# Soluções de Alto Desempenho para a Programação da Produção *Flow Shop*

**Marcelo Seido Nagano (EESC/USP)** drnagano@usp.br

**Fábio José Ceron Branco (EESC/USP)** fbranco@sc.usp.br

**João Vitor Moccellin (EESC/USP)** jvmoccel@sc.usp.br

*Resumo: Este trabalho é direcionado ao problema de programação de operações flow shop permutacional. Um novo método heurístico construtivo, com o objetivo de minimizar a Duração Total da Programação (makespan) é apresentado e comparado com os melhores heurísticos construtivos reportados na literatura, conhecidos por NEH e NEHKK. Os resultados da experimentação computacional mostraram que o novo método obtém soluções de alta qualidade em comparação com os métodos avaliados para o conjunto de problemas teste de Taillard.*
*Palavras-chave: Programação da produção; flow-shop permutacional; métodos heurísticos.*

## 1. Introdução

O problema de programação de operações *flow shop* é um problema da produção no qual *n* tarefas devem ser processadas, na mesma seqüência, em cada máquina de um conjunto de *m* máquinas distintas. Um caso específico de programação *flow shop*, denominado permutacional, é quando em cada máquina mantém-se a mesma ordem de processamento das tarefas. A solução do problema consiste em determinar dentre as ($n$!) seqüências possíveis das tarefas aquela que otimize uma determinada medida de desempenho, sendo uma das mais utilizadas a duração total da programação (*makespan*). Na teoria que estuda a complexidade dos problemas de natureza combinatorial, o problema em questão é classificado como *NP-hard* de forma que pode ser resolvido eficientemente de maneira ótima (solução exata) somente em casos de pequeno porte. Assim, muitos métodos heurísticos têm sido propostos para a solução do problema.

Para a modelagem deste tipo de problema é necessário assumir algumas hipóteses:

- Cada máquina está disponível continuamente, sem interrupções;
- Cada operação tem, no máximo, uma sucessora e uma precedente (fluxo unidirecional);
- Cada operação pode ser executada por apenas uma máquina;
- Sempre que uma operação é iniciada a mesma é finalizada sem que haja interrupção no seu processamento;
- Os tempos de preparação das máquinas (*setup*) estão inclusos nos tempos de processamento e estes independem das tarefas precedentes e;
- Cada máquina processa apenas uma tarefa de cada vez e cada tarefa é processada por apenas uma máquina de cada vez.

Os métodos heurísticos podem ser classificados de diversas maneiras. Uma delas classifica-os em Construtivos ou Melhorativos, dependendo da forma de obtenção da solução. Os métodos construtivos se caracterizam pelo fato de gerarem apenas uma solução que será a solução final do problema (PALMER, 1965; CAMPBELL *et al.*, 1970; GUPTA, 1971; DANNENBRING, 1977; NAWAZ *et al.*, 1983; KOULAMAS, 1998; DAVOUD POUR, 2001; NAGANO &

MOCCELLIN, 2002; KALCZYNSKI & KAMBUROWSKI, 2007). Esta solução pode ser gerada:

- Diretamente a partir da ordenação das tarefas segundo índices de prioridade calculados em função dos tempos de processamento das tarefas, como por exemplo: Palmer (1965) e Koulamas (1998);
- Escolhendo-se a melhor seqüência das tarefas a partir de um conjunto de seqüências também obtidas utilizando-se índices de prioridade associados às tarefas: Campbell, Dudek & Smith (1970) e Hundal & Rajgopal (1988);
- Ou ainda, a partir da geração sucessiva de seqüências parciais das tarefas (subseqüências) até a obtenção de uma seqüência completa através de algum critério de inserção de tarefas, como por exemplo: Nawaz *et al*. (1983), Nagano & Moccellin (2002), Kalczynski & Kamburowski (2007).

No caso dos métodos melhorativos, obtém-se uma solução inicial e posteriormente através de algum procedimento iterativo (geralmente envolvendo trocas de posições das tarefas na seqüência) busca-se obter uma seqüência das tarefas melhor que a atual quanto à medida de desempenho adotada, por exemplo: Osman & Potts (1989), Ogbu & Smith (1990), Widmer & Hertz (1989), Taillard (1990), Reeves (1993), Nowicki & Smutnicki (1996) e Grabowski & Wodecki (2004) que aplicaram a técnica de busca tabu (BT). Outros métodos são o *iterated local search* (*ILS*) de Stützle (1998) e, mais recentemente, dois novos métodos baseados no *ant-colony optimization* (*ACO*) de Rajendran & Ziegler (2004) e o método baseado no procedimento *scatter search* de Noorul Haq *et al.* (2007).

Dentre os métodos construtivos mais conhecidos, salienta-se o NEH proposto por Nawaz *et al.* (1983), que tem sido considerado até hoje como o melhor método heurístico construtivo para minimizar o *makespan* em *flow shop* permutacionais, com alta qualidade de solução e eficiência computacional. O Algoritmo NEH é bastante simples, podendo ser resumido como segue:

*Fase 1* - Ordenação inicial das tarefas segundo os respectivos tempos totais de processamento ( regra *LPT - Longest Processing Time*).

*Fase 2* - Construção da seqüência solução a partir da inserção gradativa das tarefas em seqüências parciais, conforme a ordenação do *Fase 1*.

Nagano & Moccellin (2002) propuseram uma heurística construtiva denominada N&M onde há uma mudança apenas na fase de ordenação inicial em relação ao NEH, sendo a ordenação inicial feita em dois passos:

*Passo 1* – Para cada tarefa calcular:

$$I_v = \sum TP_v - max \sum LBY_{Jv}$$

onde

$$max \sum LBY_{Jv} = \max_{\substack{u=1 \\ u \neq v}}^{n} \left\{ \sum_{k=1}^{m-1} LBY_{uv}^{k} \right\}$$

$$\sum TP_v = \sum_{k=1}^{m} p_{kv}$$

$p_{ki}$ = tempo de processamento da tarefa *i* na máquina *k*;

*n* = número de tarefas;

$m$ = número de máquinas.

$$LBY_{uv}^{k} = max\left(0, \left(p_{k+1,u} - p_{k,v}\right) - UBX_{uv}^{k}\right)$$

$$UBX_{uv}^{k} = max\left(0, UBX_{uv}^{k-1} + \left(p_{k-1,v} - p_{k,u}\right)\right)$$

sendo,

$$UBX_{uv}^{1} = 0$$

A tarefa $u$ é imediatamente precedente da tarefa $v$.

*Passo 2* – Ordenar todos os $I_i$ de forma decrescente.

A próxima fase é idêntica ao NEH. Este método apresentou resultados superiores com relação ao NEH (Nawaz *et al*., 1983).

Mais recentemente, Kalezynski & Kamburowski (2007) propuseram um novo método heurístico construtivo denominado NEHKK. O método proposto apresentou uma modificação do método NEH no processo de construção da seqüência solução, onde em caso de empate das soluções parciais nas subseqüências obtidas, um procedimento heuristico baseada na regra de JOHNSON (Johnson, 1954) é aplicado. Os resultados obtidos através da experimentação computacional verificaram o melhor desempenho comparados com o N&M (Nagano & Moccellin, 2002) e o NEH (Nawaz *et al*., 1983).

Neste trabalho, reportam-se sucintamente os resultados de uma pesquisa com o objetivo de desenvolver um novo método heurístico construtivo com desempenho superiores ao NEH e NEHKK. Tal método é baseado no procedimento de construção da seqüência solução utilizando-se de forma combinada as vizinhanças de busca de inserção e permutação de tarefas. O novo método proposto será avaliado verificando as características essenciais de um método heurístico: adequado equilíbrio entre a qualidade da solução e esforço computacional, simplicidade e facilidade de implementação.

## 2. Método heurístico proposto

O método heurístico proposto neste trabalho apresenta a seguinte estrutura sendo descrita a seguir.

*Passo 1* – Calcule, para cada tarefa, a soma dos tempos de processamento em todas as máquinas;

*Passo 2* – Ordene as $n$ tarefas de acordo com os valores não-crescentes das somas dos tempos de processamento;

*Passo 3* – Selecione as duas primeiras tarefas da ordenação obtida no *Passo 2*, seqüenciando-as de maneira a minimizar o *makespan*, considerando-se somente essas duas tarefas e que ocuparão as duas primeiras posições da seqüência $S$.

*Passo 4* – Para $k$ = 3 a $n$,

- Selecione a tarefa que ocupa a $k$-ésima posição na ordenação obtida no *Passo 2*;
- Acrescente a tarefa na $k$-ésima posição da seqüência atual $S$, e calcule seu respectivo *makespan* para a seqüência parcial;
- Considerando toda a Vizinhança de Inserção da seqüência parcial com $k$ tarefas, constituída de $(k-1)^2$ seqüências, determine a seqüência $S_i$ associada ao menor *makespan*;

- Se o *makespan* de $S_i$ é menor que de $S$, atualize $S$ com $S_i$ e, considerando toda a Vizinhança de Permutação da seqüência parcial com $k$ tarefas, constituída de $k(k-1)/2$ seqüências, determine a seqüência $S_p$ associada ao menor *makespan*;
- Se o *makespan* de $S_p$ é menor que de $S$, atualize $S$ com $S_p$;
- Se o *makespan* de $S_i$ *não* é menor que de $S$, considerando toda a Vizinhança de Permutação da seqüência parcial com $k$ tarefas, constituída de $k(k-1)/2$ seqüências, determine a seqüência $S_p$ associada ao menor *makespan*;
- Se o *makespan* de $S_p$ é menor que de $S$, atualize $S$ com $S_p$.

## 3. Experimentação computacional e método de análise

Para a avaliação do desempenho do método proposto, denominado NBM-*FS*, foram realizadas experimentações computacionais utilizando os problemas de Taillard (1993) (http://ina2.eivd.ch/collaborateurs/etd/problemes.dir/ordonnancement.dir/ordonnancement.html).

Neste trabalho, o NBM-*FS* foi codificado em linguagem *DELPHI* e processado em um microcomputador *Pentium* IV de 3.4 GHz, com 1 *GByte* de *RAM*.

As estatísticas usadas para avaliar o desempenho do método 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 a melhor duração total da programação (*makespan*), e o número total de problemas resolvidos. A segunda quantifica o desvio relativo ($DR_h$) que o método $h$ obtém em relação a melhor duração total da programação obtido para um mesmo problema, sendo calculado conforme segue:

$$DR_h = \frac{(M_h - M_*)}{M_*}$$

onde,

$M_h$: duração total da programação obtido pelo método $h$;

$M_*$: melhor duração total da programação obtido pelo(s) método(s), para um determinado problema.

## 4. Análise dos resultados

Conforme apresentado anteriormente os resultados obtidos pelo método NBM-*FS* foram comparados com os métodos heurísticos NEH (Nawaz *et al*., 1983) e NEHKK (Kalezynski & Kamburowski, 2007). Os resultados principais são apresentados nas Tabelas 1, 2 e 3.

A Tabela 1 apresenta a procentagem de sucesso e o desvio relativo médio entre os métodos avaliados. Verifica-se a superioridade do método NBM-*FS* para o conjunto de problemas em relação as estatísticas utilizadas. Para tarefas iguais e superiores a 50 o método NBM-*FS* apresentou 100% de sucesso, e o desvio relativo médio igual a zero.

Tabela 1 – Desvio Relativo Médio e Porcentagem de Sucesso entre os métodos NEH, NEHKK e NBM-*FS*

| | | Porcentagem de Sucesso (%) | | | Desvio Relativo Médio (%) | | |
|---|---|---|---|---|---|---|---|
| *n* | *m* | NEH | NEHKK | NBM-*FS* | NEH | NEHKK | NBM-*FS* |
| 20 | 5 | 20 | 50 | 70 | 3,229 | 1,192 | 1,240 |
| | 10 | 0 | 0 | 100 | 2,374 | 2,220 | 0,000 |
| | 20 | 0 | 10 | 90 | 1,780 | 1,626 | 0,306 |
| Média | | 6,67 | 20,00 | 86,67 | 2,461 | 1,679 | 0,515 |
| 50 | 5 | 10 | 10 | 100 | 0,538 | 0,491 | 0,000 |
| | 10 | 0 | 0 | 100 | 2,349 | 2,285 | 0,000 |
| | 20 | 0 | 0 | 100 | 2,454 | 2,170 | 0,000 |
| Média | | 3,33 | 3,33 | 100,00 | 1,780 | 1,649 | 0,000 |
| 100 | 5 | 20 | 30 | 100 | 0,421 | 0,319 | 0,000 |
| | 10 | 0 | 0 | 100 | 1,215 | 0,975 | 0,000 |
| | 20 | 0 | 0 | 100 | 2,052 | 1,885 | 0,000 |
| Média | | 6,67 | 10,00 | 100,00 | 1,229 | 1,060 | 0,000 |
| 200 | 10 | 10 | 10 | 100 | 0,760 | 0,714 | 0,000 |
| | 20 | 0 | 0 | 100 | 2,069 | 1,939 | 0,000 |
| Média | | 5,00 | 5,00 | 100,00 | 1,414 | 1,327 | 0,000 |
| 500 | 20 | 0 | 0 | 100 | 1,173 | 1,181 | 0,000 |
| Média | | 0,00 | 0,00 | 100,00 | 1,173 | 1,181 | 0,000 |

A Tabela 2 apresenta o desvio relativo médio por classes de problemas para os métodos NEH, NEHKK e NBM-*FS* em relação as melhores soluções obtidas para os problemas testes de Taillard. Pode-se verificar que o método NBM-*FS* apresentou desvio relativo médio inferiores aos métodos NEH e NEHKK para tarefas superiores a 20, consubstanciando a superioridade do método proposto, já mostrada pela tabela anterior.

Tabela 2 – Desvio Relativo Médio dos métodos NEH, NEHKK e NBM-*FS* em relação as melhores soluções para os problemas testes de Taillard

| *n* | *m* | NEH | NEHKK | NBM-*FS* |
|---|---|---|---|---|
| 20 | 5 | 4,564 | 2,485 | 2,543 |
| | 10 | 4,601 | 4,450 | 2,182 |
| | 20 | 3,731 | 3,576 | 2,231 |
| Média | | 4,299 | 3,503 | 2,319 |
| 50 | 5 | 0,727 | 0,680 | 0,189 |
| | 10 | 5,073 | 5,006 | 2,666 |
| | 20 | 6,660 | 6,361 | 4,105 |
| Média | | 4,153 | 4,016 | 2,320 |
| 100 | 5 | 0,527 | 0,425 | 0,106 |
| | 10 | 2,215 | 1,972 | 0,988 |
| | 20 | 5,345 | 5,172 | 3,227 |
| Média | | 2,696 | 2,523 | 1,440 |
| 200 | 10 | 1,258 | 1,212 | 0,495 |
| | 20 | 4,435 | 4,303 | 2,320 |
| Média | | 2,847 | 2,758 | 1,408 |
| 500 | 20 | 2,066 | 2,074 | 0,883 |
| Média | | 2,066 | 2,074 | 0,883 |

Conforme observado na Tabela 3 o método NBM-*FS* apresentou soluções iguais para os problemas de Taillard para 8 problemas testes (Ta008, Ta031, Ta033, Ta038, Ta061, Ta063, Ta067 e Ta069). Estes resultados comprovam que o método NBM-*FS* apresentou soluções de alta qualidade, já que os melhores resultados para os problemas testes de Taillard foram obtidos por métodos meta-heurísticos.

Tabela 3 – Comparação dos métodos em relação as melhores soluções para os problemas teste de Taillard

| Problema | Melhor Solução | NEH | NEHKK | NBM-*FS* |
|---|---|---|---|---|
| Ta001 | 1278 | **1286** | **1286** | 1297 |
| Ta002 | 1359 | **1365** | **1365** | 1370 |
| Ta003 | 1081 | 1159 | **1132** | **1132** |
| Ta004 | 1293 | 1325 | 1325 | **1301** |
| Ta005 | 1235 | 1305 | 1305 | **1250** |
| Ta006 | 1195 | 1228 | 1225 | **1198** |
| Ta007 | 1234 | 1278 | **1251** | **1251** |
| Ta008 | 1206 | 1223 | 1227 | **1206*** |
| Ta009 | 1230 | 1291 | 1274 | **1253** |
| Ta010 | 1108 | 1291 | **1127** | 1253 |
| Ta011 | 1582 | 1680 | 1654 | **1598** |
| Ta012 | 1659 | 1729 | 1776 | **1717** |
| Ta013 | 1496 | 1557 | 1538 | **1532** |
| Ta014 | 1377 | 1439 | 1433 | **1406** |
| Ta015 | 1419 | 1502 | 1502 | **1448** |
| Ta016 | 1397 | 1453 | 1433 | **1427** |
| Ta017 | 1484 | 1562 | 1555 | **1517** |
| Ta018 | 1538 | 1609 | 1620 | **1583** |
| Ta019 | 1593 | 1647 | 1647 | **1617** |
| Ta020 | 1591 | 1653 | 1656 | **1622** |
| Ta021 | 2297 | 2410 | 2443 | **2344** |
| Ta022 | 2099 | 2150 | 2134 | **2133** |
| Ta023 | 2326 | 2411 | 2418 | **2379** |
| Ta024 | 2223 | 2262 | 2263 | **2252** |
| Ta025 | 2291 | 2397 | 2384 | **2328** |
| Ta026 | 2226 | 2349 | 2349 | **2253** |
| Ta027 | 2273 | 2362 | 2383 | **2336** |
| Ta028 | 2200 | 2249 | 2249 | **2242** |
| Ta029 | 2237 | 2320 | 2313 | **2293** |
| Ta030 | 2178 | 2277 | **2220** | 2288 |
| Ta031 | 2724 | 2733 | 2732 | **2724*** |
| Ta032 | 2834 | 2843 | 2843 | **2838** |
| Ta033 | 2621 | 2640 | 2650 | **2621*** |
| Ta034 | 2751 | **2782** | **2782** | **2782** |
| Ta035 | 2863 | 2868 | 2871 | **2864** |
| Ta036 | 2829 | 2850 | 2845 | **2835** |
| Ta037 | 2725 | 2758 | 2735 | **2732** |
| Ta038 | 2683 | 2721 | 2695 | **2683*** |
| Ta039 | 2552 | 2576 | 2607 | **2554** |
| Ta040 | 2782 | 2790 | 2786 | **2783** |
| Ta041 | 2991 | 3135 | 3149 | **3126** |
| Ta042 | 2867 | 3032 | 3059 | **2944** |
| Ta043 | 2839 | 2986 | 3000 | **2924** |
| Ta044 | 3063 | 3198 | 3196 | **3121** |
| Ta045 | 2976 | 3160 | 3160 | **3094** |
| Ta046 | 3006 | 3178 | 3158 | **3100** |
| Ta047 | 3093 | 3277 | 3252 | **3156** |
| Ta048 | 3037 | 3123 | 3137 | **3042** |
| Ta049 | 2897 | 3002 | 3025 | **2964** |
| Ta050 | 3065 | 3257 | 3187 | **3156** |
| Ta051 | 3850 | 4082 | 4093 | **3986** |
| Ta052 | 3704 | 3921 | 3938 | **3847** |
| Ta053 | 3640 | 3927 | 3868 | **3795** |
| Ta054 | 3720 | 3969 | 3913 | **3895** |
| Ta055 | 3610 | 3835 | 3874 | **3732** |
| Ta056 | 3681 | 3914 | 3878 | **3826** |
| Ta057 | 3704 | 3952 | 3959 | **3875** |
| Ta058 | 3691 | 3938 | 3969 | **3849** |

| | | | | |
|---|---|---|---|---|
| Ta059 | 3743 | 3952 | 3951 | **3902** |
| Ta060 | 3756 | 4079 | 4015 | **3915** |
| Ta061 | 5493 | 5519 | 5514 | **5493*** |
| Ta062 | 5268 | 5348 | **5284** | **5284** |
| Ta063 | 5175 | 5219 | 5196 | **5175*** |
| Ta064 | 5014 | **5023** | **5023** | **5023** |
| Ta065 | 5250 | 5266 | 5266 | **5265** |
| Ta066 | 5135 | **5139** | **5139** | **5139** |
| Ta067 | 5246 | 5259 | 5288 | **5246*** |
| Ta068 | 5094 | 5120 | 5129 | **5099** |
| Ta069 | 5448 | 5489 | 5489 | **5448*** |
| Ta070 | 5322 | 5341 | 5341 | **5328** |
| Ta071 | 5770 | 5846 | 5886 | **5783** |
| Ta072 | 5349 | 5453 | 5460 | **5388** |
| Ta073 | 5676 | 5824 | 5743 | **5705** |
| Ta074 | 5781 | 5929 | 5926 | **5853** |
| Ta075 | 5467 | 5679 | 5647 | **5540** |
| Ta076 | 5303 | 5375 | 5373 | **5331** |
| Ta077 | 5595 | 5704 | 5681 | **5675** |
| Ta078 | 5617 | 5760 | 5752 | **5694** |
| Ta079 | 5871 | 6032 | 5993 | **5960** |
| Ta080 | 5845 | 5918 | 5922 | **5903** |
| Ta081 | 6202 | 6541 | 6502 | **6466** |
| Ta082 | 6183 | 6523 | 6494 | **6373** |
| Ta083 | 6271 | 6639 | 6534 | **6482** |
| Ta084 | 6269 | 6557 | 6611 | **6460** |
| Ta085 | 6314 | 6695 | 6590 | **6499** |
| Ta086 | 6364 | 6664 | 6725 | **6557** |
| Ta087 | 6268 | 6632 | 6631 | **6487** |
| Ta088 | 6401 | 6739 | 6839 | **6648** |
| Ta089 | 6275 | 6677 | 6630 | **6460** |
| Ta090 | 6434 | 6677 | 6683 | **6580** |
| Ta091 | 10862 | 10942 | 10986 | **10872** |
| Ta092 | 10480 | 10716 | 10722 | **10559** |
| Ta093 | 10922 | 11025 | 11025 | **10967** |
| Ta094 | 10889 | **11057** | **11057** | **11057** |
| Ta095 | 10524 | 10645 | 10575 | **10556** |
| Ta096 | 10329 | 10458 | 10476 | **10345** |
| Ta097 | 10854 | 10989 | 10965 | **10885** |
| Ta098 | 10730 | 10829 | 10798 | **10770** |
| Ta099 | 10438 | 10574 | 10594 | **10465** |
| Ta100 | 10675 | 10807 | 10795 | **10758** |
| Ta101 | 11195 | 11625 | 11640 | **11450** |
| Ta102 | 11203 | 11675 | 11781 | **11497** |
| Ta103 | 11281 | 11852 | 11789 | **11585** |
| Ta104 | 11275 | 11803 | 11764 | **11562** |
| Ta105 | 11259 | 11685 | 11624 | **11464** |
| Ta106 | 11176 | 11629 | 11658 | **11400** |
| Ta107 | 11360 | 11833 | 11798 | **11535** |
| Ta108 | 11334 | 11913 | 11883 | **11595** |
| Ta109 | 11192 | 11673 | 11665 | **11527** |
| Ta110 | 11288 | 11869 | 11805 | **11558** |
| Ta111 | 26059 | 26670 | 26743 | **26365** |
| Ta112 | 26520 | 27232 | 27167 | **26817** |
| Ta113 | 26371 | 26848 | 26925 | **26658** |
| Ta114 | 26456 | 27055 | 26988 | **26666** |
| Ta115 | 26334 | 26727 | 26788 | **26482** |
| Ta116 | 26477 | 26992 | 27111 | **26698** |
| Ta117 | 26389 | 26797 | 26731 | **26525** |
| Ta118 | 26560 | 27138 | 27027 | **26793** |

| | | | | |
|---|---|---|---|---|
| Ta119 | 26005 | 26631 | 26627 | **26265** |
| Ta120 | 26457 | 26984 | 26987 | **26685** |

Quanto ao tempo computacional, o método NBM-*FS* apresentou-se mais acentuado. Entretanto, o tempo médio computacional foi de 6,5 segundos por problema de 100 tarefas, sendo assim não relevante sobre os aspectos práticos.

## 5. Considerações finais

Os resultados experimentais mostraram que o método heurístico NBM-*FS* apresentou um desempenho superior tanto na porcentagem de sucesso e no desvio relativo médio para o conjunto de problemas avaliados, em comparação com os melhores métodos referenciados na literatura para o problema (NEH e NEHKK), além de obter soluções de altíssima qualidade.

O problema clássico de sequenciamento de tarefas em um ambiente de produção *flow shop* tem sido objeto de intensos esforços de pesquisa nos últimos 50 anos e, para fins práticos, tal problema pode ser considerado já resolvido. Na literatura, as medidas de desempenho mais utilizadas têm sido a duração total da programação (*makespan*) e a soma dos tempos de fluxo (*total flowtime*), com freqüência maior para a primeira. Entretanto, tendo em vista sua complexidade, a busca de métodos heurísticos simples e cada vez eficazes quanto à qualidade da solução ainda permanece como uma direção de pesquisa.

A realização da pesquisa relatada neste trabalho foi motivada pelas considerações acima, procurando resgatar as características essenciais de um método heurístico, ou seja, adequado equilíbrio entre a qualidade da solução e a eficiência computacional, simplicidade e facilidade de implementação.

## Referências

DANNENBRING, D. G. *An evaluation of flow-shop sequencing heuristics*. Management Science. V. 23, p.1174-1182, 1977.

CAMPBELL, H. G.; DUDEK, R. A. & SMITH, M. L. *A heuristic algorithm for the n-job, m-machine sequencing problem*. Management Science. V. 16, p.630-637, 1970.

DAVOUD POUR, H. *A new heuristic for the n-job, m-machine flow-shop problem*. Production Planning and Control. V. 12, n. 07, p.648-653, 2001.

GRABOWSKI, J. & WODECKI, M. *A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion*. Computer & Operations Research. V. 31, p.1891-1909, 2004.

GUPTA, J. N. D. *A functional heuristic algorithm for the flow-shop scheduling problem.* Operational Research Quarterly. V. 22, p.39-47, 1971.

HUNDAL, T. S. & RAJGOPAL, J. *An extension of Palmer's heuristic for the flow-shop scheduling problem*. International Journal of Production Research. V. 26, p.1119-1124, 1988.

JOHNSON, S.M. *Optimal two-and three-stage production schedules with setup times included*. Naval Research Logistics Quarterly. V. 1, p.61-68, 1954.

KALCZYNSKI, P. J. & KAMBUROWSKI, J. *On the NEH heuristic for minimizing the makespan in permutation flow shops*. Omega, The International Journal of Management Science. V. 35, p.53-60, 2007.

KOULAMAS, C. *A new constructive heuristic for the flowshop scheduling problem*. European Journal of Operational Research. V. 105, p.66-71, 1998.

NAGANO, M. S. & MOCCELLIN, J. V. *A high quality solution constructive for flow shop sequencing.* Journal of the Operational Research Society. V. 53, p.1374-1379, 2002.

NAWAZ, M.; ENSCORE JR., E. & E., HAM. *A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem*. Omega, The International Journal of Management Science. V. 11, p.91-95, 1983.

NOORUL HAQ, A.; SARAVANAN, M.; VIVEKRAJ, A.R. & PRASAD, T. *Scatter search approach for general flowshop scheduling problem*. Journal The International Journal of Advanced Manufacturing Technology. ISSN 0268-3768. V. 31, n.7-8, 2007.

NOWICKI, E. & SMUTNICKI, C. *A fast tabu algorithm for the permutation flow-shop problem*. Journal of Operational Research. V. 91, p.160-175, 1996.

OGBU, F.A. & SMITH, D.K. *The application of the simulated annealing algorithms to the solution of the* $n/m/C_{max}$ *flowshop problem*. Computers & Operations Research. V. 17, n.3, p.243-253, 1990.

OSMAN, I. H. & POTTS, C. N. *Simulated annealing for permutation flow-shop scheduling*. Omega, The International Journal of Management Science. V. 17, p.551-557, 1989.

PALMER, D. S. *Sequencing jobs through a multi-stage process in the minimum total time - A quick method of obtaining a near optimum*. Operational Research Quarterly. V. 16, p.101-107, 1965.

RAJENDRAN, C. & ZIEGLER, H. *Ant-colony algorithms for permutation flowshop scheduling to minimize makespan/total flowtime of jobs*. European Journal of Operational Research. V. 155, p.426-436, 2004.

REEVES, C. R. *A genetic algorithm for flowshop sequencing*. Computers and Operations Research. V. 22, p.5-13, 1995.

STÜTZLE, T. *Applying iterated local search to the permutation flow shop problem*. Technical report, AIDA-98-04, TU Darmstadt, FG Intellektik, 1998.

TAILLARD, E. *Some efficient heuristic methods for the flow shop sequencing problem*. European Journal of Operational Research. V. 47, p.65-74, 1990.

WIDMER, M. & HERTZ, A. *A new heuristic method for the flow shop sequencing problem*. European Journal of Operational Research. V. 41, p.186-193, 1989.