

7

Geração de cortes para o problema de alocação de custos de serviços para sistemas de transmissão

7.1

Geração de cortes

Problemas de programação linear com um número exponencial de restrições (ou variáveis) aparecem frequentemente em aplicações práticas. Para alguns desses problemas é possível determinar uma representação implícita de suas restrições, isto é, as restrições não precisam estar completamente armazenadas na memória do computador. Esquemas de geração de cortes para a solução de tais problemas baseiam-se na enumeração de um subconjunto das restrições, na solução de um problema linear restrito, na identificação de uma restrição violada pela solução ótima corrente dentre as restrições ainda não enumeradas (problema de separação) e, na solução do novo problema estendido, até que nenhuma restrição violada possa ser encontrada. Grötschel, Lovasz e Schrijver[64] mostraram que quando existe um algoritmo polinomial para identificar uma restrição violada entre as restrições ainda não enumeradas, o esquema de geração de cortes também será polinomial. Entretanto, para a maior parte das aplicações, não existe um algoritmo polinomial para resolver o problema de separação. Uma forma de contornar essa dificuldade consiste em usar um procedimento heurístico para resolver o problema de separação e aplicar um algoritmo de separação exata apenas quando a heurística não for capaz de encontrar uma restrição violada. O objetivo da separação heurística consiste em reduzir o número de vezes em que o algoritmo exato de separação é aplicado.

Nesse capítulo será apresentado um procedimento paralelo para geração heurística de cortes, usado para a solução de problemas de programação linear com um número exponencial de restrições. Também será estudada a distribuição dos tempos para valor alvo do procedimento usado para fazer a separação heurística.

7.1.1

Um modelo para a alocação de custos de sistemas de transmissão

Seja $G = (V, E)$ um grafo não direcionado associado a uma rede de transmissão. Os nós de G representam usinas, centros de demanda e subestações. Esses nós são conectados por arestas que representam as linhas de transmissão. Para cada conexão, é conhecida a sua capacidade instalada (eventualmente nula) e o seu número máximo de expansões adicionais, assim como a capacidade e o custo incremental de cada expansão realizada. O *Problema de Planejamento de Expansão de Transmissão* (PPET) consiste em determinar o número de expansões (linhas de transmissão) que devem ser adicionadas a cada conexão, de forma que o custo de expansão de uma rede viável capaz de suprir os centros de demanda a partir da energia fornecida pelas usinas seja minimizado. A rede de transmissão é operada por um grupo de K agentes. Cada um desses agentes fornece energia a partir da usina operada por ele para os centros de demanda que tem como clientes. O *Problema de Alocação de Custos de Sistemas de Transmissão* (PACST) consiste na atribuição de custos de expansão a cada agente do grupo, de forma que (1) a soma dos custos atribuídos aos agentes seja igual ao custo total do serviço, e (2) a soma dos custos atribuídos a qualquer subconjunto de agentes de S não exceda o custo em que incorreriam caso decidissem operar o seu próprio subsistema isoladamente.

Campodónico e Pereira [26] formularam o PACST como um problema de programação linear com um número exponencial de restrições. Seja Z^* o custo mínimo de expansão para construir uma rede viável satisfazendo-se as restrições de produção e de demanda dos K agentes do grupo. O valor de Z^* é fornecido como entrada para o PACST e é obtido resolvendo-se o problema de planejamento de expansão de transmissão descrito a seguir. O modelo usado nesse capítulo para o problema de planejamento de expansão de transmissão é uma simplificação do modelo real. Esse problema foi formulado matematicamente por Garver [54] e é tratado usando-se modelos mais realistas em [88, 119, 120, 122, 131]. Seja $b_i^0 > 0$ (respectivamente, $b_i^0 < 0$) a demanda (respectivamente, o fornecimento) associado ao nó $i \in V$. Seja $b_i^k > 0$ (respectivamente, $b_i^k < 0$) a demanda (respectivamente, o fornecimento) resultante da implantação do agente $k \in \{1, \dots, K\}$ no nó $i \in V$. Para cada aresta $(i, j) \in E$, são conhecidos os valores da capacidade inicial instalada $\bar{c}_{ij}$, do custo incremental d_{ij} e da capacidade incremental c_{ij} associados a cada expansão adicional, assim como o número máximo de expansões possíveis u_{ij} . Duas variáveis são associadas a cada aresta: uma variável contínua f_{ij} que representa o fluxo na aresta e uma variável inteira

x_{ij} que denota o número de expansões realizadas. Portanto, o PPET pode ser formulado como:

$$(PPET) \left\{ \begin{array}{ll} \text{Minimizar} & \sum_{(i,j) \in E} d_{ij} x_{ij} \quad (1.0) \\ & \sum_{j:(i,j) \in E} f_{ij} = b_i^0 + \sum_{k=1}^K b_i^k \quad \forall i \in V \quad (1.1) \\ & |f_{ij}| \leq \bar{c}_{ij} + c_{ij} x_{ij} \quad \forall (i,j) \in E \quad (1.2) \\ & 0 \leq x_{ij} \leq u_{ij} \quad \forall (i,j) \in E \quad (1.3) \\ & x_{ij} \in \mathbb{N} \quad \forall (i,j) \in E \quad (1.4) \end{array} \right.$$

Seja Z_S o custo mínimo de expansão associado à operação da subrede por um subconjunto isolado de agentes $S \subseteq \{1, \dots, K\}$ e θ_k o custo atribuído ao agente $k \in \{1, \dots, K\}$. O PACST pode ser formulado como:

$$(PACST) \left\{ \begin{array}{ll} \text{Minimizar} & \Delta \quad (2.0) \\ & \sum_{k=1}^K \theta_k + \Delta = Z^* \quad (2.1) \\ & \sum_{k \in S} \theta_k \leq Z_S \quad \forall S \subseteq \{1, \dots, K\}, \quad S \neq \emptyset \quad (2.2) \\ & \theta_k \geq 0 \quad k = 1, \dots, K \quad (2.3) \\ & \Delta \geq 0 \quad (2.4) \end{array} \right.$$

A variável Δ na função objetivo pode ser vista como uma medida de inviabilidade. As restrições (2.1) e (2.2) representam as condições estabelecidas anteriormente para a atribuição de custos. Um valor ótimo estritamente positivo para a função objetivo significa que as duas condições não podem ser satisfeitas simultaneamente. Os valores ótimos $Z_S, \forall S \subset \{1, \dots, K\}$ mostrados na formulação do PACST são obtidos calculando-se subproblemas $PPET(S)$, onde o total das demandas e fornecimentos são substituídos pelos subtotais correspondentes ao subconjunto S de agentes. O problema $PPET$ é uma generalização do problema conhecido como *Fixed Charge Network Flow* (FCNF) [115]. Como o FCNF é NP-difícil o $PPET$ também é NP-difícil. Portanto, a enumeração completa de todas as restrições requer o cálculo de $2^K - 1$ problemas NP-difíceis (um para cada subconjunto não vazio de agentes), o que na prática não é eficiente.

A solução do PACST por uma abordagem de geração de cortes requer

a solução de um problema de separação a cada iteração. O resultado obtido para cada problema de separação calculado é a restrição do tipo (2.2) mais violada pela solução corrente $\bar{\theta}$, onde $\bar{\theta}_k$ é o custo alocado ao agente $k = 1, \dots, K$. Esse problema de separação foi formulado por Campodónico e Pereira [26] como o problema de programação inteira mista mostrado a seguir, onde variáveis binárias λ_k são associadas a cada agente $k = 1, \dots, K$:

$$\begin{aligned}
 (\text{SEP}) \left\{ \begin{array}{ll} \text{Minimizar} & \sum_{(i,j) \in E} d_{ij} x_{ij} - \sum_{k=1}^K \bar{\theta}_k \lambda_k \quad (3.0) \\ & \sum_{j:(i,j) \in E} f_{ij} = b_i^0 + \sum_{k=1}^K b_i^k \lambda_k \quad \forall i \in V \quad (3.1) \\ & |f_{ij}| \leq \bar{c}_{ij} + c_{ij} x_{ij} \quad \forall (i,j) \in E \quad (3.2) \\ & 0 \leq x_{ij} \leq u_{ij} \quad \forall (i,j) \in E \quad (3.3) \\ & x_{ij} \in \mathbb{N} \quad \forall (i,j) \in E \quad (3.4) \\ & \lambda_k \in \{0, 1\} \quad k = 1, \dots, K \quad (3.5) \end{array} \right.
 \end{aligned}$$

Cada atribuição completa de valores 0-1 às variáveis λ corresponde a um subconjunto de agentes $S = \{k : \lambda_k = 1, k = 1, \dots, K\}$. No problema de separação, para cada uma dessas possíveis atribuições, precisa-se calcular a diferença entre (a) o custo ótimo de expansão associado ao subproblema PPET(S), definido pelo subconjunto S de agentes, e (b) o custo total alocado aos agentes em S de acordo com a solução corrente $\bar{\theta}$. Caso essa diferença seja estritamente negativa para algum subconjunto de agentes S , então a restrição (2.2) associada a S estará violada pela solução corrente. O subconjunto de agentes associado à restrição mais violada será aquele com a menor diferença negativa.

A solução exata de casos de PACST com muitas variáveis requer um alto custo computacional, devido ao tempo necessário para a solução do problema de separação a cada iteração da abordagem de geração de cortes. Na próxima seção, um procedimento para separação heurística será proposto para acelerar a solução do problema de separação.

7.2

Separação heurística

O procedimento de geração de cortes usado para a solução do problema mestre PACST pode ser definido como mostrado a seguir:

Procedimento Solução_PACST:

1. Encontre uma alocação de custos viável $\bar{\theta}$ para o problema de programação linear restrito definido pelas restrições já enumeradas.
2. Resolva o problema de separação SEP, encontrando alguma solução viável que possua valor negativo para a função objetivo (não necessariamente o valor ótimo). Caso uma restrição violada seja encontrada, isto é, uma solução viável para o SEP com valor negativo para a função objetivo, adicione essa restrição ao problema linear restrito corrente e repita o primeiro passo acima.
3. Senão, a alocação de custos corrente é ótima para o problema mestre.

Essa abordagem baseia-se na identificação de uma restrição violada no segundo passo, ou na prova de que não existe nenhuma restrição violada, quando a solução corrente do problema restrito é a solução ótima do problema mestre. Essa etapa pode ser feita, por exemplo, usando-se um procedimento de *branch-and-bound* para resolver o problema SEP. Porém, para acelerar essa etapa do problema mestre, a solução do problema de separação pode ser feita por uma heurística. Dada uma alocação de custos $\bar{\theta}$, deve-se encontrar alguma restrição violada, definida pelas variáveis λ e x de uma solução viável do problema de separação que satisfaça

$$\sum_{(i,j) \in E} d_{ij} x_{ij} < \sum_{k=1}^K \bar{\theta}_k \lambda_k,$$

isto é, uma solução em que o valor da função objetivo no problema de separação seja negativo. O procedimento de separação pode ser implementado para retornar após encontrar qualquer corte (qualquer solução viável para o SEP que apresente um valor negativo para a função objetivo) e não necessariamente a restrição mais violada. Dessa forma, o seguinte procedimento baseado na solução heurística do problema de separação é proposto para acelerar a solução do PACST:

Procedimento Solução_Acelerada_PACST:

1. Encontre uma alocação de custos viável $\bar{\theta}$ para o problema de programação linear restrito definido pelas restrições já enumeradas.

2. Resolva o problema de separação SEP, usando uma heurística para obter uma solução viável que apresente um valor negativo para a função objetivo (não necessariamente a solução ótima). Caso uma restrição violada seja encontrada, adicione essa restrição ao problema linear restrito corrente e repita o primeiro passo acima. Senão, execute o passo abaixo.
3. Aplique o procedimento de separação exata para identificar uma restrição violada. Nesse caso, a solução ótima do problema de separação SEP é obtida. Como no passo anterior, caso uma restrição violada seja encontrada, adicione essa restrição ao problema linear restrito corrente e repita o primeiro passo.
4. Caso nenhuma restrição violada seja encontrada, então a alocação de custos corrente é ótima para o problema mestre.

O algoritmo de separação heurística é formado por dois componentes principais:

- Uma *busca local* no espaço de subconjuntos de agentes (ou, de forma equivalente, no espaço de vetores característicos λ), para obter o subconjunto de agentes que determinará uma restrição violada; e
- A *construção heurística* de uma rede viável para um subconjunto de agentes S investigado durante a busca local resolvendo $\text{PPET}(S)$, que corresponde a uma solução viável x para o problema de separação SEP associada ao vetor característico λ .

A *construção heurística* de uma rede viável onde é calculada de forma aproximada a solução do problema $\text{PPET}(S)$ para um subconjunto de agentes S é descrita em [115]. Esse procedimento é composto por duas fases. Na primeira fase, uma solução viável é construída através da solução de uma seqüência de problemas de fluxo máximo. Caso as capacidades correntemente instaladas não sejam suficientes para permitir o envio do fluxo definido pelos fornecimentos e demandas dos agentes em S , as capacidades de algumas arestas do corte mínimo são aumentadas (através da solução de um pequeno problema da mochila) e um novo problema de fluxo máximo é calculado, até que uma solução viável seja encontrada. A segunda fase é um procedimento usado para diminuir o número de expansões feitas em cada aresta, através do rerroteamento do fluxo corrente (possivelmente aumentando a capacidade de arestas que possuam um custo inferior). Uma vez que o valor ótimo aproximado $\sum_{(i,j) \in E} d_{ij}x_{ij}$ do problema

$PPET(S)$ seja determinado pela heurística de duas fases descrita acima, um valor aproximado da violação associada à restrição definida por S é computado subtraindo-se $\sum_{k=1}^K \bar{\theta}_k \lambda_k$ do valor ótimo aproximado calculado para $PPET(S)$. Caso a diferença seja negativa, então uma restrição violada terá sido encontrada.

A *busca local* é iniciada a partir de um subconjunto de agentes, que poderá ser: (i) o conjunto vazio (nenhum agente), (ii) o conjunto completo de agentes (todos os agentes), (iii) um subconjunto de agentes gerado aleatoriamente, ou (iv) um subconjunto de agentes associado a um dos cortes gerados na iteração anterior. A vizinhança V da solução corrente é formada por todos os subconjuntos de agentes que diferem do subconjunto corrente por exatamente um agente (inserções ou eliminações de exatamente um agente). Para cada uma dessas soluções vizinhas, definidas por um subconjunto de agentes S , o problema $PPET(S)$ é calculado de forma aproximada através da heurística de duas fases descrita no parágrafo anterior. Caso a diferença $\sum_{(i,j) \in E} d_{ij} x_{ij} - \sum_{k=1}^K \bar{\theta}_k \lambda_k$ seja negativa, então uma restrição violada terá sido encontrada e o procedimento de separação heurística é terminado, retornando o corte encontrado. Caso contrário, o procedimento de busca local prossegue com a seleção aleatória de um dos vizinhos da solução corrente. Para isso, é montada uma lista restrita de candidatos (LRC) conforme feito para o método GRASP. Seja $c(S)$ o valor da diferença $\sum_{(i,j) \in E} d_{ij} x_{ij} - \sum_{k=1}^K \bar{\theta}_k \lambda_k$ para uma solução representada pelo conjunto de agentes S , na vizinhança V da solução corrente. Sejam $c(\underline{S})$ e $c(\bar{S})$ os valores da menor e da maior diferença para as soluções na vizinhança V da solução corrente, respectivamente. A lista restrita de candidatos LRC será formada por soluções v na vizinhança V tais que $LRC = \{v \in V \mid c(v) \leq c(\underline{v}) + \alpha(c(\bar{v}) - c(\underline{v}))\}$, onde α é um parâmetro do programa, selecionado aleatoriamente no intervalo $[0, 1]$ a cada vez que o procedimento de separação heurística é executado. Uma solução será escolhida aleatoriamente da LRC e o movimento será feito para essa solução. O procedimento de busca local é terminado quando uma restrição violada é identificada e adicionada ao problema restrito, ou após examinar um certo número **MaxTentativas** de vizinhanças, sem que uma restrição violada tenha sido encontrada.

O pseudo-código para o procedimento de separação heurística é apresentado na Figura 7.1. O procedimento **Separacao_Heuristica** retorna uma restrição violada para uma dada alocação de custos $\bar{\theta}_k, k = 1, \dots, K$, caso seja possível encontrá-la em no máximo **MaxTentativas** iterações. Caso contrário, o procedimento de separação exata é executado.

```

procedimento Separação_Heurística(seed, dados_rede, dados_agentes,  $\bar{\theta}$ )
1   tentativas = 0;
2   limite = 0;
3   Gere um subconjunto inicial de agentes S;
4    $\bar{\Theta}_S = \sum_{k \in S} \bar{\theta}_S$ ;
5    $B_S = \sum_{k \in S} b_S$ ;
6   Resolva PPET(S) com uma heurística, obtendo uma solução viável x;
7    $Z_S = \sum_{(i,j) \in E} d_{ij} x_{ij}$ ;
8   se  $Z_S - \bar{\Theta}_S < \textit{limite}$ 
9       então Achou = verdadeiro;
10      senão Achou = falso;
11  fim-se;
12  enquanto Achou == falso e tentativas < MaxTentativas faça
13      para  $k = 1, \dots, K$  faça
14          se  $k \in S$ 
15              então remova agente k:  $S^k = S \setminus \{k\}$ ;  $\bar{\Theta}_S^k = \bar{\Theta}_S - \bar{\theta}_k$ ;  $B_S^k = B_S - b_k$ ;
16              senão insira agente k:  $S^k = S \cup \{k\}$ ;  $\bar{\Theta}_S^k = \bar{\Theta}_S + \bar{\theta}_k$ ;  $B_S^k = B_S + b_k$ ;
17              fim-se;
18              Resolva PPET( $S^k$ ) com uma heurística, obtendo uma solução viável x;
19               $Z_S^k = \sum_{(i,j) \in E} d_{ij} x_{ij}$ ;
20              se  $Z_S^k - \bar{\Theta}_S^k < \textit{limite}$  então
21                  Achou = verdadeiro;
22                   $m = k$ ;
23              fim-se;
24          fim-para;
25          se Achou == falso então
26              Construa a LRC usando valores de  $Z_S^k - \bar{\Theta}_S^k, k = 1, \dots, K$ ;
27              Selecione aleatoriamente um agente m da LRC;
28          fim-se;
29           $S = S^m$ ;
30           $Z_S = Z_S^m$ ;
31           $\bar{\Theta}_S = \bar{\Theta}_S^m$ ;
32           $B_S = B_S^m$ ;
33          tentativas = tentativas + 1;
34      fim-enquanto;
35      retorne (S, ZS, Achou);
fim Separação_Heurística
    
```

Figura 7.1: Procedimento de separação heurística.

7.3

Estratégia paralela

Nessa seção será descrito o procedimento paralelo de geração de cortes usando $\rho \geq 2$ processos. Um procedimento de *branch-and-bound* será sempre executado por um processo, enquanto que os $\rho - 1$ processos restantes irão executar uma separação heurística. Cada processo executando uma separação heurística será iniciado de uma solução diferente (um subconjunto diferente de agentes). Além disso, vários cortes podem ser identificados em cada iteração do procedimento de geração de cortes. Uma nova iteração do problema mestre é iniciada caso pelo menos uma restrição violada tenha sido encontrada. Quando todos os procedimentos de separação heurística falham para encontrar um corte, a iteração corrente continua até que o processo que executa a separação exata encontre o corte mais violado ou determine que não existem mais cortes.

A Figura 7.2 descreve em pseudo-código o procedimento de geração de cortes paralelo. O processo que executa a separação exata será usado como um processo centralizador das informações obtidas globalmente pelos processos usados na aplicação. Na linha 1 o identificador de cada processo é determinado. O identificador do processo central será $meu_rank = 0$. O identificador do processo é usado para alterar as sementes usadas pelo gerador de números aleatórios na linha 4, de forma que cada processo será iniciado por uma semente diferente (essas sementes são usadas pelos procedimentos de separação heurística). O processo central inicializa o problema linear restrito com a restrição $\sum_{k=1}^K \theta_k = Z^*$ na linha 3. No laço das linhas 7 a 31 os processos irão iteragir de forma síncrona, até que uma alocação viável de custos seja encontrada.

```

procedimento Geração_Cortes_Paralelo(seed, dados_rede, dados_agentes)
1   meu_rank = OBTER_RANK();
2   se meu_rank == 0
3       então Inicialize o problema linear restrito com  $\sum_{k=1}^K \theta_k = Z^*$ ;
4       senão seed = seed + meu_rank;
5   fim-se;
6   alocacao_viavel = falso;
7   enquanto alocacao_viavel == falso faça
8       Sincronize os processos;
9       se meu_rank == 0 então
10          Resolva o problema linear restrito, obtendo alocação de custos  $\bar{\theta}$ ;
11          ENVIAR_PARA_TODOS( $\bar{\theta}$ );
12          Resolva o problema SEP usando a separação exata;
13          restricoes_recebidas = RECEBE_RESTRIÇÕES();
14          se restricoes_recebidas == 0 então
15              Enviar mensagens para interromper separações heurísticas;
16          fim-se;
17          se Restrições foram encontradas globalmente
18              então Adicione as restrições ao problema linear restrito;
19              senão alocacao_viavel = verdadeiro;
20          fim-se;
21          ENVIAR_PARA_TODOS(alocacao_viavel);
22      senão
23          RECEBE( $\bar{\theta}$ );
24          Separação_Heurística(seed, dados_rede, dados_agentes,  $\bar{\theta}$ );
25          se Restrições foram encontradas pelo processo meu_rank então
26              Envie restrições encontradas para processo 0;
27              Enviar mensagens para interromper demais separações;
28          fim-se;
29          RECEBE(alocacao_viavel);
30      fim-se;
31  fim-enquanto;
32  retorne ( $\bar{\theta}$ );
fim Geração_Cortes_Paralelo
    
```

Figura 7.2: Procedimento paralelo de geração de cortes.

A cada iteração do laço das linhas 7 a 31, o processo que executa a separação exata irá resolver o problema linear restrito, obtendo uma alocação de custos $\bar{\theta}$ na linha 10. Em seguida, na linha 11, esse processo enviará para os demais processos os valores calculados para $\bar{\theta}$. Na linha 12, o processo de separação exata resolve o problema de separação através de um algoritmo *branch-and-bound*. Durante a execução do procedimento de *branch-and-bound*, são feitas chamadas a uma rotina que verifica a existência de alguma mensagem que indique que o procedimento de *branch-and-bound* deve ser interrompido. Caso esse tipo de mensagem seja recebida, o procedimento de separação exata é finalizado. Na linha 13, o processo de separação exata recebe restrições enviadas pelos processos de separação heurística. Caso o processo central não tenha recebido nenhuma restrição dos processos de separação heurística, é necessário que uma mensagem seja enviada a cada um desses processos para que eles interrompam as suas buscas. Isso é feito nas linhas 14 a 16 do pseudo-código. Na linha 18, caso restrições violadas tenham sido encontradas por algum processo (exato ou heurístico) na iteração corrente, o processo central adiciona essas novas restrições ao problema linear restrito. Caso nenhuma restrição violada tenha sido encontrada globalmente, o processo central atribui à variável de controle *alocacao_viavel* o valor **verdadeiro**, na linha 19, indicando que uma alocação viável foi encontrada. Na linha 21, o processo central envia aos demais uma mensagem indicando o estado da alocação de custos corrente.

Um processo de separação heurística inicia cada iteração com o recebimento dos valores de $\bar{\theta}$ (linha 23) calculados pelo processo central. Em seguida, na linha 24, o procedimento de separação heurística é executado para os valores $\bar{\theta}$ recebidos. A cada iteração da busca local do procedimento de separação heurística, verifica-se a existência de mensagens enviadas pelos outros processos, que indiquem que o procedimento de separação heurística deve ser interrompido. Caso esse tipo de mensagem seja recebida, ou caso algum corte tenha sido encontrado, o procedimento de separação heurística é terminado. As restrições violadas encontradas por cada processo de separação heurística são enviadas para o processo central na linha 26 para serem inseridas no problema linear restrito. Além disso, quando uma restrição violada é encontrada por um processo de separação heurística, ele envia uma mensagem para os demais processos indicando que todos os procedimentos de separação devem ser interrompidos (linha 27). Finalmente, na linha 29, os processos de separação heurística recebem uma mensagem enviada pelo processo central indicando se a alocação de custos corrente é viável.

Caso uma alocação viável de custos tenha sido encontrada, os proces-

sos irão sair do laço das linhas 7 a 31 e a alocação de custos obtida será retornada na linha 32.

7.4

Resultados computacionais

Nessa seção serão apresentados os resultados computacionais do programa seqüencial, assim como os da abordagem paralela de geração de cortes desenvolvida para o PACST. Os programas testados são:

1. **SEP-E**: programa de geração de cortes que faz somente separações exatas;
2. **SEP-H**: programa de geração de cortes que usa heurísticas para acelerar a etapa de separação, como descrito na Seção 7.2. Um número máximo de iterações para o procedimento de separação heurística `MaxTentativas=200` foi usado nesses experimentos;
3. **SEP-PAR**: programa paralelo para geração de cortes para o PACST, descrito na Seção 7.3.

7.4.1

Ambiente computacional

Os experimentos foram feitos em um computador SGI Challenge com 16 processadores MIPS R10000 de 196 MHz, 12 processadores MIPS R10000 de 194 MHz e 7.6 Gb de memória.

Os programas foram escritos em C e foram compilados com o compilador SGI MIPSpro C usando as opções `-Ofast -64`. O código paralelo usa a implementação da especificação do MPI 1.2, encontrada no *Message Passing Toolkit* 1.4 da SGI. O CPLEX versão 5.0 foi usado para resolver os problemas de programação linear e os problemas de programação inteira mista de forma exata (*branch-and-bound*).

No programa seqüencial, os tempos corridos foram medidos usando-se a função `gettimeofday` do sistema operacional. Os tempos corridos do programa paralelo foram medidos usando-se a função `MPI_WT` do MPI. Deve-se notar que ao contrário das estratégias apresentadas para o algoritmo GRASP, os programas **SEP-H** e **SEP-PAR** executando em um processador são diferentes. O **SEP-PAR** executando em um processador faz somente a separação exata a cada iteração, enquanto que o **SEP-H** faz a separação exata somente quando o procedimento de separação heurística não encontra uma

Tabela 7.1: Dados dos agentes instalados na rede elétrica para o problema 19-1. Para cada agente são mostrados o índice, os índices dos nós da rede a partir de onde ele fornece e consome energia e sua demanda em unidades de potencia.

agente	nó de fornecimento	nó de consumo	potência(MW)
01	244	234	317.0
02	253	41	154.0
03	253	218	345.0
04	227	218	46.0
05	251	223	531.0
06	251	234	1259.0
07	211	234	1631.0
08	227	235	49.0
09	202	239	405.0
10	211	239	207.0
11	21	246	212.0
12	211	254	647.0
13	211	218	100.0
14	211	235	60.0
15	246	242	100.0
16	242	256	84.0
17	211	252	153.0
18	211	256	155.0
19	212	256	132.0

restrição violada. Portanto, as acelerações médias são calculadas dividindo-se os tempos de SEP-H pelos tempos de SEP-PAR para dois, três, cinco e nove processadores. Como SEP-H não usa a biblioteca MPI, funções diferentes são usadas para medir o tempo de execução de cada abordagem. Os tempos medidos não levam em conta o tempo necessário para fazer a entrada dos dados, inicializar o gerador de números aleatórios e imprimir a saída dos dados.

Os números aleatórios usados pelo procedimento de separação heurística são obtidos através de chamadas ao gerador de números pseudo-aleatórios fornecido pelo compilador C.

7.4.2 Problemas testes

Os programas desenvolvidos foram testados para um caso resumido do problema de alocação de custos em sistemas de transmissão, derivado da rede elétrica brasileira. A rede testada é formada por 79 nós e 283 arestas (onde 134 podem ser replicadas) e é operada por um conjunto de 19 agentes. A Tabela 7.1 mostra os dados dos 19 agentes implatados na rede. Nas Tabelas 7.2, 7.3, 7.4, 7.5, 7.6 e 7.7 são mostrados os dados das arestas da rede. As Tabelas 7.8 e 7.9 mostram os dados dos nós da rede.

Tabela 7.2: Dados das arestas (1 a 50) da rede elétrica usada nos problemas testados. Para cada aresta é mostrado o índice, os índices dos seus nós de destino e origem, a situação (instalada ou projetada), a capacidade máxima em unidades de potência, o custo por unidade de demanda e o número máximo de expansões. O custo de cada expansão é obtido multiplicando-se a capacidade máxima da aresta pelo custo por unidade de demanda.

aresta	origem	destino	situação	capacidade máxima	custo por MW	expansões
1	243	267	instalada	360.00	0	0
2	267	272	instalada	360.00	0	0
3	260	254	instalada	660.00	0	0
4	201	33	instalada	660.00	0	0
5	201	202	instalada	660.00	0	0
6	201	202	instalada	660.00	0	0
7	202	204	instalada	660.00	0	0
8	204	205	instalada	660.00	0	0
9	202	205	instalada	660.00	0	0
10	202	203	instalada	660.00	0	0
11	205	210	instalada	660.00	0	0
12	205	210	instalada	660.00	0	0
13	210	256	instalada	660.00	0	0
14	210	256	instalada	660.00	0	0
15	261	53	instalada	1320.00	0	0
16	261	53	instalada	1320.00	0	0
17	261	53	instalada	1320.00	0	0
18	21	53	instalada	1320.00	0	0
19	21	53	instalada	1320.00	0	0
20	21	53	instalada	1320.00	0	0
21	12	21	instalada	1320.00	0	0
22	12	21	instalada	1320.00	0	0
23	12	21	instalada	1320.00	0	0
24	37	40	instalada	120.00	0	0
25	37	40	instalada	120.00	0	0
26	38	40	instalada	120.00	0	0
27	38	41	instalada	120.00	0	0
28	38	39	instalada	120.00	0	0
29	38	39	instalada	120.00	0	0
30	39	41	instalada	120.00	0	0
31	263	41	instalada	120.00	0	0
32	205	208	instalada	660.00	0	0
33	208	216	instalada	660.00	0	0
34	260	223	instalada	660.00	0	0
35	260	223	instalada	660.00	0	0
36	260	208	instalada	660.00	0	0
37	260	208	instalada	660.00	0	0
38	260	208	instalada	660.00	0	0
39	260	208	instalada	660.00	0	0
40	260	216	instalada	660.00	0	0
41	260	256	instalada	660.00	0	0
42	216	254	instalada	660.00	0	0
43	216	256	instalada	660.00	0	0
44	255	256	instalada	660.00	0	0
45	206	255	instalada	660.00	0	0
46	206	216	instalada	660.00	0	0
47	203	216	instalada	660.00	0	0
48	203	216	instalada	660.00	0	0
49	203	206	instalada	660.00	0	0
50	203	208	instalada	660.00	0	0

Tabela 7.3: Dados das arestas (51 a 100) da rede elétrica usada nos problemas testados. Para cada aresta é mostrado o índice, os índices dos seus nós de destino e origem, a situação (instalada ou projetada), a capacidade máxima em unidades de potência, o custo por unidade de demanda e o número máximo de expansões. O custo de cada expansão é obtido multiplicando-se a capacidade máxima da aresta pelo custo por unidade de demanda.

aresta	origem	destino	situação	capacidade máxima	custo por MW	expansões
51	203	208	instalada	660.00	0	0
52	203	208	instalada	660.00	0	0
53	203	208	instalada	660.00	0	0
54	200	233	instalada	720.00	0	0
55	213	214	instalada	360.00	0	0
56	214	246	instalada	360.00	0	0
57	214	219	instalada	360.00	0	0
58	219	227	instalada	360.00	0	0
59	219	224	instalada	360.00	0	0
60	224	227	instalada	360.00	0	0
61	224	241	instalada	360.00	0	0
62	224	225	instalada	360.00	0	0
63	225	241	instalada	360.00	0	0
64	221	224	instalada	360.00	0	0
65	221	224	instalada	360.00	0	0
66	221	241	instalada	360.00	0	0
67	221	241	instalada	360.00	0	0
68	229	241	instalada	360.00	0	0
69	227	229	instalada	360.00	0	0
70	227	229	instalada	360.00	0	0
71	212	215	instalada	720.00	0	0
72	212	215	instalada	720.00	0	0
73	215	222	instalada	720.00	0	0
74	222	228	instalada	720.00	0	0
75	200	217	instalada	720.00	0	0
76	217	228	instalada	720.00	0	0
77	228	234	instalada	810.00	0	0
78	228	234	instalada	810.00	0	0
79	228	232	instalada	720.00	0	0
80	232	234	instalada	720.00	0	0
81	232	234	instalada	720.00	0	0
82	232	234	instalada	720.00	0	0
83	200	228	instalada	720.00	0	0
84	211	220	instalada	720.00	0	0
85	250	251	instalada	780.00	0	0
86	250	251	instalada	780.00	0	0
87	220	242	instalada	720.00	0	0
88	226	242	instalada	720.00	0	0
89	220	273	instalada	720.00	0	0
90	231	273	instalada	720.00	0	0
91	242	273	instalada	720.00	0	0
92	240	257	instalada	720.00	0	0
93	226	231	instalada	720.00	0	0
94	226	257	instalada	720.00	0	0
95	231	240	instalada	720.00	0	0
96	246	247	instalada	360.00	0	0
97	246	247	instalada	360.00	0	0
98	247	249	instalada	360.00	0	0
99	247	249	instalada	360.00	0	0
100	215	217	instalada	720.00	0	0

Tabela 7.4: Dados das arestas (101 a 150) da rede elétrica usada nos problemas testados. Para cada aresta é mostrado o índice, os índices dos seus nós de destino e origem, a situação (instalada ou projetada), a capacidade máxima em unidades de potência, o custo por unidade de demanda e o número máximo de expansões. O custo de cada expansão é obtido multiplicando-se a capacidade máxima da aresta pelo custo por unidade de demanda.

aresta	origem	destino	situação	capacidade máxima	custo por MW	expansões
101	229	243	instalada	360.00	0	0
102	229	236	instalada	360.00	0	0
103	236	243	instalada	360.00	0	0
104	229	235	instalada	360.00	0	0
105	228	233	instalada	720.00	0	0
106	235	252	instalada	360.00	0	0
107	218	221	instalada	360.00	0	0
108	262	218	instalada	360.00	0	0
109	262	221	instalada	360.00	0	0
110	262	221	instalada	360.00	0	0
111	262	221	instalada	360.00	0	0
112	230	241	instalada	360.00	0	0
113	230	241	instalada	360.00	0	0
114	230	237	instalada	360.00	0	0
115	230	237	instalada	360.00	0	0
116	237	238	instalada	360.00	0	0
117	237	238	instalada	360.00	0	0
118	238	239	instalada	360.00	0	0
119	238	239	instalada	360.00	0	0
120	243	252	instalada	360.00	0	0
121	207	209	instalada	720.00	0	0
122	209	211	instalada	720.00	0	0
123	209	226	instalada	720.00	0	0
124	207	212	instalada	720.00	0	0
125	12	27	instalada	990.00	0	0
126	12	27	instalada	990.00	0	0
127	12	27	instalada	990.00	0	0
128	12	27	instalada	990.00	0	0
129	37	205	instalada	90.00	0	0
130	37	205	instalada	90.00	0	0
131	207	206	instalada	450.00	0	0
132	249	250	instalada	336.00	0	0
133	249	250	instalada	336.00	0	0
134	249	250	instalada	336.00	0	0
135	221	222	instalada	336.00	0	0
136	217	218	instalada	336.00	0	0
137	234	237	instalada	336.00	0	0
138	234	237	instalada	336.00	0	0
139	211	246	instalada	336.00	0	0
140	226	227	instalada	240.00	0	0
141	226	227	instalada	240.00	0	0
142	226	227	instalada	240.00	0	0
143	231	243	instalada	240.00	0	0
144	231	243	instalada	240.00	0	0
145	231	243	instalada	240.00	0	0
146	272	273	instalada	240.00	0	0
147	272	273	instalada	240.00	0	0
148	212	213	instalada	336.00	0	0
149	257	252	instalada	240.00	0	0
150	257	252	instalada	240.00	0	0

Tabela 7.5: Dados das arestas (151 a 200) da rede elétrica usada nos problemas testados. Para cada aresta é mostrado o índice, os índices dos seus nós de destino e origem, a situação (instalada ou projetada), a capacidade máxima em unidades de potência, o custo por unidade de demanda e o número máximo de expansões. O custo de cada expansão é obtido multiplicando-se a capacidade máxima da aresta pelo custo por unidade de demanda.

aresta	origem	destino	situação	capacidade máxima	custo por MW	expansões
151	233	223	instalada	540.00	0	0
152	200	260	instalada	5999.40	0	0
153	200	261	instalada	5999.40	0	0
154	200	262	instalada	5999.40	0	0
155	200	263	instalada	5999.40	0	0
156	243	267	projetada	360.00	27.06	3
157	267	272	projetada	360.00	17.61	3
158	260	254	projetada	660.00	28.58	1
159	201	33	projetada	660.00	3.57	3
160	201	202	projetada	660.00	19.65	3
161	202	204	projetada	660.00	14.29	3
162	204	205	projetada	660.00	18.93	3
163	202	205	projetada	660.00	32.68	3
164	202	203	projetada	660.00	39.29	3
165	205	210	projetada	660.00	37.51	3
166	210	256	projetada	660.00	18.22	3
167	261	53	projetada	1320.00	86.76	1
168	21	53	projetada	1320.00	73.74	1
169	37	40	projetada	120.00	4.08	2
170	38	40	projetada	120.00	2.80	2
171	38	41	projetada	120.00	10.96	2
172	38	39	projetada	120.00	4.40	2
173	39	41	projetada	120.00	7.04	2
174	263	41	projetada	120.00	16.00	2
175	205	208	projetada	660.00	24.47	3
176	208	216	projetada	660.00	18.57	3
177	260	223	projetada	660.00	25.90	2
178	260	208	projetada	660.00	42.33	2
179	260	216	projetada	660.00	52.33	3
180	260	256	projetada	660.00	26.79	3
181	216	254	projetada	660.00	28.58	3
182	216	256	projetada	660.00	25.72	3
183	255	256	projetada	660.00	43.40	3
184	206	255	projetada	660.00	56.62	3
185	206	216	projetada	660.00	58.05	3
186	203	216	projetada	660.00	66.08	2
187	203	206	projetada	660.00	26.08	3
188	203	208	projetada	660.00	54.47	3
189	200	233	projetada	720.00	16.84	1
190	213	214	projetada	360.00	9.73	3
191	214	246	projetada	360.00	24.94	1
192	214	219	projetada	360.00	5.49	1
193	224	227	projetada	360.00	2.99	1
194	224	241	projetada	360.00	16.46	3
195	224	225	projetada	360.00	4.12	3
196	225	241	projetada	360.00	13.09	3
197	221	241	projetada	360.00	16.83	2
198	212	215	projetada	720.00	37.07	2
199	215	222	projetada	720.00	31.52	3
200	222	228	projetada	720.00	34.21	3

Tabela 7.6: Dados das arestas (201 a 250) da rede elétrica usada nos problemas testados. Para cada aresta é mostrado o índice, os índices dos seus nós de destino e origem, a situação (instalada ou projetada), a capacidade máxima em unidades de potência, o custo por unidade de demanda e o número máximo de expansões. O custo de cada expansão é obtido multiplicando-se a capacidade máxima da aresta pelo custo por unidade de demanda.

aresta	origem	destino	situação	capacidade máxima	custo por MW	expansões
201	200	217	projetada	720.00	17.55	3
202	217	228	projetada	720.00	39.04	3
203	228	234	projetada	810.00	30.45	2
204	228	232	projetada	720.00	18.09	3
205	232	234	projetada	720.00	20.60	3
206	200	228	projetada	720.00	29.55	3
207	211	220	projetada	720.00	25.79	3
208	250	251	projetada	780.00	56.71	3
209	220	242	projetada	720.00	17.39	3
210	226	242	projetada	720.00	19.43	3
211	220	273	projetada	720.00	39.41	3
212	231	273	projetada	720.00	40.14	3
213	242	273	projetada	720.00	41.62	3
214	240	257	projetada	720.00	29.60	3
215	226	231	projetada	720.00	64.38	3
216	226	257	projetada	720.00	82.88	3
217	231	240	projetada	720.00	33.30	3
218	247	249	projetada	360.00	20.08	2
219	215	217	projetada	720.00	30.81	3
220	229	243	projetada	360.00	26.81	2
221	229	236	projetada	360.00	13.09	2
222	236	243	projetada	360.00	15.09	2
223	228	233	projetada	720.00	14.51	1
224	235	252	projetada	360.00	7.75	3
225	218	221	projetada	360.00	15.71	1
226	262	221	projetada	360.00	23.48	1
227	237	238	projetada	360.00	30.88	1
228	238	239	projetada	360.00	26.53	1
229	243	252	projetada	360.00	14.09	2
230	207	209	projetada	720.00	17.55	1
231	209	211	projetada	720.00	29.55	3
232	209	226	projetada	720.00	62.33	3
233	207	212	projetada	720.00	25.07	3
234	27	200	projetada	720.00	67.63	2
235	54	60	projetada	720.00	25.76	3
236	200	60	projetada	720.00	39.54	3
237	201	203	projetada	660.00	56.79	3
238	240	245	projetada	720.00	44.78	3
239	231	250	projetada	720.00	111.94	3
240	220	250	projetada	720.00	53.73	3
241	211	250	projetada	720.00	57.31	3
242	248	250	projetada	720.00	22.39	3
243	240	244	projetada	720.00	89.55	3
244	244	245	projetada	720.00	89.55	3
245	245	253	projetada	720.00	17.91	3
246	211	212	projetada	720.00	38.51	3
247	215	226	projetada	720.00	39.40	3
248	235	237	projetada	360.00	12.91	3
249	235	238	projetada	360.00	24.07	3
250	246	249	projetada	360.00	37.57	3

Tabela 7.7: Dados das arestas (250 a 283) da rede elétrica usada nos problemas testados. Para cada aresta é mostrado o índice, os índices dos seus nós de destino e origem, a situação (instalada ou projetada), a capacidade máxima em unidades de potência, o custo por unidade de demanda e o número máximo de expansões. O custo de cada expansão é obtido multiplicando-se a capacidade máxima da aresta pelo custo por unidade de demanda.

aresta	origem	destino	situação	capacidade máxima	custo por MW	expansões
251	248	251	projetada	720.00	92.24	3
252	240	253	projetada	720.00	32.38	3
253	231	257	projetada	720.00	22.20	3
254	211	248	projetada	720.00	32.95	3
255	216	255	projetada	660.00	28.58	3
256	234	257	projetada	720.00	12.64	3
257	226	259	projetada	720.00	26.87	3
258	212	226	projetada	720.00	32.24	3
259	222	226	projetada	720.00	44.78	3
260	217	222	projetada	720.00	26.87	3
261	209	212	projetada	720.00	37.61	3
262	12	27	projetada	990.00	11.71	1
263	37	205	projetada	90.00	3.58	2
264	207	206	projetada	450.00	6.07	1
265	249	250	projetada	336.00	4.98	3
266	221	222	projetada	336.00	4.98	3
267	217	218	projetada	336.00	4.98	3
268	234	237	projetada	336.00	4.98	3
269	211	246	projetada	336.00	4.98	3
270	226	227	projetada	240.00	4.60	2
271	231	243	projetada	240.00	4.60	3
272	272	273	projetada	240.00	4.60	3
273	212	213	projetada	336.00	4.98	3
274	257	252	projetada	240.00	4.60	3
275	233	223	projetada	540.00	6.39	3
276	201	43	projetada	450.00	6.07	3
277	53	54	projetada	990.00	11.71	3
278	210	41	projetada	90.00	3.58	3
279	210	60	projetada	450.00	6.07	3
280	245	239	projetada	336.00	4.98	3
281	248	247	projetada	336.00	4.98	3
282	216	215	projetada	450.00	6.07	3
283	255	259	projetada	450.00	6.07	3

Tabela 7.8: Dados dos nós (12 a 229) da rede elétrica usada nos problemas testados. Para cada nó são mostrados o índice, o nome, o consumo e a geração de potência em MW.

índice	nome	geração	consumo
12	itaipu7	0	0
21	ivaipora7	1528.0	0
27	itaipu 5	3390.0	958.0
33	rosana 4	272.0	0
37	assis 2	0	62.0
38	xavantes 2	352.0	95.0
39	jurumirim 2	185.0	203.0
40	l.n.garcez 2	60.0	178.0
41	botucatu 2	120.0	479.0
53	itabera 7	0	0
200	sao paulo 5	936.0	9580.0
201	p.primaver 4	1531.0	499.0
202	taquarucu 4	20.0	45.0
203	jupia 4	4982.0	718.0
204	capivara 4	544.0	147.0
205	assis 4	0	134.0
206	a.vermelha 4	1174.0	592.0
207	a.vermelha 5	0	0
208	bauru 4	234.0	391.0
209	s.simao 5	1429.0	0
210	botucatu 4	0	0
211	itumbiara 5	0	0
212	marimbondo 5	1134.0	0
213	marimbondo 3	0	0
214	p.colombia 3	279.0	216.0
215	araraquara 5	0	0
216	araraquara 4	0	828.0
217	campinas 5	0	0
218	campinas 3	0	178.0
219	v.grande 3	323.0	0
220	emborcacao 5	1937.0	372.0
221	pocos 3	0	542.0
222	pocos 5	0	0
223	taubate 4	0	476.0
224	estreito 3	939.0	0
225	mascarenha 3	407.0	401.0
226	jaguara 5	0	0
227	jaguara 3	436.0	398.0
228	c.paulista 5	0	8.0
229	pimenta 3	0	151.0

Tabela 7.9: Dados dos nós (230 a 259) da rede elétrica usada nos problemas testados. Para cada nó são mostrados o índice, o nome, o consumo e a geração de potência em MW.

índice	nome	geração	consumo
230	itutinga 3	0	0
231	b.h. 5	0	470.0
232	angra 5	2182.0	67.0
233	taubate 5	0	0
234	adriano 5	1257.0	4216.0
235	barb-jfora 3	278.0	998.0
236	gafanhoto 3	0	151.0
237	adriano 3	231.0	0
238	campos 3	409.0	287.0
239	vitoria 3	143.0	544.0
240	mesquita 5	88.0	829.0
241	furnas 3	1116.0	0
242	n.ponte 5	582.0	0
243	b.h. 3	97.0	0
246	itumbiara 3	549.0	570.0
247	bandeirant 3	0	807.0
249	brasilvia 3	118.0	994.0
250	brasilvia 5	0	0
251	a.tocantis 5	1008.0	0
252	t. sul 3	0	480.0
254	conchal 4	0	0
255	rib.preto 4	188.0	848.0
256	sta.barb. 4	0	1341.0
257	t.sul 5	0	0
260	sao paulo 4	0	0
261	sao paulo 7	0	0
262	sao paulo 3	0	0
263	sao paulo 2	0	0
267	t.marias 3	954.0	1159.0
272	t.oeste 3	0	0
273	t.oeste 5	0	0
43	p.primaver 5	0	0
54	itabera 5	0	0
60	botucatu 5	0	0
244	jequitinho 5	0	0
245	vitoria 5	0	0
248	bandeirant 5	0	0
253	doce 5	0	0
259	r.preto 5	0	0

Tabela 7.10: Agentes que fazem parte de cada problema teste.

[illegible]

Para melhor testar as aplicações desenvolvidas, foram gerados 13 problemas testes adicionais para o PACST: quatro problemas com dez agentes, quatro problemas com 11 agentes, quatro problemas com 12 agentes e um problemas com 16 agentes. Todos os problemas testes gerados usam a mesma rede de 79 nós e 283 arestas. Os agentes que operam a rede em cada problema teste gerado são subconjuntos obtidos aleatoriamente a partir dos 19 agentes que operam a rede no caso original. A Tabela 7.10 mostra para cada problema teste, os agentes que são implantados na rede.

7.4.3

Os experimentos

A Tabela 7.11 mostra os resultados computacionais das execuções sequenciais e paralelas da abordagem de geração de cortes desenvolvida para o PACST. São mostrados para os 14 problemas testados: o tempo de SEP-E, o tempo de SEP-H e, em seguida, o tempo e a aceleração de SEP-PAR para dois, três, cinco e nove processadores. Todos os tempos mostrados são tempos corridos e medidos em segundos.

Para SEP-H, foram feitas cinco execuções independentes para cada um dos quatro casos, onde o procedimento de separação é iniciado por: (i) o conjunto vazio de agentes, (ii) o conjunto completo de agentes, (iii) um subconjunto de agentes gerado aleatoriamente e (iv) um subconjunto de agentes associado a um dos cortes gerados na iteração anterior. Portanto, os tempos mostrados na Tabela 7.11 para SEP-H são as médias de 20 execuções realizadas para cada problema. Para SEP-E, apenas uma execução foi feita para cada problema teste, visto que esse programa não possui nenhuma etapa probabilística ou parâmetros que possam ser ajustados.

Os tempos mostrados para SEP-PAR são as médias dos tempos de cinco execuções independentes para cada número de processadores considerado. Cada processo que executa a separação heurística é iniciado de uma solução diferente (um subconjunto diferente de agentes), como mostrado a seguir:

Tabela 7.11: Resultados computacionais das execuções sequenciais e paralelas das abordagens de geração de cortes desenvolvidas para o PACST. São mostrados: a média dos tempos de execução de SEP-E, a média dos tempos de execução de SEP-H e a média dos tempos de execução e a aceleração de SEP-PAR para dois, três, cinco e nove processadores. Os tempos mostrados são tempos corridos e são medidos em segundos.

problema	SEP-E tempo (s)	SEP-H tempo (s)	número de processadores para SEP-PAR							
			2		3		5		9	
			tempo (s)	aceleração	tempo (s)	aceleração	tempo (s)	aceleração	tempo (s)	aceleração
10-1	1589.50	877.55	621.05	1.41	555.09	1.58	418.85	2.10	472.38	1.86
10-2	2250.06	2750.10	1346.33	2.04	1239.55	2.22	1521.28	1.81	1505.68	1.83
10-3	5522.47	1510.10	789.62	1.91	785.52	1.92	2546.87	0.59	1482.61	1.02
10-4	124.02	233.00	77.39	3.01	82.36	2.83	69.44	3.36	60.02	3.88
11-1	1205.23	1703.45	876.57	1.94	1008.10	1.69	1007.20	1.69	784.61	2.17
11-2	691.02	779.00	338.29	2.30	251.47	3.10	346.80	2.25	291.23	2.67
11-3	282.15	303.90	143.27	2.12	169.40	1.79	157.34	1.93	133.30	2.28
11-4	666.80	359.00	260.97	1.38	213.42	1.68	201.76	1.78	236.31	1.52
12-1	399.21	356.85	229.97	1.55	178.80	2.00	134.21	2.66	102.75	3.47
12-2	16051.89	8276.80	4057.62	2.04	964.15	8.58	3643.11	2.27	4985.21	1.66
12-3	17961.64	13864.05	8192.34	1.69	8947.16	1.55	9566.44	1.45	8918.21	1.55
12-4	1663.23	1267.35	740.47	1.71	822.96	1.54	661.98	1.91	530.43	2.39
16-1	64165.43	34078.79	8916.92	3.82	15252.66	2.23	14577.56	2.34	14194.23	2.40
19-1	127184.57	63021.28	27613.48	2.28	23956.71	2.63	21427.41	2.94	7918.48	3.52

- Dois processadores: o procedimento de separação heurística é iniciado por um conjunto vazio de agentes;
- Três processadores: além dos processos usados nas execuções com dois processadores, é acrescentado um processo onde o procedimento de separação heurística é iniciado por um subconjunto de agentes gerado aleatoriamente a cada iteração do problema mestre;
- Cinco processadores: além dos anteriores, são acrescentados um procedimento de separação heurística iniciado pelo subconjunto de agentes associado ao último corte gerado e outro procedimento de separação heurística iniciado pelo conjunto completo de agentes; e
- Nove processadores: além dos anteriores, são acrescentados quatro processos onde os procedimentos de separação heurística são iniciados por subconjuntos de agentes gerados aleatoriamente a cada iteração do problema mestre.

Cada valor de aceleração mostrado é calculado dividindo-se a média dos tempos de execução de SEP-H pela média dos tempos das execução de SEP-PAR para um dado número de processadores. Essas médias são os tempos mostrados na Tabela 7.11 para SEP-H e SEP-PAR, e foram calculadas como explicado nessa subseção.

A Tabela 7.12 mostra as reduções percentuais dos tempo de execução para as abordagens seqüenciais e paralelas propostas para geração de cortes para o PACST. A redução percentual do tempo de execução de SEP-H em relação ao tempo de execução de SEP-E é indicada na Tabela por “redução H-E”. A “redução PAR-E” indica a redução percentual do tempo de execução de SEP-PAR (com dois processadores) em relação ao tempo de execução de SEP-E. A redução percentual do tempo de execução de SEP-PAR (com dois processadores) em relação ao tempo de execução de SEP-H é indicada na Tabela por “redução PAR-H”. A redução percentual do tempo de execução de um algoritmo $X \in \{\text{SEP-H}, \text{SEP-PAR}\}$ em relação ao tempo de execução de um algoritmo $Y \in \{\text{SEP-E}, \text{SEP-H}\}$ para $X \neq Y$, é calculada através de uma regra de três entre o tempo total de X (100%) e a diferença entre os tempos de execução de X e Y (redução percentual X-Y).

Observa-se que o uso da separação heurística em SEP-H permitiu uma redução média de 13.91% no tempo de execução, quando comparado ao tempo de SEP-E. Porém, observa-se que essa redução é maior para os problemas que precisam de muito tempo computacional. Por exemplo, os tempos médios de execução dos problemas 12-2, 16-1 e 19-1 foram reduzidos de 16051.89, 64165.43 e 127184.57 segundos para 8276.80, 34078.79 e 63021.28

Tabela 7.12: Reduções médias percentuais dos tempos de execução para as diferentes abordagens do programa de geração de cortes para o PACST. A “redução H-E” indica a redução percentual do tempo de execução de SEP-H em relação ao tempo de execução de SEP-E. A “redução PAR-E” indica a redução percentual do tempo de execução de SEP-PAR (com 2 processadores) em relação ao tempo de execução SEP-E. A “redução PAR-H” indica a redução percentual do tempo de execução de SEP-PAR (com 2 processadores) em relação ao tempo de execução de SEP-H.

problema	redução H-E (%)	redução PAR-E (%)	redução PAR-H (%)
10-1	44.79	60.93	29.23
10-2	-22.22	40.16	51.04
10-3	72.66	85.70	47.71
10-4	-87.87	37.60	66.79
11-1	-41.34	27.27	48.54
11-2	-12.73	51.04	56.57
11-3	-7.71	49.22	52.86
11-4	46.16	60.86	27.31
12-1	10.61	42.39	35.56
12-2	48.44	74.72	50.98
12-3	22.81	54.39	40.91
12-4	23.80	55.48	41.57
16-1	46.89	86.10	73.83
19-1	50.45	78.29	56.18
média	13.91	57.44	48.51

segundos, respectivamente, apresentando reduções de 48.44 %, 46.89 % e 50.45 %, respectivamente. Para os problemas testes 10-2, 10-4, 11-1, 11-2 e 11-3 observa-se que o tempo de execução de SEP-H é superior ao tempo de execução de SEP-E. Isso pode ser explicado devido ao fato de que como esses problemas testes são pequenos, a separação exata é feita de forma eficiente.

As reduções obtidas usando-se SEP-PAR com dois processadores foram em média de 57.44% e de 48.51% em relação aos tempos de execução de SEP-E e SEP-H, respectivamente. Isso equivale a dizer que uma aceleração aproximadamente igual a 2 foi obtida para SEP-PAR em relação a SEP-H, como observado na Tabela 7.11. Reduções elevadas do tipo PAR-E e PAR-H são observadas para os problemas grandes. Por exemplo, para 12-2, 16-1 e 19-1 as reduções PAR-E foram de 74.72 %, 86.10 % e 78.29 %, respectivamente, e as reduções PAR-H foram de 50.98 % , 73.83 % e 56.18 %, respectivamente. Ou seja, para o problema 19-1, o tempo computacional foi reduzido de 127184.57 segundos para 27613.48 segundos, introduzindo-se a separação heurística em SEP-E e em seguida, paralelizando-se essa aplicação (usando-se apenas dois processadores).

Verifica-se na Tabela 7.11 que a aceleração de SEP-PAR não cresce proporcionalmente ao número de processadores usados. Em uma execução paralela, apenas um processador estará fazendo a separação exata. Os de-

Tabela 7.13: Numero médio de separações e de separações exatas realizadas para cada problema teste, para o programa seqüencial e para o programa paralelo com dois, três, cinco e nove processadores.

prob.	número de processadores									
	1		2		3		5		9	
	total	exat.	total	exat.	total	exat.	total	exat.	total	exat.
10-1	41.9	5.8	33.2	3.0	29.4	3.2	26.8	3.0	22.8	3.4
10-2	43.3	8.8	37.0	4.0	30.6	3.4	29.8	3.4	30.0	4.2
10-3	48.4	4.4	38.4	2.6	39.2	2.8	36.0	4.0	31.4	3.8
10-4	56.4	6.3	38.8	6.4	40.8	5.4	35.0	4.6	33.8	3.4
11-1	54.1	8.9	42.2	4.6	44.2	4.6	37.4	4.4	33.6	3.4
11-2	68.7	7.0	49.2	4.0	48.0	3.4	44.2	4.2	45.8	3.8
11-3	59.0	7.0	44.4	7.2	44.8	7.2	39.4	6.2	37.4	5.8
11-4	53.1	6.6	43.6	3.6	42.6	3.4	42.2	2.8	36.4	4.6
12-1	63.2	6.5	47.6	5.8	44.2	4.0	41.6	2.6	35.4	1.8
12-2	69.3	11.5	55.0	3.6	59.8	3.4	48.4	4.2	45.4	5.2
12-3	72.8	11.2	54.2	4.2	55.0	5.4	50.0	5.0	45.0	5.0
12-4	55.5	5.8	40.4	3.2	44.6	3.4	31.6	3.4	31.8	2.6
16-1	88.3	13.7	73.4	4.8	83.8	5.2	71.6	4.2	80.4	5.0
19-1	154.6	24.9	90.0	4.8	84.6	3.8	76.0	3.8	84.2	3.2

mais processadores estarão fazendo a separação heurística e não apresentam nenhuma garantia de encontrar uma restrição violada. Quando os processos de separação heurística falham para encontrar uma restrição violada, o procedimento de separação exata executará o *branch-and-bound* por completo, até que o corte mais violado seja encontrado ou até que seja verificado que não existem cortes violados. O tempo de execução de uma iteração do programa mestre onde a separação exata é feita é de 723.02 segundos em média, enquanto que o tempo de execução de uma iteração onde a separação heurística é feita é de 20.73 segundos em média. Dessa forma, quando é necessário executar o procedimento de *branch-and-bound* por completo, a iteração precisará de um tempo computacional em média 35 vezes maior do que o tempo de uma iteração onde a separação heurística é feita. Além disso, o tempo computacional gasto pelas separações exatas feitas em uma execução do programa paralelo corresponde em média a 74.07% do tempo total do programa e não é paralelizado. As médias dos tempos de execução das separações exatas e separações heurísticas foram medidas considerando-se todas as execuções realizadas para SEP-H e SEP-PAR. Para calcular as médias dos tempos de execução das separações em SEP-PAR, o tempo necessário para executar cada iteração do problema mestre foi medido. Caso uma restrição violada tenha sido encontrada por um processo de separação heurística em uma dada iteração, o tempo medido é usado no cálculo do tempo médio para fazer a separação heurística. Caso contrário, o tempo medido é usado no cálculo do tempo médio das separações exatas.

A ordem em que as restrições são geradas pode alterar drasticamen-

te o comportamento do programa, visto que a cada iteração do problema mestre, subproblemas de separação diferentes são gerados. Portanto, subproblemas de separação mais difíceis podem ser gerados nas execuções com um número maior de processadores. Por esse motivo, é possível observar na Tabela 7.11 algumas acelerações decrescentes à medida que o número de processadores é aumentado. A Tabela 7.13 mostra, para cada problema teste, o número médio de separações e o número médio de separações exatas realizadas para o programa seqüencial e para o programa paralelo com dois, três, cinco e nove processadores. Observa-se uma redução considerável no número total de separações realizadas pelo programa paralelo, quando comparado ao número total de separações realizadas pelo programa seqüencial. Isso ocorre porque no programa paralelo, a cada iteração do problema mestre, várias restrições são freqüentemente geradas, sendo obtidas por processos diferentes. Além disso, na abordagem seqüencial, muitas vezes o procedimento de separação heurística é interrompido precocemente após esgotar o seu número máximo de iterações **MaxTentativas**, fazendo com que a restrição mais violada seja obtida pelo procedimento de separação exata. Deve-se notar porém, que um aumento considerável em **MaxTentativas** pode levar à obtenção de tempos computacionais mais elevados visto que, ainda assim, em muitas iteração do programa mestre, o procedimento de separação heurística não será capaz de obter uma restrição violada. Portanto, o parâmetro **MaxTentativas** deve ser cuidadosamente ajustado em **SEP-H**. Em **SEP-PAR**, o uso do parâmetro **MaxTentativas** não é necessário, pois processos de separação exata e heurística executam em paralelo até que um corte violado seja encontrado ou que seja verificado que a solução corrente é ótima para o problema mestre. Observa-se que, para o programa paralelo, o número de separações exatas não decresce quando o número de processadores aumenta. Isso justifica o fato de que acelerações lineares não são observadas para **SEP-PAR**.

7.4.4

Distribuição da probabilidade do tempo para valor alvo

Para analisar a distribuição da probabilidade do tempo para valor alvo do procedimento de separação heurística apresentado nesse capítulo, foram feitas pequenas alterações no algoritmo da Figura 7.1. Primeiramente, esse procedimento foi separado do programa de alocação de custo, para ser executado de forma isolada. O programa de separação heurística recebe como entrada além dos dados dos K agentes e da rede, uma alocação de

custos $\bar{\theta}_k, k = 1, \dots, K$ e um valor alvo. O valor alvo é um limite superior para o valor da função objetivo do problema de separação SEP, para o vetor característico λ associado aos agentes $k = 1, \dots, K$ fornecidos como entrada. A alocação de custos é obtida executando-se SEP-H e são valores calculados em alguma iteração do problema mestre, onde uma restrição violada foi encontrada pelo procedimento de separação heurística. O programa de separação heurística é executado até encontrar uma solução com custo inferior ao valor alvo. Para isso, a linha 2 do algoritmo da Figura 7.1 foi alterada para fazer a variável *limite* receber o valor alvo fornecido como entrada. Dessa forma, dada uma alocação de custos $\bar{\theta}$, o programa será executado até encontrar alguma restrição definida pelas variáveis viáveis λ e x do problema de separação que satisfaça

$$\sum_{(i,j) \in E} d_{ij}x_{ij} - \sum_{k=1}^K \bar{\theta}_k \lambda_k < \textit{limite},$$

isto é, uma solução em que o valor da função objetivo no problema de separação seja inferior ao valor alvo.

As Figuras 7.3 e 7.4 mostram os gráficos Q-Q e os gráficos com as distribuições teóricas e empíricas sobrepostas para o procedimento de separação heurística, respectivamente. A metodologia apresentada no Capítulo 3 é usada para plotar esses gráficos. Foram considerados os problemas testes 10-1, 11-1, 12-1 e 19-1. Para cada problema teste, três valores alvos foram analisados. Cada gráfico apresentado nessas figuras é plotado a partir dos pontos gerados por 200 execuções independentes do procedimento de separação heurística: 66 execuções iniciadas a partir do conjunto vazio de agentes, 67 execuções iniciadas a partir do conjunto completo de agentes e 67 execuções iniciadas a partir de um conjunto de agentes gerado aleatoriamente. Além disso, em cada execução do programa de separação heurística para um par problema teste/valor alvo é usada uma seqüência de números aleatórios diferente para escolher o próximo movimento a ser realizado durante a busca local no espaço de subconjuntos de agentes.

A Figura 7.3 é formada por 12 gráficos Q-Q, um para cada par problema teste/valor alvo para o procedimento de separação heurística. Da mesma forma, a Figura 7.4 é formada por 12 gráficos com as distribuições empíricas e teóricas sobrepostas, um para cada par problema teste/valor alvo. Os gráficos são dispostos em quatro linhas em cada figura. Cada linha corresponde a um problema teste estudado. Em cada linha das figuras, a dificuldade de encontrar uma solução para um dado valor alvo aumenta

Tabela 7.14: Problemas testes usados para estudar a distribuição empírica de probabilidade da variável aleatória *tempo para valor alvo* do procedimento de separação heurística. São mostrados, para os quatro problemas testados, o custo da restrição mais violada RMV no problema de separação (calculado pelo procedimento de *branch-and-bound* do Cplex), o valor alvo e os parâmetros estimados.

problema	RMV	alvo	parâmetros estimados		
			μ	λ	$ \mu /\lambda$
10-1	-9222.96	0	-0.278	2.217	0.125
		-200	-0.171	2.140	0.079
		-400	-0.154	2.226	0.069
11-1	-7288.26	0	0.252	3.438	0.073
		-200	0.333	3.666	0.090
		-400	0.365	3.560	0.102
12-1	-3346.56	0	0.436	17.850	0.024
		-200	0.260	17.875	0.014
		-400	0.952	21.059	0.045
19-1	-20942.4	0	-4.155	16.748	0.248
		-300	-4.217	17.198	0.245
		-1000	-4.420	18.286	0.241

da esquerda para a direita. Nos gráficos mais à esquerda, a separação heurística é interrompida após encontrar uma solução com custo inferior a zero, conforme é feito no programa de alocação de custos. Na Tabela 7.14 são mostrados os valores alvos considerados, assim como os parâmetros estimados pela metodologia.

Observa-se na Figura 7.3 que o afastamento dos pontos em relação à reta estimada é pequeno. Além disso, como nas distribuições da probabilidade do tempo para valor alvo estudadas nos capítulos anteriores, observa-se que à medida que a dificuldade de obter o valor alvo aumenta, os pontos tornam-se mais ajustados à reta estimada. Portanto, verifica-se que os tempos para valor alvo medidos para o procedimento de separação heurística apresentado nesse capítulo adaptam-se bem a uma distribuição exponencial de dois parâmetros.

A cada iteração do problema mestre, uma alocação de custos $\bar{\theta}_k$, $k = 1, \dots, K$ é gerada. Para algumas dessas alocações de custos, os procedimentos de separação heurística não serão capazes de identificar uma restrição violada, visto que a parcela $\sum_{(i,j) \in E} d_{ij}x_{ij}$ da expressão $\sum_{(i,j) \in E} d_{ij}x_{ij} - \sum_{k=1}^K \bar{\theta}_k \lambda_k < 0$ é um limite superior para o custo mínimo de expansão da rede calculado por uma heurística. Nesses casos, a restrição mais violada deverá ser obtida pelo procedimento de separação exata, caso a alocação de custos não seja viável. Portanto, apesar de valores pequenos de $|\mu|/\lambda$ terem sido observados para a maior parte dos pares problema teste/valor alvo testados, acelerações lineares não são observadas para SEP-PAR, conforme

explicado na subseção anterior.

7.5

Conclusão

Nesse capítulo foram apresentadas abordagens para geração de cortes para o problema de alocação de custos de serviços para sistemas de transmissão. A primeira abordagem consiste em um esquema de geração de cortes em que cada etapa de separação é feita de forma exata por um procedimento de *branch-and-bound*. Em seguida, com o objetivo de acelerar a primeira abordagem proposta, desenvolveu-se uma abordagem onde heurísticas são usadas na etapa de separação, usando-se o procedimento de *branch-and-bound* somente quando o procedimento heurístico não é capaz de encontrar uma restrição violada. Também foi apresentada uma abordagem paralela para geração de cortes, implementada para o paradigma de memória distribuída. Nessa abordagem, um procedimento de *branch-and-bound* será sempre executado por um processo, enquanto que os demais $p - 1$ processos irão executar a separação heurística. Em [4] foi apresentada uma abordagem paralela de geração de cortes para o PACST, implementada para o paradigma de memória compartilhada. Essa abordagem é semelhante à abordagem paralela apresentada nesse capítulo.

Os programas SEP-E, SEP-H e SEP-PAR foram testados e comparados na Seção 7.4. Verificou-se que SEP-H é em média 13.91% mais rápido de que SEP-E e que SEP-PAR (usando dois processadores) é em média 57.44% e 48.51% mais rápido do que SEP-E e SEP-H, respectivamente. Para os problemas que precisam de um tempo computacional elevado, reduções maiores são observadas.

Foi estudada a distribuição do tempo para valor alvo do procedimento de separação heurística apresentado nesse capítulo. Como no método GRASP, observou-se que os tempos para valor alvo medidos para o procedimento de separação heurística adaptam-se bem a uma distribuição exponencial de dois parâmetros. Nos Capítulos 3, 4 e 5 mostrou-se que a distribuição da probabilidade do tempo para valor alvo do algoritmo GRASP puro, assim como de algumas de suas variantes, adapta-se bem a uma distribuição exponencial de dois parâmetros. Nesse capítulo, verificou-se que os tempos para solução de um procedimento de busca local, onde o conceito de lista restrita de candidatos introduzido no contexto da metaheurística GRASP é usado para escolher o próximo movimento realizado, também adaptam-se a uma distribuição exponencial de dois parâmetros. Esse procedimento de

busca local é um exemplo de algoritmo onde o critério de parada por valor alvo é usado na prática, pois é interrompido após obter uma solução com custo inferior a um valor alvo igual a zero.

No próximo capítulo serão apresentadas as conclusões da tese, assim como propostas de trabalhos futuros.

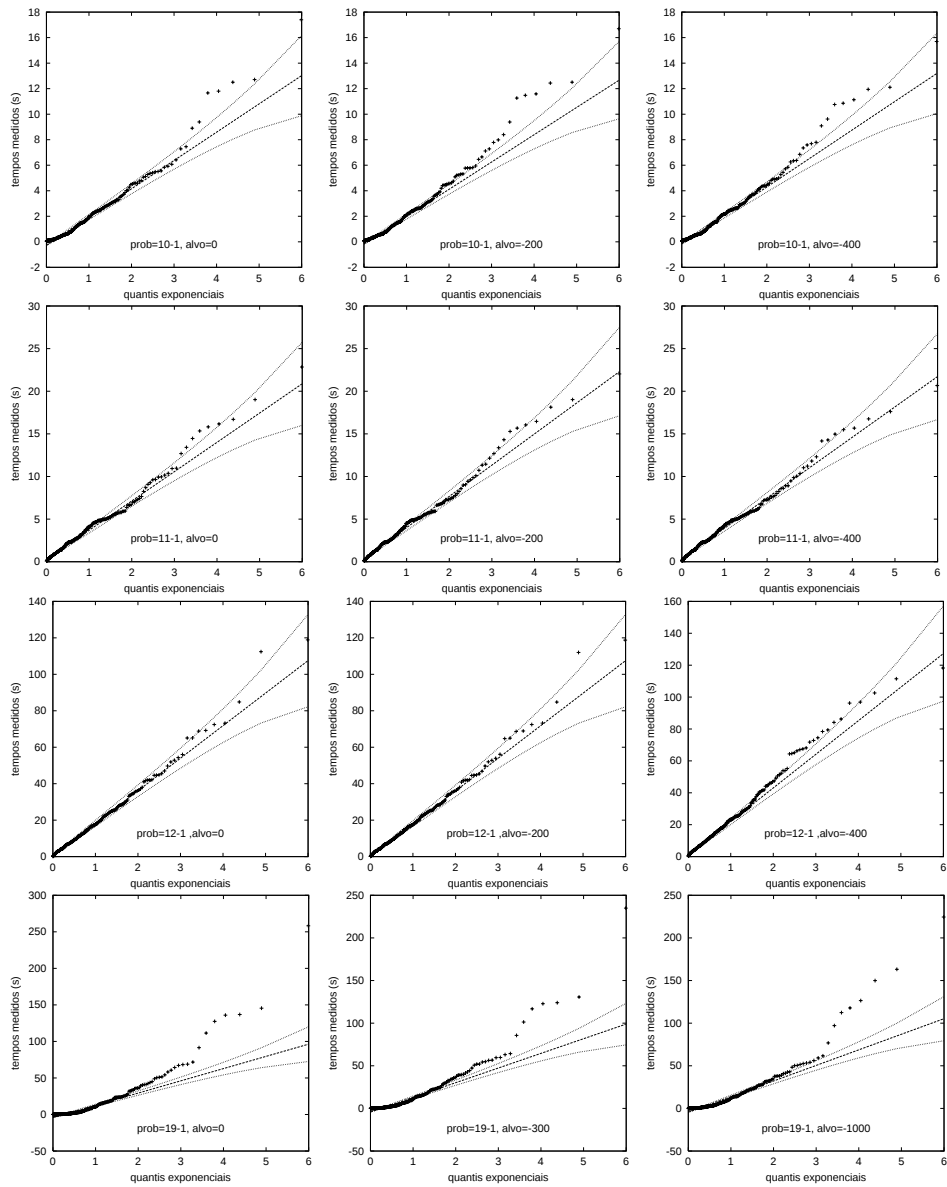


Figura 7.3: Gráficos Q-Q para problemas 10-1, 11-1, 12-1 e 19-1.

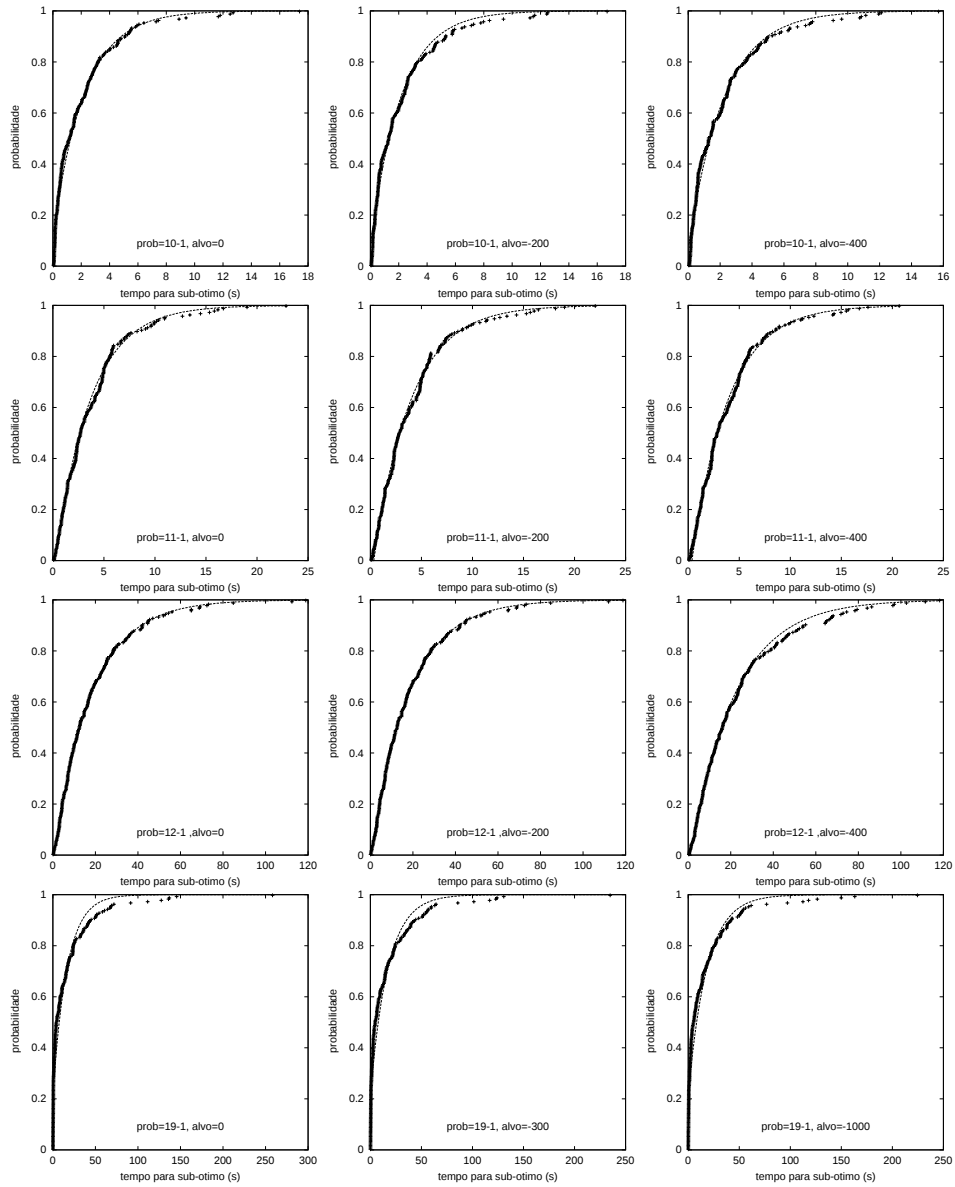


Figura 7.4: Gráficos das distribuições exponenciais para problemas 10-1, 11-1, 12-1 e 19-1.