

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO
ENGENHARIA DE PRODUÇÃO

CHRISTOPHER RENAN MARINHO DE SOUSA

**PROGRAMAÇÃO DA PRODUÇÃO UTILIZANDO HEURÍSTICAS NO
AMBIENTE DE MÁQUINAS PARALELAS NÃO RELACIONADAS**

PONTA GROSSA

2020

CHRISTOPHER RENAN MARINHO DE SOUSA

**PROGRAMAÇÃO DA PRODUÇÃO UTILIZANDO HEURÍSTICAS NO
AMBIENTE DE MÁQUINAS PARALELAS NÃO RELACIONADAS**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia de Produção, do Departamento Acadêmico de Engenharia de Produção, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Everton Luiz de Melo

PONTA GROSSA

2020



Ministério da Educação
**UNIVERSIDADE TECNOLÓGICA FEDERAL DO
PARANÁ**
CÂMPUS PONTA GROSSA
Departamento Acadêmico de Engenharia de Produção



TERMO DE APROVAÇÃO DE TCC

PROGRAMAÇÃO DA PRODUÇÃO UTILIZANDO HEURÍSTICAS NO AMBIENTE DE MÁQUINAS PARALELAS NÃO RELACIONADAS

por

Christopher Renan Marinho de Sousa

Este Trabalho de Conclusão de Curso (TCC) foi apresentado em 15 de julho de 2020 como requisito parcial para a obtenção do título de Bacharel em Engenharia de Produção. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Dr. Everton Luiz de Melo
Prof. Presidente da banca

Prof. Dr. Fábio José Ceron Branco
Membro titular

Prof. Dr. Juan Carlos Claros Garcia
Membro titular

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso”.

RESUMO

SOUSA, C. R M. **Programação da produção utilizando heurísticas no ambiente de máquinas paralelas não relacionadas**. 2020. 67 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Produção) – Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2020.

O problema de programação da produção em ambiente de máquinas paralelas não relacionadas é recorrente em empresas manufatureiras com máquinas que podem produzir os mesmos produtos em tempos distintos e não relacionados. São utilizados critérios de otimização para selecionar a melhor programação, entretanto, o método de programação matemática, o qual resulta na melhor solução para o problema, possui um elevado tempo de execução computacional quanto mais tarefas e máquinas estejam envolvidas no problema. Foram desenvolvidos nesse trabalho, iniciando pela análise e adaptação de heurísticas, métodos eficientes para a programação de tarefas visando a minimização da soma do *makespan* e do atraso total, com prioridade para o *makespan*, e realizou-se a comparação dos mesmos com o método da programação matemática. A pesquisa bibliográfica, parte inicial do trabalho, contribuiu para a modelagem matemática do método de programação inteira mista que foi implementado no *software* Lingo. Foram desenvolvidas duas heurísticas construtivas e três de melhoria resultando em doze métodos distintos pela combinação das heurísticas. Os métodos heurísticos puderam ser comparados ao modelo matemático para instâncias pequenas. Já para instâncias maiores o modelo matemático não foi capaz de encontrar uma solução no tempo limite estipulado devido ao aumento na complexidade do problema. Dentre os métodos heurísticos, a combinação resultante da Heurística construtiva 1, Heurística de melhoria 1 e Heurística de melhoria 3 obteve 67% dos sucessos possíveis, sendo assim o método mais satisfatório.

Palavras-chave: Programação de tarefas. Máquinas paralelas não relacionadas. Heurísticas. Critérios de otimização. Minimização do *makespan*.

ABSTRACT

SOUSA, C. R. M. **Production scheduling using heuristics in the non-related parallel machines environment.** 2020. 67 p. Work of Conclusion Course (Graduation in Production Engineering) - Federal Technology University - Paraná. Ponta Grossa, 2020.

The problem of scheduling production in an unrelated parallel machine environment is recurrent in manufacturing companies with machines that can produce the same products at different and unrelated times. Optimization criteria are used to select the best programming, however, the mathematical programming method, which results in the best solution to the problem, has a high computational execution time the more tasks and machines are involved in the problem. Efficient methods for scheduling tasks were developed in this work, starting with the analysis and adaptation of heuristics, aiming at minimizing the sum of the *makespan* and the total delay, with priority for the *makespan*, and the comparison was made with the method of mathematical programming. The bibliographic research, an initial part of the work, contributed to the mathematical modeling of the mixed integer programming method that was implemented in the Lingo software. Two constructive heuristics and three improvement heuristics were developed, resulting in twelve different methods by combining the heuristics. The heuristic methods could be compared to the mathematical model for small instances. For larger instances the mathematical model was not able to find a solution within the stipulated time limit due to the increase in the complexity of the problem. Among the heuristic methods, the resulting combination of Constructive Heuristic 1, Improvement Heuristic 1 and Improvement Heuristic 3 obtained 67% of possible successes, thus being the most satisfactory method.

Keywords: Scheduling. Unrelated parallel machines. Heuristics. Optimization criteria. Makespan minimization.

LISTA DE FIGURAS

Figura 1 - Modelo de transformação	17
Figura 2 - Decisões do sequenciamento de produção	20
Figura 3 - Gráfico de Gantt para máquinas paralelas não relacionadas	22
Figura 4 - Matriz de operações e tempos de processamento em <i>job shop</i>	23
Figura 5 - Gráfico de Gantt para <i>flow shop</i>	24
Figura 6 - Exemplo para ambiente de máquinas paralelas não relacionadas	27
Figura 7 - Exemplo de utilização da tarefa 0	29
Figura 8 - Classificação da pesquisa	33
Figura 9 - Etapas de desenvolvimento	34
Figura 10 - Pseudocódigo da HC1	42
Figura 11 - Exemplo de alocação das tarefas de acordo com a HC1	42
Figura 12 - Gráfico de Gantt para a solução da HC1	43
Figura 13 - Pseudocódigo da HC2	43
Figura 14 - Gráfico de Gantt para a solução da HC2	44
Figura 15 - Pseudocódigo da HM1	45
Figura 16 - Gráfico de Gantt para a solução da HM1	45
Figura 17 - Pseudocódigo da HM2	46
Figura 18 - Gráfico de Gantt para a solução da HM2	46
Figura 19 - Pseudocódigo da HM3	47
Figura 20 - Gráfico de Gantt para a solução de HM3	48
Figura 21 - Funcionamento da HM3 com solução de início da HC1	53

LISTA DE QUADROS

Quadro 1 - Relação de dados contidos em uma instância	39
Quadro 2 - Combinação e sequência das heurísticas	50
Quadro 3 - Resultados para o makespan em UT	51
Quadro 4 - Comparação entre os resultados de HC1 e HC2	52
Quadro 5 - Comparação de melhoria das HMs em HC1 para o <i>makespan</i>	54
Quadro 6 - Comparação de melhoria das HMs em HC2 para o <i>makespan</i>	55
Quadro 7 - Resultados para o atraso total em UT	56
Quadro 8 - Comparação de melhoria das HMs em HC1 para o atraso total	57
Quadro 9 - Comparação de melhoria das HMs em HC2 para o atraso total	58
Quadro 10 - Dados comparativos entre os resultados das heurísticas e do modelo matemático	59

LISTA DE TABELAS

Tabela 1 - Instância para máquinas paralelas não relacionadas.....	22
Tabela 2 - Relação de instâncias utilizadas	40
Tabela 3 - Resultados para o modelo matemático	49
Tabela 4 - Relação de sucessos para as combinações heurísticas	61

LISTA DE SIGLAS

HC1	Heurística Construtiva 1
HC2	Heurística Construtiva 2
HM1	Heurística de Melhoria 1
HM2	Heurística de Melhoria 2
HM3	Heurística de Melhoria 3
IC	Índice Crítico
IP	Índice de Prioridade
MDE	Menor Data de Entrega
MTP	Menor Tempo de Processamento
PCP	Planejamento e Controle da Produção
PEPS	Primeira que Entra Primeira que Sai
PO	Pesquisa Operacional
UT	Unidade de Tempo

SUMÁRIO

1 INTRODUÇÃO	11
1.1 PROBLEMA	12
1.2 JUSTIFICATIVA	12
1.3 OBJETIVOS	13
1.3.1 Objetivo Geral	14
1.3.2 Objetivos Específicos	14
1.4 DELIMITAÇÃO DO TEMA	14
1.5 ORGANIZAÇÃO DO TRABALHO	15
2 REVISÃO BIBLIOGRÁFICA	16
2.1 ETAPAS DO SISTEMA PRODUTIVO	16
2.2 PCP	17
2.3 PCP E PO	18
2.4 PROGRAMAÇÃO DE TAREFAS	20
2.5 AMBIENTE DE PRODUÇÃO	21
2.5.1 Máquina Única	21
2.5.2 Máquinas paralelas	21
2.5.3 <i>Job Shop</i>	23
2.5.4 <i>Flow Shop</i>	24
2.6 CRITÉRIOS DE OTIMIZAÇÃO	24
2.6.1 <i>Flow Time</i>	25
2.6.2 Atraso	25
2.6.3 Adiantamento	26
2.6.4 <i>Makespan</i>	26
2.7 <i>MAKESPAN</i> EM MÁQUINAS PARALELAS NÃO RELACIONADAS	27
2.8 A IMPORTÂNCIA DAS HEURÍSTICAS	30
2.9 REGRAS DE PRIORIDADE PARA EXECUÇÃO DE TAREFAS	31
2.10 HEURÍSTICAS DE MELHORAMENTO	32
3 METODOLOGIA	33
3.1 CLASSIFICAÇÃO DA PESQUISA	33
3.2 ETAPAS DE DESENVOLVIMENTO	34
3.2.1 Revisão Bibliográfica	35
3.2.2 Identificação dos Critérios de Otimização e Regras de Prioridade	35
3.2.3 Obtenção dos Dados	35
3.2.4 Levantamento e Aplicação Computacional do Modelo Matemático para o Problema	36
3.2.5 Levantamento de Heurísticas para a Resolução do Problema	36
3.2.6 Implementação Computacional das Heurísticas	37
3.2.7 Comparação dos Métodos Utilizados	37
4 DESENVOLVIMENTO	38

4.1	OBTENÇÃO DOS DADOS	38
4.2	IMPLEMENTAÇÃO DO MODELO MATEMÁTICO	41
4.3	MÉTODOS HEURÍSTICOS	41
4.3.1	Heurística Construtiva 1	41
4.3.2	Heurística Construtiva 2	43
4.3.3	Heurística de Melhoria 1	44
4.3.4	Heurística de Melhoria 2	46
4.3.5	Heurística de Melhoria 3	47
5	RESULTADOS	49
5.1	RESULTADOS DO MODELO MATEMÁTICO	49
5.2	RESULTADOS DAS HEURÍSTICAS	50
5.2.1	Resultados para o Critério <i>Makespan</i>	51
5.2.2	Resultados para o Critério Atraso Total	55
5.2.3	Análise Comparativa dos Resultados Entre as Heurísticas e o Modelo Matemático	59
5.2.4	Análise Comparativa Entre as Combinações Heurísticas	60
6	CONCLUSÕES	62
	REFERÊNCIAS.....	64

1 INTRODUÇÃO

As decisões tomadas na área do Planejamento e Controle da Produção (PCP) classificam-se em três níveis de acordo com o horizonte de tempo em que são aplicadas: o estratégico, políticas de longo prazo; o tático, planos de médio prazo para a produção; e o operacional, programação de curto prazo, com programação e controle da produção (OLIVEIRA, 2014).

Dentre as atribuições da programação da produção tem-se a alocação e o sequenciamento de tarefas. Segundo Etcheverry (2012), a programação de tarefas ou sequenciamento de tarefas possui fundamental importância para o bom desempenho operacional e financeiro das empresas. Ainda de acordo com Etcheverry (2012), uma programação de tarefas eficiente propicia um diferencial competitivo para as empresas, razão pela qual o tema é objeto de estudo de variados trabalhos na literatura.

De acordo com Tubino (2007), a alocação de tarefas está relacionada às variáveis tempo de preparação, taxa de produção, etc. Já o sequenciamento de tarefas envolve tempo de operação, nível de estoque, entre outros.

Taha (2008) afirma que para a resolução de problemas da produção as alternativas devem ser avaliadas de acordo com critérios de otimização e restrições. Um dos critérios de otimização mais utilizado é a minimização da duração total da programação ou *makespan*. Também se pode destacar a importância da minimização do atraso e do adiantamento de tarefas (HARTMANN; BRISKORN, 2010).

De acordo com Pfund, Fowler e Gupta (2004), com os variados ambientes de manufatura tem-se variados equipamentos similares, porém, com diferentes características de desempenho. Conseqüentemente os ambientes geralmente contam com máquinas paralelas que podem ser ou não relacionadas.

Quando os critérios de otimização são aplicados ao ambiente de máquinas paralelas não relacionadas para a programação de tarefas o problema é, normalmente, considerado NP-difícil. O termo NP-difícil refere-se à complexidade do problema, ou seja, o tempo de execução computacional dos modelos para obtenção da solução ótima cresce exponencialmente conforme o número de máquinas ou tarefas aumenta (ETCHEVERRY, 2012). Assim, o tempo de execução dos modelos

matemáticos em *softwares* pode comprometer a tomada de decisão ao não entregá-la em tempo hábil, por exemplo.

A utilização da programação inteira não é prática para problemas de minimização do *makespan*. Por isso são encontrados diversos métodos que oferecem uma solução factível para a programação de tarefas (PFUND; FOWLER; GUPTA, 2004).

Em casos nos quais a viabilidade para encontrar a solução ótima não é satisfatória é necessário o uso de métodos alternativos, como as heurísticas, para se ter uma boa solução para o problema (TAHA, 2008).

No presente contexto, este trabalho desenvolveu heurísticas para a alocação e ordenação de tarefas em máquinas paralelas não relacionadas de modo a minimizar a soma do *makespan* e dos atrasos das tarefas, com prioridade para o *makespan*.

1.1 PROBLEMA

Como desenvolver métodos heurísticos para a programação de tarefas em um ambiente de máquinas paralelas não relacionadas objetivando a minimização da soma do *makespan* e do atraso das tarefas? Sendo o *makespan* o critério prioritário, ou seja, trata-se de um problema multiobjectivo hierarquizado.

1.2 JUSTIFICATIVA

A otimização da utilização dos recursos se dá por meio de um bom sequenciamento das ordens de produção realizado na programação da produção (SLACK; CHAMBERS; JOHNSTON, 2002).

De acordo com Tubino (2007) a programação de tarefas possui relação direta com o estoque, uma vez que as ordens de produção são geradas de acordo com a política de estoque da organização.

O atraso e adiantamento de tarefas podem gerar penalidades para a empresa na forma de multas e/ou má influência na confiabilidade (RONCONI; KAWAMURA, 2010).

Ao minimizar o *makespan* é assegurado um bom balanço em relação ao tempo de utilização dos recursos, por isso, em se tratando do ambiente de máquinas paralelas não relacionadas, a minimização do *makespan* é um critério com considerável importância (PINEDO, 2008).

Observando-se os critérios de otimização, o ambiente produtivo e o esforço computacional elevado para a resolução desse tipo de problema, por ser classificado como NP-difícil, o presente trabalho buscará desenvolver métodos heurísticos capazes de oferecer soluções factíveis e satisfatórias em tempo reduzido em relação aos métodos exatos.

De acordo com Pfund, Fowler e Gupta (2004), no ambiente manufatureiro, são raras as ocasiões onde somente um objetivo é considerado. Além do crescimento das despesas fabris quanto ao espaço ocupado e maquinário utilizado, se tem também a necessidade de atender os requisitos do cliente. Daí se dá a importância de relacionar diferentes critérios para a obtenção de uma solução que atenda melhor à empresa.

A minimização conjunta do *makespan* e do atraso total proporciona um bom equilíbrio na utilização dos recursos, conforme discutido por Pinedo (2008), sem descuidar do cliente no que se refere ao prazo de entrega das tarefas, resultando em maior confiabilidade do cliente em relação à empresa (RONCONI; KAWAMURA, 2010).

1.3 OBJETIVOS

Os objetivos se dividem em geral e específicos.

1.3.1 Objetivo Geral

Desenvolver métodos heurísticos eficientes para o problema de máquinas paralelas não relacionadas com critérios de minimização do *makespan* e minimização do atraso total.

1.3.2 Objetivos Específicos

- i. Realizar a revisão de literatura para fundamentar teoricamente o trabalho;
- ii. Adaptar instâncias referentes ao ambiente de máquinas paralelas não relacionadas;
- iii. Implementar em *software* o modelo matemático para a solução exata do problema;
- iv. Analisar métodos heurísticos existentes para a resolução do problema estudado;
- v. Implementar heurísticas construtivas e de melhoria para o problema em *software*;
- vi. Avaliar as heurísticas desenvolvidas comparando-as ao método exato em relação aos critérios e ao tempo de execução computacional.

1.4 DELIMITAÇÃO DO TEMA

O presente trabalho está limitado ao estudo de métodos heurísticos para a programação de tarefas no ambiente de máquinas paralelas não relacionadas com os critérios de otimização minimização do *makespan* e minimização do atraso total, para a resolução do problema e comparação dos mesmos com a solução encontrada pelo método matemático.

1.5 ORGANIZAÇÃO DO TRABALHO

O presente trabalho de conclusão de curso está organizado em seis capítulos.

O Capítulo 1 traz a contextualização do tema, a definição do tema, a justificativa do trabalho, os objetivos geral e específicos e a delimitação da pesquisa.

No Capítulo 2 é apresentada a revisão bibliográfica para a construção do trabalho contendo os temas: etapas do sistema produtivo, PCP, PCP e Pesquisa operacional (PO), programação de tarefas, ambiente de produção, critérios de otimização, *makespan* em máquinas paralelas não relacionadas, a importância das heurísticas, regras de prioridade para a execução de tarefas e heurísticas de melhoramento.

O Capítulo 3 traz a metodologia contendo a classificação da pesquisa e as etapas de desenvolvimento do trabalho.

O Capítulo 4 contém o desenvolvimento do trabalho, apresenta como foi realizada a obtenção dos dados e a elaboração dos métodos heurísticos desenvolvidos. Já no Capítulo 5 são apresentados os resultados obtidos pela aplicação dos métodos desenvolvidos e do modelo de programação matemática, sucedido pela comparação dos métodos entre si e para com o método de programação matemática.

Por fim, no Capítulo 6, são discutidas as conclusões do trabalho desenvolvido, contendo a retomada dos objetivos, resultados e considerações finais.

2 REVISÃO BIBLIOGRÁFICA

A revisão bibliográfica pertinente ao presente trabalho está descrita no presente capítulo.

Os tópicos abordados são: etapas do sistema produtivo, PCP, PCP e PO, programação de tarefas, ambiente de produção, critérios de otimização, *makespan* em máquinas paralelas não relacionadas, a importância das heurísticas, regras de prioridade para a execução de tarefas e heurísticas de melhoramento.

2.1 ETAPAS DO SISTEMA PRODUTIVO

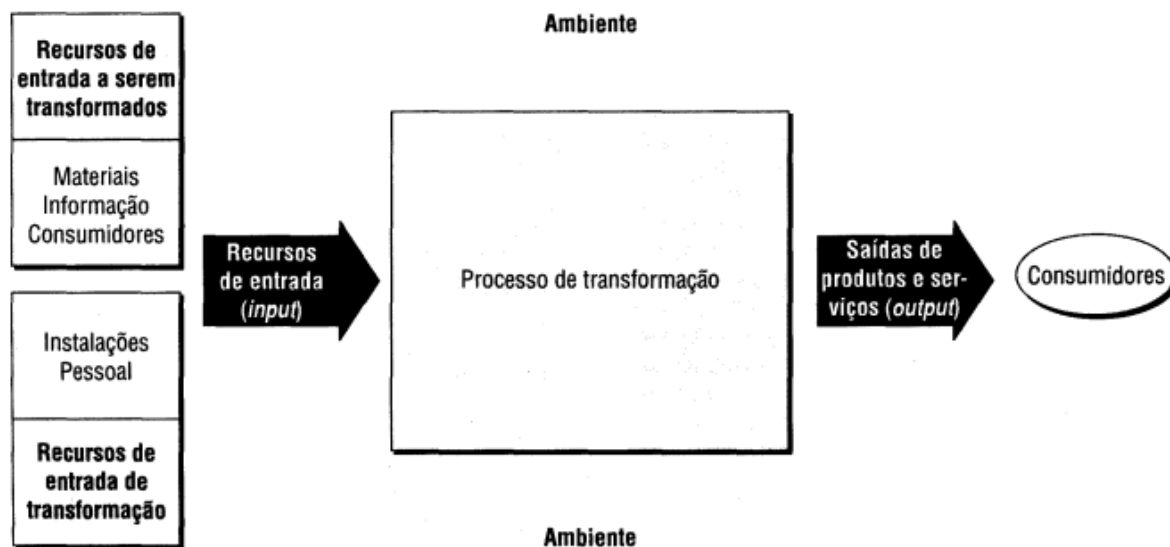
De acordo com Tubino (2007), as empresas são caracterizadas como um sistema que transforma, por meio de etapas, entradas (insumos) em saídas (produtos), os quais possuem valor para o cliente.

Slack, Chambers e Johnston (2002) definem o processo de produção como sendo o que executa um processo de transformação. A transformação é a utilização de recursos para modificar a condição ou estado de algo para produzir *outputs*.

Para Peinado e Graeml (2004) “a função produção está focada na transformação de certos insumos em algum resultado desejado”. O processo envolve recursos a serem transformados (insumos) e recursos transformadores (funcionários, instalações, máquinas, entre outros).

O modelo apresentado na Figura 1 exemplifica o sistema produtivo.

Figura 1 - Modelo de transformação



Fonte: Slack, Chambers e Johnston (2002)

O modelo apresentado é amplamente difundido na literatura sobre o tema. Ainda segundo Slack, Chambers e Johnston (2002), as operações podem produzir somente bens ou somente serviços, entretanto, a maioria resulta em ambos. Nesse trabalho será abordado o processo de produção que resulta em bens ou produtos.

De acordo com Tubino (2007) para que as etapas do sistema produtivo possam ser executadas de maneira satisfatória o mesmo precisa ser pensado em relação aos prazos. Daí se dá a importância do PCP para o sistema produtivo.

2.2 PCP

Pasquini (2016) define o PCP como “o departamento da organização que determina o que vai ser produzido, quando vai ser produzido, quanto vai ser produzido, onde vai ser produzido e como vai ser produzido.”

Trierweiller et al. (2008) dizem que:

O PCP pode ser definido como a coordenação dos departamentos de uma organização com foco voltado ao atendimento da demanda de vendas ou programação da produção, de modo que as mesmas sejam atendidas nos prazos e quantidades exigidas.

Para Mesquita e Castro (2008) um PCP eficiente é aquele que atinge seus objetivos, sendo eles: redução dos *lead times* de produção, dos custos de produção e estoque e cumprimento de prazos.

Nota-se que as definições citadas possuem em comum a importância do cumprimento de prazos. Segundo Oliveira (2014), os níveis do PCP podem ser caracterizados de acordo com os prazos a serem atendidos:

- Nível Estratégico: políticas de longo prazo (Planejamento estratégico da produção).
- Nível Tático: planos de médio prazo (Plano mestre da produção).
- Nível Operacional: planos de curto prazo (Programação da produção).

Segundo Slack, Chambers e Johnston (2002), no que se refere ao horizonte destes prazos, tem-se que o longo prazo é medido em meses, o médio prazo em semanas e o curto prazo em dias.

A programação da produção “se encarrega de fazer o sequenciamento das ordens emitidas, de forma a otimizar a utilização dos recursos” (SLACK, CHAMBERS e JOHNSTON, 2002).

2.3 PCP E PO

Arenales et al (2011) dizem que:

PO é a aplicação de métodos científicos a problema complexos para auxiliar no processo de tomada de decisões, tais como projetar, planejar e operar sistemas em situações que requerem alocações eficientes de recursos escassos.

Para a resolução do problema, é necessária a observação cuidadosa para que o mesmo seja formulado corretamente e, em seguida, a coleta de dados pertinentes (HILLIER; LIEBERMAN, 2006).

Segundo Taha (2008), três perguntas devem ser respondidas para a resolução do problema:

1. Quais são as alternativas para a decisão?

2. Quais as restrições para a tomada de decisão?
3. Qual o critério objetivo para avaliar as alternativas?

Posterior à coleta de dados se tem a construção do modelo que representa o problema. Esse modelo é composto por um sistema de equações e de expressões matemáticas, onde estão relacionadas à função objetivo, às variáveis de decisão e às restrições do problema, representadas por desigualdades ou equações (HILLIER; LIEBERMAN, 2006).

A solução para o modelo é determinada por algoritmos. O algoritmo possui regras de cálculo fixas que são realizadas múltiplas vezes e a cada iteração a solução se aproxima da ideal ou ótima (TAHA, 2008).

Os campos em que a PO tem sido aplicada são vários, como transportes, manufatura, telecomunicações, construção, planejamento financeiro, assistência médica e serviços públicos (HILLIER; LIEBERMAN, 2006).

De acordo com Pantuza (2016), com a competição do mercado mundial cada vez mais forte, as empresas necessitam reduzir seus custos. E uma forma de atingir esse objetivo pelo uso aperfeiçoado dos fatores de produção. Daí se tem a utilização da PO para a resolução desse tipo de problema.

Dentre os tipos de problemas de planejamento e controle que podem ser resolvidos com a utilização da PO tem-se: otimizar operações de abastecimento e distribuição de produtos para diminuir o custo, realizar o mix dos componentes de um produto de forma otimizada e atendendo as exigências de qualidade de objetivando o aumento do lucro, otimizar o desenho de rotas de transporte rodoviário de modo a diminuir a distância percorrida, otimizar o número de funcionários disponíveis por turno para diminuir os custos com mão de obra (HILLIER; LIEBERMAN, 2006).

Nesse trabalho foi abordada a utilização da PO para a programação da produção, na qual as tarefas devem ser alocadas em máquinas seguindo um critério de otimização.

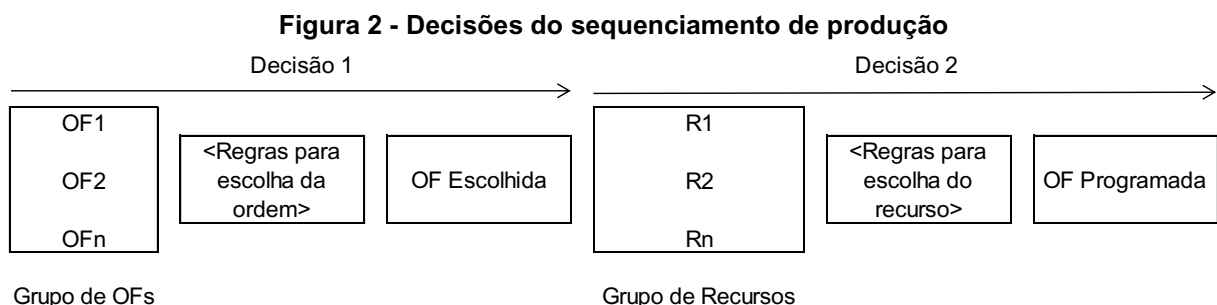
2.4 PROGRAMAÇÃO DE TAREFAS

De acordo com Tubino (2007), a programação de tarefas se dá baseada em duas questões principais:

1. A escolha da ordem a ser processada dentre uma lista de ordens de fabricação planejadas e;
2. A escolha do recurso a ser usado dentre uma lista de recursos disponíveis no centro de trabalho.

Ainda segundo Tubino (2007), a escolha da ordem a ser processada é feita com base em características do item a ser processado, como tempo de operação, importância para o cliente e nível do estoque. Para a escolha do recurso, as variáveis que influenciam na tomada de decisão são: tempo de *setup*, taxa de produção, capacidade disponível e tempo disponível.

A Figura 2 simplifica o problema de tomada de decisão envolvendo ordens de fabricação e recursos, os termos citados estão representados por “OF” e “R”, respectivamente.



Fonte: Adaptado de Tubino (2007)

Segundo Penna et al (2012), problemas de programação de tarefas têm sido acrescidos de diferentes critérios de otimização, como a minimização do atraso e do adiantamento devido ao crescimento da filosofia *just-in-time*.

Pinedo (2008) atribui as tarefas, os recursos e o tempo disponível como sendo as variáveis trabalhadas pela programação de tarefas e seu objetivo é otimizar determinado critério.

A programação de tarefas está também relacionada à administração do estoque, uma vez que, após definida a política de estoque é que são geradas as necessidades de compras, fabricação e montagem dos produtos (TUBINO, 2007).

De acordo com Arenales et al (2011), a qualidade de um programa de produção pode ser avaliada de acordo com medidas de desempenho. As medidas de desempenho mais utilizadas são: *makespan*, tempo de fluxo total, atraso máximo, atraso total e número de tarefas atrasadas.

2.5 AMBIENTE DE PRODUÇÃO

Com a automatização do manuseio de materiais e a inserção de máquinas de controle numérico, objetivando a produção automatizada em grande escala, o ambiente de produção adquiriu novas características (ARENALES et al, 2011).

Dentre os ambientes de produção se destacam: máquina única, máquinas paralelas, *flow shop* e *job shop*. A seguir foram expostos de forma pormenorizada os ambientes citados.

2.5.1 Máquina Única

Segundo Caetano et al (2016), o ambiente de máquina única é caracterizado por possuir apenas uma máquina na qual as ordens são processadas, devem ser observados o tempo de produção, data de entrega, data de liberação, etc.

Pinedo (2008) destaca a importância do modelo de máquina única ao dizer que os resultados encontrados se aplicam não só ao modelo de máquina única em si, mas também fornecem apoio para tratar de ambientes de máquinas mais complicados.

2.5.2 Máquinas paralelas

Um ambiente de máquinas paralelas consiste de máquinas similares ou não relacionadas utilizadas na programação de tarefas realizadas em simultâneo (EZUGWU, 2019).

Para Arenales et al (2011), ambientes de máquinas paralelas são divididos em três categorias: idênticas, uniformes e não relacionadas. Máquinas idênticas são aquelas que possuem o mesmo tempo de processamento e mesmo tempo de setup.

Máquinas uniformes ocorrem, por exemplo, quando uma máquina antiga possui tempo de processamento e tempo de setup proporcionais aos de uma máquina nova. Já máquinas não relacionadas possuem tempo de processamento e de setup sem nenhuma relação lógica.

A Tabela 1 apresenta um cenário de máquinas paralelas não relacionadas e contem as tarefas a serem processadas bem como o tempo de processamento das tarefas em cada uma das máquinas disponíveis.

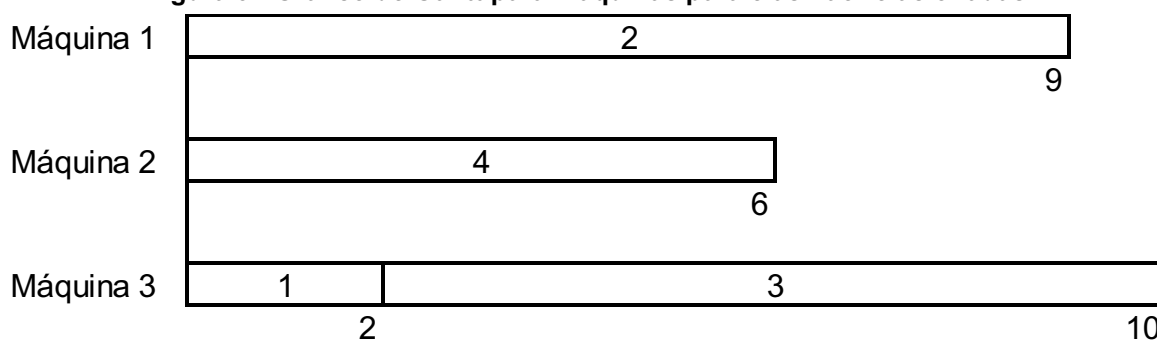
Tabela 1 – Instância para máquinas paralelas não relacionadas

Tarefa	M1	M2	M3
1	1	7	2
2	9	5	8
3	4	6	8
4	7	6	5

Fonte: autoria própria

Na Figura 3 é apresentado um gráfico de Gantt contendo uma das resoluções para o problema da Tabela 1.

Figura 3 - Gráfico de Gantt para máquinas paralelas não relacionadas



Fonte: Adaptado de Celini (2017)

O gráfico apresentado possui quatro tarefas e três máquinas. Os números abaixo das barras de tarefas nas máquinas representam os tempos utilizados para processar as tarefas. Cada tarefa pode ser realizada em qualquer uma das máquinas, mas o tempo de processamento de uma mesma tarefa pode variar de máquina para máquina.

Segundo Ezugwu (2019), o modelo de máquinas paralelas em sido amplamente estudo devido a sua aplicabilidade em diversos setores como linhas de produção, gerenciamento de hospitais (horários de médicos e enfermeiras),

gerenciamento de tarefas para computadores com múltiplos processadores, entre outros.

2.5.3 Job Shop

De acordo com Arenales et al (2011), um ambiente *job shop* tradicional é caracterizado por possuir n tarefas para serem processadas em m máquinas, sendo que cada tarefa possui uma ordem correta em relação às máquinas pela qual a mesma deve passar.

Na Figura 4 é apresentado um exemplo, com cinco tarefas e três máquinas, para o ambiente de *job shop*, no qual a matriz O possui a ordem de máquinas pelas quais cada tarefa deve passar. Já a matriz P contém os tempos de processamento das tarefas em cada máquina.

Figura 4 - Matriz de operações e tempos de processamento em *job shop*

$$\mathbf{O} = \begin{bmatrix} 2 & 1 & 3 \\ 1 & 2 & 3 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \\ 3 & 1 & 2 \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} 5 & 7 & 10 \\ 9 & 5 & 3 \\ 5 & 8 & 2 \\ 2 & 7 & 4 \\ 8 & 8 & 8 \end{bmatrix}$$

Fonte: Arenales et al (2011)

No exemplo tem-se que a primeira tarefa deve passar pelas máquinas 2, 1 e 3, nessa exata ordem. E possui tempos de processamento 5, 7 e 10 unidades de tempo nas máquinas 2, 1 e 3 respectivamente.

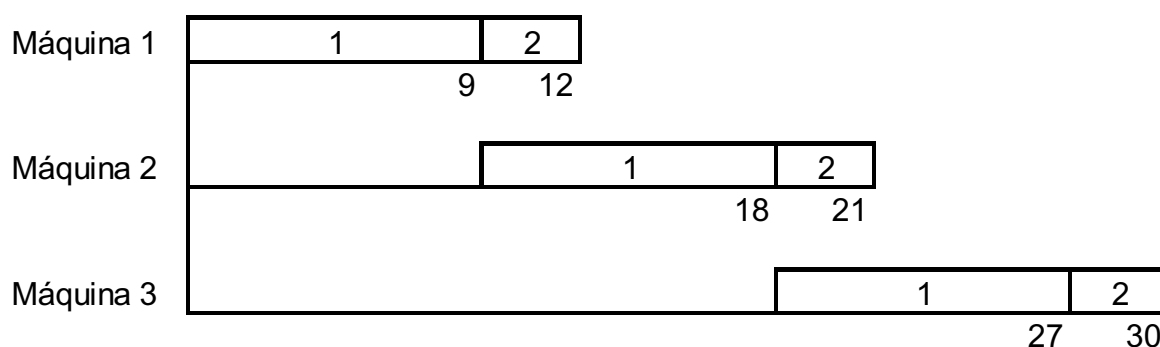
Morales e Ronconi (2015) afirmam que o modelo *job shop* clássico representa o modelo operacional de pequenas empresas manufatureiras, com grande variedade de produtos e pouco volume de produção. Também segue a premissa de que cada produto é originado de um pedido realizado pelo cliente com características diferenciadas para o produto.

2.5.4 Flow Shop

Quando o ambiente conta com n tarefas para serem processadas em m máquinas e todas as tarefas possuem o mesmo roteiro em todas as máquinas tem-se o ambiente *flow shop*, que nada mais é do que um caso particular do *job shop* (ARENALES et al, 2011).

A Figura 5 exemplifica o ambiente *flow shop*.

Figura 5 - Gráfico de Gantt para *flow shop*



Fonte: Adaptado de Celini (2017)

A Figura 5 apresenta um ambiente *flow shop* com duas tarefas e três máquinas. Nota-se que todas as tarefas seguem a mesma ordem de máquinas, sendo: máquina 1, máquina 2 e máquina 3, respectivamente.

Segundo Nagano, Moccellini e Lorena (2004), as medidas usuais para a solução do problema *flow shop* “referem-se à minimização da duração total da programação (*makespan*) e a minimização do tempo médio de fluxo”.

2.6 CRITÉRIOS DE OTIMIZAÇÃO

Segundo Hillier e Lieberman (2006), “determinar os objetivos apropriados é um aspecto muito importante na definição de um problema” e, para problemas de sequenciamento de tarefas e alocação, esses objetivos são traduzidos em critérios de otimização.

São variados os critérios de otimização e, os mesmos devem ser escolhidos de acordo com a natureza do problema e observando o ambiente em que a produção ocorre (CELINI, 2017).

Levando em consideração esses fatores tem-se uma ampla produção de trabalhos utilizando diferentes critérios de otimização, como por exemplo: minimização dos custos provenientes do adiantamento e atraso de tarefas (PENNA et al, 2012), otimização da alocação de trabalhadores (PANTUZA, 2016) e otimização do sequenciamento de tarefas objetivando diminuir o atraso e o consumo de energia das máquinas (LI et al, 2016).

Foram discutidos na sequência os critérios mais comumente utilizados na literatura sobre o tema.

2.6.1 *Flow Time*

Segundo Arenales et al (2011), o tempo de fluxo total corresponde ao tempo de conclusão de todas as tarefas e o mesmo possui relação com o estoque em processamento.

De acordo com Celini (2017), a utilização do critério de otimização visando minimizar o *flow time* auxilia as empresas a controlar o seu estoque em processo e, por consequência, administrar melhor seu espaço físico em se tratando do espaço necessário para alocar o estoque em processo.

Analisando a Figura 5 tem-se que o *flow time* da produção foi de 57 unidades de tempo, resultado proveniente da soma dos instantes de término das duas tarefas na última máquina, sendo 27 para a tarefa 1 e 30 para a tarefa 2.

2.6.2 Atraso

Tarefas atrasadas são aquelas que são concluídas após a data estabelecida para a entrega.

As mesmas podem incorrer em penalizações (ARENALES et al, 2011), além de influenciar negativamente a confiabilidade da empresa quanto a prazos e entregas (RONCONI; KAWAMURA, 2010). Assim, é necessário que o atraso seja minimizado na alocação e sequenciamento de tarefas.

Dentre os casos particulares do critério atraso, objetivando a minimização, podem-se citar o atraso total, o número de tarefas atrasadas e o atraso máximo. O primeiro é resultado da soma dos atrasos de todas as tarefas. O número de tarefas atrasadas indica a quantidade de tarefas que foram entregues após o prazo. Já o atraso máximo é relacionado ao valor máximo de atraso encontrado.

Ao atribuir valor monetário para unidades de tempo de atraso das tarefas se tem a priorização das tarefas que resultariam em uma maior multa (CELINI, 2017).

2.6.3 Adiantamento

Tarefas adiantadas são aquelas que são finalizadas antes da data prevista para a sua entrega. Segundo Arenales et al (2011), os custos de estoque estão diretamente ligados ao adiantamento de tarefas, uma vez que as mesmas incorrem em um maior custo de manutenção do estoque.

O adiantamento pode também afetar a qualidade do produto (RONCONI; KAWAMURA, 2010). Na indústria alimentícia, por exemplo, um cliente pode recusar o recebimento do produto uma vez que o mesmo foi produzido antes da data estabelecida e seu prazo de validade já está reduzido.

Com o crescimento da filosofia *just in time* a utilização dos critérios de minimização de atraso e de adiantamento tem sido aplicados em conjunto por muitos autores (RONCONI; KAWAMURA, 2010).

2.6.4 *Makespan*

O *makespan* é resultado do tempo de processamento necessário para que todas as tarefas sejam concluídas e pode ser utilizado para avaliar a utilização do sistema (ARENALES et al, 2011).

De acordo com Lin, Pfund e Fowler (2011), a minimização do *makespan* faz com que as tarefas sejam concluídas o mais rápido possível e também corroboram para que a utilização das máquinas seja maximizada.

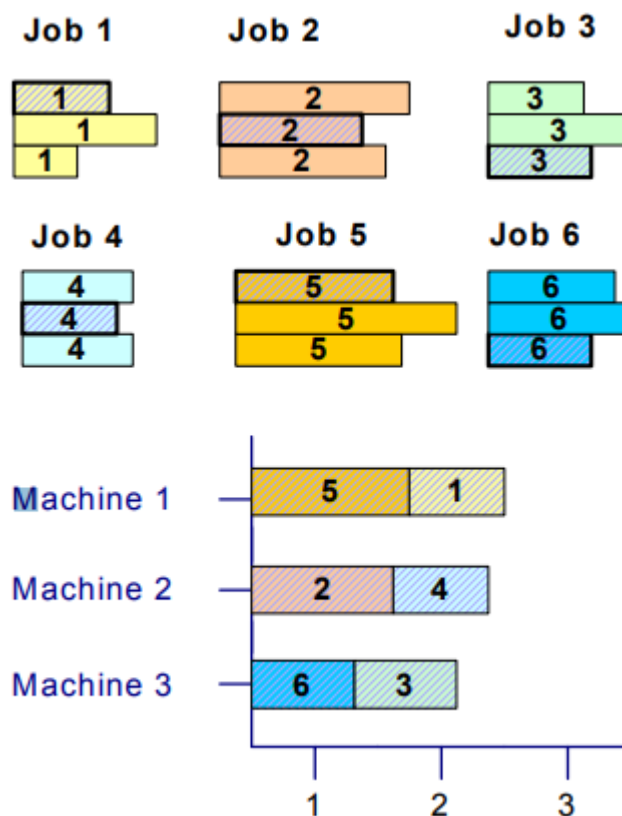
2.7 MAKESPAN EM MÁQUINAS PARALELAS NÃO RELACIONADAS

Segundo Pinedo (2008), são comuns problemas de balanceamento de carga em máquinas paralelas, e a utilização da minimização do *makespan* garante um bom balanceamento.

No ambiente de máquinas paralelas não relacionadas o critério de otimização mais estudado é a minimização do *makespan* (PFUND; FOWLER; GUPTA, 2004).

A Figura 6 apresenta um exemplo de ambiente de máquinas paralelas não relacionadas e em seguida a indicação do *makespan* do processo.

Figura 6 - Exemplo para ambiente de máquinas paralelas não relacionadas



Fonte: Pfund, Fowler e Gupta (2004)

Nesse exemplo têm-se seis tarefas e três máquinas não relacionadas. Cada tarefa foi alocada na máquina que possui o menor tempo de processamento para que a mesma seja processada. A conclusão da tarefa 1 do exemplo apresenta o *makespan* da produção, pois é a última tarefa a ser concluída.

Como o tempo total de processamento em uma máquina independe da ordem em que as tarefas estão sequenciadas, o problema é simplificado para tratar da alocação das tarefas nas máquinas (POTTS, 1985).

O modelo matemático de programação linear apresentado por Potts (1985) adaptado para a minimização conjunta de *makespan* e atraso total, com prioridade para o *makespan*, segue as seguintes premissas: cada tarefa i ($1 \dots n$) deve ser processada sem interrupção em uma das m máquinas; cada máquina só pode processar uma tarefa por vez. Cada tarefa i está disponível para ser iniciada no tempo zero e requer um tempo de processamento positivo p_{ik} se estiver alocada na máquina k .

O objetivo é minimizar o tempo máximo de conclusão das tarefas C_{max} juntamente com a minimização do atraso total das tarefas. O modelo de programação inteira mista é apresentado na sequência.

Parâmetros:

- p_{ik} Tempo de processamento da tarefa i na máquina k ;
- d_i Data de entrega da tarefa i ;
- G Número grande.

Variáveis:

- C_{max} Instante de término da última tarefa (*makespan*);
- C_{ik} Instante de término do processamento da tarefa i na máquina k ;
- x_{ijk} Assume 1 de a tarefa i precede imediatamente a tarefa j na máquina k e 0 caso contrário.

- T_i Atraso da tarefa i ;

Modelo:

$$\text{Minimizar } C_{max} + (\sum_{i=1}^n T_i) / G \quad (1)$$

$$\text{Sujeito a: } \sum_{i=1}^n \sum_{j=0}^n p_{ik} x_{ijk} \leq C_{max} \quad (k=1 \dots m) \quad (2)$$

$$\sum_{k=1}^m \sum_{i=0}^n x_{ijk} = 1 \quad (j=1 \dots n) \quad (3)$$

$$\sum_{j=1}^n x_{0jk} \leq 1 \quad (k=1 \dots m) \quad (4)$$

$$\sum_{i=0; i \neq h}^n x_{ihk} - \sum_{j=0; j \neq h}^n x_{hjk} = 0 \quad (h=1 \dots n)(k=1 \dots m) \quad (5)$$

$$C_{jk} \geq C_{ik} - G + (p_{jk} + G) X_{ijk} \quad (i=0 \dots n)(j=1 \dots n)(k=1 \dots m) \quad (6)$$

$$T_i \geq C_{ik} - d_i \quad (i=1\dots n)(k=1\dots m) \quad (7)$$

$$X_{ijk} = \{0,1\} \quad (i=0\dots n)(j=0\dots n)(k=1\dots m) \quad (8)$$

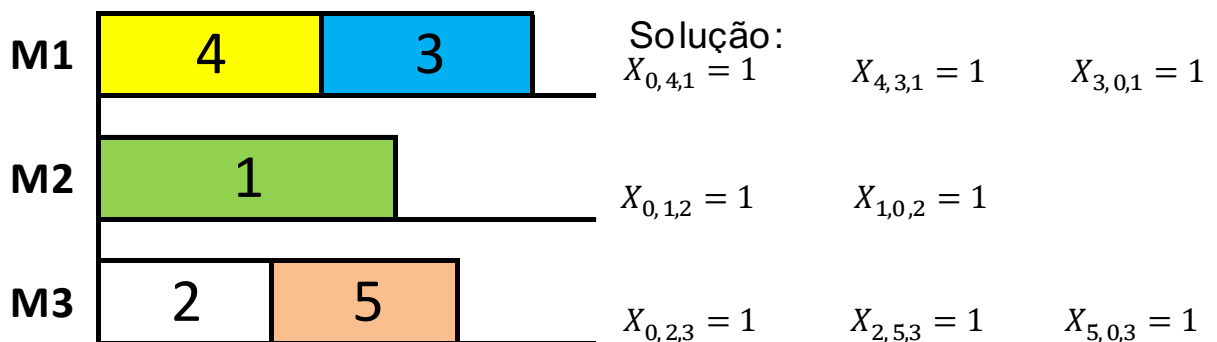
$$C_{ik} \geq 0 \quad (i=0\dots n) \quad (k=1\dots m) \quad (9)$$

$$T_i \geq 0 \quad (i=1\dots n) \quad (10)$$

A função objetivo (1) minimiza o C_{max} e o atraso total dando prioridade ao C_{max} , pois o mesmo assume somente valores inteiros enquanto que o atraso total é dividido por um número grande G , sendo que essa divisão, ao resultar em valor menor que 1, garante que C_{max} não seja prejudicado para uma melhoria no valor do atraso total. A restrição (2) garante que o C_{max} é tão grande quanto o tempo total de processamento em cada máquina. A restrição (3) assegura que cada tarefa deve ter uma única predecessora em uma única máquina. A restrição (4) limita a uma a quantidade de sequências de tarefas em cada máquina. A restrição (5) define que cada tarefa deverá ter somente uma tarefa sucessora, excetuando-se a tarefa 0(zero). A restrição (6) assegura instantes de término coerentes com o tempo de processamento para cada tarefa. A restrição (7) calcula o atraso de cada tarefa e garante a sua não negatividade. As restrições (8), (9) e (10) definem o tipo das variáveis, bem como condições de não negatividade.

No modelo é utilizada a tarefa fictícia 0. Sua função é indicar o início e o fim da sequência de tarefas em cada máquina. A Figura 7, em conjunto com um gráfico de Gantt, exemplifica uma alocação de tarefas e a utilização da tarefa 0 na solução do problema.

Figura 7 - Exemplo de utilização da tarefa 0



Fonte: autoria própria

No exemplo da Figura 7 tem-se cinco tarefas alocadas em três máquinas. Por ser uma tarefa fictícia, a tarefa 0 não aparece no gráfico de Gantt, mas sim na

solução do modelo, na qual nota-se que a mesma precede a tarefa inicial e sucede a tarefa final em todas as máquinas.

2.8 A IMPORTÂNCIA DAS HEURÍSTICAS

Em alguns casos, os modelos de programação de tarefas são muito complexos e a busca pela solução ótima torna-se inviável, por isso utilizam-se heurísticas para encontrar uma solução para o problema. Essa solução pode ser a ótima ou uma boa solução (TAHA, 2008).

A heurística é proposta como uma solução viável, devido à complexidade do problema e ao tempo necessário para a execução de métodos exatos, que não são viáveis do ponto de vista de sistema produtivos reais (MARÍN; GARCÍA; GÓMEZ, 2014).

De acordo com Pfund, Fowler e Gupta (2004), grande parte dos problemas de programação de tarefas no ambiente de máquinas paralelas não relacionadas é considerada NP-difícil, portanto a resolução utilizando programação inteira convencional se torna impraticável para problemas com instâncias grandes.

Assim, quanto maior o número de tarefas e máquinas envolvidas no problema, maiores são as dificuldades para se encontrar a solução ótima devido ao grande número de soluções possíveis (CELINI, 2017).

Diante deste cenário, é comum a utilização de heurísticas para a resolução desse tipo de problema. Marín, García e Gómez (2014) dizem que “no sentido geral, heurísticas são métodos ou procedimentos necessários para executar uma tarefa”.

Um método heurístico encontrará uma solução viável para o problema, mas essa solução pode não ser a ótima para o mesmo. Entretanto, um método heurístico bem elaborado geralmente fornece soluções próximas à solução ideal para o problema (HILLIER; LIEBERMAN, 2006).

Segundo Ruiz e Rodríguez (2010), as heurísticas mais simples são as regras de despacho ou de alocação, com regras simples para a alocação de tarefas em máquinas. Ainda segundo Ruiz e Rodríguez (2010), um grande número de artigos realiza o estudo de heurísticas.

2.9 REGRAS DE PRIORIDADE PARA EXECUÇÃO DE TAREFAS

De acordo com Tubino (2007) as regras de prioridade:

São heurísticas usadas para selecionar, a partir de informações sobre características dos itens ou lotes e/ou sobre o estado do sistema produtivo, qual dos lotes esperando na fila de um grupo de recursos terá prioridade de processamento”.

Ainda segundo Tubino (2007), tempo de processamento e data de entrega são as informações mais relevantes para a tomada de decisão.

As principais regras de prioridade, de acordo com Tubino (2007), são descritas na sequência. Ressalta-se que essas regras têm sua aplicabilidade dependente do ambiente produtivo, pois não existem regras de prioridade que sejam boas para todas as situações (TUBINO, 2007).

Primeira que Entra Primeira que Sai (PEPS): é a regra onde os lotes serão processados de acordo com sua chegada ao recurso.

Menor Tempo de Processamento (MTP): faz com que os lotes sejam processados de acordo com os menores tempos de processamento no recurso.

Menor Data de Entrega (MDE): faz com que os lotes sejam processados de acordo com as menores datas de entrega.

Índice de Prioridade (IP): é a regra na qual os lotes serão processados de acordo com a prioridade atribuída ao cliente ou ao produto.

Índice Crítico (IC): os lotes serão processados de acordo com o menor resultado para $(\text{data de entrega} - \text{data atual}) / \text{Tempo de processamento}$.

Celini (2017) afirma que mais de uma regra podem ser aplicadas para compor uma heurística, sendo que a variação da ordem em que cada regra é aplicada interfere no resultado final.

2.10 HEURÍSTICAS DE MELHORAMENTO

No final da década de 1970 foram trabalhadas as chamadas heurísticas de melhoramento, que consistem em movimentos de reordenação e realocação de tarefas entre máquinas (MULLER; DIAS; ARAÚJO, 2002).

Segundo Penna et al (2012), a heurística de refinamento/melhoria é empregada objetivando encontrar uma solução melhor para o problema e, caso a heurística de melhoramento não a encontre, a solução da heurística construtiva é mantida.

Dentre as aplicações de heurísticas de melhoria podem-se citar problemas de colheita estudado por Junqueira e Morabito (2018) e problemas de corte e empacotamento estudado por Martin et al (2018).

Um procedimento de melhoria bastante aplicado na literatura é o de melhoria local ou busca local. Partindo de uma solução inicial, se faz uma busca nas vizinhanças da solução buscando uma solução experimental melhor do que a inicial. O processo segue até que não seja encontrada nenhuma solução melhor do que a atual nas vizinhanças (HILLIER; LIEBERMAN, 2006).

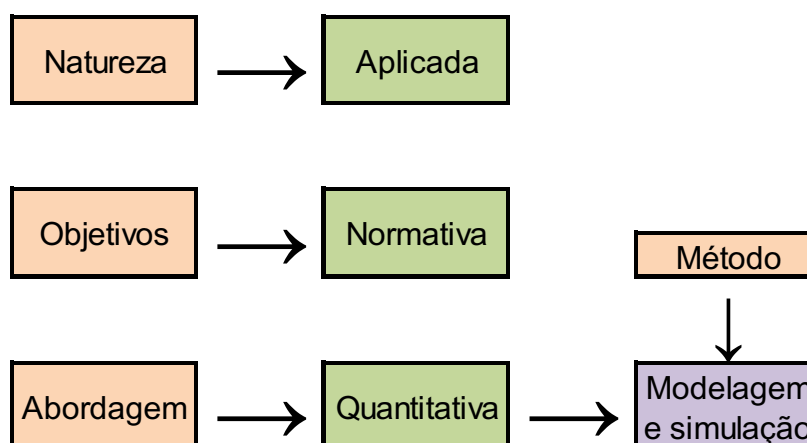
3 METODOLOGIA

No presente capítulo será apresentada a metodologia do trabalho desenvolvido. A mesma compreende a classificação da pesquisa e as etapas de desenvolvimento acompanhada da descrição das mesmas.

3.1 CLASSIFICAÇÃO DA PESQUISA

A Figura 8 contém informações a respeito da classificação da pesquisa.

Figura 8 - Classificação da pesquisa



Fonte: Adaptado de Turrioni e Mello (2012)

Quanto à natureza, a pesquisa pode ser classificada como aplicada, haja vista que a mesma contempla o desenvolvimento de heurísticas que possam ser aplicadas a problemas reais (TURRIONI; MELLO, 2012).

De acordo com Miguel et al (2012), os objetivos caracterizam a pesquisa como normativa, uma vez que são buscadas alternativas para melhorar resultados já presentes na literatura.

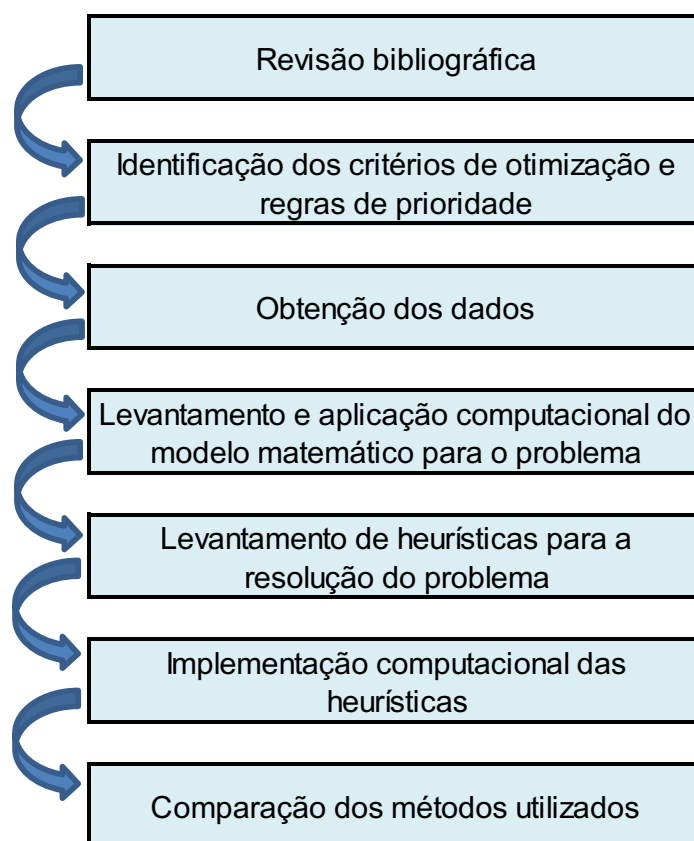
A abordagem da pesquisa pode ser descrita como quantitativa, na qual as informações e resultados do problema são quantificados e analisados (TURRIONI; MELLO, 2012).

De acordo com Bertrand e Fransoo (2002), o método para a resolução do problema utilizado nesse trabalho é o modelagem e simulação. Modelos representam processos operacionais reais e auxiliam na tomada de decisão.

3.2 ETAPAS DE DESENVOLVIMENTO

As etapas para o desenvolvimento do presente trabalho estão dispostas na Figura 9.

Figura 9 - Etapas de desenvolvimento



Fonte: Autoria própria

Cada etapa apresentada no esquema é descrita em detalhes na sequência.

3.2.1 Revisão Bibliográfica

A revisão bibliográfica contempla definições referentes ao tema estudado e traz a fundamentação teórica necessária para a realização do trabalho.

Trabalhos de autores clássicos foram abordados. Tem-se também a incorporação de informações pertinentes ao tema estudado provenientes de artigos científicos publicados em periódicos e congressos.

Os periódicos utilizados foram sondados, por meio da plataforma Sucupira, quanto ao seu qualis na área Engenharias III de acordo com a CAPES e possuem classificação entre A1 e B3 assim, garante-se que os artigos pesquisados possuem relevância para o tema em questão.

As bases de dados Scielo e Scopus foram então utilizadas para encontrar os artigos analisados. Realizou-se a busca por meio da combinação das palavras-chave programação de tarefas, máquinas paralelas não relacionadas, heurísticas, critérios de otimização e minimização do *makespan*. Os trabalhos de artigos foram filtrados em relação ao ano de publicação, compreendendo o período 2009-2020.

Os artigos selecionados foram aqueles que possuem dados relevantes para o tema abordado no trabalho, contemplando também trabalhos com recorrentes citações na literatura.

3.2.2 Identificação dos Critérios de Otimização e Regras de Prioridade

Os critérios de otimização comumente utilizados para a resolução de problemas referentes ao tema estudado estão identificados na revisão bibliográfica.

As regras de prioridade utilizadas são as pertinentes aos critérios de minimização selecionados.

3.2.3 Obtenção dos Dados

Para a implementação do modelo matemático e posterior avaliação das heurísticas são necessários dados referentes ao ambiente de máquinas paralelas não relacionadas, chamados de instâncias.

Essas instâncias contêm informações como a quantidade de máquinas e tarefas a serem executadas, tempo de processamento das tarefas nas máquinas e data de entrega das tarefas.

As instâncias foram obtidas por meio da plataforma *OR-Library*, a qual contém uma coleção de instâncias para problemas de PO, e posteriormente adaptadas.

Na *OR-Library*, foram selecionadas doze instâncias encontradas em “*Other Sources*” no menu “*scheduling*”. A primeira encontra-se nas instâncias referentes à *open shop*, as instâncias restantes referem-se ao sequenciamento *flow shop*. As instâncias foram então adaptadas para o ambiente de máquinas paralelas não relacionadas contendo o número de tarefas, o número de máquinas, tempos de processamento e data de entrega das tarefas. Essa adaptação objetiva garantir a aleatoriedade nos tempos de processamento das tarefas nas máquinas, característica do ambiente de máquinas paralelas não relacionadas.

3.2.4 Levantamento e Aplicação Computacional do Modelo Matemático para o Problema

Com base na revisão bibliográfica foi implementado, utilizando o *software* Lingo 15.0, o modelo matemático aplicando as instâncias que foram verificadas.

Os dados relevantes são o resultado final do método quanto ao critério de minimização escolhido, bem como o tempo de execução computacional do mesmo para cada instância.

3.2.5 Levantamento de Heurísticas para a Resolução do Problema

Foram analisadas heurísticas presentes na literatura para a elaboração das heurísticas construtivas e de melhorias avaliadas.

As heurísticas foram desenvolvidas baseando-se nos critérios de otimização previamente selecionados.

3.2.6 Implementação Computacional das Heurísticas

Após a obtenção dos métodos heurísticos, as etapas para a implementação do método foram transformadas em linguagem de programação do tipo C. Os métodos foram executados no *software* CodeBlocks 17.12 e, juntamente com a solução, foram gerados gráficos de Gantt para ilustrar a configuração dos resultados obtidos.

3.2.7 Comparação dos Métodos Utilizados

Os resultados obtidos pelos métodos heurísticos e modelo matemático foram comparados quanto aos resultados e tempo de execução computacional.

A quantidade de sucessos, ou seja, número de instâncias para as quais determinado método foi o mais eficiente, foi utilizada para avaliar os melhores métodos.

4 DESENVOLVIMENTO

O presente capítulo aborda o desenvolvimento das etapas realizadas para a conclusão do trabalho. As seções que compõem o capítulo são obtenção dos dados, implementação do modelo matemático, métodos heurísticos.

4.1 OBTENÇÃO DOS DADOS

Seguindo o que foi abordado na Subseção 3.2.3, foram obtidas e adaptadas instâncias da plataforma *OR-Library* intituladas “*Scheduling instances*” publicadas por E. Tailard em “*Benchmarks for basic scheduling problems*” no ano de 1993.

As instâncias são caracterizadas pelos seguintes dados:

- Quantidade de tarefas
- Quantidade de máquinas
- Tempo de processamento de cada tarefa em cada máquina
- Datas de entrega das tarefas

Dentre os dados citados, as datas de entrega necessitaram ser geradas por não estarem presentes nas instâncias definidas. Segundo Nosrati et al (2020), a geração de dados para o teste de um programa é de grande importância, pois os mesmos serão capazes de testar o desempenho e encontrar falhas no programa. Para tal foi desenvolvido o seguinte algoritmo com as etapas para a definição da data de entrega para cada tarefa:

- 1- Para cada instância, calcular a relação n/m entre número de tarefas (n) e número de máquinas (m);
- 2- Para cada tarefa, multiplicar n/m pela média dos tempos de processamento da tarefa nas máquinas e definir o resultado como data de entrega máxima;
- 3- Para cada tarefa, definir a data de entrega mínima como o tempo de processamento da tarefa na máquina mais rápida;
- 4- Para cada tarefa, calcular a diferença entre data de entrega máxima - data de entrega mínima;
- 5- Para cada tarefa, dividir a diferença obtida na etapa 4 pelo número 100;

- 6- Para cada tarefa, construir uma matriz com cem linhas e uma coluna, na qual a primeira linha corresponde a data de entrega mínima e essa é incrementada pelo resultado da etapa 5 até atingir a data de entrega máxima, na centésima linha.
- 7- Para cada tarefa, gerar um número aleatório no intervalo 1-100 o qual se refere à linha da matriz e definir seu conteúdo como a data de entrega para a tarefa.

O algoritmo criado com características de aleatoriedade garante que nenhuma tarefa possua data de entrega menor que seu tempo de processamento na máquina mais rápida, ou seja, nenhuma tarefa estará atrasada no início da resolução heurística e também permite que nem todas as tarefas possuam datas de entrega muito longas, podendo assim gerar dados que possam testar as heurísticas quanto a sua capacidade de reduzir o atraso total.

O Quadro 1 exemplifica os dados presentes em uma instância com doze tarefas e quatro máquinas.

Quadro 1 - Relação de dados contidos em uma instância

Tarefas	Tempo de processamento				Data de entrega
	Máquina 1	Máquina 2	Máquina 3	Máquina 4	
1	20	59	50	62	47
2	66	63	31	69	35
3	63	95	45	32	44
4	40	72	93	82	50
5	36	22	15	14	49
6	66	35	51	40	85
7	70	28	19	22	27
8	21	17	81	10	54
9	26	28	51	45	97
10	26	50	21	70	104
11	30	35	18	29	58
12	52	18	37	46	105

Fonte: autoria própria

A primeira coluna possui o nome de cada tarefa, a segunda coluna os tempos de processamento da tarefa nas máquinas representadas pelas subcolunas

e a última coluna a data de entrega da tarefa. Tempo de processamento e data de entrega possuem valores em “unidade de tempo” e podem representar a unidade necessária para o problema desde que estejam na mesma escala. Logo, se Tempo de processamento está em “dias”, Data de entrega deve ser representada também em “dias”.

Na Tabela 2 tem-se a relação de instâncias utilizadas nesse trabalho.

Tabela 2 - Relação de instâncias utilizadas

Instância	Nº de tarefas	Nº de máquinas
1	4	4
2	20	5
3	20	10
4	20	20
5	50	5
6	50	10
7	50	20
8	100	5
9	100	10
10	100	20
11	200	10
12	200	20

Fonte: autoria própria

O nível de dificuldade para a resolução do problema é aumentado conforme aumentam o número de tarefas e máquinas. As instâncias menores foram utilizadas para que se possa comparar efetivamente os resultados das heurísticas com os resultados da programação matemática.

As instâncias maiores representam com maior fidelidade o que se pode encontrar no ambiente produtivo, grande número de tarefas e máquinas, buscando testar assim a utilização das heurísticas estudadas para a resolução de problemas de porte realista.

4.2 IMPLEMENTAÇÃO DO MODELO MATEMÁTICO

O modelo matemático descrito na Seção 2.7 foi implementado no software Lingo versão 15.0, com licença para estudante, instalado em um computador pessoal com 2GB de memória RAM, processador Intel Celeron E3400 2.60Ghz, dois núcleos, sistema operacional Windows 7 Home Premium de 64bits.

Foram executadas as 12 instâncias distintas e registrados os valores das funções-objetivo encontrados, bem como o tempo que o software levou para chegar ao valor. Estipulou-se um tempo máximo de execução de 3600 segundos, haja vista que tempos muito longos para tomada de decisões não são desejáveis na indústria.

Assim, o melhor valor obtido pelo modelo matemático até o tempo máximo foi o valor utilizado para comparação com as heurísticas, podendo esse não ser a solução ótima para o problema.

4.3 MÉTODOS HEURÍSTICOS

As heurísticas utilizadas no trabalho foram implementadas pelo próprio autor, algumas delas combinadas com regras de prioridade conhecidas na literatura.

4.3.1 Heurística Construtiva 1

A Heurística Construtiva 1 (HC1) é a primeira desenvolvida objetivando a redução do *makespan* e do atraso total das tarefas. A mesma se dá pela alocação da tarefa na máquina em que a mesma possua o menor tempo de processamento, seguindo a lógica MTP. O pseudocódigo contendo o funcionamento dessa heurística pode ser observado na Figura 10.

Figura 10 - Pseudocódigo da HC1

Procedimento	HC1	(Entrada: instância)
1. início;		
2. enquanto	(houver tarefa não alocada)	faça:
3.	Identificar em qual máquina k a tarefa i tem o menor tempo de processamento;	
4.	Alocar a tarefa i na máquina k ;	
5. fim enquanto		
6.	Devolver (a seqüência de tarefas alocadas em cada máquina);	
7. fim.		

Fonte: autoria própria

A entrada para o método heurístico é a instância, já o retorno do algoritmo é a seqüência de tarefas alocadas em cada máquina. Os dados referentes ao *makespan* e atraso total são obtidos a partir dessa seqüência retornada. A instância representada no Quadro 1 será utilizada para exemplificar o funcionamento da heurística na Figura 11.

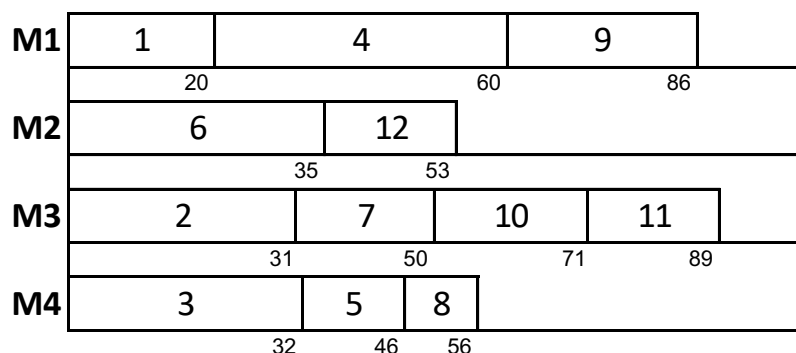
Figura 11 - Exemplo de alocação das tarefas de acordo com a HC1



Fonte: autoria própria

Iniciando pela tarefa 1 até chegar na tarefa 12 foi selecionada a máquina com menor tempo de processamento para cada tarefa e alocada a tarefa na máquina selecionada. O gráfico de Gantt que ilustra essa alocação está na Figura 12.

Figura 12 - Gráfico de Gantt para a solução da HC1



Fonte: autoria própria

A máquina 3 alocou o maior número de tarefas e é também a última máquina a terminar de processar suas tarefas, portanto o instante de término da última tarefa dessa máquina, 89, representa o *makespan* da solução.

4.3.2 Heurística Construtiva 2

A segunda Heurística Construtiva (HC2) desenvolvida nesse trabalho identifica dois dados referentes a cada instância e, seguida de uma análise condicional, escolhe a máquina na qual a tarefa será alocada.

O pseudocódigo dessa heurística pode ser observado na Figura 13.

Figura 13 - Pseudocódigo da HC2

Procedimento	HC2	(Entrada: instância)
1.	início;	
2.	enquanto (houver tarefa não alocada) faça:	
3.	Identificar em qual máquina a tarefa i tem o menor tempo de processamento;	
4.	Identificar em qual máquina a tarefa i tem o segundo menor tempo de processamento;	
5.	Se (a diferença entre a média dos tempos de processamento da tarefa i e a data de entrega da tarefa i for maior ou igual a 0):	
6.	Alocar a tarefa i na máquina mais rápida;	
7.	fim se	
8.	Senão	
9.	Alocar a tarefa i na segunda máquina mais rápida;	
10.	fim senão	
11.	fim enquanto	
12.	Devolver (a sequência de tarefas alocadas em cada máquina);	
13.	fim.	

Fonte: autoria própria

O método de ordenação segue a regra MDE, no qual as tarefas com menor data de entrega terão prioridade no processamento na máquina em que foi alocada.

O pseudocódigo dessa heurística pode ser observado na Figura 15.

Figura 15 - Pseudocódigo da HM1

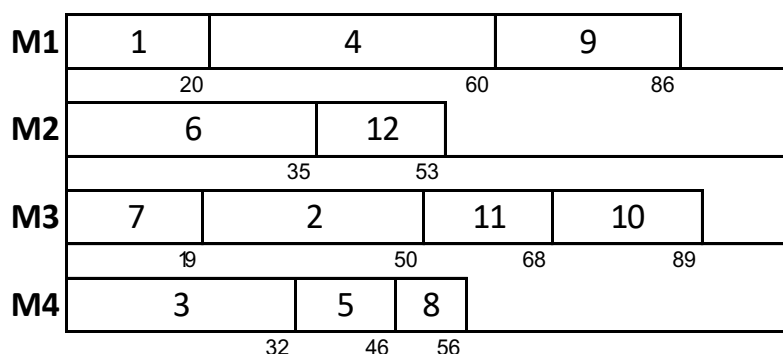
Procedimento	HM1	(Entrada: devolução da heurística construtiva)
1. início;		
2. para (cada máquina k):		
3. Ordemar as tarefas de forma não decrescente de acordo com a data de entrega;		
4. fim para		
5. Devolver (a sequência de tarefas alocadas em cada máquina);		
6. fim.		

Fonte: autoria própria

Essa heurística percorre cada máquina e reordena as tarefas alocadas em cada máquina seguindo a regra de prioridade MDE. Com isso, as tarefas que possuem menor data de entrega são processadas primeiro e as chances das tarefas atrasarem devem diminuir.

O gráfico de Gantt da Figura 16 retrata a reordenação das tarefas pela HM1 segundo o retorno do exemplo de aplicação da HC1.

Figura 16 - Gráfico de Gantt para a solução da HM1



Fonte: autoria própria

A máquina 3 é a única que possui tarefas a serem reordenadas e, seguindo a regra de prioridade, todas as tarefas alocadas nessa máquina trocaram de posição, a tarefa 7 é a primeira a ser processada por possuir menor data de entrega, já a tarefa 10 é a última por possuir maior data de entrega.

4.3.4 Heurística de Melhoria 2

Seguindo o proposto pela HM1, a Heurística de Melhoria 2 (HM2) recebe o retorno obtido por alguma heurística construtiva e reordena as tarefas de cada máquina seguindo a regra MTP.

O pseudocódigo dessa heurística pode ser observado na Figura 17.

Figura 17 - Pseudocódigo da HM2

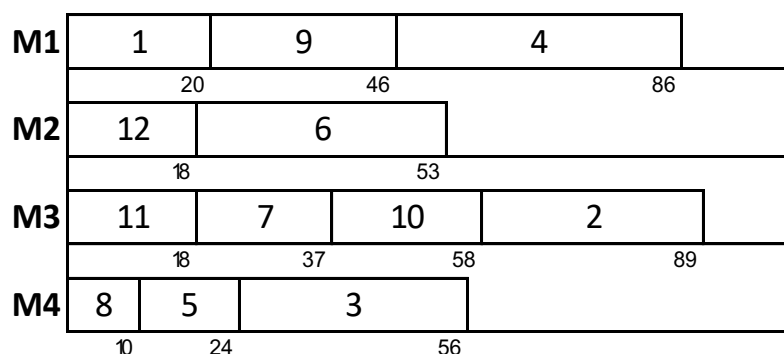
Procedimento	HM2	(Entrada: devolução da heurística construtiva)
1. início;		
2. para (cada máquina k):		
3. Ordenar as tarefas de forma não decrescente de acordo com o tempo de processamento;		
4. fim para		
5. Devolver (a sequência de tarefas alocadas em cada máquina);		
6. fim.		

Fonte: autoria própria

Essa heurística percorre cada máquina e reordena as tarefas alocadas em cada máquina seguindo a regra de prioridade MTP. Com isso, as tarefas que possuem menor tempo de processamento possuem prioridade na máquina em que está alocada.

O gráfico de Gantt da Figura 18 retrata a reordenação das tarefas pela HM2 segundo o retorno do exemplo de aplicação da HC1.

Figura 18 - Gráfico de Gantt para a solução da HM2



Fonte: autoria própria

Aplicando a HM2 ao resultado de HC1 obteve-se a reordenação das tarefas em todas as máquinas, na máquina 3, por exemplo, seguindo a regra de prioridade

MTP a tarefa 11 tornou-se a primeira na ordem de processamento e a tarefa 2 passou a ser a última.

4.3.5 Heurística de Melhoria 3

A Heurística de Melhoria 3 (HM3), diferentemente das heurísticas de melhoria anteriores, realiza a realocação de tarefas entre as máquinas seguindo o critério de diferença entre o número de tarefas entre a máquina com maior número de tarefas e a máquina com o menor número de tarefas, bem como a condição de não piorar o *makespan* obtido com a solução de entrada.

O pseudocódigo contendo o funcionamento dessa heurística pode ser observado na Figura 19.

Figura 19 - Pseudocódigo da HM3

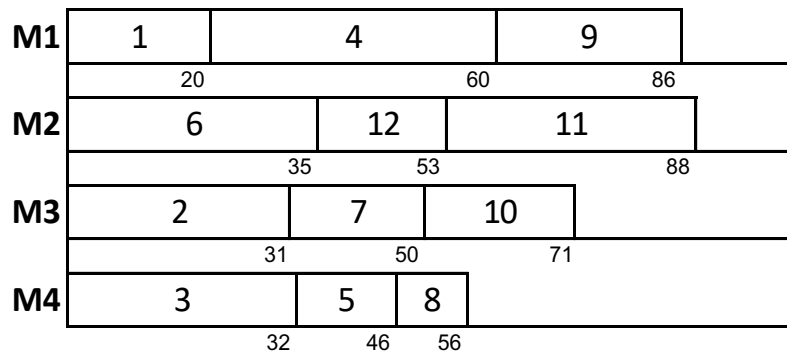
Procedimento	HM3	(Entrada: Solução inicial para o problema)
1. início;		
2.	Identificar o <i>makespan</i> ;	
3.	Identificar a máquina com o maior número de tarefas;	
4.	Identificar a máquina com o menor número de tarefas;	
5. enquanto	(a diferença entre o número de tarefas da máquina com maior número de tarefas e da máquina com o menor número de tarefas for maior que 1) faça:	
6.	Transferir a última tarefa da máquina com o maior número de tarefas para a próxima posição livre na máquina com menor número de tarefas;	
7.	Identificar o <i>makespan</i> novo;	
8.	Se (<i>makespan</i> é menor ou igual a <i>makespan</i> novo) faça:	
9.	Desfazer a transferência;	
10.	Forçar fim enquanto;	
11.	fim se	
12.	senão	
13.	<i>makespan</i> recebe <i>makespan</i> novo;	
14.	Atualizar as quantidades de tarefas nas máquinas envolvidas	
15.	Identificar a máquina com o maior número de tarefas;	
16.	Identificar a máquina com o menor número de tarefas;	
17.	fim senão	
18. fim enquanto		
19.	Devolver (a sequência de tarefas alocadas em cada máquina);	
20. fim.		

Fonte: autoria própria

Dada uma alocação de tarefas a HM3 identifica, a máquina com maior número de tarefas e a máquina com menor número de tarefas. Se a diferença entre o número de tarefas for maior que um, realiza a transferência da última tarefa da máquina com mais tarefas para a última posição livre da máquina com menos tarefas. Caso o *makespan* melhore, a etapa descrita continua a ser executada, caso o *makespan* se mantenha ou piore, a alocação anterior é devolvida e a heurística finalizada.

O gráfico de Gantt da Figura 20 ilustra a aplicação da HM3 para o exemplo utilizado em HC1.

Figura 20 - Gráfico de Gantt para a solução de HM3



Fonte: autoria própria

Durante o primeiro ciclo a HM3 transferiu a tarefa 11, inicialmente alocada na máquina 3, para a máquina 2. Como o *makespan* diminuiu, a nova alocação foi mantida e, já que a diferença entre o número de tarefas da máquina com o maior número de tarefas e a máquina com o menor número de tarefas passou a ser igual a zero, a heurística foi finalizada.

Para as heurísticas contrutivas HC1 e HC2 têm-se a possibilidade de diferenciação tanto para o *makespan* quanto para o atraso total. Nas heurísticas de melhoria HM1 e HM2 somente o valor do atraso total pode ser alterado, haja vista que as tarefas não são trocadas de máquina. A aplicação da HM3 possibilita a redução tanto do *makespan* quanto do atraso total.

5 RESULTADOS

O presente capítulo contém os resultados da implementação do modelo matemático e das heurísticas abordadas no capítulo 4 para as mesmas doze instâncias. Está dividido em resultados do modelo matemático, resultados das heurísticas e análise comparativa entre os métodos utilizados.

5.1 RESULTADOS DO MODELO MATEMÁTICO

Conforme descrito na seção 4.2 as doze instâncias foram solucionadas por programação matemática usando o modelo matemático buscando a minimização do *makespan* e do atraso total, sendo o primeiro critério prioritário.

A Tabela 3 contém os resultados da aplicação do modelo, sendo eles a instância estudada, o tempo de execução, o *makespan* e o atraso total. Unidade de Tempo (UT) é a escala para o *makespan* e atraso total.

Tabela 3 - Resultados para o modelo matemático

Instância	Tempo (s)	Makespan (UT)	Atraso Total (UT)
1	1,33	59	12
2	1206,75	84	0
3	287,65	53	9
4	11,67	23	0
5	3600	[232]	[544]
6	3600	-	-
7	3600	-	-
8	3600	-	-
9	3600	-	-
10	3600	-	-
11	3600	-	-
12	3600	-	-

Fonte: autoria própria

O modelo matemático foi capaz de resolver, no tempo limite estabelecido de 3600s, somente até a Instância 4, com 20 tarefas e 20 máquinas. Para a Instância 5, com 50 tarefas 5 máquinas, o modelo encontrou uma solução, que pode ou não ser a solução ótima, no tempo limite estabelecido. A partir da Instância 6 o modelo não foi capaz de encontrar uma solução no tempo limite.

5.2 RESULTADOS DAS HEURÍSTICAS

Observando as propriedades das heurísticas descritas na subseção 4.3.5, foram realizadas doze combinações de aplicações para as heurísticas. As combinações utilizadas estão descritas na no Quadro 2.

Quadro 2 - Combinação e sequência das heurísticas

Combinação	Sequência das heurísticas
1	HC1
2	HC1+HM3
3	HC1+HM1
4	HC1+HM1+HM3
5	HC1+HM2
6	HC1+HM2+HM3
7	HC2
8	HC2+HM3
9	HC2+HM1
10	HC2+HM1+HM3
11	HC2+HM2
12	HC2+HM2+HM3

Fonte: autoria própria

HC1 e HC2 foram utilizadas, cada uma, na etapa inicial de seis métodos e geraram alocações que puderam então ser melhoradas pelas HMs. As heurísticas de melhoria HM1 e HM2 foram aplicadas separadamente a ambas HC1 e HC2, pois realizam ordenações que sobrepõe uma a outra. Já HM3 foi aplicada ao fim de todas as sequências distintas, uma vez que a mesma realiza a troca de tarefas entre as máquinas.

Assim, partindo dos cinco métodos heurísticos apresentados, foram desenvolvidos doze métodos heurísticos distintos para a alocação e sequenciamento das tarefas. Ao incrementar a quantidade de métodos, a quantidade de soluções também aumenta e, com isso, foi possível escolher de modo mais refinado o melhor resultado entre os doze possíveis para cada instância trabalhada.

Utilizando as mesmas heurísticas podem ser gerados outros métodos que podem ou não ser melhores do que os utilizados, porém, os métodos utilizados já cumprem seu objetivo de testar o funcionamento de cada heurística criada.

5.2.1 Resultados para o Critério *Makespan*

O Quadro 3 traz os resultados para a aplicação das combinações heurísticas relevantes para o cálculo do *makespan* e os resultados da solução pelo modelo matemático. As combinações HC1+HM1, HC1+HM2, HC2+HM1 e HC2+HM2 foram excluídas da análise, haja vista que as HMs não modificam o valor do *makespan* obtido pelas HCs sozinhas. Os valores em destaque foram os melhores encontrados para os métodos heurísticos.

Quadro 3 - Resultados para o makespan em UT

Inst.	Modelo matemático	HC1	HC1+HM3	HC1+HM1+HM3	HC1+HM2+HM3	HC2	HC2+HM3	HC2+HM1+HM3	HC2+HM2+HM3
1	59	86	72	72	59	86	72	72	59
2	84	109	98	88	88	211	177	171	171
3	53	115	88	88	82	96	79	84	84
4	23	45	45	45	45	45	45	45	45
5	[232]	237	237	237	237	582	582	582	582
6	-	75	75	71	75	157	157	157	157
7	-	38	38	32	27	107	89	59	89
8	-	418	418	418	418	899	825	825	825
9	-	137	137	137	137	275	275	275	275
10	-	56	56	56	56	91	91	91	91
11	-	335	293	305	303	597	564	541	493
12	-	86	86	86	86	153	153	153	153

Fonte: autoria própria

Confirmando o descrito por Taha (2008), devido à sua complexidade, o modelo matemático não foi capaz de encontrar uma solução para todas as instâncias no tempo estipulado, enquanto que os métodos heurísticos obtiveram soluções para todas as instâncias.

As soluções obtidas pelos métodos heurísticos são viáveis e, na Instância 1, HC1+HM2+HM3 e HC2+HM2+HM3 chegaram no mesmo valor encontrado para a solução ótima. Considerando seus melhores resultados, as heurísticas obtiveram, em média, valores 50% maiores para as instâncias nas quais o modelo matemático foi capaz de encontrar a solução final no tempo estipulado. Também obtiveram soluções para 7 instâncias em que o modelo matemático não obteve sequer solução parcial após 1h.

O Quadro 4 apresenta a comparação dos resultados para o *makespan* obtidos pelas heurísticas HC1 e HC2 somente.

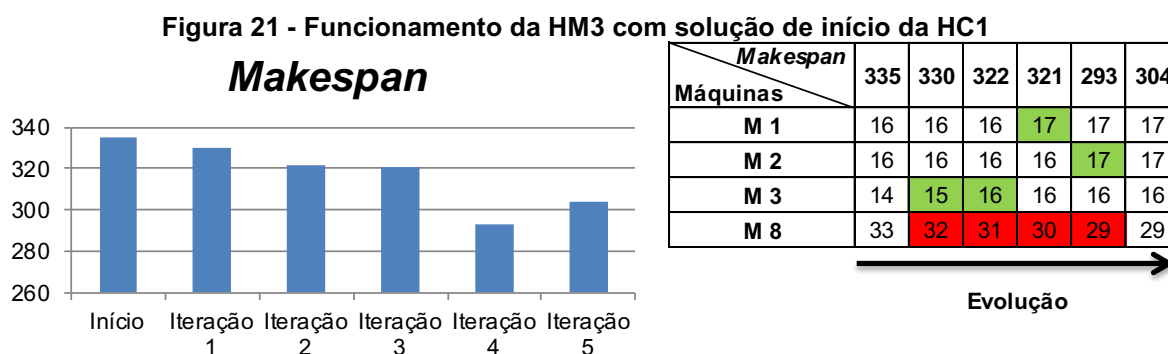
Quadro 4 – Comparação entre os resultados de HC1 e HC2

Instância	HC1	Comparativo	HC2
1	86	0%	86
2	109	HC1 (-48%)	211
3	115	HC2 (-17%)	96
4	45	0%	45
5	237	HC1 (-59%)	582
6	75	HC1 (-52%)	157
7	38	HC1 (-64%)	107
8	418	HC1 (-54%)	899
9	137	HC1 (-50%)	275
10	56	HC1 (-38%)	91
11	335	HC1 (-44%)	597
12	86	HC1 (-44%)	153

Fonte: autoria própria

Observando o Quadro 3 e o Quadro 4 nota-se que as soluções partindo da HC1 foram, em grande maioria, melhores do que as soluções derivadas de métodos com início em HC2. Logo, conclui-se que, para o critério de *makespan*, o método de alocação proposto por HC2 que, de acordo com o condicional, pode alocar a tarefa na segunda máquina com menor tempo de processamento, não é tão efetivo quando o proposto em HC1 que aloca cada tarefa em sua máquina com menor tempo de processamento.

A Figura 21 apresenta o funcionamento da HM3 aplicada à solução obtida com HC1 para a Instância 11, que contém 200 tarefas e 10 máquinas. O preenchimento em verde indica a máquina que recebeu uma tarefa a cada iteração e em vermelho tem-se a máquina que perdeu a última tarefa a cada iteração. Os valores internos indicam as quantidades de tarefas nas máquinas a cada iteração.



Fonte: autoria própria

De acordo com a Figura 21, é possível observar a evolução da melhoria do *makespan* a cada iteração de HM3. Inicialmente tem-se que a máquina com maior número de tarefas é a Máquina 8, com 33 tarefas, e a máquina com menor número de tarefas é a Máquina 3 com 14 tarefas. Após a primeira iteração a Máquina 8 transferiu a última tarefa alocada nela para a Máquina 3 e passou a ter 32 tarefas enquanto a Máquina 3 ficou com 15 tarefas. Como o *makespan* melhorou essa nova alocação foi mantida e assim as iterações seguiram sucessivamente até a Iteração 5 na qual, após a transferência de tarefas, obteve um *makespan* maior que o atual. Assim, a solução obtida na Iteração 4 foi restaurada e a HM3 finalizada.

O Quadro 5 traz os resultados comparativos da combinação das HMs em relação à HC1.

Quadro 5 - Comparação de melhoria das HMs em HC1 para o *makespan*

Instâncias	HC1	HC1+HM3	HC1+HM1+HM3	HC1+HM2+HM3
1	86	-16%	-16%	-31%
2	109	-10%	-19%	-19%
3	115	-23%	-23%	-29%
4	45	0%	0%	0%
5	237	0%	0%	0%
6	75	0%	-5%	0%
7	38	0%	-16%	-29%
8	418	0%	0%	0%
9	137	0%	0%	0%
10	56	0%	0%	0%
11	335	-13%	-9%	-10%
12	86	0%	0%	0%

Fonte: autoria própria

Em relação à HC1, HC1+HM3 foi capaz de reduzir o *makespan* em quatro instâncias. A combinação HC1+HM1+HM3 reduziu o *makespan* em seis oportunidades e HC1+HM2+HM3 em cinco. HC1+HM2+HM3 possui a maior soma das taxas de redução, chegando a 31% na Instância 1 e 29% nas Instâncias 3 e 7. Isso pode ser explicado pela utilização da HM2 no método. Em cada máquina, as tarefas mais lentas foram colocadas no final da sequência e com a HM3 as mesmas foram transferidas para máquinas com menor quantidade de tarefas, proporcionando a redução no *makespan*.

No Quadro 6 observa-se os resultados comparativos das HMs em HC2.

Quadro 6 - Comparação de melhoria das HMs em HC2 para o *makespan*

Instâncias	HC2	HC2+HM3	HC2+HM1+HM3	HC2+HM2+HM3
1	86	-16%	-16%	-31%
2	211	-16%	-18%	-19%
3	96	-18%	-13%	-13%
4	45	0%	0%	0%
5	582	0%	0%	0%
6	157	0%	0%	0%
7	107	-41%	-45%	-39%
8	899	-8%	-8%	-8%
9	275	0%	0%	0%
10	91	0%	0%	0%
11	597	-6%	-9%	-17%
12	153	0%	0%	0%

Fonte: autoria própria

Em relação à HC2, HC2+HM3 foi capaz de reduzir o *makespan* em seis instâncias. A combinação HC2+HM1+HM3 reduziu o *makespan* nas mesmas seis oportunidades e HC2+HM2+HM3 também em seis. A combinação HC2+HM1+HM3 obteve a maior taxa de redução, com 45%, contudo, assim como o encontrado nas comparações para a HC1, a combinação de HC2 com HM2+HM3 proporcionou a maior redução se somadas as taxas.

5.2.2 Resultados para o Critério Atraso Total

O Quadro 7 traz os resultados para a aplicação das combinações heurísticas para o cálculo do atraso total e os resultados da solução pelo modelo matemático. As combinações HC1+HM1, HC1+HM2, HC2+HM1 e HC2+HM2 foram mantidas na análise, haja vista que as HMs interferem no valor do atraso total obtido pelas HCs sozinhas. Os valores em destaque foram os melhores encontrados para os métodos heurísticos.

Quadro 7 - Resultados para o atraso total em UT

	Modelo Matemático	HC1	HC1+HM3	HC1+HM1	HC1+HM1+HM3	HC1+HM2	HC1+HM2+HM3	HC2	HC2+HM3	HC2+HM1	HC2+HM1+HM3	HC2+HM2	HC2+HM2+HM3
1	12	36	22	36	22	39	12	36	22	36	22	39	12
2	0	169	158	0	0	33	33	461	384	170	112	233	120
3	9	146	138	17	9	77	65	138	120	9	9	43	43
4	0	22	22	9	9	21	21	22	22	9	9	21	21
5	[544]	987	987	17	17	445	445	2947	2947	1519	1519	2149	2149
6	-	112	112	0	0	45	45	455	455	15	15	136	136
7	-	11	11	0	0	11	11	66	25	0	0	59	9
8	-	1099	1099	9	9	511	511	8629	7661	216	216	5997	5667
9	-	368	368	0	0	147	147	1800	1800	0	0	620	620
10	-	117	117	0	0	29	29	226	226	0	0	21	21
11	-	1446	1367	0	0	402	295	6722	6438	26	26	2840	2649
12	-	117	117	0	0	74	74	812	812	0	0	121	121

Fonte: autoria própria.

Observa-se no Quadro 7 que os métodos heurísticos conseguiram equiparar o valor encontrado para o atraso total por meio do método matemático nas três primeiras instâncias. Para a Instância 5, alguns métodos heurísticos superaram o modelo matemático, entretanto a solução do método matemático para essa instância não é ótima. A partir da Instância 6, somente tem-se os valores de atraso total para os métodos heurísticos.

Assim como para o critério *makespan*, os métodos baseados na HC1 obtiveram a maior quantidade de melhores resultados para o atraso total. Ao colocar algumas tarefas na segunda máquina mais rápida para elas, na maior parte das vezes HC2 não obteve resultados tão bons quanto HC1, sendo que em algumas instâncias o valor para atraso total de HC2 foi quase 8 vezes maior do que o atraso total para HC1.

Observa-se também que HM1 está presente na ampla maioria de melhores resultados. Isso se deve pela prioridade MDE utilizada na mesma. Colocando as tarefas que deveriam ser entregues primeiro com prioridade no sequenciamento, proporcionou um menor atraso total para as instâncias.

O Quadro 8 traz os resultados comparativos da combinação das HMs em HC2.

Quadro 8 - Comparação de melhoria das HMs em HC1 para o atraso total

Instâncias	HC1	HC1+HM3	HC1+HM1	HC1+HM1+HM3	HC1+HM2	HC1+HM2+HM3
1	36	-39%	0%	-39%	8%	-67%
2	169	-7%	-100%	-100%	-80%	-80%
3	146	-5%	-88%	-94%	-47%	-55%
4	22	0%	-59%	-59%	-5%	-5%
5	987	0%	-98%	-98%	-55%	-55%
6	112	0%	-100%	-100%	-60%	-60%
7	11	0%	-100%	-100%	0%	0%
8	1099	0%	-99%	-99%	-54%	-54%
9	368	0%	-100%	-100%	-60%	-60%
10	117	0%	-100%	-100%	-75%	-75%
11	1446	-5%	-100%	-100%	-72%	-80%
12	117	0%	-100%	-100%	-37%	-37%

Fonte: autoria própria

Excetuando-se a combinação HC1+HM3, todas as combinações heurísticas resultaram em um melhor resultado para o *makespan* para a maioria das instâncias.

Métodos contendo HM2, cuja regra de prioridade é MTP, proporcionaram reduções de até 80% no atraso total. Já para método contendo HM1, a redução chegou nos 100%, zerando o atraso total. Isso demonstra a efetividade da regra de prioridade MDE para a redução do atraso total. A combinação HC+HM1+HM3 obteve os melhores resultados dentre os métodos comparados, proporcionando redução no atraso total para as doze instâncias.

No Quadro 9 observa-se os resultados comparativos das HMs em HC2.

Quadro 9 - Comparação de melhoria das HMs em HC2 para o atraso total

Instâncias	HC2	HC2+HM3	HC2+HM1	HC2+HM1+HM3	HC2+HM2	HC2+HM2+HM3
1	36	-39%	0%	-39%	8%	-67%
2	461	-17%	-63%	-76%	-49%	-74%
3	138	-13%	-93%	-93%	-69%	-69%
4	22	0%	-59%	-59%	-5%	-5%
5	2947	0%	-48%	-48%	-27%	-27%
6	455	0%	-97%	-97%	-70%	-70%
7	66	-62%	-100%	-100%	-11%	-86%
8	8629	-11%	-97%	-97%	-31%	-34%
9	1800	0%	-100%	-100%	-66%	-66%
10	226	0%	-100%	-100%	-91%	-91%
11	6722	-4%	-100%	-100%	-58%	-61%
12	812	0%	-100%	-100%	-85%	-85%

Fonte: autoria própria

Exceto pela combinação HC2+HM3, todas as combinações heurísticas obtiveram melhorias para o atraso total na maior parte das instâncias. HM2 chegou a alcançar 91% de redução no atraso total para as combinações nas quais esteve aplicada e HM1 chegou nos 100% de redução do atraso total para algumas instâncias em suas combinações. As combinações HC2+HM1+HM3 e HC2+HM2+HM3 obtiveram redução no atraso total para as doze instâncias, mas a primeira citada conta com a maior soma das taxas de redução.

Tanto para HC1 quanto para HC2 as heurísticas de melhoria obtiveram êxito na sua função e aperfeiçoaram a solução inicial para grande parte das instâncias. HM3, quando melhorou o *makespan*, foi capaz de manter ou melhorar o atraso total.

5.2.3 Análise Comparativa dos Resultados Entre as Heurísticas e o Modelo Matemático

Para a análise comparativa dos resultados foram selecionados os melhores resultados entre os métodos heurísticos para o *makespan*, seu respectivo atraso total e tempo de execução computacional. Em caso de empate entre os métodos heurísticos tanto para o *makespan* quanto para o atraso total, foi selecionada a combinação com o menor tempo de execução. Esses valores foram então comparados com os resultados do modelo matemático.

O Quadro 10 relaciona os dados pertinentes para a comparação. Em destaque, com preenchimento, estão os melhores valores para cada instância.

Quadro 10 - Dados comparativos entre os resultados das heurísticas e do modelo matemático

Inst	Método	<i>Makespan</i>		Atraso total		Tempo	
		Modelo matemático	Heurística	Modelo matemático	Heurística	Modelo matemático (s)	Heurística (ms)
1	HC2+HM2+HM3	59	59 (0%)	12	12 (0%)	1,33	0,015
2	HC1+HM1+HM3	84	88 (+5%)	0	0 (0%)	1206,75	0,078
3	HC2+HM3	53	79 (+49%)	9	120 (1233%)	287,65	0,093
4	HC2+HM1	23	45 (+96%)	0	9	11,67	0,093
5	HC1+HM1	[232]	237 (+2%)	[544]	17 (-97%)	3600	0,111
6	HC1+HM1+HM3	-	71		0	3600	0,109
7	HC1+HM2+HM3	-	27		11	3600	0,14
8	HC1+HM1+HM3	-	418		9	3600	0,14
9	HC1+HM1	-	137		0	3600	0,197
10	HC1+HM1+HM3	-	56		0	3600	0,266
11	HC1+HM3	-	293		1367	3600	0,312
12	HC1+HM1+HM3	-	86		0	3600	0,469

Fonte: autoria própria

De acordo com o Quadro 10, para o *makespan*, as heurísticas obtiveram o valor ótimo para 1 (uma) instância e, considerando a Instância 5, as heurísticas ficaram a 38% do valor ótimo. A partir da Instância 6 as heurísticas obtiveram

resultados em no máximo 0,469ms, enquanto que a programação matemática não chegou sequer em um limitante com 3600s.

Para o atraso total, as heurísticas obtiveram o valor ótimo em duas instâncias e superaram o limitante encontrado pela programação matemática para a Instância 5. Porém, é importante destacar que o resultado obtido pelo modelo matemático para a Instância 5 pode não ser o ótimo e uma possível melhora no valor do *makespan* obtido pela heurística pode acarretar em um aumento do atraso total.

Rodar todos os 12 métodos para as 12 instâncias necessitaria de apenas 26,89ms, valor aproximadamente 49 vezes menor do que o tempo de execução do modelo matemático para somente a Instância 1, que contem a menor quantidade de tarefas e máquinas.

Em relação ao tempo necessário, a programação matemática mostrou-se inviável caso aplicada em problemas reais, confirmando o descrito por Marín, García e Gómez (2014). Portanto as heurísticas cumprem sua função ao oferecer soluções factíveis para o problema de minimização da soma do *makespan* e do atraso total em um tempo muito pequeno.

5.2.4 Análise Comparativa Entre as Combinações Heurísticas

Objetivando comparar as combinações heurísticas foi atribuído 1 ponto de sucesso para cada combinação sempre que, considerando cada instância, a mesma possua valor igual ao menor valor encontrado para *makespan* e outros 0,5 cada vez que a mesma combinação atinja o menor valor para atraso total, ou seja, aplicada a uma instância, se dada combinação heurística possui o menor valor para *makespan* e para atraso total, essa combinação possui 1,5 sucessos.

Assim, tem-se na Tabela 4 a relação de sucessos para as combinações heurísticas.

Tabela 4 - Relação de sucessos para as combinações heurísticas

Combinação	Sucessos
HC1	6
HC1+HM3	7
HC1+HM1	9
HC1+HM1+HM3	12
HC1+HM2	6
HC1+HM2+HM3	9,5
HC2	1
HC2+HM3	2
HC2+HM1	1,5
HC2+HM1+HM3	1,5
HC2+HM2	1
HC2+HM2+HM3	2,5

Fonte: autoria própria

De acordo com a Tabela 4, observa-se que a combinação HC1+HM1+HM3 possui o maior número de sucessos, já as combinações HC2 e HC2+HM2 possuem o menor número de sucessos. Todos os métodos heurísticos obtiveram ao menos um sucesso, demonstrando a relevância da utilização de vários métodos, pois para determinada instância, o método menos efetivo pode gerar melhor solução do que o método mais efetivo no geral. O método de pontuação de sucessos utilizado corrobora com a hierarquia dos critérios que, por sua vez, dá prioridade para a otimização dos recursos internos a empresa, o maquinário e seus custos, frente ao ambiente externo, os clientes.

6 CONCLUSÕES

Objetivando o desenvolvimento de métodos heurísticos eficientes para a minimização da soma do *makespan* e do atraso total, com prioridade para o *makespan*, no ambiente de máquinas paralelas não relacionadas, foram apresentadas as dificuldades do modelo matemático na resolução desse tipo de problema e destacada a importância das heurísticas.

Instâncias pré-existentes foram adaptadas e utilizadas na execução do modelo matemático e dos métodos heurísticos desenvolvidos, esses últimos contaram com doze métodos distintos. Por fim, realizou-se a comparação entre os resultados obtidos por meio da programação matemática e dos métodos heurísticos quanto ao valor do *makespan*, do atraso total e do tempo necessário para as suas respectivas execuções.

Partindo da minimização da soma do *makespan* e do atraso total conclui-se que as heurísticas são alternativas funcionais à resolução pelo modelo matemático. Porém, uma vez que o modelo seja capaz de ser executado até o final, o método heurístico pode no máximo igualar o modelo matemático quanto a melhor solução para o problema.

Observando os resultados obtidos pode-se concluir que, ainda que possam não alcançar valores iguais ao método matemático para instâncias com pequena quantidade de tarefas e máquinas, a utilização de variadas combinações heurísticas é eficiente para encontrar uma solução satisfatória. Levando em consideração instâncias maiores o método heurístico possui ampla vantagem em relação ao modelo matemático devido ao tempo de execução, pois o ambiente competitivo das indústrias requer soluções cada vez mais rápidas e satisfatórias.

A combinação heurística com maior número de sucessos foi a HC1+HM1+HM3 que, ao alocar inicialmente as tarefas nas máquinas de acordo com o MTP, seguida da ordenação por MDE em cada máquina e por último a tentativa de equilibrar o número de tarefas nas máquinas possibilitou maior número de melhores resultados para o *makespan* e atraso total.

Nota-se também o benefício da utilização de vários métodos e combinações heurísticas, pois a execução das mesmas é muito rápida e aumenta a quantidade de soluções, proporcionando a escolha de uma solução mais eficiente para o problema.

Resultados melhores podem ser obtidos ao utilizar diferentes métodos de alocação e ordenação de tarefas, bem como aplicar diferentes combinações entre as heurísticas utilizadas. Instâncias com dados diferentes dos utilizados podem fazer com que uma dada heurística possua mais ou menos sucessos.

Assim, conclui-se que os objetivos desse trabalho foram alcançados com o desenvolvimento de heurísticas para a resolução do problema da minimização da soma do *makespan* e do atraso total que obtiveram resultados, em média, 38% maiores do que os obtidos pelo modelo matemático no *makespan* e superaram ou igualaram o modelo matemático em quatro das cinco oportunidades possíveis no atraso total.

Considerando o tempo de execução, as heurísticas comprovaram sua eficiência, uma vez que a execução conjunta dos doze métodos desenvolvidos para todas as instâncias é aproximadamente 49 vezes menor do que o tempo necessário para a execução do método matemático com somente uma instância, sendo essa a menor utilizada, com 4 tarefas e 4 máquinas.

As heurísticas desenvolvidas podem ser refinadas e as instâncias utilizadas podem ainda ser aplicadas para diferentes critérios de otimização, como por exemplo, a minimização do adiantamento das tarefas em trabalhos futuros.

REFERÊNCIAS

ARENALES, M. et al. **Pesquisa Operacional: para cursos de engenharia**. Rio de Janeiro: Elsevier, 2011.

BERTRAND, J. Will M.; FRANSOO, Jan C. Operations management research methodologies using quantitative modeling. **International Journal Of Operations & Production Management**, [s.l.], v. 22, n. 2, p.241-264, fev. 2002. Emerald. <http://dx.doi.org/10.1108/01443570210414338>.

CAETANO, J. P. F. et al. Planejamento e controle da produção: um estudo de caso de uma prestadora de serviços na industrialização de fertilizantes. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO. 36., 2016, João Pessoa. **Anais... ABEPRO: João Pessoa, 2016.**

CELINI, M. R. J. **Heurísticas para o problema de minimização da soma de adiantamentos e atrasos no ambiente de máquinas paralelas não relacionadas**. 2017. 74 f. TCC (Graduação) - Curso de Engenharia de Produção, Universidade Tecnológica Federal do Paraná - UTFPR, Ponta Grossa, 2017.

ETCHEVERRY, G. V. **Programação de tarefas em máquinas paralelas não-relacionadas com tempos de setup dependentes da sequência**. 2012. 54 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal do Rio Grande do Sul. Porto Alegre, 2012.

EZUGWU, A. E. Enhanced symbiotic organisms search algorithm for unrelated parallel machines manufacturing scheduling with setup times. **Knowledge-Based Systems**, v. 172, p. 15-32, may. 2019. <https://doi.org/10.1016/j.knosys.2019.02.005>

HARTMANN, S.; BRISKORN, D. A survey of variants and extensions of the resource-constrained Project scheduling problem. **European Journal of Operation Research**, v. 207, n. 1, p. 1-14, nov. 2010. <https://doi.org/10.1016/j.ejor.2009.11.005>

HILLIER, F. S.; LIEBERMAN, G. J. **Introdução à pesquisa operacional**. 8. ed. São Paulo: McGraw-Hill, 2006.

JUNQUEIRA, R. A. R.; MORABITO, R. Programação e sequenciamento das frentes de colheita de cana-de-açúcar: modelo e métodos de solução para problemas de grande porte. **Gestão & Produção**, v. 25, n. 1, p. 132-147, mar. 2018. <http://dx.doi.org/10.1590/0104-530x2647-16>.

LI, Z. et al. Unrelated parallel machine scheduling problem with energy and tardiness cost. **The International Journal of Advanced Manufacturing Technology**, v. 84, n. 1-4, p. 213-226, apr. 2016. <https://doi-org.ez48.periodicos.capes.gov.br/10.1007/s00170-015-7657-2>.

LIN, Y. K.; PFUND, M. E.; FOWLER, J. W. Heuristics for minimizing regular performance measures in unrelated parallel machine scheduling problems. **Computers & Operations Research**, v. 38, n. 6, p. 901-916, jun. 2011. <https://doi.org/10.1016/j.cor.2010.08.018>.

MARÍN, J. A. A.; GARCÍA, J. A. G.; GÓMEZ, D. C. Applying TOC Heuristics to Job Scheduling in a Hybrid Flexible Flow Shop. **Dyna rev.fac.nac.minas**, v. 81, n. 186, p. 113-119, jul./ago. 2014. <http://dx.doi.org/10.15446/dyna.v81n186.39489>.

MARTIN, M. et al. Modification of Haessler's sequential heuristic procedure for the one-dimensional cutting stock problem with setup cost. **Production**, v. 28, e20170105, oct. 2018. <http://dx.doi.org/10.1590/0103-6513.20170105>.

MESQUITA, M.A.; CASTRO, R. L. Análise das práticas de planejamento e controle da produção em fornecedores da cadeia automotiva brasileira. **Gestão & Produção**, v. 15, n. 1, p. 33-42, jan./abr. 2008. <http://dx.doi.org/10.1590/S0104530X2008000100005>.

MIGUEL, P. A. C. et al. **Metodologia de pesquisa em engenharia de produção e gestão de operações**. 2. Ed. Rio de Janeiro: Elsevier, ABEPRO, 2012.

MORALES, S. G.; RONCONI, D. P. Formulações matemáticas e estratégias de resolução para o problema *job shop* clássico. **Production**, v. 26, n. 3, p. 614-625, jul./set. 2016. <http://dx.doi.org/10.1590/0103-6513.058512>.

MÜLLER, F. M.; DIAS, O. B.; ARAÚJO, O. C. B. Algoritmo para o problema de sequenciamento em máquinas paralelas não-relacionadas. **Production**, v. 12, n. 2, p. 6-17, 2002. <http://dx.doi.org/10.1590/S0103-65132002000200002>.

NAGANO, M. S.; MOCCELLIN, J. V. e LORENA, L. A. N. Programação da produção flow shop permutacional com minimização do tempo médio de fluxo. In: SIMPÓSIO BRASILEIRO PESQUISA OPERACIONAL, 36., 2004, São José Del-Rei. **Anais... SOBRAPO**: São José Del-Rei, 2004.

NOSRATI, M.; HAGHIGHI, H.; VAHIDI ASL, M. Using likely invariants for test data generation. **Journal of Systems and Software**, v. 164, 110549, fev. 2020. <https://doi.org/10.1016/j.jss.2020.110549>.

OLIVEIRA, R. A. M. **Importância do planejamento e controle da produção (PCP) dentro de uma organização**: O caso de uma empresa textil. 2014. 30 f. TCC (Graduação) - Curso de Administração, Faculdade Cenecista de Capivari - Facecap, Capivari, 2014.

PANTUZA JUNIOR, Guido. Uma abordagem multiobjetivo para o problema de sequenciamento e alocação de trabalhadores. **Gestão & Produção**, v. 23, n. 1, p. 132-145, mar. 2016. <http://dx.doi.org/10.1590/0104-530X1432-14>

PASQUINI, N. C. Planejamento e controle da produção (PCP): estado da arte. **Revista Tec. Fatec AM**, v. 3, n. 2, p. 81 – 97, mar. 2016.

PEINADO, J.; GRAEML, A. R. **Administração da produção**: Operações Industriais e de Serviços. 1. Ed. Curitiba: UnicenP, 2007.

PENNA, P. H. V. et al. Uma heurística híbrida para minimizar custos com antecipação e atraso do sequenciamento da produção em uma máquina. **Production**, v. 22, n. 4, p. 766-777, dec. 2012. <http://dx.doi.org/10.1590/S0103-65132012005000020>

PFUND, M.; FOWLER, J. W.; GUPTA, J. N. D. A survey of algorithms for single and multi-objective unrelated parallel-machine deterministic scheduling problems. **Journal of the Chinese Institute of Industrial Engineers**, v. 21, n 3, p. 230-241, jan. 2004. <https://doi.org/10.1080/10170660409509404>.

PINEDO, M. L. **Scheduling**: Theory, algorithms, and systems. 3. ed. New York: Springer, 2008.

POTTS, C. N. Analysis of a linear programming heuristic for scheduling unrelated parallel machines. **Discrete Applied Mathematics**, v. 10, n. 2, p. 155-164, feb. 1985. [https://doi.org/10.1016/0166-218X\(85\)90009-5](https://doi.org/10.1016/0166-218X(85)90009-5).

RONCONI, D. P.; KAWAMURA, M. S. The single machine earliness and tardiness scheduling problem: lower bounds and a branch-and-bound algorithm. **Computational & Applied Mathematics**, v. 29, n. 2, p.107-124, jun. 2010. <http://dx.doi.org/10.1590/s1807-03022010000200002>.

RUIZ, R.; RODRÍGUEZ, J. A. V. The hybrid flow shop scheduling problem. **European Journal of Operational Research**, v. 205, n. 1, p.1-18, aug. 2010. <https://doi.org/10.1016/j.ejor.2009.09.024>.

SLACK, N.; CHAMBERS, S.; JOHNSTON, R. **Administração da Produção**. 2. ed. São Paulo: Atlas, 2002.

TAHA, H.A. **Pesquisa operacional: uma visão geral**. 8. ed. São Paulo: Pearson Prendice Hall, 2008.

TRIERWEILLER, A. C. et al. Um modelo de planejamento e controle da produção em uma empresa de mineração em Santa Catarina. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO. 28., 2008, Rio de Janeiro. **Anais...** ABEPRO: Rio de Janeiro, 2008.

TUBINO, D. F. **Planejamento e controle da produção: teoria e prática**. São Paulo: Atlas, 2007.

TURRIONI, J. B.; MELLO, C. H. P. **Metodologia de pesquisa em engenharia de produção**. 1st ed. Itajubá: UNIFEI, 2012.