

UMA HEURÍSTICA DE GERAÇÃO DE COLUNAS PARA O PROBLEMA DE FORMAÇÃO DE CÉLULAS DE MÁQUINAS E PARTES

Geraldo Ribeiro Filho

UNISUZ – Faculdade Unida de Suzano
Rua José Correia Gonçalves 57
08675-130 Suzano – SP
geraldorf@uol.com.br

Luiz Antonio N. Lorena

LAC/INPE- Instituto Nacional de Pesquisas Espaciais
Caixa Postal 515
12201-970 - São José dos Campos – SP
lorena@lac.inpe.br

Resumo

Formação de células de máquinas e partes (FCMP) é um problema de criação de células de manufatura visando melhor fluxo de produção de sub-sistemas gerenciáveis. É um problema importante e de difícil solução amplamente estudado na literatura de manufatura industrial. Este artigo analisa uma heurística baseada em geração de colunas aplicada a FCMP, atribuindo peças a famílias de acordo com suas similaridades. Em seguida uma heurística de busca local associa máquinas às famílias de partes, formando as células de manufatura. Experimentos computacionais confirmam a viabilidade da nova heurística comparando seus resultados com as melhores soluções conhecidas.

Palavras-chave: células de manufatura, geração de colunas, heurística

Abstract

The Machine-Part Cell Formation (MPCF) is the problem of creating manufacture cells aiming best production flow of manageable sub-systems. It is an important difficult problem largely studied in industrial manufacturing literature. This paper examines a column generation heuristic applied to MPCF, assigning parts to families according to their similarities. A further step assigns machines to families of parts to form manufacturing cells. Extensive computational experiments confirm the feasibility of the column generation heuristic compared to best known solutions.

Keywords: manufacturing cell, columns generation, heuristic

1. Introdução

A competição internacional e a conseqüente necessidade de respostas rápidas para as demandas do mercado têm levado as empresas a considerar várias abordagens não tradicionais para o controle e projeto de sistemas de manufatura. A “tecnologia de grupo” (Burbidge (1969)) decompõe sistemas de manufatura em subsistemas gerenciáveis, através do agrupamento de partes similares em famílias, e de máquinas em células para manufatura dessas famílias de partes. A automação e controle são simplificados através da criação de células independentes. A análise do fluxo de produção de Burbidge (1963) é uma das primeiras metodologias associadas à tecnologia de grupo. Chamamos esse problema de Formação de Células de Máquinas e Partes (FCMP).

Várias abordagens de otimização foram propostas na literatura para criar células de manufatura. Heurísticas para FCMP geralmente trabalham com uma representação matricial de zeros e uns, indicando quais máquinas são utilizadas para produzir cada parte. Basicamente, os algoritmos mudam posições de linhas e colunas para produzir blocos de uns, formando famílias de partes e células de máquinas simultaneamente. Os trabalhos de McCormick Jr. et al. (1972), King (1980), Chandrasekharan e Rajagopalan (1989) e Venugopal e Narendran (1993) apresentam uma análise sobre matrizes zero-um para extrair propriedades e sugerem algoritmos de formação de células. A Figura 1 mostra uma representação matricial na forma original para 10 máquinas (linhas) e 15 partes (colunas). A Figura 2 mostra a mesma matriz após o agrupamento de uns que associa partes a famílias e máquinas a células através do movimento de linhas e colunas.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0								1			1			1	1
1	1			1						1				1	
2	1	1							1						1
3			1		1	1				1					
4		1		1	1		1		1				1		
5		1		1										1	
6		1		1			1		1				1		
7											1			1	1
8								1					1	1	1
9			1		1						1				

Figura 1: Matriz original com 10 máquinas e 15 partes.

	1	3	6	8	11	0	2	4	5	9	7	10	12	13	14
1						1	1	1		1					1
3						1	1	1	1						
9						1	1		1						
2	1				1		1								
4	1	1	1	1	1			1							
5	1	1			1										
6	1	1	1	1	1										
0											1	1	1	1	
7												1		1	1
8											1		1	1	1

Figura 2: Matriz resultante do agrupamento de máquinas e partes.

Muitas técnicas de otimização para FCMP têm sido propostas na literatura. Métodos de agrupamento hierárquico (Gupta e Seifoddini (1990)), não-hierárquico (Chandrasekharan e Rajagopalan (1986a), Jayakrishnan Nair e Narendran (1998)), técnicas baseadas em grafos (Rajagopalan e Batra (1975), Lin et al. (1996)), redes neurais (Malave e Ramchandran (1991), Guerrero et al. (2002)), em lógica *fuzzy* (Torkul et al. (2006), Xu e Wang (1989) e Papaioannou e Wilson (2008)) e em meta-heurísticas como Simulated Annealing (Venugopal e Narendran (1992a)), Busca Tabu (Lei e Wu (2006)) e Algoritmos Genéticos (Venugopal e Narendran (1992b), Gonçalves e Resende (2004), Ribeiro Filho e Lorena (2000)). Jaumard et al. (1999) modelaram o problema de FCMP como problema de programação linear de dupla alocação com grande número de variáveis, que é então resolvido por geração de colunas. Os trabalhos de Joines et al. (1995), Gonçalves e Resende (2004) e Papaioannou e Wilson (2008) apresentam revisões detalhadas de metodologias alternativas.

FCMP é um problema de *clustering* e foi modelado como um problema de p -medianas em alguns trabalhos, começando com Kusiak (1987) e sendo explorado por Won e Currie (2004), entre outros. A busca de p vértices medianos em uma rede (grafo) é um problema clássico de localização. O objetivo é localizar p instalações (medianas) para minimizar a soma das distâncias de cada nó de demanda até a mediana mais próxima. Senne et al. (2007) apresentaram uma abordagem de geração de colunas para problemas de p -medianas, onde as colunas formam os grupos e novas colunas são geradas usando a relaxação de programação linear de um problema de cobertura de conjuntos com restrição de cardinalidade. O trabalho também examina algumas questões sobre instabilidade no processo de geração de colunas, recomendando uma estratégia simples para tratar este efeito indesejável.

Este trabalho apresenta um novo modelo para FCMP como um problema de particionamento com uma restrição de cardinalidade. A abordagem de geração de colunas para problemas de p -medianas introduzida por Senne et al. (2007) é adaptada para construir associações viáveis de partes em famílias. As associações são baseadas em distâncias de Hamming ou Jaccard definidas para cadeias binárias. Outro passo da heurística associa máquinas às famílias de partes para formar células de manufatura.

O restante deste artigo está organizado da seguinte forma: a seção 2 introduz a formulação de p -medianas e o algoritmo de geração de colunas, na seção 3 são apresentados algoritmos de atribuição de partes a células de máquinas e de máquinas a famílias de partes, a seção 4 apresenta os resultados computacionais, e considerações finais são apresentadas na seção 5.

2. Geração de colunas e p -medianas

A geração de colunas é uma ferramenta poderosa para resolver problemas de programação linear de grande porte. Esta técnica pode ser usada quando as colunas do problema não são conhecidas com antecedência e uma enumeração completa de todas as colunas não é uma opção, ou quando o problema é reescrito usando decomposição de Dantzig e Wolfe (1960). A técnica de geração de colunas é uma escolha natural em diversas aplicações, tais como problemas de corte de estoques, roteirização de veículos e programação de tripulações.

Na forma clássica de geração de colunas, o algoritmo itera entre um subproblema gerador de novas colunas e um problema mestre restrito. A solução do problema mestre produz uma determinada solução dual, que é utilizada no subproblema visando determinar se existe alguma coluna que pode ser acrescentada ao problema mestre.

O problema de FCMP pode ser formulado como o seguinte problema de particionamento de conjuntos com restrição de cardinalidade:

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

$$\text{(SP-Pmed):} \quad v(\text{SP} - \text{Pmed}) = \text{Min} \sum_{j=1}^m c_j x_j \quad (1)$$

$$\text{Sujeito a} \quad \sum_{j=1}^m a_{ij} x_j = 1 ; i = 1, \dots, n \quad (2)$$

$$\sum_{j=1}^m x_j = p \quad (3)$$

$$x_j \in \{0,1\}. \quad (4)$$

Onde:

$n = |N|$, é o número de partes pertencentes ao conjunto de partes N ,

$S = \{S_1, S_2, \dots, S_m\}$, é um conjunto de subconjuntos de N ,

$[a_{ij}]_{n \times m}$, é uma matriz com $a_{ij} = 1$ se $i \in S_j$, e $a_{ij} = 0$ caso contrário;

$$c_j = \text{Min}_{i \in S_j} \left(\sum_{k \in S_j} d_{ik} \right); \quad (5)$$

$[d_{ij}]_{n \times n}$ é uma matriz de distâncias (Hamming ou Jaccard); e

m é o número de colunas e p o número de famílias de partes a serem criadas.

A distância de Hamming entre duas seqüências de mesmo comprimento é o número de posições nas quais os símbolos correspondentes são diferentes.

O índice de Jaccard, também conhecido coeficiente de similaridade de Jaccard, mede a similaridade entre conjuntos de símbolos, e é definido como o tamanho da interseção dividido pelo tamanho da união desses conjuntos. A distância de Jaccard, que mede a diferença entre conjuntos, é complementar ao coeficiente de Jaccard e é obtida subtraindo o coeficiente de Jaccard de 1.

Dadas duas seqüências binárias, A e B , sejam as seguintes combinações de símbolos de A e B :

- M_{11} representa o número total de símbolos de A e B que têm valor 1 em ambas as seqüências.
- M_{01} representa o número total de símbolos de A e B em que o símbolo de A é 0 e o símbolo de B é 1.
- M_{10} representa o número total de símbolos de A e B em que o símbolo de A é 1 e o símbolo de B é 0.
- M_{00} representa o número total de símbolos de A e B que têm valor 0 em ambas as seqüências.

A distância de Jaccard entre A e B é dada por:

$$d_{A,B} = \frac{M_{01} + M_{10}}{M_{01} + M_{10} + M_{11}} \quad (6)$$

O problema (SP-Pmed) formulado acima foi utilizado para problemas de localização de facilidades por Senne et al. (2007). Se S é o conjunto de todos os subconjuntos de N , a formulação permite encontrar uma solução ótima para o problema de p -medianas. No entanto, o número de subconjuntos de S pode ser enorme, e um conjunto parcial de colunas é então considerado em seu lugar. Outra aproximação considera o problema a ser resolvido por geração de colunas como a seguinte versão de cobertura de conjuntos de (SP-Pmed):

$$(SC-Pmed): \quad v(SC - Pmed) = \text{Min} \sum_{j=1}^m c_j x_j \quad (7)$$

$$\text{Sujeito a} \quad \sum_{j=1}^m a_{ij} x_j \geq 1 ; i = 1, \dots, n \quad (8)$$

$$\sum_{j=1}^m x_j = p \quad (9)$$

$$x_j \in [0,1]. \quad (10)$$

O problema (SC-Pmed) também é conhecido como problema mestre restrito no contexto de geração de colunas. Após a definição de um conjunto inicial de colunas, este problema é resolvido e os seus custos duais finais $\pi_j, j = 1, \dots, n$ e α são usados para gerar novas colunas resolvendo-se o seguinte sub-problema

$$(Sub-Pmed): \quad v(Sub - Pmed) = \text{Min}_{i \in N} \left[\text{Min}_{y_j \in \{0,1\}} \sum_{j \in N} (d_{ij} - \pi_j) y_j \right] \quad (11)$$

O problema (Sub-Pmed) é resolvido por inspeção fazendo (para cada $i = 1, \dots, n$) $y_j = 1$ se $d_{ij} - \pi_j \leq 0$, e $y_j = 0$ se $d_{ij} - \pi_j > 0$. A coluna $\left[\frac{y_j}{1} \right]$ é adicionada ao problema (SC-Pmed) se $v(Sub-Pmed) < \alpha$. Na prática, para $i = 1, \dots, n$, todas as colunas satisfazendo $\left[\text{Min}_{y_j \in \{0,1\}} \sum_{j \in N} (d_{ij} - \pi_j) y_j \right] < \alpha$ podem ser adicionadas ao conjunto de colunas, acelerando o processo de geração de colunas.

O algoritmo de geração de colunas pode ser representado na seguinte forma:

1. Encontre um conjunto inicial de colunas para o problema (SC-Pmed);
2. Resolva (SC-Pmed) e retorne os valores duais $\pi_j, j = 1, \dots, n$ e α ;
3. Resolva o subproblema correspondente (Sub-Pmed) adicionando a (SC-Pmed) as colunas $\left[\frac{y_j}{1} \right]$ que satisfazem $\left[\text{Min}_{y_j \in \{0,1\}} \sum_{j \in N} (d_{ij} - \pi_j) y_j \right] < \alpha, i = 1, \dots, n$;
4. Se não forem encontradas novas colunas no passo (3) então pare, senão vá para o passo (2);

5. Resolva o problema (SP-Pmed) com as colunas geradas.

Resolver o problema (SP-Pmed) no passo (5) produz uma associação viável de partes a famílias. Um passo adicional deve ser realizado para associar as máquinas a células que produzem as famílias de partes. Estas associações são descritas na seção 3 e este processo caracteriza a heurística de geração de colunas.

Os problemas intermediários (SC-Pmed) no passo (2) e o problema de particionamento de conjuntos do passo (5) são resolvidos pelo solver CPLEX (ILOG, 2006). O teste de parada no passo (4) é geralmente combinado com outro critério, tal como um número máximo de colunas geradas no processo.

O conjunto inicial de colunas no passo (1) pode ser gerado de várias maneiras, simples ou com maior custo de processamento. Por exemplo, colunas podem ser geradas aleatoriamente escolhendo um número de partes para cada coluna e aleatoriamente escolhendo cada uma dessas partes. Por outro lado, o processo de geração pode implementar alguma abordagem heurística para criar colunas iniciais melhores e talvez acelerar a fase de geração de colunas.

Neste trabalho, o número de partes em cada coluna inicial foi dado por um número aleatório com distribuição normal com média $E = \frac{\text{número_de_partes}}{\text{número_de_células}}$ e desvio padrão

$$S = \frac{E}{4}, \text{ e as partes foram então aleatoriamente escolhidas.}$$

3. Associações

Resolver um problema de FCMP por geração de colunas envolve um passo posterior para associar as máquinas às células. O objetivo de cada célula de máquinas é produzir uma família de partes. Portanto, devemos associar cada máquina a uma específica família de partes obtida na fase prévia de geração de colunas.

As associações são feitas com uma heurística de busca local apresentada por Gonçalves e Resende (2004) que, em seu trabalho, de maneira oposta, associa partes a células de máquinas geradas por um algoritmo genético.

A heurística consiste de um procedimento de melhoria que é iterativamente aplicado. Cada iteração k do procedimento começa com um dado conjunto inicial de famílias de partes $P_k^{INITIAL}$, produz um conjunto de células de máquinas M_k^{FINAL} , e um conjunto de famílias de partes P_k^{FINAL} . Duas matrizes bloco-diagonais, B_{IF} e B_{FF} , podem ser obtidas combinando $P_k^{INITIAL}$ com M_k^{FINAL} e P_k^{FINAL} com M_k^{FINAL} , respectivamente. Entre essas duas matrizes, aquela com maior eficácia é escolhida como matriz resultante da iteração k . O procedimento é encerrado quando $P_k^{FINAL} = P_k^{INITIAL}$ ou a eficácia da matriz resultante da iteração k é menor ou igual à eficácia da matriz resultante da iteração $k-1$, (para $k > 2$). Caso contrário, o procedimento faz $P_{k+1}^{INITIAL} = P_k^{FINAL}$ e continua com a iteração $k+1$.

Cada iteração k da heurística de busca local consiste dos dois seguintes passos:

(1) Associação de máquinas ao conjunto inicial de famílias de partes $P_k^{INITIAL}$. Máquinas são associadas uma de cada vez, em qualquer ordem. Uma máquina é associada à família de partes que maximiza uma aproximação da eficácia, isto é, a máquina é associada à família de partes P^* , dada por

$$P^* = \arg \max_P \left\{ \frac{N_1 - N_{1,P}^{Out}}{N_1 - N_{0,P}^{In}} \right\} \quad (12)$$

Onde:

$\arg \max$ é o argumento que maximiza a expressão;

N_1 é número o total de 1's na matriz;

$N_{1,P}^{Out}$ é o número total de 1's fora dos blocos da diagonal se a máquina é associada a uma família de partes P ;

$N_{0,P}^{In}$ é o número total de 0's dentro dos blocos da diagonal se a máquina é associada a uma família de partes P .

Neste passo, a heurística gera um conjunto de células de máquinas M_k^{FINAL} . Seja μ_k^1 a eficácia da matriz bloco-diagonal definida por $P_k^{INITIAL}$ e M_k^{FINAL} .

(2) Associação de partes ao conjunto de células de máquinas M_k^{FINAL} obtido no passo (1). Partes associadas a células de máquinas uma de cada vez, em qualquer ordem. Cada parte é associada à célula de máquinas que maximiza uma aproximação da eficácia, isto é, é associada à célula M^* , dada por

$$M^* = \arg \max_M \left\{ \frac{N_1 - N_{1,M}^{Out}}{N_1 - N_{0,M}^{In}} \right\} \quad (13)$$

Onde:

$\arg \max$ é o argumento que maximiza a expressão;

N_1 é o número total de 1's na matriz;

$N_{1,M}^{Out}$ é o número total de 1's fora dos blocos da diagonal se a parte é associada à célula M ;

$N_{0,M}^{In}$ é o número total de 0's dentro dos blocos da diagonal se a parte é associada à célula M .

Neste passo, a heurística de busca local gera um novo conjunto de famílias de partes P_k^{FINAL} . Seja μ_k^2 a eficácia da matriz bloco-diagonal definida por P_k^{FINAL} e M_k^{FINAL} . A matriz bloco-diagonal resultante desta iteração tem uma eficácia dado por $\mu_k = \max(\mu_k^1, \mu_k^2)$. Se $P_k^{FINAL} = P_k^{INITIAL}$ ou $\mu_k \leq \mu_{k-1}$ então o processo iterativo é

encerrado e a matriz bloco-diagonal da iteração $k-1$ é o resultado. Caso contrário, o processo faz $P_{k+1}^{INITIAL} = P_k^{FINAL}$ a continua para o passo (1) da iteração $k+1$.

Assim, dado um conjunto inicial de partes $P^{INICIAL}$ obtido com a resolução do problema (SP-Pmed), o algoritmo de busca local para associação de máquinas a células pode ser representado da seguinte forma:

1. Enquanto a condição de parada for falsa:
 - 1.1. Obter M^{FINAL} a partir de $P^{INICIAL}$ associando máquinas às famílias de partes de acordo com o critério descrito pela expressão 12;
 - 1.2. Obter P^{FINAL} a partir de M^{FINAL} associando partes às células de máquinas de acordo com o critério descrito pela expressão 13;
 - 1.3. B_{IF} = matriz bloco-diagonal obtida com $P^{INICIAL}$ e M^{FINAL} ;
 - 1.4. B_{FF} = matriz bloco-diagonal obtida com P^{FINAL} e M^{FINAL} ;
 - 1.5. e_{IF} = eficácia da solução representada por B_{IF} ;
 - 1.6. e_{FF} = eficácia da solução representada por B_{FF} ;
 - 1.7. Se $e_{IF} > e_{FF}$ então
 - 1.7.1. $B_{sol} = B_{IF}$ é a matriz da solução final;
 - 1.7.2. $e_{sol} = e_{IF}$;
 Senão
 - 1.7.3. $B_{sol} = B_{FF}$ é a matriz da solução final;
 - 1.7.4. $e_{sol} = e_{FF}$;
 - 1.8. Condição de parada = $P^{FINAL} == P^{INICIAL}$ ou a partir da segunda iteração e_{sol} é menor ou igual a e_{sol} da iteração anterior;
 - 1.9. Se condição de parada é falsa, $P^{INICIAL} = P^{FINAL}$;
2. Retorna matriz bloco-diagonal B_{sol} com eficácia e_{sol} .

No trabalho de Gonçalves e Resende (2004) não é citado, mas o procedimento de associação acima descrito pode levar a soluções inválidas com menos de duas máquinas em uma célula ou menos de duas partes em uma família. Para garantir a validade da solução foi implementado um passo adicional após a associação de máquinas às famílias de partes e após a associação de partes às células de máquinas, antes do cálculo da eficácia da associação.

Este procedimento adicional faz uma lista de eventuais células de máquinas inválidas, e enquanto esta lista não estiver vazia, repetidamente seleciona uma célula da lista aleatoriamente, procura em todas as outras células com ao menos três máquinas, e transfere para a célula inválida selecionada a máquina que produzir o maior incremento na eficácia da solução. Se a célula inválida se torna válida com a transferência, ela é retirada da lista de células inválidas. Um processo análogo se faz para famílias de partes inválidas, e assim o procedimento certamente gera uma solução válida.

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

A eficácia de agrupamento foi medida com um coeficiente que considera o número de zeros dentro dos agrupamentos e o número de uns fora dos agrupamentos, respectivamente, representando a compactação dos agrupamentos e o movimento de partes entre células:

$$\text{Coef} = \frac{e - e_1}{e + e_0} \quad (14)$$

Onde:

e é o número de 1's na matriz

e_0 é o número de 0's dentro dos agrupamentos

e_1 é o número de 1's fora dos agrupamentos

O valor ideal do coeficiente é 1 (sem zeros dentro e sem uns fora dos agrupamentos), e melhor é o agrupamento quando maior for o coeficiente.

4. Resultados Experimentais

Os resultados experimentais foram obtidos com algumas instâncias usadas anteriormente no trabalho de Gonçalves e Resende (2004) e uma instância adicional com uma matriz de máquinas e partes 20x35 (Burbidge (1969)). A Tabela 1 relaciona as instâncias com a origem na literatura e tamanho.

Tabela 1 – Instâncias usadas nos testes.

	Origem da Instância	Máquinas	Partes	Células
1	King e Nakornchai (1982)	5	7	2
2	Waghodekar e Sahu (1984)	5	7	2
3	Seifoddini (1989)	5	18	2
4	Kusiak e Cho (1992)	6	8	2
5	Kusiak e Chow (1987)	7	11	3
6	Boctor (1991)	7	11	3
7	Seifoddini e Wolfe (1986)	8	12	3
8	Chandrasekharan e Rajagopalan (1986a)	8	20	3
9	Chandrasekharan e Rajagopalan (1986b)	8	20	2
10	Mosier e Taube (1985a)	10	10	3
11	Chan e Milner (1982)	10	15	3
12	Askin e Subramanian (1987)	14	24	5
13	Stanfel (1985)	14	24	5
14	McCormick et al. (1972)	16	24	5
15	Srinivasan et al. (1990)	16	30	4
16	King (1980)	16	43	5
17	Carrie (1973)	18	24	6
18	Mosier e Taube(1985b)	20	20	5
19	Kumar et al. (1986)	20	23	5
20	Carrie (1973)	20	35	4
21	Boe e Cheng (1991)	20	35	5
22	Chandrasekharan e Rajagopalan (1989)	24	40	7
23	Chandrasekaran e Rajagopalan (1989)	24	40	7
24	Chandrasekharan e Rajagopalan (1989)	24	40	7
25	Chandrasekharan e Rajagopalan (1989)	24	40	9
26	Chandrasekharan e Rajagopalan (1989)	24	40	9
27	Chandrasekharan e Rajagopalan (1989)	24	40	9
28	McCormick et al. (1972)	27	27	4
29	Carrie (1973)	28	46	9
30	Kumar e Vannelli(1987)	30	41	11
31	Stanfel (1985)	30	50	12
32	Stanfel (1985)	30	50	11
33	McCormick et al. (1972)	37	53	2
34	Burbidge(1969)	30	35	4

Os testes foram feitos com as duas distâncias, Hamming e Jaccard, na aplicação do algoritmo de geração de colunas. Como medida de desempenho foi usada a eficácia de agrupamento.

A Tabela 2 mostra os resultados computacionais da heurística de geração de colunas usando distância de Hamming, e a Tabela 3 mostra os resultados correspondentes usando a distância de Jaccard.

As colunas nas tabelas 2 e 3 mostram o número da instância relacionada na Tabela 1, a eficácia de agrupamento encontrada na literatura (Gonçalves e Resende (2004)), valores máximo, mínimo e médio da eficácia para dez execuções do algoritmo, tempo médio total, tempo médio para a fase de geração de colunas e tempo médio para a fase de resolução do

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

modelo de programação inteira, a média do número de voltas no loop de geração de colunas e o número médio de colunas geradas nas dez execuções do algoritmo. As tabelas trazem ainda uma última linha com valores médios de cada coluna.

Resultados em negrito nas colunas de máxima eficácia e da literatura indicam resultados obtidos pelo algoritmo melhores ou iguais aos da literatura. Em particular, as tabelas 2 e 3 mostram que para as instâncias de números 21 e 33 foram encontrados resultados melhores do que a literatura usando tanto a distância de Hamming quanto a distância de Jaccard.

Nestes experimentos, para todas as instâncias, a fase de geração de colunas começou com 300 colunas iniciais, aleatoriamente geradas. Foi adotado o limite máximo de 300 iterações para o processo de geração de colunas, e como critério adicional de parada foi adotado um limite máximo de 15 mil colunas geradas.

Tabela 2 – Resultados obtidos usando distância de Hamming.

Inst	Eficácia				Tempo (seg)			Loop	Cols
	Liter	Max	Min	Med	Tot	GC	PI		
1	0,7368	0,7368	0,7368	0,7368	0,08	0,04	0,04	0,3	0,6
2	0,6250	0,6250	0,6087	0,6103	0,05	0,02	0,04	1,0	1,6
3	0,7959	0,7959	0,7959	0,7959	1,06	0,24	0,82	75,6	907,6
4	0,7692	0,7692	0,7692	0,7692	0,08	0,04	0,04	1,2	4,2
5	0,5313	0,5313	0,4688	0,5157	0,28	0,12	0,15	61,7	459,0
6	0,7037	0,7037	0,7037	0,7037	0,19	0,08	0,11	32,6	283,0
7	0,6830	0,6830	0,6830	0,6830	0,21	0,08	0,12	34,1	297,9
8	0,8525	0,8525	0,8525	0,8525	0,12	0,05	0,08	11,9	173,7
9	0,5872	0,5833	0,5833	0,5833	1,18	0,42	0,76	216,2	1198,1
10	0,7059	0,7059	0,7059	0,7059	0,06	0,03	0,04	2,3	9,1
11	0,9200	0,9200	0,9200	0,9200	0,09	0,04	0,05	6,4	64,2
12	0,6986	0,6533	0,6026	0,6313	1,06	0,32	0,74	126,7	1379,5
13	0,6933	0,6933	0,6282	0,6776	0,66	0,19	0,47	68,6	857,9
14	0,5258	0,4851	0,4112	0,4524	0,18	0,07	0,11	44,4	191,2
15	0,6783	0,6783	0,6783	0,6783	1,58	0,47	1,11	164,6	1692,0
16	0,5486	0,5486	0,5353	0,5426	3,32	0,93	2,40	273,2	2505,2
17	0,5446	0,5273	0,3966	0,4547	0,19	0,08	0,11	39,8	193,6
18	0,4296	0,4109	0,3769	0,3987	0,21	0,08	0,13	40,1	257,4
19	0,4965	0,4887	0,3778	0,4564	1,01	0,29	0,72	128,3	1163,3
20	0,7622	0,7614	0,7614	0,7614	3,05	0,78	2,27	148,9	2625,2
21	0,5807	0,5815	0,5397	0,5547	1,06	0,32	0,74	107,9	1206,1
22	1,0000	1,0000	1,0000	1,0000	0,26	0,08	0,18	14,8	421,1
23	0,8511	0,8511	0,8511	0,8511	0,66	0,18	0,48	40,1	1046,1
24	0,7351	0,7351	0,7351	0,7351	0,72	0,25	0,48	82,0	1093,1
25	0,5197	0,5063	0,4383	0,4730	0,41	0,10	0,31	27,8	494,3
26	0,4706	0,4545	0,4167	0,4422	0,80	0,27	0,53	109,4	1103,2
27	0,4487	0,4348	0,3765	0,4217	0,33	0,12	0,21	45,9	408,2
28	0,5427	0,5180	0,4448	0,5010	1,15	0,33	0,82	158,9	1100,7
29	0,4462	0,4367	0,3577	0,4025	4,17	0,64	3,53	215,2	2288,7
30	0,5848	0,5283	0,4667	0,4950	1,45	0,23	1,22	41,5	1090,4
31	0,5966	0,5966	0,5525	0,5764	0,82	0,22	0,60	73,3	915,8
32	0,5051	0,5025	0,4697	0,4865	2,58	0,82	1,76	216,5	2107,5
33	0,5642	0,5672	0,5616	0,5633	48,49	17,52	30,98	284,7	7638,0
34	0,7571	0,7571	0,7571	0,7571	1,82	0,50	1,32	102,9	1960,0
Med	0,6495	0,6418	0,6115	0,6294	4,4	1,3	3,1	90,9	1439,0

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

Tabela 3 – Resultados obtidos usando distância de Jaccard.

Inst	Eficácia				Tempo (seg)			Loop	Cols
	Liter	Max	Min	Med	Tot	GC	PI		
1	0,7368	0,7368	0,7368	0,7368	0,8	0,6	0,2	300,0	993,6
2	0,6250	0,6250	0,6087	0,6120	0,5	0,4	0,1	270,0	482,2
3	0,7959	0,7959	0,7959	0,7959	4,7	1,1	3,6	300,0	3658,0
4	0,7692	0,7692	0,7692	0,7692	0,9	0,6	0,3	240,6	1207,2
5	0,5313	0,5313	0,4848	0,5205	0,9	0,7	0,3	300,0	1568,2
6	0,7037	0,7037	0,7037	0,7037	0,6	0,4	0,2	270,3	971,8
7	0,6830	0,6830	0,6830	0,6830	1,1	0,7	0,4	300,0	1662,6
8	0,8525	0,8525	0,8525	0,8525	1,5	0,6	0,9	271,6	1859,6
9	0,5872	0,5872	0,5766	0,5815	2,1	0,8	1,3	300,0	2000,3
10	0,7059	0,7059	0,7059	0,7059	0,5	0,4	0,1	152,0	582,2
11	0,9200	0,9200	0,9200	0,9200	0,5	0,2	0,3	65,1	658,2
12	0,6986	0,6625	0,5062	0,6357	1,8	0,7	1,1	243,2	1980,1
13	0,6933	0,6933	0,6711	0,6853	1,4	0,7	0,7	272,1	1522,6
14	0,5258	0,5152	0,4356	0,4822	2,1	0,8	1,3	300,0	3094,6
15	0,6783	0,6783	0,6783	0,6783	2,1	0,9	1,2	300,0	2329,9
16	0,5486	0,5486	0,5486	0,5486	7,1	1,8	5,3	300,0	4908,0
17	0,5446	0,5321	0,4727	0,4982	2,0	0,8	1,2	300,0	2684,1
18	0,4296	0,4074	0,3688	0,3959	1,4	0,7	0,6	300,0	2021,3
19	0,4965	0,4545	0,4514	0,4520	1,2	0,7	0,4	300,0	1295,5
20	0,7622	0,7614	0,7614	0,7614	6,6	1,8	4,8	298,3	4898,0
21	0,5807	0,5815	0,5285	0,5691	3,4	1,0	2,5	300,0	3693,3
22	1,0000	1,0000	1,0000	1,0000	0,3	0,1	0,2	10,7	385,6
23	0,8511	0,8511	0,8511	0,8511	4,5	1,1	3,4	203,5	4034,2
24	0,7351	0,7351	0,7351	0,7351	3,5	1,1	2,3	297,7	3835,6
25	0,5197	0,5188	0,4417	0,4905	3,9	1,1	2,9	298,7	4604,0
26	0,4706	0,4277	0,3631	0,4042	4,2	1,1	3,2	297,5	4320,9
27	0,4487	0,4251	0,3869	0,4070	3,9	1,1	2,9	300,0	4265,4
28	0,5427	0,5228	0,4729	0,4992	1,6	0,7	0,9	300,0	1333,3
29	0,4462	0,4309	0,3690	0,4014	6,4	1,3	5,1	296,1	4708,3
30	0,5848	0,5370	0,4798	0,5094	5,4	1,3	4,1	300,0	5917,7
31	0,5966	0,5966	0,5385	0,5700	4,1	1,1	3,1	300,0	3959,5
32	0,5051	0,5025	0,4488	0,4696	4,3	1,2	3,2	299,0	4059,4
33	0,5642	0,5672	0,5672	0,5672	56,4	17,8	38,5	300,0	8698,3
34	0,7571	0,7571	0,7571	0,7571	6,7	1,8	4,8	290,5	4797,6
Med	0,6438	0,6358	0,6080	0,6250	4,4	1,4	3,0	269,9	2911,5

Nas tabelas 2 e 3 vemos que usando a distância de Hamming tivemos resultados iguais ou melhores que a literatura em 20 das 34 instâncias (58,8%), e usando a distância de Jaccard obtivemos resultados iguais ou melhores que a literatura em 21 das 34 instâncias (61,8%).

Os testes foram realizados em um microcomputador Pentium IV de 3GHz, com código escrito em C#, utilizando a biblioteca Concert 2.6 do solver CPLEX, versão 11.

5. Considerações Finais

Este trabalho examinou uma nova heurística para o problema de FCMP. O problema de FCMP é modelado como um problema de geração de agrupamentos associando partes a famílias em um problema de particionamento de conjuntos com restrição de cardinalidade. As células de máquinas e famílias de partes são criadas baseadas nas associações obtidas na geração de colunas e um procedimento de busca local.

Os resultados computacionais obtidos com a heurística de geração de colunas foram muito bons, apresentando eficácias de agrupamento tão boas quanto as listadas na literatura, e até melhores em alguns casos. Pesquisa complementar está em curso para tratar instâncias de problemas de larga escala, como por exemplo, o efetivo gerenciamento de colunas removendo colunas improdutivas ao longo do processo.

Agradecimento: Os autores agradecem as sugestões e correções de dois revisores anônimos e também ao editor da revista. O segundo autor agradece ao Conselho Nacional de Desenvolvimento Científico e Tecnológico -CNPq (proc. 305225/2006-5 e 411837/2008-3).

Referências

- Askin, R. G., e Subramanian, S. (1987). A cost-based heuristic for group technology configuration. *International Journal of Production Research*, 25, 101–113.
- Boctor, F. (1991). A linear formulation of the machine-part cell formation problem. *International Journal of Production Research*, 29(2):343-356.
- Boe, W., e Cheng, C. H. (1991). A close neighbor algorithm for designing cellular manufacturing systems. *International Journal of Production Research*, 29(10), 2097–2116.
- Burbidge, J. L. (1963). Production flow analysis. *Production Engineer*, 42:742-752.
- Burbidge, J. L. (1969). An introduction of group technology. In, seminar on group Technology, Turin.
- Carrie, S. (1973). Numerical taxonomy applied to group technology and plant layout. *International Journal of Production Research*, 11, 399–416.
- Chan, H. M., e Milner, D. A. (1982). Direct clustering algorithm for group formation in cellular manufacture. *Journal of Manufacturing System*, 1, 65–75.
- Chandrasekharan, M. P. e Rajagopalan, R. (1986a). An ideal seed non-hierarchical clustering algorithm for cellular manufacturing. *International Journal of Production Research*, 24(2):451-464.
- Chandrasekharan, M. P., e Rajagopalan, R. (1986b). MODROC: An extension of rank order clustering for group technology. *International Journal of Production Research*, 24(5), 1221–1233.
- Chandrasekharan M. P. e Rajagopalan, R. (1989). Groupability: Analysis of the properties of binary data matrices for group technology. *International Journal of Production Research*, 27(6):1035-1052.
- Dantzig, G. e Wolfe P. (1960) Decomposition Principle for Linear Programs. *Operations Research* 8, pp. 101-111.
- Gonçalves, J. F. e Resende, M. G. C. (2004). An evolutionary algorithm for manufacturing cell formation. *Computers and Industrial Engineering* 47: 247-273.

- Guerrero, F., Lozano, S., Smith, K.A., Canca, D., Kwok, T. (2002). Manufacturing cell formation using a new self-organizing neural network. *Computers and Industrial Engineering* 42, 377-382.
- Gupta, T., Seifoddini, H. (1990). Production data based similarity coefficient for machine-component grouping decisions in the design of a cellular manufacturing system. *International Journal of Production Research* 28(7), 1247-1269.
- ILOG. (2006) ILOG CPLEX 10.0: user's manual. France. 478 p.
- Jayakrishnan Nair, J.G., Narendran, T.T. (1998). CASE: a clustering algorithm for cell formation with sequence data. *International Journal of Production Research* 36(1), 157-179.
- Jaumard, B., Labit, P. e Ribeiro, C. C. (1999). A column generation approach to cell formation problems in cellular manufacturing. *Le Cahiers du GERAD*, G-99-20.
- Joines, J.A., King, R.E. e Culbreth, C.T. (1995). A comprehensive review of production-oriented manufacturing cell formation techniques. *International Journal of Flexible Automation and Intelligent Manufacturing* 3, 254-264.
- King, J. R. (1980). Machine-component grouping formation in group technology. *International Journal of management Science*, 8(2):193-199.
- King, J. R., Nakornchai, V. (1982). Machine-component group formation in group technology: Review and extension. *International Journal of Production Research*, 20(2), 117-133.
- Kumar, K. R., Kusiak, A., e Vannelli, A. (1986). Grouping of parts and components in flexible manufacturing systems. *European Journal of Operations Research*, 24, 387-397.
- Kumar, K. R., e Vannelli, A. (1987). Strategic subcontracting for efficient disaggregated manufacturing. *International Journal of Production Research*, 25(12), 1715-1728.
- Kusiak, A. (1987). The generalized group technology concept, *International Journal of Production Research* 25 (4) 561-569.
- Kusiak, A., e Chow, W. (1987). Efficient solving of the group technology problem. *Journal of Manufacturing Systems*, 6(2), 117-124.
- Kusiak, A., e Cho, M. (1992). Similarity coefficient algorithm for solving the group technology problem. *International Journal of Production Research*, 30, 2633-2646.
- Lei, D., Wu, Z. (2006). Tabu search for multiple-criteria manufacturing cell design. *International Journal of Advanced Manufacturing Technology* 28, 950-956.
- Lin, T.L., Dessouky, M.M., Kumar, K.R., Ng, S.M. (1996). A heuristic based procedure for the weighted production - cell formation problem. *IIE Transactions* 28, 579-589.
- McCormick Jr., W. T. ; Schweitzer, P. J. e White, T. W. (1972). Problem decomposition and data reorganization by a cluster technique. *Operations Research*, 20(5):993-1009.
- Mosier, C. T., e Taube, L. (1985a). The facets of group technology and their impact on implementation, *OMEGA*, 13(6), 381-391.
- Mosier, C. T., e Taube, L. (1985b). Weighted similarity measure heuristics for the group technology machine clustering problem. *OMEGA*, 13(6), 577-583.
- Papaioannou, G., Wilson, J.M., 2008. Fuzzy extensions to integer programming models of cell-formation problems in machine scheduling. *Annals of Operations Research*, Available online 10.1007/s10479-008-0423-1.
- Rajagopalan, R. e Batra, J. L. (1975). Design of cellular production systems: A graph theoretic approach. *International Journal of Production Research*, 13(6):567-579.

- Ribeiro Filho, G. e Lorena, L. A. N. (2000). A Constructive Evolutionary Approach to the Machine-Part Cell Formation Problem . In: A. Fleury; H. Yoshisaki; L. B. M. Guimaraes; J. L. D. Ribeiro. (Org.). Buildings Competencies for International Manufacturing - Perspectives for developing countries. Porto Alegre: UFRGS/FEENG, p. 340-348.
- Seifoddini, H. (1989). Single linkage versus average linkage clustering in machine cells formation applications. *Computers and Industrial Engineering*, 16(3), 419–426.
- Seifoddini, H., e Wolfe, P. M. (1986). Application of the similarity coefficient method in group technology. *IIE Transactions*, 18, 3, 266-270.
- Senne, E. L. F.; Lorena, L. A. N e Pereira, M. A. (2007). A simple stabilizing method for column generation heuristics: an application to p-median location problems. *International Journal of Operations Research*, v. 4, p. 1-9.
- Srinivasan, G., Narendran, T., e Mahadevan, B. (1990). An assignment model for the part-families problem in group technology. *International Journal of Production Research*, 28, 145–152.
- Stanfel, L. (1985). Machine clustering for economic production. *Engineering Costs and Production Economics*, 9, 73-8.
- Torkul, O., Cedimoglou, I.H., Geyik, A.K. (2006). An application of fuzzy clustering to manufacturing cell design. *Journal of Intellegent & Fuzzy Systems* 17, 173-181.
- Venugopal, V. e Narendran, T. T. (1992a). Cell formation in manufacturing systems through simulated annealing: an experimental evaluation. *European Journal of Operational Research*, 63(3):409-422.
- Venugopal, V., Narendran, T.T. (1992b). A genetic algorithm approach to the machine-component grouping problem with multiple objectives. *Computers and Industrial Engineering* 22(4), 469-480.
- Venugopal, V. e Narendran, T. T. (1993). Design of cellular manufacturing systems based on asymptotic forms of a Boolean matrix. *European Journal of Operational Research*, 67:405-417.
- Waghodekar, P. H., & Sahu, S. (1984). Machine-component cell formation in group technology MACE. *International Journal of Production Research* 22, 937-948.
- Won, Y. e Currie, K. R. (2004). Efficient p-median mathematical programming approaches to machine-part grouping in group technology manufacturing. *Engineering Optimization* Vol. 36, No. 5, 555–573
- Xu, H. e Wang, H. P. (1989). Part family formation for gt applications based on fuzzy mathematics. *International Journal of Production Research*, 27(9):1637-1651.