

**UNIVERSIDADE DE SÃO PAULO**  
**Escola de Engenharia de São Carlos**

---

**REDUÇÃO DO ESTOQUE EM PROCESSAMENTO EM  
SISTEMAS DE PRODUÇÃO *FLOWSHOP*  
PERMUTACIONAL: UMA AVALIAÇÃO DOS MÉTODOS  
HEURÍSTICOS**

*João Carlos Soriano Sampaio Januário*

---



São Carlos - SP

**JOÃO CARLOS SORIANO SAMPAIO JANUÁRIO**

**REDUÇÃO DO ESTOQUE EM PROCESSAMENTO EM  
SISTEMAS DE PRODUÇÃO *FLOWSHOP*  
PERMUTACIONAL: UMA AVALIAÇÃO DOS  
MÉTODOS HEURÍSTICOS**

Trabalho de Conclusão de Curso  
Escola de Engenharia de São Carlos  
Universidade de São Paulo  
Curso de Engenharia Elétrica com ênfase em Eletrônica

ORIENTADOR: Marcelo Seido Nagano

São Carlos

2011

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica preparada pela Seção de Tratamento  
da Informação do Serviço de Biblioteca – EESC/USP

J35r

Januário, João Carlos Soriano Sampaio  
Redução do estoque em processamento em sistemas de  
produção *flowshop* permutacional : uma avaliação dos  
métodos heurísticos / João Carlos Soriano Sampaio  
Januário ; orientador Marcelo Seido Nagano -- São  
Carlos, 2011.

Monografia (Graduação em Engenharia Elétrica com  
ênfase em Eletrônica) -- Escola de Engenharia de São  
Carlos da Universidade de São Paulo, 2011.

1. Pesquisa operacional. 2. *Flowshop* permutacional.  
3. Estoque de processo. 4. Métodos heurísticos. I.  
Titulo.


# FOLHA DE APROVAÇÃO

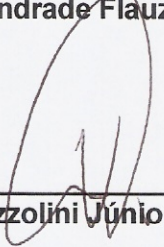
Nome: João Carlos Soriano Sampaio Januário

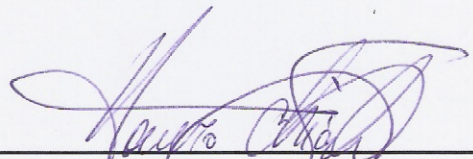
Título: "Redução do Estoque em Processamento em Sistemas de Produção *FlowShop* Permutacional: Uma Avaliação dos Métodos Heurísticos"

Trabalho de Conclusão de Curso defendido e aprovado  
em 6 / 6 / 2011,

com NOTA 10,0 (Dez , zero), pela comissão julgadora:

  
\_\_\_\_\_  
Prof. Dr. Rogério Andrade Flauzino - EESC/USP

  
\_\_\_\_\_  
Prof. Dr. Walther Azzolini Júnior - EESC/USP

  
\_\_\_\_\_  
Prof. Associado *Homero Schiabel*  
Coordenador da CoC-Engenharia Elétrica  
EESC/USP

## DEDICATÓRIA

Por nunca medirem qualquer custo ou esforço para me apoiar, eu dedico este trabalho aos meus pais, Francisco e Andréa. Todas as conquistas e realizações que um dia alcançarei só serão possíveis porque Deus preparou pais tão sábios e dispostos a me educar.

Também dedico este trabalho aos meus amigos Gabriel e Marlow, que um dia foram colegas e hoje são mais que irmãos. Eu os agradeço pela companhia constante, pela paciência em me escutar, pelos conselhos, e por fazerem dos meus anos em São Carlos anos alegres e de profundo aprendizado.

## AGRADECIMENTOS

À Deus por sempre me animar e nunca me deixar esquecer o seu imenso amor por mim. A Ele rendo graças pelos anos difíceis, mas prazerosos, que passei durante minha graduação em São Carlos.

Aos meus irmãos, Thaisa, Larissa, Mariana e Gustavo, que sempre me divertiram e apoiaram, e também aos meus familiares, que mesmo distantes mostraram seu apoio.

Ao Prof Marcelo S. Nagano pela paciência em me orientar ao longo de quase três anos. Por todo o conhecimento e conselhos passados a mim, muito obrigado.

Aos meus amigos de curso. Em especial, Márcio, pelo exemplo de humildade e companheirismo; Fernando, José Arildo e Shamir, pela companhia e amizade, sempre me recebendo na república “do Bahia” e ajudando a superar cada desafio que a graduação apresentasse; Bruno, que em diversas circunstâncias sempre estive presente oferecendo sua amizade. Que as nossas amizades durem ao tempo e à distância.

# SUMÁRIO

<b>LISTA DE FIGURAS</b> .....	<b>I</b>
<b>LISTA DE TABELAS</b> .....	<b>II</b>
<b>RESUMO</b> .....	<b>III</b>
<b>ABSTRACT</b> .....	<b>IV</b>
<b>1. INTRODUÇÃO</b> .....	<b>1</b>
<b>2. O PROBLEMA <i>FLOWSHOP</i> PERMUTACIONAL</b> .....	<b>2</b>
<b>3. REVISÃO DOS MÉTODOS HEURÍSTICOS</b> .....	<b>4</b>
<b>4. DESCRIÇÃO DOS MÉTODOS HEURÍSTICOS</b> .....	<b>8</b>
HEURÍSTICAS SIMPLES .....	8
<i>Framinan e Leisten - FL (Framinan, et al., 2003)</i> .....	8
<i>Laha e Chakraborty - H (Laha, et al., 2008)</i> .....	9
<i>Laha e Chakravorty - LC (Laha, et al., 2010)</i> .....	10
HEURÍSTICAS COMPOSTAS .....	10
<i>Framinan e Leisten – FL-IH7 (Framinan, et al., 2003)</i> .....	11
<i>Framinan, Leisten e Usano – C2 (Framinan, et al., 2005)</i> .....	11
<i>Nagano e Moccellin – NM (Nagano, et al., 2008)</i> .....	12
<i>Li, Wang e Wu – ICH3 (Li, et al., 2009)</i> .....	13
<b>5. MÉTODO DE PESQUISA</b> .....	<b>16</b>
HEURÍSTICA SIMPLES .....	16
HEURÍSTICAS COMPOSTAS .....	22
TODAS AS HEURÍSTICAS .....	28
<b>6. CONSIDERAÇÕES FINAIS</b> .....	<b>37</b>
<b>7. APÊNDICE</b> .....	<b>38</b>
<b>8. REFERÊNCIAS</b> .....	<b>40</b>

## LISTA DE FIGURAS

FIGURA 1 – REPRESENTAÇÃO DO PROBLEMA <i>FLOWSHOP</i> PERMUTACIONAL COM O TEMPO DE FLUXO DA TAREFA .....	3
FIGURA 2 - DESCRIÇÃO DAS FASES DAS HEURÍSTICAS.....	15
FIGURA 3 - PORCENTAGEM DE SUCESSO PARA AS HEURÍSTICAS SIMPLES .....	21
FIGURA 4 - DESVIO RELATIVO MÉDIO PARA AS HEURÍSTICAS SIMPLES.....	21
FIGURA 5 - TEMPO DE PROCESSAMENTO RELATIVO PARA AS HEURÍSTICAS SIMPLES.....	22
FIGURA 6 - PORCENTAGEM DE SUCESSO PARA AS HEURÍSTICAS COMPOSTAS.....	27
FIGURA 7 - DESVIO RELATIVO MÉDIO PARA AS HEURÍSTICAS COMPOSTAS .....	27
FIGURA 8 - TEMPO DE PROCESSAMENTO RELATIVO PARA AS HEURÍSTICAS COMPOSTAS .....	28
FIGURA 9 - PORCENTAGEM DE SUCESSO PARA TODOS OS MÉTODOS AVALIADOS.....	31
FIGURA 10 – DESVIO RELATIVO MÉDIO PARA TODOS OS MÉTODOS AVALIADOS .....	34
FIGURA 11 - TEMPO DE COMPUTAÇÃO ABSOLUTO PARA TODAS AS HEURÍSTICAS.....	35
FIGURA 12 - TEMPO DE COMPUTAÇÃO RELATIVO PARA TODAS AS HEURÍSTICAS .....	35
FIGURA 13 – DIAGRAMA DE GANTT PARA O MÉTODO LR EM CONDIÇÃO DE <i>FLOWSHOP</i> PERMUTACIONAL.....	38



## LISTA DE TABELAS

TABELA 1 - PORCENTAGEM DE SUCESSO E DESVIO RELATIVO MÉDIO PARA PROBLEMAS DE PEQUENO E MÉDIO PORTE PARA AS HEURÍSTICAS SIMPLES .....	17
TABELA 2 – PORCENTAGEM DE SUCESSO E DESVIO RELATIVO MÉDIO PARA PROBLEMAS DE GRANDE PORTE PARA AS HEURÍSTICAS SIMPLES .....	18
TABELA 3 - TEMPO MÉDIO DE COMPUTAÇÃO PARA PROBLEMAS DE PEQUENO E MÉDIO PORTE PARA AS HEURÍSTICAS SIMPLES .....	19
TABELA 4 - TEMPO MÉDIO DE COMPUTAÇÃO PARA PROBLEMAS DE GRANDE PORTE PARA AS HEURÍSTICAS SIMPLES .....	20
TABELA 5 - PORCENTAGEM DE SUCESSO E DESVIO RELATIVO MÉDIO PARA PROBLEMAS DE PEQUENO E MÉDIO PARA AS HEURÍSTICAS COMPOSTAS .....	23
TABELA 6 - PORCENTAGEM DE SUCESSO E DESVIO RELATIVO MÉDIO PARA PROBLEMAS DE GRANDE PORTE PARA AS HEURÍSTICAS COMPOSTAS .....	24
TABELA 7 - TEMPO MÉDIO DE COMPUTAÇÃO EM SEGUNDOS PARA PROBLEMAS DE PEQUENO E MÉDIO PORTE PARA AS HEURÍSTICAS COMPOSTAS .....	25
TABELA 8 - TEMPO MÉDIO DE PROCESSAMENTO EM SEGUNDOS PARA PROBLEMAS DE GRANDE PORTE PARA AS HEURÍSTICAS COMPOSTAS .....	26
TABELA 9-PORCENTAGEM DE SUCESSO PARA PROBLEMAS DE PEQUENO E MÉDIO PORTE.....	29
TABELA 10 - PORCENTAGEM DE SUCESSO PARA PROBLEMAS DE GRANDE PORTE.....	30
TABELA 11 - PORCENTAGEM DE DESVIO RELATIVO MÉDIO PARA PROBLEMAS DE PEQUENO E MÉDIO PORTE .....	32
TABELA 12 - PORCENTAGEM DE DESVIO RELATIVO MÉDIO PARA PROBLEMAS DE GRANDE PORTE.....	33

## RESUMO

Este trabalho trata do problema de Programação de Operações em um ambiente de produção *Flowshop*, tendo como objetivo minimizar o estoque em processamento. Recentemente, vários novos métodos heurísticos foram propostos para solução do problema, e muitos deles não foram avaliados em conjunto para identificar o melhor método heurístico. Por esse motivo, o presente trabalho tem por objetivo principal avaliar por meio de uma extensa experimentação computacional todos os melhores métodos heurísticos reportados na atual literatura. Os resultados finais da experimentação computacional identificaram os melhores métodos para duas categorias de heurísticas (Métodos Heurísticos Simples e Métodos Heurísticos Compostos) para o conjunto de problemas tratados.

**Palavras chave:** *Flowshop* permutacional, Estoque de processo, Métodos heurísticos.

## ABSTRACT

This work deals with the traditional permutation *flowshop* scheduling problem with the objective of minimizing total *flowtime*, therefore reducing in-process inventory. Recently, several of new heuristics have been developed for the scheduling problem solution, and many of them have not evaluated in conjunction for identify the best one. Therefore, the main objective of this work is compare computationally all the best heuristics described at literature. The final results show the best heuristic for two categories, Simple and Compose.

**Keywords:** Permutation *flowshop*, In-process inventory, Heuristics.

## 1. Introdução

---

Ao longo dos anos a competitividade do mercado passou a exigir o controle da produção cada vez mais fino, o que resultou na busca por uma melhoria no desempenho dos sistemas produtivos, e de todos os indicadores, para assim, reduzir tanto o tempo total da programação como também o estoque em processamento. Por esta importância, o problema de programação da produção tem sido intensamente estudado na literatura.

Desde o primeiro trabalho reportado por (Johnson, 1954) na obtenção da solução ótima para o problema com duas máquinas, diversos métodos já foram propostos, abordando as mais diversas formas de algoritmos que vão de heurísticas construtivas até metaheurísticas do tipo: Algoritmos Genéticos, Busca Tabu, e outros. Essa diversidade de métodos se justifica pela natureza do problema que é classificada como sendo NP-hard, assim, a maioria dos estudos nessa área fornece bons resultados, mas não as soluções ótimas.

Focando apenas nas heurísticas construtivas, percebe-se na literatura diversos métodos propostos de forma independente sem que haja ao certo uma comparação atual entre eles. Dessa forma, este trabalho propõe a seleção dos melhores métodos propostos na literatura atual para um estudo efetivo, concluindo qual é o estado da arte para o problema estudado.

A fim de uma melhor compreensão das atividades realizadas neste projeto, esta monografia é organizada da seguinte maneira:

- Na seção 2 é apresentada com mais detalhes os aspectos envolvendo o problema a ser pesquisado: suas características e particularidades;
- Na seção 3 é feita uma revisão temporal da literatura, indicando como se deu o desenvolvimento das heurísticas que abordam o problema estudado;
- Após uma revisão de toda a literatura, a seção 4 contém as heurísticas selecionadas para a análise, descrição e sua implementação computacional;
- A seção 5 mostra os resultados experimentais da comparação das heurísticas descritas na seção 4;
- Por fim, a seção 6 contém as considerações finais acerca dos resultados obtidos, juntamente com possíveis trabalhos futuros que possam tomar como base este projeto.

## 2. O Problema *Flowshop* Permutacional

---

A Programação da Produção pode ser definida de maneira geral como a alocação de recursos através do tempo. Esses recursos podem ser entendidos como sendo matérias primas, mão de obra, ferramentas, equipamentos, por isso, esse termo é genericamente chamado de “máquinas” em várias situações na literatura (Baker, 1974).

Taillard (1993) define em linhas gerais o problema de programação da produção, no qual um conjunto de  $n$  tarefas  $\{J_1, J_2, \dots, J_n\}$  devem ser processadas em  $m$  máquinas  $\{M_1, M_2, \dots, M_m\}$  que estão disponíveis. Esses problemas podem ser classificados de acordo com as restrições tecnológicas e também pelos objetivos da programação no sistema produtivo, assim, uma das classificações possíveis é o *flowshop* permutacional, onde a ordem de processamento de todas as  $n$  tarefas nas  $m$  máquinas é a mesma.

O sistema de produção *flowshop* permutacional tem por objetivo determinar entre as  $n!$  possíveis seqüências aquela que minimiza alguma função objetivo ou critério estabelecido, sendo as mais usuais a duração total da programação (*makespan*) e a soma total dos tempos de fluxo das tarefas (*total flowtime*). A primeira dessas funções está relacionada com o uso eficiente dos recursos de produção, e a segunda com a velocidade do processamento de um conjunto de tarefas, reduzindo, assim, o estoque em processamento. Nagano, et al. (2008) listaram algumas hipóteses que devem ser adotadas para a modelagem nesse ambiente de produção:

- Cada estágio de produção tem somente uma específica máquina;
- As tarefas possuem um seqüenciamento idêntico para todas as máquinas;
- Os tempos de processamento das tarefas nas máquinas são conhecidos e fixos;
- Os tempos de *setup* das máquinas estão incluídos nos tempos de processamento das tarefas, e são independentes da seqüência;
- Para um determinado tempo, cada tarefa é processada em apenas uma máquina, e cada máquina processa apenas uma tarefa;
- Sempre que uma operação é iniciada, a mesma é finalizada sem que haja interrupção no seu processamento;
- Cada operação tem, no máximo, uma sucessora e uma precedente (fluxo unidirecional).

Neste trabalho mais uma hipótese é considerada: a data de liberação de cada tarefa será zero, assim, o tempo de conclusão de uma tarefa e o seu tempo de fluxo serão equivalentes. Isso permite com que a soma dos tempos de fluxo, ou seja, a soma do tempo necessário para que cada tarefa possa ser processada por cada uma das máquinas, seja adotada como critério de avaliação para este trabalho. Esta soma é apresentada pela expressão 1.

$$F = \sum_{i=1}^n C_{im} \quad (1)$$

Na expressão 1,  $C_{im}$  é o tempo de fluxo da tarefa  $i$  ao passar por todas as  $m$  máquinas no processo. Uma exemplificação desse conceito pode ser visualizada na Figura 1.

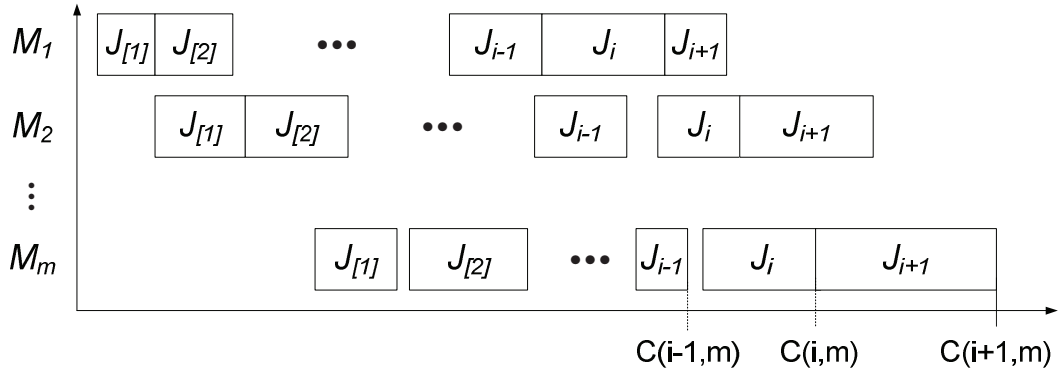


Figura 1 – Representação do problema *flowshop* permutacional com o tempo de fluxo da tarefa

Seguindo uma descrição de Nagano, et al. (2008) o problema de seqüenciamento *flowshop* permutacional para  $n$  tarefas e  $m$  máquinas pode ser definido matematicamente a seguir.

Seja  $\sigma = J_{[1]}, J_{[2]}, \dots, J_{[n]}$  a seqüência de tarefa a serem programadas nas máquinas, onde  $J_{[i]}$  é a tarefa  $i$ -ésima da seqüência em  $\sigma$ . O tempo de processamento da tarefa  $J_{[i]}$  na máquina  $j$  ( $j=1,2,\dots,m$ ) é definido por  $p_{ij}$ .

Pelas hipóteses adotadas para o problema, o tempo de início da tarefa  $J_{[i]}$  na máquina  $j$  é dado pela expressão 2.

$$S_{ij} = \max\{C(i, j - 1), C(i - 1, j)\} \quad (2)$$

Onde,  $C(i, j - 1)$  é considerado como sendo o tempo de fluxo da tarefa  $J_{[i]}$  na máquina  $(j - 1)$ , e  $C(i - 1, j)$  o tempo de fluxo da tarefa  $J_{[i-1]}$  na máquina  $j$ .

Como as tarefas em um problema de programação de produção podem estar precedidas por outras, o seu tempo de fluxo será calculado de forma recursiva dado pelas expressões 3 à 5.

$$C(i, j) = \max\{C(i, j - 1), C(i - 1, j)\} + p_{ij} \quad (3)$$

Onde,

$$C(i, 0) = 0 \quad (4)$$

$$C(0, j) = 0 \quad (5)$$

Pelo uso da recursividade é possível se chegar ao tempo de fluxo de cada uma das tarefas  $J_{[i]}$ ,  $i=1, 2, \dots, n$ , em todas as máquinas do processo. Assim, se obtém  $C(i, m)$ , que será usado na expressão 1 para calcular o soma dos tempo de fluxo.

### 3. Revisão dos Métodos Heurísticos

---

Nos últimos anos o estudo da minimização do tempo de fluxo em ambiente *flowshop* permutacional foi tema de importantes trabalhos publicados na área de seqüenciamento de tarefas. Uma breve descrição de como se procedeu o desenvolvimento dos métodos heurísticos nesse período é apresentada a seguir.

Rajendran et al. (1991) desenvolveram três heurísticas que obtiveram melhor desempenho que os métodos heurísticos propostos por Gupta (1972) e Miyazaki et al. (1978), em termos da qualidade de solução e também quanto ao esforço computacional. Na continuidade de suas pesquisas, Rajendran (1993) propôs um novo método que apresentou melhor desempenho do que as heurísticas acima mencionadas em termos de qualidade da solução, porém com alto esforço computacional.

Ho (1995) propôs uma heurística composta de diferentes iterações no processo de melhoria de uma solução inicial, a partir da obtenção de um ótimo local por permutação de pares de tarefas adjacentes, melhorando posteriormente a solução por movimentos de inserção de tarefas. Tal método heurístico apresentou desempenho melhor que os anteriores reportados na literatura, embora, como esperado, com desvantagem quanto ao tempo de computação, pelo fato de não ser uma heurística construtiva, apresentando semelhanças com metaheurísticas como *Simulated Annealing* e Busca Tabu. Em 2003 esse método foi novamente estudado em conjunto com diversos outros métodos heurísticos por Framinan et al. (2003). Neste trabalho foi mostrado para a maioria dos casos numéricos estudados que vários desses métodos possuem um desempenho pior que uma seqüência formada aleatoriamente para um conjunto usual de parâmetros, sendo a única exceção a heurística proposta por Ho (1995).

Rajendran et al. (1997) propuseram a heurística RZ, que consiste de duas fases: na primeira, uma seqüência inicial é gerada utilizando uma regra de prioridade similar ao *shortest weighted total processing time*; na segunda fase, a seqüência inicial obtida é melhorada por meio de inserções das tarefas em seqüências parciais, sucessivamente obtidas.

Wang et al. (1997) desenvolveram duas heurísticas denominadas LIT (*less idle time*) e SPD (*smallest process distance*). Suas heurísticas não são comparadas com as existentes, mas sim com um limitante inferior do tempo médio de fluxo, proposto por Ahmadi et al. (1990).

Woo et al. (1998) desenvolveram uma heurística denominada WY, comparando seu desempenho com a heurística de (Rajendran, 1993), e também com adaptações do NEH (Nawaz, et al., 1983) e CDS (Campbell, et al., 1970), para o critério de minimização do tempo médio de fluxo. Eles verificaram que seu método apresentava desempenho superior aos outros, principalmente comparado com Rajendran (1993). Muitas comparações foram feitas entre os métodos WY e RZ, (Woo e Yim, 1998) e (Rajendran e Ziegler, 1997), respectivamente, indicando que para problemas com um número pequeno de tarefas RZ apresenta um desempenho melhor, enquanto que o oposto ocorre quando o número de tarefas cresce. Com respeito ao tempo de computação, WY é bem mais lento que RZ.

Liu e Reeves (2001) apresentaram uma heurística baseada no desenvolvimento de uma função de indexação para ordenar as tarefas de uma dada seqüência. Esta heurística pode ainda conter a geração de  $x$  diferentes seqüências obtidas pelo uso de cada uma das  $x$  primeiras tarefas ordenadas inicialmente, e, dessa forma, se constrói a seqüência final selecionando as tarefas uma a uma pela função de indexação. Dessas seqüências, aquela que apresentar a menor soma dos tempos de fluxo é submetida a busca *pairwise exchange*.

Allahverdi e Aldowaisan (2002) apresentaram sete heurísticas denominadas IH $x$  ( $x = 1,2,\dots,7$ ). O melhor desempenho foi alcançado pela heurística IH7, a qual é constituída de três fases. Na primeira fase, uma solução inicial é obtida aplicando-se a heurística WY (Woo e Yim, 1998). Esta primeira solução é utilizada na segunda fase como seqüência inicial para a heurística RZ (Rajendran e Ziegler, 1997). Finalmente, a segunda solução é melhorada por meio de uma busca local, com procedimentos de permutação nas posições das tarefas.

Framinan, Leisten e Rajendram (Framinan, et al., 2003) propuseram um trabalho investigando a extensão da heurística NEH (Nawaz, et al., 1983), para a minimização do tempo de fluxo total. Nesse estudo os autores investigaram 177 diferentes abordagens de ordenação das tarefas na primeira fase do NEH. Todas essas abordagens foram examinadas experimentalmente, e se concluiu que o melhor resultado foi obtido ordenando as tarefas pela ordem decrescente da soma dos valores absolutos das diferenças dos tempos de processamento de uma tarefa com todas as outras, considerando cada máquina do problema. Neste mesmo trabalho também foram avaliadas todas as possíveis combinações de cinco diferentes ordenações iniciais, sendo a melhor delas usada com solução inicial do NEH, o que gerou uma adaptação do método que apresentou soluções de melhor qualidade que as heurísticas WY (Woo e Yim, 1998) e RZ (Rajendran e Ziegler, 1997).

Framinan e Leisten (2003) propuseram o método simples denominado FL e com ele uma modificação do IH7 (Allahverdi e Aldowaisan, 2002), FL-IH7. Essa modificação altera o modo que se obtém a solução inicial do processo de três fases do IH7. Em FL-IH7 não mais é obtida a solução inicial por WY, mas sim pelo método proposto FL, que utiliza, inicialmente, o mesmo procedimento de obtenção de seqüências parciais do NEH (Nawaz, et al., 1983), porém ordenando as tarefas de acordo com a soma dos tempos de processamento não-decrescentes. A seguir, as seqüências parciais são melhoradas por um procedimento de busca completa nas respectivas Vizinhanças de Permutação, *pairwise interchange*. Por meio de uma experimentação computacional constatou-se que FL é superior as heurísticas WY e RZ, e FL-IH7 apresenta um desempenho melhor quando comparado com IH7 original.

Framinan, Leisten e Usano (2005) classificaram as heurísticas do problema *flowshop* com critério de minimização do tempo de fluxo total em duas categorias: heurísticas simples e compostas de acordo com a estrutura na qual são propostas. Seguindo essa classificação eles propuseram duas heurísticas compostas: C1 e C2. A primeira, C1, considera como solução inicial aquela gerada pelo método LR, Liu e Reeves (2001), seguida pelo método construtivo FL, Framinan e Leisten (2003). A



heurística C2 vem a complementar C1, adicionando RZ (Rajendran e Ziegler, 1997), mais uma busca local usando Vizinhança de Inserção. A heurística C2 foi comparada com os métodos FL e FL-IH7, como também com IH7 original, Allahverdi e Aldowaisan (2002), e foi comprovada sua superioridade.

Nagano e Moccellini (2008) desenvolveram um método cuja fase de melhoramento consistia de uma composição de duas formas de busca nas vizinhanças aplicadas de forma sucessivas na seqüência parcial: a primeira forma de busca na vizinhança é *shift neighbourhood*, e a segunda a *interchange neighbourhood*. Esta heurística foi comparada com o método C2, e apresentou resultados melhores tanto em relação ao tempo de computação quanto a qualidade das soluções. No mesmo ano, Laha e Chakraborty (2008), introduziram na literatura a heurística simples H que inicialmente foi utilizada apenas como parte de uma das fases das metaheurísticas H-1 e H-2. Em Laha e Sarin (2009) o método H foi novamente apresentado e comprovaram sua superioridade com a melhor heurística simples existente, FL. Entretanto a heurística H não foi comparada com a C2, o que não permite definir qual delas apresenta melhor desempenho.

Li, Wang e Wu (2009) propuseram três heurísticas compostas pelos métodos FL (Framinan e Leisten, 2003), RZ (Rajendran e Ziegler, 1997), e de buscas locais derivadas das Vizinhanças de Permutação e Inserção para o melhoramento das soluções. Através de um estudo detalhado das fases que compõe um método heurístico os autores determinaram quais os métodos que apresentavam os melhores desempenhos. Em suas diferentes combinações apresentaram as heurísticas compostas ICH1, ICH2 e ICH3, sendo ICH3 o que apresentou o melhor desempenho mesmo quando comparado com os melhores métodos até o momento, FL-IH7 e C2.

Recentemente, Laha e Chakraborty (2010) desenvolveram um método chamado de LC, e compararam com as heurísticas WY e FL. Pela comparação os autores puderam verificar a superioridade do método proposto em relação à qualidade das soluções e o esforço computacional.

De acordo com a revisão da literatura efetuada neste trabalho percebe-se que vários métodos propostos para o problema de minimização do tempo de fluxo total foram elaborados de forma independente, não sendo realizada uma comparação efetiva entre eles. O único trabalho que busca comparar os principais métodos da literatura é Framinan, et al., 2005, mas ele se apresenta desatualizado, pois considera o C2 como melhor método para a programação *flowshop* permutacional com critério de minimização do tempo de fluxo total. Outro aspecto importante a ser levantado é referente ao desempenho do método proposto por Nagano e Moccellini (2008) que apesar de apresentar resultados superiores que o C2 não foi considerado pelos demais pesquisadores para fins de comparação.

Considerando os aspectos apresentados acima referentes à revisão realizada, este trabalho tem como objetivo identificar e comparar os melhores métodos heurísticos segundo a classificação apresentada por Framinan et al. (2005), sendo esta classificação dividida em duas categorias: Métodos Heurísticos Simples e Métodos Heurísticos Compostos.

O capítulo a seguir tem como objetivo apresentar os métodos a serem implementados e avaliados para esta pesquisa.

## 4. Descrição dos Métodos Heurísticos

---

Devido ao grande número de métodos heurísticos propostos para o problema de minimização da soma dos tempos de fluxo é conveniente classificá-los em categorias distintas, nas quais se leva em consideração a estrutura do método proposto, indicando se há ou não fase de melhoramento da solução. Seguindo essa proposta, Framinan et al. (2005) sugeriu que os métodos heurísticos sejam classificados dentro de três fases: desenvolvimento (*index development*), construção da solução (*solution construction*) e melhoramento da solução (*solution improvement*).

Todos os métodos propostos podem ser desenvolvidos dentro dessas três fases, contendo uma ou mais. Na fase de desenvolvimento da solução há uma ordenação das tarefas através de um critério específico do problema, nela não se procura a minimização da soma dos tempos de fluxo, o que se busca é apenas uma ordenação simples que servirá como base para o desenvolvimento do método. A fase de construção da solução é onde o método se caracteriza, mas, apesar de prover a estrutura do método, ela pode não garantir uma seqüência final com uma avaliação melhor que a seqüência gerada pela primeira fase. A última fase, melhoramento da solução, se compromete, diferente das outras, a explorar a seqüência formada até o momento, procurando obter a melhor disposição segundo o critério de avaliação escolhido. Esta última fase faz uso de alguma busca local, ou da associação delas, objetivando alcançar de forma eficiente a melhor seqüência para o problema abordado. Além das três fases, cada método heurístico pode ser classificado de acordo com o uso ou não de algum outro método já existente. Dessa forma, há a classificação em métodos compostos e métodos simples.

Este trabalho tem como objetivo fazer um estudo dos principais métodos heurísticos propostos na literatura, classificando-os segundo os critérios aqui apresentados. Uma análise prática e didática das heurísticas analisadas indicará qual é o estado da arte para o problema. Assim, a seguir serão descritos os métodos a serem implementados para fins de comparação.

### Heurísticas Simples

Através de uma análise na literatura, as heurísticas descritas como as mais promissoras e, assim, as que foram selecionadas para este trabalho são apresentadas abaixo.

#### Framinan e Leisten - FL (Framinan, et al., 2003)

##### Fase de Desenvolvimento

*Passo 1:* Para cada tarefa  $i$ , calcule o tempo total de processamento  $T_i$  que é dado pela expressão 6.

$$T_i = \sum_{j=1}^m t_{ij} , \quad \text{Para } i = 1, 2, \dots, n. \quad (6)$$

*Passo 2:* Ordene as tarefas em ordem ascendente dos valores de  $T_i$  para todas as máquinas.

### Fase de Construção da Solução

*Passo 1:* Considere  $k=2$ . Dada a seqüência ordenada pela primeira fase, selecione as duas primeiras tarefas e escolha como seqüência parcial aquela que obter o menor valor da soma dos tempos de fluxo.

*Passo 2:* Faça  $k=k+1$ . Selecione a tarefa da posição  $k$  da seqüência gerada pela primeira fase e insira em todas as  $k$  possíveis posições da seqüência parcial. Dentre as  $k$  seqüências parciais, escolha a melhor pelo critério de tempo de fluxo.

*Passo 3:* Dada a seqüência parcial faça uma permutação entre as tarefas da posição  $i$  e da posição  $j$ , sendo  $1 \leq i < k$  e  $i < j \leq k$ . Dessa permutação, selecione pelo critério de tempo de fluxo total a melhor dentre todas as possíveis seqüências.

*Passo 4:* Pare se  $k=n$ , caso contrário retorne para o *Passo 2*.

Por fim, a seqüência final será igual a seqüência parcial obtida para  $k=n$ .

Uma vez que o cálculo do tempo de fluxo total para uma seqüência de  $k$  tarefas é  $O(km)$ , a complexidade deste método é  $O(n^4 m)$ , pois a fase de construção da solução é executada para todo  $k=2, 3, \dots, n$  o que gera uma complexidade  $O(kmkk(k+1)/2)$  ou  $O(n^4 m)$ .

### **Laha e Chakraborty - H (Laha, et al., 2008)**

#### Fase de Desenvolvimento

*Passo 1:* Para cada tarefa  $i$ , calcule o tempo total de processamento  $T_i$  que é dado pela expressão 6.

*Passo 2:* Ordene as tarefas em ordem ascendente dos valores de  $T_i$  para todas as máquinas.

#### Fase de Construção da Solução

*Passo 1:* Considere  $k=2$ . Da seqüência ordenada pela primeira fase selecione as duas primeiras tarefas e pela escolha como seqüência parcial aquela que obter o menor valor da soma dos tempos de fluxo.

*Passo 2:* Faça  $k=k+1$ . Selecione a tarefa na posição  $k$  da seqüência gerada pela primeira fase e a insira em todas as  $k$  possíveis posições da seqüência parcial em formação. Dentre essas  $k$  seqüências, escolha a melhor pelo critério de tempo de fluxo como a seqüência parcial.

*Passo 3:* Dada uma seqüência parcial selecione cada tarefa, exceto a tarefa já trabalhada no *Passo 2*, e a insira em todas as suas  $k-1$  possíveis posições. De todas as seqüências formadas determine a melhor pelo critério de tempo de fluxo total e a indique como sendo a nova seqüência parcial.

*Passo 4:* Pare se  $k=n$ , caso contrário retorne para o *Passo 2*.

Ao fim da heurística a seqüência final será assumida como sendo a seqüência parcial obtida para o caso de  $k=n$ . Sua complexidade é a mesma que a heurística de Framinan and Leisten, (Framinan, et al., 2003),  $O(n^4m)$ .

### **Laha e Chakravorty - LC (Laha, et al., 2010)**

#### Fase de Desenvolvimento

*Passo 1:* Para cada tarefa  $i$ , calcule o tempo total de processamento  $T_i$  que é dado pela expressão 6.

*Passo 2:* Ordene as tarefas em ordem ascendente dos valores de  $T_i$  para todas as máquinas.

#### Fase de Construção da Solução

*Passo 1:* Selecione as quatro primeiras tarefas da seqüência ordenada pela primeira fase e gere seqüências pela permutação dessas tarefas. Das 24 (vinte e quatro),  $4!$ , seqüências obtidas separe as melhores  $k$  pelo critério de menor tempo de fluxo total.

*Passo 2:* Faça  $z=5$ . A tarefa cuja posição é  $z$  na seqüência ordenada na primeira fase será inserida em todas as  $z$  posições de todas as  $k$  seqüências formadas pelo *Passo 1*. Entre todas as  $z \times k$  seqüências formadas serão escolhidas as  $k$  melhores pelo critério de menor tempo de fluxo total.

*Passo 3:* Faça  $z=z+1$ . Se  $z=n$  siga para o *Passo 4*, caso contrário retorne ao *Passo 2*.

*Passo 4:* Entre todas as  $k$  seqüências formadas até o momento, considere a melhor delas como sendo a seqüência  $b$ . Para todas as  $k$  seqüências, selecione cada tarefa e a insira em todas as possíveis posições de sua respectiva seqüência, atualizando  $b$  caso a seqüência gerada seja melhor.

Depois de percorrer todas as tarefas de todas as  $k$  seqüências  $b$  será a seqüência final.

A complexidade dessa heurística é determinada principalmente pelos *Passo 2* e *Passo 4*. No *Passo 2* serão geradas  $z \times k$  seqüências, sendo  $z=5, 6, \dots, n$ , e sua complexidade será  $O(zm \cdot z \cdot zk)$  ou  $O(n^3m)$ . Para o *Passo 4*, a complexidade é assumida com sendo  $O(n(n-1) \cdot k \cdot nm)$  ou  $O(n^3m)$ . Dessa forma, a complexidade da heurística proposta é  $O(n^3m)$ .

### **Heurísticas Compostas**

A junção de métodos heurísticos, característica das heurísticas compostas, geralmente acarreta em resultados melhores que os das heurísticas simples para a maior parte dos casos, entretanto, se perde no esforço computacional. Dentro da literatura as heurísticas que melhor relacionam a qualidade dos resultados com o tempo de computação necessário para sua execução estão postas a seguir.

### **Framinan e Leisten – FL-IH7 (Framinan, et al., 2003)**

Essa heurística é uma modificação da já existente IH7, proposta por Allahverdi e Aldowaisan (2002). Através de um estudo prévio, comparando a função WY proposta por Woo e Yim (2005), com a função FL proposta por Framinan e Leisten (2003) é possível verificar a superioridade da função FL. Dessa forma, no desenvolvimento da heurística FL-IH7 ao invés de se formar a seqüência inicial pelo método WY, usa-se FL e como conseqüência seus resultados se mostraram melhores que os de IH7. Além do método FL, a heurística FL-IH7 também utiliza a fase de construção do método RZ e a busca local FPE-R (*forward pairwise exchange – restart*).

O método FL é o mesmo descrito na seção de heurísticas simples, já a fase utilizada do método RZ é apenas a aplicação vizinhança de inserção em uma dada seqüência considerando a ordem da seqüência inicial. O último método inserido na heurística FL-IH7 é o FPE-R que se caracteriza por trocar cada tarefa posicionada em  $k$  da seqüência inicial com todas as posições posteriores a  $k$ . Por ter a característica de *restart* todas as vezes que for encontrada uma solução melhor que a anterior o método se inicia,  $k=1$ , sendo somente findado quando nenhuma melhora for encontrada após  $k$  ser igual a  $n$ . Após este último método, a melhor seqüência pelo critério de minimização do tempo de fluxo será a seqüência final da heurística.

As fases desse método podem ser descritas da seguinte forma.

#### Fase de Desenvolvimento

*Passo 1:* Aplique o método FL para obter uma primeira ordenação das tarefas.

#### Fase de Construção da Solução

*Passo 1:* Utilize a fase de construção do método RZ para construir uma seqüência a partir daquela fornecida pelo *Passo 1*.

#### Fase de Melhoramento da Solução

*Passo 1:* Use a busca local FPE-R.

Essa heurística terá a mesma complexidade que a heurística que a originou, IH7,  $O(n^4m)$  que é a complexidade do método FL usando a soma total dos tempos de fluxo como critério de avaliação.

### **Framinan, Leisten e Usano – C2 (Framinan, et al., 2005)**

A geração de uma seqüência base para o desenvolvimento da heurística é feita com a utilização do método LR, proposto por Liu e Reeves (2001). Este método se destaca por considerar não apenas a influência que cada tarefa tem no desempenho de uma dada seqüência parcial, mas também a sua importância para a seqüência como um todo já que sua escolha irá gerar tempos de ociosidades distintos nas máquinas. Outro ponto que esse método ressalta é o grau de importância da

posição de uma tarefa na determinação do tempo de fluxo de uma máquina. Uma forma de considerar todas essas influências é através de uma função de indexação (*index function*) que será usada para classificar cada tarefa e, assim, determinar sua posição na seqüência. O método LR vem acompanhado de um parâmetro  $x$ ,  $LR(x)$ , para indicar o número de vezes que o procedimento da heurística irá se repetir após uma ordenar inicial feita pela função de indexação. Uma explicação mais detalhada da estrutura da função de indexação junto com o procedimento do método  $LR(x)$  está posta no Apêndice.

Outros métodos acoplados a essa heurística são os métodos FL, RZ e a busca local FIE-R. O método FL será utilizado da mesma forma que o RZ foi usado na heurística FL-IH7, ou seja, apenas a fase de construção será inserida na heurística C2, já FIE-R (*forward insertion exchange – restart*), se caracteriza por ser uma busca de inserção, assim, seguindo a ordem da seqüência inicial cada uma das tarefas será inserida em todas as posições posteriores a essa dada tarefa. A propriedade de *restart* garante que toda às vezes em que a busca encontrar uma solução melhor que a inicial o método FIE-R se inicia. Ao fim do método a melhor seqüência pela minimização do tempo de fluxo total será a seqüência final.

O detalhamento deste método em fases é apresentada abaixo.

#### Fase de Desenvolvimento

*Passo 1:* Aplique o método  $LR(x)$ , para  $x \in \{1, 1/n, n\}$ . A melhor seqüência dentre as formadas por cada  $x$  será tida como seqüência inicial.

*Passo 2:* O método FL será empregado na solução fornecida pelo *Passo 1* para gerar a seqüência inicial da heurística C2.

#### Fase de Construção da Solução

*Passo 1:* A partir da seqüência inicial utilize a fase de construção do método RZ.

#### Fase de Melhoramento da Solução

*Passo 1:* Basta aplicar o busca local FIE-R para resultar na solução final da heurística C2.

### **Nagano e Moccellin – NM (Nagano, et al., 2008)**

No decorrer das três fases da heurística NM os métodos que são utilizados para a geração de novas seqüências são a heurística NEH adaptada para tempo de fluxo junto com o uso de duas buscas locais, a SN (*shift neighbourhood*) seguida da IN (*interchange neighbourhood*).

#### Fase de Desenvolvimento

*Passo 1:* Para cada tarefa  $i$ , calcule o tempo total de processamento  $T_i$  que é dado pela expressão 7.

$$T_i = \sum_{j=1}^m t_{ij} , \quad \text{Para } i = 1, 2, \dots, n. \quad (7)$$

*Passo 2:* Ordene as tarefas em ordem ascendente dos valores de  $T_i$  para todas as máquinas.

### Fase de Construção da Solução

Para utilizar a adaptação da heurística NEH devem-se seguir os seguintes passos.

*Passo 1:* Seja  $k=2$ . Das duas primeiras tarefas da seqüência inicial selecione a melhor disposição pelo critério de minimização do tempo de fluxo total e a considere como a seqüência parcial da heurística.

*Passo 2:* Faça  $k=k+1$ . Insira a tarefa localizada na posição  $k$  da seqüência inicial em todas as  $k$  posições da seqüência parcial. Esta seqüência parcial será atualizada com a melhor entre todas as possíveis seqüências geradas pela inserção da tarefa  $k$ .

*Passo 3:* Se  $k=n$ , assuma a seqüência parcial como a seqüência final, caso contrário retorne ao Passo 2.

### Fase de Melhoramento da Solução

*Passo 1:* Considere  $j=3$ .

*Passo 2:* Para a seqüência formada pelas primeiras  $j$  tarefas da seqüência inicial, aplique a busca SN seguida pela busca IN. Ao fim de cada busca atualize a seqüência parcial com a melhor solução encontra segundo o critério de tempo de fluxo total.

*Passo 3:* Faça  $j=j+1$  e retorne ao Passo 2 caso  $j$  seja menor ou igual a  $n$ . Se  $j=n+1$  assuma a seqüência parcial já alcançada como a seqüência final da heurística.

A complexidade dessa heurística é ditada por seu último estágio, onde a combinação de buscas locais provoca uma complexidade de  $O(n^3)$ . Como ainda é preciso considerar a complexidade do critério de avaliação pelo tempo de fluxo total, a complexidade da heurística é tida com o sendo  $O(n^4 m)$ .

### **Li, Wang e Wu – ICH3 (Li, et al., 2009)**

A heurística ICH3 foi proposta como uma heurística iterativa, ou seja, ao fim do desenvolver de seus passos se a seqüência encontrada for melhor que a inicial todo o processo é repetido, considerando apenas essa nova seqüência como a seqüência inicial. Da mesma forma que a heurística C2 esse método também utiliza o procedimento LR e a fase de construção de RZ, aplicando em sua última fase a busca local FPE-R (*forward pairwise exchange – restart*). Esta busca é semelhante à aplicada na heurística C2, a diferença se encontra no tipo de vizinhança usada para o melhoramento da



solução, isso porque neste caso se utiliza uma vizinhança permutação enquanto que em C2 é usado uma vizinhança de inserção.

#### Fase de Desenvolvimento

*Passo 1:* Aplique o método LR( $x$ ), para  $x \in \{n/m\}$ . Nos casos em que essa razão não for inteira usa-se o inteiro mais próximo, desde que seu valor seja maior ou igual a um.

#### Fase de Construção da Solução

*Passo 1:* Use a fase de construção do método RZ para construir uma seqüência a partir daquela fornecida pela primeira fase.

#### Fase de Melhoramento da Solução

*Passo 1:* Aplicar o busca local FPE-R.

Por sua característica iterativa, os passos da heurística ICH3 serão repetidos até nenhuma melhora ser mais possível ou então até um tempo limite ser alcançado. Para este trabalho, assim como proposto por Li et al. (2009), será aceito como critério de parada um número máximo de 20 iterações para a heurística.

O esforço computacional dessa heurística é mais influenciado pela busca local aplicada em sua última fase, pois sua complexidade é  $O(n^4cm)$ , onde  $c$  é uma constante que depende do número de iterações necessárias para uma dada instância. Considerando o pior caso, temos que a complexidade de FPE-R é  $O(n^4m)$ , enquanto que para a mesma situação a complexidade de RZ é  $O(n^3m)$ . Assim, conclui-se que a complexidade de ICH3 é  $O(n^4m)$

Todos os métodos apresentados neste trabalho podem ser sumarizados pela Figura 2.

		FASES		
		Desenvolvimento	Construção	Melhoramento
Heurísticas Simples	FL	Ordenação	FL-C	
	H	Ordenação	H-C	
	LC	Ordenação	LC-C	
Heurísticas Compostas	FL-IH7	FL	RZ-C	FPE-R
	C2	LR(x) / FL	RZ-C	FIR-R
	NM	Ordenação	NEH- <i>flowtime</i>	SN / IN
	ICH3	LR(x)	RZ-C	FPE-R

**Figura 2 - Descrição das fases das heurísticas**

Na Figura 2 é importante observar três pontos. Primeiro é o índice *C* que vem a indicar o uso da fase construtiva de determinado método; segundo é o método *NEH-flowtime* usado para descrever a adaptação feita na heurística *NEH* para tempo de fluxo; por último, destaca-se na fase de desenvolvimento a ordenação das tarefas em ordem ascendente da soma dos tempos de processamento de cada máquina.

## 5. Método de Pesquisa

---

Os métodos heurísticos foram analisados e implementados em linguagem de programação C e processados em um microcomputador Intel Core i5 (2,4 GHz) e 4 GB de memória RAM. Na experimentação computacional foram utilizados uma amostra de 5200 problemas com o número de máquinas  $m \in \{5, 10, 15, 20\}$  e o número de tarefas  $n \in \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130\}$ , correspondendo a 100 problemas para cada combinação ( $m \times n$ ). Os tempos de processamento das operações são números inteiros gerados aleatoriamente a partir de uma distribuição uniforme no intervalo [1, 99]. Os problemas testes foram classificados em duas classes: problemas de Pequeno e Médio Porte com o número de tarefas variando entre 10 a 70, e problemas de Grande Porte com o número tarefas variando entre 80 a 130.

As estatísticas usadas para avaliar o desempenho dos métodos foram a Porcentagem de Sucesso e o Desvio Relativo Médio. A primeira é definida pelo quociente entre o número total de problemas para os quais o método obteve o melhor tempo médio de fluxo e o número total de problemas resolvidos. Obviamente, quando os dois métodos obtêm o melhor tempo médio de fluxo para o mesmo problema, suas Porcentagens de Sucesso são simultaneamente melhoradas. Já o Desvio Relativo ( $DR_h$ ) quantifica o desvio que o método  $h$  obtém em relação ao melhor tempo médio de fluxo obtido para um mesmo problema, sendo calculado conforme a expressão 8:

$$DR_h = \frac{\bar{F}_h - \bar{F}_o}{\bar{F}_o} \quad (8)$$

Onde,  $\bar{F}_h$  = tempo médio de fluxo obtido pelo método  $h$ ; e  $\bar{F}_o$  = melhor tempo médio de fluxo obtido pelos métodos, para um determinado problema.

A análise dos resultados se procederá, primeiramente, pelo estudo de cada categoria em separado, primeiro serão consideradas as heurísticas simples, e em seguida as heurísticas compostas. Assim, serão levantadas as análises estatísticas para todos os métodos pertencentes a uma determinada categoria. Após essa análise será feito um estudo no qual todos os métodos serão avaliados em conjunto, sem levar em consideração a categoria pertencente.

### Heurística Simples

As heurísticas simples estudadas nesse trabalho foram a FL, H e LC. Os resultados principais da experimentação computacional são ilustrados nas tabelas Tabela 1 à Tabela 4 e Figura 3 à Figura 5.

As Porcentagens de Sucesso e os Desvios Relativos Médios são sumarizados nas Tabela 1 e Tabela 2, já as Tabela 3 e Tabela 4 apresentam os Tempos Médios de Processamento de cada método.

**Tabela 1 - Porcentagem de Sucesso e Desvio Relativo Médio para problemas de pequeno e médio porte para as heurísticas simples**

Tarefas (n)	Máquinas (m)	FL		H		LC	
		PS	DMR	PS	DMR	PS	DMR
10	5	17,00	1,70	42,00	0,59	91,00	0,04
	10	14,00	1,28	33,00	0,49	98,00	0,01
	15	14,00	0,93	36,00	0,40	96,00	0,01
	20	17,00	0,93	35,00	0,38	93,00	0,03
	Média	15,50	1,21	36,50	0,46	94,50	0,03
20	5	2,00	2,35	40,00	0,54	59,00	0,32
	10	0,00	2,39	55,00	0,32	48,00	0,26
	15	1,00	2,11	49,00	0,33	53,00	0,28
	20	1,00	1,80	42,00	0,33	58,00	0,20
	Média	1,00	2,16	46,50	0,38	54,50	0,26
30	5	2,00	2,19	55,00	0,37	43,00	0,47
	10	1,00	2,81	67,00	0,19	32,00	0,59
	15	2,00	2,69	67,00	0,22	31,00	0,64
	20	0,00	2,30	68,00	0,16	33,00	0,42
	Média	1,25	2,50	64,25	0,23	34,75	0,53
40	5	1,00	2,41	76,00	0,09	23,00	0,56
	10	0,00	3,23	79,00	0,16	21,00	0,85
	15	0,00	2,83	80,00	0,14	20,00	0,73
	20	0,00	2,86	83,00	0,07	17,00	0,74
	Média	0,25	2,83	79,50	0,11	20,25	0,72
50	5	1,00	2,25	72,00	0,13	27,00	0,59
	10	1,00	2,92	83,00	0,08	16,00	0,87
	15	0,00	3,21	82,00	0,10	18,00	0,87
	20	0,00	3,03	79,00	0,11	21,00	0,86
	Média	0,50	2,85	79,00	0,10	20,50	0,80
60	5	0,00	2,20	86,00	0,06	14,00	0,80
	10	0,00	3,01	80,00	0,10	20,00	0,95
	15	0,00	3,08	90,00	0,04	10,00	1,05
	20	0,00	3,13	94,00	0,02	6,00	0,99
	Média	0,00	2,86	87,50	0,06	12,50	0,95
70	5	0,00	2,05	85,00	0,06	15,00	0,61
	10	0,00	2,85	81,00	0,10	19,00	0,97
	15	0,00	3,00	84,00	0,08	16,00	0,93
	20	0,00	3,25	90,00	0,03	10,00	1,10
	Média	0,00	2,79	85,00	0,07	15,00	0,90

**Tabela 2 – Porcentagem de Sucesso e Desvio Relativo Médio para problemas de grande porte para as heurísticas simples**

Tarefas (n)	Máquinas (m)	FL		H		LC	
		PS	DMR	PS	DMR	PS	DMR
80	5	0,00	1,85	89,00	0,03	11,00	0,64
	10	0,00	2,90	83,00	0,05	17,00	0,93
	15	0,00	3,00	86,00	0,04	14,00	1,01
	20	0,00	2,93	91,00	0,04	9,00	0,99
	Média	0,00	2,67	87,25	0,04	12,75	0,89
90	5	0,00	1,85	78,00	0,06	22,00	0,64
	10	0,00	2,65	93,00	0,03	7,00	1,02
	15	0,00	2,99	95,00	0,03	5,00	1,09
	20	0,00	3,10	95,00	0,02	5,00	1,17
	Média	0,00	2,65	90,25	0,03	9,75	0,98
100	5	0,00	1,73	82,00	0,05	18,00	0,64
	10	0,00	2,82	92,00	0,03	8,00	0,99
	15	0,00	3,03	93,00	0,03	7,00	1,16
	20	0,00	3,12	96,00	0,02	4,00	1,28
	Média	0,00	2,67	90,75	0,03	9,25	1,02
110	5	0,00	1,57	88,00	0,03	12,00	0,58
	10	0,00	2,63	95,00	0,02	5,00	1,03
	15	0,00	2,89	95,00	0,02	5,00	1,21
	20	0,00	3,10	100,00	0,00	0,00	1,31
	Média	0,00	2,55	94,50	0,02	5,50	1,03
120	5	0,00	1,70	88,00	0,02	12,00	0,65
	10	0,00	2,54	97,00	0,01	3,00	1,07
	15	0,00	2,86	95,00	0,01	5,00	1,11
	20	0,00	3,05	94,00	0,00	6,00	1,11
	Média	0,00	2,54	93,50	0,01	6,50	0,99
130	5	0,00	1,60	89,00	0,03	11,00	0,64
	10	0,00	2,40	96,00	0,02	4,00	1,06
	15	0,00	2,80	96,00	0,01	4,00	1,14
	20	0,00	3,00	97,00	0,01	3,00	1,24
	Média	0,00	2,45	94,50	0,02	5,50	1,02

**Tabela 3 - Tempo Médio de Computação para problemas de pequeno e médio porte para as heurísticas simples**

Tarefas (n)	Máquinas (m)	FL	H	LC
10	5	0,013	0,005	0,051
	10	0,026	0,026	0,101
	15	0,046	0,04	0,162
	20	0,057	0,047	0,206
	Média	0,142	0,118	0,52
20	5	0,127	0,148	0,295
	10	0,317	0,314	0,671
	15	0,475	0,507	1,138
	20	0,569	0,674	1,546
	Média	1,488	1,643	3,65
30	5	0,537	0,672	0,95
	10	1,248	1,538	2,298
	15	2,004	2,479	3,605
	20	2,714	3,278	4,727
	Média	6,503	7,967	11,58
40	5	1,615	2,31	2,154
	10	4,008	4,967	5,04
	15	6,111	7,725	7,979
	20	8,155	10,399	10,678
	Média	19,889	25,401	25,851
50	5	3,979	4,973	3,961
	10	8,973	11,692	9,192
	15	14,348	19,405	15,002
	20	19,334	26,123	21,764
	Média	46,634	62,193	49,919
60	5	7,754	10,104	6,701
	10	18,221	24,221	15,413
	15	28,996	39,146	25,944
	20	39,215	52,853	34,816
	Média	94,186	126,324	82,874
70	5	14,113	19,555	10,574
	10	32,765	43,91	24,223
	15	51,579	71,205	38,765
	20	70,232	91,106	53,059
	Média	168,689	225,776	126,621

**Tabela 4 - Tempo Médio de Computação para problemas de grande porte para as heurísticas simples**

Tarefas (n)	Máquinas (m)	FL	H	LC
80	5	24,433	32,601	15,629
	10	55,082	73,904	35,826
	15	86,311	120,242	57,977
	20	118,414	165,726	79,565
	Média	284,24	392,473	188,997
90	5	37,071	49,733	21,434
	10	82,769	113,568	48,714
	15	132,473	184,63	78,571
	20	181,101	255,814	109,102
	Média	433,414	603,745	257,821
100	5	53,158	75,32	30,057
	10	120,875	174,044	67,055
	15	192,164	279,73	108,041
	20	262,99	387,252	149,275
	Média	629,187	916,346	354,428
110	5	76,777	109,896	38,93
	10	172,822	248,872	85,551
	15	276,669	404,481	141,762
	20	381,029	560,667	200,215
	Média	907,297	1323,916	466,458
120	5	106,243	153,215	52,419
	10	240,113	348,219	114,596
	15	393,491	564,286	183,9
	20	541,961	783,727	251,174
	Média	1281,808	1849,447	602,089
130	5	147,354	206,535	61,778
	10	331,832	471,53	136,023
	15	534,529	769,3	222,371
	20	745,832	1075,375	311,196
	Média	1759,547	2522,74	731,368

A figura 3 vem a ilustrar os resultados agrupados das tabelas 1 e 2 para Porcentagem de Sucesso.

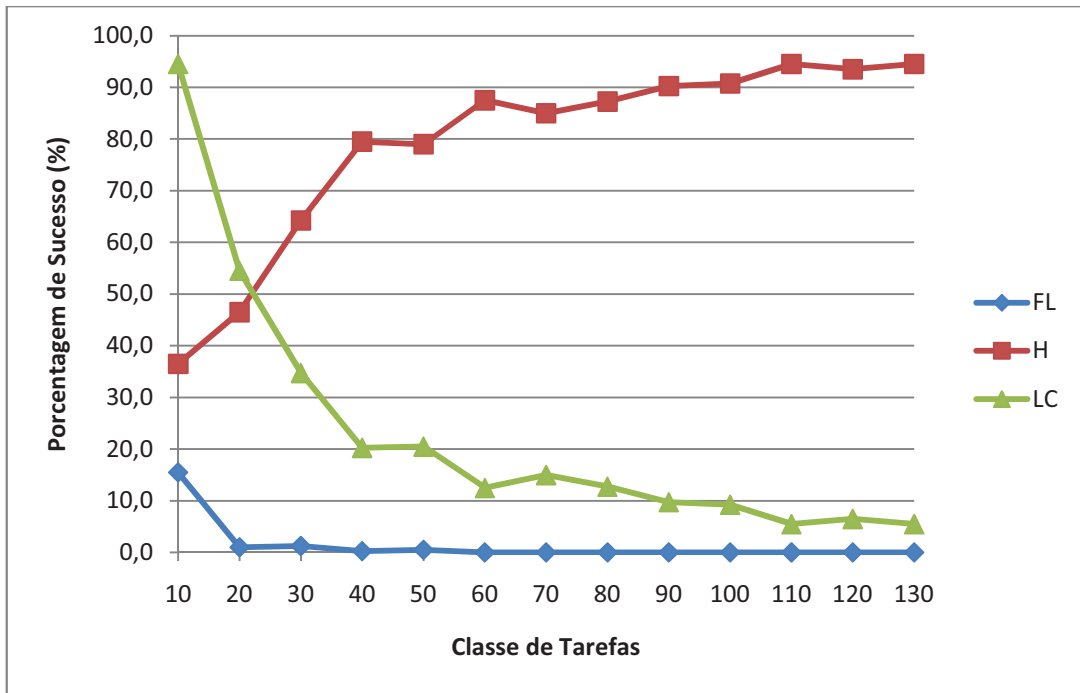


Figura 3 - Porcentagem de Sucesso para as heurísticas simples

A figura 4 vem a ilustrar os resultados agrupados das tabelas 1 e 2 para Desvio Relativo Médio.

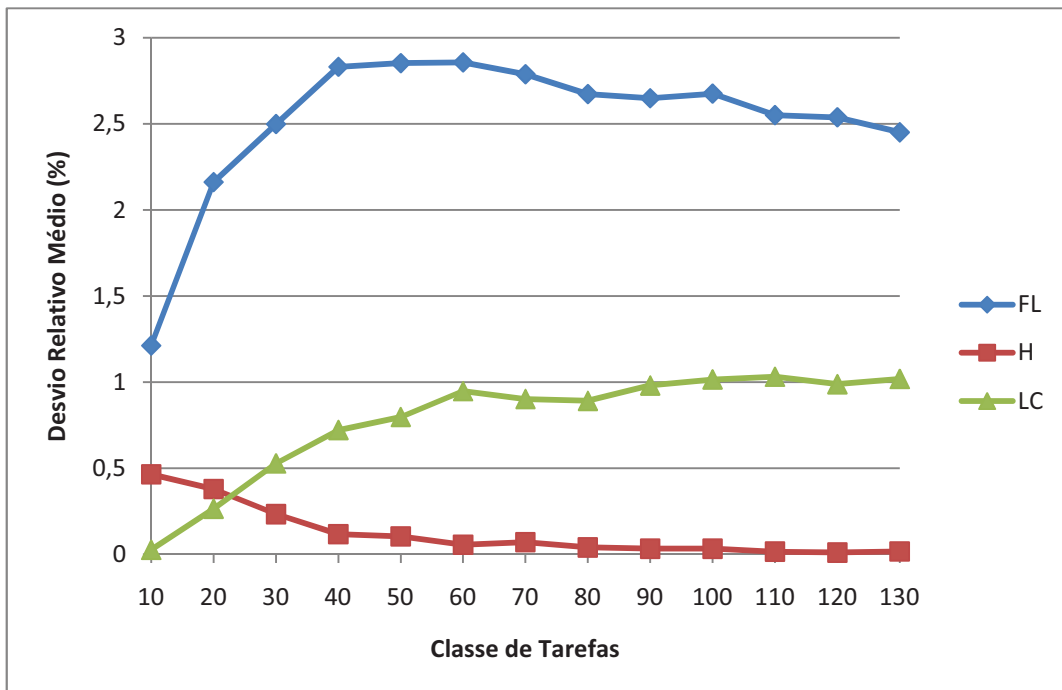


Figura 4 - Desvio Relativo Médio para as heurísticas simples



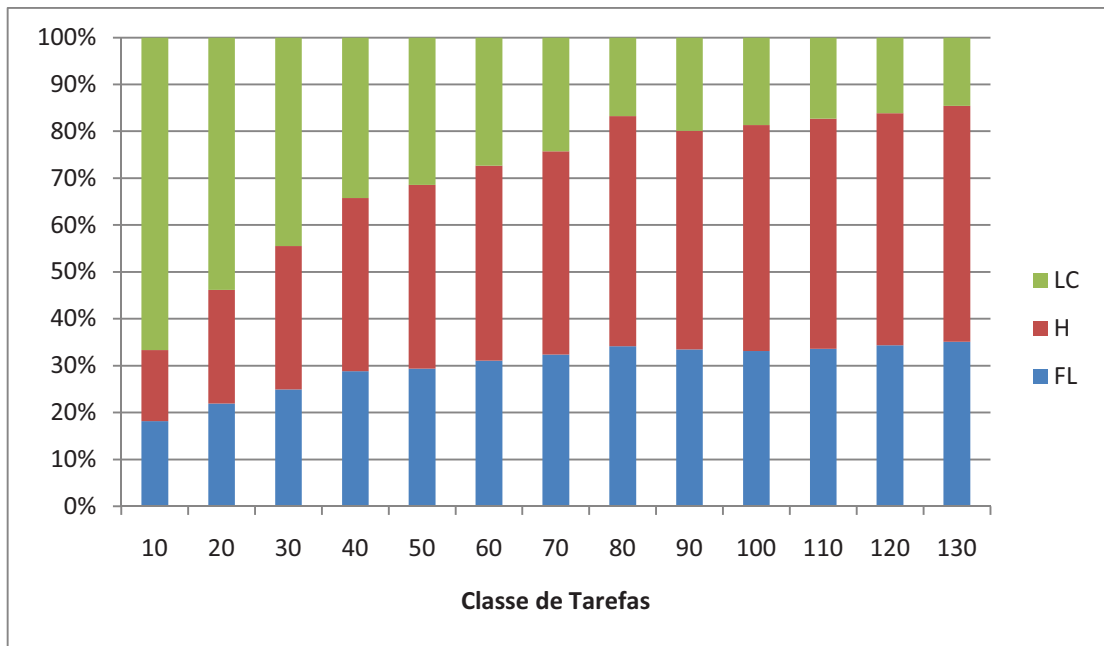


Figura 5 - Tempo de Processamento Relativo para as heurísticas simples

Analisando primeiramente a Porcentagem de Sucesso para as heurísticas simples, a Figura 3 mostra que para problemas de pequeno porte, 10 e 20 tarefas, a heurística LC apresenta desempenho superior, todavia, à medida que o número de tarefas aumenta para um dado problema o método H começa a se sobrepôr ao método LC com uma tendência de crescimento da Porcentagem de Sucesso com o aumento do porte do problema. O método FL se mostra inferior aos outros dois métodos para em qualquer um dos portes do problema.

A Figura 4 vem a destacar a superioridade do método H diante das demais heurísticas simples. Novamente, para problemas de pequeno porte, problemas com 10 e 20 tarefas, o método LC possui um Desvio Relativo menor que o método H, mas à medida que o número de tarefas cresce o método H vem a apresentar soluções cada vez mais próximas a solução ótima, ou seja, seu Desvio Relativo é cada vez mais próximo de zero. Observando a Figura 5 conclui-se que a superioridade do método H para problemas de grande porte vem acompanhada de um alto esforço computacional comparado com os demais métodos, principalmente com o método LC que tem sua taxa reduzida com o aumento do número de tarefas.

### Heurísticas Compostas

As heurísticas compostas implementadas e estudadas foram a C2, ICH3, NM e FL-IH7. Os resultados principais da experimentação computacional são ilustrados nas Tabela 5 à 8 e Figura 6 à 8. As Tabela 5 e 6 apresentam os resultados dos métodos avaliados para as estatísticas propostas para essa análise: Porcentagens de Sucesso e o Desvio Relativo Médio.

**Tabela 5 - Porcentagem de Sucesso e Desvio Relativo Médio para problemas de pequeno e médio para as heurísticas compostas**

Tarefas (n)	Máquinas (m)	C2		ICH3		NM		IH7-M	
		PS	DMR	PS	DMR	PS	DMR	PS	DMR
10	5	54,00	0,39	70,00	0,21	59,00	0,48	50,00	0,42
	10	68,00	0,20	65,00	0,18	56,00	0,34	59,00	0,32
	15	56,00	0,25	62,00	0,18	51,00	0,27	60,00	0,23
	20	59,00	0,19	60,00	0,20	63,00	0,28	57,00	0,23
	Média	59,25	0,26	64,25	0,19	57,25	0,34	56,50	0,30
20	5	22,00	0,68	35,00	0,41	28,00	0,62	23,00	0,64
	10	31,00	0,62	33,00	0,50	26,00	0,61	13,00	0,80
	15	19,00	0,68	35,00	0,48	29,00	0,67	18,00	0,73
	20	22,00	0,53	30,00	0,45	29,00	0,51	22,00	0,66
	Média	23,50	0,63	33,25	0,46	28,00	0,61	19,00	0,71
30	5	17,00	0,76	39,00	0,36	31,00	0,68	13,00	0,89
	10	19,00	0,84	34,00	0,56	30,00	0,68	17,00	0,90
	15	17,00	0,81	43,00	0,44	25,00	0,64	15,00	0,91
	20	24,00	0,63	26,00	0,48	29,00	0,51	22,00	0,67
	Média	19,25	0,76	35,50	0,46	28,75	0,63	16,75	0,84
40	5	17,00	0,60	53,00	0,21	21,00	0,69	10,00	0,85
	10	23,00	0,79	50,00	0,27	17,00	0,81	10,00	1,17
	15	21,00	0,67	35,00	0,45	28,00	0,58	16,00	0,90
	20	12,00	0,82	34,00	0,49	42,00	0,39	12,00	0,86
	Média	18,25	0,72	43,00	0,36	27,00	0,62	12,00	0,95
50	5	15,00	0,64	59,00	0,21	15,00	0,66	11,00	0,79
	10	21,00	0,77	47,00	0,32	20,00	0,69	12,00	0,96
	15	18,00	0,76	51,00	0,30	25,00	0,76	6,00	0,94
	20	11,00	0,84	40,00	0,41	30,00	0,60	19,00	0,89
	Média	16,25	0,76	49,25	0,31	22,50	0,68	12,00	0,90
60	5	13,00	0,62	63,00	0,12	17,00	0,63	7,00	0,76
	10	13,00	0,82	59,00	0,23	17,00	0,76	11,00	0,97
	15	7,00	0,86	51,00	0,33	32,00	0,47	10,00	0,98
	20	18,00	0,68	48,00	0,32	28,00	0,59	7,00	0,88
	Média	12,75	0,74	55,25	0,25	23,50	0,61	8,75	0,90
70	5	11,00	0,58	65,00	0,09	12,00	0,56	12,00	0,66
	10	11,00	0,90	59,00	0,20	21,00	0,73	9,00	1,04
	15	10,00	0,81	56,00	0,23	29,00	0,61	5,00	0,95
	20	18,00	0,72	46,00	0,33	25,00	0,46	11,00	0,94
	Média	12,50	0,75	56,50	0,21	21,75	0,59	9,25	0,90

**Tabela 6 - Porcentagem de sucesso e Desvio Relativo Médio para problemas de grande porte para as heurísticas compostas**

Tarefas (n)	Máquinas (m)	C2		ICH3		NM		IH7-M	
		PS	DMR	PS	DMR	PS	DMR	PS	DMR
80	5	10,00	0,54	69,00	0,09	15,00	0,53	6,00	0,70
	10	10,00	0,74	53,00	0,21	30,00	0,61	7,00	0,96
	15	7,00	0,81	59,00	0,21	29,00	0,68	6,00	0,98
	20	8,00	0,74	60,00	0,19	26,00	0,49	6,00	0,90
	Média	8,75	0,71	60,25	0,17	25,00	0,58	6,25	0,88
90	5	16,00	0,51	64,00	0,09	13,00	0,57	7,00	0,66
	10	10,00	0,68	64,00	0,20	23,00	0,64	3,00	1,03
	15	11,00	0,72	63,00	0,17	23,00	0,60	3,00	1,03
	20	20,00	0,60	51,00	0,22	22,00	0,50	7,00	0,89
	Média	14,25	0,63	60,50	0,17	20,25	0,58	5,00	0,90
100	5	8,00	0,51	68,00	0,08	17,00	0,51	7,00	0,60
	10	13,00	0,66	64,00	0,11	16,00	0,68	7,00	0,98
	15	14,00	0,69	55,00	0,22	20,00	0,50	11,00	0,97
	20	9,00	0,70	58,00	0,19	25,00	0,54	8,00	0,83
	Média	11,00	0,64	61,25	0,15	19,50	0,56	8,25	0,84
110	5	15,00	0,36	66,00	0,07	14,00	0,41	5,00	0,59
	10	13,00	0,65	60,00	0,16	23,00	0,66	4,00	0,97
	15	12,00	0,73	57,00	0,18	25,00	0,53	6,00	0,96
	20	9,00	0,70	63,00	0,20	23,00	0,56	5,00	0,98
	Média	12,25	0,61	61,50	0,15	21,25	0,54	5,00	0,88
120	5	12,00	0,42	61,00	0,09	17,00	0,46	10,00	0,55
	10	7,00	0,78	69,00	0,11	15,00	0,66	9,00	0,91
	15	7,00	0,67	70,00	0,15	20,00	0,60	3,00	0,94
	20	14,00	0,64	58,00	0,20	23,00	0,42	5,00	0,83
	Média	10,00	0,63	64,50	0,14	18,75	0,53	6,75	0,81
130	5	14,00	0,40	58,00	0,07	21,00	0,42	7,00	0,53
	10	7,00	0,67	71,00	0,10	15,00	0,49	7,00	0,80
	15	10,00	0,72	66,00	0,12	19,00	0,64	5,00	0,90
	20	13,00	0,60	62,00	0,15	23,00	0,47	2,00	0,91
	Média	11,00	0,60	64,25	0,11	19,50	0,51	5,25	0,79

As Tabela 7 e Tabela 8 apresentam os Tempos de Médios de Computação para os métodos compostos.

**Tabela 7 - Tempo Médio de Computação em segundos para problemas de pequeno e médio porte para as heurísticas compostas**

Tarefas (n)	Máquinas (m)	C2	ICH3	NM	FL-IH7
10	5	0,021	0,02	0,025	0,019
	10	0,044	0,04	0,069	0,027
	15	0,064	0,071	0,074	0,037
	20	0,065	0,09	0,102	0,054
	Média	0,194	0,221	0,27	0,137
20	5	0,165	0,212	0,29	0,139
	10	0,377	0,399	0,524	0,315
	15	0,591	0,625	0,824	0,477
	20	0,831	0,816	1,118	0,669
	Média	1,964	2,052	2,756	1,6
30	5	0,749	0,854	1,144	0,642
	10	1,755	1,728	2,517	1,393
	15	2,767	2,897	4,106	2,087
	20	3,54	3,58	5,456	2,788
	Média	8,811	9,059	13,223	6,91
40	5	2,059	2,359	3,413	1,729
	10	5,135	4,993	7,824	3,794
	15	8,202	8,44	12,494	6,126
	20	10,94	11,003	16,741	7,97
	Média	26,336	26,795	40,472	19,619
50	5	4,683	5,853	8,034	3,88
	10	11,938	11,811	18,465	8,767
	15	20,112	17,866	29,806	13,817
	20	26,642	24,406	40,128	18,734
	Média	63,375	59,936	96,433	45,198
60	5	9,532	10,603	16,077	7,948
	10	24,091	23,619	37,696	17,388
	15	38,945	37,024	60,458	28,789
	20	52,065	47,689	81,603	37,801
	Média	124,633	118,935	195,834	91,926
70	5	17,651	19,435	29,57	14,212
	10	42,29	41,611	68,584	32,99
	15	69,134	60,432	112,89	48,126
	20	95,435	79,442	150,202	65,64
	Média	224,51	200,92	361,246	160,968

**Tabela 8 - Tempo Médio de Processamento em segundos para problemas de grande porte para as heurísticas compostas**

Tarefas (n)	Máquinas (m)	C2	ICH3	NM	FL-IH7
80	5	28,531	33,042	50,078	25,26
	10	71,677	66,814	115,054	52,18
	15	115,797	104,06	186,119	81,155
	20	157,057	140,072	258,316	109,183
	Média	373,062	343,988	609,567	267,778
90	5	151,167	52,859	77,824	38,689
	10	321,925	101,463	177,535	80,698
	15	501,293	168,319	291,025	127,308
	20	680,367	215,284	413,15	168,881
	Média	1654,752	537,925	959,534	415,576
100	5	229,737	77,174	120,051	58,01
	10	492,251	167,369	280,942	122,178
	15	765,812	249,228	456,155	190,258
	20	1049,002	320,876	626,017	254,12
	Média	2536,802	814,647	1483,165	624,566
110	5	337,714	113,056	174,228	85,596
	10	726,708	248,706	388,809	173,512
	15	1126,306	345,664	630,714	272,667
	20	1529,896	457,572	882,617	376,642
	Média	3720,624	1164,998	2076,368	908,417
120	5	465,919	157,871	238,589	117,4
	10	1006,66	339,996	543,016	244,734
	15	1584,917	490,722	885,593	376,368
	20	2064,144	609,323	1228,056	525,562
	Média	5121,64	1597,912	2895,254	1264,064
130	5	628,846	198,948	320,323	156,718
	10	1351,164	437,284	727,038	323,553
	15	2077,728	658,235	1192,792	517,723
	20	2838,097	895,23	1669,949	704,417
	Média	6895,835	2189,697	3910,102	1702,411

Todos os resultados estão sumarizados de maneira mais sucinta nas Figura 6 à 8.

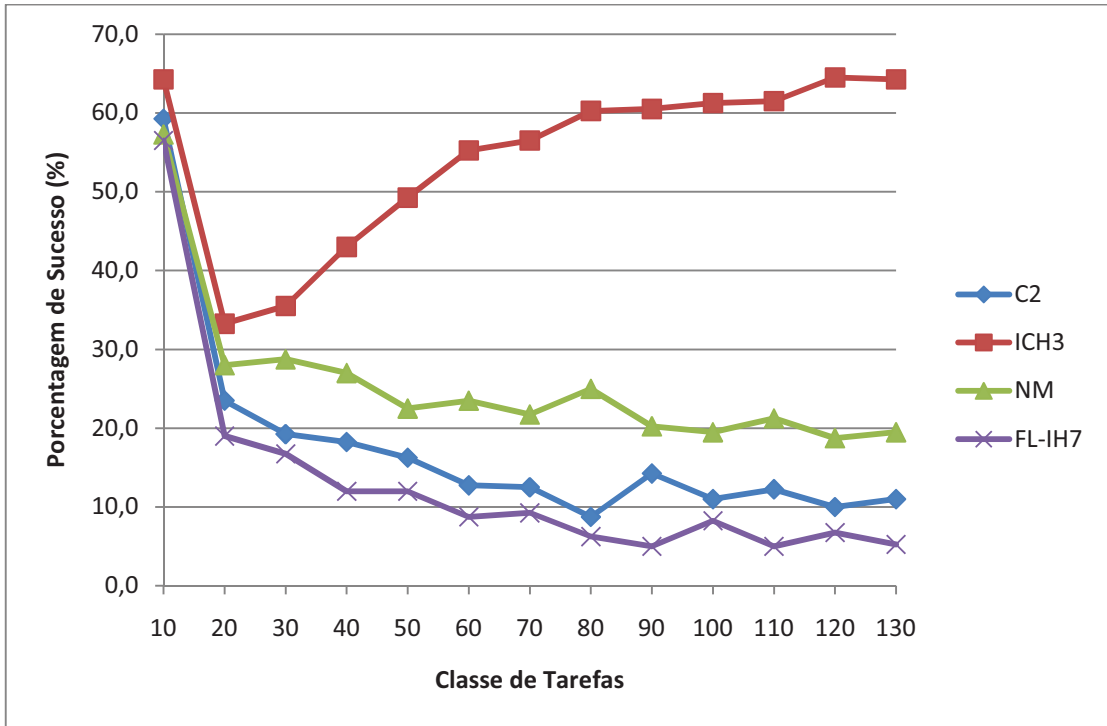


Figura 6 - Porcentagem de Sucesso para as heurísticas compostas

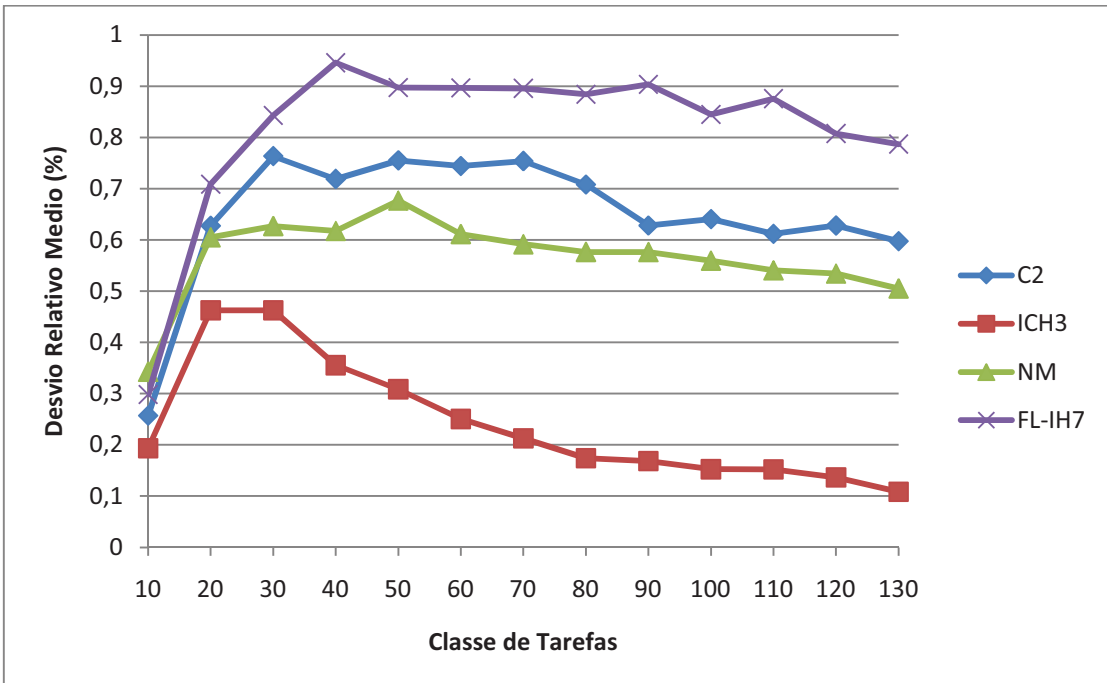
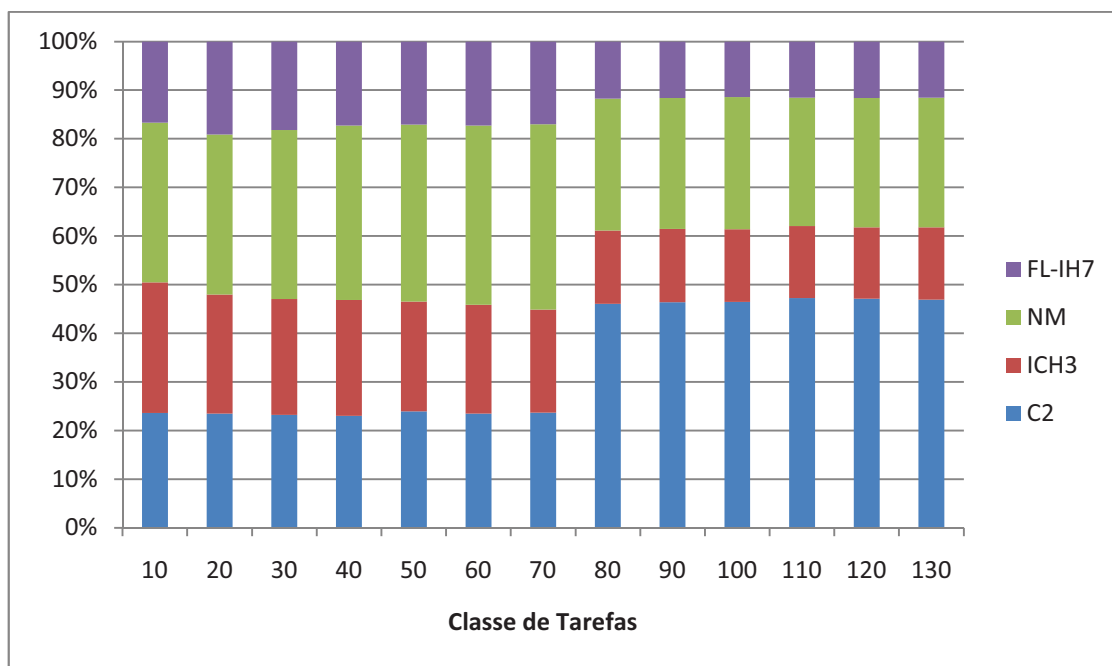


Figura 7 - Desvio Relativo Médio para as heurísticas compostas



**Figura 8 - Tempo de Processamento Relativo para as heurísticas compostas**

A Figura 6 e a Figura 7 mostram de maneira clara a superioridade da heurística ICH3 diante das demais. Enquanto que a Porcentagem de Sucesso dos métodos NM, C2 e FL-IH7 apresentam um comportamento parecido com a variação do número de tarefas do problema, a heurística ICH3 se destoa, aumentando sua porcentagem com o aumento do porte do problema. Pela Figura 7 a diferença de desempenho entre os métodos se torna mais evidente, facilitando a classificação. Dessa forma, o NM gera soluções de melhor qualidade comparado com o C2 e com o FL-IH7, e o método FL-IH7 é o que apresenta o pior desempenho dentre todos os comparados na classe de heurísticas compostas.

A análise da Figura 8 junto com as estatísticas vistas acima ressalta ainda mais a capacidade do método ICH3 em obter soluções de alta qualidade com um esforço computacional menor que as demais heurísticas.

### **Todas as Heurísticas**

Visando comparar o desempenho de cada heurística sem que haja distinção de categorias, nesta seção serão levantados os valores da Porcentagem de Sucesso e do Desvio Relativos Médio para todos os métodos em conjunto. As Tabela 9 à Tabela 12 e as Figura 9 e Figura 10 apresentam esses dados.

Tabela 9- Porcentagem de Sucesso para problemas de pequeno e médio porte

Número de tarefas (n)	Número de máquinas (m)	FL	H	LC	C2	ICH3	NM	FL-IH7
10	5	17,00	38,00	77,00	48,00	62,00	52,00	46,00
	10	13,00	26,00	76,00	54,00	55,00	43,00	46,00
	15	11,00	30,00	77,00	49,00	57,00	47,00	53,00
	20	15,00	30,00	75,00	54,00	53,00	55,00	53,00
	Média	14,00	31,00	76,25	51,25	56,75	49,25	49,50
20	5	1,00	12,00	21,00	18,00	27,00	17,00	15,00
	10	0,00	18,00	15,00	22,00	26,00	20,00	8,00
	15	0,00	10,00	9,00	15,00	34,00	24,00	13,00
	20	0,00	13,00	14,00	16,00	27,00	21,00	15,00
	Média	0,25	13,25	14,75	17,75	28,50	20,50	12,75
30	5	0,00	9,00	7,00	15,00	34,00	25,00	10,00
	10	0,00	15,00	7,00	17,00	28,00	20,00	13,00
	15	0,00	17,00	6,00	12,00	36,00	18,00	11,00
	20	0,00	20,00	4,00	19,00	23,00	20,00	15,00
	Média	0,00	15,25	6,00	15,75	30,25	20,75	12,25
40	5	0,00	12,00	2,00	16,00	44,00	19,00	8,00
	10	0,00	19,00	1,00	20,00	38,00	15,00	7,00
	15	0,00	14,00	3,00	17,00	28,00	26,00	12,00
	20	0,00	22,00	3,00	9,00	27,00	31,00	9,00
	Média	0,00	16,75	2,25	15,50	34,25	22,75	9,00
50	5	0,00	5,00	0,00	14,00	56,00	14,00	11,00
	10	0,00	18,00	2,00	16,00	40,00	14,00	10,00
	15	0,00	20,00	1,00	14,00	43,00	17,00	5,00
	20	0,00	26,00	3,00	8,00	30,00	18,00	15,00
	Média	0,00	17,25	1,50	13,00	42,25	15,75	10,25
60	5	0,00	11,00	1,00	10,00	58,00	14,00	6,00
	10	0,00	16,00	2,00	11,00	49,00	16,00	7,00
	15	0,00	24,00	1,00	5,00	41,00	23,00	7,00
	20	0,00	36,00	1,00	11,00	36,00	14,00	2,00
	Média	0,00	21,75	1,25	9,25	46,00	16,75	5,50
70	5	0,00	9,00	0,00	11,00	60,00	12,00	8,00
	10	0,00	9,00	0,00	10,00	55,00	18,00	8,00
	15	0,00	20,00	0,00	8,00	45,00	25,00	2,00
	20	0,00	34,00	2,00	9,00	32,00	18,00	6,00
	Média	0,00	18,00	0,50	9,50	48,00	18,25	6,00



Tabela 10 - Porcentagem de Sucesso para problemas de grande porte

Número de tarefas (n)	Número de máquinas (m)	FL	H	LC	C2	ICH3	NM	FL-IH7
80	5	0,00	4,00	0,00	10,00	66,00	14,00	6,00
	10	0,00	11,00	0,00	10,00	46,00	26,00	7,00
	15	0,00	17,00	0,00	7,00	50,00	22,00	4,00
	20	0,00	24,00	1,00	5,00	46,00	22,00	2,00
	Média	0,00	14,00	0,25	8,00	52,00	21,00	4,75
90	5	0,00	3,00	0,00	16,00	62,00	12,00	7,00
	10	0,00	9,00	0,00	9,00	59,00	21,00	2,00
	15	0,00	18,00	0,00	6,00	56,00	18,00	2,00
	20	0,00	24,00	0,00	16,00	40,00	16,00	4,00
	Média	0,00	13,50	0,00	11,75	54,25	16,75	3,75
100	5	0,00	4,00	0,00	8,00	64,00	17,00	7,00
	10	0,00	10,00	0,00	8,00	63,00	13,00	6,00
	15	0,00	23,00	0,00	11,00	43,00	16,00	7,00
	20	0,00	27,00	0,00	5,00	45,00	19,00	4,00
	Média	0,00	16,00	0,00	8,00	53,75	16,25	6,00
110	5	0,00	1,00	0,00	14,00	66,00	14,00	5,00
	10	0,00	9,00	0,00	12,00	55,00	20,00	4,00
	15	0,00	19,00	0,00	9,00	45,00	22,00	5,00
	20	0,00	26,00	0,00	5,00	50,00	15,00	4,00
	Média	0,00	13,75	0,00	10,00	54,00	17,75	4,50
120	5	0,00	5,00	0,00	11,00	59,00	16,00	9,00
	10	0,00	14,00	0,00	7,00	61,00	11,00	7,00
	15	0,00	17,00	0,00	7,00	59,00	16,00	1,00
	20	0,00	22,00	0,00	10,00	48,00	17,00	3,00
	Média	0,00	14,50	0,00	8,75	56,75	15,00	5,00
130	5	0,00	6,00	0,00	13,00	55,00	20,00	6,00
	10	0,00	11,00	0,00	7,00	65,00	13,00	4,00
	15	0,00	12,00	0,00	7,00	59,00	17,00	5,00
	20	0,00	19,00	0,00	9,00	54,00	17,00	1,00
	Média	0,00	12,00	0,00	9,00	58,25	16,75	4,00

A Figura 9 apresenta os resultados sumarizados.

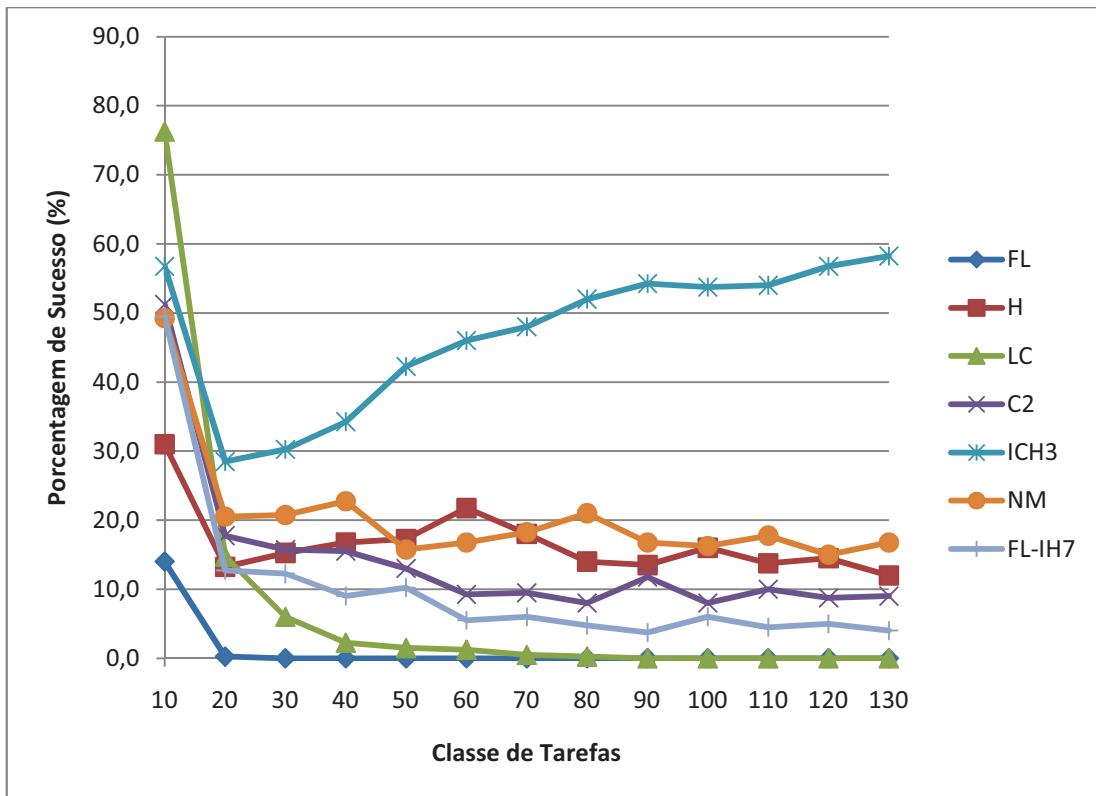


Figura 9 - Porcentagem de Sucesso para todos os métodos avaliados

As Porcentagens de Sucesso apresentada na Figura 9 vem a destacar a capacidade dos métodos compostos em obter soluções melhores que as fornecidas pelos métodos simples. Apesar de a superioridade ser evidente, a heurística simples H ainda obteve resultados equivalentes ao da heurística NM tanto para pequeno como para grande porte, sendo inferior apenas a heurística ICH3 que apresenta uma tendência de crescimento da Porcentagem de Sucesso com o aumento do porte do problema, o que a consolida como a melhor heurística dentre todas as estudadas.

As Tabela 11 e Tabela 12 apresentam os Desvios Relativos Médios dos métodos.

Tabela 11 - Porcentagem de Desvio Relativo Médio para problemas de pequeno e médio porte

Número de tarefas (n)	Número de máquinas (m)	FL	H	LC	C2	ICH3	NM	FL-IH7
10	5	1,8710	0,7539	0,2110	0,4558	0,2767	0,5436	0,4818
	10	1,4107	0,6197	0,1398	0,2538	0,2368	0,3962	0,3727
	15	1,0475	0,5128	0,1278	0,2797	0,2166	0,3013	0,2591
	20	1,0383	0,4863	0,1362	0,2207	0,2253	0,3104	0,2616
	Média	1,3419	0,5932	0,1537	0,3025	0,2389	0,3879	0,3438
20	5	2,7542	0,9336	0,7154	0,7558	0,4892	0,7017	0,7195
	10	2,8384	0,7655	0,7029	0,7141	0,5959	0,7040	0,8940
	15	2,5868	0,7998	0,7535	0,7424	0,5498	0,7378	0,7965
	20	2,1845	0,7077	0,5756	0,5889	0,5071	0,5691	0,7175
	Média	2,5910	0,8017	0,6868	0,7003	0,5355	0,6781	0,7819
30	5	2,9974	1,1599	1,2586	0,8193	0,4208	0,7354	0,9482
	10	3,4657	0,8293	1,2320	0,9268	0,6463	0,7671	0,9788
	15	3,2318	0,7534	1,1747	0,8718	0,4988	0,6945	0,9681
	20	2,8103	0,6584	0,9174	0,7262	0,5751	0,6016	0,7667
	Média	3,1263	0,8502	1,1457	0,8360	0,5352	0,6997	0,9154
40	5	3,1848	0,8531	1,3238	0,6259	0,2368	0,7193	0,8735
	10	3,9701	0,8834	1,5823	0,8770	0,3619	0,8979	1,2585
	15	3,4160	0,7173	1,3107	0,7358	0,5161	0,6426	0,9671
	20	3,3532	0,5574	1,2238	0,8879	0,5586	0,4615	0,9362
	Média	3,4810	0,7528	1,3602	0,7816	0,4183	0,6803	1,0088
50	5	3,0338	0,9044	1,3670	0,6563	0,2221	0,6725	0,8054
	10	3,6276	0,7737	1,5655	0,8456	0,3919	0,7569	1,0348
	15	3,8688	0,7414	1,5140	0,8841	0,4224	0,8870	1,0635
	20	3,5888	0,6471	1,4064	0,9510	0,5135	0,7066	1,0021
	Média	3,5298	0,7666	1,4632	0,8342	0,3875	0,7557	0,9765
60	5	2,9699	0,8201	1,5630	0,6394	0,1476	0,6558	0,7807
	10	3,7615	0,8313	1,6881	0,8660	0,2806	0,8078	1,0133
	15	3,6297	0,5683	1,5835	0,9865	0,4507	0,5900	1,1054
	20	3,5743	0,4581	1,4268	0,7900	0,4282	0,6972	0,9930
	Média	3,4838	0,6694	1,5653	0,8205	0,3268	0,6877	0,9731
70	5	2,8386	0,8323	1,3853	0,5916	0,1065	0,5771	0,6761
	10	3,6740	0,9125	1,7805	0,9357	0,2385	0,7695	1,0725
	15	3,6974	0,7659	1,6111	0,8695	0,2841	0,6668	1,0044
	20	3,6695	0,4420	1,5150	0,8328	0,4359	0,5678	1,0450
	Média	3,4699	0,7381	1,5730	0,8074	0,2663	0,6453	0,9495

Tabela 12 - Porcentagem de Desvio Relativo Médio para problemas de grande porte

Número de tarefas (n)	Número de máquinas (m)	FL	H	LC	C2	ICH3	NM	FL-IH7
80	5	2,6711	0,8361	1,4464	0,5467	0,0914	0,5354	0,7086
	10	3,7269	0,8592	1,7405	0,7832	0,2541	0,6536	0,9999
	15	3,7039	0,7229	1,6946	0,8684	0,2624	0,7350	1,0319
	20	3,4700	0,5624	1,5208	0,8142	0,2693	0,5625	0,9778
	Média	3,3930	0,7451	1,6006	0,7531	0,2193	0,6216	0,9295
90	5	2,6411	0,8386	1,4152	0,5154	0,0932	0,5689	0,6609
	10	3,4331	0,7939	1,7898	0,7104	0,2265	0,6680	1,0608
	15	3,7015	0,7186	1,7929	0,7733	0,2237	0,6604	1,0907
	20	3,6698	0,5692	1,7318	0,6812	0,2995	0,5765	0,9703
	Média	3,3614	0,7301	1,6824	0,6701	0,2107	0,6184	0,9457
100	5	2,4693	0,7862	1,3744	0,5241	0,0927	0,5205	0,6156
	10	3,5846	0,7762	1,7373	0,6916	0,1385	0,7118	1,0089
	15	3,6411	0,6226	1,7579	0,7526	0,2898	0,5697	1,0326
	20	3,6629	0,5459	1,8098	0,7862	0,2812	0,6285	0,9148
	Média	3,3395	0,6827	1,6699	0,6886	0,2006	0,6076	0,8930
110	5	2,3037	0,7487	1,3054	0,3647	0,0684	0,4150	0,5874
	10	3,3921	0,7573	1,7746	0,6809	0,1868	0,6850	1,0005
	15	3,5568	0,6708	1,8670	0,7966	0,2440	0,5954	1,0288
	20	3,6021	0,4854	1,7968	0,8108	0,3145	0,6727	1,0929
	Média	3,2137	0,6656	1,6859	0,6632	0,2034	0,5920	0,9274
120	5	2,3640	0,6759	1,3126	0,4264	0,0978	0,4682	0,5604
	10	3,2267	0,6804	1,7489	0,8128	0,1369	0,6850	0,9382
	15	3,5160	0,6492	1,7559	0,7326	0,2114	0,6657	1,0038
	20	3,5402	0,4815	1,5960	0,7126	0,2730	0,4903	0,9010
	Média	3,1617	0,6217	1,6034	0,6711	0,1798	0,5773	0,8509
130	5	2,2057	0,6324	1,2386	0,4093	0,0722	0,4231	0,5378
	10	3,1154	0,7145	1,7673	0,6855	0,1147	0,5089	0,8213
	15	3,4258	0,6231	1,7498	0,7414	0,1445	0,6663	0,9206
	20	3,5482	0,5395	1,7807	0,6516	0,2013	0,5206	0,9673
	Média	3,0738	0,6274	1,6341	0,6219	0,1332	0,5297	0,8118

A Figura 10 apresenta o desvio relativo médio para todos os métodos avaliados.

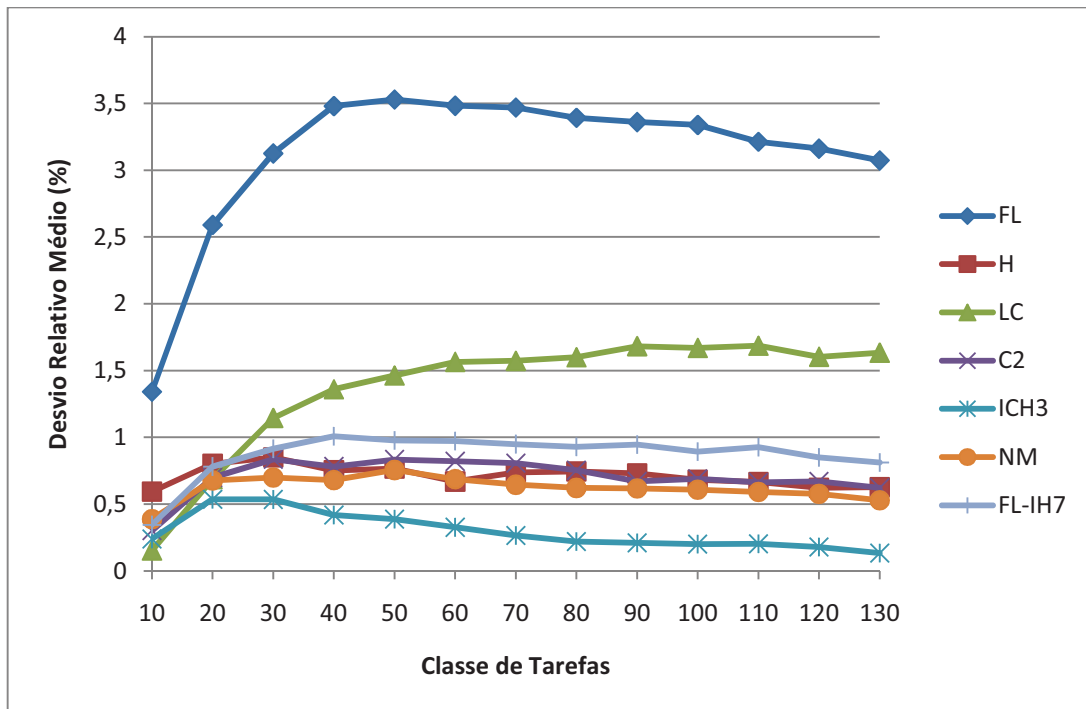


Figura 10 – Desvio Relativo Médio para todos os métodos avaliados

Mais uma vez os métodos compostos se mostram mais eficientes em seu desempenho considerando as hipóteses analisadas. A heurística simples H é a única heurística simples que apresenta resultados equiparáveis com as heurísticas compostas, sendo seus resultados próximos aos obtidos pelas heurísticas NM e C2. A heurística ICH3 tem os menores desvios, se aproximando cada vez mais de zero com o aumento do porte do problema.

Comparando os tempo de computação apresentados nas, Tabela 4, Tabela 7 e Tabela 8 foi possível construir as Figura 11 e Figura 12, apresentando esses valores em termos absolutos (em minutos) e relativos.

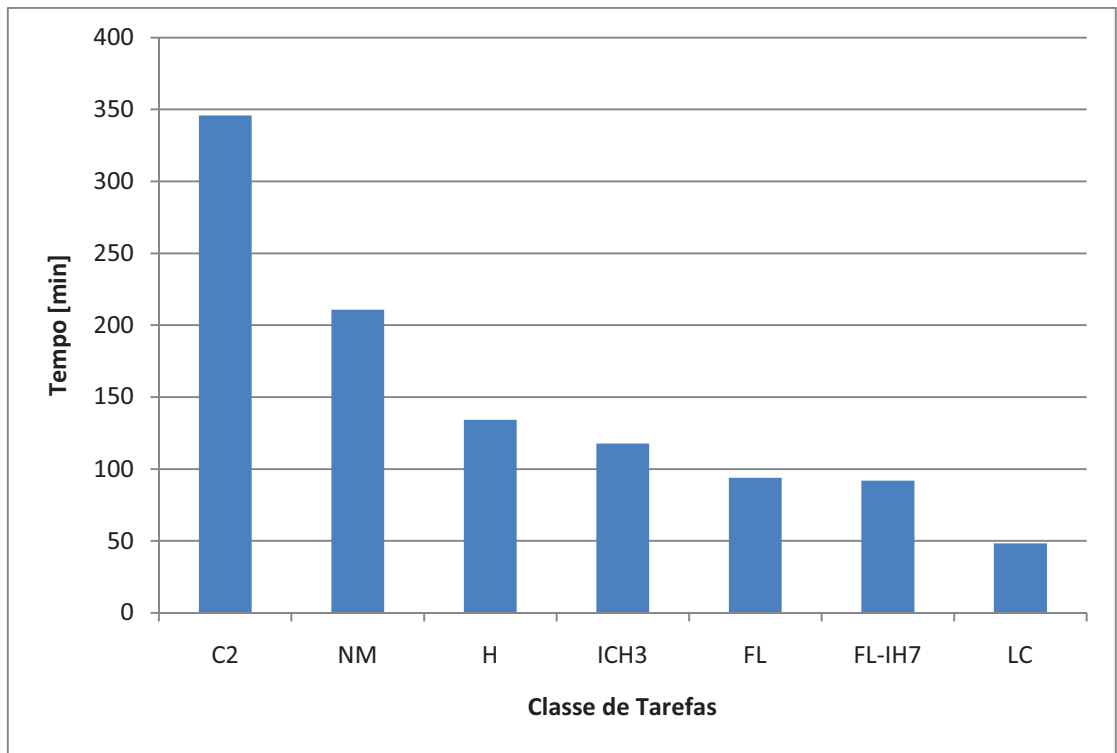


Figura 11 - Tempo de Computação absoluto para todas as heurísticas

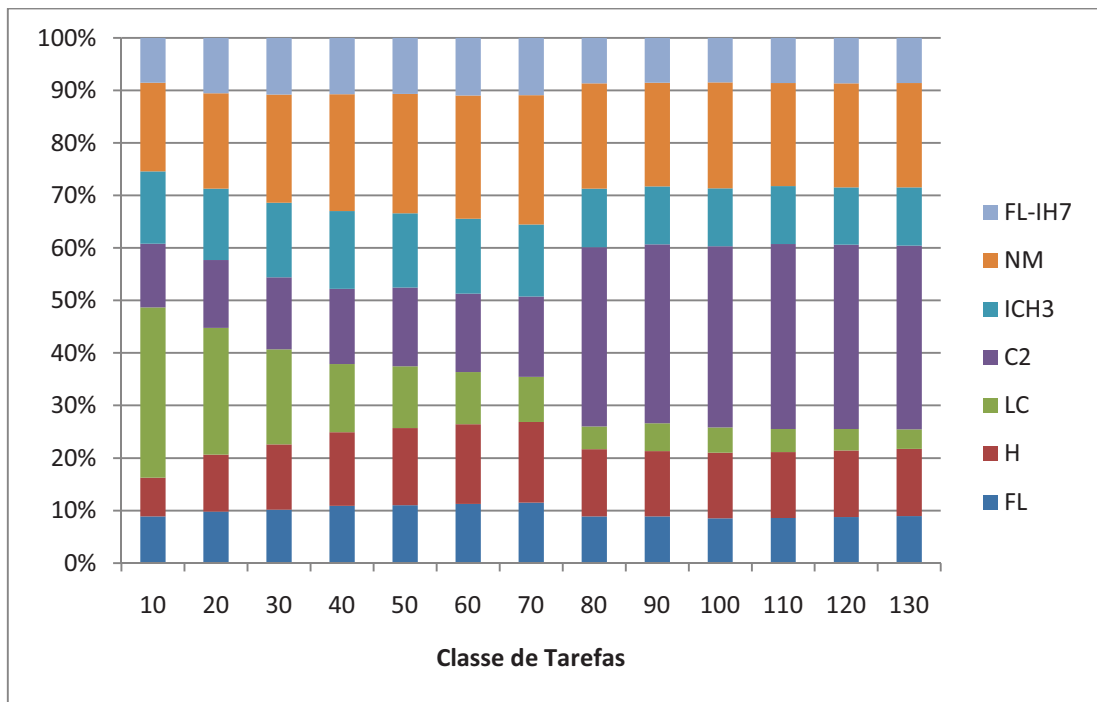


Figura 12 - Tempo de Computação relativo para todas as heurísticas

A Figura 11 e Figura 12 mostram uma diferença significativa entre os tempos de computação das heurísticas. Os métodos compostos, como por exemplo, o C2 e o NM, em

geral são os que apresentaram os maiores tempos de processamento, apesar disso, o método composto ICH3 se coloca num tempo inferior a heurística simples H, e como já visto, os resultados gerados por ICH3 são de qualidade superior aos das demais heurísticas.

## 6. Considerações Finais

---

A análise das heurísticas simples permite concluir que para problemas de pequeno porte, 10 e 20 tarefas, o método LC detém os melhores resultados de Porcentagem de Sucesso e Desvio Relativo Médio para as heurísticas simples, mas à medida que os problemas se tornam de médio e grande porte a heurística H se destaca e apresenta resultados bem superiores as demais, mas para isso é necessário um esforço computacional também superior. Já para as heurísticas compostas a análise dos resultados obtidos permite concluir que para problemas de pequeno, médio e grande porte, a heurística ICH3 foi a que apresentou os melhores resultados de Porcentagem de Sucesso e de Desvio Relativo Médio. Quanto ao tempo de processamento, apenas a heurística FL-IH7 obteve um esforço computacional menor que ICH3, o que ressalta ainda mais a qualidade deste último.

Comparando todas as heurísticas vê-se que para o conjunto de problemas de 10 e 20 tarefas o resultado médio obtido pela heurística simples LC ainda se mantém melhor, mas, em linhas gerais, o maior esforço computacional das heurísticas compostas reflete em soluções de melhor qualidade diante das heurísticas simples.

Uma análise comparativa é importante como referência literária, assim, como proposta de pesquisa para trabalhos futuros há a extensão da análise realizada neste trabalho para outros ambientes de controle de produção, como, por exemplo, a condição de *no-idle flowshop*.



## 7. APÊNDICE

O método LR, Liu e Reeves (2001), se baseia na caracterização de cada tarefa por uma função de indexação, *index function*, a qual é constituída por duas partes: peso das tarefas e o tempo de fluxo artificial. A primeira parte considera a importância da posição de cada tarefa no cálculo do tempo de fluxo, já a segunda determina o grau de influência nas tarefas posteriores àquela a qual se está analisando.

Antes de descrever os procedimentos de cada uma das partes é válido discretizar a seqüência em programada, S, (*Schedule sequence*) e não programada, U, (*Unschedule sequence*). O propósito do método é fazer com que S, inicialmente com nenhuma tarefa, seja completamente preenchida com as tarefas presentes na seqüência U. Assim, sendo  $k$  a posição na qual se pretende alocar uma tarefa de U, dependendo de  $k$  os pesos para cada tarefa serão descritas pela expressão. 9.

$$IT_{ik} = \sum_{j=2}^m w_{jk} \cdot \max\{C(i, j-1) - C([k], j), 0\} \quad (9)$$

Onde,

$$w_{jk} = \frac{m}{j + k \cdot \frac{(m-j)}{(n-2)}} \quad (10)$$

O termo IT determina o peso das tarefas em função da posição pretendida a ser preenchida em S. Esse critério privilegia as posições iniciais, pois sua influência no tempo de fluxo total é maior que para as demais posições.

A segunda parte do método LR considera o efeito de cada análise em S nas tarefas que permaneceram em U. Para isso cria-se uma tarefa artificial constituída por todas as tarefas presentes em U, exceto a tarefa a qual se pretende acoplar a S. Um esquema dessa análise é apresentado por Liu, et al. (2001) e está mostrado na Figura 13.

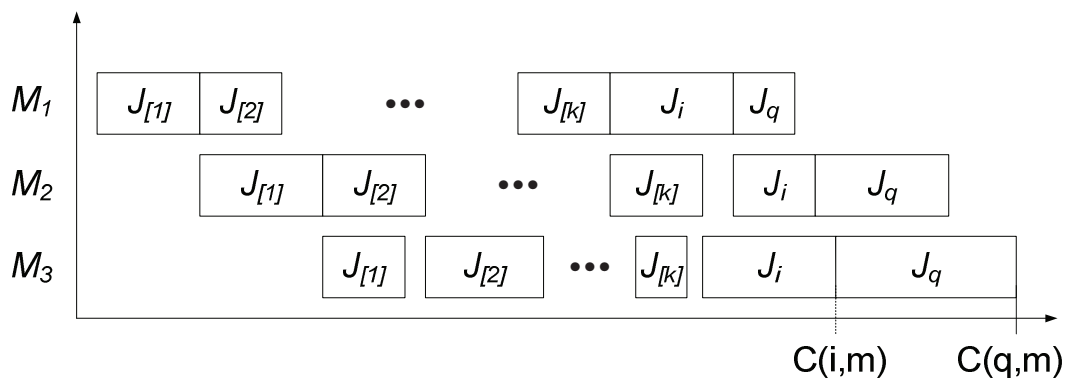


Figura 13 – Diagrama de Gantt para o método LR em condição de *flowshop* permutacional

A tarefa  $q$ , composta pela junção de todas as tarefas de U, terá o seu tempo de processamento definido pela expressão 11.

$$t(q, j) = \sum_{\substack{r \in U \\ r \neq i}} \frac{t(r, j)}{(n - k - 1)} \quad (11)$$

Pela expressão 11 é possível definir as expressões 12 à 15 que indicarão o tempo de fluxo de uma certa tarefa de forma recursiva.

$$C(i, 1) = C([k], 1) + t(i, 1) \quad (12)$$

$$C(i, j) = \max\{C([k], j), C(i, j - 1)\} + t(i, j), \quad j = 2, \dots, m \quad (13)$$

$$C(q, 1) = C(i, 1) + t(q, 1) \quad (14)$$

$$C(q, j) = \max\{C(i, j), C(q, j - 1)\} + t(q, j), \quad j = 2, \dots, m \quad (15)$$

Ao fim, calcula-se a ponderação dessa segunda parte do método LR pelo peso AT dado pela expressão 16.

$$AT_{ik} = C(i, m) + C(q, m) \quad (16)$$

Com os dois pesos definidos, a forma pela qual eles se combinam é expressa pela expressão 17.

$$f_{ik} = (n - k - 2) \cdot IT_{ik} + AT_{ik} \quad (17)$$

Estando todas as tarefas de U ponderadas para cada valor de  $k$ , é possível determinar qual delas tem menor valor de  $f_{ik}$  para ser acoplada em S na posição indicada por  $k$ . No momento em que  $k$  for igual ao número total de tarefas,  $n$ , o método chegará ao seu fim, pois U estará vazio enquanto que S terá  $n$  tarefas.

## 8. Referências

---

- Ahmadi R.H. e Bagchi U.** Improved Lower Bounds for Minimizing the Sum of Completion Times of  $n$  Jobs over  $m$  Machines. // *European Journal of Operational Research*. - 1990. - Vol. 44. - pp. 331-336.
- Aldowaisan T. e Allahverdi A.** New Heuristics for  $m$ -machine no-wait Flowshop to Minimize Total Completion Time. // *OMEGA*. - 2004. - Vol. 32. - pp. 345-352.
- Allahverdi Ali e Aldowaisan Tariq** New heuristics to minimize total completion time in  $m$ -machine flowshops // *International Journal of Operation Research*. - 2002. - pp. 71-83.
- Baker K. R.** Introduction to sequencing and scheduling. - New York : Wiley, 1974.
- Campbell H. G., Dudek R. A. e Smith M. L.** A Heuristic Algorithm for  $n$ -job,  $m$ -machine Sequencing Problem. // *Management Science*. - 1970. - Vol. 16. - pp. 630-637..
- Chakravarthy Karunakaran e Rajendran Chandrasekharan** A heuristic for scheduling in a flowshop with the bicriteria of makespan and maximum tardiness minimization // *Production Planning & Control*. - 1999. - Vol. 10. - pp. 707-714.
- Framinan J.M., Leisten R. e Rajendran C.** Different initial sequences for the heuristic of Nawaz, Enscore and Ham to minimize makespan, idle time in the static permutation flowshop. // *International Journal of Production Research*. - 2003. - Vol. 41. - pp. 121-148.
- Framinan Jose M. e Leisten R.** An efficient constructive heuristic for flowtime minimisation in permutation flow shop // *Omega*. - 2003. - pp. 311-317.
- Framinan Jose M., Leisten Rainer e Usano Rafael Ruiz** Comparison of heuristics for flowtime minimisation in permutation flowshops // *Computers & Operations Research*. - 2005. - pp. 1237-1254.
- Garey M. R., Johnson D. S. e Sethi R.** The Complexity of Flowshop and Jobshop Scheduling. // *Mathematics of Operations Research*. - 1976. - Vol. 1. - pp. 117-129.
- Gigante Rodrigo Luiz** Heurística construtiva para a programação de operações flow shop permutacional. - São Carlos , 2010.
- Gupta J. N. D.** Heuristic Algorithms for Multistage Flowshop Scheduling Problem. // *AIIE Transactions*. - 1972. - Vol. 4. - pp. 11-18.
- Ho J.C.** Flowshop Sequencing with Mean Flowtime Objective. // *European Journal of Operational Research*,. - 1995. - Vol. 81. - pp. 571-578.
- Johnson S. M.** Optimal two- and three-stage production schedules with setup times included. // *Naval Research Logistics Quarterly*. - 1954. - Vol. 1. - pp. 61-68.
- Laha Dipak e Chakraborty Uday K.** An efficient heuristic approach to total flowtime minimization in permutation flowshop scheduling // *International J Adv Manuf Technol*. - 2008. - pp. 1018-1025.
- Laha Dipak e Chakravorty Arindam** A new heuristic for minimizing total completion time objective in permutation flow shop scheduling // *Int J Adv Manuf Technol*. - 2010.

- Laha Dipak e Sarin Subhash C.** A heuristic to minimize total flow time in permutation flow shop // Omega. - 2009. - pp. 734-739.
- Li Xiaoping, Wang Qian e Wu Cheng** Efficiente composite heuristics for total flowtime minimization in permutation flow shops // Omega. - 2009. - pp. 155-164.
- Liu Ji Yin e Reeves Colin R.** Constructive and composite heuristic solutions to the  $P/\sum C_i$  scheduling problem // European Journal of Operational Research. - 2001. - pp. 439-452.
- Miyazaki S., Nishiyama N. e Hashimoto F.** An Adjacent Pairwise Approach to the Mean Flowtime Scheduling Problem. // Journal of the Operations Research Society of Japan. - 1978. - Vol. 21. - pp. 287-299.
- Nagano Marcelo e Moccellini J. V.** Reducing mean flow time in permutation flow shop // Journal of the Operational Research Society. - 2008. - pp. 939-945.
- Nawaz M., Enscore E. E. e Ham I.** A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. // Omega. - 1983. - Vol. 11. - pp. 91-95.
- Rajendran C. e Chaudhuri D.** An Efficient Heuristic Approach to the Scheduling of Jobs in a Flowshop. // European Journal of Operational Research. - 1991. - Vol. 61. - pp. 318-325.
- Rajendran C. e Ziegler H.** An efficient Heuristic for Scheduling in a Flowshop to Minimize Total Weighted Flowtime of Jobs. // European Journal of Operational Research. - 1997. - Vol. 103. - pp. 129-138.
- Rajendran C.** Heuristic Algorithm for Scheduling in a Flowshop to Minimise Total Flowtime. // International Journal Production Economics. - 1993. - Vol. 29. - pp. 65-73.
- Rinnooy Kan A.H.G.** Machine Scheduling Problems: Classification, Complexity, and Computations. // The Hague: Nijhoff. - 1976..
- Taillard E.** Benchmarks for basic scheduling problems. // European Journal of Operational Research. - 1993. - Vol. 64. - pp. 278-285.
- Wang C., Chu C. e Proth J. M.** Heuristic Approaches for  $n/m/F/\sum C_i$  Scheduling Problems. // European Journal of Operational Research. - 1997. - Vol. 96. - pp. 636-644.
- Woo D. S. e Yim H. S.** A heuristic Algorithm for Mean Flowtime Objective in Flowshop Scheduling. // Computers & Operations Research. - 1998. - Vol. 25. - pp. 175-182.
- Woo H. S. e Yim D. S.** A heuristic algorithm for mean flowtime objective in flowshop scheduling // Computers & Operations Research. - 2005. - Vol. 25. - pp. 175-182.