

**UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE – UERN**  
**FACULDADE DE CIÊNCIAS EXATAS E NATURAIS – FANAT**  
**DEPARTAMENTO DE INFORMÁTICA – DI**

Getúlio Ferreira da Silva Gomes

**OTIMIZAÇÃO NO AGENDAMENTO DE CIRURGIAS ELETIVAS MODELADO  
COMO JOB-SHOP SCHEDULING**

MOSSORÓ – RN 2016

Getúlio Ferreira da Silva Gomes

**OTIMIZAÇÃO NO AGENDAMENTO DE CIRURGIAS ELETIVAS MODELADO  
COMO JOB-SHOP SCHEDULING**

Monografia apresentada à Universidade do Estado do Rio Grande do Norte como um dos pré-requisitos para obtenção do grau de bacharel em Ciência da Computação, sob orientação do Prof. Dr. Francisco Chagas de Lima Júnior.

MOSSORÓ – RN, junho de 2016.

**Catálogo da Publicação na Fonte.  
Universidade do Estado do Rio Grande do Norte.**

Gomes, Getúlio Ferreira da Silva

Otimização no agendamento de cirurgias eletivas modelado como job-shop scheduling. / Getúlio Ferreira da Silva Gomes. - Mossoró, RN, 2016.

31 p.

Orientador(a): Prof. Dr. Francisco Chagas de Lima Júnior.

Monografia (Bacharel em Ciência da Computação).Universidade do Estado do Rio Grande do Norte.

1. Cirurgias Eletivas - Job-shop scheduling. 2. Métodos heurísticos. I. Lima Júnior, Francisco Chagas de. II. Universidade do Estado do Rio Grande do Norte. III.Título.

UERN/BC

CDD 004

Getúlio Ferreira da Silva Gomes

**OTIMIZAÇÃO NO AGENDAMENTO DE CIRURGIAS ELETIVAS MODELADO  
COMO JOB-SHOP SCHEDULING**

Monografia apresentada como pré-requisito para obtenção do título de Bacharel em Ciência da Computação da Universidade do Estado do Rio Grande do Norte – UERN, submetida à aprovação da banca examinadora composta pelos seguintes membros:

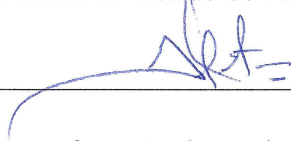
Aprovado em: 22/06/2016

Banca Examinadora



---

Prof. Dr. Francisco Chagas de Lima Júnior (Orientador)  
Universidade do Estado do Rio Grande do Norte – UERN



---

Prof. Dr. Carlos Heitor Pereira Liberalino  
Universidade do Estado do Rio Grande do Norte – UERN



---

Prof. Me. Jéssica Neiva de Figueiredo Leite  
Universidade do Estado do Rio Grande do Norte – UERN

Aos meus pais Gilberto e Noelma que,  
através dos seus esforços, me  
proporcionaram educação e disciplina para  
crescer e agir de forma digna.

## **AGRADECIMENTOS**

Gostaria de agradecer primeiramente à Deus, pois sem ele eu não chegaria à lugar algum, ao meu orientador, colegas e professores que possibilitaram grande aprendizado dentro da minha jornada acadêmica e aos meus amigos e familiares por me apoiarem.

Soli Deo gloria

## RESUMO

O problema de agendamento de salas cirúrgicas consiste em alocar recursos hospitalares (sala e equipe médica) para casos de intervenção cirúrgica, além da determinação do instante ideal (ou ótimo) para a realização do procedimento. O problema de otimização do agendamento de cirurgias eletivas proposto neste trabalho, corresponde a um caso particular de *Job Shop Scheduling*, e pode ser modelado mais especificamente, como um problema de agendamento de máquinas paralelas idênticas com tempos de preparação de máquina dependente da sequência. O problema foi aqui intitulado de Problema de Otimização de Agendamento d Cirurgias eletivas – POACE e para este são propostos um modelo exato e dois métodos heurísticos – um algoritmo guloso aleatório e um método de busca local *2-opt*. Os algoritmos propostos foram testados com dados obtidos no Centro de Oncologia e Hematologia de Mossoró-RN, tendo tais dados utilizados de forma “blindada”, ou seja, sem a identificação de médicos, pacientes ou tipo de procedimento cirúrgico. A consolidação dos métodos propostos no presente trabalho possibilitará a implementação de metaheurísticas, em particular a GRASP para o citado problema.

**Palavras-chaves:** Cirurgias Eletivas, *Job-shop scheduling*, Métodos heurísticos



## ABSTRACT

The Surgical Case Scheduling problem it consists in allocate hospital resources (living and medical staff) to case of surgical intervention, in addition to determining the ideal moment (or optimal time) for the procedure. The optimization problem of scheduling elective surgery proposed here corresponds to a particular case of Job Shop Scheduling, and more specifically can be modeled as an identical parallel machine scheduling problem with sequence dependent setup times. The problem here was titled “scheduling optimization problem of elective surgeries” and for this are proposed an exact model and two heuristics - a random greedy algorithm and method of local search  $2-opt$ . The algorithms proposed was tested using data of the Cancer Center and Hematology Mossoro-RN., And such data used in an "armored", that is, without identifying physicians, patients or surgical procedure. The consolidation of the methods proposed in this paper will enable the implementation of metaheuristics, particularly GRASP to the aforementioned problem.

**Key-words:** Elective surgeries, *Job-shop scheduling*, Heuristic methods.

## **ÍNDICE DE FIGURAS**

Figura 1. Grafo disjuntivo – Adaptado de Pinedo (2002).....	16
Figura 2. Lista Restrita de Candidatos – LRC (Fonte – Lima Júnior, 2009) .....	18
Figura 3. Exemplo de busca local 2-opt, para o PCV (Fonte – Lima Júnior, 2009).....	19
Figura 4. Pseudocódigo para heurística construtiva gulosa aleatória para o POACE. ....	24
Figura 5. Pseudocódigo para heurística de busca local para o POACE. ....	25
Figura 6. Esboço da construção gulosa aleatória e instante de tempo da busca local .....	26

# SUMÁRIO

INTRODUÇÃO .....	9
1. Motivação .....	9
2. Objetivo.....	10
3. Contribuições do Trabalho .....	10
4. Organização do Texto .....	11
II. FUNDAMENTAÇÃO TEÓRICA.....	12
1. Problemas de otimização combinatória .....	12
2. O Problema de <i>Job-shop Scheduling</i> .....	13
3. Modelo padrão de Programação Linear Inteira para JSP .....	13
4. Representação de JSP utilizando grafo disjuntivo.....	15
5. O Problema de Otimização de agendamento de cirurgias eletivas - POACE .....	16
6. Metaheurísticas.....	17
6.1. Metaheurística GRASP .....	17
III. TRABALHOS RELACIONADOS.....	19
IV. ABORDAGEM PROPOSTA.....	20
1. Modelo Matemático para o POACE.....	20
2. Aspectos práticos relativos ao Problema.....	23
3. Heurísticas Propostas para o Problema .....	23
4. Heurísticas Construtiva:.....	23
5. Heurísticas de busca local 2-opt: .....	24
6. Construção de Instâncias com os dados reais.....	25
V. RESULTADOS .....	27
VI. CONCLUSÕES E TRABALHOS FUTUROS .....	27
REFERÊNCIAS.....	28

# INTRODUÇÃO

## 1. Motivação

A gestão estratégica de serviços de saúde em hospitais públicos ou privados consiste em uma problemática de alta complexidade, dado o grande número de variáveis a serem consideradas no processo operacional deste contexto. Dentre os principais fatores que influenciam e desafiam o gestor neste processo decisório estão: necessidade de redução de custos, atendimento de prazos legais estabelecidos e, principalmente, a melhoria no atendimento aos pacientes durante o tratamento.

Em um hospital de grande porte, o centro cirúrgico consiste em uma das unidades que requer a maior atenção da gestão, pois trata-se de um dispositivo indispensável na realização de procedimentos essenciais à vida, além de ser a unidade de maior custo operacional nestes hospitais. Dentre os diversos tipos de cirurgias a serem realizadas em um centro cirúrgico, existem as denominadas cirurgias eletivas que consiste em cirurgias previamente conhecidas ou planejadas. São exceções a estes tipos de cirurgias àquelas consideradas cirurgias de emergência.

A problemática então abordada consiste em propor o melhor agendamento de salas cirúrgicas, definindo qual a sequência de priorização ideal, bem como, qual o melhor instante de realização de cada uma das cirurgias. A solução do problema é estabelecer uma agenda cirúrgica, com o objetivo de minimizar o horário de término da última cirurgia.

Segundo Carter & Tovey (1992) o problema de agendamento de cirurgias eletivas é considerado na literatura um problema clássico de otimização combinatória, pertencente à classe NP-Difícil. Portanto, o uso de métodos exatos de otimização é inviável para resolver o problema, do ponto de vista de tempo de resolução e custo computacional.

Como já mencionado, as cirurgias de emergência necessitam de salas e equipes médicas que estejam de plantão para atender os pacientes. Tendo em vista que é algo de extrema importância e totalmente imprevisível, seria necessário sempre haver uma sala disponível para cirurgias emergenciais, isso faria com que as cirurgias eletivas fossem supridas com as salas restantes, deixando assim uma equipe e uma sala ociosa. Visando minimizar a ocorrência de situações como as descritas anteriormente, propõe-se neste trabalho a modelagem do problema de agendamento de cirurgias eletivas como um caso particular de *Job-shop Scheduling* - JSP, aplicado a um estudo de caso utilizando dados

reais do Centro de Oncologia e Hematologia da cidade de Mossoró, no Rio Grande do Norte.

O método de otimização desenvolvido neste trabalho poderá ser aplicado como técnica de apoio à tomada de decisão no processo de agendamento de cirurgias eletivas em hospitais públicos ou privados, contribuindo assim com a melhoria de vários aspectos relevantes tais como: melhoria da qualidade do atendimento aos pacientes, através da redução do tempo de espera pela intervenção cirúrgica, redução de custos operacionais e melhoria na gestão dos recursos hospitalares, uma vez que a otimização do agendamento das cirurgias contribuirá também para a redução da ociosidade dos equipamentos e dos recursos humanos relativos ao centro cirúrgico.

## **2. Objetivo**

### **2.1. Objetivo Geral:**

Elaborar um modelo de otimização para aplicação como técnica de apoio à tomada de decisão no processo de agendamento de cirurgias eletivas em hospitais públicos ou privados.

### **2.2. Objetivos Específicos:**

- Modelar o problema de agendamento de cirurgias eletivas como um caso particular de *Job-shop Scheduling*.
- Elaborar instancias de testes para o problema proposto utilizando dados hospitalares reais.
- Propor heurísticas (construtivas e de busca local) aplicadas à problemática em estudo, para utilização na implementação de metaheurísticas – em particular para a metaheurística GRASP.

## **3. Contribuições do Trabalho**

O presente trabalho propõe a aplicação de um modelo baseado *em Job-shop scheduling* para otimização de agendamento de cirurgias eletivas, tendo como estudo de caso a utilização de dados do Centro de Oncologia e Hematologia de Mossoró. O modelo e as heurísticas propostas e as instâncias de testes produzidas poderão ser utilizados na implementação de métodos exatos ou aproximados a fim de melhorar a quantidade de atendimentos, o tempo de espera e a disponibilidade para cirurgias de emergência.

#### **4. Organização do Texto**

A escrita do texto deste trabalho está organizada no formato de seções, a seção II apresenta uma breve fundamentação teórica sobre problemas de otimização combinatória como *Job-shop scheduling - JSP*, em particular o caso do problema de agendamento de máquinas paralelas idênticas com tempos de preparação de máquina dependente da sequência. Nesta seção são ainda apresentados um modelo matemático clássico para o *JSP*, e tal problema modelado como um grafo disjuntivo. A seção III apresenta alguns trabalhos relacionados ao tema. A seção IV apresenta a abordagem proposta, a seção V relata os resultados obtidos, enquanto a seção VI apresenta as conclusões e indica algumas possibilidades de trabalhos futuros.

## II. FUNDAMENTAÇÃO TEÓRICA

### 1. Problemas de otimização combinatória

Na otimização combinatória são realizados estudos sobre problemas que possuem um número finito de soluções possíveis e, portanto, possuem uma solução ótima partindo de uma listagem de todas estas soluções. Infelizmente este método é inviável tendo em vista o grande número de soluções possíveis. Formalmente, um problema de otimização combinatória -POC, consiste em encontrar uma solução  $X$  para determinado problema  $P$ , levando em consideração um conjunto de restrições  $R$ , submetido a determinados critérios  $C$ . O grau de dificuldade inerente a resolução de um POC, está relacionado ao crescimento exponencial do tempo gasto para analisar todas as possibilidades de solução, à medida que as instâncias do problema crescem.

O problema clássico de otimização associado ao contexto da problemática abordada neste trabalho, corresponde ao Problema de Agendamento de Salas Cirúrgicas (do inglês, *Surgical Case Scheduling* - SCS), que consiste em alocar recursos hospitalares para casos de intervenção cirúrgica, além da determinação do instante ideal (ou ótimo) para a realização do procedimento. O problema de otimização do agendamento de cirurgias eletivas aqui proposto, pode ser visto como um caso particular de *Job Shop Scheduling*, e pode ser modelado mais especificamente, como um problema de agendamento de máquinas paralelas idênticas com tempos de preparação de máquina dependente da sequência (ou em inglês - *Identical Parallel Machine Scheduling Problem with Sequence Dependent Setup Times*).

O Problema de SCS foi considerado por Carter & Tovey (1992) um problema clássico de otimização combinatória, pertencente à classe NP-Difícil, e, portanto, tem sua resolução computacional inviabilizada via métodos exatos (ou exaustivos), visto que o custo computacional envolvido no processo tornaria inviável a tomada de decisão em tempo real. A implicação prática da classificação de um dado problema como NP-Difícil, é a constatação teórica da não existência de algoritmos exatos que garantam resolver instâncias de grande porte, de forma ótima, em um tempo computacional aceitável (polinomial).

Na abordagem de problemas NP-difíceis, as metaheurísticas são métodos que tem obtido relevante sucesso quando aplicadas a problemas da classe NP, tais como: roteamento

ou escalonamento de veículos [Backer et al. 2000], empacotamento de caixas em containers [Faroie et al. 2003], projetos de computadores e de chips VLSI [Wang & Chen 1995], logística de produção e transporte de petróleo [Barros 2001], sequenciamento de DNA [Blazewicz et al. 2004], alocação de trabalhadores ou máquinas a tarefas [Miller et al. 1999], dentre outros.

## 2. O Problema de *Job-shop Scheduling*

O clássico problema de *Job-shop Scheduling* – JSP, ou programação de tarefas, consiste em um conjunto independente de postos de trabalho – ou tarefas, cada um com sua própria ordem de processamento através de um conjunto de máquinas. Cada tarefa tem um conjunto ordenado de operações, cada uma das quais deve ser processada em uma máquina pré-definida, o que se deseja é minimizar o instante de término da última tarefa no contexto geral do problema (do inglês *makespan*). O problema JSP é conhecido por ser fortemente NP-árduo (Johnson, Zwaneveld et al., Strusevich), também considerado por Yamada e Nakano, como um dos mais complexos membros desta classe.

Mais formalmente, o problema JSP pode ser formulado como: dado um conjunto de  $M = \{m_1, m_2, \dots, m_i, \dots, m_M\}$  máquinas, um conjunto  $N = \{t_1, t_2, \dots, t_j \dots t_N\}$  de tarefas e conjunto de operações  $O = \{o_1, o_2, \dots, o_k, \dots, o_M\}$ , definidas para cada tarefa  $j$ . A sequência de operações deve ser estabelecida e determinada no início de funcionamento do ambiente (início da produção da fábrica, por exemplo), cada operação deve ser executada em uma única máquina por um determinado período contínuo de tempo e cada máquina pode realizar apenas uma tarefa por vez. O objetivo do problema é minimizar o instante de término da tarefa  $t_N$ , ou seja, o tempo de finalização da última operação completada no sequenciamento.

Uma visão geral com histórico das principais técnicas utilizadas, bem como, importantes resultados teóricos obtidos para o JSP, podem ser encontrados em Jain and Meeran (1999).

## 3. Modelo padrão de Programação Linear Inteira para JSP

Existem diferentes formulações para o problema de *Job-shop Scheduling*, dentre as quais destacam-se a de Bowman (1959), a de Wagner (1959) e a formulação inteira mista de Manne (1960), como as primeiras publicações que se registram na literatura. Neste trabalho será adotado como modelo padrão o apresentado por Aiex et al., (2003), o qual será descrito a seguir:



Dado um conjunto de  $M$  máquinas, sendo  $|M|$  a cardinalidade do conjunto  $M$ , e o conjunto  $T$  de tarefas (com  $|T|$  sendo cardinalidade do conjunto  $T$ ), seja  $O_1^j \gg O_2^j \gg \dots \gg O_M^j$  o conjunto ordenado das  $|M|$  operações da tarefa  $j$ , onde  $O_k^j \gg O_{(k+1)}^j$  indica que a operação  $O_{(k+1)}^j$  somente pode ser iniciada após o término da operação  $O_k^j$ . Considere  $O$  o conjunto de operações possíveis. Cada operação  $O_k^j$  é definida por dois parâmetros:  $M_k^j$  representa a máquina onde a tarefa  $O_k^j$  será processada e  $p_k^j = p(O_k^j)$  é o tempo de processamento da operação  $O_k^j$ . Adotando  $t(O_k^j)$  como o instante de início da  $k$  –ésima operação  $O_k^j \in O$ , a formulação para o JSP é dada da seguinte forma:

*Minimizar*  $C_{max}$

*Sujeito à:*

$$C_{max} \geq t(O_k^j) + p(O_k^j), \quad \forall O_k^j \in O, \quad (1)$$

$$t(O_k^j) \geq t(O_i^j) + p(O_i^j) \quad \forall O_i^j \ll O_k^j, \quad (2)$$

$$t(O_k^j) \geq t(O_i^i) + p(O_i^i) \vee t(O_k^j) \geq t(O_i^j) + p(O_i^j) \quad \forall i, j \in T : M_{O_i^i} = M_{O_i^j}, \quad (3)$$

$$t(O_k^j) > 0, \quad \forall O_k^j \in O, \quad (4)$$

Onde  $C_{max}$  é o *makespan* a ser minimizado.

As restrições em (1) garantem que a sequência de processamento das operações em cada tarefa correspondem a uma pré-determinada ordem, garantindo que  $C_{max}$  assumirá o valor da data de finalização da última tarefa do sequenciamento. As restrições em (2) exigem que exista uma única tarefa em cada máquina em um determinado instante de tempo. E as restrições (3) garantem que uma vez iniciado o processamento de uma operação, ele não pode ser interrompido até a sua conclusão. Qualquer solução viável (uma permutação de  $T$  em cada uma das máquinas em  $M$ ) que respeite as restrições (1), (2), (3) e (4) é chamada uma programação, sequenciamento, agendamento ou *scheduling* para a situação problema.

Considerando os aspectos práticos relativos à complexidade computacional do problema JSP, é importante salientar que nos casos em que tal problema consiste de  $n$  tarefas e  $m$  máquinas o número de possíveis agendamentos é extremamente grande, isto é,  $(n!)^m$  possibilidades e assim, as melhores técnicas de resolução são os métodos

aproximativos ou heurísticos baseados em sequenciamento ou regras de prioridades [Domdorf and Pesch, 1995].

#### 4. Representação de JSP utilizando grafo disjuntivo

Segundo Pinedo (2002), um grafo disjuntivo, denotado por  $G = (N, A, B)$  é um grafo direcionado  $G$ , com um grupo de  $N$  vértices e dois grupos de arcos  $A$  e  $B$ . Os vértices  $N$  correspondem todas as operações  $(i, j)$  que precisam ser executadas pelas  $n$  tarefas. Os chamados arcos conjuntivos  $A$  representam as rotas relativas à realização das tarefas. Se o arco  $(i, j) \rightarrow (k, j)$  pertence à  $A$ , então a tarefa  $j$  tem que ser processado na máquina  $i$  antes de ser processada na máquina  $k$ , ou seja, operações  $(i, j)$  precedem operações  $(k, j)$ , duas operações que procedem de duas tarefas diferentes e que tem que ser processadas na mesma máquina são conectadas por dois arcos chamados disjuntivos que vão em direções opostas. Os arcos disjuntivos  $B$  formam  $m$  grafos de arcos duplos, um grafo para cada máquina. Todas as operações (vértices) no mesmo grafo tem que ser realizadas na mesma máquina. Todos os arcos originados de um vértice conjuntivo ou de um vértice disjuntivo tem o mesmo tempo de processamento da operação que é representada pelo vértice. Além disso, existe uma origem  $U$  e um destino  $V$ , o vértice de origem  $U$  tem  $n$  arcos conjuntivos procedendo para a primeira operação das  $n$  tarefas, e o vértice destino  $V$  tem  $n$  arcos conjuntivos chegando de todas as operações anteriores. Os arcos procedidos da origem têm comprimento zero.

No grafo  $G$  é possível identificar dois subgrafo distintos, o subgrafo  $D$  que, representado pelos arcos disjuntivos selecionados e o subgrafo próprio  $G(D)$  definido pelo grupo dos arcos conjuntivos e pelo subgrafo  $D$ . O *makespan* de um possível agendamento é determinado pelo caminho mais longo do grafo  $G(D)$  da origem  $U$  para o destino  $V$ . Este caminho mais longo consiste em um grupo de operações das quais a primeira começa com o tempo zero e a última termina com o tempo do *makespan*. Cada operação sobre este caminho é imediatamente seguida por uma outra operação na mesma máquina ou a próxima operação de um mesmo job em outra máquina. O problema de minimização do *makespan* consiste em uma seleção de arcos disjuntivos que minimizam o comprimento do caminho mais longo. A *figura 1* apresenta, à seguir, um exemplo de grafo disjuntivo.

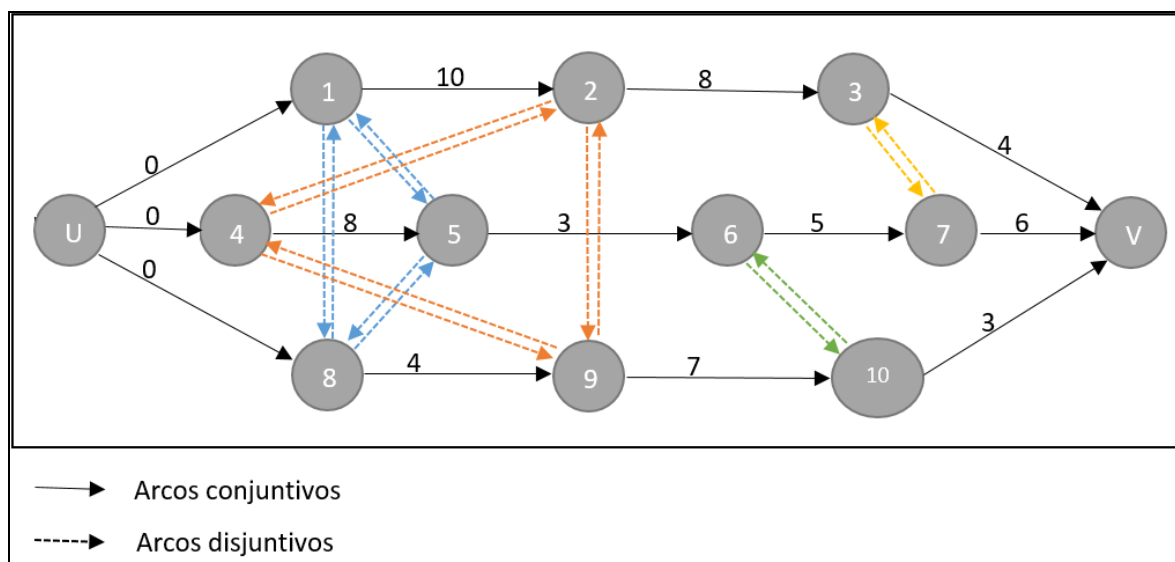


Figura 1. Grafo disjuntivo – Adaptado de Pinedo (2002)

## 5. O Problema de Otimização de agendamento de cirurgias eletivas - POACE

Dados do Portal da Saúde (site do governo federal), documentam que o número de cirurgias eletivas no Brasil cresceu 11,7% em dois anos, passando de 2.120.580 em 2012 para 2.370.039 em 2014. São exemplos de cirurgias eletivas as cirurgias de varizes, ortopédicas, de urologia e otorrinolaringologia, dentre outras. Estas cirurgias caracterizam-se pela ausência do caráter de urgência, comum em muitos outros tipos de intervenções cirúrgicas.

O aspecto não emergencial deste tipo de cirurgia permite que o gerenciamento logístico da mesma possa ser planejado a priori. Apesar do conhecimento prévio da demanda das cirurgias eletivas, se planejamento ou agendamento sequencial depende de uma série de fatores que contribuem para a contextualização de uma problemática de alta complexidade, dado o grande número de variáveis envolvidas no modelo de resolução.

De acordo com Proença (2010) o planejamento de cirurgias eletivas classifica-se em três fases: planejamento de casos mistos, planejamento mestre de cirurgias e agendamento de casos eletivos. A fase de planejamento de casos mistos é a fase responsável pela análise de disponibilidade em horas das salas de cirurgias a serem distribuídas para os diversos cirurgiões, fase que é realizada anualmente.

A segunda fase, a de planejamento mestre de cirurgias, enquadra-se em um nível estratégico da gestão hospitalar e consiste no desenvolvimento de uma agenda cirúrgica - um documento cíclico, que define o número e o tipo de salas de operações disponíveis, as horas em que as salas estão funcionando, definindo, ainda, cirurgiões ou grupos de cirurgias que tem prioridade de uso sobre o tempo das salas cirúrgicas.

Na terceira e última fase do processo de planejamento é estabelecido o agendamento de cada cirurgia em uma base diária. Esta fase situa-se em um nível operacional e consiste em agendar cirurgias previamente conhecidas (eletivas), desconsiderando casos de cirurgias de emergência. O foco da pesquisa realizada neste trabalho está exatamente em otimizar o agendamento cirurgias de casos eletivos.

O problema a ser resolvido será então tratado como um caso particular de *Job-shop scheduling*, que é o caso de problema de sequenciamento de tarefas em máquinas paralelas idênticas, com tempo de preparação dependente da sequência, onde um conjunto de  $T$  tarefas, serão distribuídas para uma única máquina  $M$ , realizá-la, sendo que nenhuma das outras máquinas estejam realizando a mesma tarefa. Cada máquina só pode realizar uma tarefa depois que a máquina anterior terminar sua execução.

## 6. Metaheurísticas

As metaheurísticas são métodos aproximativos que dependem de boas estratégias de intensificação/diversificação baseadas em conhecimento prévio do problema, que podem guiar o processo de busca para uma solução ótima global, com o propósito de fugir da armadilha de ficar preso em um ótimo local. Boas estratégias (ou boas heurísticas) alternam de forma adequada entre os processos de intensificação e diversificação, mantendo o equilíbrio entre estes dois processos durante a busca pela solução ótima. As metaheurísticas se diferenciam das heurísticas convencionais por seu caráter estocástico que possibilita a fuga dos pontos de ótimo local. Os ótimos locais são responsáveis pela estagnação do processo de busca pela solução ótima em problemas de otimização combinatória.

### 6.1. Metaheurística GRASP

A metaheurística GRASP - Greedy Randomized Adaptive Search Procedure proposta por Feo e Resende, (1995) é um processo iterativo multipartida onde cada iteração é composta por duas fases: uma fase de construção e uma fase de melhoria - normalmente busca local. Na fase de construção uma solução viável para o problema é criada e a fase de busca local tem como objetivo tentar melhorar a solução obtida na fase anterior. As iterações GRASP são independentes, isto é, na iteração corrente não se leva em conta nenhuma informação das iterações anteriores. O critério de parada normalmente usado, é um número máximo de iterações. Ao final da execução do GRASP a melhor solução até então obtida, é a solução final para o algoritmo. Maiores detalhes sobre a metaheurística GRASP podem ser encontrados em Lima Júnior (2009).

Na fase construtiva do GRASP uma solução é construída um elemento por vez, de forma iterativa, até que uma solução viável para o problema seja concluída. Em qualquer passo intermediário deste processo de construção uma solução incompleta é dita uma solução parcial para o problema.

Cada inserção de um novo elemento na solução parcial é feita através da escolha aleatória em uma lista restrita de candidatos (LRC). Existem basicamente duas estratégias de construção de uma LRC, uma baseia-se na cardinalidade da LRC, outra na qualidade dos elementos que a compõem. A figura 2 apresenta de forma resumida o funcionamento da LRC e a utilização do parâmetro  $\alpha$  na determinação do critério de gula e aleatoriedade com base na cardinalidade da LRC.

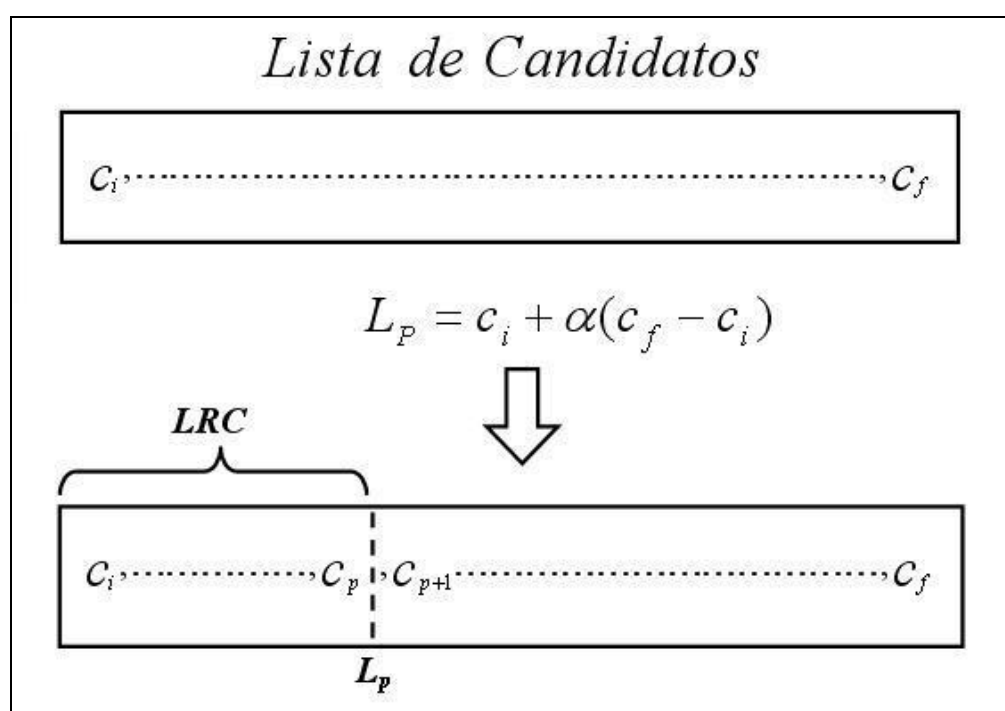


Figura 2. Lista Restrita de Candidatos – LRC (Fonte – Lima Júnior, 2009)

A fase de melhoria da metaheurística GRASP consiste tipicamente de um procedimento de busca local que visa aperfeiçoar a solução obtida na fase de construção que nem sempre é um ótimo local. A busca local atua de forma iterativa através da substituição sucessiva da solução corrente pela melhor encontrada em sua vizinhança. Existem buscas que trocam de soluções logo que uma solução de melhor qualidade é encontrada e outras que avaliam todos os vizinhos e somente depois substituem a solução corrente pela mais interessante até então encontrada.

Existem diversos algoritmos de busca local e sua classificação pode ser feita com base no tamanho da vizinhança explorada, ou seja, considerando o número de movimentos que podem ser utilizados para transformar uma rota em outra. Dentre tais algoritmos, um dos mais famosos é o *2-Opt* que foi proposto inicialmente por Croes (1958). A figura 3 apresenta um exemplo de busca local *2-Opt* para o problema do caixeiro viajante.

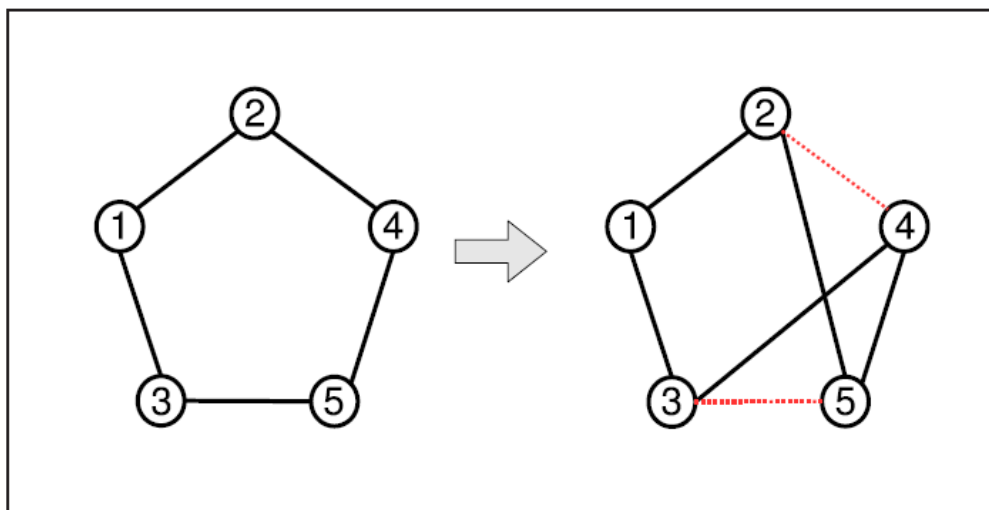


Figura 3. Exemplo de busca local 2-opt, para o PCV (Fonte – Lima Júnior, 2009).

### III. TRABALHOS RELACIONADOS

Existem na literatura algumas publicações recentes sobre o problema de agendamento de cirurgias eletivas. Nesta seção serão apresentadas alguns destes trabalhos, com destaque para suas metodologias, contribuições científicas e resultados obtidos.

Em Andrade, et al. (2012) os autores apresentam uma proposta de resolução para o problema de agendamento de cirurgias eletivas utilizando uma abordagem por metaheurísticas. Eles modelaram o problema como um caso de programação em máquinas paralelas idênticas com tempos de preparação de máquina dependente da sequência e utilizando as metaheurística GRASP - *Greedy, Randomized Iterated Search Procedure* e ILS (do inglês, *Iterated Local Search*) e desenvolveram experimentos computacionais que permitiram reduzir o uso de recursos nas cirurgias eletivas em dois hospitais da cidade de Belo Horizonte/MG. Os resultados obtidos no trabalho possibilitaram um considerável ganho de tempo e recurso ao se realocar as cirurgias.

Em Teixeira & Chang (2012) os autores apresentam uma proposta, utilizando o Data Mart, para resolver o problema de superlotação de cirurgias eletivas do Hospital São Paulo - UNIFESP. Através de simulações feitas com base em informações como: data e

tipo da cirurgia; horário de entrada e saída da sala e do centro cirúrgico; horário de início e fim da cirurgia e da anestesia. Como resultado destas simulações foi possível estimar um tempo de duração de cada tipo de cirurgia, desenvolvendo à partir destas estimativas uma central de agendamento que utilizaria o *Data Mart* para preencher o tempo disponível de cada sala, evitando superlotação de uma sala e tempo ocioso de outra.

Em Mehdi et al (2007) os autores apresentam uma proposta para reduzir os custos e tempo excessivo do uso das salas de cirurgias, tendo como referência que os custos com cirurgias em geral passam de 40% dos gastos de um hospital. Foi feita uma análise utilizando a simulação de Monte Carlo, que são simulações sucessivas repetidas por um elevado número de vezes para calcular probabilidades heurísticas, combinada com uma abordagem de geração de colunas. Como resultado foi possível concluir que é possível gerar uma solução próxima à ótima para variáveis com valores aleatórios como as cirurgias eletivas e de emergência.

Em Essen et al (2011) os autores apresentam uma proposta para conciliar as cirurgias de emergência com as cirurgias eletivas e reduzir assim o tempo de espera dos pacientes de emergência e custos com ambas as cirurgias. Foi modelado o problema como um caso de programação em máquinas paralelas idênticas com tempos de preparação de máquina dependente da sequência e utilizando as metaheurísticas *Simulated Annealing* (Recozimento Simulado) e *Tabu Search* (Busca Tabu). Como resultado foi possível reduzir o intervalo entre as cirurgias, ou BII (*Break-in-interval*), em 20% e concluir que o algoritmo “*Fixed Goal Values*” que visa aplicar o BII de forma adequada para reduzir o tempo e conseguir ganhar tempo para antecipar as cirurgias é viável.

#### **IV. ABORDAGEM PROPOSTA**

Nesta seção serão apresentados em maiores detalhes os aspectos relativos à abordagem proposta para o problema estudado.

##### **1. Modelo Matemático para o POACE**

Nesta subseção será apresentado de maneira mais formal o Problema de Otimização de Agendamento de Cirurgias Eletivas – POACE, o qual será modelado como o problema de sequenciamento de tarefas em máquinas paralelas idênticas, com tempo de preparação

dependente da sequência. O modelo aqui apresentado baseia-se no adotado por Ravetti & Mateus (2003), o qual será descrito a seguir.

Como mencionado, o POACE será caracterizado como um caso particular de Job-shop scheduling, no caso, problema de programação em máquinas paralelas, para isso na construção do modelo matemático será considerada a equivalência entre máquina e sala de cirurgia; e tarefa e cirurgia.

Considere um conjunto de  $S$  salas cirúrgicas, sendo  $|S|$  a cardinalidade do conjunto  $S$ , e o conjunto  $C$  de cirurgias a serem realizadas (com  $|C|$  sendo cardinalidade do conjunto  $C$ ). Uma vez que uma cirurgia começa a ser realizada nenhuma interrupção é permitida, ou seja, para poder começar outra cirurgia a anterior deve ter sido concluída.

Em uma sala pode ser realizada uma cirurgia somente depois de ela ter sido preparada adequadamente e o tempo de preparação gasto por essa sala depende não só do tipo cirurgia a ser realizada, mas também da cirurgia que estava sendo realizada anteriormente na mesma, ou seja, o tempo de preparação das salas depende da sequência em que as cirurgias são realizadas.

Considere a seguinte notação para o problema:

#### **Parâmetros:**

$Q$ : Quantidade de cirurgias a serem realizadas.

$i$ : indexação para as cirurgias a serem realizadas,  $i = 1, 2, \dots, Q$ .

$S$ : Quantidade de salas de cirurgias disponíveis.

$k$ : indexação para as salas de cirurgias,  $k = 1, 2, \dots, S$ .

$rt_i$ : Tempo de realização da cirurgia  $i$  em qualquer uma das de salas.

$st_i$ : Tempo de início da cirurgia  $i$  em qualquer uma das salas.

$p_{ij}$ : Tempo de preparação da sala entre a cirurgia  $i$  e a cirurgia  $j$ .

$T$ : Tempo necessário para a conclusão de todas as cirurgias.

$M$ : Um número muito grande.

#### **Variáveis de decisão:**

$X_{ik}$ :  $X_{ik} = 1$ , se a cirurgia  $i$  é realizada na sala  $k$ ,  $X_{ik} = 0$ , caso contrário.

$Y_{ijk}$ :  $Y_{ijk} = 1$ , se a cirurgia  $i$  precede a cirurgia  $j$ , na sala  $k$ ,

$Y_{ijk} = 0$ , caso contrário.

Minimizar  $T_{max}$



*Sujeito à:*

$$T_{max} \geq st_i + rt_i \quad \forall i \quad (1)$$

$$\sum_k^s X_{ik} = 1 \quad \forall i \quad (2)$$

$$(1 - X_{ik}).M + (1 - X_{jk}).M + (1 - Y_{ijk}).M + st_j \geq st_i + rt_i + p_{ij} \quad \forall i, j, k \quad (3)$$

$$(1 - X_{ik}).M + (1 - X_{jk}).M + Y_{ijk}.M + st_i \geq st_j + rt_j + p_{ji} \quad \forall i, j, k \quad (4)$$

$$\sum_i^q X_{ik} \geq \sum_i^q X_{ik-1} \quad \forall k > 1 \quad (5)$$

$$X_{ik}, Y_{ijk} \in \{0, 1\} \quad (6)$$

### Explicitação do modelo

No que diz respeito a função objetivo o que se deseja é minimizar o instante de término da última tarefa  $T_{max}$  que corresponde ao makespan máximo para o problema. As restrições em (1) garantem que  $T_{max}$  assumirá o valor da data de finalização da última cirurgia agendada. As restrições em (2) garantem que cada cirurgia será agendada em uma única sala.

Restrições em (3) e (4) estabelecem o sequenciamento das cirurgias em cada sala, de forma que, no caso das restrições em (3), se uma cirurgia  $i$  precede uma cirurgia  $j$  em uma mesma sala  $k$ , o tempo de início  $st_j$  da cirurgia  $j$ , deve ser posterior ao término da cirurgia  $i$ , mais o tempo necessário para preparar a sala para receber a cirurgia  $j$ . Assim, o lado esquerdo das inequações permite que a restrição  $st_j \geq st_i + rt_i + p_{ij}$  seja válida, somente quando as cirurgias  $i$  e  $j$  são executadas na mesma sala  $k$ , sendo que  $i$  precede  $j$  no sequenciamento. A mesma lógica é válida para o caso em que a cirurgia  $j$  precede a cirurgia  $i$ , que é tratado nas restrições em (4).

As restrições em (5) estabelecem uma espécie de corte permitindo que, por critério de qualidade, uma sala  $k$  receba maior quantidade de cirurgias que uma outra sala  $k - 1$ . As especificações em (6) definem o conjunto universo das variáveis de decisão como sendo binário.

## 2. Aspectos práticos relativos ao Problema.

No contexto do problema em estudo, as cirurgias eletivas são aquelas que possuem uma data e hora certa para ocorrer, já as cirurgias de emergência podem ocorrer aleatoriamente durante o dia, com isso o intuito de criar um modelo de agendamento das cirurgias eletivas como Job-shop Scheduling tem por finalidade de estabelecer critérios de otimização que tornem possível a disponibilidade de uma equipe e sala de cirurgia para tais situações. Desta forma, a situação real e a abordagem para a modelagem caracterizam-se pelos seguintes aspectos:

- Cada sala possui uma sequência de cirurgias, determinada pelos tipos de cirurgias que serão realizadas. Com isso pode-se considerar que cada cirurgia é uma tarefa e cada sala de cirurgia é uma máquina. Isso possibilita a utilização do Job-Shop Scheduling para localizar e aplicar melhorias no agendamento das cirurgias.
- De acordo com o modelo matemático da seção 4.1 (POACE), cada cirurgia será agendada uma única vez em uma sala, quando houver uma cirurgia posterior à que está sendo realizada, será necessário aguardar a finalização da cirurgia em realização para que se inicie a seguinte, tornando assim impossível o início de uma cirurgia durante o período de execução de outra.

## 3. Heurísticas Propostas para o Problema

Como já mencionado na subseção 1.5.1 a metaheurística GRASP, por ser um procedimento *multi-start* iterativo guloso-aleatório, realiza soluções aleatórias e depois compara o resultado entre buscas locais realizadas, com o propósito de atingir o ótimo global. A seguir serão apresentadas as heurísticas gulosa-aleatórias e de busca local que poderão ser utilizadas na metaheurística GRASP ou em qualquer outra metaheurística com estas características. O pseudocódigo apresentado na figura 4 denota a heurística construtiva proposta.

## 4. Heurísticas Construtiva:

A heurística construtiva proposta consiste em gerar de forma gulosa aleatória uma solução  $s$  com base na lista de cirurgias (LC). Inicialmente por não possuir uma solução, é atribuído o valor nulo à Solução, depois é avaliada a LC e criada uma lista restrita de cirurgias (LRC). Após feita a LRC, é gerada de forma gulosa aleatória uma solução baseada nesta lista. Obtendo-se à uma escolha aleatória para o elemento  $e$  com base no parâmetro de entrada  $\alpha$ , o elemento  $e$  (uma cirurgia da lista LRC) é atribuído à solução em construção  $s$ .

```

Procedimento Construção Gulosa Aleatória(OrdemAleatória)
  Solução ← ∅
  Avaliar candidatos
  Enquanto Solução ≠ Solução Ótima faça
  Construa Lista Restrita de Candidatos (LRC)
  S elecione um elemento s da LRC aleatoriamente
  Solução ← s
  Reavalie os candidatos
  Fim enquanto
  Retorne Solução
  Fim Construção Gulosa Aleatória.

```

Figura 4. Pseudocódigo para heurística construtiva gulosa aleatória para o POACE.

### 5. Heurísticas de busca local 2-opt:

A heurística de busca local proposta consiste em trocar duas cirurgias entre opções possíveis (viáveis) com o propósito de melhorar, ou seja, diminuir o tempo tal do agendamento das cirurgias na solução corrente.

Neste contexto, o algoritmo atribui o valor verdadeiro para *Melhora*, indicando que enquanto houver uma melhora no agendamento das cirurgias é possível otimizá-lo. Após ler a Solução, é feita a busca pelas cirurgias de maior prioridade e realizada a inversão *2-opt*, ou seja, a troca de posição entre duas cirurgias até encontrar um resultado melhor.

```

Procedimento Busca_Local(Solução)
  Melhora ← Verdadeiro
  Enquanto (Melhora = Verdadeiro) faça
    Leia Solução
    Busque cirurgia com maior prioridade
    Selecione vizinhança da solução usando 2 – opt(novaOrdem)
    TempoVizinho ← novaOrdem
    Se (TempoVizinho < Solução) então
      Solução ← TempoVizinho
      Melhora ← Falso
    Senão então
      Selecione outro vizinho
      Melhora ← Verdadeiro
    Fim se
  Fim se
Fim enquanto
Retorne Solução
Fim Busca Local

```

Figura 5. Pseudocódigo para heurística de busca local para o POACE.

Conseguindo um resultado melhor (ótimo local), é atribuído à *Melhora* o valor falso e armazenado em *Solução* a nova sequência.

## 6. Construção de Instâncias com os dados reais

Para realizar o teste com os dados cedidos, pelo Centro de Oncologia e Hematologia de Mossoró, foram usados os tempos de início e fim de cada cirurgia para chegar à uma média de tempo de duração, em seguida foi analisado qual a cirurgia tinha maior prioridade, após identificar o tempo médio das cirurgias, em horas, e criar a lista de prioridades, é criada a lista de cirurgias LC para a execução da heurística construtiva

A maior dificuldade na construção das instâncias foi estabelecer um critério para gerar aleatoriamente os valores das cirurgias de emergência, tendo em vista que não é possível prever quando e qual o tipo de cirurgia de emergência irá ocorrer. A escolha do critério influenciou diretamente no desenvolvimento das heurísticas. A dificuldade mencionada diz respeito a estabelecer um critério que obtenha o tempo total das cirurgias

que ofereça suporte às emergências igual ou melhor ao método convencional, onde não há sala e equipe para emergências.

Com os dados obtidos no Centro de Oncologia e Hematologia de Mossoró (COHM) foram criadas instâncias de teste dividindo as cirurgias do dia 01/04/2015. A figura 6 apresenta três exemplos de instâncias no formato utilizado nas heurísticas. A primeira (de cima para baixo) denota um exemplo de agendamento realizado pelo COHM. Na sequência, é apresentado um exemplo no formato utilizado na heurística gulosa aleatória onde foram selecionados elementos na lista de cirurgias (LC). Mais abaixo, é apresentado um exemplo de como ocorre na fase de busca local, onde são trocadas as posições de dois elementos (cirurgias) e comparado o resultado, caso esse resultado seja melhor que o anterior ele assume a posição como o melhor resultado.

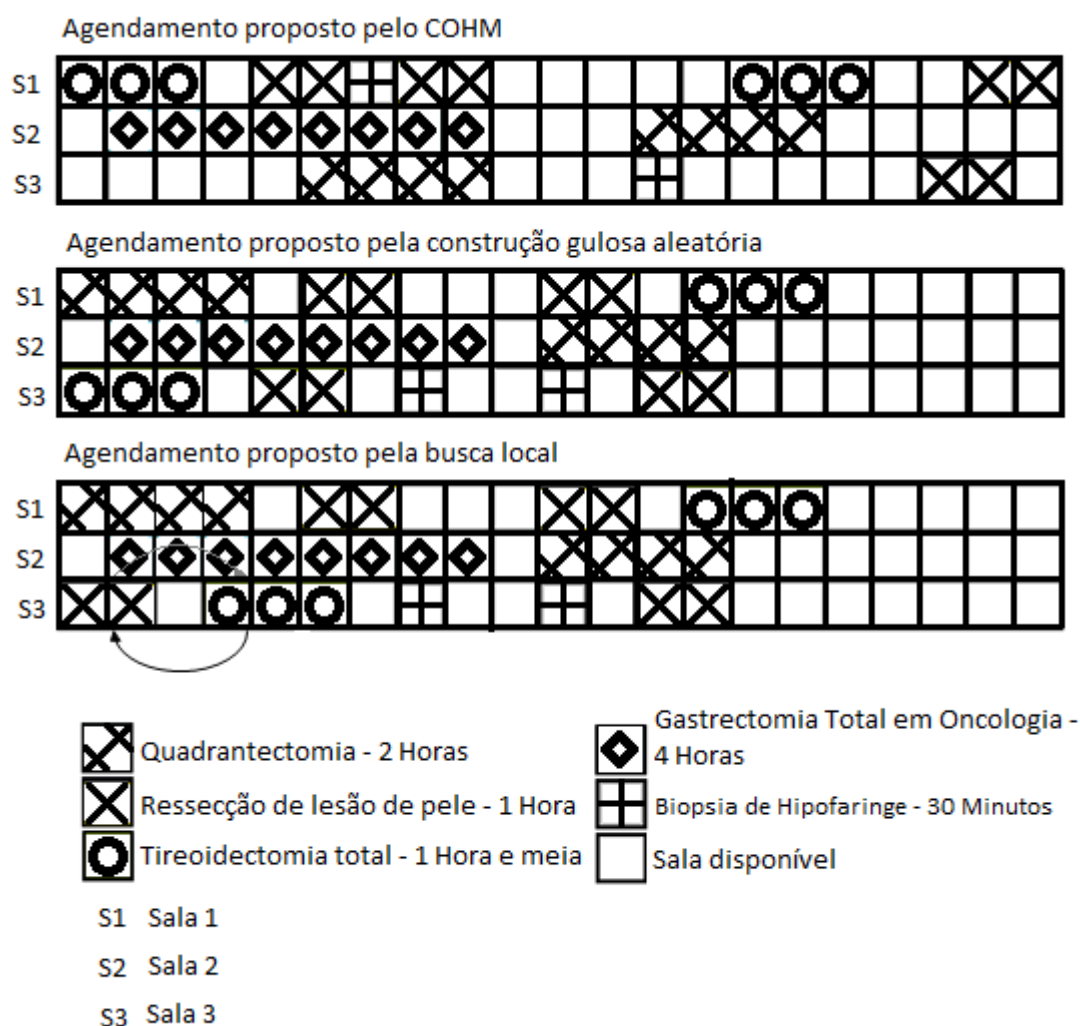


Figura 6. Esboço da construção gulosa aleatória e instante de tempo da busca local

## V. RESULTADOS

Para modelar o POACE como um caso particular de *Job Shop Scheduling* se fez necessário considerar as cirurgias como tarefas e as salas como máquinas, foram levadas em consideração a questão de que cada cirurgia é ininterrupta e que outras cirurgias só podem ser realizadas com o término da anterior. Neste contexto foram propostas duas heurísticas, uma construtiva e uma de busca local para o POACE.

As heurísticas implementadas foram aplicadas ao POACE com o objetivo de gerar uma lista de cirurgias onde houvesse a possibilidade de atender as cirurgias de emergência sem perder a qualidade no atendimento às cirurgias eletivas.

Foram realizados teste preliminares (validação) com as heurísticas e nestes testes foram considerados o tempo de duração de cada cirurgia e o seu nível de prioridade destas. Os resultados obtidos possibilitaram a ordenação das cirurgias a fim de serem atendidas as cirurgias de maior prioridade e as de menor tempo em salas diferentes.

Como resultado deste trabalho forma também geradas algumas instâncias de teste para os algoritmos implementados com dados reais obtidos no Centro de Oncologia e Hematologia de Mossoró. Todos os dados utilizados na construção das instâncias forma trabalhados de forma “blindada”, ou seja, não se teve acesso ao nome ou qualquer outro dado que possibilitasse a identificação de médicos ou pacientes.

## VI. CONCLUSÕES E TRABALHOS FUTUROS

As contribuições do presente trabalho possibilitarão o aprofundamento em trabalhos futuros para o tema em estudo.

As heurísticas aqui implementadas poderão ser utilizadas na metaheurística GRASP ou em qualquer outra que utilize heurísticas gulosa aleatória e/ou busca local.

Com base nos resultados obtidos nos testes preliminares (validação), percebe-se que é possível manter a quantidade de cirurgias e ter a ocupação por curto prazo de uma das salas, deixando-a ocupada por um curto período de tempo, reduzindo o tempo de espera para cirurgias de emergência.

Limitações e adaptações no cronograma deste trabalho impossibilitaram a implementação das heurísticas propostas no contexto da metaheurística GRASP.

Como trabalhos futuros podem ser apontados as seguintes possibilidades:

- Implementar a metaheurística GRASP aplicada ao POACE, utilizando as heurísticas gulosa aleatória e busca local aqui propostas.

- Testar as já citadas heurísticas na implementação de outras metaheurísticas para comparar os resultados e determinar o melhor método para otimizar o POACE.
- Aplicar os métodos propostos acima em estudos de caso em tempo real em hospitais.

## REFERÊNCIAS

- ADAMS, J., Balas, E., and Zawack, D. (1988), "The shifting bottleneck procedure for job shop scheduling", *Management Science* 34, 391-401.
- AIEX, R. M., Binato, S., and Resende, M. G. C. (2003). Parallel grasp with path-relinking for job shop scheduling. *Parallel Computing*, 29:393–430.
- A.S. Jain, S. Meeran, Deterministic job-shop scheduling: past, present and future, *Eur. J. Oper. Res.* 113 (1999) 390–434.
- BACKER, Bruno De, Philip Kilby, Patrick Prosser & Paul Shaw (2000), "Solving vehicle routing problems using constraint programming and metaheuristics", *Journal of Heuristics* 6, 501–523.
- BARROS, C. A. (2001), Uma aplicação de GRASP na otimização do emprego da unidade móvel de pistoneio, Dissertação de mestrado, Natal-RN, Brasil.
- BOWMAN, E.H. (1959), "The scheduling sequencing problem", *Operations Research* 7, 621.
- BLAZEWICZ, Jacek, Piotr Formanowicz, Marta Kasprzak, Wojciech T. Markiewicz & Aleksandra Swiercz (2004), "Tabu search method for DNA sequencing by hybridization with isothermic libraries", *Computational biology and chemistry* 28 (1), 11–19.
- CARTER, M.W. and Tovey, C. A. (1992). When is the classroom assignment problem hard? *Operations Research*, 40(1):28–30.
- CROES, G. A. (1958), A method for solving traveling salesman problems, em "Operations Research", Vol. 6, pp. 791–812.
- DORNDORF, U., and Pesch, E. (1995), "Evolution based learning in a job shop scheduling environment". *Computers & Operations Research* 22, 25-40.
- FAROE, O., D. Pisinger & M. Zachariasen (2003), "Guided local search for the threedimensional bin packing problem", 15, 267–283.
- LIMA JÚNIOR, F. C. (2009) Algoritmo Q-learning como Estratégia de Exploração e/ou Exploração para as Metaheurísticas GRASP e Algoritmo Genético. Tese (Doutorado) - Universidade Federal do Rio Grande do Norte – UFRN, Natal- RN.
- MARINE, A.S. (1960), "On the job shop scheduling problem", *Operations Research* 8, 219-223.
- MILLER, David M., Hui-Chuan Chen, Jessica Matson & Qiang Liu (1999), "A hybrid genetic algorithm for the single machine scheduling problem", *Journal of Heuristics* 5(4), 437–454.

- PROENÇA, I. M. (2010). Planejamento de Cirurgias Eletivas - Abordagens em Programação Inteira. PhD thesis, Tese de Doutorado, Departamento de Estatística e Investigação Operacional, Lisboa.
- RAVETTI, Martín G. Mateus G. R. (2003), “Sequenciamento de Tarefas Com Máquinas Paralelas, Permitindo Atrasos e com Tempos de Preparação de Máquina Dependentes da Sequência.”, XXXV SBPO, Natal/RN.
- TEIXEIRA, P. P., Chang J. J. (2012), “MODELAGEM: SISTEMA DE AGENDAMENTO DE CIRURGIAS ELETIVAS EM UM HOSPITAL”.
- WAGNER, H.M. (1959), "An integer linear programming model for machine scheduling", Naval Research Logistics Quarterly 6,
- WANG, Xiao-Dong & Tom Chen (1995), “On performance and area optimization of VLSI systems using genetic algorithms”, VLSI Design 3(1), 43–51.