

Algoritmo Genético Aplicado nas Funções do CEC2014.

Igor Abreu da Silva
Programa de Engenharia Elétrica
Universidade Federal do Rio de Janeiro
igorabreu.ds@poli.ufrj.br

I. INTRODUÇÃO

Este relatório faz parte do trabalho desenvolvido para a disciplina Métodos Computacionais Inspirados na Natureza (CPC881). Ministrada em 2019.2 no Programa de Engenharia Civil da COPPE/UFRJ em que o principal objetivo é minimizar determinadas funções da competição do *Congress on Evolutionary Computation* (CEC2014) [1] através da utilização de um algoritmo genético (AG).

Algoritmo genético é um tipo de algoritmo evolucionário concebido inicialmente por Holland [2]. O Algoritmo genético escrito para este trabalho foi disponibilizado de maneira gratuita, tanto para a alteração quanto para utilização, em um repositório do Git, podendo ser encontrado no seguinte endereço: <https://github.com/iabreuda/GeneticAlgorithms>

II. FUNDAMENTAÇÃO TEÓRICA

Os principais componentes de um algoritmo evolucionário podem ser divididos em: Representação; função aptidão; população; seleção de pais; combinação; mutação e seleção de sobreviventes [3]. A escolha de cada um dos parâmetros podem variar, sendo discutidas e definidas na fundamentação teórica II e nos resultados experimentais IV.

A. Representação

Representação consiste em uma forma de modelar as variáveis de um problema. As principais utilizadas são: representação Binária, Inteira e Real.

B. Função custo

Função a ser maximizada ou minimizada.

C. População

Conjunto de possíveis soluções. O tamanho da população pode variar e a população geralmente é gerada aleatoriamente através de uma distribuição uniforme limitada no espaço de busca. É extremamente importante manter a diversidade de uma população para que se evite situações em que a solução convirja para um mínimo local.

D. Seleção de Pais

Mecanismo que selecionará os pais que serão responsáveis pela geração da próxima população. Podemos subdividir essa funcionalidade em duas partes, em que a primeira é responsável por atribuir uma probabilidade de seleção para cada indivíduo e a segunda se encarrega por escolher indivíduos progenitores a partir desta probabilidade.

E. Recombinação

Combina, com uma probabilidade p_c , os cromossomos de dois pais previamente selecionados de tal forma que estes sejam capazes de gerar dois descendentes para a próxima geração.

F. Mutação

Cria pequenas alterações nos genes do cromossomo de um indivíduo, com uma probabilidade p_m , gerando um nova solução que será adicionada na próxima geração.

G. Seleção de Sobreviventes

De todos os pais e filhos, que geraram ou foram gerados nesta geração, a função da seleção de sobreviventes consiste em definir quais destes indivíduos irão compor a população, ou conjunto de soluções, da próxima geração.

III. METODOLOGIA

As funções de aptidão a serem analisadas estão descritas na tabela I [1]

Função	Nome	Mínimo Global
f_1	Rotated High Conditioned Elliptic	100
f_2	Rotated Bent Cigar	200
f_6	Shifted and Rotated Weierstrass	600
f_7	Shifted and Rotated Griewank's	700
f_9	Shifted and Rotated Rastrigin's	900
f_{14}	Shifted and Rotated HGBat	1400

TABLE I: Funções de Aptidão.

A representação utilizada no problema será composta pelo conjunto \mathbb{R} . Onde a população inicial possui dimensão $D \in [10, 30]$, ou seja, um indivíduo pode ser representado por um vetor $v = [x_1, x_2, \dots, x_D]$, sendo $\{x_n \in \mathbb{R} \mid [-100, 100]\}$. Para o tamanho da população inicial, usaremos um tamanho $\{T \in \mathbb{I} \mid [30, 80]\}$ [4], para probabilidade de mutação $p_m = 0.01$ [4]. Manteremos fixo o número de indivíduos na população em todas as gerações. O critério de parada é definido pelo número máximo de avaliações, ou seja, o número de vezes que a função que calcula a aptidão é chamada pelo código e não é permitido que esse valor ultrapasse $10000 \times (Dimensão)$.

Primeiramente foi executado o AG Básico para todas as funções, onde as definições de cada componente está descrita na tabela II.

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	30
Recombinação	Recombinação de 1 Ponto
Probabilidade de Recombinação (p_c)	0.85
Mutação	Troca de Valor
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Roleta)
Numero de Pais	15
Seleção de Sobreviventes	Geracional

TABLE II: O Algoritmo genético simples.

Função	Melhor	Pior	Média	V.Esperado
f_1	7.41×10^8	7.47×10^9	9.65×10^8	100
f_2	3.45×10^{10}	4.80×10^{10}	3.41×10^{10}	200
f_6	619.57943	619.82433	619.81361	600
f_7	890.10057	907.05796	895.36499	700
f_9	1113.73521	1120.54108	1115.26809	900
f_{14}	1557.92611	1559.75996	1558.55753	1400

TABLE III: Estatísticas do SGA para 10 dimensões e parâmetros da tabela II

Estes resultados serão utilizados como base comparativa para as modificações de componente feitas na parte experimental IV

Cada algoritmo foi executado 51 vezes e os resultados são calculados baseados em uma media dessas 51 execuções. Como é impraticável realizar em tempo hábil todas as combinações de componentes, iremos modificar uma por vez e verificar se ocorre ou não melhora na solução final. No caso de melhora passaremos a utilizar esse novo método para as próximas execuções até encontrar um AG que nos entregue um resultado mais próximo do esperado.

Os métodos a serem utilizados neste trabalho estão descritos na tabela IV, e seus algoritmos podem ser encontrados no livro *Introduction to evolutionary computing*[3].

Os experimentos foram realizados em um notebook Lenovo G50-70: processador *Intel Core i7-4510U* com 8 núcleos, 8 GB de memória RAM e uma placa de vídeo *AMD Radeon R5 M230*. Todos os códigos foram desenvolvidos utilizando-se a linguagem de programação *Python 3.7*.

Componente	Métodos
Dimensão	10, 30
Tamanho da População	30, 50, 80
Recombinação	Crossover-I e Aritméticas
Mutação	Troca, Uniforme e Gaussiana
Probabilidade Seleção de Pais	Proporcional à aptidão e Ranking
Seleção de Pais	Roleta, SUS e Torneio
Seleção de Sobreviventes	Geracional, Aptidão

TABLE IV: Componentes e métodos a serem testados.

IV. RESULTADOS EXPERIMENTAIS

A. Ranking Exponencial

Primeiramente verificamos se ocorre alguma melhora ao modificar a probabilidade de cada indivíduo ser selecionado como pai para um ranking exponencial.

$$p_{exp}(i) = \frac{1 - e^{-1}}{n} \quad (1)$$

Em que i é a posição ocupada pelo indivíduo comparado com outros da mesma população, em que $i = 1$ significa o indivíduo com menor aptidão e n é o fator de normalização.

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	30
Recombinação	Recombinação de 1 Ponto
Probabilidade de Recombinação (p_c)	0.85
Mutação	Troca de Valor
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Ranking Exponencial (Roleta)
Numero de Pais	15
Seleção de Sobreviventes	Geracional
Critério de Parada	10000 \times (<i>Dimensão</i>)

TABLE V: O Algoritmo genético com Ranking exponencial.

Função	Melhor	Pior	Média	V.Esperado
f_1	1.86×10^{11}	1.89×10^{11}	1.87×10^{11}	100
f_2	9.53×10^{11}	9.84×10^{11}	9.70×10^{11}	200
f_6	622.40421	622.90886	622.77548	600
f_7	2303.72747	2319.68514	2314.77048	700
f_9	1293.34952	1297.95282	1297.12562	900
f_{14}	1726.18954	1729.44022	1728.33199	1400

TABLE VI: Estatísticas do GA para 10 dimensões e parâmetros da tabela V

Todos os resultados pioraram com a utilização de um ranking exponencial.

B. SUS

SUS é a abreviação para *Stochastic universal sampling*, é um algoritmo de seleção de pais que evita as possíveis repetições de geradores que podem ocorrer no algoritmo de roleta tradicional.

Em relação ao SGA, tivemos melhores em grande parte dos campos, sendo assim, passaremos a utilizar o SUS como parâmetro de comparação para as próximas modificações.

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	30
Recombinação	Recombinação de 1 Ponto
Probabilidade de Recombinação (p_c)	0.85
Mutação	Troca de Valor
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (SUS)
Numero de Pais	15
Seleção de Sobreviventes	Geracional
Critério de Parada	10000 \times (Dimensão)

TABLE VII: O Algoritmo genético utilizando SUS como mecanismo de seleção de pais.

Função	Melhor	Pior	Média	V.Esperado
f_1	9.62×10^8	2.48×10^9	1.01×10^9	100
f_2	1.96×10^{10}	2.48×10^{10}	1.98×10^{10}	200
f_6	619.39136	619.83561	619.57748	600
f_7	853.63644	875.61589	856.14976	700
f_9	1083.92257	1091.32821	1087.13063	900
f_{14}	1527.55184	1535.65566	1531.77856	1400

TABLE VIII: Estatísticas do GA para 10 dimensões e parâmetros da tabela VII

C. Torneio

Nesse algoritmo de seleção por torneio utilizamos 4 competidores que são selecionados aleatoriamente. O melhor entre eles se torna pai para a formação da próxima geração, os outros voltam para a população e podem ser selecionados novamente até que o número de pais responsáveis por gerar filhos seja completo.

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	30
Recombinação	Recombinação de 1 Ponto
Probabilidade de Recombinação (p_c)	0.85
Mutação	Troca de Valor
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Torneio)
Numero de Pais	15
Seleção de Sobreviventes	Geracional
Critério de Parada	10000 \times (Dimensão)

TABLE IX: O Algoritmo genético utilizando Torneio como mecanismo de seleção de pais.

Função	Melhor	Pior	Média	V.Esperado
f_1	7.51×10^8	2.62×10^9	8.13×10^8	100
f_2	2.68×10^{10}	3.35×10^{10}	2.70×10^{10}	200
f_6	609.86180	610.21468	609.87356	600
f_7	737.41398	746.73616	737.72472	700
f_9	978.32340	983.93245	978.51037	900
f_{14}	1407.53515	1407.70022	1407.54065	1400

TABLE X: Estatísticas do GA para 10 dimensões e parâmetros da tabela IX

Com a modificação para seleção por torneio obtivemos melhoras em f_1 , f_6 , f_7 , f_9 e f_{14} atingindo, assim, um melhor resultado geral e tornando a seleção por torneio o novo parâmetro de comparação.

D. Combinação Aritmética

Consiste em aleatoriamente selecionar um gene do primeiro pai e multiplicá-lo por um valor aleatório r entre 0 e 1, em seguida, multiplica-se o mesmo gene do segundo pai pelo seu complemento $(1-r)$.

Existem outros tipos de recombinação aritméticas, entretanto, elas não apresentaram melhoras relevantes quando comparada com a RAU.

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	30
Recombinação	Recombinação Aritmética Única
Probabilidade de Recombinação (p_c)	0.85
Mutação	Troca de Valor
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Torneio)
Numero de Pais	15
Seleção de Sobreviventes	Geracional
Critério de Parada	10000 \times (Dimensão)

TABLE XI: O Algoritmo genético utilizando Recombinação Aritmética Única.

Função	Melhor	Pior	Média	V.Esperado
f_1	3.39×10^8	8.41×10^9	1.21×10^9	100
f_2	8.83×10^9	2.09×10^{11}	2.02×10^{10}	200
f_6	605.56974	611.40072	606.92599	600
f_7	712.46273	789.23242	723.18836	700
f_9	932.99681	990.37796	945.49764	900
f_{14}	1401.76817	1418.66079	1403.97339	1400

TABLE XII: Estatísticas do GA para 10 dimensões e parâmetros da tabela XI

Mesmo com a piora geral nos piores resultados, obtivemos avanço na média e nos melhores resultados. Sendo assim, a combinação aritmética única se torna uma solução melhor do que a recombinação de um ponto.

E. Mutação Uniforme

Um novo valor aleatório é gerado uniformemente entre os limites do espaço de busca e atribuído ao gene que sofrerá mutação.

A mutação uniforme encontrou melhores resultados mínimos.

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	30
Recombinação	Recombinação Aritmética Única
Probabilidade de Recombinação (p_c)	0.85
Mutação	Uniforme
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Torneio)
Numero de Pais	15
Seleção de Sobreviventes	Geracional
Critério de Parada	$10000 \times (Dimensão)$

TABLE XIII: O Algoritmo genético utilizando Mutação Uniforme.

Função	Melhor	Pior	Média	V.Esperado
f_1	2.60×10^8	1.39×10^{10}	1.41×10^9	100
f_2	1.01×10^9	9.59×10^{10}	8.35×10^9	200
f_6	604.12033	609.89520	605.68776	600
f_7	703.51067	802.00510	715.85551	700
f_9	925.43308	988.30859	939.28946	900
f_{14}	1400.69997	1426.27214	1403.48528	1400

TABLE XIV: Estatísticas do GA para 10 dimensões e parâmetros da tabela XV

F. Mutação Gaussiana

Um valor aleatório a partir de uma distribuição gaussiana $\mu = 0$ e $\sigma = \frac{LimiteSuperior - LimiteInferior}{20}$ é gerado e somada ao valor antigo do gene.

Foram testados $\sigma = [1, 5, 10, 15, 20, 30, 40, 50]$, os melhores valores ocorreram para:

$$\sigma = 10 = \frac{LimiteSuperior - LimiteInferior}{20}. \quad (2)$$

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	30
Recombinação	Recombinação Aritmética Única
Probabilidade de Recombinação (p_c)	0.85
Mutação	Gaussiana
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Torneio)
Numero de Pais	15
Seleção de Sobreviventes	Geracional
Critério de Parada	$10000 \times (Dimensão)$

TABLE XV: O Algoritmo genético utilizando Mutação Gaussiana.

Função	Melhor	Pior	Média	V.Esperado
f_1	1.66×10^8	1.06×10^{10}	1.28×10^9	100
f_2	7.72×10^8	3.98×10^{10}	5.54×10^9	200
f_6	604.20219	609.87757	605.78207	600
f_7	702.99780	777.12629	713.47546	700
f_9	933.32047	985.03260	946.40963	900
f_{14}	1400.63337	1414.88640	1403.09287	1400

TABLE XVI: Estatísticas do GA para 10 dimensões e parâmetros da tabela XV

O número de melhoras maior que a quantidade de pioras justifica adotar a mutação gaussiana como padrão para as próximas execuções.

G. Seleção de Sobreviventes Baseada em Aptidão

A forma anterior priorizava indivíduos mais novos, ou seja, gerados mais recentemente. Na seleção de sobrevivente baseado em aptidão, os progenitores e seus filhos competem para avaliar quem tem a maior aptidão e irá compor a próxima geração.

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	30
Recombinação	Recombinação Aritmética Única
Probabilidade de Recombinação (p_c)	0.85
Mutação	Gaussiana
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Torneio)
Numero de Pais	15
Seleção de Sobreviventes	Baseada em Aptidão
Critério de Parada	$10000 \times (Dimensão)$

TABLE XVII: O Algoritmo genético utilizando sobrevivência baseada em aptidão.

Função	Melhor	Pior	Média	V.Esperado
f_1	2.01×10^8	2.80×10^9	4.82×10^8	100
f_2	3.63×10^7	8.26×10^9	1.27×10^9	200
f_6	604.06900	607.83637	605.05572	600
f_7	701.10594	728.51212	705.11603	700
f_9	934.26451	974.03790	943.22796	900
f_{14}	1400.53091	1406.51125	1401.51456	1400

TABLE XVIII: Estatísticas do GA para 10 dimensões e parâmetros da tabela XVII

Seleção baseada em aptidão melhorou consideravelmente em grande parte dos aspectos analisados.

H. População inicial com 80 indivíduos

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	80
Recombinação	Recombinação Aritmética Única
Probabilidade de Recombinação (p_c)	0.85
Mutação	Gaussiana
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Torneio)
Numero de Pais	15
Seleção de Sobreviventes	Baseada em Aptidão
Critério de Parada	$10000 \times (Dimensão)$

TABLE XIX: O Algoritmo genético com população inicial de 80.

Função	Melhor	Pior	Média	V.Esperado
f_1	2.10×10^8	6.59×10^9	6.12×10^8	100
f_2	2.16×10^7	2.16×10^{10}	2.24×10^9	200
f_6	603.74999	608.80998	605.04579	600
f_7	701.06073	744.63294	706.03977	700
f_9	923.94480	976.94898	935.99702	900
f_{14}	1400.51077	1412.06647	1401.95415	1400

TABLE XX: Estatísticas do GA para 10 dimensões e parâmetros da tabela XIX

Mesmo tendo piorado na segunda e terceira coluna os mínimos foram melhores na maior parte, senda assim, a solução com 80 indivíduos parece ser melhor.

I. População inicial com 50 indivíduos

Componente	Método
Dimensão	10
Representação	Real
População	Aleatória Uniforme
Tamanho da População	50
Recombinação	Recombinação Aritmética Única
Probabilidade de Recombinação (p_c)	0.85
Mutação	Gaussiana
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Torneio)
Numero de Pais	15
Seleção de Sobreviventes	Baseada em Aptidão
Critério de Parada	$10000 \times (Dimensão)$

TABLE XXI: O Algoritmo genético com população inicial de 50.

Função	Melhor	Pior	Média	V.Esperado
f_1	1.67×10^8	1.06×10^{10}	1.28×10^9	100
f_2	2.91×10^7	1.49×10^{10}	1.72×10^9	200
f_6	604.03133	608.14811	605.04998	600
f_7	701.09933	734.93521	705.51736	700
f_9	928.86546	977.50029	939.69816	900
f_{14}	1400.53135	1410.11905	1401.88398	1400

TABLE XXII: Estatísticas do GA para 10 dimensões e parâmetros da tabela XIX

Não foi possível encontrar mínimos melhores que a execução anterior.

V. CONCLUSÕES

Após uma série de testes, a configuração de execução, tanto para 10 como para 30 dimensões, que obteve os melhores resultados foi a da tabela XXIII:

Componente	Método
Representação	Real
População	Aleatória Uniforme
Tamanho da População	80
Recombinação	Recombinação Aritmética Única
Probabilidade de Recombinação (p_c)	0.85
Mutação	Gaussiana
Probabilidade de Mutação (p_m)	0.01
Seleção de Pais	Proporcional à aptidão (Torneio)
Numero de Pais	15
Seleção de Sobreviventes	Baseada em Aptidão
Critério de Parada	$10000 \times (Dimensão)$

TABLE XXIII: Algoritmo final utilizado.

Os valores encontrados na melhor utilização para 10 dimensões estão na tabela XX e para 30 dimensões na tabela XXIV

Função	Melhor	Pior	Média	V.Esperado
f_1	1.02×10^8	4.59×10^9	4.25×10^8	100
f_2	2.48×10^8	3.10×10^{10}	3.83×10^9	200
f_6	617.65754	624.45554	619.16483	600
f_7	701.09929	736.59681	704.51488	700
f_9	1085.23669	1179.87806	1108.33023	900
f_{14}	1400.50696	1421.55140	1403.80860	1400

TABLE XXIV: Estatísticas do GA para 30 dimensões e parâmetros da tabela XXIII

Função = Min	Melhor D10	Melhor D30	Err. Rel D10	Err. Rel D30
$f_1 = 100$	2.10×10^8	1.02×10^8	-/-	-/-
$f_2 = 200$	2.16×10^7	2.48×10^8	-/-	-/-
$f_6 = 600$	603.74999	617.65754	0.62%	2.94%
$f_7 = 700$	701.06073	701.09929	0.15%	0.16%
$f_9 = 900$	923.94480	1085.23669	2.66%	20.58%
$f_{14} = 1400$	1400.51077	1400.50696	0.04%	0.04%

TABLE XXV: Comparação XIX

As duas primeiras funções não chegaram nem próximas do valor esperado. Já as outras, embora possua um erro maior do que o solicitado no enunciado do trabalho, tiveram um erro relativo baixo em grande parte dos casos.

REFERENCES

- [1] Q. B. Y. Liang, J. J. and P. N. Suganthan, "Problem definitions and evaluation criteria for the cec 2014 special session and competition on single objective realparameter numerical optimization," Tech. Rep., 2013.
- [2] J. H. Holland, *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press, 1975. [Online]. Available: <http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0472084607>
- [3] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*. Springer, 2003.
- [4] J. J. Grefenstette, "Optimization of control parameters for genetic algorithms." *IEEE Trans. Systems, Man, and Cybernetics*, vol. 16, no. 1, pp. 122–128, 1986. [Online]. Available: <http://dblp.uni-trier.de/db/journals/tsmc/tsmc16.html#Grefenstette86>