

Implementação de Mecanismos de Ajuste de Parâmetros de Controle do Otimizador do Lobo Cinzento Multiobjetivo via Lógica Nebulosa

Allan Christian Krainski Ferrari*, Gideon Villar Leandro**, Leandro dos Santos Coelho**, Carlos Alexandre Gouvea da Silva*, Douglas Antonio Firmino Pelacini*, Cristiano Osinski**

*Centro Universitário UNIFACEAR, Av. das Araucárias, 3803, 83707-067, Araucária, Paraná, Brasil (e-mail: allan.ferrari@unifacear.edu.br, carlos.gouvea@unifacear.edu.br, douglas.pelacini@ufpr.br)

** Universidade Federal do Paraná UFPR, Caixa Postal 19011, 81531-980, Curitiba, Paraná, Brasil (e-mail: gede@eletrica.ufpr.br, leandro.coelho@ufpr.br, cristiano.osinski@ufpr.br)

Abstract: The Multi-Objective Grey Wolf Optimizer (MOGWO) is a recent metaheuristic in solving multi-objective problems. This article implements two mechanisms for adjusting the control parameters of the MOGWO. The techniques are based on fuzzy logic and were originally proposed to improve the convergence of the single-objective version of the MOGWO. To evaluate the two fuzzy implementations, five multiobjective test problems were used, where each algorithm was submitted to a sequence of ten runs. The inverted generational distance was used as a metric for the comparison and evaluation of the algorithms. The results show that the two fuzzy mechanisms improve the convergence of the MOGWO.

Resumo: O otimizador do lobo cinzento multiobjetivo (MOGWO - *Multi-objective Grey Wolf Optimizer*) é uma metaheurística recente na resolução de problemas do tipo multiobjetivo. Este artigo faz a implementação de dois mecanismos de ajuste nos parâmetros de controle do MOGWO. As técnicas são baseadas em lógica nebulosa e foram propostas originalmente para melhorar a convergência da versão mono-objetivo do MOGWO. Para avaliar as duas implementações nebulosas, foram utilizados cinco problemas teste multiobjetivo, onde cada algoritmo foi submetido a uma sequência de dez simulações. A distância geracional invertida foi utilizada como métrica para comparação e avaliação dos algoritmos. Os resultados mostram que os dois mecanismos nebulosos melhoram a convergência do MOGWO.

Keywords: metaheuristics; multi-objective optimization; wolf optimizer; fuzzy systems; Pareto.

Palavras-chaves: metaheurísticas; otimização multiobjetivo; otimizador do lobo; sistemas nebulosos; Pareto.

1. INTRODUÇÃO

Os problemas de otimização podem ser do tipo mono-objetivo com um objetivo a ser otimizado ou multiobjetivo com mais de um objetivo a serem otimizados. Na área da engenharia existem problemas reais que são muito complexos. A multi-objetividade é uma característica que torna esses problemas mais desafiadores e que necessitam de ferramentas específicas para lidar com eles. A utilização de metaheurísticas é uma opção viável na resolução desses tipos de problemas (Marler e Arora, 2004).

Neste contexto, as metaheurísticas são muito populares na resolução de problemas de otimização que são considerados difíceis ou complexos para a implementação dos métodos clássicos de otimização, pois muitas dessas técnicas são baseadas na informação do gradiente e abordagens de programação matemática. Estes algoritmos poderosos são geralmente inspirados no comportamento de algum processo natural que pode ser representado por um fenômeno físico ou por um comportamento biológico ou social de certos grupos de animais (Salcedo-Sanz, 2016).

Uma das metaheurísticas mais recentes voltada a resolução de problemas com mais de um objetivo é o otimizador do lobo cinzento multiobjetivo (MOGWO - *Multi-objective Grey Wolf Optimizer*) proposto por Mirjalili et al. (2016). Este algoritmo é a versão multiobjetivo do otimizador do

lobo cinzento (GWO - *Grey Wolf Optimizer*) proposto por Mirjalili et al. (2014). O MOGWO mimetiza o comportamento de hierarquia e caça dos lobos cinzentos em um espaço de busca multiobjetivo.

No trabalho de Mirjalili et al. (2016), o algoritmo MOGWO foi comparado com outras metaheurísticas multiobjetivo, como exemplo: a otimização por enxame de partículas multiobjetivo (MOPSO - *Multi-Objective Particle Swarm Optimization*) e o Algoritmo Evolucionário Multiobjetivo baseado em Decomposição (MOEA/D - *Multi-Objective Evolutionary Algorithm based on Decomposition*). Diante desses algoritmos, o MOGWO mostrou-se competitivo e até superior na otimização de funções multiobjetivos que foram propostas para competição de algoritmos no Congresso de Computação Evolucionária 2009 (CEC 2009 - *Congress on Evolutionary Computation*) (Zhang et al., 2008).

Este artigo tem como objetivo implementar duas técnicas de ajuste nos parâmetros de controle do algoritmo MOGWO visando melhorar a convergência do mesmo. Estas técnicas são baseadas em lógica nebulosa e foram originalmente propostas para melhorar a convergência do GWO. A implementação das variantes do GWO no MOGWO foram avaliadas através da otimização de funções multiobjetivo propostas no CEC 2009. A contribuição deste trabalho resume-se na implementação e avaliação de métodos que

foram bem sucedidos no processo de otimização do GWO, a versão mono-objetivo do MOGWO. Isto possibilita a constatação do teorema *No Free Lunch* (NFL) que diz que não há um único algoritmo que seja superior de forma unânime na resolução de todos os tipos de problemas (Wolpert e Macready, 1997).

Este trabalho está estruturado em três partes: métodos, resultados e conclusão. Nos métodos é feita uma apresentação da metaheurística MOGWO e das técnicas de ajuste dos parâmetros de controle que foram aplicadas no mesmo. Na seção referente aos resultados, é mostrado o desempenho obtido pela implementação das técnicas, sendo feito em seguida uma comparação e análise com a versão convencional do MOGWO. Finalmente, na conclusão são apresentados as restrições e os desdobramentos para futuros trabalhos.

2. MÉTODOS

Nesta seção são apresentados os fundamentos básicos do algoritmo MOGWO e as variantes nebulosas do GWO que foram implementados no MOGWO visando melhorar a convergência do mesmo.

2.1 Algoritmo MOGWO

A metaheurística GWO mimetiza a vida selvagem dos lobos cinzentos que são predadores naturais que estão no topo da cadeia alimentar. Estes animais vivem em grupos onde existe uma hierarquia social dominante. O lobo alfa (α) é responsável por liderar o grupo, no GWO representa a melhor solução. Depois do α na hierarquia, vem o lobo beta (β) e lobo delta (δ) que representam a segunda e terceira melhor solução respectivamente. Estes lobos possuem a função de ajudar o lobo α . A classe mais baixa da hierarquia é representada pelos lobos omega (ω) que se submetem aos lobos das classes superiores (Mirjalili et al., 2014).

A estrutura do GWO é inspirada no mecanismo de caça dos lobos cinzentos. Os lobos α , β e δ são responsáveis por liderar a caçada de uma presa, enquanto que os lobos ω acompanham os três lobos na pesquisa de um ótimo global. Durante o processo de caça, a presa deixa de se mover quando é cercada pelos lobos. As equações que representam este comportamento são seguintes (Mirjalili et al., 2014):

$$\vec{D} = |\vec{C}\vec{X}_p(t) - \vec{X}(t)|, \quad (1)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A}\vec{D}, \quad (2)$$

onde t indica a iteração atual, $\vec{X}_p(t)$ é o vetor posição do melhor lobo, o vetor \vec{D} representa a distância do movimento aleatório do melhor lobo em relação ao lobo ω na iteração atual, $\vec{X}(t)$ e $\vec{X}(t+1)$ indicam respectivamente a posição de um lobo ω na iteração atual e na iteração a um passo à frente. As variáveis \vec{A} e \vec{C} são vetores coeficientes sendo calculados pelas seguintes equações:

$$\vec{A} = 2\vec{a}\vec{r}_1 - \vec{a}, \quad (3)$$

$$\vec{C} = 2\vec{r}_2, \quad (4)$$

onde os elementos de \vec{a} decrescem linearmente de 2 a 0, a cada iteração. Os vetores \vec{r}_1 e \vec{r}_2 são valores gerados de forma aleatória com distribuição uniforme no intervalo [0 1].

O vetor \vec{A} representa o comportamento de exploração territorial do lobo cinzento e decresce a cada iteração por estar em função de \vec{a} . Quando $|\vec{A}| > 1$, os lobos estão explorando o espaço de busca (divergindo) para encontrar uma presa em potencial para perseguir e atacar. Quando $|\vec{A}| < 1$, os lobos estão atacando uma presa (convergindo). O vetor \vec{C} simula o efeito dos obstáculos quando os lobos estão perseguindo a presa. Ao contrário do vetor $|\vec{A}|$, o vetor $|\vec{C}|$ age de forma aleatória, logo isto pode facilitar ou dificultar a captura da presa (Mirjalili et al., 2014).

Os lobos cinzentos são capazes de localizar uma presa potencial, em seguida cercar e depois atacar. A posição dos lobos ω é atualizada em função da localização dos lobos α , β e δ que lideram a caçada, pois possuem melhor conhecimento sobre a localização de uma possível presa. As equações que representam este comportamento são as seguintes:

$$\vec{D}_\alpha = |\vec{C}_\alpha\vec{X}_\alpha(t) - \vec{X}(t)|, \quad (5)$$

$$\vec{D}_\beta = |\vec{C}_\beta\vec{X}_\beta(t) - \vec{X}(t)|, \quad (6)$$

$$\vec{D}_\delta = |\vec{C}_\delta\vec{X}_\delta(t) - \vec{X}(t)|, \quad (7)$$

$$\vec{X}_1 = \vec{X}_\alpha(t) - \vec{A}_\alpha\vec{D}_\alpha, \quad (8)$$

$$\vec{X}_2 = \vec{X}_\beta(t) - \vec{A}_\beta\vec{D}_\beta, \quad (9)$$

$$\vec{X}_3 = \vec{X}_\delta(t) - \vec{A}_\delta\vec{D}_\delta, \quad (10)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (11)$$

As equações de (5), (6), (7), (8), (9) e (10) representam a implementação do mecanismo de caça dos lobos cinzentos. Estas equações são uma expansão das equações (1) a (4), e consideram a influência individual dos lobos α , β e δ . A equação (11) atualiza a posição dos lobos ω para a próxima iteração sendo basicamente uma média dos valores \vec{X}_1 , \vec{X}_2 e \vec{X}_3 que representam respectivamente a influência individual dos lobos α , β e δ . Os lobos ω tendem a se mover para o centro dos três melhores lobos. No final de cada iteração a hierarquia do grupo é atualizada (Mirjalili et al., 2014).

A diferença entre o GWO e o MOGWO, é que este possui dois mecanismos adicionais que são responsáveis por explorar soluções em um espaço de busca multiobjetivo. O primeiro mecanismo consiste em um arquivo que é responsável por armazenar soluções ótimas de Pareto não dominadas. O segundo mecanismo é uma estratégia de seleção do líder que ajuda selecionar os lobos α , β e δ a partir das soluções salvas no arquivo (Mirjalili et al., 2016).

O arquivo é uma unidade de armazenamento simples que pode salvar ou recuperar soluções ótimas de Pareto não dominadas obtidas no decorrer das iterações. O arquivo possui um controlador que faz o controle quando uma solução quer entrar no mesmo ou quando o próprio arquivo está cheio. Existe um número limite de soluções ótimas de Pareto não dominadas que o arquivo pode armazenar. No decorrer das iterações, as soluções não dominadas obtidas até então são comparadas com as soluções residentes do arquivo. Os resultados dessas comparações podem ser as seguintes (Mirjalili et al., 2016):

- Se uma nova solução é dominada por pelo menos uma das soluções residentes do arquivo, então a nova solução não deve ter permissão para entrar no arquivo;
- Se uma nova solução domina uma ou mais soluções residentes do arquivo, então as soluções dominadas dentro do arquivo devem ser omitidas e a nova solução será apta a entrar no arquivo;
- Se nem a nova solução e nem as soluções residentes do arquivo se dominam, então a nova solução deve ser adicionada ao arquivo;
- Se o arquivo estiver lotado, o mecanismo de rede deve ser executado primeiro para reorganizar a segmentação do espaço objetivo e encontrar o segmento mais lotado para omitir uma de suas soluções, então a nova solução deve ser inserida no segmento menos lotado. Isto visa melhorar a diversidade final da frente ótima de Pareto aproximada.

A probabilidade de excluir uma solução é aumentada proporcionalmente ao número de soluções no hipercubo (segmento). Para remover soluções se o arquivo estiver cheio, os segmentos mais aglomerados são selecionados primeiro, e uma solução é omitida de um deles aleatoriamente para fornecer um espaço para a nova solução. Há um caso especial em que uma solução é inserida fora dos hipercubos. Neste caso, todos os segmentos são estendidos a fim de cobrir as novas soluções. Portanto, os segmentos de outras soluções também podem ser alterados (Mirjalili et al., 2016).

No trabalho de Mirjalili et al. (2016), o mecanismo de seleção do líder escolhe o mínimo de segmentos lotados no espaço de busca multiobjetivo e oferece uma de suas soluções não dominadas como lobos α , β e δ . O método de seleção por roleta é utilizado na escolha dos lobos líderes utilizando a seguinte probabilidade para cada hipercubo:

$$P_i = \frac{c}{N_i}, \quad (12)$$

onde c é uma constante maior que 1 e N_i corresponde ao número de soluções ótimas de Pareto obtidas no i -ésimo segmento. O pseudocódigo referente a esta metaheurística está representado em Algoritmo 1.

2.2 Ajuste dos parâmetros \vec{a} e \vec{C} do MOGWO via sistema Fuzzy em função do número da iteração corrente

No trabalho de Rodriguez et al. (2017), o sistema de inferência fuzzy utiliza o número da iteração corrente normalizada, assumindo valores entre 0 e 1, para o ajuste simultâneo dos parâmetros de controle \vec{a} e \vec{C} do GWO. Este modelo nebuloso utiliza o modelo linguístico do tipo Mamdani com funções de pertinência triangular e método de defuzzificação por centroide. Os termos linguísticos adotados foram: “low” (baixo), “medium” (médio) e “high” (alto). O sistema fuzzy com as respectivas funções de pertinência e o conjunto de regras estão representadas respectivamente na Fig. 1 e na Tabela 1. Os parâmetros das funções pertinências adotadas neste trabalho são os mesmos utilizados no trabalho de Rodriguez et al. (2017).

O conjunto de regras da Tabela 1, representa o raciocínio do trabalho proposto por Rodriguez et al. (2017). A primeira regra representa o início do processo de otimização onde os

Algoritmo 1: Pseudocódigo MOGWO

Iniciar a população dos lobos $X_i (i = 1, 2, \dots, n)$.
 Iniciar os parâmetros \vec{a}, \vec{A} e \vec{C} .
 Calcular os valores objetivos de cada lobo.
 Encontrar as soluções não dominadas e iniciar o arquivo com elas.
 Selecionar \vec{X}_α pelo mecanismo de seleção do líder.
 Excluir o α do arquivo.
 Selecionar \vec{X}_β pelo mecanismo de seleção do líder.
 Excluir o β do arquivo.
 Selecionar \vec{X}_δ pelo mecanismo de seleção do líder.
 $t = 1$.
Enquanto $t <$ número máximo de iterações **faça**:
 Para cada lobo **faça**:
 Atualizar a posição pelas equações (5)-(11).
 Fim Para
 Atualizar os parâmetros \vec{a}, \vec{A} e \vec{C} .
 Calcular os valores objetivos de todos os lobos.
 Encontrar as soluções não dominadas.
 Atualizar o arquivo com as soluções não dominadas obtidas.
 Se o arquivo estiver cheio **faça**:
 Executar o mecanismo de rede para omitir uma das soluções do arquivo.
 Adicionar novas soluções no arquivo.
 Fim Se
 Se alguma solução das novas adicionadas no arquivo estiver localizada fora dos hipercubos **faça**:
 Atualizar as redes para cobrir as novas soluções.
 Fim Se
 Selecionar \vec{X}_α pelo mecanismo de seleção do líder.
 Excluir o α do arquivo.
 Selecionar \vec{X}_β pelo mecanismo de seleção do líder.
 Excluir o β do arquivo.
 Selecionar \vec{X}_δ pelo mecanismo de seleção do líder.
 $t = t + 1$.
Fim Enquanto
 Retorna o arquivo.

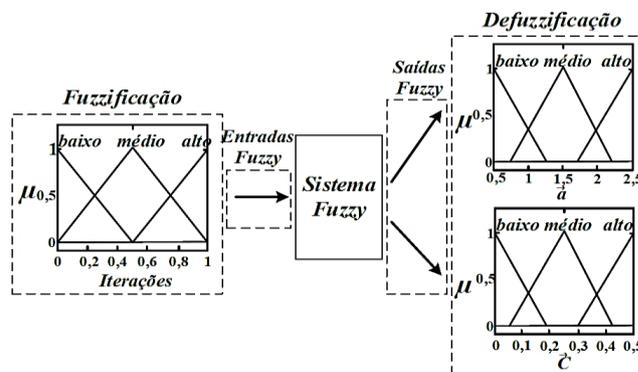


Fig. 1. Sistema fuzzy proposto por Rodriguez et. al. (2017).

Tabela 1. Conjunto de regras do sistema nebuloso proposto por Rodriguez et al. (2017)

Inferência para \vec{a} e \vec{C}	
1ª Se: Iterações é baixo	Então: \vec{a} é alto e \vec{C} é baixo
2ª Se: Iterações é médio	Então: \vec{a} é médio e \vec{C} é médio
3ª Se: Iterações é alto	Então: \vec{a} é baixo e \vec{C} é alto

lobos estão na busca de uma presa em potencial, nesta fase a influência dos obstáculos é mínima. A segunda regra representa a perseguição da presa, no decorrer das iterações os lobos vão cercando cada vez mais a presa, porém a

influência dos obstáculos aumenta ainda mais. A terceira regra representa o final do processo de otimização onde lobos cercam a presa para atacar, onde os obstáculos podem ajudar ou atrapalhar os lobos na finalização do ataque à presa.

2.3 Ajuste dos parâmetros \vec{a} e \vec{C} do MOGWO via sistema Fuzzy em função do valor do fitness normalizado

O sistema de inferência nebuloso, proposto por Ferrari et al. (2019), ajusta os parâmetros \vec{a} e \vec{C} por meio do valor do fitness normalizado (NFV - *Normalized Fitness Value*) de cada lobo. Neste sistema nebuloso, cada lobo está associado diretamente a um par próprio de valores \vec{a} e \vec{C} que são iniciados respectivamente com valores 2 e 0. Estes valores são atualizados durante o processo de otimização da função objetivo. O modelo fuzzy recebe os valores atuais dos parâmetros \vec{a} e \vec{C} , e também o valor referente ao desempenho mensurado do NFV de cada lobo da população. A saída é representada pelas variações dos parâmetros essenciais do GWO, denominados de $\Delta\vec{a}$ e $\Delta\vec{C}$. A partir do fitness de cada lobo, o valor do respectivo NFV é calculado por:

$$NFV = \frac{fitness - fitness_{min}}{fitness_{max} - fitness_{min}} \quad (13)$$

O valor de NFV pertence ao intervalo [0 1], onde o fitness de cada lobo é obtido diretamente pelo valor ótimo da função objetivo. O método nebuloso proposto por Ferrari et al. (2019), foi aplicado em problemas de otimização do tipo minimização, logo o $fitness_{min}$ representa o melhor lobo enquanto que o $fitness_{max}$ representa o pior. Na implementação deste método no MOGWO, o valor do fitness dos lobos foi obtido através da soma direta dos objetivos das funções multiobjetivo utilizadas neste trabalho. As equações que atualizam os parâmetros \vec{a} e \vec{C} de cada lobo são as seguintes:

$$\vec{a}^{t+1} = \vec{a}^t - \Delta\vec{a} \quad (14)$$

$$\vec{C}^{t+1} = \vec{C}^t + \Delta\vec{C} \quad (15)$$

O sistema fuzzy e conjunto de regras estão representados respectivamente na Fig. 2 e Tabela 2.

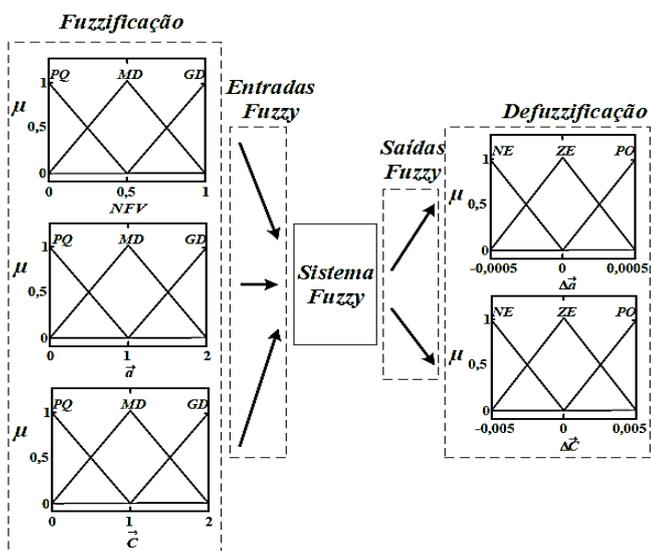


Fig. 2. Sistema fuzzy proposto por Ferrari et al. (2019).

Tabela 2. Conjunto de regras do sistema nebuloso proposto por Ferrari et al. (2019)

Inferência para $\Delta\vec{a}$ e $\Delta\vec{C}$	
1ª Se: NFV é PQ e \vec{a} é PQ	Então: $\Delta\vec{a}$ é ZE
2ª Se: NFV é PQ e \vec{a} é MD	Então: $\Delta\vec{a}$ é NE
3ª Se: NFV é PQ e \vec{a} é GD	Então: $\Delta\vec{a}$ é NE
4ª Se: NFV é MD e \vec{a} é PQ	Então: $\Delta\vec{a}$ é PO
5ª Se: NFV é MD e \vec{a} é MD	Então: $\Delta\vec{a}$ é ZE
6ª Se: NFV é MD e \vec{a} é GD	Então: $\Delta\vec{a}$ é NE
7ª Se: NFV é GD e \vec{a} é PQ	Então: $\Delta\vec{a}$ é PO
8ª Se: NFV é GD e \vec{a} é MD	Então: $\Delta\vec{a}$ é ZE
9ª Se: NFV é GD e \vec{a} é GD	Então: $\Delta\vec{a}$ é NE
10ª Se: NFV é PQ e \vec{C} é PQ	Então: $\Delta\vec{C}$ é PO
11ª Se: NFV é PQ e \vec{C} é MD	Então: $\Delta\vec{C}$ é PO
12ª Se: NFV é PQ e \vec{C} é GD	Então: $\Delta\vec{C}$ é ZE
13ª Se: NFV é MD e \vec{C} é PQ	Então: $\Delta\vec{C}$ é PO
14ª Se: NFV é MD e \vec{C} é MD	Então: $\Delta\vec{C}$ é ZE
15ª Se: NFV é MD e \vec{C} é GD	Então: $\Delta\vec{C}$ é NE
16ª Se: NFV é GD e \vec{C} é PQ	Então: $\Delta\vec{C}$ é PO
17ª Se: NFV é GD e \vec{C} é MD	Então: $\Delta\vec{C}$ é ZE
18ª Se: NFV é GD e \vec{C} é GD	Então: $\Delta\vec{C}$ é NE

O sistema nebuloso utiliza o modelo linguístico de Mamdani com funções de pertinência em formato triangular, operador de agregação de antecedentes mínimo, operador de agregação de regras máximo e defuzzificação por centro do máximo. Os termos linguísticos das funções de pertinências, utilizados para descrever os valores semânticos das variáveis de entrada (NFV, \vec{a} e \vec{C}) são declarados como: PQ (Pequeno); MD (Médio) e GD (Grande). Os valores semânticos das variáveis de saída ($\Delta\vec{a}$ e $\Delta\vec{C}$) são declarados como: NE (Negativo); ZE (Zero) e PO (Positivo). Os parâmetros das funções pertinências adotadas neste trabalho são os mesmos utilizados no trabalho de Ferrari et al. (2019).

A base de conhecimento do trabalho de Ferrari et al. (2019) utilizada na elaboração do conjunto de regras (Tabela 2), segue os seguintes preceitos:

- O início do processo de otimização do algoritmo GWO representa o começo de uma caçada. Nesta etapa inicial, o parâmetro \vec{a} inicia com um valor grande, enquanto que o parâmetro \vec{C} inicia um valor relativamente pequeno. Para os lobos da população que apresentarem um valor NFV grande, médio ou pequeno, então o valor do coeficiente \vec{a} dos mesmos deverá ser diminuído, enquanto que o valor do parâmetro \vec{C} deverá ser aumentado. Isto significa que esses lobos estão explorando um território e escolhendo uma presa em potencial para cercar e atacar;
- O final do processo de otimização é representado pelo fim da perseguição da presa. Em geral, o parâmetro \vec{a} apresenta um valor pequeno, enquanto que o parâmetro \vec{C} apresenta um valor grande. Isto significa que os lobos cercaram uma presa em potencial e estão se preparando para o ataque. Para os lobos da população que apresentarem um valor NFV pequeno, logo, os valores dos parâmetros \vec{a} e \vec{C} relacionados aos mesmos devem ter uma variação desprezível, pois isto significa que esses lobos convergiram para uma solução;
- No decorrer das iterações um lobo pode apresentar um valor NVF médio ou “grande”, porém seus parâmetros \vec{a} e \vec{C} podem apresentar valores que são mais condizentes com final do processo de otimização (valor de \vec{a} pequeno e

valor de \bar{C} grande). Nesta situação pode ocorrer problemas de convergência prematura, onde o algoritmo pode ficar preso em algum mínimo local. Para evitar esta situação, é necessário aumentar o valor do parâmetro \bar{a} e diminuir o valor do parâmetro \bar{C} de forma a aumentar o território de exploração dos lobos.

3. RESULTADOS

Para este trabalho, a implementação dos algoritmos e a obtenção dos seus respectivos resultados foi realizada por meio da utilização do ambiente computacional Matlab®. A seguir serão apresentados as funções multiobjetivos, os critérios de análise e comparação adotados neste trabalho. Após isso, são apresentados os resultados obtidos pelos algoritmos e a análise estatística dos mesmos.

3.1 Problemas Teste Multiobjetivo

Para avaliar a convergência dos mecanismos nebulosos propostos por Rodriguez et al. (2017) e Ferrari et al. (2019), foram utilizados 5 problemas de otimização multiobjetivo sem restrição (*unconstrained*) que estão descritos na Tabela 3. Estas funções foram propostas por Zhang et al. (2008) com a finalidade de competição entre as metaheurísticas no CEC 2009. Estas cinco funções foram escolhidas porque também foram utilizadas no trabalho de Mirjalili et al. (2016).

Tabela 3. Problema teste multiobjetivo

Funções Multiobjetivo	
UF1	$f_1 = x_1 + \frac{2}{ J_1 } \sum_{j \in J_1} \left[x_j - \sin \left(6\pi x_1 + \frac{j\pi}{n} \right) \right]^2$ $f_2 = 1 - \sqrt{x_1} + \frac{2}{ J_2 } \sum_{j \in J_2} \left[x_j - \sin \left(6\pi x_1 + \frac{j\pi}{n} \right) \right]^2$ $J_1 = \{j j \text{ é ímpar e } 2 \leq j \leq n\}, \quad J_2 = \{j j \text{ é par e } 2 \leq j \leq n\}$
UF2	$f_1 = x_1 + \frac{2}{ J_1 } \sum_{j \in J_1} y_j^2, \quad f_2 = 1 - \sqrt{x_1} + \frac{2}{ J_2 } \sum_{j \in J_2} y_j^2$ $J_1 = \{j j \text{ é ímpar e } 2 \leq j \leq n\}, \quad J_2 = \{j j \text{ é par e } 2 \leq j \leq n\}$ $y_j = \begin{cases} x_j - \left[0,3x_1^2 \cos \left(24\pi x_1 + \frac{4j\pi}{n} \right) + 0,6x_1 \right] \cos \left(6\pi x_1 + \frac{j\pi}{n} \right) & j \in J_1 \\ x_j - \left[0,3x_1^2 \cos \left(24\pi x_1 + \frac{4j\pi}{n} \right) + 0,6x_1 \right] \cos \left(6\pi x_1 + \frac{j\pi}{n} \right) & j \in J_2 \end{cases}$
UF4	$f_1 = x_1 + \frac{2}{ J_1 } \sum_{j \in J_1} h(y_j), \quad f_2 = 1 - x_1^2 + \frac{2}{ J_2 } \sum_{j \in J_2} h(y_j)$ $J_1 = \{j j \text{ é ímpar e } 2 \leq j \leq n\}, \quad J_2 = \{j j \text{ é par e } 2 \leq j \leq n\}$ $y_j = x_j - \sin \left(6\pi x_1 - \frac{j\pi}{n} \right), j = 2, \dots, n, \quad h(t) = \frac{ t }{1+e^{2 t }}$
UF5	$f_1 = x_1 + \left(\frac{1}{2N} + \varepsilon \right) \sin(2N\pi x_1) + \frac{2}{ J_1 } \sum_{j \in J_1} h(y_j)$ $f_2 = 1 - x_1 + \left(\frac{1}{2N} + \varepsilon \right) \sin(2N\pi x_1) + \frac{2}{ J_2 } \sum_{j \in J_2} h(y_j)$ $J_1 = \{j j \text{ é ímpar e } 2 \leq j \leq n\}, \quad J_2 = \{j j \text{ é par e } 2 \leq j \leq n\},$ $N \text{ é um inteiro, } \varepsilon > 0, \quad y_j = x_j - \sin \left(6\pi x_1 - \frac{j\pi}{n} \right), \quad j = 2, \dots, n$
UF7	$f_1 = \sqrt[3]{x_1} + \frac{2}{ J_1 } \sum_{j \in J_1} y_j^2, \quad f_2 = 1 - \sqrt[5]{x_1} + \frac{2}{ J_2 } \sum_{j \in J_2} y_j^2$ $J_1 = \{j j \text{ é ímpar e } 2 \leq j \leq n\}, \quad J_2 = \{j j \text{ é par e } 2 \leq j \leq n\}$ $y_j = x_j - \sin \left(6\pi x_1 - \frac{j\pi}{n} \right), \quad j = 2, \dots, n$

Nos problemas teste multiobjetivo (detalhes na Tabela 3) a variável x_i representa os parâmetros que serão otimizados e n é a dimensão das funções multiobjetivos que neste trabalho foi definido é igual a 10. Pode-se observar na Tabela 3 que as

funções multiobjetivo UF1, UF2, UF4, UF5 e UF7 possuem dois objetivos a ser otimizados.

3.2 Critérios de Análise e Comparação

Com o objetivo de avaliar a convergência dos mecanismos nebulosos implementados no MOGWO durante o processo de otimização das funções multiobjetivos, os algoritmos utilizados neste trabalho foram submetidos a uma série de 10 simulações. Cada simulação foi realizada com 1000 iterações. Cada algoritmo utiliza uma população composta por 100 lobos e o tamanho do arquivo para o armazenamento de soluções não dominadas é igual a 100.

Para mensurar o desempenho dos algoritmos foi utilizado como métrica a distância geracional invertida IGD (*Inverted Generational Distance*) que é definida pela seguinte equação:

$$IGD = \frac{\sqrt{\sum_{i=1}^r d_i^2}}{r} \quad (16)$$

onde r é o número de elementos da Frente de Pareto (PF-*Pareto Front*) e d_i é a distância euclidiana entre a i -ésima solução da PF e o ponto mais próximo do conjunto aproximado PF obtido pelo algoritmo (Figueiredo, 2017). O IGD foi escolhido como métrica porque a PF das funções da Tabela 3 é conhecida.

3.3 Resultados Obtidos

Os resultados obtidos estão representados na Tabela 4 e nas Figuras 3 e 4. Os mecanismos propostos por Rodriguez et al. (2017) e Ferrari et al. (2019) são identificados por “Fuzzy Iter” e “Fuzzy NfV” respectivamente.

Tabela 4. Resultado do IGD das funções (Tabela 3)

	Função UF1		
	MOGWO	Fuzzy Iter	Fuzzy NfV
IGD	0,0964	0,0880	0,1055
Média	0,0322	0,0180	0,0350
Desvio padrão	0,0956	0,0858	0,1013
Mediana	0,0593	0,0668	0,0639
Mínimo	0,1599	0,1137	0,1661
Máximo			
	Função UF2		
	MOGWO	Fuzzy Iter	Fuzzy NfV
IGD	0,0597	0,0567	0,0498
Média	0,0100	0,0133	0,0116
Desvio padrão	0,0600	0,0604	0,0452
Mediana	0,0429	0,0402	0,0350
Mínimo	0,0725	0,0779	0,0710
Máximo			
	Função UF4		
	MOGWO	Fuzzy Iter	Fuzzy NfV
IGD	0,0960	0,0881	0,0922
Média	0,0107	0,0056	0,0097
Desvio padrão	0,0967	0,0869	0,0891
Mediana	0,0804	0,0805	0,0824
Mínimo	0,1127	0,0966	0,1138
Máximo			
	Função UF5		
	MOGWO	Fuzzy Iter	Fuzzy NfV
IGD	0,3716	0,4792	0,3425
Média	0,1429	0,1757	0,1164
Desvio padrão	0,3641	0,4491	0,3469
Mediana	0,1785	0,2426	0,1243
Mínimo	0,6735	0,8765	0,5327
Máximo			
	Função UF7		
	MOGWO	Fuzzy Iter	Fuzzy NfV
IGD	0,0826	0,1035	0,0779
Média	0,0145	0,0144	0,0194
Desvio padrão	0,0806	0,1006	0,0730
Mediana	0,0676	0,0807	0,0509
Mínimo	0,1174	0,1279	0,1123
Máximo			

