

UNIVERSIDADE FEDERAL DE MINAS GERAIS

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO / ESCOLA DE ENGENHARIA

Modelagem do Sequenciamento no Processo de Lingotamento Contínuo por Métodos Exatos e Heurísticos

Allan Messeder Caldas Bretas

Orientador: Prof. Dr. Martín Gómez Ravetti

Belo Horizonte

Julho, 2013

UNIVERSIDADE FEDERAL DE MINAS GERAIS

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO / ESCOLA DE ENGENHARIA

Modelagem do Sequenciamento no Processo de Lingotamento Contínuo por Métodos Exatos e Heurísticos

Allan Messeder Caldas Bretas

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de Minas Gerais para a obtenção do título de Mestre em Engenharia de Produção.

Área de Concentração: Produção e Logística

Linha de Pesquisa: Otimização de Sistemas Produtivos

Orientador: Prof. Dr. Martín Gómez Ravetti

Belo Horizonte

Julho, 2013

*“Dedico este trabalho ao meu pai,
maior incentivador da minha vida
acadêmica e exemplo de humildade
e perseverança.”*

Resumo

O presente trabalho aborda o problema de sequenciamento da produção no setor de Lingotamento Contínuo de uma siderúrgica brasileira. A função objetivo do problema exalta o *trade-off* entre o custo de mistura e o custo de *setup*. Uma das duas opções deve ser escolhida entre quaisquer dois *jobs* que estejam em posições subsequentes no sequenciamento e as afinidades entre os tipos de aço determinam o custo de mistura. São desenvolvidos dois modelos matemáticos para resolver o problema, que apresentam bons resultados em instâncias de pequeno porte. É desenvolvido um algoritmo que utiliza a meta-heurística ILS como estratégia para a resolução de instâncias de grande porte. Instâncias reais foram fornecidas e permitem a comparação dos métodos com a solução apresentada pela empresa.

Palavras-Chave: Sequenciamento, Lingotamento Contínuo, Programação Linear Inteira Mista, Métodos Heurísticos.

Abstract

This dissertation approaches the production scheduling problem in the Continuous Casting sector of a Brazilian steel company. The objective function of the problem works the trade-off between the mix cost and the setup cost. One of these two options must be chosen between any two jobs in subsequent positions on sequencing, and the affinities between the steel types determine the mixing cost. Two mathematical models are developed to solve the problem, which present good results on small instances. Furthermore, an algorithm is developed using Iterated Local Search (ILS) metaheuristic as a strategy for solving large instances. Actual instances are provided and allow the comparison of the methods to the solution presented by the company.

Keywords: Scheduling, Continuous Casting, Mixed Integer Linear Programming, Heuristic Methods.

Agradecimentos

Gostaria de agradecer, primeiramente, à Deus pela força e superação.

À UFMG pela oportunidade.

Aos professores Martín e Maurício pelos ensinamentos e dedicação.

À Usiminas pelo incentivo à minha formação.

À minha família pelo apoio, incentivo e paciência.

À minha namorada Letícia, pelo incentivo, companheirismo e compreensão.

Aos meus amigos, colegas e professores do DEP que sem dúvida contribuíram para a realização deste trabalho.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xi
Lista de Siglas	xiii
1 Introdução	1
1.1 Objetivos	2
1.2 Justificativa	2
1.3 Contexto Siderúrgico Nacional	3
1.4 Contexto do Problema	4
1.4.1 O Lingotamento Contínuo	5
1.4.2 Delimitação do Problema	7
2 Revisão Bibliográfica	9
2.1 O Sequenciamento da Produção	10
2.1.1 Sequenciamento em Máquinas Paralelas	12
2.1.2 Sequenciamento em Aciaria	18
2.1.3 Sequenciamento em Máquina Simples	24
2.2 Definição do Problema	27
3 Modelos Matemáticos	29
3.1 Modelo 1: Variáveis de data de conclusão	30

3.2	Modelo 2: Variáveis indexadas pela posição	33
3.3	Instâncias Reais	35
3.4	Resultados Computacionais dos Modelos Matemáticos	36
4	Heurísticas e Espaço de Busca	42
4.1	Análise de <i>Landscape</i>	42
4.1.1	Representação e Estruturas de Vizinhança	43
4.1.2	Análise Comparativa das Vizinhanças	48
4.1.2.1	Número de Mínimos Locais	49
4.1.2.2	Distância Entre Mínimos Locais	51
4.1.2.3	Análise de Robustez	55
4.1.3	Análise de Simetria	56
4.1.4	Características do Espaço de Busca	58
4.2	Métodos Heurísticos	60
4.2.1	Heurísticas Construtivas	61
4.2.1.1	Garantia de Qualidade para as Heurísticas Construtivas	65
4.2.2	Buscas Locais	68
4.2.3	<i>Iterated Local Search</i>	70
4.2.3.1	Calibração dos parâmetros para o método ILS	72
4.3	Geração de Instâncias Artificiais	74
4.4	Resultados Computacionais dos Métodos Heurísticos	75
4.4.1	Resultados dos Testes das Heurísticas Construtivas	75
4.4.2	Resultados dos Testes da Meta-heurística ILS	80
5	Conclusão	85
5.1	Contribuições Acadêmicas	86
5.2	Trabalhos Futuros	87
	Referências Bibliográficas	88

Lista de Figuras

1.1	Produção de Aço Bruto no Brasil.	3
1.2	Macro-fluxo de processo de uma usina siderúrgica integrada	4
1.3	Macro-fluxo de processo de uma usina siderúrgica integrada	5
1.4	Tipos de Processos Utilizados.	6
1.5	Fluxo do aço na máquina de lingotamento.	7
2.1	Principais Classificações para Problemas de Sequenciamento	11
2.2	Sequenciamento em Máquinas Paralelas	12
2.3	Sequenciamento em Máquinas Paralelas	18
2.4	Elementos do Sequenciamento em Aciaria	20
2.5	Sequenciamento em Máquina Simples	24
3.1	Gráficos de Gantt para instância IRM42	40
3.2	Gráficos de Gantt para instância IRM42	40
4.1	Algoritmo de busca local na vizinhança A	44
4.2	Algoritmo de busca local na vizinhança B	45
4.3	Algoritmo de busca local na vizinhança C	46
4.4	Distâncias dos mínimos locais - viz A	53
4.5	Distâncias dos mínimos locais - viz B	54
4.6	Distâncias dos mínimos locais - viz C	54
4.7	Simetria devido à combinação de séries	57

4.8	Simetria devido à existência de <i>jobs</i> iguais	58
4.9	Distâncias relativas dos mínimos locais das três vizinhanças	59
4.10	Algoritmo Heurística Construtiva	62
4.11	Heurística Construtiva 1	64
4.12	Comparação entre as vizinhanças	69
4.13	Algoritmo <i>ILS</i>	71
4.14	Comparação do DPR para os valores testados dos parâmetros K e N.	73
4.15	Comparação do DPR para os valores testados do parâmetro NP.	73
4.16	Calibração parâmetro NIT	74
4.17	Heurística Construtiva 3 comparada à solução da empresa	78
4.18	Heurística Construtiva 1 comparada à solução dos modelos	79
4.19	Heurística Construtiva 4 comparada à solução da empresa	79
4.20	Heurística Construtiva 4 comparada à solução dos modelos	80
4.21	Gráfico de Gantt IRM20 - Comparação entre ILS e Modelo 1	83
4.22	Gráfico de Gantt IRM38 - Comparação entre ILS e Modelo 1	84
5.1	Apêndice C - Análise de Resíduos Parâmetros ILS	107

Lista de Tabelas

3.1	Solução Empresa x Solução Modelo 1 - IRM01 a IRM12	38
3.2	Solução Empresa x Solução Modelo 2 - IRM01 a IRM12	38
3.3	Solução Empresa x Solução Modelo 1 - IRM37 a IRM48	39
3.4	Solução Empresa x Solução Modelo 2 - IRM37 a IRM48	39
3.5	Solução Modelo 1 x Solução Modelo 2	39
4.1	Descrição instância IRM05	47
4.2	Descrição instância IRM08	47
4.3	Descrição instância IRM43	48
4.4	Descrição instância IRM39	48
4.5	Erro relativo por tipo de instância.	49
4.6	Comparação número de mínimos locais por vizinhança	50
4.7	Análise comparativa do desempenho das vizinhanças.	51
4.8	Distância média entre mínimos locais	52
4.9	Análise comparativa GMD	55
4.10	Resumo Heurísticas Construtivas	65
4.11	Análise de dominância entre as vizinhanças A, B e C.	69
4.12	Valores testados para cada parâmetro nos testes de calibração.	72
4.13	Geração instâncias artificiais	75
4.14	Desempenho das heurísticas construtivas - 50 <i>jobs</i>	76
4.15	Desempenho das heurísticas construtivas - 100 <i>jobs</i>	76

4.16	HC 1 e HC 2 x Modelo 1 - IRM01 a IRM12	77
4.17	HC 3 e HC 4 x Modelo 1 - IRM01 a IRM12	78
4.18	Comparação Modelo 1 x ILS - IRM13 a IRM24	81
4.19	Comparação Modelo 1 x ILS - IRM37 a IRM48	82
5.1	Apêndice A - Sol. Empresa x Sol. Modelo 1 - IRM01 a IRM12	93
5.2	Apêndice A - Sol. Empresa x Sol. Modelo 2 - IRM01 a IRM12	94
5.3	Apêndice A - Sol. Empresa x Sol. Modelo 1 - IRM13 a IRM24	94
5.4	Apêndice A - Sol. Empresa x Sol. Modelo 2 - IRM13 a IRM24	95
5.5	Apêndice A - Sol. Empresa x Sol. Modelo 1 - IRM25 a IRM36	95
5.6	Apêndice A - Sol. Empresa x Sol. Modelo 2 - IRM25 a IRM36	96
5.7	Apêndice A - Sol. Empresa x Sol. Modelo 1 - IRM37 a IRM48	96
5.8	Apêndice A - Sol. Empresa x Sol. Modelo 1 - IRM37 a IRM48	97
5.9	Apêndice B - HC 1 e HC 2 x Modelo 1 - IRM01 a IRM12	98
5.10	Apêndice B - HC 1 e HC 2 x Modelo 1 - IRM13 a IRM24	99
5.11	Apêndice B - HC 3 e HC 4 x Modelo 1 - IRM01 a IRM12	99
5.12	Apêndice B - HC 3 e HC 4 x Modelo 1 - IRM13 a IRM24	100
5.13	Apêndice B - Heurísticas Construtivas + BL C - 25 <i>jobs</i>	100
5.14	Apêndice B - Heurísticas Construtivas + BL C - 50 <i>jobs</i>	101
5.15	Apêndice B - Heurísticas Construtivas + BL C - 75 <i>jobs</i>	101
5.16	Apêndice B - Comparação Modelo 1 x ILS - IRM01 a IRM12	102
5.17	Apêndice B - Comparação Modelo 1 x ILS - IRM13 a IRM24	103
5.18	Apêndice B - Comparação Modelo 1 x ILS - IRM25 a IRM36	104
5.19	Apêndice B - Comparação Modelo 1 x ILS - IRM37 a IRM48	105
5.20	Apêndice C - Análise de Variância Parâmetros ILS	106

Lista de Siglas

AG: Algoritmos Genéticos.

BT: Busca Tabu.

DPR: Desvio Percentual Relativo.

EDD: *Earliest Due Date* – data de entrega mais cedo.

GMD: Gradiente Médio de Descida.

GRASP: *Greedy Randomized Adaptive Search Procedure* – Procedimento de busca adaptativa gulosa e randomizada.

HC: Heurística Construtiva

ILS: *Iterated Local Search* – Busca Local Iterativa.

LC: Lingotamento Contínuo

MLC: Máquina de Lingotamento Contínuo.

PLIM: Programação Linear Inteira Mista.

PCP: Planejamento e Controle da Produção.

PMP: Plano Mestre de Produção.

PSMP: Problema de Seqüenciamento em Máquinas Paralelas

PSUMAA: Problema de Seqüenciamento de uma Máquina por Antecipação e Atraso da Produção.

SA: *Simulated Annealing* - Recozimento simulado.

Capítulo 1

Introdução

O presente trabalho estuda o sequenciamento da produção no setor de Lingotamento Contínuo de uma siderúrgica. Tal problema envolve diversos elementos de alta relevância como custos de parada de máquina e custos de produção dependentes da sequência. Esse tipo de sequenciamento não foi largamente estudado, de modo que poucos trabalhos foram desenvolvidos até o momento com abordagens semelhantes à que foi realizada nesta pesquisa. Por ser aplicado a um caso real de uma siderúrgica brasileira, este trabalho visa fornecer contribuição aos meios acadêmico e industrial e possibilitar novos avanços nesse tipo de problema.

O problema do sequenciamento da produção de aço por Lingotamento Contínuo ou *Steelmaking-continuous Casting Problem* já foi abordado por alguns autores, mas não há uma uniformidade entre os trabalhos, variando abordagem, objetivo e restrições do problema. Neste trabalho ele é tratado primeiramente como um sequenciamento em máquinas paralelas no qual os custos envolvidos na realização ou não da parada de máquina têm especial relevância.

Este estudo é fruto de uma pesquisa em andamento na Universidade Federal de Minas Gerais em que estão diretamente envolvidos o autor do trabalho, o aluno de doutorado Aloísio C. Gomes Jr e os Prof. Martín G. Ravetti e Mauricio C. de Souza.

A estrutura do trabalho está definida da forma descrita a seguir. No capítulo 1 é introduzido o tema da pesquisa e são apresentados o contexto do problema, o contexto siderúrgico atual e uma delimitação do problema de sequenciamento. No capítulo 2 é feita a revisão bibliográfica, que se divide em problemas de sequenciamento em máquinas paralelas, problemas de sequenciamento em aciarias e problemas de sequenciamento em máquina simples. No capítulo 3 são apresentados os modelos desenvolvidos

para resolver o problema de sequenciamento, as instâncias fornecidas pela empresa e os resultados dos modelos para essas instâncias. No capítulo 4 é realizada uma análise do espaço de busca do problema, são apresentados os métodos heurísticos desenvolvidos, assim como as instâncias artificiais geradas e por fim são apresentados os resultados para os testes com o método heurístico. O capítulo 5 apresenta algumas conclusões, as contribuições acadêmicas da pesquisa e os próximos passos, respectivamente. Por fim são apresentadas as referências bibliográficas e nos apêndices A e B são exibidos resultados de alguns testes mencionados ao longo do texto.

1.1 Objetivos

O objetivo principal da pesquisa é a formulação e implementação de métodos capazes de resolver o problema de sequenciamento no Lingotamento Contínuo considerando as principais restrições e características do problema de sequenciamento e do processo produtivo no caso real.

Para alcançar esse objetivo, objetivos secundários são necessários como a realização de um estudo bibliográfico amplo, que seja capaz de apontar os principais trabalhos já desenvolvidos sobre o assunto e as tendências para o mesmo. Além disso é necessário identificar modelos matemáticos aplicáveis ao problema e formular modelos próprios considerando as características do sistema. Assim como testar métodos heurísticos que possam apresentar desempenho satisfatório para instâncias de grande porte.

1.2 Justificativa

O presente trabalho é justificado principalmente pela importância que a redução de custos possui hoje no setor siderúrgico, principalmente no brasileiro. O setor vive atualmente uma crise histórica, que coloca como estratégia número um a redução de custos, principalmente relacionados ao processo de produção. A aplicação de um modelo nos moldes do estudado pode render uma economia que no processo atual é impossível de ser realizada devido à falta de uma ferramenta especializada.

1.3 Contexto Siderúrgico Nacional

A indústria siderúrgica brasileira possui longo histórico de atividade, posicionando o país em lugar de destaque entre os produtores de aço em várias décadas. A última, em especial, foi marcada por grandes investimentos em modernização e ampliação das plantas. Com a exportação em grandes níveis e o mercado interno consumindo cada vez maiores quantidades, a indústria do aço teve momentos muito favoráveis que possibilitaram o crescimento das empresas aqui instaladas. Esse bom momento encontrava-se em seu auge quando surgiu a crise econômica de 2008, naquela época as siderúrgicas batiam recordes de produção e viam a demanda do mercado interno aumentar em grandes escalas.

A crise representou forte queda de produção e grande capacidade ociosa das plantas. Desde então, é observado um elevado desbalanceamento entre a capacidade mundial de produção e o consumo de produtos siderúrgicos, fazendo com que os preços permaneçam em patamares baixos e a concorrência elevada. O momento atual é de investir na modernização das plantas para se mostrar um forte concorrente, mas com cautela para não gastar aonde o retorno pode não aparecer. As diretrizes são reduzir custos e aumentar a produtividade. Nos últimos meses a produção de aço permaneceu estável, o que não projeta um cenário muito otimista para os próximos meses.

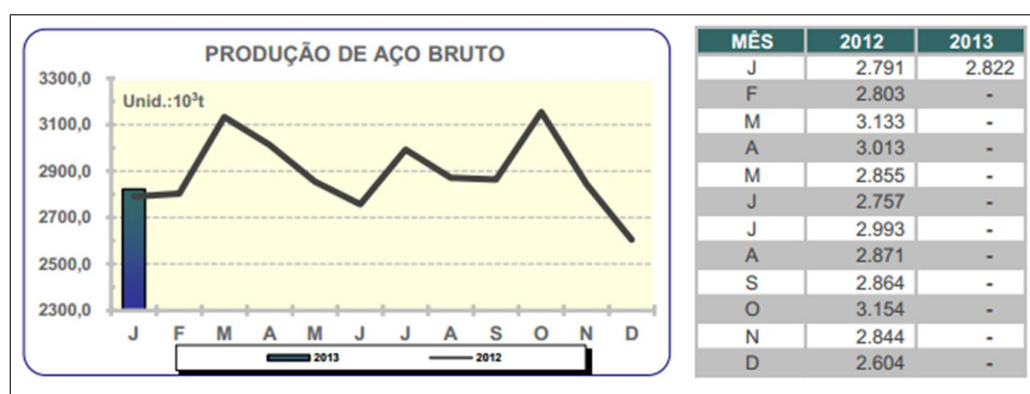


Figura 1.1: Fonte: Instituto Aço Brasil

Apesar da situação delicada que vive o setor, o Brasil é hoje o quinto maior exportador de aço líquido e exporta para mais de cem países. Seu parque produtor de aço possui 29 usinas, administradas por 11 grupos empresariais diferentes. Segundo o Instituto Aço Brasil, a capacidade instalada do país em 2011 era cerca de 47,8 milhões de toneladas por ano e a produção anual tem alcançado níveis de 35,2 milhões de toneladas. A indústria siderúrgica brasileira emprega 137.134 colaboradores e os

principais setores consumidores de aço são a construção civil, o setor automotivo, bens de capital, máquinas e equipamentos (incluindo agrícolas) e utilidades domésticas e comerciais.

1.4 Contexto do Problema

O trabalho apresentado foi desenvolvido no setor de Lingotamento Contínuo de uma usina siderúrgica integrada. Esse tipo de usina é caracterizado por ter setores destinados à produção de toda a cadeia de produção do aço, Redução, Aciaria e Laminação. A Figura 1.2 ilustra o fluxo de uma usina integrada.

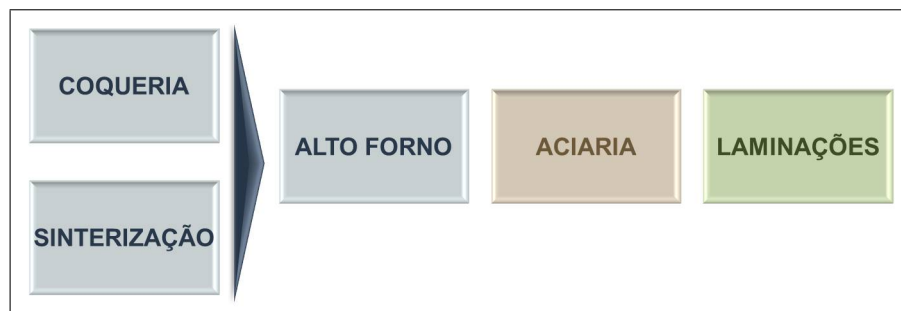


Figura 1.2: Macro-fluxo de processo de uma usina siderúrgica integrada

A Coqueria é a área da redução onde ocorre a produção do coque, combustível composto em sua maioria por carvão. Já a Sinterização é onde é produzido o sinter, rico em minério de ferro. A terceira área da redução é o Alto-Forno, nele o sinter é fundido através da queima do coque e o produto resultante é o ferro-gusa, que é líquido e possui grande quantidade de ferro.

Os processos subsequentes estão na Aciaria, onde o ferro-gusa será primeiramente refinado através da retirada de elementos químicos como enxofre, fósforo e carbono e da adição de outros requisitados à especificação de qualidade do aço. Após passar pelo equipamento Convertedor LD, no refino primário, o ferro-gusa já pode ser considerado aço devido à quantidade de ferro na mistura, que passa a ser mais elevada. Os equipamentos de refino secundário, como Forno Panela e Desgaseificador a Vácuo têm a função de ajustar a composição química e a temperatura do aço para esse passar à fase seguinte, na qual ele será solidificado em forma de placas no processo conhecido como lingotamento contínuo (LC). Nesse processo, o aço líquido contido em painéis é vazado através de uma válvula para um equipamento chamado distribuidor que por sua vez promove o escoamento do aço para os veios de solidificação. O processo é

contínuo, de modo que o aço de uma panela a ser vazada terá contato com o aço da panela anterior e da panela seguinte. O intervalo entre panelas é monitorado para que não falte aço na máquina de lingotamento contínuo. A Figura 1.3 ilustra o processo da aciaria.

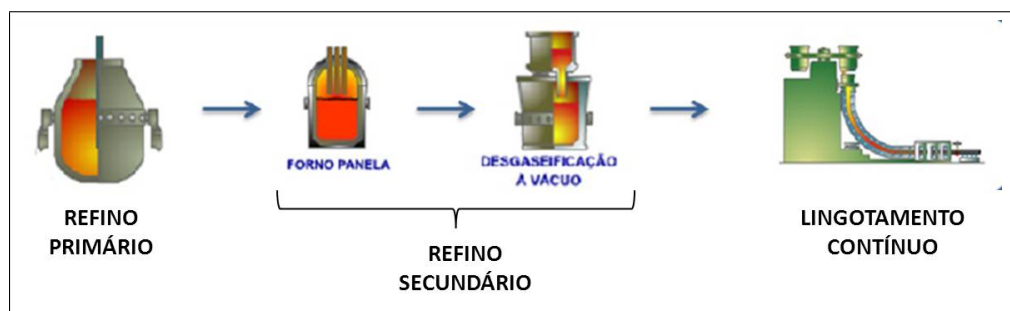


Figura 1.3: Macro-fluxo de processo de uma usina siderúrgica integrada

As fases seguintes são a Laminação à Frio e a Laminação à Quente onde as placas são laminadas para adquirir as características dos produtos finais, que podem ser chapas ou bobinas, variando composições químicas e propriedades mecânicas de acordo com a aplicação do aço.

Como o foco deste estudo está na programação da produção do lingotamento contínuo, o próximo item irá tratar com mais detalhes tal processo. A revisão bibliográfica será conduzida primeiramente de forma a apresentar trabalhos em contextos semelhantes e aplicáveis ao caso estudado. Posteriormente serão analisados trabalhos no mesmo contexto, que também possuem como desafio o sequenciamento no LC.

1.4.1 O Lingotamento Contínuo

O processo de Lingotamento Contínuo da forma como acontece atualmente surgiu na década de 1950, quando revolucionou o processo de solidificação do aço. Até então, o processo dominante era o chamado Lingotamento convencional, que consistia no derramamento do aço líquido em moldes para a solidificação e formação do produto desejado. Com a chegada do lingotamento contínuo, foi possível alcançar maior rendimento, produtividade, qualidade e eficiência de custos, devido aos benefícios da produção contínua. Passou a ser possível gerar um produto mais padronizado e de custo mais baixo, além de maior controle do processo através da automação. Conforme apresentado na Figura 1.4, o Lingotamento Contínuo atualmente é muito mais utilizado do que o convencional.

PRODUTOS	JANEIRO		13/12 (%)	OUT 2012	NOV 2012	DEZEMBRO		12/11 (%)	Unid.: 10 ³ t ÚLTIMOS 12 MESES
	2013(*)	2012				2012(*)	2011		
- ACIARIA									
OXIGÊNIO(LD)/EOF	2.137,3	2.136,2	0,1	2.418,6	2.147,0	2.253,3	2.163,3	4,2	26.568,5
ELÉTRICO	684,3	654,8	4,5	735,5	696,8	351,1	511,0	(31,3)	8.144,0
TOTAL	2.821,6	2.791,0	1,1	3.154,1	2.843,8	2.604,4	2.674,3	(2,6)	34.712,5
- LINGOTAMENTO									
CONVENCIONAL	59,5	62,1	(4,2)	85,4	63,2	44,7	81,1	(44,9)	927,0
CONTÍNUO	2.760,5	2.728,7	1,2	3.067,2	2.778,4	2.557,5	2.590,5	(1,3)	33.765,8
- AÇO P/FUNDIÇÃO	1,6	0,2	700,0	1,5	2,2	2,2	2,7	(18,5)	19,7

Figura 1.4: Fonte: Instituto Aço Brasil

A Figura 1.5 ilustra o processo e a seguir será apresentada uma explicação resumida do mesmo. Estando a máquina preparada para iniciar o processo, a panela contendo aço líquido (1) chega à máquina de lingotamento contínuo (MLC) através da torre giratória, que possui dois berços para receber panelas. Um desses berços fica acima do distribuidor (2) e fica com a panela que estiver vazando aço. Enquanto isso, o outro berço fica do lado oposto da torre e está pronto para receber outra panela, o que deve acontecer poucos minutos antes da primeira panela terminar de ser vazada, pois caso contrário a temperatura do aço da segunda panela iria diminuir muito enquanto ela espera até que a outra se esvazie. Quando a primeira panela esvazia, ela é tampada e então a torre gira, trazendo a segunda panela para iniciar o vazamento. Esse processo não pode demorar, pois isso comprometeria a continuidade do lingotamento. Quando a torre termina de girar, a primeira panela é então retirada pela ponte rolante, liberando espaço para mais tarde receber a terceira panela. O distribuidor é um recipiente que recebe o aço da panela e tem a função de alimentar os moldes (3) de forma suave e constante. Os moldes trabalham realizando um movimento oscilatório, que visa impedir a aderência do aço às suas paredes. Os moldes são refrigerados a água, já iniciando o processo de resfriamento do aço. À medida que o aço vai descendo no molde, ele vai sendo resfriado e começa a entrar em contato com os rolos (4), que fazem sua conformação em forma de placa. Após os rolos o aço passa pelo oxicorte, que corta o aço conferindo ao mesmo o comprimento da placa desejado.

A preparação da MLC envolve muitas pessoas e muito tempo. Esse *setup* geralmente envolve, dentre outras atividades, a troca do distribuidor, a preparação do molde e o posicionamento da barra falsa, que fica localizada dentro do molde e vai guiando o primeiro fluxo de aço vazado na máquina. A barra vai então descendo e depois é retirada. Além do *setup* ser demorado, ele apresenta um custo relevante.

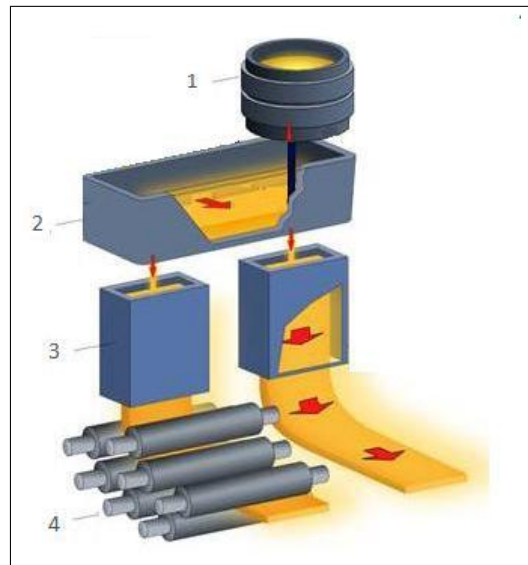


Figura 1.5: Fonte: Böhrer Welding, adaptado.

1.4.2 Delimitação do Problema

O problema de sequenciamento tratado neste trabalho está localizado na Aciaria. Em problemas como esse, as placas de aço normalmente são definidas como os *jobs* e a abordagem do problema varia, podendo ser um *Job Shop*, quando são considerados vários estágios de produção na Aciaria ou ainda um problema de sequenciamento em máquina simples ou em máquinas paralelas, quando é considerado como único estágio o Lingotamento Contínuo. No caso deste trabalho, o foco será no Lingotamento Contínuo.

O problema de sequenciamento consiste em determinar a ordem em que as placas serão processadas. O objetivo está relacionado à minimização dos custos. No processo em questão os principais são o custos de mistura, gerado quando *jobs* de grades de aço diferentes são processados em sequência, e de *setup*, operação que interrompe a produção e evita a geração de mistura. Outras características gerais do problema são:

- Uma série, ou sequência, é o nome dado a um conjunto de placas processado ininterruptamente;
- O setor de Lingotamento Contínuo estudado possui duas máquinas;
- A série não pode exceder um tamanho limite estipulado de acordo com a vida útil do distribuidor;
- A adjacência entre dois *jobs* em uma série envolve, além do custo de mistura, a

exigência de que as placas geradas tenham larguras similares;

No capítulo 2 serão discutidos os problemas de sequenciamento em máquinas paralelas e em máquinas simples, assim como diversas abordagens diferentes para o problema de sequenciamento em aciarias. Ao fim desse capítulo é feita uma definição precisa do problema estudado e das características consideradas.

Capítulo 2

Revisão Bibliográfica

O planejamento da produção de uma organização envolve decisões de diferentes naturezas e necessidades. Decisões de curto prazo como alocações de efetivo e sequenciamento de ordens de produção e decisões de longo prazo como aquisição de maquinário e ampliação de plantas fazem parte da rotina de um departamento de planejamento da produção. Segundo Contador [13], o planejamento e controle da produção (PCP) visa determinar as respostas para as questões de quanto produzir e quando produzir, considerando o dinamismo do mercado e os objetivos estratégicos da empresa. Para isso, o PCP deve gerenciar o fluxo de materiais e a alocação de pessoas e equipamentos, utilizando das capacidades interna e dos fornecedores para atender às necessidades dos clientes [8].

Conforme apresentado por Fleury & Wanke [18], o Planejamento da Produção pode ser subdividido hierarquicamente em estratégico, tático e operacional. O nível estratégico se concentra no planejamento da capacidade da organização, são atividades direcionadoras dos outros níveis, como identificação das necessidades dos clientes e determinação do nível de serviço, estipulando as especificações necessárias para satisfazer o cliente. Além disso, são planejadas questões estruturais como ampliação da capacidade produtiva e localização de instalações, além de políticas de atendimento e distribuição. Aqui se concentram as decisões de longo prazo, referentes ao futuro da organização nos próximos seis meses ou mais.

O nível tático é marcado pela definição dos processos e planejamento de operações, vendas e estoque. As decisões nesse nível possuem um horizonte de médio prazo e algumas de suas entradas foram determinadas pelo nível estratégico. As questões tratadas podem variar desde o arranjo físico do chão de fábrica e o regime de turnos

utilizado à política de estoques adotada. Outros processos dizem respeito ao planejamento de compra, à utilização das matérias-primas e à previsão de demanda. Todo esse conjunto de informações e decisões constitui o Planejamento Agregado da produção, que é definido por Moreira [39] como "o processo de balanceamento da produção com a demanda (...)".

No nível operacional a frequência de decisões já é bem maior, as mesmas se dão em curto ou curtíssimo prazo e dão sequência ao planejado nos níveis superiores. Função executada pelo Plano Mestre de Produção (PMP), que a partir das informações da configuração do processo produtivo geradas no Planejamento Agregado, define quais itens serão produzidos e em quais quantidades em um determinado período (semanas ou meses) de modo a atender à demanda e aos requisitos de qualidade. Decisões para menores períodos ainda como máquinas utilizadas e data de produção do item, que às vezes são tomadas em dias ou até horas se concentram, principalmente, no sequenciamento da produção, processo que é o foco deste estudo.

2.1 O Sequenciamento da Produção

A entrega dos lotes de produtos no prazo é hoje requisito mínimo para qualquer organização, além disso, uma programação bem feita pode significar maior capacidade de atendimento, menores custos e conseqüentemente maior competitividade. Com isso, diversos problemas relacionados à programação da produção e sequenciamento estão sendo tratados na literatura. A diferença entre esses estudos está na abordagem em questão e nas especificidades consideradas. Cada tipo de sistema pode ser tratado de diferentes formas, de acordo com os objetivos do trabalho e/ou das características do próprio sistema.

O sequenciamento da produção, ou *scheduling*, consiste em definir a ordem de execução das operações em um sistema de modo a alcançar um ou mais objetivos. No entanto, essa definição deve obedecer a algumas regras que podem ser mais simples ou mais complexas dependendo do sistema e dos objetivos em questão.

Os problemas de *scheduling* normalmente são classificados de acordo com o fluxo de operações dos *jobs* ou tarefas nas máquinas e também quanto ao número de estágios e máquinas em cada estágio. As principais dessas classificações são apresentadas na Figura 2.1.

Um mesmo sistema pode ser tratado de duas ou mais maneiras diferentes de-

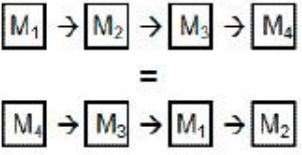









TIPO	DESCRIÇÃO	ILUSTRAÇÃO
<i>Job Shop</i>	Cada tarefa possui uma seqüência de máquinas a ser cumprida.	$J_1: \{M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow M_4\}$ $J_2: \{M_1 \rightarrow M_3 \rightarrow M_2 \rightarrow M_4\}$ $J_3: \{M_4 \rightarrow M_2 \rightarrow M_3 \rightarrow M_1\}$
<i>Flow Shop</i>	Todas as tarefas possuem a mesma seqüência de máquinas.	$J_1: \{M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow M_4\}$ $J_2: \{M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow M_4\}$ $J_3: \{M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow M_4\}$
<i>Open Shop</i>	Não existe seqüência preestabelecida para o processamento das tarefas, apenas um conjunto de tarefas a serem processadas. A ordem é irrelevante.	$J_1: \{M_1, M_2, M_3, M_4\}$ 
<i>Flow Shop Permutacional</i>	Além de cada tarefa ter a mesma seqüência, a seqüência de tarefas em cada máquina é a mesma.	$J_1 \rightarrow$    J_2  J_3   $J_4 \leftarrow$
Máquina Única	O sistema possui apenas uma máquina para o processamento de todas as tarefas.	$J_3 \ J_2 \ J_1 \rightarrow$ 
Máquinas Paralelas	O sistema possui mais de uma máquina para a mesma operação.	$J_3 \ J_2 \ J_1 \rightarrow$  

Figura 2.1: Principais Classificações para Problemas de Sequenciamento

pendendo do foco da pesquisa. A adequação ou não da classificação (*Job Shop*, *Flow Shop*, etc.) adotada ao sistema produtivo vai depender da abrangência, pois a modelagem deve conter os principais elementos e restrições para a tomada de decisão do sistema. Não existe uma maneira mais ou menos correta de se classificar o sistema. A classificação deve ser representativa e não excluir partes importantes do mesmo. No próximo item, será apresentada uma revisão bibliográfica sobre problemas de sequenciamento em máquinas paralelas, classificação que será adotada para modelar o sistema produtivo estudado.

2.1.1 Sequenciamento em Máquinas Paralelas

O problema de sequenciamento em máquinas paralelas (PSMP) pode ser separado em diferentes classificações de acordo com o tipo das máquinas, critérios de otimização, recursos e a abordagem em questão. Usualmente, nesse tipo de problema o processamento de uma tarefa por uma máquina é definido como *job*. Tal problema é caracterizado primeiramente pela existência de apenas um estágio de produção e mais de uma máquina nesse estágio. Todos os *jobs* devem passar por esse estágio e serem processados em uma das máquinas disponíveis.

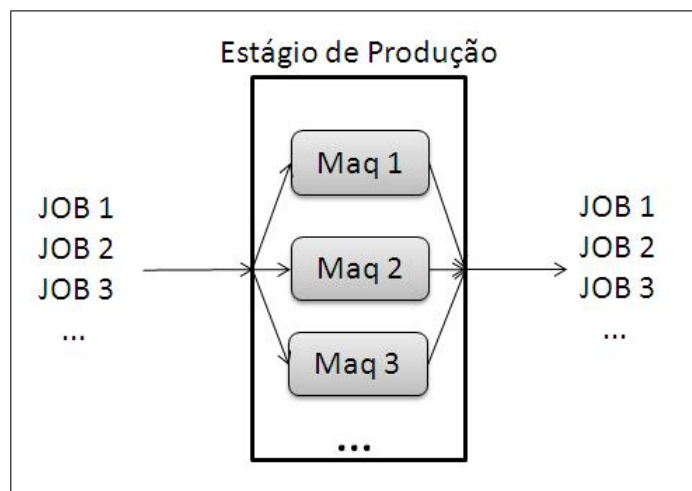


Figura 2.2: Sequenciamento em Máquinas Paralelas

Os ambientes em que pode ser enquadrado esse problema são máquinas idênticas; máquinas uniformes e; máquinas não relacionadas. O primeiro deles refere-se a problemas em que as máquinas do sistema apresentam o mesmo tempo de processamento para cada *job*. O segundo tipo define sistemas em que as máquinas possuem velocidades diferentes, assim como os tempos de processamento, que são, porém, proporcionais para a realização de qualquer *job*. Por fim o problema de máquinas paralelas não-relacionadas refere-se a máquinas nas quais as velocidades são diferentes e não há nenhuma relação entre os tempos de processamento dos *jobs* nas diferentes máquinas.

O PSMP mais clássico e estudado é o sequenciamento em máquinas paralelas idênticas com o objetivo de minimização do *makespan* $P_m || C_{max}$. Segundo Garey & Johnson [19], o PSMP para máquinas não-relacionadas, assim como para máquinas idênticas, trata-se de um problema *NP-Hard*, sendo improvável a solução ótima em tempo polinomial.

O PSMP possui diversas derivações. Podem ser considerados, por exemplo,

setups dependentes ou independentes da sequência de *jobs* nas máquinas. No segundo caso, a complexidade da formulação geralmente é menor, pois a realização do *setup* não está atrelada ao tipo de produto que está na máquina, o que restringe menos o problema. A existência de *preemption*, quando um *job* pode ter seu processamento iniciado em uma máquina, ser interrompido e depois finalizado em outra máquina diferente também pode ser observada. Sistemas em que isso é possível possuem uma modelagem mais complexa, pois dentre outros fatores, o número de variáveis tende a ser maior devido ao aumento de possibilidades para a alocação dos *jobs* nas máquinas. Derivações na função objetivo também são estudadas, como a minimização do atraso total e minimização do adiantamento ou atraso ponderado, quando se insere uma penalidade para o adiantamento e/ou atraso de cada *job*.

As funções objetivo mais comuns são relacionadas ao tempo, mas não é difícil observar estudos que consideram também os custos do sequenciamento. A minimização de custos pode estar associada ao custo do processo produtivo, da alocação de máquina ou ainda de realização do *setup*, de modo que é possível encontrar trabalhos em que mais de um tipo de custo é considerado. Problemas multiobjetivo também podem incluir objetivos relacionados a tempo e a custo simultaneamente, o que normalmente acontece com a definição de penalidades monetárias para o atraso, por exemplo.

O problema de sequenciamento de máquinas paralelas já foi largamente estudado, de modo que é possível encontrar algumas pesquisas e compilações de trabalhos sobre o assunto. Cheng & Sin [11] apresentam um abrangente estudo sobre sequenciamento em máquinas paralelas e destacam as subáreas com potencial para desenvolvimento de novas pesquisas na época. Uma pesquisa similar foi desenvolvida por Mokotoff [38], mas com o foco em máquinas paralelas idênticas.

Glass *et al.* [21] abordaram o PSMP com *setups* dependentes da sequência e objetivo de minimização do atraso total em um estudo no qual foram comparadas três meta-heurísticas, Busca Tabu (BT), Algoritmos Genéticos (AG) e *Simulated Annealing* (SA). Adomopolus & Pappis [1] propuseram uma heurística polinomial incluindo datas de entrega comuns e penalidades para adiantamentos e atrasos. O mesmo problema foi estudado por Pereira-Lopes & De Carvalho [33] através da implementação de um algoritmo do tipo *branch-and-price*.

O PSMP para máquinas paralelas não-relacionadas foi estudado por Rocha *et al.* [46]. O problema considerava datas de entrega e *setup* dependente da sequência e da máquina. Foram comparados modelos de programação inteira-mista e um algoritmo

branch and bound que utilizava como limite superior uma solução gerada pela heurística *Greedy Randomized Adaptive Search Procedure* (GRASP). Instâncias com até 30 *jobs* foram utilizadas nos modelos que tinham como função objetivo a minimização do *makespan* adicionado ao atraso ponderado. Os dois modelos matemáticos utilizados na comparação são provenientes das formulações de Wagner [50] e Manne [35]. Paula *et al.* [42] abordaram o mesmo problema e utilizaram instâncias com até 180 *jobs* e 6 máquinas, obtendo soluções ótimas para um tempo razoável de solução. Foi proposto um algoritmo *non-delayed relax-and-cut*, baseado na relaxação lagrangiana de uma formulação indexada no tempo deste problema.

O modelo baseado na formulação de Manne [35] apresentado por Rocha *et al.* [46] é descrito a seguir. Os parâmetros são os seguintes:

- N : conjunto de *jobs* a serem processados;
- M : conjunto de máquinas;
- d_i : data de entrega desejada para o *job* i ;
- w_i : é a prioridade (ou peso) do *job* i ;
- p_{im} : é o tempo de processamento do *job* i na máquina m ;
- s_{ijm} : é o tempo de *setup* do *job* i para o *job* j na máquina m ;
- G : é uma constante de valor muito elevado.

As variáveis de decisão da formulação são definidas como:

- t_i : data de início do processamento do *job* i ;
- $\alpha_{im} = \begin{cases} 1, & \text{se o } \textit{job } i \text{ é processado na máquina } m; \\ 0, & \text{caso contrário.} \end{cases}$
- $\beta_{ijm} = \begin{cases} 1, & \text{se os } \textit{jobs } i \text{ e } j \text{ são processados na máquina } m \text{ e} \\ & i \text{ é processado antes de } j; \\ 0, & \text{caso contrário.} \end{cases}$
- ρ_i : é o atraso do *job* i ;
- Z : tempo máximo para completar todos os *jobs* (*makespan*).

A seguir o modelo apresentado por Rocha *et al.* [46]:

$$\min \left(Z + \sum_{i \in N} (\rho_i * w_i) \right) \quad (2.1)$$

$$\sum_{m \in M} \alpha_{im} = 1 \quad \forall i \in N \quad (2.2)$$

$$Z - t_i + (1 - \alpha_{im})G \geq p_{im} \quad \forall i \in N, m \in M \quad (2.3)$$

$$\rho_i - t_i + (1 - \alpha_{im})G \geq p_{im} - d_i \quad \forall i \in N, m \in M \quad (2.4)$$

$$(1 - \alpha_{im})G + (1 - \alpha_{jm})G + (1 - \beta_{ijm})G + t_j - t_i \geq p_{im} + s_{ijm} \quad \forall i < j, (i, j) \in N, \quad (2.5)$$

$$\forall m \in M$$

$$(1 - \alpha_{im})G + (1 - \alpha_{jm})G + (\beta_{ijm})G + t_i - t_j \geq p_{jm} + s_{jim} \quad \forall i < j, (i, j) \in N, \quad (2.6)$$

$$\forall m \in M$$

$$\alpha_{im} \in \{0, 1\} \quad \forall i \in N, m \in M \quad (2.7)$$

$$\beta_{ijm} \in \{0, 1\} \quad \forall i < j, (i, j) \in N, \quad (2.8)$$

$$\forall m \in M$$

$$t_i, \rho_i \geq 0 \quad \forall i \in N \quad (2.9)$$

$$Z \geq 0 \quad (2.10)$$

Conforme citado anteriormente, a função objetivo visa minimizar o *makespan* acrescido do atraso ponderado. O grupo de restrições (2.2) tem a função de garantir que cada *job* passe por uma máquina. As restrições (2.3) fazem o cálculo do *makespan* e as restrições (2.4) fazem o cálculo do atraso de cada *job*. Em (2.5) e (2.6) é garantido que o *job j* iniciará seu processamento somente após o *job i* e o *setup* de *i* para *j*. Isso se ambos estiverem na mesma máquina *m* ($\alpha_{im} = \alpha_{jm} = 1$) e se *j* é processado após *i* na máquina *m* ($\beta_{ijm} = 1$). O mesmo acontece quando *j* preceder *i*. As restrições (2.7) a (2.10) definem o domínio das variáveis.

Muitas vezes, o atendimento no prazo e a menor utilização da máquina para um dado lote deixam de figurar entre os principais objetivos do sequenciamento. Isso acontece em ambientes onde esses cenários não resultam em um custo mínimo para o processo ou ainda existem novos elementos que possuem grande importância financeira. No cenário encontrado na maioria dos mercados atuais, em que a concorrência é cada vez mais forte e qualificada, o controle e redução dos custos do processo se torna o primeiro objetivo do gestor. De modo que um sequenciamento ótimo passa a não considerar o menor atraso total, mas o menor custo do processo, dado um limite aceitável

de atraso, por exemplo. Outra situação onde os custos se tornam o objetivo principal pode ser observada quando o *setup* tem um custo de execução elevado, independente do tempo de duração e da sequência de *jobs*. Nesse caso, é importante fazer o menor número de *setups* possível, mas sujeito a outras condições, como incompatibilidade de *jobs* na mesma sequência ou ainda outros custos que podem surgir ao tentar agrupar produtos de lotes diferentes.

Um problema onde a minimização do custo do atraso e do custo de ociosidade da máquina foram o foco foi estudado por Cao *et al.* [10]. Tratou-se de um sequenciamento em máquinas paralelas onde atender os clientes no prazo podia representar um tempo de espera de máquina dispendioso para a companhia. Os autores desenvolveram um modelo de otimização combinatória e posteriormente um algoritmo heurístico baseado na meta-heurística Busca Tabu para localizar soluções ótimas ou próximas do ótimo. Instâncias consideradas grandes foram resolvidas, incluindo mais de 60 *jobs* e 6 máquinas.

Fang & Lin [16] estudaram recentemente um PSMP para máquinas não relacionadas onde as velocidades das máquinas podiam ser ajustadas ao longo do processo. Tal característica permite flexibilidade para minimizar o custo relacionado à eletricidade gasta no processo, no entanto isso pode ocasionar num tempo maior de duração do processo. O problema associado era o de alocar os *jobs* às máquinas e determinar a sequência e a velocidade de cada máquina no processamento de cada *job*. O objetivo era o de minimizar o atraso ponderado e o custo total de energia. Foram propostos dois algoritmos heurísticos e um algoritmo do tipo *Particle Swarm Optimization* (PSO), que ofereceu resultados de qualidade em um tempo computacional considerado aceitável. Foram resolvidas instâncias com até 50 *jobs* e 5 máquinas.

Um problema de sequenciamento em máquinas idênticas com janela de tempo fixa para a entrega dos *jobs* foi estudado recentemente por Janiak *et al.* [25]. Foram considerados custos de adiantamento e atraso, além de um custo adicional que é considerado apenas quando os *jobs* não são entregues dentro da janela de tempo. O objetivo era o de definir um sequenciamento assim como um tamanho e uma posição para a janela de tempo. Isso é conseguido quando a soma do custo associado ao adiantamento, ao atraso e à posição da janela de tempo é o menor possível. Propriedades que definem soluções ótimas foram estabelecidas e algoritmos de programação dinâmica baseados nessas propriedades foram desenvolvidos para solucionar o problema.

Diferentes funções objetivo foram testadas por Krasik *et al.* [31] para um

problema de sequenciamento em máquinas paralelas idênticas e múltiplos produtos. Cada produto possuía uma demanda e um custo de *setup* associado à produção do lote. As funções objetivo testadas eram definidas por três componentes: custo de adiantamento, custo de atraso e custo de *setup*.

Em alguns problemas o custo do *setup* é diretamente proporcional ao tempo de *setup*, conseqüentemente, um sequenciamento ótimo em relação a tempo também é ótimo em relação a custo. No entanto, existem outros cenários em que essa proporcionalidade não existe e um *trade-off* de custos de diferentes naturezas pode ser observado. Anglani *et al.* [2] desenvolveram um programa matemático para aplicação em um caso real com uma lógica *fuzzy* considerando incerteza nos tempos de processamento. O processo permitia ajuste do tempo de processamento dos *jobs*, o que pode ser interessante para o atendimento da demanda. O *trade-off* ocorre entre a minimização do custo total de *setup* e a maximização do grau mínimo de necessidade definido para o atendimento das datas de entrega. Quanto mais *setups*, mais fácil é atender às datas de entrega, porém o custo será elevado. O método passa pela solução de um modelo aproximado do modelo original, que é um programa não linear inteiro-misto. O desvio médio em relação ao programa linear inteiro-misto derivado do modelo original foi de apenas 1,5 %. Foram resolvidos 10 problemas teste de modo que cada um continha 4 tipos diferentes de produtos e a demanda correspondia a 60 dias de trabalho, resultados considerados muito bons para o problema.

Gurel & Akturk [22] tratam o problema da minimização do custo total de manufatura e do tempo para completar todos os *jobs* simultaneamente em máquinas paralelas idênticas. No processo analisado os tempos de processamento são controláveis e o custo de manufatura aumenta com a diminuição do tempo de duração do processo. Com isso os dois objetivos não podem ser minimizados ao mesmo tempo. Tratavam-se de máquinas CNC em que se podia reduzir o tempo de processamento aumentando a velocidade de corte ou a taxa de alimentação da máquina, o que ocasionava no aumento do custo de produção. Assim é gerado o *trade-off* onde de um lado está o sequenciamento ótimo em tempo, com o mínimo tempo de finalização possível, e do outro o sequenciamento ótimo em custo, no qual o tempo total de finalização tende a ser o máximo possível. São provadas algumas propriedades de otimalidade, o que permitiu o desenvolvimento de um algoritmo heurístico eficiente. Como exemplo, uma das propriedades é a de que a regra SPT (*Shortest Processing Time*) é ótima para o primeiro objetivo. Mas tal *trade-off* não está na função objetivo da formulação, o modelo apresentado é adaptado para um objetivo de minimização do custo total de

manufatura sujeito a uma restrição de tempo total de finalização dos *jobs*. Testes foram realizados com até 200 *jobs* e 6 máquinas e testes com a utilização de *solvers* de programação não linear garantiram a identificação de ótimos locais. A heurística também alcançou ótimos locais e com uma economia de tempo considerável.

Tendo em vista os tipos de problemas apresentados na revisão bibliográfica, este trabalho pode ser classificado como um sequenciamento em máquinas paralelas no qual os custos do processo possuem extrema relevância. A Figura 2.3 busca localizar o problema estudado frente aos apresentados.

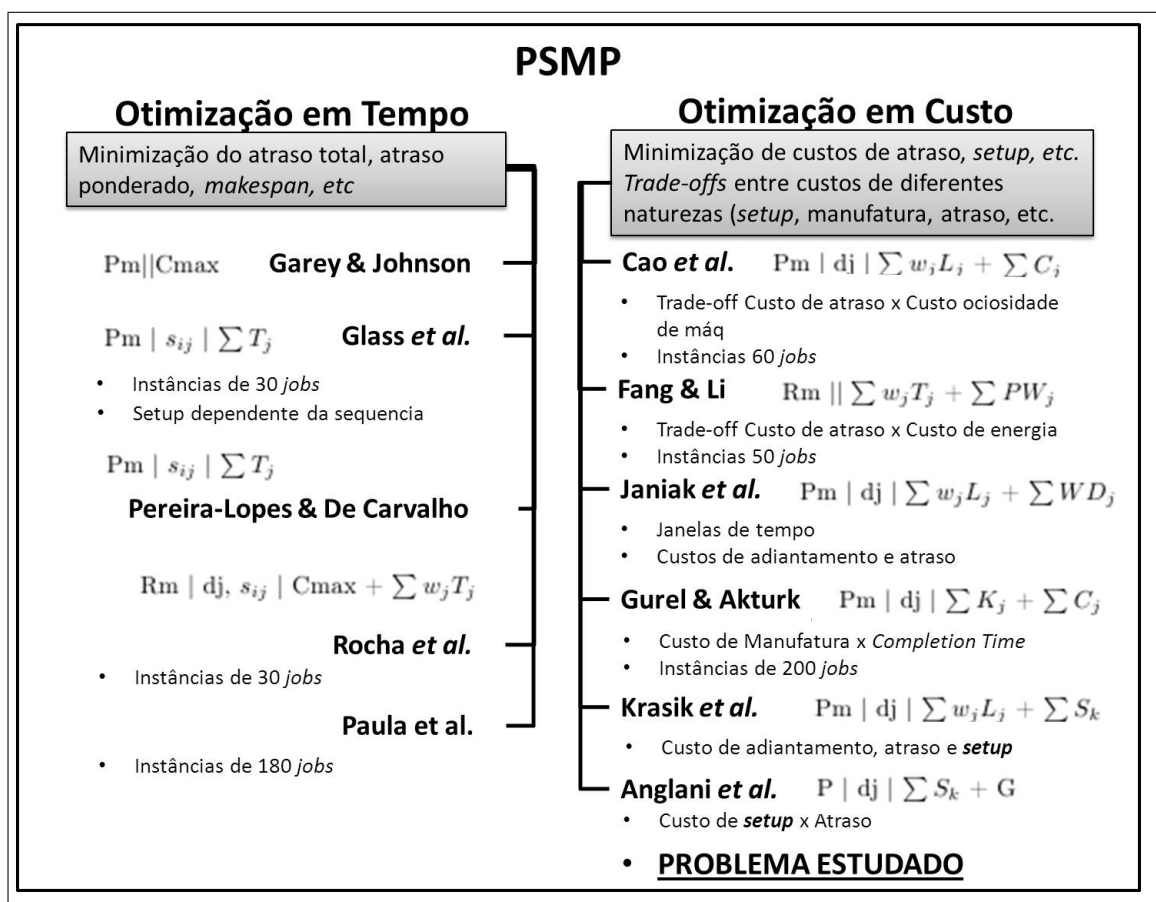


Figura 2.3: Sequenciamento em Máquinas Paralelas

2.1.2 Sequenciamento em Aciaria

O estudo de problemas de planejamento da produção no processo produtivo de aciarias via pesquisa operacional já vem sendo explorado há pouco mais de duas décadas, sendo que é possível encontrar alguns trabalhos sobre sequenciamento nesse tipo de ambiente. Boa parte desses estudos incluem aplicações em casos reais, todavia, não

há uma grande uniformidade entre os trabalhos encontrados na literatura, isso devido às diferenças entre os sistemas produtivos, ao contexto em que a empresa está inserida e ao enfoque dado ao estudo. É comum que fábricas com produção em larga escala estejam preocupadas com o alcance de grandes patamares de produção, já siderúrgicas com capacidades menores e um mix de produtos mais variado estão preocupadas com a redução dos custos envolvidos no processo. Ultimamente a maioria das companhias siderúrgicas brasileiras pode ser enquadradas no segundo perfil, tendo em vista a forte concorrência chinesa.

Em trabalhos de sequenciamento em aciarias com lingotamento contínuo (SCC - *Steelmaking-continuous Casting*) normalmente as panelas são consideradas os *jobs* a serem sequenciados nos estágios, normalmente compostos por refino primário, refino secundário e lingotamento contínuo. Esse último é o responsável pela necessidade de agrupamento das panelas em séries ou *casts*, devido à natureza contínua do processo. Entre uma série e outra ocorre a operação de *setup*, na qual, dentre outras atividades, ocorre a substituição do distribuidor. A vida útil do distribuidor é uma restrição que aparece com frequência em um problema do tipo SCC. Outro conceito importante é o de grade de aço, que define um tipo de aço quanto à composição química. Uma família de aço contém diferentes grades, mas com certa similaridade. A produção consecutiva de dois *jobs* com grades muito diferentes é algo que todo modelo visa evitar ou impedir, pois ocasionaria na produção de um subproduto que, muitas vezes, possuirá baixo valor de mercado: a placa de mistura.

É possível encontrar desde trabalhos que consideram o planejamento integrado da produção do aço, que chegam a englobar a aciaria, com todos os seus estágios, e a laminação, até estudos que analisam apenas o sequenciamento no lingotamento contínuo, local onde é gerada a maioria das restrições do processo. A figura 2.4 ilustra os principais elementos do sequenciamento em Aciaria.

Tang *et al.* [47] apresentam uma pesquisa abrangente sobre programas de gestão da produção integrada do aço. Tais programas são destinados a definir o sequenciamento das áreas de refino, lingotamento contínuo e laminação. Tal planejamento integrado fornece ganhos consideráveis, principalmente na redução do estoque de placas, no aumento de entregas no prazo e na economia de energia. Os autores referenciam alguns trabalhos sobre programação e sequenciamento do aço e classificam os trabalhos em classes de acordo com o tipo de método, variando de métodos baseados em pesquisa operacional, que incluem algoritmos exatos e heurísticos, até métodos que necessitam da interação homem-máquina para gerar o plano de produção.

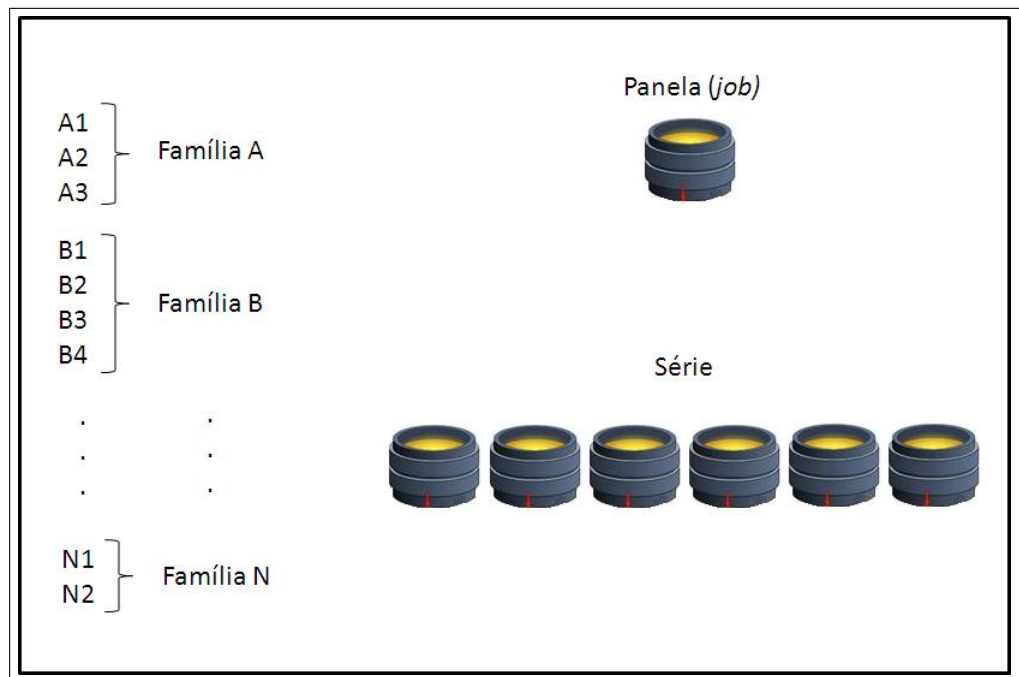


Figura 2.4: Elementos do Sequenciamento em Aciaria

Harjunkoski & Grossmann [23] desenvolvem um algoritmo de decomposição para solução de problemas de sequenciamento no ambiente SCC. O estudo propôs a desagregação do problema original, de forma que os tipos de aço foram divididos em famílias e estas separadas em grupos e então era solucionada uma sequência de pequenos problemas para cada grupo e, por fim, o sequenciamento dos grupos era realizado. Todos esses problemas são do tipo Programação Linear Inteira-Mista (PLIM). Os autores consideraram o tempo de *setup* na MLC dependente da sequência e ainda manutenção dos equipamentos.

Pacciarelli & Pranzo [41] solucionam um problema do tipo *Flow Shop* para um processo produtivo de uma aciaria que contava com um Convertedor LD (refino primário), dois equipamentos de refino secundário e uma Máquina de Lingotamento Contínuo. Com o objetivo de minimização do *makespan*, um procedimento a partir da técnica *beam search* foi desenvolvido e apresentou bons resultados. As principais restrições consideradas pelos autores foram de tempo limite de espera entre duas painelas de aço (*jobs*) em uma operação e agrupamento de *jobs* com a mesma grade. Caso *jobs* idênticos fossem processados consecutivamente, esses eram agrupados e não haveria restrição de tempo de espera entre eles na MLC. Além disso, foi considerada a perecibilidade no *buffer* antes da MLC e tempos de *setup* independentes da sequência entre duas séries consecutivas.

Bellabdaoui & Teghem [7] apresentam um estudo de caso de uma aplicação do problema SCC em uma empresa do grupo Arcelor, na Bélgica. Uma formulação de PLIM foi apresentada para resolver o problema com o objetivo de maximizar a produtividade. Dentre as restrições consideradas estão o agrupamento de *jobs* para a alocação em uma das máquinas de lingotamento contínuo, restrições de precedência entre os *jobs* de uma mesma sequência, tempo de espera dos *jobs* entre cada estágio e não existência de tempo ocioso na máquina de lingotamento contínuo. As instâncias resolvidas possuíam de 6 a 8 *jobs* e foram resolvidas por um *solver* comercial.

Tang *et al.* [48] desenvolvem um algoritmo heurístico para tratar o problema de sequenciamento no lingotamento contínuo. O objetivo considerava a maximização do número de corridas (*jobs*) em uma série. Nesse estudo não é obrigatório o sequenciamento de todos os *jobs* disponíveis, mas é penalizada na função objetivo a não inclusão de um *job* no sequenciamento. Além dessa penalização, são considerados também o custo de *setup* e um custo associado à diferenças entre as grades de aço de *jobs* subsequentes. O número de corridas em uma série é limitado pela vida útil do distribuidor (dada em número de corridas), pela similaridade entre as grades dos *jobs* e pela largura das placas. A lógica utilizada é a de que para um dado número de corridas (*input*), quanto mais corridas serem inseridas em uma série, menos *setups* serão realizados. Os autores utilizaram de procedimentos heurísticos para o agrupamento de corridas. Heurísticas construtivas proporcionam a organização dos *jobs* em listas tendo como o primeiro critério a grade de aço e o segundo a largura (em ordem decrescente) ou vice-versa e, posteriormente, faziam o agrupamento em séries levando em consideração o limite de variação de largura e o tamanho máximo de série. Uma das heurísticas elaboradas é descrita a seguir. Por fim, é executada uma rotina baseada na meta-heurística *Iterated Local Search* (ILS). É importante notar que os métodos aplicados não se preocupam com o sequenciamento interno das séries.

- *Heurística de Agrupamento de Painéis de Tang et al. [48]:*
 - Passo 1: Organizar as painéis em uma lista que tenha como primeiro critério a similaridade entre grades de aço e como segundo critério a largura, sendo o primeiro em ordem crescente e o segundo em ordem decrescente.
 - Passo 2: Retirar as painéis da lista inicial e inserir em uma nova lista usando como critério a possibilidade de processar as painéis subsequentemente de acordo com a grade.
 - Passo 3: Organizar a nova lista em ordem decrescente de larguras.

- Passo 4: Retirar as painéis da lista para formar as séries de modo que seja respeitado o limite de variação de largura e o tamanho máximo de uma série. Repetir este passo até que a nova lista esteja vazia e, com isso, as séries formadas.

Uma aplicação em caso real do problema de sequenciamento em aciarias foi estudada por Atighehchian *et al.* [4], que trataram o sequenciamento das etapas de refino e Lingotamento Contínuo como um problema de *flow shop* híbrido. Métodos de otimização não-linear são combinados com o método de colônia de formigas na estratégia utilizada. Duas fases consistem na construção da solução, a fase de alocação dos *jobs* nas máquinas e a fase de sequenciamento dos *jobs* nas máquinas. Os *jobs* em cada máquina são agrupados em séries, desde que satisfaçam restrições de largura e similaridade de grades de aço com o *job* subsequente. Na função objetivo do problema são minimizados o número de quebras na sequência (tempo entre séries); tempo de espera das painéis antes do LC; violação do tempo limite de espera, que ocasiona em aço de baixa qualidade; custo do *makespan* e; ganho com a quantidade produzida. Este último é contabilizado com o sinal negativo na função objetivo.

Outra aplicação em caso real foi apresentada por Missbauer *et al.* [37] em que se buscou maximizar o número de *jobs* produzidos em um dia. O problema também considerava as fases de refino e posteriormente o Lingotamento Contínuo. Assim como Atighehchian *et al.* [4], os autores trataram o problema como um *flow shop* híbrido e apresentaram um modelo de programação linear inteira-mista para resolver o mesmo. No contexto do problema as pontes rolantes que fazem o transporte das painéis de aço entre os estágios foram consideradas facilidades e o transporte foi considerado como uma operação. A implantação e calibração do sistema em uma siderúrgica austríaca durou cerca de dez anos.

O mesmo problema é ainda tratado por Tang *et al.* [49], que desenvolveram um método que combina Relaxação Lagrangiana e Geração de Colunas para resolver o problema. O problema tratado foi definido, assim como os anteriores em agrupar *jobs* em sequências menores de acordo com afinidades de grades de aço e dimensões das placas, no entanto foi utilizado um recurso para reduzir o número de combinações e diminuir ou eliminar os resíduos. Os *jobs* ou painéis são primeiramente agrupados em *charge-lots*, que são constituídos de painéis que possuem a mesma composição química e a mesma largura. Dessa forma o tamanho do problema é reduzido e instâncias maiores podem ser resolvidas com maior facilidade. Para a fase de testes foram utilizadas

instâncias geradas a partir de um histórico de produção de sete dias de uma grande siderúrgica chinesa e também instâncias geradas aleatoriamente. Soluções próximas do ótimo foram conseguidas com o método combinado, de modo que os resultados foram considerados satisfatórios.

Mais recentemente Li *et al.* [32] trabalharam no mesmo problema com o objetivo de minimização do *makespan*. Um modelo de programação linear inteira-mista foi proposto para alocar sequências pré-determinadas de *jobs* no horizonte de planejamento. A grande diferença entre o problema tratado neste trabalho e nos anteriores é a de que os autores consideram que as sequências que vão passar pelas máquinas já estão definidas a priori. O contexto em que está a empresa na qual foi realizado o estudo é o de produção em larga escala, várias máquinas em paralelo podem ser encontradas em cada estágio da Aciaria. Além disso, os *jobs* são agrupados em um estágio anterior de programação, restando para o modelo desenvolvido o sequenciamento das séries de painéis nas máquinas. Um contexto contrário ao estudado por Li *et al.* [32] foi alvo do estudo de Zaroni & Zavanella [51]. O objetivo do trabalho foi de minimização dos custos de manutenção de estoques, dos custos de produção e dos custos de atraso na entrega. O cenário do problema considerava uma pequena siderúrgica com restrições de espaço para armazenagem, além de um mix de grades de aço muito variado. Com isso, o objetivo final visava determinar um sequenciamento ótimo de produção considerando o espaço disponível para estoque. As instâncias para testes foram geradas aleatoriamente com base no ambiente estudado.

É possível notar uma vasta gama de problemas de sequenciamento em aciarias. Abordagens e adaptações variam de acordo com o contexto da pesquisa ou do cenário em que está inserida a siderúrgica analisada. Podem ser destacados alguns pontos que acabam por determinar a definição de como o problema será abordado. O primeiro deles diz respeito à relação entre a empresa e o mercado na qual ela atua. Nesse sentido podem ser definidos os objetivos do modelo de modo a satisfazer necessidades de redução de custo, de nível de serviço e atendimento ao cliente ou ainda de maximização da produção. Outro fator importante é o mix de produtos com a qual a companhia trabalha. Uma grande variedade de tipos de aço implica em um desafio maior no momento de montar as séries, podendo até mesmo resultar em séries que não exploram toda a vida útil do distribuidor. Além disso outros fatores como custo e duração do *setup* e geração de estoque podem ser relevantes a ponto de serem destacados no objetivo do modelo.

2.1.3 Sequenciamento em Máquina Simples

Esta seção destina-se a apresentar o problema de sequenciamento em máquina simples. A necessidade de aprofundar neste tipo específico de problema foi identificada ao longo da pesquisa, a partir dos primeiros testes realizados para o problema. A justificativa para essa alteração é apresentada na seção 3.4.

O problema de sequenciamento em máquina simples (PSUMAA) é classificado como NP-difícil [43]. Nesse tipo de problema o sequenciamento possui apenas um estágio e uma máquina. Com isso, todos os produtos são processados em uma única máquina. Sistemas produtivos em que existem estágios em série e integrados também podem ser tratados como máquina simples.

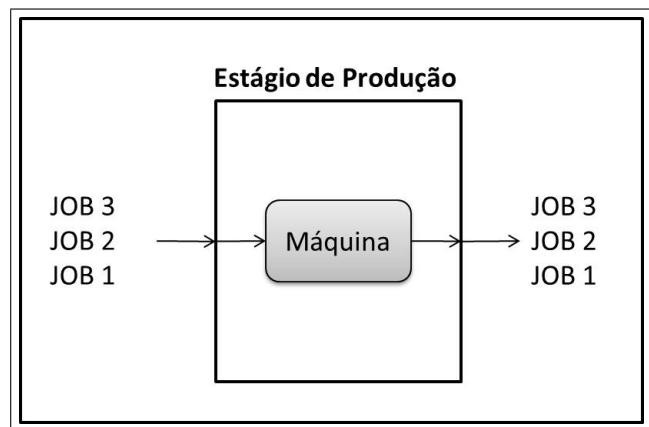


Figura 2.5: Sequenciamento em Máquina Simples

Na pesquisa desenvolvida por Keha *et al.* [29] é apresentada uma compilação e uma comparação quanto à performance computacional de diferentes tipos de formulações para o problema de sequenciamento em uma máquina. São analisadas formulações de programação linear inteira-mista com quatro tipos de variáveis de decisão diferentes:

- Variáveis de data de conclusão;
- Variáveis indexadas no tempo;
- Variáveis de ordenação linear;
- Variáveis indexadas pela posição/alocação.

Para os tipos de formulação analisados não é permitido *preemption* e as funções

objetivo analisadas são: o tempo total ponderado para completar os *jobs*; o atraso máximo; o número de *jobs* atrasados; e o atraso total ponderado;

As comparações são feitas através, principalmente, do valor de função objetivo da relaxação linear, do número de nós explorados e do tempo de processamento para instâncias em que a solução ótima foi obtida em até uma hora. A formulação com a indexação de posição foi considerada a melhor escolha por que a relaxação linear dessa formulação pode ser resolvida mais rápido e um número maior de nós pode ser explorado em um valor fixo de tempo.

Problemas de sequenciamento bi-critério em uma máquina com variáveis de data de conclusão são apresentados por Bagchi [5] em seu trabalho. A abordagem bi-critério acontece através da minimização simultânea dos objetivos da média e da variação do tempo de conclusão, o mesmo acontece para o tempo de espera dos *jobs*. O objetivo final considerado é a minimização do custo total de sequenciamento. Para isso é definida uma função linear do custo de sequenciamento em função da média do tempo de conclusão e da variação do mesmo. O problema de sequenciamento em uma máquina com objetivo de minimização da variância das datas de conclusão foi também estudado por Kanet [28].

Koulamas & Kyparisis [30] estudaram o problema de sequenciamento em máquina simples com tempo de *setup* dependente da sequência e variáveis de data de conclusão. Diferentes tipos de função objetivo são considerados, minimização do *makespan*, soma do tempo de conclusão dos *jobs*, diferença absoluta entre os tempos de conclusão e uma combinação dos dois últimos objetivos. É mostrado que o problema padrão de sequenciamento de máquina simples com *setup* dependente da sequência e com qualquer uma das funções objetivo consideradas pode ser solucionado com um procedimento de ordenação com complexidade $O(n \log n)$, onde n é o número de *jobs*.

Assim como métodos exatos são aplicados ao problema de sequenciamento em uma máquina, métodos heurísticos eficazes também podem ser encontrados na literatura, incluindo busca local simples [34], colônia de formigas [6], *Simulated Annealing* (SA) [40] e *Iterated Local Search* (ILS) [15]. Besten *et. al.* [15] apresentam um algoritmo extremamente potente, cujos resultados superam os da abordagem *Iterated Dynasearch* [12], considerada muito eficaz. A eficiência do algoritmo ILS apresentado é boa parte justificada pelo método *variable neighborhood descent* (VND) utilizado na busca local. No entanto, os autores ponderam que a eficiência do método em problemas de sequenciamento depende fortemente das particularidades do problema analisado. O

PSUMAA com restrição de data de entrega comum foi estudado por Feldmann & Biskup [17], que aplicaram as meta-heurísticas *Simulated Annealing* e *Threshold Accepting* (TA). Foram alcançados resultados próximos do ótimo a partir da solução de 140 instâncias com mais de 1000 *jobs*.

As formulações de Wagner [50] e Manne [35] também são aproveitadas em problemas de sequenciamento em uma máquina. Dois modelos para sequenciamento em máquina simples e *setup* dependente da sequência são apresentados por Bustamante [9], o primeiro baseado na formulação de Manne [35] e o segundo baseado na formulação de Wagner [50]. Os modelos apresentaram desempenho equivalente e resolveram instâncias de até 10 *jobs*. A seguir será descrito o modelo baseado na formulação de Manne [35] que foi desenvolvido por Bustamante [9].

PARÂMETROS:

- n : número de *jobs* a serem processados.
- P_i : tempo de processamento do *job* i .
- D_i : data desejada para o término do *job* i (data de entrega).
- S_{ij} : tempo de preparação da máquina para processar o *job* j depois do *job* i .
- α_i : custo de antecipação da produção do produto i por unidade de tempo (manutenção do estoque).
- β_i : custo de atraso da produção do produto i por unidade de tempo.
- M : um valor muito grande.

VARIÁVEIS DE DECISÃO:

- s_i : data de início do processamento do *job* i .
- y_{ij} : variável que determina a sequência de produção, se $y_{ij}=1$ o *job* j é processado depois do *job* i e 0 caso contrário.
- e_i : tempo de antecipação do *job* i .
- t_i : tempo de atraso do *job* i .

Desta forma, tem-se o seguinte modelo de PLIM:

$$\text{minimizar } Z = \sum_{i=1}^n (\alpha_i e_i + \beta_i t_i) \quad (2.11)$$

$$\text{sujeito a: } s_j - s_i - (M + S_{ij})y_{ij} \geq P_i - M \quad \forall i, j = 1, 2, \dots, n \text{ e } i \neq j \quad (2.12)$$

$$y_{ij} + y_{ji} = 1 \quad \forall i, j = 1, 2, \dots, n \text{ e } i \neq j \quad (2.13)$$

$$s_i + P_i + e_i - t_i = D_i \quad \forall i = 1, 2, \dots, n \quad (2.14)$$

$$s_i \geq 0 \quad \forall i = 1, 2, \dots, n \quad (2.15)$$

$$e_i \geq 0 \quad \forall i = 1, 2, \dots, n \quad (2.16)$$

$$t_i \geq 0 \quad \forall i = 1, 2, \dots, n \quad (2.17)$$

$$y_{ij} \in \{0, 1\} \quad \forall i, j = 1, 2, \dots, n \quad (2.18)$$

No modelo apresentado a função objetivo(2.11) visa a minimização dos custos de antecipação e atraso. O conjunto de restrições (2.12) calcula as datas de início do processamento dos itens na máquina. Nas restrições (2.13) é garantido que o valor de y_{ji} será 0, caso o *job j* seja processado após o *job i* ($y_{ij}=1$).Em (2.14) são calculados o atraso e a antecipação de acordo com a data de entrega do *job*. Nas restrições (2.15) a (2.18) são definidos os domínios das variáveis do problema.

2.2 Definição do Problema

A definição do escopo do trabalho foi norteadada pelas principais características do ambiente e pelos aspectos mais relevantes que precisam ser considerados na programação da aciaria em questão. O ambiente estudado é composto por duas máquinas de lingotamento contínuo. Tais equipamentos estão disponíveis para receber a programação de painéis de aço (tratadas como *jobs*) que devem ser agrupadas em séries.

O problema tratado neste trabalho é o sequenciamento de painéis em um lingotamento contínuo onde as despesas com *setup* têm grande relevância, principalmente pelo fato de a companhia possuir um mix de produtos muito variado. Tal mix faz com que a programação da produção algumas vezes coloque em sequência painéis com tipos de aço que não são totalmente compatíveis, o que gera uma despesa de outra natureza, despesa devido à placa de mistura. A placa de mistura é a parte do aço solidificado que contém aço de duas painéis, caso essas painéis possuam aços diferentes, essa placa de mistura tem uma composição química que não pode ser enquadrada em nenhum

dos dois tipos e, por muitas vezes, origina um aço com valor de mercado mais baixo. Apesar de terem sido introduzidos na última seção os principais conceitos sobre esse tipo de problema, são confirmados a seguir os principais aspectos do problema.

- Um *job* corresponde a uma panela de aço;
- Uma série corresponde a um conjunto de panelas processadas sem interrupção (*setup*);
- Sequenciamento em máquinas paralelas idênticas;
- Há um tamanho limite para uma série, em virtude da vida útil do distribuidor;
- Existe um limite de variação máximo de largura entre duas panelas;
- Há um custo associado à operação de *setup*;
- Há um custo associado ao sequenciamento de duas panelas de grades diferentes de forma subsequente (custo de mistura);
- A mistura não ocorre quando as panelas estão em séries diferentes, ou seja, quando há uma operação de *setup* entre as panelas.

A proposta é explorar o *trade-off* entre custo de *setup* e custo de mistura. A estratégia é possibilitar a realização dos tipos de mistura que ofereçam uma despesa menor que a despesa correspondente à operação de *setup*. Essa abordagem não foi observada nos trabalhos revisados, que em sua maioria consideram uma matriz de combinação entre os tipos de aço.

No próximo capítulo serão apresentados dois modelos para o sequenciamento do problema estudado. Os modelos foram desenvolvidos a partir de abordagens para problemas de sequenciamento em máquinas paralelas.

Capítulo 3

Modelos Matemáticos

Considerando o apresentado na revisão bibliográfica foi definido como estratégia inicial o desenvolvimento de modelos de programação linear inteira-mista para tentar resolver o problema de sequenciamento. A seguir são apresentados os dois modelos formulados e algumas peculiaridades de cada um. Posteriormente são apresentados os resultados, limites e considerações acerca de cada modelo. Em comum, os modelos possuem os seguintes parâmetros de entrada:

N = conjunto de *jobs*;

M = conjunto de máquinas;

$b_{ij}^m = 0$, se não pode haver mistura de i para j , por causa de composição química ou variação na largura das placas geradas. Caso contrário, $b_{ij}^m = 1$;

p_{im} = tempo de processamento do *job* i na máquina m ;

l_{max} = intervalo de tempo máximo entre o fim de um *setup* e o início do próximo;

f_{ij}^m = custo da placa de mistura gerada de i para j na máquina m ;

s^m = tempo de *setup* na máquina m ;

h^m = custo de um *setup* na máquina m (preço do distribuidor);

\bar{z} = tempo máximo para término de todos os *jobs*.

3.1 Modelo 1: Variáveis de data de conclusão

O primeiro modelo foi desenvolvido a partir do modelo de sequenciamento em máquinas paralelas apresentado por Manne [35]. As variáveis de decisão que indicam as datas de início e conclusão das tarefas continuam assim definidas. No entanto, muitas alterações foram necessárias em virtude das peculiaridades do problema, o que impossibilita comparações com o modelo original. A seguir é descrito o modelo 1.

S = conjunto que define número máximo de *setups* permitidos;

G = constante de valor muito elevado.

Variáveis de decisão:

t_i = data de início do processamento do *job* i ;

$\alpha_{im} = 1$, se o *job* i for processado na máquina m e 0, caso contrário;

$\beta_{ij}^m = 1$, se o *job* j for processado imediatamente após de *job* i na máquina m e 0, caso contrário;

$q_{ij}^m = 1$, se há formação de placa de mistura do *job* i para o *job* j na máquina m e 0, caso contrário;

θ_k^m = data de início do k -ésimo *setup* na máquina m ;

$w_{ik}^m = 1$, se é feito o k -ésimo *setup* na máquina m após o processamento do *job* i e 0, caso contrário.

Modelo 1:

$$\min \sum_{m \in M} \sum_{i \in N} \sum_{j \in N} f_{ij}^m q_{ij}^m + \sum_{m \in M} \sum_{k \in S} \sum_{i \in N} h^m w_{ik}^m \quad (3.1)$$

$$\sum_{m \in M} \alpha_{im} = 1 \quad \forall i \in N \quad (3.2)$$

$$q_{ij}^m - \beta_{ij}^m + \sum_{k \in S} w_{ik}^m \geq 0 \quad \forall i, j \in N, m \in M \quad (3.3)$$

$$\sum_{i \in N} w_{ik}^m \leq 1 \quad \forall k \in S, m \in M \quad (3.4)$$

$$\sum_{k \in S} w_{ik}^m - \alpha_{im} \leq 0 \quad \forall i \in N, m \in M \quad (3.5)$$

$$(1 - \beta_{ij}^m)G + t_j - t_i - s^m \sum_{k \in S} w_{ik}^m \geq p_{im} \quad \forall m \in M, i \in N \cup \{0\}, j \in N \cup \{\hat{n}_m\} \quad (3.6)$$

$$-(1 - \beta_{ij}^m)G + t_j - t_i - s^m \sum_{k \in S} w_{ik}^m \leq p_{im} \quad \forall m \in M, i \in N \cup \{0\}, j \in N \cup \{\hat{n}_m\} \quad (3.7)$$

$$\sum_{j \in N \cup \{\hat{n}_m\}} \beta_{ij}^m - \alpha_{im} = 0 \quad \forall i \in N, m \in M \quad (3.8)$$

$$\sum_{i \in N \cup \{0\}} \beta_{ij}^m - \alpha_{jm} = 0 \quad \forall j \in N, m \in M \quad (3.9)$$

$$\sum_{j \in N} \beta_{0j}^m = 1 \quad \forall m \in M \quad (3.10)$$

$$\sum_{i \in N} \beta_{i\hat{n}_m}^m = 1 \quad \forall m \in M \quad (3.11)$$

$$q_{ij}^m \leq b_{ij}^m \quad \forall i, j \in N, m \in M \quad (3.12)$$

$$\sum_{i \in N} w_{ik}^m - \sum_{i \in N} w_{i(k-1)}^m \leq 0 \quad \forall k \in S, k \geq 2, m \in M \quad (3.13)$$

$$\theta_k^m - t_i + (1 - w_{ik}^m)G \geq p_{im} \quad \forall i \in N, k \in S, m \in M \quad (3.14)$$

$$\theta_k^m - t_i - (1 - w_{ik}^m)G \leq p_{im} \quad \forall i \in N, k \in S, m \in M \quad (3.15)$$

$$\theta_k^m - \theta_{k-1}^m \leq l_{max} + s_m \quad \forall k \in S, k \geq 2, m \in M \quad (3.16)$$

$$\theta_1^m - l_{max} \sum_{i \in N} w_{i1}^m \leq 0 \quad \forall m \in M \quad (3.17)$$

$$\theta_k^m - G \sum_{i \in N} w_{ik}^m - \theta_{k-1}^m \leq 0 \quad \forall k \in S, k \geq 2, m \in M \quad (3.18)$$

$$\theta_k^m + G \sum_{i \in N} w_{ik}^m - \theta_{k-1}^m \geq 0 \quad \forall k \in S, k \geq 2, m \in M \quad (3.19)$$

$$t_{\hat{n}_m} - \theta_{|S|}^m \leq l_{max} + s_m \quad \forall m \in M \quad (3.20)$$

$$t_{\hat{n}_m} - G \sum_{i \in N} w_{i1}^m \leq l_{max} \quad \forall m \in M \quad (3.21)$$

$$t_{\hat{n}_m} \leq \bar{z} \quad \forall m \in M \quad (3.22)$$

$$\sum_{i \in N} \alpha_{i(m-1)} - \sum_{i \in N} \alpha_{im} \geq 0 \quad \forall m \in M, m \geq 2 \quad (3.23)$$

$$\alpha_{im} \in \{0, 1\} \quad \forall i \in N, m \in M \quad (3.24)$$

$$w_{ik}^m \in \{0, 1\} \quad \forall i \in N, k \in S, m \in M \quad (3.25)$$

$$\beta_{ij}^m \in \{0, 1\} \quad \forall m \in M, i \in N \cup \{0\}, j \in N \cup \{\hat{n}_m\} \quad (3.26)$$

$$t_i \geq 0 \quad \forall m \in M, i \in N \cup \{0, \hat{n}_m\} \quad (3.27)$$

$$\theta_k^m, q_{ij}^m \geq 0 \quad \forall i, j \in N, k \in S, m \in M \quad (3.28)$$

Dois *jobs* artificiais são utilizados na formulação para abrir e fechar a sequencia em cada máquina. A variável t_0 e os parâmetros p_{0m} e $p_{\hat{n}_m}$ são fixados em zero.

A equação (3.1) apresenta a função objetivo, que visa minimizar o custo devido ao material proveniente de mistura gerado somado ao custo de *setup*. As restrições (3.2) associam um *job* i a uma única máquina m . Em (3.3) é assegurado que após o processamento do *job* i duas situações podem ocorrer: mistura ou *setup*. O conjunto de restrições (3.4) garantem que somente um *job* poderá ser alocado ao *setup* da posição k na máquina m . As restrições (3.5) garantem que após o *job* i somente poderá haver *setup* se aquele estiver alocado na máquina m . Em (3.6) e (3.7) é calculado o sequenciamento através das datas de início dos *jobs* em cada máquina m e é assegurado que não haverá ociosidade na máquina. As restrições (3.6) e (3.7) garantem que, caso $\beta_{ij}^m = 1$, será respeitado o tempo de processamento de i (p_{im}) após o início desse *job* e, caso haja *setup*, será respeitado um tempo igual a $p_{im} + s^m$. O conjunto de restrições (3.8) assegura que dois *jobs* i e j apenas poderão estar em posições adjacentes na máquina m se ambos forem alocados na máquina m . As restrições (3.10) e (3.11) asseguram que apenas um *job* sucederá e precederá os *jobs* 0 e \hat{n}_m , respectivamente.

Em (3.12) é garantido que a mistura entre i e j só ocorrerá quando $b_{ij}^m = 1$ e em (3.13) é determinado que o k -ésimo *setup* só ocorrerá se o $(k - 1)$ -ésimo *setup* já houver ocorrido na máquina m . As restrições (3.14) e (3.15) definem a data de início do *setup* na posição k na máquina m e as restrições (3.16) definem um intervalo limite entre o fim do $(k - 1)$ -ésimo *setup* e o início do k -ésimo *setup*. O conjunto de restrições

(3.17) garante que o primeiro *setup* na máquina m será realizado antes do limite l_{max} . Já nas restrições (3.18) e (3.19) é imposto que caso o k -ésimo *setup* na máquina m não for realizado, sua data de início seja igual a data de início do $(k - 1)$ -ésimo *setup*. As restrições (3.20) garantem que a data de início do último *setup* na máquina m seja após a data $t_{\hat{n}_m} - l_{max} - s^m$. Nas restrições (3.18) a (3.20) é assegurado que o limite entre *setups* l_{max} será respeitado também no fim da sequência, ou seja, após o último *setup* na máquina m . Já em (3.21) é garantido que o limite entre *setups* l_{max} será respeitado para o tamanho da sequência caso não haja *setup*. As restrições (3.22) asseguram que todos os *jobs* sejam processados dentro do horizonte de planejamento (\bar{z}). As restrições (3.23) têm o objetivo de eliminar simetrias do problema e as restrições (3.24) a (3.28) definem o domínio das variáveis.

3.2 Modelo 2: Variáveis indexadas pela posição

O Modelo 2 foi formulado a partir do modelo apresentado por Wagner [50], mas assim como no Modelo 1 muitas alterações foram necessárias, chegando a descaracterizar a formulação original, que produz o sequenciamento a partir da indexação da posição que os *jobs* ocupam na máquina.

Variáveis de decisão do Modelo 2:

t_m^k = data de início do processamento do k -ésimo *job* na máquina m ;

α_{im}^k = 1, se o *job* i for o k -ésimo *job* processado na máquina m e 0, caso contrário;

q_{ij}^m = 1, se há formação de placa de mistura do *job* i para o *job* j na máquina m e 0, caso contrário;

w_m^k = 1, se é feito *setup* após ser processado o k -ésimo *job* da máquina m e 0, caso contrário.

Modelo 2:

$$\min \sum_{m \in M} \sum_{i \in N} \sum_{j \in N} q_{ij}^m f_{ij}^m + \sum_{m \in M} \sum_{k \in N} h^m w_m^k \quad (3.29)$$

$$\sum_{m \in M} \sum_{k \in N} \alpha_{im}^k = 1 \quad \forall i \in N \quad (3.30)$$

$$\sum_{i \in N} \alpha_{im}^k \leq 1 \quad \forall k \in N, m \in M \quad (3.31)$$

$$\sum_{i \in N} \alpha_{im}^k - \sum_{i \in N} \alpha_{im}^{k-1} \leq 0 \quad \forall k \in N, k \geq 2, m \in M \quad (3.32)$$

$$\alpha_{jm}^k + \alpha_{im}^{k-1} - w_m^{k-1} - q_{ij}^m \leq 1 \quad \forall i, j, k \in N, k \geq 2, i \neq j, m \in M \quad (3.33)$$

$$q_{ij}^m \leq b_{ij}^m \quad \forall i, j \in N, m \in M \quad (3.34)$$

$$t_m^1 = 0 \quad \forall m \in M \quad (3.35)$$

$$t_m^k - t_m^{k-1} - \sum_{i \in N} \alpha_{im}^{k-1} p_{im} - w_m^{k-1} s_m = 0 \quad \forall k \in N, k \geq 2, m \in M \quad (3.36)$$

$$-\left(\sum_{v=k}^{l-1} w_m^v\right) l_{max} + t_m^l + \sum_{i \in N} \alpha_{im}^l p_{im} - t_m^k \leq l_{max} \quad \forall m \in M, k, l \in N, l > k \quad (3.37)$$

$$w_m^k - \sum_{i \in N} \alpha_{im}^k \leq 0 \quad \forall k \in N, m \in M \quad (3.38)$$

$$t_m^n + \sum_{i \in N} \alpha_{im}^n p_{im} \leq \bar{z} \quad \forall m \in M \quad (3.39)$$

$$\sum_{i \in N} \sum_{k \in N} \alpha_{im}^{k-1} - \sum_{i \in N} \sum_{k \in N} \alpha_{im}^k \geq 0 \quad \forall m \in M, m \geq 2 \quad (3.40)$$

$$\alpha_{im}^k, w_m^k \in \{0, 1\} \quad \forall i, k \in N, m \in M \quad (3.41)$$

$$t_m^k, q_{ij}^{km} \geq 0 \quad \forall i, j, k \in N, m \in M \quad (3.42)$$

Na equação (3.29) é apresentada a função objetivo, que é a mesma do modelo 1. Em (3.30) é garantido que um *job* i seja processado em somente uma posição k e em somente uma das m máquinas. Em (3.31) é assegurado que em cada posição k de cada máquina m haverá no máximo um *job* e esta alocação só é possível caso haja outro *job* na posição $k - 1$ da máquina m (3.32). No conjunto de restrições (3.33) é garantido que, caso dois *jobs* sejam processados nas posições $k - 1$ e k na máquina m , entre eles deverá ocorrer *setup* ou mistura. Já nas restrições (3.34) é limitada a ocorrência de mistura para quando $b_{ij}^m = 1$.

A restrições (3.35) definem as datas de início para os *jobs* na primeira posição de cada máquina e as restrições (3.36) calculam a data de início dos *jobs* a partir da data de início do processamento do *job* na posição anterior somado ao tempo de processamento do *job* na máquina e ao tempo de *setup*, caso haja. As restrições (3.37)

asseguram o limite l_{max} entre dois *setups* e as restrições (3.38) impedem a ocorrência de *setup* após a posição k caso não haja *job* alocado nessa posição. Em (3.39) é definido o horizonte de planejamento (\bar{z}) para a execução dos *jobs*. As restrições (3.40) objetivam amenizar o efeito da simetria do problema e as restrições (3.41) a (3.42) definem o domínio das variáveis.

3.3 Instâncias Reais

A geração do primeiro grupo de instâncias para o problema foi realizada a partir de informações fornecidas pela organização. Dados históricos do sequenciamento no Lingotamento Contínuo referentes a um ano de produção foram fornecidos, além de alguns documentos e tabelas complementares. Nesses registros de produção constavam as grades de aço e as famílias de aço das panelas produzidas, a composição das séries, a máquina em que a panela foi processada, a largura e ainda qual produto seria obtido daquela panela no futuro. Documentos complementares com o valor de mercado dos produtos e a composição química de cada grade de aço também foram utilizados. A composição química foi necessária para, a partir de uma tabela fornecida pela empresa, calcular a quantidade de mistura que seria gerada em uma eventual mistura de aços de grades diferentes.

Nos dados fornecidos pela empresa foi possível identificar 238 tipos diferentes de aço (grades diferentes) e esses foram agrupados em 7 famílias de acordo com semelhanças de composição química.

A partir do histórico fornecido foram selecionados alguns períodos e extraídas 48 instâncias que contêm de 54 a 72 *jobs*. A partir dessas instâncias iniciais foram ainda geradas 48 instâncias menores para possibilitar a resolução nos modelos matemáticos. Para essa geração houve a preocupação em manter as principais características das instâncias maiores, como número de séries e tipos diferentes de aço.

Os tempos de processamento dos *jobs* são determinados de acordo com a grade de aço do *job* e a família a qual ela pertence. A cada família é atribuído um tempo de processamento, que pode assumir três valores: 40, 42 e 44 minutos. Já o tempo de *setup* não possui variação, é fixado em 90 minutos, e o tempo máximo entre dois *setups* consecutivos é de 528 minutos, que corresponde ao processamento de 12 *jobs* de 44 minutos. Segundo a empresa, 12 corridas é um valor médio para a vida útil do distribuidor, que pode ser um dos motivos da realização do *setup*.

O parâmetro b_{ij}^m corresponde a uma matriz que define a possibilidade de sequenciamento consecutivo entre dois *jobs* i e j de acordo com a diferença de largura entre eles. Caso essa diferença seja menor ou igual ao limite de variação de largura máximo, que é de 150 mm, o valor do parâmetro é 1, caso contrário esse valor é zero. O horizonte de planejamento corresponde à soma dos tempos de processamento dos *jobs*.

O custo de mistura entre dois *jobs* i e j foi definido em algumas etapas. A partir da tabela citada no primeiro parágrafo desta seção é possível extrair a quantidade de mistura gerada no caso de um sequenciamento consecutivo de quaisquer tipos de *jobs*. Em paralelo, a partir de outro procedimento, foi possível determinar qual a composição química aproximada de uma placa de mistura resultante da mistura de dois tipos de aço e ainda o possível destino da placa. Essa pode ser laminada, vendida como placa ou reaproveitada como sucata, cada destino desses contém um respectivo valor de mercado. Dessa forma foi possível calcular o custo de oportunidade associado à produção da placa de mistura entre os *jobs* i e j , f_{ij}^m . Os elementos e a fórmula são respectivamente:

- A quantidade de mistura gerada (QM);
- O valor de mercado da mistura gerada (VMM);
- O valor de mercado do aço que seria produzido caso não houvesse a mistura (VMA);

$$f_{ij}^m = \text{QM} \times (\text{VMA} - \text{VMM})$$

3.4 Resultados Computacionais dos Modelos Matemáticos

A partir de análises sobre testes preliminares conduzidos com os modelos para sequenciamento em máquinas paralelas, foi possível perceber certa dificuldade dos modelos em resolver instâncias maiores. Outra análise foi possível quanto ao valor da função objetivo, que permite a abordagem do problema como sequenciamento de máquina simples sem prejuízos. Considerando uma instância qualquer I para o problema de sequenciamento tratado, a única diferença entre as soluções ótimas obtidas por um

modelo de máquinas paralelas e por um modelo de máquinas simples seria a realização de um *setup* a menos no modelo de máquinas paralelas. Com isso, na tentativa de possibilitar melhor desempenho dos modelos, para todas as instâncias resolvidas foi considerado $M = 1$.

Com o intuito de verificar o desempenho dos modelos desenvolvidos foram conduzidos testes nas 48 instâncias menores (11 a 24 *jobs*, derivadas das originais, pois os mesmos não resolveram instâncias maiores. Outras informações das instâncias são:

- Limite de variação de largura: 150 mm;
- Espessura das placas: 252 mm;
- Tempo de *setup*: 90 minutos;
- Tempo de lingotamento entre 40 e 44 minutos.

Os testes foram conduzidos em um computador do tipo Intel(R) Xeon(R) CPU X5690@3.47GHz, com 24 processadores e 132 GB de memória RAM. Os modelos foram implementados e resolvidos através do software de otimização CPLEX 12.4, utilizando-se a linguagem AMPL. Foi estabelecido um tempo limite de 7200 segundos para obter a solução ótima da instância. Os resultados serão apresentados em grupos de 12 instâncias para cada um dos dois modelos. Nesta seção são apresentados apenas parte dos resultados, o restante está no Apêndice A.

Nas Tabelas 3.1, 3.2, 3.3, 3.4 e 3.5 fornecem para cada instância o número de *jobs* NL, a solução fornecida pela empresa (SolEmp), obtida pelos programadores com as metas de reduzir o número de *setups* e o número de misturas, além da função objetivo obtida pelo modelo (1 ou 2, dependendo da tabela), do GAP de otimalidade (G%), do tempo de processamento do modelo (TC) e do percentual de redução do modelo em relação à solução da empresa (% red).

Pode-se afirmar que os modelos apresentaram um bom desempenho. O tempo computacional nas instâncias com até 12 *jobs* foi de apenas alguns segundos, extremamente baixo e mesmo nas instâncias maiores, em que o limite de 7.200 segundos foi alcançado, foi possível observar melhora na função objetivo em relação à solução apresentada pela empresa. Para todas as instâncias os modelos apresentaram solução igual ou inferior à solução apresentada pela empresa, com exceção da solução apresentada pelo Modelo 2 para a instância IRM25. O Modelo 1 mostrou leve superioridade no percentual médio de melhora em relação ao Modelo 2, mas ambos com um desvio padrão

Tabela 3.1: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 1 para as instâncias reduzidas de 1 a 12

Modelo 1 - IRM01 a IRM12						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM01	12	6280 (0)	6280 (0)	0,00	0,12	0,00
IRM02	12	6554 (0)	6554 (0)	0,00	1,12	0,00
IRM03	12	27708 (0)	6306 (1)	0,00	0,62	77,24
IRM04	12	27178 (0)	12336 (2)	0,00	2,11	54,61
IRM05	11	37026 (0)	16691 (2)	0,00	0,61	54,92
IRM06	11	46500 (0)	18464 (1)	0,00	0,87	60,29
IRM07	12	50929 (0)	18587 (3)	0,00	20,60	63,50
IRM08	11	20486 (0)	9324 (0)	0,00	2,83	54,49
IRM09	12	15260 (0)	12711 (0)	0,00	0,94	16,70
IRM10	12	6218 (0)	162 (0)	0,00	0,06	97,39
IRM11	12	13020 (0)	9684 (0)	0,00	5,14	25,62
IRM12	12	11051 (0)	6290 (1)	0,00	7,08	43,08

Tabela 3.2: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 2 para as instâncias reduzidas de 1 a 12

Modelo 2 - IRM01 a IRM12						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM01	12	6280 (0)	6280 (0)	0,00	34,27	0,00
IRM02	12	6554 (0)	6554 (0)	0,00	3,52	0,00
IRM03	12	27708 (0)	6306 (1)	0,00	5,98	77,24
IRM04	12	27178 (0)	12336 (2)	0,00	5,18	54,61
IRM05	11	37026 (0)	16691 (3)	0,00	3,17	54,92
IRM06	11	46500 (0)	18464 (1)	0,00	9,67	60,29
IRM07	12	50929 (0)	18587 (3)	0,00	59,53	63,50
IRM08	11	20486 (0)	9324 (0)	0,00	6,55	54,49
IRM09	12	15260 (0)	12711 (0)	0,00	18,34	16,70
IRM10	12	6218 (0)	162 (0)	0,00	1,00	97,39
IRM11	12	13020 (0)	9684 (0)	0,00	38,93	25,62
IRM12	12	11051 (0)	6290 (1)	0,00	5,21	43,08

acima de 20%, e em metade das instâncias o primeiro modelo apresentou melhoras superiores a 50%. O Modelo 2 encontrou a solução ótima em 22 das 48 instâncias, já o Modelo 1 encontrou a solução ótima em 24 instâncias. Não é possível afirmar a partir dos testes que um modelo supera o outro, pois os resultados são bastante semelhantes. De modo que ambos são promissores para o desenvolvimento da pesquisa.

Tabela 3.3: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 1 para as instâncias reduzidas de 37 a 48

Modelo 1 - IRM37 a IRM48						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM37	20	30562 (1)	21680 (3)	25,68	7200,26	29,06
IRM38	23	103968 (1)	40722 (3)	18,32	7200,56	60,83
IRM39	20	99229 (1)	47901 (5)	0,00	10,55	51,73
IRM40	24	32051 (1)	18716 (1)	30,27	7204,57	41,61
IRM41	24	51986 (1)	24399 (3)	45,69	7200,29	53,07
IRM42	20	60527 (1)	25507 (3)	0,00	3856,50	57,86
IRM43	21	20136 (1)	14459 (2)	0,00	4753,47	28,19
IRM44	21	92011 (1)	38020 (4)	2,14	7200,32	58,68
IRM45	19	43386 (1)	23474 (2)	0,00	5130,99	45,89
IRM46	22	41130 (1)	16880 (1)	0,00	407,03	58,96
IRM47	17	23981 (1)	9206 (1)	92,01	7200,25	61,61
IRM48	14	56808 (1)	24683 (3)	0,00	9,69	56,55

Tabela 3.4: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 2 para as instâncias reduzidas de 37 a 48

Modelo 2 - IRM37 a IRM48						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM37	20	30562 (1)	26239 (2)	75,71	7202,28	14,15
IRM38	23	103968 (1)	61729 (2)	80,13	7205,07	40,63
IRM39	20	99229 (1)	47901 (5)	77,20	7200,47	51,73
IRM40	24	32051 (1)	20468 (1)	39,88	7200,69	36,14
IRM41	24	51986 (1)	26051 (3)	52,92	7199,87	49,89
IRM42	20	60527 (1)	25507 (3)	53,71	7200,38	57,86
IRM43	21	20136 (1)	14459 (2)	44,10	7200,38	28,19
IRM44	21	92011 (1)	38020 (4)	83,87	7200,31	58,68
IRM45	19	43386 (1)	23474 (2)	47,04	7200,28	45,89
IRM46	22	41130 (1)	18317 (1)	33,05	7200,34	55,47
IRM47	17	23981 (1)	9206 (1)	0,00	432,76	61,61
IRM48	14	56808 (1)	24683 (3)	0,00	685,82	56,55

Tabela 3.5: Comparação entre o Modelo 1 e o Modelo 2 para testes com as instâncias reduzidas

Modelo	% Médio de Melhora	Desvio Padrão	% Soluções Ótimas	% de Melhoras acima de 50%
MOD 1	42,3	22,9	50,0 (24)	50,0 (24)
MOD 2	40,0	24,7	45,8 (22)	43,8 (21)

Para exemplificar o funcionamento dos modelos, na Figura 3.1 e na Figura 3.2 são exibidos gráficos de Gantt que apresentam a solução dada pela empresa e a solução fornecida pelo modelo para as mesmas instâncias.

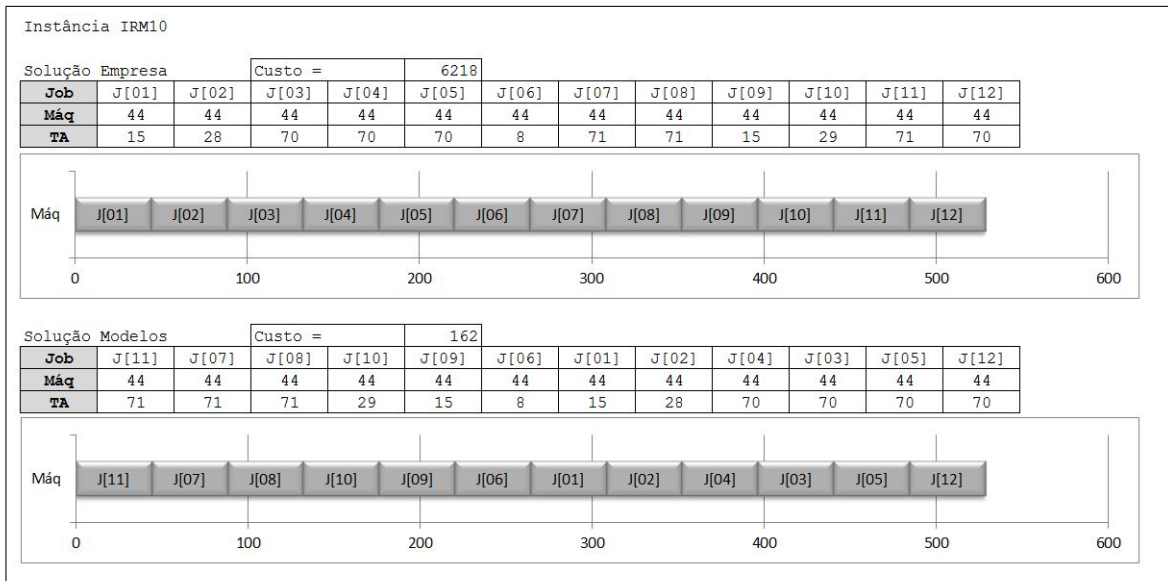


Figura 3.1: Gráficos de Gantt para instância IRM42

Na Figura 3.1, o modelo apresenta uma solução onde uma redução de custo considerável é conferida apenas reordenando os *jobs* na série, sem a necessidade da inserção de um *setup* para evitar a geração de mistura. Já na Figura 3.2 a redução do custo de sequenciamento acontece por meio da realização de mais uma operação de *setup* e com isso evita-se gerar a mistura entre alguns *jobs*.

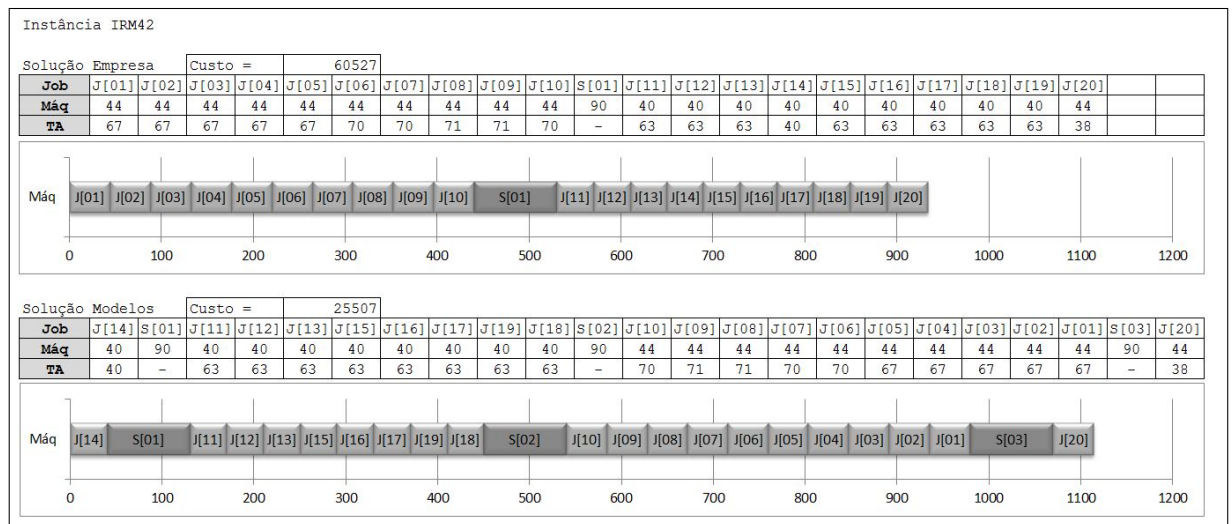


Figura 3.2: Gráficos de Gantt para instância IRM42

Os *jobs* 14 e 20 são separados dos demais, o que implica que a mistura deles com quaisquer outros implicaria em um custo de mistura mais elevado que o custo do *setup* adicionado. Na prática, o resultado relevante para esta instância é o de que os

jobs 14 e 20 não devem ser sequenciados nesse momento, pois implicariam na realização de dois *setups* para a produção de apenas duas painelas. O mais indicado nessa situação seria sequenciar as demais séries e aguardar para que num próximo período hajam *jobs* com grades de aço e/ou larguras mais compatíveis com esses *jobs*.

Capítulo 4

Heurísticas e Espaço de Busca

Apesar do bom desempenho apresentado pelos modelos 1 e 2 para instâncias pequenas, é necessária a busca por outras estratégias para tentar resolver o problema. Em virtude das peculiaridades do processo, é desejável que instâncias maiores sejam resolvidas e os modelos deixam de ser uma alternativa quando são considerados problemas teste de 60 *jobs* devido ao alto tempo de processamento necessário. Além disso, é importante conhecer o espaço de soluções do problema, a distribuição dos mínimos locais e a irregularidade do espaço. Essa análise possibilitará uma escolha mais assertiva do método heurístico para solução do problema. Com isso, este capítulo apresenta uma seção destinada à análise do *landscape* e outra que apresenta os métodos heurísticos desenvolvidos para permitir a resolução de instâncias maiores em tempo hábil.

4.1 Análise de *Landscape*

A análise de *landscape* proposta nessa seção tem o objetivo de fornecer informações importantes sobre os tipos de instâncias dos problemas. Para alcançar essa meta, algumas etapas foram definidas:

- Definição da forma de representação e estruturas de vizinhança;
- Classificação das Instâncias;
- Análise Comparativa das Vizinhanças
 - Número de mínimos locais;

- Distância entre mínimos locais;
 - Análise de Robustez / Suavidade.
-
- Análise de simetria;

4.1.1 Representação e Estruturas de Vizinhaça

Para representar uma solução do problema de sequenciamento foi definida uma representação em forma de um único vetor. Nesse vetor os *jobs* são representados por números inteiros positivos e os *setups* são representados por números inteiros negativos.

$$S = \{5 \ 8 \ 9 \ -1 \ 1 \ 3 \ 2 \ -2 \ 6\}$$

Foram definidas três estruturas de vizinhaça para serem testadas e comparadas. Os algoritmos que apresentam um laço das buscas locais correspondentes às três estruturas estão representados nas Figuras 4.1, 4.2 e 4.3 e posteriormente é feita uma explicação das funções utilizadas nos métodos.

Vizinhaça A: Movimento de Troca Adjacente: Seleciona um *job* (ou *setup*) e troca-o de posição com o *job* da posição seguinte.


```
procedimento VizA
1  para  $i = 1$  até  $i = nv - 1$  faça
2       $s(i) \leftarrow s_0(i + 1)$ ;
3       $s(i + 1) \leftarrow s_0(i)$ ;
4       $s \leftarrow elimina\_setup(s)$ ;
5       $s \leftarrow calcula\_penalidade(s)$ ;
6      se  $(f(s) < f(s_0) \&\& pen(s) = 0)$  então
7           $s_0 \leftarrow s$ ;
8      senão
9           $s \leftarrow s_0$ ;
10     fim-se
11 fim-para
12 Retorne  $s_0$ ;
fim VizA;
```

Figura 4.1: Considerando nv o tamanho do vetor solução, essa busca local possui $n - 1$ vizinhos e a complexidade para varrer todos os vizinhos é $O(n)$. No algoritmo, s é o vetor solução da solução avaliada e s_0 é o vetor da melhor solução encontrada. $s(i)$ indica a posição i do vetor s .

Vizinhança B: Movimento de Troca Simples: Seleciona um *job* (ou *setup*) e troca-o de posição com outro *job* qualquer.

```

procedimento VizB
1  para  $i = 1$  até  $i = nv$  faça
2    para  $j = 1$  até  $j = nv$  faça
3      se  $(i < j)$  então
4         $s(i) \leftarrow s_0(j);$ 
5         $s(j) \leftarrow s_0(i);$ 
6         $s \leftarrow elimina\_setup(s);$ 
7         $s \leftarrow calcula\_penalidade(s);$ 
8        se  $(f(s) < f(s_0) \&\& (pen(s) = 0))$  então
8           $s_0 \leftarrow s;$ 
9        senão
10          $s \leftarrow s_0;$ 
11       fim-se
12     fim-se
13   fim-para
14 fim-para
15 Retorne  $s_0;$ 
fim VizB;

```

Figura 4.2: Considerando nv o tamanho do vetor solução, essa busca local possui $(n - 1)n/2$ vizinhos e a complexidade para varrer todos os vizinhos é $O(n^2)$. No algoritmo, s é o vetor solução da solução avaliada e s_0 é o vetor da melhor solução encontrada. $s(i)$ indica a posição i do vetor s .

Vizinhança C: Movimento de Inserção: Seleciona um *job* (ou *setup*) e insere em outra posição.

```

procedimento VizC
1  para  $i = 1$  até  $i = nv$  faça
2    para  $j = 1$  até  $j = nv$  faça
3      se  $(i \neq j)$  então
4         $s(i) \leftarrow s_0(j);$ 
5         $s \leftarrow \text{redistribui\_vetor}(s);$ 
6         $s \leftarrow \text{elimina\_setup}(s);$ 
7         $s \leftarrow \text{calcula\_penalidade}(s);$ 
8        se  $(f(s) < f(s_0) \&\& (pen(s) = 0))$  então
9           $s_0 \leftarrow s;$ 
10       senão
11          $s \leftarrow s_0;$ 
12       fim-se
13     fim-se
14   fim-para
15 fim-para
16 Retorne  $s_0;$ 
fm VizC;

```

Figura 4.3: Considerando nv o tamanho do vetor solução, essa busca local possui $(n - 1)n - (n - 1)$ vizinhos e a complexidade para varrer todos os vizinhos é $O(n^2)$. No algoritmo, s é o vetor solução da solução avaliada e s_0 é o vetor da melhor solução encontrada. $s(i)$ indica a posição i do vetor s .

Os procedimentos $\text{elimina_setup}(s)$ e $\text{calcula_penalidade}(s)$ são utilizados nos três algoritmos. O primeiro consiste em verificar após o movimento existem *setups* adjacentes, na primeira posição do vetor e/ou na última posição do vetor, caso exista, o *setup* é eliminado e o tamanho do vetor ajustado. O segundo procedimento corresponde a verificar se há penalidades quanto à variação máxima de largura entre *jobs* adjacentes e/ou quanto ao tempo entre dois *setups*. O movimento para o vizinho candidato só é possível caso não haja penalidades $pen(s) = 0$. A função $\text{redistribui_vetor}(s)$ utilizada na busca local C tem o objetivo de redefinir as novas posições dos *jobs* após a inserção.

Para a análise do espaço de busca, foram escolhidas quatro instâncias dentre as instâncias que o Modelo 1 conseguiu fornecer a solução ótima. O critério para a escolha foi selecionar instâncias que representassem bem os tipos de instâncias mais recorrentes para o sequenciamento. Isso implica em dificuldades diferentes quanto à

combinação entre os *jobs*. Apesar de a análise ser desenvolvida apenas sobre instâncias pequenas, que podem não ter significância suficiente para generalizar conclusões sobre o *landscape* do problema, indicações importantes sobre a estrutura podem ser levantadas. As instâncias são apresentadas em ordem crescente de dificuldade nas Tabelas 4.1, 4.2, 4.3 e 4.4.

Tabela 4.1: Informações sobre a instância IRM05, escolhida para a análise do espaço de busca das vizinhanças.

INSTÂNCIA IRM05	
NÚMERO DE <i>JOBS</i>	11
GRADES DE AÇO	4
COMPATIBILIDADE ENTRE LARGURAS	Alta, todos os <i>jobs</i> podem ser combinados
MAIOR DIFERENÇA DE LARGURAS	90 mm
DIFICULDADE DA INSTÂNCIA	Baixa
SOLUÇÃO ÓTIMA (número de <i>setups</i>)	16691(2)

Tabela 4.2: Informações sobre a instância IRM08, escolhida para a análise do espaço de busca das vizinhanças.

INSTÂNCIA IRM08	
NÚMERO DE <i>JOBS</i>	11
GRADES DE AÇO	6
COMPATIBILIDADE ENTRE LARGURAS	Alta, apenas 2 <i>jobs</i> pouco compatíveis
MAIOR DIFERENÇA DE LARGURAS	130 mm
DIFICULDADE DA INSTÂNCIA	Média-baixa
SOLUÇÃO ÓTIMA (número de <i>setups</i>)	9324(0)

As instâncias IRM05 e IRM08 podem ser consideradas as menos complexas do ponto de vista de compatibilidade entre os *jobs*. A primeira possui apenas 4 grades de aço diferentes e a segunda possui 6, que para o número total de *jobs* (11) não é um número tão baixo. No entanto, quando o critério analisado é a largura, ambas possuem várias possibilidades para combinar os *jobs*. Na instância IRM05 a diferença máxima de largura entre os *jobs* é de 90 mm e, como o limite máximo é de 150 mm, quaisquer *jobs* podem estar adjacentes na solução. Já a instância IRM08 possui uma diferença de largura um pouco superior (130 mm), mas também está dentro do limite e com isso permite a adjacência de quaisquer *jobs*.

Tabela 4.3: Informações sobre a instância IRM43, escolhida para a análise do espaço de busca das vizinhanças.

INSTÂNCIA IRM43	
NÚMERO DE <i>JOBS</i>	21
GRADES DE AÇO	6
COMPATIBILIDADE ENTRE LARGURAS	Média-baixa
MAIOR DIFERENÇA DE LARGURAS	495 mm
DIFICULDADE DA INSTÂNCIA	Média
SOLUÇÃO ÓTIMA (número de <i>setups</i>)	14459(2)

Tabela 4.4: Informações sobre a instância IRM39, escolhida para a análise do espaço de busca das vizinhanças.

INSTÂNCIA IRM39	
NÚMERO DE <i>JOBS</i>	20
GRADES DE AÇO	15
COMPATIBILIDADE ENTRE LARGURAS	Baixa
MAIOR DIFERENÇA DE LARGURAS	515 mm
DIFICULDADE DA INSTÂNCIA	Alta
SOLUÇÃO ÓTIMA (número de <i>setups</i>)	47901(5)

As instâncias IRM43 e IRM39 são bem mais complexas que as apresentadas anteriormente (IRM05 e IRM08). A complexidade maior da instância IRM43 está na diferença de largura entre os *jobs*, a maior diferença é de 495 mm, o que impede o posicionamento adjacente de boa parte dos *jobs*. No entanto, a compatibilidade de grades de aço é grande, pois nos 21 *jobs* apenas 6 grades de aço diferentes podem ser observadas. A instância IRM39 foi classificada como o tipo de instância com complexidade mais alta, pois além de uma elevada diferença máxima de largura (515 mm) poucos *jobs* possuem a mesma grade de aço. De modo que podem ser observadas 15 grades diferentes nos 20 *jobs* da instância.

4.1.2 Análise Comparativa das Vizinhanças

Os testes de comparação foram realizados para as quatro instâncias selecionadas e foram planejados para definir o método heurístico mais apropriado para a resolução do problema de sequenciamento. Dois conjuntos de mil soluções são considerados para cada vizinhança, um contendo soluções aleatórias e outro constituído

pelos respectivos mínimos locais encontrados para as soluções aleatórias. Os testes foram gerados utilizando um gerador de soluções aleatório e sementes independentes.

A primeira análise desenvolvida é referente ao erro relativo dos conjuntos de soluções. Na Tabela 4.5 é apresentado o erro médio relativo (RE) nos dois conjuntos de soluções (aleatórias e mínimos locais) para as quatro instâncias. O erro relativo para uma solução é calculado a partir de sua função objetivo: $(f_s - f_{known})/f_{known}$. Onde f_s é a FO da solução avaliada e f_{known} é a melhor solução conhecida, que no caso é a solução ótima do problema.

Tabela 4.5: Para cada instância é apresentado o ótimo global (OT Global) e para cada vizinhança o erro médio relativo para os conjuntos com soluções aleatórias (RE_{aleat}) e com os mínimos locais (RE_{min}), assim como os respectivos intervalos de confiança a um nível de significância de 5%. Para essa análise $RE = (f_s - f_{ot})/f_{ot}$.

ERRO RELATIVO POR TIPO DE INSTÂNCIA						
Instância	OT Global	Vizinhança	RE_{aleat}	IC $RE_{aleat} \pm$	RE_{min}	IC $RE_{min} \pm$
IRM05	16691	VIZA	4,40	0,06	2,77	0,05
		VIZB	4,35	0,06	0,85	0,03
		VIZC	4,36	0,06	0,62	0,02
IRM08	9324	VIZA	10,11	0,21	4,92	0,13
		VIZB	10,22	0,21	1,22	0,08
		VIZC	10,05	0,21	0,12	0,02
IRM43	14459	VIZA	3,17	0,05	2,45	0,05
		VIZB	3,19	0,05	0,68	0,05
		VIZC	3,20	0,05	0,48	0,03
IRM39	47901	VIZA	6,64	0,04	5,90	0,03
		VIZB	6,65	0,04	2,36	0,02
		VIZC	6,66	0,04	1,27	0,02

Com os valores obtidos na Tabela 4.5 já é possível perceber que as vizinhanças B e C alcançam valores menores para o RE_{min} , o que significa que essas vizinhanças possuem valores de função objetivo dos mínimos locais mais próximos da função objetivo da solução ótima.

4.1.2.1 Número de Mínimos Locais

A análise de número de mínimos locais é uma das principais análises sobre o espaço de busca e fornece informações que podem dizer muito sobre o espaço de busca.

Hoos & Stützle [24] afirmam que o número de mínimos locais tem relação direta com a dificuldade de resolução da instância. A Tabela 4.6 contém os dados necessários para a análise.

Tabela 4.6: Comparação entre as vizinhanças quanto ao número de mínimos locais encontrados (Número ML), o número de funções objetivo com valores diferentes (Número FO's) e o número médio de mínimos locais por função objetivo (Razão ML/FO).

NÚMERO DE MÍNIMOS LOCAIS				
Instância	Vizinhança	Número ML	Número FO's	Razão ML/FO
IRM05	VIZ A	1000	765	1,31
IRM05	VIZ B	991	150	6,61
IRM05	VIZ C	948	65	14,59
IRM08	VIZ A	999	974	1,03
IRM08	VIZ B	736	247	2,98
IRM08	VIZ C	126	19	6,63
IRM43	VIZ A	1000	983	1,02
IRM43	VIZ B	1000	938	1,07
IRM43	VIZ C	1000	692	1,45
IRM39	VIZ A	1000	994	1,01
IRM39	VIZ B	1000	986	1,01
IRM39	VIZ C	999	680	1,47

Analisando os resultados é possível perceber que a instância IRM08 pode ser considerada a mais fácil sob o aspecto de número de mínimos locais, pois foi a que apresentou menor quantidade de mínimos locais (Número ML) para as três vizinhanças analisadas. Além disso, a vizinhança C é a que apresenta menor número de mínimos locais em quase todas as instâncias. Na última coluna é apresentado o número médio de mínimos locais para um mesmo valor de função objetivo, ou seja, quantos mínimos locais diferentes possuem o mesmo valor de FO **Média ML/FO**, o que confirma a suspeita quanto à existência de simetria no espaço de soluções do problema.

Foram feitos testes também para a percepção da eficiência dos operadores das vizinhanças. Na Tabela 4.7 são apresentados a solução ótima da instância, a média das funções objetivo dos mínimos locais, o menor mínimo local encontrado e o tempo de busca de cada vizinhança em cada instância.

Tabela 4.7: As comparações são feitas quanto à média da FO dos mínimos locais encontrados (Média ML), quanto ao menor valor de FO encontrado (Menor ML) e quanto ao tempo médio para percorrer todos os vizinhos e finalizar a busca local.

DESEMPENHO DAS BUSCAS LOCAIS					
Instância	Vizinhança	OT Global	Média ML	Menor ML	Tempo Médio Busca
IRM05	VIZ A	16691	62873	22889	0,00
IRM05	VIZ B	16691	30935	16691	0,00
IRM05	VIZ C	16691	26993	16691	0,25
IRM08	VIZ A	9324	55214	9498	0,00
IRM08	VIZ B	9324	20743	9324	0,00
IRM08	VIZ C	9324	10485	9324	0,33
IRM43	VIZ A	14459	99750	59858	0,00
IRM43	VIZ B	14459	48526	14459	0,10
IRM43	VIZ C	14459	32883	14459	0,17
IRM39	VIZ A	47901	165421	99771	0,00
IRM39	VIZ B	47901	80380	53752	0,11
IRM39	VIZ C	47901	70671	48010	2,47

Apesar do tempo de busca superior às demais vizinhanças, a vizinhança C se destaca na análise dos valores de função objetivo dos mínimos locais, a média desses valores foi bastante inferior às demais vizinhanças em todas as instâncias. A vizinhança A não encontrou a solução ótima para nenhuma das instâncias, enquanto as vizinhanças B e C encontraram a função objetivo da solução ótima em 3 das 4 instâncias analisadas.

4.1.2.2 Distância Entre Mínimos Locais

O número de mínimos locais de um espaço de busca pode ser considerado um indicador da extensão do espaço de busca. Quanto mais mínimos locais, mais extenso tende a ser o espaço de busca. No entanto, essa informação pode levar a conclusões erradas no caso de espaços de busca que contenham maior proximidade entre mínimos locais. Quando, por exemplo, são observados *clusters* no espaço de busca, a distância entre os mínimos locais tende a ser menor, mesmo que sejam observados muitos mínimos locais.

Segundo Mattfeld *et al.* [36] a distância média entre um conjunto de soluções reflete a extensão do espaço de busca. Por exemplo, para uma instância de 10 *jobs*, quanto mais próximo de 10 for a distância média entre os mínimos locais, maior é a extensão do espaço de busca e mais movimentos serão necessários para converter uma

solução em outra.

Existem diversas métricas para mensurar a distância entre as soluções, de modo que deve-se escolher uma ou mais que sejam mais apropriadas para o problema. Neste estudo será utilizada uma adaptação da métrica de adjacência utilizada por Reeves [45]. A fórmula é dada por:

$$DIST(s1, s2) = n - n_{adj}$$

Na fórmula, n é o número de *jobs* da instância e n_{adj} é o número de *jobs* que são adjacentes em $s1$ e em $s2$ simultaneamente. A comparação é feita para todos os pares i e j e caso um *job* i preceda j nas duas soluções, $n_{adj} = n_{adj} + 1$. Assim, a distância máxima entre duas soluções é n e a mínima é 0.

A métrica de distância foi utilizada para analisar as distâncias relativas dos mínimos locais e comparar a extensão do espaço de busca das vizinhanças analisadas.

Tabela 4.8: Análise comparativa da distância entre os mínimos locais (Distância Média) nas três vizinhanças para cada instância. Além da Distância Média em cada instância e cada vizinhança são exibidos também os limites (LI e LS) dos intervalos de confiança (IC) a um nível de significância de 5%.

DISTÂNCIA ENTRE MÍNIMOS LOCAIS				
Instância	Vizinhança	Distância Média	IC	
			LI	LS
IRM05	VIZA	9,67	9,67	9,67
	VIZB	8,67	8,66	8,67
	VIZC	8,31	8,31	8,31
IRM08	VIZA	9,60	9,60	9,60
	VIZB	8,54	8,54	8,55
	VIZC	7,22	7,22	7,23
IRM39	VIZA	6,95	6,95	6,95
	VIZB	5,88	5,88	5,89
	VIZC	5,51	5,51	5,52
IRM43	VIZA	13,93	13,92	13,93
	VIZB	15,50	15,50	15,50
	VIZC	15,16	15,15	15,16

Para as instâncias IRM05, IRM08 e IRM39 pode ser observado uma tendência em que a vizinhança A possui o espaço de busca mais extenso e a vizinhança C possui o menos extenso. Essa tendência confirma a análise de número de mínimos locais, na

qual a vizinhança C obteve menor número de mínimos locais. Mas para a instância IRM43, considerada mais complexa, a vizinhança A passa a ter o menos extenso e a vizinhança B se mostra com o espaço de busca com maior extensão. Como o intervalo de confiança é da ordem de 10^{-3} , as diferenças entre as médias são suficientes para a análise.

Comparando as distâncias encontradas para as instâncias, é possível perceber uma diferença de padrão na instância IRM39. Considerando que a distância máxima para duas soluções dessa instância é 20, essa instância apresenta os mínimos locais bem mais próximos do que a instância IRM43 por exemplo, cuja distância máxima possível é de 21.

Utilizando a solução ótima encontrada pelo Modelo 1 como referência foi possível mensurar a distância relativa dos mínimos locais encontrados pelas três vizinhanças. Tal análise foi feita para a instância IRM43 e será exibida nos gráficos das Figuras 4.4, 4.5 e 4.6.

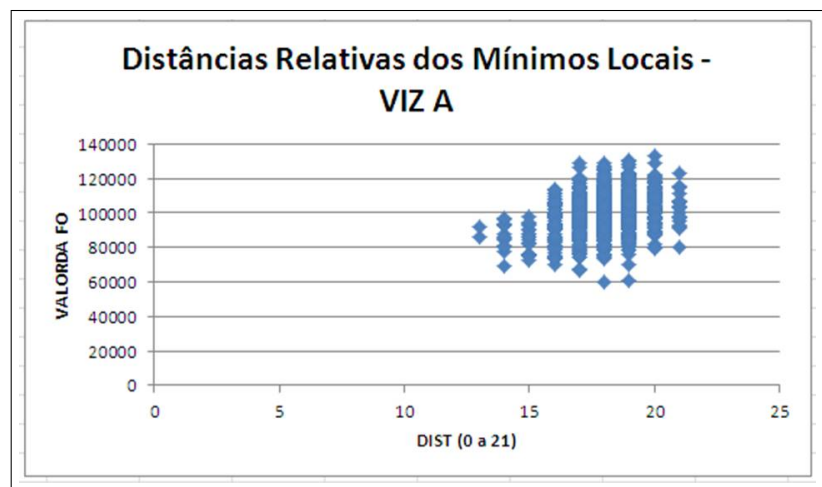


Figura 4.4: Distâncias dos mínimos locais encontrados pelo operador da vizinhança A relativas ao mínimo global.

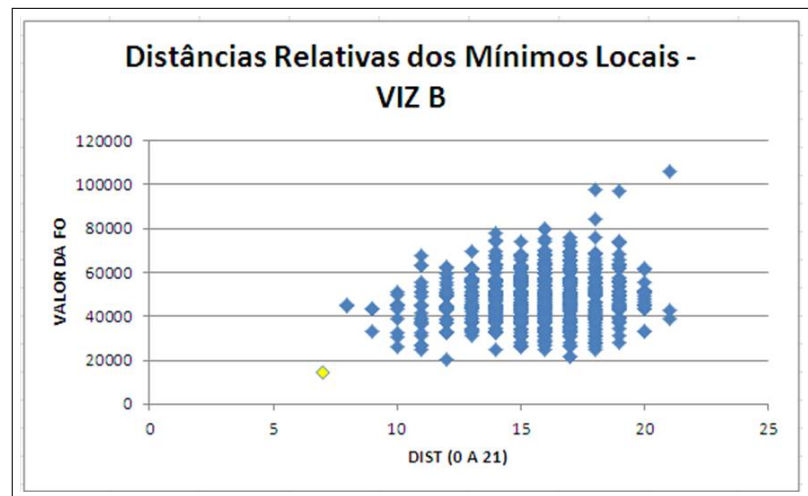


Figura 4.5: Distâncias dos mínimos locais encontrados pelo operador da vizinhança B relativas ao mínimo global. O ponto em amarelo representa um ótimo global, mas não o mesmo usado como referência.

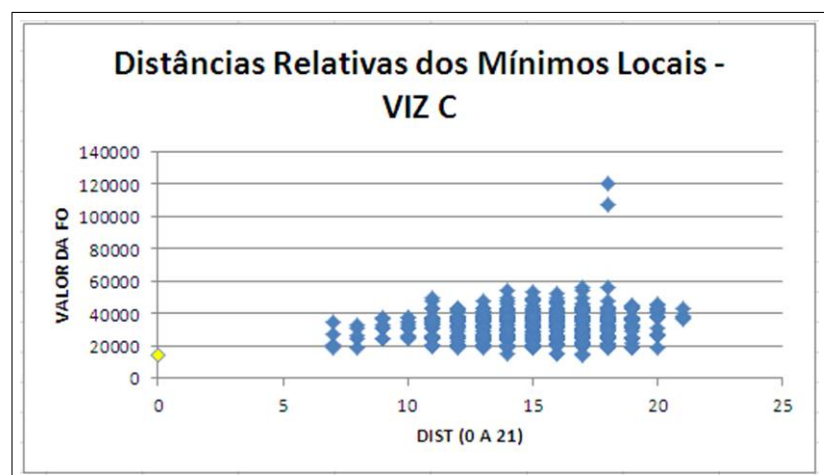


Figura 4.6: Distâncias dos mínimos locais encontrados pelo operador da vizinhança C relativas ao mínimo global. O Ponto em amarelo representa um ótimo global encontrado pela vizinhança, que coincide com o ótimo global usado como referência.

A partir dos gráficos é possível constatar os resultados apresentados anteriormente. A vizinhança C apresentou mínimos locais com valores menores de função objetivo. Já a vizinhança A alcançou valores maiores, mais distantes dos mínimos globais. A vizinhança B obteve uma área de alcance intermediária, chegando a dividir espaço com a vizinhança C nos valores para a FO em torno de 20.000 e com a vizinhança A nos valores para a FO em torno de 60.000.

4.1.2.3 Análise de Robustez

A robustez figura entre os 3 principais tipos de análise do espaço de busca segundo Mattfeld *et al.* [36]. É importante ter informações quanto à irregularidade do espaço de busca, existência de picos, vales, platôs, etc. A distância entre os mínimos locais pode dar indícios para concluir a respeito da irregularidade da superfície do espaço de busca. O fato dos ótimos locais estarem espalhados pelo espaço de busca pode significar a existência um espaço de busca acidentado, mas a análise de robustez/suavidade é mais recomendada para esse fim.

Hoos & Stützle [24] afirmam que o número de mínimos locais pode fornecer informações quanto à robustez do espaço de busca. Sob essa ótica, um espaço de busca pode ser considerado acidentado caso o seu tamanho cresça exponencialmente com o número de ótimos locais. No entanto, como é inviável a determinação do número exato de ótimos locais, métodos de estimativa são utilizados. A análise realizada consiste em determinar o Gradiente Médio de Descida (GMD) e foi feita novamente para as quatro instâncias e para as três vizinhanças definidas. O gradiente médio de descida indica o quanto acidentado é o espaço de busca. Utilizando o erro médio relativo das soluções aleatórias e dos mínimos locais já calculados foi possível calcular o GMD como $(RE_{aleat} - RE_{min})/1$. A comparação está na Tabela 4.9.

Tabela 4.9: Análise comparativa do Gradiente Médio de Descida (GMD) calculado entre as soluções aleatórias e os mínimos locais nas três vizinhanças para cada instância. IC GMD indica o intervalo de confiança a um nível de confiança de 95%.

SUAVIDADE DOS ESPAÇOS DE BUSCA			
Instância	Vizinhança	GMD	IC GMD \pm
IRM05	VIZA	1,63	0,05
	VIZB	3,50	0,03
	VIZC	3,74	0,02
IRM08	VIZA	5,19	0,13
	VIZB	9,00	0,08
	VIZC	9,93	0,02
IRM43	VIZA	0,72	0,05
	VIZB	2,51	0,05
	VIZC	2,73	0,03
IRM39	VIZA	0,74	0,03
	VIZB	4,29	0,02
	VIZC	5,38	0,02

A vizinhança A aparenta ser menos acidentada, o que aliado ao fato dela fornecer valores médios de mínimos locais mais elevados ajuda a confirmar a existência de vales menos profundos, mais rasos. Já a vizinhança C apresenta os valores mais elevados para o gradiente médio de descida. Isso implica na existência de vales mais íngremes no espaço de busca, o que faz sentido se forem analisados os baixos valores médios dos mínimos locais encontrados nessa vizinhança. Tal conclusão já é um indicativo de que estratégias de intensificação são interessantes para a vizinhança C. Mais uma vez a vizinhança B ocupou um patamar intermediário, mas com valores bem próximos da vizinhança C, o que faz com que as mesmas conclusões para o espaço de busca podem ser tiradas para aquela vizinhança. A conclusões a respeito da robustez da vizinhança C já são um indicativo de que estratégias de intensificação são maias interessantes.

4.1.3 Análise de Simetria

A partir dos resultados encontrados é possível comprovar a forte existência de simetria para todas as estruturas de vizinhança. É possível perceber também que o numero médio de mínimos locais para um mesmo valor de função objetivo diminui com o aumento da complexidade das instâncias. A simetria no problema analisado pode ter duas causas diferentes:

- Simetria devido à combinação das séries.

Nesse caso, uma solução simétrica consiste na mesma solução, mas com a(s) série(s) trocadas de lugar. Este tipo de simetria aumenta com o número de séries que a solução apresenta. Caso haja apenas uma série, não haverá solução equivalente, caso haja duas séries (um *setup*, haverão duas soluções de mesma FO. De modo que a fórmula para o número de soluções simétricas é:

$$NE^1 = ns!$$

Onde NE^1 é o número de espelhos da solução para esse tipo de simetria e ns é o número de séries. A figura 4.7 ilustra a existência de soluções simétricas devido à combinação de séries. No caso analisado, $NE^1 = 2! = 2$.

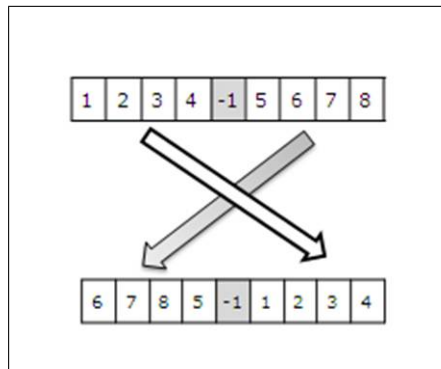


Figura 4.7: Exemplo de situação em que duas soluções são simétricas (mesma FO) devido à possível troca de posição das séries em relação ao *setup*.

- Simetria devido à existência de *jobs* iguais.

Para este tipo de simetria, uma solução simétrica consiste em uma solução que possui como diferença da primeira a troca de posição entre dois *jobs* exatamente iguais (mesmo tipo de aço e mesma largura). Esse tipo de simetria cresce com o número de *jobs* de cada tipo e o número de soluções simétricas pode ser dado por:

$$NE^2 = (T1 \times NT1)! \times (T2 \times NT2)! \times (T3 \times NT3)! \times \dots \times (TP \times NTP)!$$

NE^2 representa o número de soluções simétricas devido à existência de *jobs* iguais. TP é binário e indica a presença ($TP = 1$) ou não ($TP = 0$) do tipo aço e valor de largura na instância e NTP representa o número de *jobs* com as mesmas especificações P (grade de aço e largura). A Figura 4.8 ilustra a existência de soluções simétricas devido à existência de *jobs* com as mesmas características. No exemplo, $NE^2 = (1)(2!)(2!)(1)(1)(1) = 4$.

Por fim, o número de soluções simétricas pode ser dado pela fórmula abaixo e, no exemplo considerado, a instância possui 8 simétricos ($2 \times 4 = 8$).

$$NE^1 \times NE^2$$

A métrica utilizada para mensurar a distância entre as soluções permite que soluções simétricas devido à combinação de séries tenham distância zero entre si, pois a adjacência entre os *jobs* é mantida em ambas. No entanto, soluções simétricas devido à existência de *jobs* iguais apresentam um valor de distância entre si.

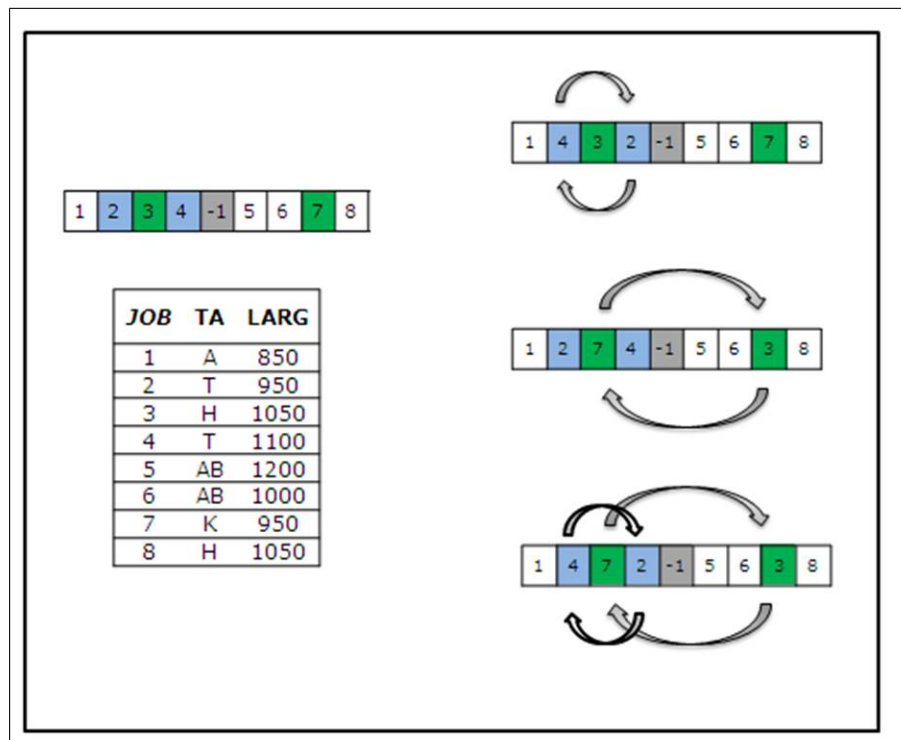


Figura 4.8: Exemplo de situação em que quatro soluções são simétricas (mesma FO) devido à possível troca de posição entre *jobs* que possuem exatamente a mesma especificação.

Para o caso analisado da instância IRM43, será calculado o número de soluções simétricas para a solução ótima. Como são três séries, NE^1 é igual a $3! = 6$. O NE^2 é igual a 32, devido à combinação de *jobs* com grades de aço iguais. Com isso a fórmula fornece um total de 192 mínimos globais. O gráfico 4.9 exhibe os mínimos locais fornecidos pelas três vizinhanças e os ótimos globais. Os pontos tem a distância relativa ao ótimo global fornecido pelo Modelo 1.

4.1.4 Características do Espaço de Busca

Após as análises desenvolvidas nas seções anteriores é possível estabelecer as características mais marcantes dos espaços de soluções das vizinhanças A, B e C.

A vizinhança A apresentou a menor eficácia do ponto de vista de valor da função objetivo. Apesar de ter apresentado um tempo de execução extremamente baixo, essa vizinhança apresentou valores de FO nos mínimos locais extremamente elevados se comparados aos valores alcançados pelas demais vizinhanças. Além disso, esses valores estiveram longe dos ótimos globais em todas as vizinhanças analisadas. O número de mínimos locais também foi muito superior se comparado aos valores

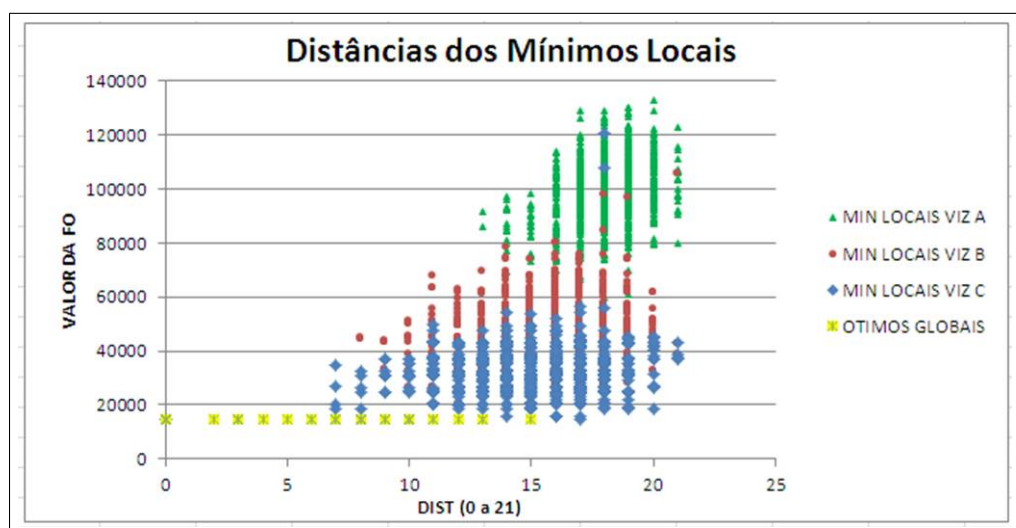


Figura 4.9: Distâncias dos mínimos locais encontrados pelas três vizinhanças e dos ótimos globais relativas ao ótimo global encontrado pelo Modelo 1.

das outras vizinhanças, tendência que não seguiu para a distância entre os mínimos locais, que foi superior em todas as instâncias, mas na instância IRM43 apresentou o menor valor dentre as vizinhanças. Esses resultados sugerem um espaço de busca mais extenso. A outra importante característica que define esse espaço de busca é a suavidade, essa vizinhança apresentou um espaço de soluções muito menos acidentado que as demais, com vales menos acentuados.

Em posição intermediária, a vizinhança B apresentou em todas as análises valores que estão entre os encontrados pela vizinhança A e C e sem apresentar vantagem computacional elevada. Os mínimos locais estão mais próximos entre si e mais próximos dos ótimos globais do que na vizinhança A, com exceção da instância IRM43, quando a vizinhança B apresentou o maior valor para a distância média entre os mínimos locais. A superfície do espaço de busca da vizinhança B é bem mais acidentada do que a superfície da vizinhança A, no entanto mais uma vez ocupa uma posição intermediária.

A vizinhança C obteve os melhores resultados quanto à eficácia. Os valores de função objetivo nos mínimos locais foram os menores dentre as vizinhanças para todas as instâncias analisadas, assim como o número de mínimos locais. A média de mínimos locais por valor de função objetivo foi sempre a maior, sugerindo maior existência de simetria do que as demais vizinhanças. Confirmando a tendência de ser um espaço de busca menos extenso, a distância média entre os mínimos locais foi também a menor dentre as vizinhanças, com exceção da instância IRM43, quando foi a segunda menor. Essa vizinhança apresentou a superfície do espaço de busca mais acidentado, aparentando possuir vales íngremes, o que confirma a tendência apresentada pelos

valores das funções objetivos dos mínimos locais.

A partir dessa análise é possível afirmar que a vizinhança C apresentou resultados mais atrativos e mostrou ser o procedimento de busca local mais promissor dentre os analisados. Como conclusões a respeito da análise de *landscape* e das características desejáveis para um método heurístico para resolução do problema podem ser destacados os seguintes fatores:

- Algumas instâncias com elevado número de mínimos locais;
- Extensão do espaço de busca também variável dependendo da instância;
- Espaço de busca acidentado, com existência de vales profundos;
- Forte presença de simetria em todas as instâncias analisadas;

Com a observação dessas características, é desejável que o método heurístico a ser aplicado contemple essa grande diferença entre as instâncias. A diversificação é um fator fundamental para alcançar as bacias de atração no espaço de busca. Assim como a intensificação é importante para percorrer rapidamente os vales até os mínimos locais. Com isso, é importante que o método permita combinar estratégias de intensificação e diversificação que se adaptem aos diferentes tipos de instâncias.

Outro fator relevante é referente à função objetivo do problema, e os possíveis custos de mistura entre os *jobs*. Como esses custos são relativos e dependem de cada par de *jobs* sucessivos, a busca local pode chegar a um mínimo local que, com uma pequena perturbação no vetor solução chegue a um outro mínimo local próximo, mas de função objetivo ainda menor. Isso promove uma hierarquia de mínimos locais e, com isso, de buscas locais.

Tendo em vista as considerações feitas a meta-heurística escolhida para tratar o problema é a *Iterated Local Search* (ILS) por possibilitar a abrangência da maioria das necessidades comentadas.

4.2 Métodos Heurísticos

A resolução de problemas através de métodos heurísticos é muito utilizada, principalmente quando aplicados a problemas reais. Os principais motivos são o tempo de processamento elevado exigido por métodos exatos e os bons resultados que podem

ser fornecidos pelas diferentes técnicas conhecidas. Assim como o processo estudado neste trabalho, muitas vezes o programador precisa de uma resposta rápida e de boa qualidade, objetivos que podem ser alcançados por métodos heurísticos.

Um método heurístico pode ser definido como um algoritmo que objetiva fornecer boas soluções para um problema em tempo relativamente baixo, mas sem garantias de qualidade. A importância dos métodos heurísticos foi adquirida nas últimas décadas através da utilização desses métodos no tratamento de problemas para os quais não são conhecidos métodos eficazes que forneçam garantias de qualidade. Sua eficiência normalmente é mensurada do bom desempenho médio, o que requer uma bateria de testes significativa.

Os métodos heurísticos utilizados nesse trabalho pertencem a três definições distintas, heurísticas construtivas, heurísticas de busca local e meta-heurísticas.

4.2.1 Heurísticas Construtivas

O objetivo de uma heurística construtiva é construir uma solução para o problema. Essa construção varia de acordo com o método adotado e pode se dar elemento a elemento de acordo com um critério de avaliação, ou ainda como resultado de uma ordenação dos *jobs* de acordo com determinado critério. Esse é o método utilizado por exemplo pela heurística *Earliest Due Date* (EDD), que propõe a organização dos *jobs* crescente da data de entrega para formar a solução. Heurísticas construtivas são normalmente utilizadas para fornecer a solução inicial do problema. Sob essa solução posteriormente é aplicado um método de busca local por exemplo.

Quatro heurísticas construtivas foram desenvolvidas para gerar soluções iniciais para o problema. A primeira delas utiliza como critério primário a grade do *job* e como critério secundário a largura, já o segundo método inverte a prioridade, colocando como primeiro critério a largura. O terceiro método foi desenvolvido utilizando como primeiro critério de agrupamento as grades de aço dos *jobs*, mas como uma avaliação referente a custo para a consolidação das séries. E, por fim, a mesma avaliação em relação a custo foi adicionada ao segundo método para gerar a heurística construtiva 4.

O intuito de organizar os *jobs* de acordo com as grades de aço é colocar em posição adjacente *jobs* que tenham um custo de mistura baixo em função de terem a mesma grade. Já a ordenação por largura tem o objetivo de colocar adjacentes *jobs*

com baixa diferença de largura, além de possibilitar maior extensão das séries, gera menor quantidade de mistura, que também é um componente da função de cálculo do custo de mistura.

A complexidade dos quatro algoritmos é $O(n \log(n))$ e a Figura 4.10 exibe o pseudo-código que representa os quatro métodos.

```

procedimento HeuristicaConstrutiva
1   $l_0 \leftarrow \text{ordenacao}(n, \text{atrib}(n), \text{conj\_crit1});$ 
2   $ns = 1;$ 
3   $q_{ns} = 0;$ 
4  enquanto ( $l_0 \neq \emptyset$ ) faça
5      para  $i = 1$  até  $n$  faça
6          se  $q_{ns} = 0$  então
7               $s_{ns} \leftarrow i;$ 
8               $q_{ns} = q_{ns} + 1;$ 
9          fim-se
10         senão se  $\text{atrib}(i) < \text{conj\_crit2}$  então
11              $s_{ns} \leftarrow i;$ 
12              $q_{ns} = q_{ns} + 1;$ 
13             se  $q_{ns} = \text{lim\_ser}$  então
14                  $ns = ns + 1;$ 
15                  $q_{ns} = 0;$ 
16             fim-se
17         fim-se
28     fim-para;
19      $ns = ns + 1;$ 
20      $q_{ns} = 0;$ 
21 fim-enquanto;
22 Retorne  $s_{ns};$ 
fim HeuristicaConstrutiva;

```

Figura 4.10: No algoritmo, l_0 é a lista inicial gerada a partir da ordenação dos *jobs* segundo o conjunto de critérios de ordenação da heurística *conj_crit1* e os atributos *atrib(n)* de todos os n *jobs* da instância. ns é o número de séries, q_{ns} é a quantidade de *jobs* em cada série e s_j é o conjunto de *jobs* pertencentes à série j . *conj_crit2* é o conjunto de critérios do método para avaliação quanto à entrada do *job* na série.

O pseudo-código na Figura 4.10 representa as quatro heurísticas, o que altera entre os métodos são os conjuntos de critérios de ordenação da lista inicial *conj_crit1* e os conjuntos de critérios de inserção do *job* i na série aberta *conj_crit2*. Em síntese, dois tipos de critério diferentes foram usados para ordenação e dois tipos de critério diferentes foram usados na alocação nas séries, são eles:

- Critérios de Ordenação:

- 1: *Largura* \rightarrow *grade*
- 2: *Grade* \rightarrow *largura*
- Critérios de Alocação:
 - A: $b_{ij}^m > 0$
 - B: $f_{ij}^m < h^m$

Como o parâmetro b_{ij}^m indica a possibilidade de haver adjacência ou não entre dois *jobs* em virtude da variação máxima de largura possível ele é utilizado como o critério de alocação A. O critério de alocação B consiste em verificar se o custo de mistura f_{ij}^m entre dois *jobs* i e j é menor que o custo de *setup*.

Heurística Construtiva 1: A primeira heurística construtiva elaborada tem como principal fator a similaridade entre as grades de aço dos *jobs*, ao organizar com esse critério são agrupados *jobs* que não acarretam em custo de mistura entre si. O outro fator determinante é a largura, pois ao organizar também pela largura é assegurado que os *jobs* de mesma grade com largura mais próxima uma da outra estarão em sequência.

- *conj_crit1*: Organizar os *jobs* em uma lista com ordem crescente de grades de aço como primeiro critério e decrescente de larguras como segundo critério (1).
- *conj_crit2*: Selecionar o primeiro *job* da lista inicial e avaliá-lo quanto à sua inserção na nova lista aberta de acordo com o limite de variação de largura (A).

A Figura 4.11 ilustra o funcionamento da Heurística Construtiva 1.

Heurística Construtiva 2: A segunda heurística construtiva coloca em segundo plano as similaridades entre grades de aço e usa a variação de largura como principal critério. A diferença em relação ao primeiro método está no *conj_crit1*. É esperado que esse método apresente melhores resultados em instâncias com alta similaridade entre grades de aço.

- *conj_crit1*: Organizar os *jobs* em uma lista com ordem decrescente de largura como primeiro critério e crescente de grades de aço como segundo critério (2).
- *conj_crit2*: Selecionar o primeiro *job* da lista inicial e avaliá-lo quanto à sua inserção na nova lista aberta de acordo com o limite de variação de largura(A).

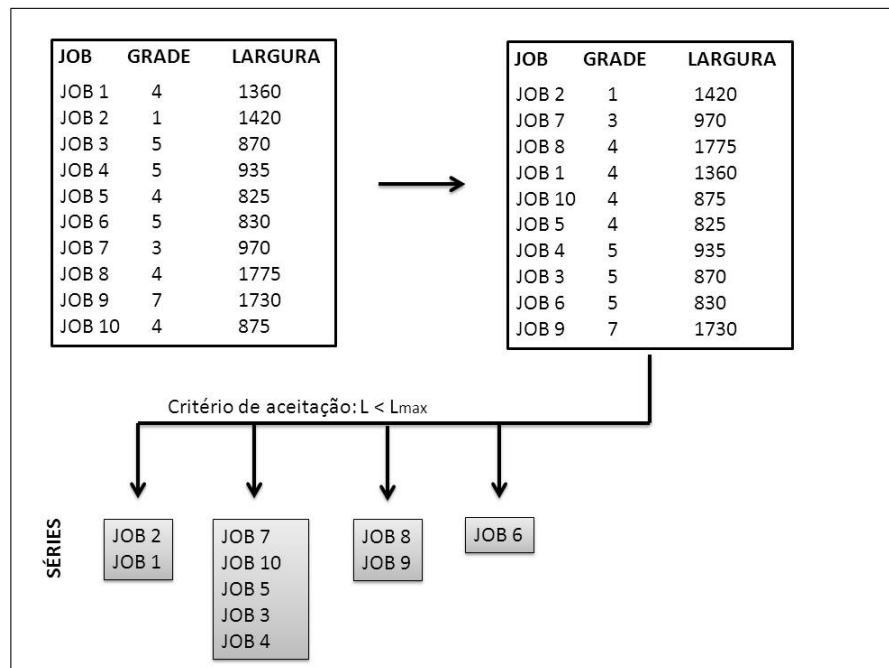


Figura 4.11: Heurística Construtiva 1

Heurística Construtiva 3: A terceira heurística desenvolvida visa abranger os critérios grade de aço e largura, nessa ordem de importância, assim como o primeiro método. No entanto nesse algoritmo há uma avaliação também quanto à custo de mistura entre *jobs* em sequencia. Com isso busca-se diminuir a inclusão de *jobs* que gerem um alto custo de mistura na mesma série.

- *conj_crit1*: Organizar os *jobs* em uma lista com ordem crescente de grades de aço como primeiro critério e decrescente de larguras como segundo critério (1).
- *conj_crit2*: Selecionar o primeiro *job* da lista inicial e avaliá-lo quanto à sua inserção na nova lista aberta de acordo com o limite de variação de largura e com o custo de mistura entre ele e o último *job* da nova lista. Se o custo de mistura for menor que o custo de *setup* a avaliação é positiva e o *job* será inserido na próxima posição da série aberta ($A + B$).

Heurística Construtiva 4: O quarto algoritmo foi desenvolvido sob a mesma estratégia do terceiro e coloca uma avaliação relacionada a custo ao alocar um *job* em uma série. No entanto trata-se de uma derivação da heurística construtiva 2 por considerar a largura como critério primário de organização da lista e a grade como critério secundário.

- *conj_crit1*: Organizar os *jobs* em uma lista com ordem decrescente de largura como primeiro critério e crescente de grades de aço como segundo critério (2).
- *conj_crit2*: Selecionar o primeiro *job* da lista inicial e avaliá-lo quanto à sua inserção na nova lista aberta de acordo com o limite de variação de largura e com o custo de mistura entre ele e o último *job* da nova lista. Se o custo de mistura for menor que o custo de *setup* a avaliação é positiva e o *job* será inserido na próxima posição da série aberta ($A + B$).

A Tabela 4.10 apresenta um resumo da utilização dos critérios.

Tabela 4.10: Resumo de como as heurística construtivas foram montadas. Para cada método são mostrados os critérios de ordenação e alocação utilizados no código.

SÍNTESE HEURÍSTICAS CONSTRUTIVAS		
Método	Ordenação	Alocação
HC 1	1	A
HC 2	2	A
HC 3	1	A + B
HC 4	2	A + B

4.2.1.1 Garantia de Qualidade para as Heurísticas Construtivas

Com o objetivo de entender o desempenho dos algoritmos e medir sua eficácia quanto à capacidade de gerar boas soluções, essa seção é destinada a discutir como responderiam os métodos no pior caso possível. Primeiramente é necessário definir para cada algoritmo o cenário mais desfavorável, que provoque o pior funcionamento do método. Em seguida, deve-se identificar qual seria a resposta dos métodos no pior caso, definindo assim uma garantia de qualidade dos mesmos.

Para isso, duas proposições foram estabelecidas e demonstradas, a primeira delas, proposição P, visa identificar as soluções apresentadas pelos algoritmos HC 1 e HC 2 no pior caso (que é o mesmo para os dois) e a proposição Q faz o mesmo, mas para HC 3 e HC 4, que também compartilham o pior cenário.

Como os métodos HC 1 e HC 2 utilizam ordenações por grade e largura e apenas restringem uma sequência de *jobs* pelo limite de variação de largura, o pior caso para eles seria aquele em que todas as larguras dos N *jobs* seriam compatíveis e todos os *jobs* possuísem grades de aço diferentes. Com isso, o algoritmo não seria capaz de separar os *jobs* e a mistura, que nesse caso possuiria valores muito altos e ocorreria em todas as situações possíveis.

Proposição P: No pior caso, as heurísticas HC 1 e HC 2 fornecem uma solução com o número mínimo de séries possível, $\lceil N/12 \rceil$, e o número máximo de misturas possível.

Demonstração: No pior caso a ordenação da HC 1 resultará em uma lista com os N *jobs* ordenados em ordem crescente de grade e decrescente de largura. Já a HC 2 resultará em uma lista com os N *jobs* ordenados em ordem decrescente de largura e crescente de grade. Como no pior caso a diferença de largura entre os *jobs* sempre será menor que o limite de variação de largura, a matriz do parâmetro b_{ij}^m terá apenas valores 1 para i diferente de j .

$$\bullet b_{ij}^m = \begin{matrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{matrix}$$

Com isso, as verificações para a inserção dos *jobs* nas séries sempre terão valor verdadeiro. E então as séries serão constituídas seguindo exatamente a ordem da lista até completar o tamanho limite da série, caso isso aconteça, o próximo *job* será o primeiro da próxima série e essa seguirá sendo preenchida pelo restante da lista.

A FO será então igual à soma dos custos de mistura dos *jobs* que estiverem subsequentes nas séries acrescido do custo de *setup*, que será o número de séries menos um, ou a função teto do número de *jobs* dividido pelo tamanho máximo de uma série $\lceil N/12 \rceil$. Assim, o valor fornecido pelos métodos pode ser expressado por:

$$\sum_{s \in S} \sum_{i \in N_s} \sum_{j \in N_s} f_{ij} \beta_{ij} + (S - 1)h$$

Onde:

- N é o número de *jobs* da instância;
- S é o número de séries a ser criado e é dado por $\lceil N/12 \rceil$;
- NJ_s é o número de *jobs* na série S e é calculado como o resto da divisão $N/12s$;
- $\beta_{ij} = 1$, se o *job* j for processado imediatamente após do *job* i e 0, caso contrário;

- f_{ij} é o custo da placa de mistura gerada de i para j ;
- h é o custo de um *setup*;

Para os métodos HC 3 e HC 4, o pior caso é diferente, pois os métodos fazem, além da verificação de variação de largura, a verificação quanto ao custo de mistura ($f_{ij}^m < h^m$). Essa condição faz com que esses métodos apresentem um teto para a FO, que será $(N - 1) \times h^m$, onde N é o número de *jobs* e h^m é o custo de realização de um *setup*. No entanto, quando os *jobs* possuem grades de aço diferentes, mas a mistura entre alguns deles não excede o custo de *setup*, o algoritmo falha em encontrar as melhores combinações, pois fica preso à ordenação crescente de grades. Com isso o pior caso passa a ser aquele em que todos os *jobs* possuem larguras compatíveis e grades diferentes, mas com custo de mistura menor do que o custo de *setup*.

Proposição Q: No pior caso, as heurísticas HC 3 e HC 4 fornecem uma solução com FO menor ou igual a $(N - 1) \times h^m$, onde N é o número de *jobs* e h^m é o custo de realização de um *setup*.

Demonstração: Nos métodos HC 3 e HC 4 os critérios para inserção dos *jobs* da lista nas séries são os mesmos:

- $b_{ij}^m > 0$
- $f_{ij}^m < h^m$

Dessa forma, como no pior caso as grades de aço dos N *jobs* são todas diferentes, o segundo critério de inclusão na série pode não ser satisfeito e cada série terá apenas um *job*. Com isso o custo de mistura f_{ij}^m será zero e a função objetivo será dada pelo número de *setups* vezes o custo de *setup*. Essa situação acontece quando o custo de mistura de todos os *jobs* adjacentes na ordenação é superior ao custo de *setup*. No entanto, na matriz de custos dos *jobs* podem haver custos de mistura que são inferiores ao custo de *setup*, mas que devido aos *jobs* não estarem em posição adjacente na ordenação por grades não serão misturados. Com isso, o algoritmo não garante que detectará os menores custos de mistura da matriz e no pior caso terá o valor da função objetivo dado, no máximo, por:

- $FO = (N - 1) \times h^m$

Com as proposições P e Q é possível ter uma garantia de qualidade dos procedimentos heurísticos, ou seja, passa-se a ter um limite máximo para as soluções

fornecidas pelos procedimentos, o que fornece uma segurança maior na fase de testes. Em seu pior caso, os métodos HC 1 e HC 2 fornecem

4.2.2 Buscas Locais

As heurísticas de busca local (ou heurísticas de refinamento) são métodos baseados no conceito de vizinhança, que é constituída por soluções do problema. Um operador da vizinhança corresponde ao tipo de movimento que uma solução pode fazer, em outras palavras o operador define as únicas possibilidades de alteração que uma solução pertencente à vizinhança pode sofrer. A vizinhança de uma solução S é o conjunto de todas as soluções S_n que diferem de S por um movimento.

O ponto de partida para uma heurística de busca local é uma solução inicial, que pode ser obtida aleatoriamente ou através de uma heurística construtiva por exemplo. A cada iteração o algoritmo altera a solução vigente chegando a um vizinho. Alguns métodos de busca local conhecidos são os métodos Primeiro Aprimorante e Melhor aprimorante. No primeiro, assim que é encontrado um vizinho com FO melhor do que a solução vigente esse vizinho passa a ser a solução vigente. Já no segundo método, todos os vizinhos são verificados e a solução vigente passa a ser o vizinho com a melhor FO.

Os procedimentos de busca local considerados para análise são definidos a partir das vizinhanças definidas na seção 4.1. A busca local considerada é do tipo primeiro aprimorante, que consiste em, a partir de uma solução inicial, percorrer os vizinhos até encontrar uma solução melhor. Quando é encontrada uma solução melhor, a solução vigente passa a ser o vizinho de menor função objetivo e a verificação dos vizinhos recomeça. A busca local finaliza quando todos os vizinhos da solução corrente são percorridos e não é encontrado nenhum com valor de FO menor.

As análises apresentadas na seção 4.1 contribuíram para conhecer as características das vizinhanças quanto à eficiência e propriedades do espaço de busca, no entanto, não foi analisada a dominância entre as vizinhanças. A abrangência de cada método e a diferença entre eles influenciam a avaliação avaliar de qual método deve ser considerado para compor um algoritmo mais poderoso, como uma meta-heurística.

É possível perceber através de um pequeno exemplo a dominância entre as vizinhanças. A Figura 4.12 ilustra a comparação entre os vizinhos gerados pelas três estruturas de vizinhança para um mesmo vetor solução.

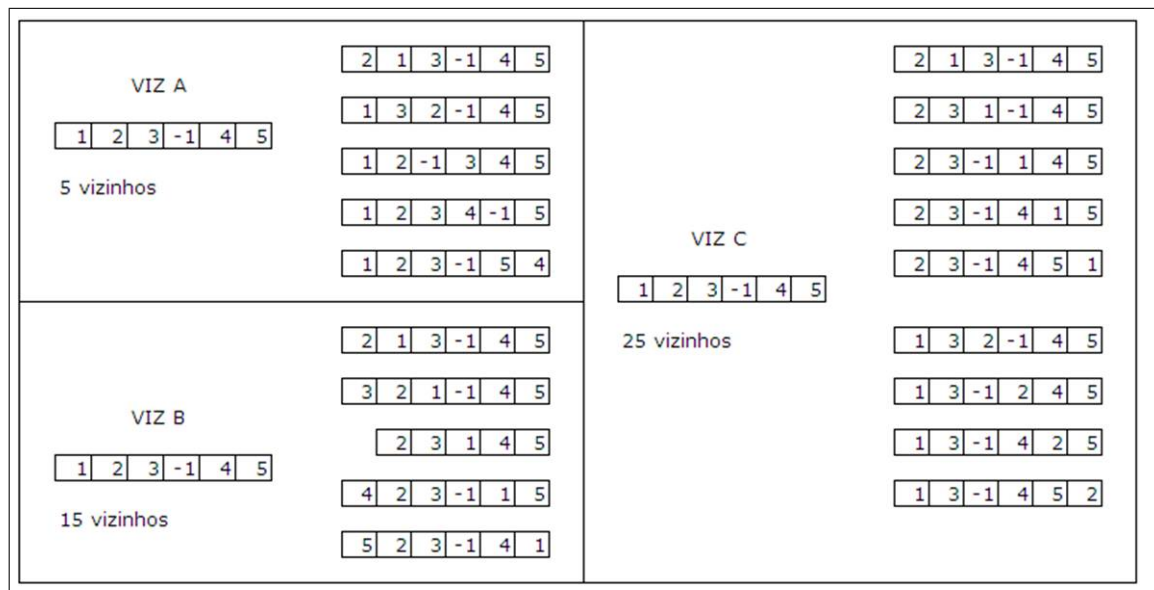


Figura 4.12: Comparação entre as vizinhanças

Para o vetor solução $S = \{1 \ 2 \ 3 \ -1 \ 4 \ 5\}$ a vizinhança A possui 5 vizinhos, enquanto as vizinhanças B e C possuem 15 e 25 vizinhos respectivamente. Considerando a solução S , são 6 vetores na vizinhança A, 16 vetores na vizinhança B e 26 vetores na vizinhança C. A análise de dominância consiste em determinar quantos desses vetores são comuns a duas ou até três das vizinhanças. A tabela 4.11 exibe esses valores.

Tabela 4.11: São exibidos para cada vizinhança o total de vizinhos (Tot Viz.), o número de vizinhos em comum das três vizinhanças (Num VC), a porcentagem de vizinhos compartilhados entre as três vizinhanças (% VC), o número de vizinhos exclusivos de cada vizinhança (VExcl.) e a porcentagem de vizinhos exclusivos em relação ao total de vizinhos (% VExcl.).

DOMINÂNCIA ENTRE AS VIZINHANÇAS DAS BUSCAS LOCAIS							
VIZ	Tot. Viz.	Num VC	% VC	VExcl. A	VExcl. B	VExcl. C	% VExcl.
A	6	6	100%	0	-	-	0%
B	16	6	38%	-	10	-	63%
C	26	6	23%	-	-	20	77%

Com a análise de dominância é possível perceber que a vizinhança A é totalmente dominada pelas vizinhanças B e C, com 100% dos seus vizinhos sendo compartilhados pelas outras vizinhanças. As vizinhanças B e C possuem respectivamente 63% e 77% dos seus vizinhos compartilhados com as demais vizinhanças. Os 14% a mais na exclusividade de vizinhos da vizinhança C são justificados pelo maior número de vizinhos dessa vizinhança em relação às demais. Não foram observadas relações de dominância entre pares de vizinhanças, o compartilhamento de vizinhos foi observado

apenas entre as três vizinhanças simultaneamente, de modo que entre as vizinhanças B e C não pode ser observada relação de dominância. Assim, a vizinhança A pode ser considerada a menos atrativa por ser totalmente dominada e a vizinhança C pode ser considerada a mais abrangente por possuir mais vizinhos exclusivos. A vizinhança B também pode ser considerada abrangente, mas em menor escala.

4.2.3 *Iterated Local Search*

A meta-heurística *Iterated Local Search* está na categoria de meta-heurísticas baseadas em busca local, em que a exploração do espaço de busca é realizada por meio de movimentos sobre a solução vigente para alcançar soluções mais promissoras.

O conceito do método ILS é o de fornecer soluções de partida para a busca local mais promissoras para alcançar ótimos locais melhores e/ou inexplorados. Novas soluções de partida são geradas a partir de uma perturbação no ótimo local vigente a busca local é aplicada até chegar ao mínimo local. O mínimo local avaliado passa a ser o ponto de partida para a próxima perturbação caso esteja dentro do critério de aceitação definido. A intensificação do algoritmo é proporcionada pela busca local e sua eficácia depende da vizinhança utilizada. A perturbação do algoritmo é responsável pela capacidade de diversificação do método e precisa ser bem dosada, pois deve ser forte o suficiente para fazer com que a busca local explore diferentes soluções e fraca o suficiente para evitar que haja um reinício aleatório.

O sucesso da meta-heurística ILS depende principalmente da escolha da busca local, das perturbações e do critério de aceitação e o método é recomendado principalmente para situações as características das instâncias variam muito, para problemas dinâmicos em que os dados variam durante a resolução do problema, para problemas multi objetivo e também para problemas em que a maior parte das meta-heurísticas falham.

Na seção 4.1.4 foram estabelecidas algumas características desejáveis para o método heurístico a ser desenvolvido para resolver o problema de sequenciamento e, de acordo com a descrição da meta-heurística ILS, o método está alinhado com tais características.

O método ILS proposto utiliza como solução inicial a solução fornecida por uma das heurísticas construtivas desenvolvidas e a busca local consiste em percorrer uma das buscas locais definidas na seção 4.1. Foi necessário definir alguns parâmetros

para a ILS aplicada, são eles:

- NP: Número de perturbações aplicadas ao ótimo local corrente;
- N: Número máximo de *jobs* deslocados em uma perturbação;
- K: Fator em porcentagem aplicado sobre a FO que indica o critério de aceitação;
- NIT: Número de iterações;

Para minimizar o efeito da simetria, na busca local foi inserida uma verificação que impede que sejam considerados movimentos entre soluções vizinhas que se diferenciem apenas por ter *jobs* com as mesmas especificações (grade e largura) em posições diferentes. Na Fig. 4.13 é apresentado o método ILS desenvolvido e posteriormente é explicado o funcionamento do mesmo.

```

procedimento ILS
1   $s_0 \leftarrow Heuristica\_Construtiva(n)$ ;
2   $s \leftarrow Busca\_Local(nv, s_0)$ ;
3  enquanto ( $num\_it < NIT$ ) faça
3    para  $n\_pert = 1$  até ( $n\_pert = NP$ ) faça
5       $s' \leftarrow Perturbacao(nv, s, N)$ ;
6       $s'' \leftarrow Busca\_Local(nv', s')$ ;
7      se ( $f(s'') < f(s_{ot}) \times K$ ) então
8         $s \leftarrow s''$ ;
9         $f(s) \leftarrow f(s'')$ ;
7      se ( $f(s'') < f(s_{ot})$ ) então
8         $s_{ot} \leftarrow s''$ ;
9         $f(s_{ot}) \leftarrow f(s'')$ ;
10     fim-se
10     fim-se
11     fim-para
12 fim-enquanto;
13 Retorne  $s_{ot}$ ;
fim ILS;

```

Figura 4.13: No algoritmo, s_0 é a solução inicial gerada após aplicação da heurística construtiva sobre os n *jobs* e s é o vetor solução resultado da aplicação da busca local sobre s_0 . nv é o tamanho do vetor s e os parâmetros K, NIT, N e NP são os mesmos calibrados para a meta-heurística.

Após a aplicação da heurística construtiva e da busca local sobre a solução encontrada, ocorre a perturbação, que consiste em selecionar aleatoriamente um bloco de *jobs* do vetor solução (número de *jobs* também aleatório e menor que N) e criar

uma nova série com esses *jobs*. Um *setup* é inserido no fim do vetor e após o *setup* são inseridos os *jobs* deslocados. Esse procedimento de perturbação acontece NP vezes. Após a etapa de perturbação é aplicada a busca local e então acontece a verificação. Caso a FO da solução avaliada seja menor do que K vezes a FO da melhor solução encontrada até o momento, a solução corrente se transforma na solução avaliada. Caso a FO da solução avaliada seja também menor do que a FO da melhor solução encontrada até o momento, a melhor solução encontrada se transforma na solução avaliada.

4.2.3.1 Calibração dos parâmetros para o método ILS

Para garantir a melhor configuração para a meta-heurística desenvolvida foi realizada uma etapa de calibração dos parâmetros utilizados. Primeiramente houve a tentativa de identificar os valores mais promissores para os parâmetros NP , N e K e posteriormente houve a calibração do parâmetro NIT . Para chegar nos melhores valores para os parâmetros foram planejados experimentos com as 40 instâncias artificiais, sendo quatro grupos de dez instâncias de 25, 50, 75 e 100 *jobs*. Para cada um dos parâmetros avaliados foram definidos possíveis valores (Tabela 4.12) e para cada combinação entre esses valores.

Tabela 4.12: Valores testados para cada parâmetro nos testes de calibração.

CALIBRAÇÃO PARÂMETROS ILS	
Parâmetro	Valores Testados
NP	1 e 2
N	4, 6 e 8
K	1.05, 1.10 e 1.20

Para a análise, a variável considerada como resposta foi o desvio percentual relativo (DPR), calculado da seguinte forma:

- $DPR = 100(f_{sol} - f_{best})/f_{best}$

Onde f_{sol} é a FO da solução analisada e f_{best} é a melhor solução conhecida para a instância.

A partir da análise de variância desenvolvida é possível concluir que não há diferença estatística significativa entre os diferentes valores dos parâmetros a um nível de confiança de 95%, o que sugere que a heurística não é sensível a essas variações de valores. A Tabela 5.20 apresenta as estatísticas da análise de variância e encontra-se no

apêndice C. De qualquer forma, foi realizada uma análise dos intervalos de confiança para os valores avaliados de cada parâmetro. A Figura 4.14 exibe a análise comparativa para o DPR para os parâmetros K e N.

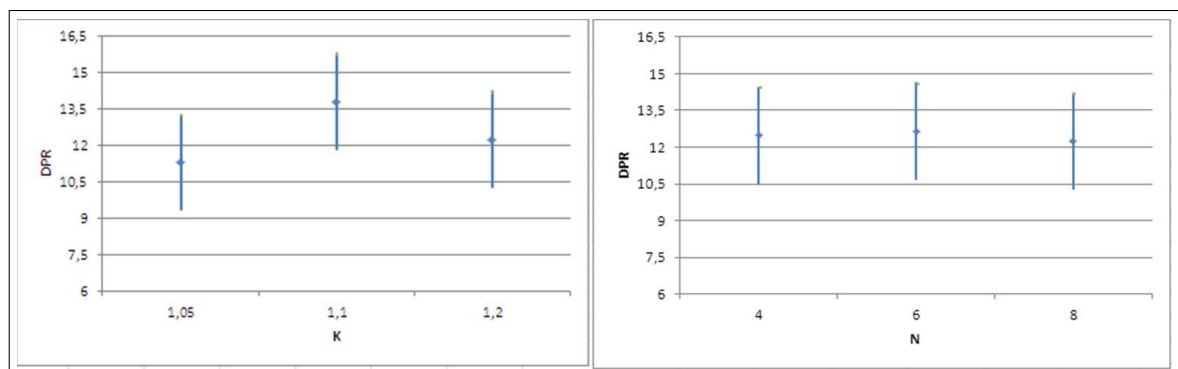


Figura 4.14: Comparação do DPR para os valores testados dos parâmetros K e N.

A partir dos gráficos é possível perceber a proximidade apresentada entre os intervalos de confiança. Para o parâmetro K o valor 1.05 (ou 5%) possui média levemente inferior aos demais valores, mas para o parâmetro N a diferença é quase imperceptível. A Figura 4.15 exibe a mesma comparação, mas para o parâmetro NP.

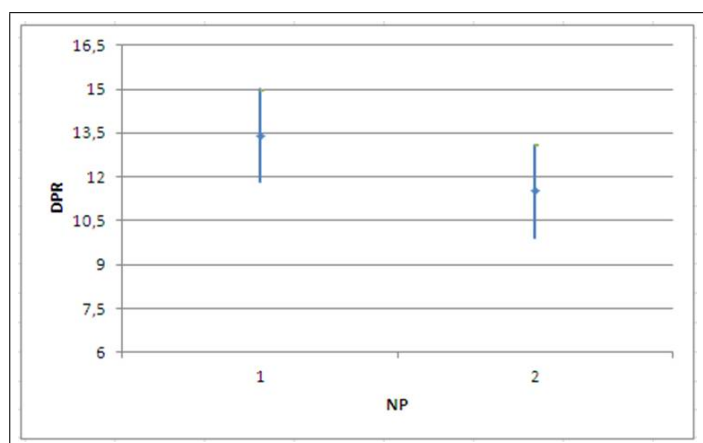


Figura 4.15: Comparação do DPR para os valores testados do parâmetro NP.

Considerando a análise do parâmetro NP, é observada uma leve superioridade nos valores encontrados quando o número de perturbações é fixado em 2, o que sugere a utilização do valor 2. No entanto, também não é suficiente para afirmar uma diferença estatística significativa entre esses valores. Assim, foram estabelecidos para os parâmetros K, N e NP os valores 5%, 6 e 2, respectivamente.

Para calibrar o parâmetro NIT foram realizados testes com seis instâncias escolhidas ao acaso dentre as instâncias artificiais já utilizando os valores para os parâmetros P, NP e K estabelecidos. Os testes consistiram em registrar a cada iteração do ILS a melhor função objetivo encontrada. Os resultados são exibidos no gráfico da Figura 4.16.

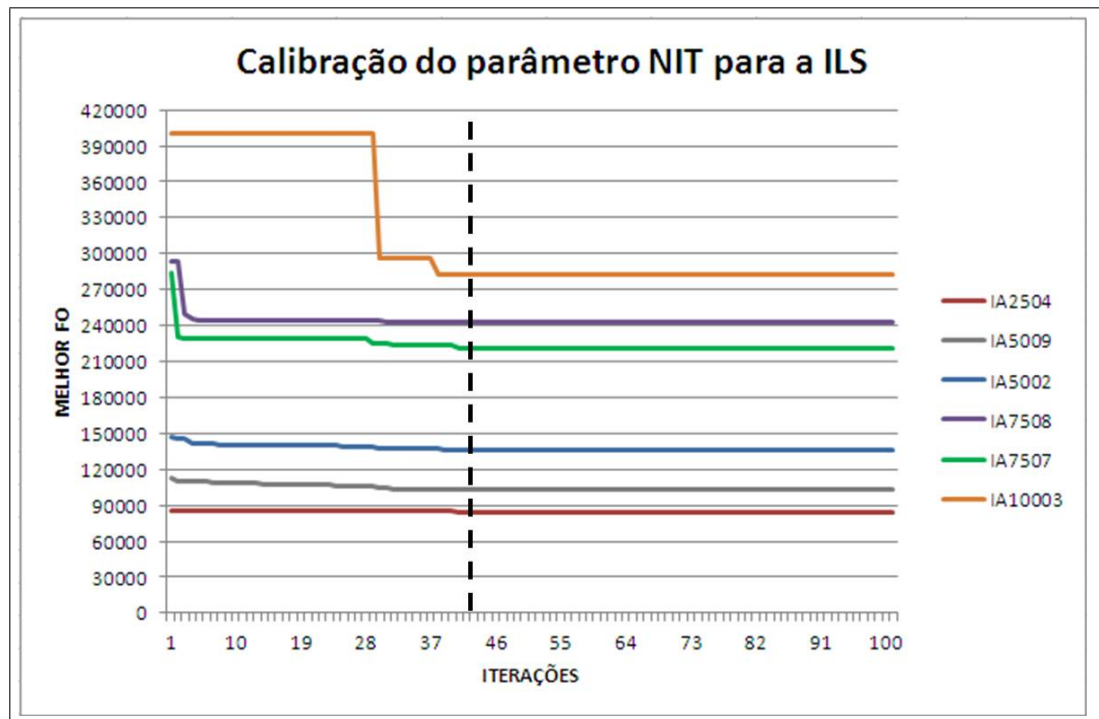


Figura 4.16: Evolução do valor do melhor ótimo local encontrado pelo ILS para seis instâncias artificiais ao longo das iterações.

Para os testes foram aplicadas 100 iterações. A instância que teve a melhor FO em 100 iterações encontrada mais cedo foi a instância IA7508, que encontrou na iteração 30, e a instância que demorou mais para encontrar o menor valor para FO foi a instância IA7507 (iteração 44). Com isso, entende-se que um valor razoável para o número de iterações seja 50.

4.3 Geração de Instâncias Artificiais

Com o objetivo de avaliar as heurísticas desenvolvidas foram geradas instâncias artificiais de 25 a 100 *jobs* totalmente independentes das instâncias reais.

Para obter as instâncias artificiais foi utilizado um gerador de instâncias desenvolvido na linguagem C++. A maioria dos parâmetros de entrada são gerados

randomicamente a partir de uma distribuição uniforme com limite inferior e superior especificados, no entanto, alguns parâmetros como o custo de mistura precisaram seguir a lógica de similaridade para a geração da matriz de custo.

Tabela 4.13: Relação dos parâmetros e o método de geração dos mesmos.

Parâmetros para Instâncias Artificiais	
PARÂMETRO	DISTRIBUIÇÃO
Tempo de Setup	U(50,150)
Limite Variação de Largura	U(100,200)
Grade de Aço	U(1,236)
Número de <i>jobs</i> de Mesma Grade	U(1,12)
Largura	U(800,2000)
Tempo de Processamento	U(30,50)
Custo de Mistura - <i>Jobs</i> de mesma grade	U(0,500)
Custo de Mistura - <i>Jobs</i> de grades diferentes	U(500,30000)

Apesar de ter sido usada a quantidade real de grades de aço diferentes (236), o custo de mistura atrelado a cada grade não foi o real com o intuito de permitir a geração de diferentes combinações de valores para testar as heurísticas. No entanto, mesmo utilizando distribuições uniformes para gerar o custo, a afinidade entre *jobs* de mesma grade deve ser mantida para não comprometer o funcionamento das heurísticas, caso contrário os métodos perderiam o sentido.

4.4 Resultados Computacionais dos Métodos Heurísticos

Esta seção está dividida em duas subseções. Na primeira são apresentados os resultados para os testes de eficiência dos quatro métodos construtivos. Já na segunda subseção são apresentados os resultados dos testes para o ILS.

4.4.1 Resultados dos Testes das Heurísticas Construtivas

O planejamento de experimentos para as heurísticas construtivas ocorreu da seguinte forma. Primeiro buscou-se observar o comportamento das heurísticas e compará-las entre si. Para isso, foram geradas instâncias de 50 e 100 *jobs*, sendo 10 de cada tipo. Os algoritmos foram então testados para essas instâncias e foi observado seu tempo computacional e a função objetivo. Em seguida, para testar a eficácia dos procedimentos, foram realizados testes com as quatro heurísticas para as 48 instâncias

menores do problema, das quais já se conhece os resultados dos modelos 1 e 2.

Tabela 4.14: Comparação de FO e tempo computacional obtidos pelas quatro heurísticas para as instâncias artificiais de 50 *jobs*

Instâncias Artificiais de 50 <i>jobs</i>									
INST	NJ	FO HC1	TC HC1	FO HC2	TC HC2	FO HC3	TC HC3	FO HC4	TC HC4
IA5001	50	680565	0.360	770991	1.313	220048	1.953	227991	1.609
IA5002	50	531356	0.766	632701	1.204	194440	1.719	181449	1.593
IA5003	50	526586	0.875	569255	1.203	202335	1.860	191512	1.703
IA5004	50	580243	0.922	766934	1.219	187404	1.750	199265	1.578
IA5005	50	653440	0.391	641978	1.329	212490	1.875	214675	1.515
IA5006	50	588258	0.844	665024	1.204	229571	1.859	203393	1.672
IA5007	50	542926	0.875	685076	1.328	206505	0.750	209659	1.578
IA5008	50	649523	0.937	803179	1.282	216352	1.859	203470	0.703
IA5009	50	609702	0.828	735709	1.406	214107	1.860	193808	1.656
IA5010	50	594506	0.906	649596	1.297	199288	1.796	204948	1.547
MÉDIA		595710.5	0.8	692044.3	1.278	208254	1.107	203017	1.606

Tabela 4.15: Comparação de FO e tempo computacional obtidos pelas quatro heurísticas para as instâncias artificiais de 100 *jobs*

Instâncias Artificiais de 100 <i>jobs</i>									
INST	NJ	FO HC1	TC HC1	FO HC2	TC HC2	FO HC3	TC HC3	FO HC4	TC HC4
IA10001	100	1481097	2.593	1458096	4.672	392859	6.734	391563	6.296
IA10002	100	1419870	2.641	1440077	4.531	428047	6.718	392510	6.171
IA10003	100	1464558	2.844	1467776	4.781	416987	6.688	399761	6.078
IA10004	100	1522361	2.796	1572714	4.485	377827	6.671	363492	6.109
IA10005	100	1349394	2.813	1522348	4.609	371232	6.672	384723	6.094
IA10006	100	1584800	2.719	1526926	4.562	411460	6.593	358350	6.015
IA10007	100	1630601	2.796	1569300	4.828	400701	6.625	352532	5.968
IA10008	100	1499453	2.953	1493723	4.531	443445	6.765	381495	6.312
IA10009	100	1639423	2.796	1592003	4.672	412895	6.953	399135	6.031
IA10010	100	1629145	2.781	1685617	4.515	405193	6.657	403267	6.125
MÉDIA		1522070.2	2.773	1532858	4.619	406064.6	6.707	382682.8	6.120

Conforme verificado nos testes, a HC 4 apresentou desempenho superior às demais para os dois conjuntos de instâncias. A média da FO encontrada para esse método foi inferior, no entanto a HC 3 por vezes apresenta a menor FO dentre os quatro métodos. Quanto ao tempo computacional, a HC 1 apresentou os menores valores, apesar de que nenhuma delas apresentou valores com ordem de grandeza tal que inviabilize sua aplicação.

A segunda etapa dos testes dos métodos heurísticos visou mensurar o desempenho dos mesmos para instâncias reais de até 24 *jobs*. Como esse foi o tamanho limite de instância que os modelos conseguiram resolver é através dessas instâncias que será possível identificar a eficácia dos algoritmos. As tabelas contendo os resultados apresentam a função objetivo encontrada por cada procedimento e a diferença, em

porcentagem, que esses resultados estão da função objetivo encontrada pelo Modelo 1 (modelo de melhor desempenho). Quanto mais distante de 100% estiver esse valor, mais distante da FO encontrada pelo modelo a solução da heurística estará. Nos casos em que esse valor é de 100%, a heurística encontrou o mesmo valor do modelo. Nas ocasiões em que a FO do modelo é ótima foi inserido um "OT" ao lado do valor e nas situações em que esse valor não é ótimo foi inserido um "MS", identificando o valor como a melhor solução encontrada. Nesta seção são apresentados os resultados para apenas metade das instâncias, o restante encontra-se no Apêndice B.

Tabela 4.16: Comparação entre as soluções encontradas pelo Modelo 1 e pelos métodos HC 1 e HC 2 para as instâncias reduzidas de 1 a 12

Heurísticas Construtivas - IRM01 a IRM12						
INST	NJ	FO Mod 1	HC1	D(%)	HC2	D(%)
IRM01	12	6280 (OT)	19204	205,8	25746	309,97
IRM02	12	6554 (OT)	11440	74,55	6930	5,74
IRM03	12	6306 (OT)	32451	414,61	75630	1099,33
IRM04	12	12336 (OT)	18242	47,88	27914	126,28
IRM05	11	16691 (OT)	42324	153,57	31485	88,63
IRM06	11	18464 (OT)	32755	77,4	32968	78,55
IRM07	12	18587 (OT)	86907	367,57	91578	392,7
IRM08	11	9324 (OT)	20623	121,18	75032	704,72
IRM09	12	12711 (OT)	14343	12,84	18767	47,64
IRM10	12	162 (OT)	1883	1062,35	7701	4653,7
IRM11	12	9684 (OT)	16548	70,88	36652	278,48
IRM12	12	6290 (OT)	17664	180,83	11558	83,75

Os resultados para os métodos HC1 e HC2 mostram uma oscilação na assertividade do método, chegando a apresentar soluções com diferenças de FO para o modelo 1 que vão de 13% a 94%. Para os métodos HC3 e HC4 é esperada uma regularidade maior, pois o critério de custo para inserção do *job* na série limita a realização de misturas com um custo muito elevado.

Como era esperado, os métodos HC3 e HC4 apresentaram um desempenho melhor e, além disso, apresentaram soluções próximas das soluções encontradas pelo Modelo 1 em algumas situações. Em geral, a Heurística Construtiva 4 foi superior, encontrando a solução ótima em três situações (IRM04, IRM07 e IRM12), enquanto a HC3 encontrou a solução ótima em duas (IRM04 e IRM07). É possível observar também que os métodos alternam quanto à maior proximidade da solução encontrada pelo Modelo 1, o que sugere o desenvolvimento de um algoritmo com os dois métodos combinados.

Apesar de apresentarem soluções piores que as dos modelos na maioria das

Tabela 4.17: Comparação entre as soluções encontradas pelo Modelo 1 e pelos métodos HC 3 e HC 4 para as instâncias reduzidas de 1 a 12

Heurísticas Construtivas - IRM01 a IRM12							
INST	NJ	FO Mod 1	HC3	D(%)	HC4	D(%)	
IRM01	12	6280 (OT)	18412	193,18	6525	3,9	
IRM02	12	6554 (OT)	11440	74,55	6930	5,74	
IRM03	12	6306 (OT)	12439	97,26	6307	0,02	
IRM04	12	12336 (OT)	12336	0	12336	0	
IRM05	11	16691 (OT)	16906	1,29	18279	9,51	
IRM06	11	18464 (OT)	23165	25,46	24648	33,49	
IRM07	12	18587 (OT)	18587	0	18587	0	
IRM08	11	9324 (OT)	12389	32,87	17767	90,55	
IRM09	12	12711 (OT)	14343	12,84	18767	47,64	
IRM10	12	162 (OT)	1883	1062,35	7701	4653,7	
IRM11	12	9684 (OT)	16548	70,88	36652	278,48	
IRM12	12	6290 (OT)	12396	97,07	6290	0	

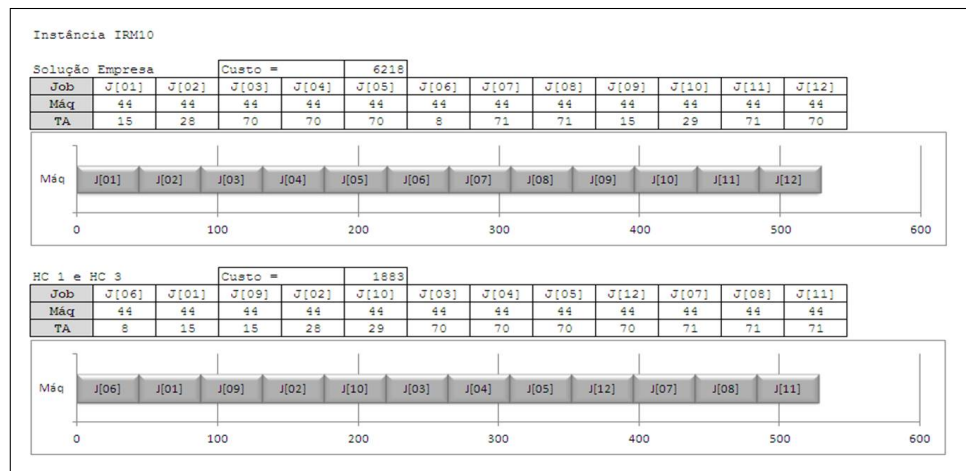


Figura 4.17: Heurística Construtiva 3 comparada à solução da empresa

soluções, as heurísticas construtivas apresentaram soluções melhores que as soluções da empresa, as Figuras 4.17 e 4.18 comparam a solução conseguida pelo método 3 às soluções alcançadas pela empresa e pelos modelos para a instância IRM10, respectivamente.

No caso da instância IRM10, o ganho mostrado na solução da heurística em relação à solução da empresa é apenas resultado de uma reordenação dos *jobs* na série, uma vez que não há *setup*. O mesmo pode ser observado quando essas duas soluções comparadas à solução apresentada pelos modelos.

As soluções da empresa, dos modelos e da HC 4 são comparadas para a instância IRM42 nas Figuras 4.19 e 4.19.

No caso da instância IRM42 a redução de custo observada nas soluções apre-

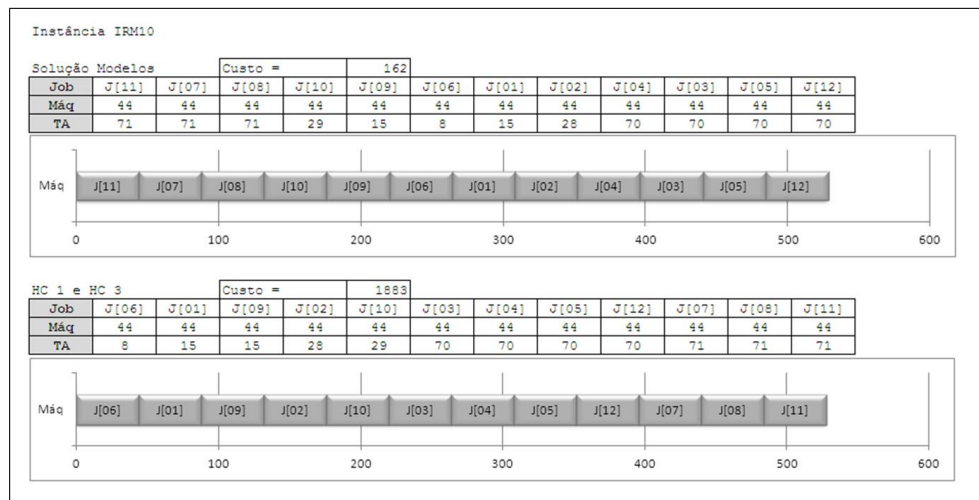


Figura 4.18: Heurística Construtiva 1 comparada à solução dos modelos



Figura 4.19: Heurística Construtiva 4 comparada à solução da empresa

sentadas pelos modelos e pela HC 4 é devido à inclusão do número de *setups*. Enquanto a empresa apresentou uma solução com apenas um *setup*, nos outros dois casos foram apresentadas soluções com três *setups*. A diferença na FO entre os modelos e a HC 4 se deve novamente à reorganização dos *jobs* nas séries, pois as séries das as duas soluções são constituídas dos mesmos *jobs*, porém em ordens diferentes.

Para verificar qual a heurística construtiva mais eficiente para achar regiões promissoras do espaço de busca foram realizados testes de comparação dos quatro métodos integrados ao método de busca local C (correspondente a percorrer a vizinhança C). Mais uma vez o método HC 4 se mostrou mais eficiente que os demais. Os resultados encontram-se no Apêndice B.



Figura 4.20: Heurística Construtiva 4 comparada à solução dos modelos

4.4.2 Resultados dos Testes da Meta-heurística ILS

O método ILS utilizado nos testes obtém como solução inicial a solução fornecida pela Heurística Construtiva 4 e a busca local consiste em percorrer a vizinhança C , que se mostrou mais eficiente nos testes da seção 4.1. Os testes foram realizados para as instâncias reais menores, para as quais são comparados os resultados do modelo 1 e da heurística ILS. O modelo 1 foi escolhido por ter obtido melhor desempenho do que o modelo 2 na maioria das instâncias menores. São apresentados na Tabela 4.18 e na Tabela 4.19 apenas os testes realizados para as instâncias 13 a 24 e 37 a 48. O resultados completos dos testes estão no apêndice B.

Tabela 4.18: Análise comparativa entre a solução fornecida pela empresa, a solução encontrada pelo Modelo 1 e a solução encontrada pelo ILS para as instâncias reais menores de 13 a 24. São apresentados na tabela a solução fornecida pela empresa (SE), a solução encontrada pelo Modelo 1 (FO), o GAP de otimalidade do modelo 1 (G%), o tempo de ciclo do modelo 1 (TC (s)), a função objetivo média encontrada nas 10 execuções do ILS (FO ILS), o tempo de ciclo de ILS (TC ILS) e o percentual de redução do modelo sobre o ILS, um percentual positivo indica superioridade do modelo (RED M1).

Modelo 1 x ILS - IRM13 a IRM24								
INST	NJ	SE	FO	G(%)	TC (s)	FO ILS	TC ILS	% RED M1
IRM13	20	39262 (1)	18438 (2)	22,06	7200,30	19600,2	104,81	6,3
IRM14	24	61018 (1)	15368 (2)	71,86	7201,83	17801,2	99,78	15,83
IRM15	22	17392 (1)	13331 (1)	13,56	7205,93	13331	93,08	0
IRM16	21	7714 (1)	6192 (1)	98,71	7200,25	6193,2	90,89	0,02
IRM17	18	38152 (1)	16635 (2)	0,00	41,00	16701,6	101,22	0,4
IRM18	20	55467 (1)	23832 (3)	46,31	7200,27	23832	99,52	0
IRM19	11	4199 (0)	1804 (0)	0,00	5,48	1804	83,64	0
IRM20	19	22297 (1)	12057 (1)	34,27	7200,29	14231	94,54	18,03
IRM21	11	37870 (0)	24536 (2)	0,00	0,97	24536	98,92	0
IRM22	23	37669 (1)	21746 (2)	42,66	7201,70	21989,6	95,39	1,12
IRM23	17	43209 (1)	22874 (3)	9,08	7200,17	22956,2	100,88	0,36
IRM24	12	6554 (0)	6554 (0)	0,00	0,65	6930	87,24	5,74

Tabela 4.19: Análise comparativa entre a solução fornecida pela empresa, a solução encontrada pelo Modelo 1 e a solução encontrada pelo ILS para as instâncias reais menores de 37 a 48. São apresentados na tabela a solução fornecida pela empresa (SE), a solução encontrada pelo Modelo 1 (FO), o GAP de otimalidade do modelo 1 (G%), o tempo de ciclo do modelo 1 (TC (s)), a função objetivo média encontrada nas 10 execuções do ILS (FO ILS), o tempo de ciclo de ILS (TC ILS) e o percentual de redução do modelo sobre o ILS, um percentual positivo indica superioridade do modelo (RED M1).

Modelo 1 x ILS - IRM37 a IRM48								
INST	NJ	SE	FO	G(%)	TC (s)	FO ILS	TC ILS	% RED M1
IRM37	20	30562 (1)	21680 (3)	25,68	7200,26	21708	101,89	0,13
IRM38	23	103968 (1)	40722 (3)	18,32	7200,56	38290,6	108,65	-5,97
IRM39	20	99229 (1)	47901 (5)	0,00	10,55	48178,6	114,95	0,58
IRM40	24	32051 (1)	18716 (1)	30,27	7204,57	19774	131,24	5,65
IRM41	24	51986 (1)	24399 (3)	45,69	7200,29	25847	96,37	5,93
IRM42	20	60527 (1)	25507 (3)	0,00	3856,50	25598	99,14	0,36
IRM43	21	20136 (1)	14459 (2)	0,00	4753,47	16645	95,53	15,12
IRM44	21	92011 (1)	38020 (4)	2,14	7200,32	38461	103,13	1,16
IRM45	19	43386 (1)	23474 (2)	0,00	5130,99	24524	96,54	4,47
IRM46	22	41130 (1)	16880 (1)	0,00	407,03	18317	91,63	8,51
IRM47	17	23981 (1)	9206 (1)	92,01	7200,25	9214	85,66	0,09
IRM48	14	56808 (1)	24683 (3)	0,00	9,69	24683	104,4	0

A partir dos resultados apresentados é possível afirmar que o método ILS obteve excelente desempenho, chegando a superar o modelo 1 na instância IRM38 e a igualar o modelo em outras instâncias, inclusive instâncias em que o modelo encontrou a solução ótima. O pior desempenho da meta-heurística, relativo ao do modelo, foi observado na instância IRM20, onde a superioridade da solução encontrada pelo modelo foi de 18,03%. Já o melhor desempenho do ILS foi observado na instância IRM38, onde a meta-heurística apresentou uma solução 5,97% melhor que a do modelo. A média do percentual de redução do modelo sobre o ILS foi de 2,57%, o que indica uma equivalência entre as estratégias quanto ao valor da FO. No entanto, o tempo médio de processamento da heurística para essas instâncias foi de 97,78 segundos.

A Figura 4.21 exibe os gráficos de Gantt para a solução apresentada pelo modelo 1 e para a solução apresentada pelo ILS. A principal diferença entre as soluções é que a heurística separou os *jobs* 17, 18 e 19 em uma série a parte e o modelo conseguiu

integrá-los em uma série maior através da mistura *JOB 17 - JOB 14* a um custo de 4004. Ao separar esses *jobs* dos demais a heurística precisou contar com mais um *setup*, a um custo de 6132. Essa diferença é responsável pelos 18,03% de superioridade do modelo.



Figura 4.21: Gráfico de Gantt IRM20 - Comparação entre ILS e Modelo 1

Na Figura 4.22 são apresentados os gráficos de Gantt do ILS e do modelo 1 para a instância IRM38. Esse foi o melhor resultado da heurística, obtendo 5,97% de superioridade da FO em comparação ao modelo. Ao contrário da última instância, a estratégia de acrescentar *setups* (provavelmente ocasionada pela perturbação do método) resultou em uma FO menor para o ILS. Ao todo foram utilizados 5 *setups* no ILS e apenas 3 no modelo. Enquanto na solução do modelo houve preferência por misturar o *job 20* com o *job 10* a um custo de 5506, no ILS houve o isolamento do *job 20* entre *setups* e o *job 10* foi misturado os *jobs 9* e *5* a custos de 1774 e 1896 respectivamente. A grande diferença entre as soluções, porém, está no *job 15*. Enquanto no modelo houve a mistura *JOB 16 - JOB 15* a um custo de 8855, na heurística houve o isolamento desse *job*, o que já ocorre na HC 4.



Figura 4.22: Gráfico de Gantt IRM38 - Comparação entre ILS e Modelo 1

Capítulo 5

Conclusão

A primeira conclusão deste trabalho é referente ao desempenho dos modelos matemáticos desenvolvidos. Ambos obtiveram bons resultados para as instâncias que conseguiram resolver. No entanto, os modelos 1 e 2 conseguiram resolver instâncias de, no máximo, 24 *jobs* apenas, o que é considerado pouco para o processo de sequenciamento na empresa estudada. Este fato implica na necessidade de tentar outras alternativas para resolver o problema.

Como parte do desenvolvimento de novas estratégias de resolução do problema, foram desenvolvidas quatro heurísticas construtivas, três procedimentos de busca local e uma meta-heurística ILS. A meta-heurística apresentou bons resultados para as instâncias menores, chegando a encontrar a solução ótima em algumas instâncias e até a superar a FO encontrada pelos modelos em outras instâncias e com um tempo de processamento muito inferior ao dos modelos. Este resultado é suficiente para sustentar a conclusão de que o método desenvolvido atende o objetivo de fornecer bons resultados a um tempo de processamento adequado ao processo produtivo, inclusive para instâncias de grande porte.

Sob olhar da companhia siderúrgica, os benefícios que a implementação de um programa que contenha o algoritmo gerado são vastos e de diferentes naturezas. Como o processo atual é manual, o mix variado de tipos de aço com que a organização trabalha gera um desafio enorme para os programadores, o que resulta em horizontes de planejamento baixos para o tipo de negócio. A avaliação quantitativa dos principais custos envolvidos no sequenciamento da aciaria permite a tomada de decisão alinhada com a atual principal diretriz do setor siderúrgico nacional: redução de custos.

A existência de um algoritmo que seja capaz de fornecer resultados como os

apresentados representa um avanço com impactos econômicos em diversas áreas da empresa. A otimização proposta para o sequenciamento evita custos de retrabalho com o retorno de sucata e custos de oportunidade de diferentes origens. Além de ter menor quantidade de sucata gerada, menor quantidade de aço puro será perdida na mistura e a produtividade do processo tem um incremento. Além disso, a otimização do sequenciamento permite a programação para horizontes ainda maiores do que os utilizados atualmente na programação, o que conseqüentemente gera ganhos de escala devido ao aumento das possibilidades de combinação de tipos de aço semelhantes. Com isso, o impacto da implementação da ferramenta chega às linhas de laminação, cujo sequenciamento pode ser beneficiado pelo horizonte mais longo de programação na aciaria.

Quanto às oportunidades identificadas, é importante observar que o campo de aplicação da pesquisa operacional na programação da produção de uma usina siderúrgica é pouco explorado em âmbito internacional e principalmente nacional. Outros estágios de produção e variáveis importantes como o custo de estoque e de não atendimento da demanda, interna e externa, podem ser incorporados a uma ferramenta como a desenvolvida nesse trabalho, formando assim um algoritmo mais completo. De modo que tanto no meio acadêmico quanto no industrial existem boas perspectivas para o desenvolvimento de pesquisas na programação da produção de siderúrgicas.

5.1 Contribuições Acadêmicas

Conforme citado anteriormente este trabalho é fruto de uma pesquisa conjunta que está sendo desenvolvida na Universidade Federal de Minas Gerais. Como primeira contribuição acadêmica da pesquisa pode-se citar os modelos matemáticos desenvolvidos em virtude da abordagem considerada para o problema. O *trade-off* entre custo de mistura e custo de realização de *setup* no Lingotamento Contínuo é pouco explorado ainda nos problemas de sequenciamento.

Além da dissertação de mestrado pretendida, a pesquisa contribui para o desenvolvimento da tese de doutorado de um dos envolvidos, na qual é estudado o problema de sequenciamento no Lingotamento Contínuo e o dimensionamento de lotes de produção para o LC de forma integrada [27]. Além da publicação de artigos e de apresentações de trabalhos em conferências internacionais [14].

5.2 Trabalhos Futuros

As próximas etapas da pesquisa serão direcionadas a refinar os métodos desenvolvidos. A simetria devido à combinação de séries ainda não foi tratada, de modo que se o algoritmo pudesse explorar uma região do espaço de busca que não contivesse espelhos o esforço computacional seria menor e o método mais eficiente.

No atual estágio da pesquisa, já houve a conversão dos modelos para tratar o sequenciamento em máquinas simples. Algumas melhoras foram observadas em relação aos modelos de sequenciamento em máquinas paralelas, no entanto instâncias com mais de 20 *jobs* ainda não podem ser resolvidas em um tempo computacional de 7.200 segundos.

Uma boa possibilidade para o avanço da pesquisa é a utilização do solução fornecida pela heurística como um *upper bound* para os modelos e observar se há grande diferença no tempo computacional. Como os modelos mostraram-se ineficazes para instâncias de grande porte a estratégia é utilizar de métodos heurísticos para alcançar esse objetivo.

Referências Bibliográficas

- [1] ADAMOPOULOS, G. I.; PAPPIS, C. P. “Scheduling under a common due-date on parallel unrelated machines”. *EJOR*, v. 105–3, p. 494–501, 1998.
- [2] ANGLANI, A.; GRIECO, A.; GUERRIERO, E.; MUSMANNO, R. “Robust scheduling of parallel machines with sequence-dependent set-up costs”. *EJOR*, v. 161, p. 704–720, 2005.
- [3] ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. *Pesquisa Operacional para Cursos de Engenharia*. 1ª Edição, Editora Campus - Elsevier, Rio de Janeiro-RJ, 2006.
- [4] ATIGHENCHIAN, A.; BIJARI, M.; TARKESH, H. “A novel hybrid algorithm for scheduling steel-making continuous casting production”. *EJOR*, v. 36, p. 2450–2461, 2009.
- [5] BAGCHI, U. B. “Simultaneous minimization of mean and variation of flow-time and waiting time in single machine systems”. *OR*, v. 37, p. 118–125, 1989.
- [6] BAUER, A.; BULLNHEIMER, B.; HARTL, R. F.; STRAUSS, C. “An ant colony optimization approach for the single machine total tardiness problem”. *Proceedings of the 1999 Congress on Evolutionary Computation*, v. 21, p. 1445–1450, 1999.
- [7] BELLABDAOUI, A.; TEGHEM, J. “A mixed-integer linear programming model for the continuous casting planning”. *Production Economics*, v. 104, p. 260–270, 2006.
- [8] BERRY, W. L. *Sistemas de Planejamento e Controle da Produção: Para o Gerenciamento da Cadeia de Suprimentos*. 1ª Edição, Bookman Companhia Ed, Porto Alegre-RS, 2006.

- [9] BUSTAMANTE, L. M. *Minimização do Custo de Antecipação e Atraso para o Problema de Sequenciamento de uma Máquina com Tempo de Preparação Dependente da Seqüência: Aplicação em uma Usina Siderúrgica*. Dissertação de mestrado, Programa de Pós Graduação em Engenharia de Produção, UFMG, Belo Horizonte, 2006.
- [10] CAO, D.; CHEN, M.; WAN, G. “Parallel machine selection and job scheduling to minimize machine cost and job tardiness”. *COR*, v. 32, p. 1995–2012, 2005.
- [11] CHENG, T. C. E.; SIN, C. C. S. “A State-of-the-art Review of Parallel Machine Scheduling Research”. *EJOR*, v. 77, p. 271–292, 1990.
- [12] CONGRAM, R. K.; POTTS, C. N.; VELDE, S. L. VAN DE. “An iterated dynasearch algorithm for the single-machine total weighted tardiness problem.”. *INFORMS Journal on Computing*, v. 14, p. 52–67, 2002.
- [13] CONTADOR, J. C. *Gestão de Operações*. 2ª Edição, Edgard Blucher LTDA, São Paulo-SP, 2001.
- [14] de SOUZA, M. C.; RAVETTI, M. G.; Jr., A. C. GOMES; BRETAS, A. M. C. “Scheduling in Continuous Ingot Casting”. 2013.
- [15] del BESTEN, M.; STUTZLE, T.; DORIGO, M. “Design of Iterated Local Search Algorithms: An Example Application to the Single Machine Total Weighted Tardiness Problem.”. 2001.
- [16] FANG, K.; LIN, B. M. T. “Parallel-machine scheduling to minimize tardiness penalty and power cost”. *Computers & Industrial Engineering*, v. 64, p. 224–234, 2013.
- [17] FELDMANN, M.; BISKUP, D. “Single-machine scheduling for minimizing earliness and tardiness penalties by meta-heuristic approaches.”. *Computers & Industrial Engineering*, v. 44, p. 307–323, 2003.
- [18] FLEURY, P.; WANKE, K. F. *Logística empresarial: A perspectiva brasileira*. Atlas Ed. Coleção COPPEAD, São Paulo-SP, 2000.
- [19] GAREY, M. R.; JOHNSON, D. S. *Computers and intractability: A guide to the theory of NP-completeness*. New York Freeman, San Francisco, CA, 1979.

- [20] GEIGER, M. J. “On heuristic search for the single machine total weighted tardiness problem – Some theoretical insights and their empirical verification.”. *EJOR*, v. 207, p. 1235–1243, 2010.
- [21] GLASS, C. A.; POTTS, C. N.; SHADE, P. “Unrelated parallel machine scheduling using local search”. *Mathl. Comput. Modelling*, v. 20(2), p. 41–52, 1994.
- [22] GUREL, S.; AKTURK, M. S. “Scheduling parallel CNC machines with time/cost trade-off considerations”. *COR*, v. 34, p. 2774–2789, 2007.
- [23] HARJUNKOSKI, I.; GROSSMANN, I. E. “A decomposition approach for the scheduling of a steel plant production”. *Computers and Chemical Engineering*, v. 25, p. 1647–1660, 2001.
- [24] HOOS, H. H.; STÜTZLE, T. *Stochastic Local Search: Foundations and Applications*. 1ª Edição, Editora Morgan Kaufmann - Elsevier, San Francisco-CA, 2005.
- [25] JANIÁK, A.; JANIÁK, W.; MIKHAIL, M.; KOVALYOV, Y.; KOZAN, E.; PESCH, E. “Parallel machine scheduling and common due window assignment with job independent earliness and tardiness costs”. *IS*, v. 224, p. 109–123, 2013.
- [26] JR, A. C. GOMES. *Problema de Sequenciamento em uma Máquina com Penalidades por Antecipação e Atraso: Modelagem e Resolução*. Dissertação de mestrado, Programa de Pós Graduação em Engenharia de Produção, UFMG, Belo Horizonte, 2007.
- [27] JR, A. C. GOMES. *Dimensionamento e Sequenciamento de Lotes em um Processo de Lingotamento Contínuo*. Qualificação de doutorado, Programa de Pós Graduação em Engenharia de Produção, UFMG, Belo Horizonte, 2012.
- [28] KANET, J. J. “Minimizing variation of flow time in single machine systems.”. *Management Science*, v. 27, p. 1453–1459, 1981.
- [29] KEHA, A. B.; KHOWALA, K.; FOWLER, J. W. “Mixed integer programming formulations for single machine scheduling problems.”. *Computers & Industrial Engineering*, v. 56, p. 357–367, 2009.
- [30] KOULAMAS, C.; KYPARISIS, G. J. “Single-machine scheduling problems with past-sequence-dependent setup times.”. *EJOR*, v. 187–3, p. 1045–1049, 2008.

- [31] KRASIK, V.; LEUNG, J.; PINEDO, M.; ZHANG, J. “Scheduling Multiple Products on Parallel Machines with Setup Costs”. *Naval Research Logistics*, v. 55, p. 654–669, 2008.
- [32] LI, J.; XIAO, X.; TANG, Q.; FLOUDAS, C. A. “Production Scheduling of a Large-Scale Steelmaking Continuous Casting Process via Unit-Specific Event-Based Continuous-Time Models: Short-Term and Medium-Term Scheduling”. *Industrial & Engineering Chemistry Research*, v. 51, p. 7300–7319, 2012.
- [33] LOPES, M. J. P.; CARVALHO, J. M. V. “A branch-and-price algorithm for scheduling parallel machines with sequence dependent setup times”. *EJOR*, v. 176–3, p. 1508–1527, 2006.
- [34] MAHESWARAN, R.; PONNAMBALAN, S. G. “An investigation on single machine total weighted tardiness scheduling problems”. *The International Journal of Advanced Manufacturing Technology*, v. 22, p. 243–248, 2003.
- [35] MANNE, A. S. “On the Jobshop Scheduling Problem”. *OR*, v. 8, p. 219–223, 1960.
- [36] MATTFELD, D. C.; BIERWIRTH, C.; KOPFER, H. “A search space analysis of the Job Shop Scheduling Problem”. *Annals of Operations Research*, v. 86, p. 441–453, 1999.
- [37] MISSBAUER, H.; HAUBER, W.; STADLER, W. “A Scheduling System for the Steelmaking Continuous Casting Process. A Case Study from the Steel-Making Industry”. *International Journal of Production Research*, v. 47, p. 4147–4172, 2009.
- [38] MOKOTOFF, E. “Parallel machine scheduling problems: A survey”. *Pacific Journal of Operational Research*, v. 18, p. 193–242, 2001.
- [39] MOREIRA, D. A. *Administração da produção e operações*. 3ª Edição, Pioneira Ed., São Paulo-SP, 1998.
- [40] NEARCHOU, A. C. “Solving the single machine total weighted tardiness scheduling problem using a hybrid simulated annealing algorithm”. *Second IEEE International Conference on Industrial Informatics*, p. 513–516, 2004.

- [41] PACCIARELLI, D.; PRANZO, M. “Production Scheduling in a Steelmaking Continuous-casting plant”. *Computers and Chemical Engineering*, v. 8, p. 2823–2835, 2004.
- [42] PAULA, M. R.; MATEUS, G. R.; RAVETTI, M. G. “A nondelayed relaxandcut algorithm for scheduling problems with parallel machines, due dates and sequence dependent setup times”. *COR*, v. 37, p. 938–949, 2010.
- [43] PINEDO, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. *Scheduling: Theory, Algorithms and Systems*. Prentice-Hall, New Jersey, 1995.
- [44] RAVETTI, M. G. *Algoritmos para o problema de sequenciamento com máquinas paralelas e tempos de preparação dependentes da sequencia*. Tese de doutorado, Programa de Pós Graduação em Ciência da Computação, UFMG, Belo Horizonte, 2007.
- [45] REEVES, C. R. “Landscapes, operators and heuristic search.”. *Annals of Operations Research*, v. 86, p. 473–490, 1999.
- [46] ROCHA, P. L.; RAVETTI, M. G.; MATEUS, G. R.; PARDALOS, G. R. “Exact algorithms for a scheduling problem with unrelated parallel machines and sequence and machine-dependent setup times”. *COR*, v. 35, p. 1250–1264, 2007.
- [47] TANG, L.; LIU, J.; RONG, A.; YANG, Z. “A review of planning and scheduling systems and methods for integrated steel production”. *EJOR*, v. 133, p. 1–20, 2001.
- [48] TANG, L.; LUO, J. “A New ILS Algorithm for Cast Planning Problem in Steel Industry”. *ISIJ International*, v. 47, p. 443–452, 2007.
- [49] TANG, L. X.; WANG, G. S.; LIU, J. Y.; LIU, J. A. “A Combination of Lagrangian Relaxation and Column Generation for Order Batching in Steelmaking and Continuous-Casting Production.”. *Naval Research Logistics*, v. 58, p. 370–388, 2011.
- [50] WAGNER, H. M. “An Intenger Programming Model for Machine Scheduling”. *Naval Research Logistics Quarterly*, v. 6, p. 131–140, 1959.
- [51] ZANONI, S.; ZAVANELLA, L. “Model and analysis of integrated production inventory system: The case of steel production”. *International Journal of Production Economics*, v. 93–94, p. 197–205, 2005.

Apêndice A

Este apêndice apresenta os resultados dos testes descritos na seção 3.4. Nas tabelas a seguir estão os resultados para os testes com os modelos 1 e 2 para todas as 48 instâncias menores.

Tabela 5.1: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 1 para as instâncias reduzidas de 1 a 12

Modelo 1 - IRM01 a IRM12						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM01	12	6280 (0)	6280 (0)	0,00	0,12	0,00
IRM02	12	6554 (0)	6554 (0)	0,00	1,12	0,00
IRM03	12	27708 (0)	6306 (1)	0,00	0,62	77,24
IRM04	12	27178 (0)	12336 (2)	0,00	2,11	54,61
IRM05	11	37026 (0)	16691 (2)	0,00	0,61	54,92
IRM06	11	46500 (0)	18464 (1)	0,00	0,87	60,29
IRM07	12	50929 (0)	18587 (3)	0,00	20,60	63,50
IRM08	11	20486 (0)	9324 (0)	0,00	2,83	54,49
IRM09	12	15260 (0)	12711 (0)	0,00	0,94	16,70
IRM10	12	6218 (0)	162 (0)	0,00	0,06	97,39
IRM11	12	13020 (0)	9684 (0)	0,00	5,14	25,62
IRM12	12	11051 (0)	6290 (1)	0,00	7,08	43,08

Tabela 5.2: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 2 para as instâncias reduzidas de 1 a 12

Modelo 2 - IRM01 a IRM12						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM01	12	6280 (0)	6280 (0)	0,00	34,27	0,00
IRM02	12	6554 (0)	6554 (0)	0,00	3,52	0,00
IRM03	12	27708 (0)	6306 (1)	0,00	5,98	77,24
IRM04	12	27178 (0)	12336 (2)	0,00	5,18	54,61
IRM05	11	37026 (0)	16691 (3)	0,00	3,17	54,92
IRM06	11	46500 (0)	18464 (1)	0,00	9,67	60,29
IRM07	12	50929 (0)	18587 (3)	0,00	59,53	63,50
IRM08	11	20486 (0)	9324 (0)	0,00	6,55	54,49
IRM09	12	15260 (0)	12711 (0)	0,00	18,34	16,70
IRM10	12	6218 (0)	162 (0)	0,00	1,00	97,39
IRM11	12	13020 (0)	9684 (0)	0,00	38,93	25,62
IRM12	12	11051 (0)	6290 (1)	0,00	5,21	43,08

Tabela 5.3: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 1 para as instâncias reduzidas de 13 a 24

Modelo 1 - IRM13 a IRM24						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM13	20	39262 (1)	18438 (2)	22,06	7200,30	53,04
IRM14	24	61018 (1)	15368 (2)	71,86	7201,83	74,81
IRM15	22	17392 (1)	13331 (1)	13,56	7205,93	23,35
IRM16	21	7714 (1)	6192 (1)	98,71	7200,25	19,73
IRM17	18	38152 (1)	16635 (2)	0,00	41,00	56,40
IRM18	20	55467 (1)	23832 (3)	46,31	7200,27	57,03
IRM19	11	4199 (0)	1804 (0)	0,00	5,48	57,04
IRM20	19	22297 (1)	12057 (1)	34,27	7200,29	45,93
IRM21	11	37870 (0)	24536 (2)	0,00	0,97	35,21
IRM22	23	37669 (1)	21746 (2)	42,66	7201,70	42,27
IRM23	17	43209 (1)	22874 (3)	9,08	7200,17	47,06
IRM24	12	6554 (0)	6554 (0)	0,00	0,65	0,00

Tabela 5.4: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 2 para as instâncias reduzidas de 13 a 24

Modelo 2 - IRM13 a IRM24						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM13	20	39262 (1)	18438 (2)	65,20	7200,37	53,04
IRM14	24	61018 (1)	15368 (2)	20,19	7203,22	74,81
IRM15	22	17392 (1)	15779 (1)	22,28	7200,34	9,27
IRM16	21	7714 (1)	6192 (1)	0,00	1267,07	19,73
IRM17	18	38152 (1)	16635 (2)	26,25	7200,22	56,40
IRM18	20	55467 (1)	23832 (3)	65,38	7200,36	57,03
IRM19	11	4199 (0)	1804 (0)	0,00	2,45	57,04
IRM20	19	22297 (1)	12057 (1)	0,00	2552,47	45,93
IRM21	11	37870 (0)	24536 (2)	0,00	11,29	35,21
IRM22	23	37669 (1)	22494 (2)	45,48	7201,69	40,29
IRM23	17	43209 (1)	22874 (3)	0,00	4779,39	47,06
IRM24	12	6554 (0)	6554 (0)	0,00	3,17	0,00

Tabela 5.5: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 1 para as instâncias reduzidas de 25 a 36

Modelo 1 - IRM25 a IRM36						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM25	24	14439 (1)	14439 (1)	31,10	7200,28	0,00
IRM26	16	28703 (2)	24199 (3)	23,93	7200,32	15,69
IRM27	19	20024 (2)	19906 (2)	58,97	7200,24	0,59
IRM28	17	32164 (1)	17433 (1)	0,00	609,67	45,80
IRM29	21	33956 (1)	12438 (1)	96,80	7200,34	63,37
IRM30	18	23662 (1)	22533 (2)	0,00	421,22	4,77
IRM31	21	72003 (1)	29028 (4)	40,15	7200,26	59,69
IRM32	19	29474 (2)	25015 (4)	23,75	7200,30	15,13
IRM33	19	33297 (1)	22300 (3)	22,12	7200,29	33,03
IRM34	22	54465 (1)	26376 (3)	4,81	7200,27	51,57
IRM35	21	28900 (1)	22192 (2)	20,44	7200,27	23,21
IRM36	21	70425 (1)	31791 (4)	15,37	7200,79	54,86

Tabela 5.6: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 2 para as instâncias reduzidas de 25 a 36

Modelo 2 - IRM25 a IRM36						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM25	24	14439 (1)	18551 (1)	33,89	7200,29	-28,48
IRM26	16	28703 (2)	24199 (3)	0,00	408,16	15,69
IRM27	19	20024 (2)	19906 (2)	33,10	7202,39	0,59
IRM28	17	32164 (1)	17433 (1)	0,00	2205,50	45,80
IRM29	21	33956 (1)	12438 (1)	49,46	7200,47	63,37
IRM30	18	23662 (1)	22533 (2)	36,51	7201,26	4,77
IRM31	21	72003 (1)	29028 (4)	78,88	7201,61	59,69
IRM32	19	29474 (2)	25015 (4)	72,67	7200,94	15,13
IRM33	19	33297 (1)	22300 (3)	44,92	7200,64	33,03
IRM34	22	54465 (1)	37745 (2)	83,75	7201,85	30,70
IRM35	21	28900 (1)	22377 (2)	71,51	7202,54	22,57
IRM36	21	70425 (1)	31791 (4)	74,86	7200,44	54,86

Tabela 5.7: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 1 para as instâncias reduzidas de 37 a 48

Modelo 1 - IRM37 a IRM48						
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red
IRM37	20	30562 (1)	21680 (3)	25,68	7200,26	29,06
IRM38	23	103968 (1)	40722 (3)	18,32	7200,56	60,83
IRM39	20	99229 (1)	47901 (5)	0,00	10,55	51,73
IRM40	24	32051 (1)	18716 (1)	30,27	7204,57	41,61
IRM41	24	51986 (1)	24399 (3)	45,69	7200,29	53,07
IRM42	20	60527 (1)	25507 (3)	0,00	3856,50	57,86
IRM43	21	20136 (1)	14459 (2)	0,00	4753,47	28,19
IRM44	21	92011 (1)	38020 (4)	2,14	7200,32	58,68
IRM45	19	43386 (1)	23474 (2)	0,00	5130,99	45,89
IRM46	22	41130 (1)	16880 (1)	0,00	407,03	58,96
IRM47	17	23981 (1)	9206 (1)	92,01	7200,25	61,61
IRM48	14	56808 (1)	24683 (3)	0,00	9,69	56,55

Tabela 5.8: Comparação entre a solução fornecida pela empresa e a solução encontrada pelo Modelo 2 para as instâncias reduzidas de 37 a 48

Modelo 2 - IRM37 a IRM48							
INST	NJ	SolEmp	FO	G(%)	TC (s)	% red	
IRM37	20	30562 (1)	26239 (2)	75,71	7202,28	14,15	
IRM38	23	103968 (1)	61729 (2)	80,13	7205,07	40,63	
IRM39	20	99229 (1)	47901 (5)	77,20	7200,47	51,73	
IRM40	24	32051 (1)	20468 (1)	39,88	7200,69	36,14	
IRM41	24	51986 (1)	26051 (3)	52,92	7199,87	49,89	
IRM42	20	60527 (1)	25507 (3)	53,71	7200,38	57,86	
IRM43	21	20136 (1)	14459 (2)	44,10	7200,38	28,19	
IRM44	21	92011 (1)	38020 (4)	83,87	7200,31	58,68	
IRM45	19	43386 (1)	23474 (2)	47,04	7200,28	45,89	
IRM46	22	41130 (1)	18317 (1)	33,05	7200,34	55,47	
IRM47	17	23981 (1)	9206 (1)	0,00	432,76	61,61	
IRM48	14	56808 (1)	24683 (3)	0,00	685,82	56,55	

Apêndice B

No Apêndice B são apresentados todos os resultados encontrados nos testes das heurísticas construtivas e da meta-heurística ILS. As tabelas contendo os resultados apresentam a função objetivo encontrada por cada procedimento e a diferença, em porcentagem, que esses resultados estão da função objetivo encontrada pelo Modelo 1 (modelo de melhor desempenho).

Tabela 5.9: Comparação entre as soluções encontradas pelo Modelo 1 e pelos métodos HC 1 e HC 2 para as instâncias reduzidas de 1 a 12

Heurísticas Construtivas - IRM01 a IRM12						
INST	NJ	FO Mod 1	HC1	D(%)	HC2	D(%)
IRM01	12	6280 (OT)	19204	205,8	25746	309,97
IRM02	12	6554 (OT)	11440	74,55	6930	5,74
IRM03	12	6306 (OT)	32451	414,61	75630	1099,33
IRM04	12	12336 (OT)	18242	47,88	27914	126,28
IRM05	11	16691 (OT)	42324	153,57	31485	88,63
IRM06	11	18464 (OT)	32755	77,4	32968	78,55
IRM07	12	18587 (OT)	86907	367,57	91578	392,7
IRM08	11	9324 (OT)	20623	121,18	75032	704,72
IRM09	12	12711 (OT)	14343	12,84	18767	47,64
IRM10	12	162 (OT)	1883	1062,35	7701	4653,7
IRM11	12	9684 (OT)	16548	70,88	36652	278,48
IRM12	12	6290 (OT)	17664	180,83	11558	83,75

Tabela 5.10: Comparação entre as soluções encontradas pelo Modelo 1 e pelos métodos HC 1 e HC 2 para as instâncias reduzidas de 13 a 24

Heurísticas Construtivas - IRM13 a IRM24						
INST	NJ	FO Mod 1	HC1	D(%)	HC2	D(%)
IRM13	20	18438 (MS)	76487	314,83	71243	286,39
IRM14	24	15368 (MS)	42435	176,13	85142	454,02
IRM15	22	13331 (MS)	55720	317,97	107804	708,67
IRM16	21	6192 (MS)	46514	651,2	16909	173,08
IRM17	18	16635 (OT)	88065	429,4	71875	332,07
IRM18	20	23832 (MS)	81078	240,21	112607	372,5
IRM19	11	1804 (OT)	4199	132,76	2339	29,66
IRM20	19	12057 (MS)	29096	141,32	31873	164,35
IRM21	11	24536 (MS)	36873	50,28	36935	50,53
IRM22	23	21746 (MS)	44419	104,26	92722	326,39
IRM23	17	22874 (MS)	48558	112,28	37668	64,68
IRM24	12	6554 (OT)	11440	74,55	6930	5,74

Tabela 5.11: Comparação entre as soluções encontradas pelo Modelo 1 e pelos métodos HC 3 e HC 4 para as instâncias reduzidas de 1 a 12

Heurísticas Construtivas - IRM01 a IRM12						
INST	NJ	FO Mod 1	HC3	D(%)	HC4	D(%)
IRM01	12	6280 (OT)	18412	193,18	6525	3,9
IRM02	12	6554 (OT)	11440	74,55	6930	5,74
IRM03	12	6306 (OT)	12439	97,26	6307	0,02
IRM04	12	12336 (OT)	12336	0	12336	0
IRM05	11	16691 (OT)	16906	1,29	18279	9,51
IRM06	11	18464 (OT)	23165	25,46	24648	33,49
IRM07	12	18587 (OT)	18587	0	18587	0
IRM08	11	9324 (OT)	12389	32,87	17767	90,55
IRM09	12	12711 (OT)	14343	12,84	18767	47,64
IRM10	12	162 (OT)	1883	1062,35	7701	4653,7
IRM11	12	9684 (OT)	16548	70,88	36652	278,48
IRM12	12	6290 (OT)	12396	97,07	6290	0

Tabela 5.12: Comparação entre as soluções encontradas pelo Modelo 1 e pelos métodos HC 3 e HC 4 para as instâncias reduzidas de 13 a 24

Heurísticas Construtivas - IRM13 a IRM24						
INST	NJ	FO Mod 1	HC3	D(%)	HC4	D(%)
IRM13	20	18438 (MS)	18572	0,73	24644	33,66
IRM14	24	15368 (MS)	24348	58,43	34088	121,81
IRM15	22	13331 (MS)	32246	141,89	23662	77,5
IRM16	21	6192 (MS)	6231	0,63	7691	24,21
IRM17	18	16635 (OT)	29005	74,36	16995	2,16
IRM18	20	23832 (MS)	23989	0,66	29509	23,82
IRM19	11	1804 (OT)	4199	132,76	2339	29,66
IRM20	19	12057 (MS)	20627	71,08	14766	22,47
IRM21	11	24536 (MS)	24944	1,66	25294	3,09
IRM22	23	21746 (MS)	30917	42,17	30198	38,87
IRM23	17	22874 (MS)	23140	1,16	24462	6,94
IRM24	12	6554 (OT)	11440	74,55	6930	5,74

A seguir são apresentados os resultados dos testes feitos com algoritmos que contemplam a integração das heurísticas construtivas com o método de busca local C.

Tabela 5.13: Comparação da média da FO apresentada por cada um 4 métodos construtivos com posterior aplicação da busca local C para 10 testes com cada uma das instâncias artificiais com 25 *jobs*

Comparação FO entre HC + BL C - Inst 25 JOBS				
Instância	HC1	HC2	HC3	HC4
IA2501	117969	128726	92257	90502
IA2502	278041	244311	112828	106938
IA2503	159516	311161	100910	105933
IA2504	270047	154765	90563	89677
IA2505	82420	138750	62793	74229
IA2506	181653	137719	84379	84552
IA2507	239684	132392	81983	82995
IA2508	323284	295435	127760	118794
IA2509	299383	276589	96467	95823
IA2510	116693	130986	97144	97078
MÉDIA	206869	195083	94708	94652

Tabela 5.14: Comparação da média da FO apresentada por cada um 4 métodos construtivos com posterior aplicação da busca local C para 10 testes com cada uma das instâncias artificiais com 50 *jobs*

Comparação FO entre HC + BL C - Inst 50 JOBS				
Instância	HC1	HC2	HC3	HC4
IA5001	522188	671591	140731	135118
IA5002	154318	582586	120735	106852
IA5003	419428	509881	114647	99523
IA5004	165057	682431	135274	120527
IA5005	191652	573458	136418	142362
IA5006	209393	604280	159432	138270
IA5007	249410	596295	120019	118008
IA5008	206764	694244	135740	139236
IA5009	260230	644774	135946	119745
IA5010	236185	595909	124055	122003
MÉDIA	261463	615545	132300	124164

Tabela 5.15: Comparação da média da FO apresentada por cada um 4 métodos construtivos com posterior aplicação da busca local C para 10 testes com cada uma das instâncias artificiais com 75 *jobs*

Comparação FO entre HC + BL C - Inst 75 JOBS				
Instância	HC1	HC2	HC3	HC4
IA7501	874048	1091812	315788	317812
IA7502	942702	920588	313589	270844
IA7503	932565	1162508	305493	300630
IA7504	993549	998657	317067	284973
IA7505	904649	987045	346206	338308
IA7506	965859	1200226	328129	326389
IA7507	919554	1121977	311028	283870
IA7508	968196	1061491	310306	292828
IA7509	978809	1055288	319981	292373
IA7510	1022342	1027950	318420	299211
MÉDIA	950227,3	1062754	318601	300724

A seguir são apresentados os testes realizados com a meta-heurística ILS. Nos

testes são comparados a solução fornecida pela empresa, os resultados encontrados pelo modelo 1 e os resultados encontrados pelo ILS para as 48 instâncias reais menores.

Tabela 5.16: Análise comparativa entre a solução fornecida pela empresa, a solução encontrada pelo Modelo 1 e a solução encontrada pelo ILS para as instâncias reais menores de 1 a 12. São apresentados na tabela a solução fornecida pela empresa (SE), a solução encontrada pelo Modelo 1 (FO M1), o GAP de otimalidade do modelo 1 (G%), o tempo de ciclo do modelo 1 (TC (s)), a função objetivo média encontrada nas 10 iterações do ILS (FO ILS), o tempo de processamento do ILS (TC ILS) e o percentual de redução do modelo sobre o ILS, um percentual positivo indica superioridade do modelo (RED M1).

Modelo 1 x ILS - IRM01 a IRM12								
INST	NJ	SE	FO	G(%)	TC (s)	FO ILS	TC ILS	% RED M1
IRM01	12	6280 (0)	6280 (0)	0,00	0,12	6480,5	88,11	3,19
IRM02	12	6554 (0)	6554 (0)	0,00	1,12	6930	87,17	5,74
IRM03	12	27708 (0)	6306 (1)	0,00	0,62	6306,5	92,17	0,01
IRM04	12	27178 (0)	12336 (2)	0,00	2,11	12336	98,9	0
IRM05	11	37026 (0)	16691 (2)	0,00	0,61	16700,8	94,54	0,06
IRM06	11	46500 (0)	18464 (1)	0,00	0,87	18464	101,36	0
IRM07	12	50929 (0)	18587 (3)	0,00	20,60	18587	114,19	0
IRM08	11	20486 (0)	9324 (0)	0,00	2,83	9324	88,87	0
IRM09	12	15260 (0)	12711 (0)	0,00	0,94	12711	88,83	0
IRM10	12	6218 (0)	162 (0)	0,00	0,06	162	87,69	0
IRM11	12	13020 (0)	9684 (0)	0,00	5,14	9700,4	86,46	0,17
IRM12	12	11051 (0)	6290 (1)	0,00	7,08	6290	94,53	0

Tabela 5.17: Análise comparativa entre a solução fornecida pela empresa, a solução encontrada pelo Modelo 1 e a solução encontrada pelo ILS para as instâncias reais menores de 13 a 24. São apresentados na tabela a solução fornecida pela empresa (SE), a solução encontrada pelo Modelo 1 (FO M1), o GAP de otimalidade do modelo 1 (G%), o tempo de ciclo do modelo 1 (TC (s), a função objetivo média encontrada nas 10 iterações do ILS (FO ILS), o tempo de processamento do ILS (TC ILS) e o percentual de redução do modelo sobre o ILS, um percentual positivo indica superioridade do modelo (RED M1).

Modelo 1 x ILS - IRM13 a IRM24								
INST	NJ	SE	FO	G(%)	TC (s)	FO ILS	TC ILS	% RED M1
IRM13	20	39262 (1)	18438 (2)	22,06	7200,30	19600,2	104,81	6,3
IRM14	24	61018 (1)	15368 (2)	71,86	7201,83	17801,2	99,78	15,83
IRM15	22	17392 (1)	13331 (1)	13,56	7205,93	13331	93,08	0
IRM16	21	7714 (1)	6192 (1)	98,71	7200,25	6193,2	90,89	0,02
IRM17	18	38152 (1)	16635 (2)	0,00	41,00	16701,6	101,22	0,4
IRM18	20	55467 (1)	23832 (3)	46,31	7200,27	23832	99,52	0
IRM19	11	4199 (0)	1804 (0)	0,00	5,48	1804	83,64	0
IRM20	19	22297 (1)	12057 (1)	34,27	7200,29	14231	94,54	18,03
IRM21	11	37870 (0)	24536 (2)	0,00	0,97	24536	98,92	0
IRM22	23	37669 (1)	21746 (2)	42,66	7201,70	21989,6	95,39	1,12
IRM23	17	43209 (1)	22874 (3)	9,08	7200,17	22956,2	100,88	0,36
IRM24	12	6554 (0)	6554 (0)	0,00	0,65	6930	87,24	5,74

Tabela 5.18: Análise comparativa entre a solução fornecida pela empresa, a solução encontrada pelo Modelo 1 e a solução encontrada pelo ILS para as instâncias reais menores de 25 a 36. São apresentados na tabela a solução fornecida pela empresa (SE), a solução encontrada pelo Modelo 1 (FO M1), o GAP de otimalidade do modelo 1 (G%), o tempo de ciclo do modelo 1 (TC (s)), a função objetivo média encontrada nas 10 iterações do ILS (FO ILS), o tempo de processamento do ILS (TC ILS) e o percentual de redução do modelo sobre o ILS, um percentual positivo indica superioridade do modelo (RED M1).

Modelo 1 x ILS - IRM25 a IRM36								
INST	NJ	SE	FO	G(%)	TC (s)	FO ILS	TC ILS	% RED M1
IRM25	24	14439 (1)	14439 (1)	31,10	7200,28	15550	93,25	7,69
IRM26	16	28703 (2)	24199 (3)	23,93	7200,32	24199	101,94	0
IRM27	19	20024 (2)	19906 (2)	58,97	7200,24	19906	90,53	0
IRM28	17	32164 (1)	17433 (1)	0,00	609,67	17452	90,8	0,11
IRM29	21	33956 (1)	12438 (1)	96,80	7200,34	12467,6	93,52	0,24
IRM30	18	23662 (1)	22533 (2)	0,00	421,22	22533	98,34	0
IRM31	21	72003 (1)	29028 (4)	40,15	7200,26	31075	105,05	7,05
IRM32	19	29474 (2)	25015 (4)	23,75	7200,30	25015	106,91	0
IRM33	19	33297 (1)	22300 (3)	22,12	7200,29	24145,6	97,63	8,28
IRM34	22	54465 (1)	26376 (3)	4,81	7200,27	28005	102,32	6,18
IRM35	21	28900 (1)	22192 (2)	20,44	7200,27	22346,8	99,27	0,7
IRM36	21	70425 (1)	31791 (4)	15,37	7200,79	31791	112,15	0

Tabela 5.19: Análise comparativa entre a solução fornecida pela empresa, a solução encontrada pelo Modelo 1 e a solução encontrada pelo ILS para as instâncias reais menores de 37 a 48. São apresentados na tabela a solução fornecida pela empresa (SE), a solução encontrada pelo Modelo 1 (FO M1), o GAP de otimalidade do modelo 1 (G%), o tempo de ciclo do modelo 1 (TC (s)), a função objetivo média encontrada nas 10 iterações do ILS (FO ILS), o tempo de processamento do ILS (TC ILS) e o percentual de redução do modelo sobre o ILS, um percentual positivo indica superioridade do modelo (RED M1).

Modelo 1 x ILS - IRM37 a IRM48								
INST	NJ	SE	FO	G(%)	TC (s)	FO ILS	TC ILS	% RED M1
IRM37	20	30562 (1)	21680 (3)	25,68	7200,26	21708	101,89	0,13
IRM38	23	103968 (1)	40722 (3)	18,32	7200,56	38290,6	108,65	-5,97
IRM39	20	99229 (1)	47901 (5)	0,00	10,55	48178,6	114,95	0,58
IRM40	24	32051 (1)	18716 (1)	30,27	7204,57	19774	131,24	5,65
IRM41	24	51986 (1)	24399 (3)	45,69	7200,29	25847	96,37	5,93
IRM42	20	60527 (1)	25507 (3)	0,00	3856,50	25598	99,14	0,36
IRM43	21	20136 (1)	14459 (2)	0,00	4753,47	16645	95,53	15,12
IRM44	21	92011 (1)	38020 (4)	2,14	7200,32	38461	103,13	1,16
IRM45	19	43386 (1)	23474 (2)	0,00	5130,99	24524	96,54	4,47
IRM46	22	41130 (1)	16880 (1)	0,00	407,03	18317	91,63	8,51
IRM47	17	23981 (1)	9206 (1)	92,01	7200,25	9214	85,66	0,09
IRM48	14	56808 (1)	24683 (3)	0,00	9,69	24683	104,4	0

Apêndice C

Neste apêndice são apresentadas é apresentada a análise de variância realizada a partir dos testes feitos para calibração dos parâmetros do método ILS. A tabela 5.20 apresenta a análise de variância para os parâmetros.

Tabela 5.20: Tabela ANOVA para os parâmetros número de *jobs* (NJ), critério de aceitação (K), número de *jobs* perturbados (N) e número de perturbações (NP).

ANOVA					
Parâmetros	G.L.	Soma Quadr.	Quadr. Médio	Estat. F	P-valor
NJ	3	25044,4041	8348,1347	40,78192772	0
K	2	775,1246478	387,5623239	1,893301827	0,1514
N	2	21,19642775	10,59821387	0,051773912	0,9495
NP	1	651,5763486	651,5763486	3,183051126	0,0748
NJ:K	6	1558,624252	259,7707087	1,269020045	0,2694
NJ:N	6	387,5316507	64,58860844	0,315525331	0,929
NJ:NP	3	532,2665019	177,4221673	0,866734697	0,458
K:N	4	264,9382284	66,23455711	0,323566044	0,8622
K:NP	2	204,448393	102,2241965	0,499380993	0,6071
N:NP	2	26,2443478	13,1221739	0,064103847	0,9379
Resíduos	688	140834,85	204,7018168		

A partir da análise não é possível rejeitar a hipótese de que as médias são iguais a um nível de confiança de 95%, pois o p-valor encontrado foi alto para todos os parâmetros e todas as interações entre parâmetros analisados. Com exceção do número de *jobs*, que é uma variável não-controlada. A figura 5.1 apresenta a análise dos resíduos para os resultados dos testes.

Figura 5.1: Análise de resíduos para os dados dos testes dos parâmetros do ILS.

