

# Comparação Entre Abordagens Mono e Multiobjetivo dos Algoritmos Genéticos Simples e Compacto

BRENO M. F. VIANA, Universidade de São Paulo, Brasil

## 1 INTRODUÇÃO

Algoritmos evolutivos (EA, do inglês *Evolutionary Algorithm*) tradicionais são baseados em teorias da evolução [4], tais algoritmos possuem a seguinte estrutura: (1) geração da população inicial de indivíduos; (2) avaliação de cada indivíduo da população seguindo certos critérios; (3) seleção de indivíduos mais aptos (identificados através de uma função objetivo); (4) geração de novos indivíduos a partir dos selecionados. Contudo, existem abordagens evolutivas que utilizam modelos probabilísticos, que não mantêm armazenamento da população e, portanto, oferecem menor gasto de memória. Além disso, tais abordagens podem considerar correlações entre genes de um indivíduo normalmente não identificadas por sistemas evolutivos tradicionais. Tais abordagens são classificadas como Algoritmos de Estimação de Distribuição (EDA, do inglês *Estimation of Distribution Algorithm*).

Neste trabalho, apresentamos um estudo comparativo entre as abordagens mono e multiobjetivas para os algoritmos genéticos. O Algoritmo Genético Simples (sGA, do inglês *Simple Genetic Algorithm*) proposto por Sampson [3], inspirado pela Teoria da Evolução de Darwin [1], e o Algoritmo Genético Compacto (cGA, do inglês *Compact Genetic Algorithm*) proposto por Harik, Lobo e Goldberg [2], que utiliza um modelo probabilístico simples visando economia de memória. Em resumo, compararemos as seguintes abordagens: **sGA monobjetivo**, **cGA monobjetivo**, **sGA multibjetivo** e **cGA multibjetivo**.

Para realização dos testes utilizamos configurações de testes variando o problema, a quantidade de genes (bits) do problema e o tamanho máximo da população. Com isso, visamos analisar a convergência e o comportamento de cada abordagem. Para uma comparação justa, não utilizamos operadores genéticos para problemas específicos. Os mesmos operadores foram utilizados distinguindo apenas devido a natureza dos problemas binários e reais.

Na seção a seguir, Seção 2, descrevemos como os experimentos foram configurados neste trabalho bem como os problemas (Subseção 2.1) os quais foram atacados. Na Seção 3 apresentamos os resultados dos experimentos e, em seguida, analisamos os resultados (Seção 4). Por fim, apresentamos as conclusões deste trabalho na Seção 5.

## 2 MÉTODOS

Para analisar as versões mono e multiobjetiva do sGA e cGA, devemos realizar diversos testes com diferentes configurações. Com isso, podemos analisar o comportamento das abordagens, verificar a convergência das soluções, e identificar em quais cenários as abordagens se mostram mais vantajosas em relação à eficiência. Portanto, no nosso trabalho variamos três características: (1) o problema (os problemas são descritos na subseção 2.1); (2) a quantidade de bits do problema (10, 20, 40, 80 e 160); e (3) o tamanho máximo da população (10, 20, 40, 80 e 160). Fixamos a quantidade de gerações em 200 e realizamos 100 execuções para cada configuração de teste (totalizando 35.000 execuções).

Além disso, dividimos os problemas entre binários e reais. Entre eles, temos três problemas binários (2.1.1, 2.1.2 e 2.1.3) e dois problemas de números reais (2.1.4 e 2.1.5). Analisamos cada um destes problemas para as versões monobjetivas do sGA e cGA. Nas versões multiobjetivas, por sua vez, utilizamos pares de problemas para binários e para números reais. Para o multiobjetivo binário utilizamos os problemas 2.1.2 e 2.1.3 e para números reais 2.1.4 e 2.1.5.

Para resumir as citações sobre as configuração dos testes utilizamos o padrão  $X/Y$ , onde  $X$  é o tamanho do cromossomo e  $Y$  o tamanho da população. Por exemplo, 10/40 se refere a uma configuração com cromossomo de tamanho 10 e população de 40 indivíduos.

## 2.1 Problemas

Para elaborar os testes, escolhemos problemas baseados em funções matemáticas. Entre elas, temos três funções consideradas **funções armadilhas** (2.1.2, 2.1.3 e 2.1.5), funções que abruptamente mudam de comportamento devido à relação entre as variáveis (genes) das soluções (indivíduos). **Funções armadilhas** podem prender algoritmos de busca em soluções ótimas locais. Consideramos um erro de  $10^{-10}$  para o cálculo do *fitness* das funções 2.1.4 e 2.1.5, pois aplicamos essas funções no contexto de números reais.

**2.1.1  $f_{OneMax}$ .** O problema OneMax tem como objetivo maximizar a soma dos bits de um cromossomo. Por exemplo, em um cromossomo de 5 bits a solução [1 0 1 1 1], com *fitness* = 4, não é a ideal. A solução ótima (a melhor possível) para um cromossomo de 5 bits é [1 1 1 1 1], com *fitness* = 5. A Tabela 1 apresenta as soluções ótimas possíveis para o tamanho dos cromossomos considerados no nosso trabalho.

$$f_{OneMax}(x) = \sum_{i=1}^n x_i \quad (1)$$

**2.1.2  $f_{trap5}$ .** O problema Trap5 também tem o objetivo de maximizar a soma dos bits de um cromossomo. Contudo, a função trap5 possui uma característica que dificulta a busca pela solução ótima (também [1 1 1 1 1], com *fitness* = 5, para um cromossomo de 5 bits). Tal característica é que a soma dos bits é subtraído de 4, que faz com que, quanto maior a soma menor o *fitness*. Contudo, a exceção é a soma igual à 5, que possui *fitness* = 5. Tal característica engana os algoritmos de forma a acharem que a quantidade maior de 0s no cromossomo seja melhor. A Tabela 1 apresenta as soluções ótimas possíveis para o tamanho dos cromossomos considerados no nosso trabalho.

$$f_{trap5}(x) = \begin{cases} 4 - x & \text{se } x < 5, \\ 5 & \text{caso contrário.} \end{cases} \quad (2)$$

A solução ótima local para este problema pode ser calculada com a seguinte função:

$$tamanho(individuo) - tamanho(individuo)/5, \text{ desde que: } tamanho(individuo)\%5 = 0 \quad (3)$$

Considerando um cromossomo de 5 bits, qualquer configuração onde apresenta apenas um bit com valor 1 é um solução ótima local.

**2.1.3  $f_{inotrap5}$ .** O problema InvTrap5 é análogo ao trap5, porém, possui o objetivo de minimizar a soma dos bits de um cromossomo. A solução ótima é [0 0 0 0 0], com *fitness* = 5, para um cromossomo de 5 bits. A Tabela 1 apresenta as soluções ótimas possíveis para o tamanho dos cromossomos considerados no nosso trabalho.

$$f_{inotrap5}(x) = \begin{cases} x - 1 & \text{se } x > 0, \\ 5 & \text{caso contrário.} \end{cases} \quad (4)$$

A solução ótima local para este problema também pode ser calculada com a Equação 3. Considerando um cromossomo de 5 bits, qualquer configuração onde apresenta apenas um bit com valor 0 é um solução ótima local.

Tabela 1. Soluções possíveis dos problemas 2.1.1, 2.1.2 e 2.1.3 para o tamanho do problema dos testes

Soluções					
tamanho	10	20	40	80	160
<i>fitness</i>	10	20	40	80	160

2.1.4  $f_{sphere}$ . O problema Sphere tem como objetivo minimizar o *fitness* da função  $f_{sphere}$ . A solução ótima para um cromossomo de 5 genes é  $[0\ 0\ 0\ 0\ 0]$ , com *fitness* = 0. Portanto, a solução ótima é composta apenas por genes 0 e possui sempre *fitness* = 0.

$$f_{sphere}(x) = \sum_{i=1}^n x_i^2 \quad (5)$$

2.1.5  $f_{rosen}$ . O problema Rosen tem como objetivo minimizar o *fitness* da função  $f_{rosen}$ . A solução ótima para um cromossomo de 5 genes é  $[1\ 1\ 1\ 1\ 1]$ , com *fitness* = 0. Portanto, a solução ótima é composta apenas por genes 1 e possui sempre *fitness* = 0. Por utilizarmos esta função no contexto de números reais, existem infinitas possíveis soluções ótimas locais, nos testes identificamos que tais soluções se aproximam do *fitness* = 4 para um cromossomo de 5, e que o valor .

$$f_{rosen}(x) = \sum_{i=1}^{n-1} (100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2) \quad (6)$$

### 3 RESULTADOS

Nesta seção, apresentamos uma visão geral dos principais resultados obtidos no experimento. Primeiramente, apresentamos os resultados dos algoritmos monobjetivos. Na subseção seguinte, apresentamos os resultados dos algoritmos multiobjetivos através de gráficos de hipervolume. Por fim, apresentamos os mapas de calor das funções armadilhas para analisar a natureza multivariada destas funções. O conjunto completo de resultados está armazenado no Google Drive e pode ser acessado através do link a seguir: <https://shorturl.at/fksZ6>.

#### 3.1 Resultados dos Algoritmos Monobjetivos

Nesta seção apresentamos os principais resultados das versões monobjetivas do **sGA** e **cGA** aplicados individualmente para os problemas apresentados na Subseção 2.1.

3.1.1  $f_{onemax}$ . A Tabela 2 apresenta a taxa de sucesso das execuções do **sGA** e do **cGA** para o problema OneMax. Observando a Tabela 2a, podemos perceber que o **sGA** apresenta bons resultados para o problema OneMax, pois consegue encontrar a solução ótima na maioria das execuções. Apenas em três casos o algoritmo não foi capaz de encontrar, em nenhuma das execuções, a solução ótima: 80/10, 160/10 e 160/20. Além disso, o **sGA** apresenta um crescimento, ou manutenção, da taxa de sucesso com o aumento da população.

C	P	10	20	40	80	160
10		100%	100%	100%	100%	100%
20		99%	100%	100%	100%	100%
40		49%	100%	100%	100%	100%
80		0%	100%	100%	100%	100%
160		0%	0%	97%	100%	100%

(a) sGA

C	P	10	20	40	80	160
10		92%	100%	100%	100%	99%
20		49%	99%	100%	32%	2%
40		4%	77%	9%	0%	0%
80		0%	1%	0%	0%	0%
160		0%	0%	0%	0%	0%

(b) cGA

Tabela 2. Taxa de sucesso da  $f_{onemax}$  (C: cromossomo | P: população)

O **cGA**, por sua vez, apresenta resultados variados (apresentados na Tabela 2b). Podemos perceber que a taxa de sucesso (a porcentagem de vezes em que o algoritmo encontrou a solução ótima) decresce com o crescimento do tamanho do cromossomo e da quantidade de indivíduos da população. Encontramos bons resultados para os

cromossomos de tamanhos 10 e 20 que apresentaram 100% de taxa de sucesso pelo menos uma vez. O cromossomo de tamanho 40 apresentou um resultado promissor apenas para a população de 20 indivíduos, com 77% de taxa de sucesso. Apenas 1% das execuções de configuração 80/20 apresentaram a solução ótima para o problema OneMax. Nenhuma execução de qualquer tamanho de população do cromossomo 160 encontrou a solução ótima.

A Figura 1 apresenta um gráfico do valor de *fitness* das soluções encontradas no **sGA** e no **cGA** para a configuração 40/40. Podemos perceber que, apesar da taxa de sucesso não ser grande, o algoritmo se aproxima da solução ótima. A Figura 2, por sua vez, apresenta um gráfico do valor de *fitness* das soluções encontradas no **sGA** e no **cGA** para a configuração 40/160. A partir da Figura 2, podemos perceber que o **cGA** se distanciou bastante da solução ótima e não obteve nenhuma taxa de sucesso.

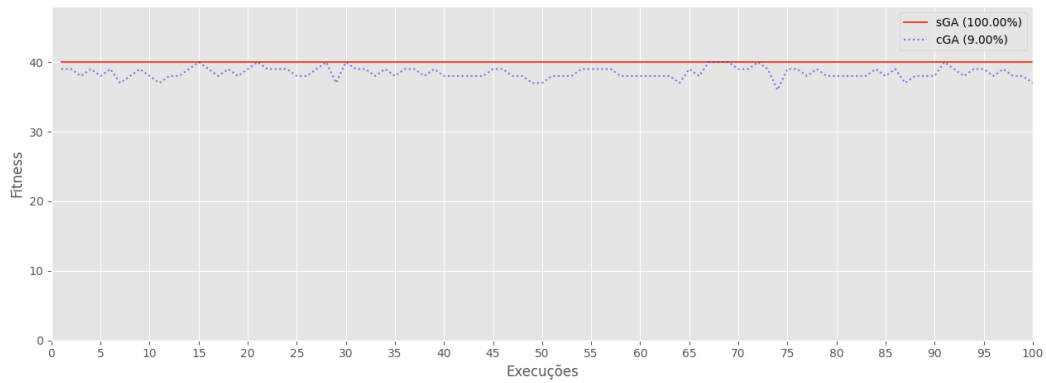


Fig. 1. Resultados da *fonemax* com configuração 40/40

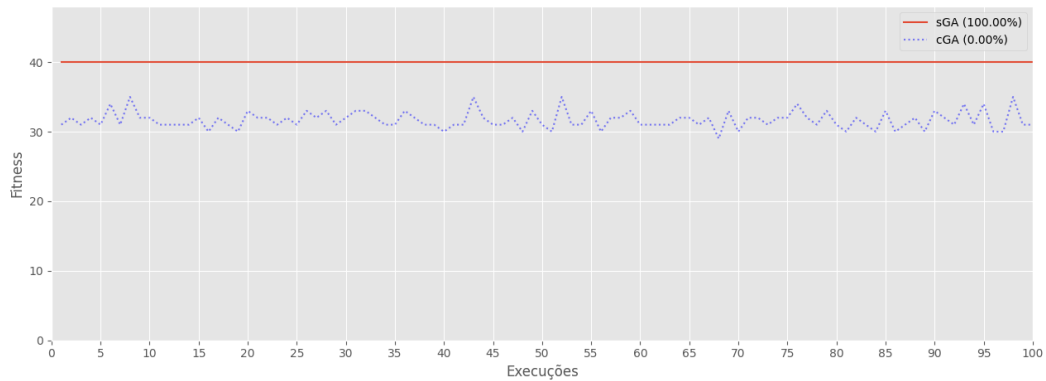


Fig. 2. Resultados da *fonemax* com configuração 40/160

**3.1.2  $f_{trap5}$ .** A Tabela 3 apresenta a taxa de sucesso das execuções do **sGA** e do **cGA** para o problema Trap5. Ao observar a Tabela 3a, percebemos que o **sGA** apresenta uma dificuldade de encontrar a solução ótima para populações pequenas. E a medida que a população aumenta a taxa de sucesso também aumenta. Apesar disso, nenhuma configuração atingiu a solução ótima. Contudo, os resultados a partir do cromossomo 40 apresentaram taxa de sucesso nula nas populações de todos os tamanhos. Ao observar a Tabela 3b observamos que o algoritmo encontra a solução ótima apenas para cromossomos de tamanho 10 e ainda apresenta taxas baixas de sucesso.

C	P	10	20	40	80	160
10		11%	10%	33%	76%	90%
20		0%	0%	3%	9%	33%
40		0%	0%	0%	0%	0%
80		0%	0%	0%	0%	0%
160		0%	0%	0%	0%	0%

(a) sGA

C	P	10	20	40	80	160
10		1%	1%	3%	10%	7%
20		0%	0%	0%	0%	0%
40		0%	0%	0%	0%	0%
80		0%	0%	0%	0%	0%
160		0%	0%	0%	0%	0%

(b) cGA

Tabela 3. Taxa de sucesso da  $f_{trap5}$  (C: cromossomo | P: população)

As figuras 3 e 4 apresentam gráficos do valor de *fitness* das soluções encontradas no **sGA** e no **cGA** para as configurações 40/40 e 40/160, respectivamente, do problema Trap5. Ao observá-las, percebemos que o **sGA** apresenta, em média, resultados próximos à solução ótima e que o aumento da população aumenta a proximidade das soluções com a solução ótima. Uma vez que a solução ótima não é alcançada mesmo com o aumento da população, podemos deduzir que as soluções encontradas são soluções ótimas locais. Percebemos também que o **cGA** com população 160 apresenta um maior distancimento da solução ótima que o **cGA** com população 40.

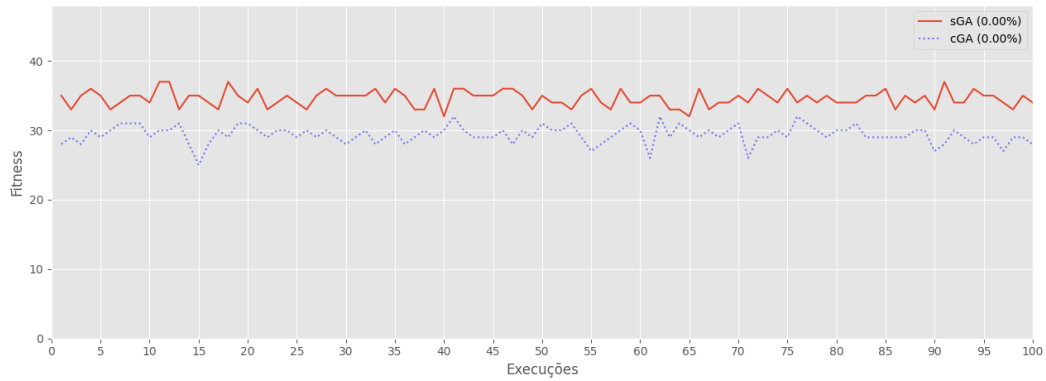


Fig. 3. Resultados da  $f_{trap5}$  com configuração 40/40

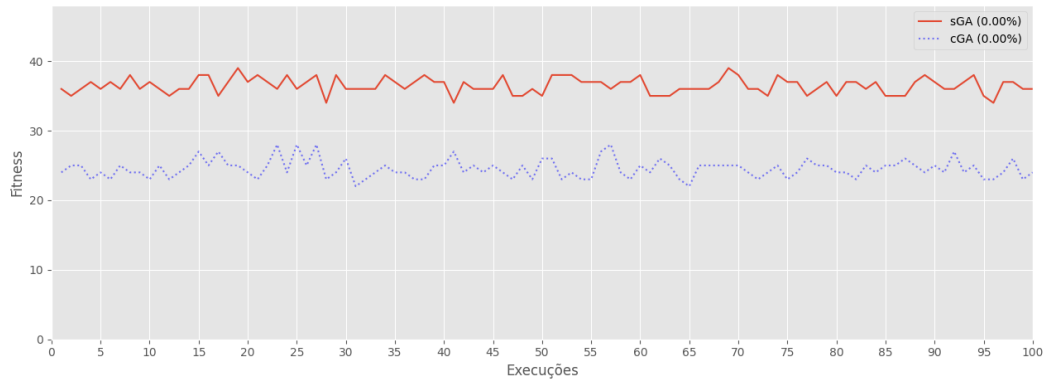


Fig. 4. Resultados da  $f_{trap5}$  com configuração 40/160

3.1.3  $f_{inotrap5}$ . A Tabela 4 apresenta a taxa de sucesso das execuções do **sGA** e do **cGA** para o problema InvTrap5. Os resultados obtidos são análogos aos resultados do problema Trap5. A única diferença mais relevante podemos identificar ao comparar as tabelas 4b e 4b. O **cGA** para o InvTrap5 apresenta um aumento da taxa de sucesso com o aumento da população, considerando apenas o cromossomo de tamanho 160. Ao contrário do **cGA** para o Trap5 que apresenta uma maior taxa de sucesso com o cromossomo de tamanho 80 e uma diminuição com o cromossomo de tamanho 160.

C	P	10	20	40	80	160
<b>10</b>		13%	12%	30%	71%	90%
<b>20</b>		0%	0%	3%	11%	33%
<b>40</b>		0%	0%	0%	0%	0%
<b>80</b>		0%	0%	0%	0%	0%
<b>160</b>		0%	0%	0%	0%	0%

(a) sGA

C	P	10	20	40	80	160
<b>10</b>		2%	1%	2%	5%	13%
<b>20</b>		0%	0%	0%	0%	0%
<b>40</b>		0%	0%	0%	0%	0%
<b>80</b>		0%	0%	0%	0%	0%
<b>160</b>		0%	0%	0%	0%	0%

(b) cGA

Tabela 4. Taxa de sucesso da  $f_{inotrap5}$  (C: cromossomo | P: população)

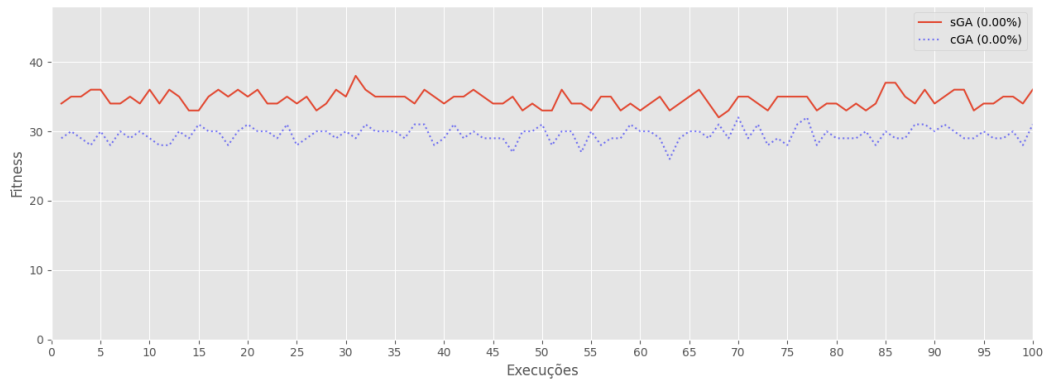


Fig. 5. Resultados da  $f_{inotrap5}$  com configuração 40/40

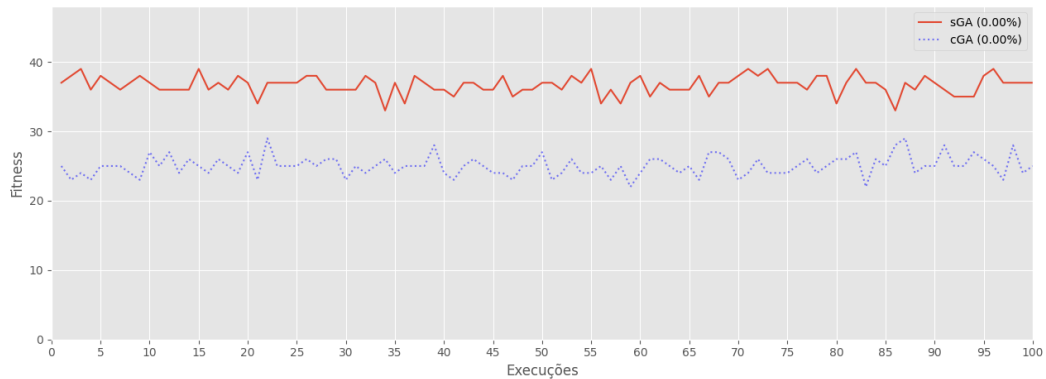


Fig. 6. Resultados da  $f_{inotrap5}$  com configuração 40/160

As figuras 5 e 6 apresentam gráficos do valor de *fitness* das soluções encontradas no **sGA** e no **cGA** para as configurações 40/40 e 40/160, respectivamente, do problema InvTrap5. Ao observá-las percebemos que o **sGA** apresenta o mesmo comportamento do problema anterior, o que indica que as soluções encontradas são soluções ótimas locais. Adicionalmente, identificamos que o **cGA** com população 160 apresenta um maior distancimento da solução ótima que o **cGA** com população 40.

3.1.4  $f_{sphere}$ . A Tabela 5 apresenta a taxa de sucesso das execuções do **sGA** e do **cGA** para o problema Sphere. Observando a Tabela 5a, podemos perceber que o **sGA** apresenta 100% de taxa de sucesso para apenas quatro configurações de testes: 10/40, 10/80, 10/160 e 20/160. Além disso, o **sGA** apresenta um crescimento brusco da taxa de sucesso de 0% à 100% das configurações 10/20 para 10/40 e da 20/80 para 20/160.

C	P	10	20	40	80	160
10		0%	0%	100%	100%	100%
20		0%	0%	0%	0%	100%
40		0%	0%	0%	0%	0%
80		0%	0%	0%	0%	0%
160		0%	0%	0%	0%	0%

(a) sGA

C	P	10	20	40	80	160
10		0%	2%	71%	100%	100%
20		0%	0%	2%	69%	99%
40		0%	0%	0%	2%	51%
80		0%	0%	0%	0%	0%
160		0%	0%	0%	0%	0%

(b) cGA

Tabela 5. Taxa de sucesso da  $f_{sphere}$  (C: cromossomo | P: população)

Podemos observar na Tabela 2b que o **cGA** apresenta uma maior taxa de sucesso maiores em mais configurações do que o **sGA**. O **cGA** perde em eficiência para o **sGA** apenas para as configurações 10/40 e 20/160. Além disso, o **cGA** apresenta um crescimento, ou manutenção, da taxa de sucesso com o aumento da população. Contudo, ao observar as configurações 10/20, 10/40, 20/40, 20/80, 40/80 e 40/160, percebemos que o crescimento da taxa de sucesso aparenta se tornar lento a medida que aumentamos o tamanho dos cromossomos.

As figuras 7 e 8 apresentam gráficos do valor de *fitness* das soluções encontradas no **sGA** e no **cGA** para as configurações 40/40 e 40/160, respectivamente, do problema Sphere. Ao observar a Figura 7 podemos observar que tanto o comportamento do **sGA** quanto o do **cGA** são irregulares. Contudo, boa parte das soluções tendem à solução ótima global. Ao observar a Figura 8 percebemos que o **sGA** conseguiu apresentar soluções muito próximas à solução global em todas as execuções, apesar de não encontrar a solução ótima nenhuma vez. O **cGA**,



Fig. 7. Resultados da  $f_{sphere}$  com configuração 40/40

por sua vez, apresentou algumas soluções mais distantes da solução ótima. Contudo, 51% das execuções o **cGA** conseguiu encontrar a solução ótima.

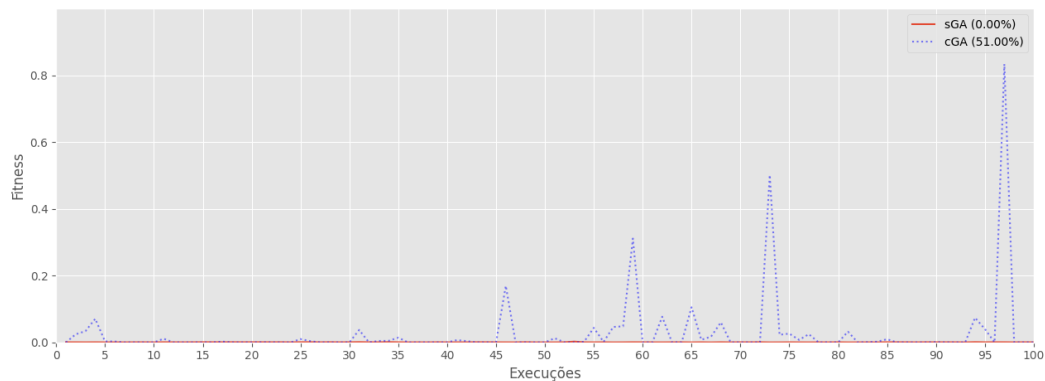


Fig. 8. Resultados da  $f_{sphere}$  com configuração 40/160

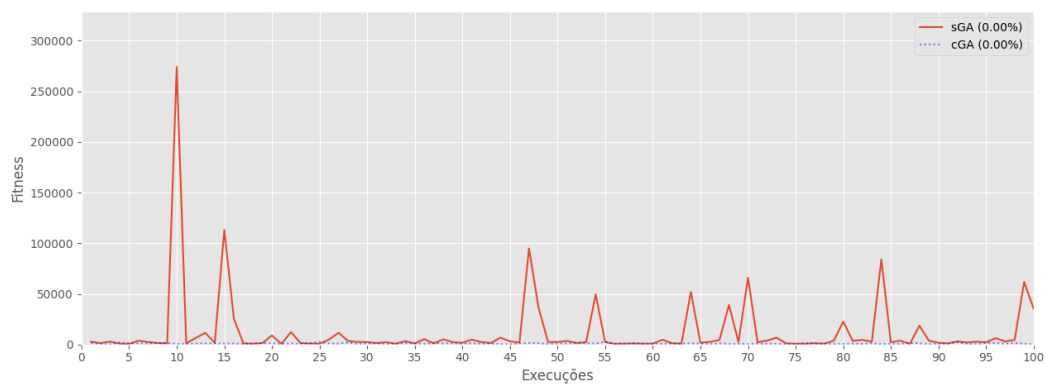


Fig. 9. Resultados da  $f_{rosen}$  com configuração 40/40

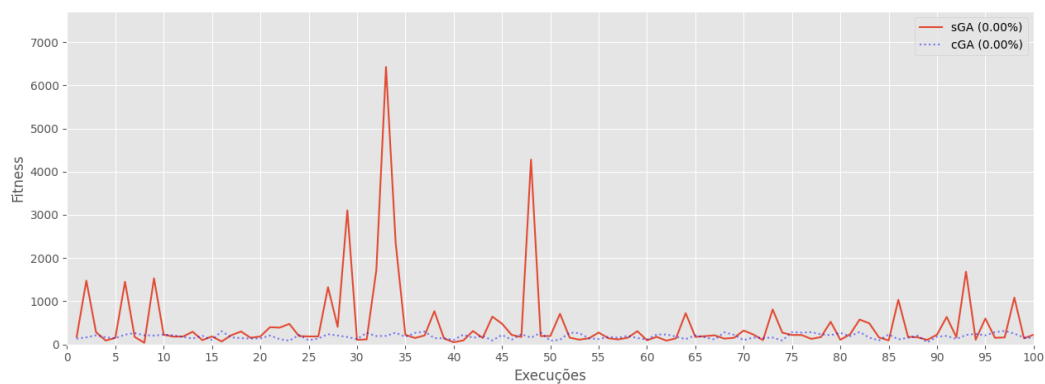


Fig. 10. Resultados da  $f_{rosen}$  com configuração 40/160



3.1.5  $f_{rosen}$ . A Tabela 6 apresenta a taxa de sucesso das execuções do **sGA** e do **cGA** para o problema Rosen. Observando a Tabela 6, podemos perceber que tanto o **sGA** quanto o **cGA** apresentam 0% de taxa de sucesso em todas as configurações de testes. Não podemos analisar detalhe algum observando apenas para a taxa de sucesso do algoritmos para esse problema. Portanto, temos que observar apenas as figuras 9 e 10.

C	P	10	20	40	80	160
10		0%	0%	0%	0%	0%
20		0%	0%	0%	0%	0%
40		0%	0%	0%	0%	0%
80		0%	0%	0%	0%	0%
160		0%	0%	0%	0%	0%

(a) sGA

C	P	10	20	40	80	160
10		0%	0%	0%	0%	0%
20		0%	0%	0%	0%	0%
40		0%	0%	0%	0%	0%
80		0%	0%	0%	0%	0%
160		0%	0%	0%	0%	0%

(b) cGA

Tabela 6. Taxa de sucesso da  $f_{rosen}$  (C: cromossomo | P: população)

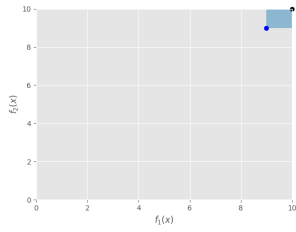
As figuras 9 e 10 apresentam gráficos do valor de *fitness* das soluções encontradas no **sGA** e no **cGA** para as configurações 40/40 e 40/160, respectivamente, do problema Rosen. Ao analisar as figuras 9 e 10 observamos os valores de *fitness* são muito altos, o que é ruim para um problema de minimização. O aumento da população apresenta uma melhora dos resultados, mas ainda longe da solução ótima. O **cGA** apresenta, em média, soluções mais próximas à 0 (*fitness* da solução ótima do problema). Além disso, o valor de *fitness* das soluções são próximos entre si, o que pode indicar a convergência para soluções ótimas locais. Caso seja interesse do leitor, a Tabela 7, que se encontra no Apêndice A, apresenta os valores de *fitness* de cada execução da configuração 40/160.

## 3.2 Resultados dos Algoritmos Multiobjetivos

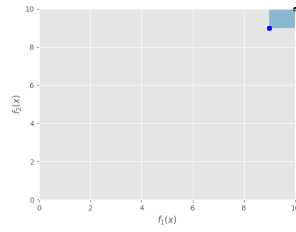
Nesta seção apresentamos os **melhores** resultados encontrados das versões multiobjetivas do **sGA** e **cGA** aplicados em pares de problemas. Para acessar os demais resultados basta acessar o link fornecido no início desta seção. No nosso experimento, utilizamos os problemas Trap5 e InvTrap5 para a versão binária utilizamos e para a versão real utilizamos os problemas Sphere e Rosen.

3.2.1  $sGA f_{trap5}$  e  $f_{invtrap5}$ . As figuras 11, 12 e 13 apresentam os gráficos de hipervolume das melhores soluções não-dominadas encontradas pelo **sGA** para os problemas Trap5 e InvTrap5 para os cromossomos de tamanho 10, 40 e 160, respectivamente. Podemos observar que as soluções não-dominadas encontradas são soluções próximas à solução utópica (dada a natureza das funções, uma solução com esse valor de *fitness* não pode ser alcançado). Ao observar apenas as figuras com população de 10 indivíduos, percebemos que as soluções não dominadas ficam mais distantes da solução utópica. Contudo, podemos observar também que, ao aumentar a população, soluções não-dominadas melhores são encontradas.

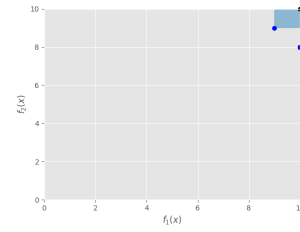
3.2.2  $cGA f_{trap5}$  e  $f_{invtrap5}$ . As figuras 14, 15 e 16 apresentam os gráficos de hipervolume das melhores soluções não-dominadas encontradas pelo **cGA** para os problemas Trap5 e InvTrap5 para os cromossomos de tamanho 10, 40 e 160, respectivamente. Podemos observar que o **cGA** encontrou soluções não-dominadas próximas à solução utópica apenas para o cromossomo de tamanho 10. Contudo, ao contrário do **sGA**, o **cGA** apresenta soluções não-dominadas mais distantes da solução utópica a medida que o tamanho do cromossomo e o tamanho da população aumentam.



(a) População de tamanho 10

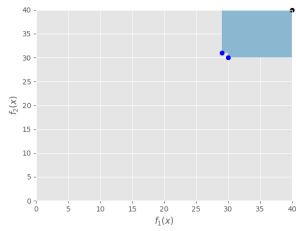


(b) População de tamanho 40

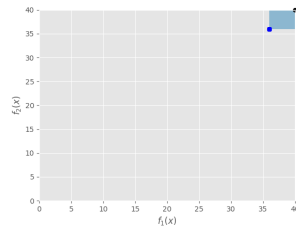


(c) População de tamanho 160

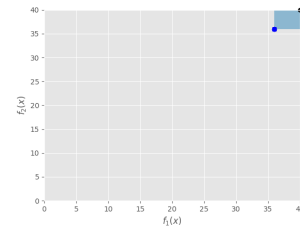
Fig. 11. Hipervolume do sGA do multiobjetivo  $f_{trap5}$  ( $f_1$ ) e  $f_{inotrap5}$  ( $f_2$ ) com cromossomo de tamanho 10



(a) População de tamanho 10

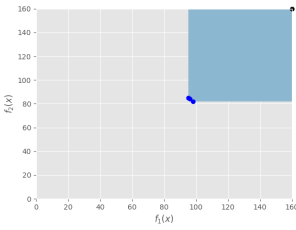


(b) População de tamanho 40

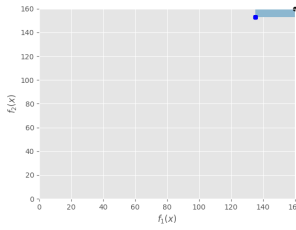


(c) População de tamanho 160

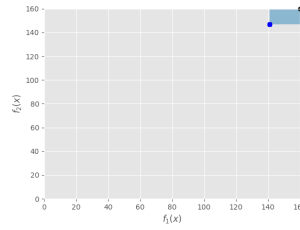
Fig. 12. Hipervolume do sGA do multiobjetivo  $f_{trap5}$  ( $f_1$ ) e  $f_{inotrap5}$  ( $f_2$ ) com cromossomo de tamanho 40



(a) População de tamanho 10

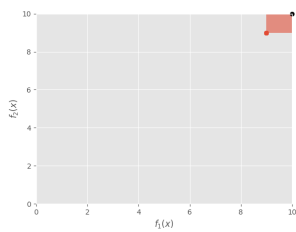


(b) População de tamanho 40

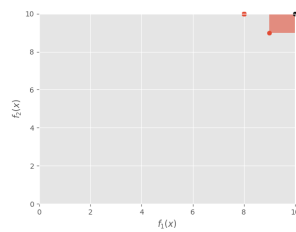


(c) População de tamanho 160

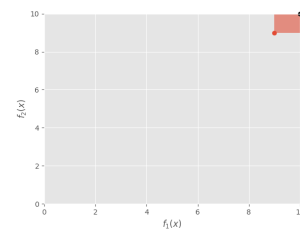
Fig. 13. Hipervolume do sGA do multiobjetivo  $f_{trap5}$  ( $f_1$ ) e  $f_{inotrap5}$  ( $f_2$ ) com cromossomo de tamanho 160



(a) População de tamanho 10



(b) População de tamanho 40



(c) População de tamanho 160

Fig. 14. Hipervolume do cGA do multiobjetivo  $f_{trap5}$  ( $f_1$ ) e  $f_{inotrap5}$  ( $f_2$ ) com cromossomo de tamanho 10

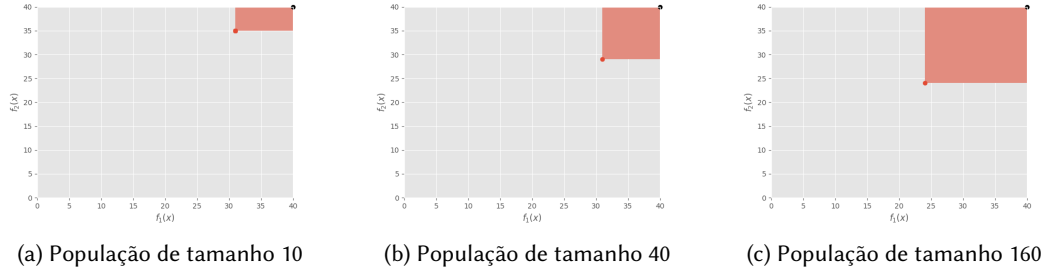


Fig. 15. Hipervolume do cGA do multiobjetivo  $f_{trap5}$  ( $f_1$ ) e  $f_{invtrap5}$  ( $f_2$ ) com cromossomo de tamanho 40

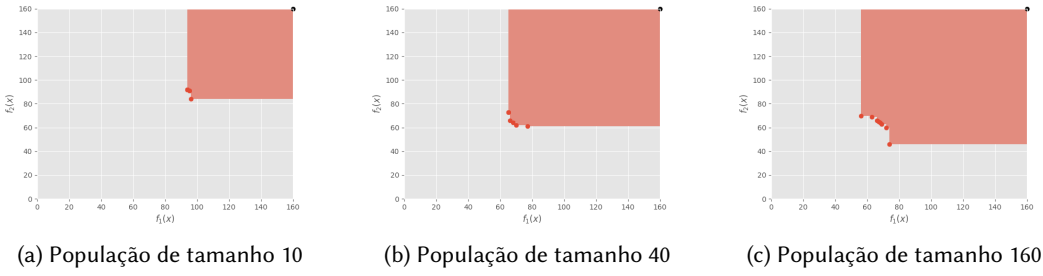


Fig. 16. Hipervolume do cGA do multiobjetivo  $f_{trap5}$  ( $f_1$ ) e  $f_{invtrap5}$  ( $f_2$ ) com cromossomo de tamanho 160

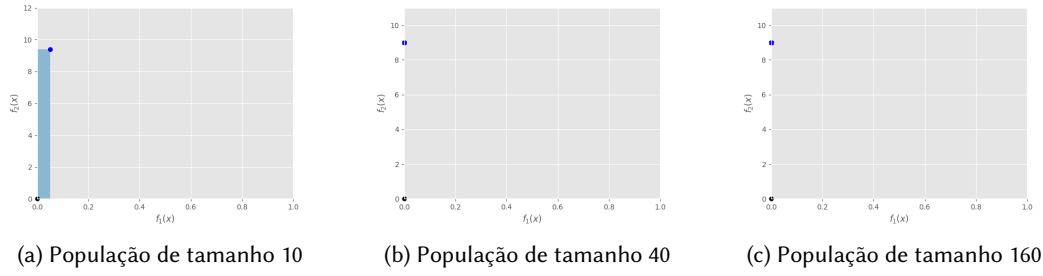


Fig. 17. Hipervolume do sGA do multiobjetivo  $f_{sphere}$  ( $f_1$ ) e  $f_{fosen}$  ( $f_2$ ) com cromossomo de tamanho 10

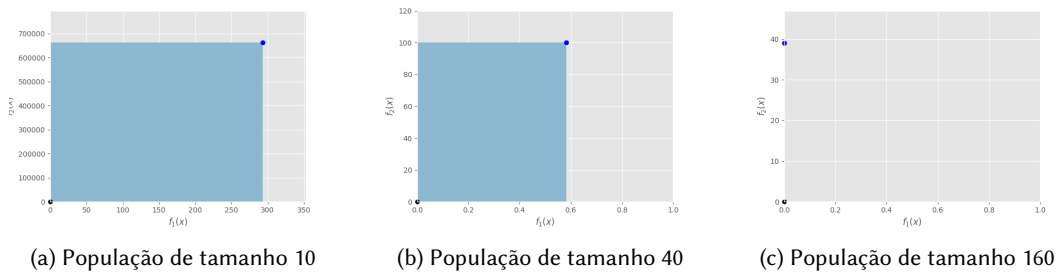
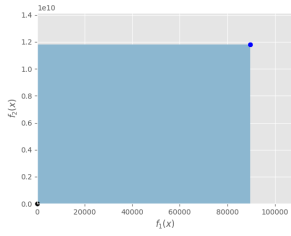
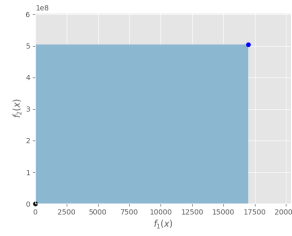


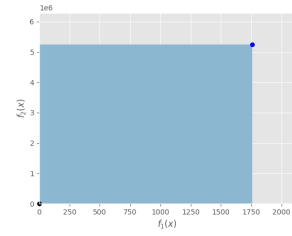
Fig. 18. Hipervolume do sGA do multiobjetivo  $f_{sphere}$  ( $f_1$ ) e  $f_{fosen}$  ( $f_2$ ) com cromossomo de tamanho 40



(a) População de tamanho 10

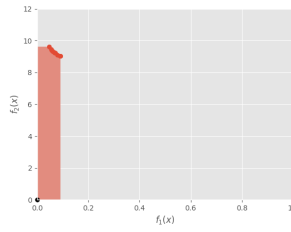


(b) População de tamanho 40

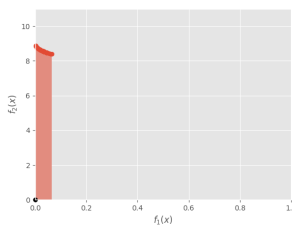


(c) População de tamanho 160

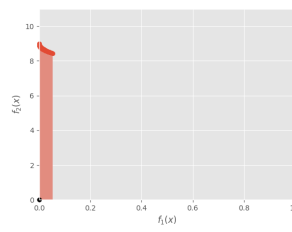
Fig. 19. Hipervolume do sGA do multiobjetivo  $f_{sphere}$  ( $f_1$ ) e  $f_{rosen}$  ( $f_2$ ) com cromossomo de tamanho 160



(a) População de tamanho 10

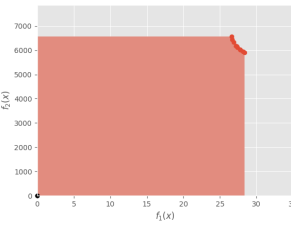


(b) População de tamanho 40

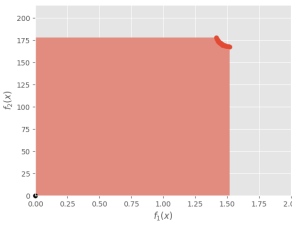


(c) População de tamanho 160

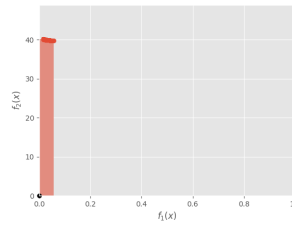
Fig. 20. Hipervolume do cGA do multiobjetivo  $f_{sphere}$  ( $f_1$ ) e  $f_{rosen}$  ( $f_2$ ) com cromossomo de tamanho 10



(a) População de tamanho 10

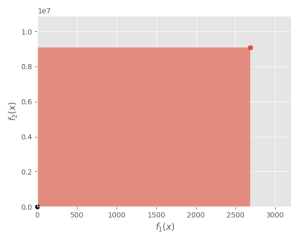


(b) População de tamanho 40

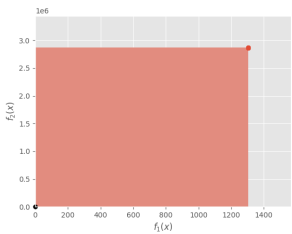


(c) População de tamanho 160

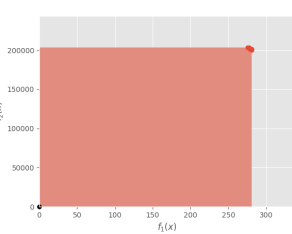
Fig. 21. Hipervolume do cGA do multiobjetivo  $f_{sphere}$  ( $f_1$ ) e  $f_{rosen}$  ( $f_2$ ) com cromossomo de tamanho 40



(a) População de tamanho 10



(b) População de tamanho 40



(c) População de tamanho 160

Fig. 22. Hipervolume do cGA do multiobjetivo  $f_{sphere}$  ( $f_1$ ) e  $f_{rosen}$  ( $f_2$ ) com cromossomo de tamanho 160

**3.2.3  $sGA$   $f_{sphere}$  e  $f_{rosen}$ .** As figuras 17, 18 e 19 apresentam os gráficos de hipervolume das melhores soluções não-dominadas encontradas pelo **sGA** para os problemas Sphere e Rosen para os cromossomos de tamanho 10, 40 e 160, respectivamente. Podemos observar que o valor de *fitness* para  $f_{rosen}$  das soluções não-dominadas trava em um ótimo local enquanto que o valor de *fitness* para  $f_{sphere}$  se aproxima ou atinge *fitness* da solução ótima para o cromossomo de tamanho 10. Para o cromossomo de tamanho 40, podemos identificar o mesmo comportamento do cromossomo de tamanho 10, mas apenas para a população de 160 indivíduos. Para o cromossomo de tamanho 160, vemos o **sGA** não encontrou as soluções não-dominadas próximas à solução utópica. Podemos observar que, para os cromossomos de tamanho 40 e 160, que quanto maior a população maiores as chances do algoritmo encontrar soluções mais próximas da solução utópica.

**3.2.4  $cGA$   $f_{sphere}$  e  $f_{rosen}$ .** As figuras 20, 21 e 22 apresentam os gráficos de hipervolume das melhores soluções não-dominadas encontradas pelo **cGA** para os problemas Sphere e Rosen para os cromossomos de tamanho 10, 40 e 160, respectivamente. Observando todos os gráficos, podemos identificar que o **cGA** possui um comportamento análogo ao **sGA**. Além disso, percebemos que o **cGA** encontrou uma maior variedade de soluções não-dominadas que o **sGA** e que para o cromossomo de tamanho 160 o **cGA** obteve resultados melhores.

### 3.3 Covariância das Populações

Nesta seção apresentamos alguns mapas de calor a fim de observar a relação entre as variáveis (genes) das funções armadilhas:  $f_{trap5}$ ,  $f_{invtrap5}$  e  $f_{rosen}$ . Apresentamos mapas de calor das execuções cujas configuração é 40/40 e apenas das versões monobjetivas do **sGA** e do **cGA**. As cores mais fortes dos mapas de calor representam uma correlação entre as variáveis.

As figuras 23a, 24a, 25a, 26a, 27a e 28a apresentam os mapas de calor das populações iniciais do **sGA** **cGA** para os problemas Trap5, InvTrap5 e Rosen. Dado que a população inicial é composta por indivíduos gerados aleatoriamente, no caso do **sGA**, ou é gerada por um modelo de distribuição ainda não representativo para o problema, então os mapas de calor de tais populações não nos dizem muito. A Figura 28a apresenta um mapa diferente dos demais mapas de calor do **cGA**, pois a evolução da versão binária é diferente da versão real.

**3.3.1  $f_{trap5}$ .** Ao observar a Figura 23b (população intermediária do **sGA**) podemos identificar que grupos de bits escuros se formaram na imagem indicando que existe uma forte relação entre os bits. Na Figura 23c (população final do **sGA**) podemos observar que o grupo mais escuro aumentou, o que reforça a indicação da relação entre os bits. Ao observar a Figura 23b, por sua vez, não conseguimos afirmar que há relação entre as variáveis, apesar de percebermos agrupamento de bits, são agrupamentos de cores mais claras. A Figura 23c apresenta um pouco mais de agrupamento de bits mais escuros, o que indica a existência de relação entre os bits. Além disso, o mapa da Figura 23c continua com uma variação muito forte de cores, o que indica que o **cGA** não converge.

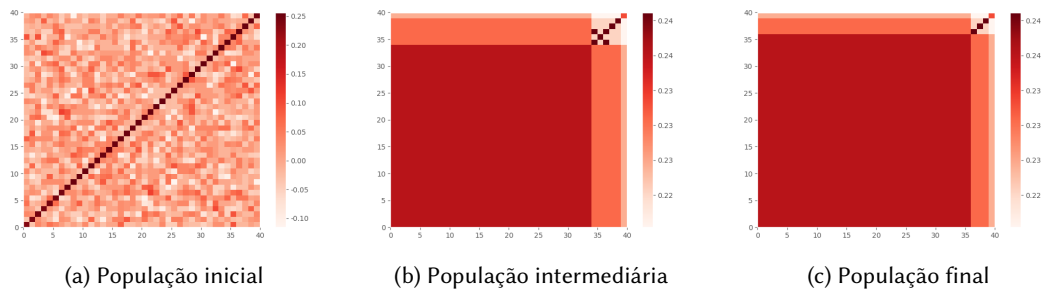
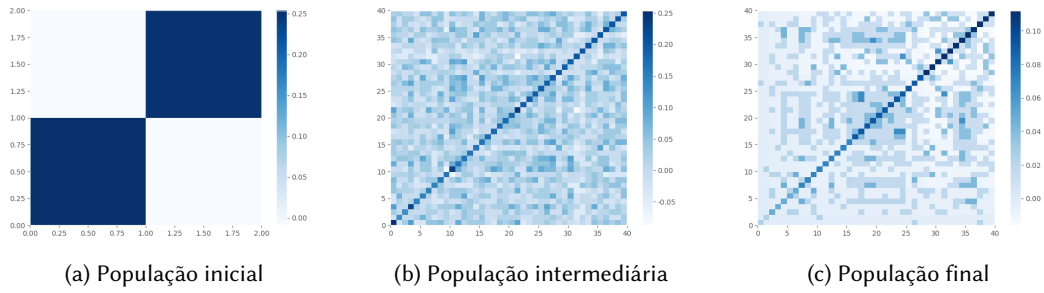
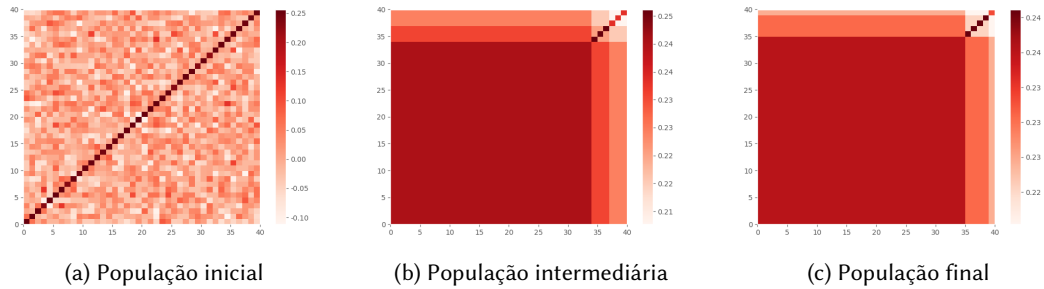
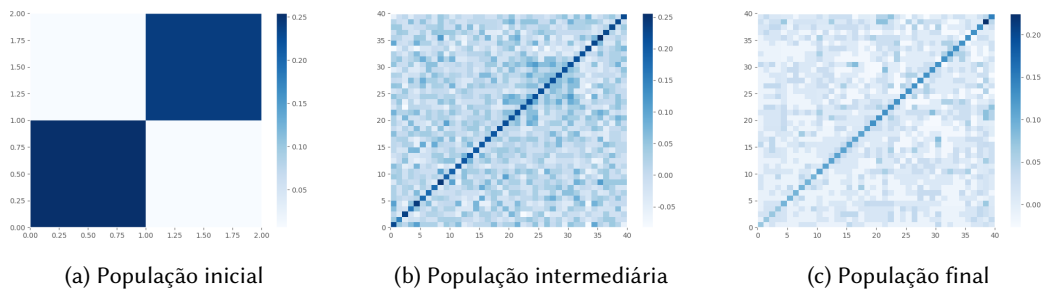


Fig. 23. Mapa de calor do **sGA** da  $f_{trap5}$

Fig. 24. Mapa de calor do cGA da  $f_{trap5}$ 

3.3.2  $f_{inotrap5}$ . Ao observar as figuras 25b (população intermediária do **sGA**), 25c (população final do **sGA**), 26b (população intermediária do **cGA**) e 26c (população final do **cGA**) percebemos que os resultados são análogos aos do problema Trap5. No **sGA** a principal diferença é apenas que um dos agrupamentos de bits da Figura 25b se torna maior na Figura 25c. No **cGA** a principal diferença é que na Figura 26c os bits se tornam mais claros, e há menos indicação de relação entre os bits no mapa. Apesar disso, os mapas indicam uma forte relação entre os bits e a não-convergência de ambos os algoritmos.

Fig. 25. Mapa de calor do sGA da  $f_{inotrap5}$ Fig. 26. Mapa de calor do cGA da  $f_{inotrap5}$ 

3.3.3  $f_{rosen}$ . Ao observar a Figura 27b (população intermediária do **sGA**) podemos identificar que grupos de genes, não muito escuros, se formaram na imagem o que pode indicar que existe uma relação entre os genes.

Na Figura 27c (população final do **sGA**) podemos observar que o mapa de calor se tornou mais claro, mas ainda podemos identificar grupos de genes, o que reforça que existe uma relação entre os genes. Além disso, o mapa ainda possui cores diferentes, e, ainda que ligeiramente diferentes, indica que o **sGA** não converge. Ao observar a Figura 27b conseguimos identificar relação entre as variáveis através do agrupamento de genes escuros. A Figura 27c também apresenta agrupamento de genes, ainda que em locais diferentes, reforçando a existência da relação entre os genes. Além disso, o mapa da Figura 27c mantém uma variação forte de cores, indicando que o **cGA** não converge.

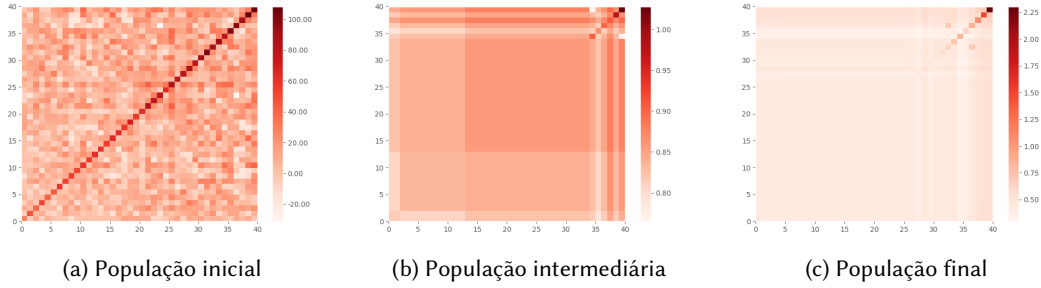


Fig. 27. Mapa de calor do sGA da  $f_{rosen}$

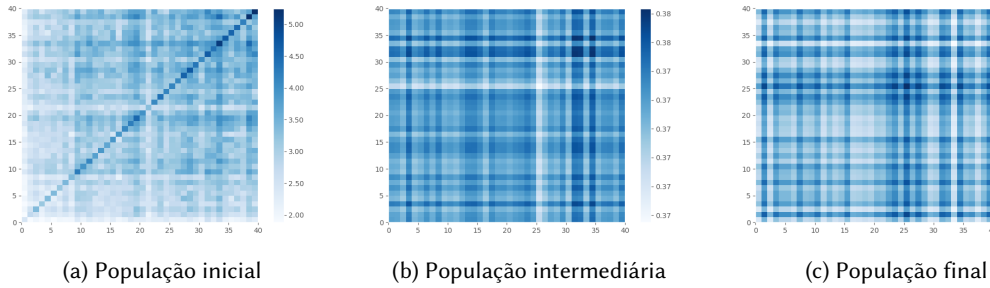


Fig. 28. Mapa de calor do cGA da  $f_{rosen}$

#### 4 ANÁLISE DOS RESULTADOS

Nesta seção apresentamos a análise dos resultados obtidos no experimento. Na nossa análise consideramos apenas a convergência dos algoritmos. Primeiramente, apresentamos a análise dos algoritmos monobjetivos e, em seguida, a análise dos algoritmos multibjetivos.

No problema OneMax, observamos que o **sGA** convergiu para todos os tamanhos de cromossomos e para quase todos os tamanhos de população. O **cGA** também apresentou convergência para muitas das configurações de testes. Contudo, uma parte considerável dos testes realizados para o problema do OneMax não convergiu, tais testes apresentam os maiores cromossomos e populações.

Nos problemas Trap5 e InvTrap5, percebemos um comportamento semelhante. Nestes problemas, o **sGA** convergiu para todas as populações do cromossomo de tamanho 10 e para o cromossomo de tamanho 20, apenas convergiu com as três maiores populações. O **cGA**, por sua vez, apresentou convergência apenas para as populações do cromossomo de tamanho 10. Além disso, apresentamos também nos mapas de calor de um dos

resultados com a configuração 40/40, apresentados na seção anterior, não convergiu bem como apresentado pelas tabelas 3b e 4b.

Ao contrário do que observamos nos problemas anteriores, o **cGA** apresenta melhores resultados para o problema Sphere que o **sGA**. Enquanto que o **cGA** convergiu para os cromossomos de tamanho 10, 20 e 40 em diferentes tamanhos de população. O **sGA** convergiu apenas para os cromossomos de tamanho 10 e 20 para em pouco menos populações. Também identificamos a diminuição do crescimento da taxa de sucesso do **cGA** com o aumento do tamanho dos cromossomos e da população.

No problema Rosen, identificamos que nenhum dos algoritmos foi capaz de encontrar a solução ótima, portanto, nenhum apresentou convergência. Podemos identificar a falta de convergência também nos mapas de calor da configuração 40/40 apresentados na seção anterior.

De modo geral, ao comparar os gráficos dos resultados dos algoritmos monobjetivos, podemos identificar que o **cGA** apresenta queda de *fitness* para os problemas binários com o crescimento da população. Contudo, o **cGA** apresenta o comportamento contrário para os problemas reais, resultando em solução de maior *fitness*. Além disso, observamos também que a convergência das soluções de ambos os algoritmos aplicados às funções armadilhas, especialmente do **cGA**, foi prejudicada devido às fortes relações entre as variáveis.

Os algoritmos multibjetivos, por sua vez, apresentaram, em geral, resultados promissores, e certos comportamentos serem semelhantes aos monobjetivos. Os resultados do **sGA** multibjetivo binário com as funções  $f_{trap5}$  e  $f_{intrap5}$  apresentaram convergência das soluções em todas as configurações e também apresentaram ter mais chances de convergir com o aumento da população. As soluções do **cGA** também apresentaram convergência, contudo, bem como a versão monobjetiva, o **cGA** apresentou uma diminuição do *fitness* das soluções com o crescimento do cromossomo ou da população.

Os resultados do **cGA** multibjetivo reais, por sua vez, apresentaram melhores que os binários, provavelmente, devido à natureza das funções  $f_{sphere}$  e  $f_{rosen}$ . Apesar disso, os resultados do **sGA** e do **cGA** se comportaram de forma semelhante. Os resultados de ambos os algoritmos convergiram para os cromossomos de tamanho 10 e 40. Para os cromossomos de tamanho 160, apesar de não ter convergido, podemos observar que as soluções tendem a convergir com o aumento da população.

## 5 CONCLUSÃO

Neste trabalho, realizamos uma comparação entre os algoritmos **sGA** e **cGA** tanto as versões monobjetivas quanto as multiobjetivas. Para tal, utilizamos as funções:  $f_{onemax}$ ,  $f_{trap5}$ ,  $f_{intrap5}$ ,  $f_{sphere}$  e  $f_{rosen}$ . Analisamos a convergência e o comportamento de cada abordagem, as quais nos permitiram concluir que:

- Para as versões monobjetivas: (1) o **sGA** apresenta tem mais chances de convergir quanto maior o tamanho da população; (2) o **cGA** apresenta queda do *fitness* para os problemas binários com o crescimento do cromossomo ou da população; (3) o **cGA** apresenta um aumento do *fitness* para os problemas reais com o crescimento da população; e (4) o *fitness* das soluções de ambos os algoritmos aplicados às funções armadilhas foi prejudicada devido às fortes relações entre as variáveis.
- Para as versões multibjetivas: (1) o **sGA** apresentou convergência das soluções em todas as configurações e maior chances de convergir com o aumento da população; (2) o **cGA** apresentou convergência semelhante à da versão monobjetiva, pois o **cGA** apresentou uma diminuição do *fitness* das soluções com o crescimento do cromossomo ou da população; e (3) o **sGA** e o **cGA** se comportaram de forma semelhante para os problemas de números reais, com a convergência para os cromossomos de tamanho 10 e 40 e a tendência de convergir com o aumento da população para o cromossomo de tamanho 160.

Caso seja interesse do leitor, o código do experimento pode ser acessado através do link: [github.com/brenov/ies-usp](https://github.com/brenov/ies-usp).



## REFERÊNCIAS

- [1] Charles Darwin. 2018. *On the Origin of Species: Or, The Preservation of the Favoured Races in the Struggle for Life*. Read Books Ltd.
- [2] Georges R Harik, Fernando G Lobo, and David E Goldberg. 1999. The compact genetic algorithm. *IEEE transactions on evolutionary computation* 3, 4 (1999), 287–297.
- [3] Jeffrey R Sampson. 1976. Adaptation in natural and artificial systems (John H. Holland).
- [4] Antonio Helson Mineiro Soares. 2014. *Algoritmos de estimação de distribuição baseados em árvores filogenéticas*. Ph.D. Dissertation. Universidade de São Paulo.

## A RESULTADOS DO PROBLEMA ROSEN 40/160

	sga	cga
1	174.7594867774	133.7596688222
2	1478.6653678388	1651914449253
3	286.5007873848	212.5804954817
4	91.2879372756	166.1149901655
5	153.4943654876	143.3890841822
6	1452.2345809122	227.376986609
7	175.5462864206	264.3589729404
8	39.927896521	213.2992652965
9	1530.9676982684	206.2499681956
10	225.3879961891	229.7356702658
11	182.1797470334	213.1159089452
12	184.808265792	174.6628616659
13	294.541663264	156.1919376978
14	100.7977814804	201.1417996891
15	186.1152549487	106.8261041207
16	69.3381797467	306.362953786
17	215.0511277469	166.0632377403
18	300.1573661091	146.1564445656
19	154.752402121	138.103845969
20	185.927072582	139.3913838387
21	400.0202493075	206.4632724372
22	390.1751731517	124.7345838135
23	477.0823625058	87.1626370472
24	194.3774356026	220.4761200695
25	186.0629068993	98.4851389307
26	187.1049311051	139.4737463566
27	1326.4062693652	239.0229384267
28	408.089678233	206.7098880446
29	3103.7623271665	173.5662728992
30	108.371964066	124.103057314
31	118.5114552881	268.7640866307
32	1719.2764611931	195.85984596
33	6423.9719181046	199.020886124
34	2349.701372329	273.9177938955
35	224.379850288	175.3963222486
36	149.4836585717	268.0774578639
37	210.1752986106	299.8203660473
38	770.4425382511	146.5414023864
39	139.5514617911	143.2297998109
40	51.5746069487	105.3060855104
41	94.9726897991	228.6157630698
42	311.7645349565	153.8929667432
43	150.3257547436	197.484296107
44	644.359281258	92.284519788
45	481.4101934913	220.4844844708
46	221.5704611412	111.1555751989
47	172.5427631603	238.6102097975
48	4278.827109428	156.7414704458
49	195.3341537608	270.9770000242
50	195.4572615152	86.8874464355
51	710.1651205941	110.9564731632
52	157.7367891475	274.9417604957
53	111.4089664794	262.7479425831
54	144.6411046346	141.4400330202
55	275.9881234976	124.9619760432
56	144.0194923415	178.4646961886
57	117.5408822559	154.7921935786
58	155.3007829966	189.196923948
59	307.6369590007	150.2212816329
60	98.4334816097	112.5033347332
61	176.4650117292	232.4731382834
62	89.4121823531	231.274682049
63	142.828455595	181.537467012
64	725.0142240673	124.0842357565
65	175.4745257222	210.9089659753
66	193.1267765695	164.0714294937
67	211.6644639634	115.0376078531
68	136.0152219392	282.4917609634
69	153.9753975266	223.1246969104
70	322.7304512612	101.8070970144
71	230.4398748374	161.9689494563
72	101.2708952697	148.4457559975
73	810.6674445249	159.7543633314
74	272.2966428418	92.3732091906
75	221.6572785167	285.678372352
76	218.5612759986	274.8757668661
77	129.1774471406	284.2748124796
78	172.2147756657	236.9702393842
79	525.6849307009	216.0362522663
80	109.174334229	268.9738597459
81	234.1556975452	192.0717185041
82	977.3615484339	290.0377057018
83	489.6027323763	155.7208254085
84	163.6087367201	96.7038624143
85	93.7528656921	232.668671732
86	1034.369947291	118.4556897037
87	178.4218407777	153.7168963963
88	164.9968562887	206.0548945085
89	106.2506979478	51.7893292592
90	227.8209912699	179.7412895095
91	637.9554331068	198.2193212439
92	168.1345398102	127.9319480747
93	1684.9019209483	220.7074008643
94	110.7281706017	245.2564823217
95	600.3630674446	211.0078371947
96	153.7414296907	283.3040620755
97	163.1735456785	313.5520366416
98	1085.9646810032	250.7959123126
99	140.7947453878	159.9967105764
100	225.8857809915	149.5413511858

Tabela 7. Resultados da  $f_{rosen}$  com configuração 40/160