

UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA QUÍMICA

Área de Concentração: Desenvolvimento de Processos Químicos



UNICAMP

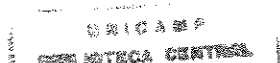
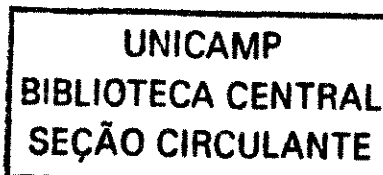
**“Otimização do posicionamento de componentes
no *layout* de plantas químicas”**

Dissertação de Mestrado apresentada à
Faculdade de Engenharia Química como
parte dos requisitos exigidos para a
obtenção do título de Mestre em
Engenharia Química.

***Autor:* Lucilene Beatriz Pissinatto**

***Orientador:* Prof. Dr. Reginaldo Guirardello**

Campinas - SP
Outubro / 2001



192702002

UNIDADE BC
 1.ª CHAMADA:
T/UNICAMP
P677o
 Ex.
 OMBO BC/47135
 ROC. 16-392/01
 C ☐ D ☒
 REC# 25 11,00
 ATA 12-01-2002
 .ª CPD

CM00162413-B

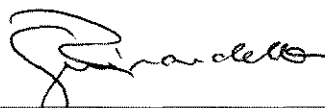
FICHA CATALOGRÁFICA ELABORADA PELA
 BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

P677o Pissinatto, Lucilene Beatriz
 Otimização do posicionamento de componentes no
 layout de plantas químicas / Lucilene Beatriz Pissinatto.--
 Campinas, SP: [s.n.], 2001.

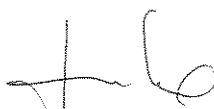
Orientador: Reginaldo Guirardello
 Dissertação (mestrado) - Universidade Estadual de
 Campinas, Faculdade de Engenharia Química.

1. Layout. 2. Modelos matemáticos. 3. Otimização
 matemática. I. Guirardello, Reginaldo. II. Universidade
 Estadual de Campinas. Faculdade de Engenharia
 Química. III. Título.

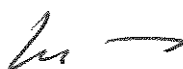
Dissertação de Mestrado defendida por *Lucilene Beatriz Pissinatto* e aprovada em *01 de outubro de 2001* pela banca examinadora constituída pelos doutores:



Prof. Dr. Reginaldo Guirardello

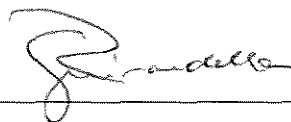


Prof. Dr. José Maurício Pinto



Prof. Dr. Roger Josef Zemp

Este exemplar corresponde à versão final da Dissertação de Mestrado em Engenharia Química defendida por *Lucilene Beatriz Pissinatto* e aprovada pela comissão julgadora em *01/10/2001*.

A handwritten signature in black ink, appearing to read 'Reginaldo', is positioned above a horizontal line.

Prof. Dr. Reginaldo Guirardello
(Orientador)

DEDICATÓRIA

Dedico este trabalho aos meus pais, *Neide e Oreste*, e aos meus irmãos, *Antonio Carlos e Gilberto*, pelo carinho, incentivo e paciência durante todo o percurso.

AGRADECIMENTOS

A Deus, pela vida.

Aos meus pais e irmãos, pelo apoio, carinho e confiança.

Ao professor Reginaldo Guirardello, pela orientação e amizade.

À FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo), pela bolsa de pesquisa e fornecimento de recursos para a realização deste trabalho.

À todos os meus amigos, que sempre me incentivaram e ajudaram de alguma forma, especialmente, Édler, Iraí, Simone, Elaine, Denise, Priscila, Érika, Michele, Rúbia, Rubão e Nicole.

À todas as pessoas que, direta ou indiretamente, ajudaram no trabalho, ou ainda, que conviveram comigo, compreendendo os momentos de dificuldade e incentivando para continuar sempre alegre e perseverante.

RESUMO

O desenvolvimento adequado do *layout* de uma planta química é muito importante por motivos de economia, operação e segurança. A otimização global da planta deve incluir a otimização do posicionamento dos componentes e o comprimento e distribuição da tubulação, satisfazendo as restrições de segurança.

O objetivo deste trabalho foi o desenvolvimento de modelos matemáticos mais eficientes, do ponto de vista computacional, para realizar a otimização do posicionamento dos componentes do *layout* de plantas químicas, através da formulação do problema em programação mista, inteira e linear (MILP), resolvendo-se com o auxílio do pacote comercial de otimização, GAMS/CPLEX. Um modelo mais eficiente implica em menor tempo de cálculo e resultados satisfatórios. Para testar os modelos foram propostos três casos de estudo, com diferentes números de componentes: uma unidade petroquímica (6 componentes); uma unidade de polimerização (7 componentes) e uma planta de recuperação de mercúrio a partir de efluentes eletrolíticos (9 componentes).

O tempo computacional para resolver cada caso está intimamente ligado ao número de componentes, ou seja, quanto maior o número, tanto maior o tempo gasto para sua resolução.

Os modelos testados variaram de caso para caso, isto é, com a adição de algumas restrições lógicas e/ou substituição de outras restrições relacionadas aos bocais e tubos, o tempo computacional diminuiu para a unidade petroquímica e para a planta de recuperação de mercúrio, no entanto, o modelo original (existente na literatura) apresentou melhores resultados para a planta de polimerização.

Portanto, verificou-se que os modelos propostos são melhores em algumas situações, mas não em outras, dependendo do caso de estudo e das condições envolvidas (número de variáveis, número de iterações e nós inspecionados pelo algoritmo B&B, etc.).

ABSTRACT

The adequate development of the layout of a chemical plant is very important because of economic, operational and safety reasons. The overall optimization of the plant must include the optimization of the placement of the components and the length and distribution of the pipes, satisfying safety constraints.

The objective of this work was the development of more efficient mathematical models to accomplish the optimization of the placement of the components of the layout of chemical plants, through the formulation of the problem as mixed integer linear programming (MILP), solved with the aid of a commercial optimization package, GAMS/CPLEX. A more efficient model implies in smaller time of calculation and satisfactory results. To test the models, three case studies were proposed, with different numbers of components: a petrochemical unit (6 components); a suspension polymerization plant (7 components) and a recovery of mercury from the effluents of electrolytic plant (9 components).

The computational time to solve each case is intimately linked to the number of components, in other words, as larger the number, so much larger the time spent for its resolution.

The tested models showed mixed results, that is, with the addition of some logical constraints or substitution of other constraints related to the nozzles and pipes, the computational time decreased for the petrochemical unit and for the plant of mercury recovery, but the original model (existent in the literature) presented better results for the suspension polymerization plant.

Therefore, it was verified that the proposed models are better in some situations, but not in others, depending on the case study and of the involved conditions (number of variables, number of iterations and nodes inspected by the algorithm B&B, etc.).

SUMÁRIO

CAPÍTULO 1. INTRODUÇÃO	01
1.1. Apresentação	01
1.2. Objetivos	02
CAPÍTULO 2. REVISÃO BIBLIOGRÁFICA	03
2.1. O <i>layout</i> e sua importância	03
2.2. Classificação do <i>layout</i>	06
2.2.1. <i>Layout</i> de produto ou linear	06
2.2.2. <i>Layout</i> de processo ou funcional	06
2.2.3. <i>Layout</i> posicional ou fixo	07
2.3. Modelos matemáticos para <i>layout</i> de processos químicos	09
2.4. Códigos computacionais para <i>layout</i>	15
CAPÍTULO 3. MODELO PROPOSTO	19
3.1. Considerações do modelo	19
3.2. Modelo matemático	20
3.3. Distâncias mínimas de segurança	21
3.4. Custos unitários	22
3.5. Modelagem implementada em GAMS (modelo original)	23
3.6. Modelo Modificado (com restrições lógicas)	29
3.7. Modelo Modificado (substituição + restrições lógicas)	30
CAPÍTULO 4. MÉTODOS E ALGORITMOS DE RESOLUÇÃO	33
4.1. Algoritmo “Branch and Bound” (B & B)	33
4.2. As etapas do algoritmo B&B aplicado aos problemas MILP	37
4.3. GAMS – “General Algebraic Modeling System”	39
CAPÍTULO 5. CASOS DE ESTUDO	41
5.1. Unidade petroquímica	41
5.2. Planta de polimerização	43
5.3. Planta de recuperação de mercúrio	45

5.4. Dados para os casos de estudo	46
5.4.1. Unidade petroquímica	47
5.4.2. Planta de polimerização	48
5.4.3. Planta de recuperação de mercúrio	49
CAPÍTULO 6. RESULTADOS E DISCUSSÃO	51
6.1. Unidade petroquímica	52
6.2. Planta de polimerização	62
6.3. Planta de recuperação de mercúrio	68
6.4. Comentários	74
CAPÍTULO 7. CONCLUSÕES	75
7.1. Sugestões para trabalhos futuros	76
REFERÊNCIAS BIBLIOGRÁFICAS	77
ANEXOS	81
ANEXO 1. UNIDADE PETROQUÍMICA	83
* Implementação do <i>modelo original</i> em GAMS/CPLEX	95
* Implementação do <i>modelo 3</i> em GAMS/CPLEX	103
ANEXO 2. PLANTA DE POLIMERIZAÇÃO	111
ANEXO 3. PLANTA DE RECUPERAÇÃO DE MERCÚRIO	123
ANEXO 4. GAMS – Descrição dos comandos principais	131

LISTA DE TABELAS

1	Os nove componentes principais da unidade petroquímica	42
2	Os componentes rearranjados para a unidade petroquímica	43
3	Os treze componentes principais da planta de polimerização	44
4	Os componentes rearranjados para a planta de polimerização	44
5	Os nove componentes principais da planta de recuperação de mercúrio	46
6	Resultados obtidos para a unidade petroquímica, com o modelo existente na literatura	52
7	Resultados obtidos para a unidade petroquímica, com o modelo modificado com as restrições lógicas (modelo 2)	53
8	Resultados obtidos para a unidade petroquímica, com o modelo modificado pela substituição de restrições e adição das restrições lógicas (modelo 3)	54
9	Custo dos suportes para cada componente da unidade petroquímica	55
10	Comprimento dos tubos, em metros, obtido pelo GAMS, para a unidade petroquímica	56
11	Custos de tubulação para a unidade petroquímica	57
12	Orientações dos componentes obtidas para a unidade petroquímica, conforme estratégia “depth first”	58
13	Orientações dos componentes obtidas para a unidade petroquímica, conforme estratégia “best bound”	58
14	Posições do centro dos componentes obtidas para a unidade petroquímica	60
15	Posições relativas entre os componentes obtidas para a unidade petroquímica, conforme a estratégia “best bound”	61
16	Comparação entre o número de equações e variáveis usadas em cada modelo, para a unidade petroquímica	62
17	Resultados obtidos para a planta de polimerização, com o modelo inicial	63
18	Resultados obtidos para a planta de polimerização, com o modelo modificado pela adição das restrições lógicas	63

19	Resultados obtidos para a planta de polimerização, com o modelo modificado pela substituição de restrições e pela adição de restrições lógicas	64
20	Custos de terreno, suportes e tubulação para a planta de polimerização	64
21	Orientações dos componentes obtidas para a planta de polimerização, conforme estratégia “depth first”	65
22	Orientações dos componentes obtidas para a planta de polimerização, conforme estratégia “best bound”	65
23	Posições do centro dos componentes obtidas para a planta de polimerização	66
24	Posições relativas entre os componentes obtidas para a planta de polimerização, para qualquer estratégia ou modelo	67
25	Comparação entre o número de equações e variáveis usadas em cada modelo, para a planta de polimerização	68
26	Resultados obtidos para a planta de recuperação de mercúrio com o modelo inicial	69
27	Resultados obtidos para a planta de recuperação de mercúrio com o modelo modificado pela adição das restrições lógicas	69
28	Resultados obtidos para a planta de recuperação de mercúrio com o modelo modificado pela substituição de restrições e pela adição das restrições lógicas	70
29	Custos de terreno, suportes e tubulação para a planta de recuperação de mercúrio	71
30	Orientações dos componentes obtidas para a planta de recuperação de mercúrio	71
31	Posições do centro dos componentes obtidas para a planta de recuperação de mercúrio	72
32	Posições relativas entre os componentes obtidas para a planta de recuperação de mercúrio, para qualquer estratégia e modelo	73
33	Comparação entre o número de equações e variáveis usadas em cada modelo, para a planta de recuperação de mercúrio	73
34	Dimensões e áreas dos componentes para a unidade petroquímica	85
35	Distâncias mínimas entre os componentes para a unidade petroquímica	85
36	Posicionamento dos bocais para a unidade petroquímica	86
37	Dados para os tubos na otimização do layout da unidade petroquímica	87

38	Dimensões e áreas dos componentes para a unidade de polimerização	113
39	Dados para os tubos na otimização do layout da planta de polimerização	113
40	Distâncias mínimas entre os componentes para a unidade de polimerização	114
41	Posicionamento dos bocais para a unidade de polimerização, componentes 1, 2 e 3.	115
42	Posicionamento dos bocais para a unidade de polimerização, componentes 4 a 7.	116
43	Dimensões e áreas dos componentes para a planta de recuperação de mercúrio	125
44	Dados para os tubos na otimização do layout da planta recuperação de mercúrio	125
45	Posicionamento dos bocais para a planta recuperação de mercúrio	126
46	Distâncias mínimas entre os componentes para a planta recuperação de mercúrio	127
47	Tipos de variáveis e seus respectivos intervalos	134

LISTA DE FIGURAS

1	Procedimento SLP (“Systematic Layout Planning”)	08
2	Estrutura do Sistema	14
3	Critério para a análise dos modelos de <i>layout</i>	15
4	As oito rotações possíveis no plano x-y	21
5	Unidade petroquímica com nove componentes principais	42
6	Planta de polimerização com treze componentes principais	43
7	Planta de recuperação de mercúrio a partir de efluentes eletrolíticos	45
8	Unidade petroquímica, modelo original, empregando-se B&B – “Depth First”	89
9	Unidade petroquímica, modelo(2), empregando-se B&B – “Best Bound”	91
10	Unidade petroquímica, modelo(3), empregando-se B&B – “Best Bound”	93
11	Planta de polimerização, modelo original, empregando-se B&B – “Best Bound”	117
12	Planta de polimerização, modelo (2), empregando-se B&B – “Best Bound”	119
13	Planta de polimerização, modelo(3), empregando-se B&B – “Best Bound”	121
14	Planta de recuperação de mercúrio , modelos (1, 2 e 3).	129

NOMENCLATURA

A_i, B_i, C_i	dimensões para o componente i
C^{per}	custo por unidade de comprimento (de perímetro)
C_k^p	custo por unidade de comprimento para o tubo k
C_{ik}^{sa}, C_{ik}^{sb}	fatores de custo unitário para o suporte do componente i
C_i^s	custo dos suportes para o componente i
C_p	custo atual
C_o	custo original
$d_{ij}^x, d_{ij}^y, d_{ij}^z$	distância de Manhattan entre os centros dos componentes i e j nas direções x , y e z
D_{ij}^h	distância mínima horizontal entre os componentes i e j
D_{ij}^v	distância mínima vertical entre os componentes i e j
E_i^l	elevação mínima vertical do componente i
$F_{ji}^x, F_{ji}^y, F_{ji}^z$	posições relativas dos bocais j na superfície do componente i
F_p^x, F_p^y, F_p^z	posições relativas dos bocais relacionadas ao terminal $k_s(k)$
F_q^x, F_q^y, F_q^z	posições relativas dos bocais relacionadas ao terminal $k_t(k)$
I_p	índice atual
I_o	índice original

$k_s(k)$	terminal dos bocais do tubo k (ponto de partida)
$k_t(k)$	terminal dos bocais do tubo k (ponto de chegada)
$l_k^{x+}, l_k^{x-}, l_k^{y+}, l_k^{y-}, l_k^{z+}, l_k^{z-}$	comprimentos dos tubos k usando a distância de <i>Manhattan</i> nas direções x, y e z
M	um número grande
N_{cp}	número de componentes
N_{pi}	número de tubos
N_{nz}	número de bocais
N_{zi}	número de bocais do componente i
x, y, z	dimensões do layout da planta
x_i, y_i, z_i	posições do centro do componente i
x_p^n, y_p^n, z_p^n	posição do bocal p
w_{il}	variável inteira para a rotação do componente i
$w_{ij}^x, w_{ij}^y, w_{ij}^z$	variáveis inteiras para as posições relativas entre os componentes i e j.

SIGLAS

B&B	Algoritmo <i>Branch and Bound</i>
GAMS	Código comercial de otimização (<i>General Algebraic Modeling System</i>)
MILP	Programação mista, inteira e linear
MIP	Programação mista e inteira
NP-completo	Polinomial não-determinístico completo

CAPÍTULO 1

INTRODUÇÃO

1. 1. Apresentação

O desenvolvimento adequado do *layout* de equipamentos e tubulações em uma planta química é muito importante por motivos de economia, de operação e de segurança. A otimização global da planta deve incluir a otimização da colocação dos componentes e o comprimento e distribuição da tubulação, satisfazendo restrições de segurança.

Devido à complexidade e ao elevado tempo de cálculo requerido para esse tipo de otimização, a otimização do *layout* de toda a planta pode ser dividida em três etapas: divisão da planta em módulos, otimização do posicionamento dos componentes e módulos, e otimização da distribuição da tubulação (GUIRARDELLO, 1993, 1994). Em cada etapa, um modelo na forma de uma programação mista, inteira e linear é utilizado para se resolver o problema, e métodos específicos são utilizados para se resolver cada modelo.

Nesse trabalho propôs-se desenvolver um modelo mais eficiente para a etapa de otimização do posicionamento dos componentes e módulos de uma planta química. Essa é justamente a etapa mais difícil do ponto de vista computacional, justificando assim um aprofundamento na busca de métodos mais eficientes.

O capítulo 2 apresenta a revisão bibliográfica, onde é destacada a importância do *layout* e os modelos empregados para otimização do mesmo, citados na literatura.

No capítulo 3 são apresentados os modelos matemáticos propostos por este trabalho, formulando-se o problema na forma de MILP.

O capítulo 4 aborda o método de resolução empregado para resolver o problema formulado em MILP, isto é, o algoritmo *Branch and Bound* (B&B) e sobre o pacote comercial, *GAMS/CPLEX*.

O capítulo 5 apresenta os casos de estudo e os dados empregados para a implementação dos modelos através do pacote *GAMS/CPLEX*.

Os capítulos 6 e 7 apresentam os resultados e a discussão dos mesmos para os casos de estudo e, finalmente, as conclusões obtidas na otimização do posicionamento de componentes em *layout* de plantas químicas, que é o tema deste trabalho.

1. 2. Objetivos

Este trabalho teve por objetivo:

- Desenvolver um modelo matemático mais eficiente para a otimização do posicionamento dos componentes do *layout* de uma planta química, capaz de incorporar um número de equipamentos que possa ser resolvido de forma exata, em um tempo aceitável.
- Realizar a modelagem em programação mista, inteira e linear (MILP), resolvendo-se com o auxílio do pacote comercial de otimização *GAMS*.
- Testar os modelos nos três casos-estudo propostos na literatura: uma unidade petroquímica (LEESLEY, 1982; GUIRARDELLO, 1993), uma planta de recuperação de mercúrio de efluentes eletrolíticos (AMORESE et al., 1977) e uma unidade de polimerização (GUNN, 1970).

CAPÍTULO 2

REVISÃO BIBLIOGRÁFICA

2.1. O *layout* e a sua importância

Um *layout* (arranjo físico) adequado é aquele que propicia a melhor utilização do espaço disponível, resultando em um processamento mais efetivo e seguro, através da menor distância, no menor tempo possível (VIEIRA, 1983).

O processo de elaboração do *layout* consiste na formulação e análise do problema, pesquisa de diferentes desenhos, seleção e especificação do modelo de *layout* a ser instalado.

A informação necessária para a elaboração de um *layout* de uma planta provém do fluxograma de processo, desenhos dos tanques com elevação e das conexões dos tubos, entre outras especificações dos equipamentos (KERN, 1977). Há ainda a necessidade de conhecimentos nas áreas de engenharia mecânica e química, bem como considerações de desenho geométrico (KERN, 1978). Além disso, são necessários conhecimentos sobre o produto, o processo e a programação da produção (FRANCIS & WHITE, 1974).

Portanto, o desenvolvimento adequado do *layout* de uma planta química faz parte de um processo mais genérico, isto é, do desenvolvimento do projeto de um processo químico. Este, por sua vez, envolve seis etapas principais (ULRICH, 1984):

1. *Concepção e definição do problema*:- deve-se conhecer as bases e hipóteses a serem aplicadas, a capacidade da planta e a distribuição de tempo. Nesta etapa, questionam-se os objetivos reais da planta, sua viabilidade econômica e sua flexibilidade para eventuais mudanças de processo.

2. *Desenvolvimento do fluxograma de processo*:- permite a familiarização com o processo, reduzindo a complexidade do problema e possibilitando identificar quais as informações ainda estão faltando.

3. *Projeto dos equipamentos*:- é necessário, pelo menos, o projeto parcial, pois o custo do equipamento é um elemento importante na economia do processo. Além disso, os vendedores ou fabricantes precisam de informações mais detalhadas, bem como do *layout* do processo, materiais de construção e espessura dos tanques, etc.

4. *Análise econômica*:- "Qual o retorno a ser esperado em relação ao dinheiro investido?" A resposta a essa pergunta envolve custos de processamento, matéria-prima, equipamentos, trabalho, além de fatores como inflação, taxas, etc. que influenciam na lucratividade do processo como um todo.

5. *Otimização*:- uma combinação de economia e engenharia. Busca-se o ótimo em relação a todos os fatores envolvidos no problema. (Aqui, pode-se incluir a otimização do posicionamento dos componentes do *layout* de plantas químicas, que é o objetivo deste trabalho.)

6. *Apresentação de relatório*:- é a apresentação organizada e clara de todo o esforço de trabalho realizado, garantindo o esclarecimento de dúvidas.

Geralmente, propõe-se um estudo do *layout* em situações onde há (VIEIRA, 1983):

- Obsolescência das instalações:- as instalações existentes tornam-se ineficientes devido à necessidade de se fabricar produtos novos, adquirir mais equipamentos, mudar o processo de fabricação e a necessidade de maior espaço para estocagem ou instalação de novas seções, etc.;
- Redução dos custos de produção;
- Variação na demanda: isto pode exigir um aumento ou um decréscimo na produção;

- Ambiente de trabalho inadequado devido aos ruídos, temperaturas anormais, pouca ventilação, má iluminação, que prejudicam o rendimento do trabalhador, bem como podem causar acidentes;

- Manuseios excessivos que provocam estragos nos materiais de produção e atrasam a produção, devido aos problemas no *layout*;

- Instalação de uma nova fábrica :- um planejamento do *layout* evita problemas futuros. Logo, a finalidade da fábrica, a quantidade a produzir, o tipo de trabalho, o espaço disponível, etc. devem ser bem definidos, previamente.

Segundo MOURA (1983), o fluxo de materiais é um dos fatores mais importantes na determinação do tamanho, forma e arranjo geral de qualquer local de fabricação. Além de determinar o arranjo das máquinas. Enquanto o *layout* está relacionado com a análise, planejamento e projeto das instalações utilizadas na produção de bens e serviços; a movimentação de materiais está ligada à fase das operações que envolvem o movimento dos materiais usados no desempenho das atividades do empreendimento. Desta forma, o mesmo autor, considera que a movimentação de materiais é a parte principal de quase todo o trabalho de *layout*.

Em suma, pode-se dizer que o estudo do *layout* da planta inclui objetivos como (FRANCIS & WHITE, 1974):

1. minimizar o investimento em equipamento
2. minimizar o tempo total de produção
3. utilizar o espaço existente de maneira mais efetiva
4. propiciar facilidade, segurança e conforto aos empregados
5. manter flexibilidade na disposição e operação
6. minimizar o custo com o manuseio (movimentação em geral) dos materiais
7. minimizar a variação de tipos de equipamentos para o manuseio dos materiais
8. facilitar o processo de fabricação
9. facilitar a estrutura organizacional.

Mas, além destes objetivos, existem restrições ao problema que devem ser incluídas. Por exemplo, restrições acerca dos níveis permissíveis de ruído, padrões referentes ao espaço entre os equipamentos, à temperatura do ambiente, à ventilação, à

iluminação; além da geometria da construção já existente, ou se for de uma planta nova, do terreno a ser construído.

2. 2. Classificação do *layout*

O *layout* pode ser classificado em três tipos básicos (FRANCIS & WHITE, 1974; VIEIRA, 1983; JAMES & ALCORN, 1991):

- 1) *Layout* de produto ou LINEAR
- 2) *Layout* de processo ou FUNCIONAL
- 3) *Layout* posicional ou FIXO

2. 2. 1 . *Layout de produto ou LINEAR*

O *layout* de produto baseia-se especificamente nos próprios produtos a serem produzidos, em processos contínuos. Portanto, as máquinas e as atividades auxiliares são organizadas de acordo com a seqüência de operações necessárias ao processamento dos produtos.

Este tipo de *layout* é aplicado em fábricas de montagem, onde o material move-se, enquanto as máquinas permanecem fixas.

Sua principal desvantagem é a descontinuidade na produção quando alguma máquina pára. Mas, em contrapartida, apresenta vantagens como a menor exigência de treinamento e qualificação profissional, bem como a menor exigência de espaço para o trabalho e a estocagem, reduzindo o volume e o manuseio do material em processo, além de facilitar o controle de supervisão, tornando possível a automação.

O *layout* de produto é empregado em indústrias automobilísticas, refinarias de petróleo, fábricas de produtos químicos, entre outras.

2. 2. 2. *Layout de processo ou FUNCIONAL*

O *layout* de processo geralmente é empregado em processos de batelada e, portanto, para a produção de diferentes produtos.

As máquinas são agrupadas de modo a realizar operações análogas, em um mesmo local. O material move-se através de seções especializadas.

No *layout* de processo, as máquinas empregadas são normalmente de uso geral, permitindo, assim, uma maior flexibilidade quanto às mudanças na seqüência das operações. No entanto, os trabalhadores precisam ser altamente qualificados devido à diversidade de produtos que pode ser produzida e há grande manuseio, podendo causar estragos no material.

As fábricas de sapato e as indústrias têxteis são exemplos de aplicação deste tipo de *layout*.

2.2. 3. *Layout posicional ou FIXO*

O *layout* posicional ou fixo também é conhecido por *layout* de produto estático, pois o produto fica parado enquanto os operadores e as máquinas se movimentam, uma vez que o produto a ser fabricado não pode ser movido facilmente.

O *layout* posicional é empregado na fabricação de navios e aviões, na construção civil, na montagem de grandes máquinas, etc.

Apesar da classificação acima, geralmente existe uma combinação dos três tipos de *layout*.

Um procedimento de planejamento de *layout* sistemático (SLP) foi desenvolvido por Muther, em 1961 (Apud: FRANCIS & WHITE, 1974) e pode ser aplicado a uma variedade de problemas envolvendo produção, transporte, estocagem, serviços de suporte e atividades de escritório. O procedimento SLP é apresentado na Figura 1, onde as etapas de 1 a 5 fazem parte da análise do problema; as etapas 6 a 9, da pesquisa e a etapa 10, a seleção.

Portanto, a elaboração de um bom *layout* é uma etapa importante no projeto de uma nova planta ou na "remodelação" de uma planta já existente. O *layout* tem grande impacto sobre a segurança, o meio ambiente, as condições de operação e de acesso para a manutenção das plantas químicas, afetando o custo total das mesmas. Assim sendo, a sua otimização é essencial.

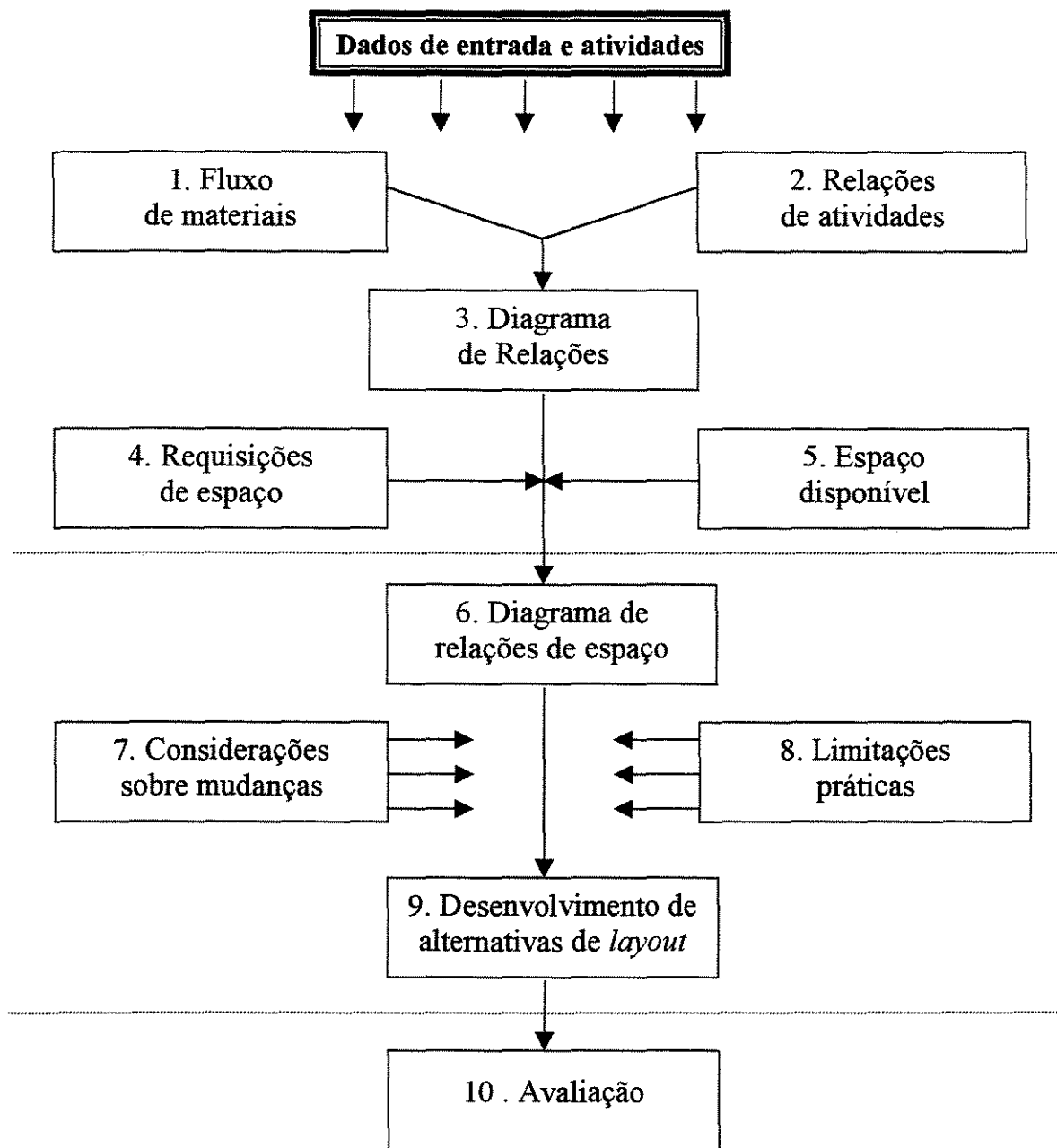


Figura 1. Procedimento SLP ("Systematic *Layout* Planning").

Fonte: FRANCIS & WHITE, 1974.

A otimização de processos ou de *layouts* é uma preocupação antiga da engenharia, uma vez que está intimamente relacionada à redução de custos. No entanto, nos dias atuais, ainda é considerada como parte essencial e básica frente aos rápidos avanços tecnológicos,

que obrigam a buscar alternativas melhores (mais econômicas, seguras e com resultados rápidos) para garantir produtos finais com preços competitivos.

Neste sentido, GREEN et al. (1999) comentam sobre uma abordagem de intensificação de processos, visando realizar plantas menores, mais baratas e mais seguras. O conceito de intensificação de processo surgiu devido à constatação que grande parte do capital empregado em uma planta química é gasto em sua estrutura de engenharia civil. Assim, reduções significativas nestes custos podem ser alcançadas reduzindo-se o tamanho desta planta. O processo de otimização considera as condições e os procedimentos de operação correntes e faz pequenos ajustes nos mesmos para produzir uma melhoria. Esse trabalho se aplica às plantas já existentes e que necessitam de uma reestruturação.

Quando a planta ainda não existe, LUDWIG (1995) cita que o engenheiro tem uma responsabilidade importante desde a seleção do terreno e do *layout* da planta, pois muitas das decisões referentes a disposição física dos prédios (construções em geral) e dos equipamentos requerem um conhecimento da sequência das operações de um determinado processo, bem como dos fatores de risco, tais como: explosão, fogo, toxicidade, etc.

Logo, existem diferentes modelos matemáticos que envolvem tais restrições e são aplicados ao processo de otimização.

2. 3. Modelos matemáticos para *layout* de processos químicos

Os modelos matemáticos podem ser usados para se obter o *layout* de processo mais econômico, sujeito a restrições que possibilitem a operação, manutenção e segurança (GUIARDELLO, 1993). Portanto, todos os modelos objetivam minimizar o custo total, desde que satisfaçam as restrições de espaço livre ao redor dos componentes.

Conforme GUIARDELLO (1993, 1994), a otimização do *layout* pode ser dividida em três etapas:

* ***Divisão da planta em módulos:*** agrupamento dos equipamentos que possuem maior número de conexões entre si (ou custo unitário de tubulação), de tal forma a minimizar a distância e o custo de tubulação colocando-se esses equipamentos em um mesmo módulo. O objetivo é a divisão da planta em módulos diferentes.

* **Otimização do posicionamento dos componentes e módulos:** emprego da otimização para posicionar os componentes do *layout* e, em seguida, os módulos da planta. Para tanto, é necessário satisfazer as restrições de distância mínima entre os componentes e minimizar os custos relacionados ao *layout*. Nesta etapa, utiliza-se um modelo matemático que permite diferentes rotações e orientações dos equipamentos, bem como diferentes posições relativas entre os pares de componentes (em três dimensões).

* **Otimização da distribuição da tubulação:** define a localização da tubulação através da construção de uma rede de pontos (vértices) e arcos (conectando os pontos), de tal forma a evitar a interferência com os equipamentos e com outras regiões restritas. A otimização é feita para cada tubo, encontrando-se o percurso de menor custo entre os pontos que correspondem aos extremos do tubo.

A parte mais difícil do problema é a otimização do posicionamento dos componentes principais, devido à não-convexidade da região viável, pois as posições de dois componentes não devem se sobrepor, e à dificuldade da complexidade computacional.

A complexidade computacional de um algoritmo usado para resolver um problema é medida pelo seu tempo de execução, ou seja, o tempo que leva para produzir a resposta final (GUIARDELLO, 1993). Para um problema de tamanho N , a complexidade será definida pelo comportamento do pior caso possível dentre todas as entradas de tamanho N . Um problema é chamado de polinomial quando existe um algoritmo para resolver esse problema no qual o pior caso possível de complexidade de tempo é limitado por um polinômio de grau N (classe P). Um problema é chamado de NP , polinomial não-determinístico (*non-deterministic polynomial*), quando, se uma solução existe, ela pode ser construída e verificada em tempo polinomial (classe NP). Em geral, os problemas da classe NP incluem todos aqueles que levam tempo exponencial em N , mas não está provado que eles não possam ser resolvidos por algoritmos polinomiais. A classe dos problemas NP -completo é a mais difícil de todas, pois inclui todos os problemas para os quais não se conhece nenhum algoritmo polinomial para resolvê-los.

No caso da otimização de *layout*, as etapas de divisão em módulos e otimização da distribuição da tubulação podem ser resolvidas por algoritmos polinomiais (classe P), e são assim consideradas etapas fáceis do ponto de vista computacional. Já a etapa de otimização

do posicionamento dos componentes e dos módulos resulta em modelos que pertencem à classe dos problemas *NP-completo*. Mesmo um modelo simples de otimização de *layout* como o “quadratic assignment problem” é do tipo *NP-completo*. Do ponto de vista prático isso significa: (i) ou utilizar um algoritmo exato para se encontrar a solução ótima de um problema com N pequeno, (ii) ou utilizar um algoritmo heurístico para se encontrar uma boa solução de um problema com N grande.

Um exemplo de algoritmo que fornece respostas aproximadas para problemas de *layout* de tamanho razoável é o algoritmo de “simulated annealing”. Para modelos na forma de “quadratic assignment problem” esse tipo de algoritmo fornece respostas muito próximas do ótimo, mas para problemas na forma de programação mista, inteira e linear (MILP) as respostas não são muito boas (GUIRARDELLO, 1993). Nesse caso, é preferível utilizar um algoritmo exato para resolver problemas de *layout* dados por uma MILP, com a conseqüente limitação do tamanho do problema a ser resolvido.

Anteriormente, GUNN (1970) já havia desenvolvido um método no qual a planta era dividida em módulos, objetivando-se minimizar os custos de tubulação e do local de construção (prédio/terreno), através do emprego de um método de pesquisa direta (método de Rosenbrock).

A divisão da planta em módulos foi realizada no intuito de satisfazer a necessidade de acessibilidade para a operação, manutenção e segurança e, ainda, para evitar a computação de um número enorme de diferentes arranjos dos componentes, caso todas as possíveis combinações fossem incluídas.

A complementação deste método foi realizada por GUNN & AL-ASADI (1985), aplicando-o em uma planta de batelada de poliéster, para encontrar o *layout* ótimo. Cada módulo foi definido como um paralelepípedo, contendo as coordenadas e posições relativas de cada bocal e evitando a sobreposição. O problema foi resolvido através de um programa computacional e o *layout* da planta baseou-se no fluxograma de processo.

AMORESE et al. (1977) aplicaram o método heurístico para obter um *layout* compacto, alocando os componentes do processo químico, empregando-se de um procedimento de aprendizagem (inteligência artificial) para escolher a melhor regra heurística. O modelo foi desenvolvido em um plano bidimensional.

Segundo AMORESE et al. (1977), um *layout* pode ser obtido através de dois procedimentos alternativos:

- distribuir os componentes sobre um plano e, então, reduzir a distância entre eles até que os limites de não interferência sejam violados ou

- dispor os componentes em uma certa ordem e, em seguida, colocá-los um a um num plano na melhor posição possível.

Este procedimento heurístico não garante achar uma solução ótima global.

Neste sentido, GEORGIADIS & MACCHIETTO (1997) acreditam que os métodos heurísticos tradicionais, usados para disposição de equipamentos dentro de plantas químicas, não são sistemáticos e não consideram todos os dados apropriados e relevantes ao problema, além de conduzi-lo à soluções sub-ótimas. Logo, propuseram uma abordagem baseada na otimização para obter um bom *layout* preliminar, sujeito às restrições de segurança, meio ambiente, construção, manutenção e operação. O problema foi formulado como um modelo de programação mista, inteira e linear (MILP).

No entanto, os mesmos autores admitem que para plantas com mais de trinta componentes, o custo computacional pode crescer consideravelmente, logo, sugerem o método heurístico ou uma solução alternativa que seria a de estabelecer “módulos de equipamentos segregados”. Assim, um módulo incluiria um grupo de unidades com características operacionais semelhantes ou unidades que necessitam estar interligadas. Após a escolha dos módulos e a especificação dos equipamentos em cada módulo, bem como da definição de suas posições relativas, o problema de *layout* da planta visaria decidir as posições relativas de cada módulo da planta.

Tendo observado que um dos principais pontos fracos da maioria dos métodos citados na literatura consiste em não levar em consideração o tamanho e a geometria dos equipamentos e, ainda, a sua orientação, GEORGIADIS et al. (1999) estenderam seu trabalho anterior focalizando este aspecto.

O problema é abordado em um dado espaço bidimensional ou tridimensional, considerando-se os custos de terreno, tubulação, transporte e de construção, além dos diferentes tamanhos e formas geométricas retangulares dos equipamentos de processo, permitindo a sua devida orientação. A forma retangular é considerada mais apropriada para problemas industriais do que as formas circulares, pois, desta forma, algum espaço extra é incluído para permitir o acesso para manutenção, inspeção, etc. A formulação deste problema é realizada como MILP, novamente, sendo implementado em linguagem de programação C++, ambiente de objetos orientados e resolvido usando o *solver CPLEX*.

Mais recentemente, SCHMIDT-TRAUB et al.(1999) apresentaram uma abordagem de otimização para *layout* empregando a ferramenta CAD (“computer-aided design”), para obter uma visualização tridimensional mais precisa. Seu objetivo principal é o de fornecer padrões e procedimentos heurísticos que auxiliem o engenheiro, considerando, às vezes, a necessidade de dados estatísticos e de alguns algoritmos.

SCHMIDT-TRAUB et al.(1999) consideram que para desenvolver um sistema CAD para o *layout* da planta são necessários conhecimentos prévios sobre a disposição espacial do equipamento, bem como obter informações de engenheiros mais experientes, além da análise de plantas já elaboradas.

Além disso, citam que a concepção do *layout* consiste em três módulos principais: modelagem do equipamento, modelagem do *layout* e disposição dos tubos, incluindo ainda a análise dos resultados gerados. A Figura 2 apresenta as informações inerentes a cada módulo citado e à análise.

Na etapa de modelagem do equipamento, são estimadas as exigências de espaço para o próprio equipamento, incluindo o espaço para a tubulação e o espaço necessário ao acesso local. Os resultados são elementos geométricos simples, tais como cubos ou cilindros que descrevem o equipamento. Além disso, são definidas as restrições operacionais, que provêm das condições de segurança, operação, manutenção e inspeção e, ainda, são gerados os pontos de conexão às tubulações do processo, a fim de serem otimizados.

Na etapa de modelagem do *layout*, o mesmo é obtido através do arranjo adequado da posição de cada equipamento dentro da estrutura da planta. Para tanto, após a definição das restrições, são formuladas as equações matemáticas pertinentes.

Quanto à disposição de tubos e à análise, são propostos algoritmos para a resolução e analisados os dados gerais sobre a concepção do *layout*, derivados de procedimentos heurísticos ou estatísticos, respectivamente.

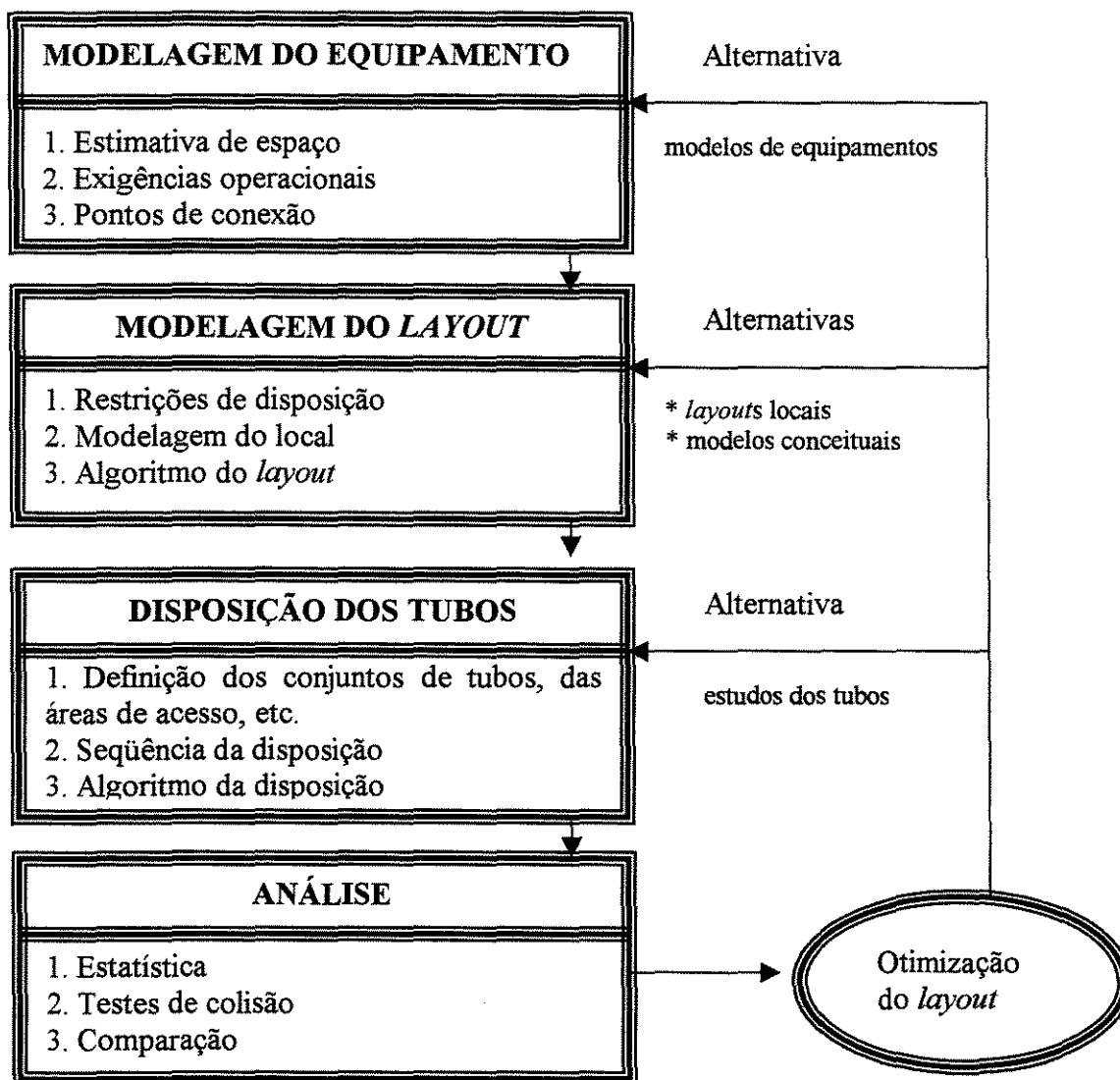


Figura 2. Estrutura do sistema.

Fonte: SCHMIDT-TRAUB et al. (1999).

O critério de análise aplicado ao modelo de *layout* é ilustrado na Figura 3.

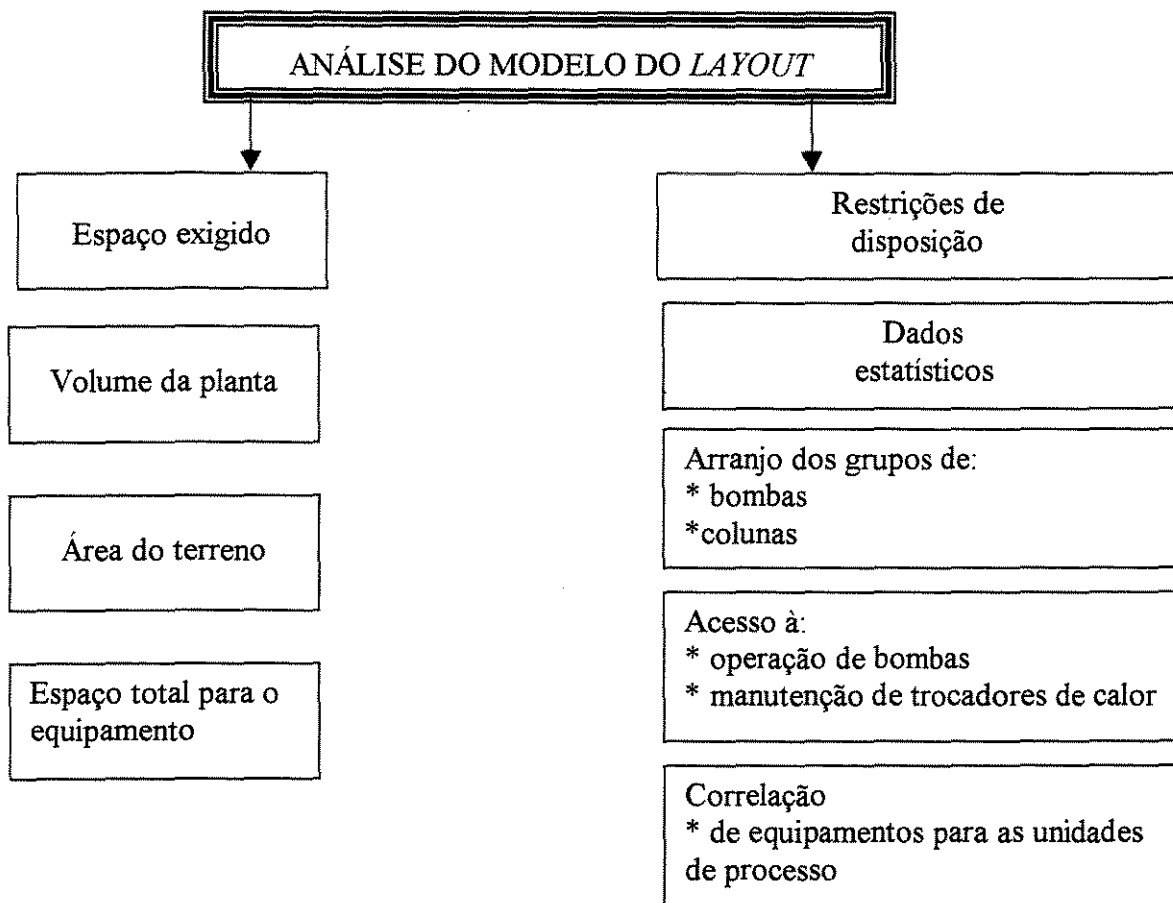


Figura 3. Critério para a análise dos modelos de *layout*.

Fonte: SCHMIDT-TRAUB et al. (1999).

2. 4. Códigos computacionais para *layout*

A aplicação de computadores ao desenvolvimento de *layouts* pode facilitar a sua elaboração, diminuindo o tempo de projeto, permitindo a sua visualização e, ainda, possibilitando realizar a otimização matemática mais rapidamente. Os sistemas desenvolvidos em CAD ("computer-aided design") facilitam muito a visualização do *layout*.

Quando empregados na otimização, o computador usa métodos matemáticos para localizar o melhor *layout* para a planta de processo. MECKLENBURGH (1982, 1985) revisou programas usados para a otimização econômica e identificou os seguintes tópicos:

- * Os métodos disponíveis não são capazes de conceber um *layout* ou trabalhar com grandes mudanças nas posições relativas dos componentes, sem a interação com o *designer* (intuição do *designer*).

- * O custo é calculado como uma função de espaço, tubos e outras conexões quando os detalhes do *layout* são inseridos ao computador. Somente aplicações limitadas deste tipo são disponíveis comercialmente, usualmente com sistemas computacionais interativos.

- * A otimização dos custos de espaço e tubos é prática se o *designer* divide a planta em seções (módulos). A otimização é feita independentemente, dentro de cada seção individual e entre todas as seções. Não são disponíveis sistemas comerciais para otimização.

- * A disposição automática dos tubos é viável e comercialmente disponível. O computador pode arranjá-los no intuito de evitar colisões com os componentes e as áreas restritas para acesso ou para outros propósitos. Esta é uma vantagem para usar rotas de tubos maiores.

- * É desejável ter uma rotina para detectar colisões entre os componentes e violações das áreas restritas.

- * No critério de otimização é possível incluir outras características além da tubulação.

Há diversos métodos e algoritmos computacionais desenvolvidos para *layout* de equipamentos que também podem ser interativos ou usar técnicas de otimização para alocação dos componentes (GUIARDELLO, 1993). Os algoritmos computacionais são classificados em dois tipos (FRANCIS & WHITE, 1974): algoritmos de construção e algoritmos de melhora (aperfeiçoamento).

O primeiro tipo consiste na seleção e disposição de equipamentos até que o projeto (desenho) do *layout* seja obtido. O segundo baseia-se na completa existência de um *layout* inicial, onde a disposição dos equipamentos é trocada a fim de aperfeiçoar o desenho do *layout*.

Alguns algoritmos de construção são os seguintes: CORELAP, ALDEP, PLANET, RMA COMP I, LSP, LAYOPT. No entanto, os mais representativos são CORELAP e ALDEP, conforme a literatura. CRAFT pode ser citado como exemplo de um algoritmo de melhora.

JAMES & ALCORN (1991) comentam sobre quatro modelos computacionais para *layouts*, inclusive sobre os já citados CORELAP, ALDEP e CRAFT:

- "Técnica de tempo elementar máximo sucessivo": é um modelo que opera conforme a abordagem PERT ("Program Evaluation and Review Technique"), determinando a quantia máxima de tempo de cada elemento no sistema de produção a ser considerado e, assim, desenvolvendo um modelo global do tempo necessário e distribuindo-o para completar a manufatura. As informações de tempo são os dados de entrada e o resultado de saída é um modelo global de manufatura baseado na máxima eficiência de tempo gasto no processo.
- Modelo CRAFT ("Computerized Relative Allocation of Facilities Technique"): este modelo usa um conjunto de parâmetros de projeto pelos quais ele pode definir se as posições de elementos são aceitáveis ou não, dentro do processo de produção. É denominado como um modelo heurístico que arranja e rearranja os elementos do sistema, visando buscar o *layout* ótimo. O *layout* é determinado pela minimização dos custos de transporte, manuseio e movimentação dos materiais em termos de distância, tempo, equipamento e assim por diante. O modelo CRAFT leva em conta as limitações dos equipamentos e utilidades, tais como dimensões totais, características de construção e forma. O programa faz uma série de arranjos até obter um custo mínimo de transporte (movimentação e manuseio) dos materiais.
- Modelo CORELAP ("Computerized Relationship *Layout* Planning"):- opera similarmente ao CRAFT, que é heurístico. Ele determina o *layout* global mais efetivo baseado nas relações entre os equipamentos e as etapas do processo de produção, sendo que tempo, distância e fatores de custo são alguns dos seus parâmetros essenciais. O *layout* é criado em função das relações que existem e serve para determinar a quantia mínima de espaço e tempo necessários para produzir o produto.
- Sistema ALDEP ("Automated *Layout* Design Programs"): é uma versão do programa CAD; integra o conjunto completo de características de desenho do *layout*, usando algumas ou todas dentre essas características como parâmetros para obter o melhor

"design" dos equipamentos. Parâmetros possíveis incluem características do produto, padrões de fluxo do produto, necessidades envolvendo o transporte dos materiais, controle do estoque, uso do espaço, necessidade de capital e custo de produção, etc. Combinando estes elementos, o ALDEP compara várias combinações, sucessivamente, até obter o melhor *layout*. A escolha dos fatores a serem usados como parâmetros é de suma importância e constitui a parte mais crítica desta programação.

Segundo FRANCIS & WHITE (1974), CRAFT é usado quando o fluxo de materiais é dominante, pois emprega procedimentos que visam minimizar o custo de transporte, manuseio e movimentação dos materiais. Por outro lado, ALDEP e CORELAP são usados quando o fator principal a ser considerado é as inter-relações de atividades e quando as constantes condições de mudança impedem a obtenção de dados numéricos precisos.

MEREDITH (1987) observa que o algoritmo CRAFT não garante o *layout* de menor custo, mas alerta que podem existir soluções melhores do que a encontrada. Ainda, cita a desvantagem de que o mesmo não é válido quando se necessita mudar a construção da fábrica e não somente a disposição dos equipamentos e processos. Para tanto, são necessários arranjos manuais. No entanto, sua vantagem é a possibilidade de se fixar algumas áreas específicas antes de realizar um novo arranjo. Além disso, alguns resultados demonstraram uma redução de custo de 20% em relação ao *layout* inicial.

Existem, ainda, outros algoritmos (GUIRARDELLO, 1993), tais como: PLANET ("Plant *Layout* and Evaluation Technique"), COFAD ("Computerized Facilities Design"), CAFLAS ("Computer Aided Facility Layout Selection"), etc.

Para *layout* de plantas de processo existem o "Intergraph PDS-Plant Design System" e o PEGS/PDMS ("Project Engineering Graphics System" / "Plant Design Management System"), que são *softwares* comerciais interativos, isto é, o projetista muda a posição dos componentes.

CAPÍTULO 3

MODELO MATEMÁTICO

3. 1. Considerações do Modelo

O modelo inicial, a partir do qual algumas mudanças foram realizadas ao longo do trabalho, foi extraído de GUIRARDELLO (1993). Independentemente dessas alterações, o problema de *layout* é modelado considerando-se os componentes como sólidos retangulares (em três dimensões), conectados por linhas que representam os tubos e partindo de pontos especificados na superfície dos mesmos. Esses sólidos não podem ocupar a mesma posição no espaço.

O problema da segurança e do espaço para operação e manutenção é considerado estabelecendo-se restrições de distância mínima entre componentes, utilizando-se a distância de Manhattan (norma 1).

O problema do custo é considerado atribuindo-se custos unitários para tubulação (por unidade de comprimento), suporte de componentes e tubulação (por unidade de altura e área), e área física da planta (por unidade de área).

Assim, a otimização do layout é feita minimizando-se o custo global, satisfazendo-se as restrições de espaço para segurança, operação e manutenção, além das restrições de posição no espaço, e outras restrições adicionais que podem ser consideradas. Embora seja

feita a inclusão do custo na função objetivo e da segurança nas restrições, isto não implica que o custo seja mais importante que a segurança, já que as restrições são tão importantes quanto a função objetivo.

Ainda, neste modelo proposto por GUIRARDELLO (1993), algumas simplificações foram feitas, com o objetivo de se obter funções lineares das variáveis:

- O custo da área da planta foi substituído por um custo de perímetro, de tal forma a obter a menor área da planta através do menor perímetro. O custo C^{per} considera uma transformação do custo por área de um quadrado em custo por perímetro (aproximação).
- O custo da tubulação foi substituído por um custo aproximado que considera somente a distância de Manhattan (norma 1) entre as posições dos bocais, sem considerar se estão atravessando os componentes ou outros tubos; esse custo tende a aproximar os componentes, mas não fornece o custo real da tubulação, que precisa ser determinada em uma etapa posterior.
- O custo de suporte de cada componente é considerado em função de sua área (fixa) e de sua altura, que pode variar; como referência, utilizou-se a área do topo do equipamento.

3. 2. Modelo Matemático

O modelo matemático utilizado permite diferentes rotações e orientações dos equipamentos (Figura 4), bem como diferentes posições relativas entre pares de componentes (em três dimensões). Essas rotações (orientações) e posições relativas são representadas no modelo por variáveis binárias inteiras (0 ou 1). A distância entre os equipamentos é dada de forma a considerar os percursos perpendiculares da tubulação (como é o caso de uma planta química real). Dessa forma, o problema poderá ser colocado na forma de um modelo de programação mista, inteira e linear.

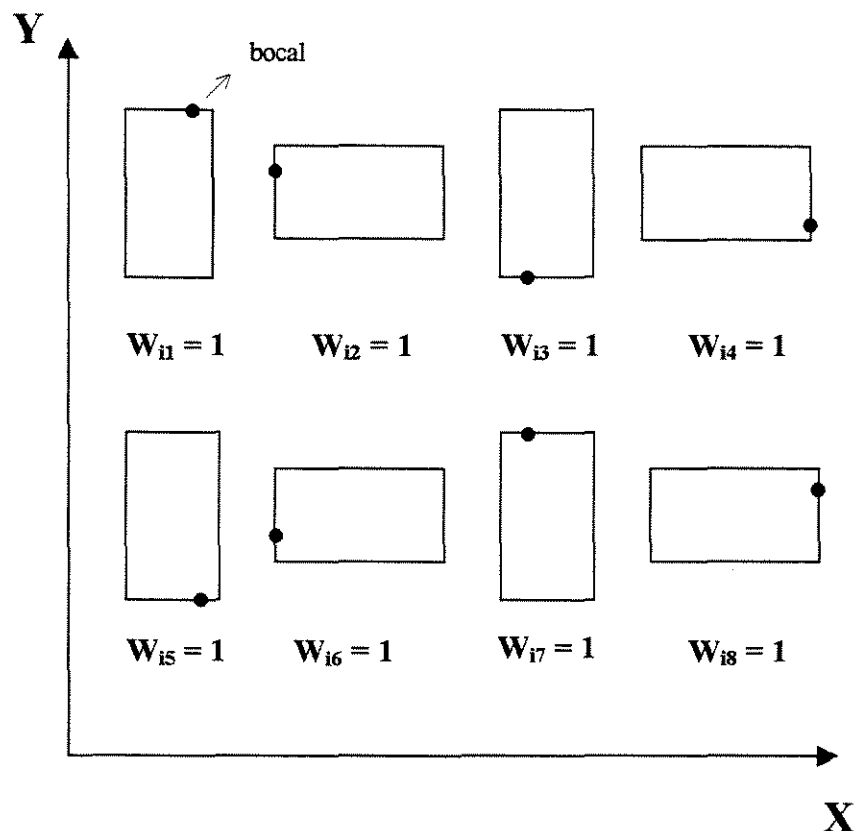


Figura 4. As oito rotações possíveis no plano x-y.

Fonte: GUIRARDELLO (1993).

A Figura 4 apresenta as posições possíveis dos bocais nos componentes, refletindo as diferentes rotações como se fossem reflexões de um espelho.

A otimização do posicionamento de componentes e módulos é a etapa mais difícil no desenvolvimento do layout. O tempo total de cálculo depende do número de componentes e do número de tubos entre os componentes. A dificuldade de resolução da otimização está relacionada principalmente com as variáveis inteiras de posição relativa, pois o conjunto de soluções factíveis não é convexo e existe um número muito grande de pontos ótimos locais que precisam ser examinados para se encontrar o ponto ótimo global.

Os problemas são modelados através da programação mista, inteira e linear (MILP), que são problemas lineares de otimização envolvendo variáveis discretas (inteiras)

e contínuas. A forma geral de um problema MILP (*Mixed Integer Linear Programming*) pode ser expressa como:

$$\text{Maximize } Z = \sum_{j=1}^n c_j x_j, \quad (1)$$

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

$$\text{e } x_j \geq 0, \text{ para } j = 1, 2, \dots, n, \quad (3)$$

$$x_j \text{ é inteiro, para } j = 1, 2, \dots, I; I \leq n. \quad (4)$$

Os problemas MILP são lineares na função objetivo e nas restrições.

3. 3. Distâncias Mínimas de Segurança

Distâncias mínimas de segurança entre componentes são determinadas para cada caso específico. Quando isto não é possível, essas distâncias podem ser estimadas a partir de tabelas disponíveis na literatura. Essa informação pode ser apresentada como uma distância mínima entre pares de equipamentos (ANDERSON, 1982; BUSH E WELLS, 1972; KAURA, 1980; E MECKLENBURGH, 1985), ou como uma distância mínima em torno de cada equipamento (BUSH E WELLS, 1972; KAURA, 1980; E MECKLENBURGH, 1985).

3. 4. Custos Unitários

Custos unitários de tubulação, suportes e tamanho da planta devem ser obtidos para cada caso específico. Quando isto não for possível, esses custos podem ser obtidos a partir de informação disponível na literatura (GUTHRIE, 1974; PETERS E TIMMERHAUS, 1991). Os custos que mais influenciam a otimização do layout são os custos de tubulação e os custos de suportes para componentes e tubulação.

3. 5. Modelagem implementada em GAMS (Modelo Original)

O modelo implementado no *software* comercial de otimização, *GAMS*, foi extraído de GUIRARDELLO (1993) e consiste de uma função objetivo de minimização do custo total da planta (Eq.5), quanto ao posicionamento dos componentes no *layout*, e de uma série de restrições visando, principalmente, a segurança e o fácil acesso à planta.

$$\begin{aligned}
 \min \quad & 2 C^{per} (x + y) + \sum_{i=1}^{N_{cp}} c_i^s + \sum_{k=1}^{N_{pi}} C_k^p [l_k^{x+} + l_k^{x-} + l_k^{y+} + l_k^{y-} + l_k^{z+} + l_k^{z-}] \\
 & x, y, z, C_i^s \\
 & l_k^{x+}, l_k^{x-}, l_k^{y+}, l_k^{y-}, l_k^{z+}, l_k^{z-} \\
 & x_i, y_i, z_i, x_p^n, y_p^n, z_p^n \\
 & d_{ij}^x, d_{ij}^y, d_{ij}^z \\
 & w_{ip}, w_{ij}^x, w_{ij}^y, w_{ij}^z
 \end{aligned} \tag{5}$$

A função objetivo acima (Eq. 5) inclui o custo de perímetro da planta, o custo de suportes de cada equipamento e o custo de tubulações. Esta função é sujeita às seguintes restrições:

- Eq. (6) - cálculo do custo de suporte de cada componente i através de uma função convexa linear por partes (4 segmentos):

$$C_i^s \geq C_{ik}^{sa} \cdot \left(z_i + \frac{1}{2} C_i \right) + C_{ik}^{sb} \tag{6}$$

$$i = 1, \dots, N_{cp} \qquad k = 1, \dots, 4$$

- Eq. (7) - cálculo da dimensão (*largura*) x da planta, a partir da posição máxima na direção x de todos os componentes:

$$x \geq x_i + \frac{1}{2} A_i \cdot (w_{i1} + w_{i3} + w_{i5} + w_{i7}) + \frac{1}{2} B_i \cdot (w_{i2} + w_{i4} + w_{i6} + w_{i8}) \tag{7}$$

$$i = 1, \dots, N_{cp}$$

- Eq. (8) - cálculo da dimensão (*comprimento*) y da planta, a partir da posição máxima na direção y de todos os componentes:

$$y \geq y_i + \frac{I}{2} B_i \cdot (w_{i1} + w_{i3} + w_{i5} + w_{i7}) + \frac{I}{2} A_i \cdot (w_{i2} + w_{i4} + w_{i6} + w_{i8}) \quad (8)$$

$$i = 1, \dots, N_{cp}$$

- Eq. (9) - cálculo da dimensão (*altura*) z da planta, a partir da posição máxima na direção z de todos os componentes:

$$z \geq z_i + \frac{I}{2} C_i \quad (9)$$

$$i = 1, \dots, N_{cp}$$

- Eq. (10, 11 e 12) - cálculo do limite mínimo para as posições centrais x_i , y_i , z_i de cada componente, respectivamente:

$$x_i \geq \frac{I}{2} A_i \cdot (w_{i1} + w_{i3} + w_{i5} + w_{i7}) + \frac{I}{2} B_i \cdot (w_{i2} + w_{i4} + w_{i6} + w_{i8}) \quad (10)$$

$$i = 1, \dots, N_{cp}$$

$$y_i \geq \frac{I}{2} B_i \cdot (w_{i1} + w_{i3} + w_{i5} + w_{i7}) + \frac{I}{2} A_i \cdot (w_{i2} + w_{i4} + w_{i6} + w_{i8}) \quad (11)$$

$$i = 1, \dots, N_{cp}$$

$$z_i \geq \frac{I}{2} C_i + E_i^I \quad (12)$$

$$i = 1, \dots, N_{cp}$$

- Eq. (13, 14 e 15) - cálculo do limite mínimo para as distâncias d_{ij}^x , d_{ij}^y e d_{ij}^z entre as posições centrais de cada par de componentes i e j , evitando sobreposição de equipamentos:

$$\begin{aligned}
d_{ij}^x \geq & \frac{1}{2} A_i \cdot (w_{i1} + w_{i3} + w_{i5} + w_{i7}) + \frac{1}{2} B_i \cdot (w_{i2} + w_{i4} + w_{i6} + w_{i8}) + \\
& + \frac{1}{2} A_j \cdot (w_{j1} + w_{j3} + w_{j5} + w_{j7}) + \frac{1}{2} B_j \cdot (w_{j2} + w_{j4} + w_{j6} + w_{j8}) + \\
& + [D_{ij}^h + M_{ij}] \cdot w_{ij}^x - M_{ij}
\end{aligned} \tag{13}$$

$$i, j = 1, \dots, N_{cp} \quad i \neq j$$

$$\begin{aligned}
d_{ij}^y \geq & \frac{1}{2} B_i \cdot (w_{i1} + w_{i3} + w_{i5} + w_{i7}) + \frac{1}{2} A_i \cdot (w_{i2} + w_{i4} + w_{i6} + w_{i8}) + \\
& + \frac{1}{2} B_j \cdot (w_{j1} + w_{j3} + w_{j5} + w_{j7}) + \frac{1}{2} A_j \cdot (w_{j2} + w_{j4} + w_{j6} + w_{j8}) + \\
& + [D_{ij}^h + M_{ij}] \cdot w_{ij}^y - M_{ij}
\end{aligned} \tag{14}$$

$$i, j = 1, \dots, N_{cp} \quad i \neq j$$

$$d_{ij}^z \geq \left[D_{ij}^y + \frac{1}{2} C_i + \frac{1}{2} C_j \right] \cdot w_{ij}^z \tag{15}$$

$$i, j = 1, \dots, N_{cp} \quad i \neq j$$

As Eq. (13), (14) e (15) atuam em conjunto com as Eq. (16), (17), (18), (19), (20) e (21).

- Eq. (16, 17 e 18) - cálculo do limite máximo para as distâncias d_{ij}^x , d_{ij}^y e d_{ij}^z . Estas equações evitam a sobreposição de equipamentos. Quando uma posição relativa entre i e j está ativa, a distância oposta tem que ser zero, senão as distâncias calculadas pelas Eq. (19, 20 e 21) poderiam ser tais que fariam as diferenças entre as posições centrais serem menores que o permitido.

$$d_{ij}^x \leq M \cdot (1 - w_{ji}^x) \tag{16}$$

$$i, j = 1, \dots, N_{cp} \quad i \neq j$$

$$d_{ij}^y \leq M \cdot (1 - w_{ji}^y) \tag{17}$$

$$i, j = 1, \dots, N_{cp} \quad i \neq j$$

$$d_{ij}^z \leq M \cdot (1 - w_{ji}^z) \quad (18)$$

$$i, j = 1, \dots, N_{cp} \quad i \neq j$$

- Eq. (19, 20 e 21) - cálculo das diferenças entre as posições centrais entre pares de componentes i e j , a partir das distâncias d_{ij}^x, d_{ij}^y e d_{ij}^z . Nota-se que se d_{ij} e d_{ji} em qualquer direção forem ambas diferentes de zero, a diferença entre as posições centrais será a diferença entre as duas.

$$x_i - x_j - d_{ij}^x + d_{ji}^x = 0 \quad (19)$$

$$i = 1, \dots, N_{cp} - 1 \quad j = i + 1, \dots, N_{cp}$$

$$y_i - y_j - d_{ij}^y + d_{ji}^y = 0 \quad (20)$$

$$i = 1, \dots, N_{cp} - 1 \quad j = i + 1, \dots, N_{cp}$$

$$z_i - z_j - d_{ij}^z + d_{ji}^z = 0 \quad (21)$$

$$i = 1, \dots, N_{cp} - 1 \quad j = i + 1, \dots, N_{cp}$$

- Eq. (22, 23 e 24) - cálculo das posições dos bocais de cada tubo, nas três coordenadas x, y e z , respectivamente:

$$x_p^n = x_i + F_{ji}^x \cdot \frac{1}{2} A_i \cdot (w_{i1} - w_{i3} + w_{i5} - w_{i7}) + F_{ji}^y \cdot \frac{1}{2} B_i \cdot (-w_{i2} + w_{i4} - w_{i6} + w_{i8}) \quad (22)$$

$$i = 1, \dots, N_{cp} \quad j = 1, \dots, N_{z_i} \quad p = j + \sum_{l=1}^{i-1} N_{z_l}$$

$$y_p^n = x_i + F_{ji}^y \cdot \frac{1}{2} B_i \cdot (w_{i1} - w_{i3} - w_{i5} + w_{i7}) + F_{ji}^x \cdot \frac{1}{2} A_i \cdot (w_{i2} - w_{i4} - w_{i6} + w_{i8}) \quad (23)$$

$$i = 1, \dots, N_{cp} \quad j = 1, \dots, N_{z_i} \quad p = j + \sum_{l=1}^{i-1} N_{z_l}$$

$$z_p^n = z_i + F_{ji}^z \cdot \frac{1}{2} C_i \quad (24)$$

$$i = 1, \dots, N_{cp} \quad j = 1, \dots, N_{z_i} \quad p = j + \sum_{l=1}^{i-1} N_{z_l}$$

- Eq. (25, 26 e 27) - cálculo dos comprimentos dos tubos a partir da posição dos bocais, nas três coordenadas x , y e z , respectivamente:

$$x_p^n - x_q^n - l_k^{x+} + l_k^{x-} = 0 \quad (25)$$

$$k = 1, \dots, N_{pi} \quad p = k_s(k) \quad q = k_t(k)$$

$$y_p^n - y_q^n - l_k^{y+} + l_k^{y-} = 0 \quad (26)$$

$$k = 1, \dots, N_{pi} \quad p = k_s(k) \quad q = k_t(k)$$

$$z_p^n - z_q^n - l_k^{z+} + l_k^{z-} = 0 \quad (27)$$

$$k = 1, \dots, N_{pi} \quad p = k_s(k) \quad q = k_t(k)$$

- Eq. (28) - cálculo da orientação dos componentes. Esta equação determina que apenas uma posição de rotação será escolhida para cada componente i :

$$\sum_{l=1}^8 w_{il} = 1 \quad (28)$$

$$i = 1, \dots, N_{cp}$$

- Eq. (29) - cálculo da posição relativa entre os componentes. Esta equação determina que apenas uma posição relativa será escolhida para cada par de componentes i e j :

$$w_{ij}^x + w_{ij}^y + w_{ij}^z + w_{ji}^x + w_{ji}^y + w_{ji}^z = 1 \quad (29)$$

$$i = 1, \dots, N_{cp} - 1 \quad j = i + 1, \dots, N_{cp}$$

- Eq. (30, 31, 32, 33, 34 e 35) - todas as variáveis citadas são não-negativas, isto é, maior ou igual a 0.

$$x, y, z \geq 0 \quad (30)$$

$$C_i^s \geq 0 \quad (31)$$

$$i = 1, \dots, N_{cp}$$

$$l_k^{x+}, l_k^{x-}, l_k^{y+}, l_k^{y-}, l_k^{z+}, l_k^{z-} \geq 0 \quad (32)$$

$$k = 1, \dots, N_{pi}$$

$$x_i, y_i, z_i \geq 0 \quad (33)$$

$$i = 1, \dots, N_{cp}$$

$$x_p^n, y_p^n, z_p^n \geq 0 \quad (34)$$

$$p = 1, \dots, N_{nz}$$

$$d_{ij}^x, d_{ij}^y, d_{ij}^z \geq 0 \quad (35)$$

$$i, j = 1, \dots, N_{cp} \quad i \neq j$$

- Eq. (36 e 37) – apresentam o domínio das variáveis binárias, relacionadas à rotação dos componentes e à posição relativa entre os componentes, respectivamente. Quando a variável assume valor 0 ela não existe e quando assume valor 1, existe.

$$w_{il} \in \{0, 1\} \quad (36)$$

$$i = 1, \dots, N_{cp} \quad l = 1, \dots, 8$$

$$w_{ij}^x, w_{ij}^y, w_{ij}^z \in \{0, 1\} \quad (37)$$

$$i, j = 1, \dots, N_{cp} \quad i \neq j$$

onde:

$$M_{ij} = \frac{1}{2} [\max \{A_i, B_{ij}\} + \max \{A_j, B_{ij}\}] \quad (38)$$

Neste modelo, os bocais sobre os componentes têm dois índices, um local para cada componente i ($j = 1, \dots, N_{z_i}$) e um outro global para todos os componentes

$$\left(p = j + \sum_{l=1}^{i-1} N_{z_l} \right).$$

A partir deste modelo foram desenvolvidos os modelos apresentados nos itens 3.6. e 3.7., objetivando-se a redução do tempo computacional.

A implementação deste modelo para a unidade petroquímica, em GAMS, encontra-se no Anexo 1.

3. 6. Modelo Modificado (com restrições lógicas)

Este modelo consiste no acréscimo de restrições lógicas (Eq. 39, 40 e 41) ao modelo original, proposto por GUIRARDELLO (1993), apresentado no item 3.5. Tais restrições estão relacionadas às posições relativas entre os componentes (no plano tridimensional) e evitam redundâncias, restringindo mais fortemente as relaxações do modelo.

$$w_{ij}^x \geq w_{ik}^x + w_{kj}^x - 1 \quad (39)$$

$$i \neq j \neq k$$

$$w_{ij}^y \geq w_{ik}^y + w_{kj}^y - 1 \quad (40)$$

$$i \neq j \neq k$$

$$w_{ij}^z \geq w_{ik}^z + w_{kj}^z - 1 \quad (41)$$

$$i \neq j \neq k$$

* Obs.: Os índices i, j e k referem-se aos componentes da planta.

Estas restrições diminuem consideravelmente o número de iterações e nós inspecionados através do algoritmo *Branch & Bound*, pelo *solver CPLEX 6.0* (disponível pelo *GAMS*) e, portanto, o tempo computacional gasto na resolução. Entretanto, elas não alteram o conjunto de soluções inteiras possíveis. Tais restrições lógicas, expressas através das Eq. (39), (40) e (41), implicam na seguinte situação:

se A está à direita de B e

se B está à direita de C ;

então A está à direita de C ,

onde A , B e C são os componentes da planta, neste exemplo. O mesmo vale para outras posições relativas entre os componentes.

3. 7. Modelo Modificado (substituição de restrições + restrições lógicas)

Este modelo é composto pelo modelo original (item 3.5), substituindo-se as restrições referentes às posições dos bocais p (x_p^a , y_p^a , z_p^a), expressas pelas Eq. 22 a 27 pelas novas restrições relacionadas ao comprimento dos tubos k , usando a distância de *Manhattan* nas direções x , y e z (l_k^{x+} , l_k^{x-} , l_k^{y+} , l_k^{y-} , l_k^{z+} , l_k^{z-}), conforme são expressas pelas Eq.(42), (43) e (44), além de incluir as restrições lógicas apresentadas no item 3.6. A implementação em *GAMS* deste modelo para a unidade petroquímica está no Anexo 1.

$$l_k^{x+} - l_k^{x-} = x_i - x_j + F_P^x \cdot \frac{1}{2} A_i \cdot (w_{i1} - w_{i3} + w_{i5} - w_{i7}) + F_P^y \cdot \frac{1}{2} B_i \cdot (-w_{i2} + w_{i4} - w_{i6} + w_{i8}) \\ - F_Q^x \cdot \frac{1}{2} A_j \cdot (w_{j1} - w_{j3} + w_{j5} - w_{j7}) - F_Q^y \cdot \frac{1}{2} B_j \cdot (-w_{j2} + w_{j4} - w_{j6} + w_{j8}) \quad (42)$$

$$l_k^{y+} - l_k^{y-} = y_i - y_j + F_P^y \cdot \frac{1}{2} B_i \cdot (w_{i1} - w_{i3} - w_{i5} + w_{i7}) + F_P^x \cdot \frac{1}{2} A_i \cdot (w_{i2} - w_{i4} - w_{i6} + w_{i8}) \\ - F_Q^y \cdot \frac{1}{2} B_j \cdot (w_{j1} - w_{j3} - w_{j5} + w_{j7}) - F_Q^x \cdot \frac{1}{2} A_j \cdot (w_{j2} - w_{j4} - w_{j6} + w_{j8}) \quad (43)$$

$$l_k^{z+} - l_k^{z-} = z_i - z_j + F_P^z \cdot \frac{1}{2} C_i - F_Q^z \cdot \frac{1}{2} C_j \quad (44)$$

onde: $k = 1, \dots, N_{pi}$ $i, j = 1, \dots, N_{cp}$ $p = k_s(k)$ $q = k_t(k)$

* Obs.: O índice i refere-se ao componente onde se encontra o bocal p e o índice j , ao componente onde se encontra o bocal q .

As Eq. (42), (43) e (44) foram formuladas através do arranjo algébrico das variáveis envolvidas, no intuito de reduzir o número total de variáveis empregadas no modelo, reduzindo assim, o número de equações gerado pelo programa em *GAMS* e, conseqüentemente, reduzindo o tempo computacional gasto na resolução do problema. As variáveis eliminadas foram aquelas relacionadas ao posicionamento dos bocais no plano tridimensional (x_p^n, y_p^n, z_p^n) .

CAPÍTULO 4

MÉTODOS E ALGORITMOS DE RESOLUÇÃO

Neste trabalho, os problemas formulados em MILP são resolvidos pelo algoritmo *Branch & Bound*, através do *solver CPLEX 6.0*, que é utilizado pelo programa desenvolvido em *GAMS* (“General Algebraic Modeling System”).

4. 1. Algoritmo *Branch and Bound* (B&B)

Os problemas em MILP podem ser resolvidos por programação linear e, mais comumente, são resolvidos utilizando-se o algoritmo *Branch and Bound* (B&B), que pode ser usado para achar uma solução ótima para problemas relativamente pequenos.

Resumidamente, este método baseia-se nos limites superiores e inferiores dos valores da função objetivo. Ele divide o problema em subconjuntos, computando os seus limites e, em seguida, a cada iteração, compara os valores obtidos anteriormente com os valores atuais, escolhendo o melhor limite. O processo de iterações continua até que a solução ótima seja encontrada.

O algoritmo B&B pode ser usado tanto para resolver problemas de programação inteira pura (IP), linear (LP) ou mista, inteira e linear (MILP).

Todo algoritmo B&B tem as mesmas três etapas básicas (HILLIER & LIEBERMAN, 1990; MURTY, 1995): “branching” (ramificação), “bounding” (definição de limites) e “fathoming” (exame detalhado para achar a solução ótima). A flexibilidade desta técnica está em como estas etapas são executadas.

Branching sempre envolve a seleção de um subproblema remanescente e a sua divisão em subproblemas menores, isto é, como seu próprio nome diz, há uma ramificação do problema. Nesta etapa, a flexibilidade está ligada à maneira como é feita a seleção e a divisão.

Bounding é usualmente empregada para resolver uma relaxação, sendo que existem diversos caminhos para formar relaxações. Na maioria dos casos, a relaxação é obtida, simplesmente, excluindo-se um conjunto de restrições que dificulta a resolução do problema. Em termos gerais, as duas características procuradas ao escolher uma relaxação são para resolver o problema de forma relativamente rápida e para obter um limite relativamente justo. No entanto, são necessárias ambas as características, pois uma somente não é adequada. Esta etapa considera e usa os limites superior e inferior para achar o valor objetivo ótimo.

Fathoming é a etapa de análise do problema. Um subproblema é considerado *fathomed* se a análise de sua relaxação revela que:

1. Soluções viáveis (“feasible”) do subproblema devem ter $Z \leq Z^*$ ou
2. O subproblema não tem solução viável ou
3. Uma solução ótima do subproblema foi encontrada.

onde Z^* é o valor mais atualizado, isto é, a melhor solução viável encontrada até o momento em relação a função objetivo Z .

O procedimento B&B é semelhante (HILLIER & LIEBERMAN, 1990) a outros problemas de programação, ao ser aplicado ao problema geral de MILP, onde algumas das variáveis são restritas a valores inteiros (mas não necessariamente apenas a 0 e 1), mas as demais são variáveis ordinárias contínuas. A forma geral do problema MILP já foi apresentada através das expressões 1 a 4.

Portanto, no caso de problemas MILP, as bases para as etapas de *bounding* e *fathoming* são obtidas resolvendo-se as relaxações de programação linear. De fato, apenas quatro mudanças são necessárias neste algoritmo em relação ao aplicado aos problemas inteiros puros ou binários.

Uma mudança envolve a escolha da variável de ramificação, que é a variável usada em qualquer iteração, onde lhe são atribuídos diferentes valores. Nas outras programações, escolhia-se automaticamente a variável mais próxima na ordem natural – $x_1, x_2, \dots, x_n$. Em se tratando de MILP, somente serão consideradas as variáveis “inteiras restritas”, que não têm um valor inteiro na solução ótima para a relaxação do subproblema corrente. Uma sugestão seria de escolher a primeira variável na ordem natural.

A segunda mudança envolve os valores atribuídos à variável de ramificação para criar os novos subproblemas menores. Para os problemas binários, a variável era fixada em 0 ou 1, respectivamente, para os dois subproblemas novos. Agora, a variável geral “inteira restrita” poderia ter um número maior de valores inteiros possíveis e seria ineficiente criar e analisar muitos subproblemas. Portanto, novamente são criados apenas dois subproblemas novos, mas especificando-se as faixas de valores para a variável.

Para explicar como isso é feito, considere que x_j é a variável de ramificação atual e x_j^* é seu valor (não-inteiro) na solução ótima para a relaxação de programação linear do problema. Assim, admitindo que:

$$[x_j^*] = \text{maior inteiro menor ou igual a } x_j^*,$$

a faixa de valores para os dois subproblemas novos é

$$x_j \leq [x_j^*] \quad \text{e} \quad x_j \geq [x_j^*] + 1, \quad (45)$$

respectivamente. Cada desigualdade torna-se uma restrição adicional para o subproblema novo. Por exemplo, se $x_j^* = 3 \frac{1}{2}$, então

$$x_j \leq 3 \quad \text{e} \quad x_j \geq 4, \quad (46)$$

são as respectivas restrições adicionais para o subproblema novo.

Quando as duas mudanças citadas estão combinadas, é possível que ocorra um fenômeno de repetição da variável de ramificação. Para ilustrar, façamos $j = 1$ no exemplo acima onde $x_j^* = 3 \frac{1}{2}$, e considere o novo subproblema onde $x_1 \leq 3$. Quando a relaxação de programação linear (relaxação-LP) for resolvida, suponha que $x_1^* = 1 \frac{1}{4}$. Então x_1 se repete

como variável de ramificação, e os dois subproblemas novos criados têm restrições adicionais, $x_1 \leq 1$ e $x_1 \geq 2$, respectivamente. Depois, quando o subproblema $x_1 \leq 1$ for resolvido, suponha que $x_1^* = \frac{3}{4}$. Então x_1 se repete novamente como variável de ramificação, e os dois subproblemas novos criados têm $x_1 = 0$ (por causa da restrição nova $x_1 \leq 0$ e a restrição de não-negatividade em x_1) e $x_1 = 1$ (por causa da restrição nova $x_1 \geq 1$ e a restrição prévia $x_1 \leq 1$).

A terceira mudança envolve a etapa de limites. Nas outras programações costumava-se arredondar (para menos) o valor de Z ; agora o limite é o valor de Z sem arredondamentos.

A última mudança envolve o terceiro teste da etapa de *fathoming*. Ao invés de considerar apenas uma solução ótima viável quando esta fosse inteira; agora o teste exige que somente as variáveis “inteiras restritas” sejam inteiras na solução ótima para o subproblema de relaxação-LP, pois isto basta para certificar-se que a solução é viável e, portanto, ótima, para o subproblema.

Em suma, o algoritmo B&B para problemas em MIP pode ser descrito como (HILLIER & LIEBERMAN, 1990):

Passo de inicialização: Fazer $Z^* = -\infty$. Aplicar as etapas: de limite, *fathoming* e teste da solução ótima que são descritas abaixo. Se não estiver *fathomed*, classificar este problema como um subproblema remanescente para realizar a primeira iteração completa, abaixo.

Passos para cada iteração:

1. *Branching*: Entre os subproblemas remanescentes (que não estão completamente analisados e, portanto, *fathomed*), selecionar aquele criado mais recentemente. Entre as variáveis “inteiras restritas” que não têm um valor inteiro na solução ótima para a relaxação-LP do subproblema, escolher a primeira na ordem natural das variáveis para ser a variável de ramificação. Fazer com que x_j seja esta variável e x_j^* , seu valor nesta solução. Ramificar a partir do nó para o subproblema criar dois novos subproblemas pela adição das respectivas restrições, $x_j \leq [x_j^*]$ e $x_j \geq [x_j^*] + 1$.
2. *Bounding*: Para cada subproblema novo, obter seus limites através da aplicação do método simplex para a relaxação-LP e usar o valor de Z para a solução ótima resultante.

3. *Fathoming* : Para cada subproblema novo, aplicar os três testes abaixo e descartar os subproblemas que estão *fathomed* por qualquer um dos testes.

Teste 1. Seu limite $\leq Z^*$, onde Z^* é o valor de Z mais atualizado.

Teste 2. A relaxação-LP não apresenta soluções viáveis.

Teste 3. A solução ótima para a relaxação-LP tem valores inteiros para as variáveis "inteiras restritas". (Se esta solução é melhor do que o valor mais atualizado, ela torna-se o novo valor atual e o teste 1 é reaplicado para todos os subproblemas que ainda não estão *fathomed* com o novo e maior Z^* .)

Teste do ótimo: Parar quando não houver subproblemas remanescentes, isto é, quando o valor mais atualizado for ótimo. Caso contrário, realizar novamente outra iteração.

4. 2. As etapas do algoritmo B&B aplicado aos problemas MILP

Uma maneira simplificada sobre o procedimento *Branch and Bound* é descrita por EDGAR & HIMMELBLAU (1989):

1. Obter a solução contínua (não inteira) do problema; designar este problema como "C".
2. A solução para C satisfaz todas as exigências inteiras? Em caso afirmativo, está terminado. Se não, prosseguir para a etapa 3.
3. A partir de uma revisão da solução para o problema C selecione uma variável inteira que tenha um valor não inteiro (α). Considere o valor de x_a , α , a ser um nóculo de partida. Construa dois nós que se ramifiquem de α pela adição das restrições:
 - a) $x_a \geq$ o menor inteiro maior que α

- b) $x_a \leq$ o maior inteiro menor que α

Chame o problema C com os problemas de restrição adicionados C^1 e C^2 . Em seguida resolva o problema contínuo para C^1 ou C^2 , digamos C^1 . Retornamos ao outro ramo mais tarde.

4. Revise a solução para C^1 , e ramifique sobre uma nova variável x_b usando duas restrições de partição como antes. Selecione o ramo que inicialmente parece ser o mais promissor para o próximo subproblema a resolver. Poderia ser necessário ramificar sobre x_a novamente se mais tarde as soluções requererem. Continue ramificando sobre as variáveis inteiras até que uma das três coisas aconteça:
 - a) Você alcança uma solução que satisfaça todas as restrições inteiras (isto poderia exigir a fixação de todas as variáveis em valores inteiros).
 - b) A função objetivo para uma solução de um subproblema torna-se inferior a qualquer solução previamente obtida que satisfaça as restrições inteiras.
 - c) Não é possível encontrar nenhuma solução viável.

Em qualquer caso, este nó terminal é considerado compreendido.

5. Em seguida, retornar a um nó anterior sobre o mesmo ramo, usando a restrição alternativa e resolvendo o subproblema de otimização. Calcular as soluções do subproblema como nós adicionais como no passo 4. Uma vez que um ramo está compreendido, repetir a operação até que finalmente alcance a solução original (nó 1). “Reserve” o melhor valor para o ramo apenas analisado para comparação com os outros ramos.

Continuar deste jeito até que todas as possíveis combinações de restrições inteiras tenham sido realmente impostas sobre o problema ou compreendidas. Agora, decida em qual limite irá usar para mudar em um nó, e então determine qual a sequência de nós a seguir na otimização, é o “problema-dependente”.

4. 3. GAMS – “General Algebraic Modeling System”

Entre 1950 e 1960, com o desenvolvimento de algoritmos e códigos computacionais, houve progressos substanciais na resolução de problemas matemáticos extensos.

No Centro de Pesquisa e Desenvolvimento do Banco Mundial (BROOKE et al., 1988), em Washington, surgiu o sistema *GAMS* (*General Algebraic Modeling System*), que foi desenvolvido a fim de que os modelos pudessem ser resolvidos em diferentes tipos de computadores sem nenhuma variação.

A estrutura de entrada ("input") do modelo GAMS consiste basicamente nos componentes citados abaixo (BROOKE et al., 1988), onde os comandos específicos do GAMS são apresentados entre parêntesis:

- Índices (*SETS*)
 - declaração
 - atribuição de elementos

- Dados (*PARAMETERS, TABLES, SCALARS*)
 - declaração
 - atribuição de valores

- Variáveis (*VARIABLES*)
 - declaração
 - definição do tipo
 - (opcional - atribuição de valores iniciais e de contorno)

- Equações (*EQUATIONS*)
 - declaração
 - definição

- Modelo e resolução do problema (*MODEL e SOLVE*)

- Demonstração (*DISPLAY*) - opcional

A descrição mais detalhada destes comandos encontra-se no Anexo 4.

CAPÍTULO 5

CASOS DE ESTUDO

O modelo descrito no item 3.5 foi implementado em GAMS inicialmente para o caso de estudo de uma unidade petroquímica (LEESLEY, 1982; GUIRARDELLO, 1993) e, em seguida, para a planta de polimerização (GUNN, 1970) e para a unidade de recuperação de mercúrio a partir de efluentes eletrolíticos (AMORESE et al., 1977). As alterações no modelo, descritas nos itens 3.6 e 3.7, também foram testadas para estes três casos de estudo.

5.1. Unidade petroquímica

O primeiro caso de estudo testado foi de uma unidade petroquímica, que é apresentada na Figura 5, cuja planta consiste de 9 componentes principais (Tabela 1). Porém, para facilitar a modelagem, bem como a programação em *GAMS*, os componentes desta unidade foram rearranjados, conforme pode-se observar na Tabela 2.

É interessante ressaltar que, neste trabalho, o agrupamento dos componentes em pequenos módulos é realizado considerando-se o fluxograma de processo, a similaridade das funções dos equipamentos e o bom senso. No entanto, para o caso específico da unidade petroquímica e para efeito de comparação com o trabalho desenvolvido por GUIRARDELLO (1993), o agrupamento dos componentes de tal planta foi o mesmo realizado pelo referido autor.

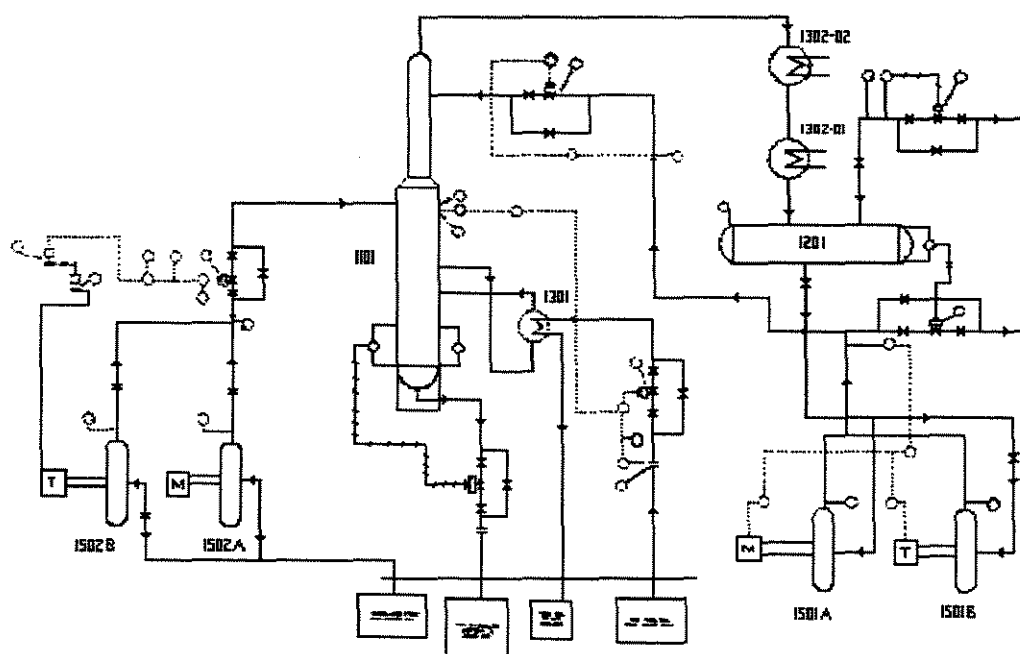


Figura 5 . Unidade petroquímica com nove componentes principais.

Fonte: GUIRARDELLO, 1993.

Tabela 1. Os nove componentes principais da unidade petroquímica.

Componentes	Nomes
1	Coluna de destilação (1101)
2	tanque de refluxo (1201)
3	refervedor (1301)
4	condensador (1302-01)
5	condensador (1302-02)
6	bomba de refluxo (1501 A)
7	bomba de refluxo (1501B)
8	bomba do produto superaquecido (1502 A)
9	bomba do produto superaquecido (1502B)

Tabela 2. Os componentes rearranjados para a unidade petroquímica.

Componentes	Nomes
1	Coluna de destilação (1101)
2	tanque de refluxo (1201)
3	refervedor (1301)
4	condensadores (1302-01 e 1302-02)
5	bombas de refluxo (1501 A e 1501B)
6	bombas do produto superaquecido (1502 A e 1502B)

5. 2. Planta de polimerização

A planta de polimerização apresentada na Fig. 6 é composta por 13 componentes principais (Tab. 3), que foram rearranjados, conforme Tab. 4.

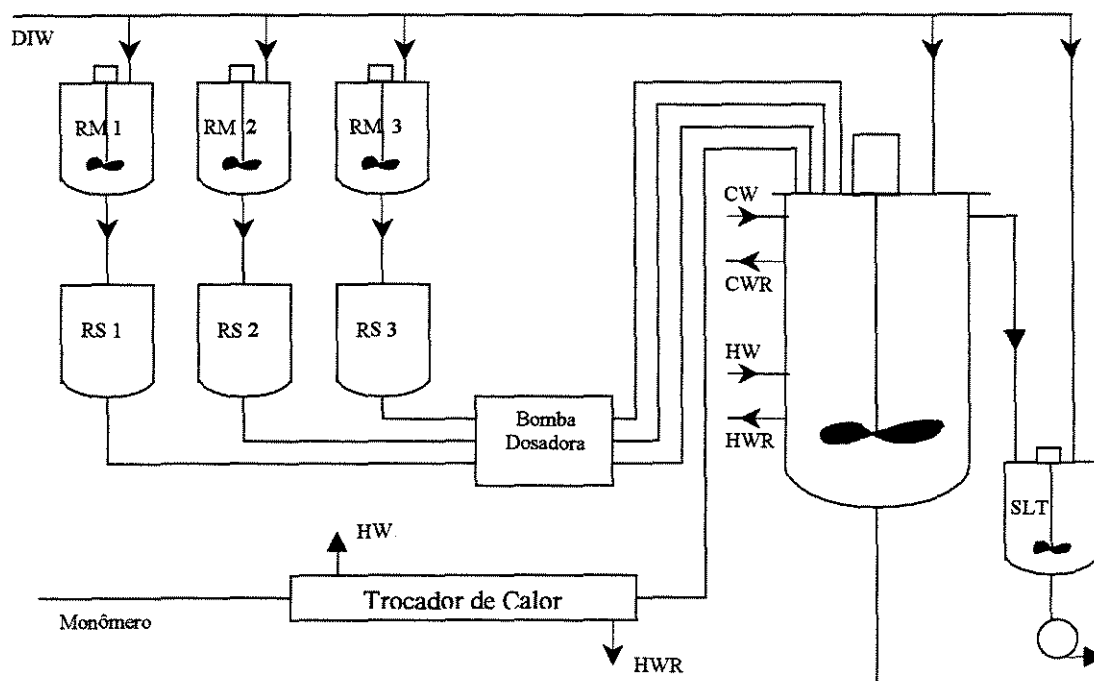


Figura 6. Planta de polimerização com treze componentes principais.

Fonte: GUNN, 1970.

Tabela 3. Os treze componentes principais da planta de polimerização.

Componentes	Nomes
1	Tanque de mistura (RM1)
2	Tanque de mistura (RM2)
3	Tanque de mistura (RM3)
4	Tanque de armazenagem (RS1)
5	Tanque de armazenagem (RS2)
6	Tanque de armazenagem (RS3)
7	Bomba dosadora
8	Bomba dosadora
9	Bomba dosadora
10	Reator
11	Tanque após reator (SLT)
12	Bomba após SLT
13	Trocador de calor horizontal

Tabela 4. Os componentes rearranjados para a planta de polimerização.

Componentes	Nomes
1	Tanques de mistura (RM 1, 2 e 3)
2	Tanques de armazenagem (RS 1, 2 e 3)
3	Bombas dosadoras
4	Reator
5	Tanque após reator (SLT)
6	Bomba após SLT
7	Trocador de calor horizontal

5. 3. Planta de recuperação de mercúrio

A planta de recuperação de mercúrio a partir de efluentes eletrolíticos (AMORESE, CENA E MUSTACCHI, 1977), apresentada na Figura 7, contém 9 componentes principais que não puderam ser rearranjados como nos outros casos de estudo, aumentando, assim, a complexidade computacional para a resolução do problema modelado em MILP.

Os nove componentes principais desta planta são listados na Tabela 5, embora não houvesse uma descrição dos componentes da mesma no artigo de onde foi retirada.

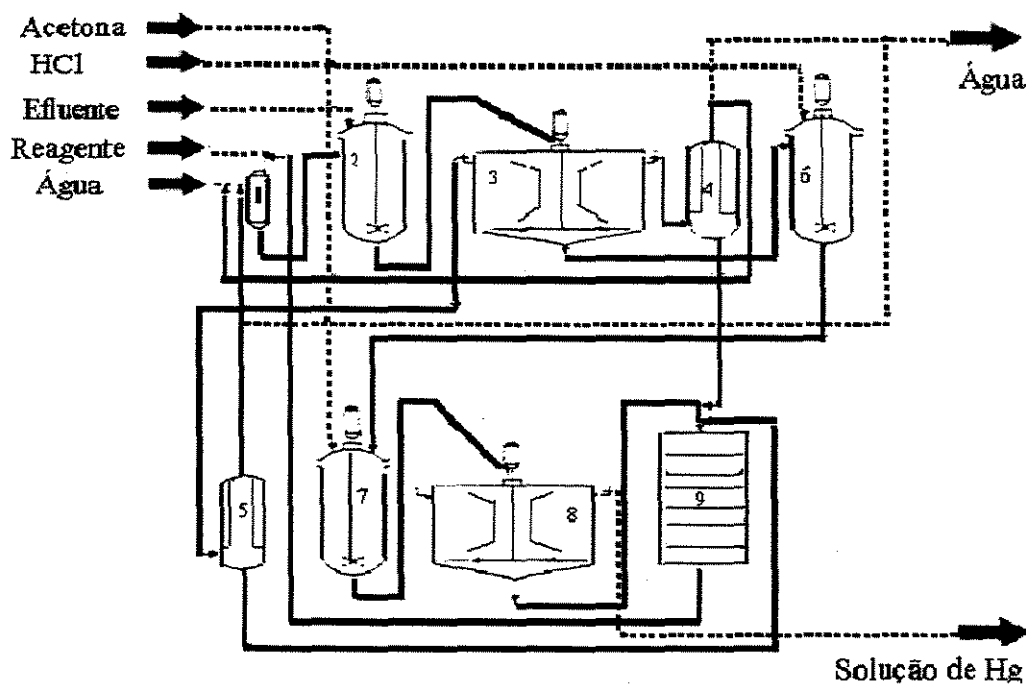


Figura 7. Planta de recuperação de mercúrio a partir de efluentes eletrolíticos.

Fonte: AMORESE, CENA E MUSTACCHI, 1977.

Tabela 5. Os nove componentes principais da planta de recuperação de mercúrio.

Componentes	Nomes
1	Misturador
2	Reator
3	Decantador
4	Filtro
5	Filtro
6	Reator
7	Reator
8	Decantador
9	Filtro

5. 4. Dados para os casos de estudo

O levantamento de dados para os três casos de estudo foi realizado a partir de informações disponíveis na literatura, como também a partir dos próprios fluxogramas de processo dos casos estudados. Alguns dados não foram encontrados, principalmente para o terceiro caso de estudo (planta de recuperação de mercúrio), logo, precisaram ser estimados através de informações de outros processos semelhantes, ou mesmo, fixados arbitrariamente.

Os dados pesquisados e/ou estimados referem-se às dimensões de equipamentos, distâncias mínimas de segurança, número e posicionamento dos tubos e bocais, bem como os custos associados aos suportes e componentes, que são necessários para o desenvolvimento e implementação dos modelos em *GAMS*.

5. 4. 1. Unidade petroquímica

Os dados usados para a implementação do caso de estudo da unidade petroquímica foram os mesmos usados por GUIRARDELLO (1993), para poder comparar os resultados a partir do modelo original.

Os dados referem-se às dimensões dos componentes, distâncias mínimas de segurança que devem existir entre os componentes e dados sobre os bocais e tubos empregados na unidade petroquímica. Estes dados são apresentados através das Tab. 34 a 37, no Anexo 1.

Quanto ao termo de elevação mínima vertical do componente i , E_i^l , ele é zero para todos os componentes neste caso; o custo por unidade de comprimento do perímetro, $C^{per} = US\$298.43/m$ e o número grande foi considerado como $M = 51,0$.

Ainda, para obter os fatores de custo unitário para o suporte do componente i , C_{ik}^{sa} e C_{ik}^{sb} , foram consideradas as correlações (Eq. 47 a 54) obtidas em GUIRARDELLO (1993):

$$C_{i1}^{sa} = \text{Área}_i * 62,8765 \quad (47)$$

$$C_{i2}^{sa} = \text{Área}_i * 95,3989 \quad (48)$$

$$C_{i3}^{sa} = \text{Área}_i * 115,9964 \quad (49)$$

$$C_{i4}^{sa} = \text{Área}_i * 140,9302 \quad (50)$$

$$C_{i1}^{sb} = \text{Área}_i * 0,0 \quad (51)$$

$$C_{i2}^{sb} = \text{Área}_i * (-99,1281) \quad (52)$$

$$C_{i3}^{sb} = \text{Área}_i * (-224,6904) \quad (53)$$

$$C_{i4}^{sb} = \text{Área}_i * (-528,6832) \quad (54)$$

onde: $i = 1, \dots, N_{cp}$

5. 4. 2. Planta de polimerização

Os dados para a planta de polimerização, utilizados para a implementação do modelo em MILP para a resolução em *GAMS*, são apresentados através das Tab. 33 a 37, no Anexo 2.

O termo de elevação mínima vertical do componente i , E_i^l , é considerado neste caso-estudo para o trocador de calor e os tanques de mistura (RM) como $E_i^l = 3,5$ m, para os demais componentes é considerado nulo.

Ainda, as correlações usadas para obter os fatores de custo unitário para o suporte do componente i , C_{ik}^{sa} e C_{ik}^{sb} , foram atualizadas em função do índice de *Marshall & Swift Equipment (Chemical Engineering, sep. 2000)* referente ao 2º. trimestre de 2000 ($I = 1089$), conforme Eq. 55 e 56, resultando nas Eq. 57 a 64.

$$C_p = C_o * (I_p / I_o) \quad (55)$$

onde : C_p = custo atual

C_o = custo original

I_p = índice atual

I_o = índice original

As correlações originais para a obtenção dos fatores de custo foram apresentadas por Guthrie (1974), mas a base de dados utilizada para os custos foi o 1º. Trimestre de 1970, cujo índice de *Marshall & Swift Equipment* era $I = 301$.

Logo,

$$C_p = C_o * (1089/301) \quad (56)$$

Desta forma, as correlações atualizadas são expressas através das Eq.57-64.

$$C_{i1}^{sa} = \acute{Area}_i * 74,1045 \quad (57)$$

$$C_{i2}^{sa} = \acute{Area}_i * 112,4344 \quad (58)$$

$$C_{i3}^{sa} = \acute{Area}_i * 132,7100 \quad (59)$$

$$C_{i4}^{sa} = \acute{Area}_i * 166,0963 \quad (60)$$

$$C_{i1}^{sb} = \acute{Area}_i * 0,000 \quad (61)$$

$$C_{i2}^{sb} = \acute{Area}_i * (-116,8295) \quad (62)$$

$$C_{i3}^{sb} = \acute{Area}_i * (-264,8136) \quad (63)$$

$$C_{i4}^{sb} = \acute{Area}_i * (-623,0909) \quad (64)$$

onde: $i = 1, \dots, N_{cp}$

E o custo por unidade de comprimento do perímetro, C^{per} , também foi atualizado pelas Eq. 55 e 56, resultando em $C^{per} = \text{US\$}351.72/\text{m}$.

5. 4. 3. Planta de recuperação de mercúrio

Os dados, apresentados através das Tab. 43 a 45, no Anexo 3, foram obtidos a partir do fluxograma de processo e de algumas informações descritas sobre a planta de recuperação de mercúrio de efluentes eletrolíticos apresentada por AMORESE, CENA E MUSTACCHI (1977). A Tab. 46 (Anexo 3) apresenta as distâncias mínimas de segurança entre os componentes.

As correlações para a obtenção dos fatores de custo são as mesmas apresentadas através das Eq.55 a 64.

As informações sobre os custos dos tubos, C_k^p , são valores arbitrários associados àqueles descritos no artigo de AMORESE, CENA E MUSTACCHI (1977), multiplicando-se por um fator de 10 (este fator foi escolhido arbitrariamente, também).

CAPÍTULO 6

RESULTADOS E DISCUSSÃO

A unidade petroquímica, a planta de polimerização e a unidade de recuperação de mercúrio foram implementados em *GAMS 2.50*, utilizando-se o *solver CPLEX 6.0*, em uma estação de trabalho “SunSparc 5 / Solaris”, com 98 Mb de memória (RAM). Inicialmente, avaliou-se o modelo de otimização proposto por GUIARDELLO (1993) e, posteriormente, foram desenvolvidos e testados outros modelos a partir do modelo inicial.

O tempo computacional de execução (tempo de *CPU*) foi avaliado, devido ao objetivo de se obter um modelo que pudesse resolver o problema de otimização através da programação mista, inteira e linear, de forma exata, em um tempo aceitável.

A complexidade de tempo para a resolução destes problemas varia exponencialmente e isto foi constatado através da adição de componentes às plantas, ou seja, a unidade petroquímica contém 6 componentes e a unidade de recuperação de mercúrio, 9 componentes, sendo que o tempo de resolução aumentou drasticamente de uma planta para outra. Logo, algumas variáveis de posição relativa entre os componentes foram fixadas para reduzir este tempo computacional, reduzindo, também, as iterações e nós inspecionados pelo algoritmo *Branch & Bound* (B&B).

A vantagem de se utilizar o pacote comercial (*GAMS*) é a possibilidade de realizar certos ajustes, que incluem desde a maior disponibilização de memória; acréscimos nos

limites de tempo de execução, de nós a serem usados, de iterações; como também mudanças relativas ao algoritmo *Branch & Bound*.

As opções testadas dentro do *solver CPLEX 6.0* referem-se à escolha dos algoritmos “depth first” e “best bound”, que são estratégias de pesquisa do algoritmo *Branch & Bound* e, também, alguns testes em relação à identificação das variáveis binárias, também chamadas *SOS* (“special ordered set”) para facilitar o processo de iterações. No entanto, esta última opção não apresentou grande melhora nos resultados para a unidade petroquímica e, portanto, foi descartada para os demais casos de estudo, de tal forma a se fazer comparações em uma mesma base.

6.1. Unidade petroquímica

As Tabelas 6, 7 e 8 apresentam alguns resultados relacionados ao tempo de execução, nós e iterações para a unidade petroquímica, onde foram realizados os testes com o modelo inicial e com os modelos modificados com as restrições lógicas, e com as substituições de restrições + restrições lógicas, respectivamente.

Tabela 6. Resultados obtidos para a unidade petroquímica, com o modelo existente na literatura.

	layout1/modelo(1) + “depth first”	layout1/modelo(1) + “depth first” + “SOS”	layout1/modelo(1) + “best bound”	layout1/modelo(1) + “best bound” + “SOS”
Tempo	31.917s	31.917s	33.621s	33.105s
Iterações MIP	1.368.147	1.368.147	1.409.567	1.409.556
Nós	273.675	273.675	271.445	271.402

* Utilizando a opção para identificar as variáveis “SOS”, binárias.

Como pode-se observar na Tab.6, empregando-se o modelo existente na literatura (modelo 1), com a opção para identificar as variáveis binárias (“SOS”), houve uma pequena redução no tempo de execução para a estratégia “best bound”, enquanto que o tempo de execução para a estratégia “depth first” foi o mesmo. Portanto, essa opção (“SOS”) foi descartada para os testes com os outros modelos.

Na Tab. 7, pode-se verificar que o novo modelo (modelo 2), com as restrições lógicas, apresentou considerável redução no tempo de execução (86,3% e 87,01%, para “depth first” e “best bound”, respectivamente) em relação ao modelo original proposto por GUIRARDELLO (1993).

Tabela 7. Resultados obtidos para a unidade petroquímica, com o modelo modificado com as restrições lógicas (modelo 2).

	layout1 / modelo (2) “depth first”*	layout1 / modelo (2) “best bound”
Tempo	4.368s	4.300s
Redução de Tempo*	86,3%	87,01%
Iterações (MIP)	177.828	176.019
Nós	23.246	22.319

* Redução percentual de tempo obtida em relação ao modelo original (1).

A Tab. 8 apresenta os resultados obtidos para o último modelo proposto (modelo 3), que substitui as restrições de posição dos bocais pelas de comprimento dos tubos e inclui as restrições lógicas. Nesta tabela, observa-se que, com a utilização do modelo 3, obteve-se os melhores resultados quanto à redução de tempo de execução do problema para a unidade petroquímica. Em relação ao modelo original, a redução de tempo foi de 94,40% e 94,88%, empregando-se as estratégias “depth first” e “best bound”, respectivamente.

Tabela 8. Resultados obtidos para a unidade petroquímica, com o modelo modificado pela substituição de restrições e adição das restrições lógicas (modelo 3).

	layout1 / modelo (3) “depth first”*	layout1 / modelo (3) “best bound”
Tempo	1.786s	1.720s
Redução de Tempo*	94,40%	94,88%
Iterações (MIP)	62.175	60.835
Nós	9.965	9.691

* Redução percentual de tempo obtida em relação ao modelo original (1).

Ainda, através dos resultados obtidos, não se pode dizer que uma ou outra estratégia de pesquisa do algoritmo *Branch & Bound* seja melhor, em termos de tempo computacional gasto na execução do problema. Às vezes, a estratégia “best bound” encontra a solução ótima com menos iterações e nós analisados do que a estratégia “depth first” (Tab. 7 e 8); outras vezes, o inverso também ocorre (Tab. 6).

O algoritmo “depth first” analisa “profundamente” a partir do nó aberto mais recentemente, expandindo a árvore de ramificações até alcançar uma solução factível ou apenas *fathomed* para partir para outra série de ramificações. Enquanto o algoritmo “best bound” ramifica-se a partir do primeiro nó, criando “galhos” diferentes e ramificações novas, ao mesmo tempo, expandindo a árvore lateralmente e consumindo mais memória para armazenamento e comparação dos valores. A sequência seguida para o processo de ramificações nestas estratégias é diferente e é isto que pode levar à análise de um número maior ou menor de nós, bem como de iterações.

O resultado obtido na função objetivo de custo (Eq.5), a partir do modelo implementado em *GAMS* para a unidade petroquímica foi o valor ótimo US\$27,857.181.

É interessante ressaltar que os dados usados neste problema (unidade petroquímica) foram exatamente iguais aos dados empregados por GUIARDELLO (1993), com o objetivo de comparar os resultados, avaliando a implementação do problema

através do pacote comercial de otimização *GAMS/CPLEX*. O valor obtido por GUIRARDELLO (1993) para a função objetivo foi US\$27,667.7, portanto, o valor obtido neste trabalho apresentou um pequeno desvio relativo de 0,68%, que pode ter sido ocasionado devido aos arredondamentos realizados pela estação de trabalho empregada.

Ainda, em relação à função objetivo, cada contribuição (custo de terreno, custo de suportes e custo de tubulação) foi avaliada, sendo que o custo com os suportes (acessórios para o suporte de tubos e componentes, entre outros) foi o mais significativo, isto é, aproximadamente 60% do valor ótimo obtido, aplicando-se o modelo original.

O custo do terreno considera apenas o plano bidimensional, onde o custo do perímetro foi adotado como $C^{per} = US\$298.43$ e as dimensões x e y foram obtidas pelo *GAMS*, 9,29m e 5,65m, respectivamente. Este custo é apresentado através da Eq. 65.

$$2*CPER*(X+Y) = US\$8,917.09 \quad (65)$$

O custo dos suportes, obtido para cada componente, é apresentado através da Tab.9, onde o custo total dos suportes é US\$16,647.67.

Tab. 9. Custo dos suportes para cada componente da unidade petroquímica (modelos 1, 2 e 3).

COMPONENTES	CS (custo dos suportes)
1	9,338.755
2	4,022.081
3	262.075
4	2,668.436
5	101.671
6	254.650
Custo Total	US\$16,647.67

O custo da tubulação varia conforme o modelo e a estratégia empregados no caso de estudo. Este custo refere-se ao termo $\sum_{k=1}^{N_{pi}} C_k^p [I_k^{x+} + I_k^{x-} + I_k^{y+} + I_k^{y-} + I_k^{z+} + I_k^{z-}]$ e, para ser calculado, foram considerados os valores de comprimento dos tubos, obtidos pelo *GAMS* (Tab. 10) e os valores de custo por unidade de comprimento. A Tab. 11 apresenta os custos por tubo e o custo total da tubulação empregada na unidade petroquímica.

As mesmas variáveis (binárias) de posição relativa entre os componentes que foram fixadas em GUIRARDELLO (1993), isto é, atribuído o valor 1, foram empregadas neste trabalho. As variáveis são: $w_{5,6}^x$, $w_{4,2}^z$, $w_{2,5}^z$, $w_{4,5}^z$.

Tabela 10. Comprimento dos tubos, em metros, obtidos pelo *GAMS*, para a unidade petroquímica.

TUBOS	LXP	LXN	LYP	LYN	LZP	LZN
1	2,40		1,72			12,862
2	2,40			0,560		12,862
3		7,175	0,256		11,90	
4	6,00			0,374		21,355
5	6,00		1,566			21,355
6		2,575		2,785	4,837	
7	5,00		2,785			3,167
8	0,150		2,785			3,167
9	0,470		3,794			2,100
10	0,705			0	4,715	
11	0,705			1,952	4,715	

Além disso, o tempo gasto por GUIARDELLO (1993) para execução do problema, implementado na época em FORTRAN 77, foi 51.513,1 segundos, aplicando-se a estratégia “depth-first” e 11.310,6 segundos, empregando-se a estratégia “best bound”, respectivamente, valores bem diferentes para cada estratégia e diferentes em relação à este trabalho. Os programas foram executados por GUIARDELLO em uma estação de trabalho também, porém, modelo *Digital DEC-station 3100*.

Tabela 11. Custos de tubulação para a unidade petroquímica.

TUBOS	CP(K)	ΣL^*	$\Sigma L^* CP(K)$
	(US\$/m)	(m)	(US\$)
1	5.249	16,982	89.139
2	5.249	15,822	83.050
3	26.247	19,331	507.381
4	5.249	27,729	145.550
5	5.249	28,921	151.806
6	39.042	10,197	398.111
7	39.042	10,952	427.588
8	39.042	6,102	238.234
9	9.843	6,364	62.641
10	14.764	5,42	80.021
11	14.764	7,372	108.804
Custo Total			US\$2,292.36

* ΣL representa $(LXP(K) + LXN(K) + LYP(K) + LYN(K) + LZP(K) + LZN(K))$

As melhores orientações dos componentes obtidas através do programa em *GAMS* variam conforme a estratégia de pesquisa do algoritmo *Branch & Bound* adotada. As Tab. 12 e 13 apresentam alguns resultados obtidos para a unidade petroquímica.

Tabela 12. Orientações dos componentes obtidas para a unidade petroquímica, conforme a estratégia “depth first”.

W_{ij} layout1 / modelo (1)	W_{ij} layout1 / modelo (2)	W_{ij} layout1 / modelo (3)
1,1	1,1	1,1
2,7	2,1	2,7
3,2	3,4	3,6
4,7	4,7	4,1
5,1	5,5	5,1
6,1	6,1	6,5

Tabela 13. Orientações dos componentes obtidas para a unidade petroquímica, conforme a estratégia “best bound”.

W_{ij} layout1 / modelo (1)	W_{ij} layout1 / modelo (2)	W_{ij} layout1 / modelo (3)
1,1	1,1	1,5
2,1	2,1	2,5
3,2	3,2	3,2
4,1	4,7	4,5
5,1	5,5	5,5
6,1	6,1	6,5

É interessante observar que as orientações mudaram tanto ao utilizar a mesma estratégia (“best bound”) em diferentes modelos, quanto ao usar estratégia diferente no mesmo modelo. Qualquer alteração, menor que seja, implicará em orientações diferentes dos componentes na planta. Para melhor visualização dos resultados obtidos pode-se consultar a Figura 1, onde foram apresentadas as orientações consideradas nos modelos.

A obtenção das coordenadas (x, y, z), para a construção dos blocos tridimensionais dos componentes, é função das seguintes condições:

* (1ª. condição) Para $w_{i,2}$, $w_{i,4}$, $w_{i,6}$, $w_{i,8}$ = 1 e $w_{i,1}$, $w_{i,3}$, $w_{i,5}$, $w_{i,7}$ = 0 ;

$$x = x_i \pm \frac{A_i}{2} \quad (66)$$

$$y = y_i \pm \frac{B_i}{2} \quad (67)$$

$$z = z_i \pm \frac{C_i}{2} \quad (68)$$

* (2ª. condição) Para $w_{i,2}$, $w_{i,4}$, $w_{i,6}$, $w_{i,8}$ = 0 e $w_{i,1}$, $w_{i,3}$, $w_{i,5}$, $w_{i,7}$ = 1 ;

$$x = x_i \pm \frac{B_i}{2} \quad (69)$$

$$y = y_i \pm \frac{A_i}{2} \quad (70)$$

$$z = z_i \pm \frac{C_i}{2} \quad (71)$$

Onde x_i , y_i e z_i são as posições do centro dos componentes e são obtidas através do programa desenvolvido em GAMS; enquanto que os parâmetros A_i , B_i e C_i , que se referem às dimensões dos componentes, provêm da literatura, conforme as tabelas apresentadas nos anexos 1, 2 e 3 deste trabalho.

Os valores 1 e 0 atribuídos às condições acima, implicam na existência ou não, respectivamente, de determinada rotação do componente.

Assim, as orientações obtidas através do programa são relacionadas às condições citadas para obter as coordenadas x_i , y_i , z_i de cada componente.

Para a visualização tridimensional dos *layouts* obtidos foram realizados os desenhos, onde são necessários somente os valores de x_i , y_i e z_i (posições centrais dos componentes) e os parâmetros A_i , B_i e C_i (dimensões dos componentes). Os valores referentes às variáveis de rotação dos componentes são necessários para a escolha adequada dos parâmetros A_i e B_i , determinados pelas condições expressas através das Eq. 66 a 71. Estes desenhos são apresentados através das Fig. 8 a 10 (Anexo 1) e os valores de x_i , y_i e z_i empregados são apresentados na Tab. 14.

Tabela 14 . Posições do centro dos componentes obtidas para a unidade petroquímica.

	layout1 / modelo (1)			layout1 / modelo (2)			layout1 / modelo (3)		
	“depth first”			“best bound”			“best bound”		
Componentes	x_i	y_i	z_i	x_i	y_i	z_i	x_i	y_i	z_i
1	0,940	1,160	11,250	0,940	1,160	11,250	0,940	4,490	11,250
2	8,585	2,780	5,655	8,585	2,780	5,655	8,585	2,870	5,655
3	3,515	4,885	0,385	3,515	4,885	0,385	3,515	0,765	0,385
4	8,115	2,825	9,530	8,115	2,825	9,530	8,115	2,825	9,530
5	8,585	1,615	0,235	8,585	1,615	0,235	8,585	4,035	0,235
6	5,180	1,500	0,375	5,180	1,500	0,375	5,180	4,150	0,375

Como pode-se observar na Tab. 14, os valores de y_i foram os únicos que mudaram no último modelo (modelo 3). Os demais valores foram iguais em todos os modelos, para a unidade petroquímica.

A Tab. 15 apresenta os resultados obtidos para as variáveis de posição relativa entre os componentes (w_{ij}^x , w_{ij}^y , w_{ij}^z), empregando-se a estratégia “best bound”. As posições em *itálico* são aquelas fixadas no início do problema, para reduzir o tempo computacional. Nesta tabela, nota-se uma pequena variação entre as posições relativas entre os

componentes da unidade petroquímica, para os diferentes modelos, demonstrando que é possível obter diferentes configurações do *layout* para uma mesma função objetivo. Ainda, essas variações também foram observadas ao empregar a estratégia “depth first”.

Tabela 15 . Posições relativas entre os componentes obtidas para a unidade petroquímica, conforme a estratégia “best bound”.

layout1 / modelo (1)			layout1 / modelo (2)			layout1 / modelo (3)		
w^x_{ij}	w^y_{ij}	w^z_{ij}	w^x_{ij}	w^y_{ij}	w^z_{ij}	w^x_{ij}	w^y_{ij}	w^z_{ij}
2,1	3,1	2,3	2,1	3,1	2,3	2,1	1,3	2,3
4,1	3,5	2,5	4,1	3,5	2,5	4,1	5,3	2,5
4,6	3,6	2,6	5,1	3,6	2,6	5,1	6,3	2,6
5,1		4,2	5,6		4,2	5,6		4,2
5,6		4,3	6,1		4,3	6,1		4,3
6,1		4,5			4,5			4,5
					4,6			4,6

Enfim, para comparação dos três modelos aplicados neste caso de estudo (unidade petroquímica) tem-se a Tab. 16, onde são apresentados os números de variáveis e equações empregados para a sua otimização. Os números referentes aos blocos de equações e blocos de variáveis correspondem àqueles empregados somente na implementação do modelo e os demais parâmetros correspondem aos números de equações e variáveis gerados pela execução do programa em *GAMS*, estando relacionados com o número de componentes da planta e com o próprio processo de iteração.

Ainda, na Tab. 16 observa-se que no modelo (2) empregou-se um número maior de equações, devido à inclusão das três restrições lógicas ao modelo original, isto é, dos 25 blocos de equações, passou para 28. Esta tabela também é útil para verificar a complexidade do problema.

Tabela 16. Comparação entre o número de equações e variáveis usadas em cada modelo, para a unidade petroquímica.

	layout1 / modelo (1)	layout1 / modelo (2)	layout1 / modelo (3)
Bloco de Equações	25	28	25
Bloco de Variáveis	24	24	21
Elementos não-zero	2.551	3.631	3.427
Equações Simples	418	778	700
Variáveis Simples	400	400	322
Variáveis Discretas	134	134	134

6.2. Planta de Polimerização

A planta de polimerização (GUNN, 1970) possui treze componentes principais, no entanto, eles foram rearranjados em pequenos grupos devido à sua similaridade, resultando em 7 componentes.

Observou-se que o acréscimo de um componente nesta planta em relação à planta petroquímica aumentou exageradamente o tempo de execução, quando implementada no modelo inicial em *GAMS*, sem fixar muitas posições relativas entre os componentes. Assim, não foi possível obter um resultado satisfatório para esta planta após 8 dias de execução na estação de trabalho, fixando-se apenas 4 variáveis (binárias) de posição relativa entre os componentes: $w_{4,5}^z$, $w_{5,6}^z$, $w_{2,3}^z$, $w_{1,2}^z$.

Em seguida, foram feitos diversos testes fixando-se novas variáveis de posição, considerando sempre o fluxograma de processo. Logo, foi escolhido um número razoável de variáveis a serem fixadas, para avaliar o desempenho nos três modelos propostos. Isto é válido, pois permite comparar os 3 modelos, que é o objetivo do trabalho. Assim, as variáveis de posição relativa entre os componentes que foram fixadas, isto é, atribuído o

valor 1, foram as seguintes: $w_{1,2}^z$, $w_{1,3}^z$, $w_{4,1}^x$, $w_{2,3}^z$, $w_{4,2}^x$, $w_{4,3}^x$, $w_{5,6}^z$; fixadas conforme o fluxograma de processo.

O resultado obtido na função objetivo de custo (Eq.5), a partir do modelo implementado em *GAMS* para a planta de polimerização foi o valor ótimo US\$22,624.71. Este valor não deve ser considerado como um valor real, uma vez que diversos dados foram retirados da literatura ou estimados arbitrariamente. O valor é interessante apenas como um parâmetro de comparação, para avaliar o desempenho dos modelos, aplicados para o mesmo caso de estudo.

As Tabelas 17, 18 e 19 apresentam alguns resultados relacionados ao tempo de execução, nós e iterações para a planta de polimerização, onde foram feitos testes com o modelo original (inicial) e com os modelos modificados com as restrições lógicas, e com as substituições de restrições + restrições lógicas, respectivamente.

Tabela 17. Resultados obtidos para a planta de polimerização, com o modelo inicial.

	layout2 / modelo (1) “depth first”	layout2 / modelo (1) “best bound”
Tempo	352s	321s
Iterações MIP	10.120	8.993
Nós	2.696	2.370

Tabela 18. Resultados obtidos para a planta de polimerização, com o modelo modificado pela adição das restrições lógicas.

	layout2 / modelo (2) “depth first”	layout2 / modelo (2) “best bound”
Tempo	1.881s	1.839s
Iterações MIP	50.328	49.272
Nós	7.825	7.609

Tabela 19. Resultados obtidos para a planta de polimerização, com o modelo modificado pela substituição de restrições e pela adição das restrições lógicas.

	layout2 / modelo (3) “depth first”	layout2 / modelo (3) “best bound”
Tempo	1.468s	1.506s
Iterações MIP	41.963	42.594
Nós	6.266	6.108

A Tab. 20 apresenta a contribuição de cada custo (terreno, suportes e tubulação) relacionada ao valor ótimo obtido para a função objetivo.

Tabela 20. Custos de terreno, suportes e tubulação para a planta de polimerização.

Custos:	Layout2
Terreno	11,044.01
Suportes	10,778.82
Tubulação	801.85
Total	US\$22,624.68

Neste caso de estudo, os custos de terreno e suportes foram mais representativos do que o custo de tubulação, relacionando-os com o valor ótimo obtido. Ressalta-se que os valores obtidos são arbitrários, não necessariamente condizentes com a realidade.

As Tab.21 e 22 apresentam as rotações obtidas para os componentes da planta.

Tabela 21. Orientações dos componentes obtidas para a planta de polimerização, conforme a estratégia “depth first”.

W_{il} layout2 / modelo (1)	W_{il} layout2 / modelo (2)	W_{il} layout2 / modelo (3)
1,4	1,4	1,2
2,8	2,4	2,6
3,7	3,5	3,5
4,3	4,7	4,1
5,2	5,2	5,2
6,1	6,7	6,1
7,1	7,3	7,3

Tabela 22. Orientações dos componentes obtidas para a planta de polimerização, conforme a estratégia “best bound”.

W_{il} layout2 / modelo (1)	W_{il} layout2 / modelo (2)	W_{il} layout2 / modelo (3)
1,4	1,4	1,2
2,8	2,4	2,2
3,1	3,5	3,1
4,3	4,7	4,5
5,2	5,2	5,2
6,1	6,7	6,1
7,1	7,3	7,1

As rotações dos componentes da planta de polimerização sofreram algumas alterações conforme a estratégia de pesquisa do algoritmo B&B adotada, porém, as equações usadas para a construção dos blocos tridimensionais obedecem as mesmas condições para qualquer modelo ou estratégia de pesquisa.

Assim, os componentes 1, 2 e 5 são relacionados à 1ª. condição ($w_{i,2}$, $w_{i,4}$, $w_{i,6}$, $w_{i,8} = 1$), de acordo com as Eq. 66 a 68 e, os componentes 3, 4, 6 e 7 são relacionados à 2ª. condição ($w_{i,1}$, $w_{i,3}$, $w_{i,5}$, $w_{i,7} = 1$), conforme Eq. 69 a 71.

As Figuras 11, 12 e 13 (Anexo 2) apresentam os *layouts* obtidos para cada modelo, onde se observam pouquíssimas diferenças no modelo (2), devido à pequena variação em algumas posições do centro dos componentes (y_i), como mostra a Tab. 23. Como pode-se observar nesta tabela, os valores de y_i foram os únicos que mudaram no modelo com as restrições lógicas (modelo 2). Os demais valores foram iguais em todos os modelos, para a planta de polimerização.

Tabela 23 . Posições do centro dos componentes obtidas para a planta de polimerização.

	layout2 / modelo (1)			layout2 / modelo (2)			layout2 / modelo (3)		
	“depth first”			“best bound”			“best bound”		
Componentes	x_i	y_i	z_i	x_i	y_i	z_i	x_i	y_i	z_i
1	0,800	2,475	5,950	0,800	2,475	5,950	0,800	2,475	5,950
2	0,800	2,475	3,025	0,800	2,475	3,025	0,800	2,475	3,025
3	1,275	3,200	0,250	1,275	1,750	0,250	1,275	3,200	0,250
4	9,175	3,137	2,050	9,175	1,812	2,050	9,175	3,137	2,050
5	5,275	1,562	2,975	5,275	3,387	2,975	5,275	1,562	2,975
6	4,825	1,562	0,375	4,825	3,387	0,375	4,825	1,562	0,375
7	5,050	2,290	5,950	5,050	2,660	5,950	5,050	2,290	5,950

A Tab. 24 apresenta os resultados obtidos para as variáveis de posição relativa entre os componentes (w_{ij}^x , w_{ij}^y , w_{ij}^z), empregando-se tanto a estratégia “best bound”, quanto a estratégia “depth first”, nos diferentes modelos. As posições em *itálico* são aquelas fixadas no início do problema, para reduzir o tempo computacional.

Tabela 24 . Posições relativas entre os componentes obtidas para a planta de polimerização, para qualquer estratégia ou modelo.

layout2 - modelos (1, 2 e 3)					
w_{ij}^x			w_{ij}^z		
4,1	4,7	6,2	1,2	7,5	
4,2	5,1	6,3	1,3	7,6	
4,3	5,2	7,1	2,3		
4,5	5,3	7,2	5,6		
4,6	6,1	7,3			

E, para a comparação dos três modelos aplicados neste caso de estudo (planta de polimerização) tem-se a Tab. 25, onde são apresentados os números de variáveis e equações empregados para a sua otimização. Os números referentes aos blocos de equações e blocos de variáveis correspondem àqueles empregados somente na implementação do modelo e os demais parâmetros correspondem aos números de equações e variáveis gerados pela execução do programa em *GAMS*, estando relacionados com o número de componentes da planta e com o próprio processo de iteração.

Assim, embora o número de blocos de equações e variáveis seja o mesmo àquele empregado na implementação dos modelos para a unidade petroquímica, os outros parâmetros aumentaram significativamente devido ao maior número de componentes na planta de polimerização.

Tabela 25. Comparação entre o número de equações e variáveis usadas em cada modelo, para a planta de polimerização.

	layout2 / modelo (1)	layout2 / modelo (2)	layout2 / modelo (3)
Bloco de Equações	25	28	25
Bloco de Variáveis	24	24	21
Elementos não-zero	3.345	5.235	4.909
Equações Simples	561	1.191	1.080
Variáveis Simples	523	523	412
Variáveis Discretas	175	175	175

6.3. Planta de Recuperação de Mercúrio

A planta de recuperação de mercúrio possui mais componentes que a planta de polimerização (2 componentes a mais), isto é, apresenta maior complexidade computacional do que os casos já estudados. Portanto, foi necessário fixar outras variáveis (binárias) de posição relativa entre os componentes, face à dificuldade já conhecida, referente ao elevado tempo computacional gasto para a resolução deste tipo de problema.

As variáveis (binárias) de posição relativa entre os componentes que foram fixadas, isto é, atribuído o valor 1, foram as seguintes: $w_{1,2}^x$, $w_{3,2}^x$, $w_{6,1}^x$, $w_{7,1}^x$, $w_{7,8}^x$, $w_{4,3}^x$, $w_{6,3}^x$, $w_{6,4}^x$, $w_{7,5}^x$, $w_{7,2}^x$, $w_{6,5}^x$, $w_{2,5}^y$, $w_{3,5}^y$, $w_{3,7}^y$, $w_{4,5}^y$, $w_{4,7}^y$, $w_{6,7}^y$, $w_{3,1}^y$, $w_{4,1}^y$, $w_{1,5}^y$. Todas as posições foram fixadas considerando-se a configuração do fluxograma de processo.

As Tabelas 26, 27 e 28 apresentam alguns resultados relacionados ao tempo de execução, nós e iterações para a planta de polimerização, onde foram feitos testes com o modelo original (inicial) e com os modelos modificados com as restrições lógicas, e com as substituições de restrições + restrições lógicas, respectivamente.

Tabela 26. Resultados obtidos para a planta de recuperação de mercúrio, com o modelo inicial.

	layout3 / modelo (1) “depth first”	layout3 / modelo (1) “best bound”
Tempo	119s	97s
Iterações MIP	2.505	2.023
Nós	795	683

As Tab.27 e 28 apresentam, também, uma comparação dos resultados obtidos com o modelo original, quanto ao tempo de execução do problema. Nesse sentido, observa-se que o emprego dos modelos modificados apresentou melhores resultados.

Tabela 27. Resultados obtidos para a planta de recuperação de mercúrio, com o modelo modificado pela adição das restrições lógicas.

	layout3 / modelo (2) “depth first”	layout3 / modelo (2) “best bound”
Tempo	78s	86s
Redução de Tempo*	34,45%	11,34%
Iterações MIP	2.696	2.988
Nós	287	310

* Redução percentual de tempo obtida em relação ao modelo original (1).

Tabela 28. Resultados obtidos para a planta de recuperação de mercúrio, com o modelo modificado pela substituição de restrições e pela adição das restrições lógicas.

	layout3 / modelo (3) “depth first”	layout3 / modelo (3) “best bound”
Tempo	78s	86s
Redução de Tempo*	34,45%	11,34%
Iterações MIP	2.696	2.988
Nós	287	310

* Redução percentual de tempo obtida em relação ao modelo original (1).

Apesar desta planta apresentar um número maior de componentes em relação às demais, o tempo computacional de execução do problema foi relativamente pequeno (2 minutos, no máximo) devido à fixação de um número maior de variáveis de posição relativa entre os componentes. Assim, esta planta apresenta maior complexidade computacional em relação às outras e, é exatamente por isso que se justifica a necessidade de se fixar um número maior de variáveis, no intuito de diminuir o tempo total gasto na execução do problema.

O resultado obtido na função objetivo de custo (Eq.5), a partir do modelo implementado em *GAMS* para a planta de recuperação de mercúrio a partir de efluentes eletrolíticos foi o valor ótimo US\$37,613.14. Este valor não deve ser considerado como um valor real, uma vez que diversos dados foram retirados da literatura ou estimados arbitrariamente. O valor é interessante apenas como um parâmetro de comparação, para avaliar o desempenho dos modelos, aplicados para o mesmo caso de estudo.

A Tab. 29 apresenta os custos de terreno, suportes e tubulação obtidos para a planta de recuperação de mercúrio a partir de efluentes eletrolíticos. Novamente, os custos com o terreno e suportes foram os mais representativos na planta otimizada.

Tabela 29. Custos de terreno, suportes e tubulação para a planta de recuperação de mercúrio.

Custos:	Layout3
Terreno	20,300.52
Suportes	15,549.02
Tubulação	1,763.56
Total	US\$37,613.10

As rotações obtidas para os componentes da planta de mercúrio não mudaram em função da estratégia de pesquisa do algoritmo B&B e houve pequena alteração em relação ao modelo original com os outros dois modelos, conforme se observa através da Tab. 30.

Tabela 30. Orientações dos componentes obtidas para a planta de recuperação de mercúrio.

W_{ij} layout3 / modelo (1)	W_{ij} layout3 / modelo (2)	W_{ij} layout3 / modelo (3)
1,1	1,1	1,1
2,6	2,6	2,6
3,4	3,4	3,4
4,8	4,8	4,8
5,3	5,3	5,3
6,3	6,4	6,4
7,1	7,1	7,1
8,1	8,1	8,1
9,5	9,3	9,3

A Tab. 31 apresenta as posições de centro dos componentes, obtidas pelo programa em *GAMS*, cujas posições foram empregadas para a visualização do layout através da Fig. 14 (em anexo). O componente 6 é o único elemento que sofre alteração, devido à mudança de condição para a construção dos blocos tridimensionais, isto é, no modelo (1) ele assume a 2ª. condição ($w_{i,1}$, $w_{i,3}$, $w_{i,5}$, $w_{i,7} = 1$), conforme Eq. 69 a 71 e nos modelos (2 e 3), a 1ª. condição ($w_{i,2}$, $w_{i,4}$, $w_{i,6}$, $w_{i,8} = 1$), de acordo com as Eq. 66 a 68. No entanto, foi construído apenas um desenho da planta, pois os parâmetros A_i e B_i são iguais para cada componente, não mudando o aspecto visual da mesma.

Tabela 31. Posições do centro dos componentes obtidas para a planta de recuperação de mercúrio.

Posições	layout3 – modelos (1), (2) e (3)								
	Componentes								
Relativas (m)	1	2	3	4	5	6	7	8	9
x_i	6,950	1,000	8,125	11,950	6,850	18,150	17,800	10,675	3,350
y_i	4,550	8,425	8,125	8,587	1,900	5,400	1,000	1,625	1,625
z_i	0,600	2,200	0,950	1,500	1,500	2,200	2,200	3,250	1,900

Todos os valores das posições de centro dos componentes são iguais, independente do modelo empregado para a planta de recuperação de mercúrio.

A Tab. 32 apresenta os resultados obtidos para as variáveis de posição relativa entre os componentes (w_{ij}^x , w_{ij}^y , w_{ij}^z), empregando-se tanto a estratégia “best bound” quanto a estratégia “depth first”, para os diferentes modelos. As posições em *itálico* são aquelas fixadas no início do problema, para reduzir o tempo computacional e, como se observa, a maioria já havia sido fixada antes do problema ser otimizado.

Tabela 32 . Posições relativas entre os componentes obtidas para a planta de recuperação de mercúrio, para qualquer estratégia e modelo.

layout2 - modelos (1, 2 e 3)						
w_{ij}^x				w_{ij}^y		
1,2	5,9	6,8	7,9	1,5	3,7	4,7
1,9	6,1	6,9	8,1	2,5	3,8	4,8
3,2	6,2	7,1	8,2	2,9	3,9	6,7
4,2	6,3	7,2	8,5	3,1	4,1	
4,3	6,4	7,5	8,9	3,5	4,5	
4,9	6,5	7,8				

Na Tab. 33 estão relacionados os números de equações e variáveis empregados na implementação do modelo e na execução do programa em *GAMS*, para a planta de recuperação de mercúrio a partir de efluentes eletrolíticos.

Tabela 33. Comparação entre o número de equações e variáveis usadas em cada modelo, para a planta de recuperação de mercúrio.

	layout3 / modelo (1)	layout3 / modelo (2)	layout3 / modelo (3)
Bloco de Equações	25	28	25
Bloco de Variáveis	24	24	21
Elementos não-zero	4.857	9.393	9.271
Equações Simples	784	2.296	2.227
Variáveis Simples	691	691	622
Variáveis Discretas	268	268	268

Nota-se que o número de equações e variáveis gerado pelo programa, conforme a Tab. 33, aumentou consideravelmente em relação às demais plantas, devido ao acréscimo de componentes, isto é, esta planta tem 9 componentes principais, enquanto a unidade petroquímica e a planta de polimerização continham 6 e 7 componentes, respectivamente.

6.4. Comentários

É interessante fazer algumas observações sobre os resultados obtidos a partir das otimizações realizadas com os diferentes modelos e diferentes casos de estudo:

- existem várias configurações alternativas que resultam na mesma função objetivo com valor ótimo; isto indica que rotações e posições relativas diferentes podem ser igualmente aceitáveis como solução ótima, para o *modelo dado*.
- o fato do valor absoluto do custo ótimo relativo aos suportes e ao perímetro ser bem maior que o valor do custo das tubulações *não* implica que este último seja menos importante, pois ele afeta diretamente o tamanho e a altura da planta. Logo, o que realmente importa é a *variação* de cada custo de uma configuração para outra e *não* o seu valor absoluto.
- o tempo de *CPU* para se encontrar o valor ótimo varia de forma imprevisível com relação à mudança de certas variáveis e/ou à mudança de certas estratégias, como “depth first” e “best bound”.
- de forma geral, pode-se dizer que em certos casos diminuir o número de nós inspecionados (através das restrições lógicas) e diminuir o número de variáveis e de equações (através da troca de restrições) leva a uma redução apreciável no tempo de cálculo, embora, em outros casos, fixar muitas posições relativas pode influenciar esse resultado.

CAPÍTULO 7

CONCLUSÕES

O desenvolvimento e a implementação do modelo existente no pacote *GAMS/CPLEX*, inicialmente para a unidade petroquímica, foi um processo lento, sendo que a etapa de execução do problema foi um dos fatores que mais interferiu neste processo. Assim, o tempo de execução obtido para o modelo existente na literatura foi de aproximadamente 9 horas (em uma estação de trabalho *SunSparc5*), uma vez que se trata de um problema *NP-completo*, onde a complexidade de tempo varia exponencialmente em N (número de componentes). No entanto, este mesmo caso-estudo apresentou considerável melhora ao ser implementado com os novos modelos, com o acréscimo de restrições lógicas e/ou substituição de algumas restrições, reduzindo o número de iterações realizadas e nós inspecionados e, conseqüentemente, o tempo gasto para a sua resolução.

A planta de polimerização, no entanto, apresentou melhores resultados, em relação ao tempo computacional, quando implementada no modelo original. Por outro lado, a planta de recuperação de mercúrio a partir de efluentes eletrolíticos apresentou melhores resultados com os modelos novos, de maneira semelhante ao ocorrido com a unidade petroquímica.

Desta forma, não se pode afirmar qual modelo é melhor para reduzir o tempo computacional, embora em dois casos de estudo os modelos novos tenham se mostrado mais eficientes. A escolha das variáveis de posição relativa entre os componentes a serem

fixadas interfere neste tempo computacional, também, portanto, é necessário um estudo mais aprimorado nesse sentido.

Ainda, pode-se concluir que o emprego do pacote comercial de otimização *GAMS* neste trabalho apresentou a vantagem no que se refere à facilidade de programação, que está diretamente relacionada ao tempo de implementação dos modelos nos diferentes casos de estudo.

Assim, pode-se comprovar que é possível resolver de forma exata problemas de otimização do posicionamento dos componentes em *layouts*, através da programação em MILP, implementado-se em *GAMS/CPLEX*, em um tempo aceitável de cálculo.

7. 1. Sugestões para trabalhos futuros

Para se atingir o objetivo de reduzir o tempo computacional, outras estratégias, além de mudar o modelo, podem:

- usar algoritmos mais eficientes para *MILP*;
- usar algoritmos com computação em paralelo.

REFERÊNCIAS BIBLIOGRÁFICAS

- AMORESE, L.; CENA, V. and MUSTACCHI, C. A heuristic for the compact location of process components. *Chemical Engineering Science* 32, 119-124, 1977.
- ANDERSON, F. V. Plant layout. In: Kirk-Othmer, *Encyclopedia of Chemical Technology*, 3rd. ed., 18, pp.23-43, 1982.
- BROOKE, A., KENDRICK, D. and MEERAUS, A. *GAMS: a user's guide*. San Francisco: The Scientific Press, 1988.
- BUSH, M. J. and WELLS, G. L. . Optimum plant layout scores on maintenance, operation, safety. *Process Engineering*, 53 (9), pp. 135-137, 1972.
- Economic Indicators. In: *Chemical Engineering*, 107 (10), p.150, september, 2000.
- FRANCIS, R.L. and WHITE, J. A. *Facility layout and location: an analytical approach*. New Jersey: Prentice Hall, 1974.
- GEORGIADIS, M.C. and MACCHIETTO, S. Layout of process plants: a novel approach. *Computers & Chemical Engineering*, 21, Suppl., pp. S337-s342, 1997.

- GEORGIADIS, M.C.; SCHILLING, G.; ROTSTEIN, G. E.; MACCHIETTO, S. A general mathematical programming approach for process plant layout. *Computers & Chemical Engineering*, 23, pp. 823-840, 1999.
- GREEN, A.; JOHNSON, B. and JOHN, A. Process intensification magnifies profits. *Chemical Engineering*, 106 (13), pp.66-73, 1999.
- GUIRARDELLO, R. *Optimization of Process Plant Layout*. Ph.D. Thesis, University of Wisconsin, Madison, USA ,1993.
- GUIRARDELLO, R. *Otimização de Layout de Processos Químicos*, Anais do 10º Congresso Brasileiro de Engenharia Química, São Paulo - SP, Vol. 2, pp. 1169-1174, 1994.
- GUNN, D. J. The optimised layout of a chemical plant by digital computer. *Computer Aided Design*, Spring ,11-16 ,1970.
- GUTHRIE, K. M. *Process plant estimating evaluation and control*. San Francisco: Craftsman Book Company of America, 1974.
- HILLIER, F.S. and LIEBERMAN, G.J. *Introduction to Mathematical Programming*. McGraw-Hill, 1990.
- JAMES, R.W. and ALCORN, P.A. *A guide to facilities planning*. New Jersey: Prentice Hall, 1991.
- KAURA, M. L. Plot plans include safety. *Hydrocarbon Processing*, 59 (7), pp. 183-194, 1980.

- KERN, R. How to manage plant design to obtain minimum cost. *Chemical Engineering*, 84 (11), pp. 130-136, 1977.
- KERN, R. Arranging the housed chemical process plant. *Chemical Engineering*, 85 (16), pp. 123-130, 1978.
- LEESLEY, M.E. *Computer-Aided Process Plant Design*. Gulf Publishing Company, Houston, pp. 1123-1182, (1982).
- LUDWIG, E.E. *Applied process design: for chemical and petrochemical plants*. Houston: Gulf Publishing, 3rd. ed., V.1, 1995.
- MECKLENBURGH, J. C.. Plant layout practice. *Chemical Engineering Symposium Series*, 73, E1-E16, 1982.
- MECKLENBURGH, J. C.. *Process plant layout*. New York: Halsted Press - John Wiley & Sons, 1985.
- MEREDITH, J.R. *The management of operations*. New York: John Wiley & Sons, 3rd. ed., 1987.
- MURTY, K. G. *Operations Research*. New Jersey: Prentice Hall, 1995.
- SCHMIDT-TRAUB, H., HOLTKÖTTER, T.; LEDERHOSE, M. and LEUDERS, P. An approach to plant layout optimization. *Chemical Engineering Technology*, 22 (2), pp. 105-109, 1999.

- TIN-LOI, F. Plastic limit analysis of plane frames and grids using GAMS. *Computers & Structures*, 54(1), pp. 15-25, 1995.
- ULRICH, G.D. *A guide to chemical engineering process design and economics*. New York: John Wiley & Sons, 1984.
- VIEIRA, A.C.G. *Manual de Layout (Arranjo físico)*. RJ: Confederação Nacional da Indústria, 1983.

ANEXOS

ANEXO 1

UNIDADE PETROQUÍMICA

Tabela 34. Dimensões e áreas dos componentes para a unidade petroquímica.

Componentes	A_i (m)	B_i (m)	C_i (m)	Área_i (m ²)
1	1,88	1,88	22,50	3,5344
2	1,41	5,56	1,41	7,8396
3	0,77	7,03	0,77	5,4131
4	0,47	5,65	2,14	2,6555
5	1,41	2,44	0,47	1,3254
6	1,80	3,00	0,75	2,7000

Fonte: GUIARDELLO (1993).

Tabela 35 . Distâncias mínimas entre os componentes para a unidade petroquímica.

Componente i	Componente j	D_{ij}^h (m)	D_{ij}^v (m)
1	2	6,00	4,80
1	3	2,40	1,80
1	4	6,00	4,80
1	5	2,40	1,20
1	6	2,40	1,20
2	3	2,40	2,70
2	4	2,40	2,10
2	5	4,50	4,20
2	6	4,50	4,20
3	4	1,50	1,80
3	5	1,50	1,80
3	6	1,50	1,80
4	5	1,50	1,20
4	6	1,50	1,20
5	6	1,50	0,90

Fonte: GUIARDELLO (1993).

Tabela 36 . Posicionamento dos bocais para a unidade petroquímica.

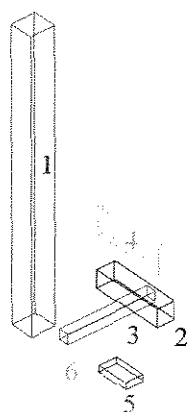
Componente <i>i</i>	<i>j</i>	<i>p</i>	F_{ji}^x	F_{ji}^y	F_{ji}^z
1	1	1	1,00	0,00	0,21
	2	2	0,00	-1,00	-0,89
	3	3	0,00	0,00	1,00
	4	4	1,00	0,00	0,94
	5	5	0,00	100	-0,57
	6	6	0,00	1,00	-0,65
2	1	7	0,00	0,69	1,00
	2	8	0,00	-0,69	1,00
	3	9	0,00	-0,77	-1,00
3	1	10	0,00	0,00	-1,00
	2	11	0,00	-0,69	1,00
	3	12	0,00	0,69	1,00
	4	13	0,00	0,92	-1,00
	5	14	0,00	0,92	1,00
4	1	15	0,00	-0,68	1,00
	2	16	0,00	-0,68	-1,00
	3	17	0,00	-0,87	-1,00
	4	18	0,00	-0,87	1,00
5	1	19	-1,00	-0,68	1,00
	2	20	-1,00	0,91	1,00
	3	21	-1,00	-0,80	0,00
	4	22	-1,00	0,80	0,00
6	1	23	-1,00	0,92	1,00
	2	24	-1,00	-0,60	1,00
	3	25	-1,00	0,76	0,00
	4	26	-1,00	-0,76	0,00

Fonte: GUIARDELLO (1993).

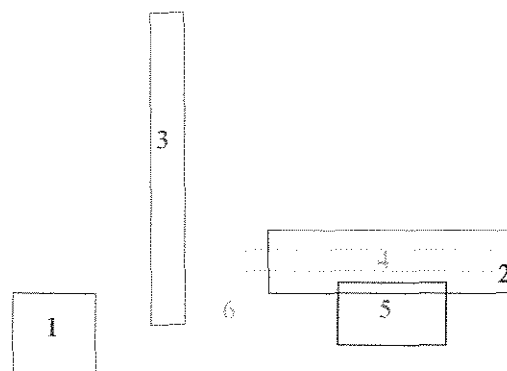
Tabela 37. Dados para os tubos na otimização do layout da unidade petroquímica.

Tubo k	$k_s(k)$	$k_t(k)$	C_k^P
1	23	1	5,249
2	24	1	5,249
3	3	15	26,247
4	19	4	5,249
5	20	4	5,249
6	5	10	39,042
7	11	6	39,042
8	12	6	39,042
9	7	16	9,843
10	9	21	14,764
11	9	22	14,764

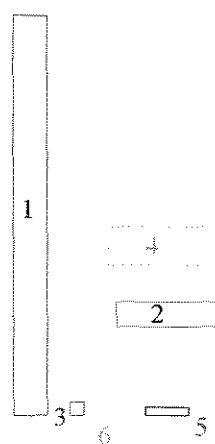
Fonte: GUIARDELLO (1993).



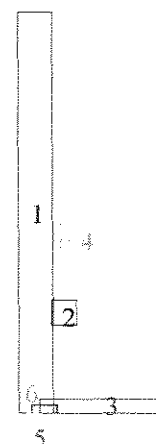
Vista Isométrica SE



Vista Superior

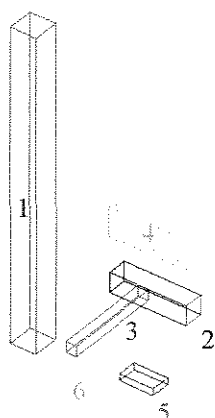


Vista Frontal

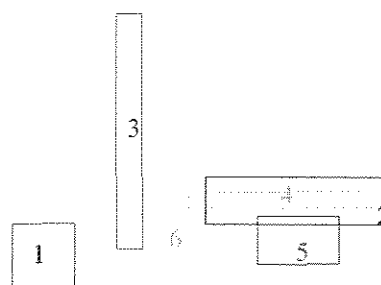


Vista Lateral

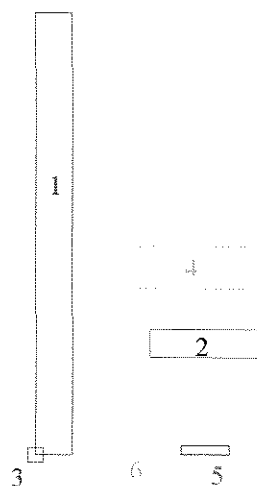
Figura 8. Unidade Petroquímica, modelo original, empregando-se B&B -" Depth First".
Componentes: (1) Coluna de Destilação, (2) Tanque de Refluxo, (3) Refervedor,
(4) Condensadores, (5) Bombas de Refluxo, (6) Bombas de Produto Superaquecido.



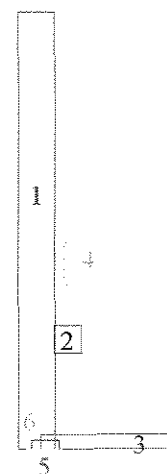
Vista Isométrica SE



Vista Superior

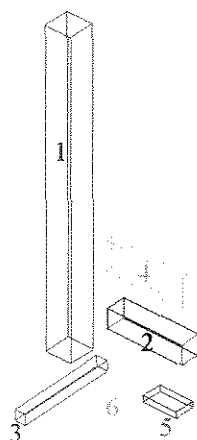


Vista Frontal

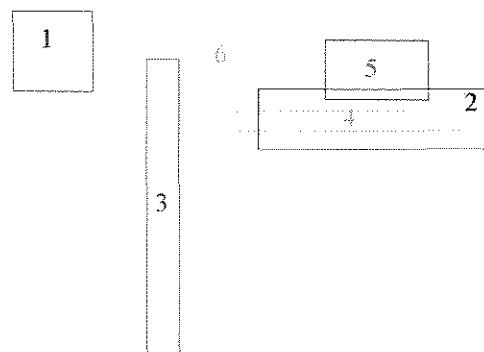


Vista Lateral

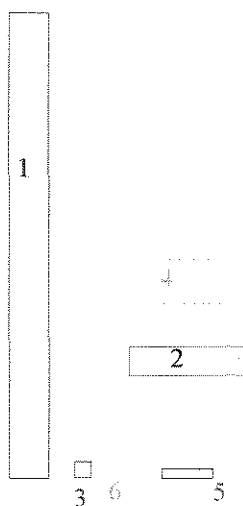
Figura 9. Unidade Petroquímica, modelo (2), empregando-se B&B - "Best Bound".
Componentes: (1) Coluna de Destilação, (2) Tanque de Refluxo, (3) Refervedor,
(4) Condensadores, (5) Bombas de Refluxo, (6) Bombas de Produto Superaquecido.



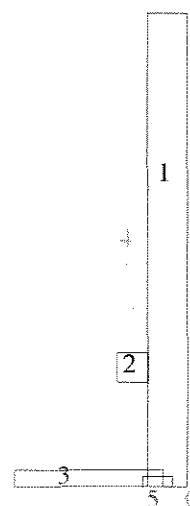
Vista Isométrica SE



Vista Superior



Vista Frontal



Vista Lateral

Figura 10. Unidade Petroquímica, modelo (3), empregando-se B&B - "Best Bound".
Componentes: (1) Coluna de Destilação, (2) Tanque de Refluxo, (3) Refervedor, (4) Condensadores, (5) Bombas de Refluxo, (6) Bombas de Produto Superaquecido.

*** IMPLEMENTAÇÃO DO MODELO ORIGINAL PARA A UNIDADE
PETROQUÍMICA, EM GAMS/CPLEX**

\$ONTEXT

----- LAYOUT_1 -----

A planta a ser otimizada e' de uma unidade petroquímica (GUIARDELLO,1993).

OBS: Originalmente, na Eq.6 existe o termo de elevacao (EL(i)), mas como para este caso-estudo ele e' nulo, foi excluido da eq. referida.

* MODELO ORIGINAL proposto por Guirardello (1993).

* Algoritmo BEST BOUND

OS ELEMENTOS I e J REPRESENTAM:

1-COLUNA DE DESTILACAO

2-TANQUE DE REFLUXO

3-REFERVEDOR

4-CONDENSADORES 1 E 2

5-BOMBAS DE REFLUXO 1 E 2

6-BOMBAS DE PRODUTO SUPERAQUECIDO 1 E 2

\$OFFTEXT

\$offlisting

SETS

I	componentes	/ 1*6 /
K	tubos	/ 1*11 /
P	nozzle	/ 1*26 /
L	rotacoes	/ 1*8 /
KK	auxiliar	/ 1*4 /

KS(K,P)	/ 1 . 23
	2 . 24
	3 . 3
	4 . 19
	5 . 20
	6 . 5
	7 . 11
	8 . 12
	9 . 7
	10 . 9
	11 . 9 /

KT(K,P) / 1 . 1
 2 . 1
 3 . 15
 4 . 4
 5 . 4
 6 . 10
 7 . 6
 8 . 6
 9 . 16
 10 . 21
 11 . 22 /

PI(P,I) / 1 . 1
 2 . 1
 3 . 1
 4 . 1
 5 . 1
 6 . 1
 7 . 2
 8 . 2
 9 . 2
 10 . 3
 11 . 3
 12 . 3
 13 . 3
 14 . 3
 15 . 4
 16 . 4
 17 . 4
 18 . 4
 19 . 5
 20 . 5
 21 . 5
 22 . 5
 23 . 6
 24 . 6
 25 . 6
 26 . 6 /;

ALIAS (I,J) ;

SCALARS

CPER custo por unidade de comprimento do tubo k /298.43/
 MM um numero grande considerado no modelo /51.00/ ;

PARAMETERS

A(I) dimensoes dos componentes i (m)

/1	1.88
2	1.41
3	0.77
4	0.47
5	1.41
6	1.80/

B(I) dimensoes dos componentes i (m)

/1	1.88
2	5.56
3	7.03
4	5.65
5	2.44
6	3.00/

C(I) dimensoes dos componentes i (m)

/1	22.50
2	1.41
3	0.77
4	2.14
5	0.47
6	0.75/

CP(K) custo por unidade de comprimento para o tubo k (dolares por m)

/1	5.249
2	5.249
3	26.247
4	5.249
5	5.249
6	39.042
7	39.042
8	39.042
9	9.843
10	14.764
11	14.764 /

AREA(I) area dos componentes i (metros quadrados),

CSA(I,KK) fator de custo unitario para o suporte do componente i ,

CSB(I,KK) fator de custo unitario para o suporte do componente i ,

M(I,J) ;

$$\text{AREA}(I) = A(I) * B(I) ;$$

$$\text{CSA}(I,'1') = \text{AREA}(I) * 62.8765 ;$$

$$\text{CSA}(I,'2') = \text{AREA}(I) * 95.3989 ;$$

$$\text{CSA}(I,'3') = \text{AREA}(I) * 115.9964 ;$$

$$\text{CSA}(I,'4') = \text{AREA}(I) * 140.9302 ;$$

$$\text{CSB}(I,'1') = \text{AREA}(I) * 0.0 ;$$

$$\text{CSB}(I,'2') = \text{AREA}(I) * (-99.1281) ;$$

$$\text{CSB}(I,'3') = \text{AREA}(I) * (-224.6904) ;$$

$$\text{CSB}(I,'4') = \text{AREA}(I) * (-528.6832) ;$$

$$M(I,J) = 0.5 * ((\text{MAX}(A(I),B(I)) + (\text{MAX}(A(J),B(J)))));$$

TABLE Dh(I,J) distancias minimas horizontais de seguranca entre os componentes (m)

	1	2	3	4	5	6
1		6.00	2.40	6.00	2.40	2.40
2	6.00		2.40	2.40	4.50	4.50
3	2.40	2.40		1.50	1.50	1.50
4	6.00	2.40	1.50		1.50	1.50
5	2.40	4.50	1.50	1.50		1.50
6	2.40	4.50	1.50	1.50	1.50	

TABLE Dv(I,J) distancias minimas verticais de seguranca entre os componentes (m)

	1	2	3	4	5	6
1		4.80	1.80	4.80	1.20	1.20
2	4.80		2.70	2.10	4.20	4.20
3	1.80	2.70		1.80	1.80	1.80
4	4.80	2.10	1.80		1.20	1.20
5	1.20	4.20	1.80	1.20		0.90
6	1.20	4.20	1.80	1.20	0.90	

TABLE NPOSI(P,I,*) posicoes dos "nozzles"

	FX	FY	FZ
1.1	1.00	0.00	0.21
2.1	0.00	-1.00	-0.89
3.1	0.00	0.00	1.00
4.1	1.00	0.00	0.94
5.1	0.00	1.00	-0.57
6.1	0.00	1.00	-0.65
7.2	0.00	0.69	1.00
8.2	0.00	-0.69	1.00
9.2	0.00	-0.77	-1.00
10.3	0.00	0.00	-1.00
11.3	0.00	-0.69	1.00
12.3	0.00	0.69	1.00
13.3	0.00	0.92	-1.00
14.3	0.00	0.92	1.00
15.4	0.00	-0.68	1.00
16.4	0.00	-0.68	-1.00
17.4	0.00	-0.87	-1.00
18.4	0.00	-0.87	1.00
19.5	-1.00	-0.68	1.00
20.5	-1.00	0.91	1.00
21.5	-1.00	-0.80	0.00
22.5	-1.00	0.80	0.00
23.6	-1.00	0.92	1.00
24.6	-1.00	-0.60	1.00
25.6	-1.00	0.76	0.00
26.6	-1.00	-0.76	0.00

VARIABLES

XX dimensao do layout
 YY dimensao do layout
 ZZ dimensao do layout
 T custo total do layout

CS(I) custo dos suportes para o componente i

LXP(K) comprimento do tubo k usando a distancia de Manhattan em x
 LXN(K) comprimento do tubo k usando a distancia de Manhattan em x
 LYP(K) comprimento do tubo k usando a distancia de Manhattan em y
 LYN(K) comprimento do tubo k usando a distancia de Manhattan em y
 LZP(K) comprimento do tubo k usando a distancia de Manhattan em z
 LZN(K) comprimento do tubo k usando a distancia de Manhattan em z

X(I) posicao do centro do componente i
 Y(I) posicao do centro do componente i
 Z(I) posicao do centro do componente i

XN(P) posicao do nozzle p
 YN(P) posicao do nozzle p
 ZN(P) posicao do nozzle p

DX(I,J) distancia Manhattan entre os centros dos componentes i e j em x,y,z
 DY(I,J) distancia Manhattan entre os centros dos componentes i e j em x,y,z
 DZ(I,J) distancia Manhattan entre os centros dos componentes i e j em x,y,z

W(I,L) variavel inteira para a rotacao do componente i
 WX(I,J) variavel inteira para a posicao relativa entre os componentes i e j
 WY(I,J) variavel inteira para a posicao relativa entre os componentes i e j
 WZ(I,J) variavel inteira para a posicao relativa entre os componentes i e j ;

POSITIVE VARIABLES XX,YY,ZZ,CS,LXP,LXN,LYP,LYN,LZP,LZN,DX,DY,DZ;

BINARY VARIABLES W, WX, WY, WZ;

WX.fx('5','6') = 1;
 WZ.fx('4','2') = 1;
 WZ.fx('2','5') = 1;
 WZ.fx('4','5') = 1;

EQUATIONS

CUSTO define a funcao objetiva
 SUPORTE(I,KK) custo de suportes para os componentes i
 EQ01(I)
 EQ02(I)
 EQ03(I)
 EQ04(I)
 EQ05(I)
 EQ06(I)
 EQ07(I,J)
 EQ08(I,J)
 EQ09(I,J)
 EQ10(I,J)
 EQ11(I,J)
 EQ12(I,J)
 EQ13(I,J)
 EQ14(I,J)
 EQ15(I,J)
 EQ16(P)
 EQ17(P)
 EQ18(P)
 EQ19(K)
 EQ20(K)
 EQ21(K)
 EQ22(I)
 EQ23(I,J) ;

CUSTO .. T =e= 2*CPER*(XX+YY) + SUM(I, CS(I))
 + SUM(K,CP(K)*(LXP(K) + LXN(K) + LYP(K) + LYN(K) + LZP(K) + LZN(K))) ;

SUPORTE(I,KK) .. CS(I) =g= CSA(I,KK)*(Z(I) + C(I)/2) + CSB(I,KK);

EQ01(I) .. XX =g= X(I) + 0.5*A(I)*(W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*B(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"));

EQ02(I) .. YY =g= Y(I) + 0.5*B(I)*(W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*A(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"));

EQ03(I) .. ZZ =g= Z(I) + 0.5*C(I);

EQ04(I) .. X(I) =g= 0.5*A(I)*(W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*B(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"));

EQ05(I) .. Y(I) =g= 0.5*B(I)*(W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*A(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"));

EQ06(I) .. Z(I) =g= 0.5*C(I);

EQ07(I,J)\$ (ORD(I) NE ORD(J)) ..
 DX(I,J) =g= 0.5 * A(I) * (W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*B(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"))
 + 0.5*A(J)*(W(J,"1")+W(J,"3")+W(J,"5")+W(J,"7"))
 + 0.5*B(J)*(W(J,"2")+W(J,"4")+W(J,"6")+W(J,"8"))
 + (DH(I,J)+M(I,J)) * WX(I,J) - M(I,J);

EQ08(I,J)\$ (ORD(I) NE ORD(J)) ..
 DY(I,J) =g= 0.5 * B(I) * (W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*A(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"))
 + 0.5*B(J)*(W(J,"1")+W(J,"3")+W(J,"5")+W(J,"7"))
 + 0.5*A(J)*(W(J,"2")+W(J,"4")+W(J,"6")+W(J,"8"))
 + (DH(I,J)+M(I,J)) * WY(I,J) - M(I,J);

EQ09(I,J)\$ (ORD(I) NE ORD(J)) ..
 DZ(I,J) =g= (DV(I,J) + 0.5*C(I) + 0.5*C(J))*WZ(I,J) ;

EQ10(I,J)\$ (ORD(I) NE ORD(J)) ..
 DX(I,J) =l= MM*(1-WX(J,I));

EQ11(I,J)\$ (ORD(I) NE ORD(J)) ..
 DY(I,J) =l= MM*(1-WY(J,I));

EQ12(I,J)\$ (ORD(I) NE ORD(J)) ..
 DZ(I,J) =l= MM*(1-WZ(J,I));

EQ13(I,J)\$ (ORD(I) LT ORD(J)) .. X(I) - X(J) - DX(I,J) + DX(J,I) =e= 0 ;

EQ14(I,J)\$(ORD(I) LT ORD(J)) .. $Y(I) - Y(J) - DY(I,J) + DY(J,I) = 0$;

EQ15(I,J)\$(ORD(I) LT ORD(J)) .. $Z(I) - Z(J) - DZ(I,J) + DZ(J,I) = 0$;

EQ16(P) .. $XN(P) = \text{SUM}(\$PI(P,I), X(I) + NPOSI(P,I,'FX')*0.5*A(I)*(W(I,"1")-W(I,"3")+W(I,"5")-W(I,"7")) + NPOSI(P,I,'FY')*0.5*B(I)*(-W(I,"2") + W(I,"4")-W(I,"6")+W(I,"8")))$;

EQ17(P) .. $YN(P) = \text{SUM}(\$PI(P,I), Y(I) + NPOSI(P,I,'FY')*0.5*B(I)*(W(I,"1")-W(I,"3")-W(I,"5")+W(I,"7")) + NPOSI(P,I,'FX')*0.5*A(I)*(+W(I,"2")-W(I,"4")-W(I,"6")+W(I,"8")))$;

EQ18(P) .. $ZN(P) = \text{SUM}(\$PI(P,I), Z(I) + NPOSI(P,I,'FZ')*0.5*C(I))$;

EQ19(K) .. $\text{SUM}(P\$KS(K,P), XN(P)) - \text{SUM}(P\$KT(K,P), XN(P)) - LXP(K) + LXN(K) = 0$;

EQ20(K) .. $\text{SUM}(P\$KS(K,P), YN(P)) - \text{SUM}(P\$KT(K,P), YN(P)) - LYP(K) + LYN(K) = 0$;

EQ21(K) .. $\text{SUM}(P\$KS(K,P), ZN(P)) - \text{SUM}(P\$KT(K,P), ZN(P)) - LZP(K) + LZN(K) = 0$;

EQ22(I) .. $\text{SUM}(L, W(I,L)) = 1$;

EQ23(I,J)\$(ORD(I) LT ORD(J)) .. $WX(I,J) + WX(J,I) + WY(I,J) + WY(J,I) + WZ(I,J) + WZ(J,I) = 1$;

MODEL LAYOUT /ALL/ ;
layout.optfile = 1;

OPTION MIP = CPLEX ;

option reslim = 1000000000;
option iterlim = 25000000;
option optcr = 0.0;
option work = 1000.0;

\$offsymxref

SOLVE LAYOUT USING MIP MINIMIZING T ;

DISPLAY XX.L, YY.L, ZZ.L, X.L, Y.L, Z.L, W.L, WX.L, WY.L, WZ.L ;

*** IMPLEMENTAÇÃO DO MODELO 3 PARA A UNIDADE PETROQUÍMICA,
EM GAMS/CPLEX.**

\$ONTEXT

----- VERSAO MODIFICADA DO LAYOUT1 -----

MODELO 3

A planta a ser otimizada e' de uma unidade petroquimica (GUIRARDELLO,1993).

OBS: Originalmente, na Eq.6 existe o termo de elevacao (EL(i)), mas como para este caso-estudo ele e' nulo, foi excluído da eq. referida.

* Algoritmo Best Bound

* Inclui Mudancas nas Equações...

OS ELEMENTOS I e J REPRESENTAM:

1-COLUNA DE DESTILACAO

2-TANQUE DE REFLUXO

3-REFERVEDOR

4-CONDENSADORES 1 E 2

5-BOMBAS DE REFLUXO 1 E 2

6-BOMBAS DE PRODUTO SUPERAQUECIDO 1 E 2

\$OFFTEXT

\$offlisting

SETS

I	componentes	/ 1*6 /
K	tubos	/ 1*11 /
P	nozzle	/ 1*26 /
L	rotacoes	/ 1*8 /
KK	auxiliar	/ 1*4 /;

ALIAS (I,J,IJ) ;

SETS

KS(K,P,I)	/ 1 . 23 . 6
	2 . 24 . 6
	3 . 3 . 1
	4 . 19 . 5
	5 . 20 . 5
	6 . 5 . 1
	7 . 11 . 3
	8 . 12 . 3
	9 . 7 . 2

	10 . 9 . 2
	11 . 9 . 2 /
KT(K,P,I)	/ 1 . 1 . 1
	2 . 1 . 1
	3 . 15 . 4
	4 . 4 . 1
	5 . 4 . 1
	6 . 10 . 3
	7 . 6 . 1
	8 . 6 . 1
	9 . 16 . 4
	10 . 21 . 5
	11 . 22 . 5 /
PI(K,I,J)	/ 1 . 6 . 1
	2 . 6 . 1
	3 . 1 . 4
	4 . 5 . 1
	5 . 5 . 1
	6 . 1 . 3
	7 . 3 . 1
	8 . 3 . 1
	9 . 2 . 4
	10 . 2 . 5
	11 . 2 . 5 / ;

SCALARS

CPER	custo por unidade de comprimento do tubo k /298.43/
MM	um numero grande considerado no modelo /51.00/ ;

PARAMETERS

A(I) dimensoes dos componentes i (m)

/1	1.88
2	1.41
3	0.77
4	0.47
5	1.41
6	1.80/

B(I) dimensoes dos componentes i (m)

/1	1.88
2	5.56
3	7.03
4	5.65
5	2.44
6	3.00/

C(I) dimensoes dos componentes i (m)

/1	22.50
2	1.41
3	0.77
4	2.14
5	0.47
6	0.75/

CP(K) custo por unidade de comprimento para o tubo k (dolares por m)

/1	5.249
2	5.249
3	26.247
4	5.249
5	5.249
6	39.042
7	39.042
8	39.042
9	9.843
10	14.764
11	14.764 /

AREA(I) area dos componentes i (metros quadrados),

CSA(I,KK) fator de custo unitario para o suporte do componente i ,

CSB(I,KK) fator de custo unitario para o suporte do componente i ,

M(I,J) ;

$$AREA(I) = A(I) * B(I) ;$$

$$CSA(I,'1') = AREA(I) * 62.8765 ;$$

$$CSA(I,'2') = AREA(I) * 95.3989 ;$$

$$CSA(I,'3') = AREA(I) * 115.9964 ;$$

$$CSA(I,'4') = AREA(I) * 140.9302 ;$$

$$CSB(I,'1') = AREA(I) * 0.0 ;$$

$$CSB(I,'2') = AREA(I) * (-99.1281) ;$$

$$CSB(I,'3') = AREA(I) * (-224.6904) ;$$

$$CSB(I,'4') = AREA(I) * (-528.6832) ;$$

$$M(I,J) = 0.5*((MAX(A(I),B(I)))+(MAX(A(J),B(J))));$$

TABLE Dh(I,J) distancias minimas horizontais de seguranca entre os componentes (m)

	1	2	3	4	5	6
1		6.00	2.40	6.00	2.40	2.40
2	6.00		2.40	2.40	4.50	4.50
3	2.40	2.40		1.50	1.50	1.50
4	6.00	2.40	1.50		1.50	1.50
5	2.40	4.50	1.50	1.50		1.50
6	2.40	4.50	1.50	1.50	1.50	

TABLE Dv(I,J) distancias minimas verticais de seguranca entre os componentes (m)

	1	2	3	4	5	6
1		4.80	1.80	4.80	1.20	1.20
2	4.80		2.70	2.10	4.20	4.20
3	1.80	2.70		1.80	1.80	1.80
4	4.80	2.10	1.80		1.20	1.20
5	1.20	4.20	1.80	1.20		0.90
6	1.20	4.20	1.80	1.20	0.90	

TABLE NPOSI(P,I,*) posicoes dos "nozzles"

	FX	FY	FZ
1.1	1.00	0.00	0.21
2.1	0.00	-1.00	-0.89
3.1	0.00	0.00	1.00
4.1	1.00	0.00	0.94
5.1	0.00	1.00	-0.57
6.1	0.00	1.00	-0.65
7.2	0.00	0.69	1.00
8.2	0.00	-0.69	1.00
9.2	0.00	-0.77	-1.00
10.3	0.00	0.00	-1.00
11.3	0.00	-0.69	1.00
12.3	0.00	0.69	1.00
13.3	0.00	0.92	-1.00
14.3	0.00	0.92	1.00
15.4	0.00	-0.68	1.00
16.4	0.00	-0.68	-1.00
17.4	0.00	-0.87	-1.00
18.4	0.00	-0.87	1.00
19.5	-1.00	-0.68	1.00
20.5	-1.00	0.91	1.00
21.5	-1.00	-0.80	0.00
22.5	-1.00	0.80	0.00
23.6	-1.00	0.92	1.00

24.6	-1.00	-0.60	1.00
25.6	-1.00	0.76	0.00
26.6	-1.00	-0.76	0.00

VARIABLES

XX dimensao do layout
YY dimensao do layout
ZZ dimensao do layout
T custo total do layout

CS(I) custo dos suportes para o componente i

LXP(K) comprimento do tubo k usando a distancia de Manhattan em x
LXN(K) comprimento do tubo k usando a distancia de Manhattan em x
LYP(K) comprimento do tubo k usando a distancia de Manhattan em y
LYN(K) comprimento do tubo k usando a distancia de Manhattan em y
LZP(K) comprimento do tubo k usando a distancia de Manhattan em z
LZN(K) comprimento do tubo k usando a distancia de Manhattan em z

X(I) posicao do centro do componente i
Y(I) posicao do centro do componente i
Z(I) posicao do centro do componente i

DX(I,J) distancia Manhattan entre os centros dos componentes i e j em x,y,z
DY(I,J) distancia Manhattan entre os centros dos componentes i e j em x,y,z
DZ(I,J) distancia Manhattan entre os centros dos componentes i e j em x,y,z

W(I,L) variavel inteira para a rotacao do componente i
WX(I,J) variavel inteira para a posicao relativa entre os componentes i e j

WY(I,J) variavel inteira para a posicao relativa entre os componentes i e j
WZ(I,J) variavel inteira para a posicao relativa entre os componentes i e j ;

POSITIVE VARIABLES XX,YY,ZZ,CS,LXP,LXN,LYP,LYN,LZP,LZN,DX,DY,DZ;

BINARY VARIABLES W, WX, WY, WZ;

WX.fx('5','6') = 1;
WZ.fx('4','2') = 1;
WZ.fx('2','5') = 1;
WZ.fx('4','5') = 1;

EQUATIONS

CUSTO define a funcao objetiva
SUPORTE(I,KK) custo de suportes para componentes i
EQ01(I)
EQ02(I)
EQ03(I)

EQ04(I)
 EQ05(I)
 EQ06(I)
 EQ07(I,J)
 EQ08(I,J)
 EQ09(I,J)
 EQ10(I,J)
 EQ11(I,J)
 EQ12(I,J)
 EQ13(I,J)
 EQ14(I,J)
 EQ15(I,J)
 EQ16(K)
 EQ17(K)
 EQ18(K)
 EQ19(I)
 EQ20(I,J)
 EQ21(I,J,IJ)
 EQ22(I,J,IJ)
 EQ23(I,J,IJ) ;

CUSTO .. T =e= 2*CPER*(XX+YY) + SUM(I, CS(I))
 + SUM(K, CP(K)*(LXP(K)+LXN(K)+LYP(K)+LYN(K)+LZP(K)+LZN(K))) ;

SUPORTE(I, KK) .. CS(I) =g= CSA(I, KK)*(Z(I) + C(I)/2) + CSB(I, KK);

EQ01(I) .. XX =g= X(I) + 0.5*A(I)*(W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*B(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"));

EQ02(I) .. YY =g= Y(I) + 0.5*B(I)*(W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*A(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"));

EQ03(I) .. ZZ =g= Z(I) + 0.5*C(I);

EQ04(I) .. X(I) =g= 0.5*A(I)*(W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*B(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"));

EQ05(I) .. Y(I) =g= 0.5*B(I)*(W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*A(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"));

EQ06(I) .. Z(I) =g= 0.5*C(I);

EQ07(I,J)\$(ORD(I) NE ORD(J)) ..
 DX(I,J) =g= 0.5 * A(I) * (W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7"))
 + 0.5*B(I)*(W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8"))
 + 0.5*A(J)*(W(J,"1")+W(J,"3")+W(J,"5")+W(J,"7"))
 + 0.5*B(J)*(W(J,"2")+W(J,"4")+W(J,"6")+W(J,"8"))
 + (DH(I,J)+M(I,J)) * WX(I,J) - M(I,J);

EQ08(I,J)\$ (ORD(I) NE ORD(J)) ..

$$\begin{aligned} DY(I,J) = & g = 0.5 * B(I) * (W(I,"1")+W(I,"3")+W(I,"5")+W(I,"7")) \\ & + 0.5 * A(I) * (W(I,"2")+W(I,"4")+W(I,"6")+W(I,"8")) \\ & + 0.5 * B(J) * (W(J,"1")+W(J,"3")+W(J,"5")+W(J,"7")) \\ & + 0.5 * A(J) * (W(J,"2")+W(J,"4")+W(J,"6")+W(J,"8")) \\ & + (DH(I,J)+M(I,J)) * WY(I,J) - M(I,J); \end{aligned}$$

EQ09(I,J)\$ (ORD(I) NE ORD(J)) ..

$$DZ(I,J) = g = (DV(I,J) + 0.5 * C(I) + 0.5 * C(J)) * WZ(I,J) ;$$

EQ10(I,J)\$ (ORD(I) NE ORD(J)) ..

$$DX(I,J) = I = MM * (1 - WX(J,I));$$

EQ11(I,J)\$ (ORD(I) NE ORD(J)) ..

$$DY(I,J) = I = MM * (1 - WY(J,I));$$

EQ12(I,J)\$ (ORD(I) NE ORD(J)) ..

$$DZ(I,J) = I = MM * (1 - WZ(J,I));$$

EQ13(I,J)\$ (ORD(I) LT ORD(J)) .. $X(I) - X(J) - DX(I,J) + DX(J,I) = e = 0 ;$

EQ14(I,J)\$ (ORD(I) LT ORD(J)) .. $Y(I) - Y(J) - DY(I,J) + DY(J,I) = e = 0 ;$

EQ15(I,J)\$ (ORD(I) LT ORD(J)) .. $Z(I) - Z(J) - DZ(I,J) + DZ(J,I) = e = 0 ;$

EQ16(K) .. $LXP(K) - LXN(K) = e = \text{SUM}((I,J)\$PI(K,I,J), X(I) - X(J))$

$$\begin{aligned} & + \text{SUM}((P,I)\$KS(K,P,I), NPOSI(P,I,'FX') * 0.5 * A(I) * (W(I,"1") - W(I,"3") + W(I,"5") \\ & - W(I,"7")) + NPOSI(P,I,'FY') * 0.5 * B(I) * (-W(I,"2") + W(I,"4") - W(I,"6") + W(I,"8"))) \\ & - \text{SUM}((P,I)\$KT(K,P,I), NPOSI(P,I,'FX') * 0.5 * A(I) * (W(I,"1") - W(I,"3") + W(I,"5") \\ & - W(I,"7")) + NPOSI(P,I,'FY') * 0.5 * B(I) * (-W(I,"2") + W(I,"4") - W(I,"6") + W(I,"8"))) ; \end{aligned}$$

EQ17(K) .. $LYP(K) - LYN(K) = e = \text{SUM}((I,J)\$PI(K,I,J), Y(I) - Y(J))$

$$\begin{aligned} & + \text{SUM}((P,I)\$KS(K,P,I), NPOSI(P,I,'FY') * 0.5 * B(I) * (W(I,"1") - W(I,"3") - W(I,"5") \\ & + W(I,"7")) + NPOSI(P,I,'FX') * 0.5 * A(I) * (W(I,"2") - W(I,"4") - W(I,"6") + W(I,"8"))) \\ & - \text{SUM}((P,I)\$KT(K,P,I), NPOSI(P,I,'FY') * 0.5 * B(I) * (W(I,"1") - W(I,"3") - W(I,"5") \\ & + W(I,"7")) + NPOSI(P,I,'FX') * 0.5 * A(I) * (W(I,"2") - W(I,"4") - W(I,"6") + W(I,"8"))) ; \end{aligned}$$

EQ18(K) .. $LZP(K) - LZN(K) = e = \text{SUM}((I,J)\$PI(K,I,J), Z(I) - Z(J))$

$$\begin{aligned} & + (\text{SUM}((P,I)\$KS(K,P,I), NPOSI(P,I,'FZ') * 0.5 * C(I)) \\ & - \text{SUM}((P,I)\$KT(K,P,I), NPOSI(P,I,'FZ') * 0.5 * C(I))) ; \end{aligned}$$

EQ19 (I) .. $\text{SUM}(L, W(I,L)) = e = 1 ;$

EQ20(I,J)\$ (ORD(I) LT ORD(J)) ..

$$WX(I,J) + WX(J,I) + WY(I,J) + WY(J,I) + WZ(I,J) + WZ(J,I) = e = 1 ;$$

EQ21(I,J,IJ)\$ ((ORD(I) NE ORD(J)) AND (ORD(I) NE ORD (IJ)) AND (ORD(J) NE ORD(IJ))) ..

$$WX(I,J) = g = WX(I,IJ) + WX(IJ,J) - 1 ;$$

EQ22(I,J,IJ)\$((ORD(I) NE ORD(J)) AND (ORD(I) NE ORD (IJ)) AND (ORD(J) NE ORD(IJ))) ..

WY(I,J) =g= WY(I,IJ) + WY(IJ,J) - 1 ;

EQ23(I,J,IJ)\$((ORD(I) NE ORD(J)) AND (ORD(I) NE ORD (IJ)) AND (ORD(J) NE ORD(IJ))) ..

WZ(I,J) =g= WZ(I,IJ) + WZ(IJ,J) - 1 ;

MODEL LAYOUT /ALL/ ;

layout.optfile = 1;

OPTION MIP = CPLEX ;

option reslim = 1000000000;

option iterlim = 25000000;

option optcr = 0.0;

option work = 1000.0;

\$offsymxref

SOLVE LAYOUT USING MIP MINIMIZING T ;

DISPLAY XX.L, YY.L, ZZ.L, X.L, Y.L, Z.L, W.L, WX.L, WY.L, WZ.L ;

ANEXO 2

PLANTA DE POLIMERIZAÇÃO

Tabela 38. Dimensões e áreas dos componentes para a unidade de polimerização.

Componentes	A_i (m)	B_i (m)	C_i (m)	Área _i (m ²)
1	3,90	1,20	1,40	4,6800
2	4,95	1,60	2,05	7,9200
3	1,55	1,45	0,50	2,2475
4	3,15	3,15	4,10	9,9225
5	1,65	1,65	2,05	3,3000
6	0,75	1,80	0,75	1,3500
7	0,30	3,90	0,30	1,1700

Tabela 39. Dados para os tubos na otimização do layout da planta de polimerização.

Tubo k	$k_s(k)$	$k_t(k)$	C_k^p
1	4	7	11,601
2	5	8	11,601
3	6	9	11,601
4	10	15	11,601
5	11	14	11,601
6	12	13	11,601
7	16	21	6,186
8	17	20	6,186
9	18	19	6,186
10	35	24	46,014
11	23	29	46,014
12	31	32	6,186

Tabela 40 . Distâncias mínimas entre os componentes para a unidade de polimerização.

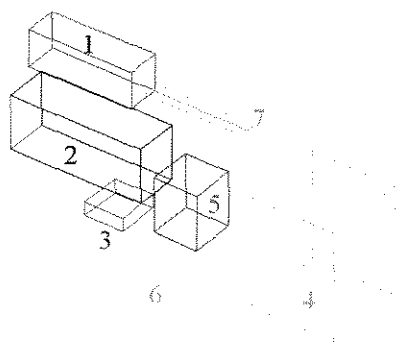
Componente i	Componente j	D_{ij}^h (m)	D_{ij}^v (m)
1	2	1,50	1,20
1	3	2,40	1,50
1	4	6,00	3,50
1	5	1,50	1,20
1	6	2,40	1,50
1	7	2,40	1,80
2	3	2,40	1,50
2	4	6,00	3,50
2	5	1,50	1,20
2	6	2,40	1,50
2	7	2,40	1,80
3	4	2,40	1,80
3	5	2,40	1,50
3	6	1,50	0,90
3	7	1,50	1,80
4	5	1,50	3,50
4	6	2,40	1,80
4	7	2,40	3,50
5	6	2,40	1,20
5	7	2,40	1,80
6	7	1,50	1,80

Tabela 41 . Posicionamento dos bocais para a unidade de polimerização,
componentes 1, 2 e 3.

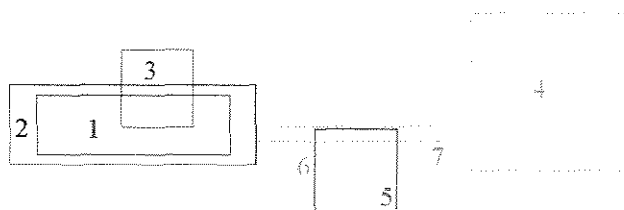
Componente <i>i</i>	P	F_{ji}^x	F_{ji}^y	F_{ji}^z
1	1	0,00	-0,50	1,00
	2	0,00	0,10	1,00
	3	0,00	0,70	1,00
	4	0,00	-0,60	-1,00
	5	0,00	0,00	-1,00
	6	0,00	0,60	-1,00
2	7	0,00	-0,60	1,00
	8	0,00	0,00	1,00
	9	0,00	0,60	1,00
	10	0,00	-0,60	-1,00
	11	0,00	0,00	-1,00
	12	0,00	0,60	-1,00
3	13	0,00	-1,00	0,50
	14	0,00	-1,00	0,00
	15	0,00	-1,00	-0,50
	16	0,00	1,00	0,50
	17	0,00	1,00	0,00
	18	0,00	1,00	-0,50

Tabela 42 . Posicionamento dos bocais para a unidade de polimerização,
componentes 4 a 7.

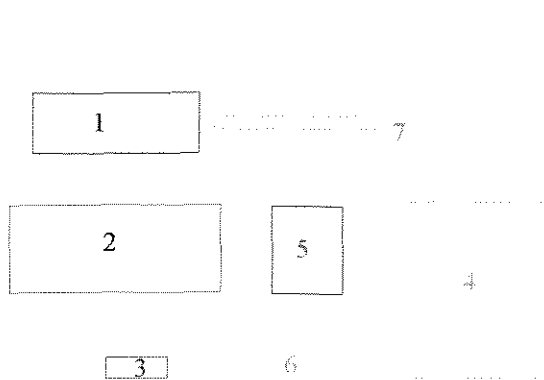
Componente <i>i</i>	P	F_{ji}^x	F_{ji}^y	F_{ji}^z
4	19	0,50	-0,50	1,00
	20	0,00	-0,50	1,00
	21	-0,50	-0,50	1,00
	22	0,00	0,70	1,00
	23	0,00	1,00	0,70
	24	0,00	-0,70	1,00
	25	0,00	-1,00	0,70
	26	0,00	-1,00	0,40
	27	0,00	-1,00	-0,40
	28	0,00	-1,00	-0,70
5	29	0,00	-0,70	1,00
	30	0,00	0,70	1,00
	31	0,00	0,00	-1,00
6	32	0,00	0,00	1,00
	33	0,00	0,70	-1,00
7	34	0,00	-1,00	0,00
	35	0,00	1,00	0,00
	36	0,00	-0,70	1,00
	37	0,00	0,70	-1,00



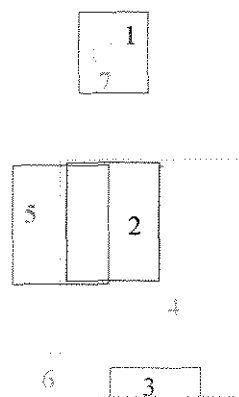
Vista Isométrica SE



Vista Superior

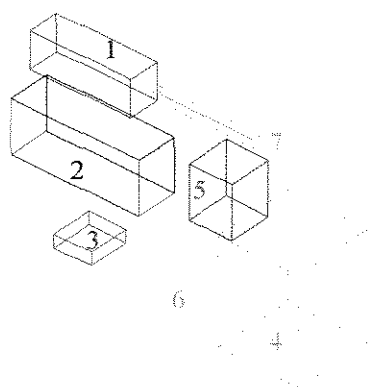


Vista Frontal

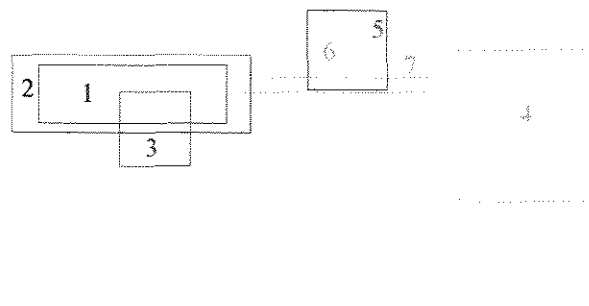


Vista Lateral

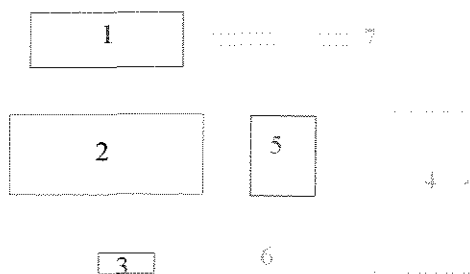
Figura 11. Planta de polimerização, modelo "original", empregando-se B&B - "Best Bound".
Componentes: (1) Tanques de Mistura, (2) Tanques de Armazenagem, (3) Bombas Dosadoras,
(4) Reator, (5) Tanque de Armazenagem, (6) Bomba, (7) Trocador de Calor Horizontal.



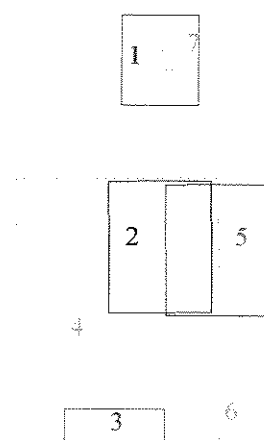
Vista Isométrica SE



Vista Superior

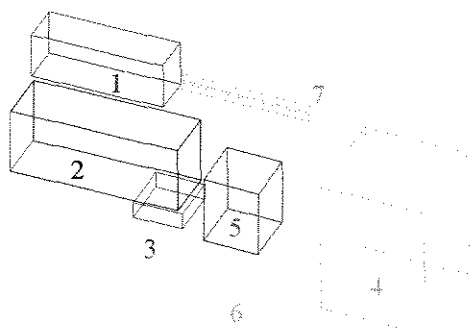


Vista Frontal

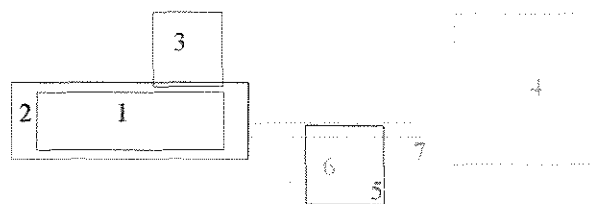


Vista Lateral

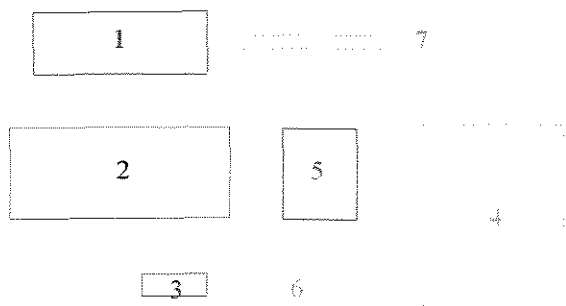
Figura12 . Planta de polimerização, modelo (2), empregando-se B&B -" Best Bound".
Componentes: (1) Tanques de Mistura, (2) Tanques de Armazenagem, (3) Bombas Dosadoras, (4) Reator, (5) Tanque de Armazenagem, (6) Bomba, (7) Trocador de Calor Horizontal.



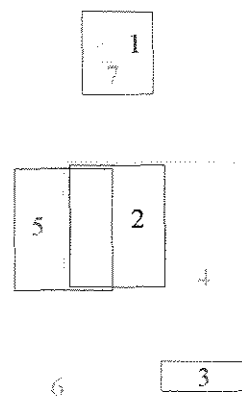
Vista Isométrica SE



Vista Superior



Vista Frontal



Vista Lateral

Figura13. Planta de polimerização, modelo (3), empregando-se B&B -" Best Bound".
Componentes: (1) Tanques de Mistura, (2) Tanques de Armazenagem, (3) Bombas Dosadoras,
(4) Reator, (5) Tanque de Armazenagem, (6) Bomba, (7) Trocador de Calor Horizontal.

ANEXO 3

PLANTA DE RECUPERAÇÃO DE MERCÚRIO A PARTIR DE EFLUENTES ELETROLÍTICOS

Tabela 43. Dimensões estimadas dos componentes para a planta de recuperação de mercúrio.

Componentes	A _i (m)	B _i (m)	C _i (m)	Área _i (m ²)
1	0,90	0,90	1,20	1,80
2	2,00	2,00	4,40	4,00
3	3,25	3,25	1,90	6,50
4	1,40	1,40	3,00	2,80
5	1,40	1,40	3,00	2,80
6	2,00	2,00	4,40	4,00
7	2,00	2,00	4,40	4,00
8	3,25	3,25	1,90	6,50
9	2,60	2,60	3,80	5,20

Tabela 44. Dados para os tubos na otimização do layout da planta de recuperação de mercúrio.

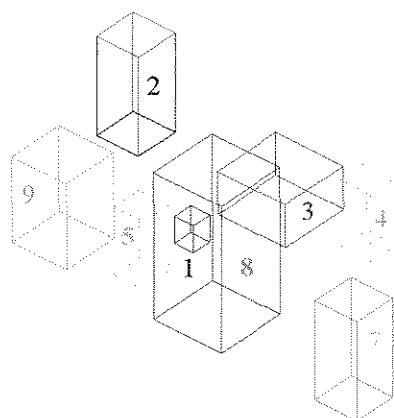
Tubo k	$k_s(k)$	$k_t(k)$	C_k^p
1	1	4	5,00
2	11	3	5,00
3	16	3	7,50
4	23	2	5,00
5	5	6	25,00
6	9	10	20,00
7	7	15	20,00
8	17	22	7,50
9	21	22	15,00
10	19	20	25,00
11	14	18	25,00
12	8	13	15,00
13	12	22	7,50

Tabela 45 . Posicionamento dos bocais para a planta de recuperação de mercúrio.

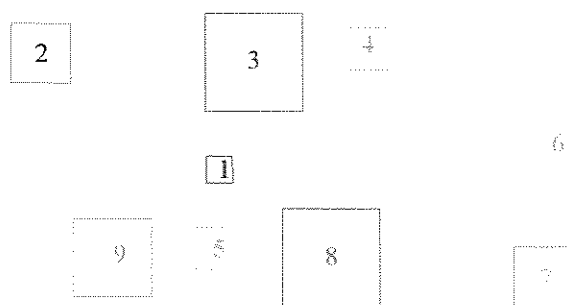
Componente i	P	F_{ji}^x	F_{ji}^y	F_{ji}^z
1	1	0,00	0,00	-1,00
	2	-0,50	0,25	1,00
	3	0,50	-1,00	0,65
2	4	0,50	-1,00	0,50
	5	0,00	0,00	-1,00
3	6	0,00	0,00	1,00
	7	0,50	-1,00	0,85
	8	0,00	0,00	-1,00
	9	-0,50	1,00	0,85
4	10	0,50	-1,00	-0,75
	11	0,00	0,00	1,00
	12	0,00	0,00	-1,00
6	13	0,00	-1,00	0,70
	14	0,00	0,00	-1,00
5	15	0,50	-1,00	-0,75
	16	0,00	0,00	1,00
	17	0,00	0,00	-1,00
7	18	0,50	0,65	1,00
	19	0,00	0,00	-1,00
8	20	0,00	0,00	1,00
	21	0,00	0,00	-1,00
9	22	0,00	0,00	1,00
	23	0,00	-0,10	-1,00

Tabela 46. Distâncias mínimas entre os componentes para a planta de recuperação de mercúrio.

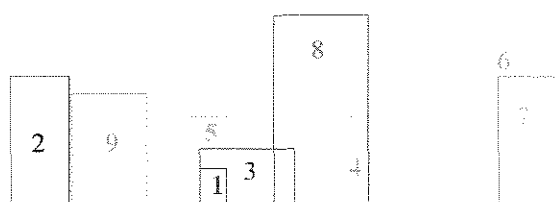
Componentes	Componente j	D_{ij}^h (m)	D_{ij}^v (m)
Misturador	2	4,50	2,40
	3	1,50	1,50
	4	1,50	1,50
	5	1,50	1,50
	6	4,50	2,40
	7	4,50	2,40
	8	1,50	1,50
	9	1,50	1,50
Reatores	3	4,50	2,40
	4	4,50	2,40
	5	4,50	2,40
	6	2,40	1,50
	7	2,40	1,50
	8	4,50	2,40
	9	4,50	2,40
Decantadores	4	1,50	1,50
	5	1,50	1,50
	6	4,50	2,40
	7	4,50	2,40
	8	1,50	1,50
	9	2,40	1,50
Filtros	5	1,50	1,50
	6	4,50	2,40
	7	4,50	2,40
	8	2,40	1,50
	9	1,50	1,50



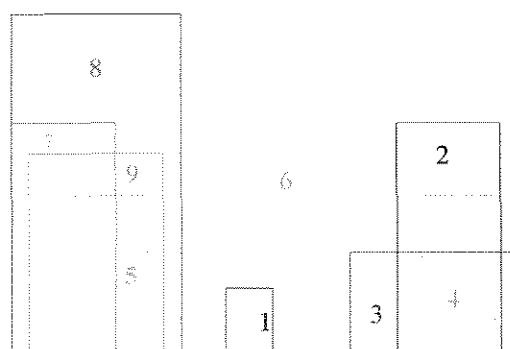
Vista Isométrica SE



Vista Superior



Vista Frontal



Vista Lateral

Figura 14. Planta de recuperação de mercúrio, modelos (1, 2 e 3).
Componentes: (1) Misturador, (2) Reator, (3) Decantador, (4) Filtro, (5) Filtro, (6) Reator, (7) Reator, (8) Decantador, (9) Filtro.

ANEXO 4

GAMS

(Descrição dos comandos principais)

** Descrição dos comandos principais usados em GAMS(BROOKE et al., 1988; TIN-LOI, 1995):*

- **SETS** : - correspondem exatamente aos índices nas representações algébricas dos modelos, aos quais podem ser atribuídos números ou nomes.

Por exemplo:

SETS

N	nomes dos componentes	/destil,tqref,referv,cond12,bbr12,bbov12/	
I	números dos componentes	/ 1 , 2 , 3 , 4 , 5 , 6 /	
J	números dos componentes	/ 1 * 6 / ;	
↓	↓	↓ ↓	
declaração	texto explicativo	atribuição	finalização do "SET"
	(opcional)	(se for uma sequência,	
		I pode ser escrito como J)	

Outro comando muito útil é o *ALIAS* que permite definir outro índice, cujos elementos são iguais ao índice já declarado. Por exemplo, o índice *I* é exatamente igual ao índice *J*:

```
SET I  números dos componentes / 1 * 6 / ;
ALIAS ( I , J ) ;
```

Além deste comando, as funções de transferência *CARD* e *ORD* podem ser usadas com os *sets*. *CARD* retorna o número de elementos em um *set*, enquanto que *ORD*, a posição relativa do elemento.

- **DADOS**

A entrada de dados pode ser feita na forma de :

- listas (parâmetros e escalares);

- tabelas;
- atribuições diretas.

Os parâmetros são definidos de forma semelhante aos índices (*sets*), sendo compostos por constantes e podendo ser um escalar, um vetor ou uma matriz de duas ou mais dimensões. No entanto, os escalares são empregados para elementos simples.

As tabelas são usadas para atribuição das estruturas de dados bidimensionais ou de dimensões superiores.

As atribuições diretas diferem das listas e tabelas, pois apresentam as expressões e cálculos separadamente.

Por exemplo:

$$\text{AREA(I)} = \text{A(I)} * \text{B(I)} ;$$

$$\text{CSA(I,K)} = \text{AREA(I)} * 62.8765 ;$$

- **VARIÁVEIS** : - usadas para declarar as variáveis de decisão para o modelo. Podem ser de cinco tipos diferentes, sendo que um tipo (*FREE*) é atribuído somente quando este não for especificado. Os tipos de variáveis e os intervalos compreendidos para cada um são apresentados na Tabela 47.

Tabela 47. Tipos de variáveis e seus respectivos intervalos.

Nome do Tipo de Variável	Intervalo permitido
Livre (<i>Free</i>)	$-\infty$ a $+\infty$
Positiva (<i>Positive</i>)	0 a $+\infty$
Negativa (<i>Negative</i>)	$-\infty$ a 0
Binária (<i>Binary</i>)	0 ou 1
Inteira (<i>Integer</i>)	0,1, . . . , 100

Fonte: BROOKE et al., 1988.

Embora os valores não possam ser inicializados na declaração das variáveis, campos adicionais podem ser usados para especificar os limites inferiores (.LO) e limites superiores (.UP), entre outros.

• **EQUAÇÕES**

As equações e desigualdades formam a essência dos modelos de otimização.

Em GAMS, coloca-se, sequencialmente, o nome da equação a ser definida, o domínio, ponto - ponto e a expressão relacionada com um operador (menor ou igual " $=L$ ", igual " $=E$ ", ou ainda, maior ou igual " $=G$ ").

A equação indexada gera, automaticamente, um grande número de equações de restrição. Logo, é necessário muita atenção na declaração, para evitar a geração desnecessária destas equações que, conseqüentemente, torna o programa mais lento em sua resolução, além de ocupar mais memória no computador.

Por exemplo:

EQUATION

EQ01(I) cálculo da dimensão x do "layout" ;

EQ01(I) .. $XX =G= X(I) + 0.5 * A(I) * (W(I, "1") + W(I, "3") + W(I, "5") + W(I, "7"))$
 $+ 0.5 * B(I) * (W(I, "2") + W(I, "4") + W(I, "6") + W(I, "8"))$;

Primeiramente, foi realizada a declaração da equação e, logo em seguida, a sua definição. No exemplo acima, a equação é indexada apenas por I , logo, o número de equações de restrição a ser gerado é exatamente igual ao número de elementos declarados no " $SET I$ ".

• **FUNÇÃO OBJETIVO**

Não há comandos específicos para declarar a função objetivo em GAMS, no entanto, convém que esta seja a primeira equação a ser definida, sua variável seja livre (sem restrições de tipo), sem domínio (índices) e, ainda, a equação seja uma igualdade.

- **MODELO E RESOLUÇÃO**

Para identificar e resolver o conjunto de equações, após a sua declaração e definição, emprega-se a palavra *model* precedido do nome do modelo e, entre barras, os nomes das equações ou simplesmente *all*, que inclui todas de uma só vez.

Por exemplo:

MODEL layout /ALL/ ;

ou

MODEL layout / OBJ, SUP, EQ01, EQ02, EQ03 / ;

Para chamar o *solver*, bem como para explicitar o tipo de otimização desejada, deve-se observar o seguinte procedimento:

- 1) Empregar a palavra *SOLVE*
- 2) Escrever o nome do modelo a ser resolvido
- 3) Escrever *USING*
- 4) Escolher um procedimento de solução, isto é:
 - *LP* para programação linear
 - *NLP* para programação não linear
 - *MIP* para programação inteira mista
 - *RMIP* para programação inteira mista relaxada
- 5) Escrever *MINIMIZING* ou *MAXIMIZING* para minimizar ou maximizar a função objetivo e o nome da variável a ser otimizada.

Por exemplo:

SOLVE LAYOUT USING MIP MINIMIZING T ;

Onde T é a variável livre que determina a função objetivo.

Existe ainda a possibilidade de se escolher o *solver* através do comando *OPTION*.

Assim,

MODEL LAYOUT /ALL/ ;

OPTION MIP = CPLEX ;

SOLVE LAYOUT USING MIP MINIMIZING T ;

- **SAÍDA DOS RESULTADOS**

Para visualizar as variáveis, parâmetros ou escalares desejados, basta empregar o comando *DISPLAY*.

Ex.: **DISPLAY XX.L, YY.L, ZZ.L ;**

- **EXECUÇÃO**

Para executar os programas em GAMS, basta digitar a palavra *gams* precedida do nome do arquivo.

Ex.: **gams layout**

- **SOLVERS**

Existem diversos *solvers* para resolver os problemas em MILP, porém, neste trabalho é empregado o *solver* *CPLEX*, que utiliza o algoritmo *Branch and Bound* (B&B).