

UMA PROPRIEDADE ESTRUTURAL DO PROBLEMA DE PROGRAMAÇÃO DA PRODUÇÃO *FLOW SHOP* PERMUTACIONAL COM TEMPOS DE *SETUP*

João Vitor Moccellin *

Departamento de Engenharia de Produção
Escola de Engenharia de São Carlos
Universidade de São Paulo (USP)
São Carlos – SP
jvmoccel@sc.usp.br

Marcelo Seido Nagano

Departamento de Engenharia de Produção
Escola de Engenharia de São Carlos
Universidade de São Paulo (USP)
São Carlos – SP
drnagano@usp.br

* *Corresponding author* / autor para quem as correspondências devem ser encaminhadas

Recebido em 08/2006; aceito em 10/2007 após 1 revisão
Received August 2006; accepted October 2007 after one revision

Resumo

Neste artigo apresenta-se uma propriedade estrutural do problema de programação da produção *flow shop* permutacional com tempos de *setup* das máquinas separados dos tempos de processamento das tarefas, a qual foi identificada a partir de investigações que foram realizadas sobre as características do problema. Tal propriedade fornece um limitante superior do tempo de máquina parada entre a sua preparação e o início de execução das tarefas. Utilizando a propriedade, o problema original de programação da produção com minimização do *makespan* pode ser resolvido de maneira heurística por meio de uma analogia com o problema assimétrico do caixeiro-viajante.

Palavras-chave: programação da produção; *flow shop* permutacional; tempos de *setup*.

Abstract

This paper deals with the permutation flow shop scheduling problem with separated machine setup times. As a result of an investigation on the problem characteristics a structural property is introduced. Such a property provides an upper bound on the idle time of the machines between the setup task and the job processing. As an application of this property, the original scheduling problem with the makespan criterion can be heuristically solved by an analogy with the asymmetric traveling salesman problem.

Keywords: production scheduling; permutation flow shop; setup times.

1. Introdução

O problema de programação *flow shop* é um problema de programação da produção no qual n tarefas devem ser processadas, na mesma seqüência, em 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 otimiza alguma medida de desempenho da programação, sendo que as usuais consistem na minimização da Duração Total da Programação (*makespan*), ou minimização do Tempo Médio de Fluxo. A primeira relaciona-se a uma utilização eficiente dos recursos (máquinas) enquanto que a segunda busca minimizar o estoque em processamento.

Esse problema de programação da produção tem sido intensamente estudado na literatura, desde os resultados reportados por Johnson (1954) para o problema com somente duas máquinas. Hejazi & Saghafian (2005) efetuaram uma ampla revisão da literatura referente ao ambiente de produção *flow shop*, mostrando que a maioria das pesquisas efetuadas considera os tempos de preparação (*setup*) das máquinas não significativos ou então os incluem nos tempos de processamento das tarefas. Isto simplifica a análise das aplicações, porém afeta a qualidade da programação quando tais tempos têm uma variabilidade relevante em função da ordenação das tarefas nas máquinas ou mesmo quando as máquinas podem ser preparadas antecipadamente para executarem as tarefas, não necessitando aguardar o término das tarefas nas máquinas precedentes.

Recentemente, o *European Journal of Operational Research* editou um número especial em comemoração aos 50 anos de publicação do primeiro artigo sobre programação da produção *flow shop* (Johnson, 1954). No *Editorial paper* dessa edição, Gupta & Stafford Jr. (2006) procuram mostrar a evolução dos problemas e métodos de solução em ambiente *flow shop* nas últimas 5 décadas. Nesse contexto, pode-se observar que a consideração de problemas com um tratamento explícito dos tempos de preparação das máquinas praticamente teve início a partir da terceira década (1975-1984), com um aumento progressivo de interesse por parte dos pesquisadores a partir da quarta década. Essa tendência de intensificação de esforços em pesquisas abordando tais problemas encontra-se detalhada na revisão da literatura efetuada por Cheng, Gupta & Wang (2000). Nesse artigo, os autores apresentam uma classificação dos problemas de programação da produção *flow shop* com tempos de *setup* e um conjunto de trabalhos relevantes realizados. Na conclusão do artigo são identificados alguns temas com potencial para futuras pesquisas, entre os quais o problema com múltiplos estágios (número de máquinas $m > 2$), sistemas de produção *no-wait* e modelos com medidas de desempenho multi-critério.

A consideração explícita dos tempos de preparação das máquinas é necessária em sistemas de produção como indústrias químicas, por exemplo, na produção de tintas onde o processo de limpeza é diferenciado dependendo da cor que estava sendo produzida e daquela que será fabricada em seguida (Simons Jr., 1992). Tal ambiente de produção também é comum em indústrias gráficas, têxteis e de produtos plásticos (Das, Gupta & Khumawala, 1995).

Basicamente, existem dois tipos de tempos de preparação de máquinas tratados separadamente dos tempos de processamento das tarefas. No primeiro, tal tempo depende somente da tarefa a ser executada (independente da seqüência). O segundo depende tanto da tarefa a ser executada quanto daquela que foi processada imediatamente antes na mesma máquina, portanto dependente da seqüência.

Este trabalho trata do problema *flow shop* permutacional com tempos de preparação das máquinas separados dos tempos de processamento das tarefas, podendo ser dependentes ou não da seqüência de execução das tarefas nas máquinas, que será denotado por FSP-TPS, e tem como objetivo apresentar uma propriedade relevante desse problema de programação da produção, proveniente de um estudo na sua estrutura. Na literatura, esse problema é considerado fortemente *NP-hard*. Essa complexidade pode ser entendida por meio da breve exposição que segue.

Garey, Johnson & Sethi (1976) provaram que quando a medida de desempenho for o tempo médio de fluxo, o problema clássico de programação da produção *flow shop* permutacional, com os tempos de *setup* incluídos nos tempos de processamento das tarefas, é fortemente *NP-hard* para um ambiente com somente 2 máquinas. A mesma complexidade ocorre na minimização da duração total da programação (*makespan*) para o ambiente de produção com 3 máquinas. Conclui-se, portanto, que para o caso geral com um número de máquinas maior que dois tais problemas também são fortemente *NP-hard*. Pinedo (1995) apresenta uma estrutura hierárquica de complexidade envolvendo as denominadas Medidas de Desempenho Regulares, duas das quais são as mencionadas acima (tempo médio de fluxo e *makespan*). Na referida estrutura, essas duas medidas de desempenho ocupam os níveis de menor complexidade, o que permite concluir que a complexidade fortemente *NP-hard* é mantida para qualquer medida de desempenho regular. Para finalizar, a consideração explícita dos tempos de *setup*, separados dos tempos de processamento das tarefas, leva o problema a um nível superior de complexidade (Pinedo, 1995). Portanto, pode-se concluir que o problema FSP-TPS, tratado neste trabalho, é fortemente *NP-hard*.

1.1 Problema FSP-TPS com tempos de preparação independentes da seqüência das tarefas

O exame da literatura indica que o primeiro trabalho para este caso do problema FSP-TPS foi desenvolvido por Yoshida & Hitomi (1979) para um *flow shop* com apenas duas máquinas, com o objetivo de minimizar o *makespan*. Os autores mostraram que o problema pode ser resolvido com solução ótima, aplicando-se a tradicional regra de Johnson (1954). Diversos outros trabalhos, com minimização do *makespan* e solução ótima, surgiram posteriormente para o mesmo caso de duas máquinas, porém com restrições adicionais como *due dates* para as tarefas (Khurana & Bagga, 1985), tempos de remoção das tarefas (*removal times*) e condição de processamento contínuo (Gupta, Strusevich & Zwaneveld, 1997). Outros trabalhos, também buscando a solução ótima, utilizaram medidas de desempenho da programação tais como a minimização do tempo médio de fluxo (Bagga & Khurana, 1986), e a minimização do *lateness* máximo (Allahverdi & Aldowaisan, 1998).

Para *flow shops* com múltiplas máquinas, no trabalho já mencionado de Yoshida & Hitomi (1979) foi mostrado que mesmo para três máquinas, a minimização do *makespan* não necessariamente é obtida por uma programação permutacional. Mesmo assim, para o caso de diversas máquinas, as pesquisas relatadas na literatura consideram somente soluções permutacionais obtidas por métodos heurísticos. Nessas pesquisas, busca-se a minimização do *makespan* sob restrições como, por exemplo, *buffer* limitado (Park & Steudel, 1991), tempos de transferência das tarefas entre as máquinas (Cao & Bedworth, 1992) e tempos de remoção das tarefas não nulos (Rajendran & Ziegler, 1997).

1.2 Problema FSP-TPS com tempos de preparação dependentes da seqüência das tarefas

Para este caso de tempos de preparação das máquinas dependentes da seqüência das tarefas, o procedimento pioneiro de solução do problema foi desenvolvido por Gupta (1969, 1975) fundamentado na técnica de busca lexicográfica.

Para *flow shops* com duas máquinas, Corwin & Esogbue (1974) propuseram um procedimento de Programação Dinâmica para obtenção da solução ótima, quanto ao *makespan*, porém com tempos de preparação dependentes da seqüência das tarefas em somente uma das máquinas. Os primeiros métodos heurísticos para o ambiente com duas máquinas foram propostos por Gupta & Darrow (1985, 1986). Os métodos foram avaliados entre si considerando diferentes relações na ordem de grandeza dos tempos de processamento das tarefas e dos tempos de preparação das máquinas.

A grande maioria dos trabalhos reportados na literatura, para o caso de múltiplas máquinas, tem como medida de desempenho o *makespan*. Mesmo com aplicação restrita a problemas de pequeno porte, diversos métodos de solução ótima foram desenvolvidos, utilizando técnicas de Programação Linear Inteira Mista (Srikar & Ghosh, 1986; Stafford & Tseng, 1990; Rios-Mercado & Bard, 1996) e métodos de enumeração *Branch & Bound* (Rios-Mercado & Bard, 1999a).

Para problemas de médio e grande porte salientam-se os métodos heurísticos. Os mais conhecidos são os métodos denominados *SETUP* e *TOTAL* propostos por Simons Jr. (1992) e um procedimento do tipo *GRASP* (*Greedy Randomized Adaptive Search Procedure*) desenvolvido por Rios-Mercado & Bard (1998). Os métodos *SETUP* e *TOTAL* fundamentam-se em uma analogia do problema FSP-TPS com o problema assimétrico do caixeiro-viajante (*ATSP – Asymmetric Traveling Salesman Problem*). No caso do *TOTAL*, a matriz de “distâncias” entre pares de tarefas é obtida considerando-se a soma dos tempos de processamento das tarefas e dos tempos de preparação em todas as máquinas. No *SETUP* consideram-se somente os tempos de preparação das máquinas. No trabalho de Rios-Mercado & Bard (1998) foi efetuada uma comparação de desempenho entre o método *GRASP* e o *SETUP*. De maneira geral, o *GRASP* superou o *SETUP* quando os tempos de preparação das máquinas foram gerados a partir de uma distribuição uniforme discreta no intervalo [1, 10] e os tempos de processamento das tarefas utilizando a distribuição [1, 99]. Entretanto, quando esta última distribuição foi utilizada para ambos, tempos de preparação e processamento, o método *SETUP* teve um desempenho melhor que o *GRASP*. Rios-Mercado & Bard (1999b) propuseram uma extensão do método *SETUP* (Simons Jr., 1992) que foi denominado *HYBRID*. Essencialmente, esse método em uma primeira fase incorpora ao *SETUP* alguns aspectos característicos do problema FSP-TPS, e na segunda fase procura melhorar a solução inicial por meio de um procedimento de busca local. Os resultados advindos de uma experimentação computacional mostraram que, em geral, o *HYBRID* pode ser considerado superior ao método *GRASP*, quando os tempos de preparação das máquinas apresentam uma maior variabilidade, por exemplo, quando são gerados a partir de uma distribuição uniforme discreta no intervalo [1, 50] em comparação com os valores de uma distribuição [1, 10].

2. Uma Propriedade Estrutural do Problema FSP-TPS

Considere-se o problema de Programação *Flow Shop* Permutacional, com n tarefas $\{J_1, J_2, \dots, J_i, \dots, J_n\}$ e m máquinas $\{M_1, M_2, \dots, M_k, \dots, M_m\}$.

No problema tradicional, com os tempos de preparação das máquinas incluídos nos tempos de processamento das tarefas, são conhecidas as seguintes relações de viabilidade entre tempos de máquina parada e tempos de espera entre operações sucessivas de uma mesma tarefa, quando se considera que a tarefa J_u precede diretamente J_v na seqüência de execução das n tarefas (French, 1982):

$$a) X_{u,v}^k + p_{k,v} + Y_{u,v}^k = Y_{t,u}^k + p_{k+1,u} + X_{u,v}^{k+1}, \text{ e} \quad (1)$$

$$b) \text{ Se } X_{u,v}^{k+1} > 0 \text{ então } Y_{u,v}^k = 0 \quad (2)$$

$$\text{Se } Y_{u,v}^k > 0 \text{ então } X_{u,v}^{k+1} = 0, \quad (3)$$

onde

$X_{i,j}^q$ = intervalo de tempo entre o término de J_i e o início de J_j na máquina M_q ;

$p_{q,j}$ = tempo de processamento de J_j na máquina M_q (incluindo o tempo de preparação da máquina);

$Y_{i,j}^q$ = intervalo de tempo entre o término de J_j na máquina M_q e o seu início na máquina M_{q+1} , com J_i precedendo diretamente J_j .

As relações (1) a (3) são ilustradas na Figura 1.

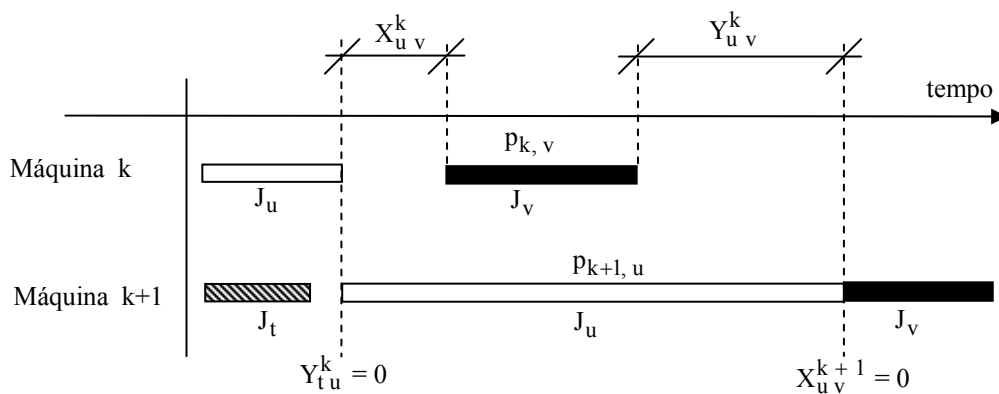


Figura 1 – Relações de viabilidade para duas tarefas adjacentes.

Para o problema FSP-TPS, objeto deste trabalho, considere-se duas tarefas adjacentes quaisquer J_u e J_v , e a atividade de preparação das máquinas nessa condição de tarefas sucessivas (independente ou não da seqüência), denotada por S_{uv} . Desta forma, S_{uv} pode ser considerada como se fosse mais uma tarefa a ser sempre executada entre J_u e J_v . Além disso, uma vez concluída a tarefa J_u em uma máquina qualquer, essa máquina já pode ser preparada para executar a tarefa J_v , não necessitando aguardar o término da operação de J_v na máquina anterior. Em algumas situações a preparação da máquina somente seria completada com a tarefa J_v na máquina, por exemplo, para o posicionamento e fixação da peça (tarefa). Nesse caso, o tempo para tal atividade é incorporado ao tempo de processamento de J_v na mencionada máquina. Portanto, não há tempo de máquina parada entre o término de J_u e o início da preparação S_{uv} .

Assim, pode-se considerar que J_u , S_{uv} e J_v seriam três “tarefas” sucessivas em uma seqüência qualquer das tarefas. Desta forma, a expressão de viabilidade (1) entre tarefas sucessivas pode ser aplicada para os pares (J_u, S_{uv}) e (S_{uv}, J_v) , levando a:

$$X_{u S_{uv}}^k + s_{k, uv} + Y_{u S_{uv}}^k = Y_{S_{tu} u}^k + p_{k+1, u} + X_{u S_{uv}}^{k+1} \quad (\text{entre } J_u \text{ e } S_{uv}) \quad (4)$$

$$X_{S_{uv} v}^k + p_{k, v} + Y_{S_{uv} v}^k = Y_{u S_{uv}}^k + s_{k+1, uv} + X_{S_{uv} v}^{k+1} \quad (\text{entre } S_{uv} \text{ e } J_v) \quad (5)$$

onde $s_{q, uv}$ é o tempo de preparação da máquina M_q para execução da tarefa J_v , que é diretamente precedida pela tarefa J_u na seqüência de execução das n tarefas.

Na expressão (4) tem-se $X_{u S_{uv}}^k = X_{u S_{uv}}^{k+1} = 0$, o que leva a

$$s_{k, uv} + Y_{u S_{uv}}^k = Y_{S_{tu} u}^k + p_{k+1, u}, \text{ e portanto, } Y_{u S_{uv}}^k = Y_{S_{tu} u}^k + p_{k+1, u} - s_{k, uv}. \quad (6)$$

Na expressão (6) é importante observar que $Y_{u S_{uv}}^k$ não necessariamente mantém a condição de ser positivo ou nulo. Nesse caso, a data de término da preparação S_{uv} na máquina M_k é maior que a data de término dessa preparação na máquina M_{k+1} , conforme ilustra a Figura 2. Ou seja, não há restrição de precedência entre as atividades de preparação de máquinas sucessivas para execução de uma tarefa qualquer. É importante observar também que o fato de $Y_{u S_{uv}}^k$ eventualmente ser negativo não elimina a validade das expressões (4) a (6). Isto pode ser constatado na Figura 2.

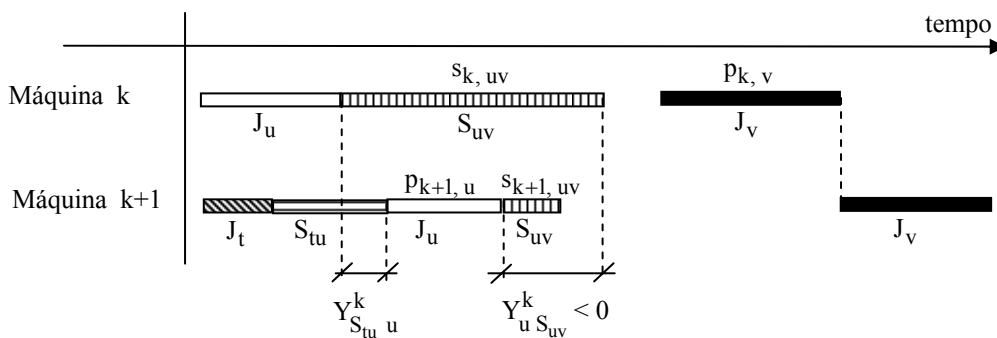


Figura 2 – Ilustração de uma situação em que $Y_{u S_{uv}}^k < 0$.

Substituindo $Y_{u S_{uv}}^k$ de (6) na expressão (5) tem-se

$$X_{S_{uv} v}^k + p_{k, v} + Y_{S_{uv} v}^k = Y_{S_{tu} u}^k + p_{k+1, u} - s_{k, uv} + s_{k+1, uv} + X_{S_{uv} v}^{k+1},$$

que leva a

$$X_{S_{uv} v}^k + (p_{k, v} + s_{k, uv}) + Y_{S_{uv} v}^k = Y_{S_{tu} u}^k + (p_{k+1, u} + s_{k+1, uv}) + X_{S_{uv} v}^{k+1}. \quad (7)$$

A expressão (7) é semelhante à tradicional expressão (1) que viabiliza, para duas tarefas sucessivas J_u e J_v , as relações entre tempos de máquina parada e tempos de espera entre operações sucessivas de uma mesma tarefa, no problema em que os tempos de preparação das máquinas são incluídos nos tempos de processamento das tarefas. A diferença fundamental é que, na expressão (7) os tempos de máquina parada $X_{i j}^q$ referem-se à tarefa J_v e à preparação S_{uv} das máquinas para sua execução, enquanto que na expressão (1) tais tempos indicam o intervalo entre o término da tarefa J_u e o início de J_v . Note-se também que os valores de $X_{i j}^q$ e $Y_{i j}^q$ mantêm a condição de serem sempre positivos ou nulos. Na expressão (7), $X_{i j}^q$ é a medida do intervalo de tempo entre o término de preparação da máquina e o início da execução da tarefa para a qual foi preparada, e $Y_{i j}^q$ é o tempo de espera entre operações sucessivas de uma mesma tarefa, as quais apresentam entre si uma relação de precedência tecnológica direta.

Desta forma, pode-se completar a expressão (7) com as demais relações de viabilidade, conforme segue:

$$\text{Se } X_{S_{uv} v}^{k+1} > 0 \text{ então } Y_{S_{uv} v}^k = 0, \text{ e} \quad (8)$$

$$\text{Se } Y_{S_{uv} v}^k > 0 \text{ então } X_{S_{uv} v}^{k+1} = 0. \quad (9)$$

A Figura 3 ilustra tais relações de viabilidade.

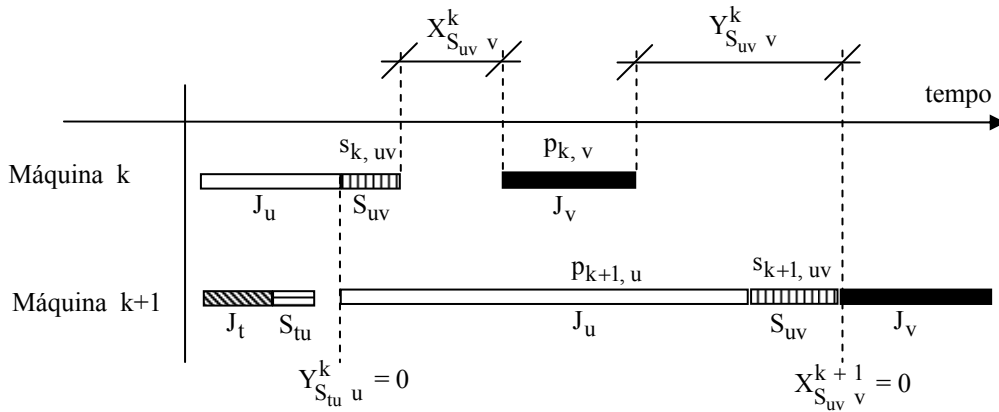


Figura 3 – Relações de viabilidade entre S_{uv} e J_v .

Seja $UBX_{S_{uv}v}^k$ um Limitante Superior de $X_{S_{uv}v}^k$, ou seja, $X_{S_{uv}v}^k \leq UBX_{S_{uv}v}^k$, para $k = 1, 2, \dots, m$.

Sendo $X_{S_{uv}v}^k \leq UBX_{S_{uv}v}^k$, adicionando-se $(p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv})$ em ambos os termos da desigualdade, tem-se:

$$X_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) \leq UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}). \quad (10)$$

A expressão de viabilidade (7) entre S_{uv} e J_v pode ser escrita como:

$$X_{S_{uv}v}^{k+1} - Y_{S_{uv}v}^k = X_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) - Y_{S_{tu}u}^k.$$

Uma vez que, por definição, $Y_{S_{tu}u}^k \geq 0$, tem-se:

$$X_{S_{uv}v}^{k+1} - Y_{S_{uv}v}^k \leq X_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}). \quad (11)$$

Utilizando-se as expressões (10) e (11) pode-se escrever que

$$X_{S_{uv}v}^{k+1} - Y_{S_{uv}v}^k \leq UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}). \quad (12)$$

A expressão (12) fornece um limitante superior para a diferença entre as variáveis $X_{S_{uv}v}^{k+1}$ e $Y_{S_{uv}v}^k$, as quais não podem ser simultaneamente positivas, conforme as expressões (8) e (9). Desta forma, pode-se enunciar a seguinte proposição:

Considere-se o problema de programação da produção *flow shop* permutacional com os tempos de preparação das máquinas separados dos tempos de processamento das tarefas, dependentes ou não da seqüência de execução das tarefas. Sejam J_u e J_v duas tarefas sucessivas quaisquer, e S_{uv} a atividade de preparação das máquinas para execução de J_v . Seja $UBX_{S_{uv}v}^k$ um limitante superior de $X_{S_{uv}v}^k$, intervalo de tempo em M_k entre o término da preparação S_{uv} e o início de J_v . Analogamente, seja $UBX_{S_{uv}v}^{k+1}$ um limitante superior de $X_{S_{uv}v}^{k+1}$.

a) Se $UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) > 0$

então $X_{S_{uv}v}^{k+1} \leq UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv})$ e

$$UBX_{S_{uv}v}^{k+1} = UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}).$$

b) Caso contrário, se $UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) \leq 0$

então $X_{S_{uv}v}^{k+1} = 0$ e $UBX_{S_{uv}v}^{k+1} = 0$.

Prova da proposição:

a) Por definição $X_{S_{uv}v}^{k+1} \geq 0$.

• Se $X_{S_{uv}v}^{k+1} = 0$ e $UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) > 0$,

obviamente $X_{S_{uv}v}^{k+1} < UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv})$.

• Se, por outro lado, $X_{S_{uv}v}^{k+1} > 0$, então pela expressão (8) $Y_{S_{uv}v}^k = 0$. Logo, utilizando-se

a expressão (12) tem-se $X_{S_{uv}v}^{k+1} \leq UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv})$.

Portanto, $UBX_{S_{uv}v}^{k+1} = UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv})$.

b) A condição $UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) \leq 0$ leva, a partir da expressão

(12), a $X_{S_{uv}v}^{k+1} - Y_{S_{uv}v}^k \leq 0$.

Tendo em vista as relações de viabilidade (8) e (9), $X_{S_{uv}v}^{k+1}$ e $Y_{S_{uv}v}^k$ são valores positivos ou nulos, mas que não podem ser simultaneamente positivos. Desta forma, a desigualdade

$X_{S_{uv}v}^{k+1} - Y_{S_{uv}v}^k \leq 0$ somente é válida se $X_{S_{uv}v}^{k+1} = 0$.

Portanto, $UBX_{S_{uv}v}^{k+1} = 0$.

Com fundamento na proposição acima, pode-se estabelecer a seguinte propriedade:

No problema FSP-TPS com m máquinas e n tarefas, sendo J_u e J_v duas tarefas sucessivas quaisquer, S_{uv} a atividade de preparação da máquina M_{k+1} para execução de J_v , e $X_{S_{uv}v}^{k+1}$ o intervalo de tempo em M_{k+1} entre o término da preparação S_{uv} e o início de J_v , então $UBX_{S_{uv}v}^{k+1}$ é um Limitante Superior de $X_{S_{uv}v}^{k+1}$, dado por:

$$UBX_{S_{uv}v}^{k+1} = \text{Max} \left[0, UBX_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) \right] \quad (13)$$

$$k = 1, 2, \dots, m-1 \quad \text{com} \quad UBX_{S_{uv}v}^1 = 0.$$

A expressão (13) permite calcular o limitante superior $UBX_{S_{uv}v}^{k+1}$ para uma máquina qualquer, de maneira recorrente, a partir de $UBX_{S_{uv}v}^1 = 0$, uma vez que na primeira máquina os intervalos de tempo $X_{S_{uv}v}^1$ são nulos.

3. Aplicação da Propriedade

Uma aplicação da propriedade apresentada é descrita a seguir.

Considerando uma seqüência qualquer σ das n tarefas, a Duração Total da Programação (*makespan*) pode ser calculada pela seguinte expressão:

$$M(\sigma) = \sum_{j=1}^n p_{m[j]} + \sum_{j=0}^{n-1} s_{m[j+1]} + \sum_{j=0}^{n-1} X_{[j+1]}^m \quad (14)$$

onde

$p_{m[j]}$ = tempo de processamento, na última máquina, da tarefa que ocupa a j -ésima posição na seqüência σ ;

$s_{m[j+1]}$ = tempo de preparação da última máquina para execução da tarefa que ocupa a posição $(j+1)$ na seqüência σ ;

$X_{[j+1]}^m$ = intervalo de tempo, na última máquina, entre o término de sua preparação e o início da execução da tarefa que ocupa a posição $(j+1)$ na seqüência σ ; sendo que

$j = 0$ corresponde a uma tarefa fictícia com tempos de processamento e preparação nulos, e que ocupa sempre a posição anterior à primeira tarefa em qualquer seqüência das n tarefas.

A partir das expressões (13) e (14) pode-se calcular um Limitante Superior do *makespan* $M(\sigma)$, conforme segue:

$$UBM(\sigma) = \sum_{j=1}^n p_{m[j]} + \sum_{j=0}^{n-1} s_{m[j+1]} + \sum_{j=0}^{n-1} UBX_{[j+1]}^m. \quad (15)$$

O primeiro termo da direita na expressão (15) é constante e independente da seqüência σ . Para o caso em que os tempos de preparação das máquinas são independentes, o segundo termo também será constante e quando os tempos de preparação das máquinas forem dependentes da seqüência das tarefas, tanto o segundo quanto o terceiro termo dependem da seqüência considerada.

Entretanto, para ambas as situações o problema original de programação da produção, quando se deseja minimizar o *makespan*, pode ser resolvido heurísticamente por meio da solução de um problema transformado, no qual objetiva-se determinar a seqüência das tarefas que minimiza o limitante superior $UBM(\sigma)$.

Tal problema transformado é análogo ao problema cíclico assimétrico do caixeiro-viajante (*ATSP*) ilustrado na Figura 4.

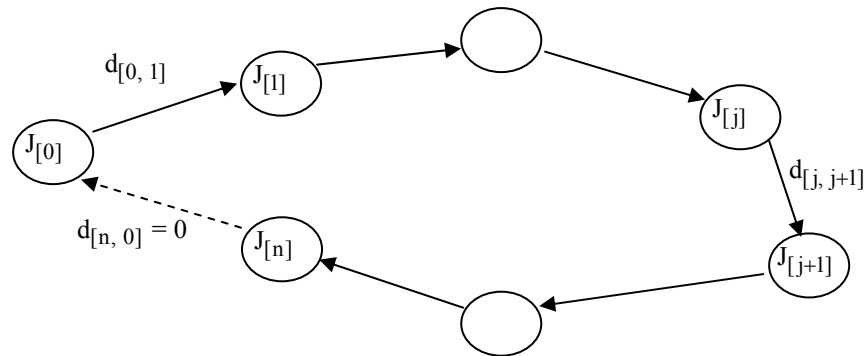


Figura 4 – Analogia do problema original de programação da produção com o problema cíclico assimétrico do caixeiro-viajante.

No problema ilustrado pela Figura 4, a cidade-base seria a tarefa fictícia $J_{[0]}$, as cidades a serem visitadas seriam as n tarefas, e uma rota viável seria a seqüência $J_{[0]}, J_{[1]}, \dots, J_{[j]}, J_{[j+1]}, \dots, J_{[n]}, J_{[0]}$, onde $J_{[j]}$ representa a tarefa que ocupa a j -ésima posição na seqüência σ .

No caso de tempos de preparação das máquinas dependentes da seqüência das tarefas, a distância entre duas cidades (tarefas) $d_{[j, j+1]}$ seria a soma $(s_{m[j+1]} + UBX_{[j+1]}^m)$, para $j = 0, 1, 2, \dots, (n-1)$. Se os tempos de preparação das máquinas forem independentes da seqüência das tarefas, tal distância seria somente o limitante superior $UBX_{[j+1]}^m$.

A modelagem de um problema de programação da produção *flow shop* por meio de um problema do caixeiro-viajante tem ocorrido frequentemente.

Framinan, Gupta & Leisten (2004) apresentam uma revisão e classificação de métodos heurísticos para o problema clássico de programação da produção *flow shop* permutacional (com os tempos de preparação das máquinas incluídos nos tempos de processamento das tarefas) tendo como medida de desempenho a minimização do *makespan*. Nesse artigo, são citados quatro trabalhos que buscam resolver heurísticamente tal problema a partir da solução de um problema de caixeiro-viajante: Gupta (1968), Stinson & Smith (1982), Widmer & Hertz (1989), e Moccellin (1995). Uma diferença básica entre esses trabalhos consiste na forma pela qual são calculadas as distâncias entre as operações sucessivas das tarefas na última máquina do *flow shop*. A analogia proposta neste trabalho entre o problema de programação da produção FSP-TPS e do caixeiro-viajante (ATSP) é similar àquela apresentada em Moccellin (1995).

Conforme já mencionado, os métodos *SETUP* e *TOTAL* (Simons Jr., 1992), e *HYBRID* (Rios-Mercado & Bard, 1999b) fundamentam-se no problema assimétrico do caixeiro-viajante para obter soluções heurísticas do problema FSP-TPS com tempos de preparação das máquinas dependentes da seqüência de execução das tarefas.

Os trabalhos acima citados, que utilizam analogia com o problema do caixeiro-viajante fazem parte de uma classe de problemas *flow shop* com *buffer* não-limitado, uma vez que as tarefas não apresentam restrições de armazenagem entre as suas operações sucessivas nas diversas máquinas. No caso de *buffer* limitado existe um número máximo de tarefas que podem ser armazenadas aguardando os respectivos processamentos nas máquinas. Outras variações do ambiente *flow shop* com restrições quanto ao processamento das tarefas denominam-se *blocking* e *no-wait flow shop*. No primeiro caso, não existe a possibilidade de armazenar as tarefas entre as suas operações sucessivas nas máquinas, de forma que uma tarefa que já tenha sido processada em uma determinada máquina não pode liberá-la, portanto bloqueando-a, até que a máquina da próxima operação esteja disponível. No segundo caso, uma tarefa uma vez iniciada não pode ser interrompida até o seu completo processamento, ou seja, não deve existir tempo de espera entre as suas operações sucessivas. Em uma recente revisão da literatura, Bagchi, Gupta & Sriskandarajah (2006) tratam desses ambientes de produção específicos, além de outros como o *flow shop* flexível e o automatizado, apresentando diversos trabalhos e formulações desses problemas de produção *flow shop* baseados em analogias com o problema do caixeiro-viajante.

4. Análise Estatística da analogia entre os problemas FSP-TPS e ATSP

A partir dos fundamentos apresentados em Mood, Graybill & Boes (1974), pode-se efetuar a análise estatística apresentada a seguir.

A analogia entre os problemas FSP-TPS e ATSP pode ser considerada adequada se os espaços de soluções dos mesmos forem equivalentes, dentro de uma faixa aceitável de aproximação. Isto significa que, se os espaços forem totalmente equivalentes, as soluções ótimas dos dois problemas também serão, ou seja, a seqüência de tarefas que minimiza o *makespan* (função-objetivo do problema original de programação da produção) é a mesma seqüência que minimiza o limitante superior $UBM(\sigma)$ (função-objetivo do problema cíclico assimétrico do caixeiro-viajante).

A condição de total equivalência pode ser estabelecida conforme segue:

Seja $DR(\sigma)$ o Desvio Relativo entre o limitante superior $UBM(\sigma)$ e o *makespan* $M(\sigma)$ de uma seqüência genérica σ , dado por:

$$DR(\sigma) = \frac{UBM(\sigma) - M(\sigma)}{M(\sigma)} \quad \text{que leva a} \quad UBM(\sigma) = (1 + DR(\sigma)) M(\sigma). \quad (16)$$

Assim, a total equivalência é condicionada a $DR(\sigma)$ assumir o mesmo valor (constante) para todas as seqüências possíveis das n tarefas, ou seja, $(n!)$ seqüências.

Obviamente, tal condição de total equivalência não é satisfeita, uma vez que os Desvios Relativos podem assumir valores diferentes. Por outro lado, uma equivalência aproximada pode ser obtida substituindo o valor $DR(\sigma)$ por uma estimativa que incorpora a sua variabilidade, por meio da expressão que segue:

$$\text{Estimativa } [DR(\sigma)] = \overline{DR} \pm S_{DR}, \text{ onde} \quad (17)$$

\overline{DR} = Desvio Relativo Médio considerando o espaço de soluções constituído das $(n!)$ seqüências, calculado por $\overline{DR} = \frac{1}{n!} \sum_{i=1}^{n!} DR(\sigma)_i$;

$$S_{DR} = \text{Desvio Padrão do desvio relativo } DR(\sigma), \text{ dado por } S_{DR} = \sqrt{\frac{\sum_{i=1}^{n!} (DR(\sigma)_i - \overline{DR})^2}{n! - 1}}.$$

Utilizando a expressão (17) em (16) obtém-se uma relação aproximada entre o limitante superior $UBM(\sigma)$ e o *makespan* $M(\sigma)$, ou seja:

$$\text{Estimativa } [UBM(\sigma)] = (1 + \overline{DR} \pm S_{DR}) M(\sigma). \quad (18)$$

A expressão (18) pode ser entendida como uma estatística que reflete o “grau de equivalência” entre os espaços de soluções dos dois problemas (programação da produção e caixeiro-viajante). É importante notar que esse grau de equivalência depende somente da amplitude do desvio padrão S_{DR} . Quanto menor o desvio padrão, maior o grau de equivalência. $S_{DR} = 0$ leva a $\overline{DR} = DR(\sigma)_i$ para todo $i = 1, 2, \dots, n!$, ou seja, um Desvio Relativo $DR(\sigma)$ constante, que seria a condição de total equivalência entre os espaços de soluções.

4.1 Um Limitante Superior do Desvio Padrão S_{DR}

Com o objetivo de avaliar a influência dos parâmetros do problema de programação da produção (número de máquinas, número de tarefas, tempos de processamento das tarefas e tempos de preparação das máquinas), foi deduzida uma expressão que fornece um limitante superior do Desvio Padrão S_{DR} em função dos mencionados parâmetros.

Para tanto, considere-se o problema FSP-TPS com m máquinas e n tarefas, e duas variáveis aleatórias independentes P e S , sendo

P = Variável aleatória dos tempos de processamento das tarefas, com média $\mu[P]$ e variância $\text{var}[P]$;

S = Variável aleatória dos tempos de preparação das máquinas, com média $\mu[S]$ e variância $\text{var}[S]$.

Sejam $\text{MIN}[P]$ e $\text{MAX}[P]$ o valor mínimo e máximo, respectivamente, dos tempos de processamento das tarefas. Analogamente, $\text{MIN}[S]$ e $\text{MAX}[S]$ para os tempos de preparação das máquinas.

Desta forma, o *makespan* $M(\sigma)$ e seu limitante superior $\text{UBM}(\sigma)$, os intervalos de tempo $X_{S_{uv}v}^{k+1}$ e os limitantes superiores $\text{UBX}_{S_{uv}v}^{k+1}$, assim como o Desvio Relativo $\text{DR}(\sigma)$, serão variáveis aleatórias dependentes de P e S .

4.1.1 Variável $\text{UBX}_{S_{uv}v}^{k+1}$

Conforme a expressão (13),

$$\text{UBX}_{S_{uv}v}^{k+1} = \text{Max} \left[0, \text{UBX}_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) \right]$$

$$k = 1, 2, \dots, m-1 \quad \text{com} \quad \text{UBX}_{S_{uv}v}^1 = 0.$$

$$\text{Portanto, para } k=1 \text{ tem-se: } \text{UBX}_{S_{uv}v}^2 = \text{Max} \left[0, (p_{1,v} + s_{1,uv}) - (p_{2,u} + s_{2,uv}) \right]$$

onde $\Delta_1(P+S) = (p_{1,v} + s_{1,uv}) - (p_{2,u} + s_{2,uv})$ é uma variável aleatória (v.a.) com

$$\text{média } \mu[\Delta_1(P+S)] = (\mu[P] + \mu[S]) - (\mu[P] + \mu[S]) = 0, \text{ e}$$

$$\text{variância } \text{var}[\Delta_1(P+S)] = (\text{var}[P] + \text{var}[S]) + (\text{var}[P] + \text{var}[S]) = 2(\text{var}[P] + \text{var}[S]).$$

A v.a. $\Delta_1(P+S)$ distribui-se simetricamente em relação ao valor zero, assumindo valores no intervalo $\text{MIN}[\Delta_1(P+S)]$ a $\text{MAX}[\Delta_1(P+S)]$, onde

$$\text{MIN}[\Delta_1(P+S)] = (\text{MIN}[P] + \text{MIN}[S]) - (\text{MAX}[P] + \text{MAX}[S]) = \text{MIN}[P+S] - \text{MAX}[P+S],$$

$$\text{e } \text{MAX}[\Delta_1(P+S)] = \text{MAX}[P+S] - \text{MIN}[P+S].$$

Assim, $\text{UBX}_{S_{uv}v}^2 = \text{Max} [0, \Delta_1(P+S)]$ será uma v.a. com as seguintes características:

- Assume valores no intervalo $[0, \text{MAX}[\Delta_1(P+S)]]$;
- $0 < \mu[\text{UBX}_{S_{uv}v}^2] < \text{MAX}[\Delta_1(P+S)]$, portanto $\text{MAX}[P+S] - \text{MIN}[P+S]$ é um limitante superior da média de $\text{UBX}_{S_{uv}v}^2$;

- $\text{var}\left[\text{UBX}_{S_{uv}v}^2\right] < \text{var}\left[\Delta_1(P+S)\right]$, portanto $2(\text{var}[P] + \text{var}[S])$ é um limitante superior da variância de $\text{UBX}_{S_{uv}v}^2$.

Para $k=2$, $\text{UBX}_{S_{uv}v}^3 = \text{Max}\left[0, \text{UBX}_{S_{uv}v}^2 + (p_{2,v} + s_{2,uv}) - (p_{3,u} + s_{3,uv})\right]$

onde $\Delta_2(P+S) = (p_{2,v} + s_{2,uv}) - (p_{3,u} + s_{3,uv})$ é uma v.a. com a mesma distribuição de probabilidades de $\Delta_1(P+S)$. Convém observar que $\Delta_1(P+S)$ e $\Delta_2(P+S)$ são variáveis aleatórias independentes, assim como $\text{UBX}_{S_{uv}v}^2$ e $\Delta_2(P+S)$.

Analogamente a $\text{UBX}_{S_{uv}v}^2$, a v.a. $\text{UBX}_{S_{uv}v}^3 = \text{Max}\left[0, \text{UBX}_{S_{uv}v}^2 + \Delta_2(P+S)\right]$ apresenta as características que seguem:

- Assume valores no intervalo $\left[0, \text{MAX}\left[\text{UBX}_{S_{uv}v}^3\right]\right]$, onde $\text{MAX}\left[\text{UBX}_{S_{uv}v}^3\right] = \text{MAX}\left[\text{UBX}_{S_{uv}v}^2\right] + \text{MAX}\left[\Delta_2(P+S)\right]$, ou seja, $\text{MAX}\left[\text{UBX}_{S_{uv}v}^3\right] = 2(\text{MAX}[P+S] - \text{MIN}[P+S])$;
- $0 < \mu\left[\text{UBX}_{S_{uv}v}^3\right] < \text{MAX}\left[\text{UBX}_{S_{uv}v}^3\right]$, portanto $2(\text{MAX}[P+S] - \text{MIN}[P+S])$ é um limitante superior da média de $\text{UBX}_{S_{uv}v}^3$;
- $\text{var}\left[\text{UBX}_{S_{uv}v}^3\right] < \text{var}\left[\text{UBX}_{S_{uv}v}^2\right] + \text{var}\left[\Delta_2(P+S)\right] < \text{var}\left[\Delta_1(P+S)\right] + \text{var}\left[\Delta_2(P+S)\right]$, portanto $4(\text{var}[P] + \text{var}[S])$ é um limitante superior da variância de $\text{UBX}_{S_{uv}v}^3$.

Seguindo um procedimento análogo, pode-se estabelecer os seguintes limitantes superiores para a média e variância de $\text{UBX}_{S_{uv}v}^{k+1}$ ($k = 1, 2, \dots, m-1$):

$$0 < \mu\left[\text{UBX}_{S_{uv}v}^{k+1}\right] < k(\text{MAX}[P+S] - \text{MIN}[P+S]); \tag{19}$$

$$\text{var}\left[\text{UBX}_{S_{uv}v}^{k+1}\right] < 2k(\text{var}[P] + \text{var}[S]), \tag{20}$$

onde J_u e J_v são duas tarefas sucessivas quaisquer.

4.1.2 Variável $UBM(\sigma)$

Conforme a expressão (15),

$$UBM(\sigma) = \sum_{j=1}^n p_{m[j]} + \sum_{j=0}^{n-1} s_{m[j+1]} + \sum_{j=0}^{n-1} UB X_{[j+1]}^m, \text{ lembrando que}$$

$p_{m[j]}$ = tempo de processamento, na última máquina, da tarefa que ocupa a j -ésima posição na seqüência σ ;

$s_{m[j+1]}$ = tempo de preparação da última máquina para execução da tarefa que ocupa a posição $(j+1)$ na seqüência σ ;

$X_{[j+1]}^m$ = intervalo de tempo, na última máquina, entre o término de sua preparação e o início da execução da tarefa que ocupa a posição $(j+1)$ na seqüência σ ; sendo que $j = 0$ corresponde a uma tarefa fictícia com tempos de processamento e preparação nulos, e que ocupa sempre a posição anterior à primeira tarefa em qualquer seqüência das n tarefas.

Tendo-se que $p_{m[0]} = 0$, a expressão (15) pode ser escrita como:

$$UBM(\sigma) = p_{m[0]} + \sum_{j=1}^n p_{m[j]} + \sum_{j=0}^{n-1} s_{m[j+1]} + \sum_{j=0}^{n-1} UB X_{[j+1]}^m, \text{ ou seja,}$$

$$UBM(\sigma) = \sum_{j=0}^{n-1} p_{m[j]} + p_{m[n]} + \sum_{j=0}^{n-1} s_{m[j+1]} + \sum_{j=0}^{n-1} UB X_{[j+1]}^m, \text{ que leva a}$$

$$UBM(\sigma) = \sum_{j=0}^{n-1} (p_{m[j]} + s_{m[j+1]}) + p_{m[n]} + \sum_{j=0}^{n-1} UB X_{[j+1]}^m. \quad (21)$$

Da teoria estatística, considerando Z_1, Z_2, \dots, Z_N um conjunto de N variáveis aleatórias, tem-se:

$$\text{a média } \mu \left[\sum_{i=1}^N Z_i \right] = \sum_{i=1}^N \mu[Z_i] \quad (22)$$

$$\text{e a variância } \text{var} \left[\sum_{i=1}^N Z_i \right] = \sum_{i=1}^N \text{var}[Z_i] + 2 \sum_{i < j} \text{cov}[Z_i, Z_j]. \quad (23)$$

Média de $UBM(\sigma)$

A partir das expressões (21) e (22),

$$\mu[UBM(\sigma)] = \sum_{j=0}^{n-1} \mu[p_{m[j]} + s_{m[j+1]}] + \mu[p_{m[n]}] + \sum_{j=0}^{n-1} \mu[UB X_{[j+1]}^m].$$

Uma vez que $\mu[p_{m[0]}] = 0$, $\mu[s_{m[1]}] = \mu[S]$, $\mu[p_{m[n]}] = \mu[P]$ e

$\mu[p_{m[j]} + s_{m[j+1]}] = \mu[p_{m[j]}] + \mu[s_{m[j+1]}] = \mu[P] + \mu[S] \quad \forall j = 1, \dots, n-1$, pode-se escrever

$$\mu[\text{UBM}(\sigma)] = n(\mu[P] + \mu[S]) + \sum_{j=0}^{n-1} \mu[\text{UBX}_{[j+1]}^m]. \quad (24)$$

A expressão (19) fornece um limitante superior da média $\mu[\text{UBX}_{S_{uv}v}^{k+1}]$ para todo $k = 1, 2, \dots, m-1$, e qualquer par (J_u, J_v) de tarefas sucessivas. Portanto, para a última máquina m ,

$$0 < \mu[\text{UBX}_{[j+1]}^m] < (m-1)(\text{MAX}[P+S] - \text{MIN}[P+S]) \quad \forall j = 0, 1, \dots, n-1. \quad (25)$$

A partir das expressões (24) e (25) pode-se estabelecer os seguintes limitantes para a média $\mu[\text{UBM}(\sigma)]$:

Limitante superior

$$\text{LS}(\mu[\text{UBM}(\sigma)]) = n(\mu[P] + \mu[S] + (m-1)(\text{MAX}[P+S] - \text{MIN}[P+S])). \quad (26)$$

Limitante inferior

$$\text{LI}(\mu[\text{UBM}(\sigma)]) = n(\mu[P] + \mu[S]). \quad (27)$$

Variância de $\text{UBM}(\sigma)$

Antes de se apresentar uma expressão para a $\text{var}[\text{UBM}(\sigma)]$, deve-se verificar as eventuais interdependências entre as variáveis aleatórias da expressão (21).

As variáveis aleatórias P (tempo de processamento das tarefas) e S (tempo de preparação das máquinas) são independentes, portanto as variáveis $(p_{m[j]} + s_{m[j+1]})$ para $j = 0, 1, \dots, n-1$, e $p_{m[n]}$ são independentes entre si.

A expressão (13),

$$\text{UBX}_{S_{uv}v}^{k+1} = \text{Max} \left[0, \text{UBX}_{S_{uv}v}^k + (p_{k,v} + s_{k,uv}) - (p_{k+1,u} + s_{k+1,uv}) \right]$$

pode ser escrita, de maneira equivalente, como

$$\text{UBX}_{[j+1]}^{k+1} = \text{Max} \left[0, \text{UBX}_{[j+1]}^k + (p_{k,[j+1]} + s_{k,[j+1]}) - (p_{k+1,[j]} + s_{k+1,[j+1]}) \right]$$

$$k = 1, 2, \dots, m-1 \quad \text{com} \quad \text{UBX}_{[j+1]}^1 = 0, \quad \text{para } j = 0, 1, \dots, n-1.$$

Assim, na última máquina m , tem-se

$$\text{UBX}_{[j+1]}^m = \text{Max} \left[0, \text{UBX}_{[j+1]}^{m-1} + (p_{m-1,[j+1]} + s_{m-1,[j+1]}) - (p_{m,[j]} + s_{m,[j+1]}) \right].$$

Portanto, pode-se notar que os n pares de variáveis $((p_{m,[j]} + s_{m,[j+1]}), UBX_{[j+1]}^m)$ para $j = 0, 1, \dots, n-1$, são dependentes, porém com correlação negativa. Desta forma, a covariância dessas variáveis é negativa. Os demais pares, ou seja, $((p_{m,[i]} + s_{m,[i+1]}), UBX_{[i+1]}^m)$ para $i = 0, 1, \dots, n-1$, e $j = 0, 1, \dots, n-1$, com $i \neq j$, são variáveis aleatórias independentes.

Para concluir a verificação de interdependência entre as variáveis que fornecem o valor de $UBM(\sigma)$, resta verificar as eventuais dependências no conjunto de n variáveis $UBX_{[j+1]}^m$, $j = 0, 1, \dots, n-1$. Para tanto, considere-se que, em uma seqüência genérica σ as tarefas J_r , J_t , J_u e J_v são sucessivas.

A expressão (13) para o par de tarefas sucessivas (J_r, J_t) fica:

$$UBX_{S_{rt}}^{k+1} = \text{Max} \left[0, UBX_{S_{rt}}^k + (p_{k,t} + s_{k,rt}) - (p_{k+1,r} + s_{k+1,rt}) \right].$$

Para $k = 1$,

$$UBX_{S_{rt}}^2 = \text{Max} \left[0, UBX_{S_{rt}}^1 + (p_{1,t} + s_{1,rt}) - (p_{2,r} + s_{2,rt}) \right] \text{ com } UBX_{S_{rt}}^1 = 0, \text{ portanto}$$

$$UBX_{S_{rt}}^2 = \text{Max} \left[0, (p_{1,t} + s_{1,rt}) - (p_{2,r} + s_{2,rt}) \right].$$

Para $k = 2$,

$$UBX_{S_{rt}}^3 = \text{Max} \left[0, UBX_{S_{rt}}^2 + (p_{2,t} + s_{2,rt}) - (p_{3,r} + s_{3,rt}) \right], \text{ que leva a}$$

$$UBX_{S_{rt}}^3 = \text{Max} \left[0, \text{Max} \left[0, (p_{1,t} + s_{1,rt}) - (p_{2,r} + s_{2,rt}) \right] + (p_{2,t} + s_{2,rt}) - (p_{3,r} + s_{3,rt}) \right].$$

Observa-se que as variáveis $(p_{1,t} + s_{1,rt})$ e $(p_{2,t} + s_{2,rt})$ contribuem para aumentar o valor de $UBX_{S_{rt}}^3$, enquanto que $(p_{2,r} + s_{2,rt})$ e $(p_{3,r} + s_{3,rt})$ contribuem para diminuir tal valor. Assim, pode-se escrever que:

$$UBX_{S_{rt}}^3 = \text{função} \left[+ (p_{q,t} + s_{q,rt}); - (p_{q+1,r} + s_{q+1,rt}) \quad q = 1, 2 \right].$$

Repetindo o procedimento para $k = 3, 4, \dots, m-1$, tem-se que na última máquina m ,

$$UBX_{S_{rt}}^m = \text{função} \left[+ (p_{q,t} + s_{q,rt}); - (p_{q+1,r} + s_{q+1,rt}) \quad q = 1, 2, \dots, m-1 \right]. \quad (28)$$

Desta forma, para os pares de tarefas sucessivas (J_t, J_u) e (J_u, J_v) pode-se escrever:

$$UBX_{S_{tu}}^m = \text{função} \left[+ (p_{q,u} + s_{q,tu}); - (p_{q+1,t} + s_{q+1,tu}) \quad q = 1, 2, \dots, m-1 \right] \text{ e} \quad (29)$$

$$UBX_{S_{uv}}^m = \text{função} \left[+ (p_{q,v} + s_{q,uv}); - (p_{q+1,u} + s_{q+1,uv}) \quad q = 1, 2, \dots, m-1 \right]. \quad (30)$$

Nas expressões (28) e (29) nota-se que $UBX_{S_{rt}t}^m$ e $UBX_{S_{tu}u}^m$ são dependentes do mesmo conjunto de variáveis $p_{q,t}$ para $q = 2, 3, \dots, m-1$. Isso indica que tais variáveis aleatórias são dependentes, porém apresentando correlação negativa, uma vez que $p_{q,t}$ ($q = 2, 3, \dots, m-1$) contribui para aumentar $UBX_{S_{rt}t}^m$ e simultaneamente para diminuir $UBX_{S_{tu}u}^m$. O mesmo acontece com $UBX_{S_{tu}u}^m$ e $UBX_{S_{uv}v}^m$, expressões (29) e (30), tendo em vista a dependência simultânea com as variáveis $p_{q,u}$ ($q = 2, 3, \dots, m-1$). Entretanto, as variáveis $UBX_{S_{rt}t}^m$ e $UBX_{S_{uv}v}^m$ são independentes, uma vez que os tempos de processamento das tarefas J_r , J_t , J_u e J_v , e os tempos de preparação das máquinas para execução das tarefas J_t e J_v são variáveis aleatórias independentes.

Desta forma, conclui-se que as n variáveis $UBX_{[j+1]}^m$, $j = 0, 1, \dots, n-1$, apresentam dependência somente nos pares sucessivos $(UBX_{[j+1]}^m, UBX_{[j+2]}^m)$, $j = 0, 1, \dots, n-2$, com covariância negativa.

Utilizando a expressão (23) e as relações de dependência entre as variáveis que fornecem o valor de $UBM(\sigma)$ na expressão (21), pode-se escrever que:

$$\begin{aligned} \text{var}[UBM(\sigma)] &= \sum_{j=0}^{n-1} \text{var}[p_{m[j]} + s_{m[j+1]}] + \text{var}[p_{m[n]}] + \sum_{j=0}^{n-1} \text{var}[UBX_{[j+1]}^m] + \\ &+ 2 \sum_{j=0}^{n-1} \text{cov}[(p_{m[j]} + s_{m[j+1]}), UBX_{[j+1]}^m] + 2 \sum_{j=0}^{n-2} \text{cov}[UBX_{[j+1]}^m, UBX_{[j+2]}^m]. \end{aligned}$$

As covariâncias da expressão acima são negativas, portanto

$$\text{var}[UBM(\sigma)] < \sum_{j=0}^{n-1} \text{var}[p_{m[j]} + s_{m[j+1]}] + \text{var}[p_{m[n]}] + \sum_{j=0}^{n-1} \text{var}[UBX_{[j+1]}^m]. \quad (31)$$

A expressão (20) fornece um limitante superior da variância $\text{var}[UBX_{S_{uv}v}^{k+1}]$ para $k = 1, 2, \dots, m-1$, e qualquer par (J_u, J_v) de tarefas sucessivas. Portanto, para a última máquina m ,

$$\text{var}[UBX_{[j+1]}^m] < 2(m-1)(\text{var}[P] + \text{var}[S]) \quad \forall j=0, 1, \dots, n-1. \quad (32)$$

Com a expressão (32) e sabendo-se que

$$\text{var}[p_{m[0]}] = 0, \quad \text{var}[s_{m[1]}] = \text{var}[S], \quad \text{var}[p_{m[n]}] = \text{var}[P], \quad \text{e}$$

$$\text{var}[p_{m[j]} + s_{m[j+1]}] = \text{var}[p_{m[j]}] + \text{var}[s_{m[j+1]}] = \text{var}[P] + \text{var}[S] \quad \forall j=1, \dots, n-1, \text{ obtém-se:}$$

$$\begin{aligned} \text{var}[\text{UBM}(\sigma)] &< n(\text{var}[P] + \text{var}[S]) + 2n(m-1)(\text{var}[P] + \text{var}[S]), \text{ ou seja,} \\ \text{var}[\text{UBM}(\sigma)] &< n(2m-1)(\text{var}[P] + \text{var}[S]). \end{aligned} \quad (33)$$

Assim, a expressão (33) fornece um limitante superior da variância de $\text{UBM}(\sigma)$, conforme segue:

$$\text{LS}(\text{var}[\text{UBM}(\sigma)]) = n(2m-1)(\text{var}[P] + \text{var}[S]). \quad (34)$$

4.1.3 Variável $M(\sigma)$

Por definição,

$$M(\sigma) = \sum_{j=1}^n p_{m[j]} + \sum_{j=0}^{n-1} s_{m[j+1]} + \sum_{j=0}^{n-1} X_{[j+1]}^m \leq \text{UBM}(\sigma) = \sum_{j=1}^n p_{m[j]} + \sum_{j=0}^{n-1} s_{m[j+1]} + \sum_{j=0}^{n-1} \text{UBX}_{[j+1]}^m.$$

Desta forma, os limitantes associados à variável $\text{UBM}(\sigma)$ podem ser adotados como limitantes do *makespan* $M(\sigma)$.

4.1.4 Variável Desvio Relativo $\text{DR}(\sigma)$

Por definição,

$$\text{DR}(\sigma) = \frac{\text{UBM}(\sigma) - M(\sigma)}{M(\sigma)}, \text{ ou seja, } \text{DR}(\sigma) = \frac{\text{UBM}(\sigma)}{M(\sigma)} - 1.$$

Assim,

$$\text{var}[\text{DR}(\sigma)] = \text{var}\left[\frac{\text{UBM}(\sigma)}{M(\sigma)}\right]. \quad (35)$$

Sabe-se que se Z e W forem duas variáveis aleatórias, então a variância $\text{var}\left[\frac{Z}{W}\right]$ é aproximadamente calculada pela expressão:

$$\text{var}\left[\frac{Z}{W}\right] \approx \left(\frac{\mu[Z]}{\mu[W]}\right)^2 \left(\frac{\text{var}[Z]}{\mu[Z]^2} + \frac{\text{var}[W]}{\mu[W]^2} - \frac{2\text{cov}[Z, W]}{\mu[Z]\mu[W]}\right).$$

Fazendo $\text{UBM}(\sigma) = Z$ e $M(\sigma) = W$, obtém-se

$$\text{var}\left[\frac{\text{UBM}(\sigma)}{M(\sigma)}\right] \approx \left(\frac{\mu[\text{UBM}(\sigma)]}{\mu[M(\sigma)]}\right)^2 \left(\frac{\text{var}[\text{UBM}(\sigma)]}{\mu[\text{UBM}(\sigma)]^2} + \frac{\text{var}[M(\sigma)]}{\mu[M(\sigma)]^2} - \frac{2\text{cov}[\text{UBM}(\sigma), M(\sigma)]}{\mu[\text{UBM}(\sigma)]\mu[M(\sigma)]}\right). \quad (36)$$

Pode-se notar, pelas expressões (14) e (15), que $UBM(\sigma)$ e $M(\sigma)$ são variáveis aleatórias dependentes e com correlação positiva, portanto apresentando covariância positiva. Assim, a partir da expressão (36) pode-se escrever que:

$$\text{var} \left[\frac{UBM(\sigma)}{M(\sigma)} \right] < \left(\frac{\mu[UBM(\sigma)]}{\mu[M(\sigma)]} \right)^2 \left(\frac{\text{var}[UBM(\sigma)]}{\mu[UBM(\sigma)]^2} + \frac{\text{var}[M(\sigma)]}{\mu[M(\sigma)]^2} \right). \quad (37)$$

Substituindo adequadamente na expressão (37) as médias e variâncias de $UBM(\sigma)$ e $M(\sigma)$ pelos correspondentes limitantes superiores e inferiores, tem-se que:

$$\text{var} \left[\frac{UBM(\sigma)}{M(\sigma)} \right] < \left(\frac{LS(\mu[UBM(\sigma)])}{LI(\mu[M(\sigma)])} \right)^2 \left(\frac{LS(\text{var}[UBM(\sigma)])}{LI(\mu[UBM(\sigma)]^2)} + \frac{LS(\text{var}[M(\sigma)])}{LI(\mu[M(\sigma)]^2)} \right), \text{ onde}$$

$$(26) \quad LS(\mu[UBM(\sigma)]) = n(\mu[P] + \mu[S] + (m-1)(MAX[P+S] - MIN[P+S]));$$

$$(27) \quad LI(\mu[UBM(\sigma)]) = n(\mu[P] + \mu[S]), \text{ sendo que } LI(\mu[M(\sigma)]) = LI(\mu[UBM(\sigma)]);$$

$$(34) \quad LS(\text{var}[UBM(\sigma)]) = n(2m-1)(\text{var}[P] + \text{var}[S]), \text{ sendo que}$$

$$LS(\text{var}[M(\sigma)]) = LS(\text{var}[UBM(\sigma)]).$$

Desta forma,

$$\text{var} \left[\frac{UBM(\sigma)}{M(\sigma)} \right] < \left(\frac{LS(\mu[UBM(\sigma)])}{LI(\mu[UBM(\sigma)])} \right)^2 \left(2 \frac{LS(\text{var}[UBM(\sigma)])}{LI(\mu[UBM(\sigma)]^2)} \right), \text{ ou seja,}$$

$$\text{var} \left[\frac{UBM(\sigma)}{M(\sigma)} \right] < 2 \frac{LS(\mu[UBM(\sigma)])^2}{LI(\mu[UBM(\sigma)]^4)} LS(\text{var}[UBM(\sigma)]), \text{ que leva a}$$

$$\text{var} \left[\frac{UBM(\sigma)}{M(\sigma)} \right] < 2n(2m-1)(\text{var}[P] + \text{var}[S]) \left(\frac{n^2(\mu[P] + \mu[S] + (m-1)(MAX[P+S] - MIN[P+S]))^2}{n^4(\mu[P] + \mu[S])^4} \right)$$

Portanto,

$$\text{var} \left[\frac{UBM(\sigma)}{M(\sigma)} \right] < \frac{2(2m-1)(\text{var}[P] + \text{var}[S]) (\mu[P] + \mu[S] + (m-1)(MAX[P+S] - MIN[P+S]))^2}{n(\mu[P] + \mu[S])^4}. \quad (38)$$

A expressão (38) fornece um limitante superior da $\text{var}[DR(\sigma)] = \text{var} \left[\frac{UBM(\sigma)}{M(\sigma)} \right]$.

Conseqüentemente, também fornece um limitante superior do desvio padrão S_{DR} dado por:

$$LS(S_{DR}) = \frac{(\mu[P] + \mu[S] + (m-1)(MAX[P+S] - MIN[P+S]))}{(\mu[P] + \mu[S])^2} \sqrt{\frac{2(2m-1)(\text{var}[P] + \text{var}[S])}{n}}. \quad (39)$$

Mesmo sem uma avaliação numérica, a expressão (39) possibilita inferir que:

- O grau de equivalência entre os espaços de soluções do problema de programação da produção e do caixeiro-viajante análogo tende a melhorar à medida que aumenta o porte do problema em termos do número n de tarefas;
- Entretanto, esse grau deve diminuir para um maior número m de máquinas;
- A adequação da analogia entre os mencionados problemas e, portanto o grau de equivalência entre os espaços de soluções apresenta uma tendência de ser melhor quanto maiores forem os tempos P de processamento das tarefas juntamente com os tempos S de preparação das máquinas, mais especificamente, quanto maior for o valor de $(\mu[P] + \mu[S])$.

4.2 Experimentação Computacional

A avaliação numérica do grau de equivalência entre os espaços de soluções do problema de programação da produção e do problema análogo do caixeiro-viajante foi efetuada por meio de uma experimentação computacional descrita a seguir.

Considerou-se um conjunto de problemas com o número de máquinas $m \in \{4, 7\}$ e o número de tarefas $n \in \{10, 20, 30, 40, 50, 60, 70, 80, 90\}$. Foram utilizadas seis relações para os tempos de processamento das tarefas e os tempos de preparação das máquinas conforme indica a Tabela 1. As relações foram definidas com base em trabalhos reportados na literatura, tais como Simons Jr. (1992), Das, Gupta & Khumawala (1995), Rajendran & Ziegler (1997), Rios-Mercado & Bard (1998, 1999b), e Weng, Lu & Ren (2001). Os tempos de processamento (P) e os tempos de preparação (S) foram considerados como variáveis aleatórias independentes com distribuição uniforme discreta.

Tabela 1 – Tempos de processamento (P) e tempos de preparação (S).

Relação	Intervalo de P	Intervalo de S
I	1-99	1-99
II	1-99	100-120
III	10-99	1-9
IV	50-99	1-49
V	1-99	1-120
VI	1-99	1-20

Sabe-se que se Z for uma variável aleatória discreta com distribuição uniforme no intervalo $[a, b]$, então a média $\mu[Z] = \frac{a+b}{2}$ e a variância $\text{var}[Z] = \frac{(b-a+1)^2 - 1}{12}$. A partir das distribuições de P e S da Tabela 1 e utilizando-se a expressão (39) podem ser obtidos os valores dos limitantes superiores do Desvio Padrão S_{DR} conforme apresentados na Tabela 2.

Tabela 2 – Limitantes superiores LS (S_{DR}) em função do número n de tarefas.

Número de máquinas	LS (S_{DR})					
	Relações de P e S					
	I	II	III	IV	V	VI
$m = 4$	$10,4/\sqrt{n}$	$2,2/\sqrt{n}$	$9,7/\sqrt{n}$	$3,0/\sqrt{n}$	$10,5/\sqrt{n}$	$12,3/\sqrt{n}$
$m = 7$	$26,3/\sqrt{n}$	$5,1/\sqrt{n}$	$24,1/\sqrt{n}$	$7,1/\sqrt{n}$	$26,5/\sqrt{n}$	$31,0/\sqrt{n}$

A Tabela 2 indica que, em termos relativos, as Relações II e IV devem levar aos menores valores do Desvio Padrão S_{DR} .

Na experimentação, para cada classe ($m \times n$) de problemas foi gerada aleatoriamente uma seqüência-semente inicial. A partir dessa seqüência-semente, foram obtidas todas as seqüências da respectiva Vizinhança de Inserção, constituída de $(n-1)^2$ seqüências. A Vizinhança de Inserção é obtida pela mudança de posição de cada uma das n tarefas para as $(n-1)$ posições diferentes na seqüência-semente. Para cada seqüência vizinha σ calculou-se o respectivo valor do Desvio Relativo $DR(\sigma)$. Tendo os valores dos desvios relativos, foram calculados os valores do Desvio Relativo Médio \overline{DR} e do Desvio Padrão do desvio relativo S_{DR} correspondentes à Vizinhança da seqüência-semente.

Com o objetivo de melhor estimar os valores de \overline{DR} e S_{DR} para o espaço de soluções ($n!$ seqüências), uma segunda seqüência-semente foi gerada da seguinte forma: dividiu-se aleatoriamente a última seqüência vizinha da seqüência-semente inicial em duas partes, invertendo-se as posições de todas as tarefas que compõem essas duas partes. Desta forma, procurou-se explorar melhor o espaço de soluções com um adequado grau de diversidade. Para a Vizinhança de Inserção da nova seqüência-semente, foram calculados os respectivos valores do Desvio Relativo Médio \overline{DR} e do Desvio Padrão do desvio relativo S_{DR} , os quais juntamente com os correspondentes à primeira seqüência-semente foram utilizados para o cálculo de \overline{DR} e S_{DR} correspondentes ao conjunto de $2(n-1)^2$ seqüências, envolvendo portanto as duas vizinhanças geradas.

Esse procedimento foi repetido sucessivamente até se atingir um valor estável para o Desvio Padrão do desvio relativo S_{DR} correspondente ao conjunto total das seqüências (soluções viáveis) até então avaliadas. Considerou-se que o Desvio Padrão S_{DR} atingiu a estabilidade quando seu valor permaneceu inalterado para 1000 seqüências-semente sucessivas, com uma precisão de 0,001 (0,1%). Isso significa que, para cada classe ($m \times n$) de problemas foram avaliadas, no mínimo, $1000(n-1)^2$ seqüências.

Os resultados obtidos na experimentação computacional são apresentados nas Tabelas 3 e 4. Os valores (N_s) indicam o número de seqüências-semente geradas para a estabilização do Desvio Padrão S_{DR} . Nas Tabelas, os melhores resultados (menores S_{DR}) estão salientados em **negrito** e os maiores S_{DR} grifados.

Tabela 3 – Resultados para tempos de preparação das máquinas independentes da seqüência das tarefas.

Classe m X n		Relações de P e S											
		I		II		III		IV		V		VI	
		Valores em porcentagem (%)											
		\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)
4	10	16,2	<u>2,9</u> (2210)	7,9	1,4 (1531)	16,8	2,8 (1575)	9,9	1,5 (1788)	15,2	2,4 (2174)	20,6	<u>2,9</u> (1638)
	20	23,8	1,8 (2204)	11,2	0,8 (1533)	23,5	1,7 (2103)	11,5	0,8 (1976)	22,0	1,5 (1435)	29,1	<u>2,0</u> (1884)
	30	25,2	1,2 (1567)	14,8	0,6 (2107)	28,3	1,4 (1857)	13,5	0,6 (1661)	27,2	1,2 (1254)	35,1	<u>1,8</u> (1389)
	40	28,6	1,0 (1297)	15,3	0,5 (1477)	33,2	1,1 (1866)	14,9	0,5 (1314)	28,6	1,0 (1147)	40,1	<u>1,4</u> (1449)
	50	30,1	0,7 (1535)	16,2	0,4 (1055)	36,8	1,0 (1175)	15,9	0,4 (1105)	29,2	0,7 (1737)	41,3	<u>1,1</u> (2041)
	60	30,2	0,6 (1680)	16,7	0,4 (1005)	38,9	<u>0,9</u> (1145)	16,9	0,3 (1129)	30,6	0,6 (1452)	41,9	<u>0,9</u> (2075)
	70	30,5	0,6 (1632)	16,7	0,3 (1005)	39,3	0,8 (1225)	17,1	0,3 (1016)	32,6	0,6 (1403)	43,8	<u>0,9</u> (1754)
	80	32,9	0,5 (1283)	16,7	0,3 (1039)	39,7	<u>0,7</u> (1458)	17,6	0,3 (1001)	34,7	0,5 (1267)	44,0	<u>0,7</u> (1043)
	90	34,2	0,5 (1764)	18,4	0,2 (1797)	42,2	<u>0,6</u> (1264)	18,1	0,2 (1026)	34,8	0,4 (1504)	45,6	<u>0,6</u> (1313)
7	10	19,1	2,6 (1965)	10,9	1,6 (2107)	17,4	2,6 (2185)	11,1	1,5 (1644)	17,2	2,7 (2028)	22,0	<u>2,9</u> (2218)
	20	25,9	1,8 (1970)	19,5	1,2 (2610)	34,1	2,3 (1939)	17,1	1,1 (1136)	27,8	1,8 (1473)	38,4	<u>2,5</u> (2021)
	30	31,7	1,3 (1875)	23,2	0,9 (1688)	41,8	1,8 (1268)	18,7	0,8 (1115)	31,8	1,3 (1656)	45,9	<u>1,9</u> (1123)
	40	38,9	1,1 (2135)	25,0	0,7 (1423)	46,8	<u>1,6</u> (1273)	21,4	0,6 (1620)	36,0	1,1 (1952)	47,2	<u>1,6</u> (1139)
	50	42,1	1,0 (1766)	25,3	0,5 (1758)	50,0	1,2 (1879)	21,9	0,5 (1330)	39,8	1,0 (1258)	52,9	<u>1,3</u> (1489)
	60	46,4	0,9 (1716)	26,3	0,5 (1089)	54,6	1,1 (1420)	22,4	0,4 (1632)	41,2	0,8 (1158)	55,7	<u>1,2</u> (1151)
	70	47,0	0,8 (1277)	26,6	0,4 (1450)	54,8	0,9 (2023)	24,8	0,4 (1010)	42,7	0,7 (1339)	58,3	<u>1,1</u> (1330)
	80	47,7	0,7 (1374)	26,7	0,4 (1043)	59,7	<u>0,9</u> (1177)	25,6	0,3 (1276)	42,7	0,6 (1083)	59,8	<u>0,9</u> (1889)
	90	47,6	0,6 (1699)	27,5	0,3 (1196)	62,5	<u>0,8</u> (1736)	27,7	0,3 (1102)	47,9	0,6 (1246)	64,7	<u>0,8</u> (1953)

Tabela 4 – Resultados para tempos de preparação das máquinas dependentes da seqüência das tarefas.

Classe m X n		Relações de P e S											
		I		II		III		IV		V		VI	
		Valores em porcentagem (%)											
		\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)	\overline{DR}	S_{DR} (Ns)
4	10	19,0	2,7 (3434)	8,1	1,5 (1387)	17,0	2,6 (1673)	9,0	1,5 (1543)	16,8	2,8 (2452)	18,5	<u>3,2</u> (2054)
	20	22,8	2,0 (1382)	13,8	0,9 (2163)	28,1	2,0 (2438)	12,4	1,0 (1446)	23,8	2,1 (1976)	29,5	<u>2,3</u> (2073)
	30	27,7	<u>1,6</u> (1427)	14,2	0,6 (1652)	30,1	1,5 (1992)	13,9	0,8 (1542)	24,9	<u>1,6</u> (1584)	35,8	<u>1,6</u> (2361)
	40	29,0	<u>1,3</u> (1933)	15,3	0,5 (1237)	34,5	<u>1,3</u> (1671)	14,7	0,6 (1493)	25,6	<u>1,3</u> (1493)	36,3	<u>1,3</u> (2084)
	50	29,6	<u>1,1</u> (2033)	16,6	0,4 (1568)	36,6	1,0 (1230)	15,1	0,5 (1064)	29,3	<u>1,1</u> (2127)	36,3	<u>1,1</u> (1446)
	60	32,3	<u>1,0</u> (1524)	16,6	0,4 (1024)	38,9	0,9 (1915)	16,0	0,5 (1056)	30,1	<u>1,0</u> (1724)	38,2	0,9 (1271)
	70	32,4	<u>0,9</u> (1494)	17,4	0,3 (1106)	39,6	0,8 (1766)	16,7	0,4 (1159)	31,8	<u>0,9</u> (1998)	39,0	0,8 (1216)
	80	33,1	<u>0,8</u> (1540)	18,7	0,3 (1099)	42,0	0,7 (1922)	16,8	0,4 (1015)	32,2	<u>0,8</u> (1792)	39,4	0,7 (1476)
	90	34,4	<u>0,7</u> (1966)	20,5	0,3 (1017)	45,8	<u>0,7</u> (1290)	18,3	0,3 (1529)	33,9	<u>0,7</u> (1578)	43,6	<u>0,7</u> (1788)
7	10	23,0	3,1 (2051)	12,9	1,7 (1243)	21,4	3,0 (2198)	10,9	1,6 (2015)	16,7	3,0 (2397)	27,6	<u>3,2</u> (2683)
	20	30,7	2,2 (2082)	19,5	1,2 (1458)	33,2	2,2 (1715)	16,2	1,1 (1929)	26,6	2,1 (1501)	36,0	<u>2,4</u> (2257)
	30	33,7	1,7 (1457)	22,2	0,9 (1120)	40,0	1,6 (2849)	20,7	0,9 (1606)	31,5	1,6 (1951)	47,8	<u>2,0</u> (1881)
	40	38,6	1,4 (1751)	22,3	0,7 (1079)	47,2	<u>1,5</u> (1838)	22,1	0,7 (1605)	37,8	1,4 (1819)	49,3	<u>1,5</u> (2614)
	50	43,0	1,3 (1701)	25,8	0,6 (1433)	53,3	1,3 (1650)	23,2	0,6 (1594)	40,3	1,3 (1571)	54,6	<u>1,4</u> (1257)
	60	44,6	1,1 (1345)	26,2	0,5 (1039)	55,4	1,1 (1743)	23,8	0,5 (1275)	42,2	1,1 (1350)	56,7	<u>1,2</u> (1205)
	70	45,3	1,0 (1884)	26,5	0,4 (1093)	56,6	1,0 (1117)	24,8	0,5 (1033)	43,8	1,0 (1315)	61,0	<u>1,1</u> (1531)
	80	46,7	0,9 (1502)	27,9	0,4 (1059)	60,1	0,9 (1821)	24,8	0,4 (1080)	44,3	0,9 (1334)	64,1	<u>1,0</u> (1898)
	90	50,3	0,8 (2016)	29,5	0,4 (1003)	63,1	0,8 (1769)	25,7	0,4 (1032)	45,8	0,8 (1302)	64,8	<u>0,9</u> (1779)

Os resultados experimentais consubstanciam as conclusões teóricas advindas da expressão (39), como também dos valores dos limitantes superiores do Desvio Padrão S_{DR} apresentados na Tabela 2.

Para ambos os casos quanto aos tempos de preparação das máquinas (independentes ou dependentes da seqüência das tarefas), o grau de equivalência entre os espaços de soluções do problema de programação da produção e do problema cíclico assimétrico do caixeiro-viajante tende a aumentar à medida que aumenta o porte do problema quanto ao número de tarefas. Como esperado, observa-se que ocorre um pequeno aumento do Desvio Padrão S_{DR} quando se aumenta o número de máquinas. As Tabelas 3 e 4 também mostram que os melhores resultados estão associados às Relações II e IV, enquanto que, na maioria das classes de problemas, os maiores S_{DR} ocorreram para a relação VI.

Comparando os resultados da Tabela 3 com os correspondentes resultados da Tabela 4, nota-se que, de maneira geral, o Desvio Padrão S_{DR} apresenta uma tendência de ser maior quando os tempos de preparação das máquinas são dependentes da seqüência das tarefas. Este fato não é explicado pela expressão (39), que teoricamente é a mesma para os dois casos de tempos de preparação. Uma possível causa de tal tendência está associada ao procedimento de geração da amostra de soluções viáveis. Para tempos de preparação independentes da seqüência das tarefas somente os valores de $\sum_{j=0}^{n-1} X_{[j+1]}^m$ e $\sum_{j=0}^{n-1} UB X_{[j+1]}^m$ sofrem alteração quando é avaliada uma nova solução (seqüência), enquanto que para tempos de preparação dependentes também ocorre variação na $\sum_{j=0}^{n-1} s_{m[j+1]}$, conforme mostram as expressões (14) e (15).

Conforme já mencionado, o grau de equivalência entre os espaços de soluções do problema de programação da produção e do caixeiro-viajante análogo depende somente da amplitude do Desvio Padrão S_{DR} . Mesmo assim, convém observar que o desvio relativo médio \overline{DR} tende a aumentar com o aumento do número de tarefas e do número de máquinas e que os menores valores estão associados às Relações II e IV que forneceram também os menores valores de S_{DR} .

5. Considerações Finais

Neste artigo foi apresentada uma propriedade estrutural do problema de programação da produção *flow shop* permutacional com tempos de preparação das máquinas separados dos tempos de processamento das tarefas, a qual fornece um limitante superior do tempo de máquina parada entre a sua preparação e o início de execução das tarefas.

Uma aplicação da referida propriedade foi proposta a partir de uma analogia do problema de programação da produção com um problema cíclico assimétrico do caixeiro-viajante. Com o objetivo de avaliar a adequação da analogia proposta foi efetuada uma experimentação computacional sobre uma amostra de problemas.

De maneira geral, com exceção dos problemas de pequeno porte ($n = 10$ tarefas), os valores do Desvio Padrão S_{DR} do Desvio Relativo $DR(\sigma)$ são suficientemente pequenos de forma que a analogia proposta entre os problemas mencionados é adequada e pode ser aceita.

Obviamente, a solução do problema cíclico assimétrico do caixeiro-viajante é uma solução aproximada (heurística) do problema original de programação da produção.

Agradecimentos

Os autores agradecem as contribuições dos revisores. O trabalho apresentado neste artigo tem o apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq.

Referências Bibliográficas

- (1) Allahverdi, A. & Aldowaisan, T. (1998). Job lateness in flowshops with setup and removal times separated. *Journal of the Operational Research Society*, **49**, 1001-1006.
- (2) Bagga, P.C. & Khurana, K. (1986). Two-machine flowshop with separated sequence-independent setup times: mean completion time criterion. *Indian Journal of Management and Systems*, **2**, 47-57.
- (3) Bagchi, T.P.; Gupta, J.N.D. & Sriskandarajah, C. (2006). A review of TSP based approaches for flowshop scheduling. *European Journal of Operational Research*, **169**, 816-854.
- (4) Cao, J. & Bedworth, D.C. (1992). Flow shop scheduling in serial multi-product processes with transfer and setup times. *International Journal of Production Research*, **30**, 1819-1830.
- (5) Cheng, T.C.E.; Gupta, J.N.D. & Wang, G. (2000). A review of flowshop scheduling research with setup times. *Production and Operations Management*, **9**, 262-282.
- (6) Corwin, B.D. & Esogbue, A.O. (1974). Two-machine flowshop scheduling problems with sequence dependent setup times: a dynamic programming approach. *Naval Research Logistics Quarterly*, **21**, 515-524.
- (7) Das, S.R.; Gupta, J.N.D. & Khumawala, B.M. (1995). A savings index heuristic algorithm for flowshop scheduling with sequence dependent set-up times. *Journal of the Operational Research Society*, **46**, 1365-1373.
- (8) French, S. (1982). *Sequencing and Scheduling: An Introduction to the Mathematics of the Job-Shop*. Ellis Horwood Ltd., West Sussex, England.
- (9) Framinan, J.M.; Gupta, J.N.D. & Leisten, R. (2004). A review and classification of heuristics for permutation flow-shop scheduling with makespan objective. *Journal of the Operational Research Society*, **55**, 1243-1255.
- (10) Garey, M.R.; Johnson, D.S. & Sethi, R. (1976). The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research*, **1**, 117-129.
- (11) Gupta, J.N.D. (1968). Travelling salesman problem – a survey of theoretical developments and applications. *Opsearch (India)*, **5**, 181-192.

- (12) Gupta, J.N.D. (1969). Economic aspects of scheduling theory. PhD Dissertation, Texas Tech University, Lubbock, Texas.
- (13) Gupta, J.N.D. (1975). A search algorithm for the generalized scheduling problem. *Computers & Operations Research*, **2**, 83-90.
- (14) Gupta, J.N.D. & Darrow, W.P. (1985). Approximate schedules for the two-machine flow-shop with sequence dependent setup times. *Indian Journal of Management and Systems*, **1**, 6-11.
- (15) Gupta, J.N.D. & Darrow, W.P. (1986). The two-machine sequence dependent flowshop scheduling problem. *European Journal of Operational Research*, **24**, 439-446.
- (16) Gupta, J.N.D. & Stafford Jr., E.F. (2006). Flowshop scheduling research after five decades. *European Journal of Operational Research*, **169**, 699-711.
- (17) Gupta, J.N.D.; Strusevich, V.A. & Zwaneveld, C. (1997). Two-stage no-wait scheduling models with setup and removal times. *Computers & Operations Research*, **24**, 1025-1031.
- (18) Hejazi, S.R. & Saghafian, S. (2005). Flowshop-scheduling problems with makespan criterion: a review. *International Journal of Production Research*, **43**, 2895-2929.
- (19) Johnson, S. (1954). Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, **1**, 61-68.
- (20) Khurana, K. & Bagga, P.C. (1985). Scheduling of job-block with deadline in $n \times 2$ flowshop problem with separated setup times. *Indian Journal of Pure Applied Mathematics*, **16**, 213-224.
- (21) Moccellin, J.V. (1995). A new heuristic method for the permutation flow shop scheduling problem. *Journal of the Operational Research Society*, **46**, 883-886.
- (22) Mood, A.M.; Graybill, F.A. & Boes, D.C. (1974). *Introduction to the Theory of Statistics*. McGraw-Hill Kogakusha, Ltd., Tokyo, Japan.
- (23) Park, T. & Steudel, H.J. (1991). Analysis of heuristics for job sequencing in manufacturing flow line work-cells. *Computers & Industrial Engineering*, **20**, 129-140.
- (24) Pinedo, M. (1995). *Scheduling: Theory, Algorithms, and Systems*. Prentice Hall, Inc., Englewood Cliffs, New Jersey.
- (25) Rajendran, C. & Ziegler, H. (1997). Heuristics for scheduling in a flowshop with setup, processing and removal times separated. *Production Planning & Control*, **8**, 568-576.
- (26) Rios-Mercado, R.Z. & Bard, J.F. (1996). The flowshop scheduling polyhedron with setup times. Technical Report ORP96-07, Graduate Program in Operations Research, University of Texas at Austin, Austin-TX.
- (27) Rios-Mercado, R.Z. & Bard, J.F. (1998). Heuristics for the flow line problem with setup costs. *European Journal of Operational Research*, **110**, 76-98.
- (28) Rios-Mercado, R.Z. & Bard, J.F. (1999a). A branch-and-bound algorithm for flowshop scheduling with setup times. *IEEE Transactions*, **31**, 721-731.
- (29) Rios-Mercado, R.Z. & Bard, J.F. (1999b). An enhanced TSP-based heuristic for makespan minimization in a flowshop with setup times. *Journal of Heuristics*, **5**, 53-70.

- (30) Simons Jr., J.V. (1992). Heuristics in flow shop scheduling with sequence dependent setup times. *Omega*, **20**, 215-225.
- (31) Srikar, B.N. & Ghosh, S. (1986). A MILP model for the n-job, m-stage flowshop, with sequence dependent setup times. *International Journal of Production Research*, **24**, 1459-1472.
- (32) Stafford, E.F. & Tseng, F.T. (1990). On the Srikar-Ghosh MILP model for the N X M SDST flowshop problem. *International Journal of Production Research*, **28**, 1817-1830.
- (33) Stinson, J.P. & Smith, W. (1982). A heuristic programming procedure for sequencing the static flowshop. *International Journal of Production Research*, **20**, 753-764.
- (34) Yoshida, T. & Hitomi, K. (1979). Optimal two-stage production scheduling with setup times separated. *AIEE Transactions*, **11**, 261-263.
- (35) Weng, M.X.; Lu, J. & Ren, H. (2001). Unrelated parallel machine scheduling with setup consideration and a total weighted completion time objective. *International Journal of Production Economics*, **70**, 215-226.
- (36) Widmer, M. & Hertz, A. (1989). A new heuristic method for the flow shop sequencing problem. *European Journal of Operational Research*, **41**, 186-193.