

FREDERICO RODRIGUES BORGES DA CRUZ

UM ALGORITMO PARA PROJETO DE REDES HIERÁRQUICAS

DISSERTAÇÃO DE MESTRADO

UNIVERSIDADE FEDERAL DE MINAS GERAIS  
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO  
BELO HORIZONTE

1991

Frederico Rodrigues Borges da Cruz  
Departamento de Ciência da Computação

## UM ALGORITMO PARA PROJETO DE REDES HIERÁRQUICAS

Dissertação apresentada ao Departamento de Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Instituto de Ciências Exatas  
Universidade Federal de Minas Gerais  
Belo Horizonte, agosto de 1991

Cruz, Frederico Rodrigues Borges da.

C957a Um algoritmo para projeto de redes hierárquicas  
[manuscrito] / Frederico Rodrigues Borges da Cruz – 1991.  
iii, 77 f. il.

Orientador: Henrique Pacca Loureiro Luna.

Dissertação (mestrado) - Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da Computação.

Referências: f.75-77

1. Computação – Teses. 2. Algoritmos de computador – Teses. 3. Redes de computadores – Planejamento -Teses. 4. Otimização combinatoria – Teses I. Luna, Henrique Pacca Loureiro II. Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da Computação. III. Título.

CDU 519.6\*62(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS

FOLHA DE APROVAÇÃO

UM ALGORITMO PARA PROJETO DE REDES HIERÁRQUICAS

FREDERICO RODRIGUES BORGES DA CRUZ

Dissertação defendida e aprovada pela banca examinadora  
constituída pelos Senhores:

Prof. HENRIQUE PACCA LOUREIRO LUNA - Orientador

Prof. LUIZ AURELIO RAGGI

Prof. ROBERTO DIEGUEZ GALVÃO

Prof. MARCIO LUIZ BUNTE DE CARVALHO

Belo Horizonte, 06 de agosto de 1991.

## SUMÁRIO

Resumo .....	ii
Abstract .....	iii
Introdução .....	1
I - Da Rede .....	2
1.1 - Descrição de Um Sistema Telefônico .....	2
1.2 - Definição do Problema Objeto .....	6
II - Do Algoritmo .....	10
2.1 - Algoritmo Geral .....	10
2.2 - Resolução do Problema P-Mediana com Custos Fixos...	20
2.3 - Resolução do Problema de Programação Inteira Mista .	27
III - Da Implementação .....	35
3.1 - Diagrama de Fluxo de Dados .....	35
3.2 - Dicionário de Dados .....	38
3.3 - Diagramas Hierárquicos .....	45
IV - Dos Testes e Resultados .....	56
4.1 - Algoritmo para Resolução do Problema PMCF .....	56
4.2 - Algoritmo para Projeto das Sub-redes .....	60
4.3 - Algoritmo Proposto .....	63
Conclusões .....	74
Referências Bibliográficas .....	75

## RESUMO

A rede de alimentação de uma central telefônica é modelada como uma rede hierárquica. Nesse tipo de rede, um nodo central (central telefônica) distribui fluxos a nodos terminais com demanda (armários de distribuição) através de um conjunto de concentradores de demanda (estágios de linha remotos). É apresentado um algoritmo em três fases para o projeto da rede hierárquica. A primeira fase consiste em localizar os concentradores através do problema das  $p$ -medianas com custos fixos. A cada concentrador corresponde uma rede para a qual uma boa topologia é determinada por meio de um problema de programação inteira mista resolvido através de um procedimento de adição e substituição de arcos. Na terceira fase, usando-se um modelo similar, faz-se a conexão dos concentradores ao nodo central. Um sistema é implementado para testes do algoritmo proposto. É apresentada a documentação do sistema por meio de diagramas de fluxos de dados, dicionário de dados e diagramas hierárquicos dos módulos. Os resultados obtidos com o algoritmo proposto indicam que boas soluções para a rede hierárquica são determinadas, embora nem sempre as melhores. Redes de porte considerável são resolvidas (200 nodos) e os resultados indicam que o tempo de processamento cresce rapidamente com o tamanho do problema.

## ABSTRACT

The primary network of a telephone switching centre is presented as a hierarchical network. In such network, a root node (switching centre) distributes flows to terminal nodes (distribution boxes) via a set of demand concentrators. A 3-phase hierarchical network design algorithm is presented. First, the fixed-charge  $p$ -median problem formulation is used to locate the concentrators. For each concentrator's network, a good topology is obtained by solving a mixed integer programming problem through addition and interchange of arcs. The third phase uses a similar model to connect the concentrators to the root node. The proposed algorithm is implemented. It is presented the data flow diagrams, data dictionary and module hierarchical diagrams of the implemented system. The 3-phase algorithm results show that good solutions can be obtained for the hierarchical network, in spite of not being the best. Large networks are solved (200 nodes) and the solutions show that the CPU time grows quickly with the problem size.

## INTRODUÇÃO

Os sistemas de telecomunicações alcançaram um alto nível de complexidade. Dessa forma, qualquer recurso financeiro destinado a investimentos deve ser bem administrado pelas empresas responsáveis. A expansão do sistema para que o atendimento à crescente demanda do mercado seja feito pelo menor custo possível exige cuidadoso planejamento. Há, então, uma forte motivação para a procura de sistemas de apoio à decisão, integrando ferramentas computacionais e técnicas de otimização, que direcionem os investimentos necessários [Luna et al., 1988].

Será proposto um algoritmo para projeto de redes hierárquicas que são derivadas de um sistema telefônico. Por essa razão, um sistema genérico será apresentado na primeira parte desse trabalho, ainda que de forma resumida. A rede hierárquica objeto do estudo, um dos sub-problemas associados aos sistemas telefônicos, será então retirada do sistema completo.

Uma vez estabelecido o problema, tratar-se-á da estratégia para sua resolução, e isso será visto na segunda parte. Será mantida a já clássica filosofia de *dividir para conquistar*. O próprio problema objeto resulta da aplicação do princípio *dividir para conquistar* ao problema de otimização de um sistema telefônico completo, muito complexo, só tratável via sub-problemas associados.

Estando bem definidos esses dois pontos - o problema e como resolvê-lo - a etapa seguinte será a implementação computacional do algoritmo proposto para resolução. A terceira parte será dedicada à documentação do sistema implementado, apresentando-se nela os diagramas de fluxos de dados, dicionário de dados e diagramas hierárquicos dos módulos.

O sistema que implementa o algoritmo proposto será cuidadosamente testado, módulo por módulo. A quarta parte mostrará resultados dos testes dos módulos que implementam os principais sub-algoritmos, sendo feitas comparações com resultados da literatura. Mostrará, em seguida, resultados dos teste feitos com o algoritmo completo.

Finalmente, serão apresentadas as conclusões sobre a metodologia proposta.



## CAPÍTULO I - DA REDE

O planejamento de um sistema telefônico como um todo, pela explosão combinatória, inviabiliza qualquer tratamento matemático-computacional exato. Alguma coisa deve ser feita para se fugir dessa limitação, pois, conforme já dito, é muito importante a criação de um ferramental de apoio. O sistema deve ser descrito em diferentes níveis de detalhamento, para que seja possível fazer uma decomposição do problema global em sub-problemas tratáveis e que mantenham coerência entre si.

### 1.1 - Descrição de Um Sistema Telefônico

Uma boa descrição e decomposição de um sistema telefônico - bem como uma especificação formal da integração entre os níveis de análise - é bem conhecida [Luna & Mateus, 1989], e segue a já clássica estratégia de *dividir para conquistar*. A fig. 1.1 ilustra a descrição da estrutura em hierarquias de um sistema telefônico, realçando os principais elementos que compoem cada sub-sistema e enfatizando os problemas de redes a resolver (elemento associado a cada ramo direito da árvore de montagem do sistema telefônico).

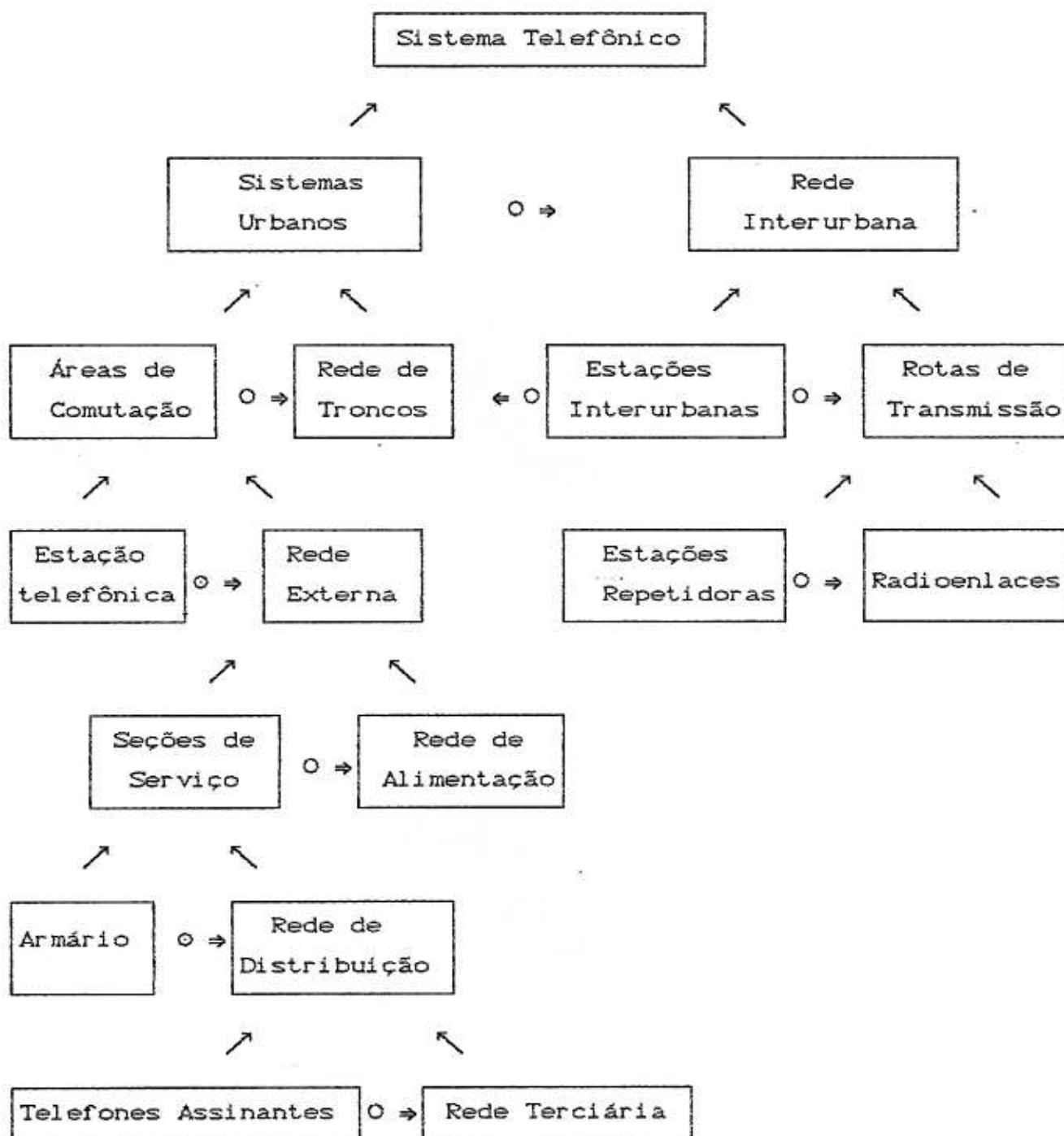


fig. 1.1 - redes de um sistema telefônico

As relações hierárquicas e de montagem são as seguinte [Luna & Mateus, 1989] :

1 - Um conjunto de *Sistemas Urbanos*, constituem um *Sistema telefônico*. Os *Sistemas Urbanos* são interligados através de uma *Rede Interurbana* (a notação "O ⇒" indica que os elementos associados ao ramo esquerdo da árvore de relações constituem os nós da rede explicitada no ramo direito da árvore).

2 - Cada *Sistema Urbano* é por sua vez constituído por uma ou mais *Áreas de Comutação*, ou zonas de filiação, por sua vez interligadas através de uma *Rede de Troncos*. No mesmo nível, a *Rede Interurbana* é composta por um conjunto de *Estações Interurbanas*, ligadas através de *Rotas de Transmissão*. Cada estação interurbana é também nó de uma rede de troncos urbana.

3 - Cada *Área de Comutação* abrange uma *Estação Telefônica* e uma *Rede Externa* (a notação "O ⇒" indica que o elemento associado ao ramo esquerdo é nó raiz da rede (árvore) explicitada à direita). Ainda no mesmo nível, mas no âmbito do sub-sistema interurbano, cada *Rota de Transmissão* é composta por *Estações Repetidoras*, ligadas por *Radioenlaces*.

4 - A *Rede Externa* cobre diversas *Seções de Serviço*, que são ligadas através da *Rede de Alimentação*, também chamada de rede primária.

5 - Cada *Seção de Serviço* possui um *Armário*, ou ponto de controle, que é nó raiz da *Rede de Distribuição*, ou rede secundária.

6 - E, finalmente, a *Rede de Distribuição*, geralmente aérea, constitui o elo de ligação dos *Telefones Assinantes*, eventualmente através de uma *Rede Terciária*.

A clássica fig. 1.2, [Tavares et al., 1981], complementa a visão geral dos problemas de planejamento, esquematizando uma rede de um sistema telefônico urbano.

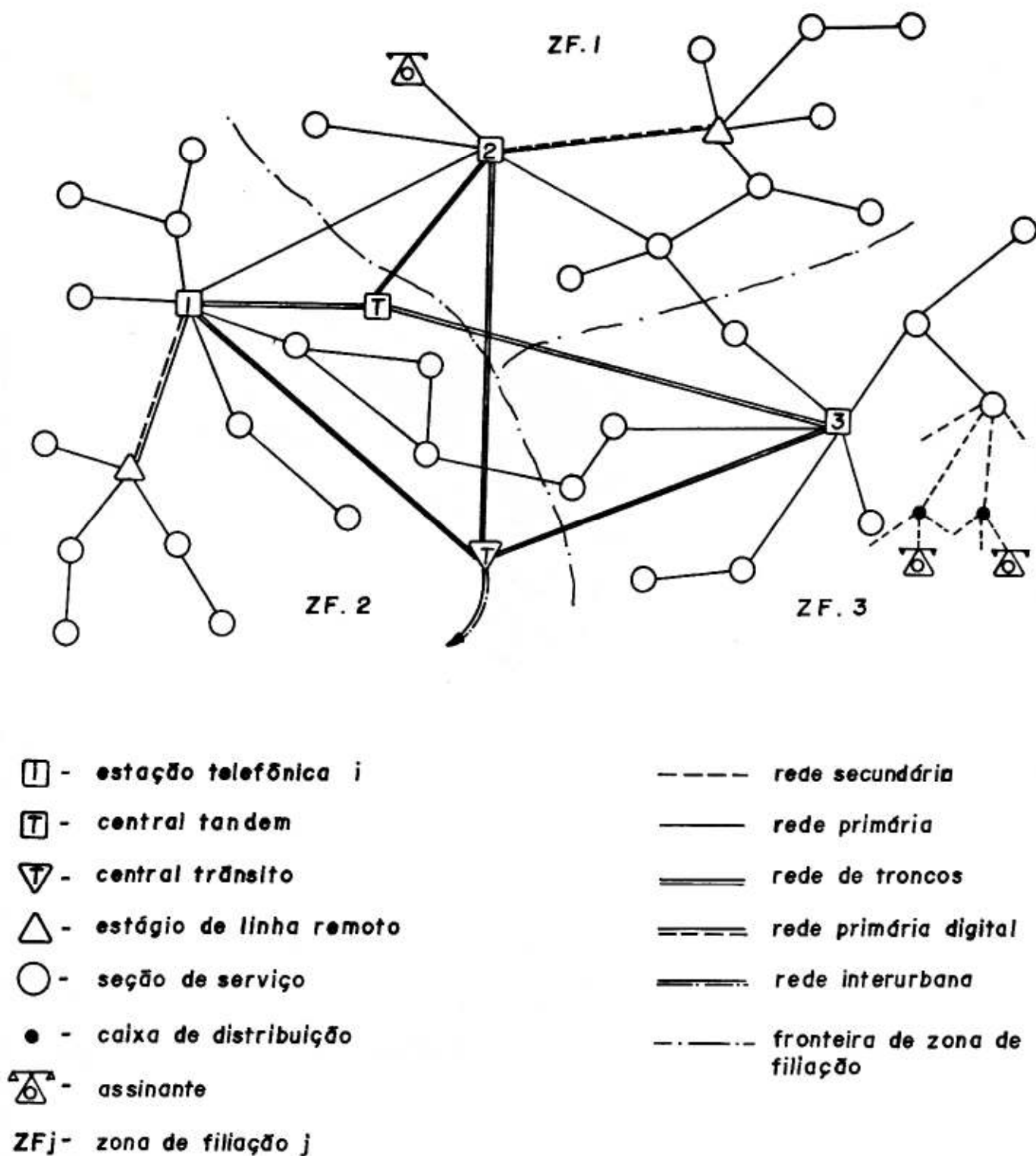


fig. 1.2 - redes de um sistema telefônico urbano

## 1.2 - Definição do Problema Objeto

Uma vez descrito o sistema, passa-se à determinação do problema a ser estudado. Ele focaliza apenas a área de abrangência de uma central telefônica. A fig. 1.3 ilustra uma rede de alimentação (ou rede primária) relativa a uma central telefônica.

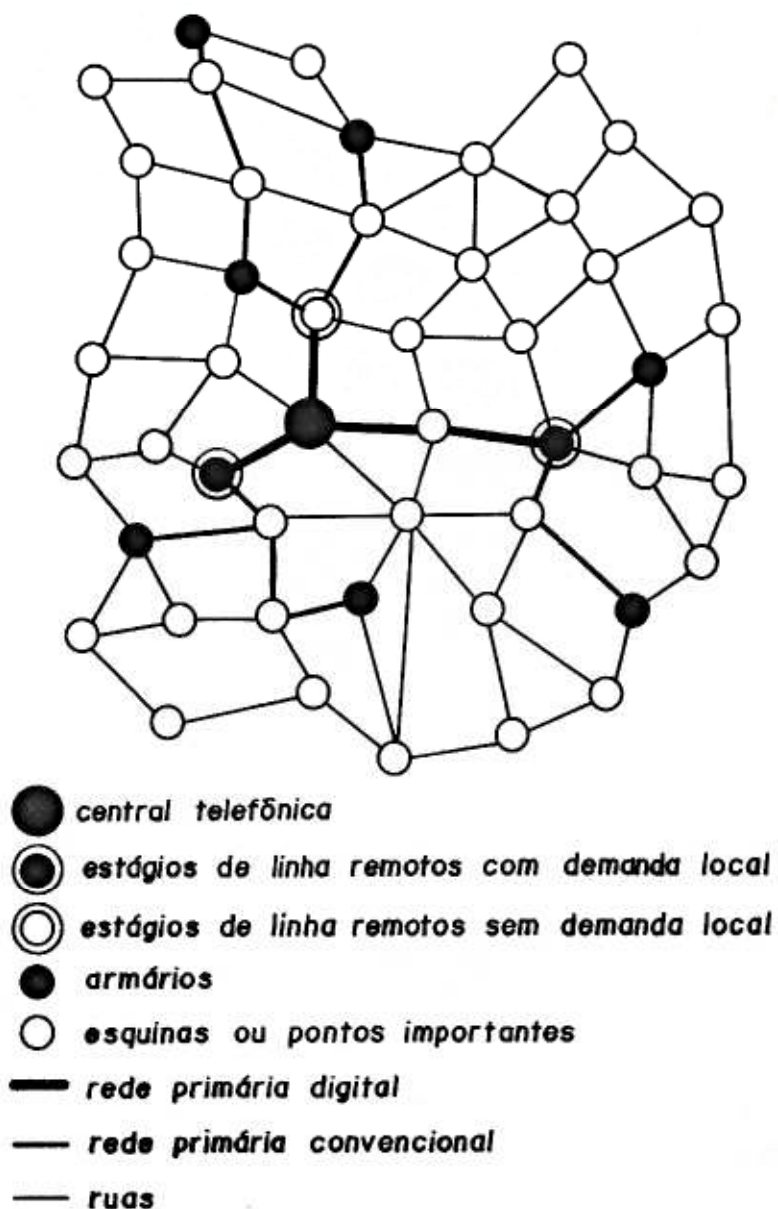


fig. 1.3 - rede de alimentação de uma central

O problema de otimização da rede de alimentação, ou rede

primária, já foi bastante explorado na literatura [Ziviani et al., 1983; Luna, 1984; Cabral, 1985; Luna et al., 1987] e não haveria razão para fazê-lo aqui novamente. Entretanto, o problema aqui proposto possui uma variante até então não incorporada. Em todas as abordagens anteriores, a rede de alimentação a ser projetada não possuía estágios de linha remotos. A sua presença dá o caráter hierárquico à rede. Do centro distribuidor de fluxos (central telefônica) aos concentradores de demanda (estágios de linha remotos) tem-se um nível hierárquico. De tais concentradores aos nodos terminais (armários de distribuição), tem-se outro nível hierárquico, figuras 1.3 e 1.4.

Quanto aos estágios de linha remotos, o melhor procedimento para sua localização seria mesmo baseado nos modelos de localização capacitada, justamente conforme feito para a localização das centrais telefônicas.

Os modelos de localização não-capacitada, como o nome indica, não levam em consideração qualquer capacidade, seja dos nós, seja dos arcos, mas apenas os custos fixos das facilidades, os custos de atendimento e as demandas dos assinantes. As estruturas de custos a serem utilizadas são, portanto, extremamente rígidas. Por isso, o tratamento de rede existente fica mais complicado, dificultando a separação entre meios em uso, ociosos ou limites de expansão ou de instalação.

Apesar disso, os estágios de linha remotos serão posicionados por um modelo de localização não-capacitada. Tais modelos apresentam um nível de complexidade inferior aos modelos capacitados, possibilitando o tratamento de problemas de dimensões maiores. As capacidades dos estágios de linha remotos não são limitadas por esse modelo, mas considerando que essas capacidades tendem a se tornar tão grandes quanto se queira, esse fato se torna pouco significativo, mesmo que na prática o sistema ainda não tenha comportamento liberado [Mateus & Luna, 1987].

Os estágios de linha remotos serão alocados entre nós que se encontram dentro da área de abrangência de uma única central, levando o problema de localização a ser tratado conjuntamente com o problema de otimização da topologia da rede de alimentação.

Para cada concentrador (estágio de linha remoto) há uma sub-rede associada, chamada local, que faz sua conexão aos pontos de

demanda (armários de distribuição) afiliados. Há também uma sub-rede, chamada de interligação, que liga todos esses concentradores ao centro distribuidor de fluxos (central telefônica).

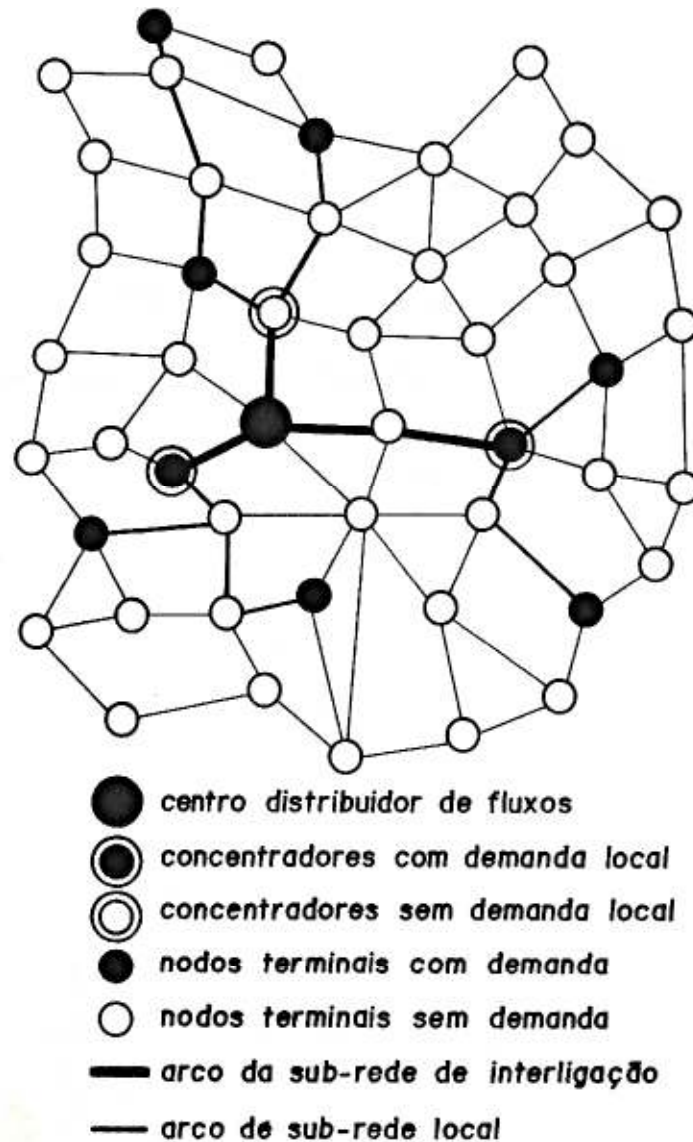


fig. 1.4 - rede em dois níveis hierárquicos

É de observação imediata que a otimização de cada uma das sub-redes que compõem a rede hierárquica (as sub-redes locais e a sub-rede de interligação) recai no antigo problema de otimização de

uma rede de alimentação (ou rede primária) sem estágios de linha remotos. Apesar de uma metodologia eficiente para obtenção de um boa solução desse problema ser conhecida [Ziviani et al., 1983; Luna, 1984; Cabral, 1985; Luna et al., 1987], um novo método de resolução é proposto.

Resumindo, o problema que se pretende estudar é o de otimização da topologia de uma rede (rede primária de um sistema telefônico urbano dentro da área de abrangência de uma central telefônica), onde um centro distribuidor (central telefônica) distribui fluxos (pares de fios telefônicos) a nodos terminais (armários de distribuição), através de concentradores de demanda (estágios de linha remotos), fig. 1.4.



## CAPÍTULO II - DO ALGORITMO

### 2.1 - Algoritmo Geral

Propõe-se que o projeto da rede em dois níveis hierárquicos seja dividida em três fases, seguindo o algoritmo apresentado abaixo, fiel à estratégia de resolução *dividir para conquistar*.

algoritmo

*localizar concentradores de demanda*

*otimizar sub-redes locais*

*otimizar sub-rede de interligação*

fim algoritmo

#### Localização dos Concentradores de Demanda

Pressupõe-se que os concentradores de demanda tenham uma capacidade superior a que estarão sujeitos, não representando erro a utilização de um modelo de localização não capacitada para sua localização. Será usado o clássico problema das  $p$ -medianas com custos fixos de instalação.

O problema das  $p$ -medianas, com ou sem custos fixos de instalação possui um amplo tratamento na literatura [Järvinen et al., 1972; Cornuejols et al., 1977; Narula et al., 1977; Eilon & Galvão, 1978; Erlenkotter, 1978; Christofides & Beasley, 1982; Hanjoul & Peeters, 1985; Guignard, 1988; Said & França, 1988; Galvão & Raggi, 1989; Mateus & Carvalho, 1991], e não representa nenhum problema a determinação do algoritmo que melhor se enquadra à

situação.

O problema geral de localização não-capacitada (PLNC), P1, possui a seguinte formulação matemática, [Galvão & Raggi, 1989]:

$$v(P1) = \min \left\{ \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{j \in J} f_j y_j \right\} \quad (2.1)$$

Sujeito a

$$\sum_{j \in J} x_{ij} \geq 1, \quad i \in I, \quad (2.2)$$

$$\sum_{j \in J} y_j \leq p, \quad (2.3)$$

$$x_{ij} \leq y_j, \quad i \in I, j \in J, \quad (2.4)$$

$$x_{ij} \in \{0,1\}, \quad i \in I, j \in J, \quad (2.5)$$

$$y_j \in \{0,1\}, \quad j \in J, \quad (2.6)$$

— onde:

$I = \{1,2, \dots, m\}$  é o conjunto de clientes a serem atendidos;  
 $J = \{1,2, \dots, n\}$  é o conjunto de pontos candidatos a receber localização de facilidades;

$c_{ij}$  é o custo para atender o cliente  $i$  pela facilidade  $j$ ;

$f_j$  é o custo fixo para se abrir a facilidade no local  $j \in J$ ;

$p$  é o número máximo de facilidades na solução e

$x_{ij}$  e  $y_j$  são as variáveis de decisão.

A variável de decisão  $x_{ij} = 1$  significa que o cliente  $i$  é servido pela facilidade  $j$  ( $x_{ij} = 0$ , caso contrário). A variável  $y_j = 1$  significa que uma facilidade é aberta no local  $j \in J$  ( $y_j = 0$ , caso contrário).

Na formulação P1, as restrições (2.2) asseguram que todos os clientes sejam atendidos. A restrição (2.3) limita o número de facilidades na solução. As restrições (2.4) dizem que um cliente  $i$  pode ser servido através do local  $j \in J$  somente se uma facilidade for aberta em  $j$ . As restrições (2.5) e (2.6) definem a integralidade das variáveis de decisão. Deve-se notar que, em princípio, pode-se definir os  $x_{ij}$  como  $x_{ij} \geq 0$ ,  $i \in I$ ,  $j \in J$ . Na ausência de restrições

de capacidade, um cliente é sempre atendido completamente pela facilidade mais próxima.

As considerações pelas quais a formulação P1 pode reproduzir as formulações conhecidas do "simple plant location problem", do problema das p-medianas ou do problema das p-medianas com custos fixos são imediatas [Galvão & Raggi, 1989].

Se  $I \equiv J$  corresponder aos vértices de uma rede, os  $c_{ij}$  aos custos de ligação de nodo  $i$  à facilidade instalada em  $j$  e  $f_j$  ao custo fixo de instalação da facilidade em  $j$ , então a formulação P1 corresponde ao problema das p-medianas com custos fixos, modelo que melhor se enquadra à situação aqui proposta.

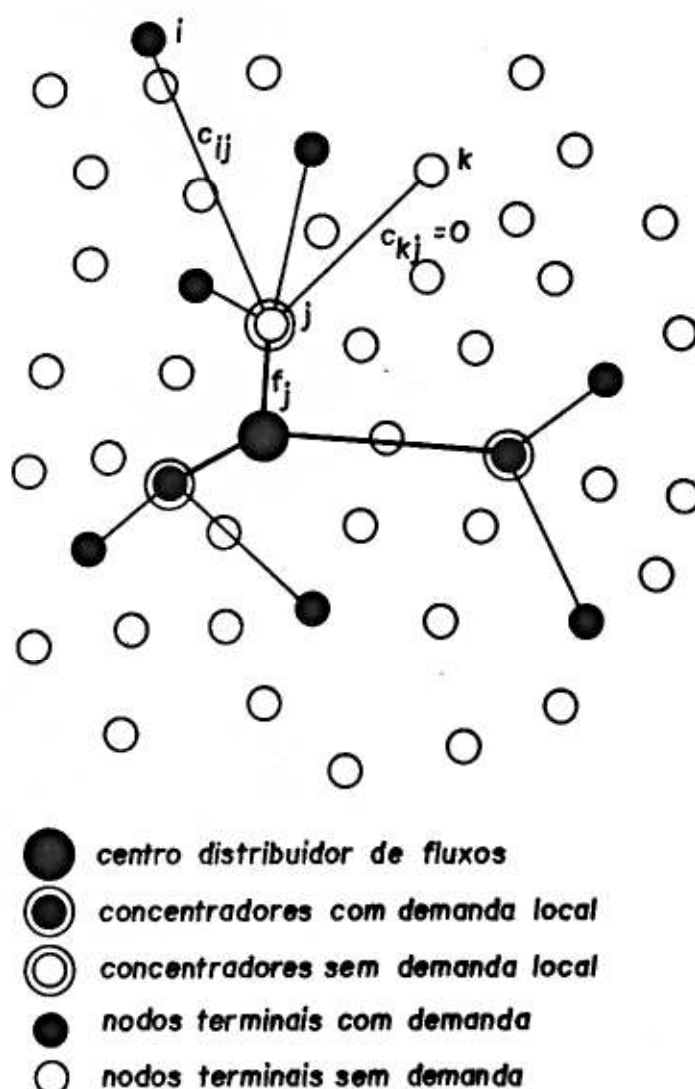


fig. 2.1 - localização dos concentradores de demanda

A solução do problema das p-medianas com custos fixos pode representar uma aproximação da rede de alimentação, desde que os custos fixos e de ligação sejam adequados. Para o conjunto ótimo de concentradores (medianas) possuir alguma relação com a rede de alimentação ótima, o problema das p-medianas com custos fixos deve ser armado com alguma habilidade.

Em primeiro lugar, algumas transformações devem ser feitas no grafo original da rede problema (por exemplo, fig. 1.4) para que o seja tratável pelo modelo das p-medianas com custos fixos. Nesse modelo, dois nós quaisquer devem ser diretamente ligados.

Os nós não diretamente conectados no grafo original serão ligados pela distância mínima. O cálculo das distâncias mínimas entre quaisquer dois nós não é uma operação problemática, pois pode ser feita em um tempo polinomial [Dijkstra, 1959].

A fig. 2.1 mostra a solução do problema das p-medianas com custos fixos para a rede ilustração da fig. 1.4. Por simples questão de clareza somente um nó branco (sem demanda) teve representada sua ligação à respectiva mediana, mas todos eles estão ligados a alguma.

Os custos fixos,  $f_j$ , são proporcionais à distância mínima entre o local de instalação da facilidade e o centro distribuidor de fluxos, mais um custo de instalação constante (custo do "hardware" do estágio de linha remoto, p.e.):

$$f_j = \alpha_{j,c} d_{j,c} + \text{cstins}_j, \quad (2.7)$$

onde:

- .  $f_j$  é o custo de instalação de uma facilidade no nó j,
- .  $\alpha_{j,c}$  é a constante de proporcionalidade para o arco (j,c),
- .  $d_{j,c}$  é a distância mínima entre o nó j e o nó central e
- .  $\text{cstins}_j$  é o custo fixo de instalação da facilidade no nó j.

A constante de proporcionalidade  $\alpha_{j,c}$  deve permitir a representação do custo total (fixo e variável) de instalação dos arcos que farão a conexão entre o concentrador j e a central

(pode-se considerar o caso de rede existente, colocando-se  $\alpha_{j,c} = 0$ , se já existe a ligação pronta entre  $j$  e  $c$ ):

$$\alpha_{j,c} = \text{intvar}_{j,c} \overline{\text{dem}} + \text{intfix}_{j,c}, \quad (2.8)$$

onde:

.  $\text{intvar}_{j,c}$  é o custo variável por unidade de distância de rede de interligação no arco  $(j,c)$ , i.e., custo por unidade de fluxo por unidade de distância naquele arco,

.  $\overline{\text{dem}}$  é a demanda média por concentrador e

.  $\text{intfix}_{j,c}$  é o custo fixo por unidade de distância de rede de interligação no arco  $(j,c)$ , i.e., o custo por unidade de distância naquele arco.

Por outro lado, os custos de ligação,  $c_{i,j}$ , são proporcionais à mínima distância entre a facilidade instalada em  $j$  e o ponto de demanda em  $i$ :

$$c_{i,j} = \alpha_{i,j} d_{i,j}, \quad (2.9)$$

onde:

.  $c_{i,j}$  é o custo de ligação do nó  $i$  à mediana  $j$ ,

.  $\alpha_{i,j}$  é a constante de proporcionalidade para o arco  $(i,j)$  e

.  $d_{i,j}$  é a distância mínima entre o nó  $i$  e o nó  $j$ .

A constante de proporcionalidade  $\alpha_{i,j}$  é nula caso o nó  $i$  não possua demanda. Não há nenhuma necessidade de se conectar nós sem demanda aos concentradores, mas a solução do problema das  $p$ -medianas sempre aloca todos eles à mediana mais próxima. Para que a solução seja uma aproximação da rede final, essa ligação, não deve afetar o valor da função objetivo do problema das  $p$ -medianas. Em outras palavras, não deve ter custo algum.

Por outro lado, se  $i$  é um nó com demanda, a constante de

proporcionalidade  $\alpha_{i,j}$  representa o custo de conexão entre ele e o concentrador de demanda instalado em  $j$  (pode-se considerar casos de rede existente fazendo  $\alpha_{i,j} = 0$ ):

$$\alpha_{i,j} = \text{locvar}_{i,j} \text{dem}_i + \text{locfix}_{i,j}, \quad (2.10)$$

onde:

.locvar<sub>i,j</sub> é o custo variável por unidade de distância de rede local no arco (i,j), i.e., custo por unidade de fluxo por unidade de distância naquele arco,

.dem<sub>i</sub> é a demanda no nodo  $i$  e

.locfix<sub>i,j</sub> é o custo fixo por unidade de distância de rede local no arco (i,j), i.e., custo por unidade de distância naquele arco.

Resumindo, é dado um artifício para se considerar os casos onde parte da rede é existente, lembrando-se, mais uma vez, que os modelos de localização não capacitada são limitados para tais considerações. As demandas em cada nó foram consideradas por uma ponderação nas distâncias mínimas, i.e., através das constantes  $\alpha_{i,j}$ , que trazem incorporadas informações de demanda, uma idéia que não é nova [Teitz & Bart, 1968]. Dessa forma, foram diferenciados os nós brancos, pontos sem demanda que não precisam ser conectados às medianas, e os nós pretos, pontos com demanda e que devem ser alocados à mediana mais próxima.

Não se pode deixar de ressaltar as simplificações impostas ao problema de projeto da rede quando se está alocando os concentradores de demanda. Não há nenhuma preocupação em se fazer as ligações entre os concentradores e os nodos terminais com demanda evitando-se o paralelismo de caminhos. As ligações são feitas pelos caminhos mínimos. O mesmo se aplica às ligações entre os concentradores e o centro distribuidor de fluxos. Não há, portanto, nenhuma garantia que a solução ótima do problema das  $p$ -medianas com custos fixos conduza à solução ótima do problema de projeto da rede hierárquica.

Ao final desse procedimento, tem-se as medianas (concentradores) alocadas e os nodos terminais com demanda, afiliados àquela mais próxima.

### Otimização das Sub-redes Locais

A fig. 2.2 ilustra as sub-redes associadas a cada concentrador de demanda, para a rede ilustração apresentada na fig. 1.4.

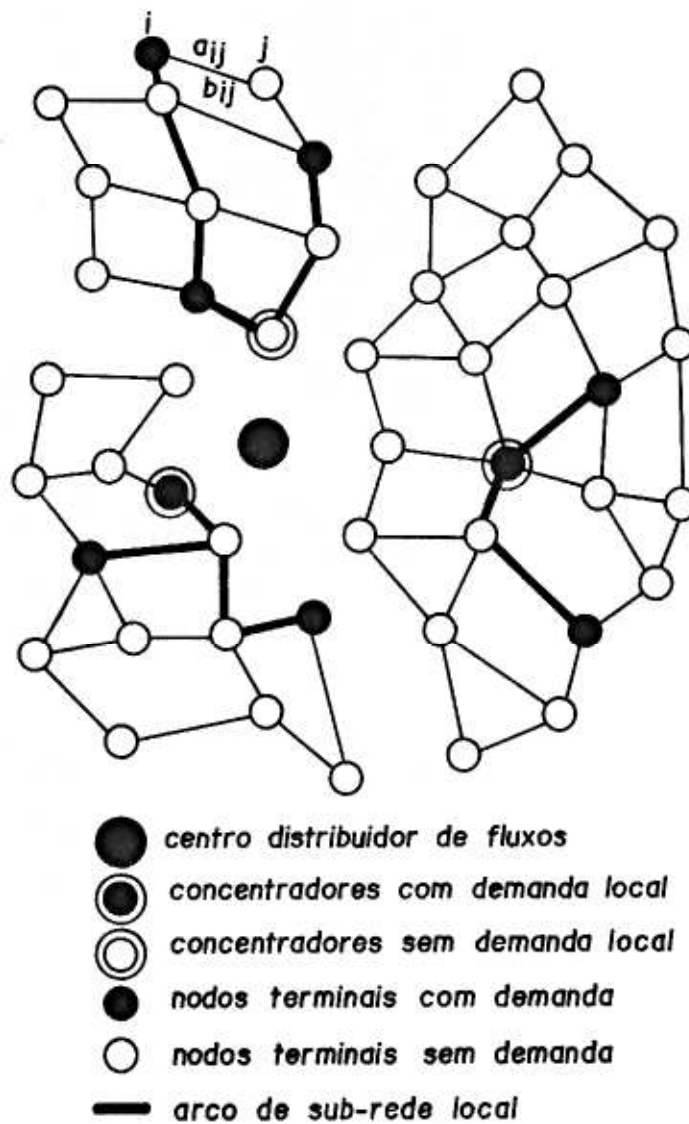


fig. 2.2 - otimização das sub-redes locais

Seja uma sub-rede associada a um concentrador de demanda um grafo conexo orientado  $G = (N, A)$ , onde  $N$  representa o conjunto de nodos e  $A$ , o conjunto de arcos. Há um nodo  $f$  origem e os outros nodos de  $N$  são particionados em dois sub-conjuntos disjuntos,  $P \subset N$  e  $B \subset N$ , onde  $P$  (pretos) é o conjunto de nodos com demanda e  $B$  (brancos) é o conjunto de nodos sem demanda.

O problema de otimização da topologia da sub-rede consiste na determinação de um sub-grafo de  $G$ ,  $G' = (N', A')$ , tal que:

- i-  $G'$  é uma árvore composta pelo nodo origem  $f$ , e todos os nodos pertencentes ao conjunto  $P$ ;
- ii- o custo da rede é mínimo;
- iii- a capacidade instalada nos arcos é suficiente para atender às demandas estipuladas para cada nodo pertencente ao conjunto  $P$ .

Conforme pode ser observado, o problema de otimização de uma sub-rede local é correspondente ao problema de otimização de uma rede de alimentação de uma central sem estágios de linha remotos, já conhecido na literatura [Luna et al., 1987; Mateus & Luna, 1989].

O modelo de programação inteira mista, P2, abaixo, sintetiza o problema de otimização das sub-redes locais [Mateus & Luna, 1989]:

$$v(P2) = \min \sum_{(i,j) \in A} (a_{ij} p_{ij} + b_{ij} y_{ij}) \quad (2.11)$$

Sujeito a

$$\sum_{(f,j) \in A} p_{fj} = \sum_{k \in P} S_k, \quad (2.12)$$

$$\sum_{(i,k) \in A} p_{ik} - \sum_{(k,j) \in A} p_{kj} = 0, \text{ para } \forall k \in B \quad (2.13)$$

$$\sum_{(i,k) \in A} p_{ik} - \sum_{(k,j) \in A} p_{kj} = S_k, \text{ para } \forall k \in P \quad (2.14)$$

$$p_{ij} \leq M y_{ij} \quad (2.15)$$

Os arcos  $(i,j) \in A$ , tais que  $y_{ij} = 1$ , formam uma árvore de  $G$ ,

$$(2.16)$$



onde :

$a_{ij}$  é o custo por unidade de fluxo através do arco  $(i,j)$ ,

$b_{ij}$  é o custo fixo para se passar fluxo através de  $(i,j)$ ,

$S_k$  é a demanda no nodo  $k$ ,

$p_{ij}$  é a demanda servida através do arco  $(i,j)$ ,

$P_{fj}$  se refere aos arcos que deixam o nodo origem  $f$ ,

$y_{ij}$  é a variável de decisão; se  $y_{ij} = 1$ , significa que o arco

$(i,j)$  foi escolhido para a topologia e  $y_{ij} = 0$ , caso contrário.

A função objetivo (2.11) minimiza o custo total resultante da soma, para cada arco, do custo do fluxo e do custo fixo de utilização desse arco.

A restrição (2.12) assegura que o fluxo total atendido pelos concentradores de demanda seja igual às demandas estipuladas pelos nodos terminais. As condições (2.13) exigem que o fluxo que chega aos nodos intermediários (sem demanda) seja igual àquele que sai deles. As condições (2.14) asseguram que a diferença entre o fluxo que chega e o que sai nos nodos que possuem demanda é igual à sua demanda. As restrições (2.13) e (2.14) garantem o princípio de conservação de fluxos.

As restrições (2.15) asseguram que a existência de fluxo através de um arco esteja condicionada à escolha desse arco na topologia, onde  $M$  é um número suficientemente grande. Finalmente, a restrição (2.16) exige que a solução seja um grafo conexo e sem ciclos.

É conhecido na literatura, [Luna et al. 1987], um problema similar, mais geral que a formulação P2, onde é definido o conceito de seções de serviço e, dentro dessas seções, podem ou não haver alternativas de localização do nodo com demanda. A resolução da formulação P2 pode ser feita por meio das heurísticas desenvolvidas para esse problema mais geral. O problema pode, também, ser solucionado pelas técnicas de relaxação lagrangeana, bastante

empregadas na resolução de problemas de programação inteira mista [Geoffrion, 1974; Fisher, 1981].

### Otimização da Sub-rede de Interligação

O problema de otimização da sub-rede que interliga os concentradores é essencialmente o mesmo de interligação dos pontos com demanda ao concentrador respectivo. A fig. 2.3 ilustra a sub-rede de interligação da rede ilustração apresentada na fig. 1.4.

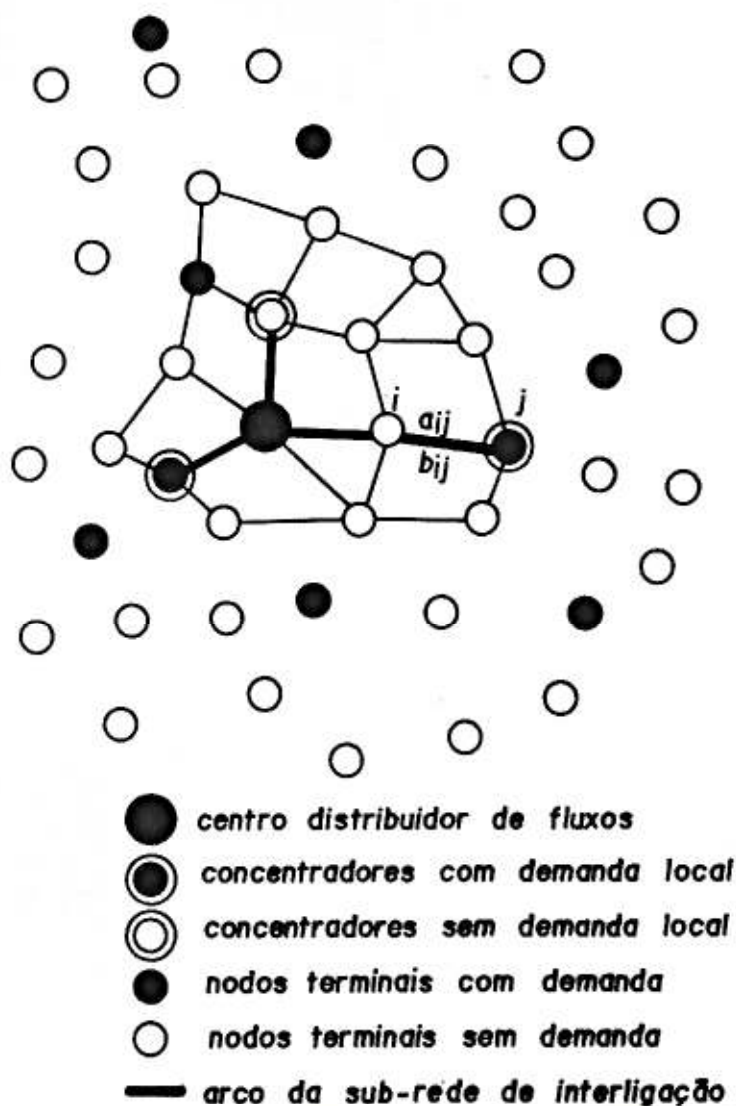


fig. 2.3 - otimização da sub-rede de interligação

Cada concentrador é análogo a um ponto com demanda e o centro distribuidor de fluxo, a um concentrador. O modelo de programação inteira mista permanece o mesmo, assim como o algoritmo para resolução.

## 2.2 - Resolução do Problema das P-medianas com Custos Fixos

Os problemas de localização não-capacitada receberam um amplo tratamento na literatura. Pelo lado do problema das p-medianas, vários foram os métodos de solução propostos. Métodos heurísticos os mais variados surgiram, como a substituição de vértices [Teitz & Bart, 1968; Galvão et al., 1986], substituição dupla de vértices [Eilon & Galvão, 1978], métodos baseados na relaxação lagrangeana [Narula et al., 1977; Said & França, 1988], relaxação com testes de redução [Christofides & Beasley, 1982; Mateus & Carvalho, 1991], métodos duais [Hanjoul & Peeters, 1985], para exemplificar alguns, sem pretensão de se fazer uma relação completa.

Pelo lado dos "*simple plant location problems*", o interesse não foi menor. Surgiram os métodos baseados em relaxação lagrangeana [Cornuejols et al., 1977], métodos duais [Erlenkotter, 1978], métodos duais ascendentes [Guignard, 1988] e etc.

Nos dois tipos de problemas, quando a solução ótima não era alcançada no nodo inicial da árvore de pesquisa, entravam em ação os algoritmos de separação e avaliação ("*branch and bound*").

Os esforços nessas duas frentes caminhavam independentemente, até ser proposta a formulação geral P1, a partir da qual se podia chegar à formulação do problema das p-medianas ou à do "*simple plant location problem*", como casos particulares. Para a formulação P1, foi proposto o seguinte algoritmo de resolução em três fases [Galvão & Raggi, 1989]:

algoritmo

```

aplicar algoritmo primal-dual
se SOLUCAO = "NAO_OTIMA" então
    aplicar algoritmo da relaxação lagrangeana
    se SOLUCAO = "NAO_OTIMA" então
        aplicar algoritmo de separação e avaliação
    fim se
fim se

```

fim algoritmo

Ao contrário do que se poderia imaginar, esse algoritmo, apesar de desenvolvido para a formulação geral P1, apresenta resultados competitivos mesmo quando comparados àqueles obtidos por meio de algoritmos específicos, seja para o "simple plant location problem" [Erlenkotter, 1978], seja para o problema das p-medianas [Boffey & Karkazis, 1987]. Além disso, esse era o único algoritmo, de todos os citados, que lidava com o problema das p-medianas com custos fixos sem necessidade de nenhuma adaptação e de estudos adicionais que uma modificação desse tipo pudesse exigir.

O algoritmo começa com uma etapa primal-dual. Se o ótimo não for alcançado, os limites gerados até então serão os limites iniciais da etapa seguinte de relaxação lagrangeana, economizando iterações de sub-gradiente. Caso a solução ótima não tenha sido atingida ainda, um algoritmo de separação e avaliação, a terceira etapa, usando os limites obtidos até aqui, é ativado. A seguir, será feita uma apresentação um pouco mais detalhada das etapas mencionadas.

**Algoritmo Primal-Dual**

A fase primal-dual segue o seguinte algoritmo [Galvão & Raggi, 1989]:

algoritmo

*aplicar heurística "gulosa" de adição de nós*  
repita  
     *criar problema dual*  
     *resolver problema dual*  
     se SOLUCAO\_DUAL > LIMITE\_INFERIOR então  
         LIMITE\_INFERIOR ← SOLUCAO\_DUAL  
         MESMO\_LIMITE\_INFERIOR ← falso  
         *criar conjunto de novos nós candidatos*  
         *aplicar heurística de substituição de nós*  
         *calcular valor da função objetivo do primal*  
         se SOLUCAO\_PRIMAL < LIMITE\_SUPERIOR então  
             LIMITE\_SUPERIOR ← SOLUCAO\_PRIMAL  
             MESMO\_LIMITE\_SUPERIOR ← falso  
         senão  
             MESMO\_LIMITE\_SUPERIOR ← verdadeiro  
         fim se  
     senão  
         MESMO\_LIMITE\_INFERIOR ← verdadeiro  
     fim se  
até ( MESMO\_LIMITE\_SUPERIOR ou MESMO\_LIMITE\_INFERIOR ou  
       ( LIMITE\_SUPERIOR - LIMITE\_INFERIOR ) <  $\epsilon$  ) )

fim algoritmo

Todos os procedimentos desse algoritmo já são explicitados em [Galvão & Raggi, 1989], embora os seguintes necessitem de leituras adicionais:

.a referência bibliográfica sobre a heurística "gulosa" de adição de nós original não foi localizada, optando-se por uma alternativa [Galvão et al., 1986];

.a resolução do problema dual segue um procedimento dual ascendente [Erlenkotter, 1978];

a heurística de substituição de nós empregou, entre as estratégias apresentadas em [Galvão et al., 1986], aquela que forneceu as melhores soluções.

O procedimento que cria o problema dual usa uma inicialização para as variáveis duais  $v_i$ , correspondente às restrições (2.2), que garante a viabilidade do dual. Para aqueles valores de  $v_i$ , o valor viável calculado para a variável dual  $r$ , correspondente à restrição (2.3), fica sempre nulo no caso do problema das p-medianas com custos fixos (PMCF). Os problemas duais criados em cada iteração são diferentes uns dos outros apenas pela variável  $F_j$ , que é igual a:

$$F_j = f_j + r, \quad \forall j \in J \quad (2.17)$$

onde:

$f_j$  é o custo fixo de instação da facilidade em  $j$  (eq. 2.7).

Sugere-se que o algoritmo primal-dual aplicado à resolução do problema das p-medianas com custos fixos (PMCF) seja simplificado em função dessa observação. Após a primeira passagem pelo laço repita até, o problema dual resolvido é sempre o mesmo. A simplificação resulta no seguinte algoritmo:

#### algoritmo

*aplicar heurística "gulosa" de adição de nós*  
*criar problema dual*  
*resolver problema dual*  
 LIMITE\_INFERIOR ← SOLUCAO\_DUAL  
*criar conjunto de novos nós candidatos*  
*aplicar heurística de substituição de nós*  
*calcular valor da função objetivo do primal*  
se SOLUCAO\_PRIMAL < LIMITE\_SUPERIOR então  
     LIMITE\_SUPERIOR ← SOLUCAO\_PRIMAL  
fim se

fim algoritmo

## Relaxação Lagrangeana

A técnica de relaxação lagrangeana se apresenta como um forte instrumento a ser utilizado na resolução de problemas de programação inteira mista [Geoffrion, 1974; Fisher, 1981]. Dentre as diversas técnicas usadas para maximizar o dual lagrangeano em otimização de grande porte, nota-se grande preferência pelo método de sub-gradientes, devido ao aspecto de não-diferenciabilidade da função objetivo do dual, em alguns pontos do domínio. A validação dessa técnica é conhecida há algum tempo [Held et al., 1974].

O algoritmo para resolução de P1, por relaxação lagrangeana, é:

### algoritmo

*inicializar valores necessários*

$T \leftarrow 0$

### repita

*montar vetor de sub-gradientes*

*calcular norma do vetor sub-gradiente*

se  $((\text{LIMITE\_SUPERIOR} - \text{LIMITE\_INFERIOR}) > \epsilon)$  e

$(\text{NORMA\_SUBGRADIENTE} > 0)$  então

*calcular tamanho do passo*

*determinar novo vetor de multiplicadores de Lagrange*

*atualizar limite inferior*

*atualizar limite superior*

$\text{MULTIPLICADORES\_VELHOS} = \text{falso}$

$T \leftarrow T + 1$

senão

$\text{MULTIPLICADORES\_VELHOS} = \text{verdadeiro}$

fim se

até  $\text{MULTIPLICADORES\_VELHOS}$  ou  $(T \geq \text{MAX\_IT})$

fim algoritmo

O algoritmo de relaxação, se necessário para determinação da solução ótima, começará exatamente do ponto onde parou o algoritmo

primal-dual. Os limites inferior e superior dessa etapa são usados como valores iniciais daquela. Os valores das variáveis duais  $v_i$  inicializam o vetor de multiplicadores de Lagrange,  $u_i$ .

Os passo intermediários são desenvolvidos em [Galvão & Raggi, 1989]. O procedimento para determinação do tamanho do passo é o único que talvez necessite de algum comentário adicional. É proposto que o seu cálculo siga a seguinte relação [Held et al., 1974]:

$$\theta_t = \alpha_t \frac{L_{\text{Sup}} - L(u^t)}{\|S^t\|}, \quad (2.18)$$

onde:

.  $L_{\text{Sup}}$  é um limitante superior para a solução ótima  $L(u^*)$ ,

.  $L(u^t)$  é a solução do dual para o vetor de multiplicadores de Lagrange corrente,  $u^t$ ,

.  $\|S^t\|$  é a norma euclideana do vetor sub-gradiente corrente,

.  $\alpha_t$  é um número real tal que  $\alpha_t \in (\epsilon, 2]$ ,  $\epsilon > 0$ .

Em [Galvão & Raggi, 1989] foi lançada a idéia da definição do valor de  $\alpha_t$  por meio de uma função contínua, embora tal função tivesse sido ali apresentada com um pequeno equívoco. Para se evitar as dificuldades computacionais provocadas por esse engano, pareceu lógico seguir a mesma filosofia e se usar a seguinte definição:

$$\alpha_t = 2e^{-t^2/n^2}, \text{ para } 0 \leq t \leq \text{MAX\_IT}. \quad (2.19)$$

Como não havia intenção de se implementar a etapa de separação e avaliação, resolveu-se prosseguir com as iterações de sub-gradiente por mais tempo que o sugerido, limitando MAX\_IT em  $2n$ , a menos que se seja dito o contrário.



Mais tarde, depois de feitos todos os teste computacionais apresentados no capítulo de testes e resultados, empregando-se a eq. 2.19, foi esclarecido, [Raggi, 1991], qual função contínua se pretendia, de fato, apresentar em [Galvão & Raggi, 1989] para se definir  $\alpha_t$ :

$$\alpha_t = \begin{cases} 2e^{-t^2/n^2} & , \text{ para } t \leq n/2, \\ 2e^{-4(t^4/n^4)} & , \text{ para } n/2 < t \leq n. \end{cases} \quad (2.20)$$

O número máximo de iterações, MAX\_IT, é limitado a  $n$ , onde  $n$  é o número de nós da rede.

Também ficou claro, [Raggi, 1991], que todos os resultados computacionais apresentados em [Galvão & Raggi, 1989] foram, na verdade, obtidos com  $\alpha_t$  definido segundo a eq. 2.20.

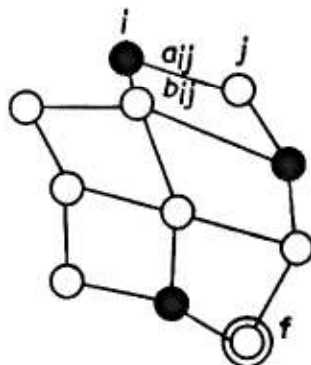
#### Algoritmo de Separação e Avaliação

Se um "gap" de dualidade é identificado ao final do algoritmo da relaxação lagrangeana, um algoritmo de separação e avaliação deve ser ativado para fechá-lo. Entretanto, entre os três problemas de localização não-capacitada, o de solução mais difícil parece ser mesmo o das  $p$ -medianas, único que exigiu a etapa de separação e avaliação nas redes aleatórias de [Galvão & Raggi, 1989].

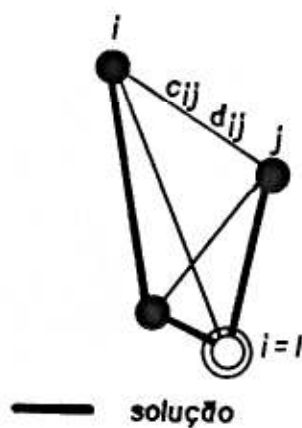
Como os "simple plant location problems" e os problemas das  $p$ -medianas com custos fixos foram resolvidos otimamente nas duas primeiras fases, nas redes aleatórias mencionadas, a terceira fase do algoritmo foi excluída da implementação.

### 2.3 - Resolução do Problema de Programação Inteira Mista

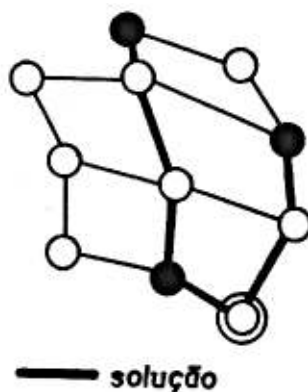
A fig. 2.4 mostra detalhadamente o processo de otimização de uma sub-rede do problema ilustrado na fig. 2.2.



a ) sub-rede original (local ou de interligação)



b ) sub-rede aproximada e a solução encontrada



c ) sub-rede original e a solução encontrada

### 2.4 - otimização de uma sub-rede

Um procedimento heurístico eficiente para resolução do modelo P2 é conhecido [Luna et al., 1987]. A técnica de relaxação lagrangeana, bastante empregada na resolução de problemas de programação inteira mista [Geoffrion, 1974; Fisher, 1981], poderia também ser tentada. Sugere-se o uso dessa última, mas não aplicada diretamente à formulação P2.

Considere-se a formulação P2 associada a uma sub-rede qualquer (local ou de interligação), fig. 2.4-a. Pode-se tentar uma relaxação como a da restrição (2.15). Nesse caso, o modelo P2, NP-difícil, [Mateus & Luna, 1989], torna-se separável em um problema de fluxos com custo mínimo (polinomial) de um lado e um problema de Steiner (NP-difícil) do outro. Mas o modelo P2 pode ser aproximado ao problema de fluxos com custos fixos (FCF), modelo P3, NP-difícil, [Hochbaum & Sergev, 1989]. O modelo P3 pode ser resolvido por relaxação lagrangeana, onde os sub-problemas gerados são de solução fácil - em tempo polinomial.

Em [Hochbaum & Sergev, 1989] são apresentadas duas relaxações diferentes para o modelo P3, onde os sub-problemas gerados, fáceis, são os seguintes:

- 1 - a primeira relaxação resulta no problema de fluxos com fonte única a custo mínimo e árvore geradora mínima;
- 2 - a segunda, no problema de árvore geradora mínima.

A aproximação é conseguida pela retirada dos nós de transbordo do problema original e pela geração do grafo completo associado aos nodos restantes, fig.2.4-b. Essa operação envolve a determinação da menor distância entre quaisquer dois nós com demanda e é polinomial [Dijkstra, 1959]. Além disso, já foi executada na fase de localização dos concentradores, podendo ser reaproveitada.

O problema de fluxos com custos fixos, modelo P3, resultado da aproximação, possui a seguinte formulação matemática [Hochbaum & Sergev, 1989]:

$$v(P3) = \min \left\{ \sum_{i=1}^n \sum_{j=2}^n c_{ij} x_{ij} + \sum_{i=1}^n \sum_{j=2}^n d_{ij} y_{ij} \right\} \quad (2.21)$$

Sujeito a

$$\sum_{p=1}^n x_{pj} - \sum_{q=2}^n x_{jq} = a_j, \quad j = 2, \dots, n, \quad (2.22)$$

$$\sum_{i=1}^n y_{ij} = 1, \quad j = 2, \dots, n, \quad (2.23)$$

$$Ay_{ij} \geq x_{ij}, \quad i = 1, \dots, n, \quad j = 2, \dots, n, \quad (2.24)$$

$$y_{ij} \in \{0,1\}, \quad x_{ij} \geq 0, \quad i = 1, \dots, n, \quad j = 2, \dots, n, \quad (2.25)$$

onde:

$n$  é o número de nós;

$a_j$  é a demanda do nó  $j$ ,  $j = 2, \dots, n$ ;

$c_{ij}$  é o custo de se mandar uma unidade de fluxo do nó  $i$  ao  $j$ ;

$d_{ij}$  é o custo fixo de se mandar fluxo do nó  $i$  ao  $j$ ;

$A = \sum_{j=2}^n a_j$  é o volume total de fluxo originário do nó raiz 1.

A função objetivo (2.21) é composta por duas partes. O termo da primeira considera o total dos custos dos fluxos e o da segunda, o total dos custos fixos associados à utilização dos arcos. As restrições (2.22) representam os requisitos de balanço de fluxo. As restrições (2.23) asseguram que exatamente um arco entra em cada nó (exceto o nó 1) e as restrições (2.24) garantem que se  $y_{ij} = 0$ , nenhum fluxo passa entre os nós  $i$  e  $j$ .

A fig. 2.4-b mostra a solução do problema de fluxos com custos fixos, modelo P3, e na fig. 2.4-c, essa solução é expressa em termos da rede original.

O problema de programação inteira mista, modelo P2, que define o projeto de cada uma das sub-redes associadas ao problema de projeto da rede hierárquica, deverá ser resolvido pelo seguinte algoritmo:

algoritmo

*criar sub-rede aproximada  
resolver problema de fluxos com custos fixos associado  
retornar à sub-rede original*

fim algoritmo

### Criação da Sub-Rede Aproximada

As sub-redes locais aproximadas criadas contêm o concentrador associado mais os nodos terminais com demanda servidos por esse concentrador. A sub-rede de interligação aproximada é composta pelo centro distribuidor de fluxos e pelos concentradores.

A única diferença entre os processos de criação dessa e daquelas sub-redes é a necessidade de correção das demandas dos concentradores. Para se criar a sub-rede de interligação aproximada deve-se considerar que a demanda de um concentrador é a soma das demandas dos nodos por ele servidos acrescida da demanda local, que pode em alguns casos ser nula. A fig. 1.4 mostra os dois tipos possíveis de concentradores: com e sem demanda local.

Quaisquer dois nodos da sub-rede aproximada são ligados por um arco com custos fixo e variável iguais ao do caminho mínimo entre eles, na sub-rede original. Para todas as sub-redes é dispensada a determinação dos caminhos mínimos entre quaisquer dois nós, uma vez que essa operação, essencial à fase de localização dos concentradores, já se encontra efetuada nesse ponto.

### Resolução do Problema de Fluxos com Custos Fixos

O problema de fluxos com custos fixos é resolvido pelo seguinte algoritmo:

algoritmo

*aplicar heurística "gulosa" de adição de arcos*  
*aplicar heurística de substituição de arcos*

fim algoritmo

O método de relaxação lagrangeana já foi utilizado [Hochbaum & Sergev, 1989] para se demonstrar que essas heurísticas são capazes de fornecer limites superiores de boa qualidade. São propostas duas relaxações diferentes e um processo para determinação dos valores iniciais dos multiplicadores de Lagrange. É sabido que o sucesso do método de otimização de sub-gradientes é fortemente dependente dos valores iniciais, sendo objeto de intensos esforços a determinação de boas estratégias de inicialização. O número de iterações requeridas pode ser reduzido significativamente quando se começa de um bom conjunto de multiplicadores lagrangeanos.

De fato, no caso do problema de fluxos com custos fixos (FCF), modelo P3, os multiplicadores de Lagrange iniciais propostos - determinados a partir do seu dual - são tão bons que pouca melhoria se consegue nos limites superiores e inferiores gerados. Não há, portanto, nenhuma vantagem em se prosseguir com as iterações de sub-gradiente, operação que envolve um relativo esforço computacional [Hochbaum & Sergev, 1989].

Resumindo:

- .é conhecido um procedimento para inicialização dos multiplicadores de Lagrange associados a duas relaxações lagrangeanas diferentes do modelo P3;
- .os limites inferior e superior gerados por esse conjunto inicial são próximos e pouco melhoráveis através de iterações de sub-gradiente;
- .é apresentado um procedimento heurístico de adição e substituição de arcos que produz um limite superior de qualidade comparável àquele obtido pelas relaxações propostas.

Por fim, não se pode perder de vista que o problema de fluxos com custos fixos (FCF), modelo P3, é uma mera aproximação para o problema original, modelo P2. Portanto, a procura por uma solução ótima para o problema de fluxos com custos fixos (FCF) não precisa ser levada às últimas consequências.

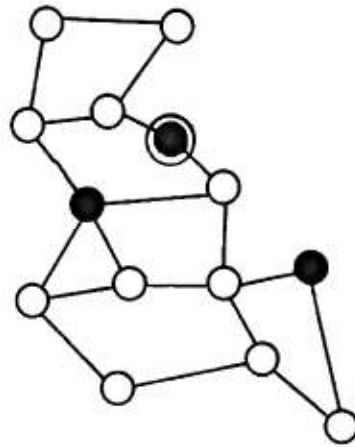
### Retorno à Sub-rede Original

A etapa de retorno ao grafo original consiste na descoberta de caminhos mínimos. A fig. 2.4-c mostra um caso onde o retorno à sub-rede original se resumiu na descoberta dos três caminhos mínimos correspondentes aos três arcos da solução da sub-rede aproximada, fig. 2.4-b.

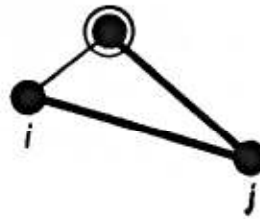
Essa procura, entretanto, nem sempre é tão simples. Considere-se, por outro lado, a otimização da sub-rede ilustrada na fig. 2.5. A fig. 2.5-b mostra a aproximação para a sub-rede original e a solução encontrada. No retorno à sub-rede original, o nodo origem  $i$  não deve se ligar ao nodo destino  $j$  diretamente, como pede a solução, fig. 2.5-b. É suficiente que seja feita a conexão do nodo  $i$  ao nodo  $k$ , fig. 2.5-c.

Sempre se tenta descobrir os arcos do caminho mínimo entre origem e destino. Se, entretanto, o destino já estiver conectado à fonte de fluxos (concentradores, no caso de sub-redes locais e centro distribuidor de fluxos, no caso da sub-rede de interligação) e se um ponto pertencente a esse caminho for alcançado, então a conexão já pode ser considerada atendida.

O algoritmo de retorno define para cada nodo um hierarquia (que não guarda nenhuma relação com a hierarquia da rede). A fonte de fluxos, por definição, tem hierarquia igual a zero. Todos os nodos ligados diretamente a ela na solução aproximada têm hierarquia igual a 1 (um). Aqueles ligados aos nodos de hierarquia igual a 1 (um) têm hierarquia igual a 2 (dois), e assim sucessivamente. A hierarquia de um nodo é a hierarquia daquele ao qual se liga na solução aproximada, acrescida de uma unidade.

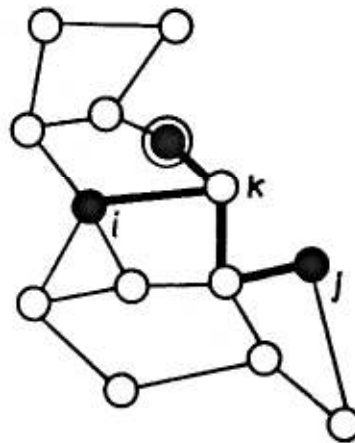


a ) sub-rede original



— solução

b ) sub-rede aproximada e a solução encontrada



— solução

c ) sub-rede original e a solução encontrada

## 2.5 - otimização de uma sub-rede



A etapa seguinte é de ligação dos nodos na rede real. Primeiramente são ligados à fonte de fluxos, pelo caminho mínimo, os nodos com hierarquia igual a um. Em seguida, os outros nodos, por ordem crescente de hierarquia, são ligados ao ponto mais próximo do caminho que liga o nodo destino à fonte de fluxos.

O algoritmo abaixo descreve o processo de retorno à sub-rede original:

algoritmo

```

calcular hierarquia de cada nó
calcular máxima hierarquia encontrada
      ( ligar nodo com hierarquia igual a zero )
ligar fonte de fluxos a ela mesma
      ( ligar nodos com hierarquias superiores )
para HIERARQUIA de 1 até HIERARQUIA_MAXIMA faça
      enquanto existir algum nodo com hierarquia corrente faça
          determinar ponto de conexão
          ligar nodo ao ponto de conexão
      fim enquanto
fim para

fim algoritmo

```

O algoritmo de retorno começa criando um arco que liga a fonte de fluxos (concentradores, no caso de sub-redes locais e centro distribuidor de fluxos, no caso da sub-rede de interligação, figuras 2.2 e 2.3) a ela mesma. Dessa forma, todos os nodos são ligados segundo um único procedimento, mesmo aqueles com hierarquia igual um.

## CAPÍTULO III - DA IMPLEMENTAÇÃO

Desenvolveu-se um sistema para projeto de redes hierárquicas OPNET. O sistema OPNET não se destina a um usuário final. O seu objetivo único é a obtenção de alguma experiência computacional com o algoritmo proposto.

Desse sistema espera-se apenas o processamento de um arquivo de entrada, contendo informações a respeito da rede que se quer otimizar. Como resultado, o sistema deve gerar um arquivo de saída, com informações suficientes para recuperar a topologia otimizada e construir uma representação da rede, como a mostrada na fig. 1.4. Arquivos intermediários gerados deverão ser mantidos, pois facilitam a tarefa de correção da implementação.

O sistema OPNET implementado será descrito a seguir em termos de diagramas de fluxos de dados, dicionário de dados e diagramas hierárquicos dos módulos.

### 3.1 - Diagrama de Fluxo de Dados

Pela análise de requisitos do sistema, pode-se identificar um entidade externa: o usuário. O usuário será responsável pela confecção do arquivo de entrada.

O diagrama de fluxo de dados (DFD) do sistema é representado em nível de topo (nível primário) na fig. 3.1, conforme convenções simbólicas consagradas [Gane & Sarson, 1983].

Antes de se passar ao segundo nível, convém lembrar que o processo 1 é desempenhado por editores de texto, já existentes. A sua função é receber a entrada pelo teclado e possibilitar a formatação adequada do arquivo de entrada - de responsabilidade do usuário.

O processo 2, ilustrado na fig. 3.2, não é propriamente parte do sistema. Sua função é gerar redes aleatórias de teste. Sua presença não é fundamental ao funcionamento do algoritmo, mas tão

somente ao seu teste. O processo 2.1 será mais detalhado na parte de testes, quando se falar no grupo de redes aleatórias. Importante, aqui, é o nível secundário do processo 3, que se encontra ilustrado na fig. 3.3.

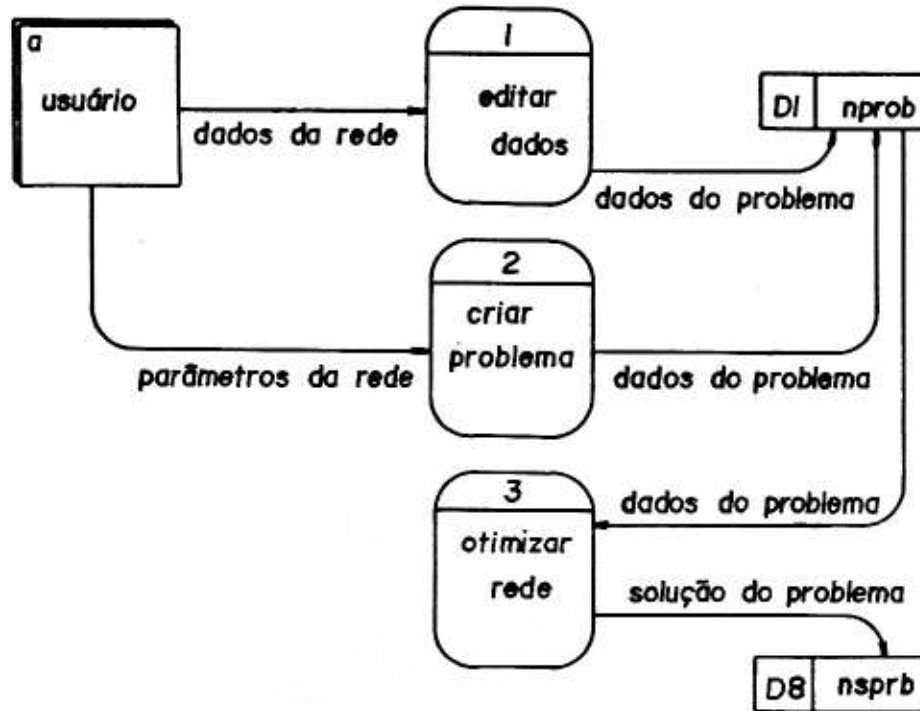


fig. 3.1 - DFD do sistema em nível primário

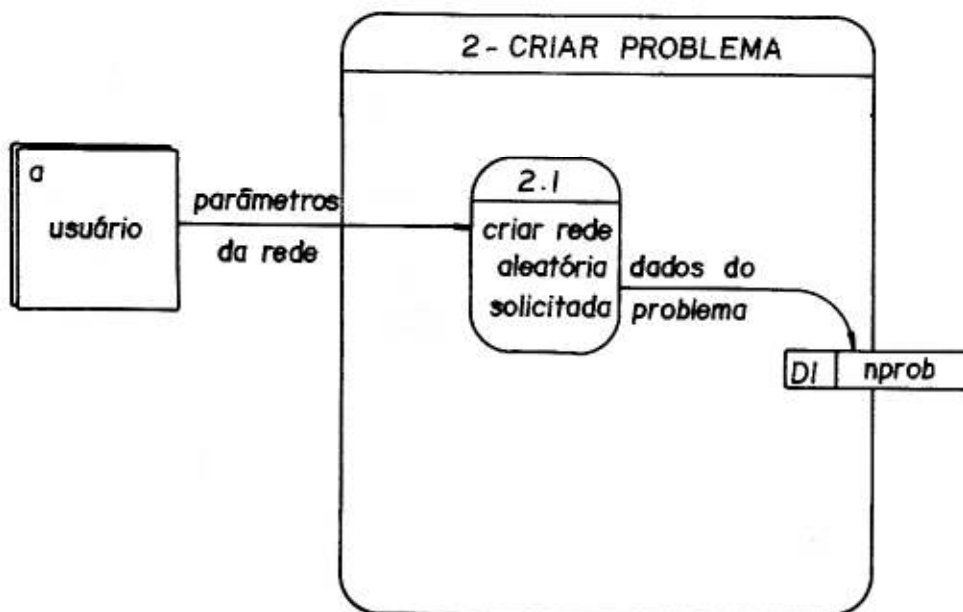


fig. 3.2 - nível secundário do processo 2

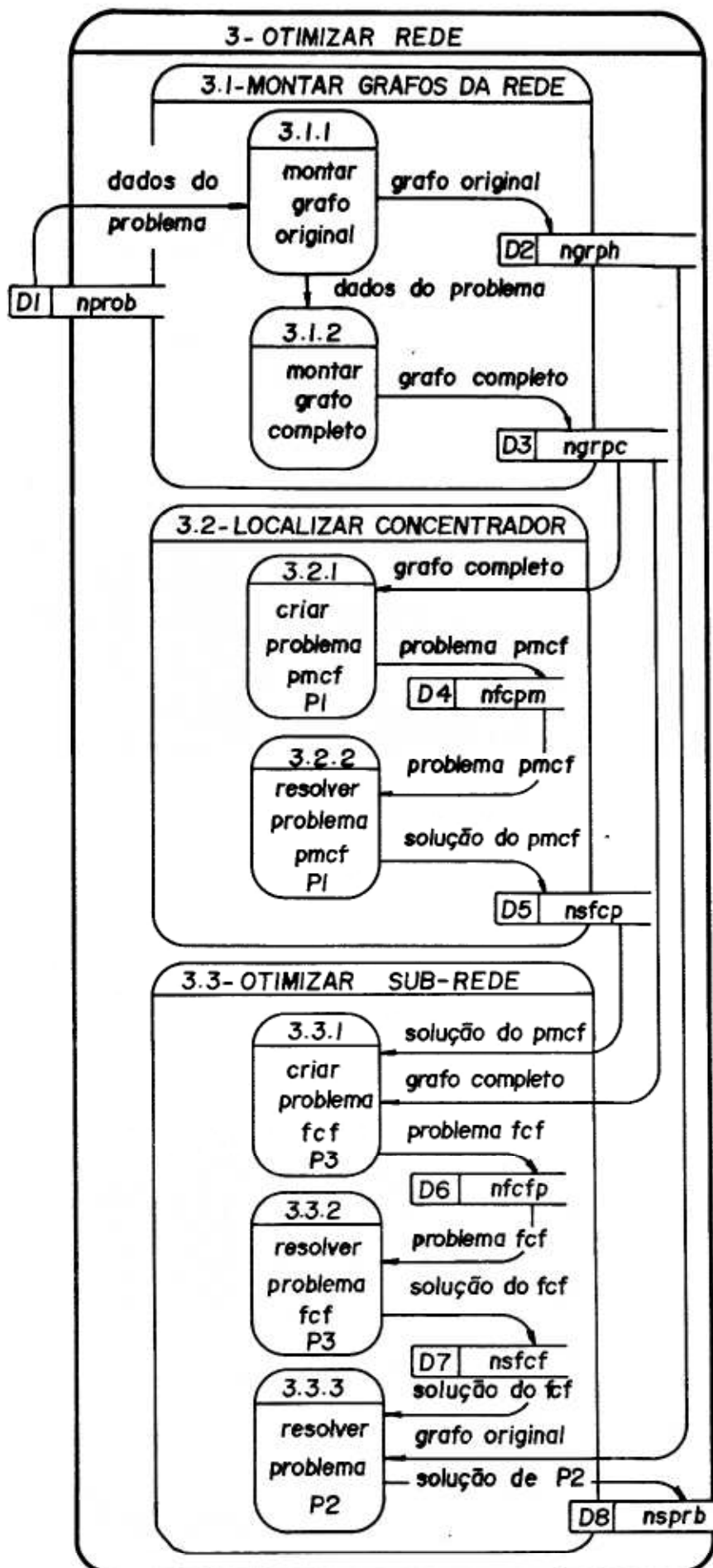


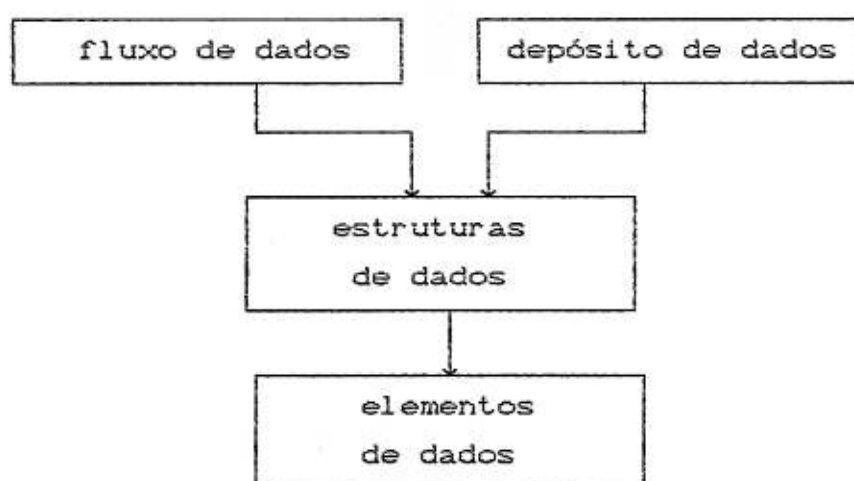
fig. 3.3 - nível secundário do processo 3

Pelo diagrama de fluxo de dados descrito na fig. 3.3, os depósitos de dados D2, D3, D5 são internos ao processo 3, não sendo conhecidos no exterior dele. São ainda mais internos os depósitos D4, D6 e D7. O depósito D4 é conhecido apenas no interior do sub-processo 3.2 e os depósitos D6 e D7, apenas no interior do 3.3. Seria suficiente se o sistema fosse implementado exatamente conforme essa descrição, pois os requisitos exigidos dele estariam perfeitamente cumpridos. Mas, objetivando-se facilitar a tarefa de correção da implementação, optou-se por deixar todos os depósitos intermediários, entre D1 e D8, externos ao processo 3.

Já se falou sobre todos os sub-processos na parte 2. Pode-se passar ao dicionário de dados, para um descrição mais detalhada das estruturas empregadas.

### 3.2 - Dicionário de Dados

A hierarquia de descrição de dados é feita conforme mostra o diagrama abaixo [Gane & Sarson, 1988]:



## Descrição dos Fluxos de Dados

Os fluxos são auto-explicativos. Mesmo assim, será dada uma breve descrição de cada um deles, bem como das estruturas incluídas:

a) dados da rede:

descrição: passa informações que definem o problema geral, por meio de uma estrutura de adjacências;

estrutura incluída: DADOS\_REDE;

b) dados do problema:

descrição: passa informações que definem o problema geral, por meio de uma estrutura de adjacências;

estrutura incluída: PROBLEMA;

c) grafo completo:

descrição: o problema geral é descrito por matriz de distâncias mínimas;

estrutura incluída: GRAFO\_COMPLETO;

d) grafo original:

descrição: forma alternativa para se descrever o problema geral, por matriz de adjacências;

estrutura incluída: GRAFO\_ORIGINAL;

e) parâmetros da rede:

descrição: compreende os parâmetros necessários para geração aleatória de um problema geral;

estrutura incluída: PARÂMETROS\_REDE;

f) problema fcf:

descrição: problema de fluxos com custos fixos (FCF) associado a uma das sub-rede do problema geral;

estrutura incluída: PROBLEMA\_FCF

g) problema pmcf:

descrição: problema das p-medianas com custos fixos (PMCF) associado ao problema geral;

estrutura incluída: PROBLEMA\_PMCF;

h) solução de P2:

descrição: solução do problema de programação inteira mista, modelo P2, associado a uma das sub-redes do problema geral; contém informações que possibilitam a recuperação da

topologia otimizada da sub-rede;

estrutura incluída: SOLUCAO\_P2;

i) solução do problema:

descrição: é uma forma alternativa para se referir ao fluxo solução de P2;

j) solução do problema fcf:

descrição: solução do problema de fluxos com custos fixos (FCF) associado a uma das sub-redes do problema geral;

estrutura incluída: SOLUCAO\_PFCF

k) solução do problema pmcf:

descrição: solução do problema das p-medianas com custos fixos (PMCF) associado ao problema geral;

estrutura incluída: SOLUCAO\_PPMCF.

#### Depósito de Dados

Pelos diagramas de fluxo de dados apresentados nas figuras 3.1, 3.2 e 3.3, podem ser identificados oito depósitos de dados. As informações relativas a cada um deles são apresentadas em seguida:

a) nfcfl:

referência: D6;

descrição: problemas de fluxos com custos fixos (FCF) associados a cada uma das sub-redes (locais ou de interligação) que compõem o problema geral;

conteúdo: estrutura PROBLEMA\_FCF;

b) nfcpm:

referência: D4;

descrição: problema das p-medianas com custos fixos (PMCF) associado ao problema geral;

conteúdo: estrutura PROBLEMA\_PMCF;

c) ngrpc:

referência: D3;

descrição: matriz de adjacências do grafo completo associado ao problema geral, mais dados restantes;

- conteúdo: estrutura GRAFO\_COMPLETO;
- d) ngrph:  
referência: D2;  
descrição: matriz de adjacências do problema geral e demais dados;  
conteúdo: estrutura GRAFO\_ORIGINAL;
- e) nprob:  
referência: D1;  
descrição: dados do problema geral, um por vez;  
conteúdo: estrutura PROBLEMA;
- f) nsfcp:  
referência: D5;  
descrição: solução do problema das p-medinas com custos fixos (PMCF) associado;  
conteúdo: estrutura SOLUCAO\_PPMCF;
- g) nsfcf:  
referência: D7;  
descrição: solução de cada um dos problemas de fluxos com custos fixos (FCF);  
conteúdo: estrutura SOLUCAO\_PFCF;
- h) nsprb:  
referência: D8;  
descrição: topologia otimizada de cada uma das sub-redes componentes do problema geral;  
conteúdo: estrutura SOLUCAO\_P2.

### Estruturas de Dados e Elementos de Dados

A seguir são apresentadas descrições das estruturas de dados e seus elementos componentes:

- a) DADOS\_REDE e PROBLEMA: estruturas que descrevem o problema geral a ser processado;
- NNOS: número de nós da rede;
- NMED: número máximo de concentradores a instalar;



CENT: número de identificação do nodo central;

LOCFIX: custo por unidade de distância de sub-rede local;

LOCVAR: custo por unidade de fluxo por unidade de distância de sub-rede local

INTFIX: custo por unidade de distância da sub-rede de interligação;

INTVAR: custo por unidade de fluxo por unidade de distância da sub-rede de interligação;

CSTINS: custo fixo de instalação dos concentradores;

DEMI\*(1-): demanda de cada nodo da rede;

NARCS: número de arcos da rede;

ARCS\*(1-): estrutura que descreve cada arco da rede;

EXTR1: identificação de uma extremidade do arco;

EXTR2: identificação da outra extremidade do arco;

DIST: comprimento do arco;

b) GRAFO\_COMPLETO: fornece a descrição do grafo completo associado ao problema geral, em matriz de adjacências;

NNOS: número de nós da rede;

NMED: número máximo de concentradores a instalar;

CENT: número de identificação do nodo central;

LOCFIX: custo por unidade de distância de sub-rede local;

LOCVAR: custo por unidade de fluxo por unidade de distância de sub-rede local

INTFIX: custo por unidade de distância da sub-rede de interligação;

INTVAR: custo por unidade de fluxo por unidade de distância da sub-rede de interligação;

CSTINS: custo fixo de instalação dos concentradores;

DEMI\*(1-): demanda de cada nodo da rede;

DIJ: estrutura contendo a distância mínima entre  $i$  e  $j$ ;

TIME: tempo de UCP gasto para transformação da estrutura GRAFO\_ORIGINAL na estrutura GRAFO\_COMPLETO;

c) GRAFO\_ORIGINAL: fornece a descrição do problema geral em matriz de adjacências;

NNOS: número de nós da rede;

NMED: número máximo de concentradores a instalar;

CENT: número de identificação do nodo central;  
 LOCFIX: custo por unidade de distância de sub-rede local;  
 LOCVAR: custo por unidade de fluxo por unidade de distância de sub-rede local  
 INTFIX: custo por unidade de distância da sub-rede de interligação;  
 INTVAR: custo por unidade de fluxo por unidade de distância da sub-rede de interligação;  
 CSTINS: custo fixo de instalação dos concentradores;  
 DEMI\*(1-): demanda de cada nodo da rede;  
 DIJ: estrutura contendo o comprimento do arco que liga o nó i ao j; se não existir, o comprimento é um número grande;  
 TIME: tempo de UCP gasto para transformação da estrutura PROBLEMA na estrutura GRAFO\_ORIGINAL;

d) PARAMETROS\_REDE: estrutura que contem os parâmetros necessários para geração aleatória de um problema geral;

NNOS: número de nós;  
 NARCS: número de arcos;  
 NOSDEM: número de nós com demanda unitária;

e) PROBLEMA\_FCF: descreve o problema de fluxos com custos fixos (FCF) para uma das sub-redes do problema geral;

NET: número de identificação da sub-rede; a sub-rede de interligação tem identificação igual a zero;

NOINDT: número de nodos dessa sub-rede;

RAIZ: número de identificação do nodo raiz;

$\left\{ \begin{array}{l} \text{LOCFIX} \\ \text{INTFIX} \end{array} \right\}$ : custo por unidade de comprimento de sub-rede local, caso NET seja diferente de zero, ou da sub-rede de interligação, caso contrário;

$\left\{ \begin{array}{l} \text{LOCVAR} \\ \text{INTVAR} \end{array} \right\}$ : custo por unidade de fluxo por unidade de comprimento de sub-rede local, caso NET seja diferente de zero, ou da sub-rede de interligação, caso contrário;

NODIND\*(1-): número da identificação de cada nodo componente da sub-rede identificada por NET;

DEMI\*(1-): demanda de cada nodo da sub-rede;

DIJ: estrutura que contem a distância mínima entre os

nodos  $i$  e  $j$  que pertencem à sub-rede identificada por NET; aqui, as identificações  $i$  e  $j$  não são as mesmas usadas nas estruturas GRAFO\_ORIGINAL, GRAFO\_COMPLETO, etc.

TIME: tempo de UCP para criação do problema;

f) PROBLEMA\_PMCF: descreve o problema das  $p$ -medianas com custos fixos (PMCF) associado ao problema geral;

NNOS: número de nós da rede;

NMED: número máximo de concentradores a instalar;

CSTLOC\*(1-): o custo de localização de cada uma das facilidades;

CSTLIG\*(1-): custo de ligação do nodo  $i$  à facilidade  $j$ ;

TIME: tempo de UCP gasto para geração da estrutura PROBLEMA\_PMCF;

g) SOLUCAO\_PFCF: descreve a solução do problema de fluxos com custos fixos (FCF), para uma determinada sub-rede do problema geral;

NET: número de identificação da sub-rede; a sub-rede de interligação tem identificação igual a zero;

NNOS: número de nodos dessa sub-rede;

RAIZ: número de identificação do nodo raiz;

$\left\{ \begin{array}{l} \text{LOCFIX} \\ \text{INTFIX} \end{array} \right\}$ : custo por unidade de comprimento de sub-rede local, caso NET seja diferente de zero, ou da sub-rede de interligação, caso contrário;

$\left\{ \begin{array}{l} \text{LOCVAR} \\ \text{INTVAR} \end{array} \right\}$ : custo por unidade de fluxo por unidade de comprimento de sub-rede local, caso NET seja diferente de zero, ou da sub-rede de interligação, caso contrário;

LSUP: limite superior da solução ótima do problema FCF;

NODIND\*(1-): número da identificação de cada nodo componente da sub-rede identificada por NET;

DEMI\*(1-): demanda de cada nodo da sub-rede;

PREDNO\*(-1): identificação do nodo predecessor de cada um dos nodos da rede;

TIME: tempo de UCP gasto para resolução do problema;

h) SOLUCAO\_PPMCF: apresenta a solução do problema das  $p$ -medianas com custos fixos (PMCF);

NNOS: número de nós; .  
 NMED: número de medianas efetivamente instaladas;  
 LSUP: limite superior da solução ótima do problema PMCF;  
 LINF: limite inferior da solução ótima do problema PMCF;  
 NITER: número de iterações de sub-gradiente executadas;  
 MEDIAN\*(1-): número de identificação de cada mediana;  
 DAD\*(1-): identificação da mediana que serve cada nodo;  
 TIME: tempo de UCP gasto para resolução do problema;

i) SOLUCAO\_P2: descreve a solução do modelo P2, associado a uma das sub-redes do problema geral;

NET: número de identificação da sub-rede; a sub-rede de interligação tem identificação igual a zero;

NNOS: número de nodos dessa sub-rede;

RAIZ: número de identificação do nodo raiz;

LSUP: limite superior da solução ótima do modelo P2;

PRDRES\*(-1): identificação do predecessor de cada nodo;

TIME: tempo de UCP gasto para resolução do problema;

### 3.3 - Diagramas Hierárquicos

O objetivo dessa parte é derivar um projeto estruturado do modelo lógico descrito nos diagramas de fluxo de dados, figuras 3.1 e 3.3. O projeto procura alcançar um sistema estruturado para alterabilidade [Gane & Sarson, 1983]:

- sistema composto de hierarquia de módulos caixa preta;
- cada módulo administrativamente (fácil de entender) pequeno;
- cada módulo alterável sem criar um efeito de propagação;
- funções do usuário isoladas no menor número possível de módulos.

Um dos objetivos ao se projetar alterabilidade é conseguir um mínimo de acoplamento entre os vários módulos, mas ainda permitindo o funcionamento do sistema:

- usando o acoplamento de dados o máximo possível;
- usando acoplamento de controle o mínimo possível;
- evitando acoplamentos mais sérios (externo, de conteúdo, etc).

Ao mesmo tempo que se minimiza o acoplamento, deve-se procurar maximizar a coesão modular:

- usando módulos funcionais tanto quanto possível;
- evitar módulos coesivos lógicos e coincidentes a todo custo.

A fig. 3.4 ilustra a estrutura hierárquica do sistema OPNET, em nível superior:

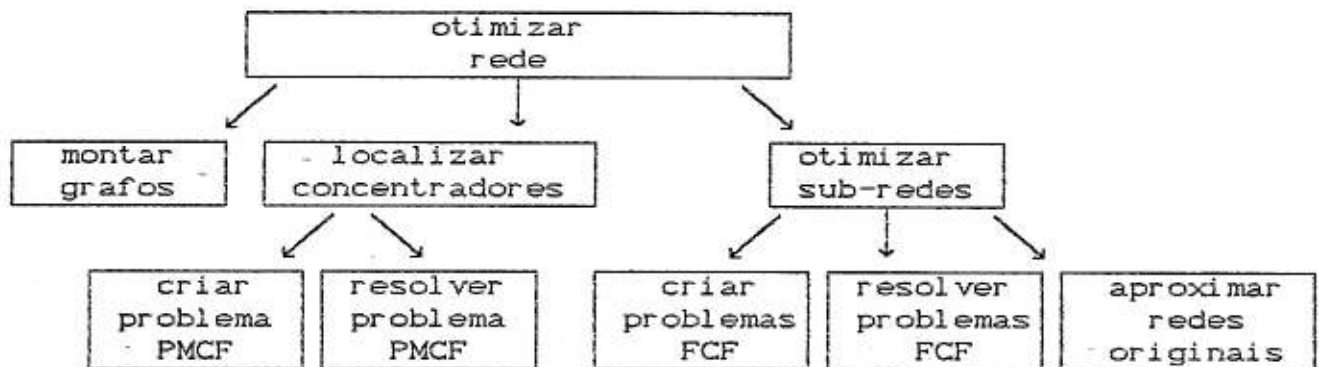


fig. 3.4 - estrutura de nível superior do sistema

Os seguintes módulos são ativados pelo sistema OPNET:

- a) módulo montar grafos:  
 fonte: arquivo MGRPHS;  
 interface: MGRPHS(nprob, ngrph, ngrpc);
- b) módulo localizar concentradores:  
 fonte: arquivo LELRS;  
 interface: LELRS(ngrpc, nfcpm, nsfcp);

o módulo *localizar concentradores* ativa os seguintes módulos:

- b1) módulo criar problema PMCF (p-medianas com custos fixos):  
 fonte: arquivo LELRS;  
 "entry point": CRFCPM(ngrpc, nfcpm);
- b2) módulo resolver problema PMCF (p-medianas com custos fixos):  
 fonte: arquivo LELRS;  
 "entry point": RSFCPM(nfcpm, nsfcp);
- c) módulo otimizar sub-redes:  
 fonte: arquivo OPSNET;  
 interface: OPSNET(ngraph, ngrpc, nsfcp, nfcfl, nsfcf, nsprb);

o módulo *otimizar sub-redes* ativa os módulos:

- c1) módulo criar problemas FCF (fluxos com custos fixos):  
 fonte: arquivo OPSNET;  
 "entry point": CRFCFL(ngrpc, nsfcp, nfcfl);
- c2) módulo resolver problemas FCF (fluxos com custos fixos):  
 fonte: arquivo OPSNET;  
 "entry point": SVFCFL(nfcfl, nsfcf);
- c3) módulo aproximar redes originais (modelo P2):  
 fonte: arquivo OPSNET;  
 "entry point": SVREAL(ngraph, nsfcf, nsprb).

A seguir serão apresentados os diagramas hierárquicos do módulos apresentados na fig. 3.4. A descrição dos arquivos utilizados é mostrada logo após a apresentação de todos eles.

### Módulo Montar Grafos

São os seguinte os módulos ativados pelo módulo *montar grafos*, fig. 3.5:

- a) módulo leituras:  
 fonte: arquivo RDWR;

- "entry point": RDPROB(nprob, PROBLEMA);  
 descrição: leitura do problema geral descrito em estrutura de adjacências, do depósito *nprob*;
- b) módulo algoritmo montar problema:  
 fonte: arquivo MGRPHS;  
 descrição: monta matriz de adjacências correspondente ao problema geral;
- c) módulo escritas:  
 fonte: arquivo RDWR;  
 "entry point": WRGRPH(ngraph, GRAFO\_ORIGINAL);  
 descrição: escrita do problema geral em matriz de adjacências, no depósito *ngraph*;
- d) módulo montar grafo completo:  
 fonte: arquivo ALGPHC;  
 "entry point": DIJKSC(GRAFO\_ORIGINAL, GRAFO\_COMPLETO);  
 descrição: cria matriz de distâncias mínimas pelo algoritmo de Dijkstra;
- e) módulo escritas:  
 fonte: arquivo RDWR;  
 "entry point": WRGRPH(ngrpc, GRAFO\_COMPLETO);  
 descrição: escrita do problema geral em matriz de distâncias mínimas, no depósito *ngrpc*;

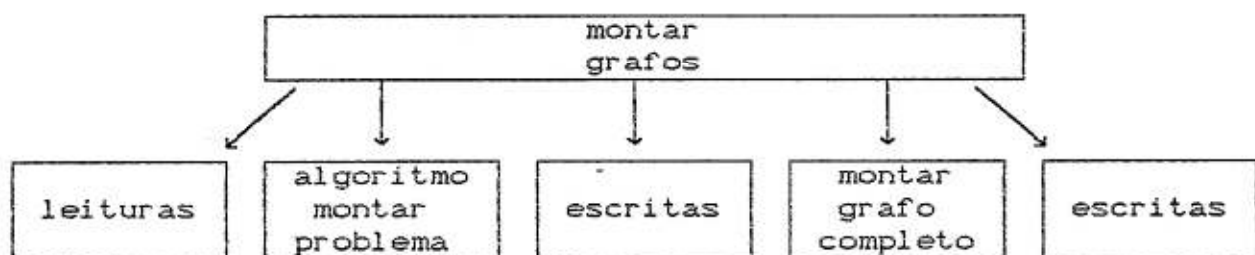


fig. 3.5 - diagrama hierárquico do módulo *montar grafos*

Abaixo, os módulos ativados pelo módulo *criar problema PMCF*, cujo diagrama hierárquico é visto na fig. 3.6.

a) módulo leituras:

fonte: arquivo RDWR;

"entry point": RDGRPH(ngRPC, GRAFO\_COMPLETO);

descrição: leitura do grafo completo associado ao problema geral, do depósito *ngRPC*;

b) módulo algoritmo montar PPMCF:

fonte: arquivo: LELRS;

descrição: montagem do problema das p-medianas com custos fixos (PMCF) associado ao problema geral;

c) módulo escritas:

fonte: arquivo RDWR;

"entry point": WRFCPM(nfcPM, PROBLEMA\_PMCF);

descrição: escrita do problema das p-medianas com custos fixos (PMCF), no depósito *nfcPM*;

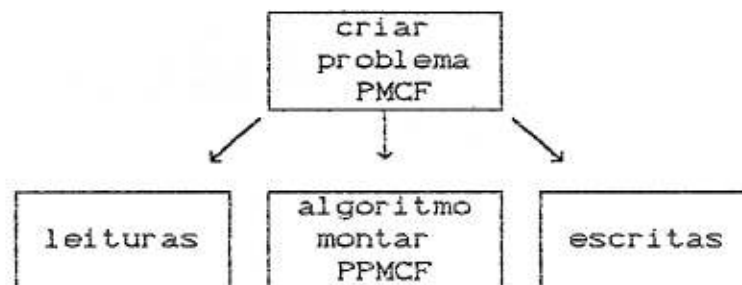


fig. 3.6 - diagrama hierárquico do módulo *criar problema PMCF*

#### Módulo Resolver Problema PMCF (P-Mediana com Custos Fixos)

Os módulos ativados pelo módulo *resolver problema PMCF* são:



## a) módulo leituras:

fonte: arquivo RDWR;

"entry point": RDFCPM(nfcpm, PROBLEMA\_PMCF);

descrição: leitura do problema das p-medianas com custos fixos (PMCF), do depósito *nfcpm*;

## b) módulo algoritmo resolver PPMCF:

fonte: arquivo ALFCPM

interface: ALFCPM(PROBLEMA\_PMCF, SOLUCAO\_PPMCF);

## c) módulo escritas:

fonte: arquivo RDWR;

"entry point": WRSFCP(nsfcp, PROBLEMA\_PMCF);

descrição: escrita da solução do problema das p-medianas com custos fixos (PMCF), no depósito *nsfcp*;

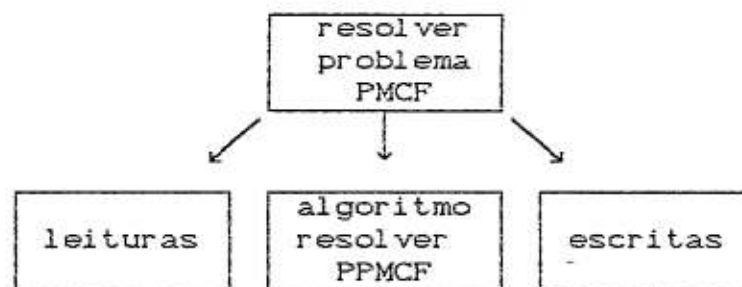


fig. 3.7 - diagrama hierárquico do módulo *resolver problema PMCF*

### Módulo Criar Problemas FCF (Fluxos com Custos Fixos)

Os seguintes módulos são ativados pelo módulo *criar problemas FCF*:

## a) módulo leituras:

fonte: arquivo RDWR;

"entry point": RDGRPH(ngRPC, GRAFO\_COMPLETO);

descrição: leitura do grafo completo associado ao problema

geral, do depósito *ngrpc*;

b) módulo leituras:

fonte: arquivo RDWR;

"entry point": RDSFCPC(nsfcp, SOLUCAO\_PFCPMD);

descrição: leitura da solução do problema das p-medianas com custos fixos (PMCF), do depósito *nsfcp*;

c) módulo criar problemas sub-rede local:

fonte: arquivo OPSNET;

descrição: criação dos problemas de fluxos com custos fixos (FCF) associados às sub-redes locais do problema geral;

o módulo *criar problemas sub-rede local* ativa os módulos:

c1) módulo criar problema sub-rede local:

fonte: arquivo OPSNET;

descrição: cria um dos problemas de fluxos com custos fixos (FCF) associados às sub-redes locais;

c2) módulo escritas:

fonte: arquivo RDWR;

"entry point": WRFCFL(nfcfl, PROBLEMA\_FCF);

descrição: escrita de um dos problemas de fluxos com custos fixos (FCF), no depósito de dados *nfcfl*;

d) módulo criar problema sub-rede de interligação:

fonte: arquivo OPSNET;

descrição: criação do problema de fluxos com custos fixos (FCF) associado à rede de interligação;

o módulo *criar problema sub-rede de interligação* ativa o módulo:

d1) módulo escritas:

fonte: arquivo RDWR;

"entry point": WRFCFL(nfcfl, PROBLEMA\_FCF);

descrição: escrita do problema de fluxos com custos fixos (FCF), no depósito de dados *nfcfl*;

Na fig. 3.8 está representado o diagrama hierárquico do módulo

criar problema FCF:

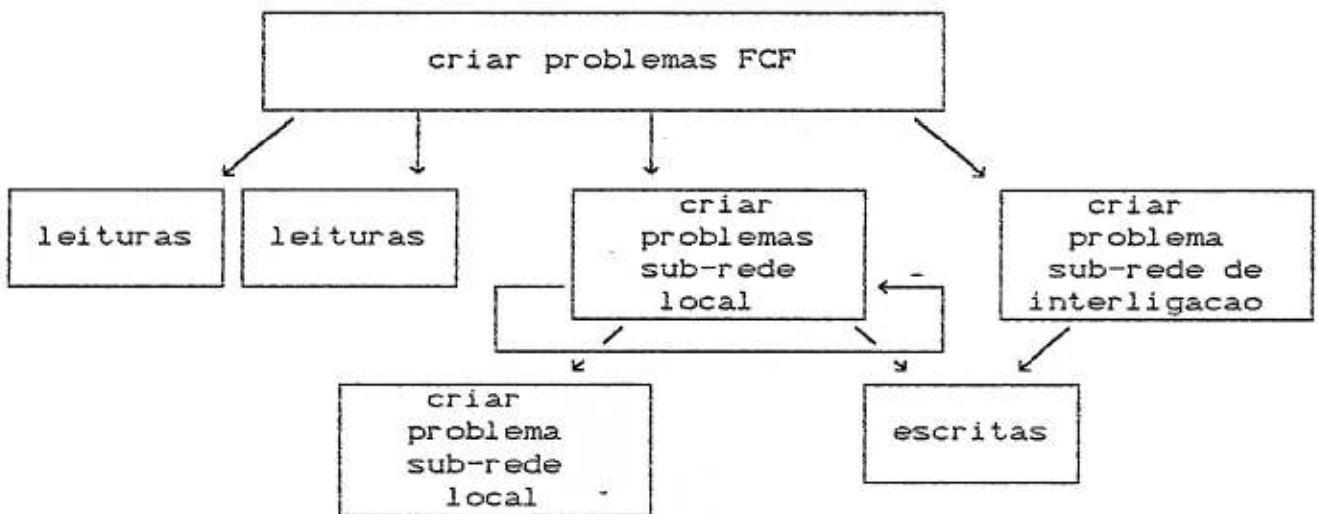


fig. 3.8 - diagrama hierárquico do módulo *criar problemas FCF*

Módulo Resolver Problemas FCF (Fluxos com Custos Fixos)

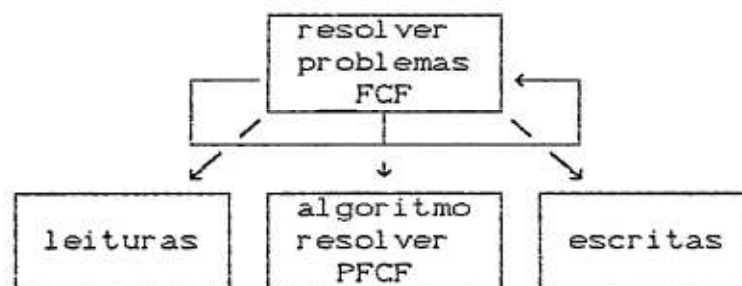


fig. 3.9 - diagrama hierárquico do módulo *resolver problemas FCF*

O módulo *resolver problemas FCF*, com o diagrama hierárquico respectivo representado na fig. 3.9, ativa os módulos abaixo:

## a) módulo leituras:

fonte: arquivo RDWR;

"entry point": RDFCFL(nfcfl, PROBLEMA\_FCF);

descrição: leitura do problema de fluxos com custos fixos (FCF) associado a uma das sub-redes do problema geral, do depósito de dados *nfcfl*;

## b) módulo algoritmo resolver PFCF:

fonte: arquivo ALFCFL;

interface: ALFCFL(PROBLEMA\_FCF, SOLUCAO\_PFCF);

## c) módulo escritas:

fonte: arquivo RDWR;

"entry point": WRSFCF(nsfcf, SOLUCAO\_PPMCF);

descrição: escrita da solução do problema de fluxos com custos fixos (FCF) associado a uma das sub-redes do problema geral, no depósito *nsfcf*;

## Módulo Aproximar Redes Originais

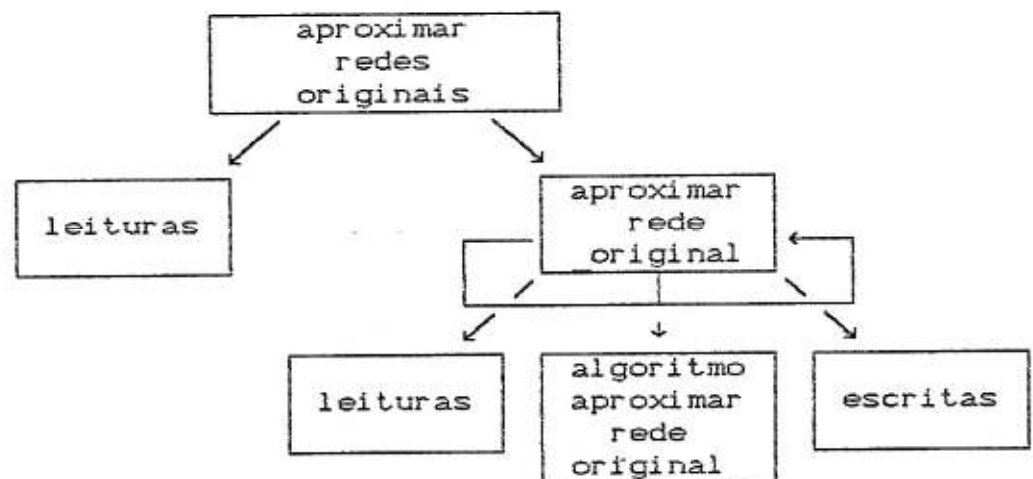


fig. 3.10 - diagrama hierárquico do módulo *aproximar redes originais*

A fig. 3.10 ilustra o diagrama hierárquico do módulo *aproximar redes originais*, que ativa os seguintes módulos:

a) módulo leituras:

fonte: arquivo RDWR;

"entry point": RDGRPH(ngmph, GRAFO\_ORIGINAL);

descrição: leitura do problema geral em matriz de adjacências, do depósito *ngmph*;

b) módulo aproximar rede original:

fonte: arquivo OPSNET;

descrição: dá uma solução aproximada para uma das sub-redes do problema geral, baseada na solução do problema de fluxos com custos fixos (FCF) associado à respectiva sub-rede aproximada;

o módulo *aproximar rede original* ativa os seguinte módulos:

b1) módulo leituras:

fonte: arquivo RDWR;

"entry point": RDSFCF( *nsfcf*, SOLUCAO\_PPMCF );

descrição: leitura da solução do problema de fluxos com custos fixos (FCF) associado a uma das sub-redes do problema geral, do depósito *nsfcf*;

b2) módulo algoritmo aproximar rede original:

fonte: arquivo ALRNET;

interface: ALRNET( SOLUCAO\_PFCF, SOLUCAO\_P2 );

b3) módulo escritas:

fonte: arquivo RDWR;

"entry point": WRSPRB( *nsprb*, SOLUCAO\_P2 );

descrição: escrita da solução do modelo P2, para uma das sub-redes do problema geral, no depósito *nsprb*.

### Conteúdo dos Arquivos

A tab. 3.1 mostra o conteúdo de cada um dos oito arquivos

utilizados pelo sistema implementado (OPNET).

tab. 3.1 - conteúdo dos arquivos utilizados no sistema OPNET

DEPÓSITO	ARQUIVO	CONTEÚDO
nprob	PROBLEM	estrutura PROBLEMA
ngrph	PRGRAPH	estrutura GRAFO_ORIGINAL
ngrpc	PRGRPHC	estrutura GRAFO_COMPLETO
nfcpm	PRFCPME	estrutura PROBLEMA_PMCF
nsfcp	PRSFCPM	estrutura SOLUCAO_PPMCF
nfcfl	PRFCFLW	estrutura PROBLEMA_FCF*
nsfcf	PRSF CFL	estrutura SOLUCAO_PFCF*
nsprb	PRSOLUT	estrutura SOLUCAO_P2*

(\*) a estrutura aparece repetidamente no arquivo;

Os três últimos arquivos contêm as estruturas respectivas repetidas tantas vezes quanto o número de sub-redes em que foi dividido o problema geral. Por exemplo, no caso da rede ilustrada na fig.1.4, o arquivo PRSOLUT contém soluções de quatro problemas:

-as três primeiras soluções correspondem às três sub-redes locais, fig.2.2;

-a quarta solução corresponde à sub-rede de interligação, fig.2.3.

## CAPÍTULO IV - DOS TESTES E RESULTADOS

O sistema OPNET, descrito anteriormente, foi implementado pela técnica de desenvolvimento "top-down" na linguagem FORTRAN 77, com todos os módulos cuidadosamente testados também pela estratégia "top-down". Todos os testes foram executados em um computador IBM 4341.

Antes de se passar aos testes do algoritmo proposto, serão mostrados os resultados obtidos com as implementações dos dois algoritmos que o compõem:

- algoritmo para resolução do problema PMCF (primeira fase);
- algoritmo para projeto das sub-redes (segunda e terceira fases);

### 4.1 - Algoritmo Para Resolução do Problema PMCF

tab. 4.1 - resultados para o problema das 33 cidades

$f_j$	p	Limite Superior	Limite Inferior	UCP <sup>(#)</sup>
184	31	6.024	6.024	4,54
295	19	8.673	8.673	8,43
500	10	11.450	11.267 <sup>(1)</sup>	5,79 <sup>(*)</sup>
1000	6	14.832	14.832	1,60
1500	6	17.832	17.832	2,54
2000	4	20.363 <sup>(1)</sup>	20.346	4,22 <sup>(*)</sup>
2500	3	22.122	22.122	2,32
3000	2	23.474	23.474	2,28
4000	2	25.474	25.474	1,34
5000	2	27.474	27.474	1,27

(#) tempo de UCP, em segundos, somente para resolução;

(\*) número máximo de iterações (MAX\_IT=3n) alcançado;

(1) limite igual ao ótimo (Erlenkotter,1978);

As tabelas 4.1 e 4.2 apresentam resultados para os clássicos problemas das 33 e das 57 cidades [Karg & Tompson, 1964].

Os resultados obtidos mostram que a implementação fornece resultados concordantes com a literatura [Erlenkotter, 1978; Mateus & Carvalho, 1991], para esses problemas clássicos.

tab. 4.2 - resultados para o problema das 57 cidades

$f_j$	p	Limite Superior	Limite Inferior	UCP <sup>(#)</sup>
50	55	2.821	2.821	30,54
200	29	9.233	9.142 <sup>(1)</sup>	52,80 <sup>(*)</sup>
500	13	15.294	15.241	25,84 <sup>(*)</sup>
1000	9	20.287	20.287	15,50
1500	7	23.923	23.923	14,13
2000	6	27.202	27.202	14,15
2500	5	30.022	30.022	12,56
3000	4	32.136	32.136	9,62
4000	3	35.547	35.547	13,87
5000	3	38.547	38.547	8,69

(#) tempo de UCP, em segundos, somente para resolução;

(\*) número máximo de iterações (MAX\_IT=3n) alcançado;

(1) limite igual ao ótimo [Erlenkotter,1978];

Não se pode usar o tempo de UCP com medida de eficiência da implementação, uma vez que eles são conhecidos para máquinas diferentes (IBM 4331, [Galvão & Raggi, 1989], IBM-PC 386, [Mateus & Carvalho, 1991]).

Uma forma melhor, é através do número de iterações gastas. As tabelas 4.3 e 4.4 comparam o número de iterações de sub-gradientes do algoritmo implementado com duas outras implementações, [Galvão & Raggi, 1989] e [Mateus & Carvalho, 1991]. As comparações só fazem sentido naqueles casos onde a implementação proposta alcançou o



ótimo. As implementações 2 e 3 resolvem todos os problemas até o ótimo sempre, eventualmente tendo que ativar um estágio de separação e avaliação ("branch and bound").

Os resultados das tabelas 4.3 e 4.4 mostram que a implementação 3 é a que gasta maior número de iterações de sub-gradientes. Isso não indica de maneira clara a superioridade da definição de uma função contínua para o tamanho do passo no método de sub-gradientes - que é empregada nas implementações 1 e 2, por meio das equações 2.19 e 2.20, respectivamente. No caso da implementação 3, usa-se um passo de tamanho também variável, mas não como uma função contínua.

tab. 4.3 - número de iterações de sub-gradiente para o problema das 33 cidades

$f_j$	implementação 1 <sup>(4)</sup>	implementação 2 <sup>(2)</sup>	implementação 3 <sup>(3)</sup>
184	0	0	1
295	93	35 <sup>(5)</sup>	102
500	99 <sup>(4)</sup>	100 <sup>(5)</sup>	163 <sup>(6)</sup>
1000	0	0	132
1500	9	21	132
2000	99 <sup>(4)</sup>	65 <sup>(5)</sup>	93
2500	54	35 <sup>(5)</sup>	73
3000	46	21	3
4000	0	26	3
5000	0	11	3

(1) implementação proposta;

(2) [Galvão & Raggi,1989];

(3) [Mateus & Carvalho,1991];

(4) número máximo de iterações (MAX\_IT=3n) alcançado antes do ótimo;

(5) número total de iterações, considerando etapa de separação e avaliação;

(6) número de iterações no nodo inicial, mas entrou na estágio de separação e avaliação.

Sabe-se que o ponto de partida para o método de sub-gradientes influencia enormemente o número gasto de iterações [Hochbaum & Sergev, 1989]. Isso pode ajudar a explicar porque a implementação 3 gastou sempre mais iterações.

Comparando as implementações 1 e 2, há uma ligeira vantagem para essa, quanto ao número de iterações. Mas como as duas se diferem em dois pontos - heurística de inicialização e função contínua que define o tamanho de passo, item 2.2 - fica difícil determinar se algum deles tem uma influência maior que o outro.

tab. 4.4 - número de iterações de sub-gradiente para o problema das 57 cidades

$f_j$	implementação 1 <sup>(1)</sup>	implementação 2 <sup>(2)</sup>	implementação 3 <sup>(3)</sup>
50	0	0	1
200	171 <sup>(4)</sup>	172 <sup>(5)</sup>	248 <sup>(6)</sup>
500	171 <sup>(4)</sup>	171 <sup>(5)</sup>	248 <sup>(6)</sup>
1000	47	53	185
1500	54	59 <sup>(5)</sup>	216
2000	55	42	93
2500	28	33	134
3000	59	36	228
4000	47	36	143
5000	50	40	248 <sup>(6)</sup>

(1) implementação proposta;

(2) [Galvão & Raggi, 1989];

(3) [Maleus & Carvalho, 1991];

(4) número máximo de iterações (MAX\_IT=3n) alcançado antes da solução ótima;

(5) número total de iterações considerando etapa de separação e avaliação;

(6) número de iterações no nodo inicial, mas entrou no estágio de separação e avaliação.

Acredita-se que os tempos de UCP da implementação proposta possam ser diminuídos pelo uso da heurística de inicialização originalmente proposta em [Galvão & Raggi, 1989]. Nota-se que um tempo considerável é gasto na inicialização, pela observância dos casos onde o ótimo foi alcançado sem necessidade da fase de relaxação lagrangeana. Onde o número de iterações atingiu o máximo, nem por isso o tempo foi absurdamente maior.

#### 4.2 - Algoritmo Para Projeto das Sub-redes

Relembrando, o projeto das sub-redes é um problema cuja solução é conhecida. O teste do algoritmo proposto para resolver esse problema tem, portanto, um termo direto de comparação.

tab. 4.5 - limites superiores gerados para a rede 1 (18 nós)

$fix^{(1)}$	$var^{(2)}$	Limite 1 <sup>(3)</sup>	UCP <sup>(4)</sup>	Limite 2 <sup>(5)</sup>
1	100	321.410	0,05	321.410
1	10	34.310	0,05	34.310
1	1	5.480	0,04	5.480
10	1	24.450	0,04	24.650
100	1	213.450	0,04	213.650

(1) custo por unidade de distância nos arcos;

(2) custo por unidade de fluxo por unidade de distância nos arcos;

(3) limite superior gerado pelo algoritmo proposto;

(4) tempo de UCP gasto pelo algoritmo proposto, em segundos;

(5) limite superior gerado pelo sistema PLOTTER (Cabral,1985).

Foram escolhidas para testes duas redes da literatura e a rede da fig 1.4, apresentada simplesmente como ilustração de redes de

alimentação em alguns trabalhos [Cabral, 1985b; Luna et al. 1987]. Foram processadas, para cada uma, várias relações entre o custo fixo total e o custo variável total da rede. Essa relação determina o grau de dificuldade do problema de fluxos com custos fixos (FCF) [Hochbaum & Sergev, 1989]. Para se ter um melhor controle sobre a relação entre o somatório dos custos fixos e o somatório dos custos variáveis, considerou-se demanda unitária para os nodos terminais com demanda (pretos). Nesse caso, tal relação se torna aproximadamente igual à relação entre custo fixo e custo variável.

tab. 4.6 - limites superiores gerados para a rede 2 (48 nós)

$fix^{(1)}$	$var^{(2)}$	Limite 1 <sup>(3)</sup>	UCP <sup>(4)</sup>	Limite 2 <sup>(5)</sup>
1	100	33.431	0,28	33.431
1	10	3.551	0,29	3.551
1	1	567	0,29	561
10	1	2.596	0,29	2.495
100	1	22.576	0,28	21.991

(1) custo por unidade de distância nos arcos;

(2) custo por unidade de fluxo por unidade de distância nos arcos;

(3) limite superior gerado pelo algoritmo proposto;

(4) tempo de UCP gasto pelo algoritmo proposto, em segundos;

(5) limite superior gerado pelo sistema PLOTTER [Cabral,1985].

A tab. 4.5 apresenta os limites superiores gerados pelo algoritmo proposto e pelo sistema PLOTTER [Cabral, 1985], para a rede 1 (18 nós e 27 arcos), [Cabral, 1985b]. A tab. 4.6, para a rede 2 (48 nós e 85 arcos), fig. 1.4. E a tab. 4.7, para a rede 3 (87 nós e 114 arcos), [Luna et al. 1987].

Os resultados começaram favoráveis ao algoritmo proposto. Para a rede 1, tab. 4.5, o sistema PLOTTER obteve soluções no máximo iguais às soluções obtidas por ele. Mas esse resultado não se manteve para as outras redes. Isso pode ser visto nas tabelas seguintes, 4.6 e 4.7. Nessas redes, o sistema PLOTTER consegue obter

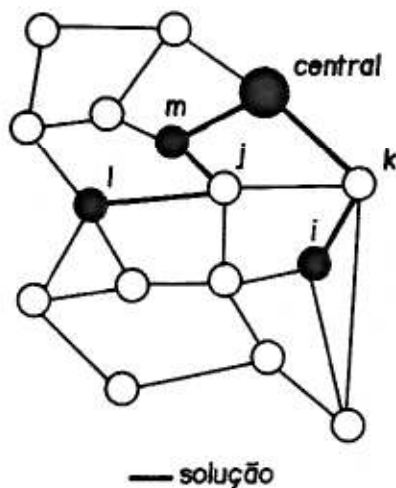
soluções melhores, naqueles casos onde os custos fixos ficam maiores que os variáveis.

tab. 4.7 - limites superiores gerados para a rede 3 (87 nós)

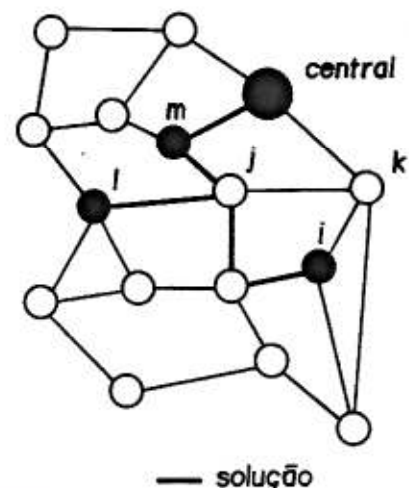
$fix^{(1)}$	$var^{(2)}$	Limite 1 <sup>(3)</sup>	UCP <sup>(4)</sup>	Limite 2 <sup>(5)</sup>
1	10	695.589	5,45	695.589
1	1	85.909	6,23	84.904
10	1	230.221	6,30	230.198

- (1) custo por unidade de distância nos arcos;  
 (2) custo por unidade de fluxo por unidade de distância nos arcos;  
 (3) limite superior gerado pelo algoritmo proposto;  
 (4) tempo de UCP gasto pelo algoritmo proposto, em segundos;  
 (5) limite superior gerado pelo sistema PLOTTER (Cabral,1985).

À luz da fig. 4.1, pode-se entender o porquê dessa diferença naqueles casos.



a) solução do algoritmo proposto



b) solução do PLOTTER

fig. 4.1 - soluções obtidas para rede 2 e  $fix/var=10$

O algoritmo proposto não consegue perceber que o nodo  $j$  é uma alternativa melhor para se ligar o nodo  $i$  à central, ainda que esse caminho seja mais longo (fig. 4.1-a). Isso não importa aqui, porque a constante  $fix$ , custo fixo no arco, é suficientemente maior que  $var$ , custo variável no arco. Caso isso não tivesse ocorrido, a ligação melhor seria realmente via nodo  $k$ . Esses são os casos onde os dois algoritmos concordam nas soluções.

O problema é que, na etapa de determinação das hierarquias dos nodos pretos, no contexto do item 2.3, os nodos brancos são retirados da rede. Em outras palavras, o algoritmo consegue determinar que a melhor opção é ligar o nodo  $i$  à central através do nodo  $m$  e ligar o nodo  $m$  à central diretamente. Mas entre ligar o nodo  $i$  à central passando por  $m$ , ou ligá-lo à central diretamente (passando pelo nodo  $k$ ), prefere a segunda opção, que é o caminho mais barato. Não consegue prever que o arco  $(j,m)$  já terá sido escolhido, quando da ligação de  $i$  a  $m$ , não sendo necessário considerar seu custo fixo novamente.

Esse problema pode ser solucionado com a adição de uma etapa de pós-otimização que estivesse atenta a esses casos. Entretanto isso depende da aplicação. Em sistemas telefônicos, em redes de alimentação convencional, por exemplo, estima-se que os custos variáveis (cabos) correspondam a 70% do custo total do arco e os custos fixos (valas e dutos) a 30%, [Luna et al., 1983]. Nesse caso, não haveria necessidade de se implementar essa etapa, pois o algoritmo estaria trabalhando com uma relação  $fix/var$  favorável.

#### 4.3 - Algoritmo Proposto

O algoritmo proposto foi testado por meio de dois grupos de duas redes cada, em crescente nível de dificuldade. As duas primeiras redes, representadas nas figuras 4.2 e 4.3, possuem as soluções ótimas para os casos de teste determináveis por inspeção. Pretende-se, com elas, verificar se para problemas pequenos o algoritmo fornece resultados aceitáveis. Não seria um bom indício se mesmo nesses problemas as soluções já fossem ruins.

O segundo grupo de testes compreende problemas aleatórios de porte prático, se fosse o caso de aplicações em redes telefônicas. A rede maior representa o tamanho máximo processável com a implementação.

#### Pequenas Redes

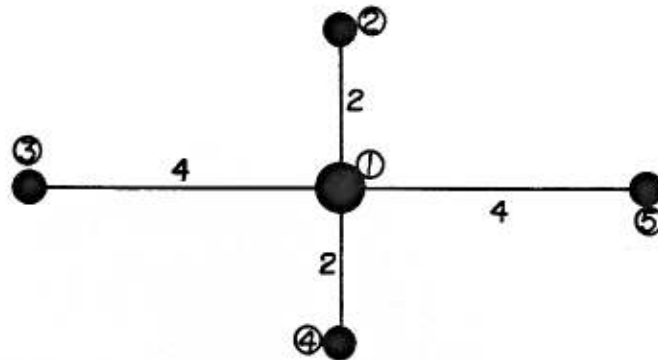


fig. 4.2 - rede de testes 4

A rede de testes 4, fig. 4.2, não poderia ser mais simples. O que seria razoável esperar desse problema é plenamente confirmado pelos resultados obtidos, apresentados na tab. 4.8.

Os dois primeiros casos de teste objetivam apenas a determinação do custo da rede para dois valores distintos do parâmetro LOCVAR. O número de medianas igual a zero é uma indicação para o sistema de que a sub-rede de interligação deve ser considerada degenerada. Por sub-rede degenerada entenda-se sub-rede composta por um único nó e um único arco, cujas extremidades coincidam.

O concentrador é confundido com a central para que seja possível determinar o custo da rede como se ela tivesse apenas uma hierarquia. São gerados dois limites - o primeiro, inferior, o segundo, superior - para referências nos casos seguintes.

O terceiro caso de teste mostra o que aconteceria caso se pudesse localizar até quatro concentradores nessa rede. Os custos fixos dos dois tipos de sub-redes - locais e de interligação - são iguais, mas os variáveis, não. Como a sub-rede de interligação é mais barata, a melhor solução é, sem dúvida, degenerar as sub-redes

locais. A solução obtida é igual ao limite inferior determinado.

O quarto caso de teste permite, no máximo, três concentradores. Pela fig. 4.2, pode-se dizer que o melhor a fazer é gastar sub-redes de interligação, mais baratas, nas maiores distâncias e sub-redes locais, mais caras, nas menores. Os concentradores são posicionados nos nodos 1, 3 e 5. O concentrador localizado no nodo 1 - em coincidência com a central - serve aos nodos 2 e 4 por meio de sub-redes locais. A sub-rede que o interliga à central é degenerada. O concentrador em 5 atende à demanda local em 5, e o concentrador em 3, à demanda local em 3. Esses dois últimos são ligados à central por sub-redes de interligação.

tab. 4.8 - testes para a rede 4

a) parâmetros dos testes

Teste	NMED	LOCFIX	LOCVAR	INTFIX	INTVAR	CSTINS
1	0 <sup>(*)</sup>	1	10	-	-	-
2	0 <sup>(*)</sup>	1	20	-	-	-
3	4	1	20	1	10	0
4	3	1	20	1	10	0
5	4	1	20	1	10	39
6	4	1	20	1	10	40

(\*) determinação do custo da rede não hierárquica;

b) resultados obtidos

Teste	Medianas	PPMCF	PFCF	PP2	UCP <sup>(#)</sup>
1	{1}	132	132	132	<1
2	{1}	252	252	252	<1
3	{4,3,5,2}	132	132	132	<1
4	{1,3,5}	198 <sup>(*)</sup>	172	172	<1
5	{2,3,4,5}	288	288	288	<1
6	{1}	292	292	292	<1

(\*) solução não ótima;

(#) soma do tempos, em segundos, para gerar grafo completo, resolver o PPMCF, resolver o PFCF e aproximar rede original.



Os dois últimos casos consideram um custo fixo de instalação dos concentradores. Para

$$\text{CSTFIX} = 39,$$

quinto caso, são possíveis as opções:

$$\text{medianas} = \{1\} \quad \Rightarrow \text{custo da rede} = 252 + 39 = 291,$$

$$\text{medianas} = \{1,3,5\} \quad \Rightarrow \text{custo da rede} = 172 + 3 \times 39 = 289,$$

$$\text{medianas} = \{2,3,4,5\} \quad \Rightarrow \text{custo da rede} = 132 + 4 \times 39 = 288.$$

O algoritmo escolheu a última, a mais barata.

Para

$$\text{CSTFIX} = 40,$$

sexto caso, as seguintes alternativas são possíveis:

$$\text{medianas} = \{1\} \quad \Rightarrow \text{custo da rede} = 252 + 40 = 292,$$

$$\text{medianas} = \{1,3,5\} \quad \Rightarrow \text{custo da rede} = 172 + 3 \times 40 = 292,$$

$$\text{medianas} = \{2,3,4,5\} \quad \Rightarrow \text{custo da rede} = 132 + 4 \times 40 = 292.$$

O algoritmo escolheu o menor número de medianas possível, uma vez que não faria diferença nenhuma escolher uma, três ou quatro.

Uma observação caberia, antes de se passar à outra pequena rede. Pode-se notar que no quarto caso de teste, o problema das p-medianas com custos fixos (PMCF) não foi resolvido até o ótimo, mesmo a rede sendo tão simples. Isso é uma indicação de que esses problemas não são assim tão fáceis.

Os resultados até então são bons. Mas não é necessário se complicar muito para que o algoritmo mostre uma deficiência. A tab. 4.9 mostra resultados para a rede 5, ilustrada na fig. 4.3.

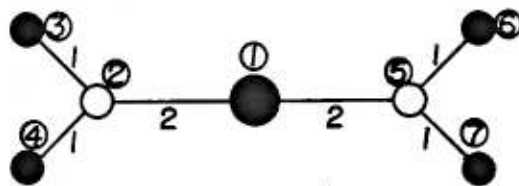


fig. 4.3 - rede de testes 5

Os dois primeiros casos degeneram a rede de interligação para determinar os custos sem concentradores, i.e., como se a rede fosse em apenas um nível hierárquico. Eles estabelecem limites inferior e superior para as soluções possíveis para rede hierárquica. Nota-se que nesses casos já aparecem diferenças entre as soluções do problema de fluxos com custos fixos (FCF) e as aproximações para a rede original. Os custos fixos dos arcos comuns a caminhos mínimos distintos - arcos (4,2) e (4,5), fig. 4.3 - são considerados duas vezes cada, na solução do problema de fluxos com custos fixos (FCF).

tab. 4.9 - testes para a rede 5  
a) parâmetros dos testes

Teste	NMED	LOCFIX	LOCVAR	INTFIX	INTVAR	CSTINS
1	0 <sup>(*)</sup>	1	10	-	-	-
2	0 <sup>(*)</sup>	1	20	-	-	-
3	4	1	20	1	10	0
4	2	1	20	1	10	0
5	4	1	20	1	10	10
6	4	1	20	1	10	30

(\*) determinação do custo da rede não hierárquica;

b) resultados obtidos

Teste	Medianas	PPMCF	PFCF	PP2	UCP <sup>(#)</sup>
1	{1}	132	132	128	<1
2	{1}	252	252	248	<1
3	{1,5,2}	128	168	168	<1
4	{2,5}	168	168	168	<1
5	{2,5}	148	188	188	<1
6	{2,5}	188	228	228	<1

(\*) solução não ótima;

(#) soma do tempos, em segundos, para gerar grafo completo, resolver o PPMCF, resolver o PFCF e aproximar rede original.

O terceiro caso, tab. 4.9, mostra qual seria a solução para a rede de testes 5 se os concentradores pudessem ser alocados em um

número máximo de quatro e a sub-rede de interligação custasse mais barato. Embora se esperasse que as redes locais, sendo mais caras, se degenerassem, somente dois concentradores foram localizados, em lugar de quatro.

O algoritmo não conseguiu determinar a solução ótima, se atrapalhando na economia de escala intrínseca da rede. A escolha pelo algoritmo dos nodos 3 e 4, fig. 4.3, implicaria na consideração dupla, pelo problema PMCF, do custo fixo do arco (4,2). Os custos totais de instalação estimados para cada nodo na etapa de localização dos concentradores, eq. 2.7, não são condicionados à escolha de outros nodos. Da forma como o problema PMCF está sendo montado, quando são alocados concentradores nos nodos 3 e 4, por exemplo, obrigatoriamente o arco (4,2) tem seu custo fixo contado por duas vezes.

O quarto caso mostra um funcionamento correto do algoritmo. Quando os concentradores são limitados a dois, os nodos escolhidos para sua localização devem ser o 2 e o 5.

Os dois últimos casos consideram dois custos de instalação diferentes. Para o quinto caso,

$$\text{CSTINS} = 10,$$

há as opções:

medianas = {1}	⇒ custo da rede = 248 + 10 = 258,
medianas = {2,5}	⇒ custo da rede = 168 + 2*10 = 188,
medianas = {3,4,6,7}	⇒ custo da rede = 128 + 4*10 = 168.

Embora não tenha sido a pior opção, o algoritmo não escolheu a melhor delas, quinto caso de teste, tab. 4.9.

O sexto caso de testes,

$$\text{CSTINS} = 30,$$

possue as alternativas de solução:

medianas = {1}	⇒ custo da rede = 248 + 30 = 278,
medianas = {2,5}	⇒ custo da rede = 168 + 2*30 = 228,
medianas = {3,4,6,7}	⇒ custo da rede = 128 + 4*30 = 248.

A escolha feita foi a melhor, conforme pode ser visto na tab. 4.9, sexto caso de teste.

### Grandes Redes Aleatórias

Os teste com redes aleatórias visam principalmente um sentimento dos tempos -de processamento para problemas de maior porte.

O segundo grupo de testes compreende as seguintes redes, geradas aleatoriamente:

-rede 6: 60 nós, 240 arcos e 10 nós aleatórios com demanda unitária;

-rede 7: 200 nós, 500 arcos e 70 nós com demanda unitária.

Esses três parâmetros definem completamente uma rede. As redes são obtidas através do seguinte algoritmo:

#### algoritmo

```

inicialize valores necessários
leia PARAMETROS
      ( criar vetor de demandas )
para CONTADOR de 1 até NNOS faça
      sortear um nó ainda não sorteado
      estabelecer demanda unitária para nó sorteado
fim para
      ( sortear coordenadas dos nodos )
para CONTADOR de 1 até NNOS faça
      sortear dois números reais no intervalo (0,NNOS)
      faça ordenada do nodo corrente igual ao primeiro número
      faça abcissa do nodo corrente igual ao segundo número
fim para

```

```

      . ( criar uma árvore geradora )
sortear extremidade 1 do arco 1
para CONTADOR de 1 até (NNOS-1) faça
  repita
    sortear extremidade 2 do arco corrente (diferente da
      sua extremidade 1)
    até novo arco não formar ciclos com os arcos já sorteados
    faça extremidade 1 do próximo arco igual à extremidade 2
      do arco corrente
fim para
      ( sortear arcos restantes )
para CONTADOR de NNOS até NARCS faça
  sortear arco que ainda não exista
fim para
calcular distâncias dos arcos sorteados
escreva PROBLEMA

fim algoritmo

```

Os números sorteados seguem uma distribuição uniforme e foram obtidos através de uma versão em FORTRAN 77 do gerador de números pseudo-aleatórios, independente da máquina, descrito em [Forsythe et al. 1977].

A tab. 4.10 apresenta resultados para a rede 6. Nos problemas aleatórios, um caso de teste adicional é considerado, porque há a necessidade de se definir a identificação do nodo central. O nodo central é aquele que minimiza o somatório de distâncias mínimas, dele a todos os outros, ou seja, é a solução do problema das  $p$ -medianas puro, com  $p$  unitário. Para que o problema das  $p$ -medianas com custos fixos (PMCF) gerado recaia nesse caso, os parâmetros INTFIX, INTVAR e CSTINS devem ser nulos (ver equações 2.7 e 2.8).

Os casos de teste 2 e 3 estabelecem os limites inferior e superior. Nota-se que a maior parcela de tempo gasto é para resolver o problema PMCF. Do primeiro caso de teste para o segundo, o acréscimo de tempo foi muito pequeno, muito embora dois algoritmos tenham sido rodados a mais nesse último.

tab. 4.10 - testes para a rede 6

## a) parâmetros dos testes

Teste	NMED	LOCFIX	LOCVAR	INTFIX	INTVAR	CSTINS
1	1 <sup>(&amp;)</sup>	1	10	0	0	0
2	0 <sup>(*)</sup>	1	10	-	-	-
3	0 <sup>(*)</sup>	1	20	-	-	-
4	10	1	20	1	10	0
5	6	1	20	1	10	0
6	10	1	20	1	10	200

(&) determinação do nodo central da rede;

(\*) determinação do custo da rede não hierárquica.

## b) resultados obtidos

Teste	Medianas	PPMCF	PFCF	PP2	UCP <sup>(#)</sup>
1	{14}	3.619	-	-	7,9
2	{14}	3.619	3.568	3.528	8,4
3	{14}	3.619	6.858	6.818	8,4
4	{13,14,20,34,38, 40,41,45,51,53}	3.208	3.608	3.568	16,7
5	{20,57,45,38,13,41}	4.819 <sup>(*)</sup>	4.726	4.726	22,9
6	{20,38,45,57,13,41}	4.906 <sup>(*)</sup>	5.926	5.926	22,9

(\*) solução não ótima;

(#) soma do tempos, em segundos, para gerar grafo completo, resolver o PPMCF, resolver o PFCF e aproximar rede original.

No quarto caso de teste é alocado o número máximo de medianas pedido, mas o conjunto não é o melhor do ponto de vista da rede, mesmo sendo ótima a solução do problema PMCF. Foi alocado um concentrador no nodo central (nodo 14), sem demanda, em detrimento de um outro com demanda. Ou seja, se um concentrador está alocado em um nodo sem demanda, um outro com demanda sobrou - eles são dez - e está sendo atendido por uma sub-rede local. Como uma das sub-redes locais não ficou degenerada, a solução obtida foi maior que o limite inferior, quando deveria ser igual.

O quinto caso limita o número de medianas em um número menor

que o número de nós com demanda. Nesse caso, alguma sub-rede local não degenerada deverá aparecer obrigatoriamente. Foi o que ocorreu e a solução se posicionou bem entre os dois limites, se aproximando mais do inferior. Nota-se que o problema PMCF associado a esse caso ficou mais difícil (o tempo gasto foi um pouco maior que o do quarto caso), a tal ponto da sua solução ótima não ter sido alcançada.

tab. 4.11 - testes para a rede 7

## a) parâmetros dos testes

Teste	NMED	LOCFIX	LOCVAR	INTFIX	INTVAR	CSTINS
1	1 <sup>(&amp;)</sup>	1	10	0	0	0
2	0 <sup>(*)</sup>	1	10	-	-	-
3	0 <sup>(*)</sup>	1	20	-	-	-
4	70	1	20	1	10	0
5	30	1	20	1	10	0
6	70	1	20	1	10	1.000
7	5	1	20	1	10	10.000

(&) determinação do nodo central da rede;

(\*) determinação do custo da rede não hierárquica.

## b) resultados obtidos

Teste	Medianas	PPMCF	PFCF	PP2	UCP <sup>(#)</sup>
1	<194>	156.420	-	-	312
2	<194>	156.420	150.984	149.462	497
3	<194>	298.620	293.184	291.662	500
4	36	122.047 <sup>(*)</sup>	183.841	182.952	1.183
5	20	177.745 <sup>(*)</sup>	217.618	217.528	925
6	26	153.224 <sup>(*)</sup>	226.784	226.235	994
7	<194,77>	290.231 <sup>(*)</sup>	301.129	299.607	654

(\*) solução não ótima;

(#) soma do tempos, em segundos, para gerar grafo completo, resolver o PPMCF, resolver o PFCF e aproximar rede original.

O sexto caso limita as medianas pelo custo de instalação:

CSTINST = 200.

São conhecidos pelos menos as três opções:

.medianas = {14}	⇒ custo da rede = 6.818 + 200 = 7.018,
.medianas = {20,57,45, 38,13,41}	⇒ custo da rede = 4.726 +1200 = 5.926,
.medianas = {13,14,20, 34,38,40, 41,45,51,53}	⇒ custo da rede = 3.528 +2000 = 5.528.

O algoritmo novamente não escolhe a melhor delas, mas escolhe uma boa opção.

Esse comportamento também é verificado na rede 7. Os resultados são mostrados na tab. 4.11. Os fatos mais relevantes são que os tempos de processamento crescem muito rapidamente e os problemas das p-medianas com custos fixos (PMCF) associados tornam-se mais difíceis ainda.

As redes testadas chamam a atenção para alguns pontos. Os problemas das p-medianas com custos fixos (PMCF) não são simples e a implementação da fase de separação e avaliação para determinação do ótimo é necessária, ao contrário do que se pensou. Mas nem mesmo sua solução até o ótimo pode garantir resultados ótimos para a rede. Não se pode esquecer que eles são apenas uma aproximação. Nos casos onde essa aproximação é mais grosseira, os resultados para o rede original não são os melhores, embora bons. Na maioria das vezes, o que se otimiza pelo problema das p-medianas com custos fixos (PMCF) não corresponde a um limite superior da solução para a rede original (por exemplo, casos de testes quatro a sete, na tab. 4.11).

O capacidade do sistema ficou limitada a uma rede de tamanho máximo igual ao da rede 7, mas esse limite pode ser ampliado. O processamento por partes, módulo por módulo, não representa a mínima dificuldade, pela preocupação que se teve com a modularidade do sistema. O problema mais grave diz respeito ao tempo de processamento que cresce rapidamente com as dimensões do problema.



## CONCLUSÕES

Apresentou-se um sistema telefônico com o objetivo de situar e melhor caracterizar o problema de projeto de redes hierárquicas, objeto desse trabalho. Um sistema foi mostrado de forma resumida e rapidamente passou-se à definição do problema objeto.

Foi proposto um algoritmo para resolução do problema que segue a já clássica estratégia *dividir para conquistar*. O algoritmo solucionou o problema via sub-problemas clássicos na literatura, um deles com solução bem conhecida, o problema das p-medianas com custos fixos. Para o outro, um modelo de programação inteira mista, menos explorado, mas com um método de solução também conhecido, foi proposto um novo algoritmo para resolução.

Projeteu-se um sistema estruturado para a alterabilidade para se fazer testes com os algoritmos propostos, que foi documentado de forma mais completa possível, por meio de convenções simbólicas consagradas. O sistema foi implementado na linguagem científica FORTRAN 77 e resultados foram obtidos em um computador de grande porte IBM-4341.

Primeiro testou-se a resolução dos sub-problemas clássicos que compõem o algoritmo e finalmente o algoritmo completo. Os resultados dos sub-problemas foram apresentados. Mostraram-se satisfatórios, inclusive para o problema de programação inteira mista, para o qual se propunha um algoritmo de resolução.

Os testes com o algoritmo completo mostraram que boas soluções para o problema de projeto de redes hierárquicas podem ser determinadas, embora nem sempre as melhores. Os problemas de p-medianas com custos fixos gerados se mostraram bem mais difíceis do que o esperado, sugerindo que a determinação do seu ótimo necessite de uma etapa que de início se pensava desnecessária.

O sistema implementado conseguiu solucionar redes de porte considerável. Esses limites podem ser ainda maiores ao custo de processamento por partes - facilmente conseguido, pela modularidade do sistema - ainda que os tempos de processamento encontrados insinuem um crescimento muito rápido com o tamanho do problema.

## REFERÊNCIAS BIBLIOGRÁFICAS

- Boffey, T.B. & Karkazis, J. P-Median and Multi-Medians. *Journal of The Operational Research Society*, 35 (1984) 57-70.
- Cabral, R.H.B. PLOTTER - Plano Otimizador de Topologia em Rede: Manual do Sistema. RT 002/85, DCC-ICEx-UFMG (1985a).
- Cabral, R.H.B. Um Sistema para Planejamento de Redes Telefônicas Urbanas. Dissertação de Mestrado. DCC-ICEx-UFMG (1985b).
- Cornuejols, G.; Fisher, M.L. & Nemhauser, G.L. Location of Bank Accounts to Optimize Float: an Analytic Study of Exact and Approximate Algorithms. *Management Science*, 23 (1977) 789-810.
- Christofides, N. & Beasley, J.E. A Tree Search Algorithm for the P-Median Problem. *European Journal of Operational Research*, 10 (1982) 196-204.
- Dijkstra, E.W. A Note on Two Problems in Connection with Graphs. *Numer. Math.*, 1 (1959) 269-271.
- Eilon, S. & Galvão, R.D. Single and Double Vertex Substitution in Heuristic Procedures for the P-Median Problem. *Management Science*, 24 (1978) 1763-1766.
- Erlenkotter, D. A Dual-Based Procedure for Uncapacitated Facility Location. *Operations Research*, 26 (1978) 992-1009.
- Fisher, M.L. The Lagrangean Relaxation Method for Solving Integer Programming Problems. *Management Science*, 27 (1981) 1-17.
- Forsythe, G.E.; Malcolm, M.A. & Moler, C.B. *Computer Methods for Mathematical Computations*, Prentice Hall, New York (1977).
- Gane, C. & Sarson, T. *Análise Estruturada de Sistemas*, LTC - Rio de Janeiro (1988).
- Galvão, R.D. & Raggi, L.A. A Method for Solving to Optimality Uncapacitated Location Problems. *Annals of Operations Research*, 18 (1989) 225-244.
- Galvão, R.D. ; Raggi, L.A. & Poubel, M.W. Método de Substituição de Vértices na Solução de Problemas de Localização em Redes. *Avances en Investigación Operativa*, Yanina, Buenos Aires, (1986) 151-162.
- Geoffrion, A.M. Lagrangean Relaxation and its Uses in Integer Programming. *Mathematical Programming Study*, 2 (1974) 82-114.

- Guignard, M. A Lagrangean Dual Ascent Algorithm for Simple Plant Location Problems. *European Journal of Operational Research*, 35 (1988) 193-200.
- Hanjoul, P. & Peeters, D. A Comparison of Two Dual-Based Procedures for Solving the P-Median Problem. *European Journal of Operational Research*, 20 (1985) 387-396.
- Held, M.; Wolfe, P. & Crowder, H.P. Validation of Subgradient Optimization. *Math. Prog.*, 6 (1974) 62-88.
- Hochbaum, D.S. & Segev, A. Analysis of a Flow Problem with Fixed Charges. *Networks*, 19 (1989) 291-312.
- Järvinen, P.; Rajala, J. & Sinervo, H. A Branch-and-Bound Algorithm for Seeking the P-Median. *Operations Research*, 20 (1972) 173-178.
- Karg, R.L. & Thompson, G.L. A Heuristic Approach to Solving Travelling Salesman Problems. *Management Science*, 10 (1964) 225-248.
- Luna, H.P.L. Modelagem Matemática do Problema de Planejamento de Rede Telefônica de Alimentação. RT 004/84, DCC-ICEx-UFMG, (1984).
- Luna, H.P.L. & Mateus, G.R. Relatório Final dos Projetos de Otimização de Redes Telefônicas. RT 025/89, DCC-ICEx-UFMG (1989).
- Luna, H.P.L. ; Mateus, G.R. & Lage, L.C.M. Modelos de Planejamento de Redes Telefônicas em Áreas Multi-centrais. RT 011/88, DCC-ICEx-UFMG (1988).
- Luna, H.P.L. ; Ziviani, N. & Cabral, R.H.B. Caracterização do Problema de Planejamento de Redes Telefônicas de Alimentação. RT 014/83, DCC-ICEx-UFMG (1983).
- Luna, H.P.L. ; Ziviani, N. & Cabral, R.H.B. The Telephonic Switching Centre Networks Problem: Formalization and Computational Experience. *Discrete Applied Mathematics*, 18 (1987) 199-210.
- Narula, S.C.; Ogbu, U.I. & Samuelsson, H.M. An Algorithm for the P-Median Problem. *Operations Research*, 25 (1977) 709-713.
- Mateus, G.R. & Luna, H.P.L. Estudo de Projetos Desenvolvidos por Convênio UNICAMP/TELEBRÁS. RT 009/87, DCC-ICEx-UFMG (1987).
- Mateus, G.R. & Luna, H.P.L. Combinatorial Optimization in Telephonic Network Planning. ALIO-EURO Workshop on Practical Combinatorial Optimization, Rio de Janeiro (1989).

- Mateus, G.R. & Carvalho, J.C.P. O Problema de Localização Não Capacitado: Modelos e Algoritmos. RT 010/91, DCC-ICEx-UFMG (1991).
- Raggi, L.A. Comunicação Pessoal. UFV, Viçosa. (1991).
- Said, A. & França, P.M. Combinando Heurísticas e Métodos de Subgradientes para Resolver o Problema das P-Medianas. Anais do 7<sup>o</sup> Congresso Brasileiro de Automática, São José dos Campos, SBA (1988) 112-117.
- Tavares, H.M.F. ; França, P.M. ; Soares Filho, S. & Araujo, E.O. Localização de Centrais Urbanas, CT 033/80 RD, RT-02, FEC-UNICAMP (1981).
- Teitz, M. & Bart, P. Heuristic Methods for Estimating the Generalized Vertex Median of Weighted Graph. Operations Research, 16 (1968) 955-961.
- Ziviani, N. ; Luna, H.P.L. & Cabral, R.H.B. Projeto Operacional do Programa PLOTTER - Plano Otimizador de Topologia em Rede. RT 018/83, DCC-ICEx-UFMG (1983).