

Centro de Pesquisas de Energia Elétrica - CEPEL

Relatório de Projeto – DSE – 3994/2024

Cliente: ONS



A pesquisa que constrói o futuro

Título: Eliminação de Cortes de Benders Redundantes da Função de Custo Futuro do modelo NEWAVE

Nº de Páginas: 43

Nº de Anexos: 00

Dados do Cliente:

OPERADOR NACIONAL DO SISTEMA ELÉTRICO – ONS

Rua Júlio do Carmo, 251 – Cidade Nova - Rio de Janeiro – RJ, CEP: 20211-160

Responsável:

Sra. Tatiana Frade Gonçalves Mundstock

Email: tatiana@ons.org.br

Departamento: Departamento de Sistemas Eletroenergéticos - DSE

Nº do PEP: CPCT.ONOE00.790.004

Centro de Lucro: C250000101

Centro de Custo: C205000021

Resumo: Este relatório aborda a eliminação de cortes dominados gerados pela Programação Dinâmica Dual Estocástica (PDDE). O crescimento excessivo de cortes de Benders durante as iterações do modelo aumenta o custo computacional sem contribuir para a solução. Para enfrentar esse desafio, foram investigados dois métodos: o Algoritmo de Shapiro Modificado e a Análise de Redundância Por Pares, além da implementação de estratégias de paralelização para otimizar a eficiência do segundo método.

Observações:

Autores:

Cristiane Barbosa da Cruz Oliveira, Robério da Rocha Barbosa, Tiago Lino Bello, André Luiz Diniz, Danielle de Freitas da Silva e Roberto José Pinto.

Palavras-Chave:

Planejamento da operação de longo/médio prazo; programação dinâmica dual estocástica; corte de Benders; cortes redundantes; eliminação de cortes;

Classificação de acesso:

☒ Público ☐ Interno ☐ Setorial ☐ Confidencial

Aprovação e data de emissão

André Luiz Diniz Souto Lima
Chefe do Departamento de Sistemas Eletroenergéticos

Tel.: (21) 2598-6046

E-mail: diniz@cepel.br

Aprovação e data de emissão

Roberto de Beauclair Seixas
Diretor de Tecnologia

Centro de Pesquisas de Energia Elétrica – CEPEL www.cepel.br

Sede: Av. Horácio Macedo, 354 - Cidade Universitária - CEP 21941-911 - Rio de Janeiro - RJ - Brasil - Tel.: 21 2598-6000

Unidade Adrianópolis: Av. Olinda, 5800 - Adrianópolis - CEP 26053-121 - Nova Iguaçu - RJ - Brasil - Tel.: 21 2666-6200

Endereço Postal: CEPEL Caixa Postal 68007 - CEP 21941-971 - Rio de Janeiro - RJ - Brasil / **Endereço Eletrônico:** cepel@cepel.br

ELIMINAÇÃO DE CORTES DE BENDERS REDUNDANTES DA FUNÇÃO DE CUSTO FUTURO DO MODELO NEWAVE

newave@cepel.br

ÍNDICE

SUMÁRIO EXECUTIVO	4
1 INTRODUÇÃO	5
2 ALGORITMOS DE ELIMINAÇÃO DE CORTES	6
2.1 Eliminação de Cortes Paralelos	6
2.2 Análise por Pares	7
2.3 Shapiro Modificado	9
3 ESTRATÉGIAS DE PARALELIZAÇÃO	12
4 ENTRADA DE DADOS	15
4.1 Dados gerais (dger.dat)	15
4.2 Arquivo com os parâmetros da eliminação de cortes (eliminacao-cortes.dat)	15
5 ARQUIVOS DE SAÍDA	17
5.1 Eco da entrada de dados (pmo.dat)	17
5.2 Relatórios do processo de eliminação de cortes	17
5.2.1 Relatório com os cortes eliminados (eliminacao-cortes-rel.csv)	17
5.2.2 Relatório com os limites considerados para cada variável de estado no processo de eliminação de cortes (eliminacao-cortes-limites-variaveis.csv)	18
5.3 Tempo de execução	19
6 RESULTADOS	20
6.1 Análise de estratégias de computação paralela na execução do algoritmo de Eliminação de Cortes Por Pares	20
6.2 Aplicação concomitante de Seleção e Eliminação de Cortes	24
6.2.1 Comparativo dos resultados do PMO	24
6.2.2 Análise de tempo de execução	26
6.2.3 Análise de cortes eliminados	30
6.3 Aplicação da Eliminação de Cortes em casos sem Seleção de Cortes	32
6.3.1 Comparativo dos resultados do PMO	32
6.3.2 Análise de tempo de execução	34
6.3.3 Análise de cortes eliminados	37
6.4 Testes numéricos observando o parâmetro de janela de iterações	38
7 BIBLIOGRAFIA	43

SUMÁRIO EXECUTIVO

Este relatório aborda a importância da eliminação de cortes redundantes/dominados no contexto da Programação Dinâmica Dual Estocástica (PDDE) aplicada ao modelo NEWAVE.

Motivação: O modelo NEWAVE utiliza a estratégia de Programação Dinâmica Dual Estocástica (PDDE) para determinar políticas ótimas para o planejamento da operação de sistemas de geração de energia elétrica. Contudo, o processo iterativo da PDDE frequentemente resulta em um crescimento excessivo no número de cortes de Benders (ou planos cortantes), muitos dos quais são redundantes, ou seja, podem ser eliminados sem afetar o valor ótimo do problema de otimização. Assim, esses cortes redundantes não contribuem para a solução, mas aumentam o custo computacional.

Metodologia: Dois métodos foram implementados e investigados para identificar e eliminar cortes redundantes durante a execução da PDDE: o algoritmo de Shapiro Modificado e a Análise de Redundância por Pares. Além disso, foram implementadas duas estratégias distintas de paralelização para o algoritmo de Análise de Redundância por Pares.

Resultados: A partir dos resultados obtidos num caso exemplo do Programa Mensal da Operação (PMO), realizado pelo Operador Nacional do Sistema (ONS), foi constatado que:

- (i) a inclusão do algoritmo de Análise Por Pares auxiliou a reduzir em 7% o tempo de execução do NEWAVE, diminuindo em até 10% o tempo das fases *backward* e *forward* da PDDE e eliminando mais de 15% de cortes redundantes;
- (ii) o algoritmo de eliminação de cortes com Shapiro Modificado diminui em até 20% o tempo das fases *backward* e *forward* da PDDE, e eliminou mais de 50% de cortes redundantes, embora eficiente na eliminação de cortes, seu alto custo computacional supera os benefícios obtidos;
- (iii) a seleção de cortes para a PDDE é importante no tempo de execução do NEWAVE, podendo contribuir numa redução no tempo de execução de até 50%;
- (iv) a estratégia de paralelismo por Períodos aplicado ao algoritmo Análise por Pares reduz em até 60% o tempo de execução desse algoritmo quando comparado com a estratégia de paralelismo por Cortes.

1 INTRODUÇÃO

Em problemas de otimização sob incertezas, como aqueles modelados por Programação Dinâmica Dual Estocástica (PDDE) [1], a resolução eficiente é essencial, especialmente em problemas de grande porte. O modelo NEWAVE [2], [3], amplamente utilizado para planejamento da operação energética no Brasil, é um exemplo consolidado de aplicação da PDDE para a resolução de problemas hidro-termo-eólicos. Nesse contexto, o modelo lida com a incerteza nas vazões afluentes aos reservatórios e à disponibilidade de ventos nos parques eólicos utilizando a PDDE para determinar a política ótima de operação de um sistema de geração de energia elétrica, composto por usinas que hidrelétricas, termelétricas, eólicas e solares.

A PDDE é estruturada em um esquema de decomposição temporal, onde as decisões para cada estágio são dependentes da solução de um subproblema dual. Esse subproblema, por sua vez, gera cortes de Benders (ou planos cortantes) que aproximam a função-valor associada a um determinado subproblema, em relação a suas variáveis de estado. No entanto, à medida que o processo iterativo avança, ocorre a geração excessiva de cortes, muitos dos quais tornam-se redundantes ao longo do processo iterativo, ou seja, a região definida por eles não altera o valor ótimo do problema de otimização. Portanto, esses cortes redundantes não contribuem mais para o refinamento da solução e, ao contrário, aumentam o custo computacional da PDDE.

De forma a contornar o número excessivo de cortes, implementou-se no passado, no modelo NEWAVE, a estratégia de seleção de cortes [4], [5], pela qual os cortes da função de custo futuro são adicionados de forma iterativa no processo de resolução de cada subproblema. Embora essa estratégia traga resultados bastantes efetivos em termos de redução no tempo de processamento, o processo de verificação dos cortes que podem ser adicionados leva em consideração todos os cortes já construídos, sejam eles dominados ou não. Portanto, a eliminação de cortes redundantes pode reduzir ainda mais o tempo de processamento e os recursos computacionais necessários sem comprometer a solução ótima. Essa prática é especialmente relevante em problemas complexos, como o NEWAVE, que envolve múltiplos cenários hidrológicos. Além disso, a eliminação de cortes melhora a estabilidade numérica do algoritmo, evitando instabilidades causadas por coeficientes desnecessários.

Diferentes abordagens têm sido propostas na literatura para identificar e eliminar cortes redundantes, com o objetivo de mitigar o crescimento excessivo do conjunto de cortes e melhorar a eficiência computacional¹. A aplicação desses métodos ao modelo NEWAVE se justifica, principalmente, devido à escala dos problemas enfrentados, onde o número de cortes cresce com o horizonte de planejamento adotado, em virtude da necessidade de um maior número de iterações da PDDE para se obter um grau de maturidade maior na função de custo futuro. Em termos práticos, uma eliminação de cortes eficiente pode reduzir o tempo de execução, especialmente em execuções do problema de otimização envolvendo grandes sistemas, como o Sistema Interligado Nacional (SIN).

¹ Ressalta-se que o escopo deste relatório se limita a técnicas de eliminação de cortes dominados. Estratégias de seleção de cortes em si não são tratadas neste documento, e podem ser encontradas em [8], [9].

2 ALGORITMOS DE ELIMINAÇÃO DE CORTES

Até a versão 30.0.1, o modelo NEWAVE possuía apenas o algoritmo de eliminação de cortes Paralelos. A partir da versão 30.1, o modelo passou a contar com a possibilidade de aplicar dois novos algoritmos: Análise de Redundância por Pares e Shapiro Modificado. Assim, o modelo passou a oferecer três algoritmos distintos para eliminação de cortes dominados, descritos a seguir.

2.1 Eliminação de Cortes Paralelos

O algoritmo de eliminação de cortes Paralelos consiste em, para cada corte construído, avaliar se existe outro corte com coeficientes iguais. Caso exista, o corte com menor valor para o termo independente (conhecido como RHS – “right hand side”) é definido como eliminado, uma vez que como são hiperplanos com coeficientes angulares iguais (isto é, paralelos), o hiperplano com o menor valor de RHS/intercepto é o dominado. A Figura 2.1 destaca um exemplo do caso no $\mathbb{R}^2$.

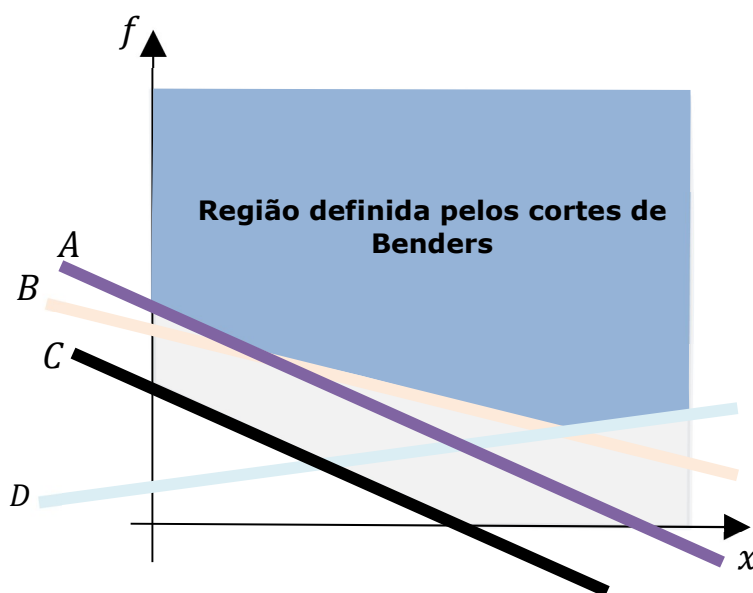


Figura 2.1 – Algoritmo Paralelos – Exemplo intuitivo.

O exemplo ilustrado na Figura 2.1 destaca quatro retas que representam a função de custo futuro (FCF) para uma variável x . Claramente, as retas A e C são paralelas. O algoritmo “Paralelos” tenta identificar a igualdade de coeficientes dos cortes criados com os cortes existentes. Nesse caso, supondo que a reta A foi o corte criado, o algoritmo identifica se as demais retas B, C e D possuem os mesmos valores de coeficientes que o corte A. A partir da existência de coeficientes iguais (retas A e C), o algoritmo Paralelos identifica o menor valor de RHS/intercepto das retas e o retira da análise, uma vez que o menor valor indicará que a reta não é necessária para a curva que define a FCF. No caso apresentado, a reta C possui o menor valor, e seria retirada.

Assim, este algoritmo pode ser definido como se segue. Considere o conjunto I o conjunto de cortes de Benders criados pelo processo de otimização:

Para cada novo corte i criado, faça:

Compare o corte i criado com os cortes do conjunto I , identificando a igualdade dos coeficientes

Se existir cortes com coeficientes iguais ao corte i :
Descarte o corte com menor RHS do conjunto I

A implementação deste algoritmo é simples e veloz, uma vez que não envolve nenhum processo de otimização para definir a redundância dos cortes e, sim, apenas um comparativo de valores.

2.2 Análise por Pares

Este procedimento é uma generalização do algoritmo exposto em [6], utilizado no modelo DESSEM, para o contexto do modelo NEWAVE, onde as variáveis de estado incluem também as energias (ou vazões) afluentes passadas. Em "Análise Por Pares", o procedimento envolve a comparação entre pares de cortes para verificar se um deles é "dominante" em relação ao outro em um determinado domínio considerado, ou seja, se ele torna o outro redundante. A Figura 2.2 destaca um exemplo intuitivo deste procedimento, sendo possível observar que a reta A é dominada pela reta B no domínio da variável x .

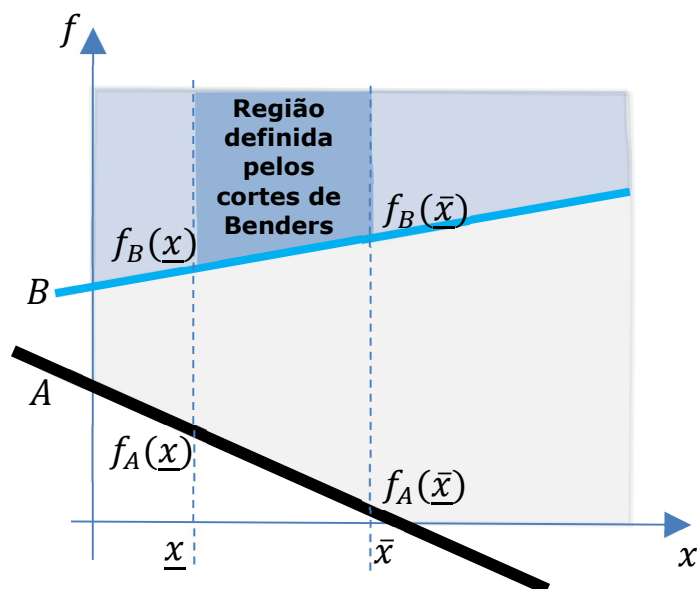


Figura 2.2 – Análise Por Pares – Exemplo intuitivo.

O algoritmo de Análise por Pares utiliza uma condição necessária e suficiente para determinar se um corte é redundante em relação a outro em um domínio específico. No contexto do NEWAVE, considere as variáveis α como custos futuros e $x \in \Gamma$ como as variáveis de estado até a iteração em que a análise de redundância é realizada. Vale ressaltar que as variáveis de estado estão em um conjunto Γ compacto, isto é, o domínio das variáveis (valores que a variável pode assumir) possuem valores mínimos e máximos. Esta análise considera cortes pertencentes a um conjunto I . Cada corte deste conjunto possui um valor escalar *RHS* e coeficientes $\{\beta_1, \beta_2, \dots, \beta_n\}$, sendo n a quantidade de variáveis de estado x . Para verificar se um corte i é redundante em relação a um corte j , i diferente de j , é observado o pior caso possível da seguinte condição,

$$\text{corte}_j \geq \text{corte}_i.$$

Manipulando-a algebricamente, sabendo da linearidade dos cortes, pode-se chegar numa condição de forma a concluir que o corte_i seja considerado redundante em relação ao corte_j ,

$$(RHS^j - RHS^i) + \sum_{k \in \{\beta_k^j - \beta_k^i > 0\}} (\beta_k^j - \beta_k^i) \underline{x} + \sum_{k \in \{\beta_k^j - \beta_k^i < 0\}} (\beta_k^j - \beta_k^i) \bar{x} \geq 0,$$

com $\underline{x}$ e $\bar{x}$ sendo os limites inferiores e superiores da variável x , respectivamente. Assim, uma vez que esta expressão forneça valores positivos, o $corte_i$ é um corte redundante e, assim, pode ser eliminado.

Apesar de existir tal condição, ela é apenas suficiente e não necessária, pois é incapaz de eliminar todos os cortes redundantes uma vez que a comparação é feita em pares, e não com o conjunto de cortes como um todo. Assim, embora seja uma abordagem rápida e eficiente, ela não pode ser generalizada para avaliar a redundância de um corte em relação a múltiplos cortes simultaneamente. Um exemplo disso é quando se observa a situação ilustrada na Figura 2.3: aplicando a condição nas combinações possíveis da reta, nenhuma será indicada para eliminação; entretanto, é visível que a reta B é redundante e não auxilia na definição da curva da FCF. Apesar desta limitação, o processo iterativo é simples contando apenas com operações básicas para definição de redundância.

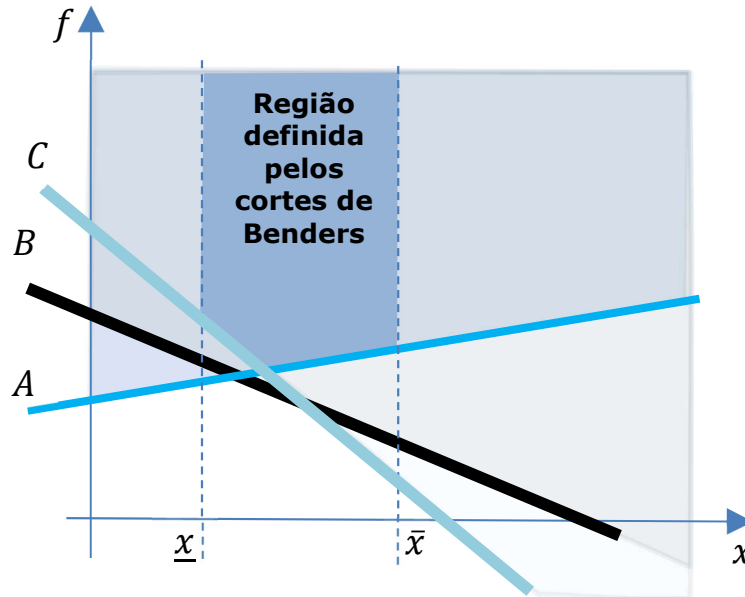


Figura 2.3 – Análise Por Pares – Exemplo de suficiência, mas não necessidade, da condição, quando se considera a comparação simultânea com todos os demais cortes.

Assim, assumindo uma tolerância para o critério de eliminação de cortes, o seguinte procedimento é realizado:

Para cada corte i , faça:

Para cada corte j , i diferente de j , faça:

Compute a seguinte expressão:

$$(RHS^j - RHS^i) + \sum_{k \in \{\beta_k^j - \beta_k^i > 0\}} (\beta_k^j - \beta_k^i) \underline{x} + \sum_{k \in \{\beta_k^j - \beta_k^i < 0\}} (\beta_k^j - \beta_k^i) \bar{x}$$

Caso o valor da expressão for maior que a tolerância adotada, então:

Descarte o corte i do conjunto I

2.3 Shapiro Modificado

O procedimento de Shapiro Modificado resulta de alterações realizadas no procedimento originalmente proposto por [7]. Na proposta original, a identificação de cortes redundantes é feita por meio da verificação da viabilidade de sistemas lineares específicos. Um corte é considerado redundante se, ao avaliar a solução ótima do problema de eliminação, ele nunca for ativado ou se sua contribuição for suplantada por outros cortes. Em outras palavras, o corte é dominado por outros ou deixa de definir a solução ótima em qualquer iteração.

No procedimento descrito por [7], a verificação de viabilidade consiste nos seguintes passos: o sinal da inequação de um corte específico é invertido, enquanto os sinais dos demais cortes são mantidos. Caso o sistema resultante não possua solução, o corte cuja inequação foi invertida é considerado redundante. Para um melhor entendimento, será ilustrado um exemplo no $\mathbb{R}^2$. Suponha a variável de decisão $x \in \Gamma$, sendo Γ um conjunto compacto, e um conjunto de inequações definidas como

$$\alpha \geq RHS^c + (\beta^c)^T x,$$

com $c \in I = \{1,2,3,4\}$, sendo I um conjunto de índices, $RHS \in \mathbb{R}$ e $\beta^c = [\beta_1, \beta_2, \dots, \beta_n]^T \in \mathbb{R}^n$ o intercepto e o coeficiente angular do hiperplano c , respectivamente, e $\alpha \in \mathbb{R}$ uma variável. Suponha que Ω seja a região definida pelos cortes de Benders. A Figura 2.4 destaca a região criada neste exemplo, com destaque a duas inequações que serão exploradas a seguir.

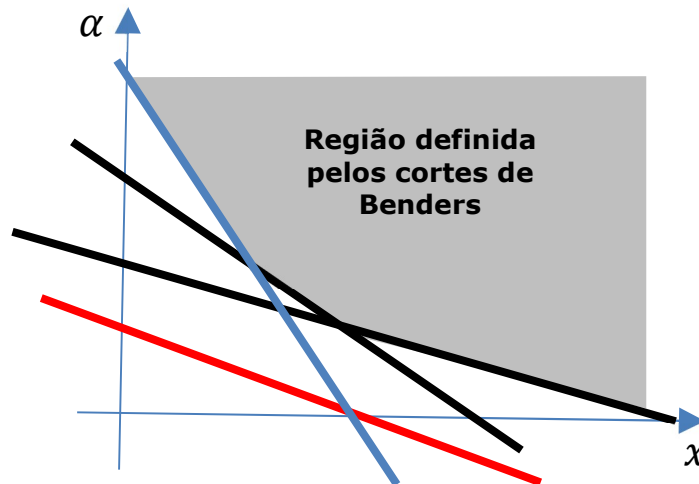


Figura 2.4 - Procedimento de Shapiro - Região definida pelos cortes de Benders no $\mathbb{R}^2$.

Assim, a identificação de um corte j dominado é realizada ao verificar se o sistema de inequações invertendo o sinal do corte j torna o sistema sem solução, isto é,

$$\begin{cases} \alpha \geq RHS^c + (\beta^c)^T x, \forall c \neq j \\ \alpha \leq RHS^j + (\beta^j)^T x \end{cases}.$$

Isso significa que serão gerados subproblemas que, caso não seja possível criar uma região (poliedro, no caso geral), o procedimento de Shapiro destaca que o *corte invertido* é um *corte redundante* aos demais. No exemplo adotado, considere que a inequação representada pela cor azul foi invertida. Logo, é possível observar a existência de uma região mesmo invertendo a inequação, o que indica que a

inequação em azul não é redundante. A Figura 2.5 destaca essa afirmação, com a região em preto sendo a solução viável do subproblema formado.

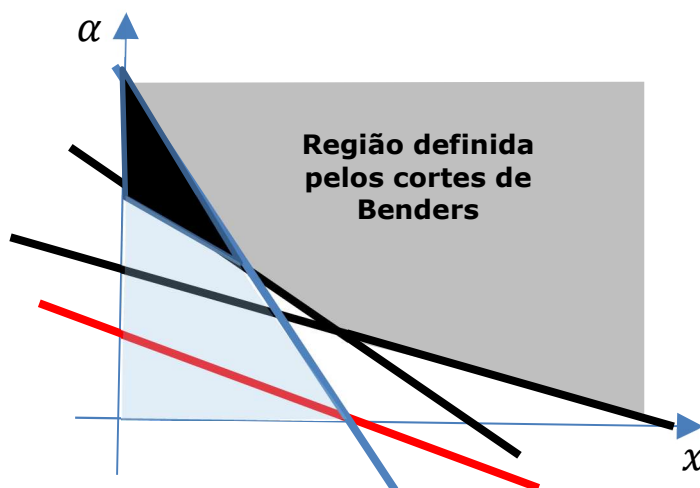


Figura 2.5 - Procedimento de Shapiro – Invertendo um corte não redundante.

Já em relação à inequação em vermelho, ao invertermos tal inequação, temos que nenhuma região pode ser formada com as demais inequações, visto que não existe interseção entre elas. Isso significa que a inequação em vermelho é redundante, logo sua retirada não afeta a curva que define a FCF. A Figura 2.6 destaca o processo descrito.

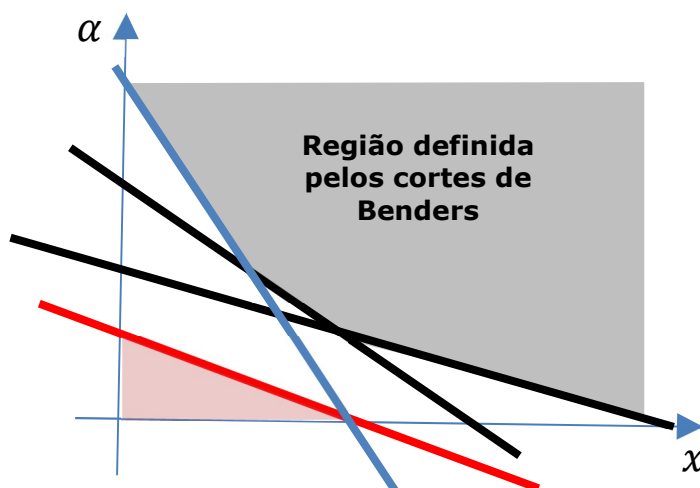


Figura 2.6 - Procedimento de Shapiro – Invertendo um corte redundante.

Assim, considerando as variáveis de estados $x \in \Gamma$, o procedimento de Shapiro verifica a viabilidade da construção de uma região definida pelos cortes de Benders ao inverter o sentido de um corte do conjunto de cortes em relação aos demais. Dessa forma, o procedimento proposto originalmente por Shapiro infere que, se a região assim definida não cria um poliedro, o corte analisado não contribui e pode ser descartado.

A implementação desta verificação deste algoritmo foi realizada como um problema de otimização, conforme apresentado a seguir. Para isso, considere as variáveis α como custos futuros e x como as variáveis de estado até a iteração em que a análise de redundância é realizada. Esta análise considera cortes pertencentes a um conjunto I . Cada corte deste conjunto possui um valor escalar RHS e coeficientes β , associados à variável de estado x . O procedimento de Shapiro Modificado introduz as variáveis de folga f e s , minimizando-as com o objetivo de verificar a não redundância.

Assim, assumindo uma tolerância para o critério de eliminação de cortes, o seguinte procedimento é realizado:

Para cada corte j , faça:

Resolva o problema de programação linear

minimizar $f_j + s_j$

sujeito a $\alpha \leq RHS^j + (\beta^j)^T x + f_j - s_j$

$\alpha \geq RHS^c + (\beta^c)^T x + f_c - s_c$, para todo $c \in I$ exceto o corte j

$f_j, s_j \geq 0$

Caso $(f_j + s_j)$ for maior que a tolerância adotada, então:

Descarte o corte j do conjunto I

Optou-se por esta formulação, como um problema de otimização incluindo variáveis de folga, a fim de garantir a existência do poliedro que o procedimento original de Shapiro descreve, assim eliminando resultados indesejados das heurísticas de pre-solve.

Esta estratégia proposta de verificação de redundância de cortes, apesar de acurada e robusta, é um procedimento com alto custo computacional. Sua acurácia em definir cortes redundantes no momento de sua aplicação é um ponto forte, mas o impacto do tempo de execução e do consumo de recursos deve ser levado em conta. Nesse sentido, a busca por alternativas menos onerosas computacionalmente pode ser necessária para equilibrar eficiência e desempenho sem comprometer o tempo computacional ao obter os cortes redundantes, como o algoritmo de Análise Por Pares por exemplo.

3 ESTRATÉGIAS DE PARALELIZAÇÃO

A versão 30.1 do modelo NEWAVE introduz a possibilidade de realizar a eliminação de cortes de Benders ao longo do processo iterativo da PDDE, trazendo também avanços na forma como a eliminação pode ser computada. Em relação à estratégia utilizada para a paralelização da computação dos algoritmos de eliminação de cortes, foram exploradas duas maneiras distintas. A primeira delas é uma evolução natural da paralelização já existente na computação das etapas *forward* e *backward* do processo iterativo da PDDE. A segunda é uma abordagem alternativa, que permite explorar maneiras mais eficientes de avaliação do conjunto de cortes, preservando a premissa fundamental de reprodutibilidade de resultados no modelo NEWAVE.

A primeira maneira de distribuir o trabalho de rotulagem dos cortes redundantes ocorre durante a etapa *backward*, imediatamente após o término de construção dos NSIM cortes de cada período t . Neste momento do processamento, há um ponto de sincronismo no paralelismo, permitindo que todos os processos disponíveis sejam utilizados para realizar a eliminação de cortes utilizando os algoritmos de Análise por Pares e Shapiro Modificado. Nesta abordagem, a eliminação de cortes construído no período t para serem utilizados em $t - 1$ é realizada de maneira que cada processo receba um subconjunto dos cortes a serem analisados. A distribuição dos cortes entre os processos é feita buscando maximizar a homogeneidade dos trabalhos em cada processo de maneira a reduzir o tempo de execução.

Para ilustrar como ocorre essa divisão, considere o seguinte exemplo: seja um caso típico de PMO com 200 cenários *forwards*, com a primeira aplicação da eliminação de cortes na 10ª iteração e execução realizada com 56 processos. Nesta situação, existem 2000 cortes a serem analisados no período. Com isso, 40 processos serão responsáveis por avaliar 36 cortes e 16 processos avaliarão 35 cortes.

Quanto à segunda estratégia de distribuição do trabalho de rotulagem dos cortes redundantes, a abordagem adotada consiste em atribuir a responsabilidade pela eliminação do conjunto completo de cortes de determinado período a um único processo. Essa diferença, ao centralizar a análise de todo o conjunto de cortes de um período, possibilita explorar o reaproveitamento de cálculos e avaliações previamente realizados, sem haver perda de reprodutibilidade de resultados mesmo com variações na quantidade de processos utilizados na computação. Adicionalmente, em comparação à primeira estratégia, essa abordagem reduz a ociosidade, eliminando os tempos de espera nos pontos de sincronismo que podem ocorrer a cada estágio na primeira solução. Em contrapartida, como essa estratégia é realizada após a passagem de um ou mais ciclos *backward*, o benefício da utilização de menos cortes de Benders nos problemas só ocorrerá no ciclo *forward* seguinte à eliminação.

Nesta segunda estratégia, todos os cortes de Benders de um dado período são enviados para um processo, que avalia cada um dos que foram recém gerados na iteração corrente contra todos os demais cortes não eliminados das iterações anteriores. No caso de algum corte ser eliminado, ou este corte não entra na lista de cortes, caso o eliminado seja o corte recém-criado, ou este corte sai da lista de cortes não eliminados, caso a eliminação seja do corte de uma iteração mais antiga. É importante frisar que, caso o corte recém-criado não seja eliminado, ele já entra na lista de cortes não eliminados e passa a participar da avaliação dos demais cortes recém-criados. Essa estratégia inicia-se com a criação de um comunicador para abranger todos os nós necessários para a avaliação dos cortes de todos os períodos do caso corrente. Caso a quantidade de processos seja maior que a quantidade de períodos, os processos excedentes não farão parte deste novo comunicador, não participando do processo, ficando ociosos durante a fase de avaliação da eliminação dos cortes e sincronizados posteriormente em uma barreira de execução ao término dessa fase.

Os processos que participam do novo comunicador (chamado de MPI_COMM_ELIM), e consequentemente da eliminação dos cortes, serão classificados em 3 possíveis tipos:

- (1) um processo para ler os arquivos de cortes e enviar os dados para os processos que efetivamente avaliarão os cortes;
- (2) um processo para receber a relação dos cortes que foram eliminados e atualizar essa informação nos arquivos de cortes;
- (3) demais processos para avaliar todos os cortes de um período por vez.

Desta forma, na quantidade de processos que são reservados para avaliar os cortes, é necessária a adição de dois processos extras para executar a leitura e a escrita das informações dos cortes. Tanto o processo que fará a leitura dos arquivos quanto o que fará a escrita precisam ter acesso aos arquivos de cortes. Como estes arquivos são manipulados pelo processo "0" do comunicador padrão do MPI (MPI_COMM_WORLD), escolheu-se esse mesmo processo para realizar as tarefas de leitura e envio dos cortes dos períodos para os demais. Para executar as tarefas de receber as listas dos cortes que foram eliminados e escrevê-las nos arquivos de cortes, foi necessário escolher um outro processo que pertencesse ao mesmo nó do processo 0 do MPI_COMM_WORLD. A escolha de um processo que está no mesmo nó do processo 0 se deve ao fato de que essa condição é necessária para que se garanta que esse processo consiga abrir e manipular os arquivos, pois ele terá acesso à pasta onde o processo 0 gerou os arquivos. Todos os demais processos do comunicador MPI_COMM_ELIM executam a avaliação dos cortes.

Inicialmente, cada processo que executa a avaliação recebe os cortes de um determinado período, de forma que, caso a quantidade de processos consiga atender a todos os períodos, cada um somente executará a avaliação completa de um período. Na Figura 3.1 pode-se observar como é executada a estratégia de paralelização com o processo 0 enviando para os processos 2, 3 até N+1, os cortes de cada período para cada processo e depois cada processo envia a sua lista de cortes eliminados para o processo 1 executar a atualização de cada um dos arquivos de cortes.

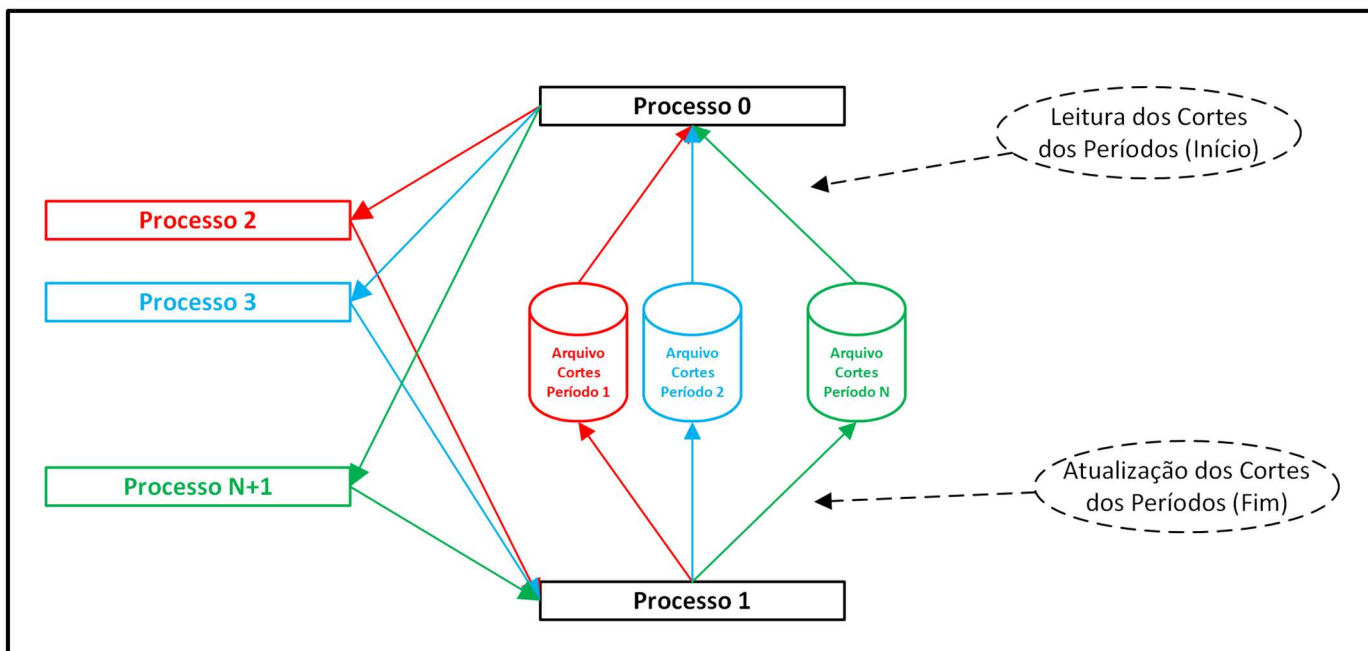


Figura 3.1 – Estratégia de Paralelização por Pares em Período Completo

Após essa primeira fase, caso a quantidade de períodos seja maior que a quantidade de processos, os dados dos cortes de cada período remanescente serão distribuídos para o primeiro processo que terminar a sua avaliação e assim sucessivamente, até encerrar a lista de períodos. Ao final da lista de períodos, cada processo recebe uma sinalização que não existe mais nenhum período com cortes a serem avaliados e saem do processo de eliminação. Nesta eventual segunda fase, o processo 0 atua de forma gerencial, enviando os dados dos cortes para o primeiro processo que encerrar a sua avaliação. Desta forma, é possível que alguns processos realizem mais avaliações do que outros, principalmente quando um processo realizar a avaliação de um período agregado, que possui cortes significativamente menores em relação aos processos que ficaram com os períodos individualizados.

Convém ressaltar que o algoritmo para eliminação de cortes paralelos continua sendo aplicado apenas no processo 0 do comunicador MPI_COMM_WORLD no momento da escrita do arquivo de cortes por período, assim como era feito antes da versão 30.1.

momentos podem ser iniciados. A ordem de execução foi planejada de forma a seguir a ordem crescente de complexidade e de tempo dos algoritmos, com os algoritmos mais complexos tendo o benefício de avaliar menos cortes.

Os limites superior (L_{sup}) e inferior (L_{inf}) do domínio das variáveis de estado associados às afluições passadas, para cada usina/REE em cada mês, utilizadas nas análises de eliminação de cortes com a aplicação do fator (fat) são definidas como:

$$L_{sup} = \overline{afl} + fat \cdot |\overline{afl} - \underline{afl}|$$

$$L_{inf} = \underline{afl} - fat \cdot |\overline{afl} - \underline{afl}|$$

Onde,

$\overline{afl}$: maior valor de afluição utilizado no cálculo da política/simulação final, para cada usina/REE em cada mês

$\underline{afl}$: menor valor de afluição utilizado no cálculo da política/simulação final, para cada usina/REE em cada mês

Adicionalmente, os cortes eliminados podem variar conforme o tipo de simulação final adotada (séries sintéticas ou históricas) ao ativar a flag para consideração dos cenários da simulação final na definição dos limites das afluições passadas.

5 ARQUIVOS DE SAÍDA

5.1 Eco da entrada de dados (pmo.dat)

A impressão do sinalizador da ativação da funcionalidade da eliminação de cortes é realizada no relatório de acompanhamento do programa, conforme mostrado na Figura 5.1.

222	ARQUIVOS DE CORTES POR PERIODOS	TODOS SERAO MANTIDOS
223		
224	JUNCAO DOS ARQUIVOS DE CORTES NUM UNICO ARQUIVO	NAO
225		
226	CONSIDERA ELIMINACAO DE CORTES	SIM, POR CORTE
227		

Figura 5.1 – Impressão do sinalizador da ativação da funcionalidade da eliminação de cortes.

Além disso, no relatório de acompanhamento do programa, também são impressos os parâmetros escolhidos pelo usuário para o procedimento de eliminação de cortes, conforme mostrado na Figura 5.2.

281	***DADOS ELIMINACAO DE CORTES***			
282				
283		PARALELOS	POR_PARES	SHAP.MODIF.
284		XXXXXXXXXXXX	XXXXXXXXXXXX	XXXXXXXXXXXX
285				
286	ALGORITMOS PARA REALIZACAO DA ELIMINACAO DE CORTES	SIM	SIM	SIM
287	ITERACAO INICIAL PARA REALIZACAO DA ELIMINACAO DE CORTES:	1	20	20
288	PASSO PARA REALIZACAO DA ELIMINACAO DE CORTES:	1	10	10
289	JANELA DE ITE. DE CONSTRUCAO DOS CORTES A SEREM AVALIADOS:	-	50	50
290				
291	FATOR APLICADO AOS LIMITES DAS AFLUENCIAS PASSADAS (%):	10.00		
292	AFLUENCIAS DA SIMULACAO FINAL NO CALCULO DOS LIMITES:	SIM		
293	IMPRESSAO DOS RELATORIOS DA ELIMINACAO DE CORTES:	SIM		

Figura 5.2 – Impressão dos parâmetros escolhidos pelo usuário para a eliminação de cortes.

5.2 Relatórios do processo de eliminação de cortes

5.2.1 Relatório com os cortes eliminados (eliminacao-cortes-rel.csv)

Este arquivo contém as seguintes informações para cada corte eliminado:

- ITE: iteração de aplicação da eliminação de cortes;
- PERIODO: período de uso do corte;
- DATA: data (mês/ano) referente ao período de uso do corte;
- REG_COR_PER: registro do corte no arquivo de cortes por períodos (cortes-xxx.dat);
- ITE_COR: iteração de construção do corte eliminado;
- ALGORITMO: algoritmo que eliminou o corte

eliminao-cortes-rel.csv							
1	ITE,	PERIODO,	DATA,	REG_COR_PER,	ITE_COR,	ALGORITMO	
2	1,	23,	11/2023,	3,	1,	PARALELOS	
3	1,	23,	11/2023,	4,	1,	PARALELOS	
4	1,	23,	11/2023,	1,	1,	PARALELOS	
5	1,	23,	11/2023,	6,	1,	PARALELOS	
6	1,	23,	11/2023,	7,	1,	PARALELOS	
7	1,	23,	11/2023,	8,	1,	PARALELOS	
8	1,	23,	11/2023,	9,	1,	PARALELOS	
9	1,	23,	11/2023,	10,	1,	PARALELOS	
10	1,	23,	11/2023,	11,	1,	PARALELOS	
11	1,	23,	11/2023,	12,	1,	PARALELOS	
12	1,	23,	11/2023,	14,	1,	PARALELOS	
13	1,	23,	11/2023,	13,	1,	PARALELOS	
14	1,	23,	11/2023,	15,	1,	PARALELOS	
15	1,	23,	11/2023,	16,	1,	PARALELOS	
16	1,	23,	11/2023,	17,	1,	PARALELOS	

Figura 5.3 – Relatório da eliminação de cortes.

5.2.2 Relatório com os limites considerados para cada variável de estado no processo de eliminação de cortes (eliminacao-cortes-limites-variaveis.csv)

Este arquivo contém as seguintes informações para cada variável de estado que compõe os cortes de cada período:

- PERIODO: período de uso do corte;
- DATA: data (mês/ano) referente ao período de uso do corte;
- ELEMENTO: coluna contendo informações quanto ao elemento de análise a ser utilizada na coluna VARIÁVEL. Essa coluna pode ser:
 - REE xxxxx: nome do REE xxxxx;
 - UHE xxxxx: nome da usina hidrelétrica xxxxx;
 - SBM xxxxx: nome do submercado xxxxx;
- VARIÁVEL: nome da variável de estado, podendo ser:
 - EARMfxxx: Energia armazenada final do REE xxx, em MWmês;
 - EAFpasxxx_yy: ENA passada do REE xxx, yy meses anteriores, em MWmês;
 - VARMfxxx: Volume armazenado final da usina hidrelétrica xxx, em hm³;
 - VAFpasxxx_yy: Volume afluente passado da usina hidrelétrica xxx, em yy meses anteriores, em hm³;
 - SGTxxxPyLz: Soma da geração térmica antecipada das usinas a GNL do submercado xxx no patamar de carga y e lag z, em MWmês;
 - MX_CGUIAxxx: Máxima violação da curva guia (VminOp) do REE xxx, em MWmês;
- LIMITE INFERIOR: valor referente ao limite inferior da variável;

- LIMITE SUPERIOR: valor referente ao limite superior da variável.

1	PERIODO,	DATA,	ELEMENTO,	VARIÁVEL,	LIMITE INFERIOR,	LIMITE SUPERIOR
2	23,	11/2023,	REE SUDESTE	EARMf001	0.000,	15433.600
3	23,	11/2023,	REE NORDESTE	EARMf003	0.000,	24945.350
4	23,	11/2023,	REE SUDESTE	EAfpas001_01	-601.732,	6619.052
5	23,	11/2023,	REE SUDESTE	EAfpas001_02	-273.000,	3003.003
6	23,	11/2023,	REE SUDESTE	EAfpas001_03	-158.934,	1748.273
7	23,	11/2023,	REE SUDESTE	EAfpas001_04	-160.834,	1769.174
8	23,	11/2023,	REE SUDESTE	EAfpas001_05	-186.587,	2052.457
9	23,	11/2023,	REE SUDESTE	EAfpas001_06	-253.375,	2787.125
10	23,	11/2023,	REE SUDESTE	EAfpas001_07	-329.359,	3622.949
11	23,	11/2023,	REE SUDESTE	EAfpas001_08	-631.230,	6943.530
12	23,	11/2023,	REE SUDESTE	EAfpas001_09	-971.270,	10683.970
13	23,	11/2023,	REE SUDESTE	EAfpas001_10	-1016.914,	11186.054
14	23,	11/2023,	REE SUDESTE	EAfpas001_11	-1406.403,	15470.433
15	23,	11/2023,	REE SUDESTE	EAfpas001_12	-803.774,	8841.514
16	23,	11/2023,	REE NORDESTE	EAfpas003_01	-857.497,	9432.467
17	23,	11/2023,	REE NORDESTE	EAfpas003_02	-388.413,	4272.543
18	23,	11/2023,	REE NORDESTE	EAfpas003_03	-432.190,	4754.090

Figura 5.4 – Relatório com os limites considerados para as variáveis de estado no processo de eliminação de cortes.

5.3 Tempo de execução

O tempo consumido na aplicação da eliminação de cortes é escrito no arquivo que fornece o tempo de execução de cada etapa do processo iterativo (newave.tim). Nesse arquivo, são escritos tanto o tempo total da eliminação de cortes ao longo de todo o processo iterativo quanto o tempo da eliminação de cortes em cada iteração que teve sua aplicação.

```

-----
Leitura de Dados:      000h 00min 01s
Calculos Iniciais:    000h 11min 06s
Calculo da Politica:  000h 07min 32s
Eliminacao de cortes: 000h 00min 22s
Simulacao Final:      000h 02min 14s
Tempo Total:          000h 21min 15s
-----

```

```

-----
Iteracao  1 - Backward: 000h 00min 45s
              Forward:  000h 00min 14s
              ElimCor:  000h 00min 03s
              Total:    000h 01min 02s
Iteracao  2 - Backward: 000h 01min 04s
              Forward:  000h 00min 14s
              ElimCor:  000h 00min 04s
              Total:    000h 01min 22s
Iteracao  3 - Backward: 000h 01min 26s
              Forward:  000h 00min 17s
              ElimCor:  000h 00min 05s
              Total:    000h 01min 48s
-----

```

Figura 5.5 – Relatório com o tempo da eliminação de cortes.

6 RESULTADOS

Os testes numéricos foram executados em máquina do Google Cloud Plataform, com 56 núcleos de processamento e um total de 448 Gigabytes de memória RAM, sendo cada caso executados com 50 nós. O caso exemplo a ser executado é o PMO híbrido de janeiro/2024, com 60 períodos mensais de planejamento da operação (sendo 12 meses individualizados e os demais 48 meses como reservatórios equivalentes), além de um período pós-estudo de 60 períodos.

Como hipóteses de execução, foram adotados que:

- o algoritmo Paralelos será executado em todas as iterações, independente das combinações;
- o fator aplicado ao limite de afluições passadas foi de 5%;
- a tolerância adotada para o critério de eliminação de cortes nos algoritmos Por Pares e Shapiro Modificado foi de 10^{-5} .

Os testes numéricos foram executados utilizando a versão 30.2 do NEWAVE, de forma a considerar as seguintes combinações dos algoritmos apresentados na Seção 2, variando alguns parâmetros nas execuções. Os gráficos a serem apresentados utilizam os dados apresentados na Seção 4 e 5 do presente relatório.

Vale destacar que existem diversos parâmetros envolvidos no processo de eliminação de cortes (vide Seção 4.1), cuja combinação, pode afetar significativamente o tempo computacional dos algoritmos. Nesta seção, não é realizado um estudo exaustivo de ajuste fino desses parâmetros, em função da complexidade computacional envolvida e da carga de experimentação necessária. Em vez disso, são apresentadas combinações selecionadas com base em análises preliminares conduzidas durante o desenvolvimento e implementação destes algoritmos, visando um equilíbrio entre eficiência e viabilidade computacional.

6.1 Análise de estratégias de computação paralela na execução do algoritmo de Eliminação de Cortes Por Pares

Para esta seção, será analisado o caso exemplo quanto à opção de paralelismo definido pelo usuário. Existem duas opções disponibilizadas: por cortes, aplicação essa realizada durante os períodos da etapa *backward*, e por período, cuja estratégia é realizada após a etapa *backward* (vide Seção 3). Como forma de comparação, será observada a combinação Paralelos + Por Pares, considerando a seleção de cortes e uma janela de iterações a serem consideradas igual a 50. A fim de estudar também os parâmetros dos algoritmos, foram executadas as seguintes combinações com o Paralelos + Por Pares:

- Opção por Cortes, iniciando na iteração 20 com passo 10;
- Opção por Períodos, iniciando na iteração 20 com passo 10;
- Opção por Cortes, iniciando na iteração 1 com passo 1;
- Opção por Períodos, iniciando na iteração 1 com passo 1.

Os resultados esperados são de alteração apenas de tempo computacional, visto que o problema se mantém.

Em relação ao resultado da PMO, o ZINF obtido no processo iterativo e o custo operacional são destacados na Figura 6.1. Não são evidenciadas diferenças significativas no ZINF, quando comparadas as opções de paralelismo de eliminação de cortes Por Pares. Para se ter uma noção da diferença do ZINF obtido com a combinação de algoritmos Paralelos+Por Pares, a Figura 6.1 também ilustra uma diferença relativa à execução do mesmo caso sem eliminação de cortes.

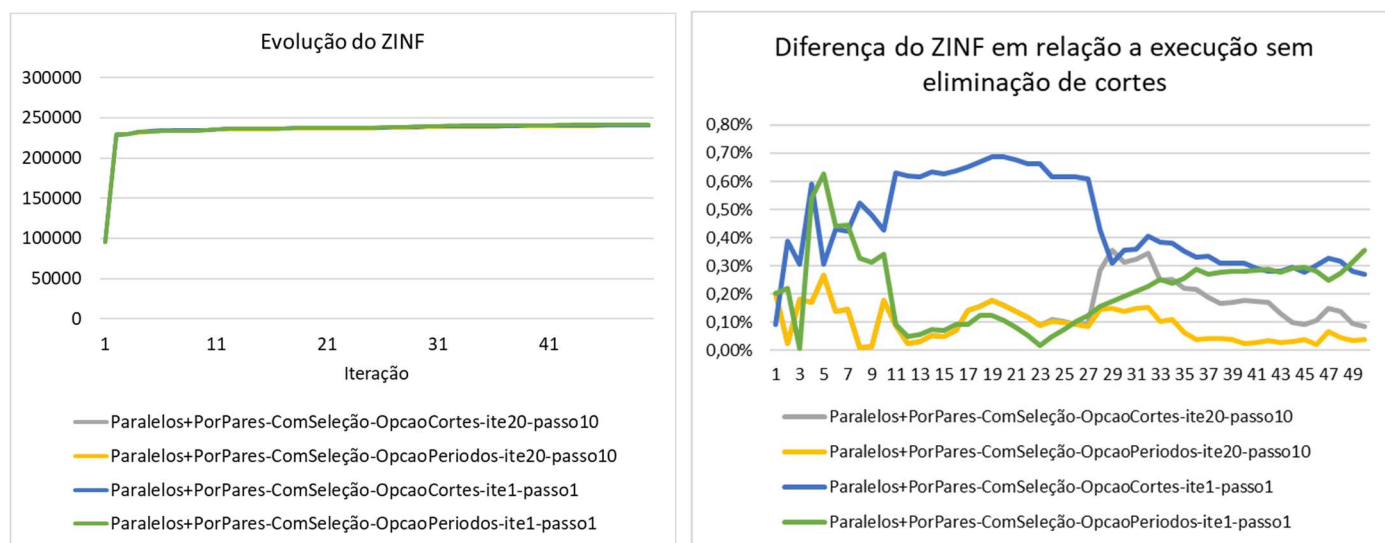
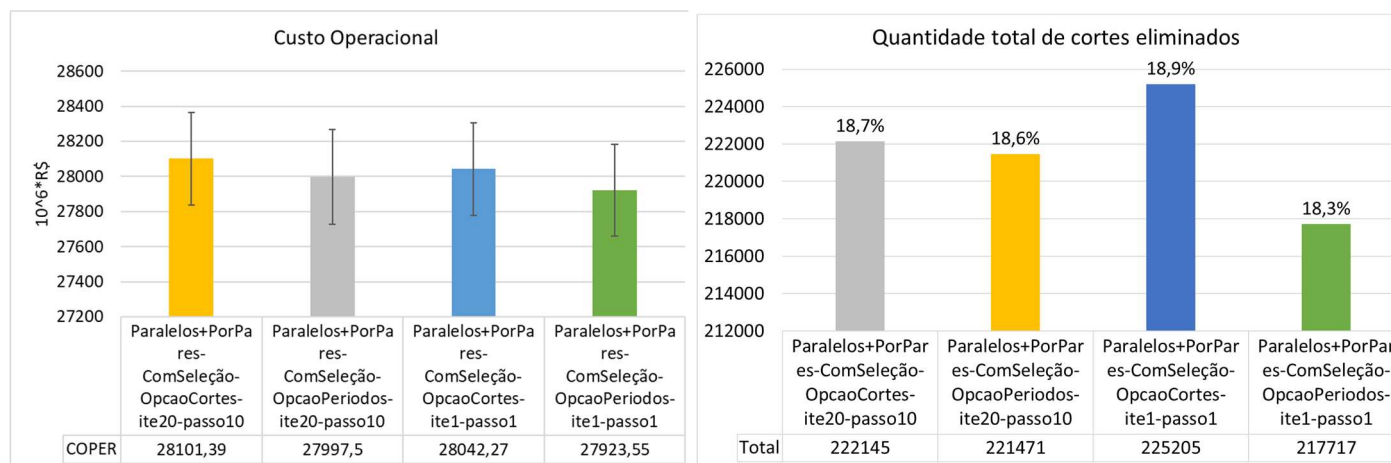


Figura 6.1 – Evolução do ZINF e a diferença dos valores entre as opções de paralelismo por cortes e por períodos.

Em relação ao custo operacional e à quantidade de cortes eliminados, conforme a Figura 6.2 ilustra, os valores obtidos estão dentro do intervalo definido no desvio padrão. Assim, não existem indícios de que a curva que define a FCF tenha sido alterada com a inclusão dos algoritmos de eliminação de cortes.



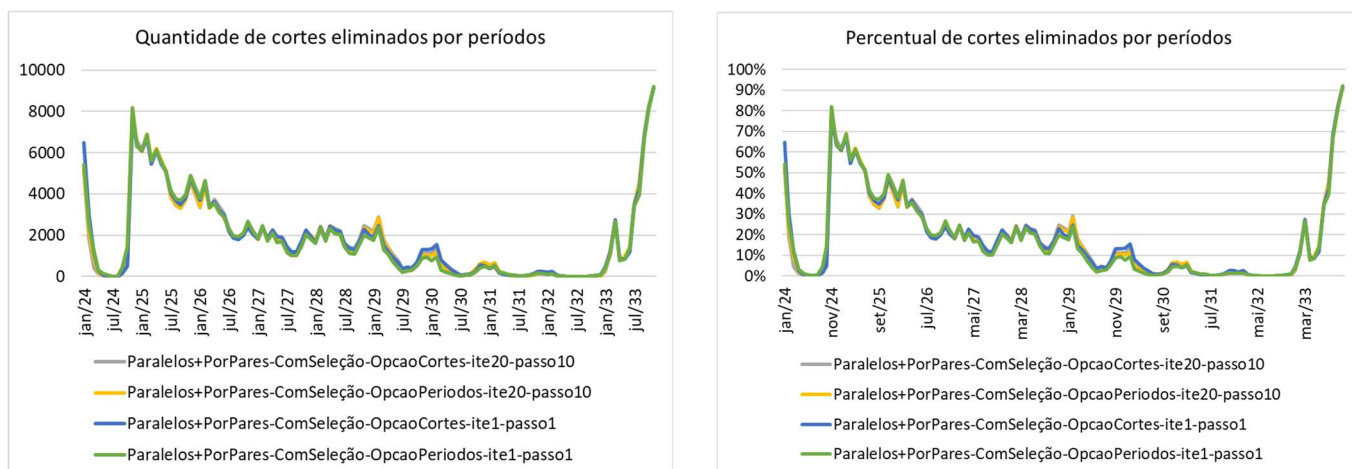


Figura 6.2 – Custo operacional e o total de cortes eliminados entre as opções de paralelismo por cortes e por períodos.

A principal diferença entre as estratégias de paralelismo recai no tempo de processamento, como será evidenciado a seguir na Figura 6.3 e Figura 6.4. Nos valores apresentados na Figura 6.3, observa-se que a opção de paralelismo por períodos reduz em o tempo de execução do algoritmo Por Pares em 50,5% quando iniciado na iteração 20, passo 10, e em 94% quando iniciado na iteração 1, passo 1. Aqui é apresentado uma evidência de que a opção de paralelismo Por Períodos é interessante quando aplicado desde o início do cálculo da política.

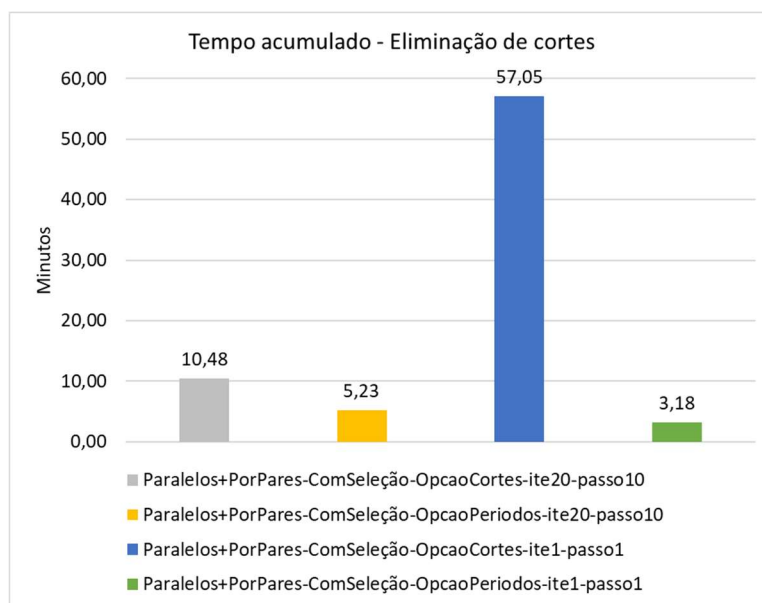


Figura 6.3 – Tempo acumulado na fase de eliminação de cortes.

Como observado na Figura 6.3, durante a execução do NEWAVE, foi constatado que a opção de paralelismo por períodos reduz o tempo da fase de Eliminação de Cortes com a combinação Paralelos+Por Pares. A Figura 6.4 ilustra a execução das fases *backward* e *forward*, bem como a eliminação de cortes e o tempo total, por iteração. Observa-se que as fases do cálculo da política sofrem um aumento de

tempo de processamento. Esse efeito é esperado, pois a opção por períodos é executada somente ao final da etapa *backward*, o que significa que o benefício de sua execução será registrado só na próxima iteração. Mesmo assim, o tempo total de execução não sofreu uma diferença significativa.

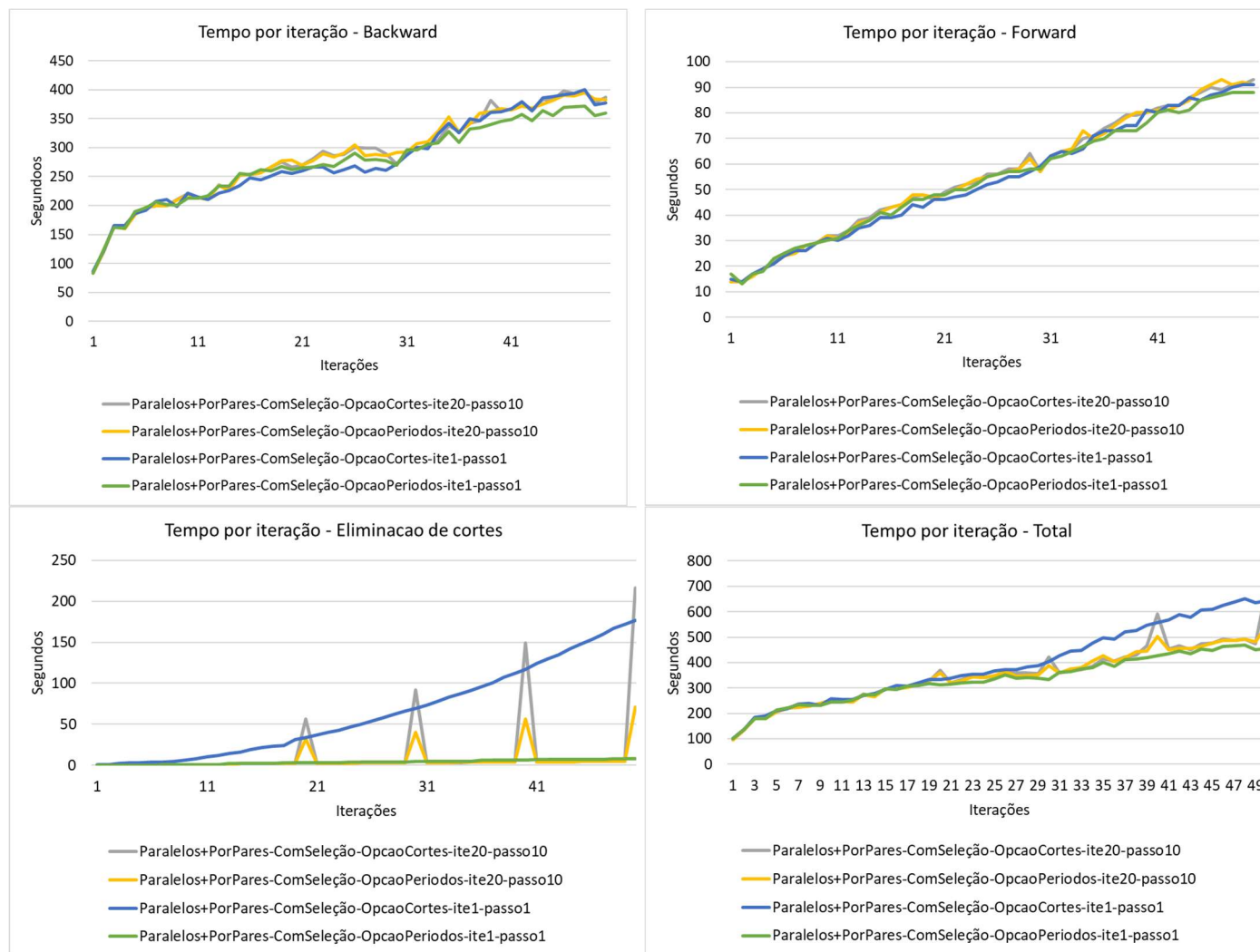


Figura 6.4 – Análise de tempo por período para o passo *forward* e *backward*.

Pela Figura 6.4 fica visível o comportamento do tempo utilizado para computar a eliminação de cortes, sendo que a opção por períodos levando menos tempo por iteração para executar o algoritmo Por Pares começando desde o início das iterações.

Desta forma, as estratégias de paralelismo possuem diferenças no tempo de processamento na fase de eliminação de cortes, sendo que a opção de paralelismo Por Períodos levou um tempo inferior para ser executada em comparação com a opção Por Cortes (vide Figura 6.3). No tempo total de execução, foi observado ganhos significativos com a opção por períodos começando desde o início do processo da PDDE (vide Figura 6.4), levando 3 minutos para a execução no tempo acumulado para a eliminação de cortes.

6.2 Aplicação concomitante de Seleção e Eliminação de Cortes

Para esta seção, será analisado o caso exemplo quando ativos a seleção de cortes, tanto na fase *forward* quanto *backward*, e os algoritmos de eliminação de cortes. Para a execução dos algoritmos de eliminação de cortes, serão comparadas as seguintes combinações:

- Sem eliminação de cortes;
- Com o algoritmo Paralelos, aplicado em toda iteração;
- Combinação Paralelos + Por Pares, com a opção de paralelismo por períodos, iniciando na iteração 1 com passo 1, com uma janela de iterações de tamanho 50 (vide Seção 6.1);
- Combinação Paralelos + Shapiro Modificado, com opção de paralelismo por cortes, iniciando na iteração 20 com passo 10, com uma janela de iterações de tamanho 50;
- Combinação Paralelos + Por Pares + Shapiro Modificado, com opção de paralelismo por cortes, sendo que ambos os algoritmos Por Pares e Shapiro Modificado iniciam na iteração 20 com passo 10, com uma janela de iterações de tamanho 50.

A combinação Paralelos + Shapiro Modificado foi utilizada para ressaltar a acurácia do último algoritmo citado, porém em testes preliminares foi identificado que o algoritmo de Shapiro Modificado eleva significativamente o tempo computacional. Assim, fixando o cálculo da política em até 50 iterações, optou-se por aplicar este algoritmo um total de 4 vezes (partindo da iteração 20 e reaplicado a cada 10 iterações posteriores).

Quanto a combinação Paralelos + Por Pares + Shapiro Modificado, foi replicado os parâmetros utilizados na combinação Paralelos + Shapiro Modificado, a fim de verificar se o algoritmo Por Pares consegue amenizar o custo computacional do algoritmo de Shapiro Modificado.

6.2.1 Comparativo dos resultados do PMO

Serão apresentadas as evidências envolvendo o valor inferior da função de custo futuro (ZINF) durante o processo iterativo de otimização para as combinações de algoritmos. Especificamente, ZINF representa o valor médio da função objetivo no primeiro estágio, ao longo das iterações, indicando um limite inferior para o valor ótimo da função objetivo do problema de planejamento da operação do sistema elétrico, que atualmente envolve parcelas associadas ao custo de operação e à medida de risco CVaR (*Conditional Value at-Risk*). Para a bateria de testes conduzida, espera-se que os valores de ZINF não sofram grandes alterações, uma vez que a proposta dos algoritmos de eliminação de cortes é de não alterar a curva que define a FCF.

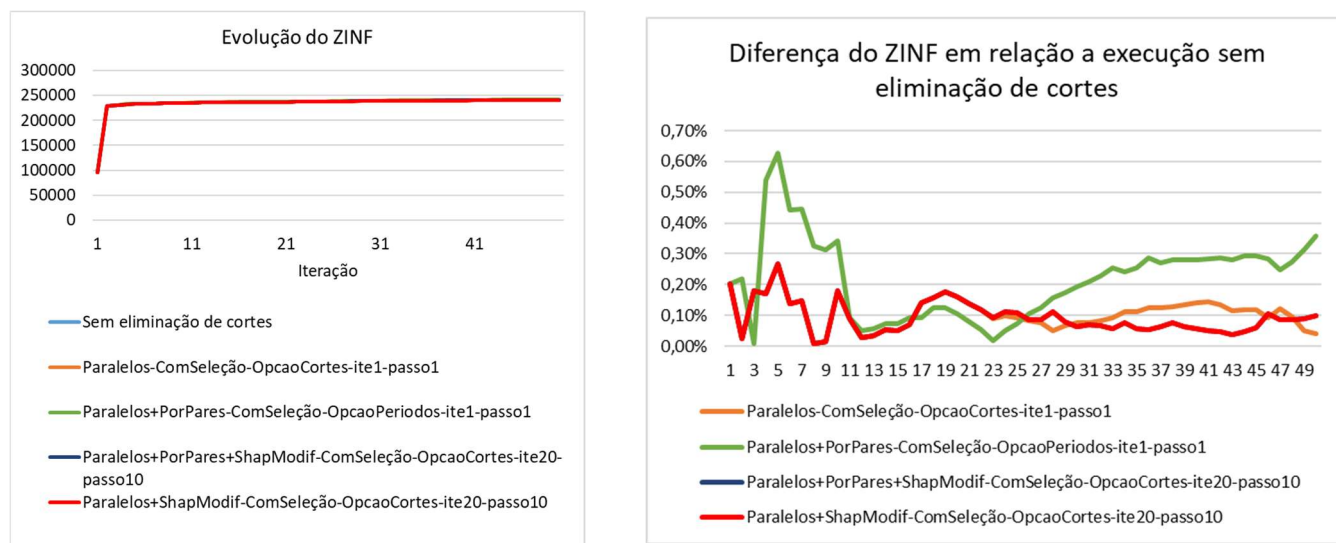


Figura 6.5 – Evolução do ZINF e a diferença dos valores entre as combinações de algoritmos com o a execução do algoritmo Paralelos.

A Figura 6.5 destaca a evolução do ZINF, bem como a diferença obtida entre as execuções sem considerar eliminação de cortes. Observa-se que as soluções obtidas são equivalentes, apesar de existir diferenças menores de 0,6%. Essa variação é justificável quanto as respostas do solver quando ocorre a extração de restrições, uma vez que o solver utilizado no processo iterativo pode encontrar outro ponto viável que reflete no mesmo valor da função objetivo. Desta forma, no processo de construção de novos cortes, ocorre a construção de cortes diferentes e, assim, obter uma leve variação na evolução do ZINF ao longo das iterações.

O impacto dessa variação pode ser observado no custo final da operação entre as execuções. Apesar de existir uma variação entre as execuções, a Figura 6.6 ilustra que os custos operativos (COPER) obtidos estão dentro do intervalo definido no desvio padrão. Assim, não existem indícios de que a curva que define a FCF tenha sido alterada com a inclusão dos algoritmos de eliminação de cortes.

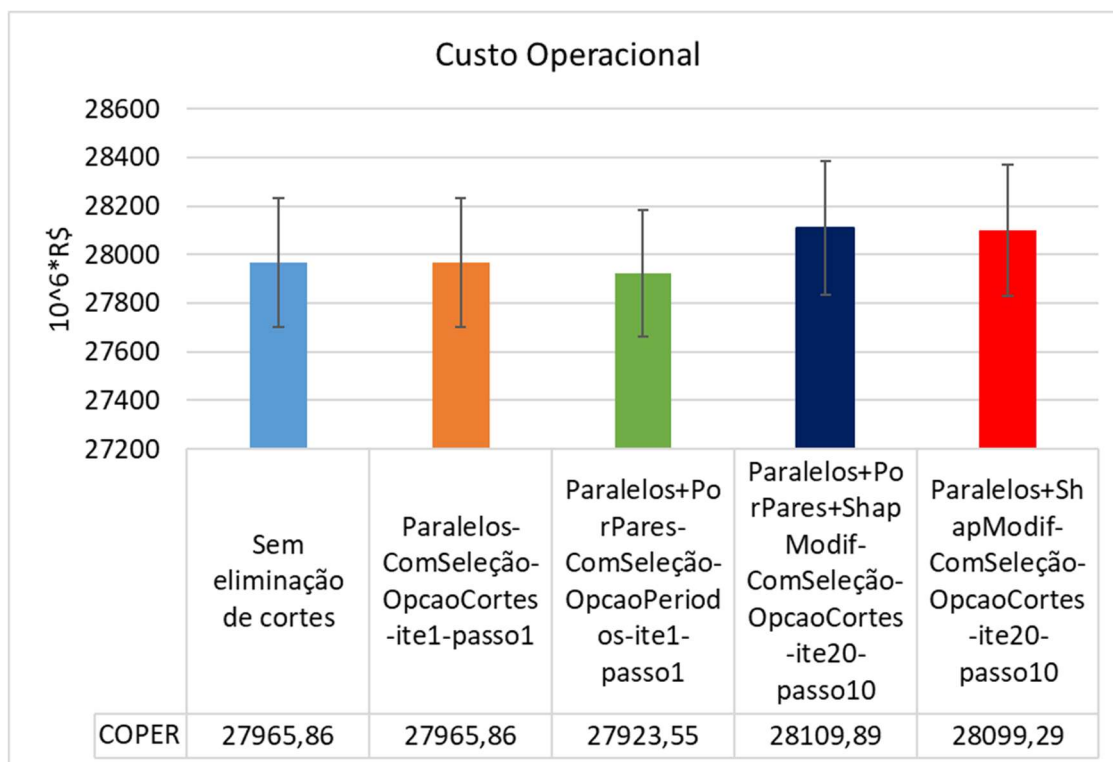


Figura 6.6 – Comparação do custo operativo do caso exemplo.

Naturalmente, a escolha do algoritmo de eliminação de cortes deve ser baseada nos objetivos específicos do estudo. Os resultados a seguir fornecerão subsídios técnicos importantes para conclusões quanto às performances dos algoritmos implementados.

6.2.2 Análise de tempo de execução

A Figura 6.7 até Figura 6.9 destacam os tempos acumulados e a redução percentual de execução dos algoritmos (em comparação a execução somente com algoritmo Paralelos) durante a execução do NEWAVE, destacando as fases *backward*, *forward* e eliminação de cortes, respectivamente.

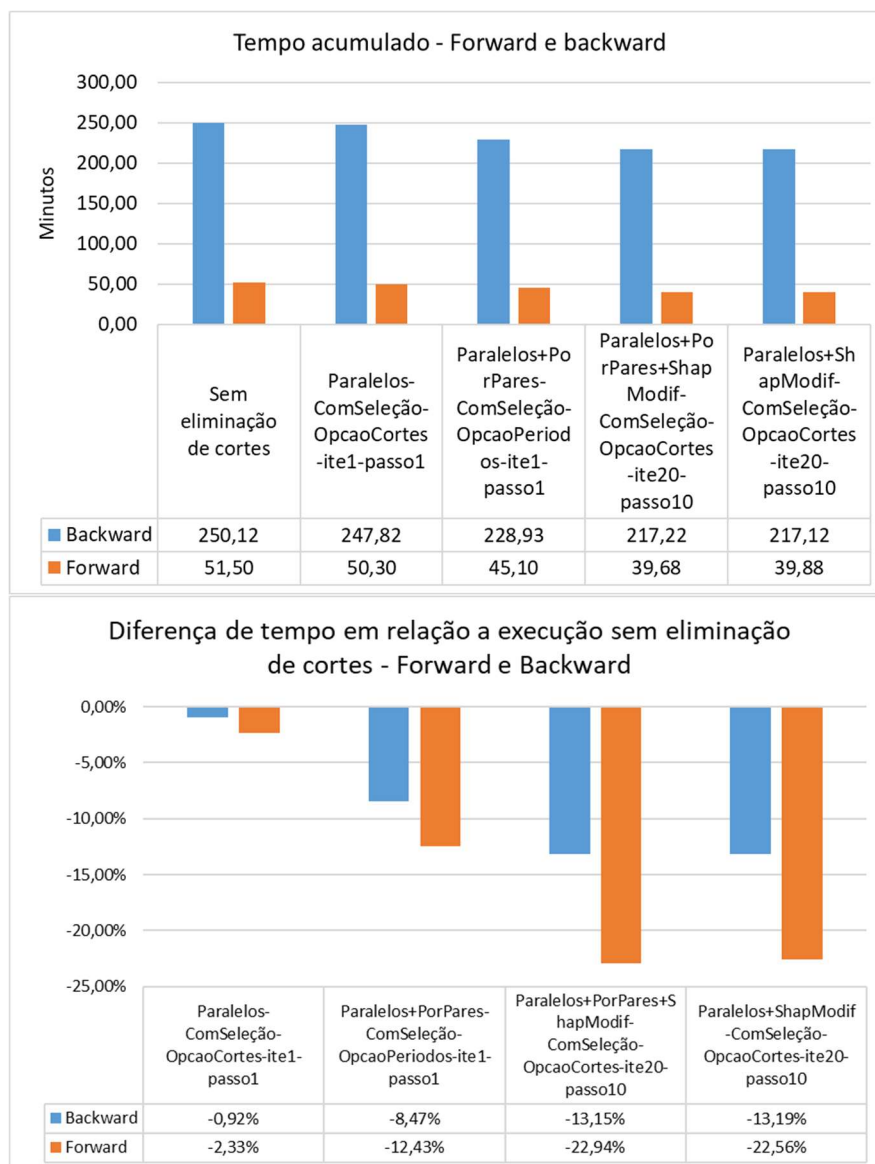


Figura 6.7 – Análise de tempo para o passo *forward* e *backward*.

Os gráficos apresentados na Figura 6.7 destacam que a execução sem considerar a eliminação de cortes e considerando o algoritmo Paralelos levaram os maiores tempos para realizar as etapas *forward* e *backward*. No caso do algoritmo paralelos, devido à natureza do próprio algoritmo, aloca-se mais espaço em memória para armazenar os demais cortes redundantes, tornando mais lenta a execução da PDDE. As demais execuções, combinando os algoritmos, apresentam uma queda de tempo acumulado durante o processo iterativo da PDDE, tornando-se uma evidência de que a inclusão destes algoritmos é benéfica para redução de tempo computacional.

Os destaques são as combinações dos algoritmos “Paralelos + Por Pares + Shapiro Modificado” e “Paralelos + Shapiro Modificado”, que reduziram mais de 20% o passo *forward* e 12% o passo *backward* em comparação com a execução somente com o algoritmo Paralelos. Essas combinações, que possuem

de diferença o algoritmo de Análise Por Pares, possuem resultados semelhantes dado que o algoritmo de Shapiro Modificado já incorpora, por construção, o algoritmo de Análise Por Pares.

De forma mais granular, a Figura 6.8 traz a evolução do tempo das combinações dos algoritmos de eliminação de cortes por iterações, tanto para o passo *forward* quanto para *backward*.

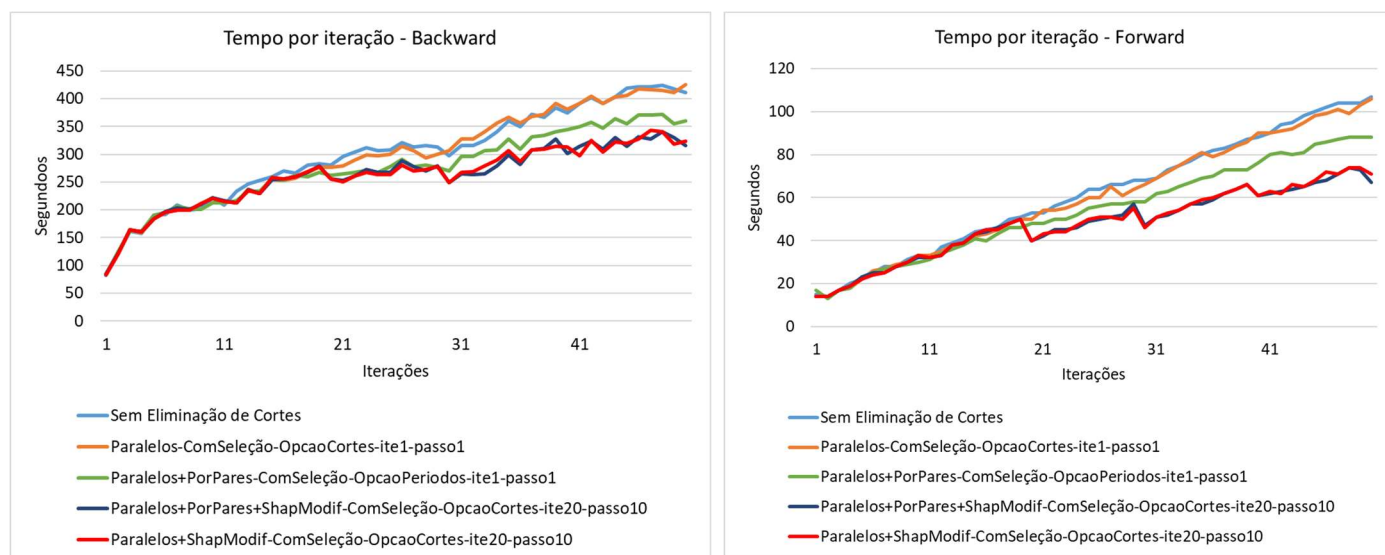


Figura 6.8 – Análise de tempo por iteração dos passos *backward* e *forward*.

Pela Figura 6.8, observa-se que os tempos possuem comportamentos diferentes a partir da 20ª iteração, iteração essa em que começa a ser aplicado os algoritmos de eliminação de corte Por Pares e Shapiro Modificado, corroborando ainda mais com o fato da redução de tempo dos passos da PDDE ao introduzir uma eliminação de cortes.

Apesar desta evidência de redução de tempo acumulado nas fases de otimização, a execução dos algoritmos de eliminação de cortes também exige tempo. A Figura 6.9 destaca o tempo acumulado e por iteração entre as comparações, evidenciando a velocidade de execução do algoritmo Paralelos (2,77 minutos) e da combinação Paralelos + Por Pares (3,2 minutos). Em contrapartida, as combinações envolvendo Shapiro Modificado possuem maiores tempos de execução, devida a solução dos subproblemas de otimização, representados nos picos em que o algoritmo foi executado.

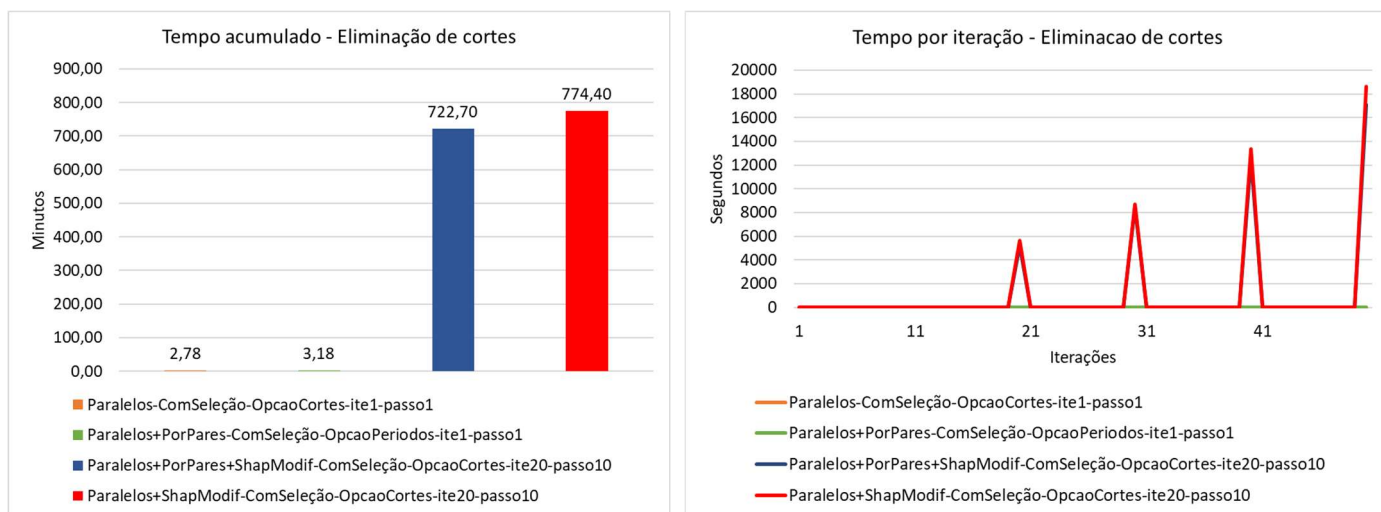
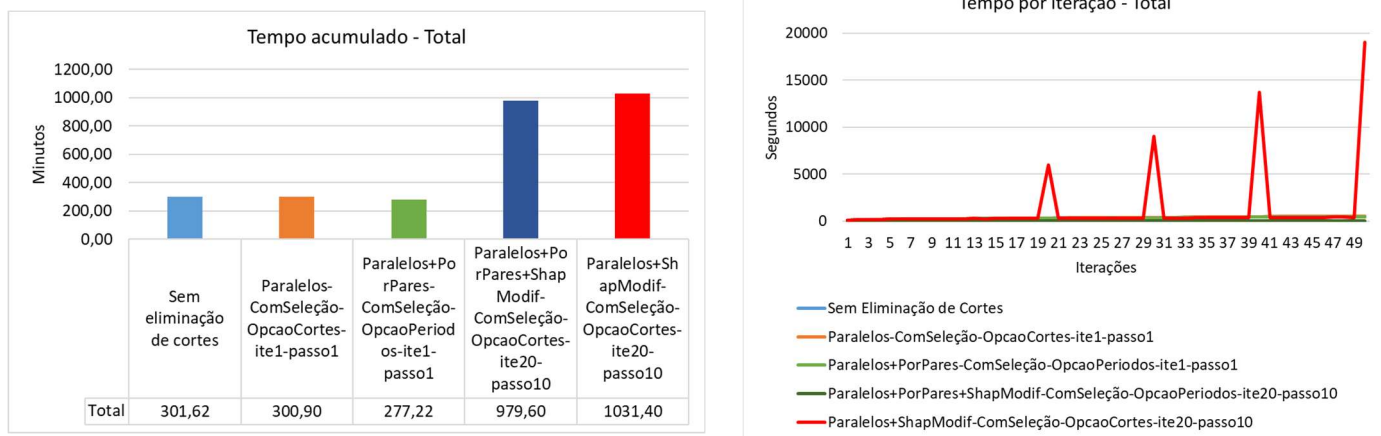


Figura 6.9 – Análise de tempo durante a eliminação de cortes.

Logo, fica evidente a existência de uma compensação envolvendo o algoritmo de Shapiro Modificado: tal algoritmo auxilia a reduzir o tempo de execução dos passos *forward* e *backward*, porém a um custo elevado de sua execução. Tal argumentação pode ser observada na Figura 6.10, onde se apresenta os valores totais de tempo de execução do caso exemplo e o valor percentual de redução de tempo em relação a execução do algoritmo Paralelos.



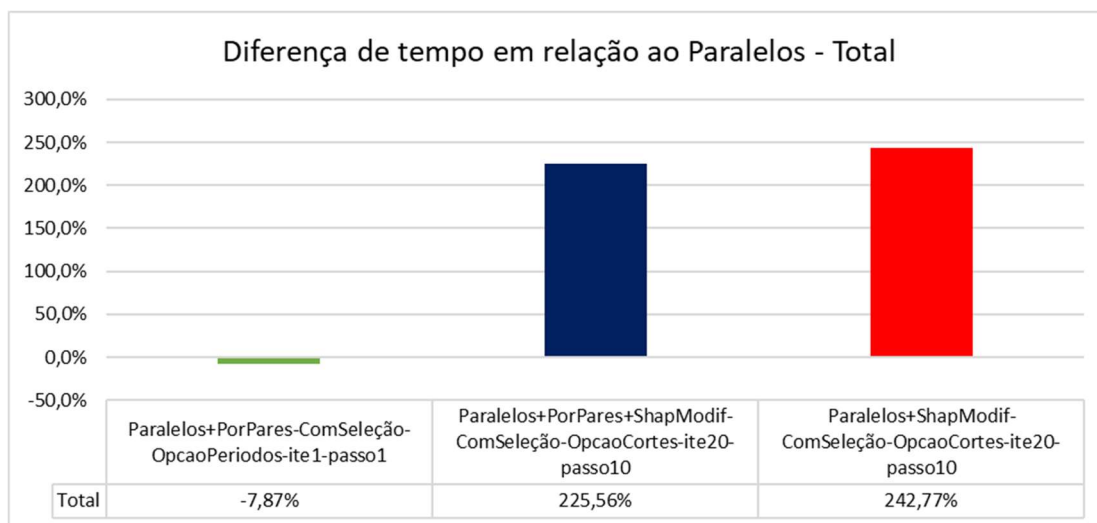


Figura 6.10 – Tempo acumulado total do caso exemplo.

Desta forma, a combinação dos algoritmos Paralelos + Por Pares apresenta boa performance computacional, apresentando uma redução de tempo de execução em relação ao algoritmo Paralelos (7,87%). Vale destacar que apesar do algoritmo Por Pares ser mais expedito, ele não consegue eliminar todos os cortes redundantes tal como o algoritmo de Shapiro Modificado faz, como será discutido na Seção 6.2.3.

Embora o algoritmo de Shapiro Modificado seja acurado na identificação de cortes redundantes, seu elevado custo computacional compromete sua aplicabilidade prática, especialmente nas situações em que o tempo de execução é crítico, como na resolução da PDDE. Nesse contexto, investir em técnicas de efficientização do algoritmo se torna essencial, dado o seu alto potencial de impacto na redução do tempo total de processamento. Melhorias na implementação, como a otimização de etapas críticas ou o uso de heurísticas, poderiam tornar o algoritmo mais competitivo, preservando sua eficácia enquanto reduz significativamente o tempo de processamento.

Outra possibilidade de aprimoramento é a aplicação da eliminação de cortes após o término da execução do NEWAVE. Isso pode ser benéfico para os modelos computacionais que utilizam a função de custo futuro do NEWAVE, ao reduzir a quantidade de cortes de Benders a serem processados. Por fim, deve-se avaliar o benefício de remover os cortes redundantes dos arquivos gerados, reduzindo seu tamanho e o seu tempo de processamento e leitura.

6.2.3 Análise de cortes eliminados

A Figura 6.11 destaca o total e o percentual de cortes eliminados nesta bateria de testes durante a execução do NEWAVE. É evidente a maior capacidade de eliminação de cortes quando se envolve o algoritmo Shapiro Modificado, em detrimento ao seu custo computacional já apresentado.

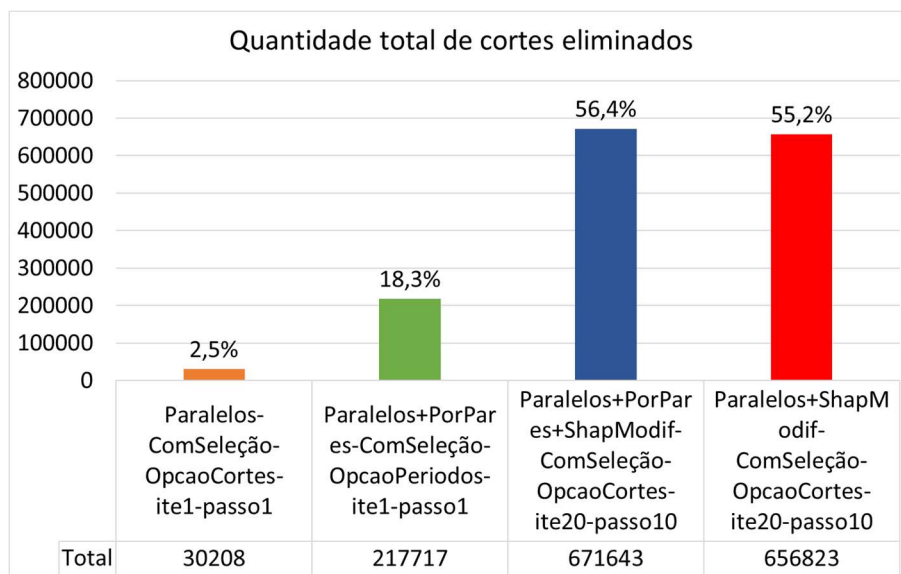


Figura 6.11 – Total de cortes eliminados.

Além disso, vale destacar também a performance de eliminação de cortes por iteração, bem como o percentual de cortes eliminados (em relação a 10000 cortes). A Figura 6.12 ilustram tais casos.

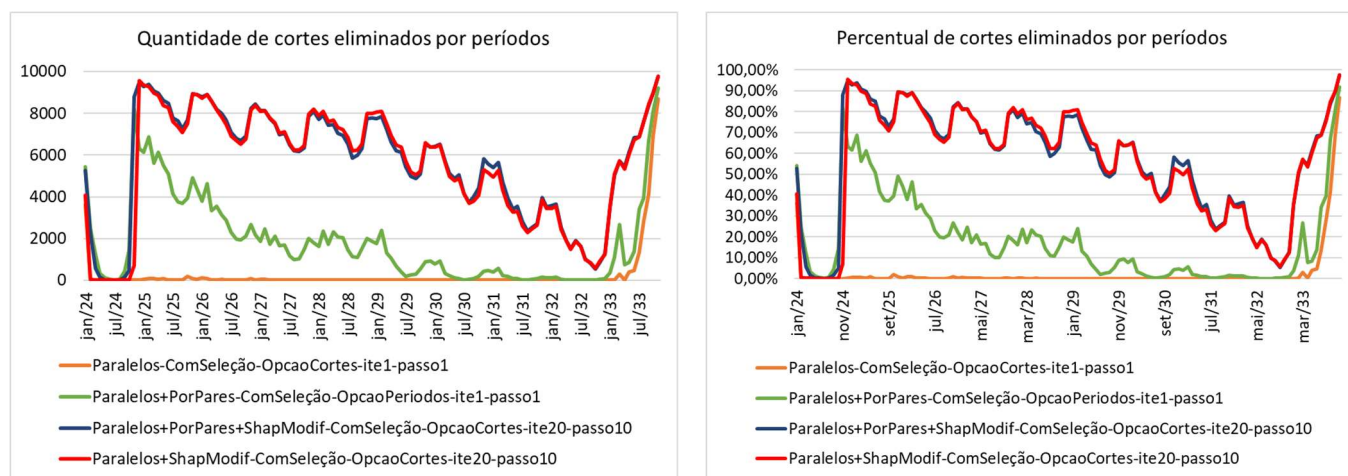


Figura 6.12 – Análise de cortes eliminados por período.

A Figura 6.12 evidencia que a execução somente com algoritmo Paralelos mantém uma eliminação de cortes constante e de menor magnitude ao longo dos períodos. Em contraste, os métodos que combinam abordagens apresentam reduções mais significativas, com picos expressivos de eliminação de cortes em períodos específicos. Destaca-se a linha amarela, que demonstra o maior número de cortes eliminados em diversos períodos, atingindo picos de aproximadamente 10000 cortes eliminados. Além disso, tem-se a evidência que em todas as combinações para este caso de PMO, uma iteração inicial interessante a ser escolhida é acima da iteração 10, bem como uma sazonalidade de queda na eliminação dos cortes.

Com os resultados apresentados, observa-se que:

- a combinação dos métodos Por Pares e Shapiro Modificado potencializa a eficiência da eliminação de cortes, especialmente em momentos críticos, apresentando-se como uma abordagem mais

robusta para lidar com a complexidade dos dados. Entretanto, o seu custo computacional é elevado;

- A inclusão do algoritmo Por Pares na combinação Paralelos + Por Pares + Shapiro Modificado reduziu pouco o tempo computacional total neste caso exemplo. Assim, abre-se margem para um estudo mais aprofundado na forma de tratar o algoritmo de Shapiro Modificado, a fim de melhorar o seu tempo de execução;
- A combinação Paralelos + Por Pares foi a mais eficiente neste teste, reduzindo o tempo computacional em 6% quando comparado com o caso sem eliminação de cortes.

6.3 Aplicação da Eliminação de Cortes em casos sem Seleção de Cortes

Para esta seção, será analisado o caso exemplo quando desativada a seleção de cortes, tanto na fase *forward* quanto *backward*. Para a execução dos algoritmos de eliminação de cortes, foi utilizada a opção de paralelismo por cortes.

Como forma de comparação, será observadas as combinações Paralelos + Por Pares e Paralelos + Shapiro Modificado, considerando a seleção de cortes. A fim de estudar também os parâmetros dos algoritmos, foram executadas variações destes algoritmos em relação as iterações iniciais e janela de iteração:

- Paralelos, iniciando na iteração 1 com passo 1;
- Paralelos + Por Pares, opção por Períodos de paralelismo, iniciando na iteração 1 com passo 1, janela de 50 iterações;
- Paralelos + Shapiro Modificado, opção por Cortes de paralelismo, iniciando na iteração 20 com passo 10, janela de 50 iterações.

O intuito deste teste numérico é avaliar o impacto da eliminação de cortes sem considerar a seleção de cortes nas fases *forward* e *backward*. A análise busca compreender como a seleção de cortes influenciam o desempenho computacional.

6.3.1 Comparativo dos resultados do PMO

Analogamente a Seção 6.2, serão apresentadas a evolução do ZINF durante o processo iterativo de otimização para as combinações de algoritmos, bem como a diferença percentual do ZINF por iterações em relação ao algoritmo Paralelos. Vale destacar aqui que os gráficos apresentados na Figura 6.13 são comparações das execuções quando não se considera seleção de cortes.

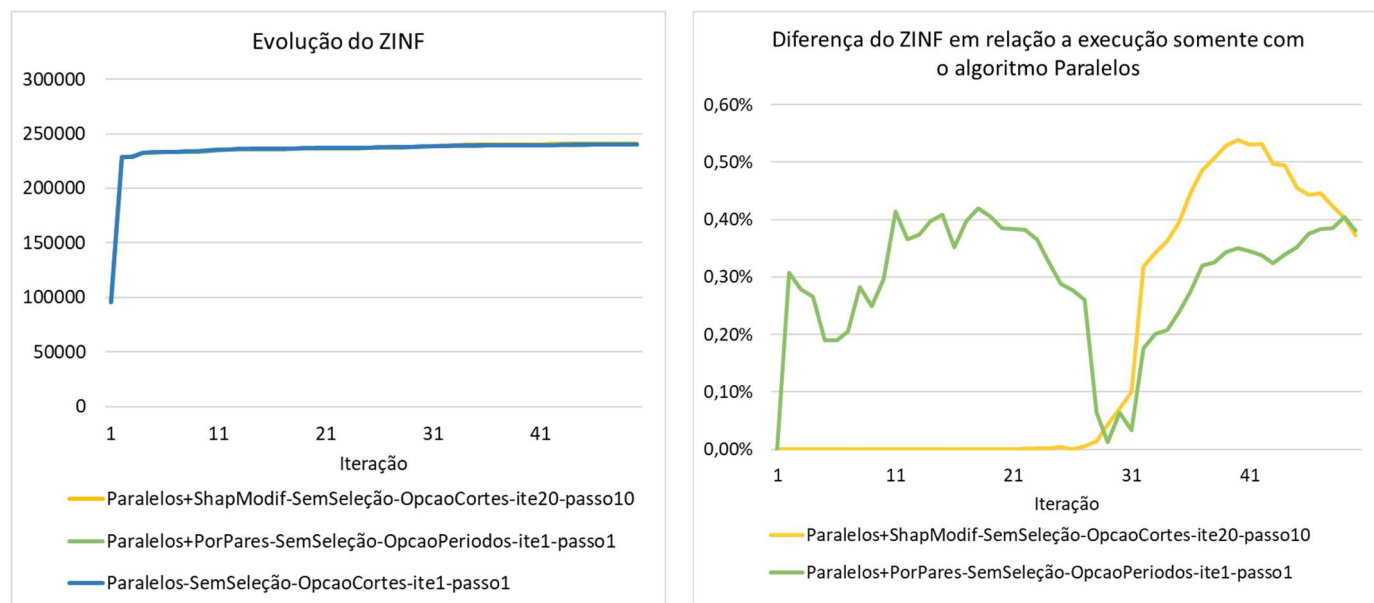


Figura 6.13 – Evolução do ZINF e a diferença dos valores entre as combinações de algoritmos com o a execução do algoritmo Paralelos.

A Figura 6.13 destaca a evolução do ZINF e a diferença percentual seguindo os resultados obtidos na Seção 6.2, obtendo apenas diferenças numéricas. Como já justificado, essa variação recai sobre as respostas do solver quando ocorre a extração de restrições, influenciando no processo de construção de novos cortes.

A Figura 6.6 ilustra que os COPER obtidos na rodada sem seleção de cortes, sendo que ambas as execuções estão dentro do intervalo definido no desvio padrão.

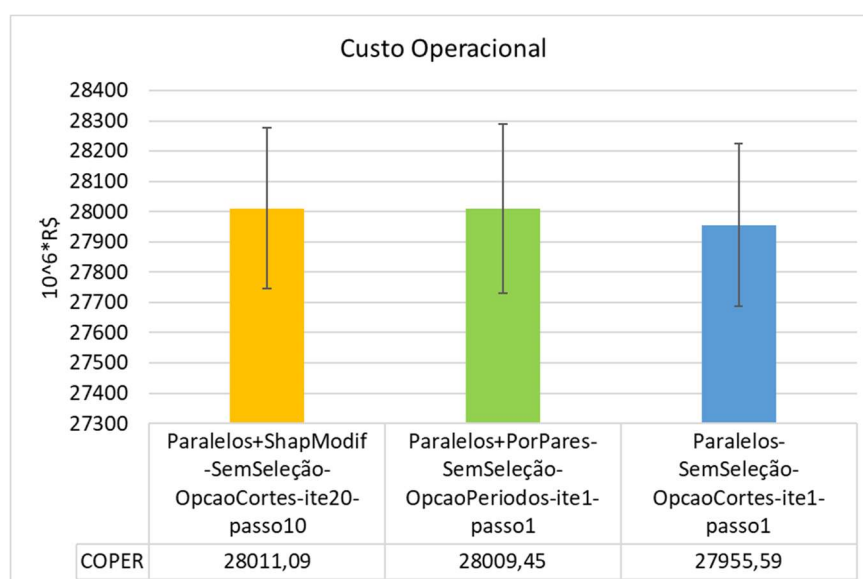


Figura 6.14 – Comparação do custo operativo do caso exemplo – sem seleção de cortes.

Assim como concluído na Seção 6.2, a inclusão dos algoritmos de eliminação de cortes não indicou alterações na curva que define a FCF do problema de otimização.

6.3.2 Análise de tempo de execução

A Figura 6.15 até a Figura 6.17 destacam os tempos acumulados e a redução percentual de execução dos algoritmos (em comparação a execução somente com algoritmo Paralelos) durante a execução do NEWAVE, destacando as fases *backward*, *forward* e eliminação de cortes, respectivamente.

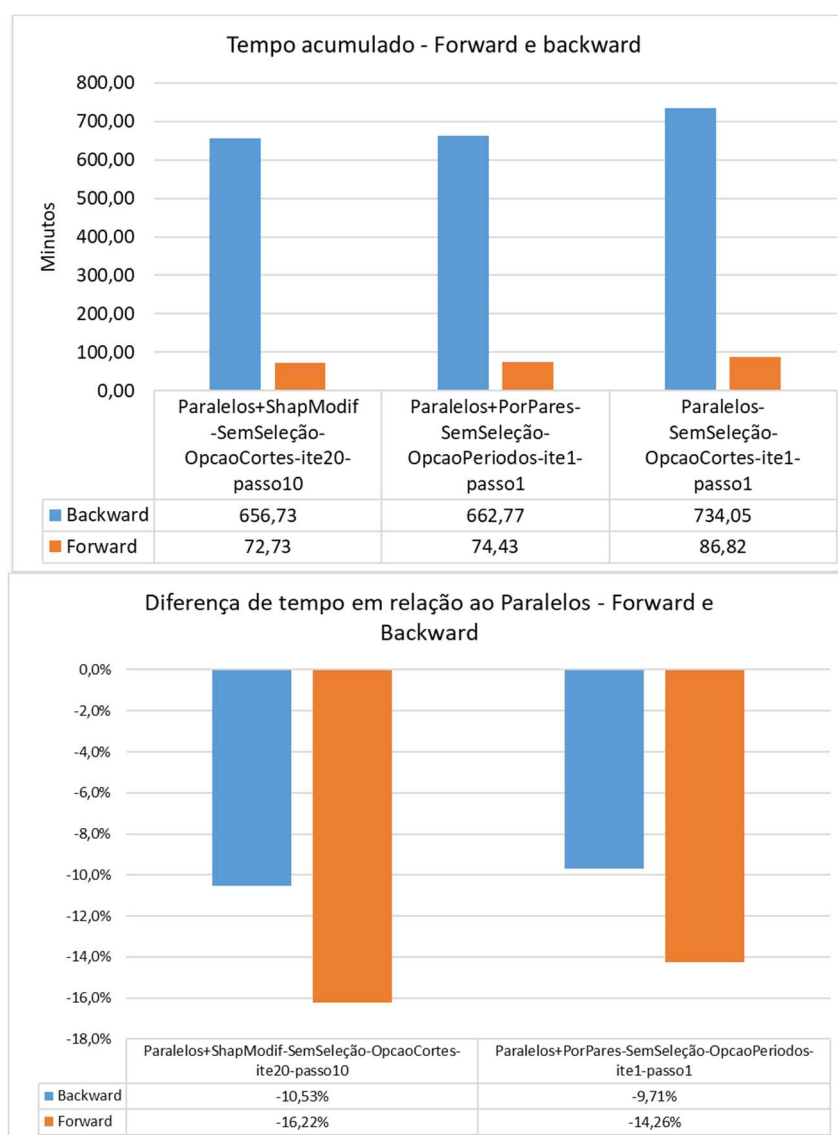


Figura 6.15 – Análise de tempo para o passo *forward* e *backward*.

Assim como ocorreu na Seção 6.2, o algoritmo Paralelos levou o maior tempo para realizar a etapa *backward*. As demais execuções, combinando os algoritmos, apresentam uma queda de tempo acumulado durante o processo iterativo da PDDE, com destaque a combinação dos algoritmos “Paralelos + Shapiro Modificado”, que reduziu 16,22% o passo *forward* e 10,53% o passo *backward* em comparação

com a execução somente com o algoritmo Paralelos. Por outro lado, comparando com os resultados apresentados na Seção 6.2.2 (vide Figura 6.7) considerando de forma concomitante a seleção e eliminação de cortes, o tempo de execução total sem considerar a seleção de cortes aumentou por volta de 400 minutos em média a execução do caso exemplo

De forma mais detalhada, a Figura 6.16 apresenta a evolução do tempo das combinações dos algoritmos de eliminação de cortes ao longo das iterações, abrangendo tanto o passo *forward* quanto o *backward*. Mais uma vez, fica evidente o efeito de utilizar algoritmos de eliminação de cortes na PDDE, uma vez que ocorre reduções consideráveis de tempo em relação a execução considerando somente o algoritmo Paralelos sem seleção de cortes.

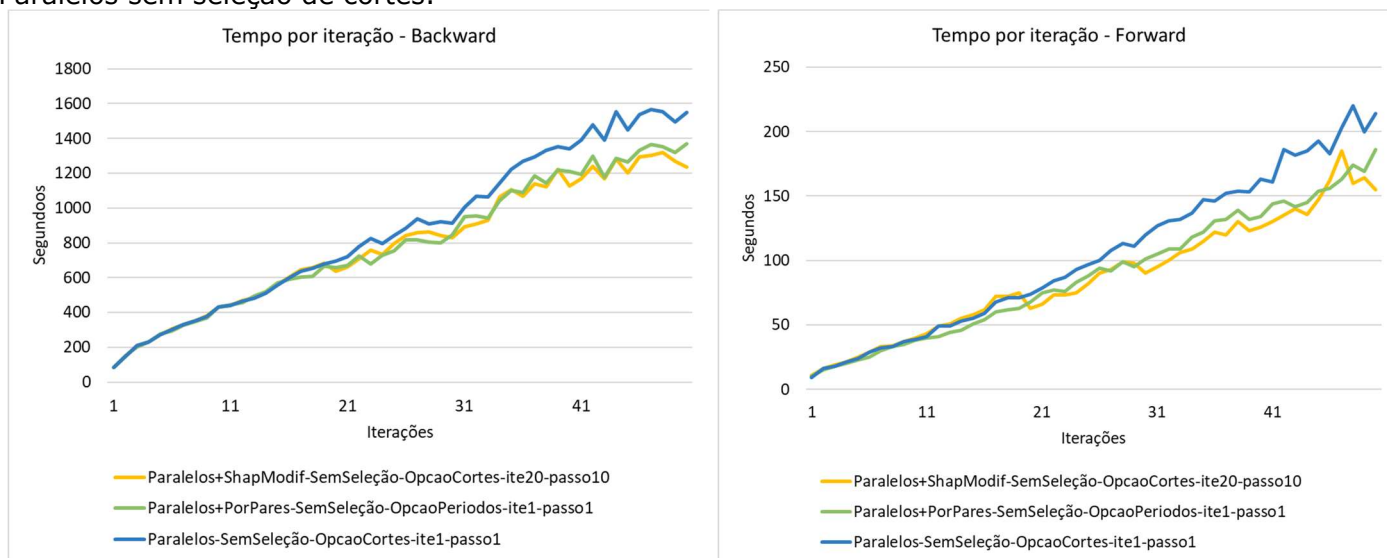


Figura 6.16 – Análise de tempo por iteração dos passos *backward* e *forward* – sem seleção de cortes.

Quanto ao tempo obtido para a fase de eliminação de cortes, a Figura 6.17 ilustra os resultados do tempo acumulado e por iteração entre as comparações. Assim como o resultado obtido considerando seleção de cortes, a Figura 6.17 evidencia a velocidade do algoritmo Paralelos e Por Pares.

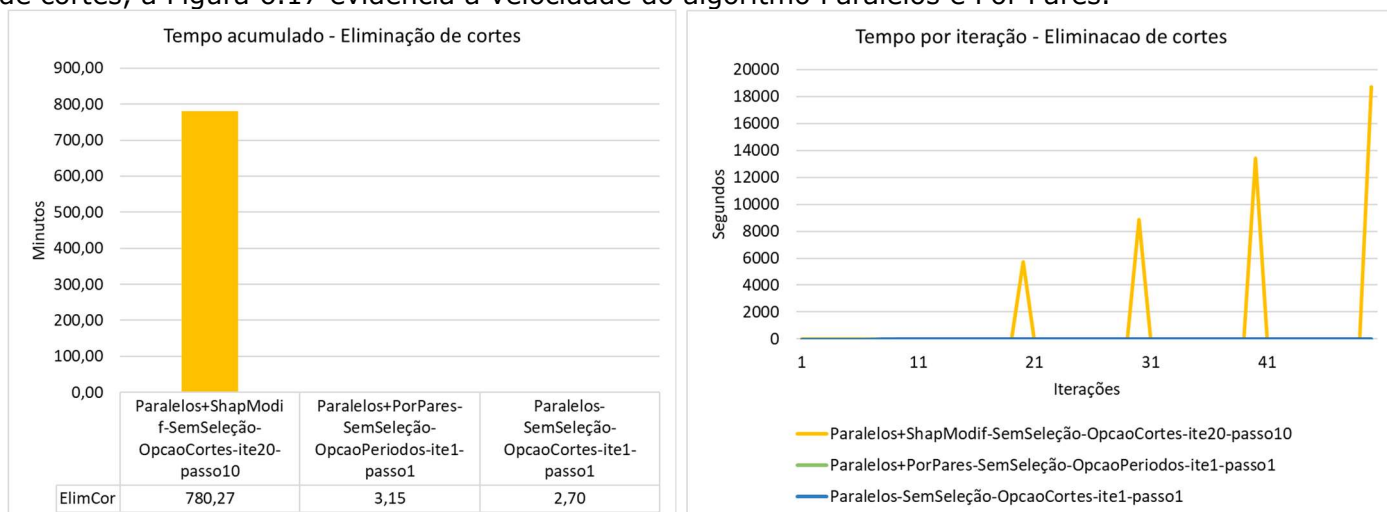


Figura 6.17 – Análise de tempo durante a eliminação de cortes - sem seleção de cortes.

Bem como constatado, o algoritmo de Shapiro Modificado auxilia a reduzir o tempo de execução dos passos *forward* e *backward*, porém a um custo computacional elevado. A Figura 6.18 reforça a argumentação, apresentando os valores totais de tempo de execução do caso exemplo e o valor percentual de redução de tempo em relação a execução do algoritmo Paralelos (sem considerar a seleção de cortes).

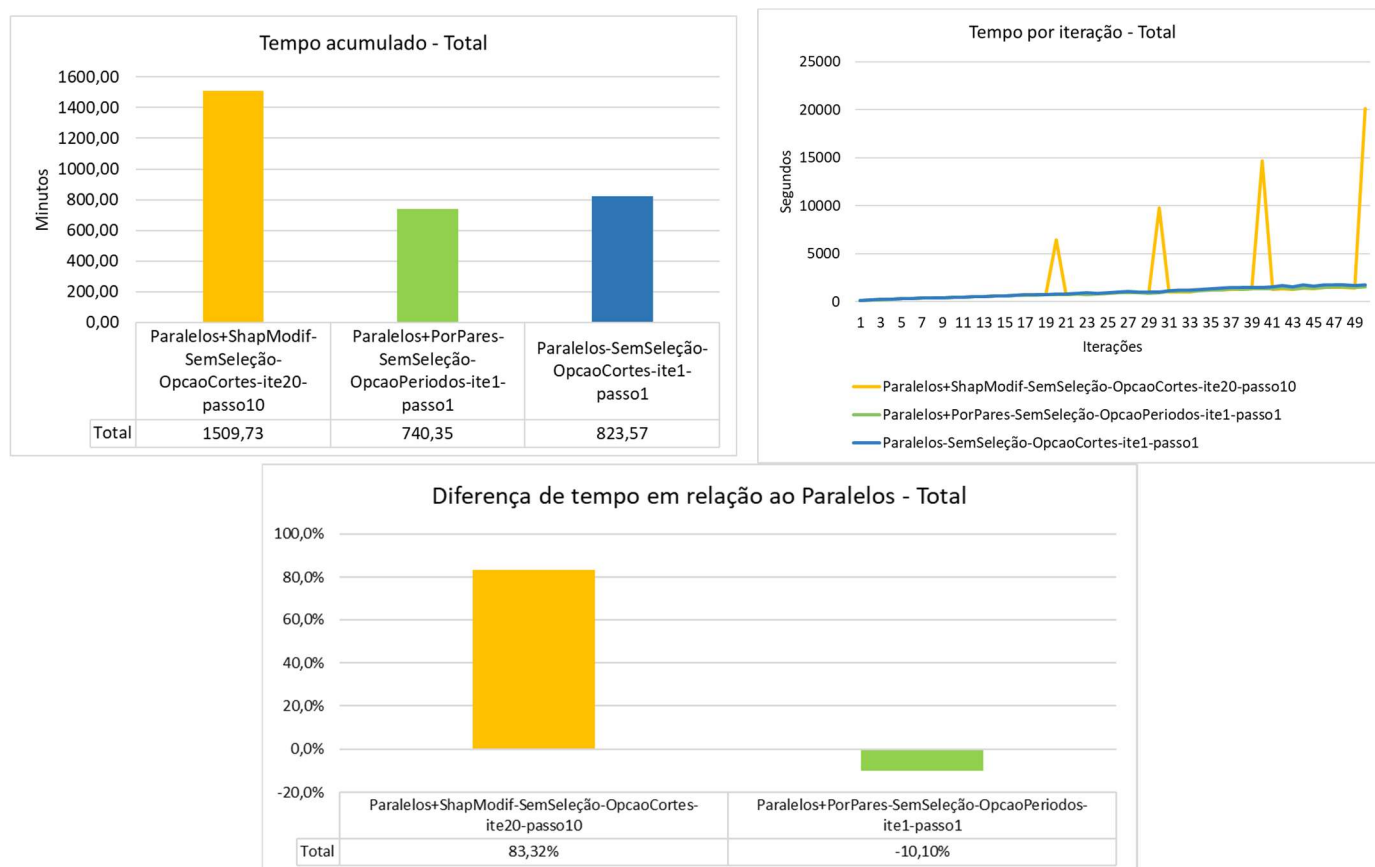


Figura 6.18 – Tempo acumulado total do caso exemplo.

A Figura 6.18 evidencia que a combinação dos algoritmos Paralelos + Por Pares apresenta boa performance computacional, reduzindo o tempo de execução em relação ao algoritmo Paralelos em aproximadamente 10,1%. Vale destacar que apesar do algoritmo Por Pares ser mais expedito, ele não consegue eliminar todos os cortes redundantes tal como o algoritmo de Shapiro Modificado faz.

Apesar desta redução apresentada, o tempo total de processamento é maior quando se observa a bateria de testes considerando a seleção de cortes na Seção 6.2. Esse aumento de tempo é esperado, uma vez que todos os cortes estão sendo incluídos na resolução dos PLs resolvidos, aumentando a carga computacional. A seguir, a Tabela 1 compila os tempos obtidos para a *backward*, *forward*, eliminação de cortes e total para as combinações de algoritmos de eliminação de cortes, com e sem seleção de cortes, com os menores tempo em negrito.

A Tabela 1 destaca, mais uma vez, a velocidade dos algoritmos Paralelos e Por Pares, obtendo os melhores tempos totais, bem como a vantagem de se utilizar a seleção de cortes de forma concomitante

com a combinação de métodos de eliminação de cortes Paralelos + Por Pares, reduzindo o tempo computacional na PDDE.

Tabela 1 – Tempos acumulados para as combinações de algoritmos de eliminação de cortes, com e sem seleção de cortes.

Combinações	Seleção de cortes	Backward (min)	Forward (min)	ElimCor (min)	Total (min)	Redução (%) de tempo quando utilizada a seleção de cortes
Paralelos	Sim	247,82	50,30	2,78	300,90	63,46%
	Não	734,05	86,82	2,70	823,57	
Paralelos+Por Pares	Sim	228,93	45,10	3,18	277,22	62,56%
	Não	662,77	74,43	3,15	740,35	
Paralelos+Shap.Modif.	Sim	217,12	39,88	774,40	1031,40	31,68%
	Não	656,73	72,73	780,27	1509,73	

6.3.3 Análise de cortes eliminados

A Figura 6.19 destaca o total e o percentual de cortes eliminados nesta bateria de testes durante a execução do NEWAVE. De acordo com a ilustração, o algoritmo Shapiro Modificado obteve um percentual maior de eliminação, apesar do seu custo computacional já apresentado. Além disso, pode-se verificar que o percentual de cortes eliminados não variou em demasia quando confrontada a inclusão ou não de seleção de cortes (vide Figura 6.11).

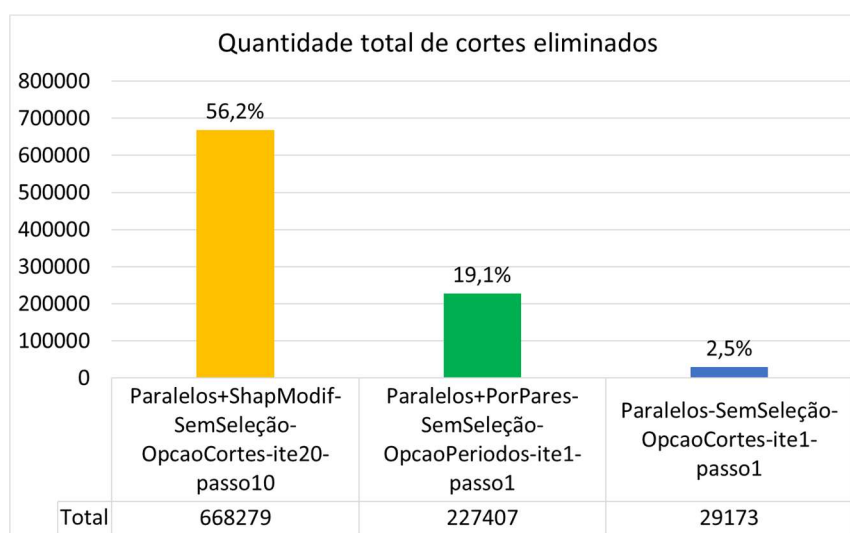


Figura 6.19 – Quantidade total de cortes eliminados – sem seleção de cortes.

A performance de eliminação de cortes por iteração, bem como o percentual de cortes eliminados (em relação a 10000 cortes), é ilustrada na Figura 6.20.

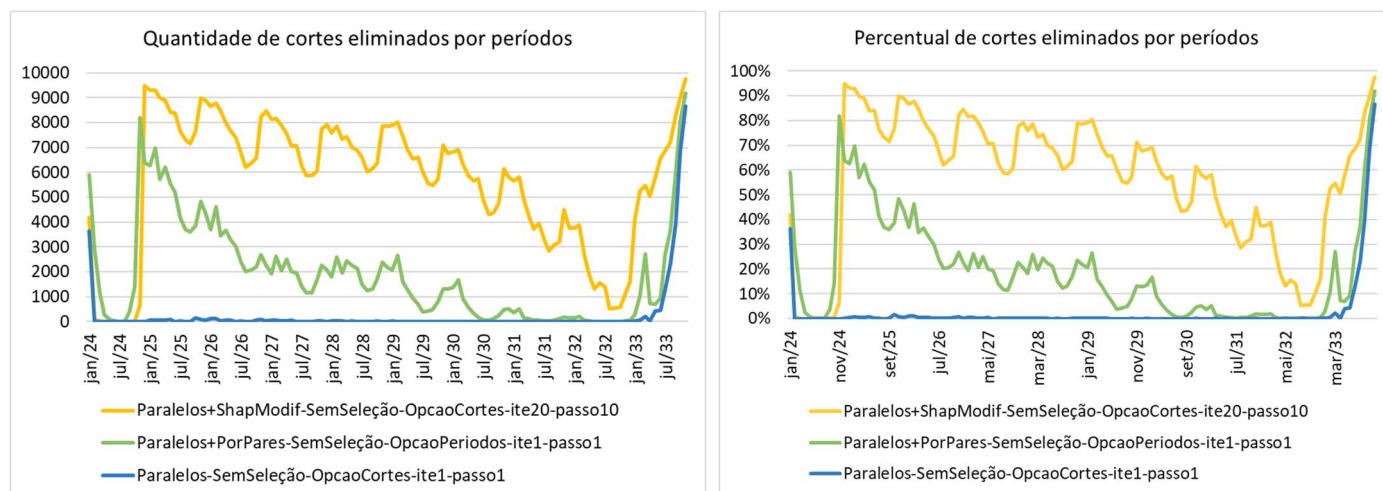


Figura 6.20 – Análise de cortes eliminados por período – sem seleção de cortes.

Os resultados ilustrados na Figura 6.20 são semelhantes àqueles destacados na Seção 6.2.3. A execução utilizando exclusivamente o algoritmo Paralelos mantém um padrão consistente de eliminação de cortes, com magnitude reduzida ao longo dos períodos. Em contrapartida, os métodos que combinam diferentes abordagens exibem variações mais acentuadas, com picos consideráveis de cortes eliminados em períodos específicos. Dentre eles, destaca-se a linha amarela, com o algoritmo de Shapiro Modificado, que apresenta o maior volume de cortes eliminados em diversos intervalos, alcançando valores próximos a 10.000 cortes eliminados.

6.4 Testes numéricos observando o parâmetro de janela de iterações

O intuito desta seção é observar o comportamento dos algoritmos ao observar a janela de iterações a ser considerada para consideração dos algoritmos de eliminação de cortes. A janela de iterações foi um parâmetro criado na expectativa de reduzir o custo computacional (por parte dos algoritmos Por Pares e Shapiro Modificado) de analisar todos os cortes. Os resultados esperados são de alteração apenas de tempo computacional, visto que o problema se mantém.

Como forma de comparação, será observadas as combinações Paralelos + Por Pares e Paralelos + Shapiro Modificado, considerando a seleção de cortes. A fim de estudar também os parâmetros dos algoritmos, foram executadas variações destes algoritmos em relação as iterações iniciais e janela de iteração:

- Paralelos + Por Pares, opção por Períodos de paralelismo, iniciando na iteração 15 com passo 1, janela de 50 iterações;
- Paralelos + Por Pares, opção por Períodos de paralelismo, iniciando na iteração 15 com passo 1, janela de 20 iterações;
- Paralelos + Shapiro Modificado, opção por Cortes de paralelismo, iniciando na iteração 20 com passo 10, janela de 50 iterações;
- Paralelos + Shapiro Modificado, opção por Cortes de paralelismo, iniciando na iteração 20 com passo 10, janela de 20 iterações.

As combinações de parâmetros utilizados foram definidas a partir dos resultados obtidos e apresentação nas Seções 6.1 e 6.2. Nestas Seções, foi identificado que o tempo de processamento nas iterações iniciais da PDDE são similares, começando a partir da 11ª iteração uma variação. Quanto a combinação Paralelos + Por Pares, motivou-se a execução desta combinação a partir da 15ª iteração visto que a Seção 6.2 já foi executado um caso começando na iteração 20, mantendo a opção de paralelismo por períodos a luz dos resultados mais significativos na redução de tempo. No caso da combinação Paralelos + Shapiro

Modificado, manteve-se a iteração inicial da execução do algoritmo Shapiro Modificado, porém alterando a janela de iterações.

Em relação ao resultado da PMO, o ZINF obtido no processo iterativo e o custo operacional são destacados na Figura 6.21. Não são evidenciadas diferenças significativas no ZINF, quando comparadas as opções de paralelismo de eliminação de cortes Por Pares. Para se ter uma noção da diferença do ZINF obtido com a combinação de algoritmos Paralelos+Por Pares, a Figura 6.21 também ilustra uma diferença relativa à execução do mesmo caso com o algoritmo Paralelos.

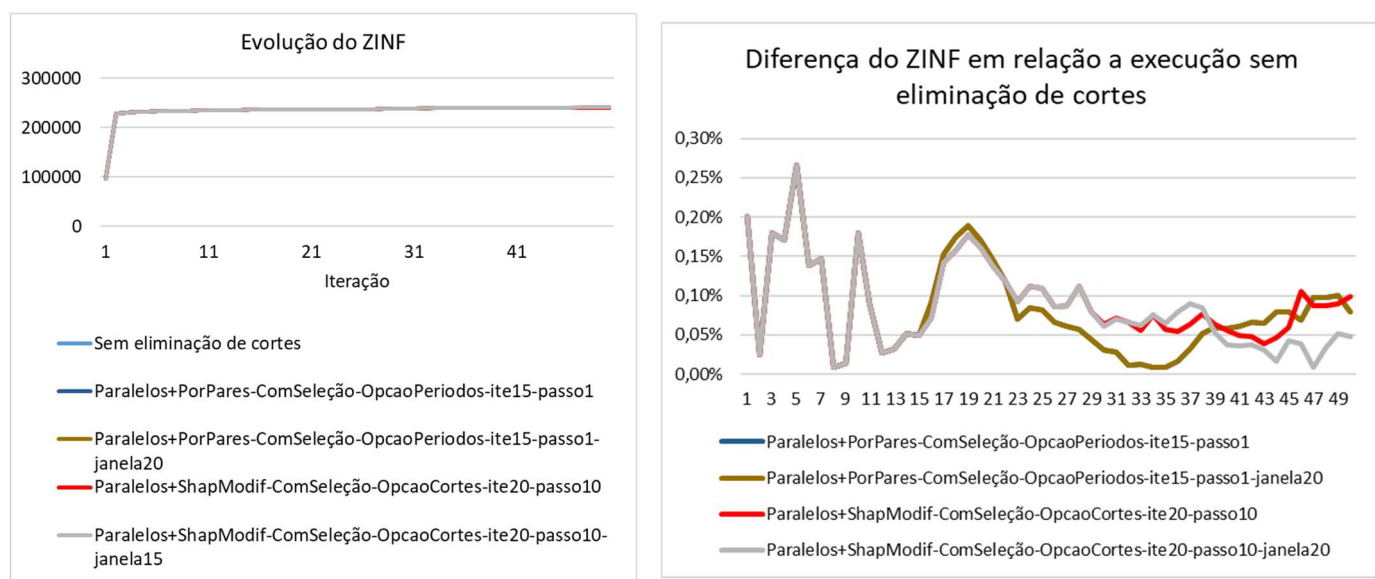
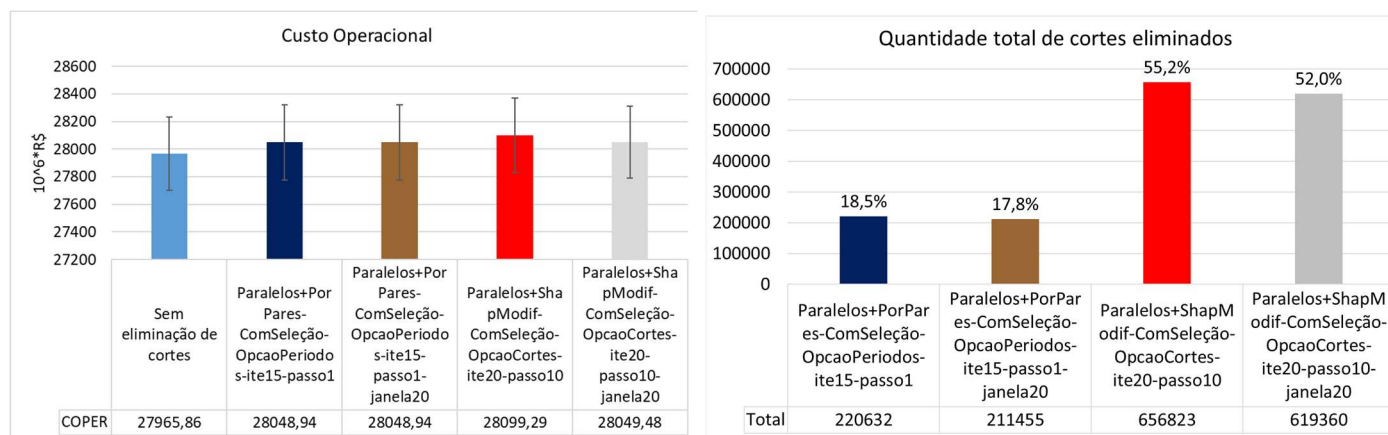


Figura 6.21 – Evolução do ZINF e a diferença dos valores entre as execuções sem eliminação de cortes.

Em relação ao custo operacional e à quantidade de cortes eliminados, conforme a Figura 6.22 ilustra, os valores obtidos estão dentro do intervalo definido no desvio padrão. Assim, não existem indícios de que a curva que define a FCF tenha sido alterada com a inclusão dos algoritmos de eliminação de cortes.



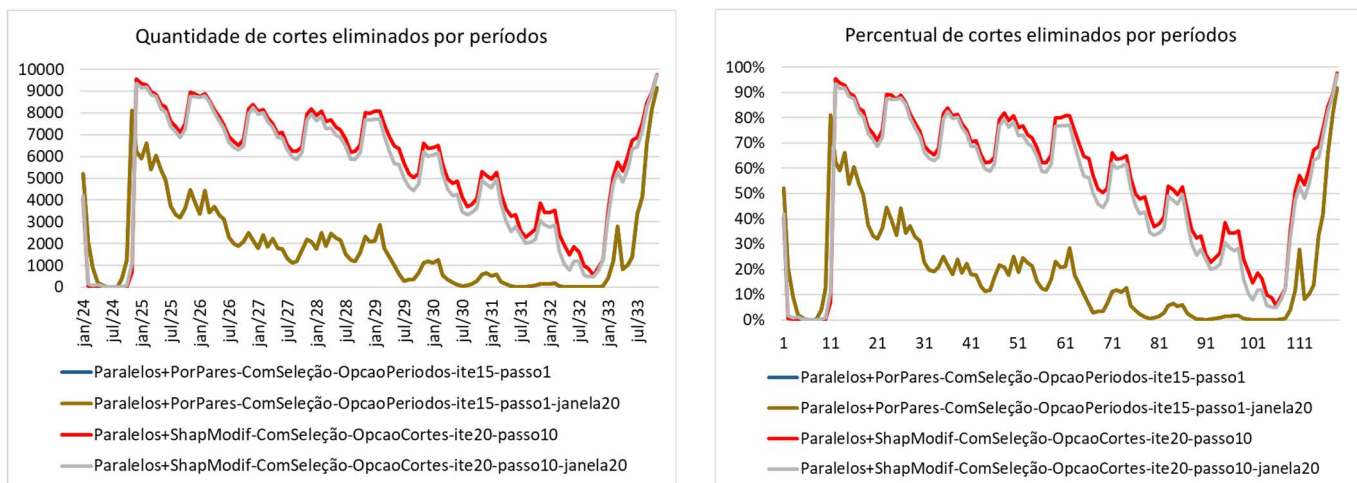


Figura 6.22 – Custo operacional e o total de cortes eliminados nos testes variando o valor da janela de iterações.

A principal diferença na alteração dos valores da janela de iteração recai no tempo de processamento, como será evidenciado a seguir na Figura 6.23e Figura 6.24. Nos valores apresentados na Figura 6.23, não se observa alterações significativas com a combinação Paralelos + Por Pares, ao contrário da combinação Paralelos + Shapiro Modificado.

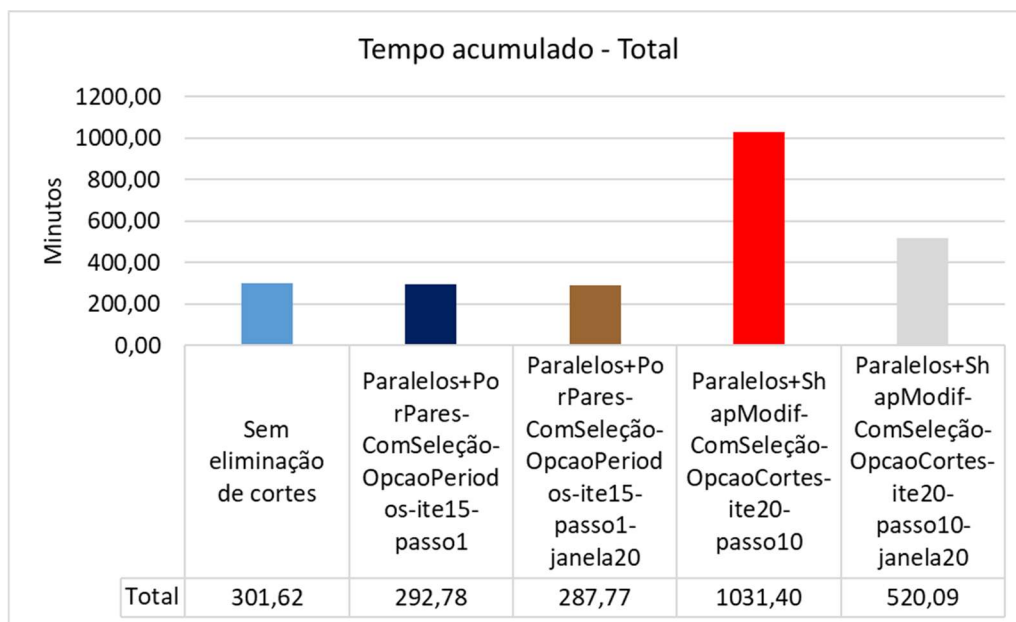


Figura 6.23 – Tempo acumulado na fase de eliminação de cortes.

Como observado na Figura 6.23, foi evidenciado o benefício de utilização da janela de iterações na execução envolvendo o algoritmo de Shapiro Modificado, reduzindo o tempo da fase de Eliminação de Cortes quando comparado com a execução com uma janela maior (50 iterações). A Figura 6.24 ilustra a execução das fases *backward* e *forward*, bem como a eliminação de cortes e o tempo total, por iteração. Observa-se que as fases do cálculo da política sofrem uma redução com a combinação Paralelos +

Shapiro Modificado, em detrimento do aumento do tempo computacional de sua execução, sendo parecidos os tempos por iteração na fase *backward* e *forward*. Quanto a combinação Paralelos + Por Pares, os tempos registrados não apresentaram diferenças significativas.

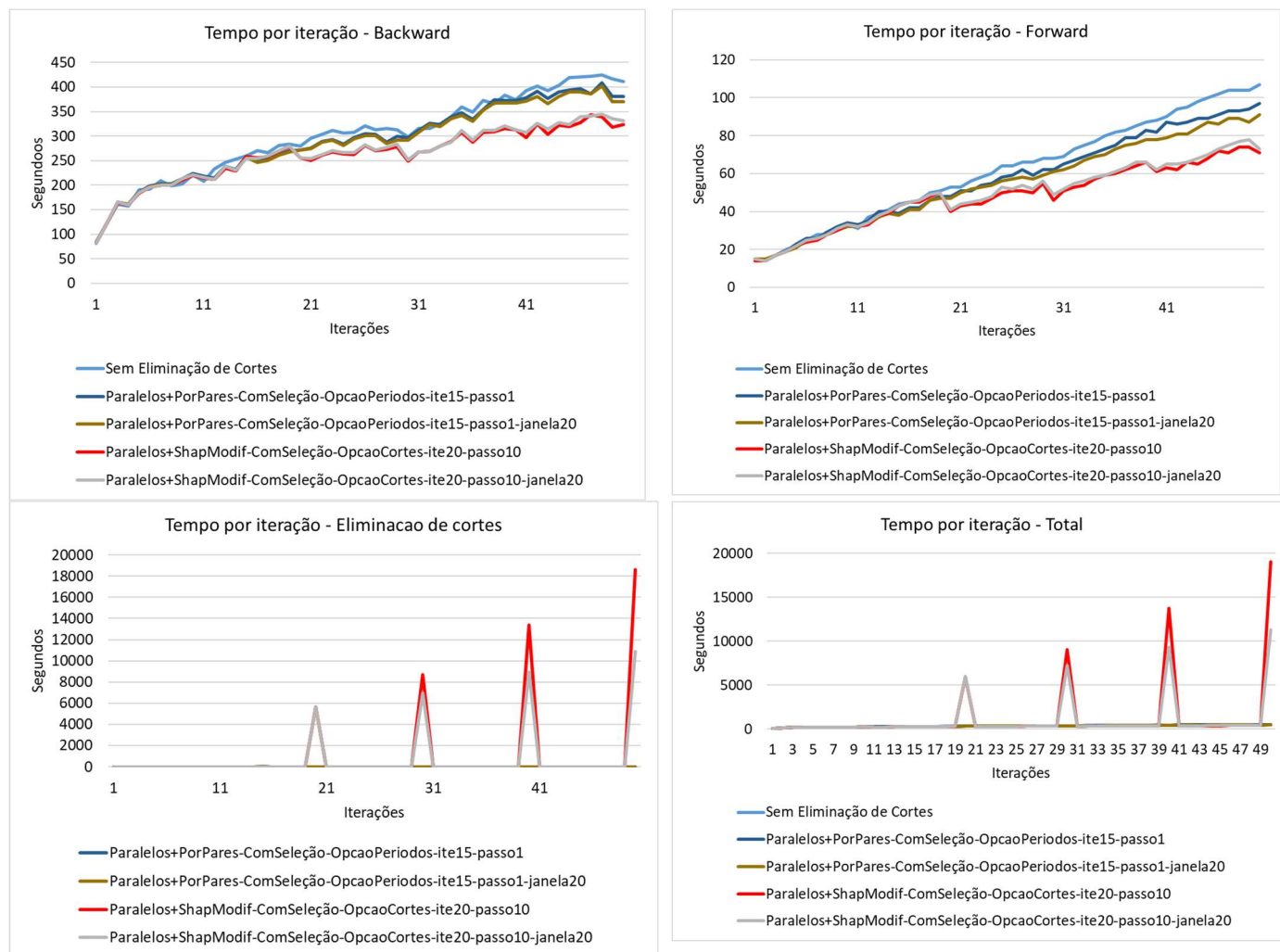


Figura 6.24 – Análise de tempo por período para o passo *forward* e *backward*.

Desta forma, o valor de janela de iteração utilizado impactou de forma significativa a execução com o algoritmo Shapiro Modificado. O parâmetro da janela de iterações implicará em quais cortes serão candidatos para análise, reduzindo o escopo de cortes que irão para teste de redundância. Assim, torna-se interessante explorar uma combinação deste parâmetro quando aplicado o algoritmo de Shapiro Modificado. Quanto à combinação Paralelos + Por Pares, não foram observadas diferenças significativas ao variar a janela, mas vale a ressalva de que esta combinação foi aplicada em todas as iterações a partir da 15ª (diferentemente do algoritmo de Shapiro Modificado).

Para ilustrar as diferenças argumentadas nessa seção, a Figura 6.25 destaca a diferença de tempo em relação a execução sem eliminação de cortes, quanto ao tempo total acumulado. Nesta figura, fica evidente o impacto da janela de iterações no algoritmo Shapiro Modificado.

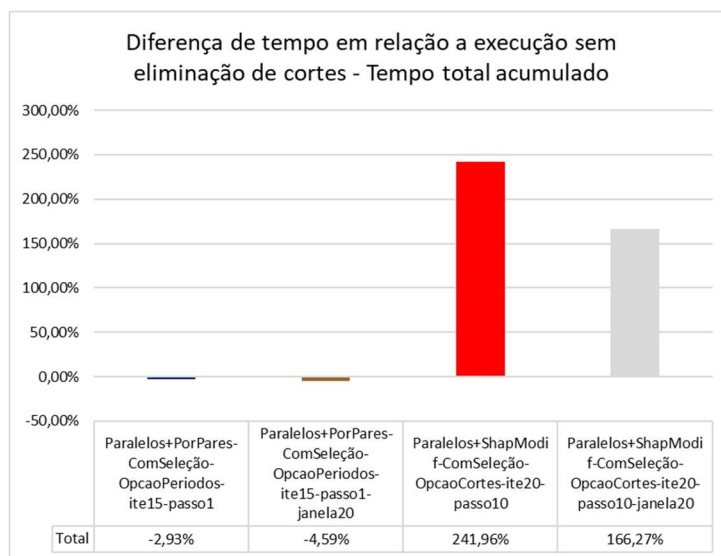


Figura 6.25 – Diferença de tempo em relação a execução sem eliminação de cortes.

7 BIBLIOGRAFIA

- [1] M. V. F. Pereira e L. M. V. G. Pinto, "Multi-stage stochastic optimization applied to energy planning," *Mathematical Programming*, vol. 52, nº 1-3, pp. 359-375, 1991.
- [2] M. E. Maceira, D. D. Penna, A. L. Diniz, R. J. Pinto, A. C. G. de Melo, C. L. V. Vasconcellos e C. B. Cruz, "Twenty Years of Application of Stochastic Dual Dynamic Programming in Official and Agent Studies in Brazil – Main Features and Improvements on the NEWAVE Model," em *20th PSCC - Power Systems Computation Conference*, Dublin, 2018.
- [3] C. B. Cruz, A. L. Diniz, T. C. Justino e R. R. Barboza, "Representação de usinas hidrelétricas individualmente e de forma agregada na programação dinâmica dual estocástica - NEWAVE Híbrido," Relatório Técnico CEPEL 1002/2022, 2022.
- [4] A. L. Diniz, M. E. Maceira, R. J. Pinto, C. L. Vasconcellos, D. D. J. Penna e C. B. Cruz, "Estratégia de seleção de cortes de Benders para redução do tempo computacional da programação dinâmica dual estocástica," Relatório Técnico CEPEL nº 11138, 2017.
- [5] A. L. Diniz, M. E. Maceira, R. J. Pinto, D. D. J. Penna e C. B. Cruz, "A. L. Diniz, M. E. Maceira, R. J. Pinto, D. D. J. Penna, C.B.Cruz, "Estratégia de seleção de cortes de Benders para redução do tempo computacional da programação dinâmica dual estocástica," em *LII Simpósio Brasileiro de Otimização Estocástica (SBPO)*, João Pessoa, Brasil, 2020.
- [6] A. P. Silva, "Análise topológica dos cortes da função de custo futuro baseada em geometria analítica: aplicação ao problema de despacho hidrotérmico," Dissertação de mestrado, Universidade Federal de Juiz de Fora, 2024.
- [7] W. Tekaya, A. Shapiro, M. Soares e J. Costa, "Worst-case-expectation approach to optimization under uncertainty," *Operational Research*, 2013.
- [8] V. de Matos, A. B. Philpott e E. C. Finardi, "Improving the Performance of Stochastic Dual Dynamic Programming," *Journal of Computational and Applied Mathematics*, vol. 290, pp. 196-208, 2015.
- [9] A. Shapiro, W. Tekaya, J. P. Costa e M. P. Soares, "Report for technical cooperation between Georgia Institute of Technology and ONS," 2011.