



## GRUPO DE ESTUDO DE ANÁLISE E TÉCNICAS DE SISTEMAS DE POTÊNCIA - GAT

### PARALELIZAÇÃO DA SOLUÇÃO DE MODELOS E OUTROS RECURSOS PARA A MELHORIA DE DESEMPENHO NA SIMULAÇÃO DE CASOS ANATEM

**NÍCOLAS ABREU ROCHA LEITE NETTO(1); LÍGIA ROLIM DA SILVA(1); FABRICIO LUCAS LIRIO(1)**  
**CENTRO DE PESQUISAS DE ENERGIA ELÉTRICA CEPEL(1)**

#### RESUMO

Foram implementadas no Anatem diversas melhorias a fim de garantir ao usuário uma simulação mais célere, tais como: um algoritmo de solução paralelizada de modelos CA e CC, um recurso para agrupamento de solução de Controladores Definidos pelo Usuário (CDUs), melhorias no algoritmo DLCA para inclusão de DELAY automático em controladores com problema de convergência e um recurso para depuração do banco de dados. Este trabalho tem como objetivo mostrar como o usuário deverá empregar os recursos supracitados de forma a obter a melhoria no desempenho computacional, mantendo a confiabilidade nos resultados que ele encontra nas versões anteriores do programa.

#### PALAVRAS-CHAVE

Transitórios Eletromecânicos, Computação Paralela, Anatem

#### 1.0 INTRODUÇÃO

O Anatem é um programa computacional desenvolvido pelo Centro de Pesquisas de Energia Elétrica, o Cepel, para a simulação de transitórios eletromecânicos em sistemas de potência de grande porte. Programa mais utilizado no Brasil para este fim, o Anatem conta com sua base de dados dinâmicos disponibilizada por agentes setoriais, o Operador Nacional do Sistema (ONS) e a Empresa de Pesquisa Energética (EPE).

A necessidade de modelagem detalhada para a correta avaliação do problema de estabilidade eletromecânica e a expansão do Sistema Interligado Nacional (SIN) implica na existência de um banco de dados crescente, em função de novos empreendimentos participantes do SIN. É também exigido dos modelos ingressantes que a modelagem deles seja a mais fiel possível ao comportamento real, com poucas simplificações concedidas aos modelos.

No caso dos modelos de geradores eólicos, por exemplo, o detalhamento exigido é elevado em função das diversas configurações de controle, e da complexidade inerente dos seus controladores. O Plano Decenal de Expansão de Energia, elaborado pela EPE, segue sinalizando para o ingresso crescente desses empreendimentos no sistema.

Naturalmente, com a expansão do banco de dados, o problema de estabilidade eletromecânica torna-se, cada vez mais, um problema computacional demandante, requerendo maior esforço computacional para a solução dos problemas de estabilidade.

Neste trabalho são apresentados três recursos para melhoria do desempenho computacional (paralelização de modelos, CDU *Clustering* e arquivo DLCA cache) e um recurso para depuração de banco de dados (opção ESTC), disponíveis desde a versão 12.0 do programa [1]. Juntos, estes recursos apresentam maior qualidade para o usuário realizar suas análises, em função do decréscimo considerável do tempo de solução, especialmente de casos problemáticos para convergência, além de permitir identificar controladores com potenciais problemas de convergência, para eventual revisão de sua modelagem.

#### 2.0 RECURSOS DO ANATEM 12 PARA GANHO DE DESEMPENHO COMPUTACIONAL E IDENTIFICAÇÃO DE CONTROLADORES COM PROBLEMAS DE CONVERGÊNCIA

Nesta seção, serão apresentados os recursos desenvolvidos na versão 12 do Anatem para a melhoria do desempenho computacional (paralelização de modelos, CDU *Clustering* e arquivo DLCA cache) e como o usuário pode melhor aproveitar esses recursos. Em seguida, será apresentado o recurso de avaliação de convergência de CDUs para identificação de controles candidatos a serem revistos.

## 2.1 Algoritmo SPMD para paralelização da solução dos modelos

O algoritmo SPMD (*Single Program Multiple Data*) de paralelização do Anatem consiste na implementação de paralelismo em que parte das tarefas típicas do Anatem são distribuídas para mais de uma unidade de processamento (núcleos de um processador multi-core) e só deve ser utilizado para solução de casos individuais.

### 2.1.1. Lei de Amdahl para estimativa de ganho computacional

Segundo a lei de Amdahl [2], existe um limite factível de ganho de desempenho com a divisão de tarefas seriais em várias unidades de processamento. Existem algumas considerações acerca de como essa lei pode ser aplicada; grosseiramente, este limite factível pode ser estimado com a seguinte equação:

$$S_{max} = \frac{1}{1 - p + \frac{p}{N}}$$

Sendo  $S_{max}$  o *speedup* máximo possível de ser alcançado dado que o percentual  $p$  do código poderia ser perfeitamente paralelizável e  $N$  o número de processos empregados. Esta relação é importante para criar uma expectativa real acerca da escalabilidade de um algoritmo com o número de *cores* empregados. A imagem a seguir ilustra combinações de um algoritmo hipotético, com diferentes percentuais de região paralelizável, e o *speedup* máximo possível para esses algoritmos:

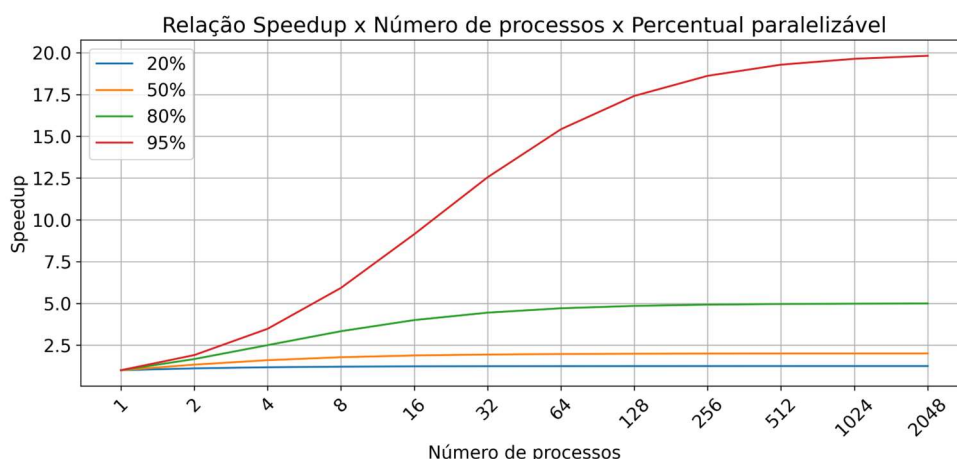


FIGURA 1 – Representação gráfica da Lei de Amdahl

### 2.1.2. Modelos paralelizáveis e modelos seriais do Anatem

Devido à natureza matemática dos modelos a que se propõe resolver, o Anatem possui diversas regiões de códigos distintas a serem resolvidas. Para cada passo de simulação, o Anatem deve resolver: os modelos CA (máquinas síncronas, máquinas de indução, parques eólicos etc.); a rede CA; os modelos CC (conversores e elos); a rede CC; as proteções (Relés e ERAC); a impressão de relatórios e arquivos de plotagem.

Algumas dessas etapas são paralelizáveis, outras são inerentemente seriais, outras ainda são paralelizáveis, mas não se justifica a implementação. A solução de modelos CA e modelos CC são paralelizáveis. A solução de rede é paralelizável, mas requer um método de solução diferente do método existente. As soluções da rede CC e das proteções são paralelizáveis, mas são pouco expressivas no tempo de execução e, portanto, a paralelização destas regiões não produz ganhos significativos. A impressão de relatórios e arquivos de plotagem (IO) são processos seriais e que não podem ser paralelizados.

Desta forma, na versão 12.0 do Anatem, foram paralelizadas as soluções de modelos CA e modelos CC, haja visto que estes representam a maior parte do esforço computacional do programa, com calendário estático de distribuição de carga entre os *cores*. É possível estimar o tempo necessário para resolver cada parcela em uma simulação por meio da opção ESTS junto do código EXSI. Essa opção realiza uma estatística detalhada da simulação (ver Tabela 1), apontando quanto tempo o programa gastou solucionando em função do tipo de modelo a ser resolvido.

### 2.1.3. Tamanho da região paralelizável de casos do Anatem

No caso do programa Anatem, o tamanho da região paralelizável depende do caso analisado em questão. Em casos com grande número de modelos de fontes eólicas, por exemplo, a região paralelizável é maior, e, portanto, o caso tem maior ganho em função do número de *cores* disponíveis em uma máquina. Para uma simulação de curto-circuito

em Colinas (TO) no BD2019 do ONS [3] e considerando-se a resolução sequencial do Anatem, têm-se os seguintes tempos computacionais para o caso com modelos de eólicas e solares e o caso sem esses modelos:

TABELA 1 – Relatórios ESTC para duas configurações do BD2019			
EXSI ESTS - Caso <b>sem</b> modelos de eólica e fotovoltaica		EXSI ESTS - Caso <b>com</b> modelos de eólica e fotovoltaica	
- Tempo total de simulação	00:03:30.91	- Tempo total de simulação	00:05:22.24
0.02% Inicialização	00:00:00.05	0.08% Inicialização	00:00:00.24
- 80.73% Laço modelo CA - rede CA	00:02:50.26	- 88.66% Laço modelo CA - rede CA	00:04:45.71
- 56.10% Modelos CA	00:01:58.33	- 77.12% Modelos CA	00:04:08.50
49.55% Modelos de máquina síncrona	00:01:44.50	27.70% Modelos de máquina síncrona	00:01:29.24
2.87% Eólicas, fotovoltaicas e CNE	00:00:06.06	47.13% Eólicas, fotovoltaicas e CNE	00:02:31.89
-> 3.68% Outros modelos CA	00:00:07.77	-> 2.29% Outros modelos CA	00:00:07.36
-> 24.62% Rede CA	00:00:51.92	-> 11.55% Rede CA	00:00:37.20
- 7.35% Laço modelo CC - rede CC	00:00:15.49	- 4.71% Laço modelo CC - rede CC	00:00:15.16
- 6.93% Modelos CC	00:00:14.62	- 4.44% Modelos CC	00:00:14.32
0.04% Modelos de elo CC	00:00:00.07	0.03% Modelos de elo CC	00:00:00.08
-> 6.90% Modelos de conversores	00:00:14.54	-> 4.42% Modelos de conversores	00:00:14.23
-> 0.41% Rede CC	00:00:00.86	-> 0.26% Rede CC	00:00:00.84
1.03% Controle de proteções (Relés+ERAC)	00:00:02.17	0.76% Controle de proteções (Relés+ERAC)	00:00:02.44
-> 10.87% Processamento de I/O	00:00:22.93	-> 5.79% Processamento de I/O	00:00:18.67

Em destaque, tabela anterior, estão os trechos da região paralelizável do caso. No cenário sem modelos dinâmicos de eólicas, o percentual da região paralelizável do caso é de aproximadamente 60%; no cenário com eólicas e fotovoltaicas, por outro lado, o percentual é aproximadamente de 80%. Segundo a lei de Amdahl, os *speedups* máximos para esses casos seriam de 2,0 e 3,0 para um computador com 6 núcleos físicos<sup>1</sup>; para um computador com 1024 núcleos físicos, esses números seriam de 2,5 e 5,0, respectivamente<sup>2</sup>.

Note que o tamanho da região paralelizável é diretamente dependente do processo de convergência: em um caso em que a rede CA apresenta problemas para convergir, o algoritmo paralelo terá sua eficácia limitada; em casos que o problema está em um modelo CA, contudo, o algoritmo SPMD consegue reduzir o efeito deste impacto na solução.

#### 2.1.4. Como utilizar o algoritmo SPMD no Anatem

Atualmente, o algoritmo SPMD no Anatem consta com 3 formas para ser ativado, a critério do usuário. A primeira forma consiste no uso de um novo botão da interface do Anatem (em destaque na Figura 2). Alternativamente, o usuário pode pressionar a tecla F5 que foi atualizada da simulação tradicional para o algoritmo SPMD.

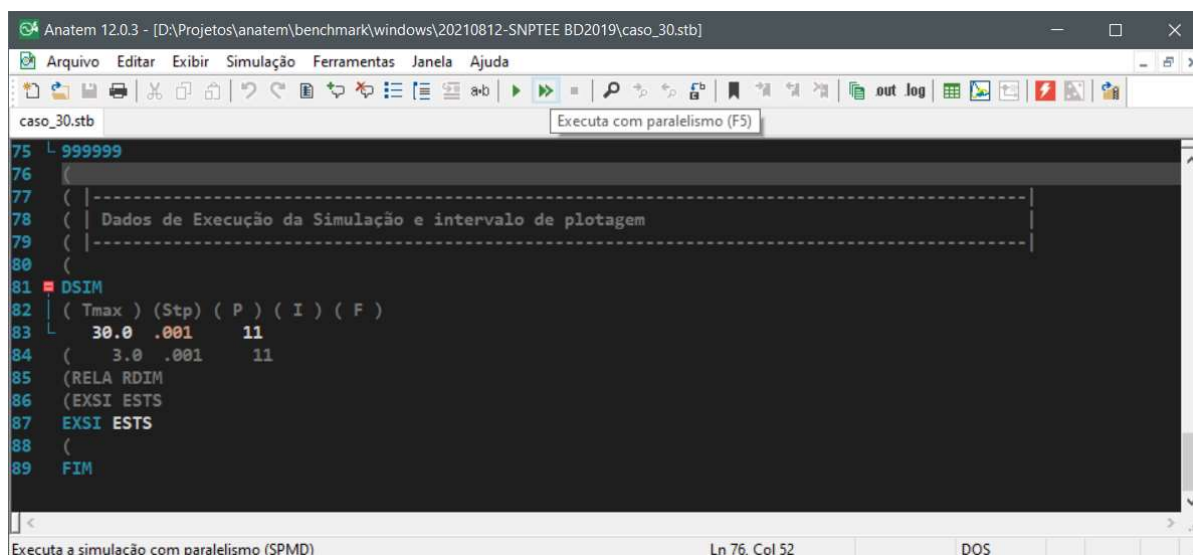


FIGURA 2 – Interface do Anatem 12.0 com destaque para o botão do algoritmo SPMD

Quando executado desta forma, o Anatem determinará automaticamente o número de processos em paralelo a serem criados (número de núcleos totais divididos por dois). Não é recomendado a utilização do computador em

<sup>1</sup> Em computadores da arquitetura x86\_64, como os da AMD e Intel, é comum o emprego da tecnologia de *núcleos virtuais*. Na prática, essas estruturas podem trazer ganhos computacionais, mas dois núcleos compartilham partes importantes da microarquitetura, prejudicando *severamente* o ganho de escala esperado.

<sup>2</sup> Em função dessa constatação, para a simulação de múltiplos casos é preferível realizá-las simultaneamente e de forma sequencial. Ou seja, neste caso é mais rápido a simulação de 4 Anatems seriais (N=1) por meio do recurso Lote da interface.

outras atividades intensivas (como outra simulação Anatem), mas atividades como navegação web ou mesmo a manipulação de casos no Anarede são atividades que não devem impactar na execução do Anatem.

Caso o usuário esteja executando o Anatem pelo *shell* do sistema operacional (CMD ou Powershell no Windows), o algoritmo SPMD pode ser evocado ao passar como parâmetro adicional na chamada do programa, o parâmetro *-n* seguido do número de processos desejados, como destacado na Figura 3.

```

nnetto@LAPTOP-876GU69C: .../20210812-SNPTEE BD2019
nnetto@LAPTOP-876GU69C: .../20210812-SNPTEE BD2019$ ./linux/anatem -i caso_30.stb -n 6
CEPEL - Centro de Pesquisas de Energia Elétrica
Análise de Transitórios Eletromecânicos - ANATEM - Verso 12.1.18-dev

Tipo de licença: / Usuário:
Empresa:

X-----X-----X-----X-----X
| UNIDADE      NOME      IDENTIFICAÇÃO      |
| LOGICA      LOGICO      DO ARQUIVO      |
X-----X-----X-----X-----X

.stb 1  TEM$DADOS /mnt/d/Projetos/anatem/benchmark/windows/20210812-SNPTEE BD2019/caso_30
      2  TEM$SAVCA /dev/null
      3  TEM$MODEL /dev/null

```

FIGURA 3 – Uso do algoritmo SPMD no modo *shell*

Por fim, o usuário pode ativar o algoritmo em apenas uma parte do seu caso Anatem; as duas formas anteriores mostram um uso do algoritmo para todos os códigos EXSI do caso. Para a simulação de alguns códigos EXSI em especial, é permitido usar a opção SPMD junto do código EXSI. Se o algoritmo for ativado por esta forma, sendo chamado sem o parâmetro *-n* ou pelo novo botão (F5) da interface, o Anatem só abrirá a região paralela nos trechos EXSI em que esta opção estiver habilitada. Neste caso, o Anatem seguirá a mesma filosofia do modo pela interface e limitará o número de processadores ativos pela metade do número máximo.

## 2.2 CDU Clustering para agrupamento de solução de controles com comunicação

Devido a forma como alguns controles CDUs têm sido modelados no banco de dados, é possível de se encontrar configurações em que parte da lógica central do controlador é feita remotamente por outro controlador. Por exemplo, em algumas configurações de parques eólicos, existe um controlador associado ao gerador eólico (fonte *shunt* controlada), mas existe um controle mestre que controla um ou mais parques (controladores não-específicos). No Anatem, a solução dos principais modelos CA está descrita na ordem da Figura 4, para um passo de iteração. Esse processo se repete até a convergência individual dos modelos, ou até que a constante IMDS seja alcançada.



FIGURA 4 – Ordem de solução dos principais modelos CA

Segundo a figura anterior, caso um modelo de eólica (fonte *shunt* controlada) espere sinais (IMPORT com local remoto) de um controlador central (não-específico), sua convergência estará prejudicada, uma vez que sua solução naturalmente se dá primeiro para as fontes *shunt* e somente depois resolverá os controladores não-específicos. Nestes casos, é necessário resolver, na melhor das hipóteses, o modelo de fonte *shunt* duas vezes, sendo uma dificuldade observada no processo de convergência.

Como alternativa, implementou-se no Anatem o código CCDU para CDU *Clustering* (ou Cluster de CDU). Por meio desse código, o usuário informa um conjunto de CDUs a serem resolvidos de forma agrupada, após a solução dos controladores não-específicos (Figura 4). É permitido associar até 10 CDUs em um único cluster, sendo que a cada linha do código CCDU, um novo cluster é definido, conforme ilustrado pela Figura 5. Na versão 12.0 do Anatem, apenas os controladores de fonte *shunt* e controladores não-específicos podem ser definidos em um cluster de CDU.

No caso deste exemplo, existem dois clusters definidos. O primeiro é definido pelos CDUs 90154, 90152, 90151, 90153; e o segundo é definido pelos controladores 10 e 20. Suponha que o CDU 10 é um CDU associado pelo DCNE e o CDU 20 é associado por um DFNT, neste exemplo os CDUs 10 e 20 não serão solucionados na sua etapa natural de solução, e serão deslocados para o final da solução de modelos CA, sendo resolvidos na ordem em que foram declarados no CCDU, ou seja, primeiro o CDU 10, e em seguida o CDU 20.

```

CCDU
(cdu1) (cdu2) (cdu3) (cdu4) (cdu5) (cdu6) (cdu7) (cdu8) (cdu9) (cdu10)
90154 90152 90151 90153
10 20
999999

```

FIGURA 5 – Exemplo de utilização do Código CCDU

Como o objetivo deste código é resolver problemas pontuais de convergência, não é esperado que o seu uso impacte no tempo de simulação, mas sim permita que os controles tenham melhor convergência, sendo potencialmente a diferença entre um caso com solução e um caso sem solução, em situações extremas. Adicionalmente, este recurso viabiliza a definição de CDUs com sinais de IMPORT e EXPORT com dependência circular, ou seja, os casos em que dois ou mais CDUs importam e exportam sinais entre si.

### 2.3 Arquivo de cache de DLCA

Na versão 11.2 do Anatem, foi disponibilizado um recurso por meio da opção DLCA para permitir ao Anatem a inserção automática de bloco DELAY nos pontos em que CDUs apresentem problemas de convergência suficientes para interromper a simulação. Para isso, o Anatem guarda cópias do estado do sistema com atualização periódica (ponto seguro) e retorna para o último ponto seguro no caso da ocorrência de problema de convergência, aplicando um DELAY automaticamente no ponto do CDU que não conseguiu convergir. Como o programa precisa recuar no tempo de simulação, os dados pertinentes ao arquivo de plotagem entre os pontos de salvamento são armazenados em *buffer*, que precisa ser descarregado em um arquivo quando um novo ponto seguro é alcançado ou então limpo quando é preciso retornar a um ponto seguro passado; e esse processo consome tempo e memória.

O DLCA é um algoritmo que é ativado quando a opção está habilitada e ele busca viabilizar a convergência de um caso que, caso contrário, seria abortado devido a um problema de convergência. Este algoritmo só atua, contudo, após a ativação do NEWT automático e do algoritmo SAD2, caso este também tenha sido aplicado junto do código EXSI, como ilustrado pela Figura 6. Ou seja, este algoritmo é o último na lista de prioridades do Anatem para tentar recuperar um caso com problema de convergência. Note ainda que o Anatem só considera o ponto como seguro caso o Anatem não esteja sob NEWT automático ou sob SAD2 ativado.

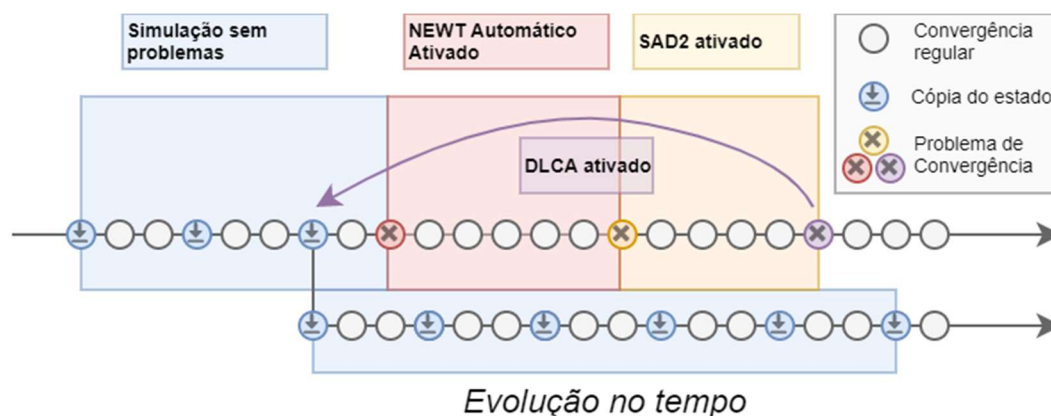


FIGURA 6 – Fluxo na ativação do algoritmo DLCA

Como forma de melhorar esta opção, criou-se um arquivo para o código DARQ com o tipo DLCA (ver Figura 7). Este novo arquivo funciona como cache do uso desta opção, onde são registrados todos os pontos em que o DLCA atuou em uma simulação anterior. Caso o uso do arquivo DLCA tenha sido declarado no código DARQ, o Anatem utilizará a informação escrita nesse arquivo para, desde o primeiro passo da simulação, aplicar o DELAY nos pontos indicados. Desta forma, não sendo esperadas novas ativações do algoritmo DLCA, esta opção pode ser retirada do EXSI, agilizando o processo de solução – dispensando o processo de criação dos pontos seguros e armazenamento em *buffer* do arquivo de plotagem.

```

DARQ
(Tipo) (C) ( Nome do Arquivo
OUT
LOG
PLT
DLCA

```

FIGURA 7 – Uso do arquivo DLCA no código DARQ

Diferentemente dos outros tipos de arquivo, o nome automático para o arquivo DLCA é sempre “cache.dlca”. O objetivo dessa decisão está em incentivar que os usuários compartilhem esse arquivo de cache em suas simulações,

para um mesmo caso ou para múltiplos casos. Desta forma, além de potencialmente acelerar a simulação, também existe um efeito de padronização dos modelos dinâmicos utilizados; ou seja, evita-se situações em que alguns casos teriam DELAY ativado e em outros, não.

#### 2.4 Identificação de controles com problemas de convergência

Dentro de uma simulação de transitórios eletromecânicos no Anatem, entre os pontos que podem prejudicar severamente o desempenho computacional está incluída a qualidade dos modelos de controladores a serem resolvidos. Devido à sua natureza, os modelos de controladores devem ser o mais bem detalhados o possível, suprimindo-se partes que não sejam pertinentes ao escopo eletromecânico; pois podem dispensar a representação malhas muito lentas (minutos) ou muito rápidas (microsegundos). Como a definição dos limites do que deve ser considerado é decorrente da experiência do modelador, modelos implementados podem ser mais ou menos bem construídos, considerando-se o desempenho computacional.



### 2.4.1. Relatório de estatística de convergência de CDUs (ESTC)

Na versão 12.0 do Anatem, implementou-se um recurso para ajudar a verificar o grau de convergência dos modelos, com a métrica de médias de solução por passo. Este recurso pode ser acessado por meio da opção ESTC junto do código EXSI (modo compacto) ou do código RELA (modo completo). A diferença é que no código EXSI, são impressos os 10 controles com maior média de solução por passo; e quando usado com o código RELA, é impressa a relação de todos os controladores ativos do sistema. A Figura 8 ilustra a impressão deste relatório:

RELATÓRIO DE ESTATÍSTICA DE CDU (PIORES)			
Controlador	Soluções		
CDU	Nome do CDU	por passo	Severidade
90154	CONJ_SBABILO	15.3486	*****
90152	CONJ_SBABILO	14.9903	*****
2673	ES_LCPRESTAG	11.5838	*****
2643	RV_LCPRESTAG	11.5838	*****
2603	RT_LCPRESTAG	11.5838	*****
2675	ES_LCPRESTBG	11.5836	*****
2645	RV_LCPRESTBG	11.5836	*****
2605	RT_LCPRESTBG	11.5836	*****
2676	ES_LCPRESTBV	11.0431	****
2646	RV_LCPRESTBV	11.0431	****

FIGURA 8 – Relatório ESTC junto do código EXSI (piores casos)

Sendo o campo “severidade” uma métrica determinada em função da distância entre a métrica “Soluções por passo” e a relação média de todos os modelos. Cada asterisco representa um desvio padrão além da média: no exemplo anterior, o modelo 90154 estaria a sete desvios padrão do resultado médio deste caso.

Naturalmente, o grau de convergência de um modelo está sujeito a várias condições (como a localidade e o tipo dos eventos aplicados, a qualidade numérica da matriz Ybarra do sistema) antes de propriamente indicar um problema de modelagem do controle. Este relatório não deve ser utilizado, portanto, como um crítico da qualidade do modelo, servindo apenas para identificar modelos que destoam dos demais em convergência. Por exemplo, em uma simulação de um evento aplicado na região Sul, é inesperado que um modelo de eólica do Nordeste apresente um elevado número de iterações para convergir.

### 2.4.2. Outras considerações acerca da convergência do processo iterativo

Nem sempre modelos com dificuldade para convergir representam problemas de desempenho computacional. Alguns modelos podem apresentar excelente grau de convergência interna (convergem em poucas interações), mas, durante o processo de solução alternado de solução entre rede CA e modelos CA, alteram a sua resposta a cada iteração. Podem apresentar um problema de convergência unicamente por estarem na ordem indevida de solução (controles comunicantes): nesta hipótese, o novo código CCDU deve auxiliar a convergência dos modelos.

Na seção “Solução de Problemas de Convergência e Desempenho” do manual do Anatem [4], são elencados os seguintes pontos que podem interferir na solução: passo de integração inapropriado, modelagem incompleta de fontes de geração, modelagem inapropriada da carga, parâmetro Vmin do modelo da carga ajustado incorretamente, constantes de convergência modificadas, excesso de eventos simultâneos, dados de falha de comutação espúrios.

O usuário também deve estar atento para a possibilidade de uma modelagem inapropriada da rede elétrica. Dois possíveis problemas incluem a relação X/R dos circuitos ser baixa e baixo valor para impedância em circuitos radiais. Ambos os problemas prejudicam a solução da rede, provocando mais iterações entre modelos CA e rede CA, aumentando o tempo computacional necessário e impactando nas métricas dos modelos CDU, indevidamente.

## 3.0 SIMULAÇÕES E DEMONSTRAÇÃO DOS RECURSOS APRESENTADOS

Nesta seção, serão apresentados os resultados do impacto dos recursos descritos nesse trabalho sobre o caso do SIN do ONS (2019). As simulações foram realizadas no Anatem 12.0.3, em um workstation em ambiente Windows com processador AMD Ryzen Threadripper 3960x, com 24/48 cores e frequência base de 3.8 GHz até 4.5 GHz.

### 3.1 Análise do impacto do arquivo de cache de DLCA

Será demonstrado o ganho de desempenho trazido pelo arquivo de cache de DLCA. O principal ganho desse recurso está na eventual repetição da simulação, e.g. quando existe algum ajuste menor. Nesses casos, como o Anatem não terá de retornar a um ponto seguro, pois os DELAYs são aplicados desde o instante inicial, a simulação deverá potencialmente ser mais rápida (em particular em casos com difícil convergência).

Utilizando uma simulação de um curto-circuito monofásico (evento APCC) de 100 ms na linha de 500 kV em Colinas (TO) do BD2019 (caso de referência) do ONS [3] com 30 segundos de simulação, obteve-se as seguintes estatísticas de simulação (Figura 9):

<p>Tempo de CPU da simulação: 00:13:48.18</p> <p>Média de soluções de rede CA por passo: 22.26</p> <p>Média de soluções de modelo CA por passo: 29.57</p> <p>Média de iterações modelos CA - rede CA por passo: 4.13</p> <p>Média de iterações modelos CC - rede CC por passo: 9.66</p> <p>Média de iterações CC - CA por passo: 1.25</p>	<pre> X-----X-----X-----X-----X-----X-----X-----X   RELATÓRIO DE INSERÇÃO DE BLOCO DELAY                     (DLCA)                         Controlador      Bloco      Adicionado         CDU      Nome do CDU      Variável      ao cache         X-----X-----X-----X-----X-----X-----X-----X     90154    CONJ_SBABILO      6140    X6140         91516    CONJ_RIACHAO      575     ETI2          91516    CONJ_RIACHAO      572     ETR2          90513    CONJ_ICARAI       575     ETI2          90534    CONJ_TRAIRI       575     ETI2          91257    CONJ_SBASTILI     575     ETI2          91541    CONJ_BRIPOTI      575     ETI2          X-----X-----X-----X-----X-----X-----X-----X </pre>
---	---

FIGURA 9 – Estatísticas de simulação sem cache de DLCA (a) e relatório automático de DLCA (b)

No caso apresentado, houve grande número de soluções de modelo e rede CA, indicando que existem controles com problema de convergência. O algoritmo DLCA precisou atuar 7 vezes neste caso (ver relatório da Figura 9.b). Com o arquivo de cache de DLCA declarado no código DARQ, na primeira vez em que o caso é simulado (ou então na ausência do arquivo cache.dlca na pasta do arquivo stb), este arquivo será escrito com os dados dos blocos de CDU em que o DELAY foi aplicado. Executando uma segunda vez este caso (e todas as vezes subsequentes), obteve-se os resultados apontados na Figura 10:

Tempo de CPU da simulação: 00:09:49.59
Média de soluções de rede CA por passo: 22.27
Média de soluções de modelo CA por passo: 29.57
Média de iterações modelos CA - rede CA por passo: 4.13
Média de iterações modelos CC - rede CC por passo: 9.66
Média de iterações CC - CA por passo: 1.25

FIGURA 10 – Estatísticas de simulação com cache de DLCA

Neste caso em particular, o *speedup* observado pelo uso deste recurso é de 1,4 vezes, para todas as execuções subsequentes à primeira. Em alguns casos outras métricas de solução do programa, como o número de soluções de modelos CA por passo podem ser melhoradas por meio deste recurso. Como o arquivo de cache de DLCA é compartilhado por todos os arquivos stb em uma mesma pasta (a menos que seja alterado o nome no DARQ), a configuração de DLCA é compartilhada entre todos os casos, potencialmente agilizando as outras simulações, mesmo que ainda não tenham sido executadas antes. O compartilhamento do cache também garante uniformidade na aplicação do DLCA entre as simulações.

### 3.2 Identificando CDUs com problema de convergência com a opção ESTC

Embora o recurso do cache de DLCA tenha melhorado o desempenho computacional, é de suspeitar que existe algum problema nos modelos, uma vez que o algoritmo DLCA precisou ser ativado muitas vezes. Pelo relatório da Figura 9.b, é possível observar que como são os mesmos blocos (mesma numeração e mesmo nome) em diferentes controles, sinalizando que pode existir espaço para melhorias no modelo CDU desses parques. Mas essa análise é incompleta, o ideal seria conhecer a estatística de convergência de cada CDU. Utilizando a opção ESTC junto do código EXSI e simulando o mesmo caso novamente (com o cache de DLCA encontrado na subseção anterior), obteve-se o relatório indicado na Figura 11.

RELATÓRIO DE ESTATÍSTICA DE CDU (PIORES)			
Controlador	Soluções		
CDU	Nome do CDU	por passo	Severidade
90154	CONJ_SBABILO	53.1992	*****
90145	CONJ_CLARGO	22.1931	*****
90146	CONJ_CLARGO	22.1931	*****
90147	CONJ_CLARGO	22.1627	*****
90148	CONJ_CLARGO	22.1627	*****

FIGURA 11 – Corte do relatório de estatística de simulação ESTC de CDUs, com cache DLCA

Uma vez que os DELAYS tenham sido aplicados nos controladores CDUs, é de se esperar que os CDUs afetados não apareçam mais como os principais responsáveis pelos problemas de convergência. Contudo, o modelo 90514 apresenta-se destacado, com um nível de severidade muito maior que os dos outros controladores listados. Analisando este CDU, é possível notar que ele é um controlador não-específico que se comunica com um controlador de fonte *shunt* controlada de número 90512. Neste caso, é indicado o uso do cluster de CDU.

### 3.3 Melhorando a convergência de parques eólicos com o cluster de CDU (CCDU)



Sabendo-se da relação dos CDUs 90154 e 90152, pode-se associá-los no código CCDU, ordenando-os conforme a conveniência. Neste caso, o CDU 90154 exporta sinal para o 90152, por isso deve ser resolvido primeiro (Figura 5). Como o próprio CDU 90154 foi incluído no rol de CDUs com DELAY adicionado, é recomendado que seja retirado do cache de DLCA (a princípio) a variável X6140 uma vez que a natureza da convergência do CDU 90154 foi alterada pelo código CCDU. Caso necessário, o algoritmo DLCA adicionará o DELAY novamente. Simulando novamente o caso a partir desse cache modificado, obteve-se os resultados da Figura 12.

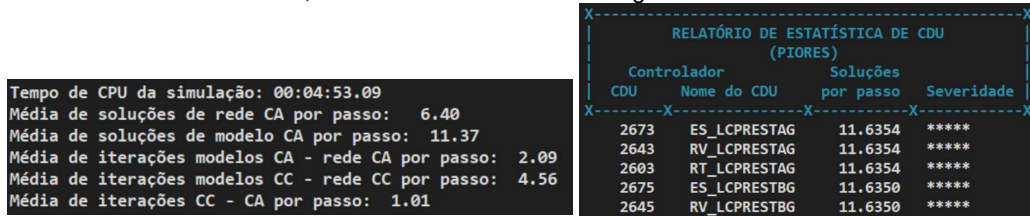


FIGURA 12 – Estatísticas de simulação (a) e corte do relatório ESTC (b) com CCDU no caso

Com a adição o cluster de CDDU, é possível notar que há melhoria significativa no desempenho (Figura 12.a), de 09min49s para 04min53s, cerca de 2,0 vezes de *speedup*. De fato, pela Figura 12.b, o CDU 90154 não mais figura como um controle com problema de convergência, melhorando os índices inclusive de outros controladores de eólicas e fotovoltaicas, apontando que eventualmente problemas de convergência em um modelo podem ser decorrentes de outros modelos próximos.

### 3.4 Análise do desempenho do algoritmo SPMD

Para a avaliação do desempenho do algoritmo SPMD, considerou-se o mesmo caso simulado anteriormente, variando-se o número de processadores envolvidos (por meio do parâmetro -n da chamada por *shell*), considerando-se ainda conjuntos de dados distintos. Em um cenário (caso A), os modelos dinâmicos de eólicas e fotovoltaicas estão ausentes no caso; no segundo cenário (caso B), estes modelos estão presentes. Como o DLCA só foi aplicado nos modelos de eólica e fotovoltaica, o cache de DLCA não está presente no primeiro cenário. Os resultados desta análise podem ser conferidos na Figura 13.

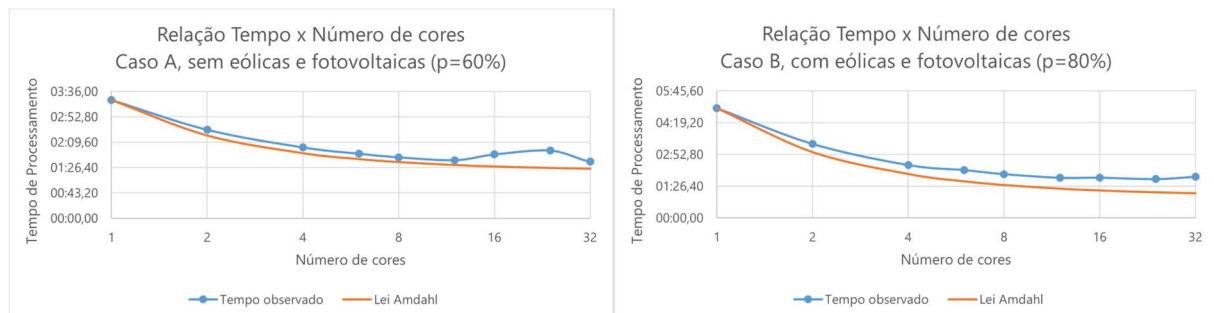


FIGURA 13 – Comparação no tempo de processamento frente o número de cores, para os dois casos A e B

Em ambos os cenários, existe aproximação entre as curvas ideal (Lei de Amdahl) e a curva obtida na simulação. A diferença entre elas se deve a diversos fatores, como: carga desbalanceada na solução (cada CDU apresenta um tempo próprio para ser resolvido causando ociosidade em alguns momentos); custo de *overhead* nas regiões paralelas; arquitetura do processador (como a memória cache L3 e L2 está distribuída pelas unidades de processamento); frequência de operação variada em função do número de cores ativos (com menos cores, a frequência de operação é maior). É também por causa desses fatores que nem sempre um maior número de cores alocados para o processo garante ganho de desempenho, como observado em ambos os cenários, a partir de 12 cores empregados.

Para o caso A, o tempo de processamento foi de 03min21s para 01min38s no melhor cenário, com 12 cores, alcançando um *speedup* de 2,03 vezes (máximo esperado de 2,22 vezes). Em termos de tempo computacional, a diferença entre o observado (01min38s) e o máximo ideal (01min30s) é de apenas 8 segundos, sugerindo adequação na implementação atual do algoritmo SPMD.

Para o caso B, o tempo de processamento foi de 04min57s para 01min49s, com 12 cores, alcançando um *speedup* de 2,71 vezes (máximo esperado de 3,75 vezes). No cenário com 12 cores, era esperado que a simulação fosse resolvida em 01min19s, uma diferença de 30 segundos. Este resultado sugere que a paralelização de modelos de eólicas e fotovoltaicas possa estar sofrendo com o desbalanço de carga (os modelos são densos e muito diferentes entre si), indicando que há espaço para melhoria na estratégia de paralelização nestes modelos.

Em função do algoritmo SPMD e dos demais algoritmos introduzidos na versão 12.0 do Anatem e apresentados neste trabalho, deve-se destacar que para uma máquina típica (com 4 cores) o desempenho em tempo computacional do caso com eólicas (cerca de 2min26s) se aproxima do desempenho do caso sem esses modelos

(cerca de 2min00s). No caso do *workstation*, com 12 cores, essa diferença foi de apenas 10 segundos. Esse resultado é particularmente importante, pois retira o ônus computacional causado pela solução dos modelos de eólica e fotovoltaica, garantindo uma maior qualidade na análise, com praticamente o mesmo custo computacional. A Figura 14 ilustra a evolução obtida com os recursos nos dois casos (A e B) simulados nesta subseção.

Um outro resultado importante evidenciado pela Figura 14 está no ganho de desempenho global observado entre o tempo computacional requerido para a simulação do caso B originalmente (13min48s) e o tempo requerido depois de todos os recursos apresentados (01min49s, com 12 cores empregados). O *speedup* observado seria de 7,57 vezes nesse caso, sendo autoevidente o ganho de produtividade trazido para o usuário.

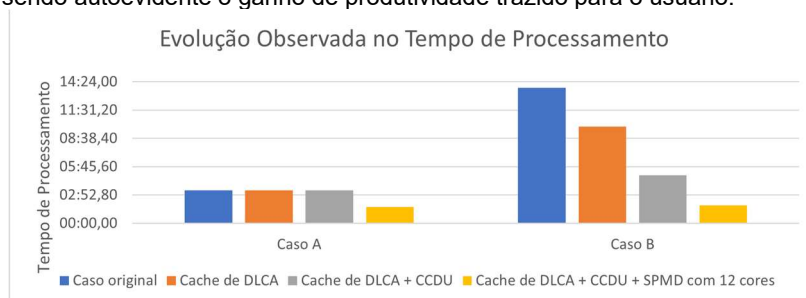


FIGURA 14 – Comparação gráfica dos ganhos obtidos com os recursos da versão 12.0 para simulações sem modelos dinâmicos de eólicas e fotovoltaicas (A) e com esses modelos no caso (B)

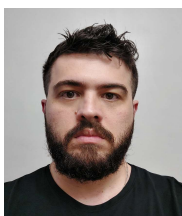
#### 4.0 CONCLUSÃO

Neste trabalho foram apresentados diversos recursos disponíveis na versão 12.0 do Anatem para agilizar a simulação de casos de estabilidade eletromecânica. O recurso do cache de DLCA permite que as atuações do algoritmo do DLCA sejam usadas em simulações subsequentes, reduzindo o custo computacional do algoritmo. O relatório de estatística de convergência de modelos CDU permite a identificação de controladores com problemas de convergência, para posterior melhorias. O CCDU permite o agrupamento de controladores de parques eólicos com controles mestres decentralizados, melhorando a convergência do conjunto e viabilizando a inicialização de CDUs com dependência circular. O algoritmo SPMD permite aproveitar melhor os recursos computacionais disponíveis para acelerar simulações em instâncias únicas de casos Anatem. Com os recursos apresentados, pode-se realizar simulações considerando os complexos modelos de eólicas e fotovoltaicas, reduzindo o ônus.

#### 5.0 REFERÊNCIAS BIBLIOGRÁFICAS

- [1] LIRIO, F. L., NETTO, N. A. R. L. e DA SILVA, L. R. "Manual do Usuário do Programa Anatem", Cepel – Centro de Pesquisas de Energia Elétrica, Fevereiro de 2021. Versão 12.0. Disponível em: <https://dre.cepel.br/manual/anatem/>
- [2] AMDAHL, G. M. "Validity of the single processor approach to achieving large scale computing capabilities". In: Proceedings of the April 18-20, 1967, Spring joint computer conference (AFIPS '67 (Spring)). Association for Computing Machinery, New York, NY, USA, 483–485.
- [3] OPERADOR NACIONAL DO SISTEMA ELÉTRICO "Casos de Referência – Estabilidade – Transitórios Eletromecânicos", visitado em: abril de 2019, Disponível em: <http://www.ons.org.br/>.
- [4] LIRIO, F. L., NETTO, N. A. R. L. e DA SILVA, L. R. "Solução de Problemas de Convergência e de Desempenho", In: "Manual do Usuário do Programa Anatem", Cepel – Centro de Pesquisas de Energia Elétrica, Fevereiro de 2021. Versão 12.0. Disponível em: [https://dre.cepel.br/manual/anatem/misc/problema\\_convergencia.html](https://dre.cepel.br/manual/anatem/misc/problema_convergencia.html)

#### 6.0 DADOS BIOGRÁFICOS



Formado em Engenharia Elétrica pela Universidade Federal do Rio de Janeiro (2014) e M.Sc. em Engenharia Elétrica (2016) pela mesma instituição. Desde 2014, trabalha como pesquisador no Centro de Pesquisas de Energia Elétrica (Cepel), onde tem atuado na área de métodos computacionais para dinâmica de sistemas de potência e análise de segurança de sistemas de potência, pertencendo à equipe de desenvolvimento do software Anatem. Suas áreas de atuação incluem computação de alto desempenho, dinâmica e confiabilidade de sistemas de potência.

#### (2) LÍGIA ROLIM DA SILVA

Possui graduação em Engenharia Elétrica pela Universidade Estadual do Rio de Janeiro (2015) e mestrado em Engenharia Elétrica pela Universidade Federal do Rio de Janeiro (2020). Desde 2015, trabalha no Centro de Pesquisas de Energia Elétrica, como pesquisadora do Departamento de Redes Elétricas, onde integra a equipe de desenvolvimento do programa Anatem. Atua na modelagem computacional de equipamentos de sistemas elétricos

de potência para análise dinâmica de transitórios eletromecânicos. Suas áreas de interesse incluem dinâmica e controle de sistemas de potência, ressonância subsíncrona e transmissão CCAT. É membra do Comitê de Estudos C2 do Cigré-Brasil.

(3) FABRICIO LUCAS LIRIO

Possui graduação e mestrado em Engenharia Elétrica pela Universidade Federal de Juiz de Fora (UFJF), respectivamente em 1998 e 2000. Doutor em Engenharia Elétrica pela Universidade Federal do Rio de Janeiro (COPPE/UFRJ) em 2007. Desde 2000 trabalha no Cepel, onde atua no Departamento de Redes Elétricas (DRE). Trabalha no desenvolvimento do Programa Anatem desde 2013, atual responsável pelo projeto Anatem, participou dos estudos da alternativa CCAT para o Aproveitamento das usinas do rio Madeira e Belo Monte. Seus campos de interesse são: Transmissão CCAT, Equipamentos FACTS, Controle aplicado a Sistemas de Potência, Estudos de Transitórios Eletromagnéticos, Eletromecânicos e Harmônicos.