

# Uma Nova Variante para Algoritmos Genéticos Baseada na Recombinação por Transformação de Bactérias

Adriana Medeiros Pinheiro

*Programa de Pós-Graduação em Engenharia Elétrica  
Universidade Federal do Pará (UFPA)*

Belém, Brasil

adrianamprf@gmail.com

George Tassiano Melo Pereira

*Programa de Pós-Graduação em Engenharia Elétrica  
Universidade Federal do Pará (UFPA)*

Belém, Brasil

george.melo7@gmail.com

Reginaldo Cordeiro dos Santos Filho

*Programa de Pós-Graduação em Engenharia Elétrica  
Universidade Federal do Pará (UFPA)*

Belém, Brasil

regicsf@ufpa.br

Claudio de Souza Sales Júnior

*Programa de Pós-Graduação em Engenharia Elétrica  
Universidade Federal do Pará (UFPA)*

Belém, Brasil

cssj@ufpa.br

**Resumo**—O Algoritmo Genético (AG) é um ramo dos algoritmos evolutivos que vem mostrando poder e sucesso como estratégia para otimizar diversos problemas, mas pode sofrer com a convergência lenta e prematura dos resultados, impactando na eficiência final obtida. Nessa perspectiva, variantes do AG estão sendo desenvolvidas, buscando obter maior convergência e precisão dos resultados. Este trabalho propõe o Algoritmo Genético de Recombinação por Transformação (RTGA) que traz um novo operador genético baseado na recombinação por transformação de bactérias. O mesmo foi comparado com outras variantes do AG na otimização de quatro funções de avaliação e obteve todos os ótimos globais.

**Palavras-chave**—computação evolutiva; algoritmo genético; problema de otimização; recombinação por transformação

## I. INTRODUÇÃO

A otimização é o processo de ajuste de características de um dado processo, matemático ou experimental, para se encontrar o valor máximo ou mínimo da função associada ao referido processo, que represente seu desempenho, obedecendo a todas as restrições impostas. Ela visa determinar a melhor configuração de projeto de um dado sistema sem ter que testar todas as possibilidades.

De acordo com [1] a otimização com o uso de Algoritmos Evolucionários (AE) vem sendo usada em vários campos da ciência com o auxílio de técnicas de computação aplicada e mais precisamente com técnicas de inteligência artificial. Nessa perspectiva, a existência de vários máximos locais e globais ou multimodalidade, é um problema clássico em que os algoritmos evolucionários podem ser empregados para a localização do ótimo global.

O Algoritmo Genético (AG) é uma ramificação dos AEs, e foi proposto por John Holland na Universidade de Michigan no início dos anos 60 [2]. Os AEs pertencem à uma classe de algoritmos de otimização inspirada no processo evolutivo

biológico. Eles recebem esse nome por se relacionarem intimamente como espécies e se adaptam ao ambiente ao longo de diversas gerações.

De acordo com [3], os algoritmos genéticos fazem parte de técnicas probabilísticas que tentam encontrar soluções em execuções diferentes usando os mesmos parâmetros, e as vezes usando a mesma população. Entre as principais vantagens do AG, se destacam: otimização de valores discretos e contínuos, pesquisa simultânea, possibilidade de trabalhar com muitas variáveis, e um conjunto de soluções candidatas. A figura 1 mostra as etapas básicas do AG.

Os AGs buscam ótimos locais a partir de uma população de indivíduos criada aleatoriamente e têm a forte capacidade de busca paralela, mas devido a todos os tipos de aleatoriedades encontradas em sua execução, problemas de resultados prematuros e a baixa velocidade de convergência podem afetar a habilidade da busca pelo ótimo global [5].

Este trabalho gerou uma nova variante do algoritmo genético baseado na recombinação por transformação de bactérias chamada *Recombination by Transformation Genetic Algorithm* (RTGA), em que causa o aumento da velocidade de convergência com alta precisão.

Foram realizados experimentos usando quatro funções multimodais em duas dimensões comparando sete variantes do algoritmo genético e o algoritmo genético clássico. Os resultados mostram que o algoritmo proposto se destacou entre o AG clássico e as sete variantes, alcançando todos os valores do ótimo global de todas as funções.

## II. TRABALHOS RELACIONADOS

Para o desenvolvimento deste trabalho, buscou-se artigos científicos que implementaram novos operadores genéticos que modificaram a estrutura do AG clássico. Os operadores

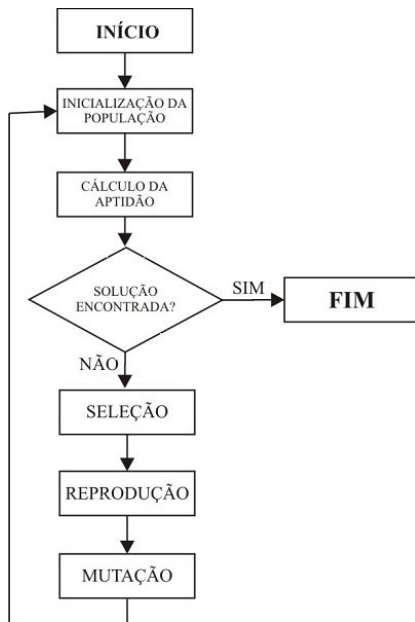


Fig. 1. Etapas do algoritmo genético. Fonte: [4]

genéticos selecionados para análises comparativas foram o *Retroviral Iterative Genetic Algorithm* (RIGA) [6], *Meiosis Genetic Algorithm* (MGA) [7], *Real-coded Quantum Genetic Algorithm* (RQGA) [8], *Chaos Genetic Algorithm* (CGA) [9], *Microbial Neuro Genetic Algorithm* (MNGA) [10], *Negative Selection Genetic Algorithm* (NSGA) [11], e *Immune Genetic Algorithm* (IGA) [5]. Estas variantes possuem em comum a finalidade de melhorar os resultados do AG clássico em termos de convergência e velocidade.

O RIGA [6] trouxe uma população auxiliar de retrovírus, no qual possui mecanismos próprios de criação e substituição de indivíduos que favoreceu o aumento da velocidade de convergência do AG clássico por trazer maior variabilidade genética ao longo das gerações. Apesar disso, observou-se o aumento do custo computacional devido a sua etapa adicional de criação de retrovírus. Através dessa observação possibilitou a criação do RTGA. No RIGA a comparação de seus resultados deu-se apenas entre ele e o AG clássico, não buscando comparar com outras técnicas similares, o que neste artigo veio à comparar.

### III. ALGORITMO GENÉTICO DE RECOMBINAÇÃO POR TRANSFORMAÇÃO

A recombinação por transformação de bactérias é o processo no qual o DNA livre no meio é tomado pela bactérias, resultando em alterações genotípicas da mesma [12]. O DNA livre capturada pela bactéria é proveniente de bactérias mortas, e não necessariamente da mesma espécie. Esse processo entretanto não ocorre sempre, pois fatores como composição do meio e estado fisiológico da bactéria são importantes para seu sucesso. A figura 2 mostra o fluxo deste processo de recombinação por transformação.

### Reprodução Bacteriana

#### Reprodução Sexuada

#### b) Transformação

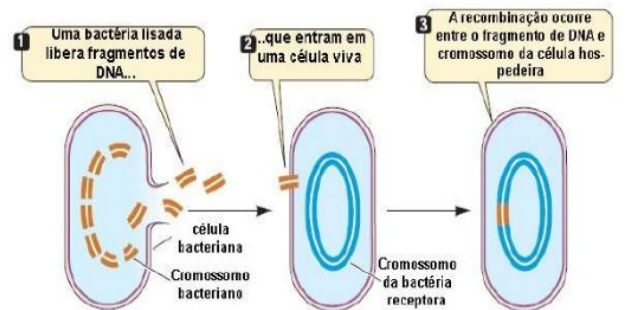


Fig. 2. Etapas da recombinação por transformação de bactérias. Fonte: [13]

Conforme a figura 2, a bactéria receptora em um meio contendo materiais genéticos dispersos captura um desses materiais e no final incorpora este fragmento em seu próprio material genético, e desta forma surge uma bactéria com um material genético diferente da original.

O RTGA proposto utiliza recombinação por transformação de bactéria, pois os indivíduos descartados (população morta chamada no algoritmo proposto) da população do AG clássico na fase de seleção dos sobreviventes representam o DNA livre. Para simular a barreira de fatores que determinam o acontecimento ou não da captura do fragmento disperso pela bactéria, fez-se o uso de uma etapa similar de uma infecção viral utilizado no RIGA, de [6].

A recombinação por transformação substitui a transcrição reversa na etapa de infecção viral de RIGA, pois realiza *crossovers* entre indivíduos mortos e um indivíduos da população. Como a função *fitness* não leva em consideração o valor de um alelo isolado na avaliação dos indivíduos, a pressão seletiva, principalmente em populações pequenas, pode resultar na perda de um alelo benéfico ou na fixação (aumento da frequência para 100%) de um alelo prejudicial em uma população, processo este conhecido como deriva genética. Portanto, a etapa de recombinação por transformação contribui como uma extensão da população e memória contra a deriva genética, pois concede novas chances para que os indivíduos mortos incluam seus alelos no decorrer das gerações.

O RTGA possui as mesmas etapas do AG, com a adição da etapa de recombinação por transformação após a seleção dos sobreviventes, como é possível ver na figura 3, onde mostra o diagrama do RTGA. Com isso, fica a critério na implementação a escolha de quais os tipos de seleção dos pais, sobreviventes, mutação e cruzamento que serão utilizados no AG. As etapas do RTGA são descritas a seguir:

- Após a etapa de seleção dos sobreviventes, os indivíduos

descartados são adicionados a uma população auxiliar de indivíduos mortos e que recebem um *fitness* de recombinação inicial igual a zero. Esse *fitness* definirá a quantidade de recombinações por transformação bem sucedidas realizadas, o que pode possuir um valor negativo;

- A recombinação por transformação acontece entre pares formados por um cromossomo morto e um cromossomo da população do AG escolhidos aleatoriamente. A quantidade de recombinações realizadas será conforme a taxa de recombinação definida. Definidos os pares a serem recombinados, aplica-se o cruzamento do AG para a geração de filhos;
- Os filhos desses cruzamentos são comparados com o seu pai proveniente da população do AG, e caso tenham melhores *fitness*, o melhor filho substitui este pai na população e é incrementado em uma unidade o *fitness* de recombinação do pai morto. Caso contrário, os filhos gerados são descartados e é decrementado em uma unidade o *fitness* de recombinação do pai morto;
- Em seguida, é realizada a seleção dos sobreviventes sobre os mortos baseado no *ranking* a partir do seu *fitness* de recombinação, no qual são eliminados todos os indivíduos mortos com menores *fitness* de recombinação até a taxa de elitismo definida;
- Se o critério de parada do AG não foi atingido, é prosseguido a execução da próxima geração.

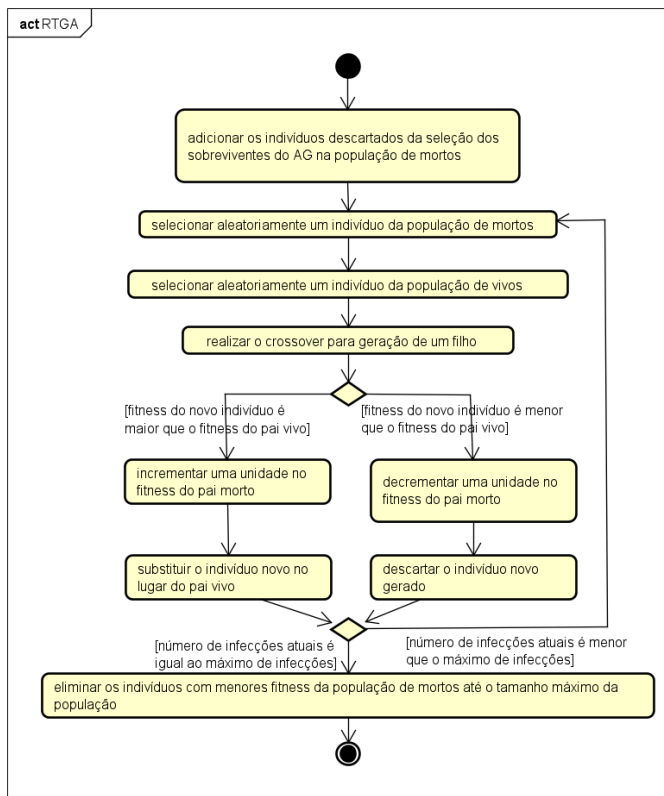


Fig. 3. Diagrama do RTGA

#### A. Parâmetros de controle do RTGA

O RTGA usa os mesmos parâmetros do AG clássico e adiciona novos para controlar o processo de recombinação por transformação. São eles:

- 1) *Taxa de recombinação por transformação na população*: a taxa de cromossomos que serão infectados na geração.
- 2) *Taxa de elitismo de mortos*: a taxa da população morta que será mantida para a próxima geração.
- 3) *Número de mortos*: o tamanho máximo na população de mortos.
- 4) *Recombinação por transformação mais fraca*: este parâmetro força a recombinação por transformação nos cromossomos com menores aptidões.
- 5) *Recombinação por transformação única*: este parâmetro força uma única recombinação por transformação por cromossomo.

O parâmetro da taxa de infecção na população irá causar uma lenta execução do algoritmo se o seu valor for maior que 50%. A definição do parâmetro da taxa de elitismo de mortos pode comprometer a evolução do algoritmo se o seu valor for maior que 50%, pois a população dos mortos não receberá novas informações genotípicas a serem recombinadas com a população do AG.

Os parâmetros de recombinação por transformação mais fraca e única quando usados causam uma rápida velocidade de execução do algoritmo, porém podem resultar no aumento do número de gerações necessárias para a convergência populacional, devido ao menor número de recombinações por transformação aplicadas.

#### IV. METODOLOGIA

Para realizar uma análise comparativa do novo algoritmo genético proposto neste trabalho com as outras variantes, foi realizada uma pesquisa em outros artigos que usam funções de validação. As funções Rastrigin (Figura 4), Griewank (Figura 5), Ackley (Figura 6) e Schaffer (Figura 7) foram as mais utilizadas pelos autores [14]–[18] e por isso foram as escolhidas para a avaliação das variantes do AG.

Na otimização matemática, a função Rastrigin é uma função não convexa usada como um problema de teste de desempenho para algoritmos de otimização; A função Griewank tem muitos mínimos locais difundidos; A função Ackley possui vários pontos de mínimos locais, porém a partir de sua borda até o seu centro acontece uma minimização do valor da função. O algoritmo genético não deve ficar preso a nenhum mínimo local e deve continuar prosseguindo pelo espaço de busca; e a função Schaffer, também multimodal, possui em grande parte uma superfície plana contendo o mesmo valor para a função, e próximo ao centro de sua superfície acontece o aumento do valor da função para logo depois uma queda abrupta para o mínimo global da função. Ela apresenta dois desafios para o algoritmo genético, como a exploração do espaço de busca até encontrar o centro da função, e vencer a barreira do aumento do valor da função nas proximidades do seu centro.

Para a experimentação, elas foram definidas em duas dimensões [8]. Além disso, cada variante foi executada um total

de 30 vezes para cada função de validação calculando-se a média dos melhores resultados.

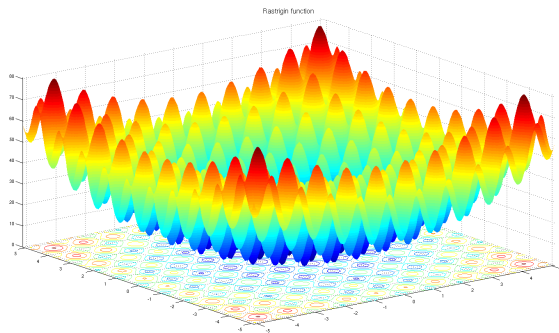


Fig. 4. Gráfico 3D da função Rastrigin. Fonte: [19]

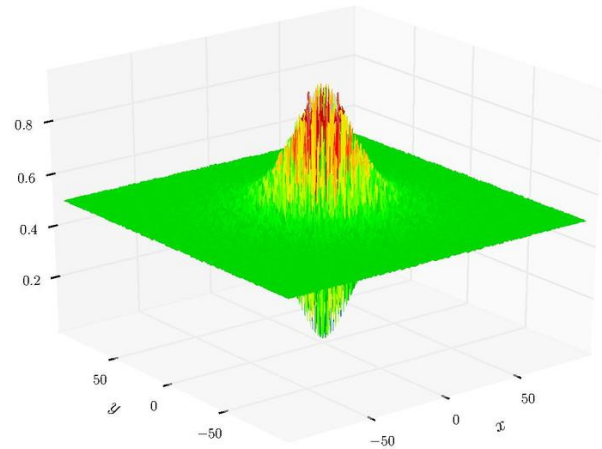


Fig. 7. Gráfico 3D da função Schaffer. Fonte: [22]

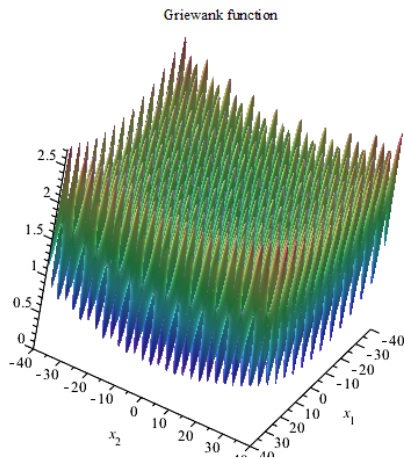


Fig. 5. Gráfico 3D da função Griewank. Fonte: [20]

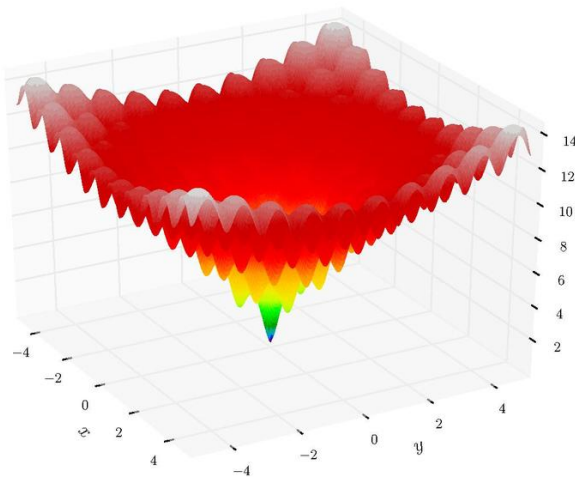


Fig. 6. Gráfico 3D da função Ackley. Fonte: [21]

Para realizar uma comparação justa das variantes foi utilizado como base a estrutura do AG com codificação real com dois genes, que representam os valores das variáveis  $x$  e  $y$  de cada função de validação. O AG forneceu, quando permitido pela variante, o uso de seus mecanismos definidos de seleção dos pais, cruzamento, mutação e seleção dos sobreviventes.

O AG clássico foi implementado com os seguintes mecanismos: seleção dos pais por torneio, cruzamento aritmético e mutação gaussiana adaptativa. A seleção dos pais por torneio deste algoritmo seleciona dois indivíduos aleatórios da população e escolhe o melhor entre eles para ser um dos pais.

O cruzamento aritmético é realizado por dois pais com a geração de dois filhos através de um alfa aleatório gerado por uma distribuição uniforme no intervalo  $[0,1]$  que determinará a maior parcela da contribuição do genótipo de um dos pais. A mutação gaussiana possui um desvio padrão adaptativo ( $dpa$ ) conforme o número de gerações dada pela Equação (1), onde  $\sigma^2$  é a variância do valor do gene a sofrer mutação.

$$dpa = \frac{\sigma^2}{10^{-3} \cdot (geracaoAtual - 1) + 1}. \quad (1)$$

A população inicial é gerada aleatoriamente, onde os valores dos genes estão contidos no intervalo entre o limite inferior e superior de cada função de validação. A seleção dos sobreviventes é feita pela substituição dos piores indivíduos da população pelos melhores filhos gerados de cada geração.

Os parâmetros do AG usados nos testes estão resumidos na Tabela I, enquanto que os valores dos parâmetros que são exclusivos das variantes RIGA e RTGA estão na Tabela II. Os valores estão conforme utilizado pelos autores [6]–[9], [17].

## V. RESULTADOS

Os resultados dos testes do RTGA e das outras variantes são encontrados na Tabela III em que mostra a média dos melhores resultados das funções de validação. Para facilitar a

TABELA I  
PARÂMETROS DO AG USADO COMO BASE.

Parâmetro	Valor
Número de indivíduos selecionados por torneio	2
Número de cruzamentos por geração	10
Probabilidade de cruzamento	90%
Probabilidade de mutação	5%
Tamanho da população	100
Valor da variância da mutação	9.0
Critério de parada	1000 gerações

TABELA II  
PARÂMETROS DO RIGA E RTGA.

Parâmetro	Valor
Taxa de infecção/recombinação	30
Taxa de elitismo	10%
Número de retrovirus/mortos	50
Infecção/recombinação mais fraca	Não
Infecção/recombinação única	Não
Taxa de infecção interna	alfa do cruzamento aritmético

leitura da tabela, foram destacados em negrito os resultados que possuem erros de aproximação menores que  $10^{-4}$ .

É possível observar que a única variante que encontrou todos os ótimos globais foi o RTGA, mostrando assim sua alta precisão de resultados. Desse modo, é correto afirmar que a variante (RTGA) foi mais eficiente para os problemas matemáticos expostos.

As funções MGA, IGA, MNGA e RIGA obtiveram desempenhos inferiores ao RTGA, com exceção para o MNGA que também encontrou o ótimo global da função Rastrigin. Já o AG e as demais variantes não alcançaram nenhuma aproximação maior que a desejada.

TABELA III  
MÉDIA DOS MELHORES RESULTADOS DAS FUNÇÕES DE VALIDAÇÃO

Função	Rastrigin	Griewank	Schaffer	Ackley
Ótimo	0	0	0	0
AG	7.5113E-1	3.1929E-2	4.84E-3	2.768E-3
MGA	<b>5.91E-4</b>	1.901E-3	<b>8.63E-10</b>	1.045E-3
MNGA	<b>0</b>	5.472E-3	2.2967E-3	<b>4.5048E-13</b>
NSGA	1.0355	4.3985E-2	3.7793E-2	5.174E-2
IGA	<b>1.7E-4</b>	1.2094E-2	<b>4.31E-6</b>	<b>7.88E-4</b>
RIGA	<b>7.8795E-6</b>	4.9154E-3	<b>4.17E-4</b>	<b>4.8365E-5</b>
RTGA	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
RQGA	2.5237E-1	2.1028E-1	9.427E-3	1.0022E-1
CGA	8.7814E-1	4.9257E-2	4.378E-3	4.429E-3

## VI. CONCLUSÃO

Este trabalho apresentou a criação de um algoritmo genético baseado na recombinação por transformação de bactérias. A aplicação biológica em conjunto com o algoritmo genético levou esse algoritmo a ser denominado Algoritmo Genético de Recombinação por Transformação (RTGA). Este artigo buscou trazer uma nova variante para o AG afim de obter melhores resultados nos experimentos realizados. O RTGA em comparação a outras variantes do AG, encontrou o ótimo global para todas as funções de testes em todas as suas execuções.

Houveram algumas limitações durante a realização deste trabalho, como a não realização da análise assintótica do algoritmo em comparação com o RIGA, além de aplicações de mais funções multimodais, ou seja, com diferentes dimensionalidades.

Como trabalhos futuros pretende-se avaliar outros aspectos entre os algoritmos abordados, tais como complexidade assintótica, resolução de problemas reais e de problemas de análises estatísticas. Além disso, será desenvolvido uma plataforma online para pesquisadores utilizarem esses novos algoritmos evolucionários.

## REFERÊNCIAS

- [1] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia Tools and Applications*, vol. 80, no. 5, pp. 8091–8126, feb 2021.
- [2] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press, 1975, second edition, 1992.
- [3] D. Costa Pereira, "OTIMIZAÇÃO DO CUSTO DE PRODUÇÃO DO CAFÉ UTILIZANDO ALGORITMO GENÉTICO," *RECIMA21 - Revista Científica Multidisciplinar - ISSN 2675-6218*, vol. 2, no. 1, pp. 252–266, feb 2021. [Online]. Available: <http://www.recima21.com.br/index.php/recima21/article/view/73>
- [4] L. Sorano, A. Pereira, N. Marranghello, A. Araujo, T. Catelam, J. Tavares, and A. Jr, "Detecção de bordas em imagens de ressonância magnética por meio de processamento de imagens com algoritmos genéticos," 11 2009.
- [5] S. Juan, S. Y. Dan, and X. min, "Research on application of iga (immune genetic algorithm) to the solution of course-timetabling problem." Proceedings of 2009 4th International Conference on Computer Science & Education, 2009, tradução nossa.
- [6] R. S. Moreira, O. N. Teixeira, and R. C. L. de Oliveira, "Mixing theory of retroviruses and genetic algorithm to build a new nature- inspired meta-heuristic for real-parameter function optimization problems." 10th International Conference on Hybrid Intelligent Systems, 2010, tradução nossa.
- [7] N. Wiriyasermkul and P. Chanvarasuth, "A meiosis genetic algorithm." Seventh International Conference on Information Technology, 2010, tradução nossa.
- [8] X. Xu, J. Jiang, J. Jie, H. Wang, and W. Wang, "An improved real coded quantum genetic algorithm and its applications," in *Proceedings - International Conference on Computational Aspects of Social Networks, CASoN'10*. IEEE, sep 2010, pp. 307–310. [Online]. Available: <http://ieeexplore.ieee.org/document/5636712/>
- [9] M. Javidi and R. Hosseinpoufard, "Chaos genetic algorithm instead genetic algorithm," *International Arab Journal of Information Technology*, vol. 12, no. 2, pp. 163–168, 2015. [Online]. Available: <http://ccis2k.org/iajit/PDF/vol.12,no.2/6039.pdf>
- [10] T. A. Tarique, M. A. Zamee, and M. I. Khan, "A new approach for pattern recognition with neuro-genetic system using microbial genetic algorithm." International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT), 2014, tradução nossa.
- [11] N.-N. LI, J.-H. GU, and B.-Y. LIU, "A new genetic algorithm based on negative selection." Proceedings of the Fifth International Conference on Machine Learning and Cybernetics, 2006, tradução nossa.
- [12] L. R. Trabulsi and F. Alterthum, in *Microbiologia*, 4th ed. Atheneu, 2005, ch. Genética Bacteriana, p. 45.
- [13] R. Físico, "Aula bacterias e doencas associadas," 2014. [Online]. Available: <https://pt.slideshare.net/aldenoramar90/aula-bacterias-edoencasassociadas>
- [14] M. Yaw, K. H. Chong, and K. Kamil, "Transform of Artificial Immune System algorithm optimization based on mathematical test function," in *2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*. IEEE, 2016, pp. 147–150. [Online]. Available: <http://ieeexplore.ieee.org/document/7893561/>
- [15] F. Li, L. Liu, and C. Jin, "A kind of composite genetic algorithm based on extreme pre-judgement," in *Proceedings of the 2007 11th International Conference on Computer Supported Cooperative Work in*

- Design, CSCWD*. IEEE, apr 2007, pp. 1038–1043. [Online]. Available: <http://ieeexplore.ieee.org/document/4281582/>
- [16] M. Ali, S. Koh, K. Chong, and D. Yap, “Hybrid Artificial Immune System-Genetic Algorithm optimization based on mathematical test functions,” *Research and Development (SCORED), 2010 IEEE Student Conference on*, no. SCORED, pp. 13–14, dec 2010. [Online]. Available: <http://ieeexplore.ieee.org/document/5704012/>
- [17] J. Fernandez-Villacanas and S. Amin, “Simulated jumping in genetic algorithms for a set of test functions,” *Proceedings Intelligent Information Systems. IIS'97*, pp. 209–215, 1997. [Online]. Available: <http://ieeexplore.ieee.org/document/645223/>
- [18] H. Kargupta, “The performance of the gene expression messy genetic algorithm on \nreal test functions,” *Proceedings of IEEE International Conference on Evolutionary Computation*, pp. 631–636, 1996. [Online]. Available: <http://ieeexplore.ieee.org/document/542674/>
- [19] Wikipedia, “Rastrigin function,” 2021. [Online]. Available: <https://en.wikipedia.org/wiki/Rastriginfunction>
- [20] —, “Griewank function,” 2021. [Online]. Available: <https://en.wikipedia.org/wiki/Griewankfunction>
- [21] —, “Ackley function,” 2020. [Online]. Available: <https://en.wikipedia.org/wiki/Ackleyfunction>
- [22] W. Commons, “Schaffer function,” 2020. [Online]. Available: <https://commons.wikimedia.org/wiki/File:Schafferfunction2.pdf>