



## PROJETO DE ARRANJOS FÍSICOS DISTRIBUÍDOS POR MEIO DE OTIMIZAÇÃO DA SIMULAÇÃO E ALGORITMOS GENÉTICOS

**Anselmo Ramalho Pitombeira Neto**

Departamento de Engenharia Mecânica  
Escola de Engenharia de São Carlos – USP  
Av. Trabalhador São-carlense, 400, São Carlos - SP  
anselmop@sc.usp.br

**Eduardo Vila Gonçalves Filho**

Departamento de Engenharia Mecânica  
Escola de Engenharia de São Carlos – USP  
Departamento de Engenharia Mecânica  
Av. Trabalhador São-carlense, 400, São Carlos - SP  
evila@sc.usp.br

### RESUMO

O objetivo deste artigo é propor um procedimento para o projeto de arranjos físicos distribuídos por meio de otimização da simulação com o uso de algoritmos genéticos. Formula-se inicialmente o projeto de arranjo físico distribuído como um problema quadrático de alocação (QAP). Constrói-se então um modelo de simulação parametrizado, de forma que a função objetivo calculada em cada solução factível seja a estimativa do inventário médio em processo, o qual é obtido como resultado da simulação. Utiliza-se um algoritmo genético para a evolução das soluções, de forma a gerar boas soluções para o problema. O método é aplicado a um problema teste, com o objetivo de verificação e experimentação. Conclui-se, por fim, que a otimização da simulação permite gerar arranjos físicos distribuídos com considerável ganho de desempenho em termos de inventário em processo.

**PALAVRAS CHAVE.** Arranjos físicos distribuídos. Otimização da simulação. Algoritmo genético. SIM - Simulação

### ABSTRACT

The objective of this paper is to present a procedure for the design of distributed layouts via simulation optimization and genetic algorithms. The design problem is initially formulated as a quadratic assignment problem (QAP). Then, a simulation model is built, so that the estimated mean work-in-process, obtained as an output of the simulation, is assigned to the value of the objective function at each feasible solution. A genetic algorithm is used so as to guide the search, yielding good solutions. The method is applied to a toy problem, with the aim of verification and experimentation. Finally, it is concluded that simulation optimization is able to generate distributed layouts with considerable improvements in terms of work-in-process.

**KEYWORDS.** Distributed layouts. Simulation optimization. Genetic algorithms. SIM - Simulation.

## 1. Introdução

O objetivo deste artigo é propor um procedimento para o projeto de arranjos físicos distribuídos por meio de otimização da simulação com o uso de algoritmos genéticos. Existem vários métodos utilizados no projeto de arranjo físico. No entanto, muitos destes métodos não consideram medidas de desempenho operacional, como inventário em processo e tempo de atravessamento. Logo, o horizonte vislumbrado aqui é o da obtenção de arranjos físicos que, intrinsecamente, apresentem baixos inventários.

A partir dos anos 80, com a implantação, pelos japoneses, de estratégias de manufatura como diferencial competitivo, indústrias em todo o mundo passaram a buscar melhorias de processo e redução de desperdícios. Essa nova visão da produção deu origem a vários movimentos inspirados em experiências bem sucedidas, tais como a manufatura *Just-in-time*, Qualidade Total, e mais recentemente a Produção Enxuta.

Uma característica comum a todos esses movimentos é a importância dada à redução de desperdícios. Sob essa ótica, movimentações desnecessárias devem ser minimizadas e inventário em processo deve ser reduzido ao máximo, de forma a gerar menores custos de capital e melhorar o nível de serviço ao cliente por meio de tempos de atravessamento mais curtos.

Arranjos físicos distribuídos surgiram como uma alternativa a arranjos tradicionais, e são obtidos pela desagregação dos departamentos funcionais e distribuição individual das máquinas no chão-de-fábrica (Montreuil et al., 1991). A lógica subjacente ao conceito deste tipo de arranjo é a de que, sob um ambiente de mercado volátil, a distribuição dos recursos aumentaria a probabilidade de se encontrar bons roteiros de produção, viabilizando também a formação de células virtuais (Nomden et al., 2006).

Com isto em mente, foram propostas muitas abordagens para o projeto deste tipo de arranjo. Montreuil, et al. (1991) estão entre os primeiros a desenvolver um procedimento para o projeto de arranjos físicos distribuídos, aos quais deram o nome de “arranjos físicos holográficos”. O método que empregaram tenta obter a melhor alocação de máquinas a locais predefinidos de forma que a proximidade entre diferentes tipos de máquinas seja uniforme. Outros artigos que tratam do projeto de arranjos físicos distribuídos são os de Benjaafar e Sheikzadeh (2000), e o de Baykasoglu (2003).

A seqüência de apresentação deste artigo é a seguinte: o item 2 faz uma introdução dos métodos e técnicas utilizadas, e descreve o procedimento proposto; o item 3 apresenta um problema-teste e os resultados obtidos com a aplicação do procedimento; finalmente, o item 4 discorre sobre algumas implicações dos resultados e explorações futuras do tema.

## 2. Metodologia

### 2.1 Problema quadrático de alocação

O problema de projeto de arranjo físico trata da disposição física de máquinas, estações de trabalho, departamentos e equipamentos de suporte no chão-de-fábrica (Tompkins et al., 2002). Há várias abordagens e metodologias na literatura, revisadas em artigos como os de Kusiak e Heragu(1987) e Singh e Sharma (2006).

Entre as várias abordagens existentes, adota-se aqui a modelagem como problema quadrático de alocação (QAP). No QAP clássico, a função objetivo é definida como o custo total de movimentação de materiais, buscando-se sua minimização respeitando-se um conjunto de restrições. Neste artigo, no entanto, a função objetivo considerada é o inventário em processo, chamado em inglês de *work-in-process* (WIP). A formulação matemática consiste nas equações 1 a 4. Seja:

- $d_{kl}$  – medida de distância entre locais  $k$  e  $l$ . Utiliza-se aqui a distância Hamming;
- $x_{ik}$  - 1 se o departamento (ou máquina)  $i$  está alocado no local  $k$ . 0 caso contrário;
- $N$  - número de departamentos (ou máquinas);

- $M$  - número de locais disponíveis ( $M \geq N$ ).

O problema é então:

$$\text{Min } E[WIP] \quad (1)$$

sujeito a

$$\sum_{i=1}^N x_{ik} = 1 \quad \forall k \in \{1, 2, \dots, M\} \quad (2)$$

$$\sum_{k=1}^M x_{ik} = 1 \quad \forall i \in \{1, 2, \dots, N\} \quad (3)$$

$$x_{ik}, x_{jl} \in \{0, 1\} \quad (4)$$

A função objetivo dada pela equação 1 é, teoricamente, o valor esperado do WIP no arranjo físico distribuído, para o qual se deseja encontrar a alocação que produz o WIP esperado mínimo. Não obstante, dada uma alocação factível, o valor esperado de WIP é estimado com base em uma amostra, gerada pelo conjunto das replicações da simulação executada para a alocação atual.

A restrição 2 assegura que somente 1 departamento é alocado ao local  $k$ , e a restrição 3 assegura que o local  $i$  terá apenas 1 departamento alocado a ele, definindo desta maneira uma relação biunívoca. A restrição 4 limita as variáveis de decisão a valores binários, 0 ou 1. O QAP é um problema combinatório e pode ser solucionado exatamente para o caso determinístico por enumeração completa ou implícita. Entretanto, o QAP faz parte da classe de problemas NP-difíceis (Garey e Johnson, 1979), para os quais existem poucos algoritmos eficientes para a obtenção de soluções ótimas. Logo, utilizam-se heurísticas e meta-heurísticas para a obtenção de “boas” soluções para os problemas. Isso é muito conveniente no âmbito da engenharia, visto que, frente à infinidade de soluções possíveis para um problema, a obtenção de uma solução satisfatória em termos econômicos é desejável e suficiente. Por essa razão, empregam-se neste artigo algoritmos genéticos para orientar a busca por soluções satisfatórias.

## 2.2 Algoritmos genéticos

Algoritmos genéticos surgiram nos anos 70 em diversos artigos e livros. A lógica deste tipo de algoritmo é inspirada na evolução Darwiniana (Goldberg, 1989; Gen e Cheng, 1997). Cada solução no espaço de soluções é considerada um indivíduo em uma população. O problema em si é tratado como o “ambiente” no qual os indivíduos estão inseridos, e a função objetivo que se pretende otimizar é vista como uma medida de adequação (ou medida de *fitness*) do indivíduo ao ambiente. À medida que o algoritmo avança, soluções factíveis passam pelos processos de seleção, reprodução e mutação, até o encontro de algum critério de parada. Em geral, o critério de parada é especificado como um número máximo de gerações a serem evoluídas. Após a execução, a solução mais adequada (com melhor valor de *fitness*) é considerada a melhor solução encontrada.

### 2.2.1 Codificação e operadores evolutivos utilizados

Algoritmos genéticos possuem propriedades desejáveis para a solução de problemas de otimização combinatória, especificamente o QAP (Tam, 1992). No caso considerado aqui, o de arranjos físicos distribuídos, a representação de uma solução factível é feita por uma *string* de

números inteiros, na qual cada posição representa um local, e o inteiro associado à posição é a máquina alocada àquele local. A figura 1 apresenta a codificação utilizada.

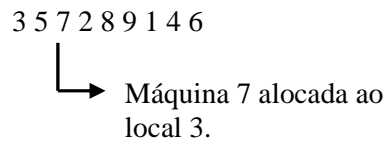


Figura 1: Codificação de um indivíduo.

Como em um arranjo físico podem existir muitas réplicas de um mesmo tipo de máquina, utiliza-se um mapeamento para representar tanto os tipos quanto as réplicas em uma mesma *string*. Na figura 2, por exemplo, a máquina de índice '2' é, na realidade, a réplica '1.2' do tipo de máquina '1'.

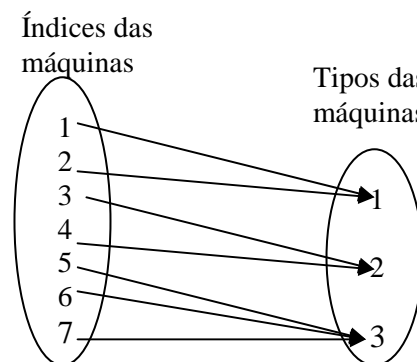


Figura 2. Mapeamento dos índices das máquinas em seus respectivos tipos.

Os operadores utilizados são os seguintes (Gen e Cheng, 1997):

- Operadores de seleção: É usado o mecanismo da roleta, no qual cada indivíduo tem uma probabilidade de seleção proporcional ao seu *fitness*, complementado pela seleção por torneio, na qual um par selecionado pela roleta é comparado, e aquele que possui maior *fitness* é eleito para reprodução;
- Operador de cruzamento: Utiliza-se o *partial mapped crossover* (PMX), o qual consiste numa troca de material genético interior a 2 pontos aleatórios das *strings*, com um procedimento que evita a produção de *strings* inválidas;
- Operador de mutação: É usada a mutação do tipo inversão, na qual 2 pontos são selecionados aleatoriamente em uma *string*, e a *substring* interior aos 2 pontos é invertida especularmente.

### 2.3 Otimização baseada em simulação

A simulação de eventos discretos trata da modelagem de um sistema real de forma a permitir a realização de experimentos estatísticos em um computador (Law e Kelton, 2000). É uma ferramenta largamente utilizada para a análise e o projeto de sistemas de manufatura, graças à sua flexibilidade e capacidade de captar a variabilidade em sistemas complexos, principalmente sistemas com filas.

Dá-se o nome de “otimização da simulação”, ou “otimização baseada em simulação” a técnicas de otimização estocástica que tentam estimar o valor da função objetivo em um

determinado vetor de decisão por meio da simulação (Fu, 2002, 2005). Tais técnicas representam uma alternativa à otimização de modelos estocásticos de manufatura, visto que existem poucos modelos capazes de captar todas as nuances existentes em um sistema real, e os mesmos baseiam-se fortemente em hipóteses simplificadoras e restritivas (Buzacott e Shanthikumar, 1993).

Encontram-se na literatura, exemplos do uso da otimização da simulação em sistemas de manufatura, as quais se tornaram possíveis com o aumento progressivo da capacidade computacional a partir do início dos anos 90. Azadivar e Tompkins (1999) utilizam um algoritmo genético e o paradigma da orientação a objetos para a otimização de um modelo de simulação de um sistema de manufatura considerando aspectos qualitativos. Pierreval e Paris (2003) tratam da otimização da própria configuração do modelo, e Azadivar e Wang (2000) apresentam uma aplicação no projeto de arranjos físicos funcionais.

A figura 3 exibe o esquema geral do procedimento de otimização da simulação (FU, 2002).

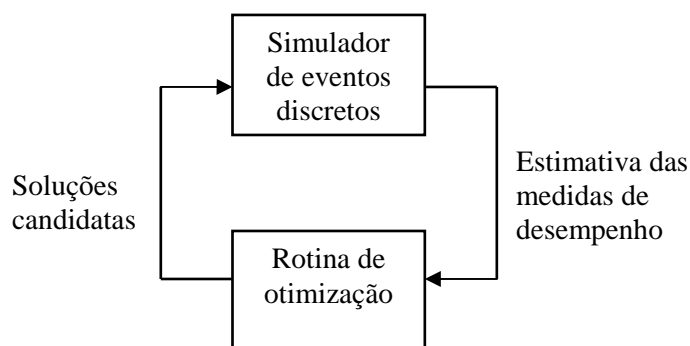


Figura 3: Procedimento de otimização da simulação.

## 2.4 Implementação computacional de um procedimento de otimização da simulação

O procedimento de otimização da simulação apresentado neste artigo baseia-se na construção de um modelo de simulação de eventos discretos de um arranjo físico distribuído. Este modelo consiste de um conjunto de máquinas, um conjunto de nós, aos quais as máquinas serão alocadas, e do sistema discreto de movimentação, o qual é uma abstração de qualquer transportador discreto. Os nós são os centros de regiões quadradas de um diagrama em blocos, no qual qualquer par de blocos adjacentes possui distância Hamming igual a 1 unidade. A figura 4 exibe um exemplo de um diagrama de blocos.



Figura 4: Representação de um chão-de-fábrica como um diagrama em blocos. Neste caso, um arranjo físico distribuído é apresentado.

O procedimento para a geração de arranjos físicos distribuídos é o seguinte:

0. Gere aleatoriamente uma população de *strings* de números inteiros, cada uma representando um arranjo físico;
1. Para cada arranjo físico na população, execute uma simulação com um número especificado de replicações, de forma a obter uma estimativa da medida de adequação (*fitness*);
2. Atribua o resultado da simulação ao *fitness* de cada indivíduo;
3. Verifique se algum indivíduo na população possui *fitness* maior que o melhor indivíduo encontrado até o momento. Caso positivo, torne esse indivíduo o melhor indivíduo até o momento;
4. Se o critério de parada for atingido, encerre a execução;
5. Gere uma nova população utilizando operadores de seleção, cruzamento e mutação, e retorne para o passo 1.

A figura 5 exibe o procedimento implementado para o projeto de arranjos físicos distribuídos.

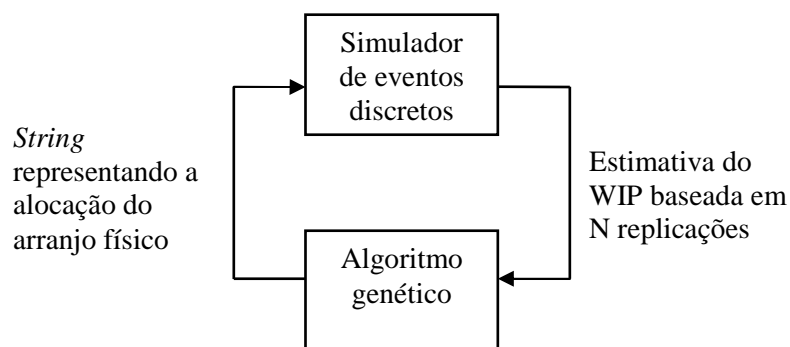


Figura 5: Procedimento para o projeto de arranjos físicos distribuídos baseado em otimização da simulação.

Para a implementação, utilizou-se o software de simulação discreta ProModel versão 7.0. Este *package* exibe uma interface ActiveX, que pode ser utilizada em qualquer linguagem que possua bibliotecas para lidar com componentes ActiveX. Neste caso, implementou-se um algoritmo genético baseado na biblioteca GALIB para linguagem C++, e na biblioteca de álgebra linear uBLAS, a qual compõe o conjunto de bibliotecas BOOST para C++. A aplicação utiliza métodos da biblioteca de tipos disponibilizada pelo objeto ActiveX do Promodel, e é compilado em .NET, no ambiente de desenvolvimento Visual C++ 2005 *express*, disponibilizado gratuitamente para fins acadêmicos.

### 3. Experimentações e resultados

Para a verificação do procedimento proposto, foi utilizado um problema teste de arranjo físico distribuído gerado aleatoriamente. O problema consiste em um chão-de-fábrica representado por um diagrama de blocos de 4 linhas por 5 colunas, configurando-se numa instância de tamanho 20. Um conjunto de 20 máquinas de 5 tipos diferentes (tabela 1) deve ser alocado a um conjunto de 20 locais disponíveis, e pretende-se obter uma alocação que possua baixo WIP. O cenário de demanda criado possui 5 famílias de peças, que são processadas de acordo com seus respectivos roteiro de produção. A tabela 2 indica os roteiros de produção e as demandas de cada família, enquanto na tabela 3 são indicados os tempos médios de produção para um lote. O tamanho dos lotes é considerado igual para todas as famílias, de tal forma que

esse fator não influenciará a análise. Para gerar variabilidade, admite-se que o processo de chegada de pedidos segue um processo de Poisson, e que os tempos de processamento em cada máquina seguem distribuições triangulares.

Tabela 1: Tipo de máquinas e os respectivos números de réplicas.

Tipo de máquina	Número de réplicas
1	5
2	2
3	4
4	6
5	3

Tabela 2: Roteiros de produção e demandas.

Família de produtos	Roteiro de produção	Demanda por unidade de tempo (fração)
P1	1 -> 4 -> 5-> 3 -> 2	0,45
P2	3 -> 4 -> 1->2	0,20
P3	5 -> 3->1	0,15
P4	1 -> 4 -> 3->5	0,10
P5	2 -> 1 -> 5->4	0,10

Tabela 3: Tempos de processamento. Os números entre parênteses indicam o número de réplicas de cada tipo de máquina.

Família de produtos	Máquina (tipo)				
	1(5)	2(2)	3(4)	4(6)	5(3)
P1	12	5	8	20	5
P2	8	8	10	15	0
P3	6	0	7	0	7
P4	10	0	12	13	9
P5	9	5	0	14	10

Nas simulações foi considerado apenas 1 transportador discreto, que em uma situação real pode ser uma empilhadeira ou paleteira. O valor de sua velocidade foi determinado por simulações iniciais de forma que o transportador tenha uma taxa de serviço suficiente para o sistema atingir o estado estacionário.

Outra questão importante na simulação de arranjos físicos distribuídos é a política de alocação de carga entre as diversas réplicas das máquinas. Visto que a tabela 2 especifica os roteiros de produção em termos apenas do tipo de máquina, e não de réplicas em particular, admite-se que todas as réplicas de um determinado tipo não apresentam diferenças significativas em termos de processamento. Logo, a decisão de qual réplica deve ser alocada para o processamento de um determinado lote depende da política de alocação de carga durante a operação do sistema real. Não obstante, adota-se aqui uma distribuição de carga uniforme entre as diversas réplicas, sem prejuízo da validade do modelo, visto que, em situações de alta utilização, as cargas de trabalho entre réplicas de um mesmo tipo de máquina se distribuem quase que uniformemente.

A simulação foi configurada para rodar um tempo de 140 horas, o que representa aproximadamente 1 mês de operação de um sistema de manufatura descontando-se intervalos. O

tempo de aquecimento considerado foi de 20 horas, para evitar que as estatísticas sejam influenciadas pelas condições iniciais do sistema (estado transiente). O número de replicações utilizado foi de 20.

O ambiente evolutivo do algoritmo genético é dado na tabela 4.

Tabela 4: Ambiente evolutivo

Parâmetros	Valor adotado
Tamanho da população	30
Número de gerações	20
Probabilidade de cruzamento	0,6
Probabilidade de mutação	1 e 0,01 a partir da segunda geração

A evolução do algoritmo genético é exibida na figura 6. Nela pode-se ver que há uma redução progressiva no valor do WIP nos arranjos físicos.

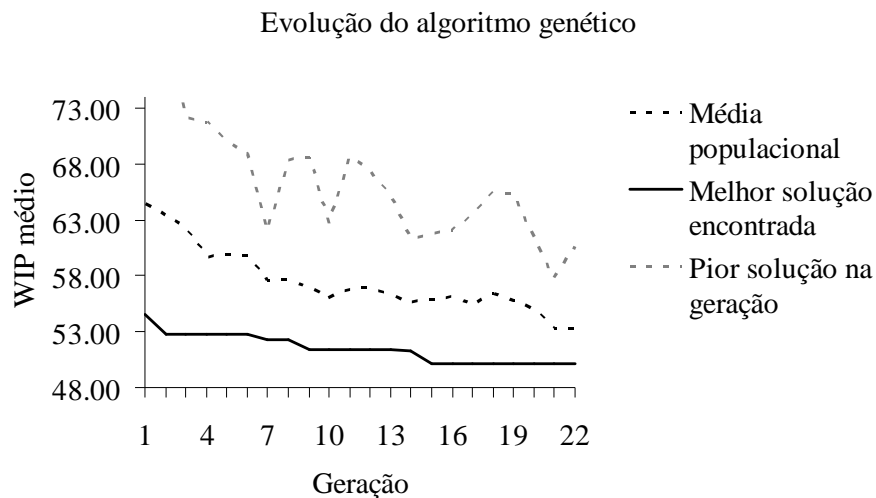


Figura 6: Evolução do algoritmo genético ao longo das gerações

**A melhor solução encontrada possui valor de WIP igual 50,2 lotes.** Ou seja, em estado estacionário, o melhor arranjo físico possui em média 50,2 lotes em processamento ou em espera no chão-de-fábrica. A *string* que representa o arranjo é:

17 14 19 1 15 2 16 20 10 5 13 9 6 18 8 12 4 11 7 3

O valor obtido representa uma melhoria de 22% em relação ao valor médio de WIP para a população inicial do algoritmo genético (64,5 lotes, em média), e uma melhoria de 42% em relação ao arranjo de pior WIP observado (86,7 lotes, em média) durante a evolução.

A figura 7 apresenta o melhor arranjo físico distribuído encontrado, enquanto a tabela 5 apresenta os dados referentes à figura.



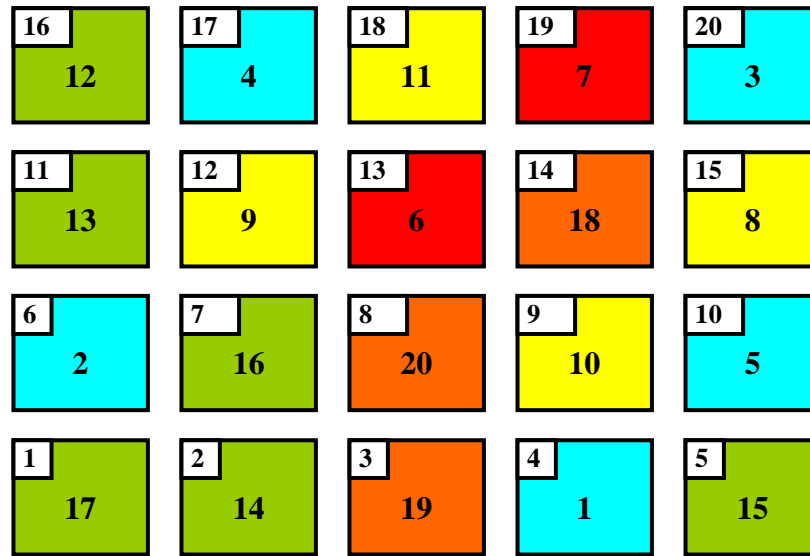


Figura 7: Arranjo físico distribuído de melhor WIP encontrado durante o processo evolutivo.

Tabela 5: Informações referentes à figura 7.

Tipo de máquina	Número de réplicas	Cor	Índices das réplicas
1	5	Azul	{1,2,3,4,5}
2	2	Vermelho	{6,7}
3	4	Amarelo	{8, 9,10,11}
4	6	Verde	{12, 13,14,15,16, 17}
5	3	Laranja	{18,19,20}

Pode-se observar que o arranjo físico obtido não se assemelha a um arranjo maximamente distribuído. De fato, algumas réplicas de máquinas foram completamente separadas, como no caso das máquinas do tipo 1 (réplicas 1,2,3,4 e 5) e tipo 3 (réplicas 8, 9, 10, 11), enquanto outras permaneceram reunidas funcionalmente, como no caso das máquinas tipo 4 (réplicas 12, 13, 14, 15, 16, 17) e tipo 5 (réplicas 18, 19 e 20). Observa-se que as máquinas do tipo 1 ficaram nas proximidades das concentrações funcionais de máquinas 4, e uma possível razão é a de que ambas são máquinas com alta carga (veja, por exemplo, Benjaafar (2002) sobre a proximidade de departamentos funcionais altamente carregados), e isso permitiria a minimização da distância percorrida pelo transportador, tanto em viagens com efetivo transporte de material, quanto em viagens em que o transportador se movimenta vazio. Isso daria alta fluidez ao fluxo entre esses 2 tipos de máquinas, diminuindo o tempo de atravessamento, e como consequência da Lei de Little, a diminuição do WIP. A concentração de máquinas tipo 4 pode se dever a um balanceamento adaptativo de capacidade, visto que os tempos de processamento neste tipo de máquina são maiores que os tempos de processamento nas máquinas do tipo 1. Um fenômeno semelhante é observado entre as máquinas do tipo 3 e 5, mas não tão fortemente, o que pode ser devido ao fato de serem máquinas menos carregadas, e pelo fato do número de máquinas do tipo 5 ser menor.

#### 4. Considerações finais

Neste artigo, procurou-se investigar o projeto de arranjos físicos distribuídos de baixo inventário com o uso de otimização da simulação e algoritmos genéticos. Apresentaram-se, em caráter introdutório, o problema de arranjo físico distribuído, visto como um problema quadrático de alocação, e como algoritmos genéticos podem ser utilizados para a exploração do espaço de soluções em busca de bons arranjos, avaliados pela simulação de eventos discretos.

Pode-se observar, pela evolução das soluções, que resultados significativos podem ser obtidos por meio da otimização da simulação de modelos de arranjos físicos. De fato, o melhor arranjo encontrado representa uma melhoria de até 42% em relação ao pior arranjo, o que significa que há grande possibilidade de geração de arranjos físicos distribuídos com bom desempenho operacional.

Muitas questões ainda precisam ser esclarecidas quanto ao projeto de arranjos físicos distribuídos, e sua implantação prática. Neste trabalho não foram abordadas questões como a influência do tamanho do lote de transferência e de produção, o número de transportadores, e a alocação de mão-de-obra. Também se fazem necessários estudos comparativos que demonstrem em que situações os arranjos distribuídos são mais vantajosos, e em quais ele se apresenta inviável.

Como trabalhos futuros, sugere-se o uso de simulação distribuída ou paralela, visto que a otimização da simulação é um procedimento computacionalmente exigente, e as soluções de instâncias maiores de problemas demandam alta capacidade de processamento para a obtenção de resultados em um tempo satisfatório. Também devem ser experimentadas outras técnicas promissoras, como a meta-heurística busca tabu, e métodos baseados em distribuições estatísticas, como o método da entropia cruzada ou *swarm intelligence*. Alguns esforços nessa direção já estão sendo empreendidos pelos autores.

#### Agradecimentos

Os autores gostariam de agradecer a CAPES, pela concessão de bolsas acadêmicas e pelo suporte às pesquisas desenvolvidas.

#### Referências

- Azadivar, F. e Tompkins, G.** (1999), Simulation optimization with qualitative variables and structural changes: a genetic algorithm approach, *European Journal of Operational Research*, 113, 169-182.
- Azadivar, F. e Wang, J.** (2000), Facility layout optimization using simulation and genetic algorithms, *International Journal of Production Research*, 38, 17, 4369-4383.
- Baykasoglu, A.** (2002), Capability-based distributed layout approach for virtual manufacturing cells, *International Journal of Production Research*, 41, 11, 2597-2618.
- Benjaafar, S.** (2002), Modeling and analysis of congestion in the design of facility layouts, *Management Science*, 48, 5, 679-704.
- Benjaafar, S. e Sheikhzadeh, M.** (2000), Design of flexible plant layouts, *IIE Transactions*, 32, 309-322.
- Buzacott, J. A. e Shanthikumar, J. G.**, *Stochastic Models of Manufacturing Systems*, Prentice Hall, Englewood Cliffs, New Jersey, EUA, 1993.
- Fu, M. C.** (2002), Optimization for simulation: theory vs. practice, *INFORMS Journal on Computing*, 14, 3, 192-215.
- Fu, M. C., Glover, F. W. e April, J.** (2005), Simulation optimization: a review, new developments, and applications, *Proceedings of the 2005 Winter Simulation Conference*, 83-95.
- Garey, M. R. e Johnson, D. S.**, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, New York, NY, EUA, 1979.
- Gen, M. e Cheng, R.**, *Genetic Algorithms and Engineering Design*, John Wiley & Sons, Inc, New York, NY, EUA, 1997.



- Goldberg, D. E.**, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Professional, 1989.
- Kusiak, A. e Heragu, S. S.** (1987), The facility layout problem, *European Journal of Operational Research*, 29, 229-251.
- Law, A. e Kelton, W. D.**, *Simulation Modeling and Analysis* 3<sup>a</sup> ed., McGraw-Hill, 2000.
- Montreuil, B., Venkatadri, U. e Lefrançois, P.** (1991), Holographic layout of manufacturing systems, *19th IIE Systems Integration Conference*, Orlando, EUA, 1-13.
- Nomden, G., Slomp, J. e Suresh, N.** (2006), Virtual manufacturing cells: a taxonomy of past research and identification of future research issues, *International Journal of Flexible Manufacturing Systems*, 17, 71-92.
- Pierreval, H. e Paris, J. L.** (2003), From ‘simulation optimization’ to ‘simulation configuration’ of systems, *Simulation Modeling Practice and Theory*, 11, 5-19.
- Singh, S. P. e Sharma, R. R. K.** (2006), A review of different approaches to the facility layout problem, *International Journal of Advanced Manufacturing*, 30, 425-433.
- Tam, K. Y.** (1992), Genetic algorithms, function optimization, and facility layout design, *European Journal of Operational Research*, 63, 322-346.
- Tompkins, J. A., White, J. A., Bozer, Y. A. e Tanchoco, J. M. A.**, *Facilities Planning* 3<sup>a</sup> ed., Wiley, 2002.