

# UM ALGORITMO EVOLUTIVO ECO-INSPIRADO APLICADO A OTIMIZAÇÃO NUMÉRICA

**Rafael Stubs Parpinelli**

Universidade do Estado de Santa Catarina  
parpinelli@joinville.udesc.br

**Heitor Silvério Lopes**

Universidade Federal Tecnológica do Paraná  
hslopes@utfpr.edu.br

**Resumo** – A busca por ideias, modelos e paradigmas computacionais inspirados na natureza sempre se mostrou de grande interesse para os cientistas da computação, em especial daqueles pertencentes à área conhecida como Computação Natural. O conceito de otimização está presente em vários processos naturais como, por exemplo, na evolução das espécies, no comportamento de grupos sociais, na dinâmica do sistema imunológico, na definição de estratégias de busca por alimento ou nas relações ecológicas de diferentes populações. Este trabalho utiliza conceitos ecológicos como habitats, relações ecológicas e sucessões ecológicas para a definição de um algoritmo eco-inspirado para otimização, chamado ECO. A abordagem proposta utiliza de forma cooperativa e coevolutiva diferentes populações de soluções candidatas, onde cada população evolui de acordo com uma meta-heurística específica e interagem utilizando conceitos ecológicos. A meta-heurística empregada neste trabalho é o algoritmo populacional *Artificial Bee Colony* (ABC). Experimentos foram feitos com quatro funções matemáticas de *benchmark* e os resultados foram comparados com aqueles obtidos pelo algoritmo sendo executado sem a utilização dos conceitos ecológicos propostos (ECO-no). O algoritmo ECO mostrou-se significativamente melhor do que o ECO-no, especialmente à medida que aumenta a dimensionalidade de cada função, possivelmente devido às interações ecológicas (intra e inter-habitats) que favorecem a coevolução das populações. Os resultados obtidos sugerem que o algoritmo eco-inspirado pode ser uma alternativa interessante para problemas de otimização numérica.

**Palavras-chave** – Otimização, busca cooperativa, co-evolução, habitats, relações ecológicas, ecologia

**Abstract** – *The search for nature-inspired ideas, models and computational paradigms always showed of great interest for computer scientists, particularly for those from the Natural Computing area. The concept of optimization is present in several natural processes, for instance, in the evolution of species, in the behavior of social groups, in the dynamics of the immune system, in the food search strategies and ecological relationships of different animal populations. This work uses the ecology concepts of habitats, ecological relationships and ecological succession to build an eco-inspired optimization algorithm, named ECO. The proposed approach uses several populations of candidate solutions that cooperates and coevolves each other, according to a given meta-heuristics. In this particular work, we used the Artificial Bee Colony algorithm as the main meta-heuristic. Experiments were done for optimizing high-dimensional mathematical functions. Results were compared with the same algorithm running without the ecology concepts previously mentioned (ECO-no). The ECO algorithm performed significantly better than ECO-no, especially as the dimensionality of the function increased, possibly thanks to the ecological interactions (intra and inter-habitats) that enabled the coevolution of populations. Results suggest that the eco-inspired algorithm can be an interesting alternative for numerical optimization.*

**Keywords** – *Optimization, cooperative search, co-evolution, habitats, ecological relationships, ecology*

## 1. INTRODUÇÃO

A natureza sempre foi uma fonte inesgotável de inspiração para modelos e paradigmas computacionais, em especial para os cientistas da computação ligados à área conhecida como Computação Natural [1]. As duas principais famílias de algoritmos de otimização que se constituem por abordagens bio-inspiradas são a computação evolucionária (CE) e a inteligência de enxames (IE) que atualmente disponibilizam uma vasta gama de estratégias para otimização de problemas [2, 3].

O conceito de otimização está presente em vários processos naturais como, por exemplo, na evolução das espécies, no comportamento de grupos sociais, na dinâmica do sistema imunológico, nas estratégias de busca por alimento e nas relações ecológicas de diferentes populações. A maioria destes processos, por si só, já serve de inspiração para o desenvolvimento de algoritmos de otimização. Porém, até então, não tem sido explorados computacionalmente em um contexto de otimização os conceitos ecológicos de habitats, relações ecológicas e sucessões ecológicas [4, 5]. Este é o foco deste trabalho que propõe um algoritmo eco-inspirado para problemas de otimização numérica.

O principal diferencial da abordagem proposta é a utilização cooperativa de diferentes populações de soluções candidatas, coevoluindo em um contexto ecológico. Cada população se comporta de acordo com uma estratégia específica de busca, empre-

gada na evolução das soluções candidatas. Além da possibilidade de utilização cooperativa de diferentes estratégias de busca, a metodologia abre a possibilidade de inserção de conceitos ecológicos no processo de otimização.

Na sequência, a Seção 2 descreve em detalhes o algoritmo eco-inspirado, a Seção 3 mostra os resultados obtidos, e a Seção 4 conclui o trabalho e aponta direcionamentos futuros.

## 2. ABORDAGEM PROPOSTA

O algoritmo eco-inspirado, abreviado para ECO, representa uma nova perspectiva para o desenvolvimento de algoritmos de busca cooperativa. O ECO é composto por populações de indivíduos (soluções candidatas para um problema a ser otimizado) e cada população evolui de acordo com uma meta-heurística. Desta forma, os indivíduos de cada população são modificados conforme os mecanismos de intensificação e diversificação, e os parâmetros iniciais, específicos de cada meta-heurística. As populações podem se comportar de forma homogênea ou heterogênea. Um comportamento homogêneo implica que todas as populações evoluem de acordo com a mesma estratégia, configurada com os mesmos parâmetros. Qualquer modificação nas estratégias ou parâmetros de pelo menos uma população já caracteriza um comportamento heterogêneo.

A inspiração ecológica advém da utilização de alguns conceitos ecológicos, tais como: habitats, relações ecológicas e sucessões ecológicas [4, 5]. Uma vez dispersas no espaço de busca, populações de indivíduos que transitam por uma mesma região constituem um habitat ecológico. Sendo assim, um habitat é um conjunto cujos elementos são populações que pertencem a uma mesma região do espaço de busca. Por exemplo, em uma hipersuperfície multimodal, cada pico pode se tornar um habitat promissor para algumas populações. Uma hipersuperfície pode possuir vários habitats. Assim como na natureza, as populações podem circular por todo ambiente. Porém, cada população pode pertencer somente a um único habitat em um dado instante de tempo  $t$ . Portanto, por definição, a intersecção entre os habitats no instante  $t$  é o conjunto vazio.

Com a definição dos habitats, duas categorias de relações ecológicas podem ser definidas. As relações intra-habitats que ocorrem entre as populações internas a cada habitat e as relações inter-habitats que ocorrem entre os habitats [4, 5].

No ECO, a relação intra-habitat empregada é a relação de acasalamento. Nesta relação, populações pertencentes a um mesmo habitat podem gerar um vínculo reprodutivo entre seus indivíduos, miscigenando as populações e favorecendo a co-evolução das mesmas. Populações pertencentes a habitats diferentes são ditas reprodutivamente isoladas, não podendo assumir tal vínculo.

A relação inter-habitats empregada é a das grandes migrações. Nesta relação, indivíduos pertencentes a um habitat migram para outros habitats com o objetivo de identificarem regiões mais promissoras à sobrevivência e encontrar novos parceiros para acasalamento.

Além dos mecanismos de intensificação/diversificação específicos de cada meta-heurística, quando considerando o contexto ecológico do algoritmo proposto, as relações intra-habitats são responsáveis pela intensificação da busca e as relações inter-habitats são responsáveis pela diversificação da busca.

Dentro da metáfora ecológica, as sucessões ecológicas representam o processo transformacional do sistema. Neste processo, os agrupamentos populacionais se formam (habitats), as relações entre as populações se estabelecem e o sistema se estabiliza através da auto-organização de seus componentes.

O Algoritmo 1 mostra o pseudocódigo da abordagem proposta. Primeiramente, no instante  $t = 0$ , são inicializados aleatoriamente todos os indivíduos de todas as populações  $Q_i$  com  $i = 1, \dots, Q$ , onde  $Q$  é o número total de populações (linha 2).

O laço principal representa as sucessões ecológicas (linhas 3 a 12). O primeiro passo a ser realizado no laço principal é a execução do período evolutivo de cada população  $Q_i(t)$  (linha 4). Neste passo, cada população explora o espaço de busca do problema livremente de acordo com seus critérios de intensificação/diversificação.

Ao final dos períodos evolutivos de todas as populações é necessário identificar a região de referência de cada população (linha 5). A região de referência é utilizada para definir os habitats do sistema. A métrica escolhida para definir a região de referência de cada população é o centróide  $C_i$ , com  $i = 1, \dots, Q$ . O centróide representa o ponto no espaço onde há a maior concentração dos indivíduos da espécie  $i$ . Uma vez calculado os centróides, verifica-se a distância entre as populações ( $D_{SS} : Q_i \times Q_k \rightarrow \mathbb{R}_0^+$ , com  $i, k = 1, \dots, Q, i \neq k$ ) e definem-se os habitats  $H_j(t)$  ( $D_{SS}(Q_i, Q_k) \leq \rho$ , sendo  $\rho \in [0..1]$  um limiar mínimo) (linha 6). A distância entre as populações é dependente da definição e representação de cada população. Nesta aplicação a distância empregada é a distância Euclidiana. Ao final deste passo todas as populações estão associadas a um habitat:  $Q_i^j(t)$ , população  $i$  pertence ao habitat  $j$  no instante  $t$ .

Com os  $H$  habitats definidos, se faz necessário a determinação das topologias de comunicação intra-habitats  $TC_j(t)$  para cada habitat  $H_j(t)$  com  $j = 1, \dots, H$  (linha 7). As topologias são definidas por uma matriz de adjacência entre as populações pertencentes a cada habitat. Uma vez definidas as topologias intra-habitats, ocorre a relação ecológica de acasalamento entre populações adjacentes (linha 8). Nesta relação, um indivíduo de cada população é escolhido utilizando a estratégia de torneio estocástico [6] e é realizada troca genética entre eles de modo a gerar um novo indivíduo. Para cada gene dos progenitores é sorteada a probabilidade do gene do progenitor 1 ou progenitor 2 compor o novo indivíduo, sendo que esses genes possuem as mesmas chances de serem escolhidos. O novo indivíduo gerado substitui um indivíduo selecionado aleatoriamente na sua população, excluindo o melhor indivíduo.

Feita a interação entre as populações de cada habitat, a topologia  $TH(t)$  de interação entre os habitats (linha 9) é definida aleatoriamente. Esta topologia inter-habitats  $TH(t)$  é utilizada para a realização da relação ecológica das grandes migrações. Nesta relação, para cada habitat é escolhida aleatoriamente uma população pertencente a ele. O melhor indivíduo desta população

migra para outro habitat aleatório e, no habitat de destino, este substitui um indivíduo escolhido aleatoriamente, excluindo o melhor indivíduo (linha 10).

O laço principal se repete até que os ciclos de sucessão ecológica alcancem um valor máximo predefinido.

---

**Algorithm 1** Pseudocódigo geral do ECO.

---

- 1: Considere  $i = 1, \dots, Q, j = 1, \dots, H$  e  $t = 0$ ;
  - 2: Inicialize cada população  $Q_i(t)$  com  $n_i$  soluções candidatas aleatórias;
  - 3: **while** critério de parada não for satisfeito **do** {Ciclos de sucessão ecológica}
  - 4:   Realize período evolutivo para cada população  $Q_i(t)$ ;
  - 5:   Aplique métrica  $C_i$  para identificar a região de referência de cada população  $Q_i(t)$ ;
  - 6:   Com os valores  $C_i$ , defina os  $H$  habitats;
  - 7:   Para cada habitat  $H_j(t)$  defina a topologia de comunicação  $TC_j(t)$  entre as populações  $Q_i^j(t)$ ;
  - 8:   Para cada topologia  $TC_j(t)$ , realize interação entre as populações  $Q_i^j(t)$ ;
  - 9:   Defina a topologia de comunicação  $TH(t)$  entre os  $H_j(t)$  habitats;
  - 10:   Para a topologia  $TH(t)$ , realize interação entre os  $H_j(t)$  habitats;
  - 11:   Incremente  $t$ ;
  - 12: **end while**
- 

### 3. EXPERIMENTOS E RESULTADOS

Experimentos foram realizados utilizando quatro funções de *benchmark*, extensivamente utilizadas na literatura para teste de métodos de otimização numérica [7]. Cada função foi testada com 2, 5, 10 e 200 dimensões e, para todas elas, o objetivo é a minimização.

A primeira função ( $f_1(\vec{x})$ ) é conhecida como a função de Schaffer F6 generalizada [8] que é fortemente multimodal. A segunda função ( $f_2(\vec{x})$ ) é a de Rastrigin que também é multimodal e baseada na função esfera com a adição de uma modulação cossenóide para produzir um grande número de mínimos locais [9]. A grande dificuldade em encontrar o ótimo global para esta função está no fato de que algoritmos de otimização numérica facilmente ficam presos a mínimos locais. A terceira função ( $f_3(\vec{x})$ ) é a função de Griewank [10] que, como as demais, é fortemente multimodal. Particularmente o número de mínimos locais para esta função aumenta exponencialmente com a sua dimensionalidade [11], o que a torna um problema de otimização muito difícil. A quarta função ( $f_4(\vec{x})$ ) é a de Rosenbrock [12] que tem a particularidade de ter uma hipersuperfície de busca na forma de um vale parabólico longo e estreito, onde se encontra o ótimo global. A Tabela 1 resume as informações sobre a funções utilizadas.

Tabela 1: Funções Numéricas

Função	Domínio	Valor Ótimo
$f_1(\vec{x}) = \sum_{i=1}^{n-1} \left( 0.5 + \frac{\sin^2(\sqrt{x_{i+1}^2 + x_i^2}) - 0.5}{(0.001(x_{i+1}^2 + x_i^2) + 1)^2} \right)$	$-100 \leq x_i \leq 100$	$f_1(\vec{0}) = 0$
$f_2(\vec{x}) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	$-5.12 \leq x_i \leq 5.12$	$f_2(\vec{0}) = 0$
$f_3(\vec{x}) = \frac{1}{4000} \left( \sum_{i=1}^n x_i^2 \right) - \left( \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) \right) + 1$	$-600 \leq x_i \leq 600$	$f_3(\vec{0}) = 0$
$f_4(\vec{x}) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	$-30 \leq x_i \leq 30$	$f_4(\vec{1}) = 0$

Os parâmetros do ECO a serem definidos são: número de populações (*N-POP*) que serão coevoluídas, número de ciclos de sucessão ecológica (*ECO-STEP*), tamanho do período evolutivo (*EVO-STEP*), tamanho do torneio (*T-SIZE*) e o limiar de proximidade  $\rho$ . Ainda não foram realizados estudos sobre o ajuste ótimo dos parâmetros. Assim, todos os parâmetros do algoritmo foram definidos empiricamente.

Em todos os experimentos foram utilizados o algoritmo populacional *Artificial Bee Colony Optimization* (ABC) [13] de forma homogênea, ou seja, todas as populações utilizam este algoritmo, com os mesmos parâmetros, para evoluírem suas soluções candidatas. Para o algoritmo ABC, o único parâmetro de controle definido é o tamanho da população (*POP-SIZE*).

Na sequência, a Seção 3.1 ilustra passo a passo o comportamento do algoritmo proposto e a aplicação computacional dos conceitos ecológicos. A Seção 3.2 mostra os resultados numéricos obtidos na otimização das funções de *benchmark*.

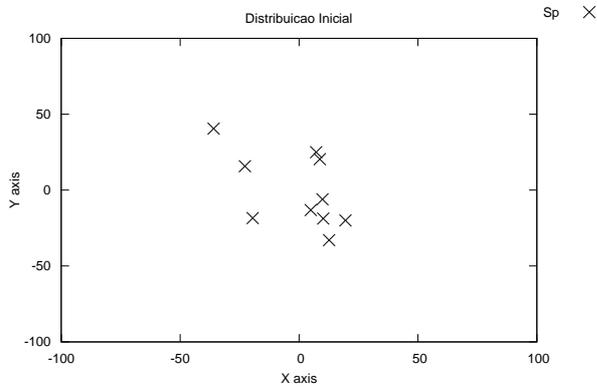
#### 3.1 Ilustração Conceitual

A função de Schaffer generalizada ( $f_1(\vec{x})$ ) foi escolhida para ilustrar o comportamento do algoritmo proposto. Como mostrado na Tabela 1,  $\vec{x}$  é definido no domínio de  $[-100, 100]$  e o mínimo global para  $f(\vec{x})$  é 0 e corresponde à solução ótima  $\vec{x}_{opt} = (x_1, x_2, \dots, x_n) = (0, 0, \dots, 0)$ . Com o propósito de permitir uma avaliação visual do comportamento do algoritmo, a função de Schaffer foi aqui definida com apenas duas dimensões.

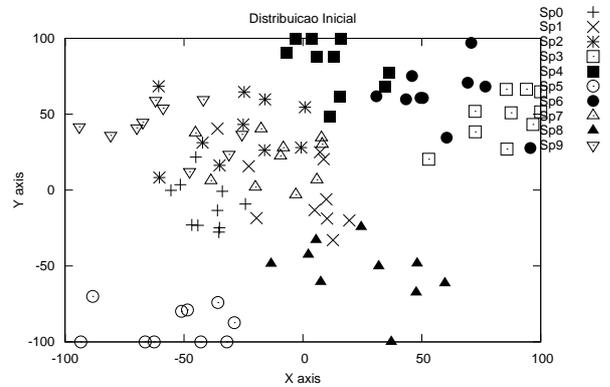
Nesta ilustração, os parâmetros utilizados foram *N-POP* = 10, *POP-SIZE* = 10, *ECO-STEP* = 100, *EVO-STEP* = 5, *T-SIZE* = 5 e  $\rho = 0,5$ . Com esta configuração, o número total de avaliações de função é de 10000 para cada população, sendo realizadas 100 avaliações para cada população a cada sucessão ecológica.

Primeiramente são iniciadas aleatoriamente todas as populações. Para tanto utiliza-se uma distribuição normal ( $N(\mu, \sigma)$ ) com média  $\mu$  (sendo escolhida aleatoriamente dentro do domínio de cada dimensão do problema) e desvio padrão  $\sigma$  (também proporcional ao domínio do problema). A amostragem aleatória é feita utilizando a transformada de Box-Muller [14].

A Figura 1(a) ilustra a distribuição inicial dos indivíduos de uma única população, enquanto que a Figura 1(b) mostra a distribuição inicial de todas as populações.



(a) Distribuição inicial de uma população.



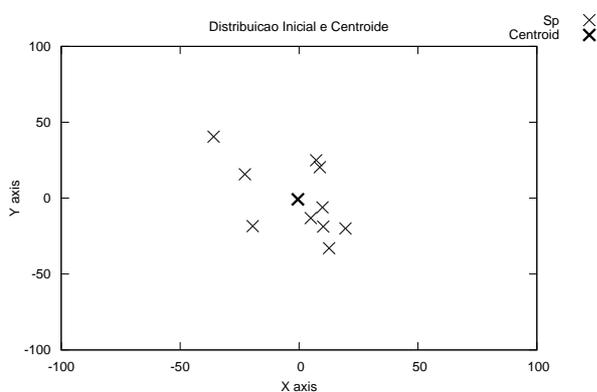
(b) Distribuição inicial de todas populações.

Figura 1: Distribuição inicial.

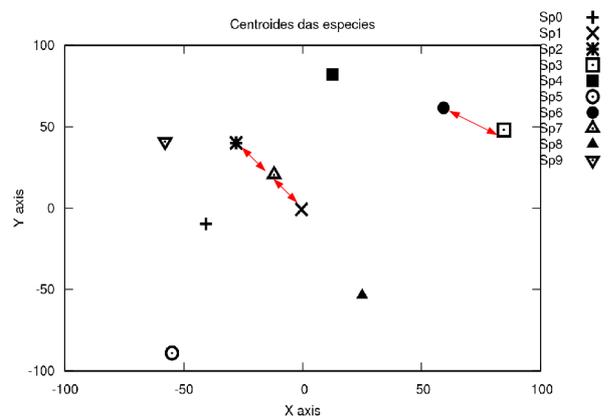
Uma vez inicializadas as populações, o algoritmo entra no laço das sucessões ecológicas que se inicia com o período evolutivo. Neste estágio, todas as populações evoluem suas soluções por um número preestabelecido de iterações. Neste exemplo, o período evolutivo foi definido em 5 iterações.

Após o período evolutivo, inicia-se a definição dos habitats do sistema. Os habitats são as regiões em que se concentram as populações. Nesta proposta, a região de referência de uma população é definida pelo seu centróide e é calculado pela Equação 1, onde  $Pop$  representa o número de indivíduos da população. A Figura 2(a) ilustra a distribuição de uma população e seu respectivo centróide. A Figura 2(b) ilustra a distribuição dos centróides de todas as 10 populações para a primeira sucessão ecológica.

$$C = \frac{\sum_{i=1}^{Pop} \vec{x}_i}{Pop} \quad (1)$$



(a) Centróide de uma distribuição inicial.



(b) Centróides de todas as populações.

Figura 2: Centróides das populações.

Uma vez calculado os centróides de cada população, verifica-se a distância entre eles e definem-se os habitats de acordo com o limiar mínimo  $\rho$  pela métrica Euclidiana normalizada. De acordo com  $\rho$ , é gerada uma matriz de adjacência onde duas

populações são ditas adjacentes entre si se elas estiverem a uma distância mínima de  $\rho$  uma da outra. Para o exemplo da Figura 2(b), as informações de adjacência são mostradas nas setas em vermelho: Sp0 não é adjacente a nenhuma população; Sp1 é adjacente a Sp7; Sp2 é adjacente a Sp7; Sp3 é adjacente a Sp6; Sp4 não é adjacente a nenhuma população; Sp5 não é adjacente a nenhuma população; Sp6 é adjacente a Sp3; Sp7 é adjacente das populações Sp1 e Sp2; Sp8 não é adjacente a nenhuma população; e Sp9 não é adjacente a nenhuma população.

Com a matriz de adjacência são gerados os habitats. A Figura 3 ilustra os habitats encontrados para a distribuição de centróides da Figura 2(b), com  $\rho = 0,5$ . Nesta figura, observa-se a existência de sete habitats:  $H_0$  composto por Sp0;  $H_1$  composto por Sp1, Sp2 e Sp7;  $H_2$  composto por Sp3 e Sp6;  $H_3$  composto por Sp4;  $H_4$  composto por Sp5;  $H_5$  composto por Sp8; e  $H_6$  composto por Sp9.

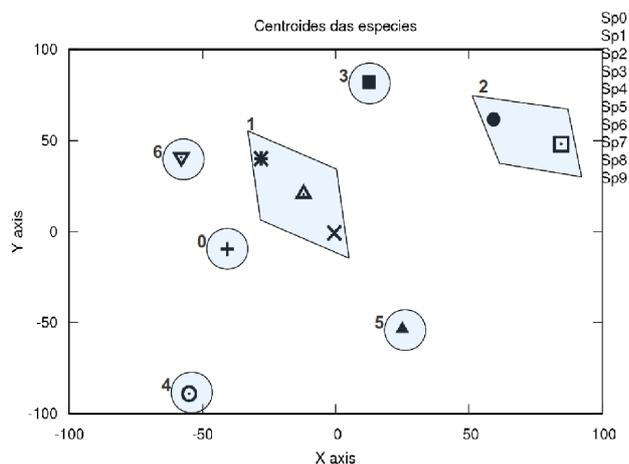


Figura 3: Distribuição inicial dos habitats.

A matriz de adjacência define também a topologia de comunicação intra-habitat. Por exemplo, no habitat  $H_1$ , composto pelas populações Sp1, Sp2 e Sp7, a população Sp1 pode estabelecer relação com a população Sp7, a população Sp2 pode estabelecer relação com a população Sp7, e a população Sp7 pode estabelecer relação com as populações Sp1 e Sp2. Neste momento tem-se definido os habitats com suas populações e topologias de comunicação.

O próximo passo é a realização da comunicação entre as populações dentro de cada habitat. No algoritmo proposto, populações que possuem adjacência entre si interagem pela relação ecológica de acasalamento. Por exemplo, a população Sp7 do habitat  $H_1$  seleciona um indivíduo de sua população e uma população adjacente para estabelecer relação. A população adjacente é escolhida aleatoriamente e, neste caso, poderia ser a população Sp1 ou Sp2. Em cada uma das populações, os indivíduos escolhidos para realizar acasalamento são selecionados utilizando a estratégia de torneio com tamanho 5. Um novo indivíduo é gerado substituindo um indivíduo escolhido aleatoriamente dentro da população adjacente. As populações Sp1 e Sp2 realizam o mesmo procedimento. Todos os habitats compostos por mais de uma população realizam a relação de acasalamento de acordo com a topologia definida pela matriz de adjacência.

Após as interações intra-habitats é necessário definir a topologia de comunicação inter-habitats para a realização da relação ecológica das grandes migrações. Nesta relação, para cada habitat é escolhida aleatoriamente uma população pertencente a ele. O melhor indivíduo desta população migra para outro habitat aleatório e, no habitat de destino, este substitui um indivíduo escolhido aleatoriamente, ressalvo o melhor indivíduo. Neste estágio do algoritmo, recomeça-se o laço das sucessões ecológicas.

A Figura 4 mostra a evolução do número de habitats a cada passo de sucessão ecológica.

Como se pode observar, o sistema evoluiu e convergiu para a formação de três habitats. A Figura 5 ilustra os habitats encontrados para a última sucessão ecológica. São eles:  $H_0$  composto por Sp0, Sp1, Sp2, Sp3, Sp4, Sp5, Sp7 e Sp9;  $H_1$  composto por Sp6; e  $H_2$  composto por Sp8. Observa-se também na Figura 5 que as populações pertencentes ao habitat  $H_0$  convergiram para o ótimo global da função analisada e os outros habitats convergiram para regiões próximas ao ótimo global.

### 3.2 Otimização Numérica

O algoritmo eco-inspirado (ECO) foi executado utilizando duas configurações. Uma configuração executa o Algoritmo 1 como descrito na Seção 2, com as definições de habitats, topologias e relações ecológicas, e outra configuração desabilitando a capacidade de criar habitats e, conseqüentemente, de definir topologias e interações. Esta segunda configuração simula a evolução completamente isolada das populações. Para cada configuração, o algoritmo foi executado 30 vezes.

Para o número de dimensões igual a 2, 5 e 10, a única modificação nos parâmetros utilizados com relação à ilustração anterior é o número de populações que foi definido em 100. Para o número de dimensões igual a 200, dado o número elevado de dimensões, foram redefinidos o número de populações para 200, o número de ciclos de sucessão ecológica para 500, e o período evolutivo para 10 iterações. Com este ajuste de parâmetros, para 200 dimensões, o número total de avaliações de função a cada

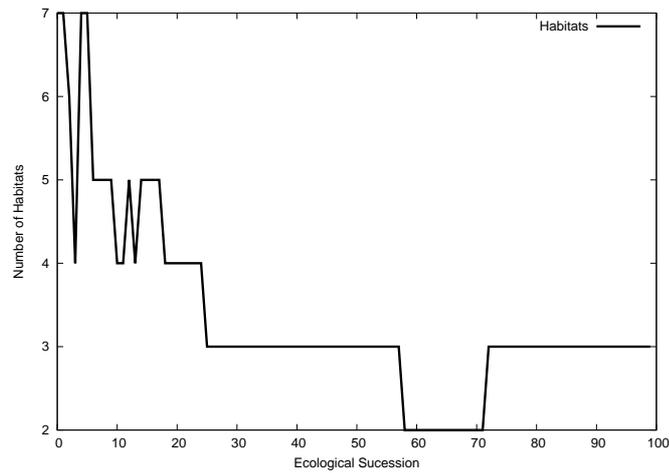


Figura 4: Evolução do número de habitats durante as sucessões ecológicas.

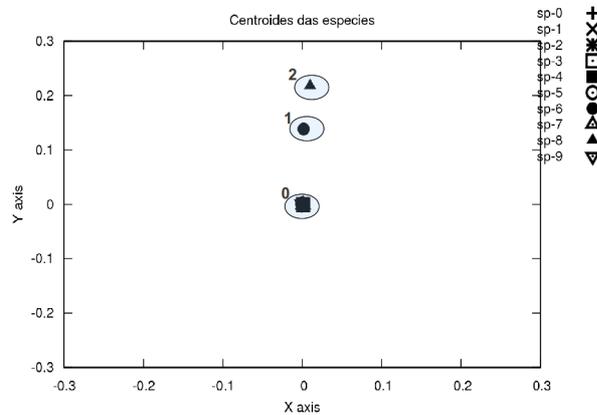


Figura 5: Distribuição final dos habitats.

execução do algoritmo passou de 10000 para 100000 avaliações para cada população, sendo realizado 200 avaliações para cada população a cada sucessão ecológica.

A Tabela 2 mostra as médias obtidas para as funções de *benchmark*. Para cada uma das funções, a primeira linha discrimina o número de dimensões da função, sendo 2, 5, 10 e 200 respectivamente. A segunda linha discrimina a configuração do algoritmo, sendo com (ECO) e sem (ECO-no) o uso dos conceitos ecológicos, respectivamente. A terceira linha (*Med melhor*) mostra a média e desvio padrão dos melhores resultados obtidos por todas as populações em cada execução. Neste experimento, a cada execução do algoritmo tem-se a evolução de 100 populações, cada qual com sua convergência. Ao final de uma execução tem-se a média dos melhores resultados das 100 populações. Ao final das 30 execuções faz-se a média das médias. A quarta linha (*Melhor global*) mostra a média e o desvio padrão do melhor resultado obtido por todas as populações em cada execução.

Analisando a Tabela 2, pode-se observar que o algoritmo ECO proposto obteve resultados significativamente melhores do que o algoritmo sendo executado sem os conceitos de habitat e interações entre populações (ECO-no) em todas as funções. Este ganho pode ser melhor observado com o aumento da dimensionalidade de cada função e se deve principalmente às interações ecológicas (intra e inter-habitats) que favorecem a coevolução das populações.

A Figura 6 mostra a evolução do número médio de habitats a cada passo de sucessão ecológica para as 30 execuções do algoritmo ECO. Como o comportamento observado é o mesmo para todas as funções, a Figura 6 mostra a evolução para a função  $f_1(\vec{x})$  com  $D = 10$ . Pode-se observar que no início do processo de otimização, com as populações mais dispersas no espaço de busca, existe um maior número de habitats. À medida que o processo de otimização avança através das sucessões ecológicas, as populações tendem a se deslocar para as regiões que cada uma converge. Como mostrado na Figura 6, o número de habitats reduz com o passar dos ciclos de sucessão ecológica, indicando que as populações tendem a convergir para pontos próximos umas das outras.

Tabela 2: Resultados obtidos para as funções de *benchmark*.

$f_1(\vec{x})$	$D = 2$		$D = 5$		$D = 10$		$D = 200$	
Modelo	ECO	ECO-no	ECO	ECO-no	ECO	ECO-no	ECO	ECO-no
Med melhor	0.0118 ± 0.0	1.7345 ± 0.6	0.5239 ± 0.1	9.6695 ± 0.4	3.9895 ± 0.3	12.6312 ± 0.3	20.5058 ± 0.2	28.5370 ± 0.8
Melhor global	0.0000 ± 0.0	0.0186 ± 0.1	0.1341 ± 0.0	3.7867 ± 1.5	2.6070 ± 0.2	10.0630 ± 0.9	18.3801 ± 0.4	27.5936 ± 0.7
$f_2(\vec{x})$	$D = 2$		$D = 5$		$D = 10$		$D = 200$	
Modelo	ECO	ECO-no	ECO	ECO-no	ECO	ECO-no	ECO	ECO-no
Med melhor	0.0022 ± 0.0	0.8426 ± 0.2	0.1694 ± 0.2	12.2897 ± 3.8	1.2263 ± 1.0	55.8775 ± 3.5	4.6851 ± 1.7	208.6492 ± 2.2
Melhor global	0.0000 ± 0.0	0.0000 ± 0.0	0.0000 ± 0.0	0.3832 ± 0.6	0.0000 ± 0.0	12.6207 ± 4.3	0.0003 ± 0.0	136.3832 ± 6.4
$f_3(\vec{x})$	$D = 2$		$D = 5$		$D = 10$		$D = 200$	
Modelo	ECO	ECO-no	ECO	ECO-no	ECO	ECO-no	ECO	ECO-no
Med melhor	$10^{-17} \pm 0.0$	$10^{-17} \pm 0.0$	0.0012 ± 0.0	0.0016 ± 0.0	0.0002 ± 0.0	1.1613 ± 1.2	$10^{-10} \pm 0.0$	$10^{-7} \pm 0.0$
Melhor global	0.0000 ± 0.0	$10^{-19} \pm 0.0$	$10^{-19} \pm 0.0$	$10^{-12} \pm 0.0$	$10^{-13} \pm 0.0$	$10^{-06} \pm 0.0$	$10^{-11} \pm 0.0$	$10^{-8} \pm 0.0$
$f_4(\vec{x})$	$D = 2$		$D = 5$		$D = 10$		$D = 200$	
Modelo	ECO	ECO-no	ECO	ECO-no	ECO	ECO-no	ECO	ECO-no
Med melhor	0.0000 ± 0.0	4.9315 ± 0.6	1.0332 ± 1.5	369.5634 ± 219.3	87141.5 ± 1562.8	$2.65 \times 10^6 \pm 3421.2$	333.0645 ± 120.0	$5.19 \times 10^6 \pm 21664.5$
Melhor global	0.0000 ± 0.0	0.0015 ± 0.0	0.0019 ± 0.0	0.4379 ± 0.2	0.0646 ± 0.0	93.2517 ± 62.7	137.8643 ± 42.0	13036.1 ± 4193.4

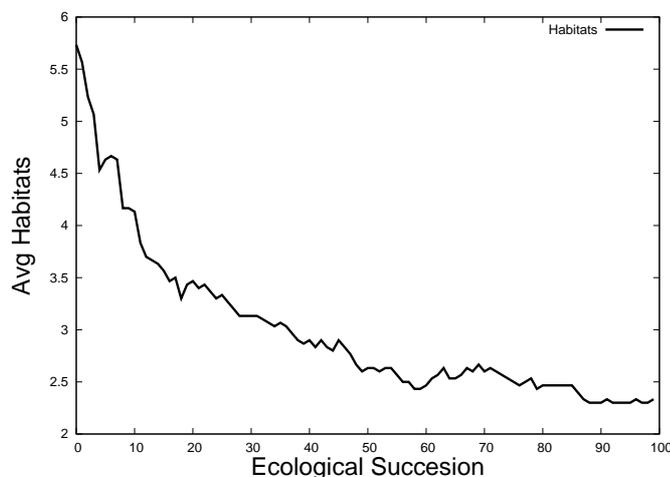


Figura 6: Número médio de habitats a cada sucessão ecológica.

## 4. CONCLUSÃO

Este trabalho apresenta um algoritmo eco-inspirado para otimização, chamado de ECO. O algoritmo proposto usa busca cooperativa onde populações de indivíduos coevoluem e interagem entre si utilizando alguns conceitos ecológicos. Cada população se comporta de acordo com os mecanismos de intensificação e diversificação, e os parâmetros de controle específicos de cada meta-heurística (neste trabalho foi utilizado o algoritmo populacional *Artificial Bee Colony Optimization* em todas as populações).

Os principais conceitos ecológicos abordados são a definição de habitats, relações ecológicas e sucessões ecológicas. Juntamente com estes conceitos, a definição de topologias de comunicação intra e inter-habitats são inseridas para compor o algoritmo. Estas características trazem maior plausibilidade biológica ao algoritmo proposto.

Experimentos foram realizados considerando a execução do algoritmo com e sem o uso dos conceitos ecológicos em diferentes funções de *benchmark* e com diferentes graus de complexidade. Os resultados mostraram que o uso de habitats e relações ecológicas influenciam significativamente no processo de coevolução das populações, levando à obtenção de melhores soluções (do que quando não são utilizados os conceitos ecológicos).

Este trabalho ainda está em desenvolvimento e, como trabalhos futuros, pretende-se analisar a influência dos parâmetros do sistema (número sucessões ecológicas, período evolutivo, número de populações, e fator  $\rho$  para criação de habitats) na qualidade das soluções, bem como utilizar diferentes meta-heurísticas no modelo proposto (sistema heterogêneo). Esta última linha de pesquisa visa analisar o efeito que cada meta-heurística pode ter nos resultados, tendo em mente que cada uma tem a sua forma própria de explorar o espaço de soluções. Atualmente está em desenvolvimento um *framework* paralelo para aumentar as capacidades computacionais do algoritmo proposto, de modo a ser aplicável a problemas de alta complexidade.

## REFERÊNCIAS

- [1] L. N. de Castro. "Fundamentals of natural computing: an overview". *Physics of Life Reviews*, vol. 4, no. 1, pp. 1–36, 2007.
- [2] A. P. Engelbrecht. *Computational Intelligence: An Introduction*. Wiley, Chichester, UK, second edition, 2007.

- [3] R. S. Parpinelli and H. S. Lopes. “New inspirations in swarm intelligence: a survey”. *International Journal of Bio-Inspired Computation*, vol. 3, no. 1, pp. 1–16, 2011.
- [4] M. Begon, C. R. Townsend and J. L. Harper. *Ecology: from individuals to ecosystems*. Blackwell Publishing, Oxford, UK, fourth edition, 2006.
- [5] R. M. C. May and A. R. McLean. *Theoretical Ecology: Principles and Applications*. Oxford University Press, Oxford, UK, 2007.
- [6] T. Blicke. “Tournament selection”. In *Evolutionary Computation*, edited by T. Bäck, D. Fogel and Z. Michalewicz, volume 2, pp. 181–186. Institute of Physics, Bristol, UK, 2000.
- [7] J. G. Digalakis and K. G. Margaritis. “An Experimental Study Of Benchmarking Functions For Evolutionary Algorithms”. *International Journal of Computer Mathematics*, vol. 79, no. 4, pp. 403–416, 2002.
- [8] C. A. Floudas and P. M. Pardalos. *A collection of test problems for constrained global optimization problems*, volume 455 of *Lecture Notes in Computer Science*. Springer, 1990.
- [9] H. Mühlenbein, D. Schomisch and J. Born. “The Parallel Genetic Algorithm as Function Optimizer”. *Parallel Computing*, vol. 17, no. 6-7, pp. 619–632, 1991.
- [10] A. Griewank. “Generalized descent for global optimization”. *Journal of Optimization Theory and Applications*, vol. 34, no. 1, pp. 11–39, 1981.
- [11] H. Cho, F. Olivera and S. Guikema. “A derivation of the number of minima of the Griewank function”. *Applied Mathematics and Computation*, vol. 204, no. 2, pp. 694–701, 2008.
- [12] H. Rosenbrock. “An automatic method for finding the greatest or least value of a function”. *The Computer Journal*, vol. 3, pp. 175–184, 1960.
- [13] D. Karaboga and B. Akay. “A Comparative Study of Artificial Bee Colony Algorithm”. *Applied Mathematics and Computation*, vol. 214, pp. 108–132, 2009.
- [14] G. E. P. Box and M. E. Muller. “A Note on the Generation of Random Normal Deviates”. *The Annals of Mathematical Statistics*, vol. 29, no. 2, pp. 610–611, 1958.