SPOLM 2009

ISSN 2175-6295 Rio de Janeiro- Brasil, 05 e 06 de agosto de 2009.

024/2009 - ANÁLISE MULTICRITÉRIO DO PROBLEMA DE SELEÇÃO DE EQUIPAMENTOS EM AMBIENTE *JUST IN TIME*, UTILIZANDO-SE DE UM ALGORITMO GENÉTICO

José André de Moura Brito
IBGE – Instituto Brasileiro de Geografia e Estatística – DPE / COMEQ.
Av.Chile, 500, 10º Andar, Centro – Rio de Janeiro – RJ.
e-mail jose.m.brito@ibge.gov.br

Luiz Cláudio Lopes Alves
IBGE – Instituto Brasileiro de Geografia e Estatística – DPE /COSEC.
Av.Chile, 500, 10º Andar, Centro – Rio de Janeiro – RJ.
e-mail lcalves@ibge.gov.br

Carlos Augusto de Alcantara Gomes
IEPG/UNIFEI
R. Sambaíba,141 – Pontinha – Araruama – MG
e-mail alcantaragomes@unifei.edu.br

Resumo

O presente trabalho propõe uma abordagem baseada em algoritmos genéticos para o problema de seleção de equipamentos em ambiente *Just in Time*. A resolução de tal problema consiste na determinação do número de máquinas que devem ser utilizadas em ***E*** estágios, de forma a minimizar o custo total de produção. Um conjunto de resultados computacionais é apresentado, incluindo-se os tempos totais de produção de cada alternativa gerada. Isto permite que se faça uma adoção de soluções viáveis para a função objetivo original, possibilitando uma análise multicritério do problema.
Palavras-Chaves: Just in Time; Seleção de Equipamentos; Algoritmos Genéticos; MetaHeurísticas.

Abstract

This paper proposes an approach based on genetic algorithms for the equipment selection problem in a Just in Time environment. The resolution of this problem consists in determining the number of machines to be used in E stages, in order to minimize the total cost of production. A set of computational results is presented, including the total production time of each generated alternative. This allows us to make an adoption of suboptimal solutions for the original objective function, enabling an multicriteria analysis of the problem.
Keywords: Just in Time; Equipment Selection; Genetic Algorithms; Metaheuristics

1. INTRODUÇÃO

GUNASEKARAN et alii (1993) propuseram uma formulação matemática não-linear para o problema de seleção de equipamentos em sistemas de manufatura *Just in*

Time. A partir do estudo desta formulação, ALVES(1997) e ALVES et alii (2001) propuseram uma formulação modificada, considerando o desenvolvimento de três relaxações (restrições) que foram incorporadas à formulação original, quais sejam: uma relaxação que considera a perda de tempo devido à quebra de máquinas, a segunda a geração de sucata e a terceira de que a demanda seria variável. Observando que a utilização desta última relaxação pode implicar em um custo de perda de venda, caso não ocorra o atendimento à 100% da demanda do público.

O presente trabalho tem por objetivo, inicialmente, avaliar a adoção de soluções viáveis, produzidas a partir da aplicação de um algoritmo genético que considera a função objetivo associada com a formulação proposta por GUNASEKARAN et alii (1993).

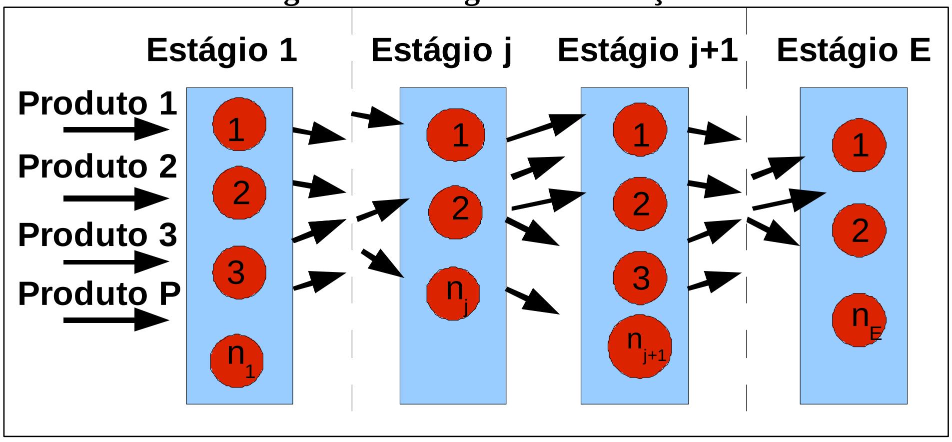
A adoção de tais soluções (produzidas pelo AG) tende a cooperar para o aumento na rapidez de produção e diminuir os custos em um sistema de manufatura *Just in Time*, e consequentemente, auxiliar em uma tomada de decisão multicritério de assumir custos maiores de investimento e/ou aluguel em equipamentos versus a redução do tempo total de produção, conforme ALVES et alli (2004).

O presente trabalho está dividido da seguinte forma: Na seção 2 temos uma descrição do problema de seleção de máquinas. A seção 3 apresenta os conceitos básicos da metaheurística algoritmos genéticos e uma descrição detalhada do algoritmo proposto para a resolução deste problema. Finalmente, na seção 4, temos os resultados computacionais e análises produzidas para um conjunto de instâncias geradas artificialmente.

2. PROBLEMA DE SELEÇÃO DE MÁQUINAS

O sistema de produção considerado neste trabalho tem múltiplos estágios, para os quais, podem existir máquinas semelhantes realizando os mesmos tipos de operações. Busca-se encontrar o número de máquinas requeridas em cada estágio que minimize o custo total de produção associado (figura 1), sem considerar nenhuma relaxação, isto é, em ambiente *Just in Time*, segundo GUNASEKARAN et alii (1993). O custo total do sistema consiste dos seguintes custos: (i) custo devido ao processamento dos produtos (ii) custo devido ao não-balanceamento nas taxas de produção; (iii) investimento em equipamentos;

Figura 1 – Estágios de Produção



A resolução de tal problema pode ser obtida mediante a aplicação da formulação proposta por GUNASEKARAN et alii (1993) e descrita a seguir, considerando-se, para tal, as seguintes premissas:

- A demanda para cada produto é assumida como sendo uniforme, determinística e conhecida.
- Não há quebra de máquinas. Esta é uma premissa simplificadora que pode ser obtida através da implementação da manutenção produtiva total (TPM) e da adoção de manutenção preventiva entre paradas.
- O custo de estoque é linearmente proporcional ao valor de estoque em processo ponderado.

- O tempo de transporte para lotes é desprezível. Uma vez que o conceito de GP tenha sido assumido para ser usado no sistema *JIT*, é razoável supor-se que não há tempo de transporte de subprodutos entre estágios.
- Pode-se assumir que as máquinas de cada estágio têm a mesma capacidade.
- Não há sucata.
- Não há custo de estoque de produto acabado, uma vez que os produtos são despachados assim que o processamento seja concluído.

2.1. FORMULAÇÃO

P : total de produtos
i : índice de produto $(i = 1,..., P)$
D_i : demanda do produto i por unidade de tempo ou ano;
E : total de estágios
j : índice de estágio $(j = 1,..., E)$

- **Para o estágio j :**

n_j **:** número de máquinas;
M_j **:** custo por máquina ou taxa de aluguel;
V_j **:** área do espaço requerido por uma máquina.

- **Para o i-ésimo produto e o j-ésimo estágio:**

AP_{ij} : tempo médio de processamento para um lote;
Q_{ij} : tamanho do lote;
O_{ij} : peso de prioridade dado no processamento (sequenciamento);
α_{ij} : custo devido ao processamento por unidade de tempo;
β_{ij} : custo de penalidade devido a uma unidade de desbalanceamento nas taxas de produção entre os estágios j e $j+1$;
PT_{ij} : tempo de processamento para um lote;
NPC_{ij} : número de ciclos de produção por unidade de tempo;
T_{ij} : tempo de processamento de uma unidade;
km_{ij} : taxa de diminuição no tempo de processamento face a um aumento de uma unidade no investimento de máquinas;
MM_{ij} : tempo de processamento por unidade quando o investimento em equipamento é zero;
λ_{ij} : taxa de produção
Cb_{ij} : custo devido à perda no tempo de processamento por unidade de tempo;
δ_{ij} : perda média percentual de tempo de processamento;
Cs_{ij} : custo devido à perda em um lote por unidade de tempo;
γ_{ij} : perda média percentual em um lote;
TM : orçamento de capital máximo disponível para investimento em máquinas;
Ω : espaço máximo disponível para máquinas na fábrica;
Mg_i : margem de lucro do produto i por unidade de tempo ou ano;
μ_i : percentagem da produção deixada de ser vendida por lote do produto i no último estágio;
Z : custo total do sistema relacionado com a seleção de equipamentos.

Considerando os parâmetros acima definidos, podemos apresentar os custos envolvidos no problema:

(i) Custo de processamento de produtos:
O custo total devido ao processamento de um produto para dada demanda, considerando todos produtos e todos os ***E*** estágios, por unidade de tempo (ano), é dado por:

$$\sum_{i=1}^{P}\sum_{j=1}^{E}\{NPC_{ij}.AP_{ij}.\alpha_{ij}\} \tag{1}$$

sendo:

$$NPC_{ij} = \frac{D_i}{Q_{ij}} \tag{2}$$

$$AP_{ij} = \frac{1}{\lambda_{ij}} \tag{3}$$

$$\lambda_{ij} = \frac{O_{ij}.n_j}{PT_{ij}} \tag{4}$$

$$PT_{ij} = Q_{ij}\, x T_{ij} \tag{5}$$

$$T_{ij} = MM_{ij} - km_{ij}\, M_j \; ; \; M_b \leq M_j \leq M_c \tag{6}$$

$$\sum_{i=1}^{P} O_{ij} = 1,\; j = 1,\ldots,E \tag{7}$$

(ii) Custo devido ao não-balanceamento das taxas de produção entre dois estágios sucessivos:

O custo total devido ao não-balanceamento nas taxas de produção considerando-se todos os produtos e estágios, é dado por:

$$\sum_{i=1}^{P}\sum_{j=1}^{E-1}\left[\left|\lambda_{ij} - \lambda_{ij+1}\right| \beta_{ij}\right] \tag{8}$$

(iii) Custo de investimento em equipamento:

O investimento total em equipamento (aluguel de equipamentos) por ano, considerando-se todos os estágios, é dado por:

$$\sum_{j=l}^{n} M_j n_j \tag{9}$$

Considerando a notação e as equações acima definidas, chegamos a seguinte formulação:

$$Min\ Z = \sum_{i=1}^{P}\sum_{j=1}^{E} NPC_{ij}\ x\ AP_{ij}\ x\ \alpha_{ij} + \sum_{i=1}^{P}\sum_{j=1}^{E-1}\left|\lambda_{ij} - \lambda_{ij+1}\right|\beta_{ij} + \sum_{j=1}^{S} M_j n_j \tag{10}$$

s.a

$$\sum_{j=1}^{E} M_j n_j \leq TM \tag{11}$$

$$\sum_{j=1}^{E} n_j V_j \leq \Omega \qquad (12)$$

A formulação matemática apresentada possui uma função objetivo não linear e não diferenciável e restrições lineares. Sendo n_j uma variável de decisão do tipo inteiro que corresponde ao número de máquinas alocadas a cada estágio ***j***.

Em vez de se utilizar uma abordagem de programação não-linear (BAZARAA, 1993) para determinar o número de máquinas por estágio, optou-se pela aplicação da metaheurística algoritmos genéticos, que será descrita mais à frente.

3. ALGORITMOS GENÉTICOS

A metaheurística algoritmos genéticos foi criada por Holland (1975) e apóia-se no desenvolvimento natural das espécies. A idéia que norteia a aplicação desta metaheurística está associada a teoria biológica dos sistemas naturais, onde os melhores "indivíduos" sobrevivem e geram descendentes com suas características hereditárias.

Portanto, analogamente, em um algoritmo genético (GLOVER E KOCHENBERGER, 2002) pode-se construir de forma aleatória uma população inicial de indivíduos (cromossomos), avaliando-se, em seguida, cada indivíduo desta população através da aplicação de uma função objetivo, selecionando-se os melhores (com menor valor função objetivo – um problema de minimização) e promovendo manipulações genéticas como cruzamento e mutação, a fim de criar uma nova população. GOLDBERG (1989), dentre outros, consolidaram o uso desta metaheurística, que pode ser adaptada de forma relativamente simples para resolver qualquer tipo de problema de otimização.

Primeiramente, deve-se determinar como os cromossomos serão representados, o que comumente é feito através de *strings* ou vetores. A população inicial pode ser gerada aleatoriamente ou utilizando alguma heurística de construção, produzindo p configurações (C_1 , C_2 , ..., C_p), onde p é o tamanho da população, que representa um pequeno subconjunto do espaço total de configurações, ou seja, de soluções viáveis do problema. Para cada cromossomo na população deve-se calcular o valor da função objetivo $f_i = f(C_i), \forall i = 1,..., p$. Finalmente, operadores genéticos devem ser aplicados à população na seguinte ordem:

(1) Reprodução dos melhores cromossomos (seleção). Baseado nos valores de f_i, as melhores sequências são selecionadas e duplicadas em substituição às piores utilizando o método da roleta viciada, do torneio, etc.

(2) Recombinação das estruturas genéticas da população, ou seja, o cruzamento (*crossover*). Esta operação possibilita a diversificação no espaço das configurações. O operador *crossover* (tradicional) escolhe aleatoriamente dois cromossomos e troca partes do seu padrão genético.

(3) Perturbação de uma configuração (mutação), isto é, a modificação de alguma porção de alguns indivíduos (gen ou gens de alguns cromossomos). Este procedimento permite a regeneração de algum gen que possa ter sido eliminado da população, de forma inesperada, em alguma etapa do processo.

Após a execução de todos estes procedimentos, obtém-se uma nova população. E sobre esta, deve-se repetir a avaliação da função objetivo para cada cromossomo e a aplicação dos operadores genéticos e assim sucessivamente, em um processo iterativo de geração de novas populações. Normalmente, são considerados os seguintes critérios de parada neste algoritmo: Número máximo gerações - Repetir os passos (1, 2, 3) por um número de vezes pré-determinado e o tempo máximo de processamento.

A saída é representada pela última população gerada, da qual o melhor elemento, isto é, o cromossomo que tem associado o menor (maior) valor da função objetivo no problema de minimização (maximização) é usualmente escolhido como a solução do problema.

3.1 ALGORITMO GENÉTICO PROPOSTO

Considerando o algoritmo proposto, a população é formada por m vetores (cromossomos), cada um com E posições que correspondem ao número de estágios do processo de fabricação. E para cada posição destes vetores, atribui-se um valor inteiro aleatório no intervalo entre 1 e $\left\lfloor \theta.\Omega / \sum_{j=1}^{E} V_j \right\rfloor$ $(\theta \geq 1)$, que corresponde ao número de máquinas consideradas em cada estágio $j, (j = 1,...,E)$. Adotando-se tais valores é possível gerar soluções viáveis de forma que as restrições (11) e (12) sejam satisfeitas. A tabela 1 a seguir tem um exemplo da representação de algumas soluções com uma distribuição do número de máquinas em 3 estágios.

Tabela 1 - Representação de Soluções com o Número de Máquinas por Estágio

Solução	***j=1***	***j=2***	***j=3***
$\mathbf{C_1}$	2	3	4
$\mathbf{C_2}$	1	5	6
$\mathbf{C_i}$	8	4	4
...	...	...	...
$\mathbf{C_m}$	3	3	3

Os m vetores $C = \{C_1, C_2, ..., C_i, ..., C_m\}$ gerados correspondem a um pequeno subconjunto S de soluções do espaço total de soluções, ou seja, representam algumas das possíveis configurações das máquinas necessárias no processo de produção. Após a construção, avalia-se a função objetivo (equação 10) para cada cromossomo C_i , ou seja, $f_i = f(C_i), \forall i = 1,...,m$, aplicando-se, em seguida, os operadores descritos abaixo:

(i) Seleção – A partir da aplicação do método do torneio (GLOVER E KOCHENBERGER, 2002), um conjunto de τ $(\tau < m)$ cromossomos é selecionado aleatoriamente da população atual e comparado, considerando o valor da função objetivo, sendo o melhor dentre estes o cromossomo selecionado. Tal método é aplicado m vezes de forma a obter uma nova população.

(ii) Cruzamento - São realizados t cruzamentos ($t < m$, corresponde a um percentual de m). Em cada cruzamento são selecionados aleatoriamente dois cromossomos C_i e C_k da atual população, efetuando-se incrementos (+1) e decrementos (-1) no número de máquinas associadas a cada posição $j, j = 1,...,E$ (estágio) de C_i e C_k (vide tabela 2).

Tabela 2 - Geração de Novos Cromossomos com Aplicação do Cruzamento

Cromossomo	***j=1***	***j=2***	***j=3***
$\mathbf{C_i}$	2	3	3
$\mathbf{C_k}$	4	5	1
$\mathbf{C_i}$ (intermediário)	**3**	3	3
$\mathbf{C_k}$ (intermediário)	**3**	5	1
$\mathbf{C_i}$ (intermediário)	**4**	3	3
$\mathbf{C_k}$ (intermediário)	**2**	5	1
$\mathbf{C_i}$ (intermediário)	4	**4**	3
$\mathbf{C_k}$ (intermediário)	2	**4**	1
$\mathbf{C_i}$ (intermediário)	4	**5**	3
$\mathbf{C_k}$ (intermediário)	2	**3**	1
...	...	...	...

Para os incrementos e os decrementos, e consequemente para a geração de novos cromossomos (soluções intermediárias), são observados os seguintes critérios:

- Se $C_i[j] < C_k[j]$, são aplicados r incrementos $(r = (C_k[j] - C_i[j])$ de uma unidade em $C_i[j]$ e p decrementos $(p \le r)$ de uma unidade em $C_k[j]$ enquanto $C_k[j] > 1$ (número de máquinas em cada estágio j maior do que 1).

- Se $C_i[j] > C_k[j]$, são aplicados p decrementos de uma unidade em $C_i[j]$, observando que $C_i[j] \ge 1$ e r incrementos $(r = (C_i[j] - C_k[j]))$ sobre $C_k[j]$.

Em cada operação de incremento/decremento, avalia-se a viabilidade da solução gerada, isto é, se o novo cromossomo satisfaz às restrições (11) e (12). Após a realização destas operações, atualizamos C_i e C_k com os cromossomos viáveis gerados que têm o menor valor da função objetivo associado (equação 10).

Além do cruzamento, também foi implementado um procedimento de busca local aplicado a cada 50 gerações do algoritmo genético, em todos os cromossomos da população, logo após o cruzamento.

Inicialmente, seleciona-se aleatoriamente para cada cromossomo C_i $(i = 1,...,m)$ uma posição ***j*** que contém o total de máquinas alocadas no estágio ***j***. E em seguida, avalia-se o valor da função objetivo para cada número inteiro (máquinas alocadas) entre $\max(C_i[j] - MaxViz, 1)$ e $C_i[j] + MaxViz$. Sendo atribuído a $C_i[j]$ um número de máquinas correspondente ao menor valor da função e que satisfaz às restrições (11) e (12).

(iii) Mutação – Foi realizada em poucos cromossomos C_i (5% da população), tendo por objetivo regenerar valores que porventura tivessem sido eliminados na reprodução ou cruzamento. Para cada um destes cromossomos C_i foi selecionada aleatoriamente uma posição j entre 1 e E, atribuindo-se a $C_i[j]$ um valor aleatório inteiro entre 1 e $\left\lfloor \theta.\Omega / \sum_{j=1}^{E} V_j \right\rfloor$. Ao efetuar tal atribuição, altera-se o número de máquinas que serão disponibilizadas em um determinado estágio no processo de produção.

4. RESULTADOS COMPUTACIONAIS E ANÁLISES

O algoritmo genético proposto neste trabalho foi implementado em linguagem Delphi e todos os experimentos computacionais foram realizados em um computador com 2Gb de memória RAM e dotado de um processador AMD de 2.3 Ghz.

Com a finalidade de avaliar e analisar a performance do novo algoritmo, foram utilizadas 20 instâncias geradas aleatoriamente. Neste conjunto de instâncias o número de estágios variou entre 2 e 10 e o número de inicial de máquinas, distribuídas em cada um dos estágios, variou entre 2 e 5.

4.1 APLICAÇÃO DO ALGORITMO E RESULTADOS

Para aplicação do algoritmo genético (***AG***), considerando cada uma das 20 instâncias, trabalhou-se com uma população de 100 cromossomos (possíveis configurações de máquinas). Ademais, no presente experimento o número de gerações foi fixado em 2000, as probabilidades de cruzamento e mutação foram fixadas em respectivamente 0.40 e 0.05, o parâmetro θ foi definido em 1.4 e o parâmetro *Maxviz* foi definido em 50.

A tabela 3 sumariza os resultados produzidos pelo ***AG*** para as 20 instâncias. Nesta tabela, a coluna um traz o número de identificação da instância, a coluna dois contém o número de estágios e as colunas três e quatro contém, respectivamente, o número de produtos e o número de máquinas por estágio. Na coluna cinco temos o tempo de CPU do algoritmo em segundos (execução das 2000 gerações) e nas colunas restantes, temos: o custo devido ao processamento (C_{proc}), o custo devido ao não balanceamento (C_{bac}), o custo de investimento em máquinas (C_{inv}), o custo total ***Z*** (equação 10) e o tempo total de produção definido por: $\sum_{i=1}^{E} \underset{i=1,...,P}{máximo} (NPC_{ij}.AP_{ij})$

Tanto o número de máquinas, quanto os custos $C_{proc}, C_{bac}, C_{inv}$, $\boldsymbol{Z}$ e o tempo total de produção $\boldsymbol{TP}$, disponibilizados para cada uma das instâncias, correspondem a uma solução inicial (primeira geração do AG), a uma solução obtida até a geração 1000 do AG (intermediária) e a melhor solução.

A partir deste resultados, podemos observar que o AG proposto pode produzir soluções viáveis de boa qualidade em um tempo computacional razoavelmente pequeno. Ademais, com a adoção destas soluções, pode-se cooperar para o aumento na rapidez de produção e diminuição dos custos em um sistema de manufatura *Just in Time*.

TABELA 3 – Resultados do Algoritmo Genético

Instância	S Estágios	P Produtos	N° de Máquinas por Estágio	Tempo AG	C_{proc}	C_{bac}	C_{inv}	Z	TP
1	**3**	**2**	5,3,4 5,3,2 2,2,2	**2**	17.479,5 22.404,5 35.025,8	41.966,3 42.419,2 14.786,7	57.500,0 49.500,0 29.000,0	**116.945,8** **114.323,7** **78.812,5**	5.578,0 6.813,0 11.360,5
2	**4**	**2**	4,6,8,10 3,2,1,3 2,2,2,2	**6**	15.533,4 46.374,4 37.542,7	54.590,7 46.408,1 150.912,2	145.000,0 48.000,0 69.000,0	**215.124,0** **140.782,6** **102.785,0**	5.135,6 13.732,4 12.262,2
3	**5**	**2**	5,2,2,4,1 4,4,5,4,4 3,2,2,2,2	**11**	52.840,0 27.177,2 51.037,7	89.178,8 45.011,0 26.440,3	72.500,0 104.000,0 55.000,0	**214.518,8** **176.188,2** **132.478,1**	14.162,9 8.083,4 14.680,9
4	**6**	**2**	4,6,8,10,16,2 3,5,5,3,5,6 2,3,2,1,2,2	**21**	27.499,2 32.998,5 73.972,9	237.720,8 75.703,2 40.247,4	227.000,0 133.000,0 59.500,0	**492.220,0** **241.701,7** **173.720,3**	8.118,0 9.892,0 21.172,9
5	**7**	**2**	9,12,9,14,13,6,8 7,6,3,3,2,5,3 2,2,3,3,4,2,2	**32**	16.121,1 42.113,0 63.323,8	211.700,5 133.798,8 53.633,9	351.500,0 144.000,0 87.000,0	**579.321,6** **319.911,8** **203.957,8**	4.656,0 11.409,1 18.810,5
6	**8**	**2**	4,8,7,6,6,8,16,20 2,4,3,3,4,4,13,14 4,5,2,3,3,3,2,3	**45**	25.386,0 48.174,4 61.327,9	213.656,3 167.077,2 87.596,8	369.000,0 229.000,0 126.500,0	**608.042,3** **444.251,6** **275.424,7**	7.455,0 14.324,3 16.393,6
7	**9**	**2**	20,27,22,10,5,2,3,3,1 3,4,2,3,7,5,6,7,4 2,4,4,4,3,3,3,5,8	**60**	56.255,4 51.493,6 59.259,0	356.975,0 152.583,9 92.469,6	462.000,0 200.000,0 176.000,0	**875.230,4** **404.077,4** **327.728,6**	14.533,9 14.647,8 17.039,9
8	**10**	**2**	1,12,6,9,15,14,1,5,7,5 13,5,6,8,6,4,3,5,8,2 10,5,5,3,4,4,5,5,5,3	**96**	73.183,1 48.249,4 50.615,9	496.520,2 282.420,2 160.672,2	374.500,0 297.000,0 238.500,0	**944.203,3** **627.669,6** **449.788,1**	22.813,3 12.646,7 13.330,6
9	**3**	**3**	7,5,7 4,4,3 3,3,3	**2**	30.380,1 52.078,2 63.035,4	54.028,7 25.056,8 19.832,5	90.500,0 54.000,0 43.500,0	**174.908,8** **131.135,0** **126.367,9**	6.472,6 11.095,0 13.146,7
10	**4**	**3**	3,7,8,9 3,3,3,4 3,3,3,3	**3**	47.038,5 76.785,9 102.652,5	78.252,3 36.175,3 30.190,0	139.500,0 67.500,0 51.000,0	**264.790,7** **180.461,2** **173.983,3**	12.417,0 20.877,8 27.461,1
11	**5**	**3**	4,3,8,10,9 4,3,5,5,5 3,3,3,3,3	**9**	57.607,8 72.684,4 101.685,5	103.963,8 57.510,6 35.451,1	169.000,0 109.000,0 75.000,0	**330.571,6** **239.194,9** **212.136,6**	10.579,5 13.843,7 19.872,8
12	**6**	**3**	4,6,8,10,1,10 4,5,4,4,2,4 2,3,4,3,3,4	**18**	116.296,2 110.576,8 107.254,1	254.781,0 99.773,6 142.629,6	199.500,0 116.500,0 171.500,0	**570.577,3** **326.850,4** **284.055,3**	26.203,0 25.282,4 24.873,6
13	**7**	**3**	2,2,4,13,3,2,5 4,4,8,7,5,3,5 3,3,8,3,3,4,4	**27**	187.924,9 113.362,5 148.882,4	245.145,0 120.668,8 63.683,0	158.500,0 173.500,0 131.000,0	**591.569,9** **407.531,3** **343.565,4**	49.942,1 29.712,3 39.023,3
14	**8**	**3**	5,7,10,3,5,3,14,6 6,6,3,2,2,5,8,8 4,5,3,3,4,4,3,5	**43**	115.553,0 154.441,5 159.592,3	387.318,3 149.751,8 114.603,7	248.000,0 197.000,0 155.000,0	**750.871,2** **501.193,3** **429.195,9**	29.196,4 38.143,1 40.445,6
15	**9**	**3**	4,6,8,10,3,15,11,14,11 6,6,7,7,7,8,5,6,5 3,3,3,3,5,5,8,4,6	**60**	85.528,9 101.896,0 157.648,6	349.447,5 123.669,3 130.414,4	404.000,0 280.000,0 190.000,0	**838.976,3** **505.565,3** **478.063,0**	18.568,3 24.319,2 35.485,9
16	**10**	**3**	6,15,7,4,11,20,21,10,11,16 9,2,7,5,8,6,9,4,4,7 4,3,6,7,4,5,5,6,5,6	**85**	86.222,4 175.051,2 177.366,3	634.133,3 320.306,7 202.767,2	582.500,0 291.000,0 251.000,0	**1.302.855,7** **786.357,9** **631.133,5**	20.647,8 44.110,5 44.361,1
17	**3**	**4**	4,6,8 5,4,5 4,3,4	**3**	64.688,1 92.612,5 92.022,5	52.238,3 135.316,3 29.545,8	85.000,0 110.500,0 52.500,0	**201.926,4** **176.220,8** **174.068,3**	10.124,2 13.230,2 15.273,3
18	**4**	**4**	6,7,3,3 4,4,3,2 4,4,3,4	**4**	113.338,1 148.806,5 126.720,5	101.721,6 55.531,0 58.173,1	98.500,0 66.000,0 84.000,0	**313.559,7** **270.337,5** **259.421,9**	26.837,5 34.156,5 29.737,5
19	**5**	**4**	7,4,10,2,3 5,6,7,6,6 4,4,4,4,4	**8**	165.473,6 109.833,4 153.910,6	177.432,2 61.801,8 32.900,6	122.500,0 143.500,0 100.000,0	**465.405,9** **312.214,1** **286.811,3**	33.542,5 20.212,5 29.137,7
20	**6**	**4**	11,10,19,8,2,8 7,6,5,4,5,8 4,4,5,4,4,6	**22**	141.091,9 146.220,8 192.779,5	291.362,4 136.617,3 89.541,4	283.000,0 174.500,0 134.000,0	**715.454,3** **457.338,1** **416.320,9**	27.896,3 30.374,5 38.843,0

4.2 ANÁLISE GRÁFICA

Considerando como exemplo a instância 10 (tabela 3),verifica-se, de acordo com ALVES et alli (2004) que há uma possibilidade de se utilizar as soluções viáveis produzidas pelo algoritmo avaliando-se, ao invés do custo total produção (eixo X), a variável tempo total de produção (eixo Y), que também pode ser tão ou mais importante no estabelecimento de estratégias empresariais (PLATTS/GREGORY, 1994). O tempo total de produção, tem acréscimos marginais crescentes para valores de custo total de produção decrescentes abaixo de certos valores (próximos aos valores da iteração intermediária, veja gráfico 1).

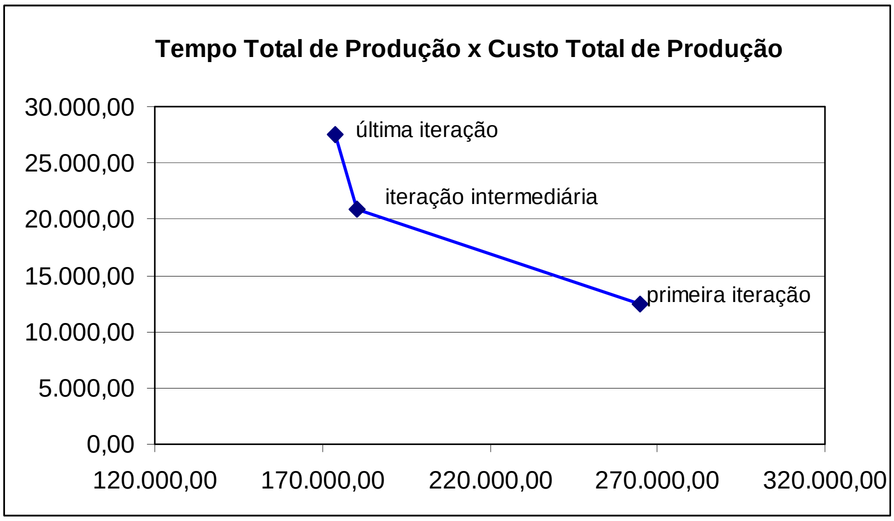


Gráfico 1 – Análise da Instância 10

Isto nos sugere que podemos realizar um *trade-off* entre estas duas variações marginais relativas, de modo que seja possível tomar uma decisão para uma política de investimento em máquinas, que seja ótima de acordo com os dois critérios mencionados, isto é, que o custo total de produção e o tempo total de produção sejam ótimos simultaneamente.

Em particular, na instância 10, podemos verificar que na iteração intermediária há a uma substancial redução no tempo total (24,0%), com um aumento muito pequeno do custo total (3,7%).

Nós poderíamos, portanto, para a iteração ótima da instância 10, decidirmos aumentar de uma unidade o número de equipamentos do 4º estágio, obtendo assim uma expressiva redução no tempo total de produção, sem um aumento significativo do custo total de produção.

A elaboração de um algoritmo de otimização multicritério global, utilizando-se da metodologia de otimização multicritério, apresentada no decorrer deste trabalho, fica como sugestão para pesquisas futuras.

5. REFERÊNCIAS BIBLIOGRÁFICAS

[1] ALVES, L.C.L., 1997. Um modelo para seleção de equipamentos em sistemas de produção Just-in-Case: O caso da relaxação do ambiente Just-in-Time**.** Tese D.Sc., COPPE, UFRJ, Rio de Janeiro, Brasil, 142p.

[2] ALVES, L.C.L., ALCANTARA GOMES, C.A., FUKS, S., 2001. Avaliação da Perda de Efetividade em um Sistema de Manufatura com Demanda Variável, através de um Modelo Não Linear de Seleção de Equipamentos**,** XXXIII SBPO, Simpósio Brasileiro de Pesquisa Operacional, Campos de Jordão, S.P., Brasil.

[3] ALVES, L.C.L., ALCANTARA GOMES, C.A., 2004. Adoção de Soluções Sub-ótimas da Função Objetivo Original de um Modelo Não-linear de Seleção de Equipamentos em Ambiente *Just in Time* e com Relaxações.**,** XXXVI SBPO, Simpósio Brasileiro de Pesquisa Operacional, São João Del-Rei, M.G., Brasil.

[4] ALVES, L.C.L., ALCANTARA GOMES, C.A., 2008. Adoption of Suboptimum Solutions of the Original Objective Function Optimizing a Nonlinear Programming Model**,** EngOpt 2008, International Conference on Engineering Optimization, Rio de Janeiro, R.J., Brasil.

[5] BAZARAA, M. S., SHERALI, H. D., SHETTY, C. M., 1979. Nonlinear programming. 2ª edição, 1993, John Wiley & Sons, inc., 638p.

[6] GLOVER, F. E KOCHENBERGER, G. A. (2002)**.** *"Handbook of Metaheuristic"*, First Edition Norwell: K1uwer Academic Publishers.

[7] GOLDBERG, D.E**.** (1989). *Genetic Algorithms in Search, Optimization and Machine Learning Reading*, Reading, MA: Addison-Wesley.

[8] GUNASEKARAN, A., GOYAL, S.K., MARTIKAINEM, T., YLI-0LLI, P., 1993. Equipment selection problem in Just-in-Time manufacturing systems. Journal of the Operational Research, v.44, n.4, p.345-353.

[9] HOLLAND, J.H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor.

[10] PLATTS, K., GREGORY, M. (1994). Competitive Manufacturing. A practical approach to the development of a manufacturing strategy. Department of Trade and Industry. London.