

## Otimização de *layout* de fábrica usando uma metaheurística de busca tabu

Valdair Candido Martins (PUCPR) [valdair@pppgia.pucpr.br](mailto:valdair@pppgia.pucpr.br)  
Marco Antônio Barbosa Cândido (PUCPR) [candido@ppgia.pucpr.br](mailto:candido@ppgia.pucpr.br)  
Leandro dos Santos Coelho (PUCPR) [lscelho@rla01.pucpr.br](mailto:lscelho@rla01.pucpr.br)

### Resumo

*Este artigo aborda uma solução computacional para um problema comumente encontrado no ambiente industrial: a geração do layout físico da fábrica onde algumas restrições (hard-constraints) não podem ser violadas. A inclusão destas restrições, embora aumente a complexidade do problema, permite lidar com o problema de layout de fábrica de maneira mais realista. O problema é solucionado em duas etapas. Na primeira, uma solução inicial baseada em heurística construtiva é gerada e na segunda, através da aplicação da metaheurística busca tabu, a solução inicial é otimizada.*

**Palavras Chaves:** *Layout*, Otimização, Busca Tabu.

### 1. Introdução

Como resultado direto da globalização de mercado observa-se uma competição acirrada entre as empresas de manufatura, levando-as a implementarem estratégias em busca da melhoria na qualidade e do aumento na produtividade dos bens produzidos. Entre as atividades importantes que influenciam na qualidade e na produtividade está o planejamento do *layout* da fábrica sob o qual o processo de fabricação será realizado. A escolha por *layouts* viáveis e eficientes permite que decisões, tais como a seleção de máquinas e equipamentos, ferramentas e operações, entre outras, sejam tomadas de forma mais apropriada. A geração de layouts mais eficientes é uma atividade bastante complexa (problema NP-Completo), segundo Garey (1979), onde erros humanos podem ser cometidos resultando em um aumento considerável nos custos de produção. Neste contexto, a aplicação de algoritmos eficazes para resolver o problema de *layout* de fábrica é de vital importância para otimizar a disposição de máquinas e pessoas numa empresa. Assim, abordagens como algoritmos genéticos (HOLLAND, 1975, GOLDBERG, 1989), *simulated annealing* (KIRKPATRICK, 1983) e busca tabu (GLOVER, 1989a, 1989b), estão sendo aplicadas com resultados promissores para a resolução do problema de layout. Para um maior detalhamento da busca tabu (BT) pode-se recorrer às referências citadas no final deste artigo (GLOVER, 1989a, 1989b, 1997).

As aplicações industriais recentes são numerosas e este assunto tem sido objeto de vários artigos de síntese (KUSIAK, 1987; MELLER, 1996). Entre os trabalhos pioneiros, destacam-se Koopmans (1957), Armour (1963), Nugent *et al.* (1968) e Vollman (1966).

Os *layouts* reais podem conter vários departamentos, e estes, sujeitos a um grande número de restrições (MARTINS *et al.*, 2003). Ao projetar o *layout*, o projetista deve considerar a estrutura das áreas de circulação (corredores, passagens, desníveis, entre outros), o sistema de movimentação de materiais, a escolha da localização das máquinas dentro dos departamentos e a localização dos pontos de entrada e/ou saída, entre outras. Os *layouts* podem ser construídos a partir de dados básicos, exemplos são o CORELAP (LEE, 1967), o ALDEP (SEEHOF, 1967), PLANET (APPLE, 1972), ou melhorados quando já existirem, como é o caso do CRAFT (ARMOUR, 1963) e o COFAD (TOMPKINS, 1976).

Este artigo está organizado da seguinte forma. Na seção 2, o escopo do problema é detalhado.

A abordagem de solução adotada é abordada na seção 3. Os resultados obtidos e conclusão são apresentados nas seções 4 e 5, respectivamente.

## 2. Escopo do problema

Uma solução é considerada viável quando não violar nenhuma das restrições a seguir:

- **Quadro externo (dimensões da fábrica):** O espaço destinado a conter todos os departamentos é retangular.
- **Área:** Os departamentos apresentam exigências individuais de área.
- **Formatos:** Os departamentos são modelados como blocos retangulares.
- **Orientação:** Livre (vertical ou horizontal) ou fixa (somente vertical ou somente horizontal).
- **Razão de aspecto:** A razão de aspecto  $a_i$  do departamento  $i$  é dada pelo comprimento  $C$  do departamento  $i$  dividido por sua largura  $L$ . Os departamentos com razão de aspecto variável permitem maior flexibilidade em seu formato retangular.
- **Sobreposição:** Em ambientes de piso único, os departamentos não podem se sobrepor.
- **Fixos e móveis:** Um departamento fixo terá sua localização conhecida *a priori*. Sendo móvel, o departamento poderá ocupar qualquer posição que não se sobreponha a outro.
- **Percentual de redução:** Uma localização disponível à atribuição de um departamento qualquer não poderá sofrer variações de área além do permitido.
- **Áreas inutilizadas (área morta):** (pilares, escadas, elevadores, etc). A existência destas áreas poderá distorcer a forma retangular da área útil do departamento  $i$ . Esta possível perda será amenizada pela razão de área morta  $\beta_i$  do departamento  $i$ , sendo definida como a área inutilizada dentro do departamento  $i$  dividida pela área total do departamento  $i$ .
- **Adjacência entre departamentos:** Em ambientes realistas, os departamentos são necessariamente adjacentes, isto é, precisam ficar próximos devido à dependência existente entre eles, ou são necessariamente não adjacentes, devendo ficar afastados por algum motivo. Os departamentos são considerados adjacentes quando suas fronteiras possuem pelo menos dois pontos em comum.
- **Adjacência entre departamento e fronteira do quadro externo:** Em muitos casos práticos, departamentos devem ficar na região fronteira do quadro externo. Em outras situações, não.

A última coluna da tabela 1 mostra o conjunto de restrições tratadas neste trabalho com o *software* AVOLI (MARTINS *et al.*, 2003) em relação as restrições tratadas por Tam (1991), Tam (1992), Tam (1998) e Chiang (2001).

Restrições	Tam (1991)	Tam (1992)	Tam (1998)	Chiang (2001)	AVOLI (este artigo)
adjacência/não adjacência					<b>x</b>
áreas desiguais	x	x	x	x	<b>x</b>
áreas ocupadas		x			<b>x</b>
departamentos fixos	x				<b>x</b>
departamentos móveis	x	x	x	x	<b>x</b>
quadro externo/delimitação		x	x	x	<b>x</b>
orientação (V/H)	x				<b>x</b>
razão de área morta		x			<b>x</b>
razão de aspecto	x	x	x		<b>x</b>
razão de redução					<b>x</b>
sobreposição	x	x	x	x	<b>x</b>
porte do problema tratado	30	30	20	40	<b>55</b>

Tabela 1 – Comparativo das restrições tratadas

### 2.1 Função objetivo

Em geral, a área de cada departamento e o custo de interconexão entre todos os pares de departamento é estimado a partir de dados básicos coletados por engenheiros de produção.

Este custo pode ser quantitativamente especificado em função do peso atribuído ao fluxo de material ou qualitativamente através de requerimentos de adjacência. O que se pretende aqui é minimizar o custo total de interconexão entre os departamentos fixos e móveis. Neste caso, a função objetivo é dada pela seguinte equação:

$$\underset{s \in S}{\text{Min}} \quad F = \sum_i^n \sum_j^n C_{ij} F_{ij} D_{ij} M^{(k+l+m)} \quad i, j = 1, 2, \dots, n, \text{ onde:}$$

$C_{ij}$ : custo do fluxo entre o departamento  $i$  e o departamento  $j$  por unidade de distância;

$F_{ij}$ : fluxo entre o departamento  $i$  e o departamento  $j$ ;

$D_{ij}$ : distância retangular centro-a-centro entre o departamento  $i$  e o departamento  $j$ ;

$K$ : número de departamentos não posicionados no *layout*;

$l$ : número de departamentos que violaram adjacência (perto);

$m$ : número de departamentos que violaram adjacência (longe);

$M$ : valor positivo definido em função do problema. Se restrições  $k$ ,  $l$  e/ou  $m$  forem violadas, a função de avaliação é altamente penalizada;

$n$ : tamanho do problema tratado.

As restrições de razão de aspecto e de área morta são definidas respectivamente por

$$a_{i(\text{min})} \leq a_i \leq a_{i(\text{máx})} \quad \text{e} \quad 0 \leq \beta_i \leq \beta_{i(\text{máx})}, \text{ onde:}$$

$a_i$ : razão de aspecto do departamento  $i$ .

$a_{i(\text{min})}$ ,  $a_{i(\text{máx})}$ : razão de aspecto mínima e máxima do departamento  $i$  respectivamente;

$\beta_i$ ,  $\beta_{i(\text{máx})}$ : razão de área morta e razão de área morta máxima do departamento  $i$ , respectivamente.

### 3. Abordagem de solução

O procedimento de solução é iniciado com o fornecimento dos dados específicos para cada problema. A escolha da seqüência de alocação dos departamentos no espaço bidimensional é realizada considerando-se a área e o relacionamento entre todos os departamentos, onde, primeiramente o maior departamento em área é inserido, seguido daquele com quem mantém o maior relacionamento. O próximo departamento a ser inserido na lista será o que mantiver o maior relacionamento com o último departamento inserido e assim sucessivamente. Caso o último departamento inserido na lista não mantenha relacionamento com nenhum outro, o próximo departamento a entrar na lista é o maior em área ainda não inserido. O procedimento continua até que todos os departamentos tenham sido inseridos. Primeiramente os departamentos do tipo “fixos” são atribuídos ao *layout* e em seguida os departamentos do tipo “móveis” provenientes da seqüência previamente gerada.

O primeiro departamento é alocado no espaço bidimensional a partir da primeira posição livre da esquerda para a direita e de cima para baixo. O segundo departamento é movimentado no sentido horário ao redor do último departamento já alocado até que se consiga atribuí-lo ao *layout* e assim sucessivamente. Se o próximo elemento da lista não puder ser alocado na vizinhança do seu antecessor, respeitando todas as restrições, então este departamento pode ser atribuído à próxima localização livre dentro do *layout*. O procedimento continua até que todos os departamentos tenham sido alocados. Caso algum departamento não seja atribuído ao *layout* ou viole alguma restrição, o procedimento é reiniciado pela reformulação da lista de alocação colocando-se o segundo maior departamento em área na segunda posição da lista, deslocando de uma posição para a direita dos demais elementos.

### 3.1 Melhoramento da solução inicial

A BT é usada como estratégia de refinamento da solução inicial com os seguintes parâmetros: (i) estrutura de memória, usada para armazenar o histórico da evolução do processo de busca, (ii) o critério de aspiração, usado para fazer o balanceamento, ou não, de uma nova configuração, com base nas informações armazenadas na estrutura de memória, e (iii) estratégias de diversificação e intensificação. Para um problema de minimização, pode-se generalizar a BT da seguinte forma:

- 1) Partir de uma solução inicial  $D$ ;
- 2) Inicializar parâmetros: lista tabu  $\leftarrow 0$ , MelhorSolução  $\leftarrow D$ ;
- 3) Enquanto um critério de terminação não for satisfeito faça: Avalie a lista de candidatos a movimento (candidato =  $\{D' \in V(D), D' \notin \text{lista tabu ou } D' \text{ satisfaz a um critério de aspiração}\}$ ): Selecione a melhor solução admissível,  $D^* \in \text{candidatos}$ ;
- 4) Atualizar a lista tabu e a solução corrente  $D \leftarrow D^*$ ; se  $F(D) < F(\text{melhor})$  então MelhorSolução  $\leftarrow D$ ;
- 5) Retornar a melhor solução.

Visão geral da BT utilizada:

- 1) Representação do problema: Lista de alocação ( $n$  elementos).
- 2) Vizinhança  $V(S)$ : a) Troca simples, b) Pseudotroca simples e c) Pseudotroca expandida.
- 3) Elementos da estrutura de controle:
  - a) Lista tabu dinâmica: i) Tempo tabu aleatório e ii) Tempo tabu variável.
  - b) Critério de aspiração: i) Um movimento deixa de ser tabu se levar a uma solução melhor que a melhor solução encontrada até o momento, ii) Se na vizinhança da solução corrente, todos os movimentos possíveis estão tabus, o que menos piora a função objetiva é liberado e iii) Se não ocorrer melhora na função objetiva em  $T$  iterações, libere todos os elementos tabus.
  - c) Critério de parada: i) Número pré-estabelecido de iterações.
- 4) Decodificação: Alocação dos departamentos.

### 3.2 Configurações da lista tabu

Foram realizados testes utilizando-se lista tabu dinâmica com duas variantes: i) Lista tabu aleatória: a cada iteração um número aleatório é escolhido dentro do intervalo especificado por  $T_{\min}$  e  $T_{\max}$  e atribuído ao elemento que acaba de se tornar tabu; ii) Lista tabu variável: semelhante a lista tabu aleatória, porém, se não ocorrer nenhuma melhora na função objetiva durante o intervalo de  $T$  iterações, todos os elementos são liberados do estado tabu.

### 3.3 Estrutura de vizinhança

Como exemplo considere a lista  $L = (A, B, C, D, E, F)$  formada pelos  $n$  departamentos a serem alocados. Usou-se três estruturas de vizinhança  $V(S)$ :

- 1) Troca simples:  $V(S)=6$ . As possíveis trocas são:  $A$  com  $B$ ,  $B$  com  $C$ ,  $C$  com  $D$ ,  $D$  com  $E$ ,  $E$  com  $F$  e  $F$  com  $A$ , escolhendo-se o melhor movimento admissível ou o movimento que satisfaça o critério de aspiração;
- 2) Pseudotroca simples: similar a troca simples, porém os movimentos não são efetivamente realizados, são apenas avaliados. Considera-se que todos os departamentos possuem a mesma área e formato, possibilitando as pseudotrocas entre si. Como as relações de fluxos entre os departamentos não mudam, o que realmente pode alterar os cálculos verdadeiros são as distâncias e os custos de transportes entre os departamentos. Assim, trabalha-se com as pseudodistâncias encontradas. Desta forma, avalia-se a pseudovizinhança sem a necessidade de alocar os departamentos para se encontrar as distâncias verdadeiras. Um subconjunto  $W$  de  $V(S)$  é testado, deste, e finalmente é escolhido para ser realizado o melhor movimento admissível;
- 3) Pseudotroca expandida:  $V(S)$  é maior, aqui o número de vizinhos é igual a  $V^*(V-1)/2$ , isto é,  $6*(6-1)/2=15$ . Utiliza a mesma lógica da pseudotroca simples, diferindo apenas no tamanho da vizinhança avaliada.

### 3.4 Estratégia de proibição e de liberação

A estratégia de proibição gerencia a inclusão na lista tabu classificando como proibidos os movimentos recém realizados. Após um movimento ter sido realizado, o atributo deste movimento é imediatamente colocado na lista tabu com um tempo tabu definido pelas configurações da lista tabu. Já a estratégia de liberação gerencia o que é retirado da lista tabu e em que momento isto ocorrerá, removendo as restrições tabus de um movimento de modo que ele possa participar em futuras soluções. Tanto na estratégia de proibição quanto na de liberação, os atributos de um movimento tabu permanecem na lista tabu durante  $T$  iterações, sendo liberados quando o tempo de permanência na lista se esgotar. O que se deve proibir e como? Proibiu-se o atributo do movimento (movimento, troca  $i$  com  $j$ ) através da regra de ativação tabu sentido do movimento (tanto o movimento na direção  $i$  para  $j$  quanto na direção  $j$  para  $i$ ), guardando-o na lista tabu. Por quanto tempo? Pelo tempo tabu, isto é, o tempo que o atributo permanece no estado tabu não podendo participar nas  $T$  soluções imediatas.

### 4. Resultados obtidos

Para validar os resultados usou-se o banco de problemas propostos por NUGENT *et al.* (1968). Os estudos de caso abordados são os problemas 5, 6, 7, 8, 12, 15, 20 e 30, com número máximo de 500 iterações. Os resultados obtidos foram comparados com os resultados de Tate (1995), Suresh (1995), Chiang (2001) e com as melhores soluções conhecidas para o problemas apresentados em NUGENT *et al.* (1968). A tabela 2 mostra os resultados encontrados nos testes. A última linha desta tabela mostra a que distância percentual estão as soluções encontradas das melhores soluções apresentadas na literatura. Para a execução dos testes utilizou-se o *software* AVOLI (MARTINS, 2002) em um computador Compaq S3200NX com 256 MB de memória RAM.

Total de departamentos	5	6	7	8	12	15	20	30
Total de iterações	100	100	100	100	200	200	500	500
Tempo total (s)	3	3	4	5	09	19	132	586
Melhor iteração	8	1	1	8	109	54	332	164
Melhor tempo (s)	0	0	0	0	04	04	88	191
Nugent <i>et al.</i> (1968)	50	86	148	214	578	1150	2570	6124
Tate (1995)	50	86	148	214	578	1150	2598	6184
Suresh (1995)	50	86	148	214	578	1150	2570	6168
Chiang (2001)	50	86	144	212	578	1110	2564	6094
Avoli	50	86	148	214	578	1150	2570	6128
Distância (%)	0,00	0,00	2,70	0,90	0,00	3,47	0,23	0,55

Tabela 2. Problemas benchmark - Análise comparativa das melhores soluções.

Como se observa na tabela 2, dos 8 problemas *benchmark*, 3 tiveram a melhor solução igualada, 3 tiveram uma solução menos de 1% inferior e 2 tiveram solução cerca de 3% inferior com tempos computacionais plenamente aceitáveis. Os resultados obtidos, quando comparados com as melhores soluções apresentadas, foram satisfatórios. Deve-se observar que os tempos computacionais apresentados neste trabalho referem-se ao tempo (em segundos) decorrido até a iteração referente à melhor solução encontrada.

Propôs-se um problema realista com 55 departamentos (50 “móveis” e 5 “fixos”). Este ambiente de teste possui 5 áreas ocupadas. Os experimentos foram realizados considerando-se as seguintes restrições: os departamentos 8, 9, 19 e 29 devem ficar na vertical, os departamentos 10 e 17 devem ficar na horizontal, o departamento 9 deve ficar perto do 47, o departamento 15 deve ficar longe do 19, o 23 deve ficar longe do departamento fixo 3 e o 14 deve ficar longe do departamento fixo 4. O departamento 20 deve ficar na borda do quadro

externo. O teste foi executado num total de 500 iterações. A última linha da tabela 3 apresenta o percentual de melhoria da solução final em relação à solução inicial.

Através deste problema é comprovada a eficácia do modelo heurístico utilizado na solução de um problema de grande porte. A melhor combinação entre vizinhanças e tempo tabus foi detectada quando se considerou  $V(S)$  pseudotroca expandida associada a lista tabu variável. A escolha mais adequada de  $T_{min}$  e  $T_{max}$ , permitiu balancear a exploração do espaço de soluções intensificando ou diversificando ao se restringir fortemente os atributos de movimento. A escolha mais apropriada de  $V(S)$ , bem como o tamanho mais adequado da lista tabu, permitiu alcançar bons resultados a constatar pelo percentual de melhoria atingido.

A melhor solução obtida quando comparada com a solução inicial apresentou uma melhoria de 33,85%. Observa-se que a escolha mais promissora da lista de candidatos à troca serve para restringir o número de elementos a serem analisados. Desta forma, pôde-se usar uma vizinhança reduzida sem comprometer a qualidade da solução.

total de departamentos	55
quadro externo (m2)	90x60
lista tabu variável	$T_{min}$ (7) e $T_{max}$ (14)
$V(S)$ expandida	20
total de iterações	500
tempo total (s)	12700
solução inicial	187770
solução final	124213
melhor iteração	216
melhor tempo (s)	5206
melhoria (%)	33,85

Tabela 3. Desempenho para o problema proposto.

A interface de trabalho do *software* AVOLI encontra-se no anexo A na figura 1. Os dados de entrada para o problema proposto nas tabelas 4 a 7 e os *layouts* correspondentes à solução inicial e à melhor solução final são mostrados nas figuras 2 e 3, respectivamente.

## 5. Conclusões

Este artigo apresentou um procedimento de otimização heurístico para solução de problemas de *layout* industrial que considera como *hard-constraints*, as restrições geométricas e as de localização frequentemente encontradas em problemas reais e não tratadas adequadamente em outros modelos.

A abordagem de solução utilizada foi dividida em 2 etapas, sendo que na primeira obteve-se, através de uma heurística construtiva uma solução inicial viável não violando nenhuma das restrições impostas, e na segunda etapa obteve-se uma solução melhorada através da utilização da busca tabu com as devidas adaptações. Duas estruturas de vizinhança foram implementadas para o problema, tomando por comparação problemas com soluções já demonstradas na literatura. Através dos testes realizados observou-se que as dimensões do quadro externo podem influenciar diretamente nos resultados obtidos, portanto, a escolha mais adequada de suas dimensões, pode produzir melhores resultados. As soluções obtidas pelo modelo heurístico se mostraram tão boas quanto as melhores soluções conhecidas para problemas *benchmark*. Um problema de maior porte foi proposto e tratado em tempo computacional aceitável, porém não foi possível fazer nenhuma comparação de resultados, uma vez que problemas semelhantes não foram encontrados na literatura.

## Referências

APPLE, J. M. & DEISENROTH, M. P. (1972) - A computerized plant layout analysis and evaluation technique

- (PLANET). *Proceedings of the 23rd Annual Conference and Convention*, AIIE, California, pp. 121-127.
- ARMOUR, G. C. & BUFFA, E. S. (1963) - A heuristic algorithm and simulation approach to relative allocation of facilities, *Management Science*, vol. 9, pp. 294-309.
- CHIANG, W. C. (2001) - Visual facility layout design system, *International Journal Production Research*, vol. 39, no. 9, pp. 1811-1836.
- GAREY, M. R. & JOHNSON, D. S. (1979) - *Computers and intractability: a guide to the theory of np-completeness*. W. H. Freeman and Company, New York.
- GLOVER, F. (1989a) - Tabu search - part I, *ORSA Journal on Computing*, vol. 1, no. 3, pp. 190-206.
- GLOVER, F. (1989b) - Tabu search - part II, *ORSA Journal on Computing*, vol. 2, no. 1, pp. 4-32.
- GLOVER, F. & LAGUNA M. (1997) - *Tabu search*, Kluwer Academic Publishers. Massachusetts.
- GOLDBERG, D. E. (1989) - *Genetic algorithms in search, optimization and machine learning*. Addison Wesley, New York, NY, USA.
- HOLLAND, J. H. (1975) - *Adaptation in natural and artificial systems*, University of Michigan, Ann Arbor, MI.
- KIRKPATRICK, S.; GELLAT, C. D. JR. & VECCHI, M. P. (1983) - Optimization by simulated annealing, *Science*, vol. 220, pp. 671-680.
- KOOPMANS, T. C. & BECKMAN, M. (1957) - Assignment problems and the location of economic activities. *Econometrica*, vol. 25, no. 53, pp. 53-76.
- KUSIAK, A. & HERAGU, S. S. (1987) - The facility layout problem, *European Journal of Operational Research*, vol. 29, pp. 229-251.
- LEE, R. C. & MOORE, J. M. (1967) - CORELAP - Computerized relationship layout planning. *Industrial Engineering*, vol. 18, pp. 195-200.
- MARTINS, V. C. (2002) - VCM-AVOLI Industrial Model. *Patent at INPI*, Register Number 00048682, Brasil.
- MARTINS, V. C., COELHO, L. S., *et al.* (2003) - Otimização de layouts industriais com base em busca tabu., *Gestão & Produção*, vol. 10, no. 1, p. 69-88. ISSN 0104-530X.
- MELLER, R. D. & GAU, K. -Y. (1996) - The facility layout recent and emerging trends and perspectives, *Journal of Manufacturing Systems*, vol. 15, no. 5, pp. 351-366.
- NUGENT, C. E., VOLLMANN, T. E. & RUMML, J. (1968) - An experimental comparison of techniques for the assignment of facilities to locations. *Operations Research*, vol. 16, pp. 150-173.
- SEEHOF, J. M. & EVANS, W. O. (1967) - Automated layout design program, *Industrial Engineering*, vol. 18, pp. 690-695.
- SURESH, G. & SAHU, S. (1995) - A genetic algorithm for facility layout, *International Journal of Production Research*, vol. 33, no. 12, pp. 3411-3423.
- TAM, K. Y. & LI, S. G. (1991) - A hierarchical approach to the facility layout problem, *International Journal of Production Research*, vol. 29, pp. 165-184.
- TAM, K. Y. (1992) - Genetic algorithms, function optimization, and facility layout design. *European Journal of Operational Research*, vl. 63, pp. 322-346.
- TAM, K. Y. & CHAN, S. K. (1998) - Solving facility layout problems with geometric constraints using parallel genetic algorithms: experimentation and findings. *International Journal of Production Research*, vol. 36, no. 12, pp. 3253-3272.
- TATE, D. M. & SMITH, A. E. (1995) - A genetic approach to the quadratic assignment problem. *Computers and Operations Research*, vol. 32, pp. 73-83.
- TOMPIKINS, J. A. & REED JR, R. (1976) - An applied model for the facilities design problem, *International Journal of Production Research*, vol. 14, no. 5, pp. 583-595.
- VOLLMAN, T. E. & BUFFA, E. S. (1966) - Facilities layout problem in perspective, *Management Science*, vol. 12, pp. 450-468.

## ANEXO

O sinal “-” identifica departamento fixo e o sinal “+” identifica departamento móvel.

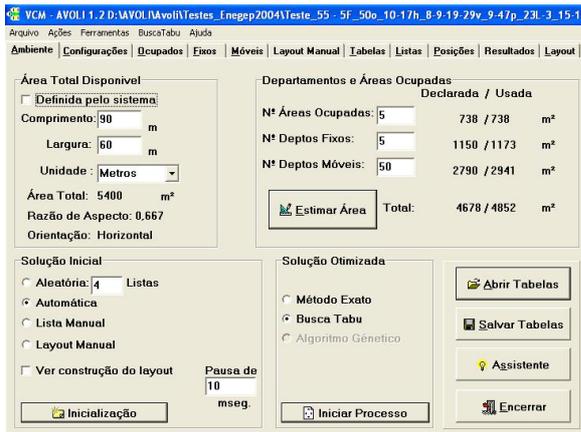


Figura 1 – Interface de trabalho do AVOLI.

Ambiente	Configurações	Ocupados	Fixos	Móveis	Layout Manual	Tabelas	Posições	Resultados	Layout
-1	60	28	1			0	.5	1	
-2	450	69	36			0	1	1	
-3	360	41	1			0	1	1	
-4	200	1	47			0	1	1	
-5	90	42	38	h		0	.6	1,2	

Tabela 4 – Geometria dos departamentos fixos.

Ambiente	Configurações	Ocupados	Fixos	Móveis	Layout Manual	Tabelas	List
1	38	1	37	3			
2	41	35	3	14			
3	60	1	12	31			
4	52	38	23	3			
5	55	57	4	36			

Tabela 5 – Geometria das áreas ocupadas.

Ambiente	Configurações	Ocupados	Fixos	Móveis	Layout Manual	Tabelas	Listas	Posições	Resultados	Layout
+1	100	0	0,7	1,0	0					
+2	80	0	1	1	0					
+3	50	0	0,7	1,3	0					
+4	60	0	0,5	0,8	0					
+5	120	0	0,9	1	0					
+6	40	0	0,6	1	0					
+7	20	0	0,7	1,4	0					
+8	40	v	1	1	0					
+9	150	v	0,8	1,1	0					
+10	120	h	0,5	1,5	0					
+11	50	0	0,7	1,1	0					
+12	10	0	0,6	1,2	0					
+13	20	0	0,95	1,5	0					
+14	30	0	0,75	1,25	0					
+15	50	0	0,9	1,1	0					
+16	20	0	0,8	1,5	0					
+17	40	h	0,4	1,4	0					
+18	20	0	0,9	1,3	0					
+19	80	v	1	1	0					
+20	100	0	0,95	1,15	0					
+21	40	0	0,5	1,5	0					
+22	50	0	0,9	1,1	0					
+23	80	0	0,6	1	0					
+24	10	0	0,9	1	0					
+25	40	0	0,8	1,1	0					
+26	10	0	0,5	1,2	0					
+27	40	0	0,8	1	0					
+28	10	0	0,5	1,3	0					
+29	80	v	0,7	1,05	0					
+30	40	0	0,9	1,1	0					
+31	40	0	.9	1,2	0					
+32	80	0	.9	1,5	0					
+33	10	0	.9	1,1	0					
+34	40	0	.8	1	0					
+35	10	0	.8	1,3	0					
+36	50	0	.8	1,2	0					
+37	20	0	.8	1,5	0					
+38	40	0	.9	1,6	0					
+39	50	0	.9	1,1	0					
+40	60	0	1	1,8	0					
+41	150	0	.7	1,6	0					
+42	210	0	.3	1	0					
+43	110	0	.5	1	0					
+44	60	0	.85	1,6	0					
+45	45	0	.9	1,2	0					
+46	30	0	.7	1,05	0					
+47	80	0	.95	1,15	0					
+48	10	0	.8	1,1	0					
+49	95	0	.75	1,2	0					
+50	30	0	.6	1	0					

Tabela 6 – Geometria dos departamentos móveis.

Ambiente	Configurações	Ocupados	Fixos	Móveis	Layout Manual	Tabelas	Listas	Posições	Resultados	Layout
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0	0
35	0	0	0	0	0	0	0	0	0	0
36	0	0	0	0	0	0	0	0	0	0
37	0	0	0	0	0	0	0	0	0	0
38	0	0	0	0	0	0	0	0	0	0
39	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0
41	0	0	0	0	0	0	0	0	0	0
42	0	0	0	0	0	0	0	0	0	0
43	0	0	0	0	0	0	0	0	0	0
44	0	0	0	0	0	0	0	0	0	0
45	0	0	0	0	0	0	0	0	0	0
46	0	0	0	0	0	0	0	0	0	0
47	0	0	0	0	0	0	0	0	0	0
48	0	0	0	0	0	0	0	0	0	0
49	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0

Tabela 7 – Matriz de fluxos simétrica.

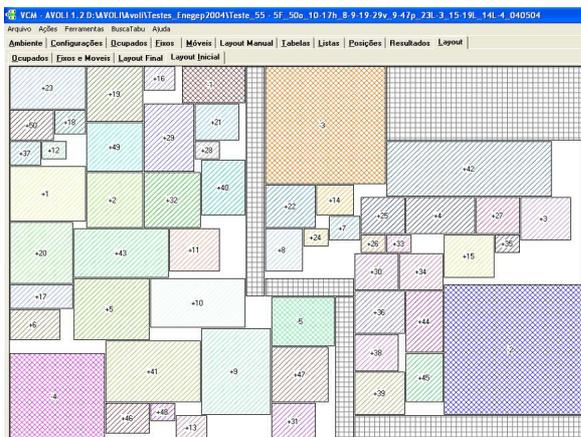


Figura 2 – Layout inicial - 55 departamentos.

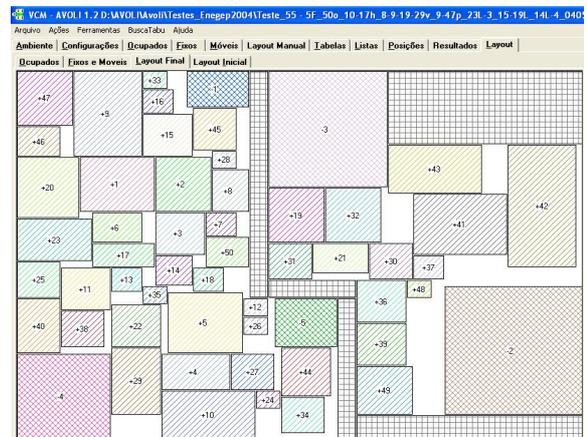


Figura 3 – Layout final - 55 departamentos.