XXIV SNPTEE
SEMINÁRIO NACIONAL DE PRODUÇÃO E
TRANSMISSÃO DE ENERGIA ELÉTRICA

22 a 25 de outubro de 2017
Curitiba - PR

CB/GOP/16

GRUPO –IX

GRUPO DE ESTUDO DE OPERAÇÃO DE SISTEMAS ELÉTRICOS- GOP

PROGRAMAÇÃO DINÂMICA DUAL: ESTRATÉGIAS EFICIENTES APLICADAS A PROBLEMAS ESTOCÁSTICOS DE COORDENAÇÃO HIDROTÉRMICA

Lilian Chaves B. dos Santos(*)
CEPEL

André L. Diniz
CEPEL

RESUMO

O planejamento hidrotérmico de curto e médio prazo visa calcular a política operativa ótima tendo em vista os recursos disponíveis e considerando riscos, afluências incertas e custos de operação do sistema. São problemas complexos, de difícil modelagem e custosos para serem processados, sendo necessários métodos de decomposição para resolvê-los, como a programação dinâmica dual (PDD). Propõem-se três estratégias para aprimorar a eficiência computacional da PDD, sem perda de suas propriedades de convergência: teste de convergência local, teste de estabilidade das variáveis de estado e compartilhamento local de cortes. Resultados são apresentados para problemas de tamanho moderado.

PALAVRAS-CHAVE

Planejamento Hidrotérmico, Programação dinâmica dual, Árvore de cenários, Otimização Estocástica, Paralelismo

1.0 - INTRODUÇÃO

Devido ao grande potencial hídrico brasileiro, o problema de coordenação hidrotérmica é crucial para o cálculo da política operativa do setor elétrico do país. A complexidade e o tamanho do sistema elétrico exigem modelos matemáticos detalhados para representá-lo, o que tem sido aprimorado e objeto de diversos estudos no setor (1). Quanto mais complexo e mais aderente à realidade for o modelo, mais custoso é modelá-lo e resolvê-lo.

Modelos de médio prazo podem ser formulados como problemas lineares estocásticos multi-estágio, que é o caso do modelo brasileiro DECOMP, desenvolvido pelo CEPEL (2). Tendo em vista a dimensão desses problemas, a estratégia de solução requer métodos de decomposição, como a Programação Dual Estocástica (PDD) (3) Entretanto, pela natureza iterativa deste método e pelo fato de percorrer todos os cenários do problema estocástico, o tempo e recursos computacionais exigidos podem se tornar proibitivos quando o número de estágios e/ou o tamanho da árvore de cenários é muito grande.

Visto isso, diversos esforços têm sido feitos para melhorar a eficiência da PDD. Com o objetivo de diminuir o tempo de processamento global podem-se ter duas abordagens. A primeira visa acelerar o processo iterativo, onde têm sido propostas estratégias como agregação de nós (4)-(7), desagregação de cortes (*multi-cut*) (8)-(10), protocolos de caminhamento em árvore (11)-(13) e construção de cortes mais fortes (14). A segunda visa tornar as iterações mais rápidas, onde se encontram os métodos de agregação de cortes (9) (10), seleção de cortes (15), (16), compartilhamento de base (13), (17), (18) e obtenção de cortes inexatos (19). Adicionalmente, com o advento das tecnologias de hardware e software, alinhadas ao ambiente multi-processado, a paralelização dos algoritmos de programação estocástica vem ganhando espaço. Os benefícios do paralelismo dependem fortemente da estrutura do algoritmo: no caso da PDD, existe uma forte dependência intra-estágios, o que prejudica o uso destas técnicas,

(*) Av Horácio Macedo, n° 354 – Cidade Universitária – CEP 21.941-911 Rio de Janeiro, RJ, – Brasil
Tel: (+55 21) 2598-6059 – Fax: (+55 21) 2598-6482 – Email: liliancbs@cepel. br

mas, por outro lado os cenários de um mesmo estágio são totalmente independentes, e podem ser resolvidos em paralelo (2), (5), (20), (21).

O presente trabalho visa apresentar métodos para melhorar a eficiência o processo da PDD na resolução do problema de coordenação hidrotérmica, tanto no sentido de tornar as iterações mais rápidas - Teste de Convergência Local (TCL) e Teste de Estabilidade das Variáveis de estado (TEV) - de forma a reduzir a quantidade de cálculos e cortes em uma iteração, quanto no sentido de diminuir o número de iterações - Compartilhamento Local de Cortes (CLC) - trocando informações entre nós para melhor aproximar a função de custo futuro.

O trabalho está organizado da seguinte forma: na seção 2, o modelo de programação estocástica aplicado ao problema de coordenação hidrotérmica usado nos experimentos é apresentado, na seção 3 estão descritos os métodos de aceleração propostos. Na seção 4 apresenta-se o caso de estudo, e experimentos e resultados são discutidos. Na seção 5 apresentam-se as conclusões do estudo.

2.0 - O MODELO DE COORDENAÇÃO HIDROTÉRMICA

Esse trabalho considera um modelo de coordenação hidrotérmica de médio prazo, com função objetivo e restrições lineares ou lineares por partes. Algumas características da modelagem são:

a. As usinas hidráulicas e térmicas são representadas de modo individualizado.

b. O horizonte de tempo é dividido em períodos de duração variável, e no final no horizonte de estudo há um acoplamento com um modelo de longo prazo (modelo NEWAVE (22)).

c. A demanda é representada em três patamares de carga: leve, média e pesada, de duração variável, devendo a demanda ser atendida em cada um dos três patamares.

d. As incertezas consideradas são as afluências às usinas hidráulicas, sendo discretizadas em um número finito de realizações por estágio, consistindo em uma árvore de cenários. Os cenários de afluências para cada período de tempo são gerados sinteticamente por um modelo Auto Regressivo Periódico (GEVAZP (23), (24)), cujos parâmetros são calculados baseando-se no registro histórico de afluências.

Na árvore de cenários construída pelo modelo, cada nó representa um período de tempo e um cenário de afluência. Esse nó é modelado em um problema de programação linear (PL), que tem como parâmetros de entrada (variáveis de estado) os valores de armazenamento no inicio do período de tempo referente ao cenário considerado, e como condição de contorno os cortes da chamada “função de custo futuro” (FCF), que representa o custo total para os cenários de afluência futuros àquele nó. O PL de cada nó é modelado conforme a seguir:

2.1 Função Objetivo

A função objetivo consiste em minimizar o custo presente e futuro da geração energética conforme a equação:

$$\min \sum_{i=1}^{NT} \sum_{p=1}^{NPAT} ct_i gt_i^{t,c,p} + \mathbb{E}[Q(t,c)],$$

onde NT é o número de usinas térmicas e $NPAT$ é o número de patamares de carga do estudo, e utilizam-se os índices: i, para usina térmica, p para patamar de carga, t para período de tempo e c para cenário. O termo, ct é o custo incremental associado à geração térmica gt, que, para todas as térmicas e patamares, compõem o custo presente do respectivo período e cenário. O custo futuro está associado ao valor esperado de custo nos estágios posteriores, dado pela função $Q(t,c)$. Outros custos são adicionados na função objetivo para que o modelo seja mais aderente à realidade, como penalidades associadas ao vertimento e turbinamento de água, que são custos muito baixos mas importantes para evitar que o modelo deflua água sem necessidade quando o valor da água está muito baixo.

2.2 Restrição de atendimento à demanda:

A restrição de atendimento à demanda para cada patamar de carga (*p*) é representada por:

$$\sum_{i=1}^{NT} gt_i^{t,c,p} + \sum_{i=1}^{NH} gh_i^{t,c,p} = dem^{p,t} \qquad p = 1, \dots, NPAT$$

sendo, para cada patamar de carga,. NH é o número de usinas hidráulicas do estudo, gh a geração hidráulica e dem a carga do período e do patamar.

2.3 Restrição de balanço hídrico:

O balanço hídrico para cada usina hidráulica i é representada pela equação:

$$Vf_i^{t,c} + \sum_{p=1}^{NPAT} \left(Q_i^{t,c,p} + S_i^{t,c,p}\right) - \sum_{j \in \Omega^{mon}} \left(Q_j^{t,c,p} + S_j^{t,c,p}\right) = Vi_i^{t,c} + A_i^{t,c} \qquad i = 1, \dots, NH$$

onde $A_i^{t,c}$ é o volume de água afluente, $Q_i^{p,t,c}$ é o volume turbinado e $S_i^{p,t,c}$ é o volume de água vertido. Os termos $Q_j^{p,t,c}$ e $S_j^{p,t,c}$, por sua vez, são os volumes turbinados e vertidos nos aproveitamentos de montante $j \in \Omega^{mon}$. $Vi_i^{t,c}$ e $Vf_i^{t,c}$ representam os volumes armazenados nos reservatórios no instante inicial e final do período t, respectivamente. No caso de usinas à fio d'água, os volumes armazenados são sempre zero.

2.4 Restrição de engolimento máximo:

O engolimento máximo de uma usina, ou turbinamento máximo, é considerado por período e patamar, e dado por:

$$Q_i^{t,c,p} \leq Qturmax_i^{t,p}, \forall i = 1, \dots, NH; p = 1, \dots, NPAT$$

onde o valor de $Qturmax$ é definido em um pré processamento do modelo, considerando o engolimento máximo da turbina e do gerador, sendo que o primeiro depende da altura de queda da usina (25).

2.5 Função de produção das usinas hidroelétricas

A energia gerada em uma usina hidroelétrica é função da vazão que passa em suas turbinas, sua altura de queda e a eficiência associada ao gerador e à turbina, que por sua vez também dependem da vazão e da altura. Essa relação é conhecida por ser não convexa. Como uma modelagem constante (linear) desta função pode não representar bem o modelo real é utilizado um modelo aproximado linear por partes. A função de produção hidráulica aproximada (FPHA) (26) é calculada em um pré-processamento ao modelo, e os cortes são inseridos em forma de restrições envolvendo os volumes, vazões turbinadas e vertidas e geração de cada uma das usinas, tal que:

$$GH_i^{t,c,p} \leq \gamma_{0\,i,t,c}^{(k)} + \gamma_{V\,i,t,c}^{(k)}(Vi_i^{t,c} + Vf_i^{t,c})/2 + \gamma_{Q\,i,t,c}^{(k)} Q_i^{t,c,p} + \gamma_{S\,i,t,c}^{(k)} S_i^{t,c,p},$$
$$k = 1, \dots, NCUT_{FPHA}^{i,t}; i = 1, \dots, NH; p = 1, \dots, NPAT$$

onde γ são os coeficientes do modelo linear por partes, para cada uma das variáveis indicadas nos sub-índices.

2.6 Função de custo futuro

Ao final do horizonte de estudo, é necessário que haja algum artifício para indicar ao modelo o benefício associado ao armazenamento de água para o futuro, informação que é fornecida por modelos de longo prazo para o planejamento hidrotérmico. Dessa forma, é feito um procedimento de acoplamento com uma função de custo futuro (FCF) no final do horizonte de estudo, proveniente do modelo NEWAVE. O custo futuro está associado à energia armazenada no final do horizonte de estudo (calculado em função do vetor de volumes armazenados nas usinas hidroelétricas), de forma que, quanto menor a energia armazenado $Earm^r$ em cada Reservatório Equivalente de Energia (REE) r representado no NEWAVE, maior o custo na função objetivo:

$$\sum_{r=1}^{NREE} \gamma_r^{(k)} Earm^r + \alpha \geq RHS^{(k)}, \qquad k = 1, \dots, NCUT_{FCF}$$

onde, para cada corte k dentre os $NCUT_{FCF}$ cortes da FCF fornecida pelo modelo NEWAVE, os termos $\gamma_r^{(k)}$ e $RHS^{(k)}$ são os coeficientes relacionados às energias armazenadas, e α é a variável de custo futuro, presente na função objetivo.

3.0 - ESTRATÉGIAS PROPOSTAS PARA MELHOR A EFICIÊNCIA DA PROGRAMAÇÃO DINÂMICA DUAL

Nesta seção são propostas três estratégias que visam melhorar a eficiência da PDD, e que podem ser utilizadas sem perda de propriedades de convergência, para acelerar o tempo de processamento dessa estratégia, quando aplicada a problemas estocásticos multi-estágio.

3.1 Teste de Convergência Local (TCL)

O TCL é uma estratégia esperta que visa evitar cálculos e armazenamento de dados que não acrescentam informações novas ao processo da PDD. A PDD tem convergência global quando o custo associado ao modelo da função de custo futuro no nó raiz (limite inferior), que é o conjunto de cortes construídos ao longo das iterações para esse nó, é suficientemente próximo ao custo futuro real (limite superior), representado pela soma dos custos individuais dos cenários futuros a este nó.

Em (11), os conceitos de limites superior e inferior de convergência são estendidos para qualquer nó da árvore no intuito de determinar se o algoritmo deve fazer passos *backward* ou *forward*. Aplicamos esse conceito local, que chamamos de TCL (executar o teste de convergência localmente nas sub-árvores do problema), para determinar se uma sub-árvore convergiu localmente, para um dado estado (armazenamento inicial nos reservatórios). Neste caso, os cortes construídos por ela não trazem informações novas para o problema global, ou seja, a função de custo futuro construída localmente para aquela sub-árvore já representa o custo futuro real de forma satisfatória. Deste modo, em caso de convergência local, não são executados os passos *backwards* desnecessários, além da construção de cortes em todos os nós pertencentes a essa sub-árvore.

Nota-se que, no teste de convergência global, as variáveis iniciais são as condições iniciais inalteradas do problema, ou seja, quando a árvore global converge, o problema encontrou a solução ótima global, para uma dada tolerância especificada. Porém no caso de sub-árvores, a convergência garante uma solução ótima local atingida para determinada condição inicial da sub-árvore, que pode variar de uma iteração para outra, dessa forma, o TCL deve ser executado em todas as iterações. Visando isso, a próxima estratégia proposta é o Teste de Estabilidade das Variáveis de Estado.

3.2 Teste de Estabilidade das Variáveis de Estado (TEV)

Em complemento à estratégia do TCL, o TEV, por sua vez, evita operações *forward* desnecessárias em algumas sub-árvores, ao longo do processo da PDD. Conforme descrito anteriormente, dado que uma sub-árvore convergiu localmente, na iteração seguinte pode haver uma variação nas condições iniciais dessa sub-árvore tal que, para este novo cenário, a convergência local não tenha sido mais alcançada. Porém, caso as condições iniciais na iteração seguinte sejam as mesmas (ou suficientemente próximas, para uma dada tolerância), ou seja, haja estabilidade das variáveis de estado, a convergência continua garantida pelo TCL da iteração anterior, o que neste caso evita os passos *forward* e *backward* na iteração corrente.

É esperado que, quando o processo iterativo da PDD esteja perto da sua convergência global, existam várias sub-árvores do problema com convergência local atingida, evitando muitos processamentos de passos *forward* e *backward* absolutamente desnecessários.

3.3 Compartilhamento Local de Estados (CLC)

Uma estratégia mais esperta consistiria no uso de informações de um nó da árvore em outro nó. Em um mesmo período e estado ("leque" da árvore de cenários), dois PLs são diferentes apenas pelo valor da variável estocástica. Assim, os cortes construídos em um nó da árvore podem ser usados em outro nó por meio de uma operação de "deslocamento", proporcional à diferença nos valores de afluência entre os cenários. Com esse procedimento, espera-se obter iterações mais lentas, uma vez que os cenários terão mais cortes, porém o acréscimo de informações ao longo da iteração pode proporcionar uma convergência mais rápida. Ou seja, o lado direito da equação de balanço hídrico de dois cenários r e s tem afluências diferentes e volumes iniciais iguais a:

$$cenario\ r\colon\ (\dots) = Vi_i^t + A_i^{t,r}\ i = 1, \dots, NH$$
$$cenario\ s\colon\ (\dots) = Vi_i^t + A_i^{t,s}\ i = 1, \dots, NH,$$

o que pode ser transformado em afluências igual e volumes iniciais diferentes, da seguinte forma:

$$cenario\ s\colon\ (\dots) = (Vi_i^t + A_i^{t,s} - A_i^{t,r}) + A_i^{t,r}\ i = 1, \dots, NH$$

ou seja, o problema resolvido pelo cenário s, é o mesmo problema resolvido pelo cenário r para um volume alternativo: $\widetilde{Vi_i^t} = Vi_i^t + A_i^{t,s} - A_i^{t,r}$.

No entanto, algumas particularidades para a aplicação desta estratégia devem ser observadas. A principal é que a variável de estado (volumes iniciais nos reservatórios) deve estar somada à variável estocástica (afluências) na equação de balanço hídrico. Adicionalmente, as variáveis de estado e estocásticas não deve aparecer em nenhuma outra restrição do problema, já que a estratégia conta com obtenção de problemas equivalentes com diferentes dados iniciais. Por último, o futuro (cenários futuros), deve ser o mesmo para os nós, ou seja, árvore deve ser simétrica.

Para tal, é preciso modificar a equação da FPHA (seção 2.5), uma vez que o volume médio considerado nesta equação leva em conta o volume inicial. A modificação consiste em usar o volume final, o que torna a modelagem menos precisa, porém permite o uso desta estratégia. Outro artifício utilizado foi para o caso de usinas a fio d'água: uma vez que não há variáveis de estado envolvidas no balanço hídrico destas usinas, porém elas contêm as variáveis aleatórias de afluência, foi necessário criar uma variável de estado para estas usinas de forma que, somadas às afluências, permite-se o compartilhamento dos cortes.

4.0 - EXPERIMENTOS E RESULTADOS

Para avaliar a eficiência e o impacto das estratégias propostas no processo de convergência da PDD, um caso de estudo de coordenação hidrotérmica foi resolvido utilizando o método PDD puro *multi-cut*, para ser comparado com as estratégias TCL, TEV e CLC. Uma estratégia de processamento paralelo também foi implementada no algoritmo da PDD para avaliar o impacto do paralelismo no tempo de processamento total e com as estratégias propostas. Na sequencia, está descrito o caso de estudo analisado, os resultados e discussões.

4.1 Caso de estudo

O caso-estudo contém 84 usinas hidráulicas do subsistema Sudeste, sendo 44 com reservatório e 40 a fio d'água, e 46 usinas térmicas. O Horizonte de estudo é de 7 meses, com períodos mensais sendo duas aberturas (cenários) para cada mês, totalizando 64 cenários e 127 nós na árvore. O PL a ser resolvido por cada nó que não seja do tipo "folha" (ou seja, que não é do último estágio), tem dimensão de 1190 variáveis e 2400 restrições na primeira iteração, o que aumenta com a adição dos cortes à medida que o processo iterativo de PDD evolui. Para os 64 nós-folha, a dimensão é de 1192 variáveis e 2501 restrições, pois há restrições de acoplamento com o custo futuro.

4.2 Resultados: redução de cálculos

No primeiro resultado, mostrado na Figura 1, compara-se a quantidade de PLs resolvidos com as três simulações em cada uma das 27 iterações que a PDD procedeu até a convergência. No passo *Forward* do algoritmo (gráfico à esquerda), todos os 127 PLs (um para cada nó) devem ser resolvidos. O método PDD e a estratégia TCL resolverão sempre todos estes PLs a cada iteração, enquanto a estratégia TEV evita que alguns destes PLs sejam resolvidos desnecessariamente. Nota-se que, nas iterações iniciais, não há muito impacto na quantidade de PLs resolvidos, porém, à medida que o método se aproxima da sua solução ótima, essa quantidade diminui consideravelmente, sendo que, para algumas iterações, apenas metade dos PLs são resolvidos. Já no passo *Backward* (gráfico à direita), onde 63 PLs são processados, observa-se que, para as estratégias propostas, esse número é menor nos dois casos. Da mesma maneira, a diferença é mais importante nas últimas iterações.

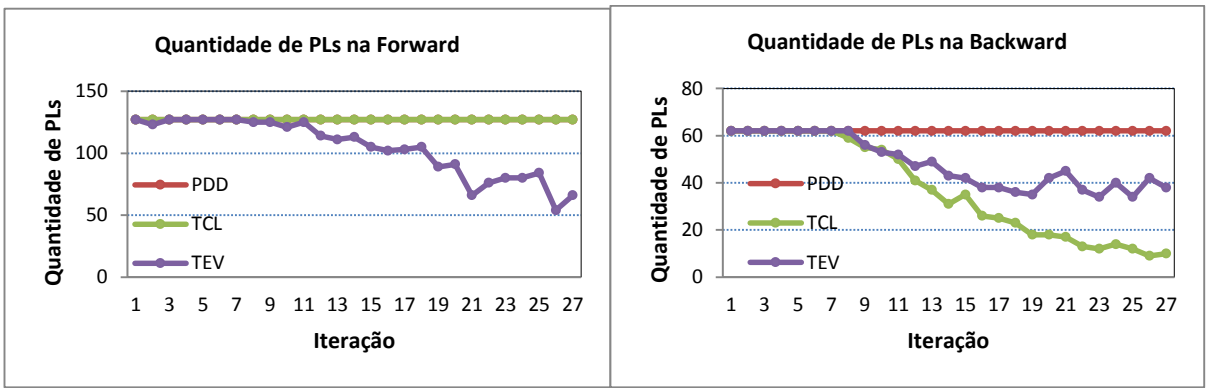

FIGURA 1 – Quantidade de PLs resolvidos por iteração no passo *Forward* e no passo *Backward* para o algoritmo PDD puro e estratégias TCL e TEV.

4.3 Resultados: redução de tempo e processamento paralelo

Para verificar o impacto no tempo de processamento das estratégias, a Figura 2 mostra o tempo gasto por iteração para as três simulações. Os valores são relativos ao tempo do caso PDD com 1 processador, que foi aproximadamente constante por iteração, com valor absoluto por volta de 100 segundos por iteração. Na Figura 3 mostra-se o tempo acumulado nas iterações, relativos ao tempo da PDD com 1 processador, que foi 2700 segundos. Foram computados os tempos utilizando 1, 8 e 16 processadores. Observam-se os seguintes aspectos:

- No caso com um processador, nas iterações iniciais há uma adição no tempo de cálculo com a estratégia TCL, de quase 10%, devido às comparações e testes que são necessárias para esta estratégia. Porém, esse tempo é compensado, uma vez que nas ultimas iterações as reduções chegam a ser de até 20%, e no tempo total houve uma redução de cerca de 5%. A estratégia TEV reduz o tempo total em 22%, uma vez que evita os PLs no passo

forward que estão em número mais significativo que a quantidade de PLs no passo *backward*. Observa-se uma redução importante de cerca de 60% nas ultimas iterações do algoritmo;

- Para o caso com mais processadores, observamos que o TLC melhora seu desempenho em relação ao método de PDD puro, e chega ser 10% mais rápido no tempo global. Isto ocorreu porque se mantiveram os ganhos de 20% de tempo nas ultimas iterações e não foi observada sobrecarga de cálculos nas iterações iniciais, uma vez que agora estão divididos entre processadores. Já o tempo total do TEV é cerca de apenas 15% melhor que o de PDD pura, uma vez que, no processamento paralelo, evitar a solução de alguns de PLs em alguns processadores pode acarretar em tempo ocioso, e não necessariamente em economia de tempo.

- Comparando o tempo de solução da PDD com 1, 8 e 16 processadores, nota-se que, ao aumentar o número de processadores em 8 vezes (1 para 8 processadores), tem-se uma diminuição de 1.8 vezes no tempo total de processamento, e ao aumentar o número de processadores em 16 vezes (1 para 16 processadores), a diminuição no tempo é de 2.7 vezes. Ressalta-se que a estrutura em árvore de cenários com dependência temporal imposta pela PDD restringe a paralelização do código, e a torna muito dependente da estrutura da árvore. Árvores com muitas aberturas (onde não há dependência temporal) tendem a ter melhor desempenho em paralelo, enquanto que, árvores mais profundas (muitos estágios temporais) e com menos aberturas, limitam o paralelismo principalmente nos primeiros estágios, que é exatamente o caso da arvore em estudo. Outras técnicas para paralelizar a PDD, de forma assíncrona, como por exemplo em (21) podem ser exploradas no intuito de melhorar essa eficiência, porém esse não foi o foco deste trabalho.

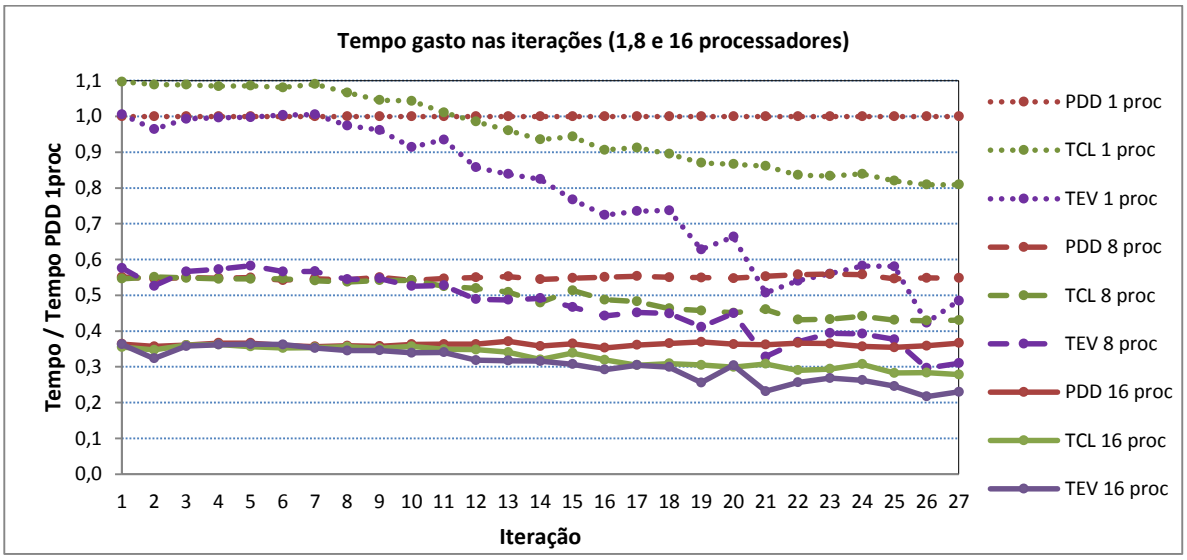

FIGURA 2 – Tempo de processamento gasto a cada iteração para os três algoritmos (PDD, TCL e TEV).

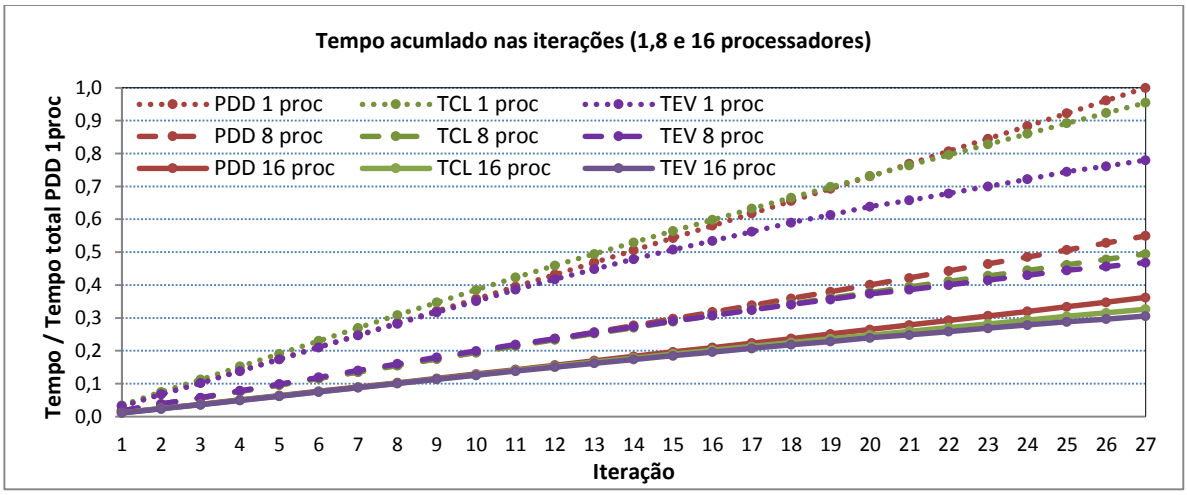

FIGURA 3 – Tempo de processamento acumulado a cada iteração para os três algoritmos (PDD, TCL e TEV).

4.4 Resultados: Compartilhamento de Cortes

Para os experimentos de compartilhamento de cortes, o caso foi alterado para atender aos requisitos necessários para aplicação dessa estratégia: a árvore de cenários foi simétrica e as usinas fio d'água também possuíam variáveis de estado na equação de balanço hídrico e função de custo futuro. Como a inserção destas podem acarretar em tempo e dificuldades maiores para convergência, são feitas três simulações: PDD pura sem as variáveis extra de estado, PDD_VE com as variáveis extra de estado, e CLC com as variáveis extra de estado.

Na Figura 4 vemos o desempenho em tempo e quantidade de iterações realizadas para cada uma das simulações, usando 1, 8 e 16 processadores. O tempo mostrado é relativo ao tempo total da PDD_VE com 1 processador que foi 3448 segundos. Pode-se observar que:

- A PDD converge com 42 iterações, porém ao adicionar as variáveis de estado para as usinas fio d'água (PDD_VE), o número de iterações aumenta para 59. Ao adicionar o CLC, o número de iterações decresce para 49. Portanto, o CLC permite uma diminuição de 20% das iterações, mas que não compensa o aumento das iterações causado pelo acréscimo das variáveis de estado.

- Com relação aos tempos de processamento, o caso sem as variáveis de estado extra realiza menos iterações, sendo mais rápido que os outros dois, considerando qualquer quantidade de processadores.

- Comparando o CLC e a PDD_VE, é possível observar um aumento no tempo de cada iteração, causado pelos cortes a mais que são inseridos no PL. Porém, a convergência mais rápida do CLC compensa esse aumento, sendo o tempo global menor principalmente com um maior número de processadores.

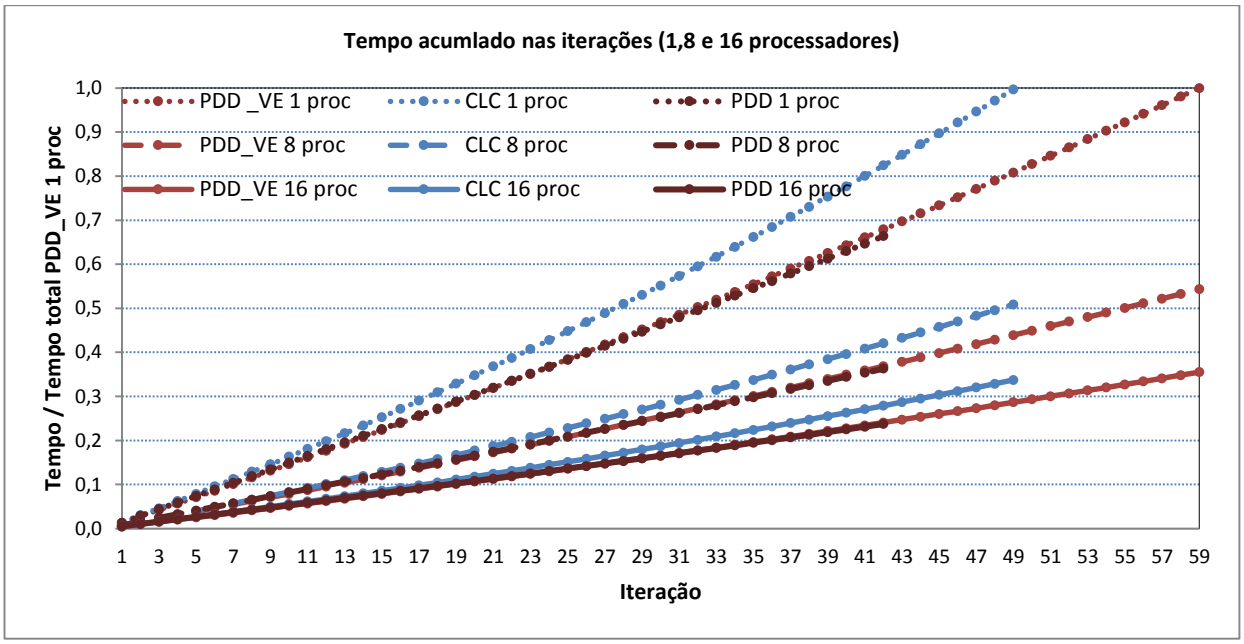

FIGURA 4 – Tempo de processamento acumulado a cada iteração, para os algoritmos PDD, PDD_VE e CLC.

5.0 - CONCLUSÕES

Ressaltando a importância de se obter algoritmos mais eficientes para resolver problemas de coordenação hidrotérmica e permitir soluções mais rápidas com um aumento da complexidade dos modelos, este artigo propõe alguns aprimoramentos na eficiência do algoritmo de programação dinâmica dual (PDD), sem perda de suas propriedades de convergência: o teste de convergência local (TCL), o teste de estabilidade das variáveis de estado (TEV) e o compartilhamento de cortes entre estados (CLC). As duas primeiras técnicas são complementares, têm baixo custo de implementação e os resultados mostraram que são capazes de evitar consideravelmente a quantidade de operações executadas no processo de solução da PDD, principalmente nas iterações finais.

O CLC mostrou cumprir seu objetivo de melhorar a taxa de convergência do algoritmo, embora o custo adicional do método seja grande limitando muito a melhora no tempo total. Além disso, as exigências para aplicação desse método, como não permitir que o armazenamento seja variável de estado em outras restrições do problema além do balanço hídrico, faz com que não compense o uso do mesmo para o caso em questão. Com relação ao processamento paralelo, a forte dependência temporal do algoritmo limita a utilização deste recurso na arvore de cenários, mesmo que, ainda haja uma redução de tempo importante ao aplicar essas técnicas, o que facilita muito os estudos que envolvem execução dos casos de grandes dimensões.

6.0 - REFERÊNCIAS BIBLIOGRÁFICAS

(1) Fortunato, L. A. M., Neto, T. A. A., Albuquerque, J. C. R., Pereira, M. V. F., “Introdução ao planejamento da expansão e operação de sistemas de produção de energia elétrica”, Universidade Federal Fluminense (1990).

(2) Pinto, R. J., Sabóia, A.L.G.P., Cabral R.N., Costa, F.S., Diniz, A.L., e Maceira. M. E. P., “Metodologia para aplicação de processamento paralelo no planejamento de curto-prazo (modelo DECOMP)”, XX SNPTEE- Seminário Nacional de Produção e Transmissão de Energia Elétrica, Recife, Novembro (2009).

(3) Birge, J. R. Decomposition and partitioning methods for multistage stochastic linear programs. Operations Research, 33(5):989-1007 (1985).

(4) Birge, J. R. and Louveaux, F. V. Introduction to Stochastic Programming. Springer-Verlag New York (1997).

(5) Dempster, M. A. H. and Thompson, R. T. Parallelization and aggregation of nested benders decomposition. Annals of Operations Research, 81(0):163-188 (1998).

(6) Dos Santos, T.N. and Diniz, A.L. A new multiperiod stage definition for the multistage benders decomposition approach applied to hydrothermal scheduling. IEEE Transactions on Power Systems, 24(3):1383-1392 (2009).

(7) Latorre, J. M., Cerisola, S., Ramos, A., and Palacios, R. Analysis of stochastic problem decomposition algorithms in computational grids. Annals of Operations Research, 166(1):355-373 (2008).

(8) Birge, J. R. and Louveaux, F. V. A multicut algorithm for two-stage stochastic linear programs. European Journal of Operational Research, 34(3):384 – 392 (1988).

(9) Trukhanov, S., Ntaimo, L., and Schaefer, A. Adaptive multicut aggregation for two-stage stochastic linear programs with recourse. European Journal of Operational Research, 206(2):395 – 406 (2010).

(10) Wolf, C. and Koberstein, A. Dynamic sequencing and cut consolidation for the parallel hybrid-cut nested l-shaped method. European Journal of Operational Research, 230(1):143 – 156 (2013).

(11) Morton, D. P. An enhanced decomposition algorithm for multistage stochastic hydroelectric scheduling. Annals of Operations Research, 64(1):211-235 (1996).

(12) Gassmann, H. I. Mslip: A computer code for the multistage stochastic linear programming problem. Mathematical Programming, 47(1):407-423 (1990).

(13) Altenstedt, F. Aspects on asset liability management via stochastic programming. PhD thesis, Chalmers University of Technology and Gotebourg University, (2003).

(14) Magnanti, T. L. and Wong, R. T. Accelerating benders decomposition: Algorithmic enhancement and model selection criteria. Operations Research, 29:464484 (1981).

(15) Shapiro, A., Tekaya, W., Soares, M. P., and da Costa, J. P. Worstcase-expectation approach to optimization under uncertainty. Operations Research, 61(6):1435-1449 (2013).

(16) de Matos, V. L., Philpott, A. B., and Finardi, E. C. Improving the performance of stochastic dual dynamic programming. Journal of Computational and Applied Mathematics, 290:196 – 208 (2015).

(17) Walkup, D. W. and Wets, R. J.-B. Lifting projections of convex polyhedra. Pacic J. Math., 28(2):465-475 (1969).

(18) Wets, R. J.-B. Large scale linear programming techniques in stochastic programming. IIASA Working Paper (1984).

(19) Zakeri, G., Philpott, A. B., and Ryan, D. M. Inexact cuts in benders decomposition. SIAM Journal on Optimization, 10(3):643-657 (1999).

(20) Linderoth, J. and Wright, S. Decomposition algorithms for stochastic programming on a computational grid. Computational Optimization and Applications, 24(2):207-250 (2003).

(21) Santos, T. N.,. Diniz, A.L., Borges, C. T., “A New Nested Benders Decomposition Strategy for Parallel Processing Applied to the Hydrothermal Scheduling Problem”, IEEE Transactions on Smart Grid, (2016).

(22) Maceira, M.E.P., Duarte, V.S., Penna, D.D.J., Moraes, L.A.M., Melo, A.C.G., “Ten years of application of stochastic dual dynamic Programming in official and agent studies in Brazil –Description of the NEWAVE program”, *16th Power Systems Computation Conference - PSCC*, Glasgow, SCO (2008).

(23) Maceira, M.E.P., Bezerra, C.V., 1997, “Stochastic Streamflow model for Hydroelectric Systems” In: Proceedings of 5th Int. Conf.Probab. Methods Applied to Power Systems, pp. 305-310, Vancouver, Canada (1997).

(24) Maceira, M. E. P.; Penna, D. D. J.; Damazio, J. M., “Geração de Cenários Sintéticos de Energia e vazão para o Planejamento da Operação Energética”, Cadernos do IME – Série Estatística, UERJ (2006).

(25) CEPEL, Centro de Pesquisas de Energia Elétrica, “Consideração do engolimento máximo das turbinas em função da altura de queda na representação da vazão turbinada máxima no modelo DECOMP”, NT nº 1(2015).

(26) Diniz, A.L., Maceira, M.E.P., “A four-dimensional model of hydro generation for the short-term hydrothermal dispatch problem considering head and spillage effects”, *IEEE Trans. Power Syst.,* v. 23, n.3, pp. 1298-1308 (2008).

7.0 - DADOS BIOGRÁFICOS

Lílian Chaves B. dos Santos: possui graduação em Engenharia Elétrica (2014) pela UFMG, MG, Brasil. Diploma de engenheiro nível master (2013) pela *Ecole Supérieure d'Ingé*nieurs en Electronique et Electrotechnique, Paris, França. Mestrado em Engenharia de Sistemas e Computação (em curso) pela UFRJ / COPPE, RJ, Brasil. É pesquisadora no CEPEL - Centro de Pesquisas de Energia Elétrica, do grupo ELETROBRÁS, desde 2014.

André L. Diniz: possui graduação em Engenharia Civil (1997), mestrado em Engenharia de Transportes (2000) e doutorado em Otimização pelo Programa de Engenharia de Sistemas e Computação (2007), todos pela UFRJ / COPPE. Em 2014 realizou um pós-doutorado no Weierstrass Institute for Applied Analysis and Stochastics. É pesquisador e chefe do Departamento de Otimização Energética e Meio Ambiente (DEA) do CEPEL - Centro de Pesquisas de Energia Elétrica, do grupo ELETROBRÁS, e professor adjunto da UERJ - Universidade do Estado do Rio de Janeiro, no departamento de Estatística. Tem grande experiência na área de programação matemática e otimização, especialmente aplicada em problemas de planejamento e operação de Sistemas de Energia Elétrica