



OTIMIZAÇÃO DE LAYOUTS INDUSTRIAIS COM BASE EM BUSCA TABU

Valdair Candido Martins

Programa de Pós-graduação em Informática Aplicada,
Pontifícia Universidade Católica do Paraná,
Rua Imaculada Conceição, 1155,
CEP 80215-901, Curitiba, PR, Brasil,
e-mail: valdir@ppgia.pucpr.br;

Leandro dos Santos Coelho

Marco Antonio Barbosa Cândido

Ricardo Ferrari Pacheco

Pontifícia Universidade Católica do Paraná,
Rua Imaculada Conceição, 1155,
CEP 80215-901, Curitiba, PR, Brasil,
Centro de Ciências Exatas e de Tecnologia, CCET,
Programa de Pós-graduação em Engenharia de
Produção e Sistemas, PPGEPS, Grupo Produtrônica,
emails: lscoelho@rla01.pucpr.br,
candido@ppgia.pucpr.br e rpacheco@ppgia.pucpr.br

Resumo

Este artigo aborda uma solução computacional para o problema de layout industrial considerando hard-constraints não tratadas em trabalhos anteriores. O problema é solucionado em duas etapas. Na primeira, uma solução inicial baseada em heurística construtiva é gerada e na segunda, por intermédio da aplicação da meta-heurística busca tabu, a solução inicial é melhorada. Como contribuição, apresenta-se a ferramenta computacional denominada Ambiente Visual para Otimização de Layout Industrial (AVOLI), a fim de gerar layouts viáveis e eficientes capazes de tratar problemas de grande porte sujeitos a um conjunto de restrições realistas.

Palavras-chave: layout, otimização, hard-constraints, busca tabu.

1. Introdução

A aplicação de metodologias eficazes de projeto e otimização para resolução do problema de FLD (*facility layout design*) é de vital importância para melhorar a disposição de equipamentos e funcionários em uma empresa. Nesse contexto, aplicações industriais e diversas metodologias para o problema de FLD têm sido abordadas na literatura (Kusiak & Heragu, 1987; Meller & Gau, 1996). Entre os trabalhos pioneiros, destacam-se Koopmans & Beckman (1957), Armour & Buffa (1963), Nugent *et al.* (1968) e Vollman & Buffa (1966), cujo foco principal era otimizar a localização relativa dos departamentos em um único pavimento, formulando o problema do layout de fábrica como um problema de atribuição quadrática (do inglês, *quadratic assignment problem* – QAP).

O QAP é um problema NP-Completo (Garey & Johnson, 1979), em geral, de difícil solução, ou seja, não há algoritmo capaz de encontrar a solução ótima de um problema de grandes proporções em tempo polinomial. O QAP pode ser interpretado da seguinte maneira: suponha que m departamentos precisam ser atribuídos a n localizações, de modo que, para cada localização j , seja atribuído um único departamento i e, da mesma forma, que cada departamento i seja atribuído a uma única localização j . Para representar a interação entre os departamentos, um peso positivo é associado a cada par de departamentos. Assim, o problema consiste em atribuir cada departamento a uma localização diferente de todos os demais, de forma que a soma dos pesos vezes as distâncias entre todos os departamentos seja a mínima possível (em problemas de minimização).

O QAP foi introduzido para modelar o problema de alocar e interagir plantas de áreas e formatos iguais (Koopmans & Beckman, 1957), podendo ser aplicado a problemas que incluem planejamento urbano, layout de painéis de controle e projetos de redes de energia (Bazaraa, 1975). Entre as abordagens usadas para tratar o problema do layout de departamento, o QAP tem sido muito utilizado (Chiang, 2001).

O problema de layout tem sido modelado por diferentes abordagens, destacando-se a teoria de grafos (Foulds & Robinson, 1978), o MIP (*mixed integer program*) (Montreuil, 1990; Montreuil *et al.*, 1993; Heragu & Kusiak, 1991) e a abordagem que utiliza estrutura de árvore binária, apresentada por Tam & Li (1991) e, posteriormente, empregada por Furtado & Lorena (1997) e Azadivar & Wang (2000). Os layouts presentes no meio industrial podem conter vários departamentos sujeitos a um grande número de restrições. Ao projetar o layout, o projetista deve considerar entre outros fatores: (a) a estrutura das áreas de circulação (corredores, passagens, desníveis e outros), (b) o sistema de movimentação de materiais, (c) a escolha da localização das máquinas dentro dos departamentos e (d) a localização dos pontos de entrada/saída. Os layouts podem ser construídos a partir de dados básicos (algoritmos construtivos) ou melhorados quando já existem (algoritmos de melhoramento).

Algoritmos como CORELAP (Lee & Moore, 1967), ALDEP (Seehof & Evans, 1967) e PLANET (Apple & Deisenroth, 1972), baseados em heurística construtiva, atribuem os departamentos à área destinada ao layout, um de cada vez, até obter a solução final. Sob outro ponto de vista, os algoritmos de melhoramento necessitam de uma solução inicial a partir da qual se inicia o procedimento de otimização. Dentre os algoritmos de melhoramento, o CRAFT (Armour & Buffa, 1963) e o COFAD (Tompkins & Reed Jr., 1976) são exemplos conhecidos (Heragu & Kusiak, 1987). Nesse contexto, a combinação de algoritmos construtivos e algoritmos de melhoramento pode resultar em abordagens promissoras de algoritmos híbridos (Elshafei, 1977; Scriabin & Vergin, 1985), e, assim, em melhor compromisso entre precisão da solução, complexidade computacional, propriedades de convergência, robustez e flexibilidade da metodologia de otimização adotada.

Outras abordagens, como algoritmos genéticos (Holland, 1975), *simulated annealing* (Kirkpatrick *et al.*, 1983) e busca tabu (Glover, 1989a, b), também têm sido aplicadas com sucesso para resolver problemas de otimização e de layout.

Os ambientes computacionais propostos na literatura, tanto os construtivos quanto os de melhoramento, atuam fundamentalmente com o objetivo de minimizar o custo de manipulação de materiais e/ou maximizar a proximidade dos departamentos. Em uma concepção de projeto mais atual, há o software FACTORY (Cimtechnologies, 1989), que atua conjuntamente com o AUTOCAD (Coraini & Nolla, 1994), representando um avanço em relação às ferramentas anteriormente desenvolvidas. Embora o problema tenha sido amplamente estudado, ainda são poucos os trabalhos que consideram restrições realistas.

Este artigo propõe a solução do problema de layout industrial considerando um conjunto abrangente de restrições, para isso utiliza uma abordagem de solução com base em busca tabu. Está organizado da seguinte maneira: a descrição e as características do problema são apresentadas na Seção 2. A abordagem de otimização de layout proposta é detalhada na Seção 3. Na Seção 4, os resultados de simulação são apresentados e discutidos. Finalmente, a conclusão e as perspectivas de futuros trabalhos são tratadas na Seção 5.

2. Escopo do problema

Uma solução viável é aquela que não viola nenhuma das restrições impostas. Considerou-se o seguinte conjunto de restrições:

1. *Quadro externo*: (dimensões da fábrica) o espaço destinado a conter todos os departamentos é retangular. Todos os departamentos devem ser alocados dentro da fábrica.
2. *Área*: os departamentos apresentam exigências individuais de área física.
3. *Formatos*: os departamentos são modelados como blocos retangulares ou quadrados.
4. *Orientação*: livre (vertical ou horizontal) ou fixa (somente vertical ou somente horizontal).
5. *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 . Departamentos com razão de aspecto variável permitem maior flexibilidade em seu formato retangular.
6. *Sobreposição*: em ambientes de piso único, os departamentos não podem se sobrepor.
7. *Fixos e móveis*: departamentos do tipo “fixos” a priori possuem localização conhecida. Sendo móvel, o departamento pode ocupar qualquer local ainda não utilizado.
8. *Porcentual de redução*: um local disponível à atribuição de um departamento qualquer não poderá sofrer variações de área além do permitido.
9. *Áreas inutilizadas (área morta)*: é comum no interior da fábrica haver regiões impedidas de serem utilizadas (pilares, escadas, elevadores, etc.). Essas áreas poderão distorcer a forma retangular da área útil do departamento i . Uma medida que represente essa possível perda de área é necessária. A razão de área morta β_i do departamento i é introduzida com esse propósito, sendo definida como a área inutilizada dentro do departamento i dividida pela área total do departamento i .
10. *Adjacência entre departamentos*: em muitos casos práticos, os departamentos são necessariamente adjacentes, isto é, precisam ficar próximos em razão da dependência existente entre eles. Em outras situações, os departamentos são necessariamente não adjacentes, devendo ficar afastados por algum motivo. Os departamentos são considerados adjacentes quando suas fronteiras apresentam ao menos dois pontos em comum.
11. *Adjacência entre departamento e fronteira do quadro externo*: em casos práticos, os departamentos podem se localizar na região fronteira do quadro externo. Em outras situações, não.
12. *Distância*: duas formas usuais de mensurar a distância entre dois pontos são: distância retangular e distância euclidiana (Tam & Li, 1991; Askin & Standridge, 1993). Se fixarmos o departamento i na localização $P_i = (x_i, y_i)$ e o departamento j na

localização $P_j = (x_j, y_j)$, então pode-se definir a distância entre os departamentos i e j por:

$$d_{ij}^R = |x_i^c - x_j^c| + |y_i^c - y_j^c| \quad (1)$$

distância retangular;

$$d_{ij}^E = \sqrt{(x_i^c - x_j^c)^2 + (y_i^c - y_j^c)^2} \quad (2)$$

distância euclidiana.

Para estimar a distância entre os departamentos i e j , no caso de layout de departamentos, usa-se distância centro a centro (euclidiana ou retangular), quando os pontos de entrada e saída não são conhecidos.

A Tabela 1 mostra o comparativo de restrições tratadas pelo AVOLI com as restrições tratadas por Tam (1991, 1992), Tam & Chan (1998), Furtado

& Lorena (1997) e Chiang (2001). Pode-se observar na Tabela 1 que o conjunto de restrições realistas considerado pelo AVOLI incorpora as restrições já tratadas em trabalhos anteriores. Outro aspecto relevante diz respeito ao maior porte dos problemas considerados e ao tratamento de todas as restrições em conjunto no mesmo problema.

O maior problema considerado por Chiang (2001) conta com 40 departamentos, em que se considera um conjunto composto por quatro restrições (este artigo não traz os dados necessários para uma efetiva comparação de resultados), ao passo que aqui se considerou um problema com 40 departamentos e outro com 55 departamentos sujeitos ao conjunto de restrições apresentado na Tabela 1. Todos os dados de entrada necessários à resolução desses problemas encontram-se descritos no Anexo A.

Tabela 1 – Comparativo das restrições tratadas na abordagem proposta em relação às apresentadas na literatura.

Restrições	Tam (1991)	Tam (1992)	Furtado & Lorena (1997)	Tam & Chan (1998)	Chiang (2001)	AVOLI
Adjacência/não adjacência						X
Áreas desiguais	X	X	X	X	X	X
Áreas ocupadas		X	X			X
Departamentos fixos	X					X
Departamentos móveis	X	X	X	X	X	X
Quadro externo		X	X	X	X	X
Orientação (vertical/horizontal)	X					X
Razão de área morta		X	X			X
Razão de aspecto	X	X	X	X		X
Razão de redução						X
Sobreposição	X	X	X	X	X	X
Quantidade de departamentos	30	30	30	20	40	55

2.1 Função objetivo

Em geral, a área de cada departamento e o custo de interconexão entre todos os pares de departamentos são estimados a partir de dados coletados por engenheiros de produção. O custo de interconexão também pode ser especificado quantitativamente por intermédio de um peso atribuído ao fluxo de material ou pode ser medido qualitativamente por requerimentos de adjacência. A função objetivo visa minimizar o custo total de interconexão entre todos os departamentos fixos e móveis e é regida pela equação:

$$\text{Min}_{s \in S} F = \sum_i^n \sum_j^n C_{ij} F_{ij} D_{ij} M^{(k+l+m)}$$

em que $i, j = 1, 2, \dots, n$ (3)

sendo:

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 e alto definido em função do problema. Se restrições k , l e/ou m forem violadas, a função de avaliação será rigorosamente penalizada;

n : porte do problema tratado, ou seja, número de variáveis a serem otimizadas.

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

$$a_{i(\text{mín})} \leq a_i \leq a_{i(\text{máx})} \text{ e } 0 \leq \beta_i \leq \beta_{i(\text{máx})}, \text{ em que:}$$

a_i : razão de aspecto do departamento i ;

$a_{i(\text{mín})}$: razão de aspecto mínima do departamento i ;

$a_{i(\text{máx})}$: razão de aspecto máxima do departamento i ;

β_i : razão de área morta do departamento i ;

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

3. Abordagem de solução

O processo de solução é iniciado com o fornecimento dos dados específicos para cada problema. Os principais procedimentos utilizados são:

- *Área total necessária (ATN)*: sugere as dimensões do quadro externo.
- *Solução inicial viável (SIV)*: gera uma solução de partida factível.
- *Algoritmo matriz tabu (AMT)*: gerencia a lista de elementos tabus.
- *Algoritmo de busca tabu (ABT)*: busca a melhor solução na vizinhança atual.
- *Algoritmo de alocação (VCM)*: executa a alocação física dos departamentos dentro do quadro externo.

3.1 Obtenção da solução inicial

O procedimento SIV inicia construindo a lista de alocação que define, para o algoritmo VCM, a seqüência de alocação dos departamentos do tipo “fixos” e do tipo “móveis” no espaço bidimensional. Na lista, primeiramente é inserido o departamento com maior área, seguido daquele com o qual mantém mais relacionamento. Em seguida, o próximo departamento a ser inserido na lista será aquele que mantiver o maior relacionamento com o último departamento inserido e assim sucessivamente. Se o último departamento inserido na lista não se relaciona com nenhum outro departamento, o próximo departamento candidato a entrar na lista será o

maior em área, que ainda não se encontra na lista de alocação. O processo continua até que todos os departamentos sejam inseridos. Com a lista já construída, inicia-se o processo de alocação física dos departamentos.

O algoritmo VCM primeiramente aloca os departamentos do tipo “fixos” e, em seguida, os do tipo “móveis”, escolhendo a melhor localização dentro da fábrica e evitando que as restrições impostas sejam violadas.

O primeiro departamento alocado ao layout é o primeiro elemento da lista, o segundo é o segundo elemento da lista e assim sucessivamente. O primeiro departamento é alocado a partir da primeira posição livre da esquerda para a direita e de cima para baixo. O próximo elemento da lista é posicionado à direita do departamento anteriormente alocado ao layout. Caso não seja possível essa atribuição, a próxima alternativa para alocar o departamento corrente ao layout inicia-se pelo deslizamento do departamento à direita do último departamento já posicionado. Caso não seja possível alocá-lo à direita, tenta-se alocá-lo abaixo, se não for possível, à esquerda e se, mesmo assim, ainda não for possível, tenta-se alocá-lo acima, sempre deslizando o departamento de uma unidade de comprimento ao redor do último departamento posicionado no layout. Se o próximo elemento da lista não puder ser alocado na vizinhança de seu antecessor, respeitando todas as restrições, então esse departamento poderá 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, obrigando o deslocamento de uma posição para a direita dos demais elementos. Os departamentos maiores têm prioridade no início da lista pelo fato de que, ao alocar primeiramente os departamentos de maior área, os menores são mais facilmente alocados nos espaços livres entre os demais já

atribuídos ao layout, conseqüentemente, há maior chance de conseguir alocar todos os departamentos. Se em cada tentativa ocorrer alguma violação, a lista de alocação será reformulada até que todas as m possibilidades tenham sido testadas ($m = df + dm$, em que df = total de departamentos do tipo “fixos” e dm = total de departamentos do tipo “móveis”). Em todos os testes realizados a solução inicial escolhida foi sempre uma solução viável, ou seja, não violou nenhuma das restrições impostas.

3.2 Melhoramento da solução inicial

Com o intuito de melhorar o resultado inicialmente encontrado, utilizou-se a busca tabu (BT) como estratégia de refinamento. A BT é um método heurístico aplicado com sucesso a diversos problemas de otimização combinatória (Glover & Laguna 1997); foi proposta inicialmente por Glover (1989a, b) para resolução de problemas de otimização, sendo um procedimento adaptativo que pode guiar um algoritmo de busca local na exploração contínua do espaço de busca, sem ser encerrado prematuramente pela ausência de vizinhos que melhorem a solução atual.

A BT explora a vizinhança de uma dada solução e seleciona a melhor solução, mesmo que esta piore a solução atual. Essa estratégia permite que a busca escape de um ótimo local e explore outra parcela do espaço de soluções. Se ocorrer retornos a um ótimo local previamente visitado (condição desejada, mas não necessária), a BT, por meio de seus mecanismos de controle, permite que a exploração do espaço de soluções prossiga, evitando o efeito de “ciclagem”. Se um movimento está presente na lista tabu, ele poderá ser aceito somente se minimizar o valor da função objetivo.

O processo no qual a BT transcende a otimalidade local se baseia em uma função de avaliação, que escolhe a cada iteração o movimento com o maior valor de avaliação na vizinhança. Para tornar a busca mais flexível, classifica-se um movimento de tabu para não-tabu por algum critério de aspiração. Esse critério

libera um movimento de seu estado tabu antes que seu tempo tabu termine. Um movimento será aceito quando suas restrições tabus não forem violadas ou quando algum critério de aspiração retirar seu estado tabu. Os parâmetros básicos da BT (Glover, 1989a, b; Glover & Laguna, 1997) são: (a) a estrutura de memória, (b) o critério de aspiração e (c) o critério de terminação. Os passos genéricos da otimização com base em BT são:

1. Partir de uma solução inicial D .
2. Iniciar os parâmetros lista tabu $\leftarrow 0$ e *Melhor Solução* $\leftarrow D$.
3. Enquanto um critério de terminação não for satisfeito, avaliar a lista de candidatos a movimento (candidato = $\{D' \in N(D), D' \notin \text{lista tabu ou } D' \text{ satisfaz a um critério de aspiração}\}$ e selecionar a melhor solução admissível; $D^* \in \text{soluções candidatas}$.
4. Atualizar a lista tabu; atualizar a solução corrente $D \leftarrow D^*$; se $F(D) < F(\text{Melhor Solução})$, então, *Melhor Solução* $\leftarrow D$.
5. Retornar à melhor solução encontrada.

Segundo Müller (1997), a BT tem se mostrado eficiente na solução de problemas de otimização combinatorial, encontrando soluções, algumas vezes, até melhores do que as encontradas por outras técnicas. Para melhor detalhamento da BT, pode-se recorrer a Glover (1989a, b).

3.2.1 Configurações da lista tabu

Foram realizados testes utilizando-se lista tabu estática e dinâmica. Na lista tabu estática o tempo tabu é fixo (TTF) e predefinido de acordo com o tamanho do problema. A lista dinâmica possui duas variantes:

Tempo tabu aleatório (TTA)

1. A cada iteração, um número aleatório é escolhido dentro do intervalo especificado por T_{\min} e T_{\max} e atribuído ao elemento tabu que acaba de entrar na lista.

Tempo tabu variável (TTV)

1. Se não ocorrer melhora na função objetivo durante o tempo de T iterações:

$$(T = \text{Tempo Tabu Corrente} = \sqrt{\text{Número de Iterações}}),$$

todos os elementos que estão na lista tabu são liberados de seu estado tabu. A lista tabu é zerada e seu tamanho é modificado para um valor escolhido dentro do intervalo especificado por T_{\min} e T_{\max} .

3.2.2 Estrutura de vizinhança

Para ilustrar como as estruturas de vizinhança funcionam, considere a lista $L(a, b, c, d, e, f, g, h, i, j)$ formada pelos n departamentos a serem alocados ($n = \text{total de departamentos}$). Foram utilizadas três estruturas de vizinhança $V(S)$:

1. *Troca simples* – TSS: (número de vizinhos $V(S) = 10$). Os possíveis movimentos de troca são: a com b , b com c , c com d , d com e , e com f , f com g , g com h , h com i , i com j e j com a , escolhendo-se o melhor movimento admissível ou o movimento que satisfaça o critério de aspiração.
2. *Pseudotroca simples* – PTS: (número de vizinhos $V(S) = 10$). Similar à troca simples, porém os movimentos não são efetivamente realizados, são apenas avaliados, sendo posteriormente escolhido o melhor deles. O melhor movimento a ser efetivamente realizado é escolhido da seguinte forma: das U pseudo-avaliações realizadas, é escolhido um subconjunto $W \in U$ para ser efetivamente testado. O melhor movimento admissível dentro desse subconjunto é realmente realizado, levando uma solução S a uma solução S' na vizinhança $V(S)$. Ser um movimento apenas avaliado significa que não é necessário utilizar o algoritmo VCM para alocação efetiva dos departamentos dentro dos limites da fábrica. A PTS considera que todos os departamentos possuem a mesma área, formato e aspecto, 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. A PTS então trabalha com as pseudodistâncias encontradas. Pode-se, assim, avaliar a pseudovizinhança sem a necessidade de alocar os departamentos para encontrar as distâncias verdadeiras. O subconjunto W é verdadeiramente testado, sendo finalmente escolhido para ser realizado o melhor movimento admissível para a geração da nova solução corrente S .

3. *Pseudotroca expandida* – PTE: $V(S)$ é maior. O número de vizinhos é igual a $V*(V-1)/2$, isto é, $10*(10-1)/2 = 45$. A PTE utiliza a mesma lógica da PTS, diferindo no tamanho da vizinhança avaliada.

O algoritmo de busca tabu (ABT) recebe uma solução inicial S_0 e tenta determinar, durante um número preestabelecido de iterações, uma solução que apresente a maior redução possível na função de avaliação.

3.2.3 Visão geral da busca tabu utilizada

Representação do problema:

Lista de alocação

Vizinhança $V(S)$:

Troca simples (TSS)

Pseudotroca simples (PTS)

Pseudotroca expandida (PTE)

Elementos da estrutura de controle:

Matriz tabu:

- Tempo tabu fixo (TTF)
- Tempo tabu aleatório (TTA)
- Tempo tabu variável (TTV)

Critério de aspiração:

- Um movimento deixa de ser tabu se levar a uma solução melhor que a melhor solução encontrada até o momento.

- Se todos os movimentos admissíveis estão tabus, o que menos piorar a função objetivo será liberado.
- Se não ocorrer melhora na função objetivo em T iterações (PTE), libere todos os elementos tabus.

Critério de parada:

- Número preestabelecido de iterações.

Decodificação (alocação física dos departamentos):

Algoritmo VCM

4. Resultados obtidos e análise comparativa

Para avaliar o desempenho do sistema AVOLI, realizaram-se testes em quatro estudos de caso.

Estudo de caso 1: reporta-se ao banco de problemas proposto por Nugent *et al.* (1968), com os problemas 5, 6, 7, 8, 12, 15, 20 e 30, apresentando um número máximo de 500 iterações cada. Os resultados obtidos foram comparados aos resultados apresentados por Tate & Smith (1995), Suresh & Sahu (1995), Chiang (2001) e às melhores soluções apresentadas por Nugent *et al.* (1968), os quais se encontram resumidos na Tabela 2. A primeira coluna dessa tabela se refere ao porte do problema tratado e a segunda, ao número de iterações. O tempo é medido em segundos. Na terceira coluna o tempo total se refere ao tempo decorrido entre a primeira e a última iterações. As colunas quatro e cinco mostram, respectivamente, em qual iteração a melhor solução (coluna 10 – AVOLI) foi encontrada e o tempo necessário para encontrá-la. A última coluna apresenta, em termos percentuais, a que distância as soluções encontradas pelo AVOLI estão das melhores soluções apresentadas por Chiang (2001). No primeiro caso utilizou-se tempo tabu fixo igual a 5 para os 4 primeiros problemas e 8 para os demais, com estrutura de vizinhança e pseudotroca simples de tamanho n . Em todos os casos, o custo de transporte entre cada par de departamentos foi considerado unitário.

Tabela 2 – Análise comparativa para o estudo de caso 1.

Total de departamentos	Total de iterações	Tempo total (s)	Melhor iteração	Melhor tempo (s)	Nugent (1968)	Tate (1995)	Surech (1995)	Chiang (2001)	AVOLI	Distância (%)
5	100	3	8	0	50	50	50	50	50	0,00%
6	100	3	1	0	86	86	86	86	86	0,00%
7	100	3	23	1	148	148	148	144	148	2,70%
8	100	3	8	0	214	214	214	212	214	0,90%
12	300	29	39	5	578	578	578	578	578	0,00%
15	200	30	120	18	1.150	1.150	1.150	1.110	1.150	3,47%
20	500	2.002	300	172	2.570	2.598	2.570	2.564	2.570	0,23%
30	500	1.159	171	393	6.124	6.184	6.168	6.094	6.128	0,55%

Como se observa na Tabela 2, dos oito problemas-teste, três obtiveram soluções iguais às das outras abordagens comparadas, três obtiveram solução menos de 1% inferior e duas, solução cerca de 3% inferior. Os tempos computacionais mostraram-se aceitáveis pela complexidade dos problemas. Nesse caso, deve-se observar que os tempos computacionais apresentados neste trabalho se referem ao tempo decorrido até a iteração referente à melhor solução encontrada.

Estudo de caso 2: utilizaram-se as matrizes de fluxos dadas por Nugent *et al.* (1968) e as dimensões dos departamentos utilizadas por Tam & Li (1991), abordando os problemas 5, 6, 7, 8, 12, 15, 20 e 30 e adotando um número máximo de 1.000 iterações. Sugeriu-se o quadro externo (comprimento x largura, em metros) para alocação dos departamentos em cada problema. A Tabela 3 mostra os resultados obtidos.

A última coluna da Tabela 3 mostra o percentual de melhoria da solução final em relação à solução inicial. Nos problemas 5, 6, 7 e 8, o modelo heurístico apresentado obteve soluções ótimas iguais às encontradas pelo método exato. Para os problemas 12, 15, 20 e 30, comparações são inapropriadas, já que o método exato não permite encontrar a solução ótima para problemas desse porte em tempo computacional razoável. Observa-se que a melhoria que utiliza a BT em relação à solução inicial varia de 14% a 32%, pode-se concluir, portanto, que a eficiência de convergência do algoritmo é promissora. Nota-se que a escolha adequada de TTV e PTE (colunas 5 e 6) pode resultar em soluções de melhor qualidade em tempo computacional razoável. Para os problemas 12, 15, 20 e 30 não se pode garantir que a solução encontrada seja a ótima.

Tabela 3 – Desempenho do algoritmo usando Nugent *et al.* (1968) e Tam & Li (1991).

Total de departamentos	Quadro externo	Total de iterações	Tempo total (s)	TTV $t_{\min}-t_{\max}$	V(S) PTE	Melhor iteração	Melhor tempo (s)	Solução inicial	Solução final Exata	Solução final Heurística	Melhoria (%)
5	14 x 09	250	9	5-9	4	38	1	272	232	232	14,71
6	13 x 11	250	10	5-9	6	93	3	456	376	376	17,54
7	16 x 10	250	12	7-12	6	96	4	1.030	676	676	24,72
8	18 x 10	250	13	7-12	6	66	3	1.210	974	974	19,5
12	19 x 13	1.000	135	5-12	10	312	41	3.229		2.347	27,31
15	20 x 15	500	141	8-18	15	378	104	7.104		4.853	31,69
20	21 x 15	1.000	329	9-20	10	527	175	13.779		9.722	29,44
30	25 x 16	1.000	1.264	8-18	10	620	783	34.274		23.312	31,98

Estudo de caso 3: nesses testes foram utilizados os problemas 8, 12, 15, 20 e 30 da segunda bateria de problemas, acrescentando-se restrições não tratadas nos modelos originais. Todos os testes foram realizados com 1.000 iterações, utilizando PTE e lista tabu dinâmica TTV. A Tabela 4 resume os resultados obtidos.

Foram consideradas as seguintes restrições adicionais:

- No problema 8 o departamento 3 e o departamento 4 devem ficar na vertical, enquanto o departamento 7 deve ficar na horizontal.
- No problema 12 o departamento 7 deve ficar na horizontal, o departamento 10, na vertical, o departamento 1, próximo ao departamento 7, o departamento 3, próximo ao 11 e o departamento 3, longe do departamento 4.
- No problema 15 o departamento 6 deve ficar na vertical, o departamento 4, longe dos departamentos 10 e 15 e o departamento 10, longe do departamento 15.
- No problema 20 o departamento 7 deve ficar na horizontal e o 8, na vertical, o departamento 14 deve ficar longe do

departamento 8 e o departamento 3, perto do departamento 13 e há 2 áreas ocupadas.

- No problema 30 o departamento 2 deve ficar na horizontal e o departamento 6, na vertical, o departamento 16 deve ficar longe do departamento 21 e o departamento 1, perto do 20, há 2 áreas ocupadas e 4 departamentos fixos.

Observa-se que a inclusão de restrições no estudo de caso 3 modificou discretamente os tempos computacionais. A melhoria das soluções obtidas, quando comparadas às soluções iniciais, variou de 22,77% a 34,26%. Quando se comparam os melhores tempos e as melhores iterações do caso 3 com o caso 2, observa-se que a inclusão de restrições, além de modificar os tempos computacionais, pode influenciar a solução final obtida.

Estudo de caso 4: aqui dois novos problemas são propostos. O primeiro possui 40 departamentos e o segundo, 55. Adota-se um número máximo de 500 iterações para cada experimento. As seguintes restrições são consideradas:

- No problema 40 foram consideradas três áreas ocupadas. Os departamentos 8, 19 e 29 devem ficar na vertical e os departamentos 4, 10 e 17, na horizontal; o departamento 5 deve ficar longe do 32 e o departamento 2, perto do 5 e também próximo ao 32.

Tabela 4 – Resultados com restrições de adjacências e de localização para o estudo de caso 3.

Total de departamentos	Quadro externo	Total de iterações	Tempo total (s)	TTV $t_{\min}-t_{\max}$	V(S) PTE	Solução inicial	Melhor iteração	Melhor tempo (s)	Solução final	Melhoria (%)
8	10 x 17	1.000	55	7-12	6	1.172	54	2	898	23,38
12	18 x 13	1.000	177	5-12	10	4.174	769	134	2.744	34,26
15	20 x 14	1.000	154	8-18	10	6.870	531	83	5.306	22,77
20	21 x 15	1.000	448	9-20	12	17.502	982	440	10.338	28,37
30	25 x 16	1.000	1.786	8-18	12	28.660	768	1.359	221.888	22,58

- No problema 55 foram consideradas 5 áreas ocupadas e 5 departamentos do tipo “fixos”. Os departamentos 8, 9, 19 e 29 devem ficar na vertical e os departamentos 10 e 17, na horizontal, o departamento 9 deve ficar perto do 47, o departamento 15, longe do 19, o departamento 23, longe do departamento fixo 3, o 14, longe do departamento fixo 4 e o departamento 20 deve ficar na borda do quadro externo.

A Tabela 5 apresenta os resultados obtidos em cada problema. A última coluna mostra o percentual de melhoria da solução final em relação à inicial. A performance das soluções para cada problema desse caso encontra-se na Tabela 6. Os layouts correspondentes à melhor solução final de cada problema, bem como todos os valores utilizados nos problemas propostos, encontram-se no Anexo A.

Nesse caso, a melhor solução obtida, quando comparada à solução inicial, teve melhora de 29,07% para o problema com 40 departamentos e de 30,27% para o problema com 55 departamentos. Observa-se que a escolha mais adequada da lista de candidatos à troca restringe o número de elementos a serem analisados. Dessa forma, pôde-se usar uma vizinhança reduzida sem comprometer a qualidade da solução encontrada. Tanto no estudo de caso 3 quanto no 4, os limites inferior e superior de TTV e o tamanho da PTE foram escolhidos com base em várias tentativas. As melhores combinações para TTV e PTE, que não comprometeram a performance do sistema AVOLI, variaram entre 5 e 14 para TTV e entre 20 e 25 para PTE, pode-se observar que o

tamanho mais apropriado de vizinhança está em torno de $n/2$ para este caso.

Por intermédio desses problemas pôde-se comprovar a eficácia do AVOLI. A melhor combinação entre vizinhança e tempo tabu foi detectada quando se considerou a lista tabu dinâmica (TTV) em conjunto com a vizinhança V(S) pseudotroca expandida (PTE). A escolha mais adequada de T_{\min} e T_{\max} permitiu balancear a exploração do espaço de soluções, intensificando ou diversificando, quando os atributos de movimento foram mais fortemente restringidos. A escolha mais apropriada do tipo de vizinhança, bem como a escolha do tamanho da lista tabu mais adequada, permitiu obter excelentes resultados, constatados pelos percentuais de melhoria atingidos nesses dois novos problemas propostos.

5. Conclusões e trabalhos futuros

Este artigo apresentou um modelo heurístico para solução de problemas de layout industrial considerando a imposição de restrições geométricas e de localização, frequentemente encontradas em problemas reais e não tratadas adequadamente em outros modelos.

A abordagem de solução utilizada foi dividida em 2 passos. No primeiro obteve-se, por meio de uma heurística construtiva, uma solução inicial viável, não violando nenhuma das restrições impostas. No segundo passo obteve-se uma solução melhorada pelo uso da busca tabu com as devidas adaptações. Nesse contexto, três estruturas de vizinhança foram implementadas para o problema e seus efeitos foram analisados, tomando por comparação problemas cujas soluções já foram demonstradas anteriormente na literatura.

Tabela 5 – Resultados da quarta bateria de problemas.

Total de departamentos	Quadro externo	Total de iterações	Tempo total (s)	TTV t_{\min} - t_{\max}	V(S) PTE	Solução inicial	Melhor iteração	Melhor tempo (s)	Solução final	Melhoria (%)
40	60 x 45	500	11.692	5-12	25	111.027	190	4.229	78.755	29,07
55	90 x 60	500	20.011	7-14	20	187.770	89	3.436	130.930	30,27

Por meio dos testes realizados, observou-se que as dimensões do quadro externo podem influenciar diretamente nos resultados obtidos, portanto, a escolha adequada de suas dimensões pode produzir melhores resultados.

As soluções obtidas pelo sistema AVOLI (Anexo A) se mostraram tão promissoras e “boas” quanto as melhores soluções conhecidas para os problemas-teste. Problemas maiores que os até então reportados foram propostos e tratados em tempo computacional aceitável, porém não foi possível realizar uma comparação detalhada de resultados, uma vez que não foram encontrados na literatura problemas semelhantes. Dentre as possibilidades de trabalhos futuros incluem-se: a) adaptar o sistema para o tratamento de layouts de múltiplos pisos (multi-níveis); b) considerar departamentos com formas irregulares; c) gerar ambiente visual mais flexível (ambiente gráfico); d) desenvolver layouts detalhados; e) desenvolver metodologia para geração automática da lista tabu dinâmica; f) considerar restrições realistas mais complexas; e g) analisar as propriedades de convergência e precisão da BT/AVOLI.

Anexo A

Este anexo apresenta a interface de trabalho do sistema AVOLI, bem como os dados de entrada para os dois problemas propostos e os melhores layouts encontrados. A Figura 1 mostra, respectivamente, a interface de trabalho do software AVOLI e a legenda utilizada para representar o layout gráfico em duas dimensões.

A identificação dos tipos de departamentos e das áreas inutilizadas apresenta o seguinte padrão: se o sinal “-” estiver antes do número, identifica departamentos do tipo “fixos” e se o sinal “+” estiver antes do número, identifica departamentos do tipo “móveis”. As Figuras 2 e 3 mostram os layouts correspondentes à solução inicial e à melhor solução final para o problema com 40 departamentos. As Figuras 4 e 5 apresentam os layouts correspondentes à solução inicial e à melhor solução final para o problema com 55 departamentos. As Tabelas 6, 7 e 8 mostram, respectivamente, as áreas ocupadas, as características geométricas de cada departamento do tipo “móveis” e a matriz de fluxo para o problema com 40 departamentos.

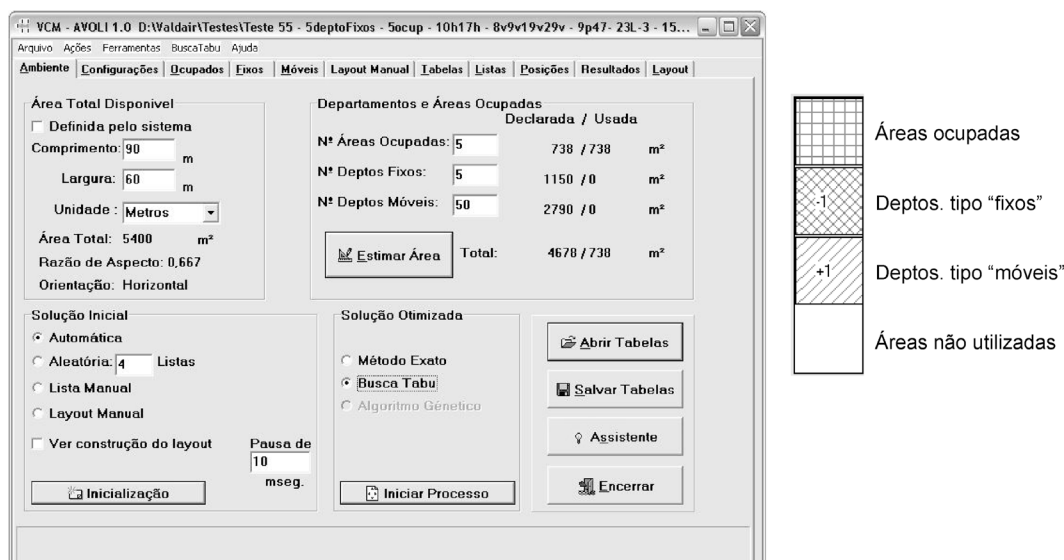


Figura 1 – Interface de trabalho do AVOLI e legenda utilizada no layout gráfico em 2D.

A Tabela 9 apresenta as características geométricas dos departamentos do tipo “fixos”. A Tabela 10 apresenta as áreas ocupadas e as Tabelas 11 e 12 apresentam, respectivamente, as características geométricas dos departamentos do tipo “móveis” e a matriz de fluxo para o problema com 55 departamentos. O sistema AVOLI foi desenvolvido em Object Pascal, usando Delphi

5.0 e executado em um computador AMD Athlon (tm) 1,2 GHz com 256 MB RAM.

Problema com 40 departamentos

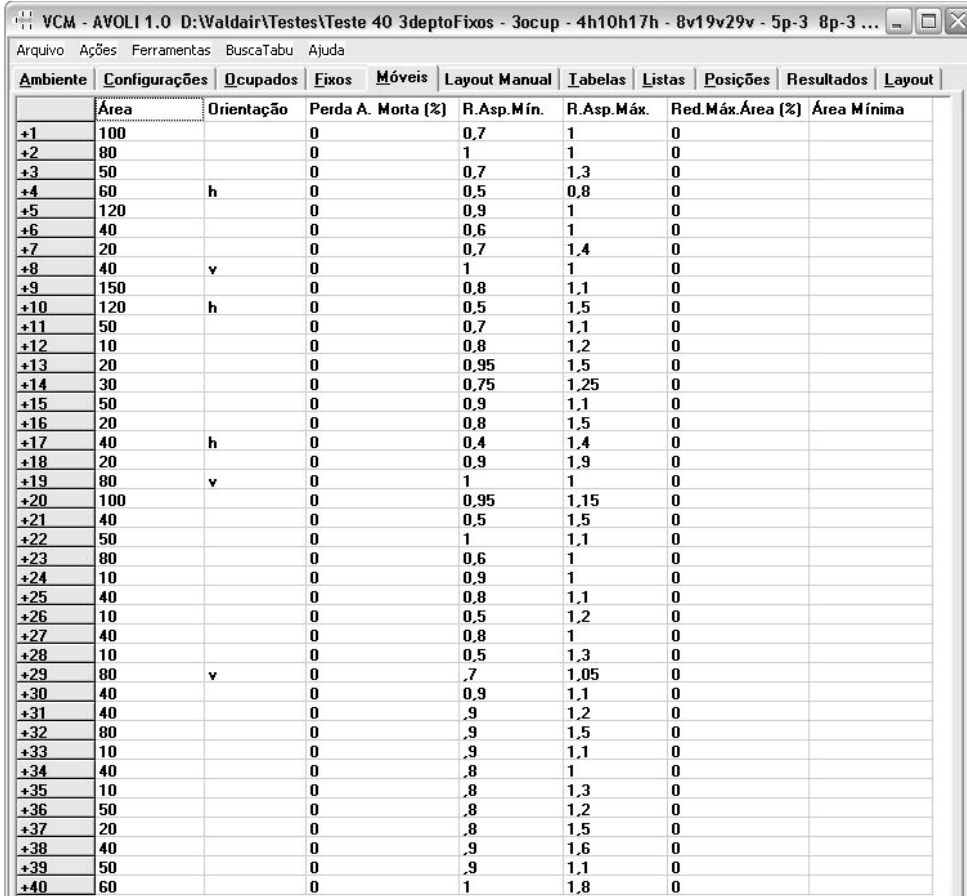
No problema com 40 departamentos há três áreas inutilizadas e identificadas sequencialmente pelos números 1, 2 e 3, conforme mostrado na Tabela 6.

Tabela 6 – Problema com 40 departamentos: áreas ocupadas.



	Início	Yinício	Largura	Comprimento
1	45	27	8	3
2	38	24	3	10
3	59	1	16	2

Tabela 7 – Características geométricas dos 40 departamentos do tipo “móveis”.



	Área	Orientação	Perda A. Morta (%)	R. Asp. Mín.	R. Asp. Máx.	Red. Máx. Área (%)	Área Mínima
+1	100		0	0,7	1	0	
+2	80		0	1	1	0	
+3	50		0	0,7	1,3	0	
+4	60	h	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	0	1	1	0	
+9	150		0	0,8	1,1	0	
+10	120	h	0	0,5	1,5	0	
+11	50		0	0,7	1,1	0	
+12	10		0	0,8	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	0,4	1,4	0	
+18	20		0	0,9	1,9	0	
+19	80	v	0	1	1	0	
+20	100		0	0,95	1,15	0	
+21	40		0	0,5	1,5	0	
+22	50		0	1	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	

Tabela 8 – Matriz de fluxo para o problema com 40 departamentos.

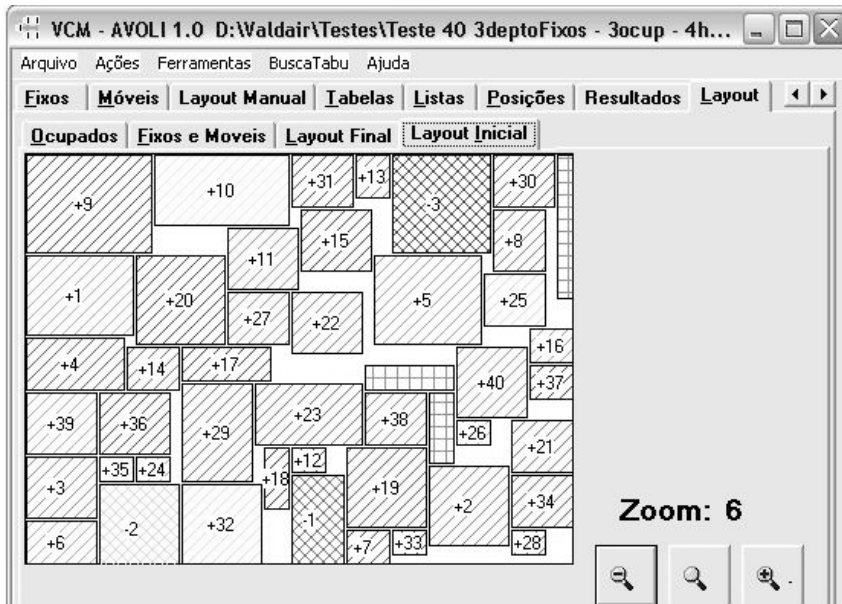


Figura 2 – Layout inicial: problema com 40 departamentos.

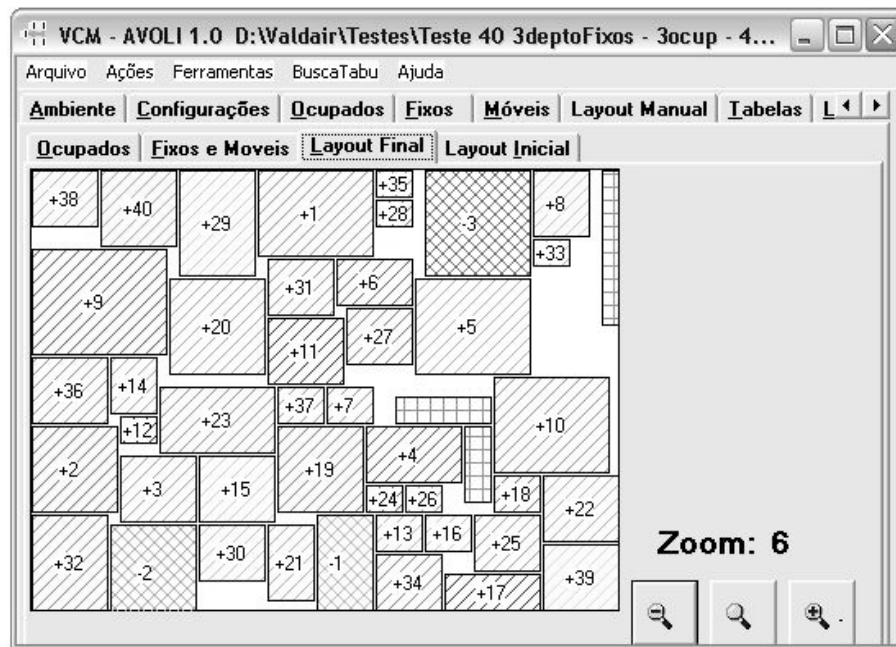


Figura 3 – Layout final: problema com 40 departamentos.

O problema possui 40 departamentos identificados seqüencialmente pelos números +1, +2, +3,..., +40. A Tabela 7 reúne todas as informações necessárias que o sistema precisa receber sobre cada departamento do tipo “móveis”.

Problema com 55 departamentos

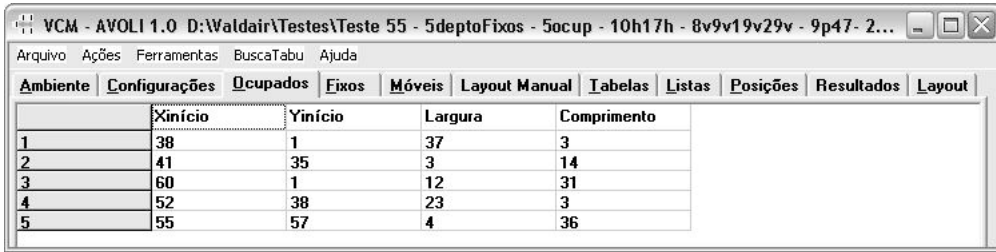
Dos 55 departamentos, cinco são do tipo “fixos” e estão cadastrados conforme a Tabela

9. Esse problema possui 5 áreas ocupadas e identificadas seqüencialmente pelos números 1, 2, 3, 4 e 5, conforme mostrado na Tabela 10.

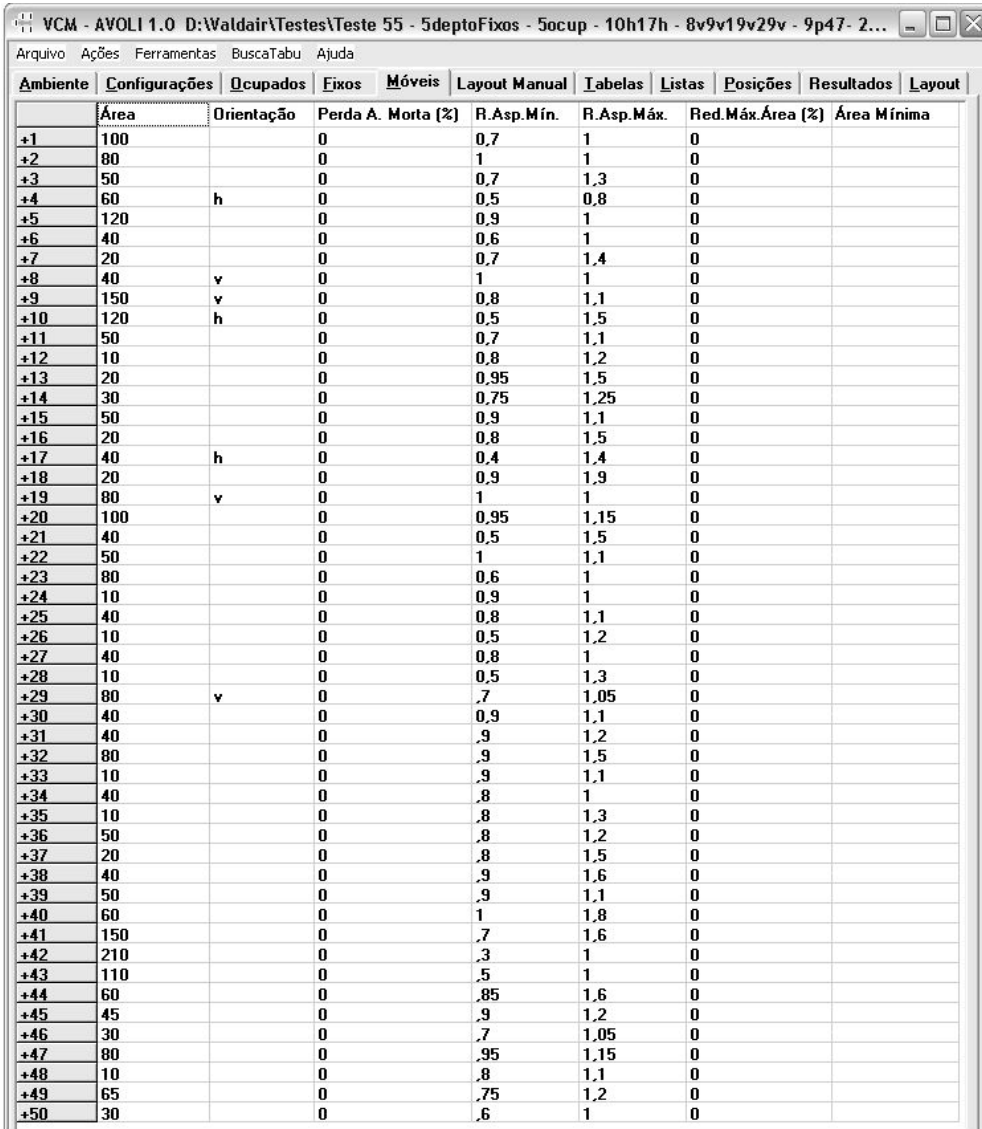
As informações dos outros 50 departamentos do tipo “móveis” são apresentadas na Tabela 11. A matriz de fluxo (volume de tráfego) entre cada par de departamentos é mostrada na Tabela 12.

Tabela 9 – Características geométricas dos departamentos do tipo “fixos”.

	Área	Xinício	Yinício	Orientação	RA Morta	R.Asp.Mín.	R.Asp.Máx.
-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	80	42	38	h	0	,6	1,2

Tabela 10 – Áreas ocupadas para o problema com 55 departamentos.


	Xinício	Yinício	Largura	Comprimento
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 11 – Características geométricas dos departamentos do tipo “móveis”.


Ambiente	Área	Orientação	Perda A. Morta (%)	R. Asp. Mín.	R. Asp. Máx.	Red. Máx. Área (%)	Área Mínima
+1	100		0	0,7	1	0	
+2	80		0	1	1	0	
+3	50		0	0,7	1,3	0	
+4	60	h	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	0	1	1	0	
+9	150	v	0	0,8	1,1	0	
+10	120	h	0	0,5	1,5	0	
+11	50		0	0,7	1,1	0	
+12	10		0	0,8	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	0,4	1,4	0	
+18	20		0	0,9	1,9	0	
+19	80	v	0	1	1	0	
+20	100		0	0,95	1,15	0	
+21	40		0	0,5	1,5	0	
+22	50		0	1	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	65		0	,75	1,2	0	
+50	30		0	,6	1	0	

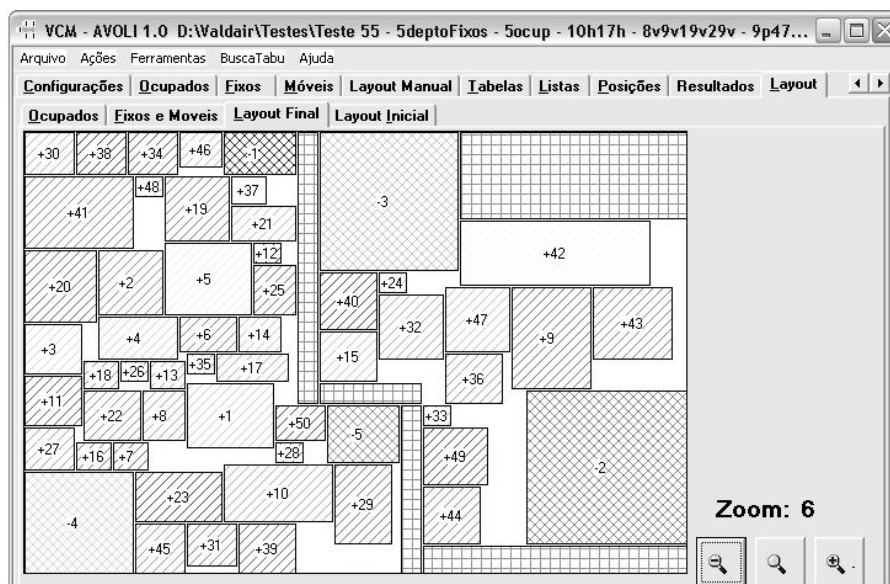


Figura 5 – Layout final: problema com 55 departamentos.

Referências Bibliográficas

- APPLE, J. M.; DEISENROTH, M. P. *A computerized plant layout Analysis and evaluation technique (PLANET)*. Proceedings of the 23rd Annual Conference and Convention, AIIE. California, p. 121-127, 1972.
- ARMOUR, G. C.; BUFFA, E. S. A heuristic algorithm and simulation approach to relative allocation of facilities. *Management Science*, v. 9, p. 294-309, 1963.
- ASKIN, R. G.; STANDRIDGE, C. R. *Modeling and analysis of manufacturing systems*. John Wiley & Sons Inc., 1993.
- AZADIVAR, F.; WANG, J. Facility layout optimization using simulation and genetic algorithms. *International Journal of Production Research*, v. 38, n. 17, p. 4369-4383, 2000.
- BAZARAA, M. S. Computerized layout design: a branch and bound approach. *AIIE Transactions*, v. 7, n. 4, p. 432-437, 1975.
- CHIANG, W. C. Visual facility layout design system, *International Journal Production Research*, v. 39, n. 9, p. 1811-1836, 2001.
- CIMTECHNOLOGIES CORP. 1989. Disponível em: <<http://www.systemflow.com>>, <<http://www.cimtechnologies.com>>.
- CORAINI, A. L. S.; NOLLA, I. M. *Autocad Release 12: curso básico e prático*. Makron Books, 1994.
- ELSHAFEI, A. N. Hospital layout as a quadratic assignment problem. *Operations Research Quarterly*, v. 28, n. 1, p. 167-179, 1977.
- FOULDS, L. R.; ROBINSON, D. F. Graph theoretic heuristics for the plant layout problem. *International Journal of Production Research*, v. 16, n. 1, p. 27-37, 1978.
- FURTADO, J. C.; LORENA, L. A. N. Otimização de layout usando busca tabu. *Gestão & Produção*, v. 4, n. 1, p. 88-107, 1997.
- GAREY, M. R.; JOHNSON, D. S. *Computers and intractability: a guide to the theory of NP-completeness*. New York: W. H. Freeman and Company, 1979.
- GLOVER, F. Tabu search – part I. *ORSA Journal on Computing*, v. 1, n. 3, p. 190-206, 1989a.

- GLOVER, F. Tabu search – part II. *ORSA Journal on Computing*, v. 2, n. 1, p. 4-32, 1989b.
- GLOVER, F. Tabu search – a tutorial. *Interfaces*, v. 20, n. 4, p. 74-94, 1990.
- GLOVER, F.; LAGUNA, M. *Tabu search*. Massachusetts: Kluwer Academic Publishers, 1997.
- HERAGU, S. S.; KUSIAK, A. The facility layout problem. *European Journal of Operational Research*, v. 29, p. 229-251, 1987.
- HERAGU, S. S.; KUSIAK, A. Efficient models for the facility layout problem. *European Journal of Operational Research*, v. 53, p. 1-13, 1991.
- HOLLAND, J. H. *Adaptation in natural and artificial systems*. University of Michigan, Ann Arbor, MI, 1975.
- KIRKPATRICK, S.; GELLAT Jr., C. D.; VECCHI, M. P. Optimization by simulated annealing. *Science*, v. 220, p. 671-680, 1983.
- KOOPMANS, T. C.; BECKMAN, M. Assignment problems and the location of economic activities. *Econometrica*, v. 25, n. 53, p. 53-76, 1957.
- KUSIAK, A.; HERAGU, S. S. The facility layout problem. *European Journal of Operational Research*, v. 29, p. 229-251, 1987.
- LEE, R. C.; MOORE, J. M. *CORELAP* – computerized relationship layout planning. *Industrial Engineering*, v. 18, p. 195-200, 1967.
- MELLER, R. D.; GAU, K. Y. The facility layout recent and emerging trends and perspectives. *Journal of Manufacturing Systems*, v. 15, n. 5, p. 351-366, 1996.
- MONTREUIL, B. *A modeling framework for integrating layout design and flow network design*. Proceeding of the Material Handling Research Colloquium, Hebron, KY, p. 43-58, 1990.
- MONTREUIL, B.; VENKATADRI, U.; RATLIFF, H. D. Generating a layout from a design skeleton. *IIE Transactions*, v. 25, n. 1, p. 3-15, 1993.
- MÜLLER, F. M. Heurísticas e metaheurísticas. In: *Escola Regional de Informática, 5. Anais...* Santa Maria, 1997, p. 19-40.
- NUGENT, C. E.; VOLLMANN, T. E.; RUML, J. An experimental comparison of techniques for the assignment of facilities to locations. *Operations Research*, v. 16, p. 150-173, 1968.
- SCRIABIN, M.; VERGIN, R. C. A cluster-analytic approach to facility layout. *Management Science*, v. 31, n. 1, p. 33-49, 1985.
- SEEHOF, J. M.; EVANS, W. O. Automated layout design program. *Industrial Engineering*, v. 18, p. 690-695, 1967.
- SURESH, G.; SAHU, S. A genetic algorithm for facility layout. *International Journal of Production Research*, v. 33, n. 12, p. 3411-3423, 1995.
- TAM, K. Y.; LI, S. G. A hierarchical approach to the facility layout problem. *International Journal of Production Research*, v. 29, p. 165-184, 1991.
- TAM, K. Y. Genetic algorithms, function optimization, and facility layout design. *European Journal of Operational Research*, v. 63, p. 322-346, 1992.
- TAM, K. Y.; CHAN, S. K. Solving facility layout problems with geometric constraints using parallel genetic algorithms: experimentation and findings. *International Journal of Production Research*, v. 36, n. 12, p. 3253-3272, 1998.
- TATE, D. M.; SMITH, A. E. A genetic approach to the quadratic assignment problem. *Computers and Operations Research*, v. 32, p. 73-83, 1995.
- TOMPIKINS, J. A.; REED Jr., R. An applied model for the facilities design problem. *International Journal of Production Research*, v. 14, n. 5, p. 583-595, 1976.
- VOLLMAN, T. E.; BUFFA, E. S. Facilities layout problem in perspective. *Management Science*, v. 12, p. 450-468, 1966.

OPTIMIZATION OF INDUSTRIAL LAYOUTS BASED ON TABU SEARCH

Abstract

This paper approaches a computational solving of the industrial layout problem considering hard-constraints not handled in previous works. The problem is solved in two steps: first a constructive-heuristic-based initial solution is generated and then tabu search heuristic is used to improve it. As a contribution, it shown a computational tool named AVOLI (Visual Environment for Optimization of Industrial Layout) that will aid planners and production engineers in the generation of layouts that contemplate real constraints.

Key words: *layout, optimization, hard-constraints, tabu search.*