Análise Paramétrica em Algoritmo de Inteligência de Enxame Utilizando Funções Benchmark

Weverson Celio Silva de Lima

Graduando em Engenharia de Computação pela Faculdade de Computação (FACOMP) da Universidade Federal do Pará (UFPA), campus Castanhal.

e-mail: weversoncelio25@gmail.com

ABSTRACT Optimization problems are present in applications in the scientific, financial, industrial and management areas, and in recent years several methodologies have emerged that aim to obtain their solution. One of these techniques is known as Swarm Intelligence (SI), based on the behavior of relatively simple beings, but who manage to solve complex problems when they are placed in a collective. In the literature, several SI algorithms were found, but there was a lack of in-depth studies on the behavior of each of its parameters. Therefore, this work presents a parametric analysis of two widely used SI algorithms, namely Particle swarm optimization (PSO) and Firefly Algorithm (FA). For this, a benchmarking study was carried out using benchmark functions in three search intervals of different sizes, in order to evaluate metrics such as accuracy, precision and average processing time. For this purpose, respectively, 315 and 207 scenarios were developed for PSO and FA. Furthermore, to compare the SI in relation to the traditional heuristic, 171 scenarios were developed for the Random Walk (RW) algorithm. With results, sets of parameters were obtained with accuracy and precision around 100% in the best scenarios of the SI algorithms, demonstrating the importance of a good parameterization for an optimal performance of the method.

INDEX TERMS Optimization Algorithm; Firefly Algorithm; PSO.

I. INTRODUÇÃO

A otimização consiste em escolher, entre várias possibilidades de realização de uma ação, aquela que possibilita atingir os melhores resultados com os menores gastos possíveis. Em vista disso, a otimização está inserida no dia a dia, podendo citar, como exemplo, o processo de comparação de preço e especificações antes de comprar um novo produto, escolhendo aquele que tem o melhor custo-benefício. Além disto, no ambiente corporativo, é comum as empresas escolheres os fornecedores que possuem os insumos mais baratos, afim de maximizar os lucros para um determinado produto. Como exemplificado, a otimização possui implicações em situações práticas, por este motivo, diversos estudos surgiram com objetivo de formalizar o problema, a fim de desenvolver técnicas para soluciona-los [1], [2]

Muitas vezes, a solução para o problema de otimização não é trivial, sendo necessários artifícios computacionais. Por este motivo, bons algoritmos de otimização possuem três princípios fundamentais. A primeira é a robustez para um bom desempenho em uma variedade de problemas. Em seguida vem a eficiência, para não depender de uma grande quantidade de recurso computacional ou um elevado tempo de processamento. Por fim a acurácia, para permitir identificar a solução com precisão, sem grande sensibilidade a erros [3].
Assim, as técnicas de otimização, geralmente, são classificadas em duas classes, sendo elas, os métodos determinísticos e os métodos estocásticos. Os métodos determinísticos, são usados quando o problema de otimização pode ser representado matematicamente, o que permitem ser usadas abordagens embasadas em cálculos diferenciais. Para isto, é necessário que o problema de otimização tenha uma função bem definida e que seja diferenciável em todos os pontos. Estas técnicas possuem teoremas que lhes garantem a convergência para uma solução ótima com alta precisão e em um tempo baixo de processamento para os problemas em que estes são aplicáveis. No entanto, em situações onde a função contém muitos pontos de ótimos locais ou tenha descontinuidades, a eficiência dos métodos determinísticos é comprometida [4]-[6].

Diante disto, surgiram os métodos estocásticos, baseados na probabilidade para tomada de decisões durante o processo de busca de quais soluções devem ser testadas como solução do problema. Para isto, possuem processos pseudoaleatórios que fazem caminhadas em sentidos “aleatórios” no espaço de busca. os métodos estocásticos começam em um ponto incialmente aleatório. A partir disto, tentam localizar os melhores valores em sua vizinhança em relação ao ponto atual, consequentemente, os valores da função vão sendo incrementados, o que geralmente permite encontrar boas soluções para o problema em questão [4]-[6]

Deste princípio, uma das metodologias estocásticas que vem sendo bastante utilizada em problemas de otimização é conhecida com inteligência de Enxame (IE) [7]-[12], sendo algoritmos bioinspirados baseados nos relacionamentos sociais encontrados em algumas espécies de animais, que são formadas por seres simples que são capazes de maximizar suas chances de sobrevivência ao desenvolver comportamentos coletivos complexos.

O presente trabalho tem como objetivo realizar um estudo paramétrico nos algoritmos baseados em inteligência de

enxame *Firefly Algorithm* (FA) e *Particle swarm optimization* (PSO), além de um terceiro algoritmo baseado na heurística tradicional, chamado de *Random Walk* (RW). Para isto, serão utilizadas algumas funções de *benchmarks*, sendo elas a Ackley, Rastrigin e Rosenbrock em intervalos de busca com tamanho diferentes, afim de encontrar os melhores valores de parâmetros. O problema apresentado é relevante, visto que, apesar de uma grande quantidade de pesquisas utilizarem os algoritmos descritos no escopo deste trabalho, nota-se uma carência de estudo sobre o impacto que os valores escolhidos dos seus parâmetros podem ter em relação ao desempenho do algoritmo.

II. ALGORITMOS UTILIZADOS

A. Heurística *Random Walk*

O *Random Walk* (RW), é uma heurística baseada no movimento irregular do grão de pólen. Onde, cada indivíduo da população comporta-se de maneira individual, ou seja, não há troca de informações entre diferentes indivíduos da mesma população, sendo que o processo de busca da solução do problema ocorre através da caminhada aleatória executada pelo algoritmo. Para compreender o funcionamento do RW, imagine que o movimento de um grão de pólen está restrito a dois sentidos possíveis, para frente ou para trás. Como mostrado na Figura 1, considerando a probabilidades idênticas para os dois casos, um grão que no instante t encontra-se na posição P, no instante seguinte $t+1$ irá se encontrar na posição P-Δ ou na posição P+Δ, onde Δ é o deslocamento efetuado pelo grão de pólen [13], [14].

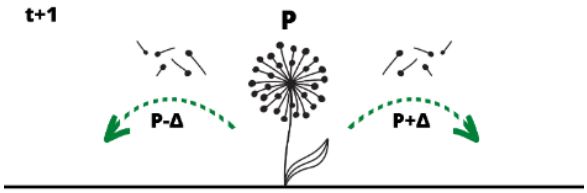

Figura 1. Movimento do grão de Pólen

No RW, a população inicial é criada de forma aleatória, conforme mostrado no fluxograma da Figura 2. Após isto, os indivíduos são classificados pela função aptidão. Para permitir a exploração em novos pontos, são substituídos os indivíduos com as menores aptidões por novos criados aleatoriamente, sendo a porcentagem de troca definida na Taxa de Substituição. O aperfeiçoamento do indivíduo ocorre através de uma caminhada aleatória ($rand$) partindo do ponto atual ($P(i)_{atual}$) para um novo ponto ($P(i)_{novo}$), sendo a distância do movimento definida no tamanho de passo ($step$), conforme mostrado:

$$P(i)_{novo} = P(i)_{atual} + step * rand \quad (1)$$

Para evitar a piora da posição do indivíduo durante as interações, a atualização só ocorrerá caso a aptidão do novo ponto ($A(i)_{novo}$) seja maior do que aptidão do ponto atual ($A(i)_{atual}$). Estes processos se repetem até que o um dos critérios de paradas seja satisfeito.

$$P(i)_{atual} = \begin{cases} P(i)_{novo}, se \;\; A(i)_{novo} > A(i)_{atual} \\ P(i)_{atual}, caso\ contrário \end{cases} \quad (2)$$

Figura 2. Fluxograma do Random Walk

B. *Inteligência de Enxame*

Na natureza, diversos animais utilizam algumas técnicas sociais com o objetivo de facilitar a caça, a coleta de alimentos, a probabilidade de acasalamento e a defesa do território contra a invasão de predadores, o que garantem uma maior adaptabilidade ao habitat ondem vivem. Para isto, os seres respondem rapidamente a ameaças e oportunidades encontradas em seus territórios, isto é possível devido ao processo de troca de informações entre os indivíduos ser feito em um tempo ágil e eficiente. Exemplos emblemáticos desta organização são encontradas em enxames de insetos, revoadas de pássaros, rebanhos de antílopes e cardumes de peixes [15].

Na computação, os algoritmos de Inteligência de Enxames (IE) são ferramentas computacionais inspiradas em comportamentos coletivos de organismos que vivem e interagem em grupos simples e autônomos. Estes algoritmos fazem uso de uma abordagem baseada em população, onde cada indivíduo representa um potencial solução para o problema. Individualmente os seres possuem regras relativamente simples e básicas para executar suas ações, mas a partir do momento que os indivíduos são colocados juntos, ocorre a interação entre os seres, emergindo o comportamento coletivo inteligente que possibilita solucionar problemas complexos de forma bastante eficaz [2], [5], [16]. Para isto, estas metaheurísticas são baseadas nos seguintes princípios:

1. O princípio da intensificação tem como objetivo encontrar o melhor valor na vizinhança do ponto ótimo. Para isto, são usadas as informações locais de um determinado ponto empregando técnicas de exploração de sua vizinhança, melhorando o seu valor em relação à função objetivo. Com isto, o processo contribui para a convergência mais rápida para o ponto de maior valor da região.
2. O processo de diversificação visa explorar outras regiões fora da vizinhança, o que não limita a busca aos pontos já encontrados. Neste caso, altos valores de diversificação provocam uma convergência mais lenta do algoritmo.

Particle Swarm Optimization

O PSO (*Particle swarm optimization*), conhecido em português como otimização por enxame de partículas, tem como inspiração a revoada de pássaros a procura de alimentos, onde inicialmente, quando levantam voo, apresentam um comportamento aleatório, mas com o passar de alguns instantes, o bando voa organizadamente. Emulando este processo, a partícula do PSO é atraída para a posição do melhor global atual e, ao mesmo tempo, tende a se mover aleatoriamente [17], conforme mostrado na Figura 3, ou seja, o PSO utiliza a aleatoriedade do número real e a comunicação global entre as partículas do enxame para a resolução do problema [18].

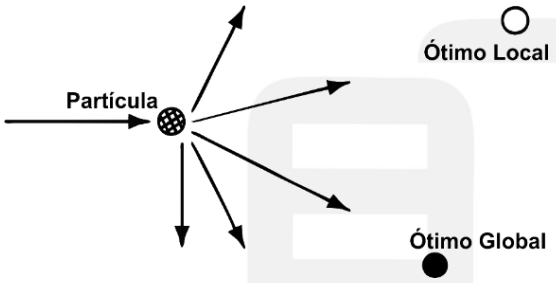

Figura 3. Possíveis movimentos da partícula

O funcionamento do algoritmo é exposto no fluxograma da Figura 4. Inicialmente, cada partícula está associada a dois vetores, posição e velocidade, inicializados de forma aleatória:

$$X_I(t) = \{x_{i1}(t), x_{i2}(t), \dots, x_{in}(t)\} \quad (3)$$

$$V_I(t) = \{v_{i1}(t), v_{i2}(t), \dots, v_{in}(t)\} \quad (4)$$

Estes valores são atualizados a cada iteração do algoritmo, em outras palavras, a partícula se movimenta no espaço de busca. Com isto, o desempenho das partículas é calculado individualmente através da função aptidão, com intuito de determinar aquela que mais se aproxima da solução do problema. A melhor posição da partícula encontrada até a iteração atual é armazenada no P_{best}, por sua vez, a melhor posição global é armazenada no G_{best}. O comportamento das partículas no espaço de busca é modelado pela velocidade de cada partícula, de acordo com:

$$V_i\,(t+1) = wv_i(t) + r_1c_1\big(P_{best}(t) - x_i(t)\big) + r_2c_2\big(G_{best}(t) - x_i(t)\big) \quad (5)$$

$$X_i\,(t+1) = X_i(t) + V_i(t+1) \quad (6)$$

Onde, w (fator inercial) determina o quanto uma partícula confia na direção que está seguindo em um determinado momento. r_1 e r_2 são números aleatórios gerados entre 0 e 1, sendo usados para ponderar o comportamento da partícula. c_1 (fator cognitivo) determina o nível de confiança que uma partícula possui nas suas experiências passadas e c_2 (fator social) determina a tendência de uma partícula seguir o melhor resultado do enxame em cada iteração [17], [19]. Após a velocidade e a posição do enxame serem atualizados, é feito a atualização de cada partícula, usando a expressão:

$$P_i\,(t+1) = \begin{cases} X_i(t+1), se \;\; f\big(X_i(t+1)\big) \geq f(P_i\,(t)) \\ P_i\,(t), caso\ contrário \end{cases} \quad (7)$$

Posteriormente, o G_{best} do enxame é atualizada por intermédio dos novos resultados de P_{best}. Este processo prossegue até que um critério de parada seja atendido. Para limitar a velocidade de uma partícula, evitando que o sistema extrapole o espaço de busca, são impostos limites, são eles:

$$V_i\,(t+1) = \begin{cases} v_{max,}, se \;\; V_i\,(t+1) > v_{max} \\ v_{min,}, se \;\; V_i\,(t+1) < v_{min} \end{cases} \quad (8)$$

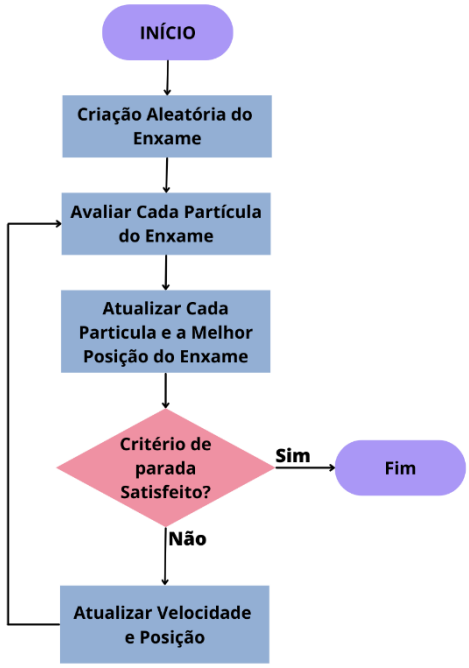

Figura 4. Fluxograma do PSO

Firefly Algorithm

O vagalume é um inseto conhecido por sua capacidade de bioluminescência. Quando adultos, além da alimentação, os vagalumes utilizam a sua luminescência para atrair os parceiros reprodutivos através da replicação de padrões luminosos [2]. Compreendendo o papel da bioluminescência na vida social dos vagalumes, foi proposto o *Firefly Algorithm* (FA), ou em português, Miríades de Vagalumes Artificiais. São permitidas algumas simplificações no processo de emulação do comportamento de atração dos vagalumes [20], são eles:

- Os vagalumes são unissex, ou seja, são atraídos por qualquer um dos indivíduos da população;
- O compartilhamento de informações entre os vagalumes é proporcional à atratividade
- Se não houver um Vagalume "mais brilhante" do que um outro em particular, este se moverá aleatoriamente;
- O brilho de um vagalume é proporcional ao valor da função objetivo.

Os vagalumes são distribuídos aleatoriamente no espaço de busca. Deste modo, cada vagalume do enxame deve representar uma solução para o problema em questão. Os Vagalumes Artificiais são representados por um vetor e um número escalar, respectivamente, a sua posição e a intensidade do seu brilho. O espaço factível é associado como o habitat dos vagalumes, sendo que cada indivíduo é visto "voando", e logo em seguida "pousando" em um ponto no espaço de busca [2].

A intensidade do brilho do vagalume é calculada usando o valor da sua posição na função objetivo. Deste modo, os vagalumes com os brilhos mais intensos são aqueles que ocupam as melhores regiões do ambiente. Como mostrado na Figura 5, o vagalume 2, em i=0, encontrasse mais próximo do ótimo global, por este motivo possui o maior brilho. Ao passar das iterações, os vagalumes "mais brilhosos" têm a capacidade de atrair com mais facilidade os outros indivíduos, levando a solução do problema de otimização [2], [21].

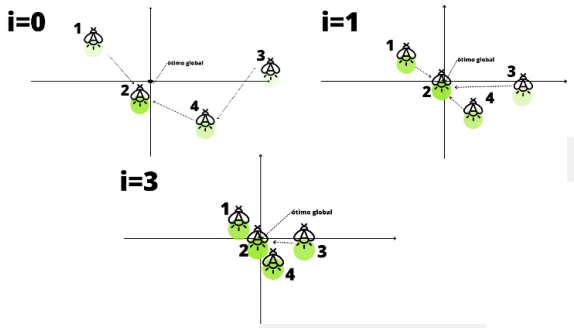

Figura 5. Atratividade dos Vagalumes

O funcionamento do FA pode ser resumido no fluxograma da Figura 6. A quantidade de luz emitida necessária para que um vagalume consiga atrair os outros depende de dois fatores: a distância entre os vagalumes e a taxa de absorção do meio. A intensidade percebida por um vagalume I_{ij} é dada por:

$$\vec{r_i} = [x_1, x_2, \dots, x_d] \tag{9}$$

$$I_i = f(\vec{r_i}) \tag{10}$$

$$I_{ij} = I_i e^{-\gamma r_{ij}^2} \tag{11}$$

Onde, $\vec{r_i}$ são o vetor de posição x_n do vagalume; d é a dimensão do problema; I_i é a intensidade da luz emitida. γ é o parâmetro de absorção da luz pelo meio; r é a distância euclidiana entre o vagalume "mais brilhante" i e o vagalume "menos brilhante" j. Por sua vez, a atratividade de um vagalume β_{ij} é expressa por:

$$\beta_{ij} = \beta_0 e^{-\gamma r_{ij}^2} \tag{12}$$

Sendo β_0 a sua atratividade inicial. A distância euclidiana r_{ij} entre dois vagalumes i e j, respectivamente, nas posições x_i e x_j é dado por:

$$r_{ij} = \left|\left|x_i - x_j\right|\right| = \sqrt{\sum_{k=1}^{d}(x_i, k - x_j k)^2} \tag{13}$$

Onde, x_i, k e x_j, k é o k-ésimo componente da coordenada espacial; x_i e x_j do i-nésimo vagalume e d é o número de dimensões. Já a movimentação de um vagalume j em direção a outro vagalume i "mais brilhante" é definida por:

$$x_j(t+1) = x_j(t) + \beta_{ji}(x_j - x_i) + \alpha\epsilon_i \tag{14}$$

Onde, x_i é a posição atual de um vagalume; t é a iteração atual da execução; $\beta_{ji}(x_j - x_i)$ é usado para representar a atratividade de um vagalume e a intensidade de luz vista por vagalumes adjacentes e $\alpha\epsilon_i$ é usado para o movimento aleatório de um vagalume, caso não exista nenhum outro vagalume "mais brilhante".

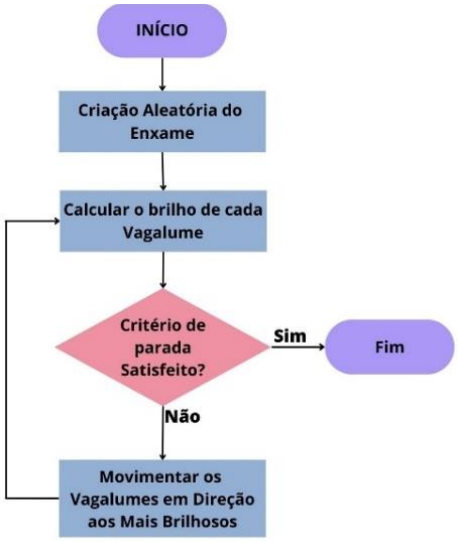

Figura 6. Fluxograma do Firefly Algorithm

III. Métricas de Desempenho

As metaheurísticas utilizam componentes estocásticos no processo de otimização, por este motivo é importante realizar uma avaliação estatística das soluções obtidas para o problema. Neste sentido, uma ou mais métrica são coletadas e comparadas em diferentes cenários de um sistema, afim de medir o seu desempenho. Comumente, são selecionadas a acurácia, a precisão e o tempo de processamento como métricas para a avaliação de desempenho em metaheurísticas. Como resultado da boa parametrização, o sistema consegue um desempenho consistente, baixo desvio padrão e a capacidade de convergência para uma solução viável [2], [22].

A. Acurácia e Precisão

A acurácia é definida como sendo a proximidade de um valor experimental com seu valor de referência verdadeiro, ou seja, tem a função de quantificar, mostrando consequentemente o grau de exatidão dos dados. A acurácia é definida em termos da média e desvio padrão das melhores aptidões alcançadas em todas as iterações. Desta forma, o sistema que tem uma média elevada dos melhores valores de aptidão com um baixo desvio padrão possuirá uma boa acurácia, de forma equacional, pode ser representada como:

$$Acurácia = \max(F_n),\ para\ \ n = 1, 2, \dots, N \tag{15}$$

$$Acurácia_{média} = \frac{\sum_{s=1}^{T} Acurácia}{T} \tag{16}$$

Onde, F_n é a aptidão, n melhores valores e T número total de simulações realizadas. Por sua vez, A precisão é o grau de consistência dos melhores resultados do algoritmo entre si. Portanto, quando menor a dispersão maior é a precisão, melhorando a estimativa de convergência das soluções.

$$Precisão = \frac{1}{\sigma_m + \Delta} \tag{17}$$

$$\sigma_m = \frac{\sigma_1 + \sigma_2 + \dots + \sigma_N}{N} \tag{18}$$

Onde, σ_m é o desvio padrão médio de cada parâmetro do problema que o algoritmo encontrou como solução; N é o número de parâmetros do problema e Δ é um número pequeno adicionado para que em caso ideal de desvio a equação não ocorra uma indeterminação matemática. Neste trabalho foi adotado $\Delta = 0{,}01$.

Sintetizando, a acurácia é a divergência entre o valor de referência e o valor estimado, e a precisão está ligada a dispersão deste valor [2], a Figura 7 faz uma analogia sobre estes conceitos e o jogo de Tiro ao alvo.

Figura 7. **Acurácia x Precisão**

B. Tempo de Simulação

Um ponto relevante para avaliação do desempenho de um algoritmo é o seu tempo de processamento. As soluções viáveis são aquelas que resolvam o problema com um tempo computacional aceitável. Neste trabalho, o tempo é contado a partir do início do laço principal até o momento em que um dos critérios de parada do algoritmo seja atingido. São considerados dois critérios: O Número Máximo de Iteração e o Erro Máximo Aceitável. O primeiro define o número de iterações máximas que poderão ser feitas pelo algoritmo dentro de uma simulação. Por sua vez, o Erro Máximo Aceitável está relacionado a qualidade obtida pelo melhor indivíduo do enxame, o que permite encerrar o algoritmo antes do número máximo de iterações seja atingido, reduzindo o tempo de execução.

C. Analises e Desempenho de Sistemas

Um sistema pode se caracterizado com um conjunto de interdependentes, de modo a formar um todo organizado. De forma geral, o sistema computacional recebe uma entrada para realizar um processamento e por fim retornar alguma saída. Além disto, todo sistema criado tem uma finalidade a ser atingida. Neste sentido, é necessário estudos para que o sistema faça a tarefa designada com a melhor performance, maior precisão, alta resistência e menor erro possível [22], [23].

Dentro das três técnicas de avaliação de desempenho: modelagem, simulação e medição; foi selecionado a última como a metodologia do presente trabalho, pois o objetivo do estudo é a comparação de diferentes algoritmos que estão disponíveis na literatura. Na medição, conhecida também como aferição, é possível capturar o desempenho do sistema em dado momento, com isso tornasse possível determinar o valor ideal de um parâmetro. Além disto, também é possível comparar de maneira justa dois ou mais sistemas.

Para aferição eficiente é necessário levar em consideração três objetos: Métricas, Parâmetros e Carga de Trabalho. Como apresentado no início desta seção, as métricas consistem nos critérios usados para avaliação do desempenho de um sistema. Já os parâmetros são os valores que alteram o comportamento do sistema. No processo de medição é escolhido os parâmetros que serão variados, enquanto os demais permanecem estáticos. Por fim, a carga de trabalho é a lista de requisições de que o sistema precisará lidar, ou seja, tem como missão os simular os cenários de uso em ambientes reais. Uma solução criada para realização de aferição dentro de um critério padronizado e confiável é chamado *benchmarking*. Normalmente as interpretações das informações obtidas durante o estudo de medição costumam ser extremamente úteis para compreender e ajustar o algoritmo, pois refletem um cenário real [22].

IV. Funções de Benchmarks

O *Benchmark*, ou teste de referência, é um conceito que surgiu no meio corporativo com intuito de comparar de forma eficaz parâmetros distintos, tendo como base padrões de testes similares. Com isso é possível comparar processos, objetos e resultados, selecionando o melhor modelo e preço para um serviço ou produto [24].

Em otimização, as funções de *benchmark* são instâncias de teste de problemas para os quais o valor ótimo é conhecido. Seu objetivo é desafiar o algoritmo, para isso, são comuns encontrar diferentes variedades de topologia. A relevância de seus resultados se dá pela proximidade entre a solução encontrada e o valor ótimo conhecido. Com isto, são aliadas na investigação de alguns aspectos importantes sobre os algoritmos, como a convergência prematura e a robustez [5].

As funções de *benchmark* podem ser classificadas em unimodais, quando apresentam um único máximo/mínimo em sua função. Ou multimodais, quando possuem vários máximos/mínimos locais, obrigando o algoritmo a evitar estes valores para encontrar o ótimo global. Neste trabalho foi utilizado a função unimodal Rosenbrock e as funções multimodais Rastrigin e Ackley para avaliar a robustez dos algoritmos, tempo de processamento, acurácia, precisão e possíveis vulnerabilidades no processo de otimização das funções.

A. Rosenbronk

Conhecida também como função Valley ou Banana, a Rosenbrock é uma função unimodal. O desenho dessa função possui formato semelhante a uma sela, conforme mostrada na Figura 8. Embora o vale seja de fácil descoberta, a convergência para ele não é trivial devido a inclinação ser muito pequena em sua curva, o que retarda a convergência na maioria dos algoritmos de otimização [17].

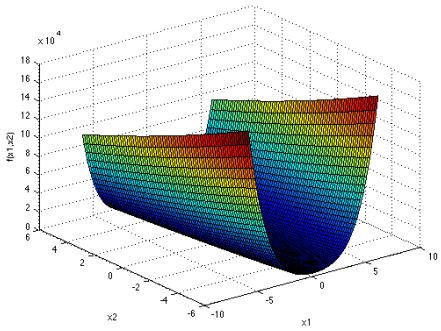

Figura 8. **Função Rosenbronk**

A função Rosenbrock é denotada pela (19). O mínimo global da função possui valor 0, e está localizado no fim do vale (vetor 1, ... ,1) conforme mostrado em (20).

$$f(X) = \sum_{i=1}^{d-1}[100(x_i + 1 - x_i^2)^2 + (x_i - 1)^2 \quad (19)$$

$$f(X^*) = 0, em\ X^* = (1, \dots, 1) \quad (20)$$

B. Rastrigin

A função Rastrigin, Figura 9, é uma função multimodal, sendo que seus mínimos locais são distribuídos regularmente e possuem alturas parecidas, o que pode dificultar o processo de busca pelo ótimo global, devido ao risco de o algoritmo facilmente ficar preso em um mínimo local [5].

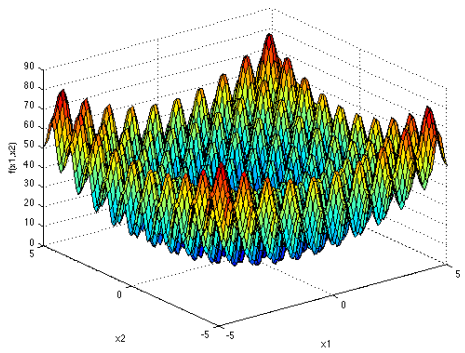

Figura 9. **Função Rastrigin**

Sua notação é definida na (21), sendo seu mínimo global igual a 0, localizado no vetor 0, ..., 0, conforme a (22).

$$f(X) = 10d + \sum_{i=1}^{d}[x_i^2 - 10\ \cos(2\pi x_i)] \quad (21)$$

$$f(X^*) = 0, em\ X^* = (0, \dots, 0) \quad (22)$$

C. Ackley

A função Ackley, Figura 10, é uma função multimodal caracterizada por uma região externa ondulada e um grande buraco no centro. A dificuldade na convergência decorre na facilidade com que os algoritmos de otimização são enganados pelos muitos mínimos locais encontrados durante a busca pelo mínimo global e induzidos a convergir prematuramente [25].

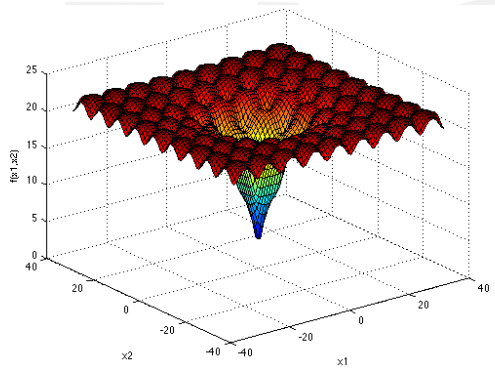

Figura 10. **Função Ackley**

Sua fórmula é representada na (23). Seu mínimo global está localizado no fim do buraco e é igual a 0, localizado no vetor 0, ..., 0, como visto na (24).

$$\mathrm{f}(X) = -\mathrm{a}\exp\left(-\mathrm{b}\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right) + \mathrm{a} + \exp(1) \quad (23)$$

$$sendo: \quad a = 20; \quad b = 0{,}2; \quad c = 2\pi$$

$$f(X^*) = 0, em\ X^* = (0, \dots, 0) \quad (24)$$

V. RESULTADOS

Esta seção apresenta a análise dos resultados do estudo paramétrico feito pelo presente trabalho. Inicialmente são apresentadas as especificações a qual os algoritmos foram submetidos. Em seguida são apresentados os cenários dos estudos de caso. Após isto, serão apresentados os resultados para os algoritmos PSO, FA e RW. Ao fim da seção, serão selecionados os melhores cenários e o melhor algoritmo de otimização entre os trabalhados.

A. Configurações dos Algoritmos e Simulações

Neste estudo, foi utilizado um computador com sistema operacional Windows 10 equipado com processador Intel i3, 6ª geração e 8 Gb de memória RAM. As simulações foram feitas utilizando o programa Matlab. Ao fim das simulações, os resultados foram exportados para uma planilha e analisado usando o programa Microsoft Excel.

No processo de busca, corriqueiramente, não é conhecido o intervalo onde está localizado o valor ótimo da função. Por este motivo, é necessário entender o funcionamento dos algoritmos em diferentes intervalos de busca. Para simular esta carga de trabalho, serão usadas três dimensões diferentes de intervalo, são elas: "P", "M" e "G", conforme mostrado na Tabela I. O Espaço de busca "P" possui um intervalo de -15 a +15 do ponto ótimo. Por sua vez, O Espaço de busca "M" possui um intervalo de -40 a +40 do valor ótimo da função. Por fim, O Espaço de busca "G" possui um intervalo de -100 a +100 do melhor valor da função.

Tabela I. **Espaço de busca**

Espaço	Intervalo de busca
P	-15 a +15
M	-40 a +40
G	-100 a +100

Para uma análise estatística, cada cenário foi repetido em 10 simulações. Os resultados apresentados nas próximas seções são as médias de cada métrica. Em cada simulação, foram utilizados dois critérios de parada: O primeiro é o Erro Máximo Aceitável definido como $1 * 10^{-15}$. Caso este critério não seja alcançado, o segundo critério de parada é acionado, sendo, o número máximo de interação, definido como 15.000. Estes valores foram selecionados para trazer uma maior dificuldade na execução do algoritmo, com isto, é possível fazer uma avaliação mais assertiva do desempenho em relação ao tempo de processamento, à acurácia e à precisão do algoritmo. A Tabela II ilustra os critérios de paradas utilizados em todos os algoritmos.

Tabela II. **Parâmetros para critério de parada**

Parâmetros	Valores
Nº de simulações	10
Nº de interações	15000
Erro Máximo Aceitável	$1 * 10^{-15}$

B. Estudo de Caso

Neste estudo de caso foram trabalhados a Heurística *Random Walk* e os algoritmos de inteligência de enxame *Particle Swarm Optimization* e *Firefly Algorithm*. Para avaliação destes, foram usadas as funções de *benchmarks*,

sendo elas: a unimodal Rosenbrock e as multimodais Ackley e Rastrigin.

O fluxo do estudo de caso se comportou da seguinte forma. Primeiramente foi selecionado um parâmetro do algoritmo para ser variado enquanto os demais eram mantidos estáticos. Foram testados seis, sete ou oito valores diferentes para cada parâmetro. Em seguida, para cada parâmetro, foi usado as três funções de *benchmark*, Ackley, Rosenbrock e Rastrigin. Por fim, em cada função foram usados os três intervalos de espaço de busca: "P", "M" e "G". A síntese do funcionamento do estudo de caso é mostrada na Figura 11.

Figura 11. Fluxo do estudo de caso

C. Simulações PSO

No estudo de caso do PSO, foram variados quatro parâmetros, são eles: tamanho da população, fator inercial (w), fator cognitivo (c_1) e fator social (c_2). Os sete valores trabalhados em cada parâmetro estão disponíveis na Tabela III. Enquanto um parâmetro era variado, os demais permaneciam estáticos, sendo os seus valores padrões estão destacados em negrito na tabela. Ademais, considerando a relação do fator cognitivo e fator social, foram adicionados cenários onde os dois parâmetros eram ajustados simultaneamente. Neste sentido, considerando as três funções de *benchmarks* utilizadas, aliada aos três espaços de buscas diferentes, tem como resultado um total de 315 cenários de simulações para o estudo de caso do PSO.

Tabela III. Valores usados por cada Parâmetro do PSO

Parâmetros	Valores
população	5; 10; 15; **20**; 25; 40 e 100
w	0,0; **0,4**; 0,8; 1,2; 1,4; 2,0 e 4,0
c_1	0,0; 0,5; 1,0; **2,0**; 2,5; 3,0 e 4,0;
c_2	0,0; 0,5; 1,0; **2,0**; 2,5; 3,0 e 4,0;

Variação do fator cognitivo

O fator cognitivo c_1 define o nível de confiança que uma partícula possui nas suas experiências passadas. Neste trabalho, foram utilizados os seguintes valores: 0,0; 0,5; 1,0; 2,0; 2,5; 3,0 e 4,0; com resultado, são obtidos 63 cenários.

Na Figura 12 é apresentada a acurácia no espaço de busca "G" para ilustrar o comportamento do parâmetro. Os valores cognitivos 0,0 e 1,0 prejudicaram o desempenho das funções Rosenbrock e Rastrigin devido à falta de confiança da partícula em suas decisões. Por sua vez, quando o valor é aumentado para 4,0, a uma piora em todas as funções, pois as partículas tendem a ser comportar egoistamente, ignorando os conhecimentos adquiridos por outros indivíduos. Em contraponto a isto, os melhores resultados para a acurácia ocorrem quando os valores estão nas regiões intermediarias dos gráficos. Neste sentido, apesar de não alcançar um bom desempenho em todas as cargas de trabalho, o valor de 2,5 teve os melhores resultados com baixo desvio, conforme mostrado na Tabela IV.

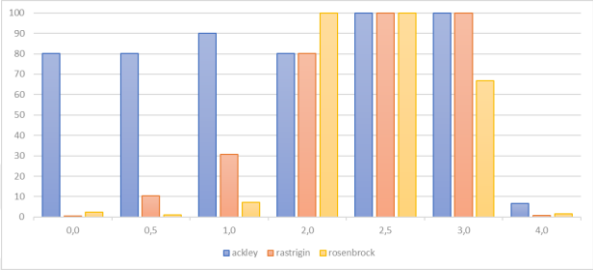

Figura 12. Acurácia do PSO em C1 no espaço de busca "G"

Tabela IV. Desvio da acurácia em C1

ACURACIA		0,0	0,5	1,0	2,0	2,5	3,0	4,0
Ackley	P	0,382	0,000	0,000	0,000	**0,000**	0,000	0,007
	M	0,000	0,000	0,000	0,000	**0,000**	0,000	0,026
	G	6,400	6,400	3,600	6,400	**0,000**	0,000	0,095
Rastrigin	P	3,741	1,154	1,194	0,318	**0,179**	0,000	0,056
	M	4,219	1,930	1,035	0,179	**0,179**	0,000	0,451
	G	2,686	1,214	0,537	0,318	**0,000**	0,000	0,698
Rosenbrock	P	2,724	2,531	3,075	0,000	**0,000**	0,000	0,051
	M	7,999	5,889	5,142	0,000	**0,000**	0,001	2,674
	G	22,221	9,446	3,012	0,000	**0,000**	5,350	2,768

De forma geral, há uma queda no desempenho dos algoritmos em relação a precisão conforme é aumentado o espaço de busca, devido a maior dificuldade do PSO em localizar o ótimo global. Deste modo, conforme mostrado no espaço de busca "G", Figura 13, obtém-se os melhores desempenhos apenas em c_1 em 2,5.

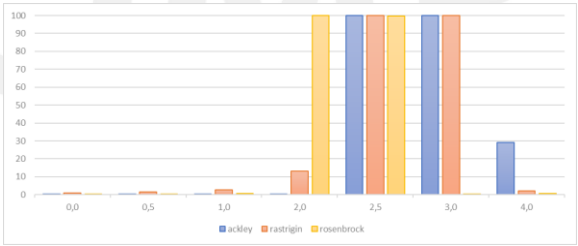

Figura 13. Precisão do PSO em C1 no espaço de busca "G"

Em relação ao tempo médio de processamento, não foram obtidos resultados que associem fortemente o fator cognitivo a alteração desta métrica. Para o melhor parâmetro, c_1 em 2,5 nos espaços de busca "P", "M" e "G" obteve os tempos, respectivamente, na função Ackley 30,6s; 0,49s e 0,62s, na função Rastrigin 2,8s; 31,4s e 0,60s, e na função Rosenbrock 28,8s; 35,8s e 30,7s.

variação fator social

O fator social c_2 influencia a partícula seguir o melhor resultado do enxame em cada iteração. Neste trabalho, foram usados os valores de 0,0; 0,5; 1,0; 2,0; 2,5; 3,0 e 4,0, totalizando 63 cenários diferentes para este parâmetro.

De forma semelhante ao fator cognitivo, os piores resultados em acurácia¸ Figura 14 e precisão, Figura 15, do fator social estão nos extremos dos gráficos. O valor de parâmetro 0,0; 0,5 e 1,0 prejudicou a troca de informações entre as partículas, consequentemente, afetou a busca do valor ótimo global, além disto, nestes valores são encontrados os maiores desvios médios da acurácia, segundo a Tabela V. Enquanto o valor de fator social mais alto, em 4,0, tornou as partículas muito sociáveis, provocando uma conversão prematura ao maior valor de ótimo local encontrado. Neste sentido, de forma geral, o melhor resultado foi quando c_2 estava em 2,5.

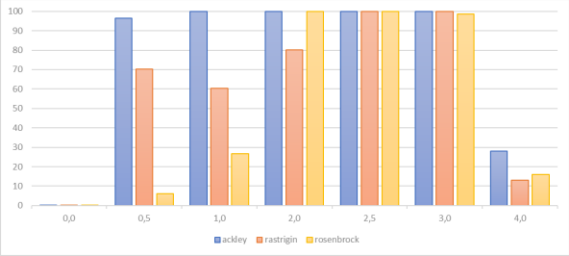

Figura 14. **Acurácia do PSO em C2 no espaço de busca "P"**

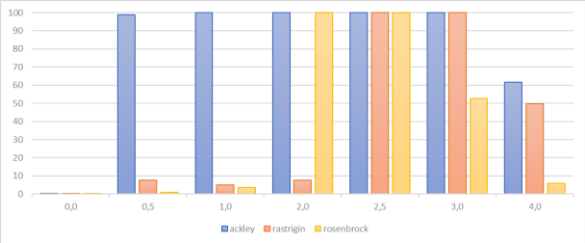

Figura 15. **Precisão do PSO em C2 no espaço de busca "P"**

Tabela V. **Desvio da acurácia em C1**

ACURACIA		0,0	0,5	1,0	2,0	2,5	3,0	4,0
Ackley	P	1,7499	0,0007	0,0000	0,0000	**0,0000**	0,0000	0,0136
	M	2,2270	5,2977	14,5782	4,7656	**5,5561**	6,1617	0,9051
	G	1,2137	1,1207	3,5650	0,0000	**3,6000**	6,3999	8,2034
Rastrigin	P	18,8139	0,4179	0,5970	0,3184	**0,0000**	0,0000	0,0409
	M	138	0,1791	0,3184	0,3184	**0,0000**	0,0000	0,2735
	G	722	1,8705	0,1791	0,4179	**0,0000**	0,0000	0,5921
Rosenbrock	P	5157	55,8757	0,3041	0,0000	**0,0000**	0,0001	0,0319
	M	202367	4,4343	1,7606	0,0000	**0,0000**	0,0004	5,6321
	G	20960509	20713	1,1730	17,8711	**0,0000**	15,5885	36,0741

Em relação ao tempo médio de processamento, também não foram obtidos resultados que associem fortemente o fator social a alteração desta métrica. Para o melhor parâmetro, c_2 em 2,5 nos espaços de busca "P", "M" e "G" obteve os tempos, respectivamente, na função Ackley 1,5s; 0,89s e 30,3s, na função Rastrigin 2,33s; 0,67s e 0,97s, e na função Rosenbrock 26,2s; 30,5s e 23,4s.

variação fator cognitivo igualado ao fator social

Considerando as características de intensificação e diversificação presentes no fator cognitivo (c_1) e fator social (c_2), também foram elaborados cenários onde os dois parâmetros são alterados simultaneamente. Neste sentido o fator cognitivo e fator social foram igualados e alterados para os valores de 0,0; 0,5; 1,0; 2,0; 2,5; 3,0 e 4,0, com resultado foram testados 63 cenários diferentes.

Em relação a acurácia, Figura 16, e a precisão, Figura 17, notasse que os valores baixos de c_1 e c_2, em 0, prejudicam o desempenho do algoritmo, visto que a população inicial é criada de maneira aleatória. Com baixo valores dos fatores cognitivo e social, o algoritmo não consegue desenvolvesse, mantendo o movimento na mesma direção criada inicialmente. Conforme o Valor dos parâmetros são acrescentados, há uma melhora do desempenho até o melhor parâmetro em 2,5. A partir disto, 3,0 e 4,0, notasse novamente uma queda de desempenho do algoritmo. Como mostrado na Tabela VI, observasse valores de desvios de acurácia maiores, do que quando os parâmetros eram alterados individualmente, acentuando este problema, principalmente, no espaço de busca "G".

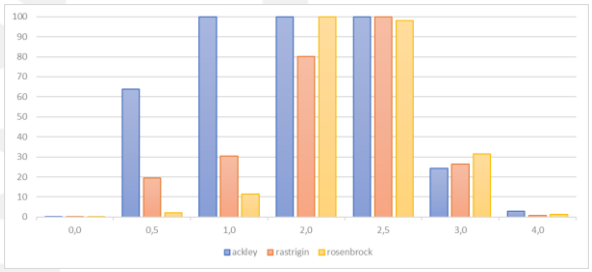

Figura 16. **Acurácia do PSO em C1=C2 no espaço de busca "P"**

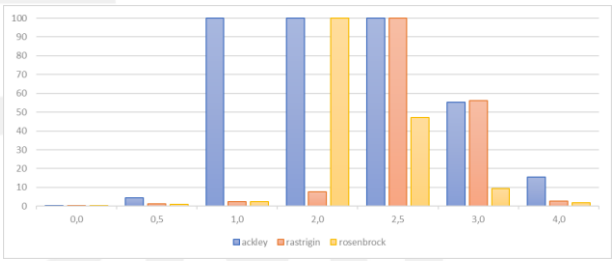

Figura 17. **Precisão do PSO em C1=C2 no espaço de busca "P"**

Tabela VI. **Desvio da acurácia em C1=C2**

ACURACIA		0,0	0,5	1,0	2,0	2,5	3,0	4,0
Ackley	P	0,9946	0,9452	0,0000	0,0000	**0,0000**	0,0217	0,2144
	M	3,1316	1,0434	0,0000	0,0000	**0,0000**	0,0316	0,4432
	G	0,9883	5,8428	3,5987	3,6000	**6,4000**	3,5342	0,6013
Rastrigin	P	17,5502	1,5921	0,8756	0,3184	**0,0000**	0,0464	0,4184
	M	146,4105	3,8206	1,2337	0,3184	**0,0000**	0,4116	1,2764
	G	733,1104	1,9090	0,8756	0,3184	**0,0000**	0,6342	1,9849
Rosenbrock	P	8649	3,8315	32,8030	0,0000	**0,0002**	0,0323	0,6736
	M	186931	4058	4,8397	5,6760	**5,6750**	0,0668	1,3850
	G	28516082	54133	20,9996	0,0000	**0,0054**	0,9462	29,8925

O tempo de processamento é mostrado na Figura 18. Notasse que os valores do centro do gráfico possuem os menores tempos médios devido ao critério de parada do Erro Máximo Aceitável ser atingido algumas vezes, diminuindo o tempo necessário para ser encerrada a iteração. Além disto, também foi observado um menor desvio no tempo nos valores de 2,5, conforme Tabela VII.

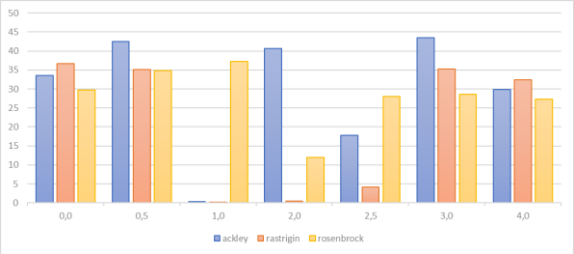

Figura 18. Tempo do PSO em C1=C2 no espaço de busca "P"

Tabela VII. Desvio do tempo em C1=C2

TEMPO		0,0	0,5	1,0	2,0	2,5	3,0	4,0
Ackley	P	4,34	5,84	19,07	18,65	**5,54**	4,69	2,62
	M	1,84	1,06	16,56	13,12	**8,73**	1,40	1,99
	G	1,19	2,36	15,12	16,35	**12,05**	2,85	1,82
Rastrigin	P	1,17	4,26	14,15	12,36	**0,68**	1,44	1,87
	M	10,73	0,68	11,69	20,02	**1,92**	2,84	2,73
	G	10,67	5,84	13,65	24,19	**2,60**	11,40	4,28
Rosenbrock	P	1,02	1,86	2,48	1,71	**2,15**	1,44	1,38
	M	1,94	1,83	4,51	3,37	**2,21**	0,48	3,73
	G	1,01	3,66	3,25	2,84	**1,93**	0,97	2,97

variação fator inercial

O fator inercial *w* influencia o quanto uma partícula confia na direção que está seguindo em um determinado momento. Neste trabalho, foram usados os seguintes valores: 0,0; 0,4; 0,8; 1,2; 1,4; 2,0 e 4,0, formando um total de 63 cenários diferentes.

Na acurácia, Figura 19, e precisão, Figura 20, notasse que, quando valores do fator inercial são 1,2 e 1,4 apresentaram péssimos resultados. Provavelmente isto foi causado pela zona de transição entre uma relação inercial forte, valores mais altos de *w*, e no outro ponto uma relação inercial mais fraca, próxima do valor zero em *w*, causando uma oscilação no comportamento da partícula, impedindo que nestes valores o algoritmo consiga se desenvolver corretamente. Por sua vez, o melhor resultado em acurácia pode ser considerado quando o valor *w* está em 0,8, pois obteve bons resultados na maioria dos cenários com baixos valores de desvio da acurácia, Tabela VIII.

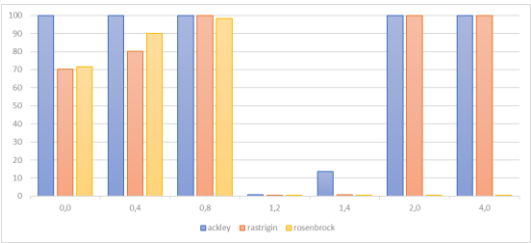

Figura 19. Acurácia do PSO em w no espaço de busca "M"

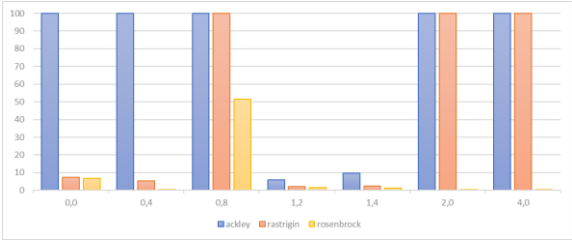

Figura 20. Precisão do PSO em w no espaço de busca "M"

Tabela VIII. Desvio da acurácia em W

ACURACIA		0,0	0,4	0,8	1,2	1,4	2,0	4,0
Ackley	P	0,0000	0,0000	**0,0000**	0,0623	0,2223	0,0000	0,0000
	M	0,0000	0,0000	**0,0000**	0,5662	0,8821	0,0000	0,0000
	G	0,0000	3,6000	**3,6000**	5,6432	0,8179	3,6000	6,4000
Rastrigin	P	0,4776	0,3184	**0,0000**	0,3412	0,3202	0,0000	0,0000
	M	0,4179	0,4776	**0,0000**	0,9633	1,2247	0,0000	0,0000
	G	0,3184	0,3184	**0,0000**	1,3413	1,3413	0,0000	0,0000
Rosenbrock	P	0,0443	0,0000	**0,0000**	0,1783	0,4677	2,3493	1,3927
	M	0,0736	7,4971	**0,0002**	1,3730	1,0859	16,5750	82,6536
	G	0,0734	17,8711	**0,0014**	14,9262	52,0836	66,0468	176379

Não foi encontrada uma relação entre fator inercial e o tempo médio de processamento, para o melhor parâmetro, *w* em 0,8, nos espaços de busca "P", "M" e "G" obteve os tempos, respectivamente, na função Ackley 12,7s; 14,8s e 11,0s, na função Rastrigin 4,96; 6,05s e 5,23s, e na função Rosenbrock 29,7s; 31,4s e 22,5s.

variação tamanho da população

O PSO faz uso de uma abordagem baseada em população, sendo que cada indivíduo representa um potencial solução para o problema. Neste trabalho, foram utilizados os valores de tamanho de população em 5; 10; 15; 20; 25; 40 e 100, totalizando 63 cenários.

Em relação a acurácia, Figura 21, e precisão, Figura 22, existe uma tendência de queda do desempenho do algoritmo em intervalo de busca maiores para o mesmo tamanho de população, devido a necessidade de mais indivíduos para a realização de uma busca satisfatória. Por este motivo, os piores resultados estão nos parâmetros com as menores populações. Neste sentido, considerando os tamanhos de busca "P", "M" e "G" nas funções de *benchmarks*, o melhor resultado, com baixo desvio da acurácia na Tabela IX, ocorre quando o tamanho de população é de 100.

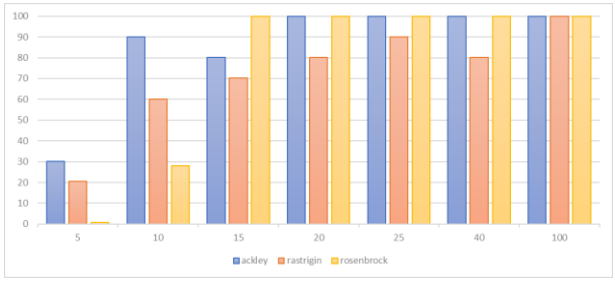

Figura 21. Acurácia do PSO em população no espaço de busca "G"

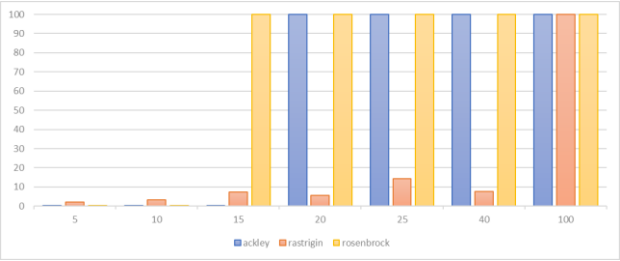

Figura 22. **Precisão do PSO em população no espaço de busca "G"**

Tabela IX. **Desvio da acurácia em população**

ACURACIA		5	10	15	20	25	40	100
Ackley	P	0,0000	0,3816	0,0000	0,0000	0,0000	0,0000	**0,0000**
	M	0,8904	0,0000	0,0000	0,0000	0,0000	0,0000	**0,0000**
	G	9,1108	3,6000	6,4000	0,0000	0,0000	0,0000	**0,0000**
Rastrigin	P	1,3530	0,5970	0,4776	0,1791	0,3184	0,0000	**0,0000**
	M	1,9103	0,7164	0,5970	0,3184	0,4179	0,0000	**0,0000**
	G	0,5360	1,3332	0,4179	0,4776	0,1791	0,3184	**0,0000**
Rosenbrock	P	3,5040	1,1151	1,6534	0,0000	0,0000	0,0000	**0,0000**
	M	3,7862	1,5646	0,0000	0,0000	0,0000	0,0000	**0,0000**
	G	1355	5,8510	0,0000	0,0000	0,0000	0,0000	**0,0000**

Existe uma tendencia de aumento do tempo médio de processamento conforme é aumentado o tamanho da população, Figura 23. Entretanto, não ocorreu um aumento demasiado no tempo de processamento no tamanho de população igual a 100. Da mesma forma, também não foi observado um desvio alto no tempo de processamento, conforme mostrado na Tabela X.

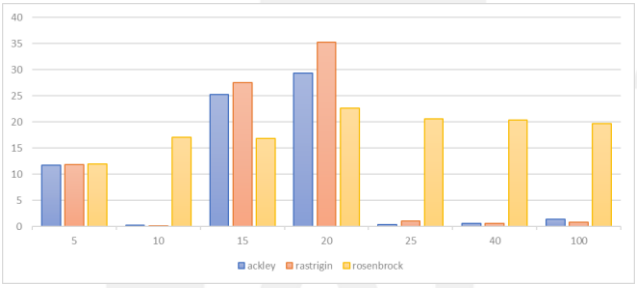

Figura 23. **Tempo do PSO em população no espaço de busca "G"**

Tabela X. **Desvio do tempo em população**

ACURACIA		5	10	15	20	25	40	100
Ackley	P	2,93	10,43	14,80	19,09	15,93	9,49	**0,05**
	M	3,63	10,08	5,84	11,24	12,75	23,46	**0,09**
	G	2,16	9,21	11,21	9,73	20,40	17,41	**0,08**
Rastrigin	P	1,25	9,42	14,93	6,09	11,58	0,20	**0,14**
	M	2,21	8,66	12,56	10,66	15,29	0,23	**0,18**
	G	3,64	9,12	11,02	11,02	8,91	17,36	**0,07**
Rosenbrock	P	1,03	0,89	3,10	1,25	1,09	1,63	**2,50**
	M	0,95	1,65	1,21	3,00	1,83	3,37	**3,64**
	G	0,26	0,70	2,50	3,39	3,06	1,89	**2,83**

D. Simulações FA

O *Firefly Algorithm* (FA) utiliza três parâmetros, são eles: atratividade inicial (β_0), absorção da luz pelo meio (γ) e tamanho de população. No estudo de caso do FA foi ajustado apenas um parâmetro por vez, sendo que os valores estão disponíveis na Tabela XI, enquanto os demais parâmetros eram mantidos inalterados, seus valores padrões estão expostos em negrito na tabela. Neste sentido, foram utilizados sete valores diferentes para o tamanho da população e oito valores para os parâmetros β_0 e γ. Para avaliação, foram utilizadas as três funções de *benchmark* em três intervalos de busca diferentes, totalizando 207 cenários de simulação.

Tabela XI. **Valores usados por cada Parâmetro do FA**

Parâmetros	Valores
população	5; 10; 15; 20; **25**; 40 e 100
β_0	0,1; 0,5; 1,0; 1,5; **2,0;** 2,5; 3,0 e 4,0;
γ	0,0; 0,1; 0,2; 0,3; **0,4**; 0,5; 0,8 e 1,0

variação β_0

A atratividade inicial (β_0) é o parâmetro responsável por definir o brilho percebido pelos vagalumes. Foram testados os valores de 0,1; 0,5; 1,0; 1,5; 2,0; 2,5; 3,0 e 4,0, totalizando 72 cenários diferentes.

Como mostrado na Acurácia, Figura 24 e precisão, Figura 25, no espaço de busca "M". Os piores desempenhos estão no valor de parâmetro em 0,1 devido à baixa capacidade de atração do vagalume. Conforme a atratividade inicial é aumentada, o desempenho do algoritmo também aumenta até o valor em β_0 2,0, obtendo também os menores desvios em acurácia, Tabela XII, onde é obtido os melhores resultados deste parâmetro. A partir disto, o valor de atratividade maior, tornasse um problema, pois podem ocorrer conversões prematuras em valores de ótimos locais em funções multimodais, prejudicando o algoritmo.

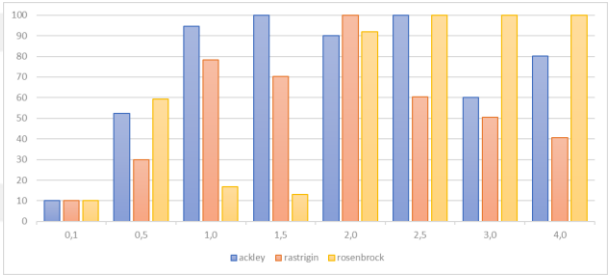

Figura 24. **Acurácia do FA em β_0 no espaço de busca "G"**

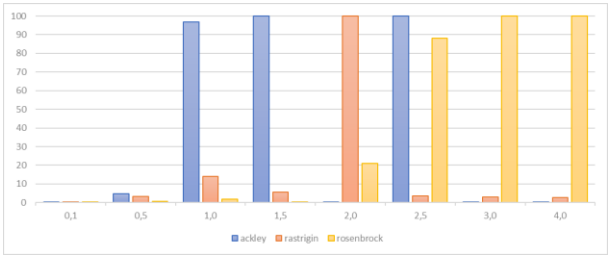

Figura 25. **Precisão do FA em β_0 no espaço de busca "G"**

Tabela XII. Desvio do acurácia em β_0

ACURACIA		0,1	0,5	1,0	1,5	2,0	2,5	3,0	4,0
Ackley	P	1,6637	0,0003	0,0074	0,0000	**0,0000**	0,0000	0,00	0,00
	M	4,6431	0,0000	0,0979	0,0000	**0,0000**	0,0000	0,00	0,00
	G	3,5129	1,0866	0,0021	0,0000	**3,6000**	0,0000	9,60	6,40
Rastrigin	P	6,8642	0,6201	0,3681	0,4776	**0,9104**	0,5970	0,42	0,52
	M	5,4070	0,3654	0,0059	0,4179	**0,4179**	0,3184	0,60	0,64
	G	112	0,5361	0,1894	0,4180	**0,0000**	0,5970	0,60	0,64
Rosenbrock	P	97,5467	0,3382	0,0684	1,0819	**0,0000**	0,0000	0,00	0,00
	M	33182	0,1195	0,6811	0,5113	**0,0000**	0,0000	0,00	7,50
	G	29361	3,5945	1,2143	7,0735	**0,0082**	0,0000	0,00	0,00

Ademais, não foram obtidos resultados que associem fortemente o β_0 a alteração no tempo de processamento, sendo o melhor parâmetro, em β_0 igual 2,5 nos espaços de busca "P", "M" e "G" obteve os tempos, respectivamente, para a função Ackley 0,67s; 1,29s e 33,66s, na função Rastrigin 35,67s; 0,52s e 35,55s e na função Rosenbrock 3,57s; 9,59s e 3,77s.

variação γ

No parâmetro absorção da luz pelo meio (γ) foram testados os valores de 0,0; 0,1; 0,2; 0,3; 0,4; 0,5; 0,8 e 1,0, totalizando 72 cenários diferentes. Considerando os valores selecionados para analise, não foram encontrados resultados expressivos na alteração do desempenho nas funções Ackley, Rastrigin e Rosenbrock nos espaços de busca "P", "M" e "G". Neste sentido, considerasse o valor padrão em 0,4 em γ como melhor parâmetro.

variação populacional

O FA também utiliza uma abordagem baseada em população, onde cada indivíduo é um potencial solução para o problema. Os valores analisados neste estudo foram 5, 10, 15, 20, 25, 40 e 100, totalizando 63 cenários.

A Figura 26 aborda os resultados de acurácia e a Figura 27 apresenta os resultados de precisão em relação ao tamanho da população "G". Observasse que a população de 40 e 100 indivíduos obtiveram um ótimo aproveitamento nas simulações, além de um baixo desvio de acurácia, Tabela XIII, isto é causado devido ao número maior de indivíduos que permite uma busca mais eficaz.

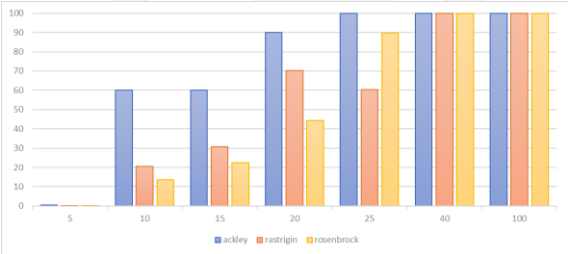

Figura 26. Acurácia do FA em população no espaço de busca "G"

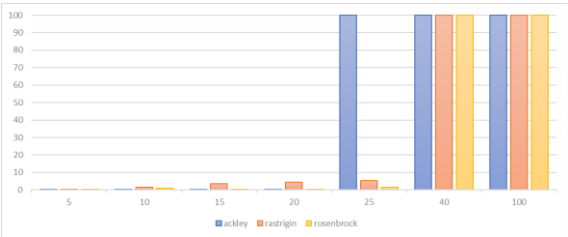

Figura 27. Precisão do FA em população no espaço de busca "G"

Tabela XIII. Desvio da acurácia em população

ACURACIA		5	10	15	20	25	40	100
Ackley	P	3,7494	0,0237	0,0000	0,0000	0,0000	0,0000	**0,0000**
	M	3,2764	0,3816	0,0000	0,0000	0,0000	0,0000	**0,0000**
	G	7,9892	8,4112	9,6000	3,6000	0,0000	0,0000	**0,0000**
Rastrigin	P	14,15	1,6316	0,9552	0,4776	0,4776	0,0000	**0,0000**
	M	44,33	2,9849	1,4725	0,7164	0,6965	0,0000	**0,0000**
	G	209	1,0232	0,4179	0,8358	0,4776	0,0000	**0,0000**
Rosenbrock	P	508	2,1585	0,1855	0,4498	0,0000	0,0000	**0,0000**
	M	49544	0,8347	0,1294	0,9104	0,0000	0,0000	**0,0000**
	G	179759	2,7179	7,5227	17,1290	0,7423	0,0000	**0,0000**

Em tempo médio de processamento, Figura 28 e seus desvios, Tabela XIV, notasse uma tendência de alta quando o tamanho da população é aumentado até 25 indivíduos. Nos valores de 40 e 100 notasse uma queda no tempo de processamento pois nestes cenários a simulação foi encerrada, em grande parte, pelo critério de parada do Erro Máximo Aceitável. Considerando todas as cargas de trabalho, o melhor valor de tamanho de população é o de 100 indivíduos.

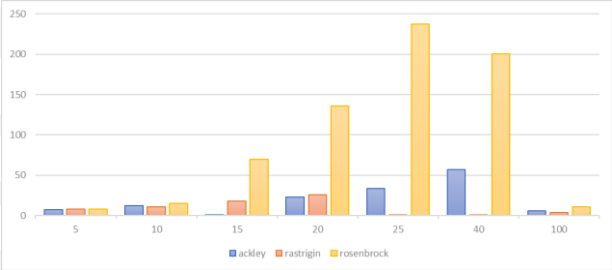

Figura 28. Tempo do FA em população no espaço de busca "G"

Tabela XIV. Desvio do tempo em população

ACURACIA		5	10	15	20	25	40	100
Ackley	P	0,49	5,34	7,70	15,10	17,44	34,81	**112,24**
	M	2,19	5,17	3,55	16,44	16,08	40,38	**66,21**
	G	1,74	0,27	5,57	4,47	14,60	22,89	**93,53**
Rastrigin	P	0,50	2,36	9,15	15,69	15,69	3,93	**21,77**
	M	0,46	3,74	5,03	12,99	13,96	4,23	**23,46**
	G	0,41	2,21	6,77	10,16	15,04	5,01	**24,04**
Rosenbrock	P	0,94	2,72	26,42	25,42	31,75	127,33	**18,55**
	M	1,00	2,22	30,13	38,60	45,76	91,22	**29,68**
	G	1,28	2,04	23,77	41,80	47,01	168,49	**26,60**

E. Simulações RW

No *Random Walk* (RW) é trabalhado três parâmetros, são eles: tamanho da população, tamanho do passo e taxa de substituição dos indivíduos. Da mesma forma que o FA e PSO, foram feitos testes alterando apenas um parâmetro por vez. Os valores testados estão disponíveis na Tabela XVI, enquanto os valores padrões são apresentados em negrito na tabela. Neste sentido, foram usados sete valores diferentes para o tamanho de população e seis valores no tamanho de passo e na taxa de substituição. Cada parâmetro foi avaliado usando as três funções de *benchmark* com os três espaços de busca diferentes.

Com resultado, para o RW foram realizado um total de 171 cenários de simulações.

Tabela XV. Valores usados por cada Parâmetro do RW

Parâmetros	Valores
população	5; 10; 15; 20; **25**; 40 e 100
passo	0,1; 0,5; **0,8**; 1,0; 1,5 e 2,0;
taxa de substuição	10%; 15%; **20%**; 30%; 40% e 50%;

variação do passo

O passo define o tamanho do deslocamento executado pelo algoritmo em cada iteração. Os valores testados neste parâmetro foram: 0,1; 0,5; 0,8; 1,0; 1,5 e 2,0, totalizando 54 simulações executadas.

A Figura 29 apresenta a acurácia e a Figura 30 aborda a precisão no tamanho de passo no espaço de busca "P". A função unimodal Rosenbrock possui uma superfície de decaimento lenta, o que torna o tamanho de passo em 0,1 o melhor valor de parâmetro. Por sua vez, as funções multimodais Ackley e Rastrigin também apresentam bons resultados em tamanho de passo pequeno, mas com desempenho mais tímido.

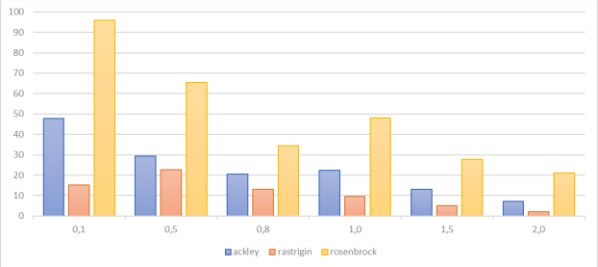

Figura 29. Acurácia do RW em passo no espaço de busca "P"

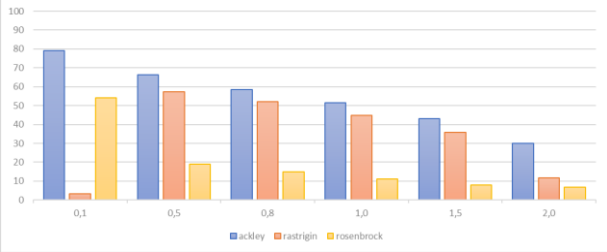

Figura 30. Precisão do RW em passo no espaço de busca "P"

Tabela XVI. Desvio da acurácia em tamanho de passo

ACURACIA		0,1	0,5	0,8	1,0	1,5	2,0
Ackley	P	0,0065	**0,0067**	0,0155	0,0205	0,0255	0,05
	M	0,4382	**0,0096**	0,0143	0,0199	0,0306	0,04
	G	0,8187	**0,0063**	0,0127	0,0155	0,0309	0,03
Rastrigin	P	0,4163	**0,0199**	0,0248	0,0573	0,1706	0,17
	M	1,1501	**0,0226**	0,0474	0,0407	0,1456	0,25
	G	4,4045	**0,0177**	0,0383	0,0398	0,1164	0,16
Rosenbrock	P	**0,0003**	0,0029	0,0111	0,0122	0,0170	0,04
	M	**0,0002**	0,0022	0,0071	0,0100	0,0350	0,05
	G	**0**	0,0048	0,0114	0,0254	0,0318	0,09

Sobre o tempo médio de processamento, não foi observado a relação entre o tamanho de passo está métrica, sendo o melhor parâmetro, em passo igual 0,5, nos espaços de busca "P", "M" e "G" obteve os tempos, respectivamente, para a função Ackley 24,5s; 18,3s e 19,5s, na função Rastrigin, 23,9s; 27,4s e 29s e na função Rosenbrock 18,6; 19,8s e 43s.

variação da taxa de substituição

A taxa de substituição define a porcentagem dos piores indivíduos que serão trocados após cada geração. Foram testados os valores de substituição em 10%; 15%; 20%; 30%; 40% e 50%, totalizando 54 simulações executadas.

Na Acurácia, Figura 31, e Precisão, Figura 32, notasse uma tendencia de queda no desempenho destas métricas quando a taxa está em 50%, principalmente na função Rosenbrock, isto significa que uma alta taxa de troca pode comprometer a evolução dos melhores indivíduos. Do outro lado, o melhor valor de taxa de substituição foi 15%, onde também apresentou um baixo desvio na Tabela XVII.

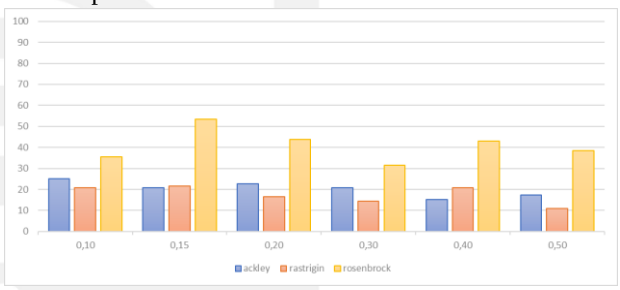

Figura 31. Acurácia do RW em substituição no espaço de busca "M"

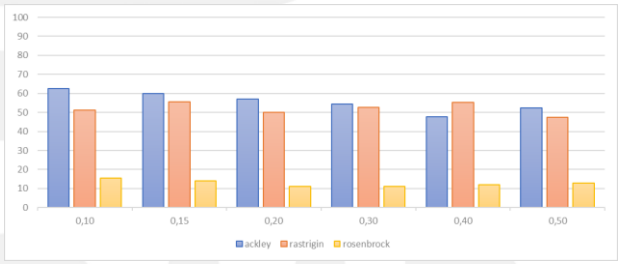

Figura 32. precisão do RW em substituição no espaço de busca "M"

Tabela XVII. Desvio da acurácia em taxa de substituição

ACURACIA		10%	15%	20%	30%	40%	50%
Ackley	P	0,0074	**0,0153**	0,0127	0,0132	0,0111	0,0168
	M	0,0081	**0,0096**	0,0149	0,0177	0,0221	0,0126
	G	0,0180	**0,0133**	0,0148	0,0099	0,0151	0,0168
Rastrigin	P	0,0708	**0,0259**	0,0315	0,0505	0,0517	0,0420
	M	0,0603	**0,0324**	0,0474	0,0515	0,0508	0,0445
	G	0,0327	**0,0466**	0,0453	0,0349	0,0336	0,0412
Rosenbrock	P	0,0036	**0,0087**	0,0080	0,0080	0,0050	0,0062
	M	0,0102	**0,0109**	0,0123	0,0194	0,0136	0,0139
	G	0	**0,0086**	0,0112	0,0366	0,0179	0,0378

Em relação ao tempo médio de processamento, o melhor parâmetro, em taxa de substituição em 15% nos espaços de busca "P", "M" e "G" obteve os tempos, respectivamente, para a função Ackley 22,2s; 22,3s e 23,2s, na função Rastrigin 22,2s; 24s e 25s e função Rosenbrock 30s; 17,9s e 27,5s.

variação populacional

Cada indivíduo é considerado uma possível solução para o problema de otimização. Neste sentido, para o RW foram testados os tamanhos de população em 5; 10; 15; 20; 25; 40 e 100, totalizando 63 simulações executadas.

Os resultados de acurácia, Figura 33, precisão, Figura 34, e os desvios de acurácia, Tabela XIX, demostram que uma população maior aumenta o desempenho, pois o algoritmo consegue explorar mais pontos, neste sentido, o melhor valor de tamanho de população é 100.

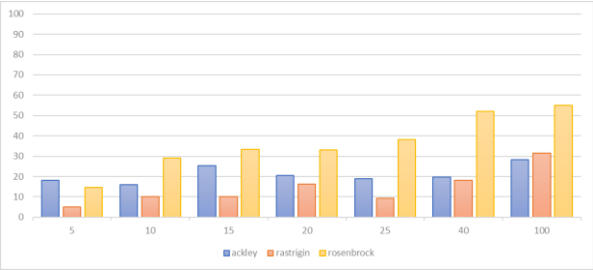

Figura 33. **Acurácia do RW em população no espaço de busca "G"**

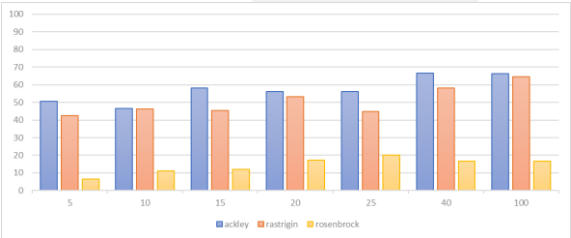

Figura 34. **Precisão do RW em população no espaço de busca "G"**

Tabela XVIII. **Desvio da acurácia em população**

ACURACIA		5	10	15	20	25	40	100
Ackley	P	0,0315	0,0227	0,0149	0,0125	0,0174	0,0090	**0,0079**
	M	0,0265	0,0179	0,0107	0,0203	0,0150	0,0123	**0,0090**
	G	0,0267	0,0242	0,0159	0,0236	0,0114	0,0094	**0,0095**
Rastrigin	P	0,0742	0,0967	0,0482	0,0361	0,0405	0,0354	**0,0122**
	M	0,0882	0,0768	0,0443	0,0374	0,0539	0,0262	**0,0248**
	G	0,0644	0,0406	0,0958	0,0355	0,0479	0,0251	**0,0190**
Rosenbrock	P	0,0244	0,0076	0,0124	0,0087	0,0123	0,0038	**0,0030**
	M	0,0316	0,0129	0,0105	0,0102	0,0131	0,0123	**0,0029**
	G	0	0,0204	0,0254	0,0112	0,0090	0,0065	**0,0047**

Além disto, a população maior, também aumenta o tempo médio de processamento, conforme mostrado na Figura 35. Isto é causado devido ao algoritmo, mesmo com a população em 100, não foi capaz de atingir o critério de parada do Erro Máximo Aceitável, apresentando um alto desvio padrão no tempo, mostrado na Tabela XIX.

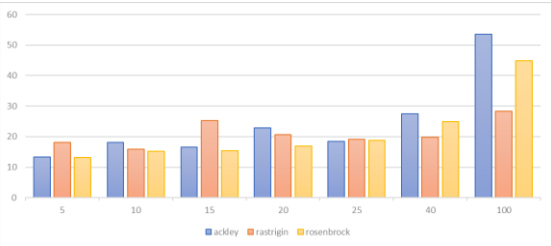

Figura 35. **Tempo do RW em população no espaço de busca "G"**

Tabela XIX. **Desvio do tempo em população**

TEMPO		5	10	15	20	25	40	100
Ackley	P	0,97	1,62	0,18	0,56	1,28	0,77	**5,90**
	M	0,42	1,26	0,67	0,32	0,86	0,54	**6,09**
	G	0,91	2,49	2,07	2,38	2,31	3,47	**10,54**
Rastrigin	P	0,16	0,39	1,08	3,32	1,09	3,79	**3,33**
	M	0,97	1,84	2,38	2,02	1,68	0,50	**2,68**
	G	1,38	1,45	1,69	1,96	1,09	1,42	**3,16**
Rosenbrock	P	3,55	0,78	1,67	1,14	1,82	1,92	**5,18**
	M	1,77	2,55	2,29	1,93	1,83	0,29	**3,32**
	G	1,03	1,58	0,20	0,16	0,10	0,18	**3,03**

F. Comparação dos Melhores Resultados

Através do processo de aferição, usando as funções de *benchmark*, é possível selecionar os melhores cenários entre cada algoritmo, e também, o melhor cenário entre todos os algoritmos analisados de maneira equalitária.

Em relação ao *Random Walk*, conforme mostrado no estudo de caso, a heurística não alcançou resultados tão alto como o FA e o PSO. Notasse uma dificuldade do RW em funções multimodais. Apesar disto, o algoritmo alcançou alguns resultados interessantes quando a população estava em 100. Os valores dos demais parâmetros neste cenário estão descritos na Tabela XX. Sendo o destaque, fica para a acurácia no espaço de busca "G" que ficou em torno de 70% na função Rosenbrock, e 30% nas funções Ackley e Rastrigin.

Tabela XX. **Melhor cenário do RW**

Parâmetros	Valores
população	100
passo	0,8
taxa de substuição	20%

No F*irefly Algorithm*, quando o tamanho de população estava em 100, o algoritmo conseguiu resultados de precisão e acurácia em torno de 100% em todas as cargas de trabalho, e com um baixo tempo de resposta. A configuração completa do cenário está exposta na Tabela XXI.

Tabela XXI. **Melhor cenário do FA**

Parâmetros	Valores
população	100
$\beta 0$	2,0
γ	0,4

No *Particle swarm optimization*, os melhores resultados foram possíveis quando o tamanho de enxame estava em 100. A configuração completa do PSO está disponível na Tabela XXII. Com resultado, foi possível alcançar acurácia e precisão em torno de 100% com um baixo tempo de execução.

Tabela XXII. Melhor cenário do PSO

Parâmetros	Valores
população	100
w	0,4
*C*1	2,0
C2	2,0

Como exposto, em contraponto a heurística RW, os algoritmos de inteligência de enxame FA e PSO obtiveram ótimos resultados, em acurácia e precisão, em seus melhores cenários. Com isto, a métrica de desempate do melhor algoritmo foi o quesito de tempo médio de processamento, mostrado na Tabela XXIV. Nota-se que o PSO obteve os menores tempos nas funções Multimodais Ackley e Rastrigin em todos os três intervalos de busca. Sendo que, na função Ackley a diferença média entre os dois algoritmos foi de 5 segundos, enquanto, na função Rastrigin foi de 3 segundos. Por sua vez, o FA apresentou melhor desempenho na função unimodal Rosenbrock com diferença média em 7 segundos.

Tabela XXIII. Tempo dos melhores cenários de algoritmo FA e PSO

Algoritmo	função	P	M	G
FA	*Ackey*	6,83s	6,38s	5,85s
	Rastringin	3,55s	3,48s	3,68s
	Rosenbrock	**15,25s***	**10,14s***	**10,72s***
PSO	***Ackey***	**1,357s***	**1,43s***	**1,39s***
	Rastringin	**0,84s***	**1,12s***	**0,86s***
	Rosenbrock	20,50s	19,40s	19,73s

* melhor resultado

VI. Conclusão

Neste trabalho foi realizado um estudo paramétrico utilizando as funções de *benchmarks* para avaliar os desempenhos de algoritmos de Inteligência de Enxame (IE). Neste sentido, foram analisados a Heurística *Random Walk*, e o algoritmo de IE *Firefly Algorithm* e *Particle swarm optimization*, utilizando diferentes espaços de busca em três funções de *benchmarks*. Os resultados apresentados durante o trabalho demostram que o processo de troca de informações feitos por algoritmos IE demostram-se bastante eficaz para a localização dos pontos ótimos em problemas de otimização, destacando-se o PSO como melhor algoritmo, apesar de em algumas situações, tenha sido ultrapassado pelo algoritmo FA. Além disto, também foi observado que a seleção incorreta de valores de parâmetros para estes algoritmos compromete profundamente a sua eficácia. Deste modo, conclui-se que este trabalho alcançou os objetivos estabelecidos, em apresentar o comportamento dos algoritmos com base nas alterações de valores de parâmetros e no tamanho do intervalo de busca, sendo este conhecimento fundamental para desenvolvimentos de futuras pesquisas usando os algoritmos apresentados. Em trabalhos futuros pretendesse explorar:

- Comparar mais algoritmos de inteligência de enxame como *Ant colony optimization algorithms, Bat Algorithm e Bacterial Foraging Optimization Algorithm;*
- Comparar o desempenho de algoritmos de inteligência de enxame com outras metaheurística baseadas em Algoritmos Evolucionários;

VII. Referências

[1] T. Boechel, Algoritmo de Otimização: uma Abordagem Híbrida Utilizando o Algoritmo das Formigas e Genético, Florianópolis: UFSC, 2003.

[2] J. F. Vidal , Identificação de Sistemas Dinâmicos Lineares Utilizando Inteligência de Enxame: Uma Abordagem Via Miríades de Vagalumes Artificiais, Castanhal: UFPA, 2020.

[3] J. Nocedal e S. J. Wright, Numerical Optimization, 2 ed., New York: Springer, 2006.

[4] S. S. Alves, METAHIS: Instrumento Educacional Complementar no Ensinoaprendizagem de Metaheurísticas Populacionais Aplicadas à Engenharia de Controle, Belém: UFPA, 2018.

[5] C. E. M. Barbosa, Algoritmos Bio-Inspirados para Solução de Problemas de Otimização, Recife: UFPE, 2017.

[6] G. C. d. C. Holtz, Traçado automático de envoltórias de esforços em, Rio de Janeiro: PUC-Rio, 2005.

[7] F. E. Bisogno e J. W. S. Prado , Projeto do Conversor Classe-E Normalizado com Aplicação de PSO, Santa Maria: SEPOC , 2021.

[8] A. M. Maia, Aplicação de Métodos de Otimização Linear e PSO na Coordenação de relés de Sobrecorrente, Niterói: UFF, 2022.

[9] B. E. E. d. Lima, Uma Proposta para Aplicação do Método PSO em Programas de Gerenciamento de Carga pelo Lado da Demanda, CORNÉLIO PROCÓPIO: UTFPR, 2020.

[10] L. A. Gigo e . C. A. d. Castro Jr, Estudo do Firefly Algorithm e aplicações em sistemas elétricos de, Campinas: Unicamp, 2020.

[11] E. F. P. d. Luz, H. F. d. C. Velho e J. C. Becceneri, algoritmo paralelo de vaga-lumes com predação na identificação de condição inicial na equação do calor, Rio de Janeiro: SPOLM, 2019 .

[12] M. P. Bez, algoritmo paralelo de vaga-lumes com predação na identificação de condição inicial na equação do calor, Porte Alegre: UFRGS, 2011.

[13] A. Jhawar, "Random Walk: Will the Drunk Man Fall Off the Cliff?," Medium, 4 jul 2020. [Online]. Available: https://medium.com/swlh/random-walk-a-comprehensive-illustration-aa13373830d1. [Acesso em 11 jun 2022].

[14] A. R. C. Gonçalves, Problema de Perseguição-Evasão Baseado em Random Walk, São Cristóvão: UFS, 2016.

[15] MDig, "Por que os peixes formam cardumes?," MDig, 06 out 2021. [Online]. Available: https://www.mdig.com.br/index.php?itemid=52936. [Acesso em 10 ago 2022].

[16] A. B. d. S. Serapião, "Fundamentos de Otimização por Inteligência de Enxames: Uma Visão Geral," *Revista Controle & Automação*, pp. 271-304, jul 2009.

[17] J. D. Elisiário, Análise Comportamental do Algoritmo Otimização por Enxame de Partículas em Espaços N-Dimensionais, Bambuí: IFMG, 2018.

[18] X.-S. Yang, Introduction to Mathematical Optimization, 1ª ed., Cambridge: CISP, 2008.

[19] J. V. d. C. Mesquita, Análise Paramétrica do Algoritmo Particle Swarm Optimization com Aplicação em Sistemas De Controle, Belém: UFPA, 2018.

[20] B. Xing e W.-J. Gao, Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms, New York: Springer, 2014.

[21] F. A. P. Paiva, . I. V. O. Leite e M. H. F. Marcone, "Um Breve Estudo Sobre os Algoritmos do Vaga-Lume e do Morcego para Otimização de Funções de Referência," *HOLOS*, pp. 190-206, 4 mai 2018.

[22] T. d. M. e. S. M. Johnson e M. Margalho , Avaliação de Desempenho de Sistemas Computacionais, Rio de Janeiro: LTC, 2011.

[23] R. Jain, Art of Computer Systems Performance Analysis Techniques For Experimental Design Measurements Simulation And Modeling, New York: John Wiley & Sons, 1991.

[24] TechTudo, "O que é benchmark?," TechTudo, 28 jun 2016 . [Online]. Available: https://www.techtudo.com.br/noticias/2016/06/o-que-e-benchmark-entenda-como-funciona-o-teste-de-desempenho-do-celular.ghtml. [Acesso em 25 mai 2022].

[25] S. Surjanovic e D. Bingham, "Virtual Library of Simulation Experiments," Simon Fraser University, 2013. [Online]. Available: https://www.sfu.ca/~ssurjano/optimization.html. [Acesso em 25 jul 2022].