

PROBLEMA DE LOCALIZAÇÃO DE CENTRAIS TELEFÔNICAS URBANAS :
UMA NOVA ABORDAGEM

REGINA CÉLIA XAVIER

Orientador :

Prof. Dr. Paulo Morelato França

Co-orientador:

Prof. Dr. Hermano de Medeiros F. Tavares

Dissertação apresentada no Instituto de Matemática, Estatística e Ciência da Computação, como requisito parcial para obtenção do título de Mestre em Matemática Aplicada.

Novembro/1983

A meus pais
JOSE e ANETE
e a minha avô ZINOCA

Este trabalho foi realizado
através do Convênio TELEBRÁS/UNICAMP
nº 033/80 e contou com o apoio finan
ceiro do Conselho Nacional de Pesqui
sa - CNPq.

AGRADECIMENTOS

A todos que de maneira direta ou indireta contribuíram para a realização deste trabalho e, particularmente,

A toda minha família e em especial a meu pai, JOSE GONÇALVES XAVIER, maior incentivador deste trabalho e que não teve tempo de vê-lo concluído;

A Paulo Morelato França pela orientação e carinho;

A Hermano M. F. Tavares pela coorientação e contribuições valiosas dadas a este trabalho;

A Akebo Yamakami pela orientação junto aos programas desenvolvidos e, principalmente, pelo apoio e incentivo nas horas mais difíceis;

A Reynaldo Arcirio de Oliveira pela confiança e carinho;

A Luiz Alberto dos Santos pela presença e pela mão estendida nos momentos em que tudo parecia perdido;

Aos amigos da UNICAMP, especialmente a Ana, Anilton, Christiano, Denise, Evandro, Flávio, JoséAngelo, Jurandir, Maria Rosa, Maristela, Marta,Raul, Secundino;

Aos amigos da DEM, que muitas vezes se sobrecarregaram para que este trabalho pudesse ser realizado.

A Mércia pela paciência e dedicado trabalho de datilografia.

ÍNDICE

CAPÍTULO I - O PROBLEMA DE LOCALIZAÇÃO DE CENTRAIS TELEFÔNICAS

1.1. Introdução	08
1.2. Modelo de Rede Urbana	11
1.3. Formulação Matemática	16
1.4. Considerações sobre a Solução	21

CAPÍTULO II - UMA PROPOSTA DE SOLUÇÃO

2.1. Introdução	22
2.2. Algoritmo de Partição de Benders	25
2.2.1. Determinação do conjunto R	26
2.2.2. Projeção em Z	29
2.2.3. Estratégia de Relaxação	32
2.2.4. O algoritmo	33
2.2.5. Convergência do Algoritmo	37

CAPÍTULO III - RESOLUÇÃO DO PROBLEMA MESTRE ATRAVÉS DE UM MÉTODO HEURÍSTICO

3.1. Introdução	39
3.2. Uma Heurística para Programação Linear Inteira - Método de Kochemberger, McCarl e Wyman	41
3.3. Resumo do Procedimento	42
3.4. Aplicação a Problemas Testes	50
3.5. Aplicação a Problemas de Localização de Centrais Telefônicas	52

CAPÍTULO IV - RESOLUÇÃO DO SUBPROBLEMA

4.1. Introdução	63
-----------------------	----

4.2.	Formulação do Problema	63
4.3.	Algoritmos de Fluxo	66
4.4.	Representação de uma Base por meio de Índices	76
4.5.	Programação do Método	81
4.5.1.	Procura do vetor que entra na base	81
4.5.2.	Procura do vetor que sai da base	82
4.5.3.	Atualização do fluxo	83
4.5.4.	Atualização do vetor multiplicador	83
4.5.5.	Reestruturação da árvore	86
4.6.	Considerações	92
4.7.	Solução Inicial	93

CAPÍTULO V - FASE DE TRANSIÇÃO: PROGRAMA TRANZE

5.1.	Introdução	96
5.2.	Estruturação do Método	97
5.2.1.	Zona de filiação aumentada	99
5.2.2.	Idéia básica	100
5.2.3.	Desenvolvimento	106
5.2.4.	Generalização	111
5.3.	A Super-iteração	117
5.4.	Resultados Obtidos	120
5.4.1.	Considerações sobre a implementação do algoritmo	120
5.4.2.	Rede exemplo	120

CAPÍTULO VI - AJUSTE FINO - UMA TÉCNICA DE PÓS-OTIMIZAÇÃO

6.1.	Introdução	124
6.2.	Idéia Básica da Técnica	125
6.3.	Desenvolvimento	132
6.4.	Resultados Obtidos	138
6.4.1.	Fluxograma resumido do método	138
6.4.2.	Considerações sobre a implementação do algoritmo	142

6.4.3. Rede exemplo	142
6.4.4. Análise dos resultados	148

CAPÍTULO VII - COMENTÁRIOS FINAIS E SUGESTÕES

7.1. Introdução	149
7.2. Considerações sobre o Método Heurístico de Solução do Problema Mestre	149
7.3. Considerações sobre a Fase de Transição	151
7.4. Considerações sobre o Ajuste Fino	152
Bibliografia	154

CAPÍTULO I - O PROBLEMA DE LOCALIZAÇÃO DE CENTRAIS TELEFÔNICAS

1.1. INTRODUÇÃO

Um sistema telefônico tem por objetivo satisfazer as necessidades de comunicação entre assinantes que residem numa determinada área. A ligação direta entre todos os assinantes torna-se inviável por ser muito onerosa e ociosa, uma vez que a parcela de assinantes que deseja se comunicar simultaneamente é muito pequena. A solução adotada é ligar cada telefone a uma central de comutação cuja função é estabelecer a comunicação temporária entre os assinantes da rede.

Por outro lado, a contínua evolução das necessidades de comunicação entre as pessoas faz com que os sistemas telefônicos tenham sua estrutura física constantemente alterada, a fim de que não se estabeleçam insatisfações decorrentes da queda de qualidade no atendimento aos seus usuários. Assim, torna-se necessário um planejamento ótimo para determinar esta evolução de tal modo que a qualidade do serviço oferecido não seja inferior a valores pré-fixados (grau de serviço), ao menor custo possível.

Um estudo global deste planejamento, que nos desse a evolução de cada um de seus componentes, reveste-se de grande complexidade devido ao grande número de variáveis

discretas e não linearidades envolvidas. Entretanto, algumas simplificações são feitas de modo a tornar possível o planejamento. Assim, a abordagem tem sido feita considerando-se uma decomposição do estudo no tempo e no espaço. Escolhe-se um período de planejamento e faz-se um estudo do estado do sistema no instante final do período. Uma outra simplificação consiste em tratar o problema de evolução em várias etapas. As etapas consistem na determinação de novos centros de fios com suas zonas de filiação e posteriormente, partindo desta localização, determinar o entroncamento entre centrais bem como o roteamento.

Será objeto deste trabalho a abordagem apenas do Problema de Localização de Centrais Locais (PLCL), sob um horizonte de longo prazo (20-30 anos).

É bom lembrar que na prática estes problemas são, em sua maioria, de grande porte, envolvendo redes com grande número de candidatos em potencial para a localização da central, e por isso difíceis de serem tratados.

A abordagem dada por ARAÚJO em [1], para a solução do PLCL, utilizou uma técnica de decomposição, que divide o problema em dois níveis. Num primeiro nível resolve-se o problema que fixa uma alternativa de localização (Programa Mestre). O segundo nível resolve um problema de fluxo de mínimo que avalia o custo da solução gerada na primeira etapa (subproblema). O processo se repete um número finito de vezes até a convergência.

O programa mestre consiste na solução de um problema inteiro com variáveis binárias, cuja performance, quanto ao tempo para convergência, diminui na medida em que o número de variáveis aumenta. Isto nos sugeriu o desenvolvimento de um procedimento heurístico de solução no sentido de acelerar a convergência (Capítulo 3). Quanto ao problema de fluxo de custo mínimo, envolvido no subproblema, procuramos desenvolver um procedimento melhorado de busca de base inicial factível que trabalhando com um menor número de arcos, aproveitasse o conhecimento da base da iteração anterior (Capítulo 5).

Finalmente, considerando ainda ARAÚJO [1], verificamos que no procedimento de escolha dos candidatos à central, este dividiu a rede em regiões candidatas e tomou uma seção de serviço (nó) como representante de cada região. A escolha destes elementos considerou as densidades de assinantes da rede. Além disso foram impostas restrições proibindo a localização de centrais em regiões próximas.

Uma vez determinada a solução ótima para estes candidatos (AJUSTE GROSSO), procurava-se refinar a localização, variando os candidatos dentro das regiões. Este procedimento apresenta desvantagens pois restringe a otimização apenas aos componentes da região candidata e não a todos os nós da rede. Dentro deste enfoque foi objeto do Capítulo 6 deste trabalho, o desenvolvimento de um procedimento automatizado de refinamento da solução obtida no AJUSTE GROSSO,

fazendo uma análise de pós-otimização sobre todos os nós da rede.

1.2. MODELO DE REDE URBANA

Para que se faça o estudo da localização de no vos centros de fios^(*) locais numa rede urbana, primeiramente é necessário que se caracterize esta rede.

Ao nível de interesse do problema aqui tratado, uma rede telefônica pode ser compreendida pela figura 1.

Todo *assinante* da rede deve estar filiado a uma *central local*.

Um assinante com desejo de falar com outro, é conectado através das centrais locais de comutação.

As centrais locais são interligadas por uma rede de cabos, chamada *rede de troncos*. A *central Tandem* não tem assinantes filiados a ela; é utilizada para estabelecer

...

(*) Entende-se por *Centro de Fios*, o ponto para onde convergem as seções de serviço de uma certa área geográfica. Cada seção de serviço é conectada a uma central local. Um mesmo Centro de Fios pode conter mais de uma Central (prefixo). No desenvolver deste trabalho vamos utilizar, por um abuso de linguagem, o termo *Central Local* para designar um *Centro de Fios*.

ligações entre as centrais com o objetivo de baratear o custo do entroncamento. A *central trânsito* encaminha as chamadas interurbanas.

Os assinantes se ligam a uma central inicialmente através de *caixas de distribuição* que, por sua vez, se ligam às *seções de serviço* as quais se ligam aos centros de fios.

As seções de serviço, também chamadas *cabinet* ou *armários* agrupam um certo número de assinantes (algumas centenas) e os ligam a uma central local através da *rede primária*, em geral uma *rede de cabos subterrâneos*.

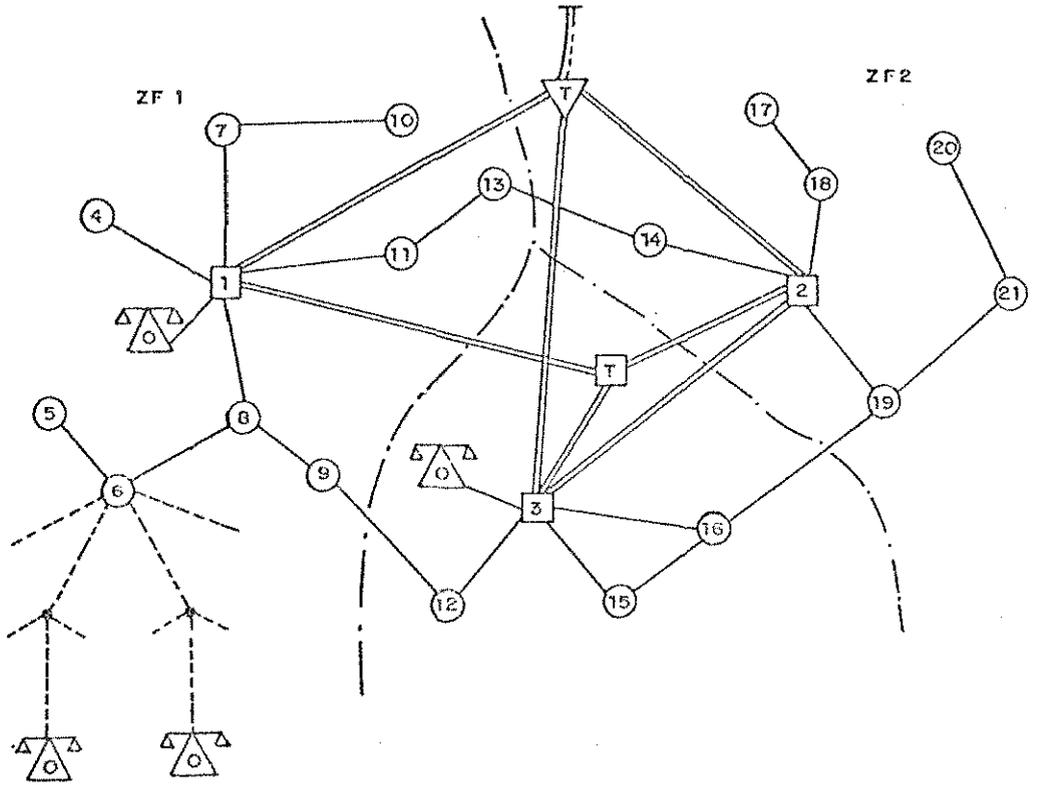
A rede secundária é uma rede de cabos, em geral aérea, que liga o assinante à seção de serviço.

A área geográfica determinada pelos assinantes que se ligam a uma central local é chamada de *zona de filiação* (ou *área de influência*) dessa central.

É possível que hajam assinantes diretamente ligados à central, sem que estejam conectados às seções de serviço; são os *assinantes diretos*.

Na tentativa de caracterizar qual a rede que realmente tem interesse para o problema de localização de centrais locais (PLCL), podemos descartar a rede secundária, visto que ela não influencia a otimização do problema.

Configura-se, então, a rede primária como a rede de estudo para o problema de localização. Nesta rede, ad



[1] - CENTRAL LOCAL DE COMUTAÇÃO J.

[T] - CENTRAL TANDEM

▽ - CENTRAL TRÂNSITO

(i) - SEÇÃO DE SERVIÇO I

o - CAIXA DE DISTRIBUIÇÃO

△ - ASSINANTE

ZF_j - ZONA DE FILIAÇÃO DA CENTRAL J

— - REDE PRIMÁRIA

- - - REDE SECUNDÁRIA

— — — REDE DE TRONCOS

— — — REDE INTERURBANA

- - - FRONTEIRAS DAS ZONAS DE FILIAÇÃO

Figura 1 - Modelo de Rede Urbana

mite-se que um centro de fios possa vir a ser implantado sobre um nó (seção de serviço) ou sobre uma aresta (duto) da rede primária. Porém, um resultado clássico da teoria de localização em redes, mostra que, nesse caso, a solução de mínimo custo dar-se-á sempre sobre um nó e não nas arestas.

A rede de estudo com relevância para a otimização da localização de centrais, é apresentada na figura 2.

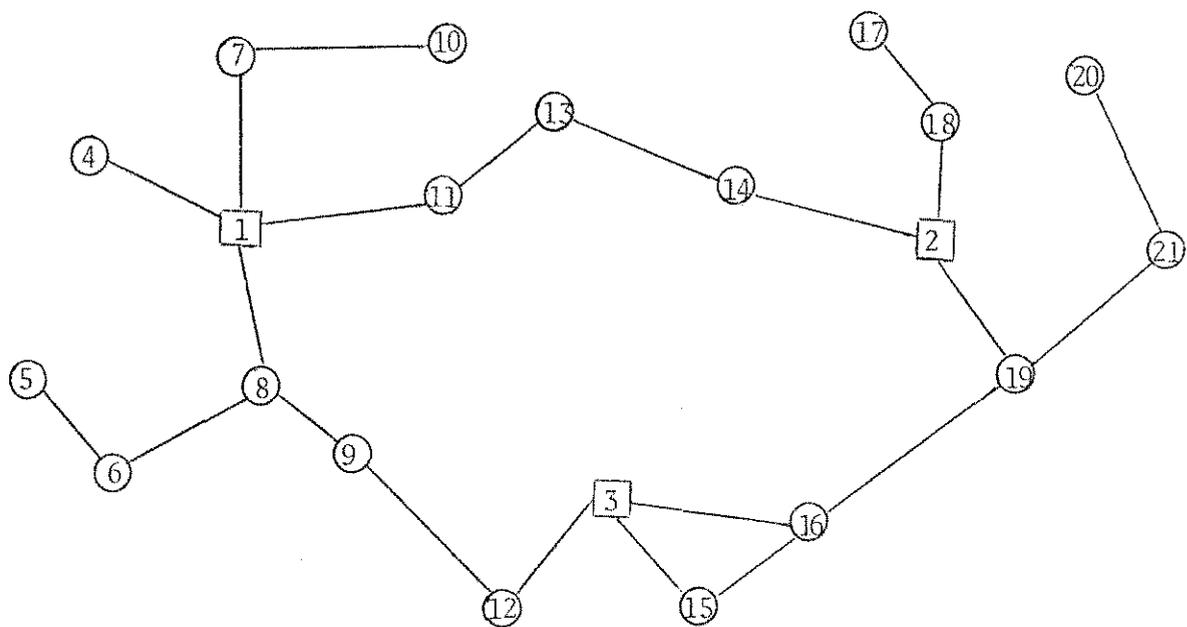


Figura 2 - Rede Primária

A definição da rede primária no ano horizonte deve compreender, numa etapa prèvia, a previsão da demanda de assinantes naquele ano, nas atuais seções de serviço, as sim como prever a demanda em regiões desabitadas e criar as seções de serviços necessàrias nestas regiões.

1.3. FORMULAÇÃO MATEMÁTICA

Segundo ^tARAÚJO [1], o problema de localização de centrais locais, pode ser formulado em duas versões. A primeira considerando o problema clàssico de transportes, de programação linear, e a segunda, um problema de fluxo de custo mìnimo numa rede. A formulação aqui abordada serà a segunda, que por sua vez, é também a sugerida por [1].

O modelo considerado, conhecido como Modelo de Fluxo de Custo Mìnimo, utiliza diretamente a rede primária mostrada na figura 2, onde cada seção de serviço i tem uma demanda de a_i assinantes e cada central j tem uma capacidade de atender b_j assinantes. O problema é fazer circular fluxo na rede de modo a satisfazer as demandas a um custo mìnimo, procurando respeitar as capacidades. Aqui, o custo do cabo é definido sobre os arcos da rede primária e não so bre os caminhos mìnimos da rede de transporte. Com isso, é possível levar em conta a capacidade dos dutos da rede de cabos.

397
 formulação matemática seria:

$$\min \sum_{i \in I} \sum_{j \in I} d_{ij} x_{ij} + \sum_{j \in J} z_j \delta_j + \sum_{j \in J} \alpha_j \left(\sum_{i \in I} x_{ij} - \sum_{k \in I} x_{jk} + a_j \right)$$

s/a

$$\sum_{i \in I} x_{ij} - \sum_{k \in I} x_{jk} \leq z_j b_j + p_j - a_j, \quad \forall j \in J \quad (1)$$

$$\sum_{i \in I} x_{ij} - \sum_{k \in I} x_{jk} \leq -a_j, \quad \forall j \in I-J \quad (2)$$

$$x_{ij} + x_{ji} \leq c_{ij} \quad \forall i \in I, \forall j \in I \quad (3)$$

$$\sum_{j \in J} z_j \leq N \max \quad (4)$$

$$\left. \begin{array}{l} z_j = 0 \text{ ou } 1; \quad \forall j \in J \\ x_{ij} \geq 0 \quad ; \quad \forall i \in I, \forall j \in J \end{array} \right\} \quad (5)$$

$$\text{restrições adicionais sobre } z \quad (6)$$

J - Conjunto de p nós constituído pelas atuais centrais e candidatos a centrais novas

I - Conjunto de todos os nós do grafo

- d_{ij} - Custo por assinante do par de fios ligando o nó i ao nó j
- δ_j - Custo fixo de localização da central j
- α_j - Custo de comutação por assinante da seção de serviço j
- a_j - Número de assinantes na seção de serviço j
- b_j - Capacidade da central futura j . Para uma central já existente é a capacidade de ampliação
- p_j - Capacidade atual da central j . Para uma central futura é zero.
- c_{ij} - Capacidade dos pares do duto (i,j)
- N_{max} - Número máximo de centrais a construir

Variáveis

- x_{ij} - Número de assinantes (pares) que passam no arco (i,j)
- z_j - Variável zero-um: se $z_j=1$ então a central z_j será localizada em j (ou ampliada); caso contrário $z_j=0$

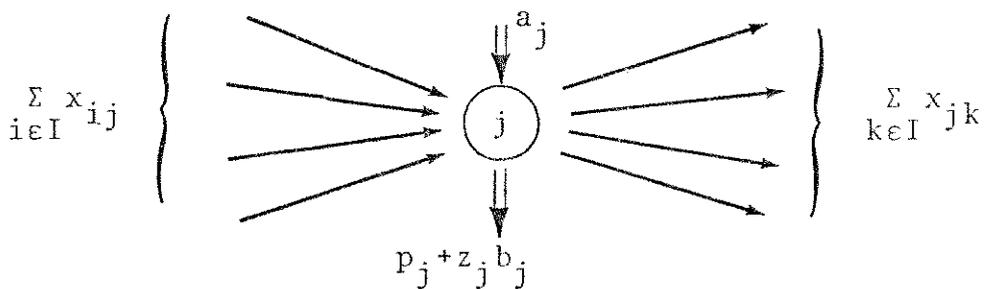
Assim, a função objetivo de (P) procura minimizar, simultaneamente, o custo dos cabos, os custos fixos e o custo de comutação. A primeira parcela fornece o custo dos cabos em todos os arcos da rede. A segunda, o custo fixo de localização e a terceira o custo de comutação.

Na parcela relativa à comutação, a expressão as

sociada ao custo α_j , indica o número de assinantes conectados à central j , supondo que seus assinantes próprios estão a ela filiados e por ela são comutados.

É possível supor que uma central saturada sirva somente como nó, de passagem, para o fluxo de um grupo de assinantes.

As restrições (1) representam um compromisso entre o fluxo não ultrapassar a capacidade de cada central $j \in J$, ao mesmo tempo em que deve satisfazer a demanda das seções de serviço. Suponha j um desses nós:



- Se $j \in J$ for uma central já existente, sofrendo uma expansão ($z_j = 1$),

$$0 \leq \sum_{i \in I} x_{ij} - \sum_{k \in I} x_{jk} \leq b_j + p_j - a_j$$

- Se $j \in J$ for uma central nova sendo instalada ($z_j = 0$)

$$0 \leq \sum_{i \in I} x_{ij} - \sum_{k \in I} x_{jk} \leq b_j - a_j \text{ pois } p_j = 0$$

- Se $j \in J$ for um local candidato à nova central mas com $z_j = 0$, então simplesmente a demanda é satisfeita.

$$\sum_{i \in I} x_{ij} - \sum_{k \in I} x_{jk} = -a_j$$

As restrições (2) procuram satisfazer a demanda nas seções de serviço.

As restrições (3) incorporam limitações de capacidade dos dutos da rede primária.

A restrição (4) limita o número máximo de centrais a construir.

As restrições (6) dão flexibilidade para impor outras relações básicas sobre as variáveis z .

O presente modelo ignora a existência de uma rede atual satisfazendo uma certa demanda, porém esta característica pode ser incorporada com uma adaptação que não apresenta dificuldades teóricas.

Assim este modelo busca a melhor política de localização, no ano horizonte, partindo de uma rede des carregada. Na otimização não consideramos os custos relativos à rede já implantada, pois esta parcela é constante em todas as alternativas e, portanto, não influencia a otimização.

1.4. CONSIDERAÇÕES SOBRE A SOLUÇÃO

Considerando que o problema (P) em estudo é um problema de programação linear mista de grande porte, optou-se por uma técnica de decomposição de BENDERS [2], que através da avaliação sucessiva de diferentes alternativas de localização converge em direção a uma solução ótima.

Como pode ser visto na figura 3, a partição de Benders divide a resolução do problema em duas partes, que pode ser vista como um esquema de decomposição a dois níveis. Num primeiro nível resolve-se um problema que chamaremos de problema (ou programa) mestre, e num segundo nível um outro problema que chamaremos de sub-problema.

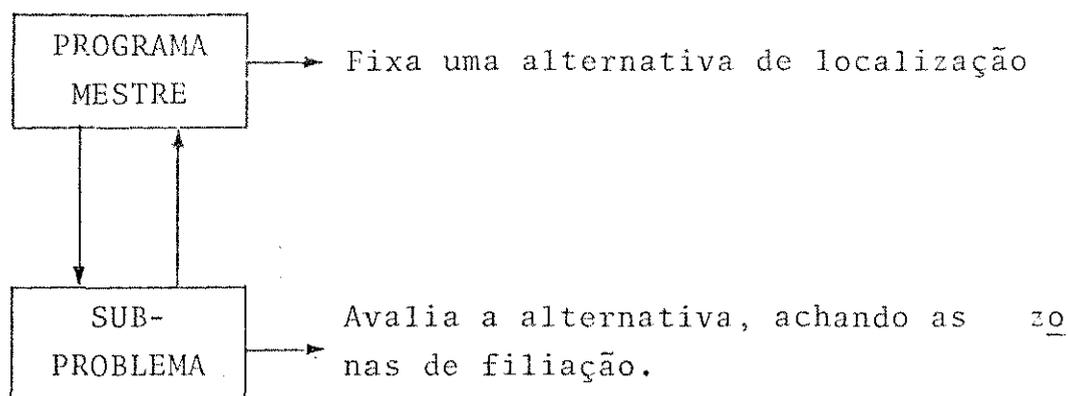


Figura 3 - Partição de Benders

À luz deste esquema, o que o programa mestre faz com (P) numa certa interação é fixar as variáveis z_j em alguma solução. Depois, o sub-problema precisa resolver um problema de fluxo de custo mínimo em (P).

CAPÍTULO II - UMA PROPOSTA DE SOLUÇÃO

2.1. INTRODUÇÃO

No capítulo 1, apresentamos uma rede telefônica urbana e alguns aspectos do planejamento de sua expansão. Uma etapa importante deste planejamento a longo prazo é aquela que determina a localização de novas centrais e/ou ampliação das existentes. Este problema, que chamaremos de PLCL, visa determinar, para o ano horizonte:

- . o número de centrais a construir ou ampliar
- . a localização e capacidade das novas centrais
- . a área de influência de cada central^(*)

de modo a minimizar os custos de cabos, construção (edificação, ar condicionado, energia), terreno e comutação.

Este capítulo se propõe a apresentar a técnica de partição de Benders que será empregada para resolver o PLCL; realçando a idéia de decomposição embutida no método de resolução.

Uma idéia particularmente atraente para resol

...

(*) No caso em que se considera a existência de uma rede atual satisfazendo a uma certa demanda, as áreas de influência das centrais podem aparecer "misturadas".

ver problemas mistos é adotar um procedimento de partição. As variáveis de fluxo são reais e as variáveis de decisão (localizar ou não) são binárias. Arbitramos uma localização, isto é, fixamos valores para z e, assim, teremos um problema linear de fluxo de custo mínimo nas variáveis reais x . A resolução deste problema nos fornece os fluxos nos arcos e o custo correspondente. Obtemos ainda as variáveis duais do Programa Linear, que nos dão uma indicação de como modificar a localização anterior. Determinada nova localização, novo programa de fluxo de custo mínimo é resolvido. Isto é repetido um certo número de vezes até que algum critério de parada seja atendido. Podemos imaginar o algoritmo consistindo de um PROGRAMA MESTRE que determina a localização (" z ") a partir da informação (variáveis duais " u ") fornecida pelo sub-problema (problema de determinação de fluxo de custo mínimo), conforme fig. 4.

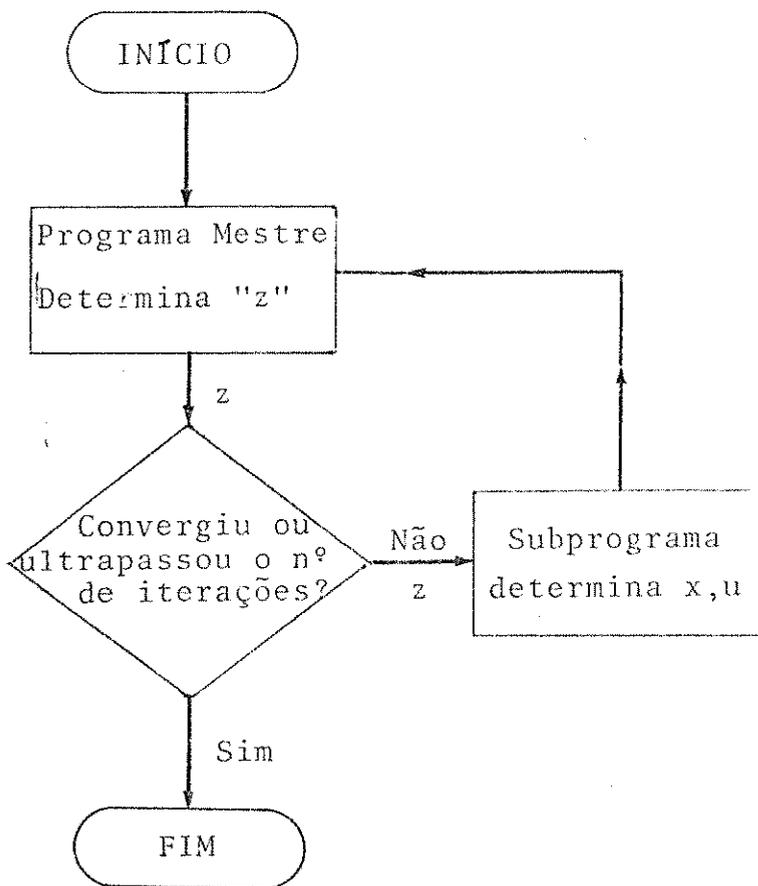


Figura 4 - Algoritmo de Partição de Benders

2.2. ALGORÍTMO DE PARTIÇÃO DE BENDERS

Desenvolver-se-á o procedimento de partição de Benders para um problema genérico (PG1) abaixo, visando com isto uma compreensão do método.

$$(PG\ 1) \quad \left\{ \begin{array}{l} \min \{cx + f(z)\} \\ \text{s/a } Ax + F(z) \geq b \\ x \geq 0 \\ z \in S \end{array} \right.$$

c - vetor $1 \times n$

x - vetor $n \times 1$

z - vetor $p \times 1$

A - matriz $m \times n$

b - vetor $m \times 1$

$f(z)$ - função escalar do vetor z

$F(z)$ - vetor de m componentes; sendo cada componente uma função escalar do vetor z .

As funções f , F podem ser não lineares. O conjunto S pode ser, por exemplo, um subconjunto de E^p , com componentes binárias (zero ou um). Porém, vamos supor, desde já, que:

1. S é um conjunto fechado e limitado
2. As funções f , F são contínuas em S .

Assim, temos um problema de programação mista

com variáveis reais não negativas x e variáveis binárias z . Aqui a idéia consiste em fixar valores para z_j , $j = 1, 2, \dots, p$, e resolver o programa linear em x , até que algum critério de parada seja atendido.

Devemos fixar vetores $z \in S$ tal que o problema resultante seja factível. Assim, determinaremos o conjunto R dos vetores z , factíveis.

$$R = \{z, z \in S \text{ e } \exists x \geq 0 / Ax \geq b - F(z)\},$$

vamos considerar que $R \neq \emptyset$.

2.2.1. Determinação do Conjunto R

Procuramos valores de z tal que seja possível encontrar $x \geq 0$ que satisfaça a inequação

$$(S_1) \quad Ax \geq b - F(z).$$

Vamos empregar o Lema de Farkas para determinar um outro sistema (S_2) tal que

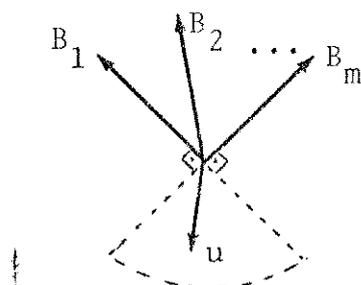
$$(S_1) \text{ admite solução} \iff (S_2) \text{ admite solução}$$

Assim poderemos determinar o conjunto R pelos valores de z tal que o sistema (S_2) tenha solução.

Considerando o Lema de Farkas, podemos afirmar que:

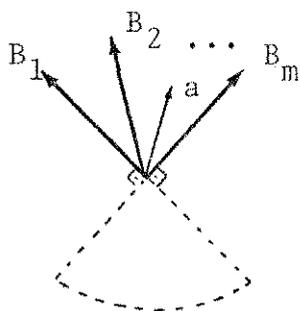
$$(1) \quad x \geq 0 / Bx = a \iff (2) \quad ua \leq 0 \text{ para } \forall u \text{ satisfazendo } uB \leq 0$$

Para ilustrar este fato vamos admitir que (2) tem solução. Seja $B = (B_1, B_2 \dots B_m)$ uma matriz representada pelos seus vetores colunas. Imaginando que estes vetores formem o cone representado na figura abaixo, o vetor \underline{u} estará contido no cone complementar



Como devemos ter

$\underline{u} \cdot \underline{B}_i \leq 0$ para $\forall \underline{u} / \underline{u} \cdot \underline{B}_i \leq 0$, temos que o vetor \underline{a} estará no cone formado pelos vetores $\underline{B}_1, \underline{B}_2, \dots, \underline{B}_m$.



Assim, podemos escrever

$$\underline{a} = \sum_{i=1}^m x_i \underline{B}_i, \quad x_i \geq 0 \quad i = 1, 2, \dots, m \quad \text{ou}$$

$$(1) \quad \underline{B}\underline{x} = \underline{a}, \quad \underline{x} \geq 0$$

Aplicando esta idéia ao sistema (S_1) , temos:

$$(S_1) \quad Ax \geq b - F(z) \\ x \geq 0$$

Após a introdução de variáveis de folga

$$Ax - \bar{x} = b - F(z) \\ x \geq 0, \bar{x} \geq 0$$

Definindo $C = \{u, u \geq 0, u(A-I) \leq 0\}$, um cone poliédrico, podemos escrever (S_2) , o sistema equivalente como:

$$(S_2) \quad u(b - F(z)) \leq 0 \quad \forall u \in C$$

Assim temos que procurar a factibilidade do sistema original (S_1) é equivalente a procurar a factibilidade do sistema (S_2) .

Sendo o conjunto R definido como

$$R = \{z \in S / u(b - F(z)) \leq 0, \forall u \in C\}$$

e sendo u_i^r , $i=1, 2, \dots, n_r$ os raios extremos do cone C , tais que para $\forall u \in C$, temos

$$u = \sum_{i=1}^{n_r} \lambda_i u_i^r, \quad \lambda_i \geq 0 \quad i = 1, 2, \dots, n_r$$

Podemos escrever $u \in C$ e $u(b - F(z)) \leq 0$ da seguinte forma

$$\sum_{i=1}^{nr} \lambda_i u_i^r (b-F(z)) \leq 0, \lambda_i \geq 0, i = 1, 2, \dots, n_r$$

Dado $\lambda_i \geq 0$, podemos reescrever R como:

$$R = \{z, z \in S / u_i^r (b-F(z)) \leq 0, i = 1, 2, \dots, n_r\}$$

2.2.2. Projeção em Z

Retornaremos à idéia inicial de fixar Z e resolver (PG1). Reescrevendo (PG1) como:

$$(PG2) \min_{Z \in R} \{f(z) + \min_{x \geq 0} \{cx / Ax \geq b-F(z)\}\}$$

e fixando z, na minimização interna, podemos resolver

$$\text{primal} \left\{ \begin{array}{l} \min cx \\ \text{s/a } Ax \geq b-F(z) \\ x \geq 0 \end{array} \right. \quad \text{ou} \quad \text{dual} \left\{ \begin{array}{l} \max u(b-F(z)) \\ \text{s/a } uA \leq c \\ u \geq 0 \end{array} \right.$$

Considerando o problema dual, reescrevemos (PG2)

$$(PG3) \min_{Z \in R} \{f(z) + \max_{u \geq 0} \{u(b-F(z)) / uA \leq c\}\}$$

Seja $D = \{u / u \geq 0, uA \leq c\}$ o conjunto de res

restrições do dual e seja

$\{u_i^P, i = 1, 2, \dots, n_p\}$ seus pontos extremos.

O conjunto $\{u_i^R, i = 1, 2, \dots, n_r\}$ já definido, é formado pelos seus raios extremos.

- Se $D = \emptyset$, o problema primal é ilimitado, e também o problema original (PG1)
- Se $D \neq \emptyset$, o dual terá solução finita. Se o problema dual fosse ilimitado, o problema primal seria infactível, o que não é possível, visto que a restrição $z \in R$ assegura sua factibilidade.

Sabendo que a solução de um programa linear limitado é um dos pontos extremos do poliedro de restrições, podemos então reescrever (PG3) como:

$$(PG4) \min_{z \in R} \{f(z) + \max_{\substack{u_i^P \\ 1 \leq i \leq n_p}} \{u_i^P (b - F(z))\}\}$$

A maneira direta de resolver (PG4) seria fixar um z factível, calcular $u_i^P(b - F(z))$ para $i = 1, 2, \dots, n_p$, e determinar o maior deles. A seguir, somar este valor a $f(z)$ e guardá-lo. Repetir-se-ia o procedimento para todos os z factíveis e escolher-se-ia aquele que resultou no menor valor de (PG4). Se o número de z factíveis fosse muito grande, este procedimento direto e simples tornar-se-ia inviável. Su

pondo que $z_j = 0,1 \quad j = 1,2,\dots,k$ teríamos, salvo restrições adicionais, 2^k combinações possíveis de z . Supondo $k = 30$, seriam aproximadamente 10^9 possibilidades.

Uma alternativa para se evitar esta enumeração completa é considerar o problema equivalente, onde T representa o menor limitante superior.

$$\begin{aligned}
 \text{(PG5)} \quad & \min T \\
 \text{s.a} \quad & f(z) + u_i^P (b - F(z)) \leq T \quad i = 1, 2, \dots, n_p \quad (1) \\
 & u_i^R (b - F(z)) \leq 0 \quad i = 1, 2, \dots, n_r \quad (2) \\
 & z \in S \quad (3)
 \end{aligned}$$

As restrições 2 e 3 exigem a factibilidade de z , impondo a necessidade de que $z \in R$.

Tendo chegado a (PG5), apresentaremos sem uma demonstração formal a equivalência entre (PG5) e (PG1)

Equivalência entre (PG1) e (PG5)

a) (PG5) tem uma solução factível \iff (PG1) tem uma solução factível

b) Se (z^*, T^*) resolve (PG5) e x^* resolve

$$\begin{cases} \min cx \\ \text{s/a } Ax \geq b - F(z^*), x \geq 0 \end{cases}$$

então (x^*, z^*) resolve (PG1)

c) Se (x^*, z^*) resolve (PG1) e $T^* = cx^* + f(z^*)$ então (z^*, T^*) resolve (PG5).

2.2.3. Estratégia de Relaxação

Resolver (PG5) apresenta a dificuldade de se de terminar todos os pontos e raios extremos do conjunto de restrições D. A estratégia de relaxação consiste em conside rar um Problema Relaxado (PGR) contendo apenas alguns pon tos e raios extremos.

$$\text{PGR} \quad \begin{cases} \min T \\ \text{s/a } f(z) + u_i^P (b - F(z)) \leq T \quad i \in I_P, I_P \{1, 2, \dots, n_p\} \\ u_i^R (b - F(z)) \leq 0 \quad i \in I_R, I_R \{1, 2, \dots, n_R\} \\ z \in S \end{cases}$$

Seja (z^k, T^k) , uma solução de (PGR)

Dois casos são possíveis:

- 1) Se (z^k, T^k) satisfaz as outras restrições de (PG5) não incluídas em (PGR), então (z^k, T^k) resolve (PG5). Neste caso $z^* = z^k$ e resolvendo

$$\begin{aligned} &\min cx \\ &\text{s/a } Ax \geq b - F(z^*) \\ &x \geq 0 \end{aligned}$$

teremos x^* e portanto (x^*, z^*) é a solução ótima de (PG1)

- 2) Se (z^k, T^k) viola alguma restrição de (PG5) procuramos de terminar a restrição mais violada. Será aquela em que $\{f(z^k) + u_i^P (b - F(z^k)), i = 1, 2, \dots, n_p\}$ é máximo. Como z^k está determinado, $f(z^k)$ é constante e, portanto, tenta-

-se determinar u_k^p , tal que:

$$u_k^p (b-F(z^k)) = \max_{1 \leq i \leq n_p} u_i^p (b-F(z^k))$$

Assim a restrição a ser acrescentada é:

$$f(z) + u_k^p (b-F(z)) \leq T$$

Caso o problema $\max_{1 \leq i \leq n_p} u_i^p (b-F(z^k))$ seja ilimitado, então

alguma restrição do tipo (2) de (PG5) (restrição de raio extremo), foi violada por não estar sendo considerada em (PGR). Determinamos então o raio extremo u_k^r e acrescentamos a seguinte restrição em (PGR)

$$u_k^r (b-F(z)) \leq 0$$

2.2.4. O Algoritmo

Recapitulando, queremos resolver:

$$\min \{cx + f(z)\}$$

$$s/a \ Ax + F(z) \geq b$$

$$x \geq 0$$

$$z \in S$$

S é um conjunto fechado e limitado.

f, F funções contínuas em S.

O algoritmo para resolver (PG1) compreende um Programa Mestre que é o Programa Relaxado (PGR) e um subproblema que é um problema linear na versão primal ou dual.

Subproblema

Primal	Dual
$\left\{ \begin{array}{l} \min cx \\ \text{s/a } Ax \geq (b-F(z^k)) \\ x \geq 0 \end{array} \right.$	$\left\{ \begin{array}{l} \max u(b-F(z^k)) \\ \text{s/a } uA \leq c \\ u \geq 0 \end{array} \right.$

Na iteração k , resolvendo o problema Primal, sua solução x^k nos dará cx^k .

Resolvendo o problema dual, sua solução u_k^p nos fornecerá $u_k^p(b-F(z^k))$ e o raio extremo u_k^r se for ilimitado.

Em programação linear, o valor ótimo do problema primal é o mesmo do problema dual.

Resolvendo o problema primal, as variáveis duais serão os multiplicadores das restrições que serão empregados para gerar nova restrição do problema relaxado.

No caso em que o Primal é infactível, obtemos dele o raio extremo do conjunto de restrições do Dual u_k^r .

Programa Mestre

(PGR)

$$\begin{aligned} & \text{Min } T \\ & \text{s/a } f(z) + u_i^P (b-F(z)) \leq T \quad i \in I_p, I_p \{1, 2, \dots, n_p\} \\ & u_i^R (b-F(z)) \leq 0 \quad i \in I_r, I_r \{1, 2, \dots, n_r\} \\ & z \in S \end{aligned}$$

Resolvendo (PGR), na iteração k teremos a solução (z_k, T_k) .

Nomenclatura do Fluxograma da Figura 5

LS : Custo da melhor solução encontrada até a presente iteração. É pois um limitante superior para o valor ótimo.

XS, ZS: Valores x e z correspondentes ao LS

PP : Problema Primal

PD : Problema Dual

VP^k : Valor do primal na iteração K

VD^k : Valor do dual na iteração K.

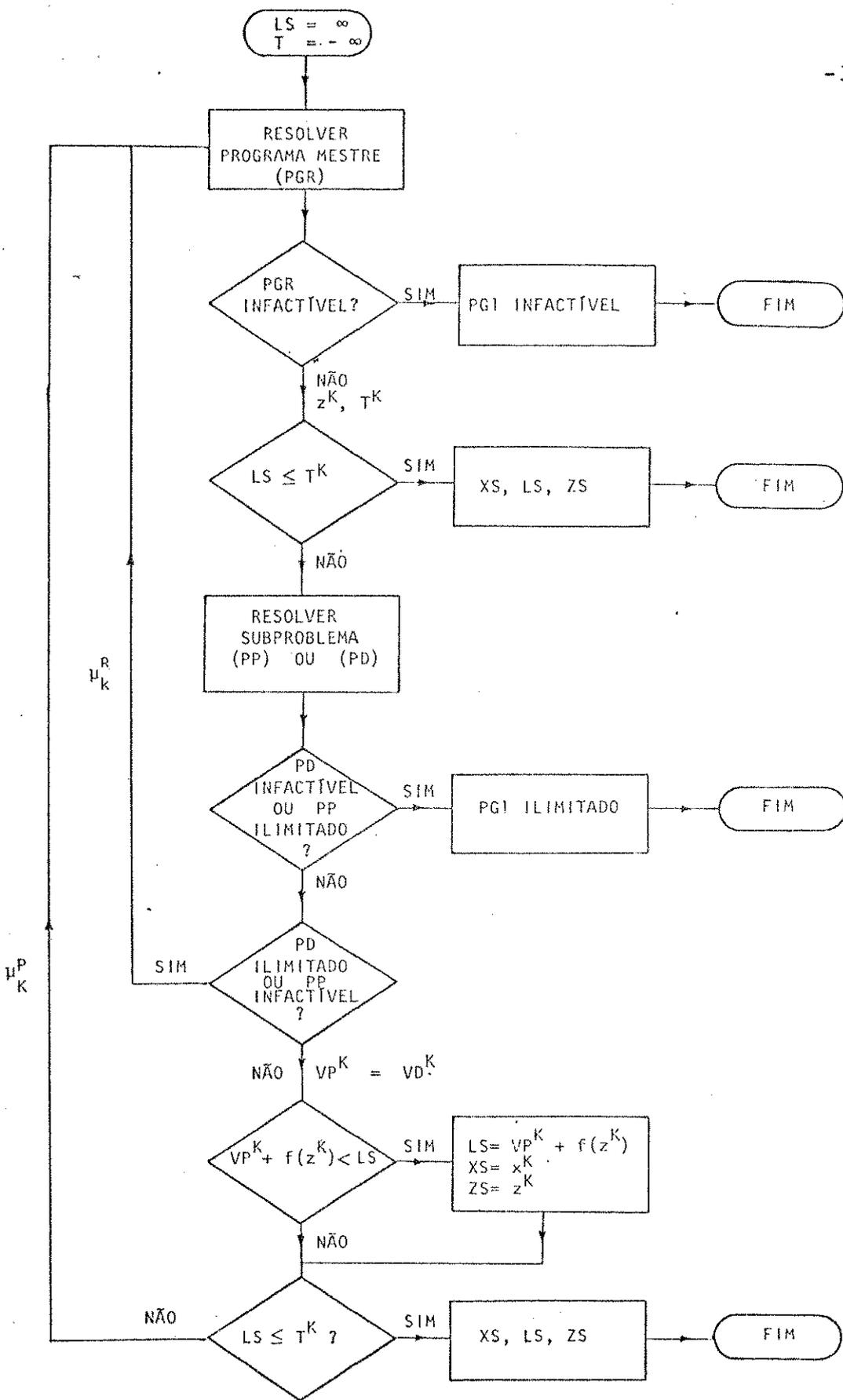


Figura 5 - Fluxograma de solução do PLCL - Algoritmo de Benders

2.2.5. Convergência do Algoritmo

O procedimento definido pelo algoritmo acima terminará após um número finito de iterações. A convergência finita é assegurada pelo fato de que o número de pontos extremos n_p e de raios extremos n_r do politopo D , do conjunto de restrições do Dual é finito. À medida que o algoritmo vai iterando, o (PGR) vai ficando mais restrito, aproximando-se do problema (PG5). Se começarmos com um (PGR) sem nenhuma restrição, após no máximo $n_p + n_r$ iterações teremos a solução do problema original (PG1). No fim do algoritmo um dos três casos pode ocorrer:

1) (PG1) é infactível:

Neste caso, (PGR) será infactível em alguma iteração.

2) (PG1) é ilimitado:

Resolvendo o Dual perceberemos a infactibilidade deste na primeira iteração.

3) (PG1) tem solução ótima finita:

Após resolver o sub-problema, se o teste de otimalidade falhar, acrescentamos uma restrição ao (PGR). Novamente, vemos que no pior caso a solução será obtida quando (PGR) = (PG5).

Como o problema relaxado é cada vez mais restrito, sua função objetivo T será monótona não decrescente

e será uma avaliação por baixo do custo da localização de terminada pelo (PGR). O cálculo do custo real é feito pelo subproblema. Este custo pode aumentar ou diminuir de uma iteração para outra. A convergência se dá quando o valor avaliado pelo Programa Mestre se iguala a algum custo real calculado pelo subproblema.

Este procedimento, apresentado no fluxograma da figura 5, foi aplicada por ARAÚJO [1], para a solução do problema de localização de centrais locais, utilizando como função objetivo uma função linear em z . Neste caso, baseado em suas experiências computacionais, a função objetivo utilizada foi o lado esquerdo da última restrição acrescentada ao (PGR). No entanto esta opção foi feita sem testar outros casos.

Dentre os enfoques clássicos de programação inteira enumerativa - decomposição, plano de corte e teoria de grupos - a abordagem escolhida por ARAÚJO para resolução do problema mestre foi a primeira, baseada em experiências bem sucedidas para problemas de médio e grande portes, relatadas em GEOFFRION e MARSTEN [3], BEALE [4], BALINSKI [5], e SALKIN [6].

O algoritmo utilizado foi baseado no trabalho original de BALAS [7] e na versão melhorada de GEOFFRION [8].

Nos capítulos seguintes apresentamos propostas alternativas de resolução do Problema Mestre e subproblema, àquelas sugeridas em [1].

CAPÍTULO III - RESOLUÇÃO DO PROBLEMA MESTRE ATRAVÉS DE UM MÉTODO HEURÍSTICO

3.1. INTRODUÇÃO

A dificuldade de resolução de problemas mistos, variáveis inteiras e reais, é um velho desafio dentro da programação matemática e tende a se agravar com o aumento do número de variáveis inteiras presentes.

O número de iterações, para o método de Benders atingir a convergência, tende a aumentar quando o número de variáveis zero-um aumenta. Assim, podemos concluir que um grande esforço computacional é exigido para a resolução do problema mestre, à medida que o número de variáveis tende a crescer.

Mesmo considerando a simplicidade algorítmica dos métodos de enumeração implícita, verificamos que a convergência do método se torna bastante demorada à medida que cresce o número de variáveis inteiras.

Observações como as anteriores motivaram o desenvolvimento de um método de Benders heurístico, buscando resolver o problema mestre através de uma rotina zero-um heurística sem degenerar, substancialmente, o critério de otimalidade do método. Assim, o desenvolvimento desta rot

na visou ter em mãos uma ferramenta que permitisse um procedimento de otimização computacionalmente mais rápido e capaz de tratar redes maiores ou redes com as quais o planejador tenha pouca familiariedade.

As grandes ressalvas à utilização de métodos heurísticos residem na sua fragilidade e característica de gerar soluções próximas às ótimas, porém não ótimas.

Por outro lado, a imprecisão dos dados reais obtidos na prática irá, dificilmente, justificar o incremento de custo de uma solução exata, bastando, ao planejador, em problemas de grande porte, a obtenção de soluções muito próximas da ótima.

Assim, baseado num estudo de comparação experimental de métodos heurísticos de programação linear, elaborado por ZANAKIS[9], onde foram analisados os algoritmos propostos por SENJU-TOYODA[10], KOCHEMBERGER, Mc CARL e WYMAN[11] e HILLIER[12], optamos pelo método proposto por KOCHEMBERGER, McCARL e WYMAN em vista de sua simplicidade algorítmica.

Este capítulo relata o desenvolvimento e resultados obtidos com a aplicação do método.

São feitos testes computacionais com problemas clássicos da literatura com a rede telefônica da Cidade de São José dos Campos e a rede telefônica da Cidade de Curitiba, comparando os resultados com os obtidos por outros algo

rítmos exatos, inclusive o método de Balas.

3.2. Uma Heurística para Programação Linear Inteira - Método de Kochemberger, McCarl e Wyman

Na construção de modelos de programação inteira, frequentemente nos deparamos com restrições de que todas as variáveis devem assumir valores inteiros. Restrições inteiras são, geralmente, necessárias devido à natureza indivisível dos itens a serem otimizados. Assim, o problema a ser considerado é o de maximizar uma função objetivo linear, dentro de um conjunto de restrições lineares, sujeito à restrição adicional que todas as variáveis sejam binárias (0 ou 1).

O problema tem, então, a seguinte configuração:

$$\begin{aligned} \text{Max } & \sum_{j=1}^n c_j x_j \\ \text{s/a } & \sum_{j=1}^n a_{ij} x_j \leq b_i \quad i = 1, 2, \dots, m \\ & x_j = 0 \text{ ou } 1 \quad j = 1, 2, \dots, n \end{aligned}$$

onde:

m : número de restrições

n : número de variáveis

c_j : ganho associado à j-ésima variável

a_{ij} : recurso requerido por unidade de variável j na

i-ésima restrição.

b_i : recurso disponível na i-ésima restrição.

Esta heurística propõe um método simples de obtenção de uma boa solução factível para problemas de programação linear inteira ou 0-1, os quais através de manipulações simples podem ser colocados na forma apresentada.

A idéia básica do método é a seguinte:

- o vetor solução inicial é constituído por todas as variáveis iguais a zero. A seguir, as variáveis são incrementadas, uma por vez, com base no aumento da função objetivo, provocado por unidade de factibilidade consumida. Convém salientar, que embora não esteja explícito, o método foi concebido para trabalhar apenas com a_{ij} e b_i não negativos, caso que ocorre no problema abordado.

3.3. Resumo do Procedimento

1. Coloque o problema na forma apresentada.
2. Faça a solução inicial $x_j = 0$ para todo $j = 1, 2, \dots, n$
3. Calcule \hat{b}_i , folga ou excesso da i-ésima restrição

$$\hat{b}_i = b_i - \sum_{j=1}^n x_j a_{ij} \quad i = 1, 2, \dots, m$$

4. Calcule \hat{a}_{ij} , porção de folga ou excesso, consumida pela variável x_j na i -ésima restrição.

$$\hat{a}_{ij} = a_{ij}/\hat{b}_i \quad \begin{array}{l} i = 1, 2, \dots, m \\ j = 1, 2, \dots, n \end{array}$$

5. Sendo $a_{ij} \geq 0$, calcule R_j , porção total de folga ou excesso consumida pela variável x_j (em todas as restrições) ou a porção de factibilidade consumida por x_j

$$R_j = \sum_{i=1}^m \hat{a}_{ij} \quad j = 1, 2, \dots, n$$

6. Calcule \hat{c}_j , variação na função objetivo provocada pela variável x_j , por unidade de factibilidade consumida

$$\hat{c}_j = c_j/R_j \quad j = 1, 2, \dots, n$$

7. Escolha o maior \hat{c}_j , ou seja, a variável que dá o maior incremento ao valor da função objetivo por unidade de factibilidade consumida.

8. Avalie se a variável x_j associada ao maior \hat{c}_j não negativo, quando incrementada torna o problema infactível.

8.a. Se o problema se torna infactível, então retorne ao passo 7, escolhendo o próximo maior c_j não negativo.

8.b. Caso contrário, incremente a variável em questão.

9. Volte ao passo 3 se a variável for incrementada, caso contrário, termine o procedimento.

Devemos observar que como o algoritmo foi desenvolvido para programação inteira, ao ser utilizado em problemas de programação linear zero-um deve-se ter o cuidado de proibir o procedimento de incrementar cada variável de mais de uma unidade. Esta técnica também é utilizável em problemas mistos (variáveis inteiras e reais) meramente incrementando as variáveis reais de quantidades fracionárias quando incrementos inteiros não forem possíveis.

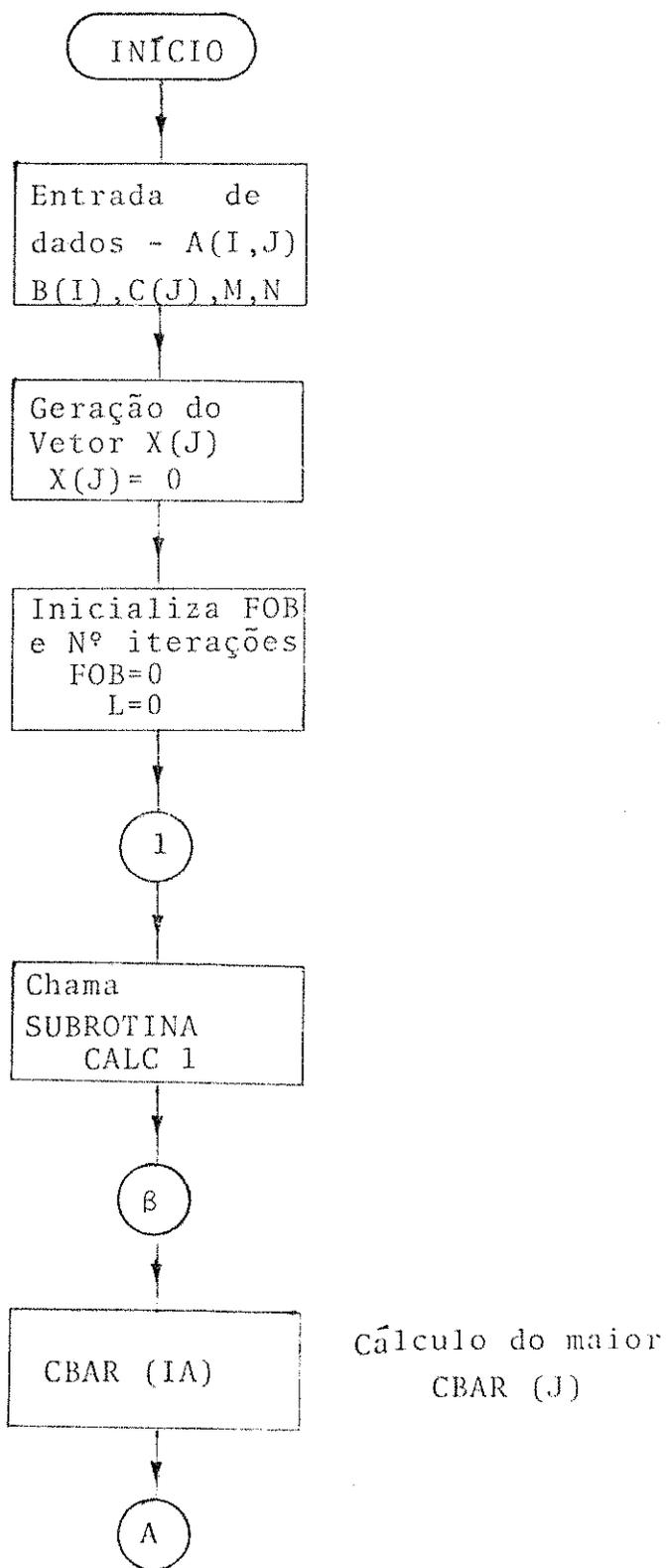
Fluxograma Resumido do Método

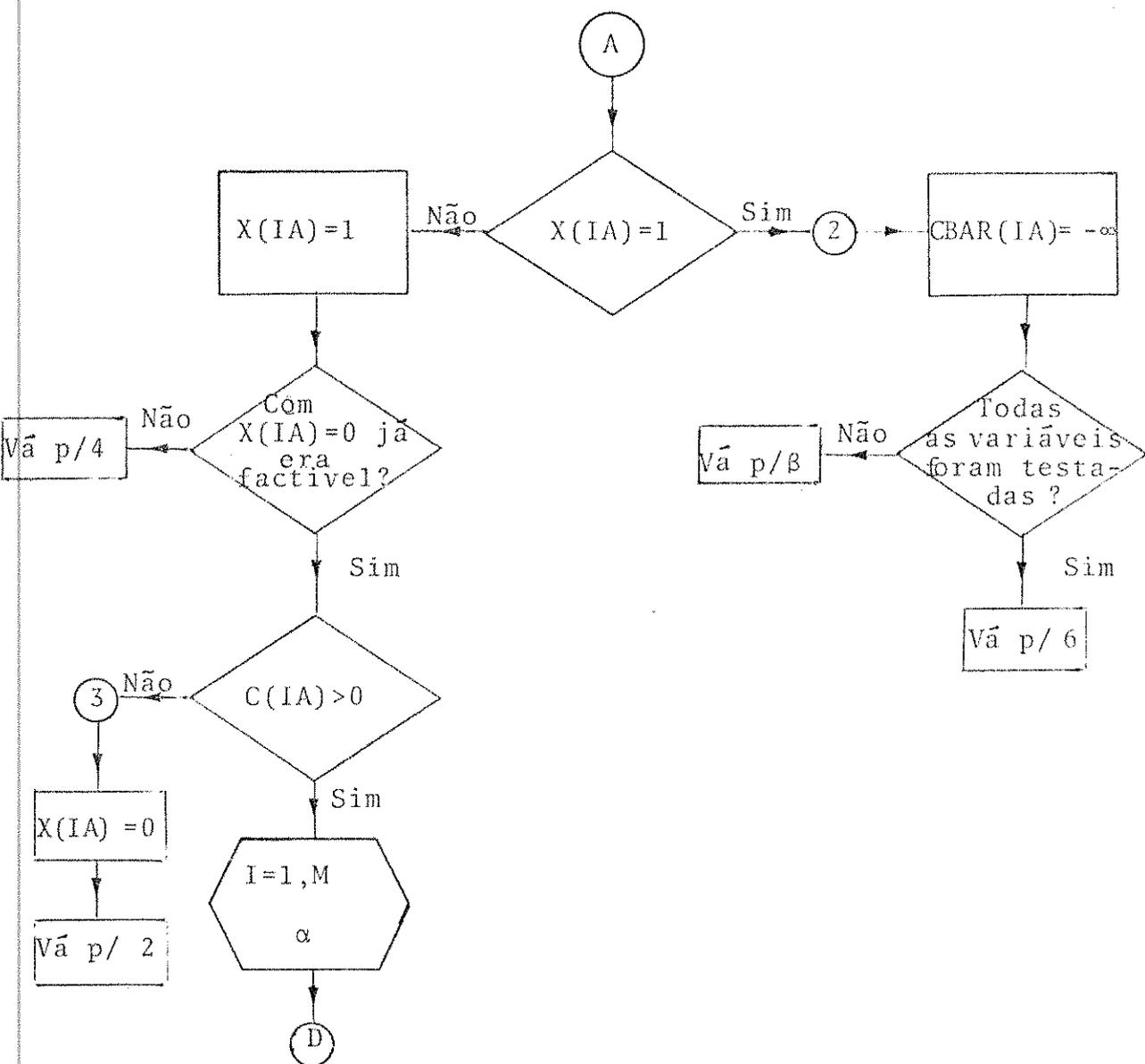
Lista de variáveis utilizadas:

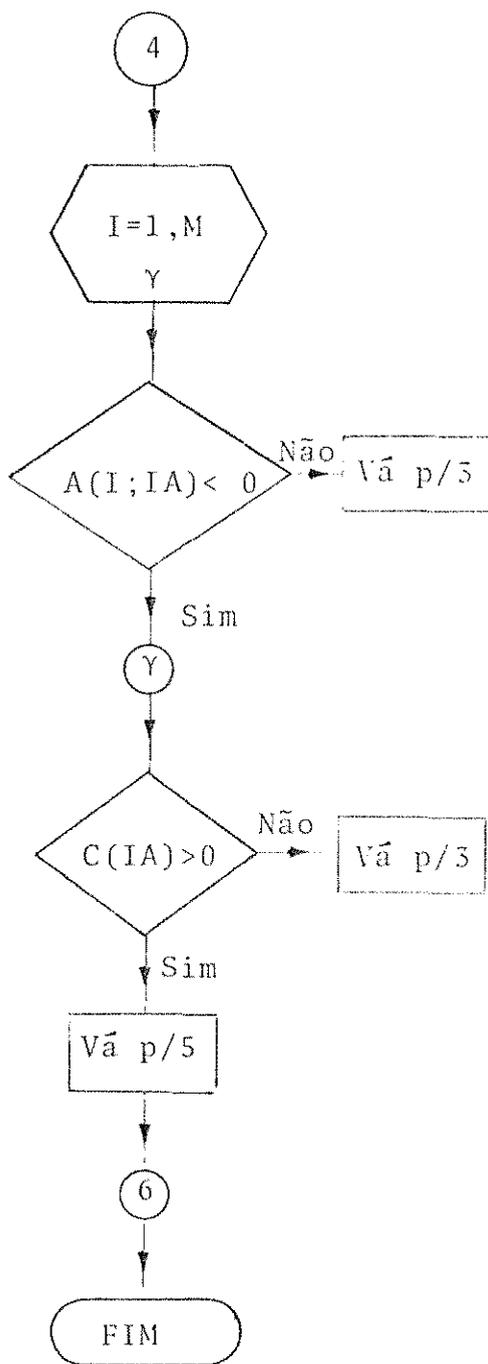
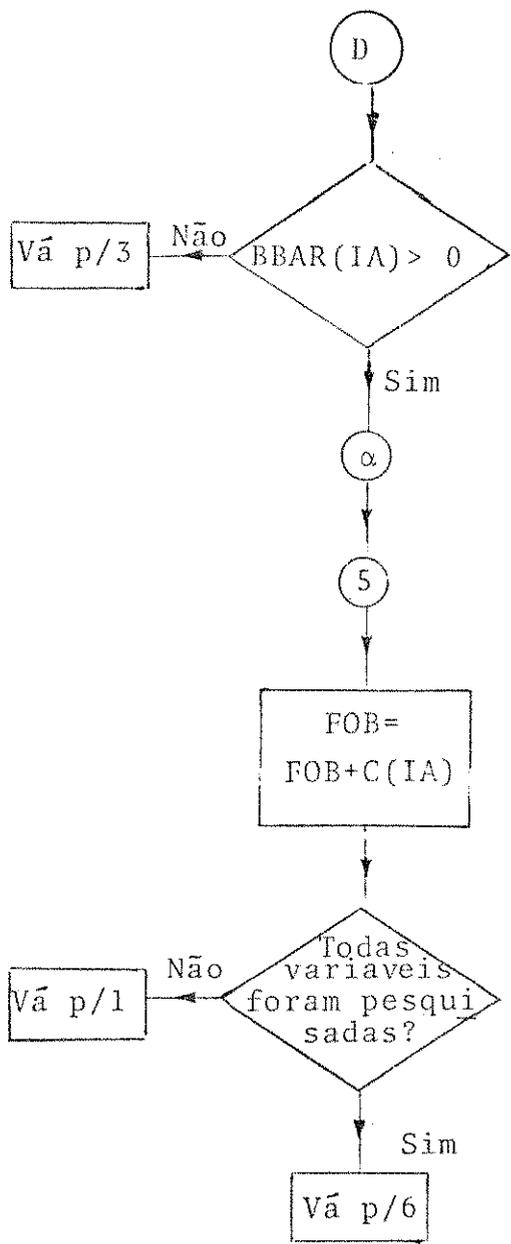
- M : Número de restrições do problema mestre geradas pelo sub-problema
- N : Número de variáveis do problema mestre
- I : Conjunto de índices das restrições $I=1,2,\dots,M$
- J : Conjunto de índices das variáveis $J=1,2,\dots,N$
- $A(I,J)$: Matriz de coeficientes do problema mestre
- $B(I)$: Vetor de recursos disponíveis do problema mestre
- $C(J)$: Vetor de ganhos do problema mestre, ou seja, os coeficientes da última restrição gerada pelo sub-problema (último corte)
- $X(J)$: Vetor solução; $X_j = 0$ ou 1
- FOB : Valor da função objetivo

- IA : Índice da variável candidata a ser incrementada numa iteração
- BBAR(I) : Vetor de folgas das restrições
- ABAR(I,J) : Matriz da porção de folga consumida, por variável, em cada restrição.
- CBAR(J) : Vetor que contém na posição j, o aumento provocado na função objetivo, pela variável x_j , por unidade de factibilidade consumida, numa iteração.
- SOMAT1 : Função que calcula a quantidade de recurso consumido pela solução X(J) na restrição I
- SOMAT2 : Função que calcula a porção total de folga consumida pela variável j ou ainda a porção de factibilidade por ela consumida
- CALC1 : Subrotina que avalia os valores de BBAR(I), ABAR(I,J) e CBAR(J)

Programa Principal







Subroutine Calc 1

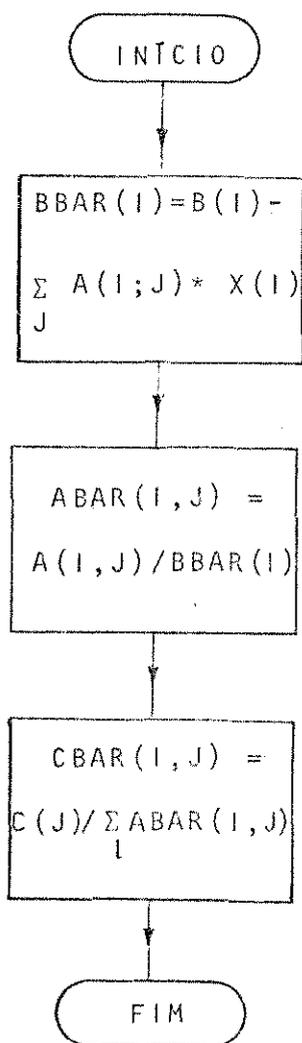


Figura 6 - Fluxograma Resumido do Método Heurístico

3.4. APLICAÇÃO A PROBLEMAS TESTES

A fim de testar a implementação do algoritmo heurístico, este foi aplicado a problemas clássicos da literatura.

Os problemas testes foram sugeridos por PETERSEN[13] WEINGARTNER[14] e SENJUO TOYODA[10], cujos resultados são apresentados na Tabela 1.

Calcula-se a porcentagem de erro entre o valor da função objetivo obtido pela heurística e o obtido pelos métodos exatos, como medida de eficiência.

Pode-se observar que mesmo sendo levadas em conta as diferenças em rapidez computacional, o tempo gasto pela heurística é muito menor que o tempo necessário para a solução do problema através dos métodos exatos examinados. Somado a isto, pode-se observar que a porcentagem de erro entre o valor da função objetivo obtido pela heurística e o obtido pelos métodos exatos nunca foi superior a 3%, justificando para a solução destes problemas, o seu uso.

TABELA 1 - Resultados obtidos em problemas testes extraídos da literatura

PROBLEMA	Nº VARIÁVEIS	Nº RESTRIÇÕES	TEMPO CPU (SEG) ⁺		FUNÇÃO OBJETIVO		PE%
			ÓTIMO	HEURÍSTICA	ÓTIMO	HEURÍSTICA	
1.Petersen N.6	39	5	300 ⁽¹⁾	1,087	10.618	10.313	2,87
2.Petersen N.7	50	5	2.160 ⁽¹⁾	1,467	16.537	16.031	3,0
3.Weingartner N.1	28	2	1,8 ⁽²⁾	0,273	141.548	141.548	0
4.Weingartner N.2	105	2	24 ⁽²⁾	3,395	1.095.445	1.095.352	0,008
5.Senju-Toyoda A	60	30	84 ⁽³⁾	5,761	7.772	7.761	0,14
6.Senju-Toyoda B	60	30	241 ⁽³⁾	8,29	8.772	8.640	0,94

(+) O tempo de CPU para heurística foi obtido com PDP-10

(1) A solução ótima foi obtida usando o algoritmo de Balas com o computador SDS-930

(2) A solução ótima foi obtida usando um algoritmo de programação dinâmica do computador IBM-7094

Utilizando outro algoritmo para a solução, os problemas 3 e 4 gastariam respectivamente 406 e 16 vezes mais tempo de CPU [9]

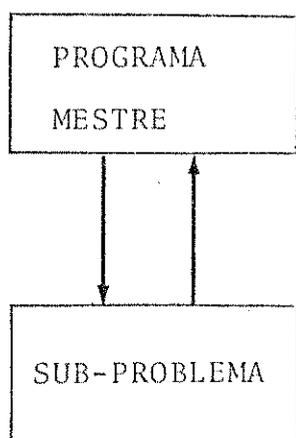
(3) A solução ótima foi obtida através do IBM/MPSX em um computador IBM 360/75

3.5. APLICAÇÃO À PROBLEMAS DE LOCALIZAÇÃO DE CENTRAIS LOCAIS

Como sabemos, a maioria dos problemas de localização de centrais locais encontrados na prática, são problemas de grande dimensão.

Devido a este fato e também por este problema estar sendo resolvido de forma conversacional, com a interferência do planejador na evolução da solução, optou-se por uma técnica de decomposição (partição de Benders), que através da avaliação sucessiva de diferentes alternativas de localização converge em direção a uma solução ótima.

A partição de Benders que divide a resolução do problema em duas partes, pode ser vista como um esquema de decomposição em dois níveis. Num primeiro nível resolve-se um problema que chamaremos de sub-problema.



Fixa uma alternativa de localização

Avalia a alternativa achando as zonas de filiação

À luz deste esquema, o que o programa mestre faz numa certa iteração é fixar as variáveis z_j em alguma solução. Depois o sub-problema avalia o custo desta solução resolvendo um problema de fluxo de custo mínimo.

O programa LOCUS desenvolvido por ARAÚJO enfoca a localização de centrais locais como um problema de otimização em busca do ótimo global.

O algoritmo heurístico foi implementado como uma rotina heurística de solução do problema mestre e inserida no programa LOCUS, gerando o programa LOCUS HEURÍSTICO. Utiliza-se a idéia de localizar as centrais em duas etapas. Na primeira etapa, baseado numa análise prévia da rede, seleciona-se um número de nós candidatos à central, seja por serem nós que concentram assinantes, seja pelo conhecimento da estrutura urbana da rede. Determinamos a solução ótima para estes candidatos e numa segunda etapa, chamada AJUSTE FINO, refinamos a localização.

No ajuste fino fazemos uma análise pós-otimização com o objetivo de ratificar a localização fornecida ou determinar uma melhor. Este procedimento se torna bastante conveniente tendo em vista que o método heurístico não garante o ótimo global, porém tem a vantagem de uma convergência mais rápida que os algoritmos de enumeração.

Deve-se observar que o conhecimento profundo do problema pode sugerir boas soluções iniciais, a partir das

quais o processo sendo iniciado reduz o número necessário de iterações.

Trabalhou-se com a rede telefônica da Cidade de São José dos Campos, de médio porte, composta por 327 nós e 364 arcos sem orientação. Após exaustiva análise da rede foram selecionados os candidatos à instalação da central.

A seguir, foi arbitrado um conjunto de localizações iniciais factíveis, aqui referidas como localizações de partida (LP), visando uma melhoria computacional.

Foram testados 8 casos com várias "LP" e com apenas uma "LP", com diversas configurações com o objetivo de comparar o desempenho do Programa LOCUS HEURÍSTICO frente ao Programa LOCUS.

Nos 6 primeiros casos foram utilizados 9 candidatos a centrais e nos dois últimos 15 candidatos.

Foram observados o número de iterações e a convergência à solução ótima utilizando, inicialmente, 10 "LP" diferentes para cada caso, e finalmente, os mesmos testes com apenas uma "LP", cujos resultados são apresentados nas tabelas 2 e 3.

De posse dos resultados das tabelas 2 e 3 foi possível observar que os dois programas se comportam de maneira diferenciada, o que nos permite fazer algumas considerações. Como já foi dito foram arbitradas localizações de partida factíveis, visando gerar restrições para o programa

TABELA 2 : Resultados obtidos nos testes com a rede de São José dos Campos, com 10 localizações de partida

CASOS	PROGRAMA LOCUS			PROGRAMA LOCUS-HEURÍSTICO			ERRO
	Nº ITERAÇÕES	(1) CONVERGIU	ITER.EM QUE OBTEM ÓTIMO	Nº ITERAÇÕES	(1) CONVERGIU	ITER.EM QUE OBTEM ÓTIMO	
1	21	SIM	(*)	13	SIM	(*)	0
2	24	SIM	22	15	SIM	13	0
3	23	SIM	(*)	13	SIM	(*)	0
4	21	SIM	18	18	NÃO	(*)	1.4
5	24	SIM	23	16	NÃO	16	1.4
5	23	SIM	(*)	14	SIM	(*)	0
7	70 (2)	NÃO	-	14	NÃO	11	13.1
8	70 (2)	NÃO	-	21	NÃO	13	9.0

(1) Convergência à solução ótima.

(2) O procedimento foi interrompido pois atingiu o número máximo de iterações permitidas.

(*) A solução ótima estava entre as localizações de partida.

TABELA 3 : Resultados obtidos nos testes com a rede de São José dos Campos, com 1 localização de partida

CASOS	PROGRAMA LOCUS			PROGRAMA LOCUS-HEURÍSTICO			% ERRO
	Nº ITERAÇÕES	(1) CONVERGIU	ITER.EM QUE OBTEM ÓTIMO	Nº ITERAÇÕES	(1) CONVERGIU	ITER.EM QUE OBTEM ÓTIMO	
1	22	SIM	(*)	15	SIM	(*)	0
2	23	SIM	15	12	NÃO	12	1.4
3	21	SIM	17	18	NÃO	7	3.1
4	22	SIM	13	11	NÃO	7	3.1
5	23	SIM	13	17	NÃO	6	3.1
6	21	SIM	15	12	NÃO	3	1.4
7	70	NÃO	-	9	NÃO	4	9.0
8	70	NÃO	-	20	NÃO	4	11.0

(1) Convergência à solução ótima.

(2) O procedimento foi interrompido pois atingiu o número máximo de iterações permitidas.

(3) A solução ótima estava entre as localizações de partida.

mestre. O objetivo deste procedimento é o de fornecer ao programa uma configuração inicial de localizações que permitisse uma mais rápida convergência.

Observando os dados das tabelas 2 e 3 verificou-se, com relação ao número de iterações, a não sensibilidade, dos dois programas, às localizações de partida.

Isto ocorre pois os dois programas trabalham com linearizações externas, tendo como função objetivo o último corte gerado (última restrição). Assim, os programas só seriam sensíveis às "LPs" se estas fossem em número muito grande, ou seja, quando os cortes (linearizações externas) fossem menos pobres, isto é quando as "LPs" já tivessem formado uma boa representação da função objetivo T.

Analisando a estrutura dos dois programas concluímos que o programa heurístico deveria incorrer em um maior número de iterações, uma vez que ele gera linearizações externas (restrições) mais pobres, contudo, o seu rígido critério de parada por infactibilidade compensa o número de iterações.

Ainda considerando as localizações de partida, é importante notar que uma análise prévia da rede em estudo pode sugerir boas localizações iniciais, o que pode favorecer a convergência do método heurístico, no que diz respeito a uma menor porcentagem de erro.

Quanto ao aspecto de convergência, considerando

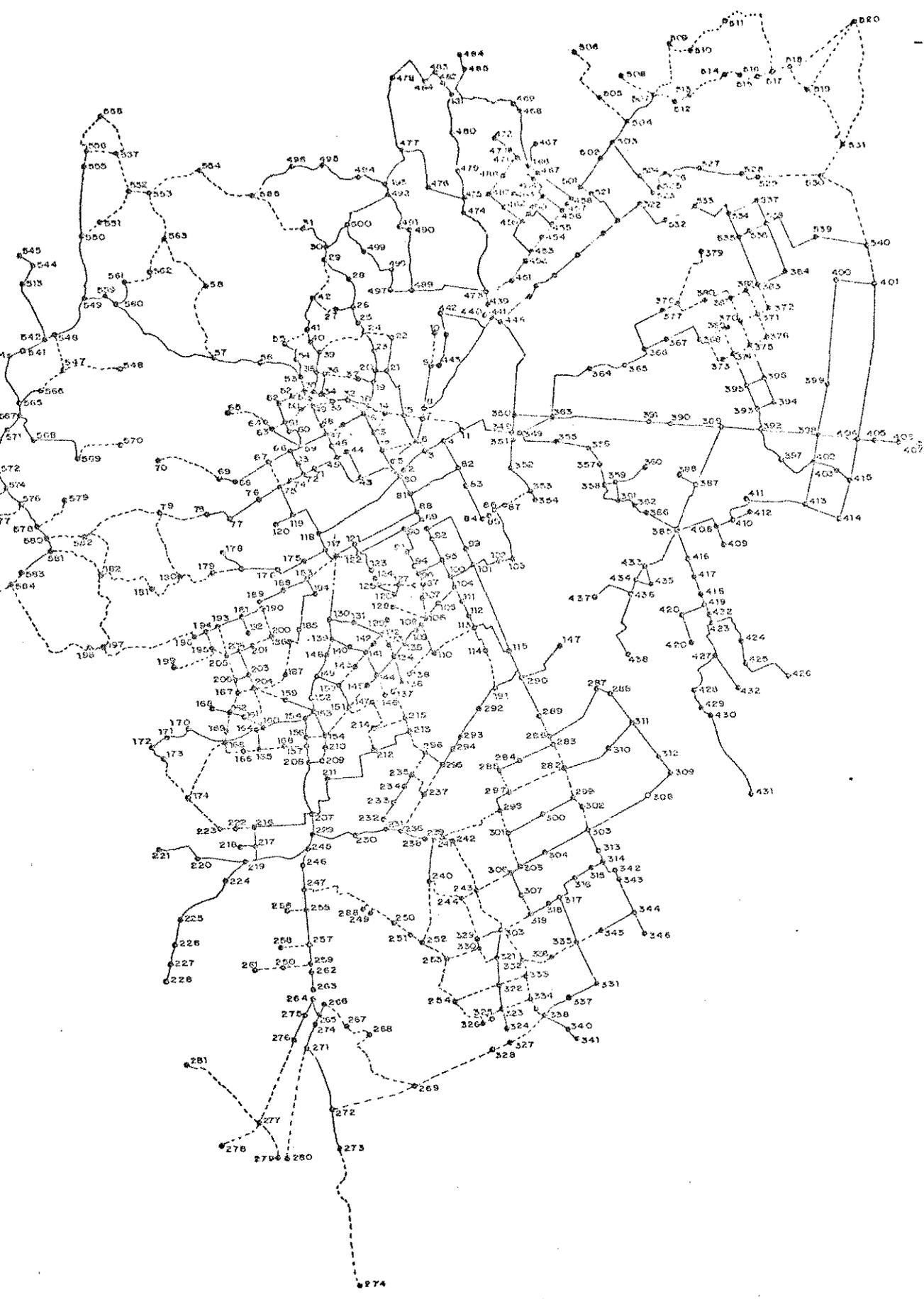


Figura 7 - Rede Telefônica Urbana de Curitiba

os testes realizados, verificamos que o programa heurístico só convergiu nos casos em que a solução ótima fazia parte do conjunto de soluções de partida. Nos casos em que o método heurístico não convergiu, a melhor solução obtida sempre se caracterizou por estar em um patamar, isto é, foi possível observar que ao chegar a uma configuração de localização muito próxima da ótima, a menos de um elemento seria a ótima, não houve mudança de solução nas iterações subsequentes. Isto se deve à estrutura rígida do algoritmo que dificulta excessivamente a troca deste único elemento que se desvia da solução ótima, o que leva o procedimento a terminar sem atingir o objetivo.

Com base na comparação entre os dois métodos, sugerimos a implementação de um algoritmo misto, que utilizaria tanto a rotina zero-um exata de BALAS, como a rotina zero-um heurística de KOCHEMBERGER, McCARL e WYMAN. Basicamente a rotina heurística seria utilizada até o momento em que se atingisse o patamar e, a partir de então, utilizar a rotina exata.

Aplicação a uma Rede de Grande Porte

O programa LOCUS-HEURÍSTICO foi utilizado na solução do problema de localização de centrais da rede telefônica urbana da Cidade de Curitiba, composta por 586 nós e 744 arcos sem orientação. (Figura 7)

Após análise prévia da rede foram escolhidos 20 candidatos a central conforme o apresentado na Figura 8.

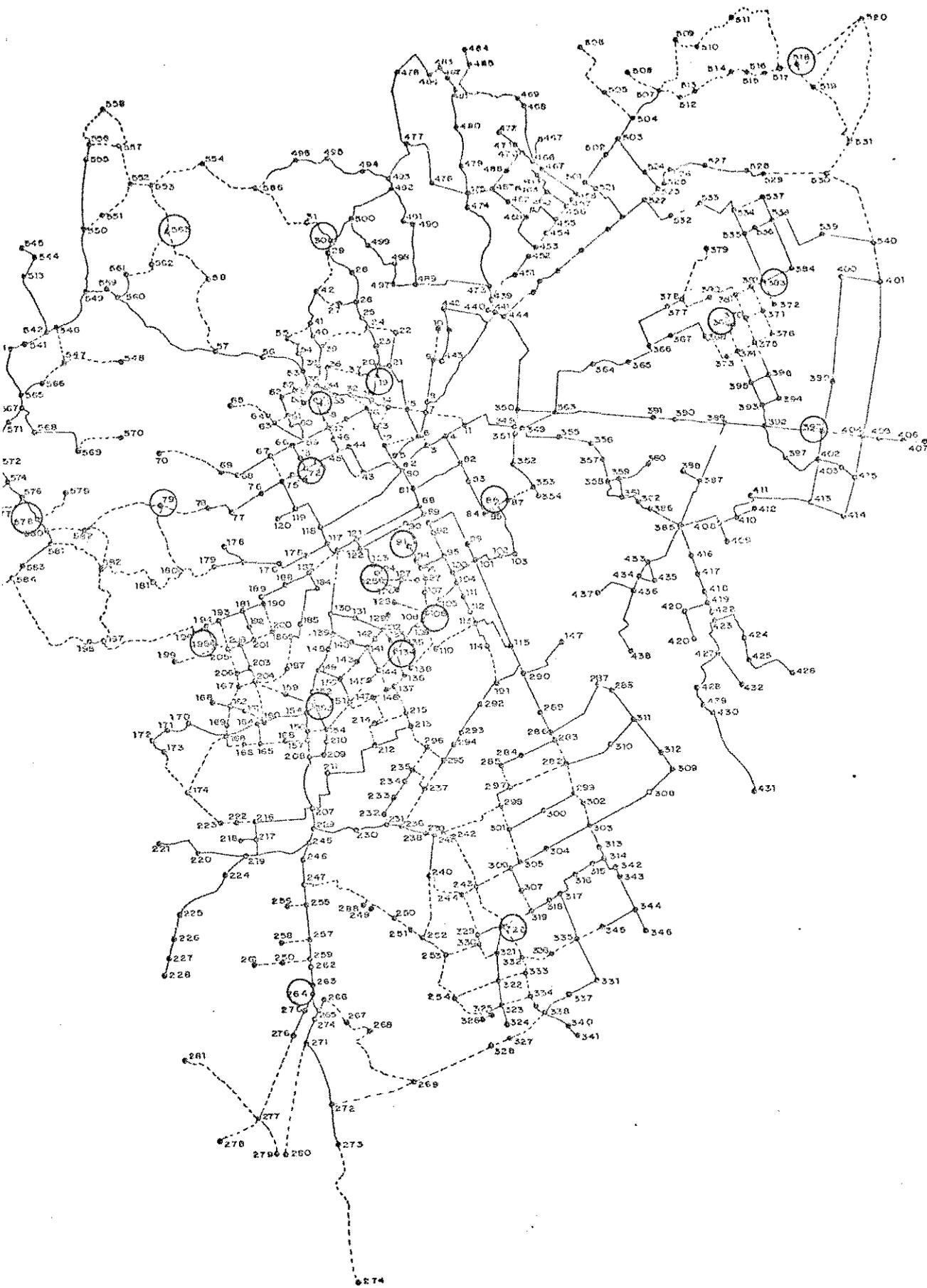


Figura 8 - Localização dos nós candidatos à central

Foi dada apenas a seguinte localização de partida

LOCALIZAÇÃO DE PARTIDA										
NÓS:	49	320	79	398	383	106	195	518	563	153

O número máximo de iterações foi limitado em 50 e o problema foi submetido à solução através dos dois métodos (exato e heurístico), cujos resultados são apresentados na tabela 4.

TABELA 4: Resultados obtidos com os programas LOCUS e LOCUS HEURÍSTICO na solução do problema de Curitiba.

	Nº DE ITERAÇÕES EFETUADAS	ITERAÇÃO EM QUE OBTEVE A MELHOR SOLUÇÃO	CUSTO DA MELHOR SOLUÇÃO (Cr\$)	POLÍTICA DE LOCALIZAÇÃO
LOCUS	50	18	79.309.601	49-320-79-398-134 -30-195-518
LOCUS				49-320-79-398-383
HEURÍSTICO	7	1	71.840.017	106-195-518-563- 153

Analisando a Tabela 4 verifica-se que o método heurístico não conseguiu gerar nenhuma política de localiza

ção melhor que a fornecida como localização de partida. Contudo podemos observar que sendo a "LP" uma solução muito boa, próxima da ótima, o programa heurístico, desde a primeira iteração se viu em um patamar interrompendo o procedimento com 7 iterações, assumindo como ótima a localização de partida, incorrendo em um erro de 2% sobre o custo da melhor solução obtida pelo método exato. Por outro lado, embora o método exato tenha obtido melhor política de localização não houve convergência com o número de iterações permitidas.

Resumindo toda a experiência podemos concluir que o método heurístico não obteve uma boa performance na solução de problemas com esta configuração, apresentando como principal demérito sua rigidez quanto a convergência.

Uma sugestão que julgamos melhorar o desempenho do método heurístico no que diz respeito ao número de iterações, seria utilizar como função objetivo uma representação de T, menor limitante superior, considerado em (PG5) no Capítulo 2.

CAPÍTULO IV - RESOLUÇÃO DO SUBPROBLEMA

4.1. INTRODUÇÃO

Nos Capítulos I e II apresentou-se a formulação do Problema (P) de Localização de Centrais Locais (PLCL), e sua abordagem segundo a Técnica de Partição de Benders.

À luz deste esquema vê-se que o sub-problema consiste em um problema de fluxo de custo mínimo em uma rede, sendo necessário, para sua resolução, um algoritmo adequado.

O procedimento de solução aqui utilizado foi baseado no trabalho de AUTHIÉ [15], cujo desenvolvimento é apresentado a seguir uma vez que seu conhecimento se torna imprescindível à compreensão dos capítulos seguintes.

4.2. FORMULAÇÃO DO PROBLEMA

O problema de fluxo de custo mínimo (linear) que aqui será tratado, pode ser formulado da seguinte maneira:

a) Seja uma rede constituída de n centros de tratamento de matéria e m ligações direcionais entre estes

centros, por onde a matéria transita. Pode-se associar a esta rede um grafo $G = (N, U)$, onde o conjunto N de nós corresponde aos centros e o conjunto U de arcos orientados corresponde às ligações direcionais entre os centros. A estrutura dessa rede fica inteiramente definida pela matriz de incidência $I(n, m)^*$ do grafo.

b) No grafo $G(N, U)$ cada nó i pode produzir, consumir ou simplesmente servir de passagem para a matéria. Seja a_i a quantidade de matéria associada ao nó i . Se $a_i > 0$ o nó i será produtor. Se $a_i < 0$, o nó i será consumidor, e se $a_i = 0$, será um nó de passagem. Fica assim definido o vetor de recursos $\underline{a} (n, 1)$.

c) É suposto que a rede é auto-suficiente, isto é, que $\sum_{i \in N} a_i = 0$. Caso num exemplo real exista um desbalançamento entre produção global e consumo global será necessário completar a rede com a introdução de um nó e alguns arcos, chamados de fechamento, por onde escoará o custo nulo, o "deficit" ou "superavit" de matéria.

d) Para satisfazer a conservação de matéria,
.../...

(*) É uma matriz onde as linhas estão associadas aos nós e as colunas aos arcos. A coluna correspondente ao arco $u = (i, j)$ terá todos os componentes nulos à exceção do i -ésimo componente que terá valor $+1$ e do j -ésimo componente que terá valor -1 .

em cada nó da rede, é preciso encaminhar a produção em direção aos centros de consumo por intermédio das ligações existentes. A quantidade de matéria escoada pelo arco u_k , chamada fluxo no arco é representada por x_k .

e) Por razões tecnológicas, o fluxo em cada arco u_k deve ser superior a um limite inferior b_k e não deve ultrapassar um limite superior c_k , chamado capacidade do arco. Para o conjunto de arcos essas restrições se escrevem $\underline{b} \leq \underline{x} \leq \underline{c}$, onde $\underline{b}(m,1)$ e $\underline{c}(m,1)$, representam os vetores de limites inferiores e superiores.

f) O custo de transportar uma quantidade x_k através do arco u_k é considerado proporcional à quantidade transportada constituindo-se, assim, numa função de custo linear $d^k x_k$. Para o conjunto de fluxos na rede define-se então um vetor de custos unitários $\underline{d}(1,m)$.

O problema aqui estudado consiste em determinar como escoar o produto dos centros de produção até os centros de consumo, através da rede existente, de modo a minimizar o custo total de transporte.

Matematicamente, o problema de fluxo de custo mínimo (linear) se apresenta como:

$$(1) \quad \left\{ \begin{array}{l} \min z = \underline{d}\underline{x} \\ \text{s/a } Ix = a \\ \underline{b} \leq \underline{x} \leq \underline{c} \end{array} \right.$$

Considerando tratar-se de um problema linear de grafos, utilizaremos algoritmos específicos para sua solução.

4.3. ALGORÍTMOS DE FLUXO

Base do Sistema

Inicialmente, é preciso caracterizar as bases do sistema $I\underline{x} = \underline{a}$. A matriz de incidência $I(n,m)$ apresenta a característica de possuir somente $(n-1)$ linhas linearmente independentes. Isto se deve ao fato de $\sum_{i \in N} I_i = 0$ onde I_i é o vetor linha da matriz de incidência I .

Exatamente motivado por essa dependência linear se exigiu, na formulação do problema, para efeito de consistência, que $\sum_{i \in N} a_i = 0$.

Reescrevendo o sistema, tem-se

$$\sum_{k \in U} I^k x_k = \underline{a} \quad (2)$$

onde I^k é o vetor coluna da matriz de incidência I .

Logo, uma base de (2) será composta de $(n-1)$ vetores colunas linearmente independentes.

Chamando o conjunto de índices associados aos

(n-1) vetores linearmente independentes de B , e \bar{B} seu complementar, podemos particionar o sistema (2) da seguinte forma

$$I^B \underline{x}_B + I^{\bar{B}} \underline{x}_{\bar{B}} = \underline{a} \quad (3)$$

As variáveis de fluxo x_k são chamadas variáveis básicas se $k \in B$ e não básicas se $k \in \bar{B}$.

O conjunto de arcos $u_B = \{u_k / k \in B\}$ permite caracterizar o grafo $H(N, u_B)$ associada à base I^B .

Solução Básica

O conjunto de restrições do problema linear (1) é:

$$\underline{I} \underline{x} = \underline{a} \quad \underline{b} \leq \underline{x} \leq \underline{c} \quad (4)$$

Um vetor \underline{x} satisfazendo (4) é chamado SOLUÇÃO FACTÍVEL de (1). Supondo conhecida uma base I^B pode-se particionar o sistema conforme (3). Atribuindo-se valores para as variáveis fora da base $\underline{x}_{\bar{B}}$ (independentes) pode-se determinar os valores das variáveis básicas \underline{x}_B (dependentes) pela resolução do sistema (3).

Particionando o conjunto de variáveis não básicas em dois subconjuntos complementares ($\bar{B} = \bar{B}_1 + \bar{B}_2$) define-se como solução básica do sistema $\underline{I} \underline{x} = \underline{a}$ relativa à base I^B , a solução do sistema que resulta da escolha das variáveis não básicas que segue:

$$\underline{x}_{\bar{B}_1} = \underline{b}_{\bar{B}_1}$$

$$\underline{x}_{\bar{B}_2} = \underline{c}_{\bar{B}_2}$$

Uma vez que a partição de \bar{B} em \bar{B}_1 e \bar{B}_2 é arbitrária, várias soluções básicas corresponderão à mesma base I^B . Uma solução BÁSICA FACTÍVEL será obtida quando as variáveis básicas satisfizerem a restrição

$$\underline{b}_B \leq \underline{x}_B \leq \underline{c}_B$$

Finalmente, uma solução básica factível de (1) que minimiza $z = \underline{d} \underline{x}$ é chamada SOLUÇÃO BÁSICA FACTÍVEL ÓTIMA.

Considerando o teorema fundamental de programação linear e considerando que as restrições (4) definem um conjunto compacto, pode-se garantir que se o problema (1) tem uma solução básica factível então terá solução básica factível ótima.

A busca da solução ótima é finita uma vez que irá se limitar à procura entre as soluções básicas factíveis, as quais são em número limitado.

O princípio do algoritmo SIMPLEX, para a resolução de problemas lineares, consiste justamente em passar de uma solução básica factível a uma outra, de tal sorte, que a função objetivo decresça a cada iteração até alcançar a solução ótima.

Para poder-se avaliar os efeitos que as mudanças, nas soluções básicas, causam sobre o valor da função objetivo, é preciso expressá-la em termos das variáveis não básicas. Para isso é fundamental a utilização do conceito de vetor multiplicador.

Vetor Multiplicador

Seja uma base I^B . Deseja-se exprimir a função objetivo em termos exclusivamente das variáveis não básicas, que são variáveis independentes.

Reescrevendo o sistema $I\bar{x} = \underline{a}$ e a função objetivo $z = \underline{d}\bar{x}$ de forma particionada em relação à base I^B , tem-se:

$$I^B \underline{x}_B + I^{\bar{B}} \underline{x}_{\bar{B}} = \underline{a} \quad (5)$$

$$\underline{d}^B \underline{x}_B + \underline{d}^{\bar{B}} \underline{x}_{\bar{B}} = z \quad (6)$$

Subtraindo-se de (6) uma combinação linear das linhas do sistema (5), onde o peso multiplicador de cada linha i é t_i , obtém-se:

$$(\underline{d}^B - \underline{t}I^B) \underline{x}_B + (\underline{d}^{\bar{B}} - \underline{t}I^{\bar{B}}) \underline{x}_{\bar{B}} = z - \underline{t} \underline{a} \quad (7)$$

onde, $\underline{t}(1,n)$ é o vetor de componentes t_i . A expressão (7) é equivalente a (6).

A fim de se obter a representação da função objetivo em termos exclusivamente das variáveis não básicas,

escolheu-se o vetor multiplicador \underline{t} de tal forma que:

$$\underline{d}^B - \underline{t}I^B = 0 \quad (8)$$

A resolução do sistema (8) fornece o valor do vetor multiplicador relativo à base I^B . Como se tratam de $n-1$ equações a n incógnitas, existe um grau de liberdade que caracteriza uma infinidade de soluções. Se \underline{t}^* é uma solução de (8) então $\underline{t}^* + \alpha \underline{e}$ também será, onde α é um escalar qualquer e $\underline{e}(1,n)$ é um vetor cujos componentes são iguais a 1.

À solução de (8) dá-se o nome de vetor POTENCIAL e corresponde na teoria de programação linear às variáveis duais do problema primal. A multiplicidade dos potenciais associados a cada solução básica se deve a existência de uma redundância entre as restrições (linhas) do sistema $I\underline{x} = \underline{a}$.

Para caracterizar esta infinidade de potenciais utiliza-se o conceito de TENSÃO, que corresponde a um vetor $\underline{\theta}(1,m)$, tal que:

$$\underline{\theta} = \underline{t}I^B$$

A tensão existe em qualquer arco do grafo porém para os arcos básicos ela é igual ao custo do arco.

Assim a qualquer arco básico ou não básico, deve corresponder uma tensão para cada solução básica.

Logo, a cada fluxo associado a uma base I^B corresponderá uma única tensão θ dada pelo componente

$$o^{ij} = t_i - t_j$$

tal que:

$$\underline{\theta} = d^B \quad (10)$$

Em síntese, qualquer vetor multiplicador ou vetor de potenciais associados a uma solução básica do problema (1) deverá ocasionar tensões nos arcos da árvore correspondente, iguais aos custos destes arcos.

Chama-se VETOR DE CUSTO RELATIVO ao vetor $\hat{d}(1,m)$, definido por:

$$\underline{\hat{d}} = \underline{d} - tI = d - \theta \quad (11)$$

e que corresponde aos coeficientes da função objetivo modificada (7). Logo, tem-se que para todo arco $u_k = (i,j) \in B$, o custo relativo

$$\hat{d}_k = \hat{d}^{ij} = d^{ij} - (t_i - t_j) \text{ deve ser nulo.}$$

A determinação do vetor de custo relativo, que permite expressar a função objetivo em termos das variáveis não básicas pode ser feito a partir de qualquer vetor potencial satisfazendo (8), percorrendo-se os arcos da árvore $H(N, u_B)$, da seguinte forma:

a) Escolha, arbitrariamente, um nó da árvore $H(N, u_B)$, atribuindo-lhe um potencial qualquer (por exemplo $t_i = 0$)

b) Seja i um nó com potencial j determinado e j um nó ainda sem potencial

• se $(i, j) \in B$ faça $t_j = t_i - d^{ij}$

• se $(j, i) \in B$ faça $t_j = t_i + d^{ji}$

Assim, ao final é obtido um vetor potencial satisfazendo (8) e a partir daí pode-se determinar as tensões e o vetor de custos relativos.

Critério de Otimalidade

Seja uma base I^B e sejam as variáveis não básicas em \bar{B}_1 e \bar{B}_2 . Seja \hat{d} , o vetor de custo relativo da base I^B . Sejam as variáveis não básicas escolhidas como:

$$\underline{x}_{\bar{B}_1} = \underline{b}_{\bar{B}_1}$$

$$\underline{x}_{\bar{B}_2} = \underline{c}_{\bar{B}_2}$$

As variáveis básicas correspondentes, \underline{x}_B , são supostas factíveis, ou seja,

$$\underline{b}_B \leq \underline{x}_B \leq \underline{c}_B$$

Esta solução básica factível será ótima se o vetor de custo relativo satisfizer as relações:

$$\hat{d}_{\bar{B}_1} \geq 0$$

(12)

$$\hat{d}_{\bar{B}_2} \leq 0$$

ou seja, o custo relativo das variáveis não básicas que estão no limite inferior é não negativo, e os das variáveis

não básicas que estão no limite superior é não positivo. Isto significa dizer que se qualquer uma destas variáveis se tornar básica não irá provocar uma diminuição no valor da função objetivo expressa em (7). Este fato caracteriza um mínimo local, porém, como o problema (1) é convexo este mínimo local é o mínimo global.

Mudança de Base

Seja uma solução básica admissível onde I^B é a base e $H(N, u_B)$ a árvore correspondente. Se as condições de otimalidade não são satisfeitas é preciso mudar a solução básica factível garantindo, entretanto a diminuição do valor da função objetivo.

A filosofia do método simplex consiste em modificar uma única variável, fora da base, por vez. Escolhendo-se a variável fora da base x_s que não satisfaça o critério de otimalidade, fazemos com que a coluna correspondente I^S seja introduzida na base. Esta coluna corresponde ao arco u_s da co-árvore $(N, u_{\bar{B}})$ pois $s \in \bar{B}$. A entrada de I^S na base pode ser interpretada como a introdução do arco u_s na árvore (N, u_B) .

O grafo (N, u_{B+S}) possui agora um ciclo elementar único, denominado μ^S , assim a mudança de fluxo da variável x_s é desejável para diminuição do valor da função objetivo. Esta mudança ocasiona uma variação de fluxo em todos os arcos do ciclo.

Assim, faz-se circular matéria pelo ciclo μ^s , através do arco u_s . Graficamente, isto é representado na figura 9

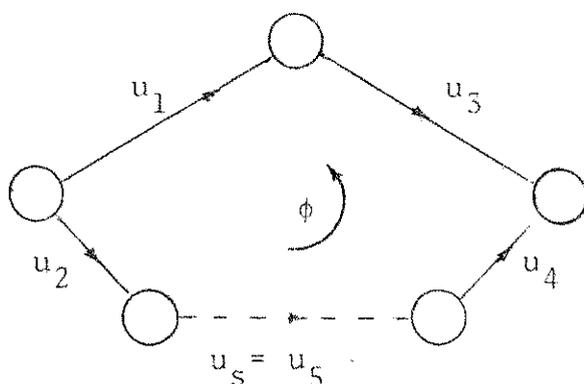


Figura 9 - Ciclo μ^s formado pela introdução do arco u_s na árvore.

O fluxo de u_s sendo aumentado de ϵ acarretará aumento de ϵ em $\{u_2, u_4\}$ e diminuição de ϵ em $\{u_3, u_1\}$. Assim, enquanto x_s varia de uma quantidade ϵ ($\epsilon > 0$ se $s \in \bar{B}_1$ ou $\epsilon < 0$ se $s \in \bar{B}_2$), as variáveis básicas (dependentes) variarão em consonância. As que pertencem ao ciclo elementar μ^s mudarão de acordo com a circulação de matéria. As que não pertencem ao ciclo permanecerão invariantes. O limite de varia

ção de x_s , que mantém a solução factível, deve ser obtido quando dentre as variáveis de fluxo que se modificam, uma atingir seu limite superior ou inferior. Caso a própria variável x_s seja a primeira a atingir seu limite, haverá mudança de solução básica sem haver mudança de base. Caso contrário, seja x_r a variável básica do ciclo μ^s , que limita a variação de fluxo, e que portanto sairá da base. Na nova base a coluna I^s tomará o lugar da coluna I^r .

A interpretação desta troca de base a nível do grafo é a seguinte: introdução do arco u_s na árvore $H = (N, u_B)$ constituindo o grafo (N, u_{B+s}) e criando um ciclo elementar μ^s . A supressão do arco u_r e μ^s gera um novo grafo $(N, u_{B+s-r}) = (N, u_{B'})$ que permanece conexo sem ciclos. É a árvore associada à nova base $I^{B'}$.

Representação de uma Base

Na programação linear clássica a informação fundamental que caracteriza as soluções básicas a cada iteração é a matriz inversa da base. É esta informação que permite realizar os cálculos necessários à mudança da base na próxima iteração.

No caso de grafos, associa-se uma árvore $H = (N, u_B)$ a cada base I^B . O vetor multiplicador e o vetor de custos relativos podem ser obtidos, como foi visto, a partir do conhecimento da árvore H . Já a coluna que entra, expressa em termos da base, corresponde ao ciclo elementar

que a introdução do arco correspondente ocasiona na árvore H.

A mudança de base que na programação linear clássica corresponde às operações matriciais sobre a inversa da base, no caso de grafos é simplificada à passagem de uma árvore (N, u_B) à outra $(N, u_{B'})$. Isto é feito através da troca de dois arcos pertencentes ao mesmo ciclo, operação esta muito mais eficiente computacionalmente.

É preciso, entretanto, representar, computacionalmente, a estrutura da árvore, de tal sorte que os cálculos necessários para efetuar as mudanças de base sejam os mais eficientes possíveis.

O método aqui utilizado e que será representado a seguir, é o proposto por GLOVER [16], o qual é considerado na literatura como superior a todos os já desenvolvidos, tanto do ponto de vista de rapidez computacional como o de memória utilizada. Ele permite representar a árvore por meio de dois índices associados aos nós do grafo.

4.4. REPRESENTAÇÃO DE UMA BASE POR MEIO DE ÍNDICES

Para apresentar os índices, que associados aos nós do grafo, permitam a representação de uma árvore é necessário antes conhecer o conceito de ÁRVORE ORDENADA.

Em uma árvore, definindo-se um nó como raiz é

possível ordenar os demais de acordo com sua posição relativa. Assim, determinando-se a cadeia (única) que liga um nó i à raiz, cada nó que faz parte dessa cadeia é chamado PREDECESSOR de i .

O conjunto de nós da cadeia, por sua vez, constitui a ASCENDÊNCIA de i . O predecessor adjacente de um nó é chamado PREDECESSOR IMEDIATO ou PAI.

Inversamente, se j é um nó da cadeia de predecessores de i , i é chamado SUCCESSOR de j . O conjunto dos nós i que têm j como seu predecessor constitui a DESCENDÊNCIA de j . O sucessor adjacente de um nó é chamado SUCCESSOR IMEDIATO ou FILHO.

Na árvore ordenada da figura 10, onde o nó 1 foi escolhido como raiz, a descendência do nó 3 é o conjunto $\{4,2,5\}$, a ascendência do nó 4 é $\{3,1\}$.

O nó 2 é predecessor imediato (Pai) do nó 5 e sucessor imediato (Filho) do nó 3.

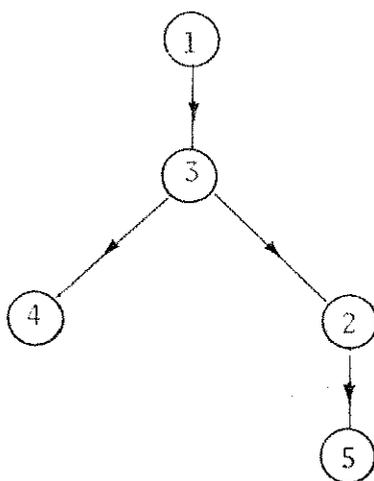


Figura 10- Árvore ordenada

Para a representação da árvore, como já foi dito, serão utilizados dois índices associados a cada nó i .

a) Predecessor Imediato de i , notado $P(i)$.

Este índice também chamado "Pai de i " corresponde exatamente ao conceito associado às árvores ordenadas. O pai da raiz é definido como zero.

b) Índice de ligação de i , notado $T(i)$.

Este índice também chamado de "Fio" permite que se percorra todos os nós da árvore em uma sequência, partindo da raiz, caracterizada da seguinte forma:

Chamando i^* a raiz da árvore; seja a notação:

$$T^2(i^*) = T(T(i^*)), \dots, T^k(i^*) = T(T^{k-1}(i^*)), \text{ etc } \dots$$

assim, examinando os nós do grafo na sequência:

$\{i^*, T(i^*), T^2(i^*), \dots, T^{n-1}(i^*)\}$ associa-se a cada nó $T^h(i^*)$, desta sequência, o conjunto de ANTERIORES $\{i^*, T(i^*), T^2(i^*) \dots, T^{h-1}(i^*)\}$ e o conjunto de POSTERIORES $\{T^{h+1}(i^*), T^{h+2}(i^*), \dots, T^{n-1}(i^*)\}$.

$T(i)$ é um sucessor imediato de i se i possui sucessor. Caso contrário, $T(i)$ é um sucessor imediato do predecessor de i mais próximo, cujos sucessores não são anteriores a i . Salvo $T^n(i^*) = T(T^{n-1}(i^*))$, que é, arbitrariamente feito nulo.

O índice $P(i)$ acima definido permite caracterizar perfeitamente a árvore (N, U_B) associada à base I^B , uma vez que através dele reconstitui-se todos os nós e arcos da base. A construção inicial destes índices bem como sua atualização a cada mudança de base podem ser realizadas de maneira simples.

Dada uma árvore (N, U_B) associada a uma base I^B e escolhido um nó raiz i^* , a construção inicial dos índices $P(\cdot)$ e $T(\cdot)$ pode ser realizada através do seguinte algoritmo.

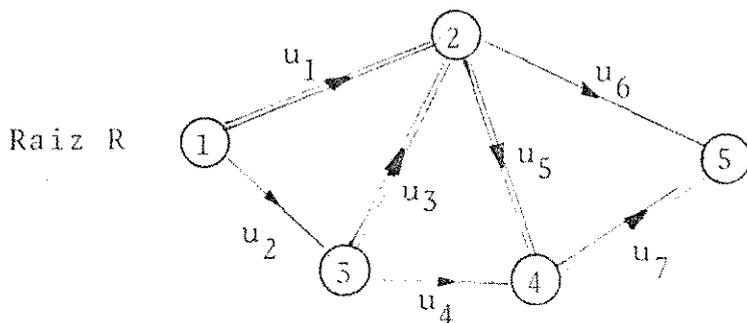
ALGORÍTMO:

PASSO INICIAL: Fazer $P(i^*) = 0$, $T(i^*) = 0$,
 $\Omega = \{i^*\}$

PASSO REPETITIVO: Procurar um arco $u_k \in U_B$ que tenha um nó extremidade $i_1 \in \Omega$ e outro nó extremidade $i_2 \notin \Omega$. Faça $P(i_2) = i_1$, $T(i_2) = T(i_1)$, $T(i_1) = i_2$, $\Omega = \Omega + \{i_2\}$ e repita este passo.

O exemplo a seguir ilustra este procedimento.

Seja o grafo (N, U) da figura 11.

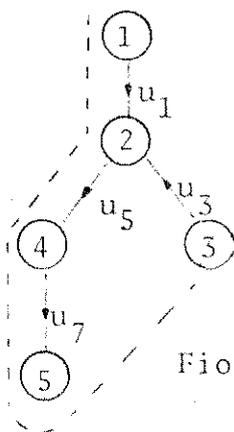


onde $N = \{1, 2, 3, 4, 5\}$ e $u = \{u_1, u_2, \dots, u_7\}$

Figura 11 - Grafo (N, u)

Seja o no 1 o no raiz e seja a rvore $\{N, u_B\}$ caracterizada por:

$N = \{1, 2, 3, 4, 5\}$ e $u_B = \{u_1, u_3, u_5, u_7\}$. A rvore ordenada e os ndices $P(\cdot)$ e $T(\cdot)$ seriam nestas condies como os apresentados na figura 12.



i	1	2	3	4	5
$P(i)$	0	1	2	2	4
$T(i)$	2	4	0	5	5

Figura 12 - rvore ordenada e ndices $P(\cdot)$ e $T(\cdot)$

O índice de ligação $T(\cdot)$ permite percorrer os nós da árvore em uma determinada sequência dada por i^* , $T(i^*)$, $T^2(i^*)$, ... onde i^* é o nó raiz. Durante a construção do fio, caso um nó possua mais de um sucessor, o fio passará primeiro por um deles que é definido como sendo o mais à esquerda.

4.5. PROGRAMAÇÃO DO MÉTODO

Serão apresentadas agora as operações que constituem esta mudança de base.

4.5.1. Procura do vetor (arco) que entra na base (árvore)

Quanto mais sofisticado o critério de escolha da variável a entrar na base, mais longo é o tempo de cálculo para realizá-la. Porém, um critério melhor, pode diminuir o número de iterações e, eventualmente, compensar em termos do tempo total de computação. Somente a experiência permite decidir qual o método mais eficiente.

De fato, estudos realizados [17] têm mostrado a importância do critério de escolha da variável que entra, sobre a duração total da otimização.

Um critério possível é: assim que for encontrado um arco que não satisfaça as condições de otimalidade,

deve-se explorar entre os arcos divergentes de seu nó origem, aquele que viola mais aquelas condições. Este parece ser o critério que melhores resultados tem dado [15].

4.5.2. Determinação do vetor (arco) que sai da base (árvore)

Para determinar o vetor que deve sair quando da entrada do vetor I^S na base I^B , basta lembrar que a expressão de I^S na base I^B é dada pelo ciclo elementar μ^S que se forma na árvore correspondente (N, U_B) . Para tanto, é preciso reconstituir este ciclo, o que é possível utilizando-se o índice $P(\cdot)$.

Seja $u_s = (i, j)$ o arco que entra e provoca o ciclo μ^S . A partir das duas extremidades do arco que entra, pesquisamos através do índice $P(i)$, até a raiz, o primeiro nó comum (nó pai comum) às duas cadeias, que caracteriza o conjunto de arcos que junto com u_s constituem o ciclo elementar desejado.

Pesquisando agora sobre os arcos do ciclo, as limitações que cada um impõe à circulação de matéria, determina-se o valor máximo factível de circulação $|\epsilon|$, e o arco u_r que sai da base.

Graficamente, isto é apresentado na figura 13.

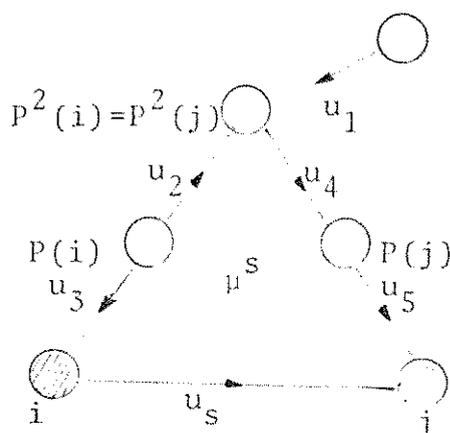


Figura 13 - Determinação do conjunto de arcos que compõem o ciclo u_s .

4.5.3. Atualização do Fluxo

No ciclo μ^s cada fluxo x_k associado ao arco u_k deve ser corrigido de $+|\epsilon|$ ou $-|\epsilon|$ de acordo com a orientação relativa ao sentido da circulação. No exemplo da figura 11 anterior, $\{u_3, u_5\}$ devem ser acrescidos de $|\epsilon|$ enquanto $\{u_2, u_4, u_1\}$ devem ser diminuídos de $|\epsilon|$.

4.5.4. Atualização do vetor multiplicador (potenciais)

Seja $H = (N, u_B)$ a árvore associada à base I^B , u_s e u_r são, respectivamente, o arco que entra e o que sai da base. A nova base é $H' = (N, u_{B'})$.

Observando a figura 14, abaixo, notamos que o par de arcos u_s, u_r caracteriza um corte no grafo dado por (A, \bar{A}) . Cada um dos dois subgrafos \bar{e} conexo sem ciclos, isto \bar{e} , \bar{e} uma \bar{a} rvore.

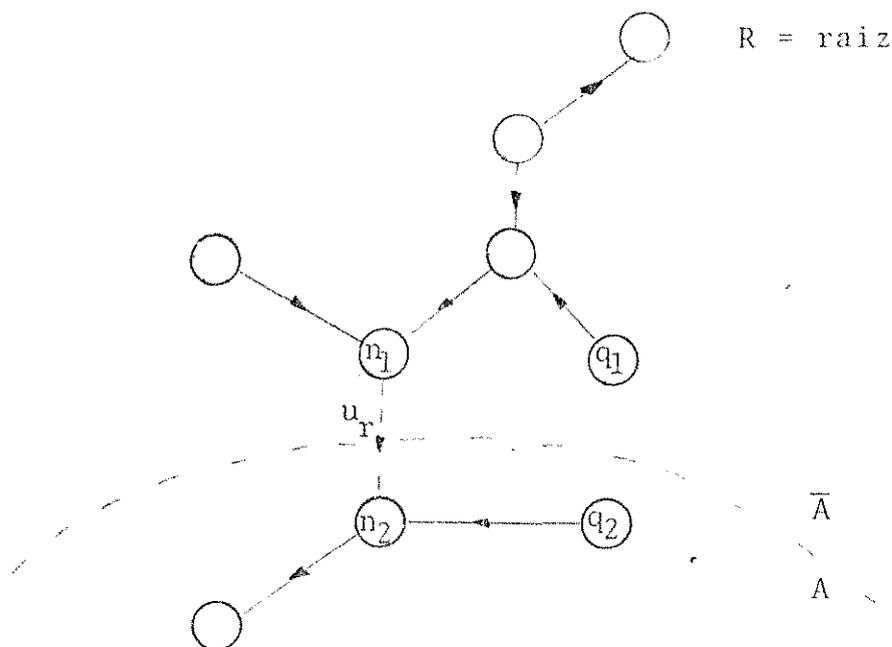


Figura 14 - Arcos u_s e u_r caracterizando os subgrafos A e \bar{A}

O vetor multiplicador relativo à nova base $I^{B'}$ deve ser tal que: $\hat{d}_{ij} = d_{ij} - (t_i - t_j) = 0 \quad (i, j) \in u_{B'}$

Convém ressaltar que o antigo vetor multiplicador relativo à base I^B satisfaz esta relação para as duas sub- \bar{a} rvoreas constituídas a partir de A e \bar{A} . Resta satisfazê-las, então, para o arco u_s , cujo custo relativo \bar{e} \hat{d}^s .

Para conseguir isto sem alterar o custo relativo dos arcos pertencentes às duas sub-árvores, basta alterar os antigos potenciais, segundo:

$$\begin{cases} t_i = t_i + \hat{d}^S & \text{se } i \in A \\ t_i = t_i & \text{se } i \in \bar{A} \text{ (que contém a raiz)} \end{cases}$$

Esta modificação mantém solidários os potenciais dos nós pertencentes à mesma sub-árvore, somente alterando um subconjunto em relação ao outro. Assim, todos os arcos cujas extremidades pertencerem ao mesmo subconjunto (A ou \bar{A}) não sofrerão alterações de custo relativo.

Os únicos arcos do grafo que terão o custo relativo modificado serão os arcos do ciclo definido pelo corte (A, \bar{A}). O sentido de variação dependerá da orientação de cada arco relativa à orientação do arco u_s . Isto sugere estratégias inteligentes para pesquisa, na próxima iteração, de arcos que são candidatos a entrar na base.

Todo o cálculo necessário à atualização dos potenciais, praticamente reside na identificação dos nós da sub-árvore A, o que pode ser feito utilizando-se os índices T(.) e P(.). Estes nós constituem a descendência do nó n_2 , extremidade inferior (filho) de u_r . Seguindo a partir de n_2 pode-se determinar a sub-árvore agindo da seguinte forma:

- marcar o nó n_2 . Fazer $j = n_2$

- Seja $i = T(j)$. Se $P(i)$ já está marcado, então marque i . Faça $j = i$ e recomece. Caso contrário, a descen
dência de n_2 é

$$\{n_2, T(n_2), T^2(n_2), \dots, j\}$$

4.5.5. Reestruturação da árvore

A operação de pivotamento de Programação Linear clássica corresponderá no caso de grafos a uma modificação dos índices $P(\cdot)$ e $T(\cdot)$ de modo a representar uma nova ba
se.

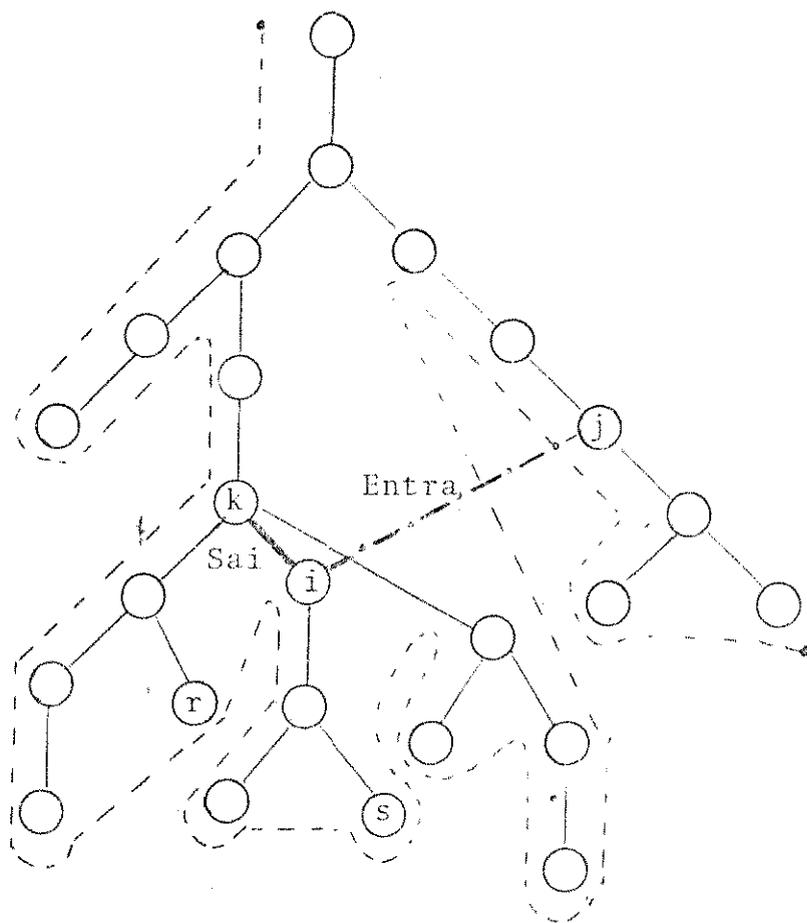
Antes de se analisar o caso que de uma troca de base, é conveniente, por razões didáticas, estudar o caso particular de uma troca de base em que o arco que entra e o arco que sai são adjacentes (tem um nó comum).

Seja o exemplo da figura 13 a seguir, onde o ar
co que entra tem extremidades i e j , enquanto o arco que sai tem extremidades i e k .

A reestruturação da árvore apresentada na figu
ra 15 consistirá dos seguintes passos:

Procedimento:

1. Identificar o nó R , ANTERIOR IMEDIATO do nó i
2. Identificar o nó S , ÚLTIMO DESCENDENTE do nó i
3. Fazer $T(R) = T(S)$; $T(S) = T(j)$; $T(j) = i$



- Fio
- . - . - Arco que entra
- _____ Arco que sai

Figura 15 - Reestruturação da árvore

4. Fazer $P(i) = j$

Após a execução do procedimento descrito, a nova base bem como o novo traçado do "Fio" podem ser vistos na figura 16. Estão, assim, atualizados os índices $P(.)$ e $T(.)$ para a nova árvore.

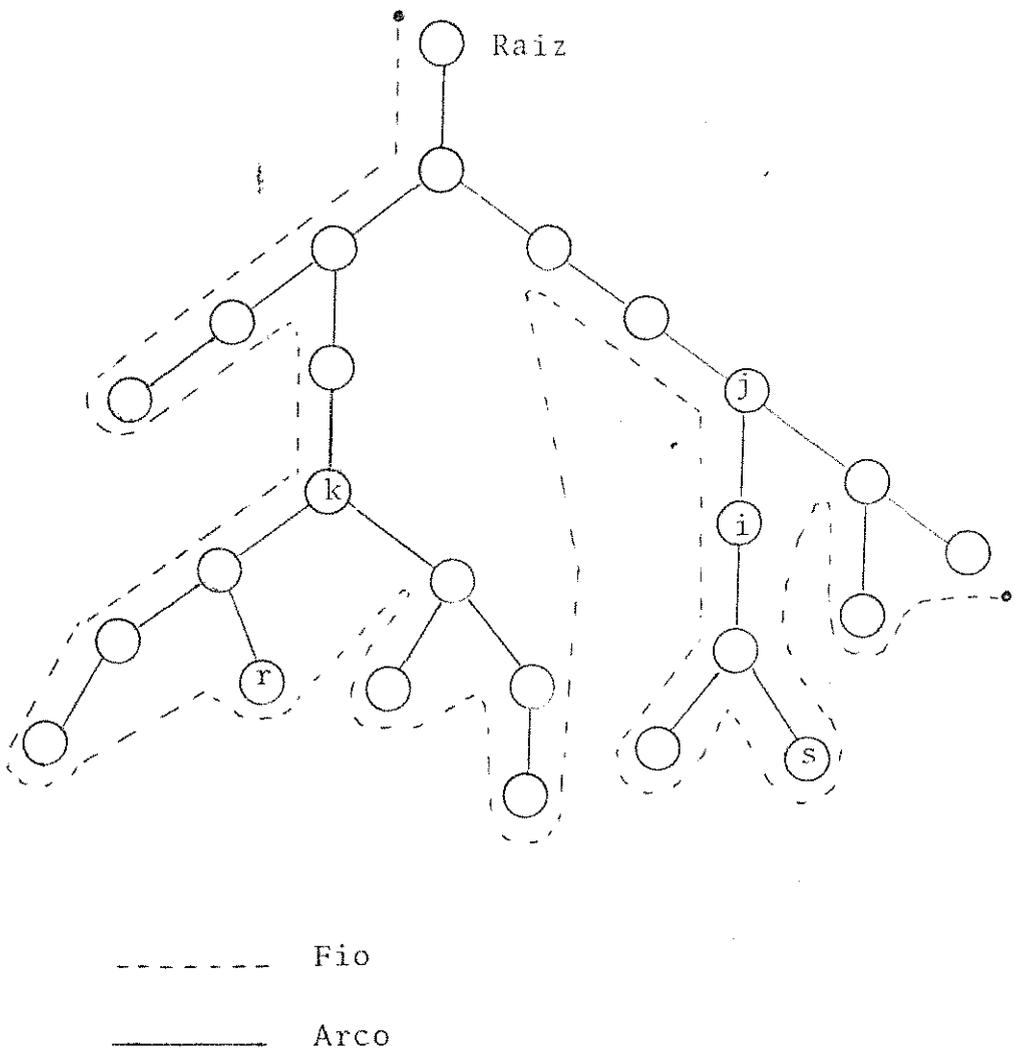


Figura 16 - Árvore reestruturada

No caso geral, entretanto, os arcos que se trocam na base não são sempre adjacentes. Porém, localizando-se a posição relativa do arco que entra e do que sai é possível realizar a troca por meio de uma sequência de trocas adjacentes.

Na figura 17, as extremidades do arco que entra u_s , possuem duas cadeias de predecessores que se encontram no "nó pai comum", definindo o ciclo.

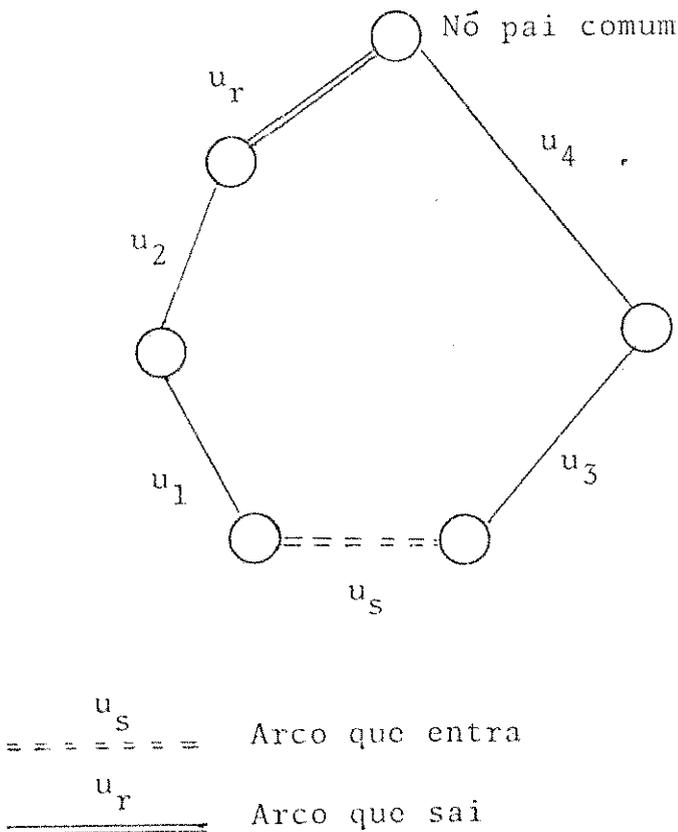


Figura 17 - Ciclo formado com a entrada de u_s

O arco que sai, U_r , situa-se na cadeia à esquerda de U_s . Isto significa que deve-se providenciar inicialmente a entrada de u_s e a saída do arco adjacente à sua esquerda, isto é, u_1 . Depois u_1 deve voltar a entrar e sair seu adjacente à esquerda, u_2 . Finalmente, u_2 volta a entrar para então sair definitivamente u_r . Assim, para a troca de base caracterizada pela entrada do arco u_s e saída do arco u_r no exemplo da figura anterior, seriam necessárias 5 trocas do tipo adjacentes.

A programação do método é constituída das cinco etapas descritas anteriormente, uma vez conhecida uma solução básica inicial. O diagrama apresentado na figura 18, resume a programação do método de resolução de um problema de fluxo de custo mínimo linear.

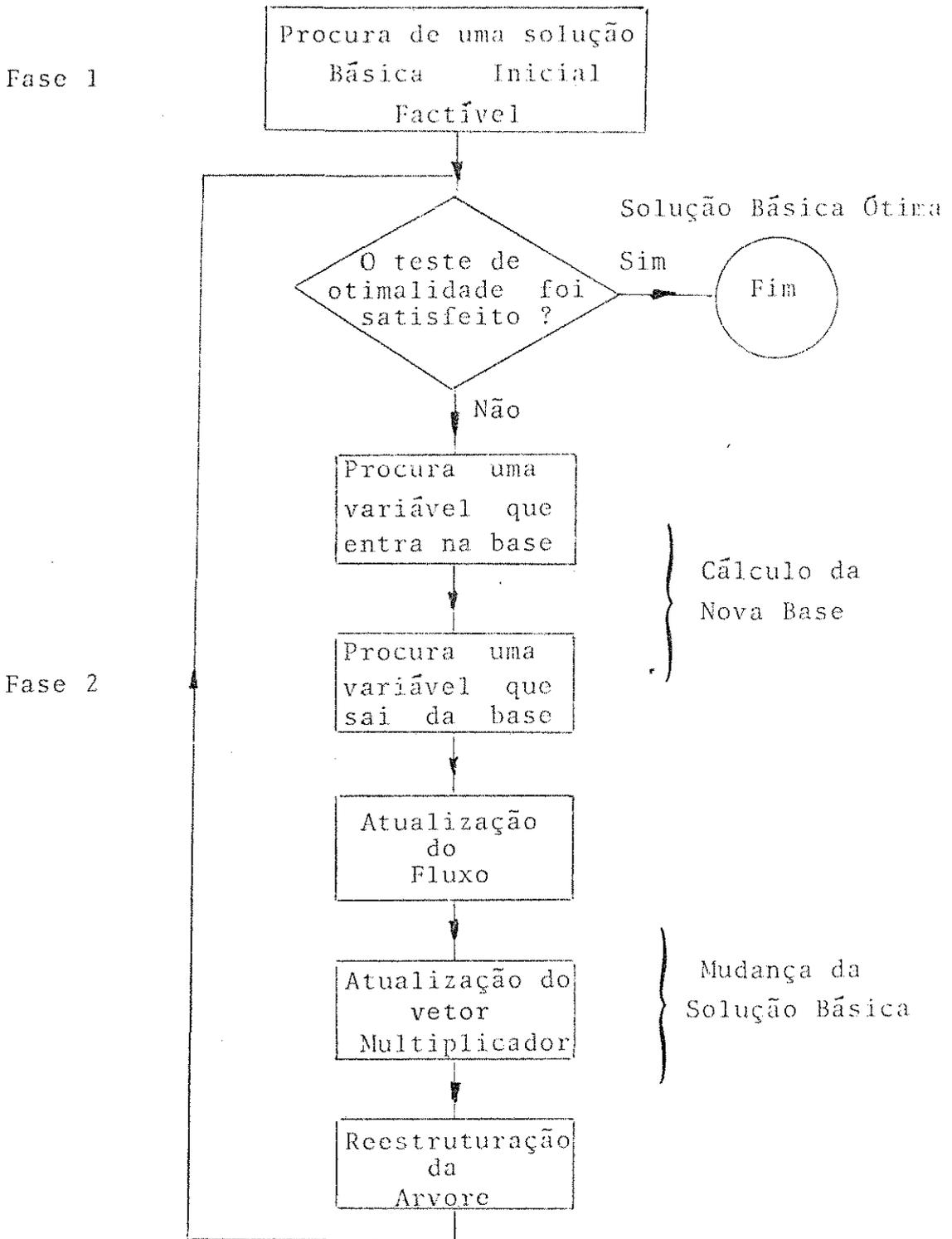


Figura 18 - Resolução de um problema de fluxo mínimo linear

4.6. CONSIDERAÇÕES

Um problema de fluxo linear se escreve sob a forma de um programa linear onde a matriz de restrições é a matriz de incidência do grafo associado à rede estudada. A uma base do sistema de restrições de igualdade corresponde uma árvore na rede e o vetor multiplicador associado a esta base é um potencial que se deduz da própria árvore. Uma mudança de solução básica é uma operação bastante simplificada em relação ao caso geral da programação linear. A introdução de uma variável independente na base equivale à introdução de um arco na árvore associada a esta base, o que gera uma "árvore aumentada" que possui um ciclo único. A supressão de uma variável básica se traduz pela supressão de um arco na "árvore aumentada", o que gera uma nova árvore.

A grosso modo as operações 4.4.1. e 4.4.2. correspondem à determinação de uma nova base enquanto as operações 4.4.3., 4.4.4. e 4.4.5. correspondem ao pivoteamento. Em programação linear clássica a duração do cálculo para se determinar a nova base geralmente é pequena se comparado com o tempo necessário para realizar o pivoteamento. No caso de redes, ao contrário, o pivoteamento sendo consideravelmente simplificado tende a torná-lo equivalente em tempo de cálculo à procura de uma nova base.

4.7. SOLUÇÃO INICIAL

Como já foi dito, a filosofia do método SIMPLEX para a resolução de problemas lineares é de caminhar de uma solução básica factível a outra, sempre garantindo a melhora da função objetivo. É preciso, portanto, para a inicialização do processo que se disponha de uma solução básica factível inicial. No caso da programação linear clássica o esforço para se conseguir esta solução inicial é bastante grande, muitas vezes superando o próprio trabalho posterior de otimização. No caso de redes, a busca da solução básica factível inicial também constitui um procedimento computacionalmente importante.

A idéia para a procura da solução inicial é a de montar um problema artificial associada ao problema original, também linear, cuja solução forneça, caso exista, uma solução básica factível para o problema original. Para isso, os n nós do grafo original, conjunto N , são particionados segundo dois sub-conjuntos complementares

$$N_1 = \{i \in N / a_i - I_i b \geq 0\}$$

$$N_2 = \{i \in N / a_i - I_i b < 0\}$$

O grafo do problema artificial deve possuir, além dos nós do grafo original, mais dois artificiais; um no FONTE (F) e outro nó SUMIDOURO (S). E além dos m arcos

arcos do grafo original, para cada no original i \bar{c} criado um arco artificial u_{m+i} ligando este no ao no artificial se gundo a regra:

• Se $i \in N_1 \implies U_{m+i} = (i, F)$

• Se $i \in N_2 \implies U_{m+i} = (S, i)$

Finalmente, um ltimo arco artificial deve li gar o no FONTE ao SUMIDOURO, $U_{m+n+1} = (F, S)$ de modo a "fe char o grafo".

A figura 19 ilustra o grafo artificial obtido.

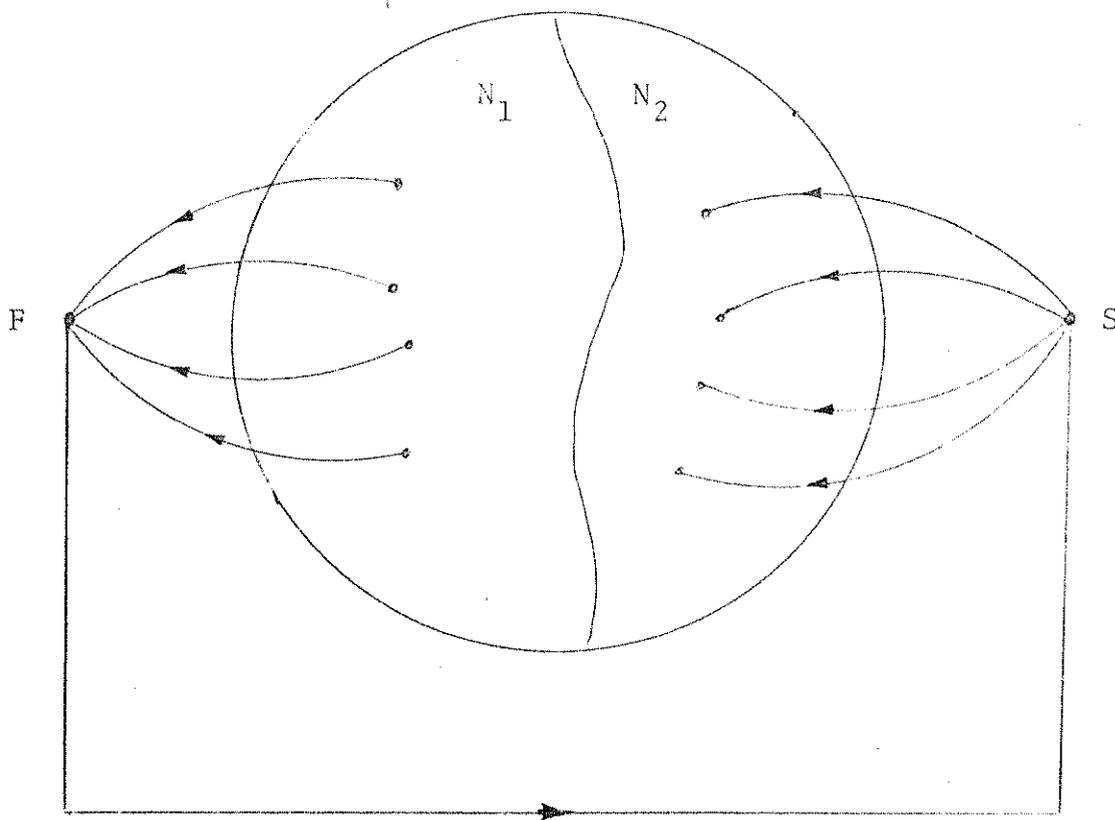


Figura 19 - Grafo artificial

Os nós $i \in N_1$ são nós produtores de matéria enquanto os nós $i \in N_2$ são nós consumidores. Uma solução básica factível para o grafo auxiliar é facilmente obtida escoando-se a produção dos nós produtores, através dos arcos artificiais, até os nós consumidores. Todos os arcos originais são utilizados no seu limite inferior.

É preciso, agora, definir uma função objetivo apropriada para que, utilizando-se o próprio método SIMPLEX, seja possível evoluir até a obtenção de uma solução básica que só empregue arcos originais para o escoamento da matéria. A função objetivo para este fim pode ser a minimização do fluxo nos arcos artificiais, isto é,

$$\text{Min } \phi = \sum_{i \in N} x_{m+i}$$

Otimizando o problema de fluxo artificial, duas opções podem ocorrer: 1) o valor ótimo ϕ é nulo, o que significa que foi possível reorientar totalmente o fluxo de matéria pela rede original. Neste caso, é preciso eliminar os arcos e nós artificiais, recuperando a rede original para a "fase de otimização"; 2) o valor ótimo ϕ é positivo, o que significa que o problema original é infactível.

CAPÍTULO V - FASE DE TRANSIÇÃO: PROGRAMA TRANZE

5.1. INTRODUÇÃO

No Capítulo 2, apresentamos uma proposta de solução do Problema de Localização de Centrais Locais, utilizando o procedimento de partição de Benders. Como foi visto naquela ocasião, o procedimento consiste em dividir a resolução do problema em duas partes com um esquema de decomposição em dois níveis. Num primeiro nível resolve-se um Programa Mestre que determina uma localização, isto é, fixa valores para z , obtendo assim um problema linear de fluxo de custo mínimo nas variáveis reais x , chamado SUBPROBLEMA. A resolução do SUBPROBLEMA nos fornece os fluxos nos arcos, o custo correspondente, bem como as variáveis duais do Programa Linear, que nos dão a indicação de como modificar a localização anterior. Determinada nova localização, novo problema de fluxo de custo mínimo é resolvido. Este procedimento é repetido até que algum critério de parada seja atingido.

Para a resolução do SUBPROBLEMA gerado é necessário que se disponha de uma solução básica factível inicial o que envolve enorme esforço computacional. Baseado nisso, uma nova forma de abordagem deste problema foi desenvolvida e implementada computacionalmente, sendo denominada FASE DE TRANSIÇÃO.

5.2. ESTRUTURAÇÃO DO MÉTODO

No diagrama de blocos apresentado na figura 18, podemos observar que a solução do SUBPROBLEMA é composta de duas fases distintas. Uma fase 1 que consiste na procura de uma solução básica factível inicial, e uma fase 2 que é o processo de resolução propriamente dito.

A FASE 1, como implementada, envolve um procedimento muito trabalhoso. Tomemos, por exemplo, um grafo com N nós e M arcos originais. Na fase 1 introduz-se $N+1$ arcos e 2 nós artificiais totalizando então uma rede de $N+2$ nós e $N+M+1$ arcos. Dentre estes arcos, M são arcos independentes não básicos. Entretanto, na FASE 2, o número de arcos independentes é de apenas $M-N$. Não obstante, o procedimento da fase 1 é tão trabalhoso quanto o da fase 2, lembrando que, a solução básica factível inicial da fase 1 é muito pobre em relação à sua solução final implicando, possivelmente, num grande número de iterações.

Recordando que a cada SUBPROBLEMA gerado pela Decomposição de Benders, o procedimento da fase 1 é executado sem aproveitar nenhuma das soluções básicas obtidas nas iterações precedentes, nos parece razoável efetuar uma fase 1 simplificada aproveitando alguma solução anterior. Logo assumiremos a execução de apenas uma fase 1 completa, no início do processo e a partir de então uma FASE DE TRANSIÇÃO.

A FASE DE TRANSIÇÃO introduz no grafo uma quantidade de arcos artificiais da ordem do número de nós escolhidos para a localização das centrais e não do número total de nós do grafo.

Vamos caracterizar o grafo em estudo considerando a rede primária, definida no Capítulo 1. A esta rede associa-se um grafo $G(N, U)$ onde a cada seção de serviço corresponde um nó e a cada duto um arco. Os nós são distintos, havendo nós que "produzem" assinantes (seções de serviço) e nós que "consomem" assinantes (as centrais atuais e as eventuais centrais a serem instaladas). É bom lembrar que a rede primária à qual nos referimos é uma "fotografia" da rede no ano horizonte, onde deseja-se implantar novos centros de fios.

Vamos eleger apenas "p" dentre os "n" nós como sendo candidatas potenciais a novas centrais. Essa seleção é possível devido ao fato de existirem nós que são obviamente locais impossíveis de abrigarem uma central telefônica.

É suposto que a rede é auto-suficiente, isto é, que $\sum_{i \in N} a_i = 0$. Caso exista um desbalanceamento entre a produção global e consumo global será necessário completar a rede com a introdução de um nó e alguns arcos chamados de fechamento, por onde escoará a custo nulo o "deficit" de matéria.

5.2.1. Zona de Filiação Aumentada

Uma central se diz SATURADA quando seu dimensionamento não comportar mais assinantes. A partir deste instante, se não houver capacidade para ampliação, o nó correspondente a esta central irá, no modelo que empregamos, se ligar ao nó onde está instalada outra central, constituindo o que chamamos de zona de filiação aumentada (Figura 20).

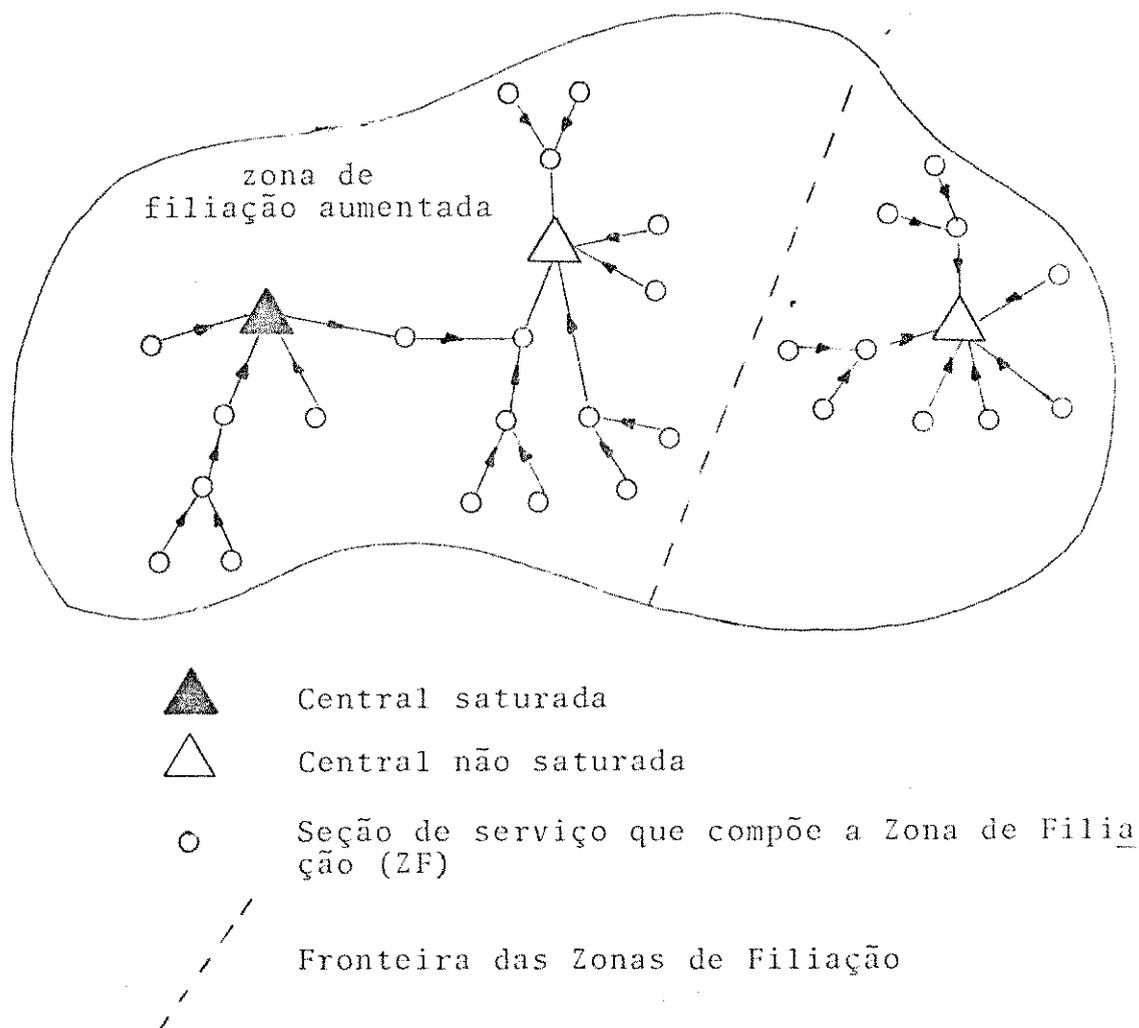


Figura 20 - Rede apresentando Zona de Filiação Aumentada.

5.2.2. Idéia Básica

Para maior clareza apresentaremos a idéia do método através de um exemplo.

Seja uma solução fornecida pela resolução de um Problema Mestre numa determinada iteração. Suponhamos que por esta solução, decidiu-se instalar centrais nos nós caracterizados por ICC; onde ICC é o conjunto de índices dos nós onde estão instaladas as centrais da solução conhecida.

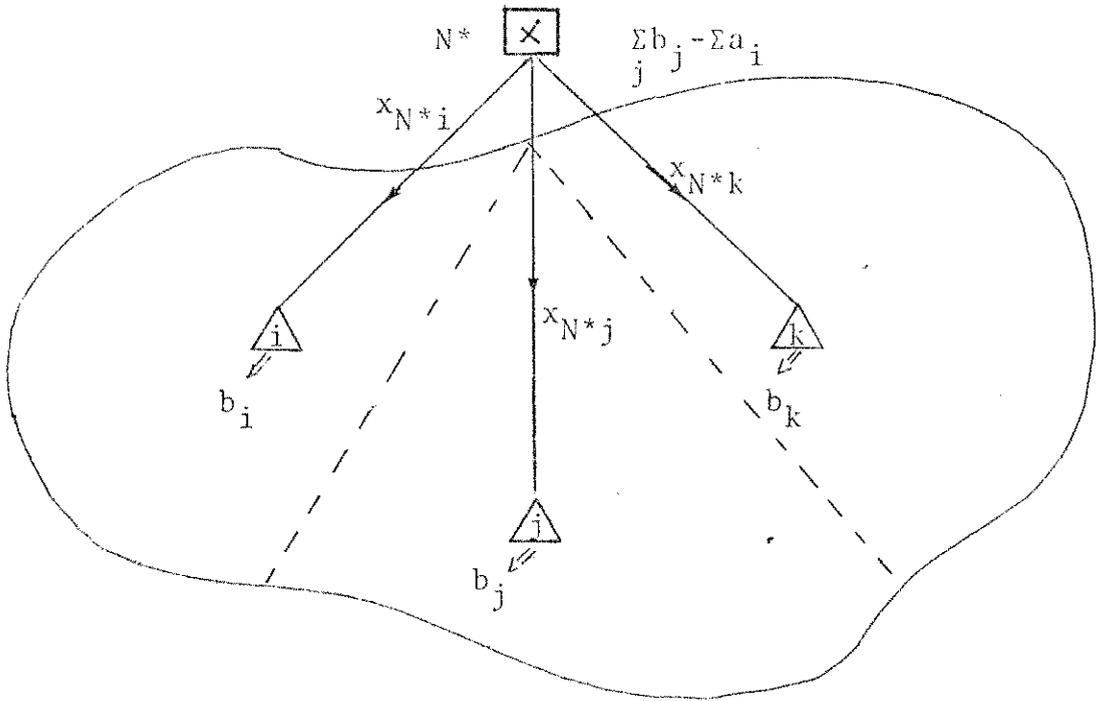
Por hipótese, vamos supor que a capacidade das centrais é maior que a demanda, ou seja:

$$\sum_{Vi} a_i < \sum_{j \in ICC} b_j$$

Por este fato vemos que existe um desbalanceamento entre a produção global e a demanda, e portanto, se faz necessária a introdução de um nó artificial N^* , chamado NÓ DE FECHAMENTO e arcos ligando as centrais a este nó. Por estes arcos, denominados ARCOS DE FECHAMENTO, irá escoar o "déficit" de matéria, a custo nulo, necessário para o balanceamento da rede conforme mostrado na figura 21. Os fluxos que irão passar pelos ARCOS DE FECHAMENTO representam as capacidades ociosas das centrais. Os arcos de fechamento presentes na solução ótima do caso ICC pertencem à árvore do grafo, ou seja, estão na base. Estes arcos tem o sentido do nó de fechamento para a central.

Seja, agora, uma nova solução do programa me
tre que determina uma nova estrutura de centrais.

Seja ICS, o conjunto de Índices dos nós onde se
rão instaladas as novas centrais que irão substituir as cen
trais da solução conhecida.



-  Nô de fechamento
-  Centrais localizadas
-  Fronteiras das zonas de filiação
-  Arcos de fechamento
-  Capacidade da Central j

Figura 21 - Rede balanceada através do nô de fe
chamento e arcos de fechamento

Nosso problema consiste em passar da solução de terminada por ICC para a solução dada por ICS, sem a necessidade de buscar uma base inicial factível, pelo processo descrito em 4.6.

Inicialmente, por simplicidade vamos supor que:

$$ICC \cap ICS = \phi$$

$$\text{observando que } \sum_{j \in ICS} b_j \geq \sum_i a_i \quad \forall i$$

Vamos partir da solução final ótima do caso ICC para obter uma solução inicial para o caso ICS.

Considerando que uma solução substituta apresenta uma nova configuração de centrais vamos chamar de ΔCC a variação entre as capacidades das centrais da solução conhecida e da solução substituta, ou seja

$$\Delta CC = \sum_{ICS} b_j - \sum_{ICC} b_j$$

Devemos considerar que a operação só será possível se as capacidades das centrais da nova solução forem suficientes para atender a demanda, ou seja, que as novas centrais comportem toda a demanda.

O que se pretende é transferir para as novas centrais todo o fluxo de matéria que antes era atendido pelas antigas. Esta transferência deverá ser feita através

dos arcos originais da árvore e os arcos de fechamento. Para se manter o balanceamento da rede, a variação de capacidade das centrais, ΔCC , deve ser injetada na rede através do nó de fechamento.

Caracteriza-se, então, o problema: como proceder a esta transferência para se chegar à nova configuração de centrais?

O procedimento é o que se segue:

Os arcos de fechamento da solução do caso ICC permanecem na base, porém tem seu sentido alterado fazendo a matéria fluir das centrais antigas para o nó de fechamento e passam, a partir de então, a ser denominados ARCOS ARTIFICIAIS. O custo associado a estes arcos passa a ser infinitamente grande. Simultaneamente, novos arcos são introduzidos no grafo, ligando os nós, onde serão instaladas as novas centrais, ao nó de fechamento N^* . O sentido destes arcos é de N^* para os nós, e seu custo nulo, passando a partir de então a ser denominados ARCOS DE FECHAMENTO. Estes novos arcos figuram fora da base, estando no seu limite superior de capacidade.

Deve-se notar também que o fluxo que irá circular nos arcos artificiais, (x_{jN^*}) $j \in ICC$, será igual à utilização real da central conhecida j , ou seja,

$$(x_{jN^*})_{\text{novo}} = b_j - (x_{N^*j})_{\text{antigo}}, \quad j \in ICC$$

ao passo que o fluxo nos arcos de fechamento será

$$x_{N^*j} = b_j, \quad j \in \text{ICS},$$

ou seja a capacidade total da nova central.

O procedimento descrito acima pode ser ilustrado pela figura 22.

Configura-se, então, o problema de se retirar da base os arcos artificiais e fazer entrar os arcos de fechamento. Isto será feito através de um processo de otimização do tipo "BIGM" clássico do método SIMPLEX.

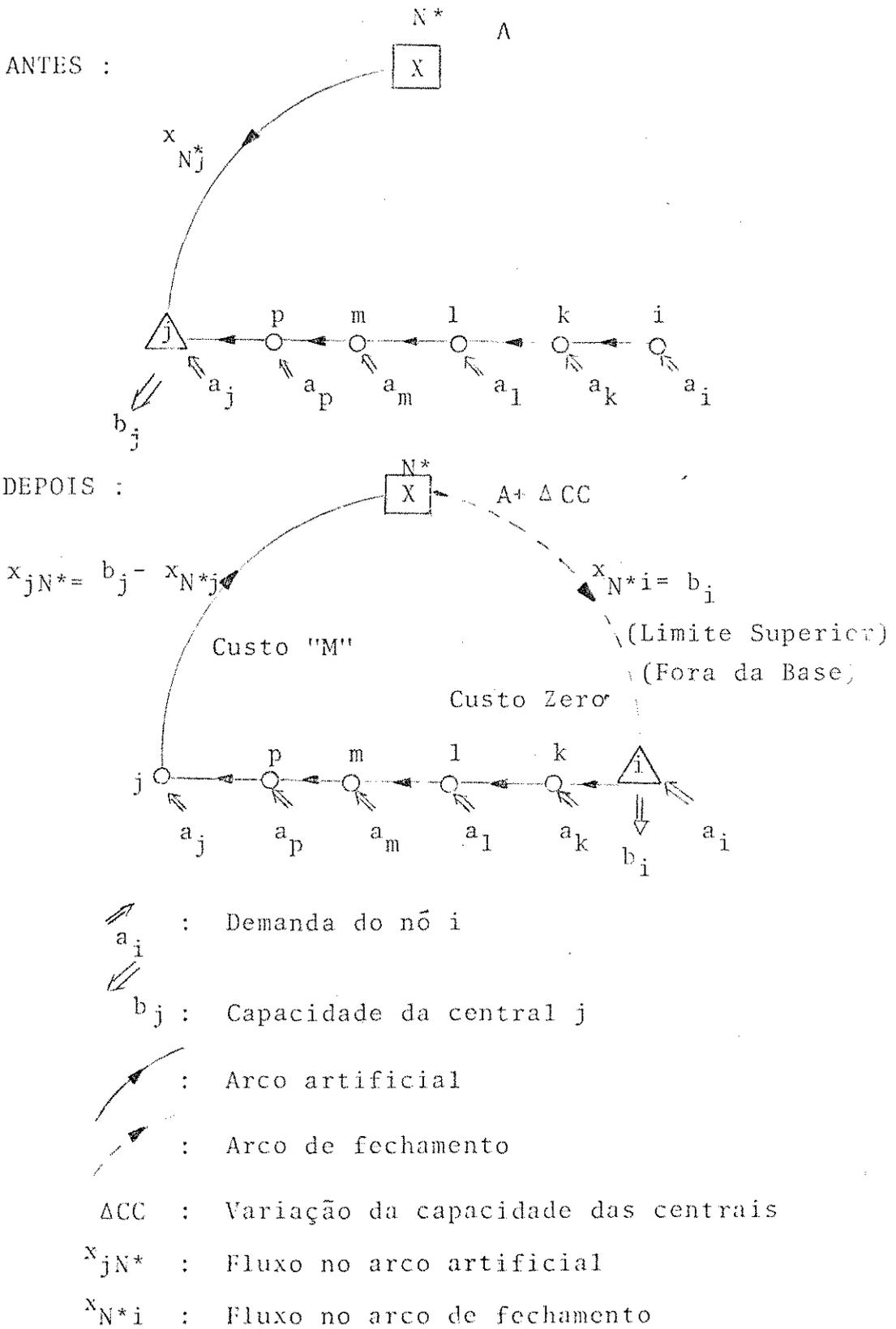


Figura 21 : Procedimento de Fase de Transição

5.2.3. Desenvolvimento

Chamemos de ϵ os custos normais e de M os custos infinitamente altos, lembrando que os custos dos arcos de fechamento são nulos.

Durante a FASE DE TRANSIÇÃO haverá duas famílias de nós; nós de BAIXO POTENCIAL e nós de ALTO POTENCIAL. Na configuração inicial da FASE DE TRANSIÇÃO teremos:

$$\text{Pot}(N^*) = 0$$

$$\text{Pot}(i) = M \quad \forall i \in \text{ICC}$$

Qualquer outro nó estará ligado a uma central por ϵ -caminhos, donde, inicialmente, todos os nós legítimos estarão com alto potencial e apenas o nó N^* , de fechamento, estará a baixo potencial. No desenrolar da FASE DE TRANSIÇÃO este baixo potencial deve "contagiar" os demais nós. Baseado nisto, podemos observar que um arco de fechamento é candidato a entrar na base. Isto é comprovado verificando o seguinte:

$$C_{N^*i} = 0 \quad i \in \text{ICS}$$

onde C_{N^*i} é o custo dos arcos de fechamento

$$\theta_{N^*i} = \text{Pot}(N^*) - \text{Pot}(i) = 0 - M = -M, \quad i \in \text{ICS}$$

onde θ_{N^*i} é a tensão no arco.

Temos então:

$$\tilde{C}_{N^*i} = C_{N^*i} - \theta_{N^*i} = M$$

onde \tilde{C}_{N^*i} é o custo relativo do arco.

Como o arco de fechamento (N^*,i) está no limite superior, ele tende a diminuir seu fluxo. Existe um ϵ -caminho, pertencente à árvore, entre i e algum $j \in ICC$, como podemos observar na figura 23.

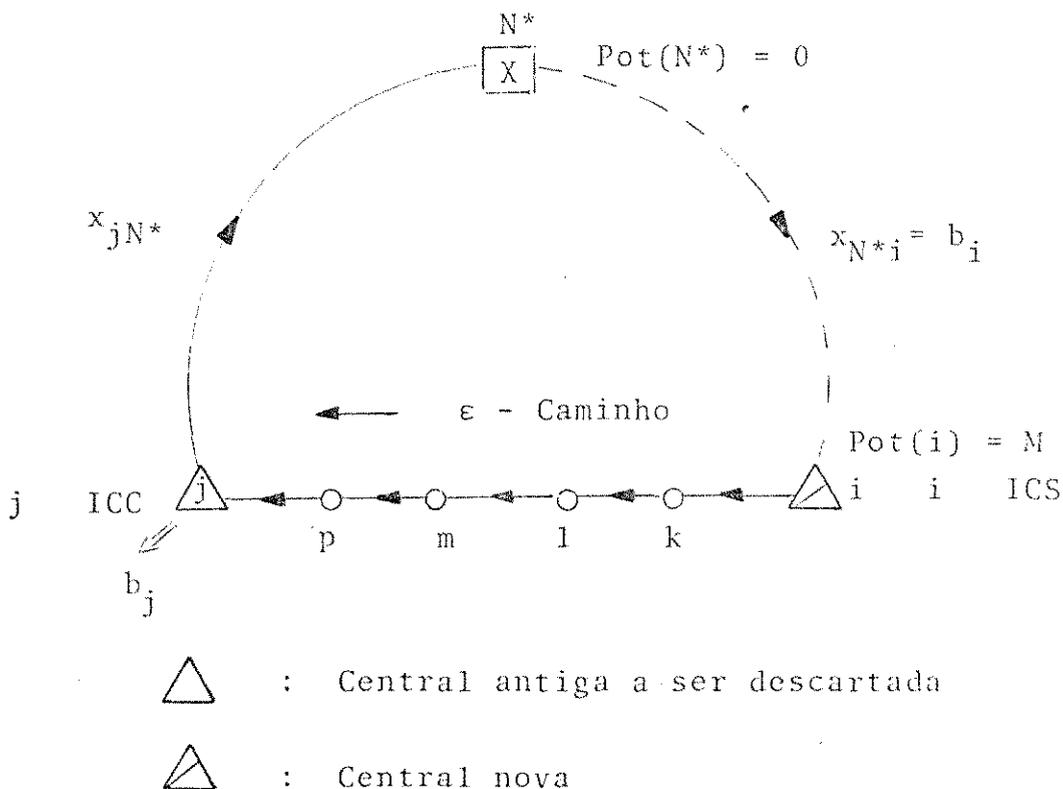


Figura 23 - ϵ -caminho entre $i \in ICS$ e $j \in ICC$

Como sabemos devemos ter

$$x_{pj} \geq x_{mp} \geq x_{lm} \geq x_{kl} \geq x_{ik}$$

portanto temos que o limitante de fluxo no ciclo será $\beta = x_{ik}$ sendo o candidato a sair da base. O nó "i" foi capturado pela família de nós de baixo potencial. Sai o arco (i,k), e o arco (k,i) é candidato a entrar na base, pois:

$$C_{ki} = \epsilon$$

$$\theta_{ki} = \text{Pot}(k) - \text{Pot}(i) = M - 0 = M$$

$$\hat{C}_{ki} = C_{ki} - \theta_{ki} = \epsilon - M = -M \ll 0$$

logo o arco (k,i) deve entrar na base tendo seu fluxo aumentado a partir do seu limite inferior. Esta situação aparece ilustrada na figura 24.

O fluxo que irá passar no ciclo será limitado por $\beta = x_{kl}$, e o nó k é "capturado" pela família de nós de baixo potencial fazendo o arco (k,l) sair da base.

A menos de situações particulares, o procedimento se repete e após alguns passos chegaremos à situação apresentada na figura 25.

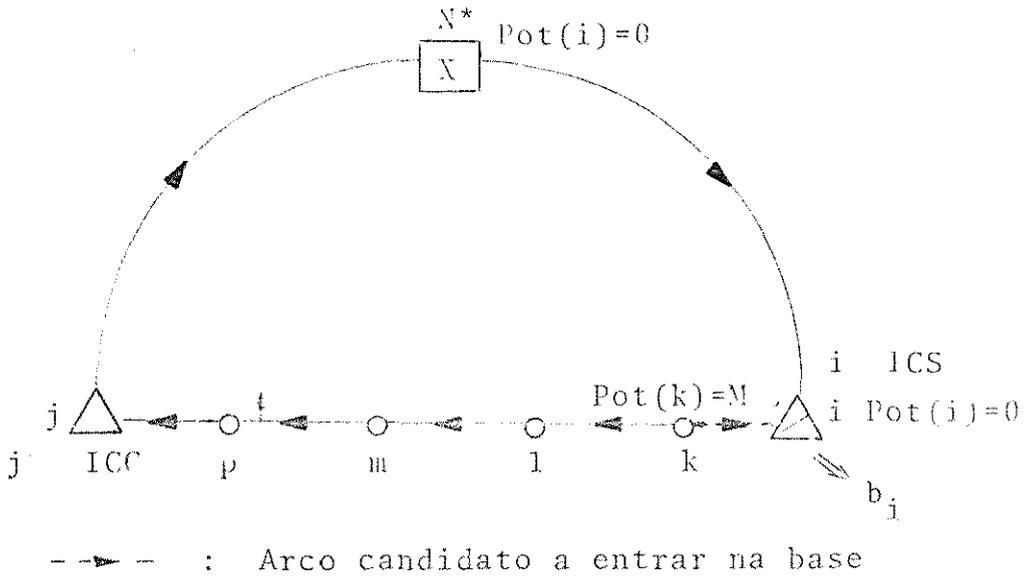


Figura 24 : Arco (k,i) candidato a entrar na base

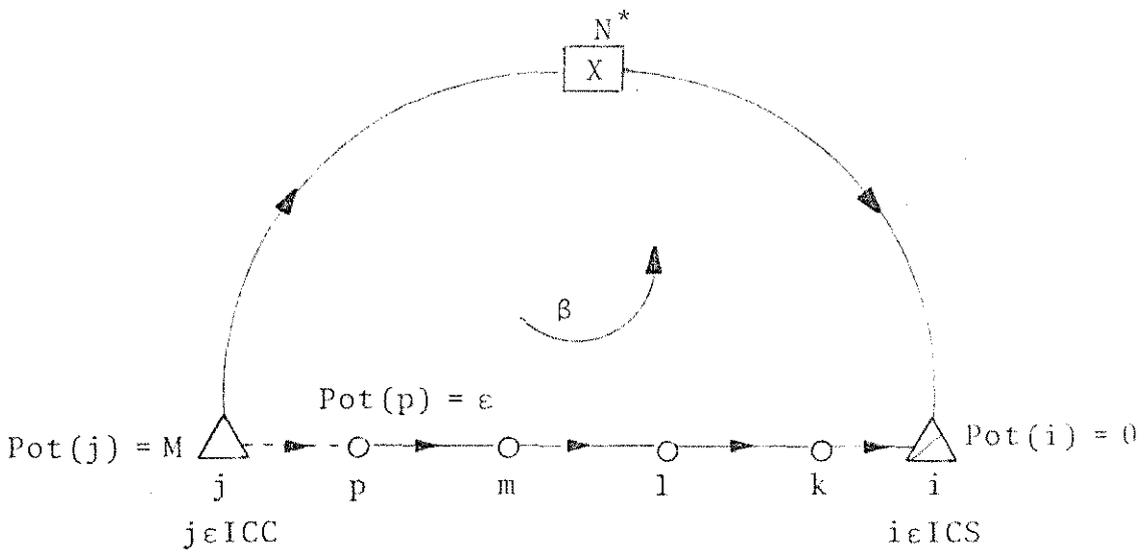


Figura 25 : Arco (j,p) candidato a entrar na base

sendo, $C_{jp} = \epsilon$

$$\theta_{jp} = \text{Pot}(j) - \text{Pot}(p) = M - \epsilon$$

$$\tilde{C}_{jp} = C_{jp} - \theta_{jp} = \epsilon - (M - \epsilon) = -M < < 0$$

Assim (j,p) é candidato a entrar na base, ou seja, finalmente os nós originalmente no ϵ caminho entre "i" e "j", tem suas demandas telefônicas desviadas para "i".

Existem duas alternativas para a limitação do fluxo β , a circular, aqui apresentadas e que serão detalhadas em 5.3.

Assim,

$$\beta = \min(x_{jN^*}, x_{N^*i})$$

a) se $\beta = x_{jN^*}$, zeramos o fluxo no arco artificial eliminando assim a infactibilidade

b) se $\beta = x_{N^*i}$, satura a nova central.

Assim, a partir do que foi apresentado podemos fazer a seguinte asserção geral:

Em uma árvore:

- Um nó "i", de família de alto potencial se liga ao nó de fechamento através de um **arco artificial** e alguns ϵ arcos no caminho $i \rightarrow N^*$, só existem nós de família de alto potencial.

- Um nó j , de família de baixo potencial se liga ao nó de fechamento através de um arco de fechamento e alguns ϵ -arcos. No caminho $j \rightarrow N^*$ só existem nós de família de baixo potencial.

5.2.4. Generalização

Seja:

ICC (conhecida) - Conjunto dos Índices dos nós da solução conhecida

ICS (substituta) - Conjunto dos Índices dos nós da solução substituta

ICP (Permanecem) - Conjunto dos Índices dos nós que permanecem na solução conhecida

ICD (Descartada) - Conjunto dos Índices dos nós que são descartados na solução substituta

ICN (novas) - Conjunto dos Índices dos nós novos na solução substituta

onde:

$$ICP = ICC \cap ICS$$

$$ICD = ICC - ICP$$

$$ICN = ICS - ICP$$

Esta situação é ilustrada na figura 26.

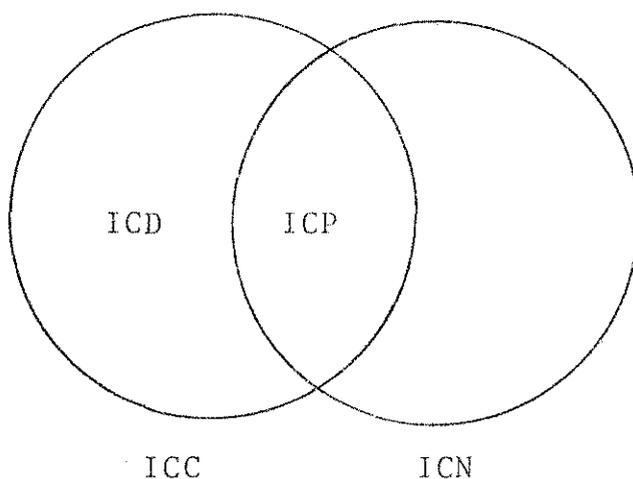


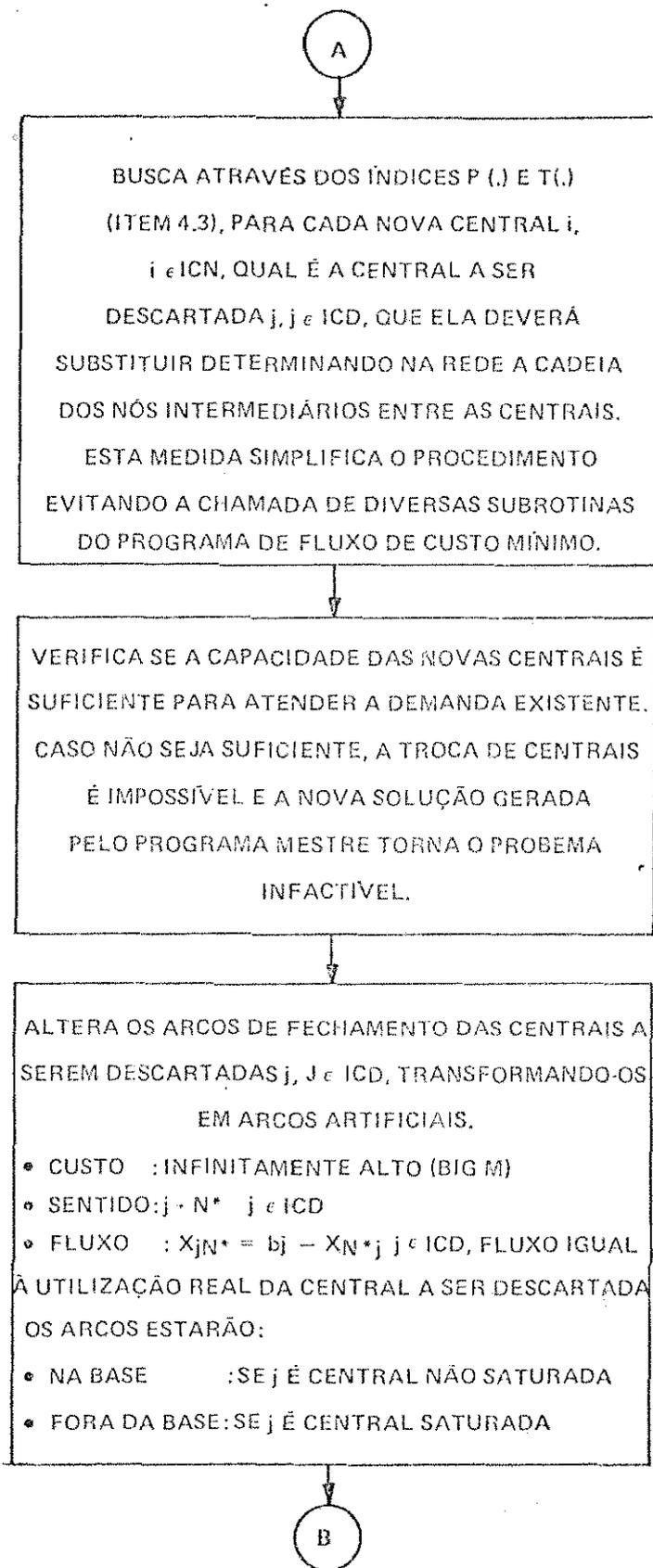
Figura 26 - Solução conhecida e solução substituta

Convém salientar que as ampliações de centrais já existentes são consideradas como a localização de uma nova central de maior capacidade.

Por outro lado, de uma iteração para outra as centrais pertencentes a ICP merecem o mesmo tratamento algorítmico que as centrais já existentes sem perspectiva de ampliação. Portanto, os conjuntos com os quais iremos efetiva

mente trabalhar são ICD e ICN, merecendo as centrais novas (ICN) arcos de fechamento e as centrais descartadas (ICD) arcos artificiais.

Dentro desta perspectiva, apresentamos a seguir o fluxograma resumido do método (Figura 27).



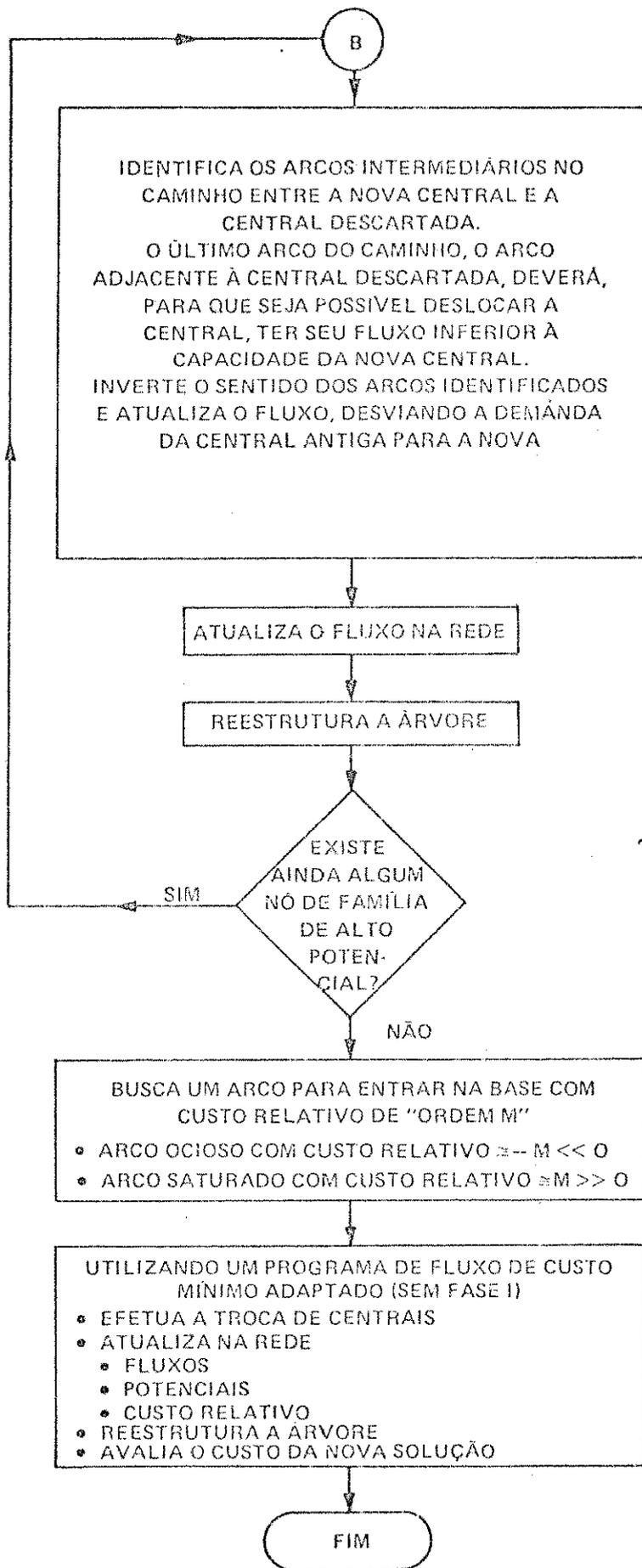


Figura 27 - Fluxograma do Método de Fase de Transição

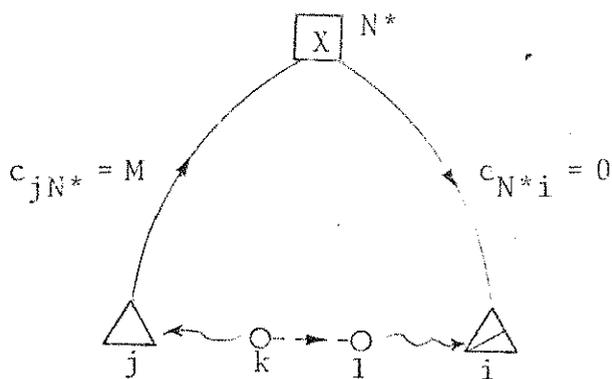
5.3. A SUPER-ITERAÇÃO

Nos itens precedentes, neste capítulo apresentamos o desenvolvimento do método da FASE DE TRANSIÇÃO. A seguir vamos pensar em grupos de iterações com o objetivo de poupar cálculos intermediários.

O arco que entra na base deve ter um custo relativo da ordem de grandeza de "m", sendo que os seus nós extremidades devem ser de famílias de potencial diferentes.

Este arco pode ser:

a) Um ϵ -arco



$$\text{Pot}(k) = M \quad \text{Pot}(l) = \epsilon$$

onde: - central descartada

- central nova

- arco candidato a entrar na base

Temos então:

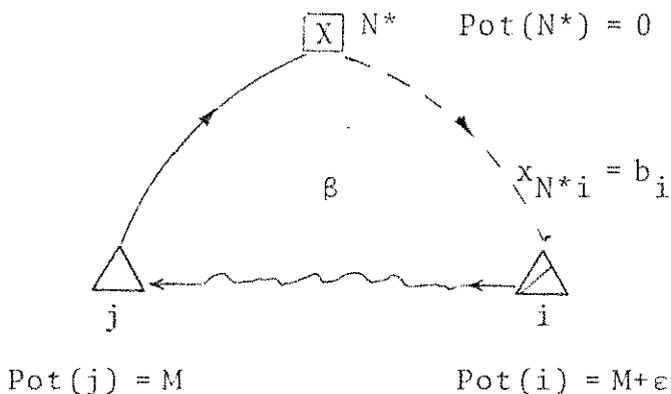
$$C_{kl} = \epsilon$$

$$\theta_{k1} = \text{Pot}(k) - \text{Pot}(1) = M - \epsilon$$

$$\hat{C}_{k1} = C_{k1} - \theta_{k1} = \epsilon - (M - \epsilon) = -M < 0$$

logo arco (k,1) deve entrar na base aumentando seu fluxo a partir de zero.

b) Um arco de fechamento



Temos então:

$$C_{N^*i} = 0 \quad \text{Pot}(N^*) = 0$$

$$\text{Pot}(i) = M + \epsilon$$

$$\theta_{N^*i} = \text{Pot}(N^*) - \text{Pot}(i) = 0 - (M + \epsilon) = -M - \epsilon < 0$$

$$\hat{C}_{N^*i} = C_{N^*i} - \theta_{N^*i} = 0 - (-M - \epsilon) = M + \epsilon > 0$$

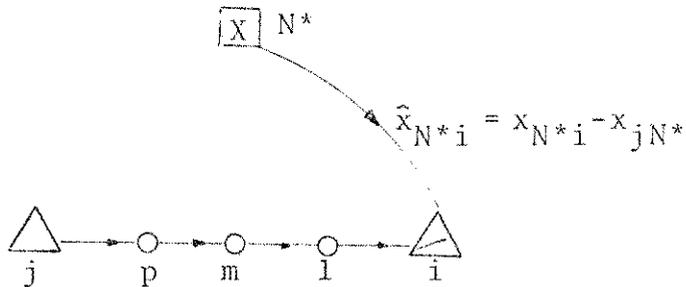
logo o arco (N*,i) entra na base, diminuindo seu fluxo, a partir de seu limite superior.

Após algumas iterações, em cada uma retirando um arco e na seguinte colocando seu "arco irmão" (arco com sentido oposto) chegamos a uma SUPER-ITERAÇÃO envolvendo um arco ARTIFICIAL e um arco de FECHAMENTO, onde:

$$\beta = \min\{x_{jN^*}, x_{N^*i}\}$$

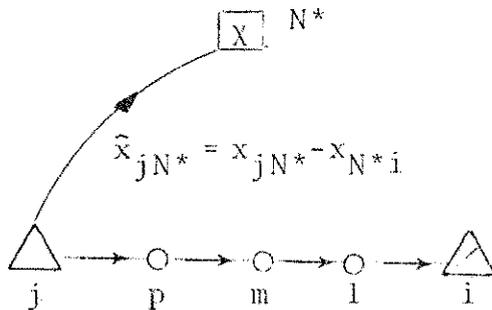
Devemos salientar que cada SUPER-ITERAÇÃO envolve um e só um arco de FECHAMENTO e um e só um arco ARTIFICIAL. Consequentemente cada SUPER-ITERAÇÃO envolve apenas uma central antiga e apenas uma central nova. Assim, os seguintes dois casos são possíveis:

I) se $\beta = x_{jN^*}$, acaba a infactibilidade



onde \tilde{x}_{N^*i} é o novo fluxo no arco (N^*,i)

II) se $\beta = x_{N^*i}$ então satura a nova central



onde \tilde{x}_{jN^*} é o novo fluxo no arco (j,N^*)

Podemos observar que o nó "m", que caracteriza um cabinet ou seção de serviços possui, no nosso modelo, as sinantes que se ligam à central antiga j e à central nova i. Esta ocorrência matemática que denominamos "cabinet re partido" (CR), ocorre quando uma central satura, sendo que o CR pode ser a própria central antiga.

A FASE DE TRANSIÇÃO é então caracterizada por uma sucessão de SUPER ITERAÇÕES, que é na realidade um passo de SIMPLEX incorporando alguma complicação adicional.

5.4. RESULTADOS OBTIDOS

5.4.1. Considerações sobre a implementação do algoritmo

O algoritmo da FASE DE TRANSIÇÃO foi implementado em linguagem FORTRAN, originando o programa TRANZE, utilizando um Computador Digital PDP-10.

O programa TRANZE foi incorporado ao programa LOCUS, na parte relativa à resolução do problema de fluxo de custo mínimo, em substituição às rotinas de construção de uma base inicial factível.

5.4.2. Rede Exemplo

O programa TRANZE foi testado na solução dos problemas de fluxo de custo mínimo envolvidos no Problema

de Localização de Centrais Urbanas na Cidade de Curitiba. Como já foi visto anteriormente (Figura 7) a rede em estudo é composta de 586 nós e 744 arcos não orientados.

Trataremos aqui dos resultados obtidos na solução dos problemas de fluxo de custo mínimo envolvidos na otimização. Serão aqui comparados os resultados obtidos pelo programa de solução do problema de fluxo de custo mínimo embutido no programa LOCUS (rotina SUBP) e os obtidos pelo programa TRANZE.

Como já foi dito, a cada iteração do método de Benders, uma política de localização de centrais é gerada e têm-se um novo problema de fluxo de custo mínimo. Considerando que as políticas de localizações obtidas são idênticas, seja com o programa LOCUS-SUBP ou com o programa LOCUS-TRANZE, a análise que iremos proceder visa avaliar a eficiência computacional dos mesmos no que diz respeito ao número de iterações e tempo de CPU gastos na solução do problema de fluxo. Assim, iremos comparar a eficiência de cada método na solução deste problema associado à cada política de localização gerada.

Os dados a serem analisados se referem à solução de 51 problemas de fluxo de custo mínimo associados à 51 iterações do método de Benders (Tabela 5).

Nº DE ITERAÇÕES NA RESOLUÇÃO DO SUBPROBLEMA		TEMPO DE CPU ^(A) NA RESOLUÇÃO DO SUBPROBLEMA		% MELHORA EM	% MELHORA EM	Nº DE ELEMENTOS	Nº DE ELEMENTOS	Nº DE ELEMENTOS
SIN	CUM	SIN	CUM			ICP	ICN	ICD
FASE DE TRANSIÇÃO	FASE DE TRANSIÇÃO	FASE DE TRANSIÇÃO	FASE DE TRANSIÇÃO	Nº DE ITERAÇÕES	TEMPO CPU	ICP	ICN	ICD
2058	2058 (1)	27,137	27,137	-	-	-	-	-
2129	152	28,744	1,883	92,8	93,4	9	1	1
2032	69	27,715	1,014	96,6	96,3	8	2	2
2023	160	27,872	2,248	92,0	92,0	7	3	3
2059	238	27,772	3,352	88,4	88,0	5	5	5
2302	179	31,341	2,254	92,2	92,8	4	6	6
2233	383	29,820	5,264	82,8	82,3	7	3	3
1885	224	24,411	2,828	88,1	88,4	6	2	2
1957	76	26,327	1,015	96,1	96,1	5	5	5
2251	387	30,744	5,271	82,8	82,8	Ø	1Ø	1Ø
1991	*997	26,464	15,166	50,0	42,7	Ø	11	11
2311	386	30,813	5,379	83,3	82,5	Ø	6	11
2167	*1176	29,125	18,657	45,7	36,0	Ø	11	6
2249	587	29,959	8,237	73,9	72,5	Ø	9	11
2120	*851	28,269	12,813	60,0	54,7	Ø	9	9
2186	379	29,718	5,308	82,7	82,1	1	9	8
2072	517	27,456	7,379	75,0	73,1	Ø	8	1Ø
2301	391	31,923	5,511	83,0	82,7	1	9	7
2300	554	30,525	8,186	75,9	73,2	Ø	1Ø	1Ø
2408	416	32,595	5,629	82,7	82,7	1	1Ø	9
2264	516	29,792	7,452	77,2	75,0	Ø	6	11
2146	464	29,933	6,351	78,4	78,8	Ø	12	6
2124	317	29,773	4,351	85,0	85,4	Ø	6	12
2191	501	30,436	6,978	77,1	77,1	Ø	9	6
2192	715	29,367	10,541	67,4	64,0	Ø	8	9
2182	599	28,560	8,850	72,5	69,0	Ø	11	8
2120	340	27,814	4,771	84,0	82,8	Ø	7	11
2137	386	29,077	5,325	82,0	81,7	Ø	9	7
2081	466	27,150	6,746	74,0	75,1	1	9	8
2142	385	28,388	5,155	82,0	81,8	Ø	9	1Ø
2136	397	28,137	5,607	81,4	80,1	1	9	8
2095	366	27,315	4,958	82,5	81,8	Ø	7	1Ø
2227	*873	29,767	13,392	61,0	55,0	1	11	6
2151	446	27,822	6,129	79,2	78,0	Ø	6	12
2139	*895	28,499	13,604	58,1	52,3	Ø	11	6
2119	452	28,145	6,1	78,6	78,3	Ø	8	11
2236	*831	29,851	12,0	62,8	60,0	Ø	9	8
2054	643	28,311	9,07	68,7	68,0	Ø	7	9
2140	*892	29,223	12,972	58,3	55,6	Ø	11	7
2104	*904	28,239	13,031	57,0	53,8	1	7	1Ø
2074	552	27,544	7,458	73,4	72,9	Ø	1Ø	8
1993	576	27,127	7,830	71,1	71,1	Ø	7	1Ø
2154	616	29,746	8,585	71,4	71,1	Ø	1Ø	7
2063	702	27,399	10,017	66,0	63,4	Ø	7	1Ø
2117	428	28,635	6,162	79,8	78,5	Ø	1Ø	7
2099	415	28,009	5,724	80,2	79,5	Ø	8	1Ø
2069	*877	28,423	13,977	57,6	50,8	Ø	8	8
1969	465	25,735	6,391	76,4	75,1	Ø	1Ø	8
2112	464	28,780	6,425	78,0	77,6	1	9	9
2038	715	27,605	10,255	65,0	62,8	Ø	7	1Ø
2154	396	29,592	5,367	81,6	81,8			

em segundos

de partida igual para as duas versões (usa FASE 1)

Como resultado imediato da Tabela 5 podemos observar a real superioridade do programa adaptado com a FASE DE TRANSIÇÃO (PROGRAMA TRANZE) sobre o procedimento convencional.

Na solução dos 51 problemas lineares de fluxo de custo mínimo obtivemos, com a FASE DE TRANSIÇÃO, uma redução média de 75% do número de iterações e de 72% no tempo de CPU.

As três últimas colunas da Tabela 5 representam as variações, da solução de uma iteração para a seguinte, em termos do número de centrais que permanecem, centrais novas e centrais descartadas. Observando então estas colunas, podemos verificar que nos casos onde o programa TRANZE realizou um maior número de iterações, (localizações 11, 13, 15, 33, 35, 37, 41 e 47), houve uma grande mudança em termos do número de centrais novas, descartadas e que permanecem, em relação à localização anterior. Além disso, observando o mapa da figura 7, constatamos que a posição dos nós escolhidos nas duas iterações são bastante distantes, justificando assim um maior esforço computacional para a mudança de solução.

Assim, avaliando todas as experiências realizadas, julgamos que o procedimento de FASE DE TRANSIÇÃO possa apresentar uma boa contribuição à solução de problemas lineares de fluxo de custo mínimo, principalmente aqueles associados a redes de grande porte.

CAPÍTULO VI - AJUSTE FINO - UMA TÉCNICA DE PÓS-OTIMIZAÇÃO

6.1. INTRODUÇÃO

Nos capítulos precedentes apresentamos técnicas alternativas para a solução do problema mestre e do subproblema.

Considerando que o problema mestre trabalha com variáveis binárias (0,1) procurando decidir dentre os "n" nós da rede quais serão os que, abrigando uma central, irão compor a localização ótima, observamos que a princípio todos os nós da rede são candidatos à central.

Uma vez que o número de centrais a serem construídas é sempre limitado, e considerando que existem 2^n possíveis soluções para o problema, optamos por uma redução do universo de candidatos à central. Para tanto procedemos a uma análise prévia da rede visando detectar nós potencialmente fortes à instalação das centrais, seja por serem pontos de concentração de assinantes, seja pelo conhecimento da estrutura urbana da rede. Assim, temos o número de candidatos a central reduzidos a "p" nós selecionados nesta análise prévia. Considerando-se que esta escolha levou em conta uma familiaridade do analista com a rede, e não um processo de otimização, foi desenvolvida uma técnica chamada AJUSTE FINO, que faz uma pós-otimização sobre os resultados

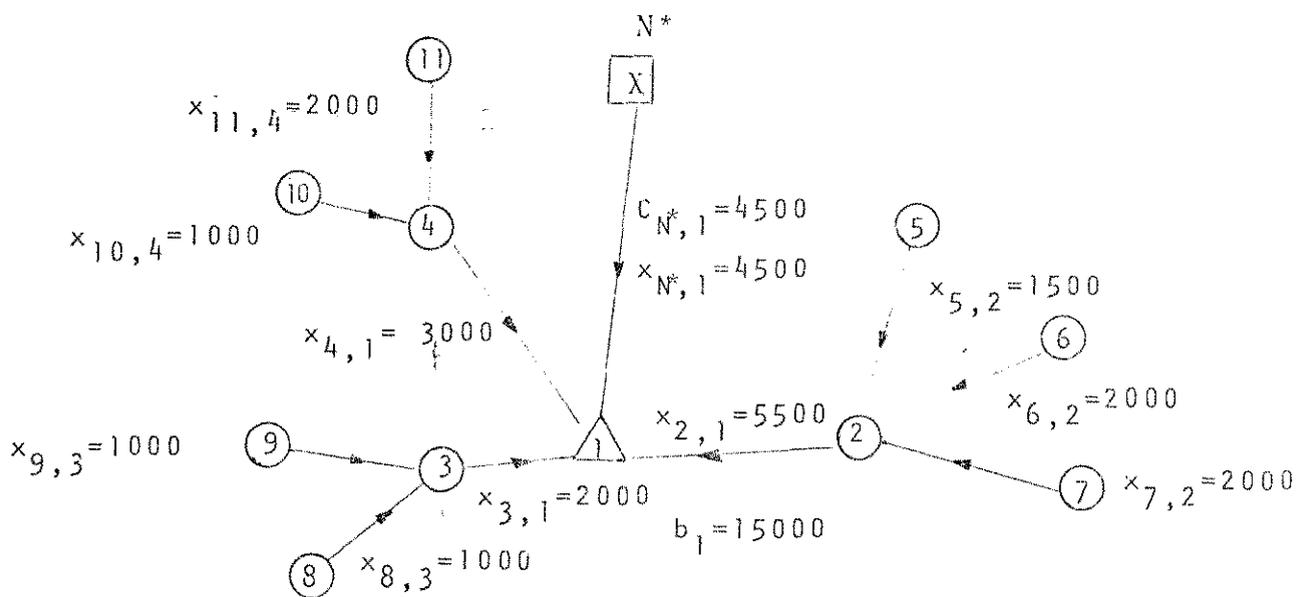
obtidos através da técnica de Benders.

Levando-se em conta que em uma rede telefônica real, devido sua dimensão e complexidade, existe dificuldade, mesmo para o planejador mais experiente, em garantir que os candidatos escolhidos são os mais representativos da região em que se encontram, tornou-se prudente a idéia de um AJUSTE FINO que, através de uma avaliação dos nós adjacentes aos escolhidos, pudesse ratificar a escolha ou deslocar a central se conveniente. Este procedimento visa atender um compromisso entre economizar cabos para a ligação dos nós à central e impedir um acréscimo nos demais custos envolvidos. Assim, para cada central nova o procedimento foi aplicado, tentando deslocar um a um cada centro de fios.

6.2. IDÉIA BÁSICA DA TÉCNICA

Para melhor compreensão apresentamos o desenvolvimento através de um exemplo. Assim, vamos considerar a zona de filiação de uma central localizada no nó 1, na solução ótima encontrada pelo método de Benders (Figura 28).

Como podemos observar na figura 28 a central localizada no nó 1 recebe todo o fluxo que transita na sua zona de filiação, através dos arcos que a ligam a seus nós adjacentes.



$C_{i,j} = 2$ i, j arcos originais na árvore

N^* = Nô de fechamento

(i) = nô pertencente à rede

\triangle_j = central localizada no nô j

$(i) \rightarrow (j)$ = arco de ligação entre o nô i e o nô j

$x_{i,j}$ = fluxo de matéria que transita no arco (i,j)

Figura 28 - Zona de filiação da central do nô 1

Cada arco possui um custo unitário associado por unidade de matéria, c_{ij} , e que por simplicidade vamos supor igual para todos os arcos da árvore (base).

Assim, vamos analisar os nós adjacentes à central (nós 2, 3, 4) e verificar se um possível deslocamento da central para um destes nós implicaria numa diminuição do custo total da solução.

O que se faz na realidade é verificar para todos os nós adjacentes se o custo de transportar toda a matéria, que passa naquele arco em direção à central, é suficientemente grande para justificar o deslocamento da central para o nó. Ao se deslocar a central para este nó estamos exatamente economizando este custo. Contudo, se o custo de desviar o fluxo dos outros nós adjacentes, até a nova localização da central, for superior à esta economia, estamos aumentando o custo total e não terá valido a pena.

Além do custo do transporte da matéria pelos arcos, temos que considerar o custo do terreno do local onde se pretende construir a central. Como este custo pode variar de nó para nó, o AJUSTE FINO deverá considerá-lo para decidir se o deslocamento da central vale a pena.

Visando então avaliar, a priori, a variação no custo total da solução, provocado pelo possível deslocamento do centro de fios, definiu-se uma medida chamada FATOR DE DECISÃO (CSI). Esta medida considera a localização da

central, os fluxos provenientes dos ns adjacentes, candidatos ao deslocamento, e o custo de transportar matria por estes arcos.

Como foi apresentado na figura 28, a central RE CEBE todo o fluxo que transita na sua zona de filiao atravs dos arcos de seus ns adjacentes. O que o fator de deciso pretende  verificar qual a ordem de grandeza do fluxo (x_{ij}) que chega  central, a partir do n adjacente i , em relao  utilizao real da central. Se este fluxo entrante for alto, a idia de deslocar a central para este n se mostra bastante atraente. Contudo, no basta esta constatao,  necessrio considerar o custo associado  utilizao do arco.

Como ilustrao, consideremos o seguinte exemplo (Figura 29)

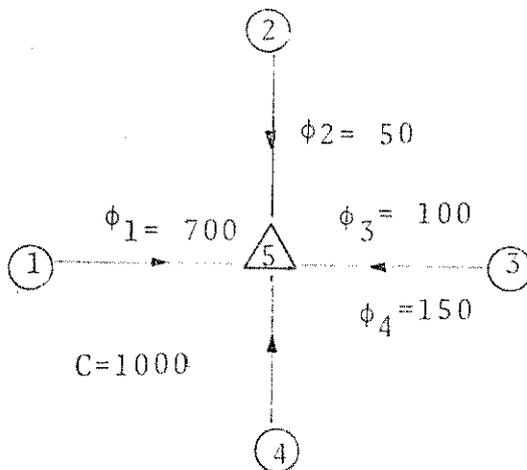


Figura 29 - Estrutura da rede antes do deslocamento da central (ajuste fino)

onde ϕ_i = fluxo no arco que liga o nó adjacente i à central

C = capacidade da Central

Temos então que

$$C = \phi_1 + \phi_2 + \phi_3 + \phi_4$$

se deslocarmos a central para o nó 1, o único fluxo que se altera é o fluxo no arco entre o nó 1 e a central (Figura 30).

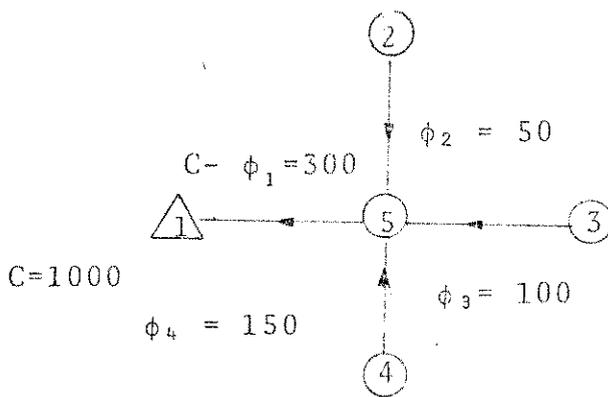


Figura 30 - Fluxos nos arcos após o deslocamento da central para o nó 1

O novo fluxo no arco será então igual a $C - \phi_1$.

Assim, deslocamos a central para o nó 1 se o fluxo original no arco for superior ao fluxo após o deslocamento, isto é se

$$\phi_1 > (C - \phi_1)$$

$$2\phi_1 > C$$

$$\phi_1 > \frac{C}{2}$$

ou seja se o fluxo original for superior a todos os demais que chegam à central.

Dentro deste espírito, o fator de decisão, CS_i foi definido a seguir:

$$CS_i = \underbrace{\left(\sum_{\ell \in I} x_{\ell j} + a_j - 2x_{ij} \right)}_C c_{ij} \quad i \in I$$

onde:

$x_{\ell j}$ = fluxo que parte do nó ℓ (adjacente) em direção à central j

I = conjunto de nós adjacentes

a_j = assinantes diretamente ligados à central

i = nó adjacente sendo analisado

C_{ij} = custo unitário de transportar matéria pelo arco i, j (arco que liga a central j ao nó adjacente i , que está sendo analisado)

Convém salientar que o custo do terreno não foi aqui considerado, sendo adicionado no final do procedimento

de avaliação do custo da solução alternativa (central deslocada).

Tomamos então, CS_k , o menor valor de CS_i .

- Se $CS_k < 0$ indica que o fluxo que chega à central j , a partir do nó adjacente k , é sozinho superior à soma de todos os demais fluxos que chegam à central j e os assinantes ligados diretamente a ela. Portanto podemos deslocar a central para o nó k , garantindo que esta solução é melhor que a anterior a menos da variação do custo do terreno.

- Se $CS_k \geq 0$ não é possível afirmar sem a avaliação do custo da nova configuração se esta solução é melhor que a antiga.

Isto é necessário pois o deslocamento da central modifica as zonas de filiação, o que pode melhorar ainda mais a solução e torná-la competitiva com a anterior.

Avaliamos então o custo da nova localização e se este for inferior ao custo da atual solução, deslocamos a central para o nó k . Caso contrário, mantemos a localização atual.

Não sendo possível deslocar a central para o nó k , passamos a analisar os outros nós adjacentes em ordem crescente de CS_i .

Ao final do procedimento, após a avaliação de todos os nós adjacentes, se foi possível o deslocamento pas

samos a analisar os nós adjacentes à nova localização, à excessão do nó relativo à antiga localização e repetimos o procedimento.

6.3. DESENVOLVIMENTO

Voltando ao exemplo da figura 28, vamos proceder todo o desenvolvimento do método.

Vamos calcular os FATORES DE DECISÃO (CS_i) dos nós adjacentes 2, 3 e 4. Temos então

$$CS_i = \left(\sum_{\ell \in I} x_{\ell j} + a_j - 2 x_{ij} \right) c_{ij} \quad i \in I$$

i	x_{ij}	c_{ij}	a_j	CS_i
2	5500	2	-	-1.000
3	2000	2	-	13.000
4	3000	2	-	9.000
$\sum x_{\ell j} = 10.500$				

Como $CS_2 < 0$ sabemos que a central instalada no nó 1 pode ser deslocada para o nó 2 pois o custo desta solução será inferior ao da anterior.

Assim, para deslocar a central para o nó 2, te

mos que transferir todo o fluxo convergente ao n^o 1, para o n^o 2, tanto o fluxo dos arcos originais como o do arco de fechamento.

Podemos verificar que o procedimento de transferir a central para o n^o adjacente é um caso particular do procedimento de FASE DE TRANSIÇÃO onde a central do n^o 1 representa a central descartada e o n^o 2 a central substituta. Logo, conforme a figura 28, é criado um arco de fechamento com custo nulo, ligando o n^o 2 ao n^o de fechamento. Este arco está fora da base no seu limite superior, igual à capacidade da central a ser descartada.

O arco de fechamento do n^o 1 tem seu sentido invertido, custo infinitamente alto e fluxo igual à real ocupação da central.

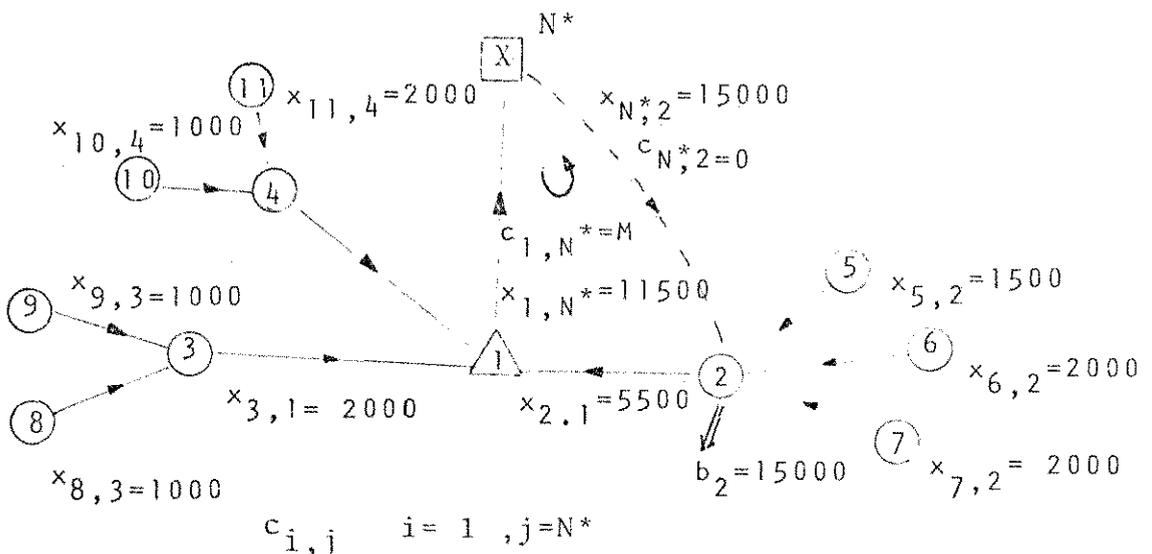


Figura 31 - Criação do arco artificial e arco de fechamento

O arco $(N^*,2)$ entra na base, formando um ciclo. O fluxo no arco $(2,1)$ limita a circulação de matéria e conseqüentemente sai da base gerando a situação apresentada na figura 32.

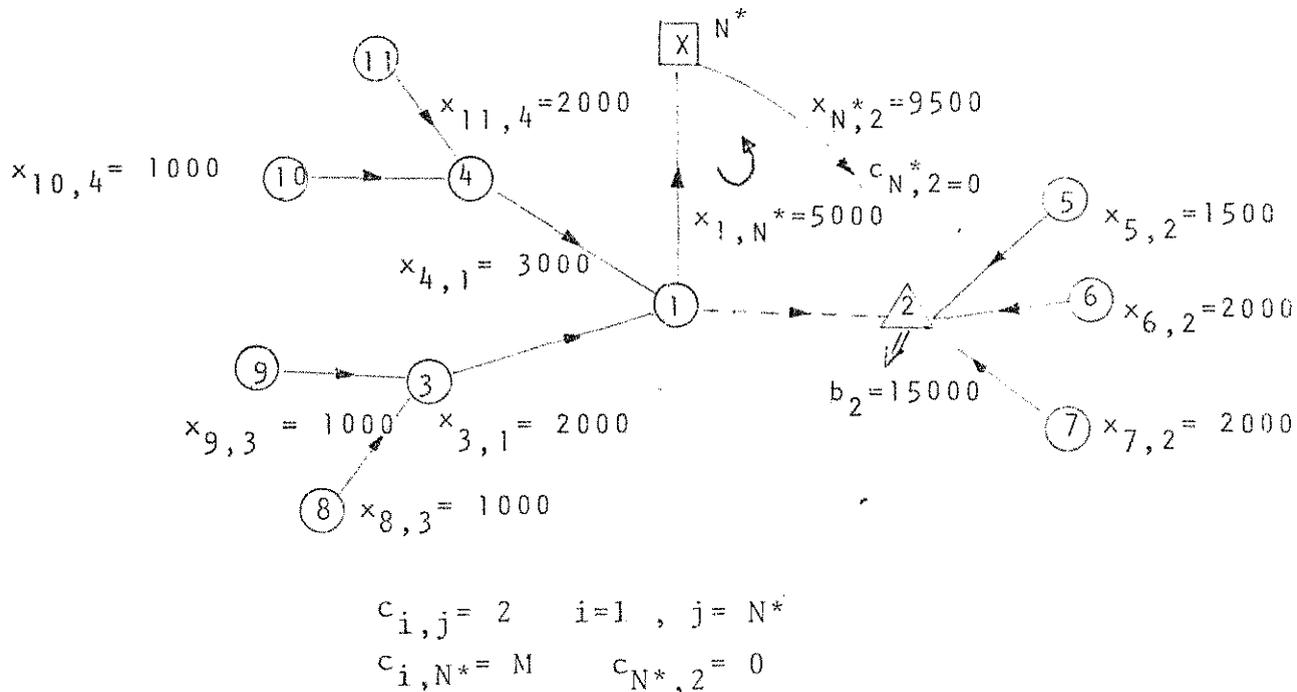


Figura 32 - Arco $(2,1)$ limita o fluxo no ciclo

Considerando que:

$$Pot(1) = M$$

$$Pot(2) = 0$$

$$\theta_{12} = Pot(1) - Pot(2) = M$$

$$\hat{C}_{12} = C_{12} - \theta_{12} = 2 - M = -M < 0$$

O arco $(1,2)$ entra na base formando um ciclo.

O arco $(1, N^*)$ limita a circulação de matéria e sai da base. Finalmente, a central foi deslocada para o nó 2, conforme a figura 33, a seguir:

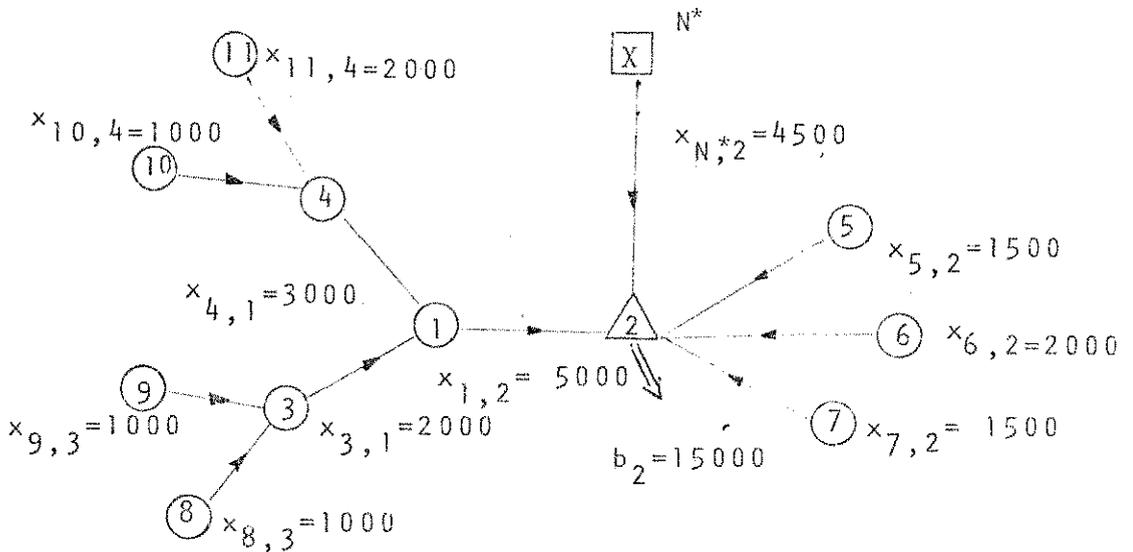


Figura 33 - Nova solução (Central deslocada para o nó 2)

A seguir avaliamos o custo da nova solução através de um PFCM, incluindo o custo do terreno e confirmamos o deslocamento da central.

Passamos a analisar os nós adjacentes ao nó 2, à exceção do nó 1, e repetimos todo o procedimento tentando melhorar ainda mais a solução.

No caso da central a ser deslocada estar satura

da, o procedimento é análogo ao descrito na fase de transição; contudo, considerando a simplicidade desta situação se comparada à fase de transição, desenvolvemos um procedimento simplificado para efetuar este deslocamento (figura 34).

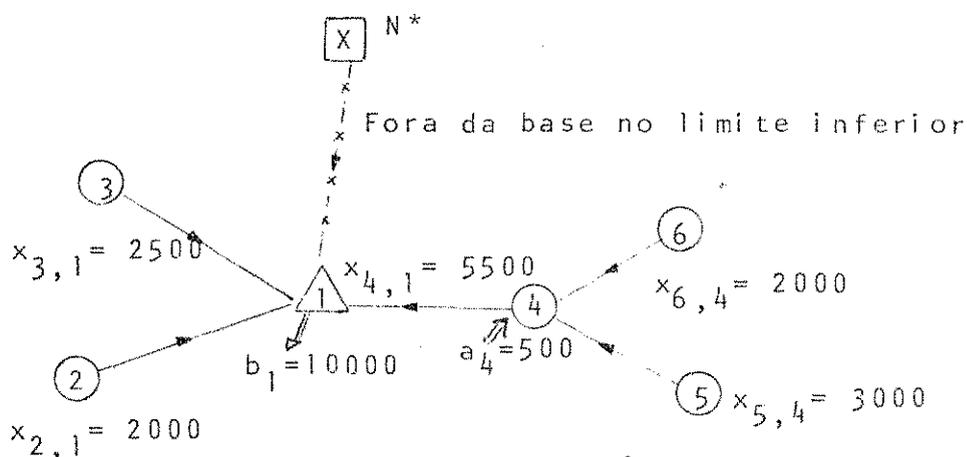


Figura 34 - Solução com Central Saturada no nó 1

Calculando o CS_i dos nós adjacentes à central do nó 1, obtemos:

i	x_{ij}	c_{ij}	a_j	CS_i
2	2.000	2	-	12.000
3	2.500	2	-	10.000
4	5.500	2	-	-2.000
$\Sigma x_{ij} = 10.000$				

Como $CS_4 < 0$ vamos deslocar a central para o nó 4. Para tanto, criamos um arco de fechamento com custo nulo, ligando o nó de fechamento ao nó 4. Fazemos este arco entrar na base com fluxo igual ã capacidade da central.

O arco (4,1) é retirado da base sendo substituído por seu oposto arco (1,4), onde irá transitar um fluxo correspondente ao fluxo que chega ao nó 1, menos o que passava pelo arco (4,1). Após a atualização dos fluxos a cen-tral foi deslocada para o nó 4 resultando na situação da fi-gura 35, observando que novamente a central saturou, tendo seu arco de fechamento fora da base no limite inferior.

Convém salientar que este exemplo tem objetivos apenas didáticos uma vez que na prática não deve ocorrer a situação de uma central nova saturar.

De qualquer forma é bom lembrar que o procedi-mento descrito acima, para centrais saturadas, é o mesmo pa-ra o caso de deslocamento de central não saturada. A dife-rença só existe a nível de reestruturação da árvore.

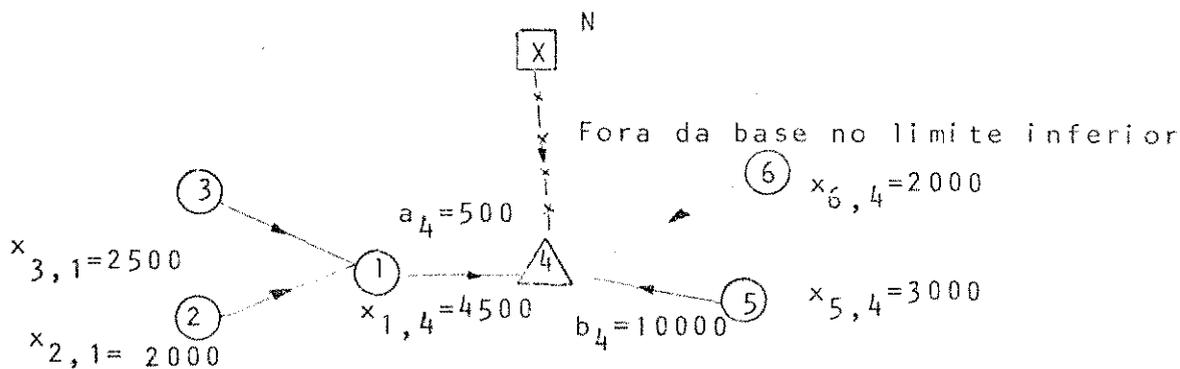
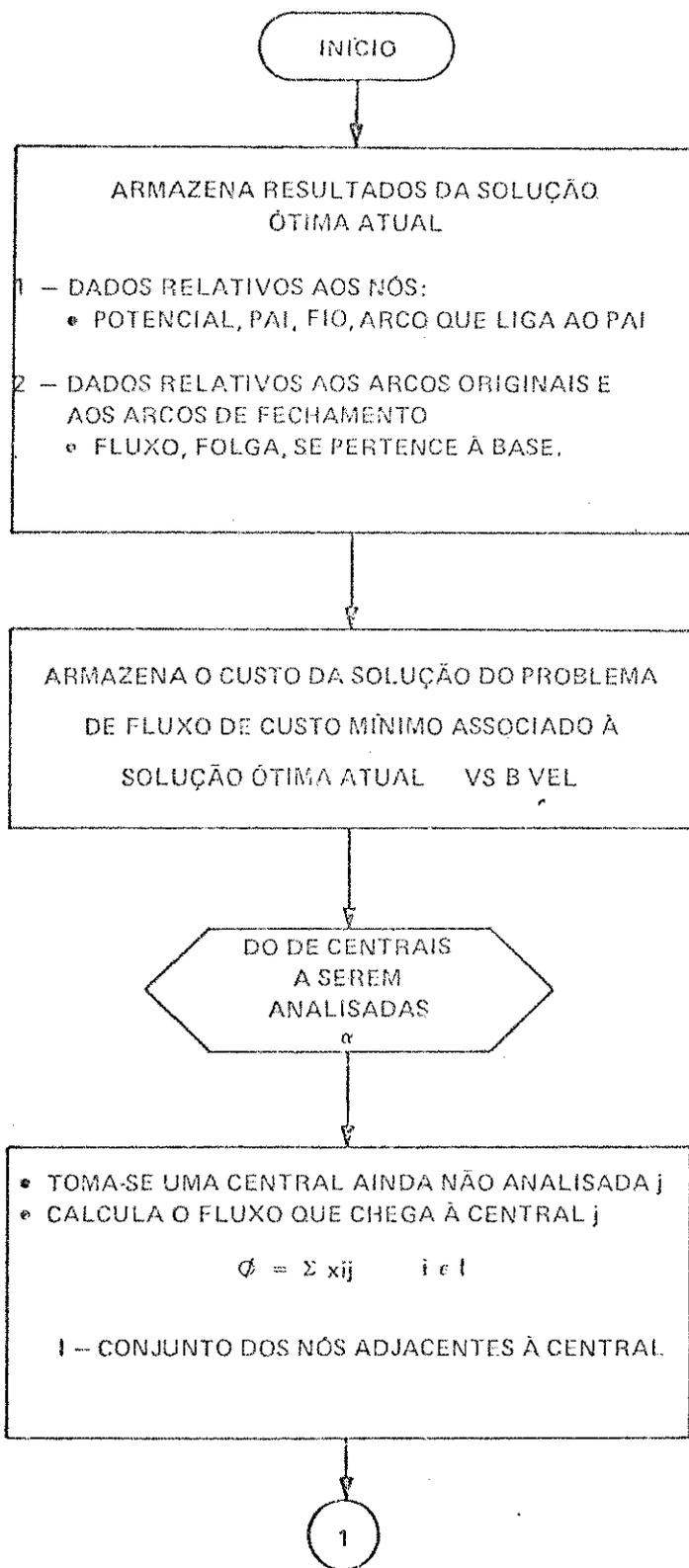


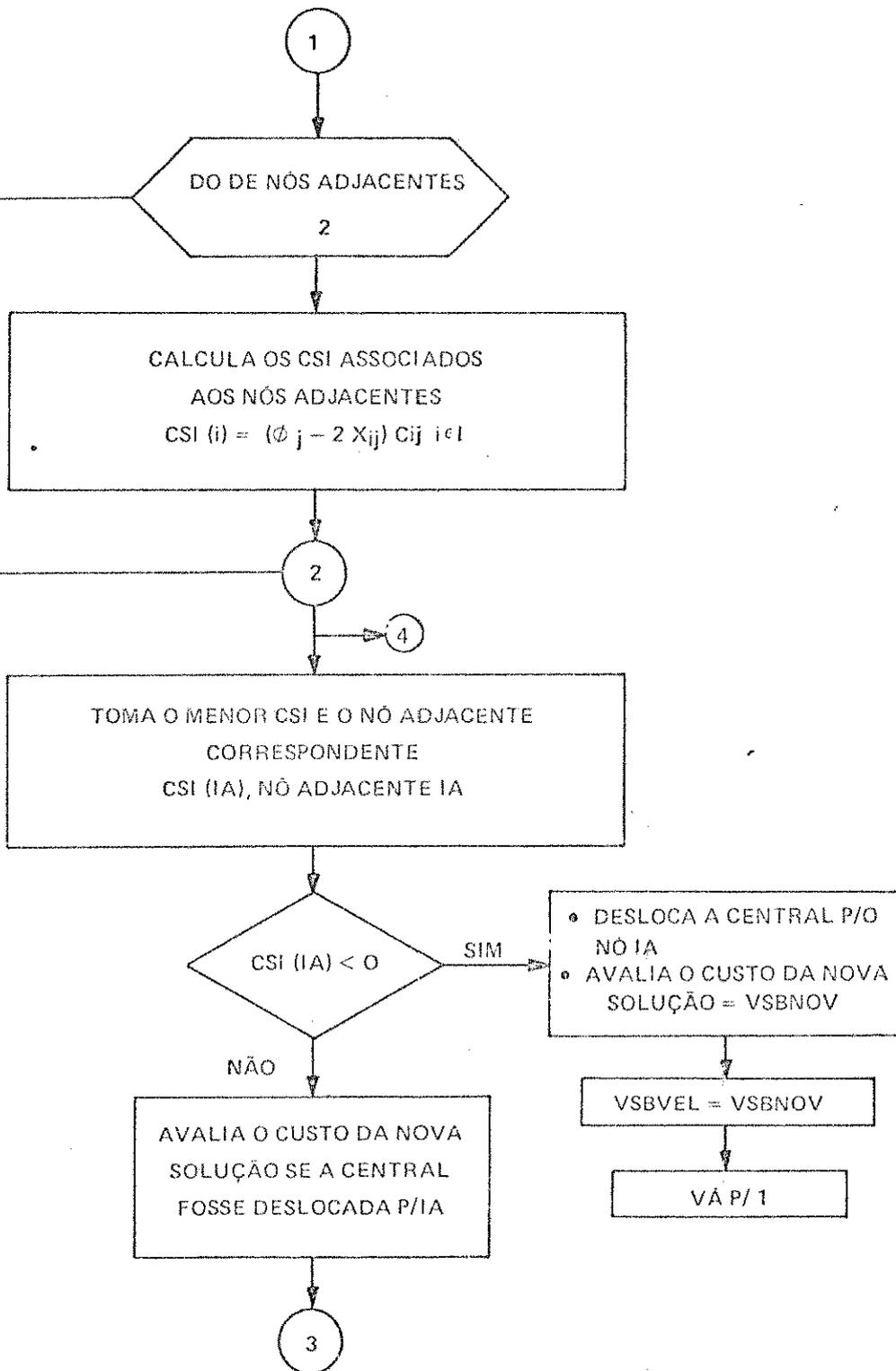
Figura 35 - Central deslocada para o nó 4 e sa-turou

6.4. RESULTADOS OBTIDOS

6.4.1. Fluxograma Resumido do Método

O procedimento descrito nas seções anteriores é apresentado na figura 36, através do fluxograma simplificado do programa FINURA, rotina que realiza o Ajuste Fino, cuja implementação discutiremos adiante.





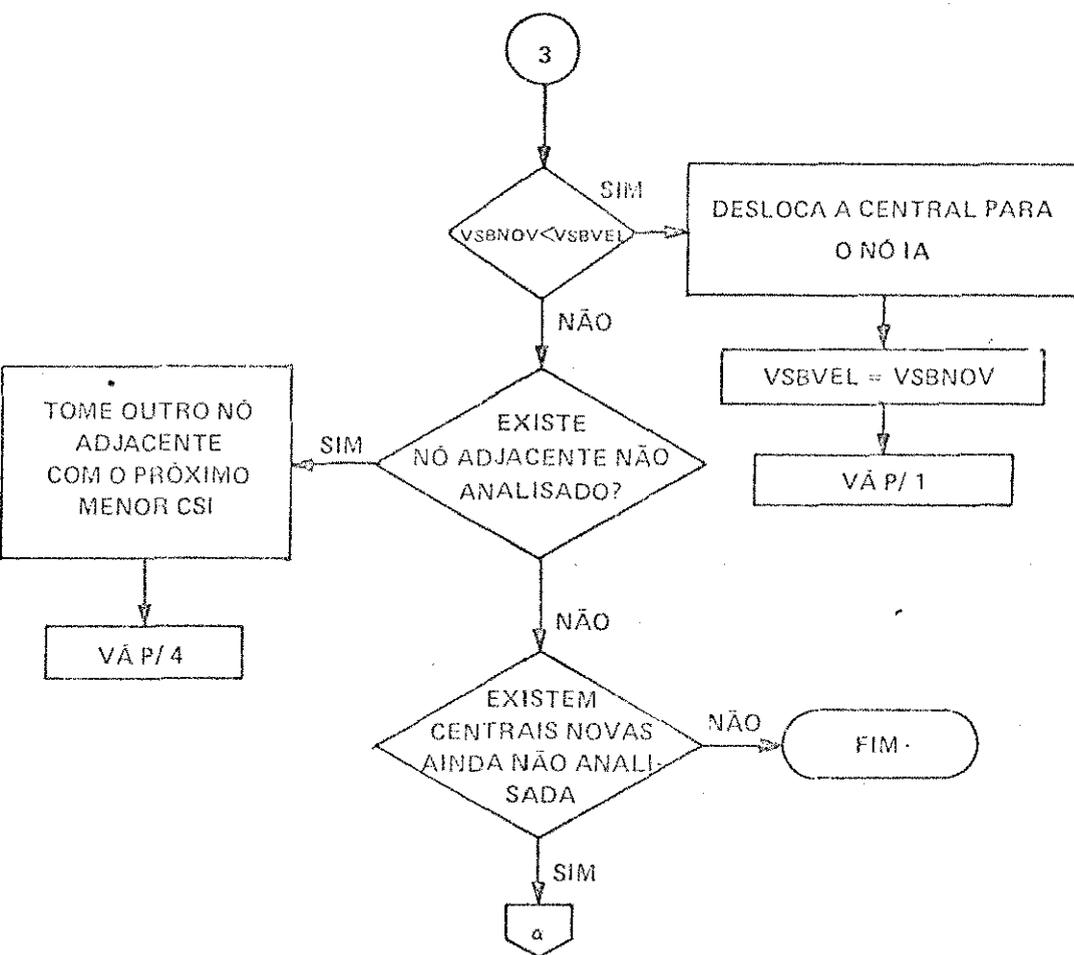


FIGURA 36: FLUXOGRAMA DO MÉTODO DE AJUSTE FINO

6.4.2. Considerações sobre a Implementação do Algoritmo

O algoritmo de AJUSTE FINO que originou o programa FINURA foi implementado em linguagem FORTRAN utilizando um Computador Digital - PDP-10.

O programa FINURA foi incorporado ao programa SUBP, que resolve o problema de fluxo de custo mínimo, já adaptado ao programa TRANZE.

O programa SUBP foi utilizado para a avaliação sucessiva dos custos das soluções com as centrais deslocadas, permitindo a decisão sobre qual seria a solução ótima.

6.4.3. Rede Exemplo

O algoritmo foi aplicado à rede telefônica urbana da cidade de Curitiba. Como já foi visto anteriormente, esta rede é composta por 586 nós e 744 arcos sem orientação.

Após detalhada análise da rede em estudo, foram escolhidos 20 nós candidatos a central, com a restrição adicional que impossibilita a instalação de centrais em candidatos vizinhos.

Os resultados apresentados referem-se à aplicação do algoritmo após obtida a solução ótima através da técnica de partição de Benders, restrita ao espaço dos 20 candidatos.

Torna-se importante reafirmar que o procedimento de Ajuste Fino não se restringe aos 20 candidatos, mas a partir deles explora através de seus adjacentes, os demais nós da rede.

A análise que vamos proceder visa mostrar a eficiência do algoritmo no sentido de aperfeiçoar, em termos de custo, a solução obtida pela técnica de Benders. Esta diminuição se deve ao fato de que ao se deslocar a central de um nó para outro, estamos economizando cabos para a ligação dos nós à central, sem incorrer em acréscimo nos demais custos envolvidos na otimização, a menos da variação no custo do terreno.

Foram escolhidos como candidatos à central os nós apresentados na Tabela 6.

Antes do AJUSTE FINO, obtivemos como solução ótima para o problema a configuração de centrais apresentada na Tabela 7 e Figura 37.

Após o procedimento de Ajuste Fino algumas centrais se deslocaram através de seus nós adjacentes chegando à solução ótima final apresentada na Tabela 8. A trajetória, segundo a qual as centrais se deslocam, é apresentada na figura 38.

Candidatos	nº dos nós
1	49
2	320
3	79
4	398
5	383
6	106
7	134
8	30
9	195
10	264
11	91
12	19
13	518
14	369
15	563
16	72
17	578
18	173
19	125
20	86

Tabela 6 - Nós candidatos à localização das centrais

CENTRAIS LOCALIZADAS ANTES DO AJUSTE FINO

NÓS: 49 320 79 398 134 30 19 578

CUSTO ASSOCIADO À LOCALIZAÇÃO: Cr\$ 7.030.960.100,00

Tabela 7 - Solução ótima gerada pela Técnica de Benders (Antes do Ajuste Fino)

CENTRAIS LOCALIZADAS APÓS O AJUSTE FINO

NÓS: 48 320 78 398 106 477 18 578

CUSTO ASSOCIADO À LOCALIZAÇÃO: Cr\$ 6.964.724.200,00

Tabela 8 - Solução ótima após o AJUSTE FINO

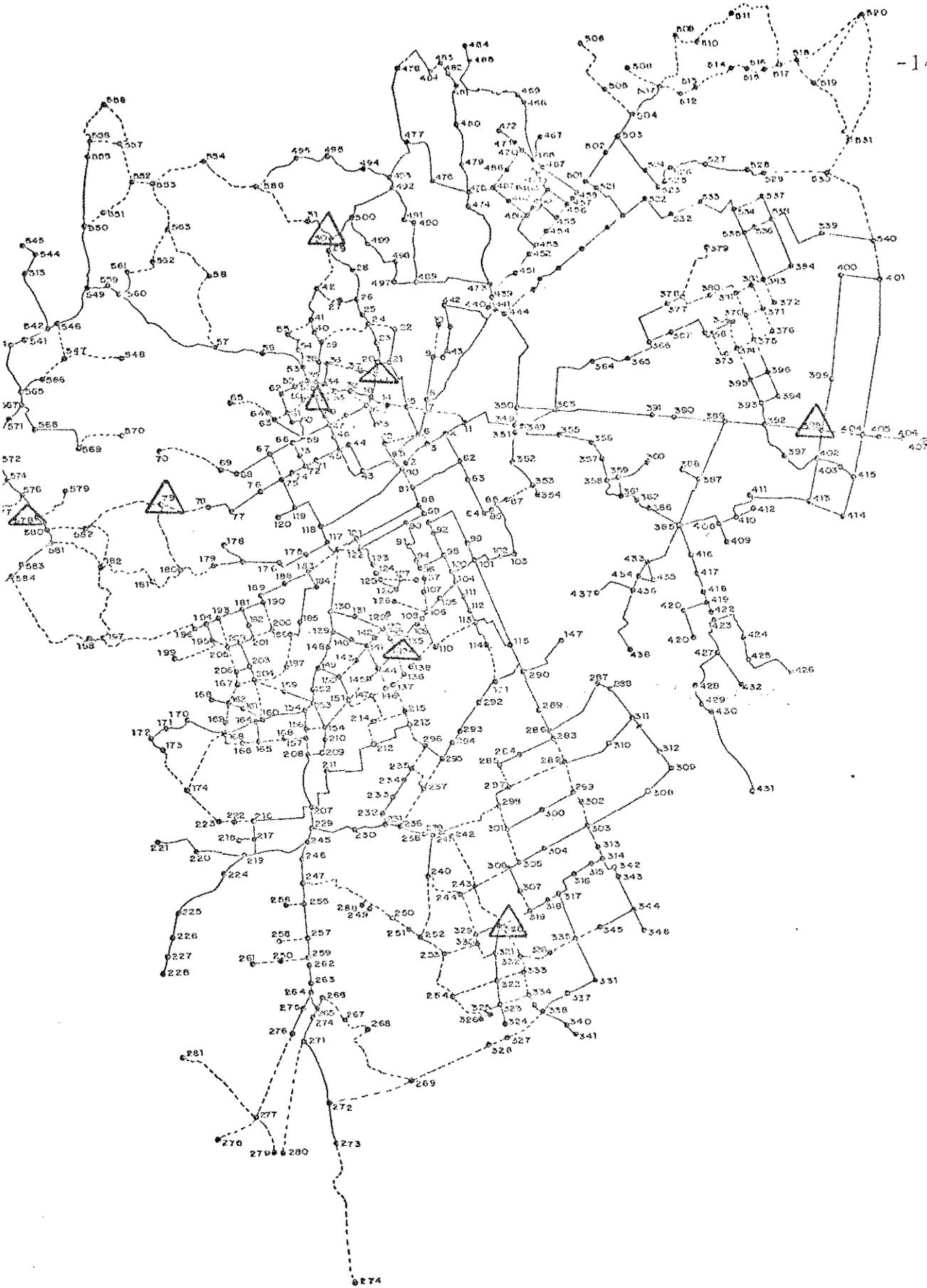


Figura 37 - Localização das Centrais antes do Ajuste Fino

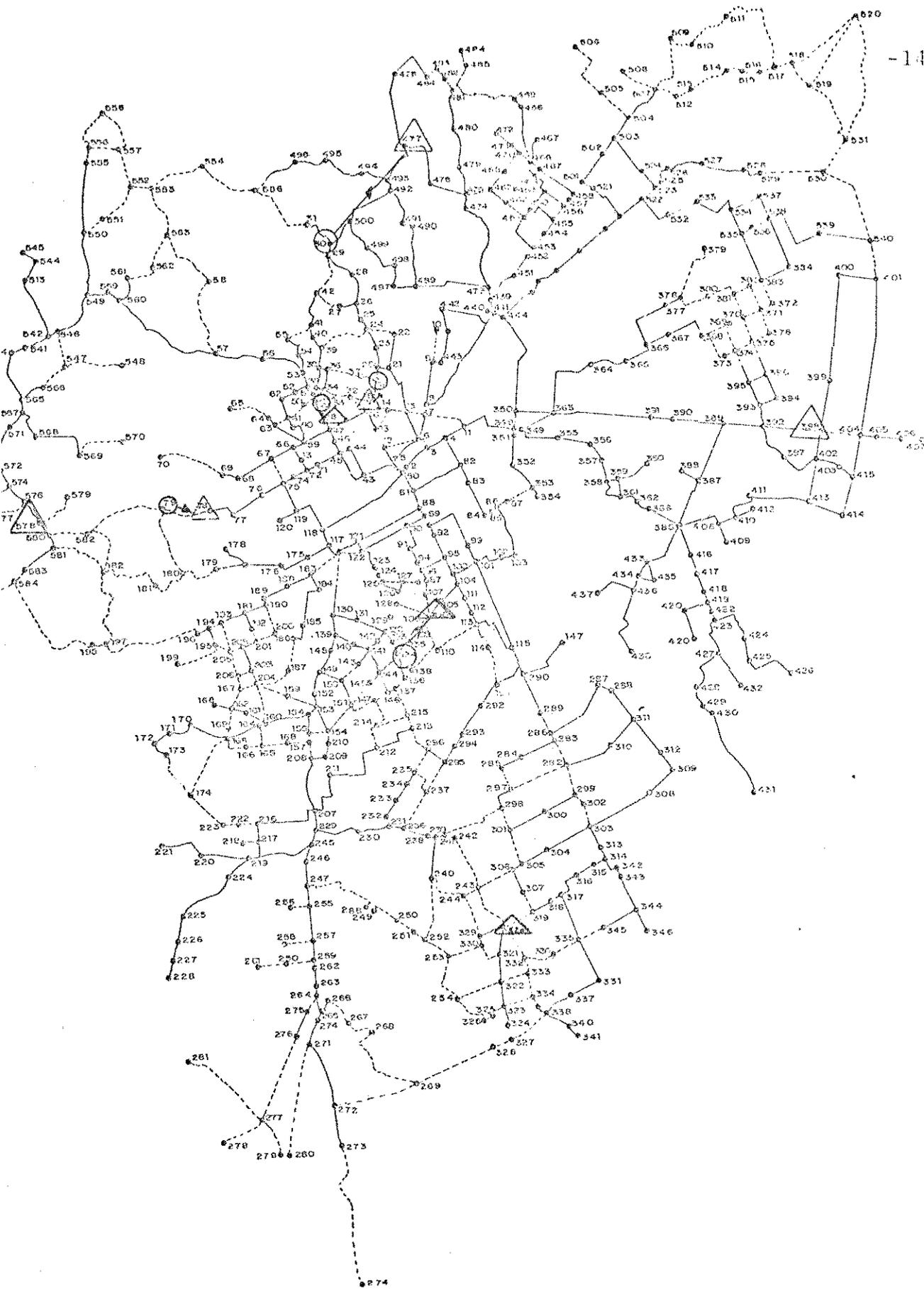


Figura 38 - Trajetória das Centrais durante o AJUSTE FINO

6.4.4. Análise dos Resultados

Comparando-se os resultados das tabelas 7 e 8 podemos observar que o procedimento de AJUSTE FINO foi responsável por uma economia de Cr\$ 66.235.900,00 ou seja cerca de 1%.

É importante observar que esta melhora no custo da solução foi obtida com um tempo de processamento de 4,082 segundos (tempo de CPU). Tal performance vem confirmar as expectativas quanto à adequação do método à aplicação em problemas com esta configuração.

CAPÍTULO 7 - COMENTÁRIOS FINAIS E SUGESTÕES

7.1. INTRODUÇÃO

O objetivo deste capítulo é fazer algumas considerações sobre as potencialidades dos algoritmos desenvolvidos, bem como suas limitações, sugerindo pontos onde estes podem ser aperfeiçoados ou mais exhaustivamente testados.

7.2. CONSIDERAÇÕES SOBRE O MÉTODO HEURÍSTICO DE SOLUÇÃO DO PROBLEMA MESTRE

Como já foi discutido no Capítulo 3, o programa LOCUS-HEURÍSTICO trabalha com linearizações externas, utilizando o último corte gerado (última restrição) como representação da função objetivo T a ser minimizada. Além disso, por tratar-se de um procedimento heurístico, gera restrições mais relaxadas e conseqüentemente menos representativas da função objetivo real T . Assim parece natural sua insensibilidade, no que diz respeito ao número de iterações, às localizações de partida (LP).

Esta constatação está baseada no fato de que o principal obstáculo em se trabalhar com linearização externa é a necessidade de um grande número de aproximantes para

se obter uma boa aproximação da função que se quer representar . Logo, seria necessário um número muito elevado de localizações de partida para se ter uma boa representação da função objetivo "modificada" e conseqüentemente diminuir o número de iterações necessárias para a convergência. Assim, acreditamos que a eficiência do método aumente se considerarmos como função objetivo do problema o epígrafo da função T. Considerando ainda o desempenho do programa LOCUS-HEURÍSTICO, devemos observar que o algoritmo heurístico utilizado só permite a inclusão de variáveis na mudança de uma solução para outra e nunca sua retirada. Esta restrição parece justificar a ocorrência de patamares de soluções, discutidos anteriormente. Dentro deste enfoque julgamos que a utilização de uma heurística mais flexível, que evolua de uma solução para outra, através de inclusão ou retirada de variáveis, venha enriquecer o procedimento da otimização.

Finalmente, embora tendo sido observada a não sensibilidade do programa LOCUS-HEURÍSTICO às localizações de partida, no que diz respeito ao tempo de convergência, é importante notar que uma cuidadosa análise prévia da rede em estudo, pode sugerir boas localizações de partida que venham favorecer a convergência do procedimento heurístico, no que diz respeito a uma menor porcentagem de erro.

7.3. CONSIDERAÇÕES SOBRE A FASE DE TRANSIÇÃO

Como sabemos a filosofia do método SIMPLEX, para a resolução de problemas lineares é de caminhar de uma solução básica factível à outra, sempre garantindo a melhora da função objetivo, sendo contudo necessário, para a inicialização do processo, que se disponha de uma solução básica inicial factível.

No procedimento clássico de busca de uma base inicial factível em uma rede de N nós e M arcos são introduzidos, além dos originais da rede, 2 nós e $N + 1$ arcos, constituindo uma rede de $N + 2$ nós e $M + N + 1$ arcos. Dentre estes arcos M são independentes não básicos. Considerando que o método SIMPLEX retira um arco por vez, a Fase 1 (busca da base inicial factível) terá no mínimo $N + 1$ iterações.

A FASE DE TRANSIÇÃO, como implementada, introduz na rede uma quantidade de arcos artificiais da ordem do número de nós escolhidos para a localização das centrais e não do número total de nós da rede. Desta forma é possível afirmar a superioridade do método de FASE DE TRANSIÇÃO sobre o procedimento convencional.

Analisando os resultados obtidos nos testes realizados com a rede telefônica de Curitiba verificamos o bom desempenho do método de FASE DE TRANSIÇÃO, o qual obteve em número de iterações uma redução média de 75% e de 72% no tempo de CPU.

Cabe ainda ressaltar que nas iterações onde o método de FASE DE TRANSIÇÃO obteve uma menor eficiência, constatou-se uma grande variação entre as soluções conhecida e substituta, tanto no que diz respeito ao número de centrais envolvidas, quanto nas localizações das mesmas.

7.4. CONSIDERAÇÕES SOBRE O AJUSTE FINO

Tentando minimizar os efeitos de se dividir a rede em regiões candidatas e se trabalhar com um nó representante de cada região, foi desenvolvido o procedimento heurístico de AJUSTE FINO, que executa uma análise de pós-otimização sobre todos os nós da rede.

A principal vantagem observada nos testes realizados foi o fato de se conseguir ampliar o universo de candidatos, aliviando as restrições originais, sem incorrer em um peso maior para o programa de otimização. Esta característica se mostra bastante válida quando associada a procedimentos heurísticos de otimização, ou quando se tem que lidar com redes de grande porte, com as quais o planejador não tem grande familiaridade.

Considerando os testes realizados, verificou-se que após a realização de um AJUSTE FINO, a reincidência do procedimento não alterou a solução já anteriormente encontrada. Contudo, mudando-se a ordem de escolha da central,

que irá iniciar o procedimento, a solução se altera. Assim, julgamos que uma avaliação de alternativas de escolha da central inicial seja conveniente, buscando a melhor política de localizações. Este procedimento poderia ser implementado computacionalmente sem nenhuma dificuldade teórica.

BIBLIOGRAFIA

- [1] ARAÚJO, E. de O. — "Localização de Centrais Telefônicas numa Rede Urbana". Tese de Mestrado, UNICAMP, 1981;
- [2] BENDERS, I. F. — Partitioning Procedure for Solving Mixed Variable Programming Problems. "Numerische Mathematik" 4: 238-252, 1962;
- [3] GEOFFRION, A. M. & MARSTEN, R. E. — "Integer Programming Algorithms: A Survey". Perspectives on Optimization: A Collection of Expository Articles, Ed. A.M. Geoffrion, Addison-Wesley, 1972;
- [4] BEALE, E. M. L. — Survey of Integer Programming. "Operational Research Quarterly" 16-nº 2: 219-238, 1965;
- [5] BALINSKI, M. L. — Integer Programming: Methods, Uses, Computation. "Management Science" 12 nº 3: 253-315, 1965;
- [6] SALKIN, H. N. — "Integer Programming". Ed. Addison Wesley Publ. Co, 1975;
- [7] BALAS, E. — An Additive Algorithm for Solving Linear Programs With Zero-One Variables. "Operations Research" 13, 517-546, 1965;

- [8] GEOFFRION, A. M. — Integer Programming by Implicit Enumeration and Balas' Method. "SIAM review" 2: 178-190, 1967;
- [9] ZANAKIS, H. Z. — Heuristic 0-1 Linear Programming: an Experimental Comparison of Three Methods. "Management Science" 2 n° 1: 91-104, 1977;
- [10] SENJU, S. and TOYODA, Y. — An Approach to Linear Programming With 0-1 Variables. "Management Science" 95: 196-207, 1968;
- [11] KOCHENBERGER, G. A., McCARL, B. A. and WYMAN, F. P., — A Heuristic for General Integer Programming. "Decision Sciences" 5 n° 1: 36-44, 1974;
- [12] HILLIER, F. S., — Efficient Heuristic Procedures for Integer Linear Programming With an Interior. "Operations Research" 17: 700-637, 1969;
- [13] PETERSEN, C. C., — Computational Experience With Variants of the Balas Algorithm Applied to the Selection of R & D Projects. "Management Science" 13: 736-750, 1967;
- [14] WEINGARTNER, H. M. and NESS, D. N., — Methods for the Solution of the Multi-Dimensional 0-1 Knapsack Problem. "Operations Research" 15, 1967;

- [15] AUTHIE, G. — Recherche d'un Flot Minimisant une
Fonction de Cout Lineaire : Methode Primale . "Note
Interne n° 18 , LAAS , Mai 1979
- [16] GLOVER, F., KARNEY, D., KLINGMAN, D., NAPIER, A., — A
Computation Study on Start Procedures, Basic Change
Criteria, and Solutions Algorithms for Transportation
Problems. "Management Science", 24, 1974;
- [17] HELGARSON, R. U., KENNINGTON, J. L. — "Algorithms for
Network Programming". John Wiley & Sons, 1980.