}()$ e encontrar uma nova política ((POWELL, 2011),(BERTSEKAS, 2012)). A Tabela 15 resume de maneira textual as etapas dos procedimentos desenvolvidos.

As políticas definidas por esse algoritmo consistem em, a cada período, decidir de forma a minimizar a soma dos custos de cancelamento, de não realização de cirurgias e de início fora da janela de tempo acrescidos à estimativa $\overline{J^{(t+1)}}(x^{(t)}, u)$ do valor de $J^{(t+1)}(x^{(t+1)})$ (6.21) em todo o horizonte de planejamento.

$$\text{Objetivo: } \min_{u \in \Xi_L^{(t)}} g_t^{PIA}(x^{(t)}, u) = C_C^t(x^{(t)}, u) + C_{NR}^t(x^{(t)}, u) + C_{IF}^t(x^{(t)}, u) + \overline{J^{(t+1)}}(x^{(t)}, u), \quad \forall t \in \{0\} \cup T \quad (6.21)$$

A região de viabilidade em um estágio t é dada pelos conjuntos de inequações (6.8) - (6.14), região $\Xi_L^{(t)}$. Observe que embora a região de viabilidade seja a mesma do método *Lookahead* a função objetivo possui uma aproximação do valor de $J^{(t+1)}$ o que torna os dois métodos diferentes.

O valor de $J^{(t+1)}(x^{(t+1)})$ foi aproximado através de uma regressão linear múltipla entre informações obtidas a partir do estado e valor observado das funções. Estas estatísticas são coletadas através da simulação dos caminhos aleatórios w .

O estado pós-decisão $x_u^{(t)}$ (*post decision state*) é definido como o estado do sistema após a aplicação da ação $u^{(t)}$ e anteriormente à realização da incerteza $\omega^{(t+1)}$. São

gerados M caminhos e para cada um destes é coletado o valor observado de $J^{(t+1)}$ (denominado $\hat{j}^{(t+1)}$) e as informações a respeito do estado pós-decisão (obtidos através das funções indicadoras).

Utilizamos as chamadas **funções indicadoras** para transformar a informação do estado pós-decisão em indicadores (escalar). Uma função indicadora é utilizada para mapear algum aspecto da variável de estado $x^{(t)}$ em um escalar ($\phi_f^{(t)}() : \mathcal{X}^t \rightarrow \mathbb{R}$). Este escalar, por sua vez, será utilizado para o valor de $J^{(t+1)}$ que aproxima $J^{(t+1)}$. Por exemplo, uma das funções indicadoras utilizadas neste trabalho é o custo futuro de início fora da janela de tempo para agendamentos realizados pela decisão $u^{(t)}$. O conjunto das funções indicadoras f é representado por $\mathcal{F}$ e na subseção 6.2.4.1 estas serão detalhadas.

Na relação (6.22), $\theta_f^{(t)}$ representam os coeficientes da regressão linear múltipla entre o valor $\hat{j}^{(t+1)}$ observado para a função $J^{(t+1)}$ e os indicadores calculados e $\theta_0^{(t)}$ representa a constante do modelo linear. Destaca-se que, no algoritmo utilizado, o que define uma política são os valores dos coeficientes utilizados. Na seção (6.2.4.2) os dois métodos alternativos utilizados para estimação destes coeficientes são descritos e posteriormente, na seção 6.2.4.1, são apresentados os indicadores utilizados nos procedimentos desenvolvidos. A ação $u^{(t)*}$ é a decisão ótima para o problema de programação inteira do estágio t (6.21), onde $x_u^{(t)}$ é o estado pós-decisão.

$$\overline{J^{(t+1)}}(x_u^{(t)}) = \theta_0^{(t)} + \sum_{f \in \mathcal{F}} \theta_f^{(t)} \phi_f^{(t)}(x^{(t)}, u^{(t)*}), \quad \forall t \in \{0\} \cup T \quad (6.22)$$

Para cada política são gerados M caminhos aleatórios para estimar o valor dos coeficientes e o PIA testa N políticas (número máximo de iterações da **Etapa 3** da Tabela 15). Ao todo são gerados $N \times M$ caminhos para estimar o valor dos coeficientes a serem utilizados. Os métodos alternativos utilizados para a estimação dos coeficientes geradores das políticas alteram levemente a estrutura do pseudocódigo apresentado. Estas diferenças, no entanto, serão apresentadas na seção (6.2.4.3) onde as etapas dos algoritmos são detalhadas.

Etapla 0:	Definir coeficientes da política inicial
Etapla 1:	a. Coletar estatísticas do desempenho da política através de simulação b. Atualizar estimativa dos coeficientes a partir das estatísticas coletadas
Etapla 2:	Definir nova política
Etapla 3:	Se número máximo de iterações tiver sido atingido, retornar política; senão, voltar à Etapla 1

Tabela 15 – Visão geral dos procedimentos desenvolvidos

6.2.4.1 Funções indicadoras

As funções indicadoras são funções utilizadas para mapear algum aspecto da variável de estado $x^{(t)}$ em um escalar ($\phi_f^{(t)}() : \mathcal{X}^t \rightarrow \mathbb{R}$). Caso os valores dos indicadores sejam correlacionados com o valor da função $J^{(t)}()$, pode-se tentar utilizá-los para realizar a aproximação da $J^{(t)}()$. A utilização de regressão linear múltipla para aproximação do valor da função gera melhores resultados para indicadores com maior correlação com $J^{(t)}()$ e que sejam independentes entre si. Com o objetivo de fomentar a busca por uma boa decisão em t , as funções indicadoras para $t + 1$ foram definidas em função do estado presente $x^{(t)}$ e da decisão tomada $u^{(t)}$ ($\phi_f^{(t+1)}() : \mathcal{X}^t \times \Xi^t \rightarrow \mathbb{R}$). Naturalmente, a partir da definição destas funções, é possível também obter o valor do indicador somente a partir do estado quando necessário. Foram utilizadas três funções indicadoras:

- **Custo futuro de início fora da janela de tempo** ($\phi_{if}^{(t)}()$) - O agendamento realizado pela decisão $u^{(t)}$ pode alocar cirurgias para períodos fora da janela de tempo. O valor deste indicador é calculado pela equação (6.23). A ideia deste indicador é, justamente, avaliar o peso deste tipo de agendamento nos custos futuros.

$$\phi_{if}^{(t)}(x^{(t)}, u^{(t)}) = \sum_{k>t} C_{IF}^{(t)}(x^{(t)}, u^{(t)}) \quad (6.23)$$

- **Média das taxas de desocupação** ($\phi_{td}^{(t)}()$) - Esse indicador considera a velocidade com que as cirurgias existentes deixariam o sistema para um dado agendamento. A ideia por trás deste é medir a taxa de ocupação das salas

em um determinado estado e a velocidade com que estas passam a ser desocupadas. Um sistema congestionado não é capaz de atender urgências e, por outro lado, a ociosidade em períodos iniciais pode gerar congestionamentos no futuro.

Para cada período posterior a t é calculada a taxa média de desocupação por período. A taxa média de desocupação consiste em subtrair, da razão entre a duração total das cirurgias que estão no sistema no período t e a capacidade das salas até o fim do dia, a razão entre a duração total residual das cirurgias que estão planejadas para estar no sistema no período k e a capacidade das salas entre k e T_{max} . Para todo $k > t$, a diferença é dividida por $k - t$. Os valores obtidos são somados para encontrar o indicador.

As variáveis l_k^M e l_k^D calculam a duração total das cirurgias presentes no sistema no período k em cada um dos conjuntos $S_M^{(k)}$ e $S_D^{(k)}$, respectivamente (6.26 e 6.27). Para o conjunto $S_I^{(k)}$, a equação (6.25) é utilizada para $k > t$ e a (6.24) para o período $k = t$. Em (6.24), as cirurgias que estiverem em $S_I^{(t)}$ mas que já deveriam ter terminado em t são computadas com duração igual a 1, as demais cirurgias, são computados os tempos residuais de espera. Em (6.25) são computados apenas os tempos residuais de espera. Em (6.26) e (6.27), considera-se a duração esperada da cirurgia no cálculo.

$$l_t^I = \sum_{s \in S_I^{(t)}} \min(1; \tau_s + \bar{\delta}_s - 1 - t) \quad (6.24)$$

$$l_k^I = \sum_{s \in S_I^{(k)}} (\tau_s + \bar{\delta}_s - 1 - k), \quad \forall k > t \quad (6.25)$$

$$l_k^M = \sum_{s \in S_M^{(k)}} (1 - d_s^{(k)}) \bar{\delta}_s, \quad \forall k \geq t \quad (6.26)$$

$$l_k^D = \sum_{s \in S_D^{(k)}} \sum_{r \in R_s} \sum_{h \geq k} y_{srh}^{(k)} \bar{\delta}_s, \quad \forall k \geq t \quad (6.27)$$

O congestionamento do sistema no período k , l_k é dado pela razão entre a soma das durações totais das cirurgias presentes no sistema e a capacidade

residual das salas até o fim do dia, equação (6.28). A equação (6.29) é usada para calcular o indicador $\phi_{td}^{(t)}(x^{(t)}, u^{(t)})$.

$$l_k = \frac{l_k^I + l_k^M + l_k^D}{|R|(T_{max} - k)}, \quad \forall k \geq t \quad (6.28)$$

$$\phi_{td}^{(t)}(x^{(t)}, u^{(t)}) = \sum_{k > t} \left(\frac{l_t - l_k}{k - t} \right) \quad (6.29)$$

Quanto maior o valor de $\phi_{td}^{(t)}(x^{(t)}, u^{(t)})$ maior a velocidade média de desocupação do sistema. Em alguns casos pode ser interessante dar preferência para agendamentos que utilizem melhor os recursos das salas nos períodos iniciais e que mantenham as salas desocupadas em períodos futuros para dar maior flexibilidade ao plano. Para esses casos, o indicador da média das taxas de desocupação pode ser útil.

- **Soma do uso de salas por cirurgias nos próximos K períodos ($\phi_{su}^{(t)}()$)**
- Esse indicador calcula o número de períodos que se espera utilizar de salas nos próximos K períodos. De cada período $k \in \{t, t+K\}$ soma-se o número de salas que se espera estar em uso, isto é, o número de cirurgias já iniciadas em t e que ainda não teriam terminado em k ($|\{s \in S_I^{(t)} | \tau_s + \bar{\delta}_s - 1 \geq k\}|$), representado por n_k^I , acrescido do número de cirurgias já agendadas para iniciar antes de k e que não teriam terminado em k ($|\{s \in S_M^{(t)} | (\tau_s \leq k) \cap (\tau_s + \bar{\delta}_s - 1 \geq k)\}|$), n_k^M , e do numero de novas cirurgias agendadas para iniciar antes de k e que não teriam terminado em k ($|\{s \in S_D^{(t)} | \tau_s + \bar{\delta}_s - 1 \geq k\}|$), n_k^D .

$$\phi_{su}^{(t)}(x^{(t)}, u^{(t)}) = \sum_{k \in \{t, t+K\}} n_k^I + n_k^M + n_k^D \quad (6.30)$$

$$n_k^I = \sum_{s \in S_I^{(t)}} 1_{[\tau_s + \bar{\delta}_s - 1 \geq k]} \quad (6.31)$$

$$n_k^M = \sum_{l \in \{t, \dots, k\}} \sum_{\substack{s \in S_M^{(t)} : \\ l + \bar{\delta}_s - 1 \geq k}} \left(\sum_{\substack{r \in Rs : \\ L_{rl}^{(t)} = 0, \\ (\eta_s, \tau_s) \neq (r, l)}} y_{srl} + 1_{[\tau_s = l]}(1 - y_{s\eta_s \tau_s}) \right) \quad (6.32)$$

$$n_k^D = \sum_{l \in \{t, \dots, k\}} \sum_{\substack{s \in S_D^{(t)} : \\ l \in T_s^{(t),a} \\ l + \bar{\delta}_s - 1 \geq k}} \sum_{r \in R_s : L_{rl}^{(t)} = 0} y_{srl} \quad (6.33)$$

O termo n_k^I não depende da decisão $u^{(t)}$, isto é, é constante dado $x^{(t)}$, portanto não há necessidade de ser computado para se resolver o problema de otimização em t . Foi utilizado $K = 6$ para cobrir a janela de tempo das cirurgias urgentes que podem chegar em $t+1$. A ideia deste indicador é identificar a capacidade do conjunto de salas de absorver urgências do período seguinte.

6.2.4.2 Atualização dos coeficientes da aproximação linear

Foram testados dois métodos para atualização dos coeficientes da regressão linear utilizados para aproximar o valor da função $J^{(t)}()$ e, consequentemente, definir uma política para a iteração n do PIA (π_n). O capítulo 8 do livro Powell (2011) fornece uma boa apresentação destes métodos.

A primeira abordagem testada utiliza a suavização exponencial para a atualização do vetor de coeficientes após a aplicação de uma política (π_n). O coeficiente $\theta^{(t), \pi_n}$ é obtido através da equação (6.34), onde $\bar{\theta}^{(t), \pi_n}$ é o vetor com a estimativa dos coeficientes dos indicadores para o período t obtido via mínimos quadrados (equação (6.35)) a partir das estatísticas coletadas em π_n .

$$\theta^{(t), \pi_n} = (1 - \alpha_n) \theta^{(t), \pi_{n-1}} + \alpha_n \bar{\theta}^{(t), \pi_n} \quad (6.34)$$

O valor da constante α_n representa o *stepsize* utilizado para a atualização dos coeficientes após a execução da política π_n . Os valores de α_n devem estar no intervalo $[0, 1]$ e valores próximos de 1 atribuem maior peso para as observações recentes.

Seja ϕ_m^{t, π_n} um vetor $|\mathcal{F} + 1| \times 1$ iniciado com a constante 1 (referente ao $\theta_0^{(t), \pi_n}$) e com os demais valores obtidos pela aplicação das funções indicadoras $\phi_f^{(t)}()$ para o m -ésimo caminho aleatório de uma política π_n . A função indicadora é aplicada sobre o estado $x^{(t), m}$ após a ação $u^{(t), m^*}$ (estado pós-decisão) observado na realização

do m -ésimo caminho aleatório da política π_n . A matriz $\phi^{t,n}$ possui dimensões $|\mathcal{F} + 1| \times M$ com cada coluna é formada por um dos vetores ϕ_m^{t,π_n} ($m = 1..M$) e representa as estatísticas dos indicadores coletados durante a execução da política π_n . Seja $\hat{j}^{(t),\pi_n,m}$ o valor observado dos custos de se aplicar a política π_n no m -ésimo caminho aleatório; o vetor $\hat{j}^{(t),\pi_n}$ com dimensão M contém todas as estatísticas de custo para a política π_n . O método dos mínimos quadrados pode ser utilizado para estimar o valor de $\bar{\theta}^{(t),\pi_n}$ após a aplicação da política π_n , equação (6.35).

$$\bar{\theta}^{(t),\pi_n} = [\phi^{t,n} \phi^{t,n'}]^{-1} \phi^{t,n} \hat{j}^{(t),\pi_n} \quad (6.35)$$

Pelo método de suavização exponencial descrito acima, a atualização dos coeficientes é feita apenas uma vez ao final da execução dos M caminhos aleatórios de uma política. Um outro método de atualização dos coeficientes é o método dos mínimos quadrados recursivo (equações (6.36) - (6.39)).

A equação (6.36) é utilizada para atualizar o valor dos coeficientes após a realização de um caminho aleatório. A matriz $B_m^{(t),\pi_n}$ é uma matriz $|\mathcal{F} + 1| \times |\mathcal{F} + 1|$ que é atualizada recursivamente através da equação (6.37). γ_m^{t,π_n} é um escalar calculado recursivamente utilizando a equação (6.38). $\hat{\epsilon}_m^t$ representa a diferença entre o valor estimado da função $\overline{J^{(t+1)}}(x^{(t),m}, u^{(t),m*})$ e o valor observado $\hat{j}^{(t),\pi_n,m}$, equação (6.39).

$$\theta_m^{(t),\pi_n} = \theta_{m-1}^{(t),\pi_n} - \frac{1}{\gamma_m^{(t),\pi_n}} B_{m-1}^{(t),\pi_n} \phi_m^{(t),\pi_n} \hat{\epsilon}_m^t \quad (6.36)$$

$$B_m^{(t),\pi_n} = \frac{1}{\lambda} \left(B_{m-1}^{(t),\pi_n} - \frac{1}{\gamma_m^{(t),\pi_n}} \left(B_{m-1}^{(t),\pi_n} \phi_m^{(t),\pi_n} \phi_m^{(t),\pi_n'} B_{m-1}^{(t),\pi_n} \right) \right) \quad (6.37)$$

$$\gamma_m^{(t),\pi_n} = \lambda + \phi_m^{(t),\pi_n'} B_{m-1}^{(t),\pi_n} \phi_m^{(t),\pi_n} \quad (6.38)$$

$$\hat{\epsilon}_m^t = \overline{J^{(t+1)}}(x^{(t),m}, u^{(t),m*}) - \hat{j}^{(t),\pi_n,m} \quad (6.39)$$

A constante λ presente nas equações (6.37) e (6.38) funciona de um modo similar à uma constante *stepsize*, onde o valor $\lambda = 1$ é ótimo para quando o processo

é estacionário, isto é, quando é necessário colocar o mesmo peso para todas as observações. Valores próximos de zero colocam maior peso nas observações mais recentes.

A atualização é recursiva e é feita para $n > 1$. Para a política π_1 os coeficientes são atualizados apenas após as M execuções da simulação utilizando o método dos mínimos quadrados tradicional (equação (6.35)). O valor de $B_1^{(t),\pi_n}$ é obtido durante a atualização dos coeficientes de π_1 através da expressão (6.40).

$$B_1^{(t),\pi_n} = \left[\phi_1^{(t),\pi_n} \phi_1^{(t),\pi_n'} \right]^{-1} \quad (6.40)$$

Problemas numéricos podem ocorrer ao utilizar este procedimento. A matriz $B_m^{(t),\pi_n}$ pode ser singular e o valor de $\gamma_m^{(t),\pi_n}$, caso seja muito próximo de zero, pode gerar instabilidade no cálculo da equação (6.37). Seja ϵ uma pequena perturbação aleatória escalar positiva. Quando a matriz $B_m^{(t),\pi_n}$ é singular, é adicionado à esta a matriz $I\epsilon$ e quando $\gamma_m^{(t),\pi_n} \approx 0$, a perturbação é adicionada à constante (ver (POWELL, 2011)).

6.2.4.3 Approximate Policy Iteration Algorithm

Os Algoritmos 5 e 6 descrevem todas as etapas deste procedimento. A diferença entre os algoritmos consiste na forma de atualização dos coeficientes da regressão. O algoritmo 5 utiliza o método da suavização exponencial enquanto o algoritmo 6 utiliza o método dos mínimos quadrados recursivos. Os algoritmos serão referenciados como *PIASE* e *PIAMQR*, respectivamente.

No *PIASE*, o vetor de coeficientes $\theta^{(t),\pi_n}$ é obtido a partir da suavização exponencial, linha 12. Já o vetor $\bar{\theta}^{(t),\pi_n}$ é obtido a partir do método dos mínimos quadrados tradicional com base nas estatísticas de custo e dos valores coletados para os indicadores a cada caminho aleatório da política π_{n-1} . As estatísticas de custo são atualizadas a cada execução m da simulação de uma política (11), nesse caso $g_t^m()$ representa o custo incorrido em t da aplicação da decisão $u^{(t),m*}$ no estado $x^{(t),m}$.

No *PIAMQR*, a política π_1 é obtida através da regressão linear múltipla realizada em batelada após a execução dos M caminhos aleatórios da simulação (linha

15). As demais políticas são obtidas através da atualização recursiva do valor dos coeficientes da política anterior (linha 13).

Os coeficientes $\theta_f^{(t),\pi_n}$ dos indicadores obtidos através das funções $\phi_f^{(t)}$ do estágio t definem a política π_n e os coeficientes $\theta_f^{(t),\pi_n,m+1}$ representam o valor estimado para o coeficiente $\theta_f^{(t),\pi_n}$ após a realização do m -ésimo caminho da política π_{n-1} . Para cada um dos M caminhos aleatórios da simulação da política π_n os custos observados $\hat{j}^{(t),n,m}$ de cada estágio t e os valores de $\phi_f^{(t)}(x^{(t)}, u^{(t)})$ são computados e utilizados para atualizar recursivamente $\theta_f^{(t),\pi_n,m+1}$. Na linha 4, $\theta_f^{(t),\pi_n,1}$ são inicializados com o valor dos coeficientes da política π_{n-1} .

As demais etapas são equivalentes para os algoritmos. Cada uma das N diferentes políticas $(\pi_0, \pi_1, \dots, \pi_{N-1})$ é avaliada via simulação por M caminhos aleatórios. Os algoritmos se iniciam aplicando a política π_0 que consiste em decidir utilizando os valores iniciais atribuídos para os coeficientes $\theta_f^{(t,\pi_0)}$, linha 2 do *PIASE* (linha 2 do *PIAMQR*).

Na linha 6 (linha 7), a decisão ótima para o estágio t é obtida após resolver o problema (6.21). Na linha 7(8) gera-se a incerteza da iteração m para o estágio $t+1$ e, na linha 8(9) aplica-se a função transição para obter o estado futuro. Esse processo se repete para todos os valores de $t \in \{0\} \cup T$.

É atribuído o valor nulo para a observação de $\hat{j}^{(T_{max}+1),n,m}$, linha 9(10). E os demais custos observados são calculados recursivamente a partir do estágio T_{max} e voltando até o estágio 0, linha 11(12). Os coeficientes das políticas são atualizados conforme relatado e, após as N iterações, retorna-se a política π_N encontrada, linha 14(18).

```

1 início
2   | inicialize  $\theta_f^{(t), \pi_0} : \forall t \in T, \forall f \in \mathcal{F}$  ;
   fim
3 para todo  $n \in \{1..N\}$  faça
4   | para todo  $m \in \{1..M\}$  faça
5     | para todo  $t \in \{0..T_{max}\}$  faça
6       |  $u^{(t), m*} \leftarrow \arg \min_{u \in \Xi_L^{(t)}} \{g_t^m(x^{(t), m}, u) + \theta_0^{(t), \pi_{n-1}} +$ 
7         |  $\sum_{f \in \mathcal{F}} \theta_f^{(t), \pi_{n-1}} \phi_f^{(t)}(x^{(t), m}, u)\}$ ;
8         | Gere a incerteza  $\omega^{(t+1)}$  ;
9         |  $x^{(t+1), m} \leftarrow f_t(x^{(t), m}, u^{(t), m*}, \omega^{(t+1)})$ ;
        fim
10      |  $\hat{j}^{(T_{max}+1), \pi_n, m} \leftarrow 0$ ;
11      | para todo  $t \in \{T_{max}, T_{max} - 1, ..., 0\}$  faça
12        |  $\hat{j}^{(t), \pi_n, m} \leftarrow g_t^m(x^{(t), m}, u^{(t), m*}) + \hat{j}^{(t+1), \pi_n, m}$ ;
        fim
      fim
    fim
  fim
12  defina os coeficientes  $\bar{\theta}^{(t), \pi_n}$  a partir da regressão linear múltipla entre
    os valores de  $\hat{j}^{(t+1), n}$  e  $\phi^{t, n}$ :  $\forall t \in T$  ;
13   $\theta^{(t), \pi_n} = (1 - \alpha_n)\theta^{(t), \pi_{n-1}} + \alpha_n \bar{\theta}^{(t), \pi_n}$ :  $\forall t \in T$ ;
    fim
14 retorna  $\theta_f^{(t), \pi_N} : \forall t \in T, \forall f \in \mathcal{F}$  ;

```

Algoritmo 5: *Policy Iteration Algorithm* com atualização dos coeficientes pelo método de suavização exponencial (*PIASE*)

```

1 início
2   | inicialize  $\theta_f^{(t),\pi_0} : \forall t \in T, \forall f \in \mathcal{F}$  ;
   fim
3 para todo  $n \in \{1..N\}$  faça
4   |  $\theta_f^{(t),\pi_n,1} = \theta_f^{(t),\pi_{n-1}} : \forall t \in T, \forall f \in \mathcal{F}$ ;
5   para todo  $m \in \{1..M\}$  faça
6     | para todo  $t \in \{0..T_{max}\}$  faça
7       |  $u^{(t),m*} \leftarrow \arg \min_{u \in \Xi_L^{(t)}} (g_t^M(x^{(t),m}, u) + \theta_0^{(t),\pi_{n-1}} +$ 
8         |  $\sum_{f \in \mathcal{F}} \theta_f^{(t),\pi_{n-1}} \phi_f^{(t)}(x^{(t),m}, u));$ 
9         | Gere a incerteza  $\omega^{(t+1)}$  ;
10        |  $x^{(t+1),m} \leftarrow f_t(x^{(t),m}, u^{(t),m*}, \omega^{(t+1)});$ 
11        fim
12        |  $\hat{j}^{(T_{max}+1),\pi_n,m} \leftarrow 0;$ 
13        para todo  $t \in \{T_{max}, T_{max} - 1, \dots, 0\}$  faça
14          |  $\hat{j}^{(t),\pi_n,m} \leftarrow g_t^M(x^{(t),m}, u^{(t),m*}) + \hat{j}^{(t+1),\pi_n,m};$ 
15          fim
16          se  $n > 1$  então atualize recursivamente o valor de  $\theta_f^{(t),\pi_n,m+1}$  a
17          partir de  $\theta_f^{(t),\pi_n,m}$ ,  $x^{(t),m}$ ,  $u^{(t),m*}$  e  $\hat{j}^{(t),\pi_n,m} : \forall t \in T, \forall f \in \mathcal{F}$ ;
18        fim
19      fim
20    fim
21  fim
22 retorna  $\theta_f^{(t),\pi_N} : \forall t \in T, \forall f \in \mathcal{F}$  ;

```

Algoritmo 6: *Policy Iteration Algorithm* com atualização dos coeficientes pelo método dos mínimos quadrados recursivo *PIAMQR*

7 Experimentos computacionais para o PDEDSARC

Este capítulo apresenta os experimentos computacionais realizados para testar as abordagens desenvolvidas para a solução do PDEDSARC proposto no Capítulo 6. No entanto, a abordagem míope descrita na seção 6.2.2 gerou resultados muito inferiores nos testes preliminares e por isso seu desempenho não será apresentado aqui. Serão detalhadas as instâncias utilizadas, o planejamento do experimento computacional, os parâmetros utilizados nas abordagens *PIASE* e *PIAMQR* e os resultados obtidos.

7.1 Características das instâncias

Assim como no Capítulo 4, as instâncias aqui utilizadas foram construídas aleatoriamente, mas possuem dimensões baseadas em uma situação real. Neste caso, a situação real é a verificada diariamente em um hospital de médio porte dedicado à saúde da mulher e à maternidade em Belo Horizonte.

Foram geradas 48 instâncias provenientes da combinação de 3 cenários de chegadas com 16 configurações de 8 diferentes tipos. Os tipos de configuração se diferenciam pelo o número esperado de cirurgias, pela disponibilidade de recursos e pela proporção entre eletivas/não-eletivas. Os cenários de chegadas de cirurgias não eletivas representam diferentes parametrizações para o processo estocástico utilizado para cada tipo de cirurgia.

Todas instâncias possuem 4 clínicas e a distribuição dos preceptores foi feita aleatoriamente para cada cenário. O cenário com 9 preceptores possui 3 clínicas com 2 preceptores e uma clínica com 3, e o cenário com 12 preceptores possui 3 em cada clínica. A disponibilidade dos dois recursos renováveis foi definida aleatoriamente para cada um dos 8 tipos de instâncias. Para cada instância, uma das salas foi definida para possuir um recurso renovável fixo do tipo 1.

O número de tipos de cirurgias é igual a 10 em todas as instâncias onde 2 clínicas possuem 2 tipos e outras 2 clínicas possuem 3 tipos cada. Em cada período, a chegada de cirurgias é definida a partir de 10 processos de Bernoulli independentes, com probabilidade definida pela combinação do número esperado de cirurgias, da proporção entre E/NE e do cenário de chegadas.

A duração das cirurgias segue distribuição uniforme com mínimo e máximo definidos aleatoriamente para cada tipo de cirurgia. Os conjuntos R_s foram definidos aleatoriamente para cada tipo de cirurgia, assim como o conjunto $S_D^{(0)}$ (conjunto de cirurgias eletivas) para cada instância.

Os custos utilizados foram os mesmos em todas as instâncias. O custo de cancelamento de um cirurgia eletiva foi usado como base para definir os outros custos. Este custo é o mesmo para todas as eletivas. Uma penalização de 60% foi aplicada ao custo de cancelamento quando a cirurgia era urgente. Os custos de não realização foram definidos como 60% destes. Os custos de início fora da janela de tempo (por período de atraso) foram definidos de maneira aleatória para cada tipo de cirurgia, eletiva ou não, dentro do intervalo de 10 a 15% do custo de cancelamento da cirurgia eletiva.

- **Número esperado de cirurgias** - Esta dimensão define, em média, quantas cirurgias eletivas e não eletivas se tornarão disponíveis durante um dia. Ele consiste na soma das cirurgias eletivas (disponíveis em x_0) com a quantidade esperada de cirurgias que chegarão ao longo do dia. Trabalhamos com dois grupos de instâncias, um com valor esperado ($E[|S|]$) de 18 cirurgias (**s18**) e outro com valor esperado de 24 cirurgias (**s24**). Essas quantidades são condizentes com o número de cirurgias realizadas na maternidade durante um dia que se distribui entre 12 e 30.
- **Disponibilidade de recursos** - A disponibilidade de recursos define a quantidade de salas e o número de preceptores. Foram testadas duas configurações de disponibilidade de recursos: **r6p9** com 6 salas, 9 preceptores e **r9p12** com 9 salas, 12 preceptores. O hospital em questão possui 6 salas e a quantidade de preceptores que atendem durante um dia é variável de-

vido à natureza do vínculo destes com o hospital, mas os valores 9 e 12 são aderentes a um dia normal de agendamentos.

- **Proporção entre eletivas/não-eletivas** - Esta proporção se refere à relação entre a quantidade de cirurgias previamente agendadas (eletivas) e a quantidade de cirurgias que chegam durante o dia (não eletivas). Foram testadas as proporções de 1/2 e de 2/1 gerando os cenários **e6** (6/12) e **e12** (12/6) para as instâncias do grupo **s18** e **e8** (8/16); e **e16** (16/8) para as instâncias do grupo **s24**. A variação destas proporções foi observada na prática.
- **Processo de chegada** - O processo de chegada das cirurgias eletivas define o congestionamento do sistema. Foram definidos três diferentes cenários para o processo de chegada das cirurgias não-eletivas. O cenário **cb** concentra 50% das chegadas nos 6 primeiros períodos, o cenário **cc** concentra 50% das chegadas entre os períodos 10 e 15 e o cenário **ch** distribui uniformemente as chegadas ao longo dos períodos. A Figura 6 apresenta a distribuição percentual esperada das chegadas ao longo do dia nos 3 cenários.

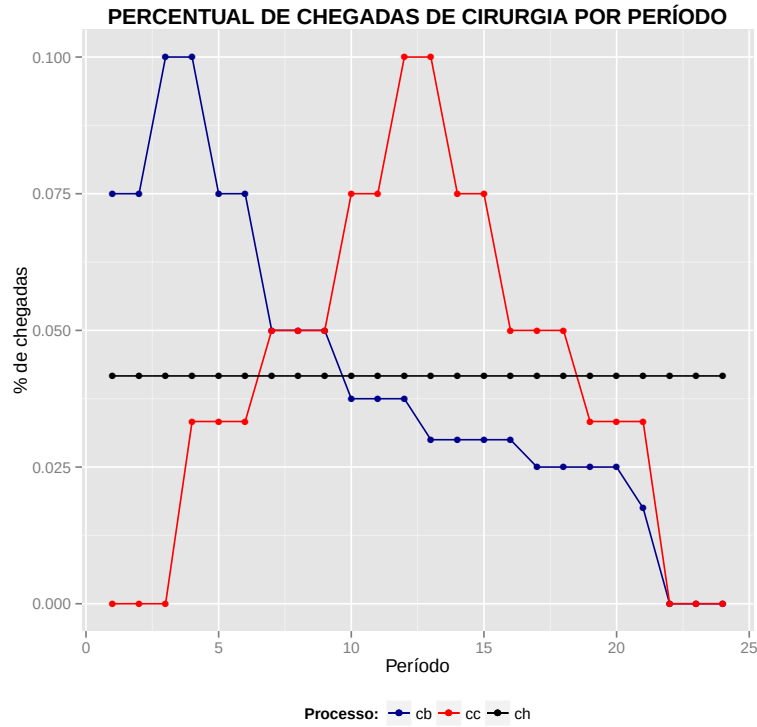


Figura 6 – Distribuição percentual esperada das chegadas ao longo do dia

7.2 Parametrização dos experimentos computacionais

Uma simulação com 30 caminhos aleatórios foi realizada para comparar o desempenho dos métodos desenvolvidos em cada instância criada. Cada caminho aleatório corresponde a um “dia diferente” para cada instância. Os experimentos foram conduzidos fixando a semente do gerador de números aleatórios para permitir a reprodução dos experimentos quantas vezes forem necessárias. O gerador de números aleatórios utilizado internamente pelo PIA é independente do gerador utilizado para a simulação do desempenho das abordagens e também teve sua semente fixada.

Além disto, garantiu-se que os 30 caminhos aleatórios gerados fossem os mesmos para cada um dos métodos permitindo, assim, a realização de um teste pareado para compará-los. O teste utilizado foi o *t-Student* unilateral para amostras pare-

adas a um nível de significância de 0.05.

A estatística de desempenho utilizada como critério de comparação foi o custo total no período 0 para cada método em cada caminho aleatório. A hipótese nula do teste, isto é, a hipótese que se desejava testar, foi a de igualdade das médias dos desempenhos das abordagens contra a hipótese de que os custos esperados das abordagens PIA fossem menores do que as do *Lookahead*.

$$H_0 : \mu_{ct}^{PIA} = \mu_{ct}^{LKD} \quad (7.1)$$

$$H_a : \mu_{ct}^{PIA} < \mu_{ct}^{LKD} \quad (7.2)$$

Um nível de significância de 0.05 garante uma probabilidade de 5% de cometer o erro do tipo I, isto é, rejeitar a hipótese H_0 quando a mesma for verdadeira. Um teste t-pareado com 30 replicações a um nível de significância de 0.05 é capaz de garantir poderes de 0.283, 0.848 e 0.996 para detectar diferenças absolutas entre as médias de 0.2, 0.5 e 0.8 desvios padrões, respectivamente, em um teste unilateral. Um valor elevado no poder do teste implica em uma baixa probabilidade de cometer o erro do tipo II - não rejeitar H_0 quando H_a for válida.

Todos os experimentos foram realizados em um computador do tipo Intel Xeon X5690 ©com 24 processadores de 3.47GHz e 132 GB de RAM com sistema operacional Linux.

Os algoritmos foram codificados em C++ utilizando o CPLEX 12.4 para resolver os subproblemas de programação inteira interfaceado pela biblioteca **Concert**. Os subproblemas foram executados com a diretriz `IloCplex::ParallelMode = 1` para garantir uma busca paralela em modo determinístico, o que permite a reprodutibilidade dos experimentos.

Os métodos estatísticos para estimação dos coeficientes da regressão foram implementados com o uso da biblioteca **Armadillo**. O tempo de execução foi contabilizado com a ajuda da biblioteca `ctime` e a contagem de tempo considera o tempo gasto por todos os processadores, isto é, a soma do tempo gasto em cada processador para resolver o problema.

7.3 Parametrização do *PIA*

Para o conjunto de instâncias com processo de chegada homogêneo (**ch**) foram utilizados os indicadores de custo futuro de início fora da janela de tempo e média das taxas de desocupação. Para os conjuntos de instâncias **cc** e **cb** foi, adicionalmente, utilizado o indicador de soma do uso de salas por cirurgias nos próximos 6 períodos. A escolha foi feita a partir do desempenho observado em testes preliminares.

Conforme descrito na seção 6.2.4.3, uma política é definida pelos coeficientes da regressão dos indicadores utilizados pelo método. O método aplicado possui dois (ou três) indicadores para cada um dos 24 períodos, isto é, possui, ao todo, 48 (ou 72) coeficientes de regressão. A cada um dos coeficientes é atribuído um valor inicial que pode interferir na velocidade de convergência dos métodos.

Os indicadores definidos demonstraram-se, no geral, correlacionados com o valor real da função $J^{(t)}()$. Foram, na maioria dos casos, significativos no modelo linear. No entanto, o modelo multivariado para a aproximação do valor da função revelou-se pouco explicativo, fato que explica a elevada variabilidade das respostas obtidas pela aplicação dos algoritmos. Uma possível relação não linear entre os indicadores e o valor da função e a existência de aspectos do problema não mapeados pelos mesmos explicam a fragilidade do modelo de regressão.

A correlação, no entanto, depende de cada cada período e varia com as instâncias. Testes preliminares mostraram que a correlação tende a ser mais fracas quando o processo de chegada é não homogêneo. É importante observar que embora o modelo não seja muito explicativo, ainda assim os algoritmos foram capazes de reduzir os custos.

Na ausência de informação a respeito do comportamento dos indicadores, à média das taxas de desocupação ($\phi_{td}^{(t)}()$), e à soma do uso de salas por cirurgias nos próximos K períodos ($\phi_{su}^{(t)}()$) foram atribuídos valores iniciais iguais a zero para seus coeficientes. No caso do indicador custo futuro de início fora da janela de tempo ($\phi_{if}^{(t)}()$), foram atribuídos os valores iniciais iguais a 1 para os coeficientes com o objetivo de replicar o procedimento *Lookahead* na política inicial π_0 . Observe que o custo futuro de início fora da janela de tempo é incorporado ao objetivo dos subproblemas resolvidos no método *Lookahead*.

Como relatado no Capítulo 6, foram testados dois métodos para atualização dos coeficientes. Para o algoritmo *PIASE*, a regra de atualização do *stepsize* utilizada foi $\alpha_n = \frac{1}{n}$. No caso do algoritmo *PIAMQR*, o valor da constante λ utilizado foi 1.01. Os valores foram definidos qualitativamente após testes com outros valores.

Outros dois parâmetros devem ser definidos para executar o método *PIA*, o número de políticas testadas (N) e o número de replicações por política (M). O número de políticas testadas tem como objetivo alcançar a convergência do método. Já o número de replicações por política influencia na qualidade da estimativa dos coeficientes dos indicadores.

A convergência do valor esperado do custo total em t_0 depende da convergência de todos os coeficientes. A título de exemplo, a Figura 7 apresenta o valor dos coeficientes dos indicadores de média das taxas de desocupação para os primeiro, décimo segundo e penúltimo períodos da aplicação do *PIAMQR* na instância **s18e6r6p9ch**.

Na figura, $\theta_{td}^{(t), \pi_{n-1}}$ representa o coeficiente do indicador taxa média de desocupação que é utilizado durante a iteração n do método *PIAMQR*. Já $\theta_{td}^{(t), \pi_n, m}$ representa o coeficiente atualizado após a avaliação do m -ésimo caminho aleatório da n -ésima iteração do mesmo método.

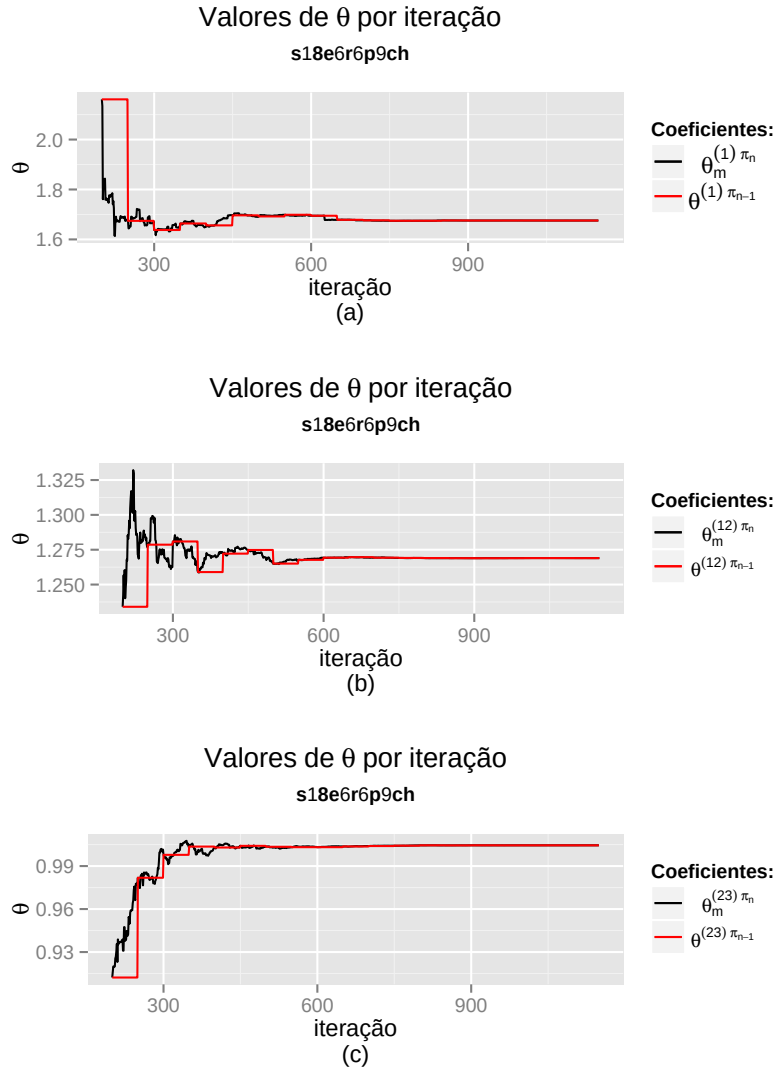


Figura 7 – Valores de $\theta_{td}^{(t), \pi_{n-1}}$ e $\theta_{td}^{(t), \pi_n, m}$ por iteração dos períodos $t = 1$ (a) $t = 12$ (b) e $t = 23$ (c) para a aplicação do *PIAMQR* na instância **s18e6r6p9ch**

O número de problemas de programação inteira a serem resolvidos aumenta linearmente com a medida $N \times M$. Portanto, a definição dos parâmetros N e M é também uma decisão de custo-benefício e este ponto de vista foi utilizado para a escolha dos valores destes parâmetros. A aplicação de testes de convergência para

os algoritmos é uma questão complexa e foge ao escopo deste trabalho.

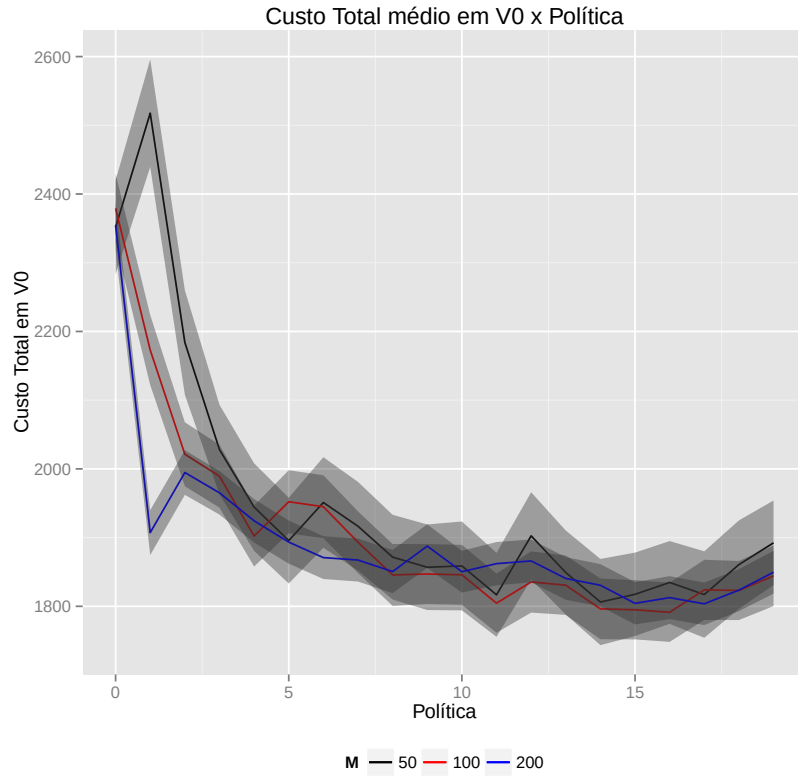


Figura 8 – Comparação do custo esperado em $t_0 \times \pi_n$ para diferentes valores M do PIAMQR

Para auxiliar a definição de N e M , foram realizados testes em uma instância com 9 cirurgias eletivas, em média 12 cirurgias não eletivas, 6 salas, 9 preceptores, processo de chegada homogêneo e utilizando o PIAMQR. A Figura 8 apresenta o gráfico do custo esperado em $t_0 \times \pi_n$ para variações de M (igual a 50, 100 e 200). Percebe-se que não há diferença visual significativa entre as séries e por este motivo optou-se por utilizar o valor de $M = 50$. No entanto, por ser importante obter uma boa estimativa inicial para os coeficientes, utilizamos $M = 200$ para avaliar π_0 .

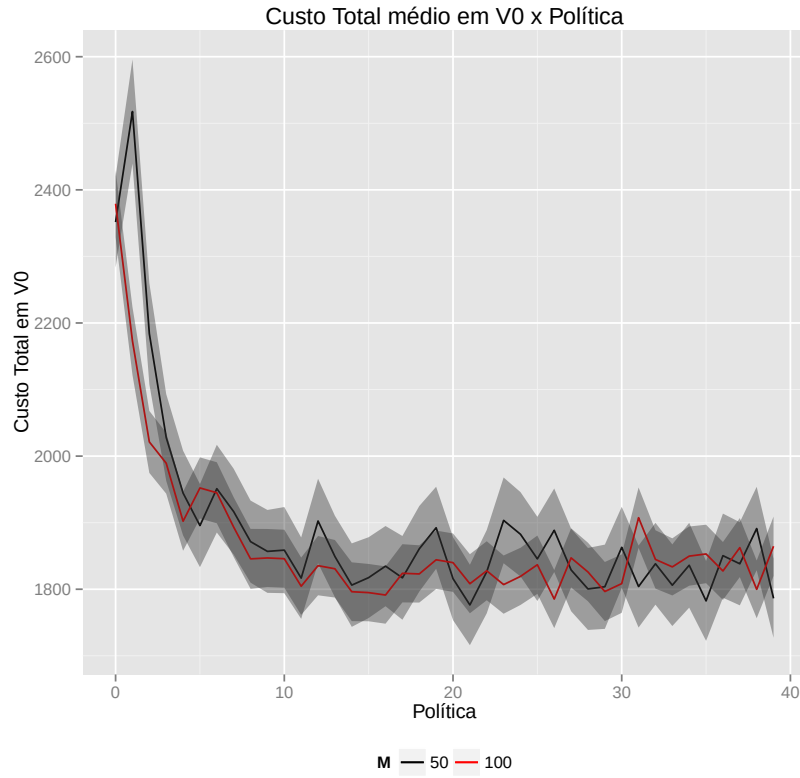


Figura 9 – Custo esperado em $t_0 \times \pi_n$ para 40 iterações do PIAMQR

A Figura 9 apresenta o gráfico do custo esperado em $t_0 \times \pi_n$ para as 40 primeiras políticas com valor de M igual a 50 e 100. É possível observar que a série sofre uma queda nos primeiros valores de n , e a partir de $n = 5$, embora não se garanta a convergência, passa a possuir uma variação menor. Empiricamente, optou-se por utilizar $N = 20$.

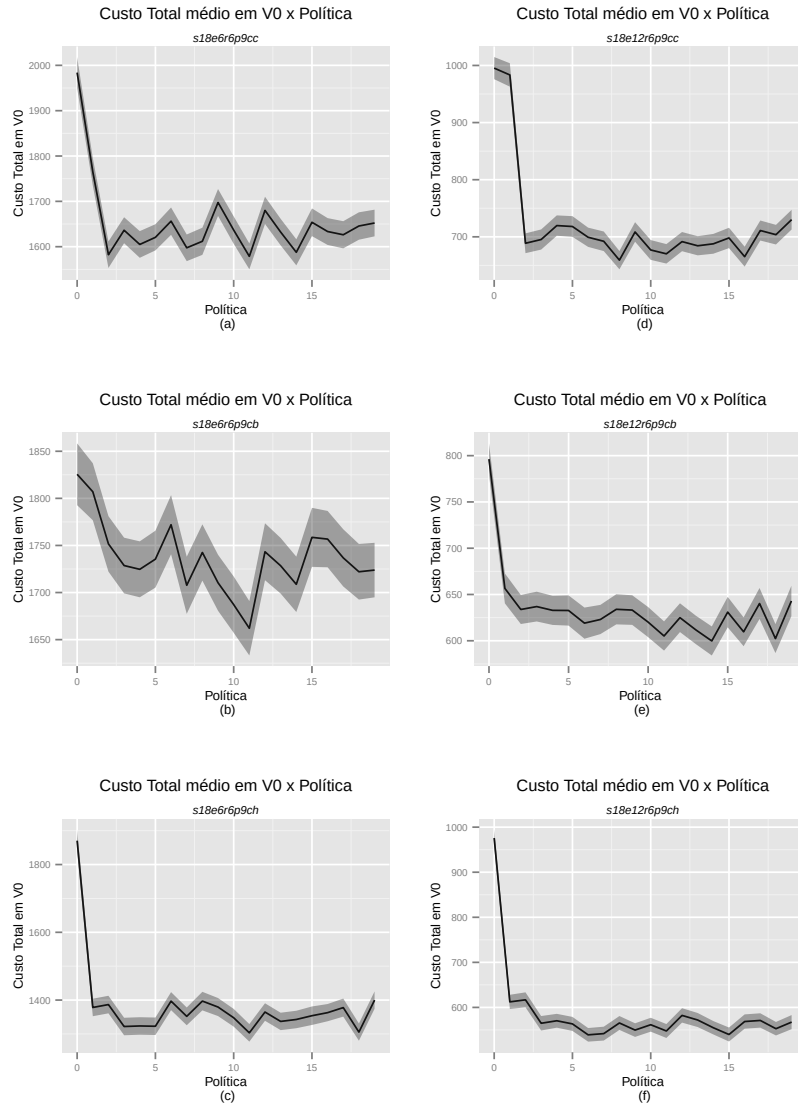


Figura 10 – Custo esperado em $t_0 \times \pi_n$ para instâncias do grupo **s18**

Uma vez que o comportamento observado nas séries da Figura 9 foi similar ao de todas as instâncias testadas, não serão apresentados todos os 48 gráficos. Como exemplo, a Figura 10 ((a)-(f)) apresenta o gráfico do custo esperado em $t_0 \times \pi_n$ das 20 primeiras políticas com valor de $M = 50$ para algumas instâncias do grupo **s18** e a Figura 11 ((a)-(f)) refere-se a algumas instâncias do grupo **s24**. Visualmente

observa-se que, na maioria dos casos, após as primeiras 5 políticas os valores das séries tendem a se estabilizar em um patamar inferior ao inicial.

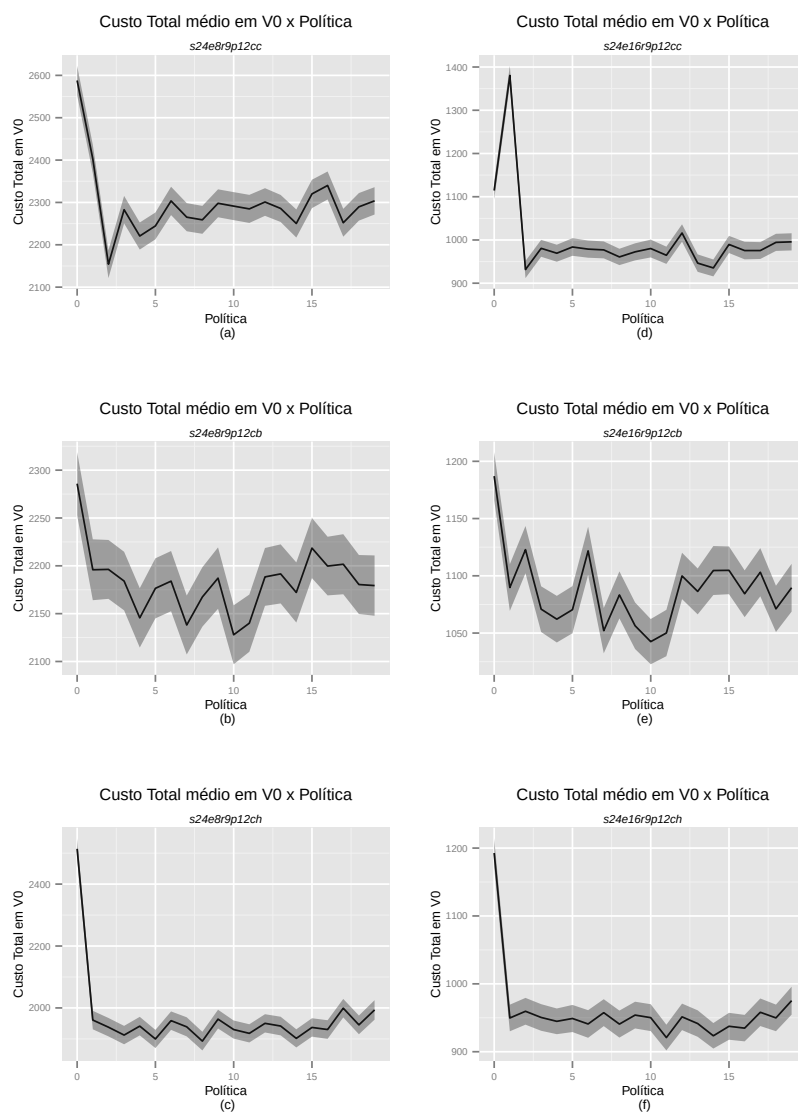


Figura 11 – Custo esperado em $t_0 \times \pi_n$ para instâncias do grupo **s24**

7.4 Resultados obtidos

7.4.1 Comparação dos métodos de atualização dos coeficientes da regressão

Para comparar a diferença entre os métodos MQR e SE, foram utilizadas as instâncias de 18 cirurgias (**s18**) como referência inicial. Devido à discrepância nos resultados encontrados, especialmente no tempo médio por replicação, optou-se por adotar como referência o método de melhor desempenho, método dos mínimos quadrados recursivos (MQR).

O gráfico da Figura 12 é um *scatterplot* dos custos médios totais encontrados para o método SE (eixo y) contra os valores encontrados para o método MQR (eixo x). As cores dos pontos diferenciam as instâncias pela relação eletiva/não eletiva, as formas diferenciam pelo processo de chegada e os rótulos de valores representam o percentual entre a diferença dos valores de MQR e SE em relação aos valores de SE. Em apenas uma instância, o MQR foi superado pelo SE. No entanto, não há diferença estatística entre a média dos valores comparados para um nível de confiança de 5%.

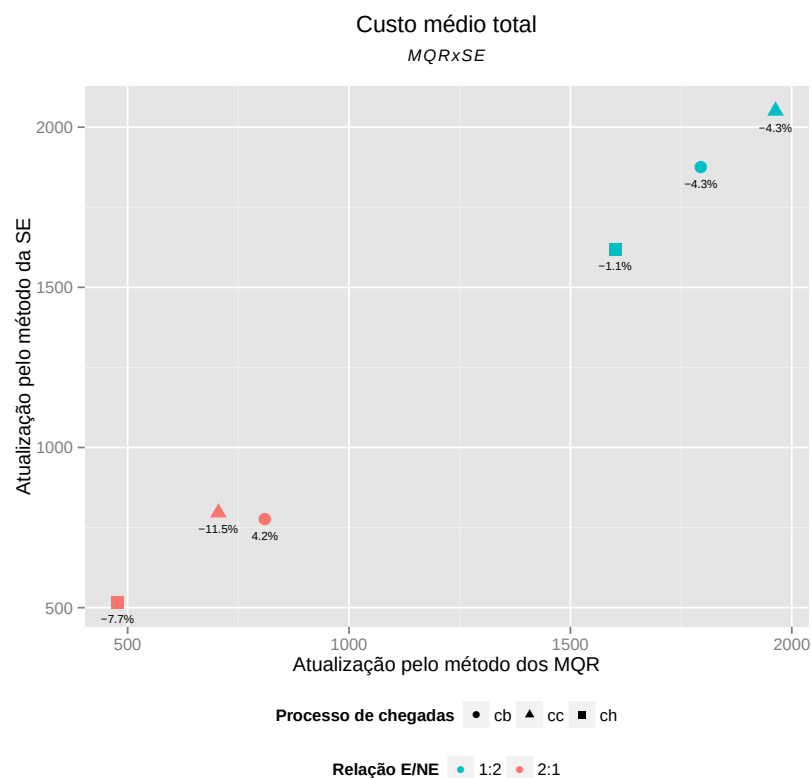


Figura 12 – Comparação dos valores de custo médio total obtidos pelos métodos MQR e SE

O gráfico da Figura 13 é um *scatterplot* dos tempos médios de execução das replicações da simulação para o método SE (eixo y) contra os valores encontrados para o método MQR (eixo x). As cores dos pontos diferenciam as instâncias pela relação eletiva/não eletiva, as formas diferenciam pelo processo de chegada e os rótulos de valores representam o percentual entre a diferença dos valores de MQR e SE em relação aos valores de SE. Em todos os casos o tempo médio de execução do método SE superou o do MQR chegando a ser 45.5% superior.

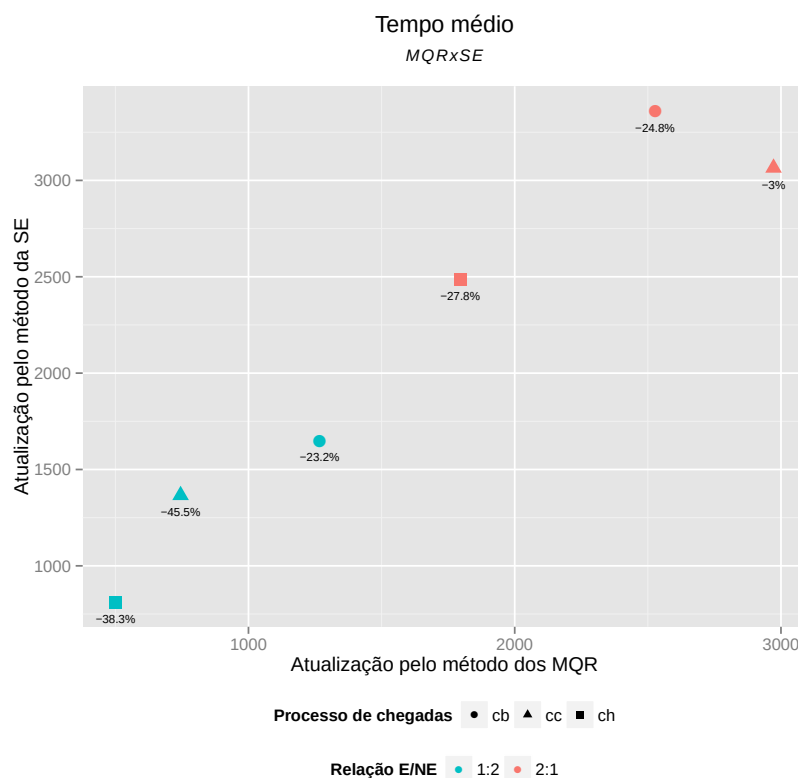


Figura 13 – Comparação dos tempos de execução dos métodos MQR e SE

7.4.2 Comparação dos algoritmos desenvolvidos para a solução do problema

As Tabelas 16 e 17 apresentam os resultados obtidos para as instâncias dos grupos **s18** e **s24**, respectivamente. A primeira coluna (*I*) identifica as instâncias. Como mencionado anteriormente, cf., seção 7.2, para cada instância foram gerados 30 caminhos aleatórios simulando diferentes dias na prática. A segunda coluna apresenta os custos médios totais obtidos através da aplicação do método *Lookahead*, a coluna $t_r(s)$ apresenta os tempos médios, em segundos, necessários para resolver o problema para um caminho aleatório, isto é, tempo necessário para resolver 24 subproblemas de programação inteira. Em seguida são apresentados os custos obtidos pelo método *PIAMQR*. A coluna P-valor refere-se à probabilidade da hipótese

nula do teste *t-student* (médias são iguais) ser verdadeira para os dados coletados. A coluna $t_e(s)$ apresenta o tempo médio em segundos que foi necessário para realizar a estimativa dos coeficientes que definem a política implementada. Por fim, a coluna $t_r(s)$ apresenta os tempos médios, em segundos, necessários para resolver o problema para um caminho aleatório, isto é, tempo necessário para resolver 24 subproblemas de programação inteira.

Embora o tempo necessário para a avaliação de um caminho aleatório tenha sido relativamente baixo, durante cada simulação são avaliados 1150 caminhos aleatórios (cf. 7.3), totalizando 28750 problemas de Programação Inteira. Caso o tempo necessário para se resolver um caminho cresça muito, pode-se inviabilizar a aplicação do método *PIA*.

É importante ressaltar que não foi imposto limite de tempo para a execução do CPLEX. A imposição poderia ocasionar o término do tempo sem ao menos uma solução viável o que travaria o algoritmo. A diretriz do CPLEX `IloCplex::EpGap` que define como critério de parada um *gap* pré definido foi testada preliminarmente e não gerou ganhos em termos de tempo, e por isso não foi utilizada. No entanto, em instâncias mais difíceis esta pode ser uma solução.

A Tabela 16 apresenta os resultados obtidos para as instâncias do grupo **s18**. Em 17 das 24 instâncias, o procedimento *PIAMQR* se saiu estatisticamente melhor do que o procedimento *Lookahead* a um nível de significância de 0.05. Em 5 instâncias, o *PIAMQR* se saiu melhor que o *Lookahead*, mas a diferença encontrada não foi suficiente para indicar uma diferença estatística a um nível de significância de 0.05. Nas instâncias **s18e12r6p9cb-2** e **s18e6r6p9cb-2** o método *Lookahead* foi melhor, sendo que, para a última, a diferença encontrada foi suficiente para indicar uma diferença estatística. Isto é, para esta instância $\mu_{ct}^{PIAMQR} > \mu_{ct}^{LKD}$ a um nível de significância de 0.05.

Os tempos computacionais necessários para avaliar um caminho aleatório foram comparáveis nos dois métodos. O tempo computacional necessário para realizar a estimativa dos coeficientes dos indicadores foram adequados ao horizonte de tempo de tomada de decisão. Em uma aplicação prática, a estimativa dos coeficientes poderia ser feita na véspera, quando seriam conhecidas as cirurgias eletivas

I	Lookahead		PIAMQR			
	Custo Médio	$t_r(s)$	Custo Médio	pvalor	$t_e(s)$	$t_r(s)$
s18e6r6p9cb-1	2127	2	1795	0.0056	1268	1
s18e6r6p9cb-2	975	0	1213	0.9949	982	1
s18e6r6p9cc-1	2225	1	1963	0.0383	745	1
s18e6r6p9cc-2	1874	1	1301	0.0003	1583	1
s18e6r6p9ch-1	2200	1	1601	< 0.0001	501	0
s18e6r6p9ch-2	1818	1	1235	< 0.0001	911	1
s18e12r6p9cb-1	1413	1	1195	0.0067	953	1
s18e12r6p9cb-2	719	1	721	0.5117	1958	2
s18e12r6p9cc-1	1862	0	1478	0.0004	888	1
s18e12r6p9cc-2	1014	1	975	0.3080	2101	2
s18e12r6p9ch-1	1738	0	1415	0.0009	764	1
s18e12r6p9ch-2	1075	1	876	0.0378	2169	2
s18e6r9p12cb-1	1008	1	810	0.0078	2527	2
s18e6r9p12cb-2	609	0	595	0.4281	3022	3
s18e6r9p12cc-1	875	2	705	0.0038	2972	3
s18e6r9p12cc-2	757	2	553	0.0302	8700	8
s18e6r9p12ch-1	784	1	477	0.0002	1795	2
s18e6r9p12ch-2	764	1	385	0.0002	4202	4
s18e12r9p12cb-1	689	1	657	0.2640	1832	2
s18e12r9p12cb-2	479	1	396	0.1195	4146	4
s18e12r9p12cc-1	649	0	467	0.0263	1704	1
s18e12r9p12cc-2	483	1	283	0.0063	4293	4
s18e12r9p12ch-1	477	0	459	0.3381	1683	1
s18e12r9p12ch-2	399	1	137	0.0021	3490	3

Tabela 16 – resultados para as instâncias do grupo **s18**

candidatas a serem realizadas no dia seguinte. O tempo máximo foi 8700s que corresponde a cerca de 2h25.

A Tabela 17 apresenta os resultados obtidos para as instâncias do grupo **s24**. Em 18 das 24 instâncias o procedimento *PIAMQR* se saiu estatisticamente melhor do que o procedimento *Lookahead* a um nível de significância de 0.05. Se for considerado um nível de significância de 0.07, o desempenho é melhor em 20 das 24 instâncias deste grupo (as instâncias **s24e8r6p9cb-2** e **s24e8r6p9cc-1** obtiveram P-valores de 0.067 e 0.055, respectivamente). Embora nas instâncias **s24e16r6p9cb-1**, **s24e16r6p9cb-2**, **s24e8r9p12cb-1** e **s24e16r9p12cb-2** o procedimento *Lookahead* tenha gerado melhores resultados, a diferença não foi suficiente para indicar uma diferença estatística entre os métodos a um nível de significância de 0.05.

I	Lookahead		Custo Médio	PIAMQR		
	Custo Médio	$t_r(s)$		pvalor	$t_e(s)$	$t_r(s)$
s24e8r6p9cb-1	3740	1	3140	0.001	2783	2
s24e8r6p9cb-2	3294	5	3118	0.0670	12521	11
s24e8r6p9cc-1	4043	6	3797	0.0550	2942	3
s24e8r6p9cc-2	4012	1	3099	< 0.0001	3667	3
s24e8r6p9ch-1	3849	3	2945	< 0.0001	2127	2
s24e8r6p9ch-2	3449	2	2673	0.0003	2465	2
s24e16r6p9cb-1	2300	1	2480	0.9212	811	1
s24e16r6p9cb-2	1751	1	1759	0.5215	2129	2
s24e16r6p9cc-1	2823	1	2413	0.0012	795	1
s24e16r6p9cc-2	2125	2	1622	0.0029	1836	2
s24e16r6p9ch-1	2493	1	1987	< 0.0001	606	1
s24e16r6p9ch-2	1829	1	1253	0.0002	2344	2
s24e8r9p12cb-1	2505	13	2523	0.5707	34105	30
s24e8r9p12cb-2	1841	3	1578	0.0058	14507	13
s24e8r9p12cc-1	2889	9	2272	0.0002	27346	24
s24e8r9p12cc-2	2163	8	1601	< 0.0001	11246	10
s24e8r9p12ch-1	2640	5	2286	0.0093	31940	28
s24e8r9p12ch-2	1971	1	1550	0.0001	25135	22
s24e16r9p12cb-1	1253	2	1115	0.0442	1759	2
s24e16r9p12cb-2	607	2	612	0.5223	3578	3
s24e16r9p12cc-1	1771	1	1121	< 0.0001	1269	1
s24e16r9p12cc-2	985	1	458	0.0002	2724	2
s24e16r9p12ch-1	1687	1	1095	< 0.0001	1294	1
s24e16r9p12ch-2	858	1	697	0.0299	4670	4

Tabela 17 – resultados para as instâncias do grupo **s24**

Os tempos computacionais necessários para avaliar um caminho aleatório foram compatíveis nos dois métodos. O tempo computacional necessário para realizar a estimativa dos coeficientes dos indicadores foram mais elevados, tendo como valor máximo (34105s) que corresponde à cerca de 9h30.

As Figuras 14 e 15 apresentam graficamente o resultado dos experimentos para as instâncias dos grupos **s18** e **s24**, respectivamente. O gráfico das figuras são *scatterplots* dos custos médios totais encontrados para a abordagem *Lookahead* (eixo y) contra os valores encontrados para o método *PIAMQR* (eixo x). As cores dos pontos diferenciam as instâncias pela relação eletiva/não eletiva, as formas diferenciam pelo processo de chegada, o tamanho varia de acordo com a disponibilidade de recursos e os rótulos de valores representam o percentual entre a diferença dos valores do *PIAMQR* e *Lookahead* em relação aos valores de *Lookahead*.

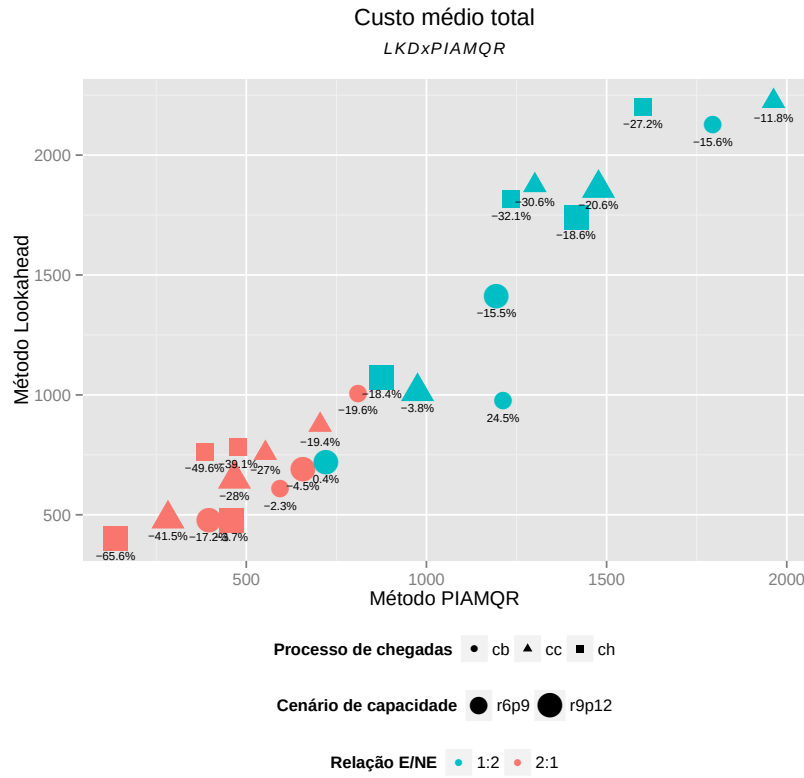


Figura 14 – *Scatterplot* de $V_0^{LKD} \times V_0^{PIA}$ para instâncias do grupo **s18**

A Figura 14 mostra que o melhor desempenho relativo do procedimento *PIAMQR* para o grupo *s18* foi -65.6% obtido para instância **s18e12r9p12ch**. O pior desempenho foi obtido para a instância **s18e6r6p9cb-2** ($+24.5\%$).

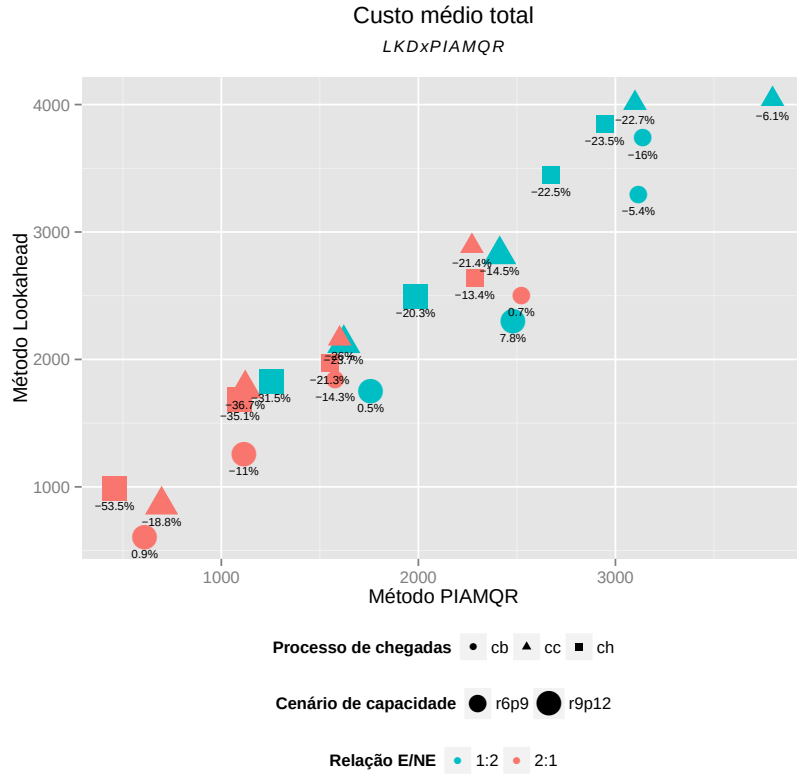


Figura 15 – *Scatterplot* de $V_0^{LKD} \times V_0^{PIA}$ para instâncias do grupo **s24**

A Figura 15 mostra que o melhor desempenho relativo do procedimento *PIAMQR* para o grupo *s24* foi -53.5% obtido para instância **s24e16r9p12ch**. O pior desempenho foi obtido para a instância **s24e8r9p12cb** ($+7.8\%$). Os dois gráficos também mostram que os custos são menores para as instâncias com relação **E/NE 2:1** e, como esperado, para as instâncias com maior disponibilidade de recursos (**r6p9**).

A Tabela 18 apresenta o resumo dos resultados encontrados para as instâncias. As estatísticas foram desagregadas pelo número esperado de cirurgias e por uma segunda dimensão relativa ao agrupamento de instâncias. A coluna *Dif(%)* é a diferença percentual obtida entre os métodos, $t_e(s)$ o tempo médio necessário para a estimativa dos coeficientes e $P - Valor < 0.05$ indica o número de instâncias em que a diferença foi estatisticamente significativa. Observa-se que, tomando como

critério a diferença percentual entre o custo médio encontrado pelo procedimento *Lookahead* e o procedimento *PIAMQR*, o ganho obtido por se aplicar o método *PIAMQR* foi ligeiramente maior no conjunto de instâncias com 18 cirurgias do que no conjunto de instâncias com 24 cirurgias. É necessário destacar, no entanto, que a garantia estatística de que o custo médio obtido pelo *PIAMQR* seja menor que os do *Lookahead* ($P - \text{valor} < 0.05$) foi observado em 17 das 24 instâncias para 18 cirurgias, enquanto no conjunto com 24 cirurgia isto foi observado em 18 das 24.

Claramente, tanto para $E[|S|] = 18$, quanto para $E[|S|] = 24$, o *PIAMQR* possuiu menor ganho relativo quando o processo de chegada é concentrado no início do expediente (grupo **cb**). Nos demais grupos de instâncias referentes ao processo de chegada, o método obteve maiores ganhos. A maior diferença de desempenho foi obtida no grupo **cc** para $E[|S|] = 24$ e no grupo **ch** para $E[|S|] = 18$. As instâncias com proporção **E/NE** de 2 : 1 apresentaram maior ganho relativo para o *PIAMQR*.

Em termos do tempo necessário para realizar a estimação dos coeficientes, as instâncias com $E[|S|] = 24$ tiveram um custo computacional maior do que os das instâncias com $E[|S|] = 18$, o que já era esperado. Embora as instâncias do grupo **r9p12** possuam uma quantidade maior de variáveis no subproblema inteiro, as instâncias mais restritas (**r6p9**) foram mais difíceis de se resolver. As instâncias com mais eletivas **E/NE 2:1** demoraram consideravelmente mais do que as instâncias com menos eletivas.

Grupo	s18			s24			Geral
	Dif(%)	$t_e(s)$	$P - \text{valor} < 0.05$	Dif(%)	$t_e(s)$	$P - \text{valor} < 0.05$	
cb	-6.24%	2086	3	-2.97%	9916	3	-4.6%
cc	-22.83%	2873	7	-28.35%	6983	7	-25.59%
ch	-31.78%	1939	7	-23.28%	9779	8	-27.53%
r6p9	-20.81%	2434	10	-15.98%	15273	9	-18.39%
r9p12	-19.76%	2165	7	-22.16%	2091	9	-20.96%
E/NE 1:2	-14.12%	1235	9	-14.72%	2931	8	-14.42%
E/NE 2:1	-26.45%	3364	8	-22.78%	11406	10	-24.61%
Geral	-20.28%	2300	17	-17.83%	8108	18	-19.06%

Tabela 18 – Resumo dos resultados

O menor ganho relativo ao método *Lookahead* obtido pelo do método *PIAMQR*

para o processo de chegada concentrado no início do período se deve, em parte, à baixa qualidade da aproximação do valor da função para este cenário. Testes preliminares mostraram que os indicadores possuem correlação mais baixa com o valor da função para estes cenários. Este fato demonstra a importância da escolha dos indicadores para a melhoria do desempenho. No entanto, um outro aspecto que pode ter influenciado é o fato de haver maior flexibilidade para a reação diante da alteração dos parâmetros nestes cenários, uma vez que há mais tempo para corrigir o planejamento. Sendo assim, o *Lookahead* pode ter obtido melhor desempenho.

Embora no geral o método *PIAMQR* tenha obtido bons resultados, ressalva-se a elevada variabilidade obtida para os custos entre as replicações. Esta característica é herdada da própria variabilidade dos cenários de incerteza e também afeta o método *Lookahead*. O uso de indicadores não lineares em relação às variáveis do problema, assim como a construção de modelos não lineares para a aproximação do valor da função, pode ser um caminho para obter melhores resultados.

A redução média de 19.06% nos custos é significativa do ponto de vista prático. Principalmente se for considerado que essa redução foi obtida em comparação com o uso de um método de otimização, o que ainda não é comum na prática. Conforme observado por outros autores, e.g., (CARDOEN et al., 2010) e também durante visitas em hospitais no decorrer do desenvolvimento desta tese, na prática o sequenciamento é feito de forma manual.

7.5 Considerações sobre a aplicação prática do método *PIA*

O tempo necessário para a obtenção da política a ser utilizada em um determinado cenário é elevado, mas é adequado quando comparado com o horizonte de tomada de decisão que é de um dia. Em um contexto prático os coeficientes seriam estimados uma única vez, na véspera, quando as informações acerca das disponibilidades de recursos e as eletivas são definidas.

Durante o dia, as sucessivas execuções, a cada estágio de decisão, dos problemas de programação inteira para se tomar a decisão são rápidas, em torno de 1s. Pelas Tabelas 16 e 17, pode-se notar que o tempo de avaliação de um caminho aleatório ($t_r(s)$) foi baixo. Como já explicado, um caminho aleatório simula todo

o expediente, i.e., simula todas as tomadas de decisão durante o horizonte de planejamento que, ao todo, são 24.

8 Conclusões

O objetivo deste trabalho foi descrever e propor solução para o problema diário de sequenciamento de cirurgias através de abordagens que fossem aderentes à realidade em termos do tratamento de recursos, da dinamicidade e incertezas do processo. Os modelos foram desenvolvidos buscando incorporar aspectos observados na prática do processo de programação de cirurgias e preencher lacunas identificadas na literatura. Os experimentos computacionais foram conduzidos em instâncias geradas a partir de dados reais. O tema abordado nesta tese é importante uma vez que a gestão eficiente de recursos é vital para a melhoria da qualidade dos serviços na área de saúde. Especificamente, o bloco cirúrgico é o setor de maior custo e maior faturamento de um hospital.

No capítulo 3, foi construído um modelo de programação inteira que considera o uso simultâneo de recursos humanos especializados, aspecto não encontrado na literatura. O modelo também incorporou características de relevância prática, tais como a existência de múltiplos blocos, salas não homogêneas, janela de tempo para os recursos humanos e preferências dos recursos humanos especializados.

Foi demonstrada, no Capítulo 4, a aplicabilidade do modelo determinístico proposto através do exemplo de uma situação real vivida em um hospital de grande porte de Belo Horizonte. O resultado obtido superou, não só o desempenho usual relatado pelo corpo médico do hospital, como também a meta definida pela direção.

O modelo desenvolvido foi capaz de gerar limites estreitos para a relaxação linear do problema inteiro. Para aproveitar este fato, foram desenvolvidas três heurísticas baseadas na formulação de programação inteira decompondo o problema em duas etapas e aproveitando os resultados da relaxação linear.

Os métodos construídos foram testados em 144 instâncias geradas aleatoriamente, mas buscam imitar o comportamento do sistema real. As instâncias geradas permitiram cobrir diferentes cenários de situações práticas.

Dentre os três métodos propostos, destaca-se a heurística *Relax-and-Fix* por ser

capaz de gerar soluções ótimas ou próximas da ótima. A heurística *Relax-and-Fix* 5% é uma variante que finaliza a busca pela solução na primeira etapa do método quando o *gap* de 5% é atingido. Essa variante se mostrou uma alternativa efetiva para encontrar boas soluções em tempos computacionais menores que uma hora.

Resultados computacionais mostraram que é possível encontrar solução ótima ou próxima da ótima para a maioria das instâncias. Os grupos de instâncias sem a restrição de janela de tempo para os cirurgiões e para as cirurgias possuem um número maior de variáveis e aparentam ser mais difíceis. Para 23 das 72 instâncias deste grupo não foi provada a otimalidade da melhor solução encontrada, isto é, essas instâncias permanecem abertas.

No capítulo 6 o problema foi tratado através de uma abordagem que considera a influência da incerteza e da dinamicidade no sistema. As incertezas de duração das cirurgias e da chegada de novas cirurgias foram incorporadas ao modelo. Reconhecendo a possibilidade de modificar o agendamento inicial ao longo do dia, a modelagem busca decidir de forma a minimizar o custo total esperado.

Para garantir a tratabilidade, o modelo para o problema dinâmico estocástico (PDE) foi construído a partir de uma simplificação do modelo determinístico proposto. No entanto, a modelagem apresentada é aderente às situações reais onde existam recursos humanos especializados em quantidade suficiente para alocar previamente à cada sala do bloco. Sob estas circunstâncias, o modelo se torna ainda mais adequado justamente por ser capaz de lidar com a incerteza e a natureza dinâmica do processo.

Durante a elaboração desta tese, não foram encontrados na literatura trabalhos que utilizassem a abordagem dinâmica estocástica para o tratamento de problemas de *allocation scheduling*. Acredita-se, portanto, que o modelo apresentado no Capítulo 6 seja uma contribuição para um tema pouco explorado.

Para a solução do PDE foram propostos e testados dois diferentes métodos. O primeiro, *Lookahead*, consiste na solução, a cada estágio de decisão, dos subproblemas de PI. Assume-se que o problema é determinístico. Este método serve como critério de comparação para a abordagem dinâmica frente à abordagem determinística.

O segundo método desenvolvido foi uma versão do *Policy Iteration Algorithm*

(PIA) com aproximação do valor função ($J^{(t)}()$) por regressão linear múltipla. Foram testados o método da suavização exponencial (SE) e o método dos mínimos quadrados recursivos (MQR) como forma de estimar os coeficientes da regressão a política.

A comparação prévia entre os métodos *PIAMQR* e *PIASE* indicou um melhor desempenho do primeiro e então este foi utilizado para realizar os testes computacionais relatados no Capítulo 7. Foram realizados 48 testes em instâncias geradas aleatoriamente mas que, assim como no caso determinístico, buscam imitar o comportamento de um sistema real. O caso real, no qual foram baseadas as instâncias, foi o bloco de cirurgias de um hospital de médio porte de Belo Horizonte-MG.

Ao todo foram gerados 8 tipos de instâncias com duas variações em cada e três processos de chegada. Os grupos se diferem em termos de quantidade média de cirurgias, disponibilidade de recursos e proporção entre a quantidade de eletivas e não eletivas. Com estes grupos buscou-se explorar diferentes cenários possíveis para a situação real de chegadas de cirurgias urgentes.

O desempenho dos métodos foi comparado através de um teste *t-student* unilateral e para 35 das 48 instâncias (72% dos casos) o método *PIAMQR* comprovou gerar um custo esperado estatisticamente menor do que o Método *Lookahead*. Em apenas uma instância o método *Lookahead* comprovou ser estatisticamente melhor do que o *PIAMQR*. Em média, o custo gerado pelo *PIAMQR* foi 19% menor do que o gerado pelo *Lookahead*. A diferença foi maior para o conjunto de instâncias com em média 18 cirurgias e com proporção eletiva/ não eletiva de 2 para 1. O método *PIAMQR* teve menores ganhos relativos em instâncias com processo de chegada de cirurgias não eletivas concentrado no início do expediente.

O desempenho do método *PIAMQR* frente ao método *Lookahead* demonstrou a abordagem dinâmica é capaz de gerar ganhos quando comparado com a solução gerada pela reotimização sucessiva do problema determinístico com a informação atualizada. Embora exista espaço para a melhoria, acredita-se que, uma vez que a abordagem dinâmica está sujeita à maldição da dimensão, o método desenvolvido representa uma contribuição desta tese.

A partir do que foi discutido e observado neste trabalho, é possível traçar caminhos

claros para pesquisas futuras, tanto utilizando a abordagem determinística, quanto a abordagem dinâmica estocástica.

Em termos de modelo para a abordagem determinística, pode-se discutir a incorporação de um horizonte de planejamento maior do que um dia. Esta demanda se deve principalmente à necessidade de se ter maior controle sobre a decisão de não agendar uma cirurgia para data corrente. O modelo seria capaz de articular a integração entre o *advanced scheduling* e o *allocation scheduling*. Ademais, em um horizonte maior, pode-se considerar como restrição a distribuição equitativa de tempo entre as diversas especialidades clínicas.

Do ponto de vista da abordagem para a solução, observou-se que em algumas situações a heurística *Relax-and-Fix*, após encontrar o valor da solução ótima na primeira etapa, não conseguia viabilizar uma alocação de recursos humanos especializados para o conjunto de cirurgias selecionadas. Portanto, pode haver muitas combinações de designação de salas e horários que levam ao valor ótimo da função objetivo, algumas destas viáveis em termos de alocação de recursos especializados, outras não.

Baseado nisto, uma linha para pesquisa futura é a introdução da ideia de *local branching* (FISCHETTI; LODI, 2003) na segunda etapa desta heurística. O modelo reduzido decidiria não só a alocação de recursos especializados, mas também poderia decidir mudar a designação de algumas cirurgias para salas e períodos dentro de uma vizinhança local obtida através da introdução de cortes do tipo *local branching*.

O propósito das vizinhanças do tipo *local branching* pode ser rearranjar o subconjunto de cirurgias entre salas e períodos de forma a elevar o número de cirurgias que possam ser aproveitadas. Em outras palavras, o *local branching* permitiria aumentar o número de cirurgias para as quais é viável a alocação de recursos especializados.

Um outra linha de pesquisa seria o desenvolvimento de uma metaheurística que, primeiramente, utilize regras heurísticas para selecionar as cirurgias e então aplique o modelo reduzido para alocar os recursos especializados. Pode-se utilizar, por exemplo, um procedimento do tipo *Variable Neighborhood Search* para aumentar

ou rearranjar o subconjunto de cirurgias candidatas à alocação.

Do ponto de vista prático, o desenvolvimento de uma metaheurística que garanta um tempo razoável de execução pode significar a diferença entre a implantação ou não do método em processos decisórios reais. Garantir um tempo razoável de execução significa não correr o risco de demandar um tempo excessivamente elevado para a encontrar a solução.

Novos aspectos da modelagem do problema por PDE podem ser explorados. Em termos da função objetivo do problema, a incorporação de custos de remarcação de cirurgias e do custo de hora extra adequariam o modelo para uma gama maior de situações reais.

Apesar do trabalho descrito nesta tese considerar indiretamente a incerteza de disponibilidade de recursos, esse aspecto pode ser modelado de forma a considerar outras situações que levam à indisponibilidade. São exemplos destas situações o atraso do recurso para início de sua janela de tempo e a indisponibilidade de um recurso material devido à quebra. Em termos de incerteza, pode-se considerar ainda a possibilidade de *No-show*, quando a cirurgia é cancelada por motivos relacionados ao paciente.

Um outro ponto que poderia ser incorporado na modelagem é a alocação de recursos humanos especializados. No entanto, esta questão aumentaria muito a complexidade do subproblema a ser resolvido a cada estágio de decisão e também da natureza da função a ser aproximada. Portanto, acreditamos que o desenvolvimento de algoritmos heurísticos eficientes para solução do problema determinístico é um pré-requisito para seguir nesta linha.

Quanto aos métodos para solução do problema dinâmico estocástico, existem inúmeros caminhos a serem seguidos. O caminho mais direto seria o desenvolvimento de procedimentos heurísticos e/ou metaheurísticos para a solução dos subproblemas combinados com os limitantes inferiores gerados pelos modelos de PI. Isto contribuiria para a melhoria do desempenho do método de solução não apenas com a redução do custo computacional do subproblema, mas também com a possibilidade de conhecer mais facilmente o espaço de estados durante o processo de aproximação. A possibilidade de uma investigação menos restritiva do

espaço de estados permitiria um maior controle sobre o *trade-off* entre diversificação (*exploration*) e intensificação (*exploitation*). Em programação dinâmica, a diversificação permite minimizar o vício existente na aproximação do valor da função causado pela realização do processo de estimação durante a aplicação de políticas já definidas (POWELL, 2011).

Lidar com os subproblemas de PLI implica na necessidade de criar funções indicadoras que possuam relações lineares com as variáveis. Este fato dificulta o uso de indicadores que estabeleçam relações não lineares com o valor a ser aproximado. Dificulta também investigar o impacto da interação entre dois indicadores na aproximação. A utilização de heurísticas para solução dos subproblemas geraria maior flexibilidade sob este aspecto.

Da mesma forma, a utilização de heurísticas permitiria o uso de outras metodologias de aproximação do valor da função, além da regressão linear. A regressão logística, a regressão kernel, as técnicas de classificação e regressão por árvores, e as redes neurais são exemplos de outros métodos comumente utilizados, segundo Powell (2011).

É necessário estudar a aplicação de técnicas de redução de variância durante a estimativa dos coeficientes da aproximação linear com o objetivo de gerar estimativas mais precisas ou reduzir o número de iterações necessárias.

A aplicação prática da abordagem determinística demonstrou que o uso da otimização pode gerar ganhos em termos do atingimento às metas do hospital. Os testes computacionais com o modelo dinâmico estocástico, demonstraram que o uso desta abordagem pode superar o uso sucessivo de modelos determinísticos na presença de incerteza.

Devido à pluralidade de aspectos considerados na modelagem, à dimensão dos problemas resolvidos e ao fato do tempo computacional demandado ser compatível com o horizonte da tomada de decisão, pode-se dizer que os modelos e métodos propostos por este trabalho possuem aderência e viabilidade de aplicação prática.

É objetivo da aplicação da Pesquisa Operacional dentro da Engenharia de Produção desenvolver modelos e métodos que sejam capazes de resolver de forma adequada problemas complexos e de relevância prática. Embora seja um grande avanço, a

existência destes tipos de modelos e métodos não garante, por si só, o uso dos mesmos nos processos decisórios das organizações. É necessário que se estude a problemática associada à implantação destes métodos. Como observado por Cardoen et al. (2010), esse tipo de estudo é muito pouco explorado na literatura e configura, portanto, uma oportunidade de pesquisa.

Referências

- BALL, M. O. Heuristics based on mathematical programming. *Surveys in Operations Research and Management Science*, v. 16, n. 1, p. 21–38, 2011.
- BEGEN, M. A.; QUEYRANNE, M. Appointment scheduling with discrete random durations. *Mathematics of Operations Research*, v. 36, n. 2, p. 240–257, 2011.
- BELIËN, J. Exact and heuristic methodologies for scheduling in hospitals: problems, formulations and algorithms. *4OR*, v. 5, p. 157–160, 2007.
- BELIËN, J.; DEMEULEMEESTER, E. A branch-and-price approach for integrating nurse and surgery scheduling. *European Journal of Operational Research*, v. 189, n. 3, p. 652 – 668, 2008.
- BERTRAND, J. W. M.; FRANSOO, J. C. Operations research methodologies using quantitative modeling. *International Journal of Operations & Production Management*, v. 22, n. 2, p. 241–264, 2002.
- BERTSEKAS, D. P. *Dynamic Programming and Optimal Control, vol. I*. [S.l.]: Athena Scientific, 2005.
- BERTSEKAS, D. P. *Dynamic Programming and Optimal Control, vol. II*. [S.l.]: Athena Scientific, 2012.
- BLAKE, J. T.; CARTER, M. W. Surgical process scheduling: a structured review. *Journal of the Society for Health Systems*, v. 5, n. 3, p. 17, 1997.
- BLAKE, J. T.; CARTER, M. W. A goal programming approach to strategic resource allocation in acute care hospitals. *European Journal of Operational Research*, v. 140, n. 3, p. 541 – 561, 2002.
- BLAKE, J. T.; DEXTER, F.; DONALD, J. Operating room managers' use of integer programming for assigning block time to surgical groups: A case study. *Anesthesia & Analgesia*, v. 94, n. 1, p. 143–148, 2002.
- BOLLAPRAGADA, S.; MORTON, T. E. A simple heuristic for computing nonstationary (s, s) policies. *Operations Research*, v. 47, 1999.
- BRUNI, M. E.; BERARDI, P.; CONFORTI, D. A stochastic programming approach for operating theatre scheduling under uncertainty. *IMA Journal of Management Mathematics*, v. 26, n. 1, p. 99–119, 2015.

- CARDOEN, B.; DEMEULEMEESTER, E.; BELIËN, J. Optimizing a multiple objective surgical case sequencing problem. *International Journal of Production Economics*, v. 119, n. 2, p. 354 – 366, 2009.
- CARDOEN, B.; DEMEULEMEESTER, E.; BELIËN, J. Sequencing surgical cases in a day-care environment: An exact branch-and-price approach. *Computers & Operations Research*, v. 36, n. 9, p. 2660 – 2669, 2009.
- CARDOEN, B.; DEMEULEMEESTER, E.; BELIËN, J. Operating room planning and scheduling: A literature review. *European Journal of Operational Research*, v. 201, n. 3, p. 921 – 932, 2010.
- CAYIRLI, T.; VERAL, E. Outpatient scheduling in health care: A review of literature. *Production and Operations Management*, v. 12, n. 4, p. 519–549, 2003.
- CESCHIA, S.; SCHAERF, A. Dynamic patient admission scheduling with operating room constraints, flexible horizons, and patient delays. *Journal of Scheduling*, p. 1–13, 2014.
- COLLART, D.; HAURIE, A. On a suboptimal control of a hospital inpatient admission system. *Automatic Control, IEEE Transactions on*, v. 21, n. 2, p. 233–238, 1976.
- DENTON, B.; VIAPIANO, J.; VOGL, A. Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health Care Management Science*, v. 10, p. 13–24, 2007.
- DENTON, B. T.; MILLER, A. J.; BALASUBRAMANIAN, H. J.; HUSCHKA, T. R. Optimal allocation of surgery blocks to operating rooms under uncertainty. *Operations Research*, v. 58, n. 4-Part-1, p. 802–816, 2010.
- DEXTER, F.; TRAUB, R. D. How to schedule elective surgical cases into specific operating rooms to maximize the efficiency of use of operating room time. *Anesthesia & Analgesia*, v. 94, n. 4, p. 933–942, 2002.
- FEI, H.; CHU, C.; MESKENS, N.; ARTIBA, A. Solving surgical cases assignment problem by a branch-and-price approach. *International Journal of Production Economics*, v. 112, n. 1, p. 96 – 108, 2008.
- FEI, H.; MESKENS, N.; CHU, C. A planning and scheduling problem for an operating theatre using an open scheduling strategy. *Computers & Industrial Engineering*, v. 58, n. 2, p. 221 – 230, 2010.
- FISCHETTI, M.; LODI, A. Local branching. *Mathematical Programming*, v. 98, n. 1-3, p. 23–47, 2003.

- GERCHAK, Y.; GUPTA, D.; HENIG, M. Reservation planning for elective surgery under uncertain demand for emergency surgery. *Management Science*, v. 42, n. 3, p. 321–334, 1996.
- GHAZALBASH, S.; SEPEHRI, M. M.; SHADPOUR, P.; ATIGHEHCHIAN, A. Operating room scheduling in teaching hospitals. *Advances in Operations Research*, p. 1 – 16, 2012.
- GRÖFLIN, H.; KLINKERT, A. Feasible insertions in job shop scheduling, short cycles and stable sets. *European Journal of Operational Research*, v. 177, n. 2, p. 763 – 785, 2007.
- GUERRIERO, F.; GUIDO, R. Operational research in the management of the operating theatre: a survey. *Health Care Management Science*, v. 14, p. 89–114, 2011.
- HANS, E.; WULLINK, G.; HOUDENHOVEN, M. van; KAZEMIER, G. Robust surgery loading. *European Journal of Operational Research*, v. 185, n. 3, p. 1038 – 1050, 2008.
- HSU, V. N.; MATTA, R. de; LEE, C.-Y. Scheduling patients in an ambulatory surgical center. *Naval Research Logistics (NRL)*, v. 50, n. 3, p. 218–238, 2003.
- HUGHES, W. L.; SOLIMAN, S. Y. Short-term case mix management with linear programming. *Hospital & health services administration*, v. 30, n. 1, p. 52, 1985.
- JEANG, A.; CHIANG, A.-J. Economic and quality scheduling for effective utilization of operating rooms. *Journal of Medical Systems*, v. 36, n. 3, p. 1205–1222, 2012.
- JEBALI, A.; ALOUANE, A. B. H.; LADET, P. Operating rooms scheduling. *International Journal of Production Economics*, v. 99, n. 1, p. 52 – 62, 2006.
- KIM, S.-C.; HOROWITZ, I. Scheduling hospital services: the efficacy of elective-surgery quotas. *Omega*, v. 30, n. 5, p. 335 – 346, 2002.
- KIM, S.-C.; HOROWITZ, I.; YOUNG, K. K.; BUCKLEY, T. A. Analysis of capacity management of the intensive care unit in a hospital. *European Journal of Operational Research*, Elsevier, v. 115, n. 1, p. 36–46, 1999.
- KOLESAR, P. A markovian model for hospital admission scheduling. *Management Science*, v. 16, n. 6, p. B–384, 1970.
- KUMAR, S.; MUTHURAMAN, K. A numerical method for solving singular stochastic control problems. *Operations Research*, v. 52, 2004.

- LAMIRI, M.; GRIMAUD, F.; XIE, X. Optimization methods for a stochastic surgery planning problem. *International Journal of Production Economics*, v. 120, n. 2, p. 400 – 410, 2009.
- LAMIRI, M.; XIE, X.; DOLGUI, A.; GRIMAUD, F. A stochastic model for operating room planning with elective and emergency demand for surgery. *European Journal of Operational Research*, v. 185, n. 3, p. 1026 – 1037, 2008.
- LI, C. L.; HSU, V. N.; XIAO, W. Q. Dynamic lot sizing with batch ordering and truckload discounts. *Operations Research*, v. 52, 2004.
- LIU, N.; ZIYA, S.; KULKARNI, V. G. Dynamic scheduling of outpatient appointments under patient no-shows and cancellations. *Manufacturing & Service Operations Management*, v. 12, n. 2, p. 347–364, 2010.
- MARCON, E.; KHARRAJA, S.; SIMONNET, G. The operating theatre planning by the follow-up of the risk of no realization. *International Journal of Production Economics*, v. 85, n. 1, p. 83 – 90, 2003.
- MARGERLIEN, J. M.; MARTIN, J. B. Surgical demand scheduling: A review. *Health Services Research*, v. 13, n. 4, p. 418–433, 1978.
- MARQUES, I.; CAPTIVO, M.; PATO, M. V. An integer programming approach to elective surgery scheduling. *OR Spectrum*, v. 34, n. 2, p. 407–427, 2012.
- MESKENS, N.; DUVIVIER, D.; HANSET, A. Multi-objective operating room scheduling considering desiderata of the surgical team. *Decision Support Systems*, v. 55, n. 2, p. 650 – 659, 2013.
- MIN, D.; YIH, Y. Scheduling elective surgery under uncertainty and downstream capacity constraints. *European Journal of Operational Research*, v. 206, n. 3, p. 642 – 652, 2010.
- PATRICK, J.; PUTERMAN, M. L.; QUEYRANNE, M. Dynamic multipriority patient scheduling for a diagnostic resource. *Operations Research*, v. 56, n. 6, p. 1507–1525, 2008.
- PHAM, D.-N.; KLINKERT, A. Surgical case scheduling as a generalized job shop scheduling problem. *European Journal of Operational Research*, v. 185, n. 3, p. 1011 – 1025, 2008.
- POWELL, W. B. *Approximate Dynamic Programming: Solving the curses of dimensionality*. [S.l.]: Wiley-Interscience, 2011.

- PRZASNYSKI, Z. H. Operating room scheduling. a literature review. *AORN Journal*, v. 44, n. 1, p. 67–79, 1986.
- RAIS, A.; VIANA, A. Operations research in healthcare: a survey. *International Transactions in Operational Research*, v. 18, n. 1, p. 1–31, 2011.
- RIISE, A.; BURKE, E. K. Local search for the surgery admission planning problem. *Journal of Heuristics*, v. 17, p. 389–414, 2011.
- ROBBINS, W.; TUNTIWONGPIBOOM, N. Linear programming a useful tool in case-mix management. *Healthcare financial management: journal of the Healthcare Financial Management Association*, v. 43, n. 6, p. 114, 1989.
- ROLAND, B.; MARTINELLI, C. D.; RIANE, F.; POCHET, Y. Scheduling an operating theatre under human resource constraints. *Computers & Industrial Engineering*, v. 58, n. 2, p. 212 – 220, 2010.
- SANTIBÁÑEZ, P.; BEGEN, M.; ATKINS, D. Surgical block scheduling in a system of hospitals: an application to resource and wait list management in a british columbia health authority. *Health Care Management Science*, v. 10, p. 269–282, 2007.
- SAREMI, A.; JULA, P.; ELMEKKAWY, T.; WANG, G. G. Appointment scheduling of outpatient surgical services in a multistage operating room department. *International Journal of Production Economics*, v. 141, n. 2, p. 646 – 658, 2013.
- SCIOMACHEN, A.; TANFANI, E.; TESTI, A. Simulation models for optimal schedules of operating theatres. *International Journal of Simulation*, v. 6, n. 12, p. 26–34, 2005.
- TIJMS, H. *Stochastic Models: An Algorithmic Approach*. [S.l.]: Wiley, 1994. (Wiley Series in Probability and Mathematical Statistics).
- VIJAYAKUMAR, B.; PARIKH, P. J.; SCOTT, R.; BARNES, A.; GALLIMORE, J. A dual bin-packing approach to scheduling surgical cases at a publicly-funded hospital. *European Journal of Operational Research*, v. 224, n. 3, p. 583 – 591, 2013.
- VREDEVELD, T. Stochastic online scheduling. *Computer Science-Research and Development*, v. 27, n. 3, p. 181–187, 2012.
- WAGNER, H. M.; WHITIN, T. M. Dynamic version of the economic lot sizing model. *Management Science*, v. 4, 1958.

WOLSEY, L. *Integer Programming*. [S.l.]: John Wiley and Sons, 1998.

WULLINK, G.; HOUDENHOVEN, M. V.; HANS, E. W.; OOSTRUM, J. M. van; LANS, M. van der; KAZEMIER, G. Closing emergency operating rooms improves efficiency. *Journal of Medical Systems*, v. 31, n. 6, p. 543–546, 2007.

ZONDERLAND, M.; BOUCHERIE, R.; LITVAK, N.; VLEGGEERT-LANKAMP, C. Planning and scheduling of semi-urgent surgeries. *Health Care Management Science*, v. 13, p. 256–267, 2010.