

PSO via linguagem Python para soluções de problemas de despacho econômico de carga [★]

Gustavo M. O. de Castro, Aline S. Gallina ^{*}
Moisés Arthur P. Borges ^{**}
Ronaldo Francisco R. Pereira, José Humberto A. Monteiro ^{***}

^{*} *eAmazônia - Energia Sustentável e Inovação, AC, (e-mail: gustavo@eamazonia.org, aline@eamazonia.org).*

^{**} *Pós-Graduação em Engenharia da Informação, Universidade Federal do ABC, SP (e-mail: moises.borges@ufabc.edu.br)*

^{***} *Centro de Ciências Exatas e Tecnológicas, Universidade Federal do Acre, AC (e-mail: ronaldo.pereira@ufac.br, humberto.monteiro@ufac.br)*

Abstract: This study aims to present the application of the Particle Swarm Optimization (PSO) algorithm to solve the Economic Load Dispatch (ED) problem in its classical formulation, considering losses in transmission lines, and the limits of the generating units. Two case studies were made to demonstrate the method's applicability, for systems with three and six generating units. Also, a comparison between two different stopping criteria was performed, by the maximum number of iterations and by error. The PSO was implemented using the Python programming language, in the Jupyter Notebook environment. The results were compared against each other, and against the classic optimization solution via Lagrange multipliers present in the literature. The results showed PSO's applicability for the considered DE problems. Another conclusion of this study is that the stopping criterion by error may return unreliable results since, during the 20 algorithm executions performed in the experiments, this criterion increased the variance of the analyzed metrics and the number of iterations. Therefore, it is concluded that the number of iterations must be known for each experiment, which guarantees the robustness and reliability of the results, considering the smaller standard deviation and percentage error in this case.

Resumo: Este trabalho tem por objetivo apresentar a aplicação do algoritmo de otimização por enxame de partículas (PSO) na solução do problema de Despacho Econômico de Carga (DE) em sua formulação clássica, considerando as perdas nas linhas de transmissão e os limites das unidades geradoras. Foram realizados dois estudos de caso para sistemas de três e seis unidades geradoras para demonstrar a aplicabilidade do método, além disso, foram feitas comparações de dois critérios de parada distintos, por número máximo de iterações e por erro. O PSO foi implementado em linguagem de programação Python em ambiente Jupyter Notebook. Os resultados foram comparados entre si e com a solução por otimização clássica via multiplicadores de Lagrange presente na literatura. Os resultados obtidos demonstraram a aplicabilidade do PSO para os problemas de DE considerados. Outra conclusão obtida deste trabalho foi que o critério de parada por erro pode retornar resultados não confiáveis, visto que nos experimentos feitos com 20 execuções do algoritmo, este critério aumentou a variância das métricas analisadas e do número de iterações, concluindo que o número de iterações para cada experimento deve ser conhecido, garantindo a robustez dos resultados.

Keywords: Economic Dispatch; Meta-Heuristics; Python; Particle Swarm Optimization.

Palavras-chaves: Despacho econômico; Meta-heurística; Python; Otimização por enxame de partículas.

1. INTRODUÇÃO

A aplicação de algoritmos que trazem soluções ótimas através de técnicas inspiradas em meta-heurísticas tem se tornado comum no âmbito da engenharia. Solucionar um sistema com poucas ou múltiplas variáveis é uma tarefa que pode ser solucionada com a aplicação de um algoritmo bioinspirado. Os algoritmos meta heurísticos objetivam encontrar soluções ótimas, mas não exatas, de um problema em troca de um menor esforço computacional necessário (CASTRO, 2022). Como exemplo de algoritmo meta heurístico, tem-se o algoritmo de otimização por enxame de partículas ou *Particle Swarm Optimization* (PSO), o qual possui diversas aplicações para resolução problemas em sistemas de não lineares presentes na literatura técnica (Alsakati et al., 2021; Hu and Yang, 2022; Zhang et al., 2021; Agrawal and Kundu, 2021; Faisal et al., 2020). Em Ramakoti et al. (2009), os autores usam PSO para ajustar matrizes de covariância de ruído para rastreamento de objetos pela abordagem de Filtro de Kalman. Os autores de Patel et al. (2020) propõem um algoritmo de sintonia para garantir condições iniciais consistentes para sistemas descritivos. Uma combinação EKF-PSO é apresentada em Vahidi-Moghaddam et al. (2020), os autores apresentam um observador adaptativo de distúrbios de tempo que é aplicado em um sistema mecânico com estimação de estados e parâmetros. Em Rayyam and Zazi (2018), é apresentada uma abordagem para controle de motores de indução baseada na estimação de estado por EKF e na sintonia de matrizes de covariância pelo método PSO.

Considerando sistemas de potência, mais especificamente, o problema de despacho econômico, a literatura técnica apresenta diversas abordagens com utilização do PSO e de outras técnicas meta heurísticas. Em Nivedha et al. (2018), apresenta uma interessante abordagem utilizando PSO para determinação da operação de custo mínimo para um sistema híbrido de geração eólica com células de combustível e geradores a diesel para alimentação de áreas remotas. Em Chen and Imani Marrani (2020), um problema não convexo de despacho econômico multi-área de grande escala é resolvido usando a abordagem híbrida entre PSO e o algoritmo de competição imperialista para minimizar o custo total de combustível em todas as áreas, satisfazendo as restrições de equilíbrio de potência, limites de geração e restrições de capacidade das linhas. Os autores de Al-Bahrani et al. (2018) apresentam o problema não convexo da utilização de unidades geradoras térmicas submetidas a efeitos de carregamento no ponto de válvula (VPL), o qual tem solução ineficiente para o método tradicional e é resolvido a partir do algoritmo multi gradiente PSO proposto. Outra abordagem interessante é apresentada em Pürlü and Türkyay (2018), os autores apresentam a técnica de Algoritmo Genético (AG) combinada com (PSO) para resolver o problema de despacho econômico dinâmico.

Nesse contexto, tem-se a grande importância dos algoritmos meta heurísticos, e.g., o algoritmo de otimização por enxame de partículas. Dessa forma, este trabalho busca tratar a solução do despacho econômico a partir da

* O presente trabalho foi realizado com apoio do programa PIBIC da Universidade Federal do Acre com recursos do CNPq

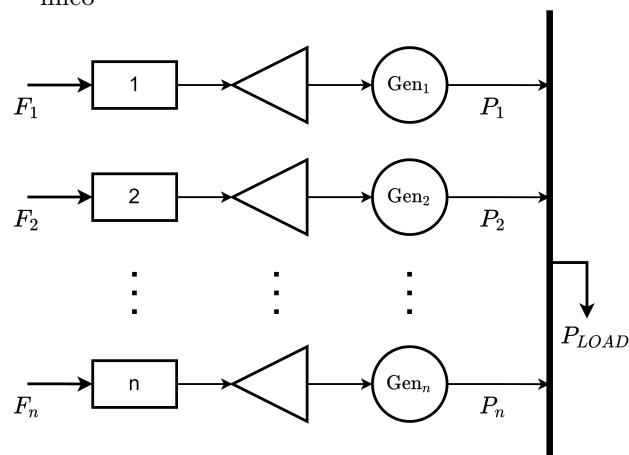
aplicação do algoritmo PSO como uma aplicação prática. Para isso, foi utilizado o ambiente Jupyter Notebook da linguagem de programação Python juntamente ao pacote Pyswarms. Realizou-se diversos experimentos para dois estudos de casos e através de métricas estatísticas, foram analisados os resultados para dois critérios de parada distintos, sendo por número máximo de iterações e por erro. As principais contribuições desse artigo são resolução do problema não-linear de despacho econômico para sistemas consolidados na literatura técnica a partir da linguagem de programação Python, sendo que os trabalhos difundidos na literatura utilizam o Matlab para isso, a validação da solução da programação apresentada a partir da comparação com a resolução clássica e a disponibilização do algoritmo de código aberto.

2. DESPACHO ECONÔMICO

O problema do Despacho Econômico (DE) envolve determinar a forma mais eficiente e barata de operar as unidades de geração de energia elétrica de um sistema, de forma a suprir a demanda total de carga, satisfazendo seus limites de operação e demais restrições do sistema.

Um sistema qualquer pode ser modelado para a resolução do problema do despacho econômico como o da Figura 1, com N geradores conectados a uma barra única suprindo uma carga equivalente P_{load} . A saída de cada unidade P_i é a potência gerada pela mesma, e as entradas F_i representam as taxa de custo, especificadas em \$/h.

Figura 1. Esquemático demonstrativo do Despacho econômico



A função de custo de um gerador individual possui características próprias, que dependem do recurso energético empregado, o preço desse recurso e a eficiência dos equipamentos. Segundo Saadat (2009), na maioria dos casos práticos, o custo de um gerador qualquer pode ser representado por uma função quadrática na forma:

$$F_i(P_i) = \alpha_i + \beta_i P_i + \gamma_i P_i^2 \quad (1)$$

Onde α_i , β_i e γ_i são os coeficientes de custo da unidade geradora i , respectivamente em \$, \$/h e \$/h². Em termos matemáticos, existe uma função que representa o custo total do sistema, que é a soma dos custos individuais de cada unidade geradora. O problema do despacho econômico é,

enfim, minimizar a função de custo F_T , que corresponde à soma das funções de custo individuais de cada gerador, estando sujeita às restrições impostas pelos limites de operação e que a soma das potências geradas individualmente seja igual à potência demandada do sistema, ou seja, a carga e as perdas.

É o custo total, que é a função a ser minimizada é:

$$F_T(P) = \sum_{i=1}^N F_i(P_i) \quad (2)$$

Na modelagem mais básica do DE, não são consideradas as perdas do sistemas e estas são modeladas como uma constante já imposta no valor da carga. Portanto, o balanço de potência deve ser satisfeito em termos das potências dos geradores P_i e da potência demandada da carga P_D como na equação:

$$\sum_{i=0}^{N_{gen}} P_i - P_D = 0 \quad (3)$$

A saída de potência de qualquer gerador termoeletrônico não deve exceder seus limites de aquecimento e nem deve estar abaixo do necessário para operação estável da caldeira, portanto, a potência gerada será restrita a um intervalo entre os valores mínimos e máximos característicos de cada gerador. Matematicamente falando, deve-se obedecer a seguinte restrição de desigualdade.

$$P_i^{min} \leq P_i \leq P_i^{max} \quad (4)$$

O DE básico consiste matematicamente falando em um problema de otimização com restrições que pode ser formalmente resolvido utilizando métodos de cálculo avançado como os multiplicadores de Lagrange. A função de Lagrange é formada adicionando à função objetivo, a função de restrição (ϕ) multiplicada por um fator indeterminado, conhecido como multiplicador de Lagrange (λ), e é mostrada na equação (5).

$$\mathcal{L} = F_T + \lambda\phi \quad (5)$$

Nos casos em que as distâncias de transmissão são curtas e a densidade de carga é alta, pode-se negar as perdas nas linhas sem afetar de forma significativa a solução do DE. No entanto, na maioria dos casos reais, as perdas nas linhas de transmissão e distribuição são um fator a ser considerado, pois a potência perdida é significativa e conta como demanda do sistema, sendo alimentada pelos geradores ativos. De acordo com Saadat (2009), para considerar o efeito das perdas na transmissão, pode-se utilizar a fórmula de perdas de Kron, como segue na equação (6).

$$P_L = \sum_{i=1}^{n_{gen}} \sum_{j=1}^{n_{gen}} P_i B_{ij} P_j + \sum_{i=1}^{n_{gen}} B_{i0} P_i + B_{00} \quad (6)$$

Onde as potências P_i e P_j são respectivamente o i -ésimo e j -ésimo elemento do vetor de potências geradas \mathbf{P} , B_{ij} são os elementos da matriz de perdas de dimensões

(n_{gen}, n_{gen}) , B_{i0} são os elementos do vetor de perdas de dimensões $(1, n_{gen})$ e B_{00} é uma constante conhecida como coeficiente de perda. Por conveniência, os nomes matriz de perdas, ou matriz \mathbf{B} , serão usados para denotar os elementos B_{ij} , B_{i0} e B_{00} .

Da dedução para encontrar a matriz de perdas, disponível em Saadat (2009), nota-se que ela é função da condição de operação atual do sistema, porém, se as condições de geração não forem muito diferentes das quais a matriz \mathbf{B} foi computada, ela poderá ser considerada como constante, sem alterações significativas no problema de DE. Dessa forma, o DE básico sofre mudanças em sua modelagem matemática com a consideração das perdas de transmissão, e o balanço de potência agora, levando em consideração a equação (6), fica.

$$\sum_{i=1}^{n_{gen}} P_i - P_D - P_L = 0 \quad (7)$$

Uma das dificuldades do cálculo analítico do problema de DE considerando as perdas envolve o fato de que o balanço de potência não é mais sujeito à uma constante, sendo considerado um fator que é função das condições de geração do sistema, o que torna o problema não-linear e mais complexo de ser resolvido.

Há também outras modelagens do DE para incluir diversas outras restrições que fazem o problema se aproximar mais da realidade, como o efeito de ponto de válvula e as zonas proibidas.

3. OTIMIZAÇÃO POR ENXAME DE PARTÍCULAS

Em 1995, James Kennedy e Russell Elberhart apresentaram o algoritmo de otimização por enxame de partículas (PSO) através do trabalho Kennedy and Eberhart (1995). O método tem inspiração no comportamento de enxames, como bandos de pássaros e cardumes de peixes e tem raízes ligadas aos algoritmos genéticos e estratégias evolucionárias. Tal técnica de otimização apresenta uma abordagem conceitual e computacionalmente simples, que requer apenas operadores matemáticos básicos e poucas linhas de códigos para implementação.

O algoritmo de otimização por enxame de partículas inicia primeiramente várias partículas, ou indivíduos, que são essencialmente possíveis soluções do problema dentro do espaço de busca, ou seja, são soluções factíveis, mas possivelmente não ótimas. Essas partículas buscam novas posições baseadas em fatores individuais e coletivos, isto é, elas possuem o conhecimento da trajetória e melhores resultados de si mesmas e das outras partículas do enxame. Matematicamente, esses fatores são, a posição de cada partícula, suas velocidades, a distância entre a posição atual e o melhor já atingido, e a distância entre a posição atual e a melhor posição entre todo o enxame. A avaliação de qual posição é melhor ou pior é dada pela função objetivo a cada iteração.

Depois de gerados os vetores de posição e velocidade de forma aleatória e dentro do espaço de busca. Cada partícula $\mathbf{x}_i = x_1, x_2, \dots, x_n$ tem sua posição alterada pelo seu vetor de velocidade $\mathbf{v}_i = v_1, v_2, \dots, v_n$ para próxima

posição do espaço, tal vetor de velocidade é função da velocidade anterior, do vetor da melhor posição atingida pela partícula \mathbf{p}_{best} e do melhor atingido pelo enxame \mathbf{g}_{best} . A equação que modifica a posição da partícula em cada iteração k é:

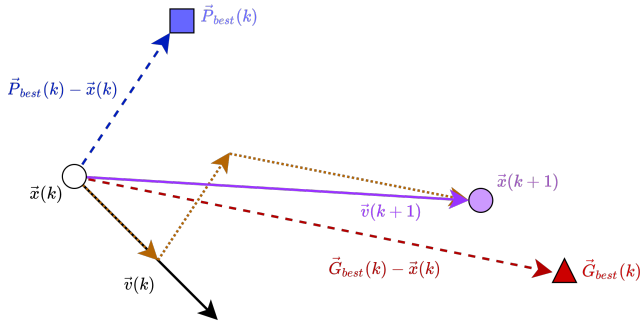
$$\mathbf{x}_i(k+1) = \mathbf{x}_i(k) + \mathbf{v}_i(k+1) \quad (8)$$

E a equação que modifica a velocidade a cada iteração k é:

$$\mathbf{v}_i(k+1) = w\mathbf{v}_i(k) + c_1r_1(\mathbf{p}_{best} - \mathbf{x}_i(k)) + c_2r_2(\mathbf{g}_{best} - \mathbf{x}_i(k)) \quad (9)$$

Na equação (9), w é o chamado fator de inércia, que de acordo com Fathi and Bevrani (2019), decresce o número de iterações necessárias. c_1 e c_2 são chamados de coeficientes de aceleração e os os termos associados a eles são respectivamente conhecidos como componente cognitivo, que representa a tendência do aprendizado de cada partícula e o componente social que representa a força de atração para a melhor posição do enxame, r_1 e r_2 são números aleatórios uniformemente distribuídos no intervalo $[0,1]$, o que introduz a natureza estocástica ao movimento das partículas. O modelo da equação (9) também é conhecido como *GBest model* e o seu esquema de atualização de posição é mostrado na Figura 2

Figura 2. Esquemático demonstrativo da mudança de posição do PSO



O primeiro termo da equação (9) é conhecido como termo de diversificação, pois sua componente pesa a tendência de explorar novas áreas no espaço de busca, o segundo e terceiro termo juntos são conhecidos como intensificação. De acordo com Lee and El-Sharkawi (2008), para favorecer a diversificação no início do processo evolutivo e a intensificação no final, o termo w é atualizado a cada iteração de acordo com a equação (10):

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \times iter \quad (10)$$

Onde w_{max} é o peso inicial, w_{min} é o peso final e $iter$ e $iter_{max}$ são respectivamente o número da iteração atual e o número máximo de iterações. Essa modificação foi introduzida depois e é conhecida como *Inertia Weights Approach* ou IWAPSO.

4. METODOLOGIA

A implementação da solução do problema do despacho econômico para três e seis unidades geradoras foi feita utilizando linguagem Python, juntamente aos pacotes SciPy, Numpy e Pyswarms, este último sendo dedicado à execução do PSO. Na implementação do PSO, a função objetivo é dada por:

$$f(\mathbf{P}) = \sum_{i=1}^{N_{gen}} C(\mathbf{P}) + \phi \left| \sum_{i=1}^{N_{gen}} P_i - P_D - P_L \right| \quad (11)$$

Os métodos meta-heurísticos têm uma certa dificuldade para lidar com restrições de igualdade que não sejam os limites do espaço de busca, então é necessário usar um fator de penalidade (ϕ) multiplicando a restrição do balanço de potência e perdas, dessa forma, quando forem escolhidas pelo PSO variáveis que não atendam essa restrição, a função objetivo será penalizada, e o algoritmo tenderá a fugir desses valores.

Para cada experimento, o PSO foi executado 20 vezes, as partículas foram iniciadas aleatoriamente dentro do espaço de busca, o fator de inércia foi variado linearmente conforme a equação (10) e das simulações foram analisados os critérios de erro em relação à solução conhecida (por multiplicador de Lagrange), erro percentual e a distância euclidiana da partícula final em relação às potências da solução conhecida, destas grandezas foram observados valores máximos, mínimos, média, desvio padrão e valor do melhor experimento.

O erro percentual foi calculado de acordo com a equação

$$Er \% = \frac{\text{Solução PSO} - \text{Solução Exata}}{\text{Solução Exata}} \times 100\% \quad (12)$$

E a distância euclidiana $d(p, q)$ entre duas partículas p e q quaisquer foi calculada de acordo com a equação (13). Onde as partículas são $p = (p_1, p_2, p_3, \dots, p_n)$ e $q = (q_1, q_2, q_3, \dots, q_n)$, n é o número de dimensões dos vetores.

$$d(p, q) = \sqrt{\sum_{i=0}^n (p_i - q_i)^2} \quad (13)$$

5. RESULTADOS

A fim de demonstrar a a eficácia do PSO para sistemas de maior complexidade para problemas de DE, o método de otimização foi aplicado em um estudo de caso para o sistema IEEE 30-barras com seis unidades geradoras, disponível em Cai et al. (2009), visualizado na Figura 3.

Como parâmetros do PSO foram adotados os valores da tabela 3. Os limites de geração e coeficientes de custo estão disponíveis nas tabelas 1 e 2 e sua matriz de perdas nas equações 14, 15 e 16.

Tabela 1. Limites de geração

Unidade	P_i^{min} [MW]	P_i^{max} [MW]
1	5	50
2	5	60
3	5	100
4	5	120
5	5	100
6	5	60

Tabela 2. Coeficientes de custo

Unidade	γ [\$/MW ²]	β [\$/MW]	α [\$/h]
1	0,01	2	10
2	0,012	1,5	10
3	0,004	1,8	20
4	0,006	1	10
5	0,004	1,8	20
6	0,01	1,5	10

Tabela 3. Parâmetros do PSO

Parâmetros	Valores
Partículas	20
Dimensão	6
Cognitivo c_1	2
Social c_2	2
Fator de inércia	$w_{min} = 0.4$ $w_{max} = 0.9$
Fator de penalidade ϕ	10

$$B_{ij} = 10^{-2} \times \begin{bmatrix} 0,1382 & -0,0299 & 0,0044 & -0,0022 & -0,0010 & -0,0008 \\ -0,0299 & 0,0487 & -0,0025 & 0,0004 & 0,0016 & 0,0041 \\ 0,0044 & -0,0025 & 0,0182 & -0,0070 & -0,0066 & -0,0066 \\ -0,0022 & 0,0004 & -0,0070 & 0,0137 & 0,0050 & 0,0033 \\ -0,0010 & 0,0016 & -0,0066 & 0,0050 & 0,0109 & 0,0005 \\ -0,0008 & 0,0041 & -0,0066 & 0,0033 & 0,0005 & 0,0244 \end{bmatrix} \quad (14)$$

$$B_{i0} = 10^{-2} \times \begin{bmatrix} -0,0107 & 0,0060 & -0,0017 & 0,0009 & 0,0002 & 0,0030 \end{bmatrix} \quad (15)$$

$$B_{00} = 9,8573 \times 10^{-3} \quad (16)$$

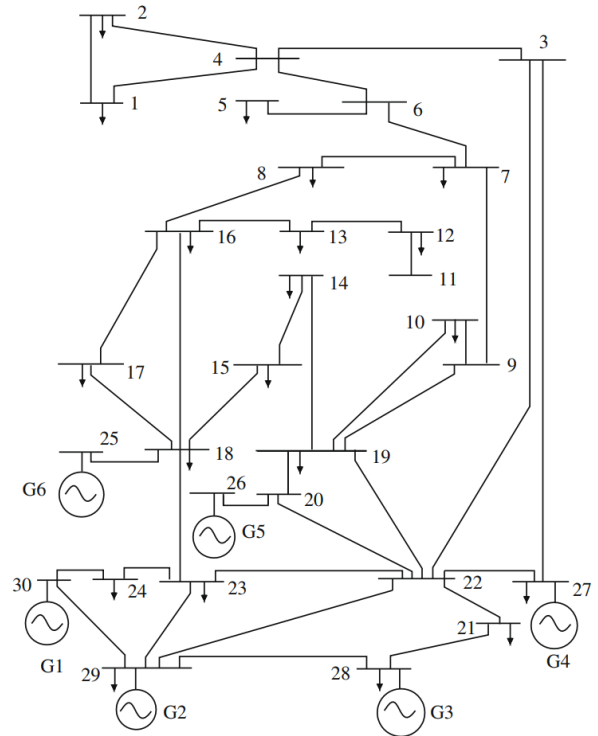
5.1 Solução via multiplicadores de Lagrange

Escrevendo o problema de otimização do sistema em termos das variáveis *slack*, temos a formulação (17)

$$\left\{ \begin{array}{l} \min \\ \text{s.a.} \end{array} \right. \begin{cases} C(P_1, P_2, P_3, \dots, P_6) \\ \sum P_i - P_D - P_L(P_1, P_2, P_3, \dots, P_6) = 0 \\ P_1 - 5 - S_1^2 = 0 \\ P_1 - 50 + S_2^2 = 0 \\ P_2 - 5 - S_3^2 = 0 \\ P_2 - 60 + S_4^2 = 0 \\ P_3 - 5 - S_5^2 = 0 \\ P_3 - 100 + S_6^2 = 0 \\ P_4 - 5 + S_7^2 = 0 \\ P_4 - 120 + S_8^2 = 0 \\ P_5 - 5 + S_9^2 = 0 \\ P_5 - 100 + S_{10}^2 = 0 \\ P_6 - 5 + S_{11}^2 = 0 \\ P_6 - 60 + S_{12}^2 = 0 \end{cases} \quad (17)$$

A função Lagrangeana, conforme visto anteriormente foi aplicada na equação (17) resultando em um sistema de 31

Figura 3. Sistema IEE 30 barras com 6 unidades geradoras



Fonte: (Cai et al., 2009)

equações e 31 variáveis que foi resolvido por Santos (2020), obtendo o resultado encontrado na tabela 4.

Tabela 4. Resultados do DE por otimização clássica

Variáveis	Solução
P_1 [MW]	11,2074
P_2 [MW]	29,1029
P_3 [MW]	58,0028
P_4 [MW]	99,4452
P_5 [MW]	52,4564
P_6 [MW]	35,5083
$\sum P_i$ [MW]	285,7230
Custo [\$/h]	605,43

5.2 Critério de parada por iterações - 6 unidades geradoras

O experimento foi aplicado ao sistema IEEE 30 barras com seis unidades geradoras para o critério de parada, por número máximo de iterações (C1), que foi variado para 200, 500, 1000, 2000 e 5000 iterações, os parâmetros do PSO foram escolhidos conforme a tabela 3. Os resultados numéricos para distância euclidiana, custo e erro percentual são exibidos nas tabelas 5, 6 e 7.

5.3 Critério de parada por erro - 6 unidades geradoras

O experimento foi repetido, agora utilizando o critério de parada por erro no mesmo sistema. O parâmetro variado foi o erro calculado em $\Delta C = 10^{-6}$ nas últimas 500, 1000 e 2000 iterações (*ftol*). Além das grandezas anteriores, também foi observado a média de iterações para cada experimento, valor máximo, mínimo, desvio padrão e valor do melhor experimento. Os resultados dos experimentos são mostrados nas tabelas 8, 9, 10 e 11.

Tabela 5. Resultados da distância euclidiana para o critério de número máximo de iterações

Iterações	Distância euclidiana [MW]		
	Média	Desvio padrão	Min
200	22,93549	7,03808	13,33998
500	15,96117	4,9763	4,71366
1000	11,15251	2,933038	6,182495
2000	9,019645	2,964362	2,254738
3000	7,942195	2,504419	2,760866
5000	6,521242	2,801955	2,482352

Tabela 6. Resultados do custo para o critério de número máximo de iterações

Iterações	Custo final [\$/h]		
	Média	Desvio padrão	Best
200	609,3234	2,190334	607,2316
500	607,4141	1,036715	605,6235
1000	606,4657	0,512665	605,776
2000	606,0677	0,340791	605,4929
3000	605,8785	0,243817	605,4896
5000	605,7749	0,270199	605,4832

Tabela 7. Resultado do erro percentual para o critério de número máximo de iterações

Iterações	Erro percentual [%]		
	Média	Desvio padrão	Best
200	0,643073	0,361782	0,297577
500	0,327715	0,171236	0,031954
1000	0,17106	0,084678	0,057156
2000	0,105331	0,056289	0,010395
3000	0,074085	0,040272	0,009837
5000	0,056968	0,044629	0,008784

Tabela 8. Resultados da distância euclidiana por critério de erro

ftol	Distância euclidiana [MW]		
	Média	Desvio	Min
500	20,78721	5,284527	13,29713
1000	17,95811	5,62534	8,214967
2000	9,509528	4,667364	2,499111

Tabela 9. Resultados do custo final para o critério de erro

ftol	Custo final [\$/h]		
	Média	Desvio	best
500	608,6364	1,476799	607,0781
1000	607,8407	1,276803	606,0706
2000	606,3322	0,775929	605,4737

Tabela 10. Resultados do erro percentual para o critério de erro

ftol	Erro percentual [%]		
	Média	Desvio padrão	Best
500	0,529599	0,243926	0,272222
1000	0,398185	0,210892	0,105817
2000	0,149015	0,128162	0,007223

Tabela 11. Dados das iterações para o DE de 6 unidades geradoras para o critério de erro

ftol	Iterações				
	Média	Desvio padrão	Max	Min	best exp
500	993,1	352,2198	1712	504	851
1000	2170,45	1267,723	6293	1001	2734
2000	6679,4	2932,466	10000	2462	10000

6. COMPARAÇÃO DOS RESULTADOS

Para analisar o critério de parada por número máximo de iterações (C1) e o critério de parada por erro (C2), iremos comparar as execuções do primeiro em relação ao número de iterações média do segundo. Por exemplo, percebe-se que o critério C2 por 1000 iterações nos retornou um valor médio de iterações de aproximadamente, 2000. Os resultados dessa execução serão comparados com o critério C1 por 2000 iterações. A tabela 12 mostra a comparação entre os critérios de número máximo de iterações igual 3000 e o critério de erro de 10^{-6} por 1500 iterações, visto que apresentam número médio de iterações similares.

Tabela 12. Comparação de dois critérios de parada para o DE de 6 unidades

Iterações		Erro (1000)	Max. Iter. (2000)
		Média	2170,45
	Desvio padrão	1267,723	X
	Max	6293	2000
	Min	1001	2000
Distância euclidiana	Média	17,95811	9,019645
	Desvio padrão	5,62534	2,964362
	Min	8,214967	2,254738
Custo	Média	607,8407	606,0677
	Desvio padrão	1,276803	0,340791
	Best	606,0706	605,4929

Nas duas tabelas podemos verificar que os critério C1 e C2 podem chegar a valores igualmente ótimos, no entanto, para o segundo critério podemos verificar que o valor médio e desvio padrão são maiores, tal conclusão se reflete tanto na distância euclidiana como no custo. Em relação ao número de iterações, foram comparadas 20 execuções do algoritmos de custo computacional semelhantes, isso significa que se o PSO for executado uma vez, ao acaso, o critério C2 tem uma probabilidade maior de retornar resultados piores, visto que há um alto valor no desvio padrão para as iterações e outras grandezas. Tal conclusão se reflete nos dois casos estudados.

7. CONCLUSÃO

O PSO se mostrou ser um método eficaz para resolução dos problemas de otimização envolvendo o DE, visto que encontrou as soluções ótimas e muito próximas das esperadas para os dois casos estudados e para números variados de iterações. Foram examinados dois critérios de parada para o PSO, por número de iterações e por erro calculado nas últimas iterações. O primeiro se revelou ser robusto e confiável para encontrar as soluções, enquanto o segundo, apesar de poder encontrar valores igualmente ótimos, sendo executado ao acaso, possui uma chance de retornar valores menos próximos da solução exata que o primeiro. Tais conclusões foram obtidas comparando as métricas estatísticas de média e desvio padrão, além dos valores mínimos. Concluiu-se portanto, que o critério de parada por erro possui uma variância maior em torno da média para uma execução do algoritmo qualquer, visto que os experimentos foram feitos analisando 20 execuções.

AGRADECIMENTOS

Os autores agradecem a Universidade Federal do Acre, e ao eAmazônia - Energia Sustentável e Inovação pelo apoio no desenvolvimento dessa pesquisa.

REFERÊNCIAS

- Agrawal, S. and Kundu, P. (2021). A unified optimal power flow modeling for vsc-hvdc converter: a novel methodology for optimal installation based on average loadability index. *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*.
- Al-Bahrani, L.T., Patra, J.C., and Stojcevski, A. (2018). Solving economic dispatch problem under valve-point loading effects and generation constrains using a multi-gradient pso algorithm. In *2018 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.
- Alsakati, A.A., Vaithilingam, C.A., Naidu, K., Rajendran, G., Alnasseir, J., and Jagadeeshwaran, A. (2021). Particle swarm optimization for tuning power system stabilizer towards transient stability improvement in power system network. In *2021 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAET)*, 1–6. IEEE.
- Cai, J., Ma, X., Li, Q., Li, L., and Peng, H. (2009). A multi-objective chaotic particle swarm optimization for environmental/economic dispatch. *Energy Conversion and Management*, 50(5), 1318–1325.
- CASTRO, G.M.O. (2022). Aplicação de pso via linguagem python para soluções de problemas de despacho econômico de carga. *Universidade Federal do Acre*.
- Chen, J. and Imani Marrani, H. (2020). An efficient new hybrid ica-pso approach for solving large scale non-convex multi area economic dispatch problems. *Journal of Electrical Engineering & Technology*, 15(3), 1127–1145.
- Faisal, S.F., Beig, A.R., and Thomas, S. (2020). Time domain particle swarm optimization of pi controllers for bidirectional vsc hvdc light system. *Energies*, 13(4), 866.
- Fathi, M. and Bevrani, H. (2019). *Optimization in electrical engineering*. Springer.
- Hu, Z. and Yang, G. (2022). Simulation control model of synchronous motor based on pso algorithm optimization in power system. *Energy Reports*, 8, 1044–1054.
- Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, 1942–1948. IEEE.
- Lee, K.Y. and El-Sharkawi, M.A. (2008). *Modern heuristic optimization techniques: theory and applications to power systems*, volume 39. John Wiley & Sons.
- Nivedha, R.R., Singh, J.G., and Ongsakul, W. (2018). Pso based economic dispatch of a hybrid microgrid system. In *2018 International Conference on Power, Signals, Control and Computation (EPSCICON)*, 1–5. IEEE.
- Patel, T., Rao, M., Purohit, J.L., and Shah, V.A. (2020). State estimation of nonlinear descriptor systems using particle swarm optimization based extended kalman filter. In *2020 European Control Conference (ECC)*, 991–996. IEEE.
- Pürlü, M. and Türkay, B.E. (2018). Dynamic economic dispatch with valve point effect by using ga and pso algorithm. In *2018 6th International Conference on Control Engineering & Information Technology (CEIT)*, 1–6. IEEE.
- Ramakoti, N., Vinay, A., and Jatoth, R.K. (2009). Particle swarm optimization aided kalman filter for object tracking. In *2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies*, 531–533. IEEE.
- Rayyam, M. and Zazi, M. (2018). Particle swarm optimization of a non-linear kalman filter for sensorless control of induction motors. In *2018 7th International Conference on Renewable Energy Research and Applications (ICRERA)*, 1016–1020. IEEE.
- Saadat, H. (2009). Power system analysis,(2nd). *McGraw-Hill Higher Education*.
- Santos, J.O.d. (2020). Metaheurísticas bio-inspiradas aplicadas em problemas de despacho econômico de carga.
- Vahidi-Moghaddam, A., Rajaei, A., Ayati, M., Vahidmoghaddam, R., and Hairi-Yazdi, M.R. (2020). Adaptive prescribed-time disturbance observer using nonsingular terminal sliding mode control: Extended kalman filter and particle swarm optimization. *arXiv preprint arXiv:2012.01026*.
- Zhang, J., Song, J., Su, C., Hu, J., and Wang, Q. (2021). Parameter identification of hvdc transmission system model based on intelligent optimization algorithm. In *2021 IEEE 11th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, 643–647. IEEE.