

Estudo Comparativo Sobre a Convergência e o Custo Computacional das Estruturas Topológicas Aplicadas à Otimização por Enxame de Partículas (PSO)

Alisson Patrick Medeiros de Lima^{*}

Ygor Alcântara de Medeiros^{**}

Lucas Vale Faustino da Silva^{***}

Wellington Candeia de Araujo^{****}

Resumo

O PSO é uma meta-heurística que foi criada para ser aplicada à uma diversidade de funções de otimização. Diante disso, o comportamento de qualquer meta-heurística é facilmente guiado por duas vertentes: os operadores de variação e os valores selecionados para os parâmetros do PSO. Portanto, não só os parâmetros da meta-heurística, mas também a topologia de vizinhança desempenha um papel fundamental no comportamento do PSO. Nesta perspectiva, este trabalho tem o objetivo de analisar o desempenho das principais topologias existentes que podem ser aplicadas ao PSO, investigando ao mesmo tempo, a convergência e o custo computacional de cada topologia. Com o propósito de realizar um amplo estudo, foram selecionadas 6 diferentes topologias: Global, Ring, Von Neumann, Random, Mesh, Star.

Palavras Chaves: PSO. Topologias de vizinhança. Meta-heurística.

* Graduando em Bacharelado em Ciência da Computação pela Universidade Estadual da Paraíba - Patos.
E-mail: alissonp.medeiros@gmail.com.

** Graduando em Bacharelado em Ciência da Computação pela Universidade Estadual da Paraíba - Patos.
E-mail: ygoralcantara5@gmail.com.

*** Graduando em Bacharelado em Ciência da Computação pela Universidade Estadual da Paraíba - Patos.
E-mail: valelucas1@gmail.com.

**** Professor doutor na Universidade Estadual da Paraíba - Patos. E-mail: wcandeia@uepb.edu.br

1 Introdução

Na ciência da computação, o PSO é um método computacional de pesquisa e otimização biologicamente inspirado nos comportamentos sociais dos pássaros, cardumes de peixes ou enxames de insetos.

Aplicações reais que usam técnicas de otimização requerem um resultado ótimo ou ao menos próximo disso. O PSO é uma proposta recente de meta-heurística para otimização em espaços vetoriais. O PSO funciona com um enxame de agentes simples com baixa capacidade de processamento que coletivamente buscam um determinado objetivo, assim simulando o comportamento de enxames naturais em busca de alimento.

Métodos clássicos de aprimoramento de sistemas tendem a aumentar sua complexidade na proporção em que a dificuldade do problema aumenta. Neste cenário, o tempo e o custo de processamento desses métodos impossibilita sua aplicação em muitos problemas práticos. A maioria dos problemas reais não necessitam obrigatoriamente de uma solução ótima, soluções próximos da ótima já são suficientes, desde que dentro de uma margem aceitável de erro, variando conforme o problema que se quer otimizar.

Este trabalho é organizado em 5 seções. Na segunda será abordado sobre o PSO, na terceira sobre os experimentos realizados, na quarta os resultados e por último as conclusões e trabalhos futuros.

2 Otimização por Enxame de Partículas

O PSO é um algoritmo de inteligência de enxames inspirado na natureza. A inteligência de enxames é uma técnica de solução para problemas que se baseiam em interações de indivíduos simples, com baixa capacidade de processamento, que interagem entre si e com o ambiente em que convivem através de regras simples. Essas populações de indivíduos formam sistemas concentrados e auto organizáveis.

O PSO foi sugerido em 1995 por Kennedy e Eberhart (KENNEDY, EBERHART, 1995), que simula a conduta de bandos de pássaros e cardume de peixes. Kennedy e Eberhart foram inspirados pelo trabalho de Heppner e Granander (HEPPNER, GRANANDER, 1990).

No PSO, o enxame de partículas voa pelo espaço de busca e cada partícula é influenciada pela sua melhor posição encontrada até o momento e pela melhor posição encontrada pelo bando. Neste contexto, cada partícula equivale uma solução plausível para um determinado problema que está sendo otimizado.

Na inicialização do algoritmo, as partículas são aleatoriamente distribuídas, ou seja, seus vetores velocidade e posição são definidos no espaço de busca. Cada partícula i é representada por três vetores (BRATTON, KENNEDY, 2007):

a- A sua posição em um espaço de busca D-dimensional: $\vec{p}_i = (p_{i1}, p_{i2}, \dots, p_{iD})$;

b- A sua melhor posição encontrada até o momento: $\vec{pb}_i = (pb_{i1}, pb_{i2}, \dots, pb_{iD})$;

c- A sua velocidade atual: $\vec{v}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$.

Neste sentido as partículas movimentam-se no espaço de busca procurando a melhor solução possível e a cada iteração o algoritmo atualiza as velocidades e posições das partículas usando as equações (1) e (2):

$$v_{id} = v_{id} + c_1 r_1 (pb_{id} - p_{id}) + c_2 r_2 (bs - p_{id}) \quad (1)$$

$$p_{id} + 1 = p_{id} + v_{id} \quad (2)$$

O algoritmo PSO clássico define que deve existir um mecanismo de condição de parada, este mecanismo pode ser determinado pelo número de iterações que o algoritmo deve executar ou até que o enxame consiga achar uma solução a qual ele não consiga melhorar diante disto. Na Figura tem-se o algoritmo do PSO.

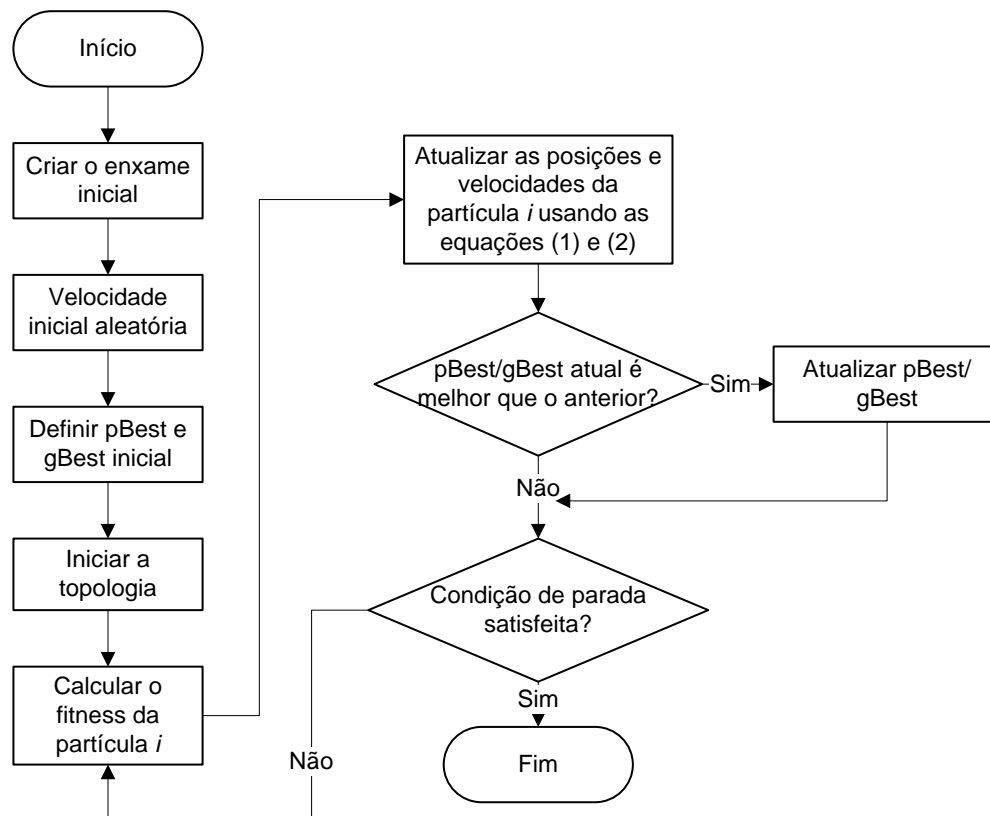


Figura 1 - Fluxograma do PSO.
Fonte: Elaborado pelos autores.

2.1 Topologias de vizinhança

No PSO, cada partícula que faz parte do enxame se comunica de forma específica com as outras partículas. Esse mecanismo de comunicação é denominado topologia de vizinhança. Diante disso, foram realizados diversos estudos para determinar se uma topologia de vizinhança influencia na convergência do PSO (KENNEDY, 1999; KENNEDY, MENDES, 2002).

Esses estudos propuseram implementações das topologias global e ring, que são hoje as mais utilizadas no PSO. Com isso ficou constatado que algumas topologias obtêm melhores resultados que outras topologias quando aplicadas em diferentes funções de otimização (KENNEDY, 1999).

Neste cenário, as topologias são mecanismos de extrema importância que definem como as partículas se comunicam, influenciando diretamente o resultado da execução do algoritmo.

3 Especificação do experimento

Esta seção aborda detalhadamente sobre as topologias utilizadas na realização do experimento. Em adição, serão explicadas sobre as funções de testes que foram utilizadas para testar o desempenho do PSO utilizando cada topologia, assim como as configurações necessárias para o funcionamento do PSO.

3.1 Topologias Ring (Local) e Global

As topologias local e global são as topologias mais conhecidas e utilizadas (KENNEDY, EBERHART, 1995). A topologia global foi a primeira a ser proposta. Nesta topologia cada partícula está conectada com todas as outras do enxame, logo uma partícula é influenciada por todo o enxame, pois esta está recebendo informações de todo o enxame. Esta topologia apresenta grandes vantagens quando utilizadas em problemas com funções unimodais, pois as partículas encontram rapidamente uma solução aceitável.

A topologia Ring (ou local) é considerada uma das melhores abordagens com maior importância do algoritmo PSO clássico. Nesta topologia as partículas estão organizadas em forma de anel, assim cada partícula se comunica diretamente com dois vizinhos. Na Figura 2 têm-se as topologias Local e Global.

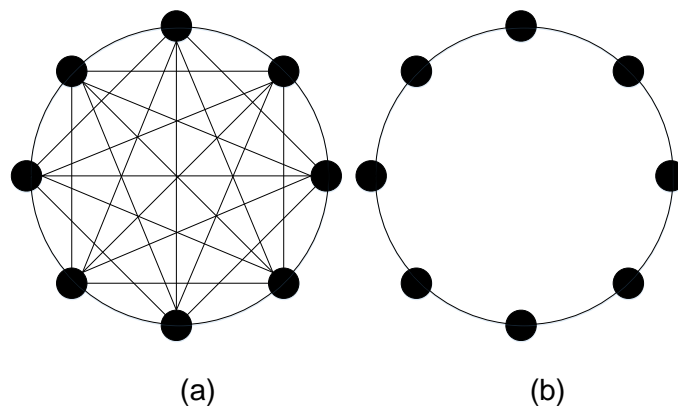


Figura 2 – (a) Topologia Global e (b) Topologia Local.
Fonte: Elaborado pelos autores.

3.2 Topologia Random

A Topologia Random determina que cada partícula está associada diretamente a uma quantidade randômica de outras partículas. Diante disso, esta topologia apresenta grandes variações em seus resultados, ocorrendo que muitas vezes ela não é comumente utilizada devido ao fato do seu comportamento ser aleatório. A figura 3 representa a topologia Random.

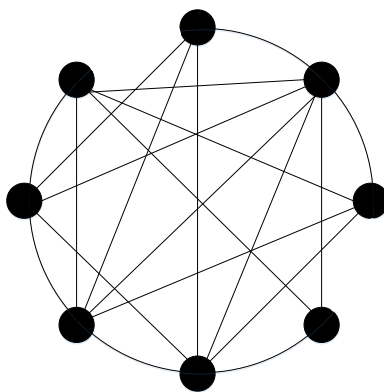


Figura 3 – Topologia Random.
Fonte: Elaborado pelos autores.

3.3 Topologia Star

Na topologia Star, a informação passa apenas por uma partícula central. Nesta partícula é centralizada as informações trocadas em todo o enxame. Com isso, esta partícula central influencia todos os membros do enxame.

Diante disso, a topologia Star separa todos os membros do enxame, fazendo com que a comunicação entre eles sempre passe pelo nó central. Para a análise da topologia Star, a partícula central será escolhida aleatoriamente. Na Figura 4 tem-se a representação gráfica da topologia Star.

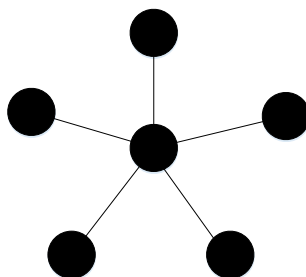


Figura 4 – Topologia Star.
Fonte: Elaborado pelos autores.

3.4 Topologia Mesh

Na topologia Mesh, tem-se a representação do enxame em forma de uma matriz, onde na maioria dos casos uma partícula comunica-se diretamente com 4 outras partículas, porém existem algumas exceções para as partículas localizadas nos cantos da matriz ou nos limites da matriz que respectivamente comunicam-se com outras três partículas.

Para as partículas que não estão nos cantos e nem nos limites na matriz, ou seja, estão nas posições centrais da matriz, estas partículas comunicam-se diretamente com as partículas localizadas nas direções: norte, sul, leste e oeste.

Para as demais partículas, as direções das partículas vizinhas dependem diretamente da sua localização. Diante disso, tem-se a topologia Mesh representada graficamente na Figura 5 a seguir.

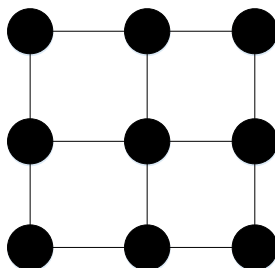


Figura 5 – Topologia Mesh.
Fonte: Elaborado pelos autores.

3.5 Topologia Von Neumann

Na topologia Von Neumann é considerada uma superfície fechada à qual é definida como o produto de dois círculos. Essa definição pode ser constatada na figura 6. Topologicamente esta estrutura é muito parecida com a topologia Mesh, com exceção de que todas as partículas do enxame se comunicam respectivamente com 4 vizinhos.

Como mostrado na figura 6, na topologia Von Neumann interliga todas as partículas localizadas nos cantos da matriz. O mesmo ocorre para as partículas localizadas nos limites da matriz. O restante do enxame é interligado seguindo a definição da topologia Mesh.

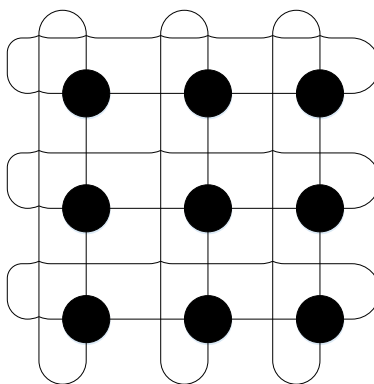


Figura 6 – Topologia Von Neumann.
Fonte: Elaborado pelos autores.

3.6 Funções de Teste e Configuração do PSO

Esta subseção explica sobre as funções de teste e configurações utilizadas para os experimentos das topologias no PSO. As funções de teste padrão, com alta complexidade, que são utilizadas para realizar a validação do PSO são descritas a seguir.

O algoritmo PSO foi usado para resolver seis funções benchmark, as quais são definidas na Tabela I. Essas funções consistem em multimodais e unimodais. As multimodais são Ackley, Griewank, Michalewicz, Rastrigin. As unimodais são Schwefel 1.2 e Sphere (SURJANOVIC, BINGHAM, 2013).

Para a realização dos testes, foi utilizada uma população de 100 partículas para todas as topologias. Ademais, para as funções unimodais foram necessárias 1000 iterações e para as demais funções multimodais foram realizadas 5000 mil iterações, com exceção da função Griewank à qual precisou utilizar apenas 2000 iterações. Na tabela 2 tem-se todas as configurações utilizadas nas funções de teste.

Função	Fórmula
Ackley $f(x_1)$	$\left(-b \sqrt{\frac{d}{1} \sum_{i=1}^d x_i^2}\right) - \exp\left(\frac{1}{d} \sum_{i=1}^d \cos(cx_i)\right) + a + \exp(1)$
Griewank $f(x_2)$	$\sum_{i=1}^d \frac{x_i^2}{4000} - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
Michalewicz $f(x_3)$	$-\sum_{i=1}^d \sin(x_i) \sin^{2m}\left(\frac{ix_i^2}{\pi}\right)$
Rastrigin $f(x_4)$	$10d + \sum_{i=1}^d [x_i^2 - 10\cos(2\pi x_i)]$
Schwefel 1.2 $f(x_5)$	$\sum_{i=1}^d \left(\sum_{j=1}^i x_j\right)^2$
Sphere $f(x_6)$	$\sum_{i=1}^d x_i^2$

Tabela 1 – Definições matemáticas sobre as funções de teste.

Função	Dimensão	Solução ótima	Espaço de busca	Espaço de busca utilizado
Ackley	30	0	-32, 32	16, 32
Griewank	30	0	-600, 600	16, 32
Michalewicz	10	-9.660	(0, π)	2.56, 3.12
Rastrigin	30	0	-5.12, 5.12	2.56, 5.12
Schwefel 1.2	30	0	-100, 100	-2, 5
Sphere	30	0	-5.12, 5.12	-5, 5

Tabela 2 – Configuração das funções de teste utilizadas no experimento.

4 Resultados das simulações

Nesta seção, é investigada a performance de diferentes topologias utilizadas no algoritmo PSO original. Para isso, utilizou-se diversas funções de teste benchmark. Essas funções são responsáveis por testar a otimização atual do comportamento de determinados algoritmos. As funções utilizadas nos experimentos são caracterizadas pela alta complexidade associada à não linearidade e múltiplos mínimos locais.

Posteriormente à tabela abaixo, será discutido sobre o resultado da convergência e do custo computacional de cada topologia ao mesmo tempo que será analisado a viabilidade de utilização de cada topologia. Tal viabilidade será feita de acordo com a velocidade de convergência e custo computacional esboçado nos gráficos de cada topologia. Na tabela 2 tem-se os resultados de convergência de todas as topologias.

Função	Mesh		Star		Global		Ring		Von Neumann		Random	
	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio
$f(x1)$	12.110	0.7473	12.666	1.1854	12.424	1.3824	12.132	0.574	10.7271	0.4901	10.806	0.3477
$f(x2)$	0.0209	0.0211	0.0217	0.0216	0.0087	0.0118	0.0029	0.0041	0.0018	0.0034	0.0283	0.0806
$f(x3)$	-9.5991	0.2163	-9.4679	0.1955	-9.5882	0.1660	-9.2729	0.2617	-9.4803	0.2309	-9.0236	0.0368
$f(x4)$	68.452	20.416	54.324	18.567	61.886	21.091	75.117	13.088	37.547	13.508	47.560	16.898
$f(x5)$	7.0E-5	9.0E-5	9.0E-5	1.2E-4	1.2E-4	1.6E-4	0.3210	0.1725	0.0280	0.0211	0.0017	0.0063
$f(x6)$	2.0E-5	2.0E-5	2.0E-5	1.0E-5	3.0E-5	7.0E-5	0.0554	0.0268	0.0044	0.0034	6.0E-5	9.0E-5

Tabela 3 – Desempenho das topologias testadas nas funções benchmark

A tabela 3 serve como base para uma observação mais precisa e acurada sobre os gráficos que serão mostrados a seguir. Cada gráfico demonstra diretamente o comportamento de cada topologia abordada neste trabalho. Para uma melhor certeza sobre os dados apurados, a quantidade de execuções para cada topologia foi definida em 100.1

Como pode-se observar na tabela 3, durante os testes realizados com a função Ackley, a topologia Von Neumann conseguiu alcançar resultados bastantes significativos com relações as outras topologias, conseguindo chegar em $f(x1) = 10,727$. A única topologia à qual conseguiu alcançar resultados próximos da topologia Von Neumann foi a topologia Random. Na figura a seguir, temos a representação gráfica da convergência de todas as topologias testadas na função Ackley.

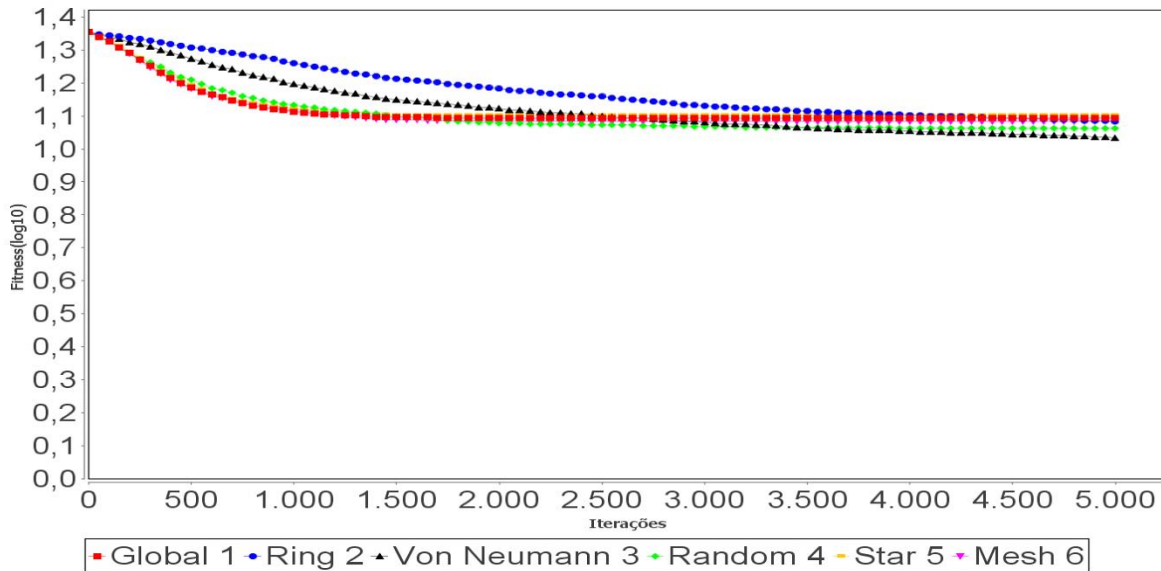


Figura 7 – Gráfico de convergência da função Ackley.

Ainda de acordo com a tabela 3 e com o gráfico da figura 8 a seguir, a topologia Von Neumann destaca-se com relação ao tempo computacional gasto para executar o PSO, ao qual gastou 2,24 segundos. A topologia Von Neumann está 10 décimos de segundos mais lenta do que a topologia Ring. A topologia Von Neumann consegue se sobressair tanto no quesito convergência quanto tempo computacional diante das demais topologias. Na figura 8 podemos analisar o custo computacional que todas as topologias atingiram.

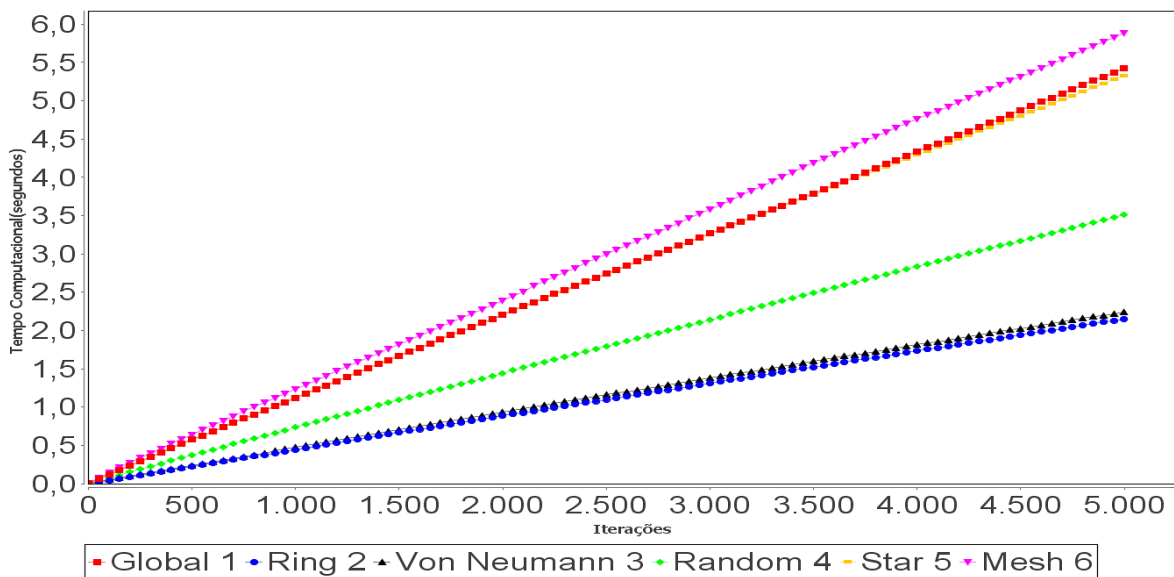


Figura 8 – Gráfico do custo computacional da função Ackley.

Em adição, observa-se na figura 9 os resultados de convergência das topologias utilizando a função Griewank. A topologia Von Neumann mais uma vez consegue alcançar resultados totalmente melhores que as demais topologias, deixando claro que esta topologia pode ser usada perfeitamente em uma diversidade de problemas que utilizem o PSO. A partir da iteração 1.250 a topologia Von Neumann já consegue resultados melhores que as demais topologias, conseguindo chegar em $f(x_2) = 0,0018$.

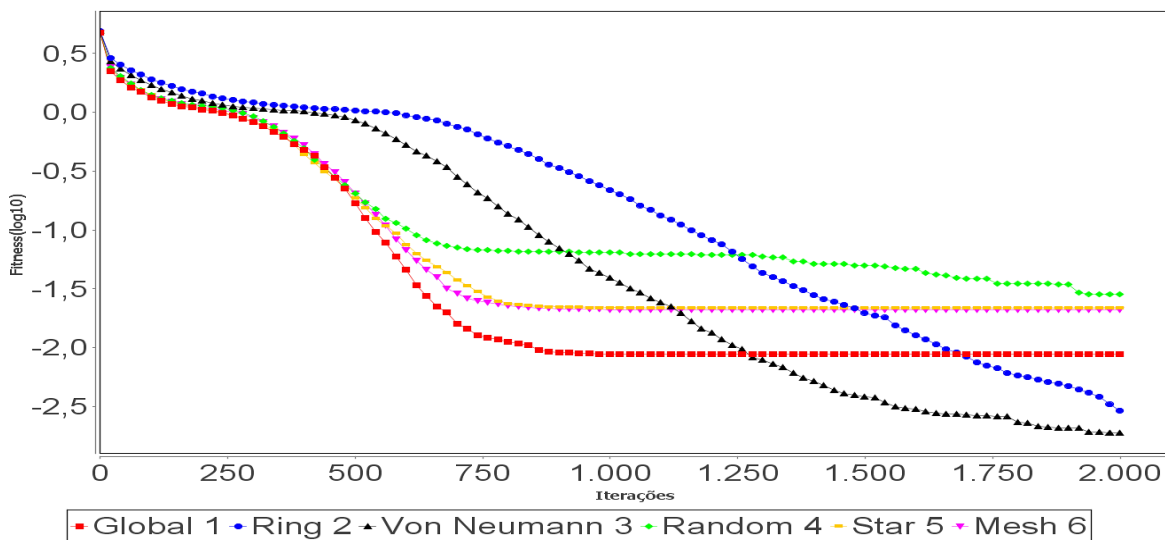


Figura 9 – Gráfico de convergência da função Griewank

Nos testes realizados na função Griewank, as topologias Ring e Von Neumann se saem melhores no item custo computacional. Podemos analisar de acordo com a figura 10, que essas duas topologias conseguem ser utilizadas sem requererem tanto tempo computacional e ambas gastaram quase 1 segundo para executarem as 2.000 iterações. Na figura a seguir, tem-se o gráfico que representa o custo computacional de todas as topologias utilizadas na função Griewank.

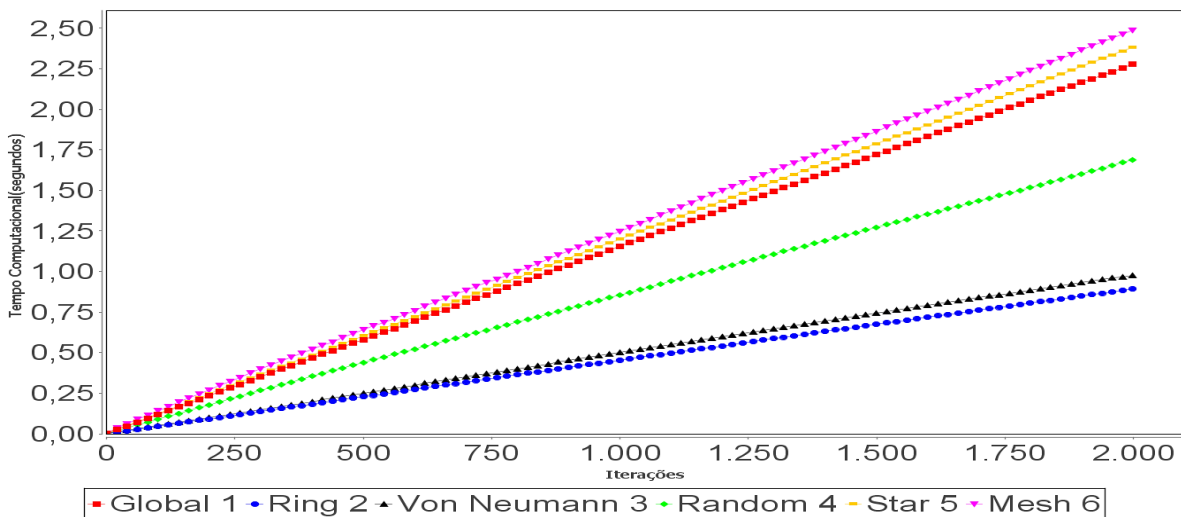


Figura 10 – Gráfico do custo computacional da função Griewank.

A função Michalewicz é uma função bastante complexa devido possuir muitos mínimos locais. Nos testes de convergência realizados com esta função, todas as topologias obtiveram resultados muito próximos, chegando muito perto do ótimo global desta função que é -9,660 com 10 dimensões, obtendo assim $f(x) = -9,5991$. De acordo com a figura 11, a topologia Mesh obtém resultados satisfatórios, conseguindo chegar num resultado muito aproximado ao esperado. A topologia Global ficou em 2º lugar no item convergência.

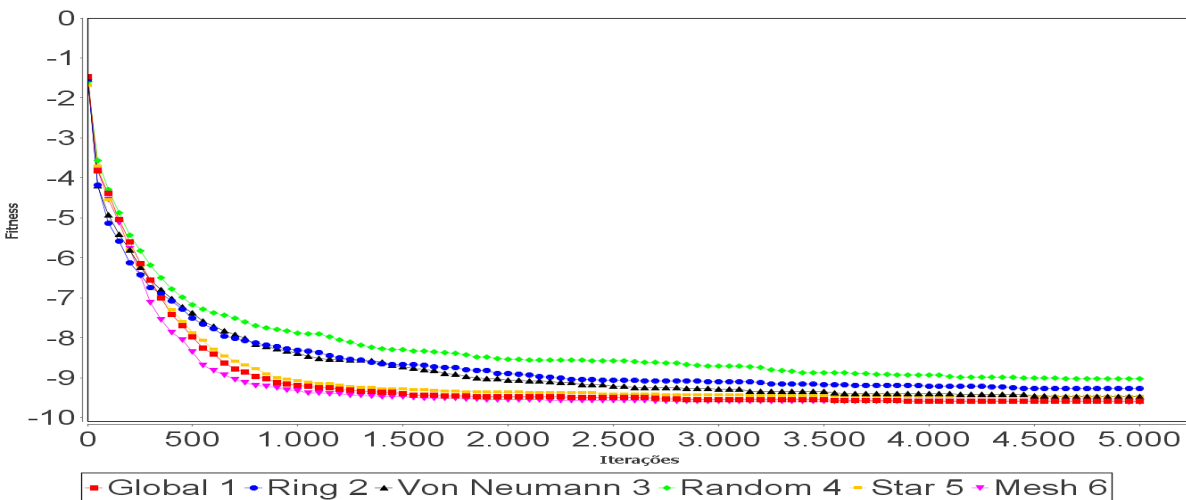


Figura 11 – Gráfico de convergência da função Michalewicz.

A grande diferença entre os resultados obtidos pelas topologias Mesh e Global está no custo computacional. Ambas as topologias obtiveram ótimos resultados, porém o custo computacional da topologia Mesh é consideravelmente mais elevado que a topologia Global. Na figura 12 temos o custo computacional de cada topologia.

A Topologia Mesh é cerca de 30 décimos de segundos mais lenta que a topologia Global, porém outra questão importante é que as outras topologias se sobressaem do que diz respeito ao custo computacional. Enquanto a topologia Mesh obtém -9.59918 em relação a convergência para um resultado ótimo, a topologia Ring consegue obter -9.27293 e a diferença entre elas duas se dá principalmente no custo computacional para serem executadas as 5.000 iterações.

Ao mesmo tempo em que a topologia Mesh gasta cerca de 3,27 segundos, a topologia Ring gasta apenas 1,91 segundos, ou seja, quase metade do tempo de execução da topologia Mesh. Isso indica que deve ter uma análise simultânea sobre a velocidade de convergência e o tempo computacional de uma determinada topologia antes de usá-la.

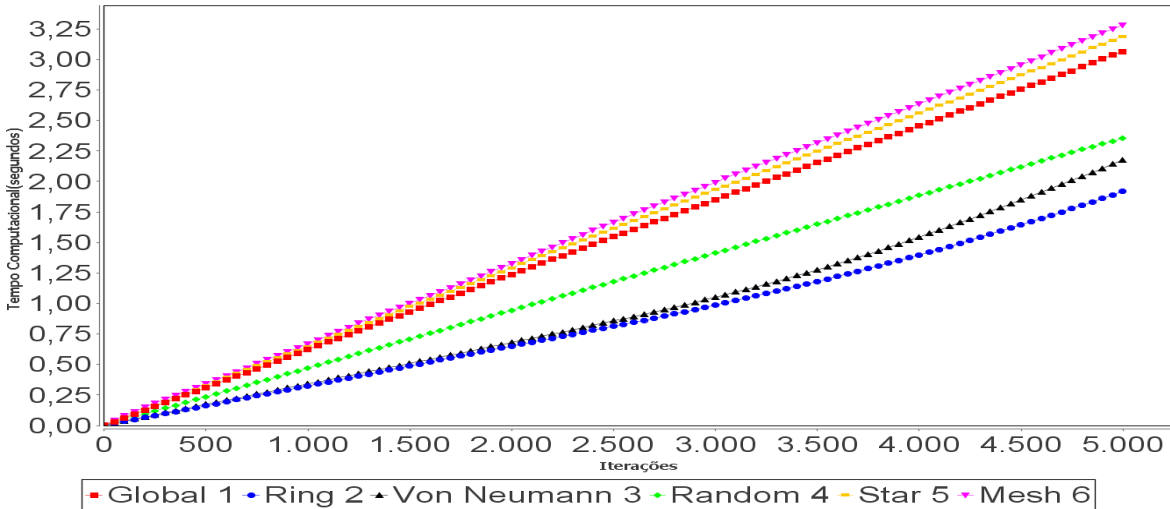


Figura 12 – Gráfico do custo computacional da função Michalewicz.

A função Rastrigin é altamente multimodal. Possui diversos mínimos locais, mas os locais dos mínimos são distribuídos regularmente. Nos testes realizados utilizando esta função, podemos observar que existe um desvio padrão muito elevado. Mesmo sendo uma função com resultados muito instáveis, a topologia Von Neumann mais uma vez tem uma velocidade de convergência maior que as demais topologias.

Com esta função de teste, a topologia Von Neumann conseguiu atingir o mínimo local de $f(x_4) = 37,5$, enquanto a segunda topologia Random que obteve resultados mais próximos, alcançou um ótimo local de $f(x_4) = 47,7$. Esta diferença pode ser notada analisando a figura 13.

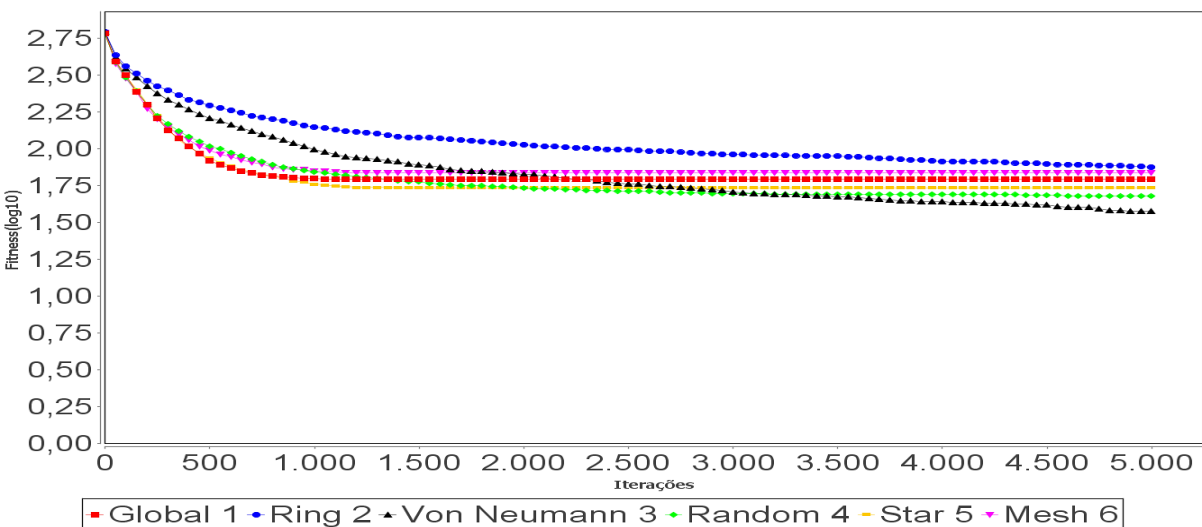


Figura 13 – Gráfico de convergência da função Rastrigin.

Outro aspecto relevante a ser observado é no que diz respeito ao custo computacional. Neste aspecto observado, as topologias Ring e Von Neumann conseguiram executar as 5.000 iterações em uma quantidade de tempo bem razoável do que outras topologias. A topologia Ring conseguiu executar as 5.000 iterações em apenas 2.12 segundos, enquanto que a topologia Mesh que obteve o pior resultado, conseguiu executar as 5.000 iterações em 6,16 segundos. Podemos constatar essas observações no gráfico da figura a seguir.

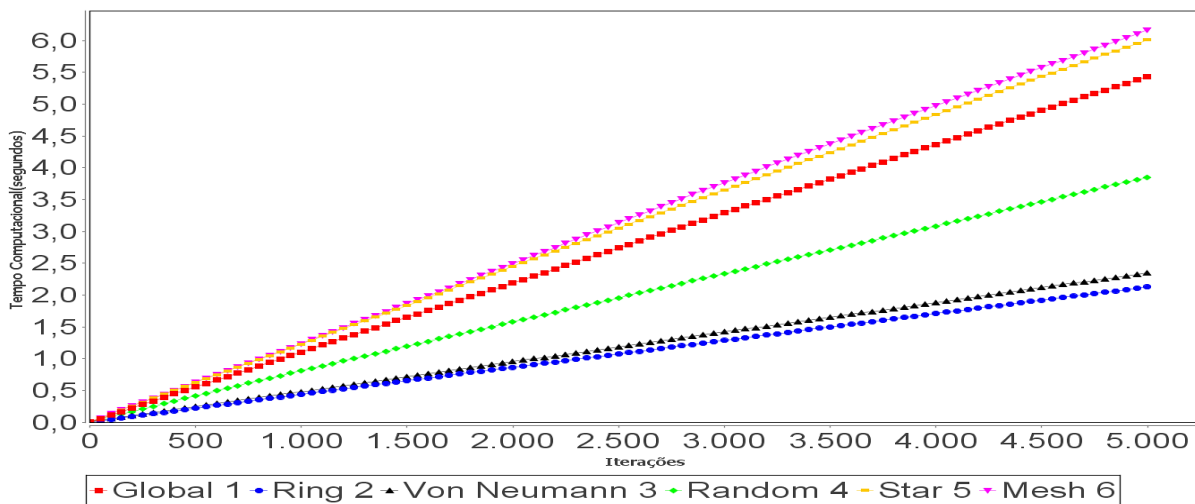


Figura 14 – Gráfico do custo computacional da função Rastrigin.

A função Schwefel 1.2 é conhecida pelo seu alto grau de complexidade e pelos seus diversos mínimos locais. Seu mínimo global encontra-se em $f(x) = 0$. Diante dos testes realizados, todas as topologias conseguiram alcançar resultados bons. Nos testes envolvendo esta função, as topologias que se destacam é a Mesh com $f(x_5) = 7,0E-5$ e a topologia Star com $f(x_5) = 9,0E-5$. Ambas conseguem atingir mesmos resultados, porém com custo computacional diferentes. Na figura 15, pode-se observar que tanto a topologia Mesh quanto a Star conseguem atingir o ótimo global.

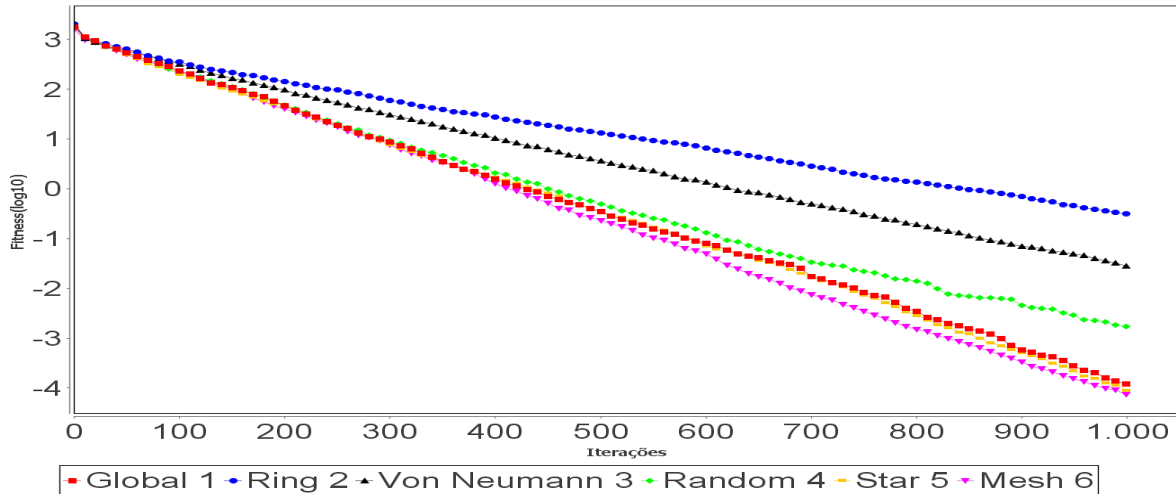


Figura 15 – Gráfico de convergência da função Schwefel 1.2.

No quesito custo computacional, todas as topologias também obtiveram resultados parecidos, não ocorrendo tanta diferença entre a topologia Von Neumann que obteve o melhor resultado e a topologia Mesh que obteve o pior resultado. Neste caso em específico, nota-se que não ouve a diferença de sequer 1 segundo. Na figura 16 temos a representação gráfica do custo computacional de todas as topologias.

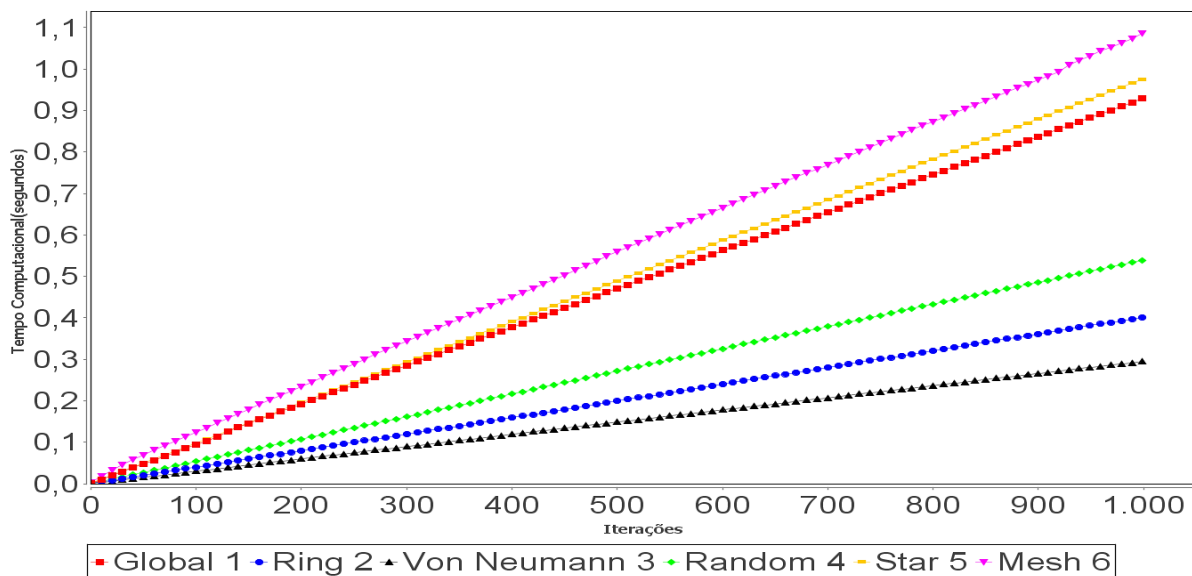


Figura 16 – Gráfico do custo computacional da função Schwefel 1.2.

A função Sphere também é uma função unimodal que possui vários mínimos locais. Esta função é contínua e convexa. Seu mínimo global também está em $f(x) = 0$. Foram utilizadas apenas 1.000 iterações na função Schwefel 1.2 e Sphere porque estas funções alcançam seu mínimo global muito rapidamente, ou seja, em

funções unimodais tem-se uma velocidade de convergência mais rápida que as multimodais.

Neste sentido, os testes realizados com a função Sphere mostram que a maioria das topologias conseguem alcançar o resultado esperado em poucas iterações, em contrapartida, as topologias Ring e Von Neumann não obtêm resultados concisos. Diante disso, a topologia Star com $f(x6) = 2,0E-5$ e Mesh também com $f(x6) = 2,0E-5$ alcançaram mesmos resultados, diferenciado apenas no desvio padrão que é baixíssimo e no custo computacional. A topologia Global também conseguiu atingir um resultado muito próximo do esperado. Na figura 17 podemos analisar a convergência de cada topologia utilizando a função Sphere.

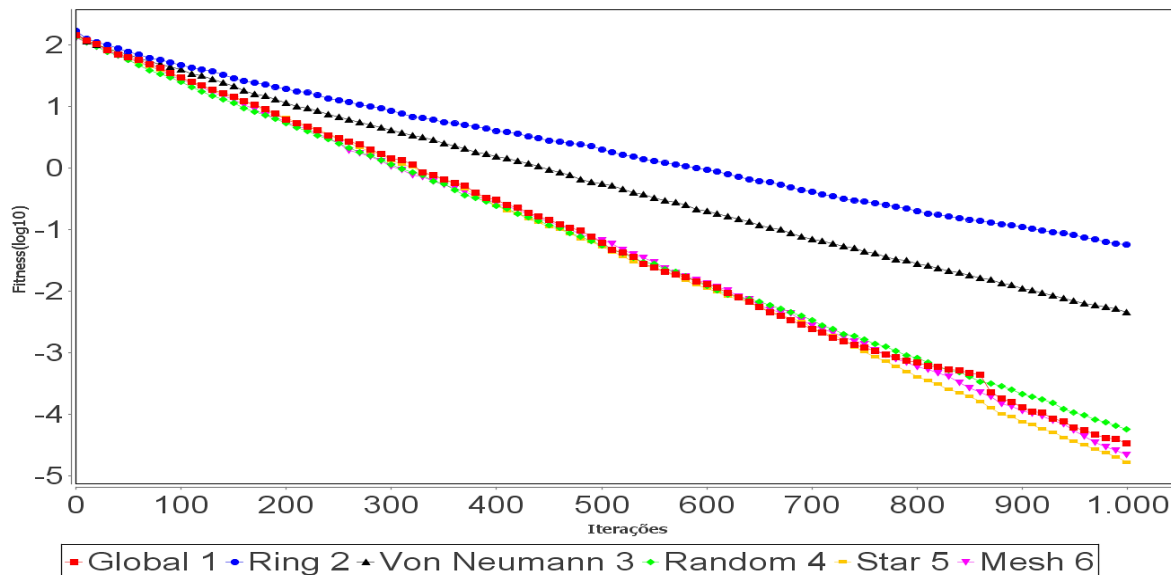


Figura 17 – Gráfico de convergência da função Sphere.

Durante os testes de custo computacional, ficou comprovado que todas as topologias conseguem executar as 1.000 iterações em um período de tempo bastante curto, tais períodos variam dentre 0,2 e 1 segundo. A topologia Global foi a que mais demorou para executar, gastando assim 0,97 segundos. Já a topologia que obteve melhores resultados foi a Von Neumann, gastando 0,25 segundos para executar as 1.000 iterações. Os resultados podem melhor serem observados na figura 18, que representa o custo computacional de cada topologia utilizando a função Sphere.

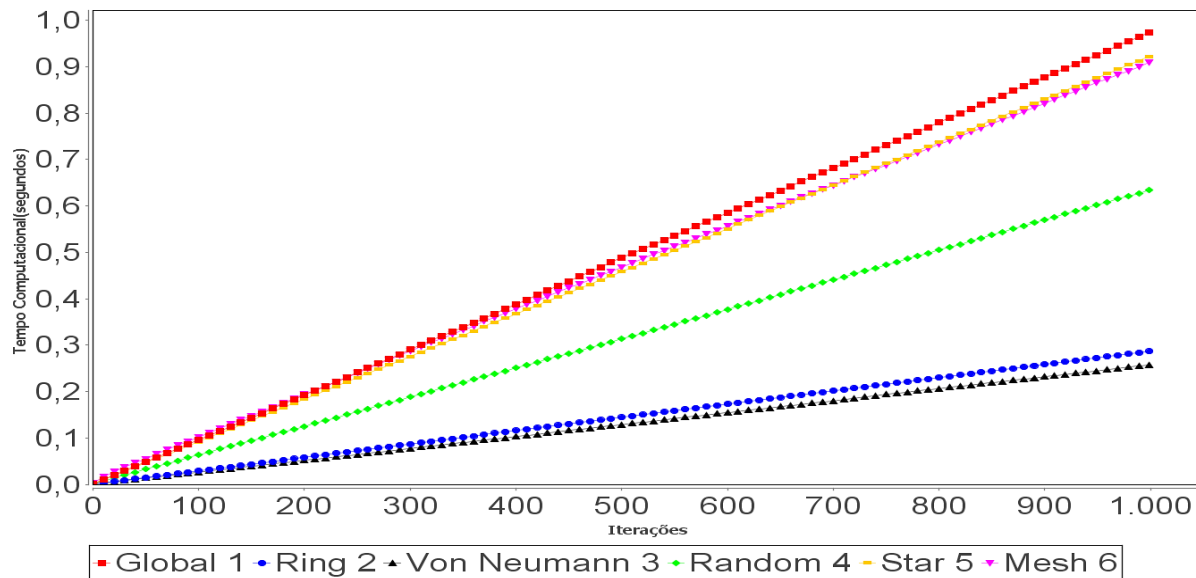


Figura 18 – Gráfico do custo computacional da função Sphere.

Através de todos os testes realizados neste trabalho, a topologia Von Neumann provou ser melhor para todas as funções multimodais utilizadas neste trabalho, com exceção da função Michalewicz, tanto no requisito velocidade de convergência, quanto no custo computacional. Esta topologia em específico, consegue ao mesmo tempo alcançar uma alta velocidade de convergência sem deixar a desejar no custo computacional exigido para tal operação.

Outro resultado que chamou bastante atenção, foram os resultados da topologia Mesh, que nas funções multimodal Michalewicz e unimodal Sphere, conseguiu uma velocidade de convergência significativa se comparada as outras topologias, porém, um fato que deve ser exposto, é que esta topologia em particular, necessita de um custo computacional bastante elevado, quando comparado as outras topologias utilizadas neste trabalho.

Nos resultados da função unimodal Sphere, a topologia Star conseguiu uma velocidade de convergência maior que as demais topologias, contudo, pode-se observar que tanto a topologia Star, Mesh e Global têm um custo computacional elevado, e essa característica deve ser examinada cuidadosamente antes da utilização dessas topologias, pois dependendo do problema que se está sendo abordado, estão topologias podem deixar a desejar no quesito custo computacional.

5 Conclusões

Em virtude dos resultados mencionados, constatamos que as topologias apresentadas neste trabalho podem ser utilizadas em uma ampla diversidade de problemas combinatórios. Tendo em vista os aspectos observados, a velocidade de convergência e o custo computacional são características essenciais para o bom funcionamento do algoritmo PSO. Diante disso, somos levados a acreditar que este trabalho pode contribuir significativamente na viabilidade da utilização das topologias aplicadas ao PSO. Concluimos, portanto, que este estudo pode ajudar diversos outros pesquisadores a desenvolverem novas formas de comunicabilidade para o PSO, tendo em vista a utilização do PSO nas diversas áreas científicas assim como na indústria.

6 Referências

Eberhart R. and Kennedy J., **A New Optimizer Using Particle Swarm Optimization Theory**, Proceedings of the Sixth International Symposium on Micro Machine and Human Science, (Nagoya, Japan), pp. 39-43, 1995.

Heppener F. and U. Grenander, **A stochastic nonlinear model for coordinated bird flocks**, The Ubiquity of Chaos, AAAS Publication, pp. 223-238, 1990.

Bratton D. and Kennedy J., **Defining a Standard for Particle Swarm Optimization**, Swarm Intelligence Symposium, 2007. SIS 2007. IEEE, pp. 120-127, 2007.

[5] Clerc M. and Kennedy J., **The Particle Swarm – Explosion, Stability, and Convergence in a Multidimensional Complex Space**, IEEE Transactions on Evolutionary Computation, pp. 58-73, 2002.

[1] Kennedy J., **Small Worlds and Mega-minds: Effects of Neighborhood Topology on Particle Swarm Optimizatón**, Proc. IEEE Congr. Evol. Comput., 3, pp. 1931-1938, 1999.