

Algoritmo *Grey Wolf Optimizer* na resolução do problema de Fluxo de Potência Ótimo Reativo com variáveis de controle discretas

Leonardo Pereira de Almeida Campos

Universidade Estadual Paulista "Júlio de Mesquita Filho"

Av. Eng. Luís Edmundo Carrijo Coube, 14-01 - Vargem Limpa, Bauru - SP, 17033-360

leonardo.pa.campos@unesp.br

Edilaine Martins Soler

Universidade Estadual Paulista "Júlio de Mesquita Filho"

Av. Eng. Luís Edmundo Carrijo Coube, 14-01 - Vargem Limpa, Bauru - SP, 17033-360

edilaine.soler@unesp.br

RESUMO

O Problema de Fluxo de Potência Ótimo (FPO) determina o estado de uma rede elétrica de potência para desempenho ótimo, satisfazendo suas restrições físicas e operacionais. Um caso particular deste problema é o Fluxo de Potência Ótimo Reativo (FPOR), que pode ser modelado matematicamente como um problema de otimização não linear, não-convexo, restrito e com variáveis discretas e contínuas. Na maioria dos trabalhos da literatura, as variáveis discretas são tratadas como contínuas, dada a dificuldade de solucionar o problema. Neste trabalho é proposta uma abordagem de solução para o problema de FPOR considerando a natureza discreta de algumas variáveis. Foi utilizada uma função penalidade, associada ao algoritmo meta-heurístico *Grey Wolf Optimizer* (GWO), para tratar as variáveis discretas do problema. Testes numéricos feitos com os sistemas benchmark IEEE de 14 e 30 barras demonstraram o potencial da abordagem proposta em obter soluções discretas de boa qualidade.

PALAVRAS CHAVE. Fluxo de Potência Ótimo Reativo, Variáveis Discretas, Grey Wolf Optimizer.

EN&PG - PO na Área de Energia, Petróleo e Gás

ABSTRACT

The Optimal Power Flow (OPF) problem determines the state of an electric power grid for optimal performance, satisfying its physical and operational constraints. A particular case of that problem is the Optimal Reactive Power Dispatch (ORPD), which can be mathematically modelled as a nonlinear, non-convex, constrained, with discrete and continuous variables optimization problem. Most approaches in the literature treat the discrete variables as continuous, given the difficulty to solve the problem. This work proposes a solution approach for the ORPD problem considering the discrete nature of some of its variables. A penalty function, associated to the Grey Wolf Optimizer (GWO) meta-heuristic algorithm, was used to manage the discrete variables. Numerical tests conducted on the 14 and 30 bus IEEE benchmark systems demonstrated the potential of the proposed approach to obtain good quality discrete solutions.

KEYWORDS. Optimal Reactive Power Dispatch, Discrete Variables, Grey Wolf Optimizer.

EN&PG - OR on Energy, Oil and Gas

1. Introdução

A civilização humana é, na contemporaneidade, extremamente dependente da energia elétrica em todos seus setores: no transporte, na saúde, na alimentação, no trabalho e no seu desenvolvimento, de forma geral. Para que a utilização dessa energia seja possível, é necessário a existência dos sistemas elétricos de potência (SEP), que englobam a geração, a transmissão e a distribuição da eletricidade. Além disso, para garantir a eficiência e a qualidade dessa energia é necessário garantir que esses sistemas operem da melhor maneira possível.

O estado ótimo instantâneo de um sistema elétrico de potência - aquele em que o sistema apresenta o melhor desempenho possível - pode ser determinado pela resolução do problema de Fluxo de Potência Ótimo (FPO), o qual é formulado como um problema de otimização [Papalexopoulos et al. 1989]. O problema de FPO foi inicialmente proposto na década de 60 [Carpentier 1962] e uma grande quantidade de trabalhos científicos foram propostos desde então, com diferentes formulações e abordagens de solução.

O tratamento eficiente das variáveis discretas do problema de FPO é reconhecido como desafiador e tem sido abordado desde a década de 80 [Tinney et al. 1988], [Papalexopoulos et al. 1989].

O problema de Fluxo de Potência Ótimo Reativo (FPOR) é um caso particular do problema de FPO em que as variáveis de controle referentes à potência ativa são fixadas e as variáveis de controle referentes à potência reativa são ajustadas de forma a otimizar um desempenho do sistema. Este problema é formulado como um problema de programação não-linear discreto misto (PNLDM), com sua formulação envolvendo funções não convexas, e variáveis discretas e contínuas e é considerado de difícil resolução.

Devido à dificuldade de solução imposta pelas variáveis discretas do problema, em muitas das abordagens da literatura que tratam da resolução de problemas de FPO os controles discretos são modelados como variáveis contínuas. Estas formulações estão longe da realidade de um sistema elétrico de potência, pois alguns controles podem somente ser ajustados por passos discretos. Soluções obtidas a partir do arredondamento das variáveis discretas, tratadas como contínuas, para os valores discretos mais próximos são aceitáveis para variáveis discretas com passos pequenos, como é o caso dos *taps* dos transformadores, mas não funcionam bem para variáveis discretas com passos maiores, como é o caso das susceptâncias dos bancos de capacitores e reatores *shunt*. O arredondamento pode gerar soluções não-ótimas e até mesmo infactíveis.

Determinar um método de solução para problemas de FPO que seja viável para ser utilizado em tempo real e que considere a natureza das variáveis inteiras e discretas destes problemas ainda é um desafio mesmo depois de mais de 50 anos da formulação destes problemas. Operadores de sistemas elétricos utilizam métodos de solução que fazem diversas aproximações, ao custo de dezenas de bilhões de dólares anualmente e a danos ambientais desnecessários por energia desperdiçada [Cain et al. 2012].

Algoritmos meta-heurísticos têm sido muito utilizados na resolução de problemas de FPO [Khamees et al. 2016], entre outros problemas de otimização de difícil resolução. Dentre os algoritmos meta-heurísticos destacamos o algoritmo *Grey Wolf Optimizer (GWO)*, que é um algoritmo estocástico baseado em população desenvolvido recentemente [Mirjalili et al. 2014], inspirado no comportamento social e hierárquico de caça das alcateias de lobos cinzentos.

O algoritmo GWO tem sido aplicado com sucesso em problemas de FPO contínuos [Majumdar et al. 2017]; [Kamboj et al. 2016]; [Jayabarathi et al. 2016], no problema de FPO multi-objetivo [El-Fergany e Hasanién 2015], e com ótimos resultados para o problema de Fluxo de Potência Ótimo Reativo [Sulaiman et al. 2015]. O algoritmo GWO também tem sido aplicado a

problemas de otimização das mais diversas áreas da engenharia [Mirjalili 2015]; [Mohanty et al. 2015]; [Mittal et al. 2016]; [Emary et al. 2016]; [Khalilpourazari e Khalilpourazary 2018].

O algoritmo GWO foi proposto originalmente para problemas com variáveis contínuas. Neste trabalho propõe-se adaptá-lo para a resolução do problema de FPOR com variáveis de controle discretas incorporando a função penalidade senoidal [Soler et al. 2013] na função *fitness*.

Este trabalho está organizado como segue: na seção 2 é descrita a formulação matemática do problema de Fluxo de Potência Ótimo Reativo; na seção 3 é descrito o algoritmo *Grey Wolf Optimizer*; na seção 4 é descrita a abordagem proposta para a solução do problema de FPOR; na seção 5 estão os resultados numéricos obtidos; na seção 6 é apresentada a conclusão deste trabalho.

2. O problema de Fluxo de Potência Ótimo Reativo

O problema de Fluxo de Potência Ótimo Reativo é modelado matematicamente como um problema de programação não linear com variáveis discretas e contínuas e pode ser representado como:

$$\begin{aligned} \min \quad & \sum_{k,m \in \Omega} g_{km} \left(\frac{1}{t_{km}^2} V_k^2 + V_m^2 - 2 \frac{1}{t_{km}} V_k V_m \cos \theta_{km} \right) \\ \text{s.a. : } \quad & \begin{cases} \Delta P_i(V, \theta, t) = 0, & \forall i \in B_{CR} \cup B_C \\ \Delta Q_j(V, \theta, t, b^{sh}) = 0, & \forall j \in B_C \\ \underline{Q}_k \leq Q_k(V, \theta, t, b^{sh}) \leq \overline{Q}_k, & \forall k \in B_{CR} \\ \underline{V}_m \leq V_m \leq \overline{V}_m, & \forall m \in B \\ t_l \in D_{t_l}, & \forall l \in T \\ b_q^{sh} \in D_{b_q^{sh}}, & \forall q \in B_{SS} \end{cases} \end{aligned} \quad (1)$$

Em que:

- **Conjuntos:** B_{CR} é o conjunto das barras com controle de reativos; B_C é o conjunto das barras de carga; B é o conjunto de todas as barras do sistema; T é o conjunto dos transformadores com controle de *tap*; B_{SS} é o conjunto de barras com bancos de capacitores ou reatores *shunt*; D_{t_l} é o conjunto de valores discretos que o *tap* do transformador l pode assumir; $D_{b_q^{sh}}$ é o conjunto de valores discretos que a susceptância *shunt* conectada à barra q pode assumir; Ω é o conjunto dos ramos do sistema; Ω_k , o conjunto de todas as barras vizinhas à barra k ;
- **Parâmetros:** $\underline{Q}_k$ e $\overline{Q}_k$ são os limites mínimo e máximo, respectivamente, de geração de potência reativa; $\underline{V}_m$ e $\overline{V}_m$ são os limites mínimo e máximo, respectivamente, das magnitudes das tensões nas barras; g_{km} , b_{km} , b_{km}^{sh} , a condutância e as susceptâncias da linha km; P_k^G , P_k^C , as potências ativas gerada e consumida, respectivamente; Q_k^G , Q_k^C , as potências reativas gerada e consumida, respectivamente; $t_{km} = 1$, caso as barras k e m estejam conectadas por uma linha de transmissão.
- **Variáveis de decisão:** As variáveis de decisão são: V é o vetor das magnitudes de tensão das barras do sistema; θ é o vetor dos ângulos de tensão nas barras; t é o vetor dos *taps* dos transformadores com controle de *tap*; b^{sh} é o vetor das susceptâncias *shunt*;

- Balanço de potência ativa pra as barras de carga e de controle reativo:

$$\Delta P_i(V, \theta, t) = P_i^G - P_i^C - \sum_{m \in \Omega_i} P_{im}(V, \theta, t) = 0 \quad (2)$$

em que,

$$P_{im}(V, \theta, t) = \begin{cases} g_{im} \frac{1}{t_{im}^2} V_i^2 \\ -\frac{1}{t_{im}} V_i V_m (g_{im} \cos \theta_{im} + b_{im} \sin \theta_{im}), & \text{se } i \text{ é um nó inicial} \\ g_{im} V_i^2 \\ -\frac{1}{t_{im}} V_i V_m (g_{im} \cos \theta_{im} - b_{im} \sin \theta_{im}), & \text{se } i \text{ é um nó final} \end{cases} \quad (3)$$

- Balanço de potência reativa nas barras de carga:

$$\Delta Q_j(V, \theta, t) = Q_j^G - Q_j^C - b_k^{sh} V_k^2 - \sum_{m \in \Omega_j} Q_{jm}(V, \theta, t, b^{sh}) = 0 \quad (4)$$

em que,

$$Q_{jm}(V, \theta, t) = \begin{cases} -\left(b_{jm} \frac{1}{t_{jm}^2} + b_{jm}^{sh}\right) V_j^2 \\ +\frac{1}{t_{jm}} V_j V_m (b_{jm} \cos \theta_{jm} - g_{jm} \sin \theta_{jm}), & \text{se } j \text{ é um nó inicial} \\ -(b_{jm} + b_{jm}^{sh}) V_j^2 \\ +\frac{1}{t_{jm}} V_j V_m (b_{jm} \cos \theta_{jm} + g_{jm} \sin \theta_{jm}), & \text{se } j \text{ é um nó final} \end{cases} \quad (5)$$

- Geração de potência reativa injetada nas barras de controle de reativo:

$$Q_k(V, \theta, t) = Q_k^C - b_k^{sh} V_k^2 + \sum_{m \in \Omega_k} Q_{km}(V, \theta, t, b^{sh}) \quad (6)$$

de forma que $Q_{km}(V, \theta, t, b^{sh})$ é dado pela equação (5).

Uma das grandes dificuldades na resolução do problema de Fluxo de Potência Ótimo Relativo (FPOR), além da não convexidade e não linearidade das funções envolvidas, é a existência de variáveis que só podem assumir valores discretos: os *taps* dos transformadores, que assumem valores discretos com tamanho de passo constante (ou seja, cada possível valor é igualmente espaçado dentro de um intervalo); e os valores das susceptâncias *shunt* dos bancos de capacitores e reatores, que assumem valores discretos com tamanhos de passos distintos. Propõe-se neste trabalho resolver o problema de FPOR utilizando-se o algoritmo *Grey Wolf Optimizer* associado a funções penalidade para o tratamento das variáveis discretas do problema.

3. O algoritmo *Grey Wolf Optimizer*

O Algoritmo *Grey Wolf Optimizer* (GWO) é um algoritmo meta-heurístico que realiza uma busca em um espaço multidimensional simulando o comportamento hierárquico de caça de uma alcateia de lobos cinzentos (*Canis lupus*), foi desenvolvido por [Mirjalili et al. 2014] e aplicado com sucesso ao problema de FPOR com variáveis contínuas por [Sulaiman et al. 2015].

No algoritmo, a busca se inicia com cada lobo em uma posição aleatória dentro do espaço de buscas multidimensional. Então, a função objetivo do problema é calculada para cada lobo, e aquele com o menor valor numérico para a função objetivo (melhor solução até então) é designado como alfa, em seguida o beta, em terceiro lugar o delta (lobos principais) e os lobos restantes são designados como ômegas (lobos caçadores). A partir disso os lobos mudam suas posições de acordo com a posição do alfa, do beta e do delta a cada iteração. Quando o critério de parada é atendido (numero máximo de iterações é atingido), a solução é dada pela última posição do lobo alfa.

Cada lobo (agente de busca) é definido pelo vetor $W_i(t) = [w_{i1}(t), w_{i2}(t), \dots, w_{in}(t)]$, onde t é a iteração atual do algoritmo e $w_{ij}(t)$ é a j -ésima coordenada do i -ésimo lobo na iteração t . As equações que regem o comportamento do algoritmo são:

$$a(t) = 2 - t \frac{2}{t_{max}} \quad (7)$$

$$A_{p,i}(t) = 2a(t)r_1 - a(t) \quad (8)$$

$$C_{p,i}(t) = 2r_2 \quad (9)$$

$$D_{p,i}(t) = |C_{p,i}p_j(t) - w_{ij}(t)| \quad (10)$$

$$X_{p,i}(t) = p_j(t) - A_{p,i}D_{p,i} \quad (11)$$

$$w_{ij}(t+1) = \frac{X_{\alpha,i}(t) + X_{\beta,i}(t) + X_{\delta,i}(t)}{3} \quad (12)$$

Onde, $p = \alpha, \beta, \delta$; $p_j(t)$ é a j -ésima coordenada do lobo p (sendo este um dos três principais) na iteração t ; $i = 1, 2, \dots, n_{lobos}$, onde n_{lobos} é o número de agentes de busca; r_1 e r_2 são números aleatórios distribuídos uniformemente em $[0, 1]$; e t_{max} é o número máximo de iterações.

O parâmetro $a(t)$, dado pela equação (7), decresce linearmente de 2 até 0 ao longo das iterações. Sua função é variar o parâmetro $A_{p,i}(t)$, dado pela equação (8), de forma que: se $t \leq \frac{t_{max}}{2}$, o algoritmo estará na fase de "*exploration*", na qual os lobos devem divergir das posições uns dos outros, no intuito de explorar o máximo possível do espaço de buscas e, assim, evitar ótimos locais; se $t > \frac{t_{max}}{2}$, o algoritmo entra na fase de "*exploitation*", na qual os lobos convergirão para as posições do alfa, beta ou delta a cada iteração, no intuito de explorar uma região menor delimitada pelos lobos com melhores resultados.

O parâmetro $C_{p,i}(t)$ é aleatório e varia no intervalo $[0, 2]$ de acordo com o valor de r_2 , sua função é servir como um "peso" para as coordenadas dos lobos alfa, beta e delta dentro da equação do parâmetro $D_{p,i}(t)$, que atua como uma "distância" entre o lobo p e o lobo i . Caso $C_{p,i}(t) \leq 1$, o lobo i se afastará da posição do lobo p na próxima iteração. Caso $C_{p,i}(t) > 1$, o lobo i convergirá para a posição do lobo p na próxima iteração. Dessa forma, há um fator de aleatoriedade que

regula a troca de posições entre os lobos caçadores e os lobos principais, aumentando a eficácia do algoritmo para evitar pontos de ótimos locais.

Por fim, o parâmetro $X_{p,i}(t)$ une os demais e funciona de modo a determinar se a posição do lobo i na iteração $t + 1$ tenderá mais para a posição do lobo alfa, beta ou delta na iteração t , através da equação (12), que serve para atualizar a posição de todos os lobos a cada iteração.

O algoritmo (1) ilustra o algoritmo *Grey Wolf Optimizer*, baseado em [Mirjalili et al. 2014] e [Nuaekaew et al. 2017]

Algoritmo 1: Algoritmo *Grey Wolf Optimizer*

Entrada: n_{lobos} , t_{max} , dados do sistema

Saída: Posição do lobo α para $t = t_{max}$, sendo esta a melhor solução encontrada

início

 Inicialização dos lobos $W_i(t)$, $i \in [1, n_{lobos}]$ e do parâmetro $a(t)$;

while $t < t_{max}$ **do**

 Calcular a função objetivo para todos os lobos;

 Determinar os agentes α , β e δ ;

 Calcular $D_{p,i}$, $A_{p,i}$, $C_{p,i}$, $X_{p,i}$, para $i \in [1, n_{lobos}]$ e $p = \alpha, \beta, \delta$;

 Atualizar a posição de cada lobo pela equação (12);

 Atualizar $a(t)$;

end

fim

4. Algoritmo proposto

O algoritmo *Grey Wolf Optimizer* não foi criado, originalmente, para lidar com problemas de otimização com restrições e nem problemas de PNLD. Para utilizá-lo para solucionar o problema de FPOR, a estratégia escolhida foi incorporar as restrições de variáveis discretas através da função de penalização Senoidal, e a restrição dos limites das magnitudes das tensões nas barras de controle de reativos do sistema por meio de uma penalização estática dada pela equação (15). As demais restrições do problema de FPOR, dadas pelas equações de balanço do sistema elétrico, são garantidas a partir da resolução do problema de Fluxo de Carga (um sistema de equações não lineares) para cada lobo, utilizando a biblioteca *PandaPower*, da linguagem de programação *Python*.

A função de Penalização Senoidal, proposta por [Soler et al. 2013], $P_{sen}(y) : \mathbb{R} \mapsto \mathbb{R}$ é dada por (13). Seja D_{y_i} o conjunto dos valores discretos que a variável y_i pode assumir.

$$P_{sen}(y) = \sum_{i=1}^{n_y} \left[\sin \left(\frac{y_i}{y_i^{sup} - y_i^{inf}} \pi + \alpha_i \right) \right]^2 \quad (13)$$

Em que, y_i^{sup} é o valor discreto superior mais próximo de y_i ; y_i^{inf} é o valor discreto inferior mais próximo de y_i ; α_i é $(0 \leq \alpha_i \leq \pi)$ é determinado de forma que $P_{sen}(y) = 0$ se, e somente se, as variáveis y_i , com $i = 1, 2, \dots, n_y$, assumem valores discretos do conjunto D_{y_i} e é dado por (14).

$$\alpha_i = \left(\left\lceil \frac{y_i^{inf}}{y_i^{sup} - y_i^{inf}} \right\rceil - \frac{y_i^{inf}}{y_i^{sup} - y_i^{inf}} \right) \pi \quad (14)$$

O operador $\lceil n \rceil$ representa o valor arredondado para cima do número n . Por construção, a função (13) é nula se y_i assumir um valor discreto do intervalo D_{y_i} , para $i = 1, 2, \dots, n_y$, e é positiva caso contrário. Caso a variável y_i possua passo discreto constante, α_i também é constante.

A função de penalização estática, como proposto por [Coello 2002], para os limites das magnitudes de tensão das barras de controle de reativo do sistema é

$$P_V(V_{i \in B_{CR}}) = \sum_{i \in B_{CR}} (V_i - V_i^{lim})^2 \quad (15)$$

Em que,

$$V_i^{lim} = \begin{cases} \bar{V}_i, & \text{caso } V_i > \bar{V}_i \\ V_i, & \text{caso } \underline{V}_i < V_i < \bar{V}_i \\ \underline{V}_i, & \text{caso } V_i < \underline{V}_i \end{cases} \quad (16)$$

Incorporando todas as funções de penalização na função objetivo do problema, obtém-se a Função Aumentada, dada por (17):

$$FA(V, \theta, t, b^{sh}) = f(V, \theta, t) + \lambda_1 P_{sen}(t) + \lambda_2 P_{sen}(b^{sh}) + \lambda_3 P_V(V_{i \in B_{CR}}) \quad (17)$$

Em que $\lambda_1, \lambda_2, \lambda_3$ são parâmetros de penalização.

O algoritmo *Grey Wolf Optimizer* é utilizado, então, para minimizar a função (17), obtendo, assim, uma solução para o problema de FPOR com variáveis discretas definido por (1). O algoritmo proposto é ilustrado pelo algoritmo 2.

Algoritmo 2: Algoritmo proposto: *Grey Wolf Optimizer* associado a funções penalidade

Entrada: n_{lobos}, t_{max} , dados da rede elétrica

Saída: Posição do lobo α para $t = t_{max}$, sendo esta a melhor solução encontrada

início

 Inicialização dos lobos $W_i(t), i \in [1, n_{lobos}]$ e do parâmetro $a(t)$;

while $t < t_{max}$ **do**

if *Variáveis fora dos limites definidos* **then**

 Retornar as variáveis para os limites;

else

 Continuar

end

 Calcular a função aumentada dada por (17) para todos os lobos;

 Resolver o problema de Fluxo de Carga para cada lobo, mapeando suas posições no *PandaPower*;

 Determinar os agentes α, β e δ ;

 Calcular $D_{p,i}, A_{p,i}, C_{p,i}, X_{p,i}$, para $i \in [1, n_{lobos}]$ e $p = \alpha, \beta, \delta$;

 Atualizar a posição de cada lobo pela equação (12);

 Atualizar $a(t)$;

end

fim

5. Resultados numéricos

Foram realizados testes numéricos com o FPOR para os sistemas IEEE de 14 e 30 barras, disponíveis na biblioteca *PandaPower* [Thurner et al. 2018], o algoritmo foi implementado na linguagem de programação *Python*, versão 3.7 e os testes foram conduzidos em um computador *desktop* customizado com processador AMD Ryzen 5 1600, modelo AF, operando na frequência de 3.6GHz, e 8GB de memória RAM DDR4 operando na frequência de 2666 MHz.

As magnitudes de tensão das barras foram consideradas dentro do intervalo $[0,94; 1,06]pu$ os *taps* dos transformadores foram considerados dentro do intervalo discreto $[0,95; 1,05]pu$, com passo discreto de $0,01pu$.

5.1. Sistema elétrico IEEE de 14 barras

O problema de FPOR com variáveis discretas para o sistema elétrico IEEE de 14 barras possui: 22 restrições de igualdade; 38 restrições de desigualdade; 27 variáveis contínuas (relativas às magnitudes de tensão V e aos ângulos de tensão θ das barras); 4 variáveis discretas (*taps* t de três transformadores com controle de *tap* e uma susceptância *shunt* b^{sh}).

No algoritmo proposto, são otimizadas 5 variáveis contínuas: as tensões das 5 barras com controle de reativo do sistema ($V_{k \in B_{CR}}$); e as quatro variáveis discretas ($t_{l \in T}$ e $b_j^{sh} \in B_{SS}$). As demais variáveis contínuas são obtidas a partir da resolução do problema de Fluxo de Carga. Foi considerado que a susceptância *shunt* conectada à barra 9 deve pertencer ao seguinte conjunto discreto: $b_9^{sh} \in \{0; 0,19; 0,34; 0,39\}$

Os resultados expostos na tabela 1 foram os melhores obtidos dentro de 25 execuções do algoritmo, com $n_{lobos} = 12$, $\lambda_1 = \lambda_2 = \lambda_3 = 100$.

Tabela 1: Resultados numéricos - sistema IEEE de 14 barras

t_{max}	$f(V, \theta, t)$ (MW)	$P_{sen}(t)$	$P_{sen}(b^{sh})$	$P_V(V)$	Tempo de execução (s)
10	13,625	$4,91 \cdot 10^{-29}$	0	0	2,77
100	13,422	$2,60 \cdot 10^{-29}$	0	0	27,33

Dentre as 25 execuções do algoritmo para o sistema IEEE de 14 barras, para $t_{max} = 10$, o valor máximo da função objetivo obtido foi 15,103 MW e o valor médio foi 13,952 MW. Para $t_{max} = 100$, o valor máximo da função objetivo obtido foi 13,790 MW e o valor médio foi 13,543 MW.

Os valores das variáveis de controle discretas correspondentes aos resultados da tabela 1 estão expostos na tabela 2. Os níveis de tensão das barras da rede e ângulos das tensões das barras para 10 iterações estão expostos na figura 1. Os níveis de tensão das barras da rede e ângulos das tensões das barras para 100 iterações estão expostos na figura 2.

Tabela 2: Variáveis de controle discretas - sistema IEEE de 14 barras

t_{max}	$t_{4-7}(pu)$	$t_{4-9}(pu)$	$t_{5-6}(pu)$	$b_9^{sh}(pu)$
10	1,05	0,95	1,05	0
100	0,95	0,95	1,05	0

Figura 1: Níveis e ângulos de tensão das barras da rede - sistema IEEE de 14 barras - 10 iterações

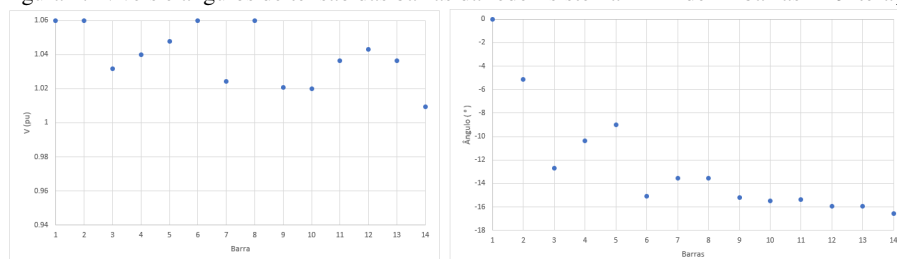
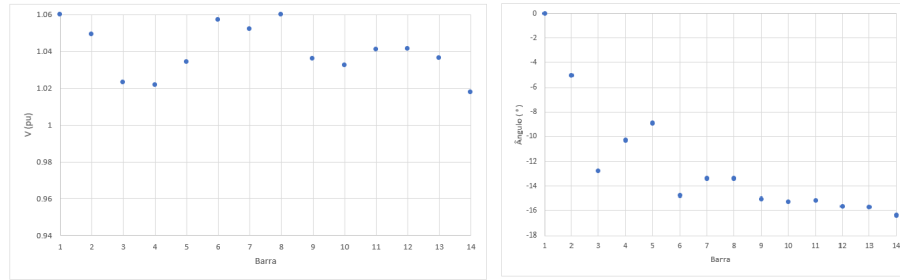


Figura 2: Níveis e ângulos de tensão das barras da rede - sistema IEEE de 14 barras - 100 iterações



5.2. Sistema elétrico IEEE de 30 barras

O problema de FPOR com variáveis discretas para o sistema elétrico IEEE de 30 barras possui: 53 restrições de igualdade; 72 restrições de desigualdade; 59 variáveis contínuas (relativas às magnitudes de tensão V e aos ângulos de tensão θ das barras); 6 variáveis discretas ($taps$ t de quatro transformadores com controle de tap e duas susceptâncias $shunt$ b^{sh}).

No algoritmo proposto, são otimizadas 5 variáveis contínuas: as tensões das 5 barras com controle de reativo do sistema ($V_{k \in B_{CR}}$); e as seis variáveis discretas ($t_{l \in T}$ e $b_{j \in B_{SS}}^{sh}$). As demais variáveis contínuas são obtidas a partir da resolução do problema de Fluxo de Carga. Foi considerado que as susceptâncias $shunt$ conectada às barras 10 e 24 do sistema devem pertencer aos conjuntos discretos: $b_{10}^{sh} \in \{0; 0,19; 0,34; 0,39\}$, $b_{24}^{sh} \in \{0; 0,05; 0,09\}$

Os resultados expostos na tabela 3 foram os melhores obtidos dentro de 25 execuções do algoritmo, com $n_{lobos} = 12$, $\lambda_1 = \lambda_2 = \lambda_3 = 100$.

Tabela 3: Resultados numéricos - sistema IEEE de 30 barras

t_{max}	$f(V, \theta, t)$ (MW)	$P_{sen}(t)$	$P_{sen}(b^{sh})$	$P_V(V)$	Tempo de execução (s)
10	18,029	$3,11 \cdot 10^{-29}$	0	0	3,32
100	17,768	$5,00 \cdot 10^{-29}$	0	0	33,29

Dentre as 25 execuções do algoritmo para o sistema IEEE de 30 barras, para $t_{max} = 10$, o valor máximo da função objetivo obtido foi 18,893 MW e o valor médio foi 18,338 MW. Para $t_{max} = 100$, o valor máximo da função objetivo obtido foi 18,278 MW e o valor médio foi 18,058 MW.

Os valores das variáveis de controle discretas correspondentes aos resultados da tabela 3 estão expostos na tabela 4. Os níveis de tensão das barras da rede e ângulos das tensões das barras para 10 iterações estão expostos na figura 3. Os níveis de tensão das barras da rede e ângulos das tensões das barras para 100 iterações estão expostos na figura 4.

Tabela 4: Variáveis de controle discretas - sistema IEEE de 30 barras

t_{max}	$t_{6-9}(pu)$	$t_{6-10}(pu)$	$t_{4-12}(pu)$	$t_{28-27}(pu)$	$b_{10}^{sh}(pu)$	$b_{24}^{sh}(pu)$
10	1,01	1,01	1,01	1,01	0	0
100	1,05	0,95	1,05	0,95	0	0

Figura 3: Níveis e ângulos de tensão das barras da rede - sistema IEEE de 30 barras - 10 iterações

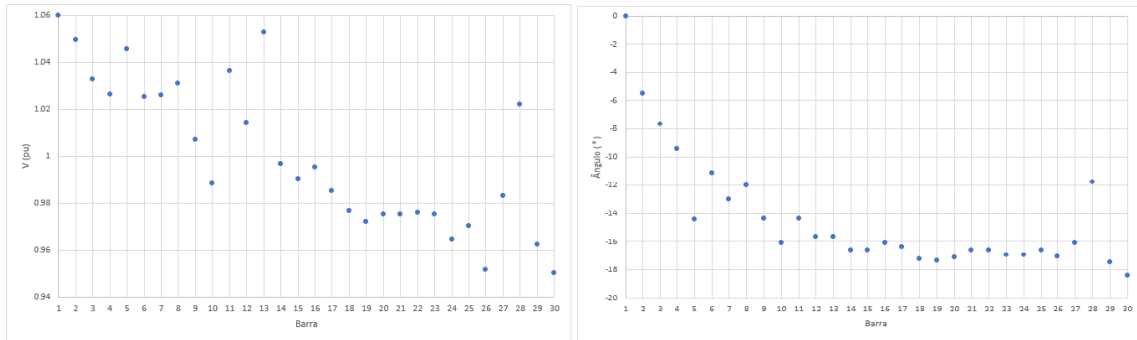
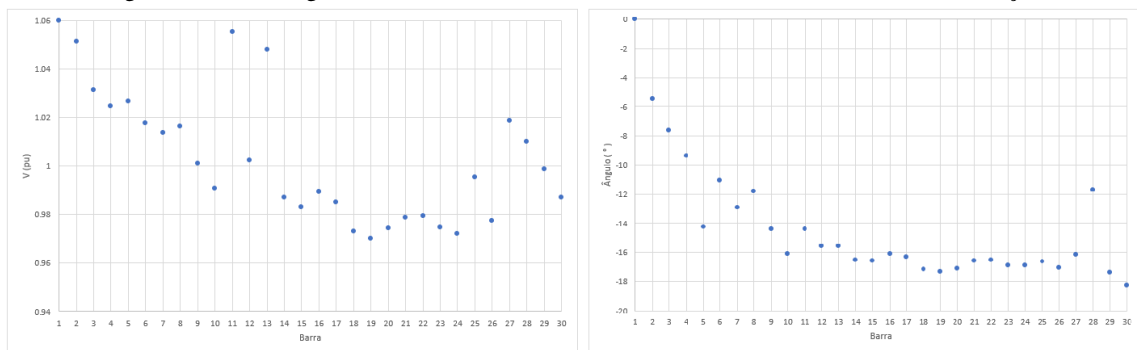


Figura 4: Níveis e ângulos de tensão das barras da rede - sistema IEEE de 30 barras - 100 iterações



5.3. Discussão dos resultados

Os resultados numéricos obtidos com o problema de FPOR para os sistemas de *benchmark* IEEE de 14 e 30 barras mostram a potencialidade da abordagem proposta para solução deste problema. Para ambos sistemas foram obtidas soluções factíveis de forma que as variáveis discretas assumiram valores discretos dentro dos respectivos intervalos, como pode ser observado também pelos valores das funções de penalização aplicadas a estas variáveis.

Para os dois sistemas, pode-se notar que o método obtém boas soluções mesmo com um baixo número de iterações (10 iterações) e com um pequeno número de agentes de busca ($n_{lobos} = 12$). A execução do algoritmo leva um tempo computacional aceitável e este está vinculado à execução do fluxo de carga para cada agente de busca, uma vez que a execução deste algoritmo leva um tempo considerável. A implementação do algoritmo de fluxo de carga paralelamente utilizando, por exemplo, o auxílio do poder de computação paralela de uma placa de vídeo, poderia diminuir consideravelmente o tempo de execução do algoritmo.

A abordagem proposta obteve resultados competitivos comparando-se com resultados de métodos exatos da literatura. Como exemplo, o Método de Newton com Programação Linear, o método de Rescalamento Não-Linear com Região de Confiança e o método de Pontos Interiores com Região de Confiança, propostos em [Silva 2019].

Dada a natureza estocástica de um algoritmo meta-heurístico, nem todos os resultados

convergem para a mesma solução ótima. Assim sendo, é importante executar o algoritmo diversas vezes para que seja possível determinar a melhor solução obtida.

6. Conclusão

O problema de Fluxo de Potência Ótimo Reativo é um problema de otimização não-linear, não-convexo, restrito e com variáveis contínuas e discretas, e é considerado de difícil solução. O objetivo de resolver este problema é obter as variáveis de controle de um sistema elétrico de potência que determinam a menor perda possível de potência ativa na rede, satisfazendo suas restrições físicas e operacionais.

Assim sendo, a resolução deste problema é de fundamental importância na operação de sistemas elétricos de potência e, ao considerar a natureza discreta de algumas de suas variáveis, é possível obter uma solução ótima com potencial aplicabilidade em sistemas elétricos reais, que possuem controles que só podem ser ajustados através de passos discretos - como é o caso dos *taps* de transformadores e a susceptância *shunt* de bancos de capacitores conectados às barras do sistema.

Este trabalho propõe uma abordagem de solução para o problema de FPOR através do algoritmo meta-heurístico *Grey Wolf Optimizer* (GWO) associado à função de penalização senoidal proposta por [Soler et al. 2013] para o tratamento das variáveis discretas do problema, sendo esta sua principal contribuição para a literatura.

Futuramente, pretende-se executar testes numéricos desta abordagem para o problema de FPOR para outros sistemas elétricos. Pretende-se também melhorar a implementação computacional do algoritmo, visando uma diminuição do tempo de execução do mesmo.

Agradecimentos

Os autores agradecem à Fundação de Amparo à Pesquisa do Estado de São Paulo - FAPESP (processo nº 2019/15950-0 - bolsista FAPESP) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPQ (processos 313495/2017-3 e 428740/2016-2).

Referências

- Cain, M. B., O'Neill, R. P., e Castillo, A. (2012). History of optimal power flow and formulations. *Federal Energy Regulatory Commission*, 1:1–36.
- Carpentier, J. (1962). Contribution a l'étude du dispatching économique. *Bulletin de la Societe Francaise des Electriciens*, 3(1):431–447.
- Coello, C. A. C. (2002). Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer methods in applied mechanics and engineering*, 191(11-12):1245–1287.
- El-Fergany, A. A. e Hasanien, H. M. (2015). Single and multi-objective optimal power flow using grey wolf optimizer and differential evolution algorithms. *Electric Power Components and Systems*, 43(13):1548–1559.
- Emary, E., Zawbaa, H. M., e Hassanien, A. E. (2016). Binary grey wolf optimization approaches for feature selection. *Neurocomputing*, 172:371–381.
- Jayabarathi, T., Raghunathan, T., Adarsh, B., e Suganthan, P. N. (2016). Economic dispatch using hybrid grey wolf optimizer. *Energy*, 111:630–641.
- Kamboj, V. K., Bath, S., e Dhillon, J. (2016). Solution of non-convex economic load dispatch problem using grey wolf optimizer. *Neural Computing and Applications*, 27(5):1301–1316.

- Khalilpourazari, S. e Khalilpourazary, S. (2018). Optimization of production time in the multi-pass milling process via a robust grey wolf optimizer. *Neural Computing and Applications*, 29(12): 1321–1336.
- Khamees, A. K., Badra, N., e Abdelaziz, A. Y. (2016). Optimal power flow methods: a comprehensive survey. *International Electrical Engineering Journal (IEEJ)*, 7(4):2228–2239.
- Majumdar, K., Das, P., Roy, P. K., e Banerjee, S. (2017). Solving opf problems using biogeography based and grey wolf optimization techniques. *International Journal of Energy Optimization and Engineering (IJEEO)*, 6(3):55–77.
- Mirjalili, S. (2015). How effective is the grey wolf optimizer in training multi-layer perceptrons. *Applied Intelligence*, 43(1):150–161.
- Mirjalili, S., Mirjalili, S. M., e Lewis, A. (2014). Grey wolf optimizer. *Advances in engineering software*, 69:46–61.
- Mittal, N., Singh, U., e Sohi, B. S. (2016). Modified grey wolf optimizer for global engineering optimization. *Applied Computational Intelligence and Soft Computing*, 2016.
- Mohanty, S., Subudhi, B., e Ray, P. K. (2015). A new mppt design using grey wolf optimization technique for photovoltaic system under partial shading conditions. *IEEE Transactions on Sustainable Energy*, 7(1):181–188.
- Nuaekaew, K., Artrit, P., Pholdee, N., e Bureerat, S. (2017). Optimal reactive power dispatch problem using a two-archive multi-objective grey wolf optimizer. *Expert Systems with Applications*, 87:79–89.
- Papalexopoulos, A. D., Imparato, C. F., e Wu, F. F. (1989). Large-scale optimal power flow: effects of initialization, decoupling and discretization. *IEEE Transactions on Power Systems*, 4(2):748–759.
- Silva, D. N. d. (2019). *Novas Abordagens Determinísticas de Otimização para Resolução do Problema de Fluxo de Potência Ótimo*. PhD thesis, Universidade de São Paulo.
- Soler, E. M., Asada, E. N., e Da Costa, G. R. (2013). Penalty-based nonlinear solver for optimal reactive power dispatch with discrete controls. *IEEE Transactions on Power Systems*, 28(3): 2174–2182.
- Sulaiman, M. H., Mustaffa, Z., Mohamed, M. R., e Aliman, O. (2015). Using the gray wolf optimizer for solving optimal reactive power dispatch problem. *Applied Soft Computing*, 32:286–292.
- Turner, L., Scheidler, A., Schäfer, F., Menke, J., Dollichon, J., Meier, F., Meinecke, S., e Braun, M. (2018). pandapower — an open-source python tool for convenient modeling, analysis, and optimization of electric power systems. *IEEE Transactions on Power Systems*, 33(6):6510–6521. ISSN 0885-8950.
- Tinney, W., Bright, J., Demaree, K., e Hughes, B. (1988). Some deficiencies in optimal power flow. *IEEE Transactions on Power Systems*, 3(2):676–683.