

# APLICAÇÃO DO ALGORITMO DE OTIMIZAÇÃO BAYESIANA COM MUTAÇÃO NO PLANEJAMENTO DA EXPANSÃO DE SISTEMAS DE TRANSMISSÃO DE ENERGIA ELÉTRICA<sup>1</sup>

**Thiago Lablonsk MEDEIROS<sup>2</sup>**  
IF/IFSP  
Aluno/IFSP

**Muriell de Rodrigues FREIRE<sup>3</sup>**  
-/UNIFEI  
Professor/IFSP

## RESUMO

Este trabalho tem como objetivo realizar uma aplicação do Algoritmo de Otimização Bayesiana (BOA) no problema do Planejamento da Expansão da Transmissão (PET). A formulação do PET será baseada no planejamento de curto prazo, considerando o critério de segurança “N-0” (é considerado a rede intacta para planejamento, ou seja, todos os elementos de transmissão operando normalmente). As melhores soluções para o problema foram encontradas através de um algoritmo evolutivo especialista, sendo utilizadas como referência para a implementação do algoritmo BOA. A partir de uma adaptação do BOA, foi implementado um operador de mutação visando gerar mais diversidade e assim melhorar as buscas. As análises foram feitas comparando o BOA original com a versão modificada. A comparação entre os algoritmos foi utilizando um método gráfico (gráfico de convergência) comparando os casos para discussão. O algoritmo pode ser utilizado não somente para o PET, mas também para outros problemas que envolvem algoritmo evolutivo.

**Palavras-chaves:** Algoritmo Evolutivo. Mutação. Redes Bayesianas. Planejamento da Expansão da Transmissão de Curto Prazo.

## INTRODUÇÃO

O Planejamento da Expansão da Transmissão (PET) é um problema clássico na área de estudos de Sistemas Elétricos de Potência e tem como objetivo fornecer um plano da expansão para o sistema de transmissão que garanta a manutenção do fornecimento de energia e de forma desejada. Dado um cenário futuro de demanda de geração e geração de energia, o PET busca encontrar uma solução ótima, com menor

---

<sup>1</sup>Iniciação Científica do Curso de Engenharia de Controle e Automação.

<sup>2</sup>thiagolablonsk@gmail.com

<sup>3</sup>muriell@ifsp.edu.br

custo de investimento e que atenda as especificações de qualidade, segurança (LEITE DA SILVA; FREIRE; HONÓRIO, 2016) e que seja capaz de fornecer informações como linhas de transmissões a serem construídas e equipamentos necessários para o funcionamento da rede, além de locais e de reforços necessários. Por se tratar de um problema com um certo grau de complexidade mesmo desconsiderando incertezas como (crescimento de carga local, perdas etc.).

A Rede Bayesiana (RB) é gerada através de uma tabela de probabilidades utilizando o teorema de Bayes. Cada nó pai possui um ou mais filhos e seus filhos são pais de outros nós ou são folhas (quando os filhos não possuem descendentes). Os nós possuem ligações em suas arestas que levam do nó pai para o nó filho (um nó pode ter mais de um pai e mais de um filho, mas nunca terá uma “seta” de volta para seu pai). Todo nó possui variáveis onde a probabilidade encontra-se dividida entre as variáveis (probabilidade máxima é 1 ou 100%). A ligação do nó pai com o filho depende da relação de probabilidades dado por Bayes que pode ou não interferir nas probabilidades dos nós filhos. Realizar inferência significa escolher um nó da rede e neste nó escolher uma das variáveis transformando a variável em 100%, ao realizar esta tarefa a probabilidade vai se propagar podendo ou não (tudo depende de como está implementado as relações de pai para filho) modificar a probabilidade dos nós filhos como um efeito domino, sendo assim os pesos das variáveis filhos podem estar relacionados aos pesos das variáveis nó pai. As conexões entre pais e filhos ou os pesos (probabilidade de cada variável) de cada nó depende diretamente das configurações da RB, tais configurações como a priori ou posteriori necessária para estabelecer as relações entre os nós.

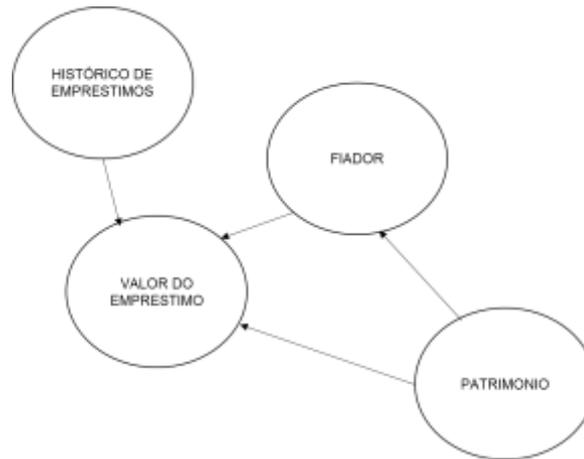


Figura 1. Estrutura de rede Bayesiana

Fonte: Autores

O BOA é um modelo probabilístico de otimização estocástico baseado nos conceitos de algoritmos evolutivos que utiliza operadores genéticos de cruzamento para gerar novos indivíduos (planos quando se trata do PET), utilizando operadores de inferência em uma RB sendo a mesma gerada a partir da estimação de distribuição de probabilidades de melhores indivíduos da população atual. Partindo de uma população inicial, os melhores candidatos são escolhidos para construir a RB. Um agente faz inferência na RB escolhendo um nó que ainda não foi selecionado aleatoriamente da RB e aleatoriamente uma variável de interesse, realizando a inferência na mesma, repetindo o processo até que todos os nós sejam inferidos assim gerando um novo indivíduo/plano. O processo de coleta de indivíduos/planos se repete até que uma condição seja satisfeita, após a coleta é feita uma verificação de factibilidade e se o mesmo é melhor que algum plano conhecido, caso seja melhor que mais de um plano o algoritmo deve substituir do que mais se aproxima do plano que está sendo avaliado (distância algébrica no caso utilizado), o processo termina quando todos forem avaliados. Com uma nova população o processo se repete até que uma condição de parada seja satisfeita.

Este projeto propõem o estudo, aplicação e melhoria do BOA (PELIKAN et al., 1999) utilizando mutação, voltado para o sistema PET onde os melhores planos que compõem a RB final que é representado por um grafo acíclico relacionando as probabilidades e dependência entre os nós relacionando causa/efeito entre variáveis do problema caso uma inferência seja feita pode ou não afetar a probabilidade de nós

filhos, sendo assim o planejador pode avaliar o conjunto de planos disposto no grafo, onde cada nó possui informações relevantes no lugar em que se aconselha a tomar uma decisão ou não. A partir das informações fornecidas, o planejador pode utilizar de sua experiência para avaliar as informações fornecidas extraindo informações através de inferências na RB para obter um plano de expansão. Os testes foram feitos para os sistemas Garver e IEEE - RTS79 (IEEE, 1979). Os algoritmos desenvolvidos utilizam de recursos probabilísticos para simular o processo biológico natural (reprodução, recombinação, seleção e com o BOA modificado a mutação).

## **DESENVOLVIMENTO**

O trabalho desenvolvido trabalha com três abordagens, sendo elas sem mutação (BOA (PELIKAN et al., 1999)) e as outras duas apelidadas de BOAME (BOA Mutação Estática) e BOAMD (BOA Mutação Dinâmica), as duas últimas são criações de autoria própria utilizando a ideia do GA (Algoritmo Genético) (LEITE DA SILVA; FREIRE; HONÓRIO, 2016). O algoritmo BOA pode ser melhor explicado na Figura 2 e 3. A ferramenta utilizada para visualizar a RB ao final do processo é distribuída pela empresa BayesFusion (GeNIe, 2019), versão acadêmica. Os resultados utilizados como base foram adquiridos pelo algoritmo evolutivo especialista AES-TEP (LEITE DA SILVA; FREIRE; HONÓRIO, 2016), onde foram extraídas as 10 melhores soluções por exaustão utilizando o critério “N-0” sem relaxamento. As 10 melhores soluções fornecem informações de reforços necessários em cada ramo para um sistema com o custo total para uma futura expansão.

Os sistemas escolhidos foram o Garver (sistema clássico) e o sistema IEEE RTS – 79 que também é amplamente utilizado para fins acadêmicos com 24 barras e 41 ramos candidatos a expansão (FREIRE, 2016).

Os três algoritmos foram testados e comparados utilizando um gráfico de convergência. Os parâmetros estudados foram: tamanho da população, o máximo de parentes que cada nó da RB poderia ter, número de interações, tamanho da amostra (algoritmo que gera a RB), a probabilidade das ligações (nó pai para nó filho) e ligações à priori. Os resultados obtidos para cada sistema são: para o Garver os melhores resultados ficaram com o número de interações igual a 3, tamanho da amostra 4, ligação

de probabilidade “0,1”, ligação à priori “0,001” e o número de gerações foi de 400. No IEEE o número de interações igual a 800, o tamanho da amostra foi alterado para cada teste a fim de mostrar a variação que será discutida em resultados já que a variação das interações afetou somente o tempo de execução, a ligação de probabilidade 0,5 e ligação à priori 0,5. Para ambos os testes (Garver e IEEE) a população é de 50, máximo de parentes igual a 2. E cada indivíduo foi replicado 800 vezes isso porque a RB consegue calcular melhor as probabilidades e construir as relações de pai para filho e esse valor foi descoberto através de vários testes onde para números acima de 800 não foi possível perceber variações na estrutura da RB sendo o valor limite para o PET.

As Figuras 5 e 6 descrevem como a mutação funciona na RB mantendo o a ideia do BOA. Um ponto que foi alterado para o BOA e aplicado aos demais são as populações (P, S e O(t)) utilizadas onde não se trabalha mais com S(t), percebeu-se que não faz sentido algum trabalhar com uma população que nunca entra em processo nenhum no caso do PET.

A mutação utilizada funciona da seguinte forma: partindo de uma taxa de mutação, verifica-se a mesma vai ocorrer, caso ocorra o valor deve aumentar ou diminuir em 1 isto ocorre para cada informação (cada reforço no caso do PET) e para cada indivíduo gerado a partir de uma amostra retirada da RB utilizando um agente de amostragem, caso o valor da informação exceda o mínimo ou o máximo (0 ou 5 respectivamente) não ocorre nenhuma mutação, na Figura 5, 6 e 7 mostra como é feito o processo de mutação na RB. A mutação estática possui um valor fixo de mutação e quando este é muito elevado o algoritmo não consegue convergir (encontrar os 10 melhores) pois ocorre mutações na maioria das vezes, enquanto na mutação dinâmica o valor da taxa de mutação acompanha a evolução do algoritmo, decaindo (menor a chance de ocorrer mutações) conforme se aproxima do melhor global conhecido, até que nunca mais ocorra mutações (encontrou o melhor global).

O agente de amostragem é utilizado em todos os algoritmos, o mesmo trabalha realizando amostragem, com um sorteio de um número inteiro aleatório que varia de 1 até o número de nós existentes na RB (que ainda não foram sorteados), após a escolha do nó o próximo passo é a seleção da variável do nó que sofrerá inferência, se o número de variáveis é maior que 1 faz se a roleta que sorteia um número real aleatório de 0 a 100. Cada nó possui uma ou mais variáveis e cada variável possui um peso

(probabilidades) que varia de 0 até 100 e para cada nó as variáveis encontram-se ordenadas em ordem crescente (independentemente do valor do peso de cada variável), a partir disto é feito um sorteio com a roleta e se o peso da variável menos o número sorteado for menor ou igual a zero, então é feita inferência na variável atual, caso contrário o resto da operação é utilizado repetir o processo para a próxima variável até que o resultado de alguma seja menor ou igual a zero. Ao realizar a inferência a probabilidade pode ser propagada ou não para seus nós filhos. Após a inferência feita e a probabilidade propagada na RB uma amostra é gerada utilizando a posição do nó e da variável que sofreu inferência na sua respectiva posição e valor.

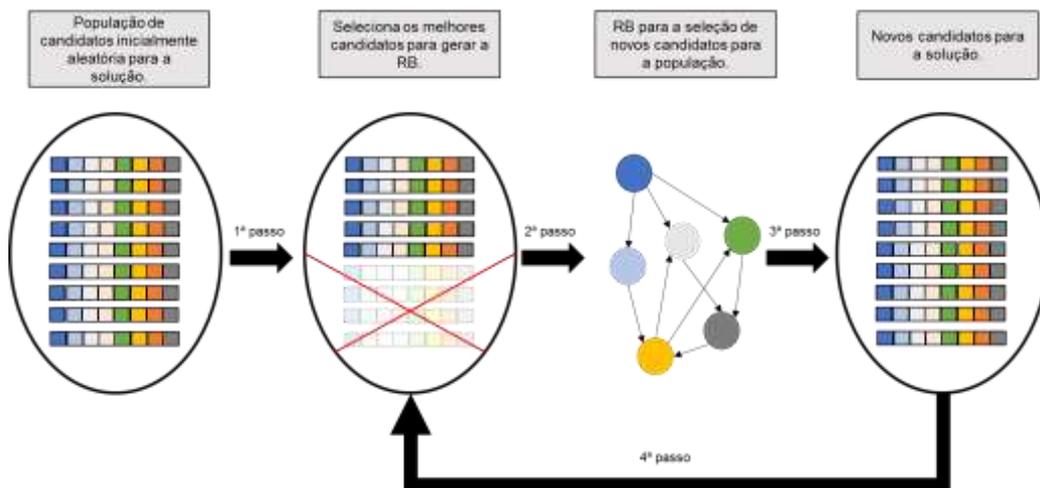


Figura 2: Ilustrativo do procedimento do algoritmo de otimização Bayesiano

Fonte: Autores

- 1:  $t$  recebe 0.
- 2: Aleatoriamente gere uma população inicial  $P(0)$ .
- 3: Selecione um conjunto de vetores promissores  $S(t)$  de  $P(t)$ .
- 4: Construa a rede  $B$  usando uma métrica escolhida e restrições.
- 5: Gere um conjunto de novos vetores  $O(t)$  de acordo com a distribuição conjunta codificada em  $B$ .
- 6: Crie uma nova população  $P(t + 1)$  substituindo alguns vetores de  $P(t)$  com  $O(t)$ .
- 7:  $t$  recebe  $t + 1$ .
- 8: Se o critério de parada não for satisfeito, vá para o passo 3.

Figura 3: Pseudocódigo do BOA original

Fonte: REGOLIN (2004)

1. T recebe 0.
2. Gera uma população factível  $P(t)$  aleatória de tamanho N.
3. Constrói uma RB a partir de  $P(t)$
4. Realiza inferência para gerar a população  $O(t)$  de tamanho N.
5. Substitui os filhos  $O(t)$  mais próximos de  $P(t)$  se factível.
6. T recebe  $T + 1$ .
7. Se o critério de parada não for satisfeito volta para o passo 3.

Figura 4: Pseudocódigo BOA Adaptado pelo Autor

Fonte: Autores

O algoritmo para construir uma RB final com um conjunto de soluções segue o seguinte procedimento BOA:

- O BOA é iniciado a partir de uma população inicial aleatória  $P(t)$  com um tamanho de N indivíduos.
  - Os  $S(t)$  melhores  $P(t)$  são selecionados a para gerar uma RB.
  - Em  $S(t)$  utilizando a métrica de pontuação mínima – *MinimumDescriptionLength* (MDL) proposto por Bouckaert (Bouckaert,1994) (esta métrica é usada com objetivo de encontrar a estrutura de rede que melhor descreva o conjunto de utilizando o mínimo de elementos possíveis para calcular a probabilidade conjunta da rede de crença, reduzindo desta forma o esforço computacional necessário no cálculo das inferências) e algoritmo de busca K2 (COOPER e HERSKOVITS, 1992) (método de busca gulosa que procura maximizar a qualidade da estrutura de Redes Bayesianas),  $S(t)$  são multiplicados 800 vezes cada uma de modo a terem relações probabilísticas entre os ramos de entrada utilizados para construir a RB e através de inferências são extraídos planos formando a população  $O(t)$  que são as amostras (quando se faz inferência as probabilidades se propagam e nós filhos tem suas probabilidades alteradas ou não), sendo  $O(t)$  do mesmo tamanho de  $S(t)$  e não repetidos. (todo esse processo agora é feito no passo 2 na  $O(t)$ ).
- Uma nova População  $P(t+1)$  é gerada usando os valores da população atual substituindo indivíduos  $O(t)$  que estão próximos (distância geométrica).
- O processo se repete a partir a seleção até que o número de gerações seja o máximo ou que o algoritmo deixe de aprender novas RB.

- O algoritmo finaliza extraindo os 10 melhores como é feito nos passos anteriores, gerando uma RB com o mesmo.

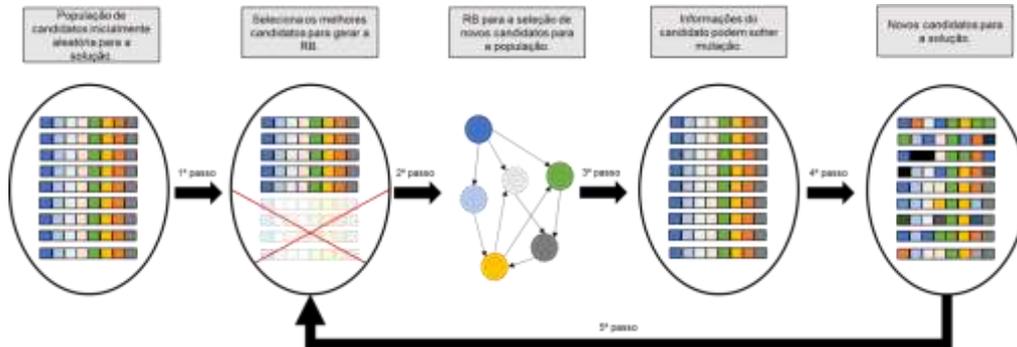


Figura 5: Ilustrativo do procedimento do algoritmo de otimização Bayesiano com mutação.

Fonte: Elaboração própria.

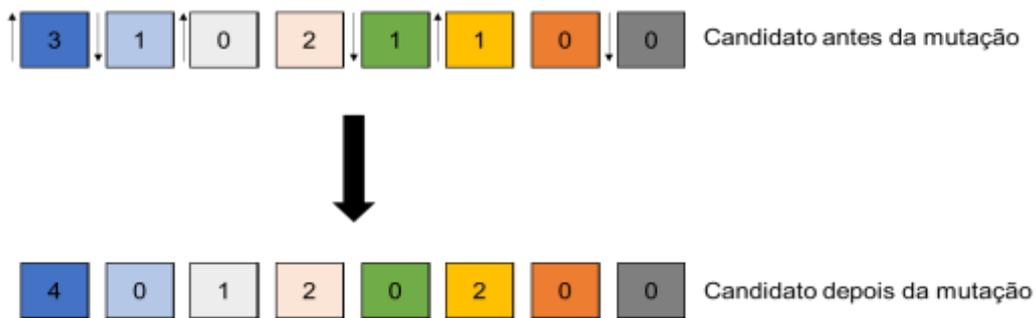


Figura 6: Ilustrativo do processo de mutação de cada candidato para a solução.

Fonte: Elaboração própria.

1. T recebe 0.
2. Gera uma população factível  $P(t)$  aleatória de tamanho N.
3. Constrói uma RB a partir de  $P(t)$
4. Realiza inferência para gerar a população  $O(t)$  de tamanho N.
5. Tenta realizar mutação simples em cada indivíduo
6. Substitui os filhos  $O(t)$  mais próximos de  $P(t)$  se factível.
7. T recebe  $T + 1$ .
8. Se o critério de parada não for satisfeito volta para o passo 3.

Figura 7: Pseudocódigo BOA - MUTAÇÃO

Fonte: Autores

Para estimar o tamanho da população, foram realizados testes que variavam de 20 até 300 e os resultados onde a variação era quase nula em relação à mudança de parâmetros (estabilidade) foi uma população de tamanho 50. Esse parâmetro também apresentou melhores desempenhos quando se trata da busca do melhor global. Uma população de 50 significa ter 50 planos diferentes inseridos em uma RB, replicado um a um 800 vezes. Em termos de algoritmo evolutivo, ambos os testados possuem uma população bem compacta que para cada geração se trabalha no máximo com 50 indivíduos diferentes enquanto outros algoritmos geram diversos indivíduos superando até mesmo a população inicial no número de indivíduos gerados. A cada geração o algoritmo faz a tentativa de inserir novos planos na população atual  $P(t)$ , isso significa que para cada geração o máximo de indivíduos gerados é o mesmo número de indivíduos da população inicial  $P(0)$  após a primeira seleção. Para um algoritmo de aprendizagem isso pode ser muito compacto, trabalhar com populações e espaços de busca bem pequenos e ser capaz de encontrar o melhor global.

O processo de mutação tem como objetivo acelerar a busca e aproveitar as RB geradas. O intuito é espalhar a busca (gerar diversidade), ao invés de apenas combinar informações semelhantes.

## **RESULTADOS OBTIDOS**

Nesta seção os resultados apresentados farão um comparativo de teste entre os algoritmos com a finalidade de demonstrar a melhoria e as vantagens e desvantagens de se utilizar o BOA para resolver o PET.

O sistema de Transmissão GARVER (GARVER, 1970), o qual está representado pelo diagrama unifilar da Figura 8. Este sistema possui 6 barras com 15 ramos (que são os nós para a RB) candidatas à expansão onde cada ramo pode receber no máximo 5 reforço. Foram considerados para o problema, redes com despacho, sem perdas e adaptativo. A Tabela 1 e Figura 9 são os 10 melhores planos para a solução geral. Os testes para o Garver consideram as configurações para a construção da RB já mencionado com 50 rodadas. O custo é o valor atribuído aos investimentos necessários para PET dado em milhões de dólares.

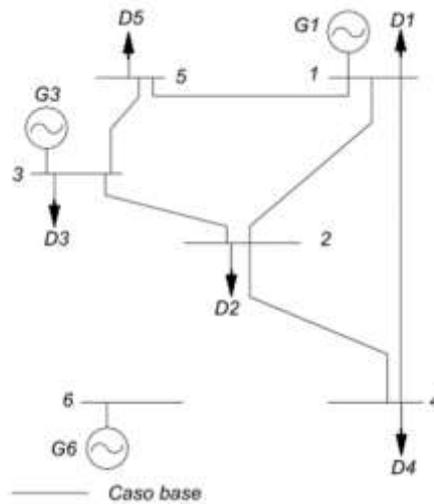


Figura 8: Diagrama unifilar para o sistema GARVER – topologia original

Fonte: VILLAJUAN MONTES, 2016

Planos	Custo	B1B3	B2B3	B2B4	B2B6	B3B5	B3B6	B4B6
1	110	0	0	0	0	1	0	3
2	130	0	0	0	1	2	0	2
3	130	0	0	0	3	2	0	0
4	130	0	1	0	1	1	0	2
5	140	0	0	0	2	1	0	2
6	148	1	0	0	1	1	0	2
7	148	0	1	0	0	1	1	2
8	148	1	0	0	3	1	0	0
9	150	0	0	1	1	1	0	2
10	160	0	1	0	3	1	0	1

Tabela 1: Os 10 melhores planos para o Garver

Fonte: Autores

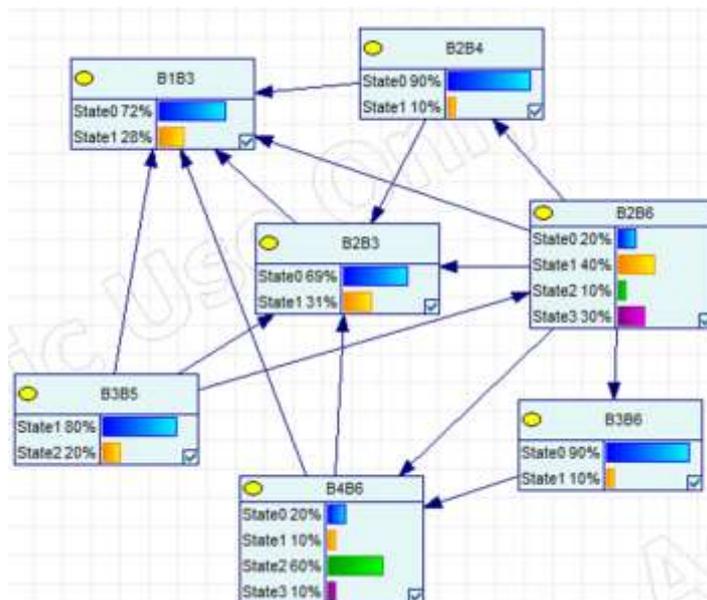


Figura 9: RB dos 10 melhores planos para o Garver.

Fonte: Elaboração própria.

Custo	B1B2	B1B3	B1B4	B1B5	B1B6	B2B3	B2B4	B2B5	B2B6	B3B4	B3B5	B3B6	B4B5	B4B6	B5B6
130	0	0	0	0	0	0	0	0	0	0	2	0	0	3	0

Figura 10: Plano que deveria estar entre os dez melhores

Fonte: Autores

A Tabela 1 apresenta os 10 melhores planos conhecidos para o Garver e que foram utilizados como base. O BOA original não foi capaz de encontrar todos os 10 melhores, mas ambos os algoritmos de mutação conseguiram encontrar. Além dos 10 melhores em algum momento o algoritmo de mutação pode encontrar um plano que se encaixa na solução dos 10 melhores e não é um subplano de outros existentes na solução. Este plano tem um custo de 130 e ao entrar na lista o plano com valor de 160 sai do top 10 e assim temos um novo top. Os impactos dessa mudança são: o algoritmo se mostra de confiança pois conseguiu mapear uma solução desconhecida até então sem a técnica de busca por exaustão, um top dez que detalha melhor o cenário, fornecendo informações mais detalhadas onde o planejador tem todas as possíveis escolhas dentro de um espaço de busca (top 10 no caso). O Garver pode ser um problema simples em comparação ao IEEE, mas isso não impediu que outros algoritmos não encontrassem este plano. Para o Garver as três implementações conseguiram convergir rapidamente.

Os gráficos abaixo Figuras 11, 12, 13 e 14 respectivamente, são os gráficos de convergência para o Garver, comparando as variáveis de mutação e o comportamento. A linha de cor vermelha e azul é o erro da média (todas as rodadas) do melhor plano de todas as rodadas em cada geração. A sombra superior é o erro do pior plano de todas as rodadas, e a inferior é o erro do melhor plano de todas as rodadas, ambos em cada geração.

Olhando para o gráfico pode se notar facilmente a melhoria que a mutação causou no algoritmo, reduzindo o erro, acelera a convergência e proporcionou um melhor espalhamento que permite buscar informações de forma mais homogênea, evitando convergir prematuramente em melhores locais. A Figura 12 ocorre uma flutuação que aparentemente pode ter ocorrido por algum ruído que foi eliminado ao aumentar a taxa de mutação Figura 13. Pelo gráfico pode se notar a parte inferior toca o zero bem antes da estagnação e com valores próximos de 40 e com populações bem pequenas o que pode ser interessante pensando em algoritmo de busca para problemas complexos onde a solução é desconhecida.

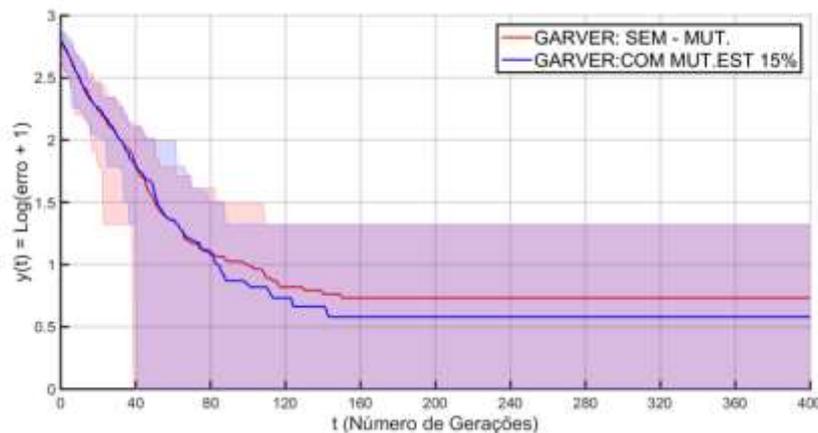


Figura 11: Gráfico de convergência sem mutação X mutação estática 15%

Fonte: Autores

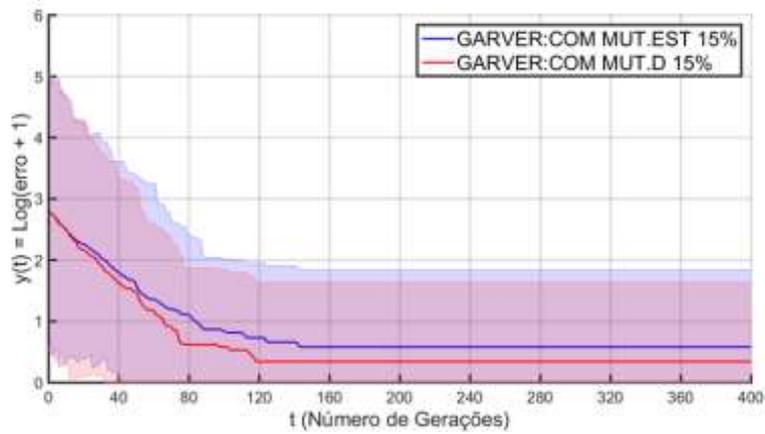


Figura 12: Gráfico de convergência mutação estática 15% X mutação dinâmica 15%

Fonte: Autores

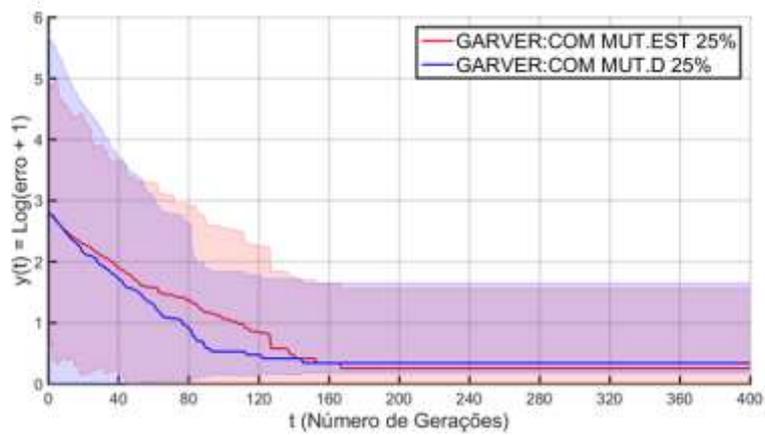


Figura 13: Gráfico de convergência mutação estática 25% X mutação dinâmica 25%

Fonte: Autores

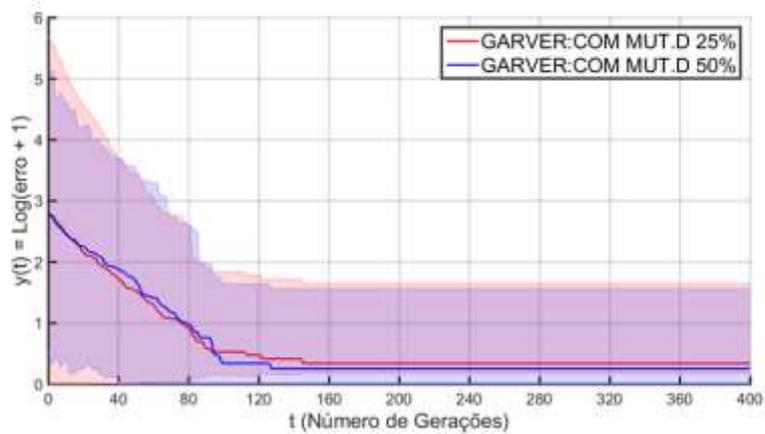


Figura 14: Gráfico de convergência mutação dinâmica 25% X mutação dinâmica 50%

Fonte: Autores

Comparando os algoritmos de mutação através do gráfico de convergência pode se notar que a mutação dinâmica é melhor que a estática, por possuírem um erro menor e convergir antes dos demais. A Figura 13 ocorre uma flutuação que aparentemente pode ter ocorrido por algum ruído que foi eliminado ao aumentar a taxa de mutação Figura 14. Na Figura 14 a mutação levou o algoritmo a convergir mais tarde, porém o erro e a estabilidade do erro foram melhores em comparação com a mutação dinâmica com 25% e isso ocorre justamente pelo motivo já mencionado (melhor espalhamento das buscas).

O sistema de Transmissão IEEE RTS-79 (IEEE24, 1979), o qual está representado pelo diagrama unifilar da Figura 15. O sistema possui 24 barras com 41 ramos (que são os nós para a RB) candidatas à expansão onde cada ramo pode receber no máximo 5 reforço. Foram considerados para o problema, redes com despacho, sem perdas e adaptativo. A Tabela 2 e Figura 16 são os 10 melhores planos para a solução geral (FREIRE, 2016). Os testes para o IEEE consideram as configurações para a construção da RB já mencionado.

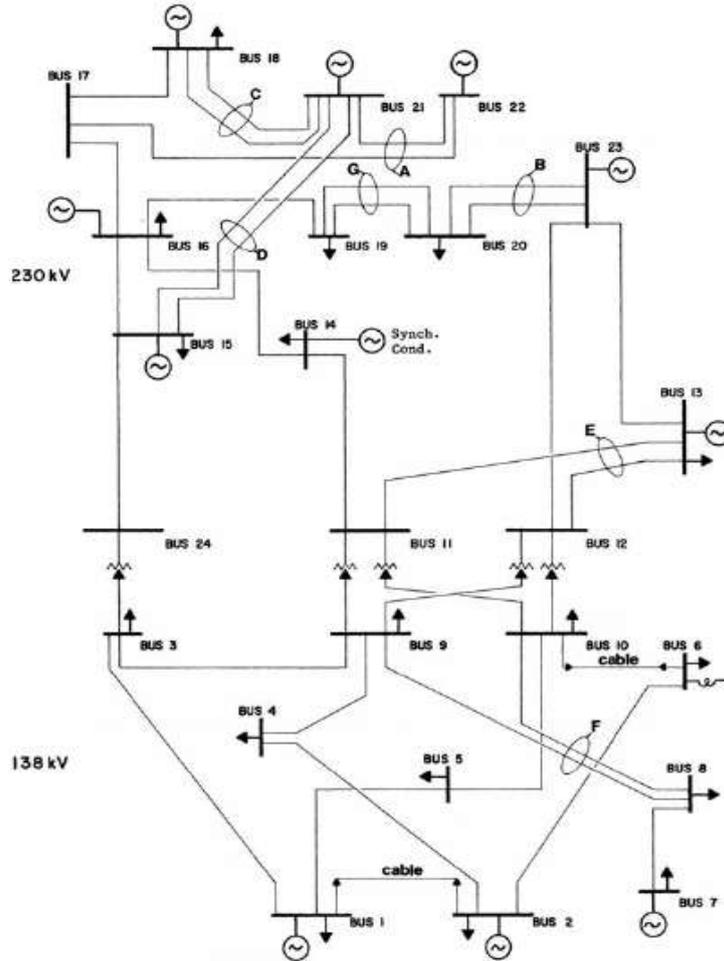


Figura 15: Diagrama unifilar para o sistema IEEE – topologia original

Fonte: VILLAJUAN MONTES, 2016

Planos	Custo	B1B2	B1B5	B4B9	B5B10	B9B12	B10B12	B11B13	B14B16	B15B16	B17B18	B13B14
1	152	0	0	0	0	0	1	0	1	0	0	0
2	163	1	0	0	0	0	1	0	0	0	0	1
3	164	0	0	0	0	0	1	1	0	0	0	0
4	174	0	1	0	0	1	0	0	1	0	0	0
5	180	0	0	0	0	0	1	0	0	0	1	1
6	182	0	1	0	0	0	1	0	0	0	0	1
7	183	0	0	0	1	0	1	0	0	0	0	1
8	184	0	0	0	0	0	1	0	0	1	0	1
9	185	1	1	0	0	1	0	0	0	0	0	1
10	187	0	0	1	0	0	1	0	0	0	0	1

Tabela 2: Os 10 melhores planos conhecidos para o IEEE RTS – 79

Fonte: Autores

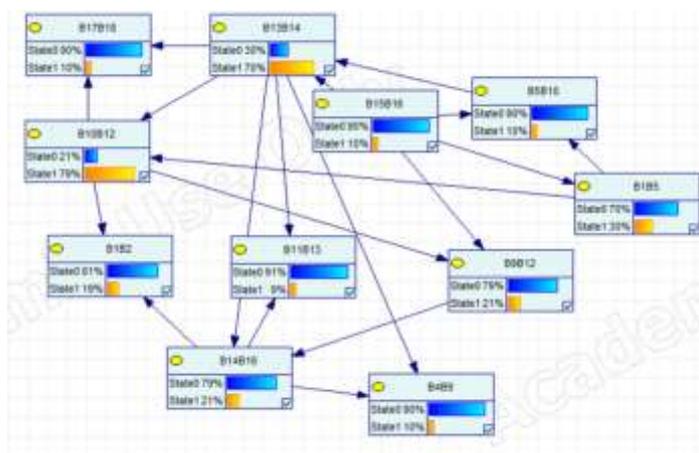


Figura 16: RB dos 10 melhores planos conhecidos para o IEEE

Fonte: Autores

A Tabela 2 e Figura 16 representam os 10 planos conhecidos, que foram utilizados como base. Para compilar o IEEE foi utilizado somente 4 rodadas, pois o problema é muito complexo e a RB muitas vezes tomava tempo exagerado para ser gerado e como o processo à cada geração gera uma nova RB tornou o processo lento (o que pode ser contornado com uma otimização do algoritmo responsável pela RB).

O resultado para o IEEE não foi diferente do Garver, onde a Tabela 3 foi encontrado um novo top 10 com um custo e expansão de ramos diferentes do da Tabela 2 (alguns podem se repetir, porém foi encontrado vários planos melhores em relação ao custo). Este resultado da Tabela 3 pode ser incorporado na Tabela 2 e formar um novo conjunto de 10 melhores. Uma nova RB pode ser gerada com esse novo top 10, com novas possibilidades para o planejador Figura 17.

Planos	Custo	B1B2	B1B5	B6B10	B7B8	B9B12	B10B12	B11B13	B14B16	B13B14	B14B23
1	152	0	0	1	2	0	1	0	1	0	0
2	155	1	0	1	2	0	1	0	1	0	0
3	160	0	0	1	2	0	1	0	0	1	0
4	164	0	0	1	2	0	1	1	0	0	0
5	168	0	0	1	3	0	1	0	1	0	0
6	170	2	0	1	2	0	1	1	0	0	0
7	172	4	0	1	2	0	1	0	0	1	0

8	174	2	0	2	2	0	1	0	1	0	0
9	177	1	1	1	2	1	0	0	1	0	0
10	180	0	0	1	3	0	1	1	0	0	0
11	184	0	0	1	2	0	1	0	0	0	1
12	184	0	0	3	2	0	1	0	1	0	0

Tabela 3: RB dos melhores planos descobertos pelo BOA

Fonte: Autores

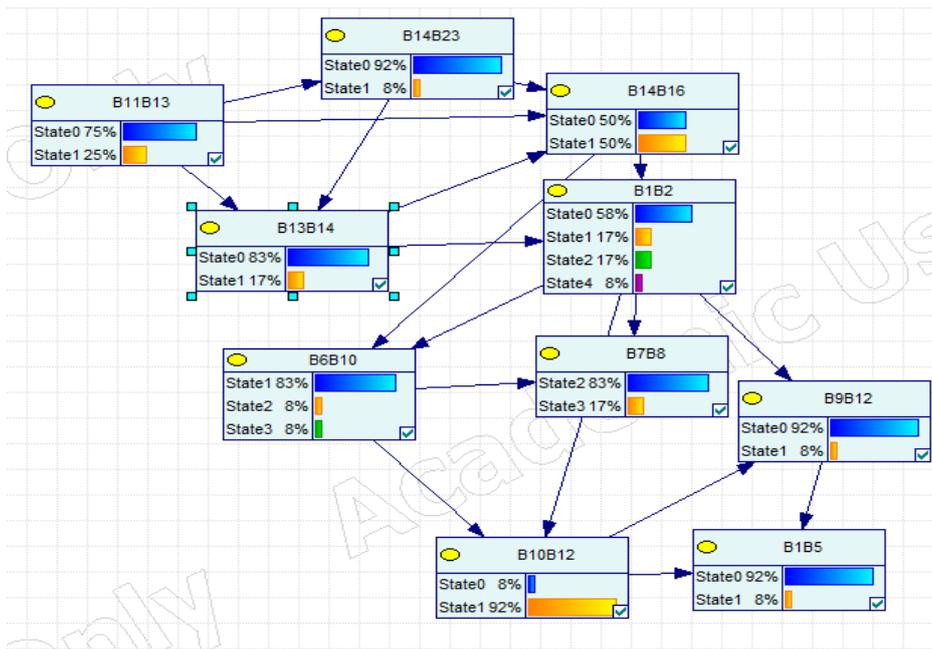


Figura 17: RB dos melhores planos descobertos pelo BOA

Fonte: Autores

Para o estudo de caso foi realizado uma análise na variável Tamanho da Amostra (TA). Alterando o valor de TA pode se notar uma variação no gráfico de convergência Figura 18.

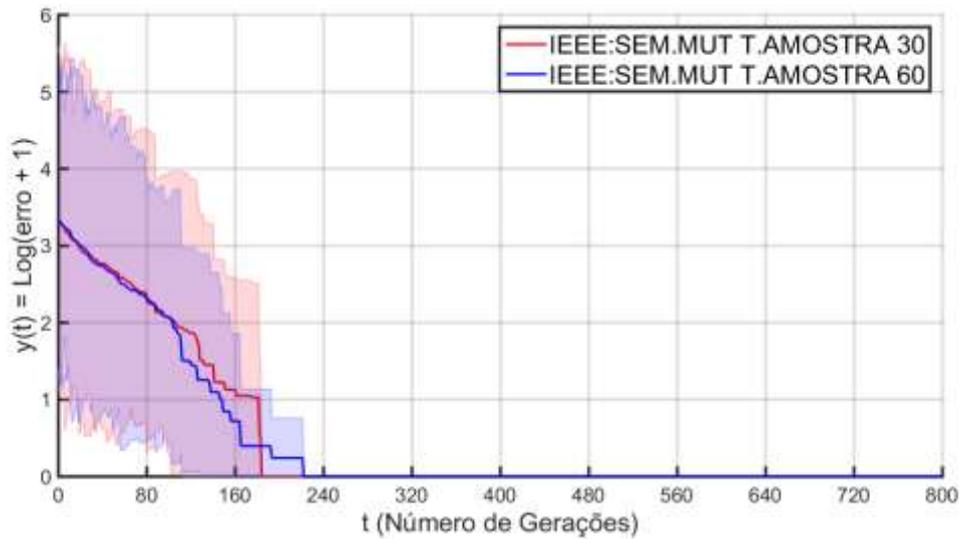


Figura 18: Gráfico de convergência sem mutação TA 30 x TA 60

Fonte: Autores

Nota-se que para o mesmo problema os algoritmos com TA maior possui maior espalhamento e pode encontrar o melhor global antes do TA menor (Figura 18 e 20). Outro comparativo é dos algoritmos BOA, BOAME e BOAMD; na Figura 19 o número de gerações necessárias para a estagnação do algoritmo com mutação dinâmica foi muito inferior ao algoritmo de mutação estática em um teste realizado para comprar os dois.

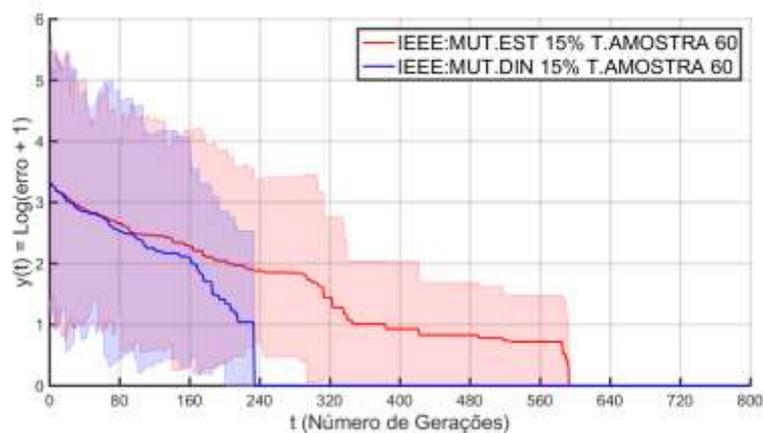


Figura 19: Gráfico de convergência BOAME 15% x BOAMD 15%

Fonte: Autores

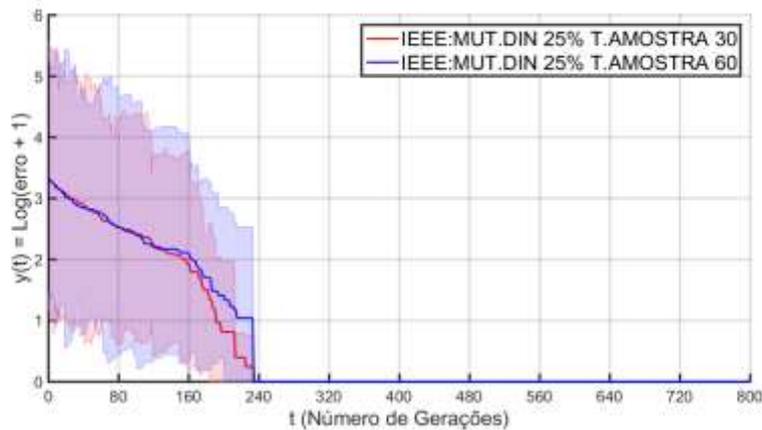


Figura 20: Gráfico de convergência BOAMD 25% x BOAMD

Fonte: Autores

A Figura 20 faz um comparativo do mesmo algoritmo utilizando mutação comparando TA e são similares em tanto em erro quanto em espalhamento dos resultados. A mutação pode ter deixado o algoritmo menos sensível à variações da configuração da RB. Isto é um resultado positivo, pois para problemas mais complexos não se sabe qual é o melhor cainho para se escolher, tornando a mutação dinâmica mais à mais genérica e indicada mais indicada para se resolver qualquer problema e tem mais chance de encontrar os 10 melhores globais para o PET.

## CONCLUSÃO

Os resultados para ambos os algoritmos são satisfatórios. Com estas ferramentas pode-se aplicar em outras áreas com as devidas adaptações. O único problema encontrado em se utilizar a ferramenta é a otimização da criação da RB.

O aprendizado via BOA com mutação pode melhorar os resultados e deixar o algoritmo mais robusto tornando ainda mais viável a utilização do algoritmo com mutação e mais confiável, além de retornar um conjunto de soluções que podem ser o conjunto de melhores globais nas quais outros algoritmos não foram capazes de mapear o mesmo e a rede ainda propicia um ambiente gráfico e amigável para o planejador utilizar seus conhecimentos para extrair informações probabilísticas da rede. As redes apresentadas ao final podem ser utilizadas por planejadores que atuam no setor elétrico.

O sistema de potência estudado é um modelo acadêmico (Garver e IEEE) então para pesquisas futuras um modelo real e mais complexo pode ser fonte de pesquisas futuras. Pelos resultados apresentados a configuração mais indicada para o PET é o BOAMD por ser menos suscetível a variações e não sofrer tanto com as configurações da RB impostas para resolver o problema.

## **REFERÊNCIAS**

BOUCKAERT, Remco R. Properties of Bayesian belief network learning algorithms. In: Uncertainty Proceedings 1994. Morgan Kaufmann, 1994. p. 102-109.

COOPER, Gregory F.; HERSKOVITS, Edward. A Bayesian method for the induction of probabilistic networks from data. *Machine learning*, v. 9, n. 4, p. 309-347, 1992.

FREIRE, Muriell de Rodrigues e. Algoritmo Evolutivo adaptativo via Multioperadores aplicado ao planejamento da Expansão de Sistemas de Transmissão. 2016. 199 f. Tese (Doutorado em Engenharia Elétrica) – Universidade Federal de Itajubá, Itajubá, 2016.

GARVER, Len L. Transmission network estimation using linear programming. *IEEE Transactions on Power Apparatus and Systems*, n. 7, p. 1688-1697, 1970.

GENIE. Versão 2.4. 2.4. ed. University of Pittsburgh: BayesFusion, LLC, 2019. Disponível em: < <https://www.bayesfusion.com/> >. Acesso em: 23 ago. 2019.

IEEE APM SUBCOMMITTEE. IEEE Reliability Test System. *IEEE Transactions on Power Apparatus and Systems*, v. 1, n. 6, p. 2047-2054, 1979.

LEITE DA SILVA, A. M. et al. Performance comparison of metaheuristics to solve the multi-stage transmission expansion planning problem. *IET Generation, Transmission & Distribution*, v. 5, n. 3, p. 360, 2011.

LEITE DA SILVA, A. M.; FREIRE, M. R.; HONÓRIO, L. M. Transmission expansion planning optimization by adaptive multi-operator evolutionary algorithms. *Electric Power Systems Research*, v. 133, p. 173–181, abr. 2016.

PELIKAN, M., GOLDBERG, D. E., & CANTÚ-PAZ, E. (1998). Linkage problem, distribution estimation, and Bayesian networks. IlliGAL Report No. 98013: Illinois Genetic Algorithm Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL.

PELIKAN, Martin; GOLDBERG, David E.; CANTÚ-PAZ, Erick. BOA: The Bayesian optimization algorithm. In: Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation-Volume 1. Morgan Kaufmann Publishers Inc., 1999. p. 525-532.

REGOLIN, EVANDRO NUNES. Programação genética e algoritmos de estimação de distribuição. 2004. 84 f. Dissertação (Mestrado em Informática) - Universidade Federal do Paraná, Departamento de Informática do Setor de Ciências Exatas da Universidade Federal do Paraná, 2004.

VILLAJUAN MONTES, Cristian Victor. Modelos de programação linear inteira mista para resolver problemas de planejamento da expansão e da operação de sistemas de transmissão de energia elétrica. 2016.

***Application of the Bayesian Optimization Algorithm with mutation on the Planning the Expansion of Power Transmission Systems***

**ABSTRACT**

*This work aims to apply the Bayesian Optimization Algorithm (BOA) to the Transmission Expansion Planning (PET) problem. The formulation of PET will be based on short term planning, considering the "N-0" (the intact network is considered for planning, that is, all transmission elements operating normally) safety criteria. The best solutions to the problem were found through a specialized evolutionary algorithm and used as reference for the implementation of the BOA algorithm. From an adaptation of the BOA, a mutation operator was implemented to generate more diversity and thus improve searches. The analyses were made comparing the original BOA with the modified version. The comparison between the algorithms was done using a graphical method (convergence graph) comparing the cases for discussion. The algorithm can be used not only for PET, but also for other problems that involve evolutionary algorithm.*

**KEYWORDS:** *Evolutionary Algorithm. Mutation. Bayesian Networks. Short Term Transmission Network Expansion Planning.*

**Enviado em: 28/12/2020**