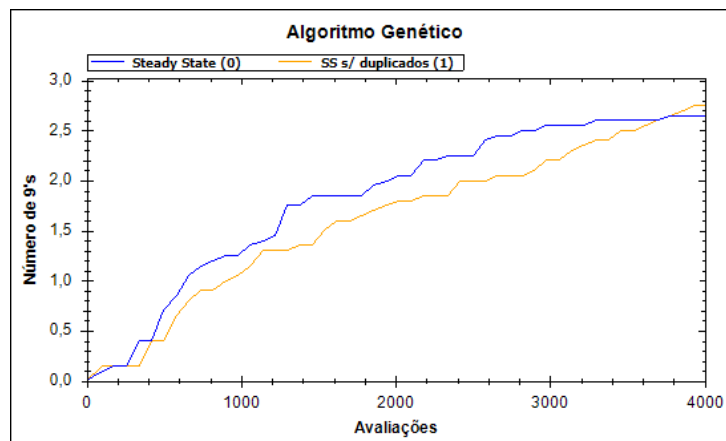
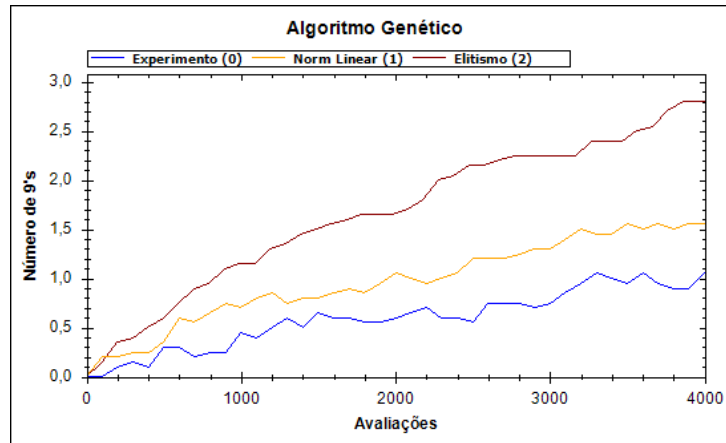


# Trabalho 1 Inteligência Computacional Aplicada

Leo Land Bairos Lomardo - 2020201  
Luigi Gosling de Alencastro - 1811556

Maio 2023

# 1 Questão 1



Analisando os resultados obtidos, percebemos que a GA 1-1 apresentou o pior resultado em comparação com as demais, tal resultado se justifica pela simplicidade do modelo, onde seu algoritmo não apresenta nenhuma outra característica além das mutações e *crossover*, características que as demais também possuem.

Ao implementar a normalização linear como método de seleção, houve uma melhora significativa em comparação com o primeiro, porém seu resultado permaneceu abaixo em comparação com os demais. No caso das GA's 2-3 e 2-4 (*Stead-State*), os resultados foram os melhores, onde nesse algoritmo ocorre uma seleção dos melhores indivíduos (GAP), preservando os melhores indivíduos, descartando a necessidade de reavaliar os mesmos. No caso do *Stead-State* com duplicadores, a população apresenta um comportamento mais estático. Já no caso sem duplicadores, ocorre a eliminação dos

indivíduos duplicados, obtendo assim uma maior eficiência de paralelismo de busca.

Na GA 2-2, onde possui elitismo, o resultado foi muito parecido com o das GA's 2-3 e 2-4, onde nesse caso a GA apresentou um crescimento mais linear. Tal resultado se justifica pelo algoritmo, onde antes mesmo da mutação, se copia o melhor cromossomo de  $P(x)$  para  $P(x+1)$ , conservando a “qualidade” do melhor indivíduo para a “geração futura”.

## 2 Questão 2

Por ser um processo estocástico, onde duas execuções com exatamente os mesmos parâmetros podem apresentar resultados diferentes, optamos por rodar 3 vezes para cada GAP, apresentando um valor médio.

Abaixo estão as tabelas com os resultados, onde destacamos os 2 piores resultados e o melhor resultado de cada algoritmo.

Stead-State com duplicador					
GAP	Resultado_1	Resultado_2	Resultado_3	Média	Devio Padrao
5	2,35	2,85	2,65	2,62	0,252
10	2,5	2,05	3,25	2,60	0,606
15	2,45	2,5	2,15	2,37	0,189
20	2,84	1,8	2,45	2,36	0,525
25	2,14	2,95	2,45	2,51	0,409
30	2,3	2	3,4	2,57	0,737
35	2,2	3,85	2,15	2,73	0,967
40	2,15	2,55	3,1	2,60	0,477
45	2,14	3	2,7	2,61	0,437
50	2,95	3,45	2,35	2,92	0,551
55	2,19	2,25	1,9	2,11	0,187
60	2,9	2,5	2,17	2,52	0,366
65	3,2	1,88	2,65	2,58	0,663
70	2,45	3,2	2,9	2,85	0,377
75	2,7	2,9	2,65	2,75	0,132
80	2,05	2,45	3,1	2,53	0,530
85	2,6	2,65	1,9	2,38	0,419
90	2,05	2,25	3,55	2,62	0,814
95	2,2	3,5	2,39	2,70	0,702
100	1,94	2,19	1,7	1,94	0,245

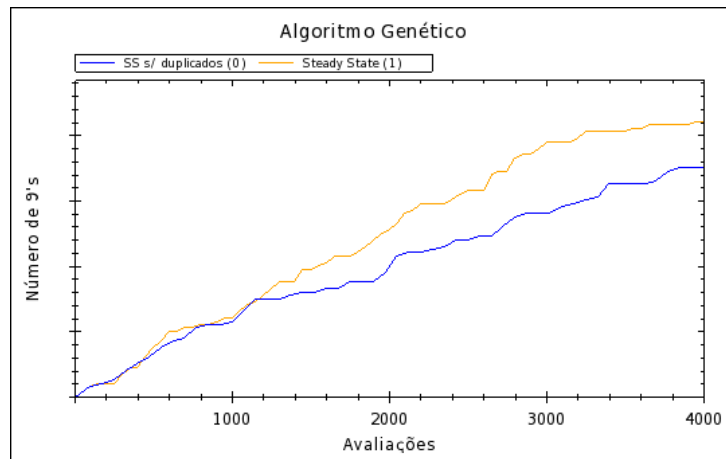
Stead-State sem duplicador					
GAP	Resultado_1	Resultado_2	Resultado_3	Média	Desvio Padrao
5	3,2	2,1	3,45	2,92	0,718
10	2,6	2,55	3,05	2,73	0,275
15	2,8	3,1	3,6	3,17	0,404
20	2,75	2,45	3,45	2,88	0,513
25	2,6	1,8	3,3	2,57	0,751
30	3,3	2,9	2,7	2,97	0,306
35	3,05	3,1	2,1	2,75	0,563
40	3,03	2,63	2	2,55	0,519
45	2,95	2,15	4,4	3,17	1,141
50	2	3,5	2,8	2,77	0,751
55	1,9	3,69	2,2	2,60	0,959
60	2,35	4,2	2,7	3,08	0,983
65	3,3	2,95	3,4	3,22	0,236
70	3,2	3,5	2,9	3,20	0,300
75	4,4	3,5	4,15	4,02	0,465
80	2,6	3,1	3,4	3,03	0,404
85	2,9	2,2	3	2,70	0,436
90	3,45	2,3	3	2,92	0,580
95	2,4	2,45	2,5	2,45	0,050
100	1,45	1,7	1,65	1,60	0,132

Como pudemos perceber, em ambos os algoritmos o valor de 100% apresentou o pior resultado, isso se deve pelo fato da variável GAP representar a porcentagem da população que é trocada. Logo, o valor de 100% representa a mesma taxa de troca que 0%, descaracterizando o algoritmo.

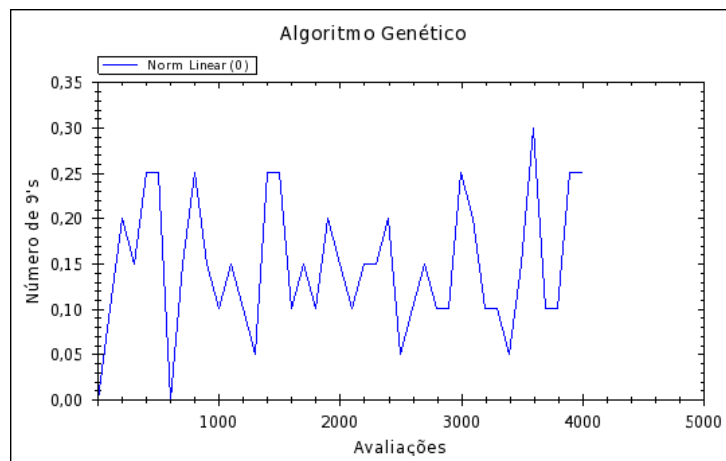
Dois pontos interessantes que analisamos, é que durante a análise dos 3 resultados de cada GAP, percebemos que no teste com GAP = 55 do *Stead-State* com duplicador apresentou o segundo menor desvio padrão, mostrando que o resultado foi constantemente pior que os demais(descartando GAP = 100%).

Outro dado analisado, foi que ao calcularmos o desvio padrão de cada GAP e depois comparar a média do mesmo para *Stead-State* com e sem duplicador, percebemos que o *Stead-State* sem duplicador apresentou resultados com caráter mais oscilatório.

Na figura abaixo podemos ver ambos os valores em comparação, onde o valor de 50% com duplicador apresentou um resultado superior ao 75% sem duplicador.



### 3 Questão 3



Como podemos ver na imagem, o gráfico da solução com os parâmetros do enunciado apresenta uma oscilação muito grande. Analisando os parâmetros, podemos afirmar que a alta taxa de mutação tornou o processo aleatório. Combinado a este fato, a baixa taxa de cruzamento, pode ter resultado no descarte de alguma característica do indivíduo que iria contribuir com a solução.

No caso oposto, onde a taxa de mutação é muito baixa, a diversidade da população cairia bastante. No caso onde a taxa de *crossover* é demasiadamente alta, pode ocorrer uma convergência prematura da rede.

## 4 Questão 4

Seguindo os valores estipulados no enunciado, e o fato de o número total de indivíduos ser igual ao tamanho da população multiplicado pelo número de gerações, realizei os testes com os seguintes parâmetros:

GA 2-2		
Gerações	População	Total indivíduos
20	20	400
20	50	1000
20	70	1400
20	100	2000
20	150	3000
20	180	3600
20	200	4000

Inicialmente, pode-se perceber o resultado do algoritmo *Elitismo(0)*, onde a população era igual a 20, apresentou um resultado pouco expressivo quando comparado com os demais, visto que ele não tinha uma quantidade de indivíduos grande o suficiente para se desenvolver.

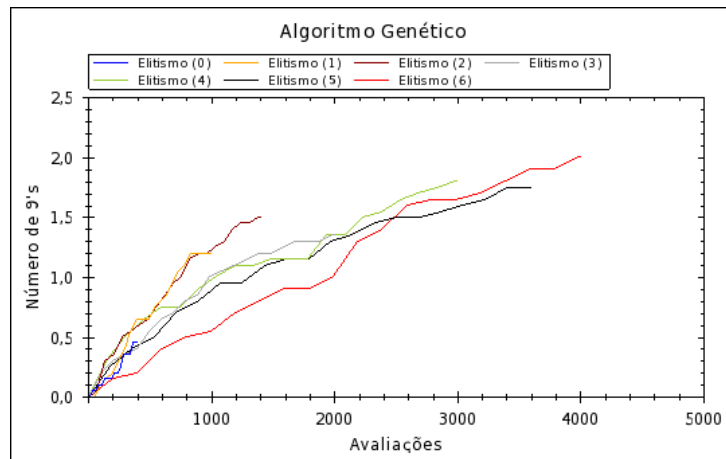
Já no *elitismo(1)* (população = 50), houve uma melhora significativa dos resultados, porém mesmo assim podemos ver que o algoritmo ainda pode se desenvolver mais com uma população maior, visto que o gráfico apresenta uma inclinação alta e nos últimos resultados, a reta não apresenta nenhum sinal de estabilizar horizontalmente, apenas crescer.

No *elitismo(2)* (população = 70), obtivemos o melhor resultado dentre os algoritmos com população menor que 100.

O caso do *elitismo(3)* (população = 100) apresentou um resultado inferior ao anterior, porém acredito que tal piora se deva à aleatoriedade do processo.

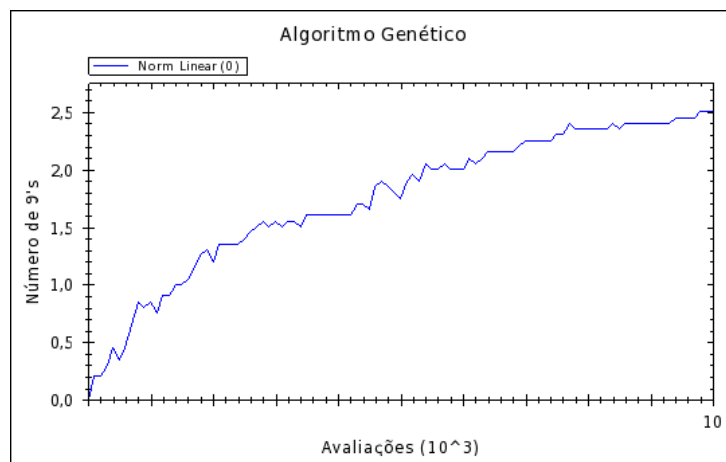
O *elitismo(4)* e *elitismo(5)* apresentaram resultados muito parecidos, e superiores aos anteriores, e pode-se perceber uma lateralização da reta no seu final.

O algoritmo que apresentou o melhor resultado foi o *elitismo(6)* onde a população era a maior de todas, resultando em uma quantidade de indivíduos significativa. Com uma quantidade total de indivíduos mais expressiva, evitamos uma convergência prematura e também permitiu ao algoritmo ter uma solução mais geral do problema.

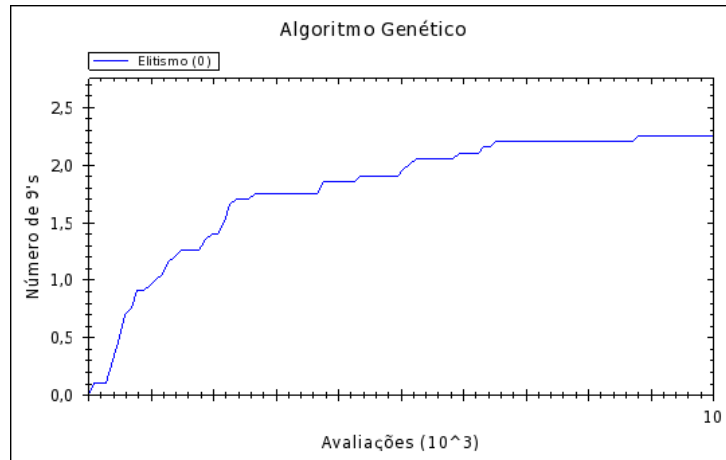


## 5 Questão 5

Na imagem abaixo temos o resultado do algoritmo referente a GA2-1 com as alterações propostas no enunciado.



Analisando a curva do gráfico, podemos ver uma lateralização quando nos aproximamos de 8000 indivíduos, mostrando que a rede está próxima de convergir. Além dessa lateralização, percebe-se uma diminuição na oscilação da curva.



Já no gráfico referente a GA2-2 com o novo valor de indivíduos totais, podemos pontuar inicialmente, o constante crescimento dos resultados, característica do elitismo. Além disso, a curva começou a ficar fortemente lateralizada próximo da marca de 6500 indivíduos.

Podemos concluir que a GA2-1 precisa de aproximadamente 8000 indivíduos e a GA2-2 6500 indivíduos.

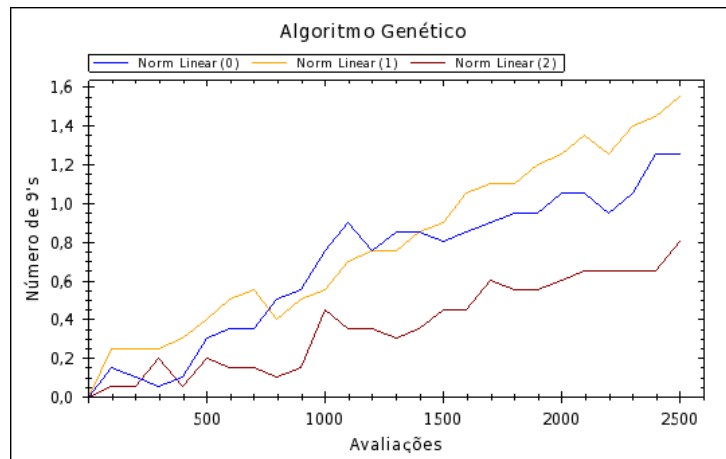
## 6 Questão 6

Analisando inicialmente o gráfico abaixo, referente a GA2-1 onde a numeração da legenda corresponde aos tipos de *crossover One Point*, *Two Point* e *Uniform*, respectivamente.

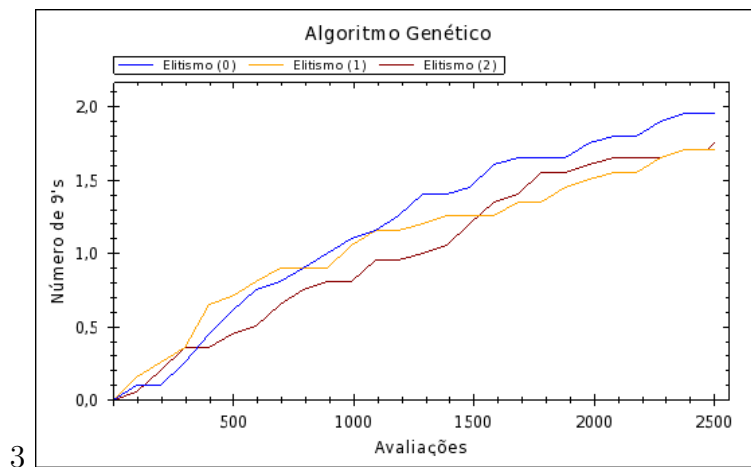
Realizamos 3 testes para cada tipo de *crossover* em cada GA, tentando buscar uma região que representa o resultado, e em todas as vezes, o *crossover Uniform* apresentou um resultado significativamente inferior aos outros 2. Isso se deve pelo fato de o *crossover Uniform* ter um poder destrutivo muito grande, visto que ele destrói com a formação de padrões, característica essa muito prejudicial na normalização linear, visto que nela os indivíduos são ordenados de acordo com sua aptidão.

Por outro lado, o *crossover One Point* e *Two Point*, apresentaram resultados superiores e de certa forma similares, onde o *Two Point* prevaleceu. Vale destacar que o *crossover Two Point* apresentou um crescimento mais constante e menos oscilatório que o *One Point*.





Já na GA2-2, o *crossover* uniforme prevaleceu sobre os demais. Tanto o *crossover One Point* e *Two Point* apresentaram um resultado final muito similar. O resultado no *crossover Uniform* se torna mais eficiente no elitismo devido ao fato que a troca de cromossomos ocorre entre os melhores indivíduos.

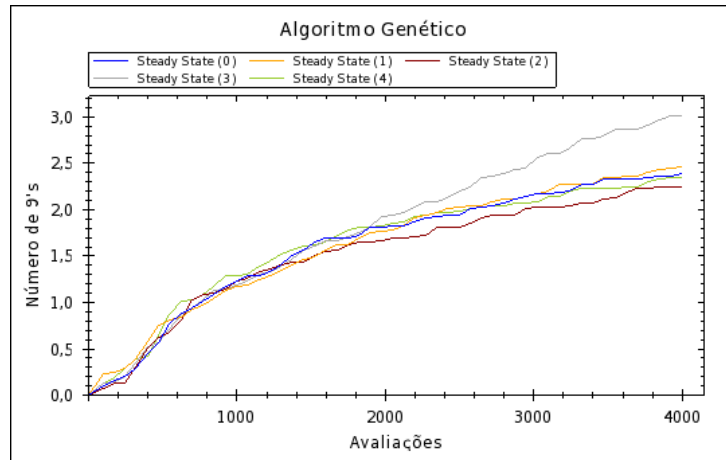


3

## 7 Questão 7

Na normalização linear, o valor máximo e mínimo definem a pressão seletiva sobre os indivíduos. Caso o valor máximo seja muito pequeno, indivíduos ruins poderão se reproduzir com maior facilidade, e caso seja muito alto, a taxa de reprodução será afetada, visto que poucos indivíduos poderão se reproduzir.

Na imagem abaixo o melhor resultado foi com  $MÁX = 200$ .



## 8 Questão 8

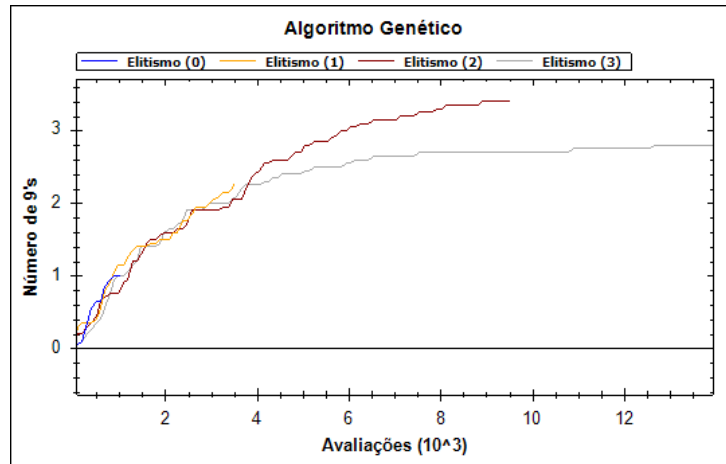
Para os testes livres, serão testados 4 valores diferentes, sem a variação de algum outro durante os testes específicos. Serão utilizados como parâmetros base os valores da tabela referente à questão 1, realizando alterações em um parâmetro ou em uma característica do algoritmo. A mudança dos parâmetros do algoritmo nos testes abaixo foi feita de maneira adaptativa, analisando os resultados previamente obtidos para determinar os valores seguintes.

### 8.1 Características da Execução

Por base, a característica do algoritmo será o *Elitismo* com  $\text{min} = 1$  e  $\text{máx} = 100$  e *Crossover One-Point*

#### 8.1.1 Avaliações

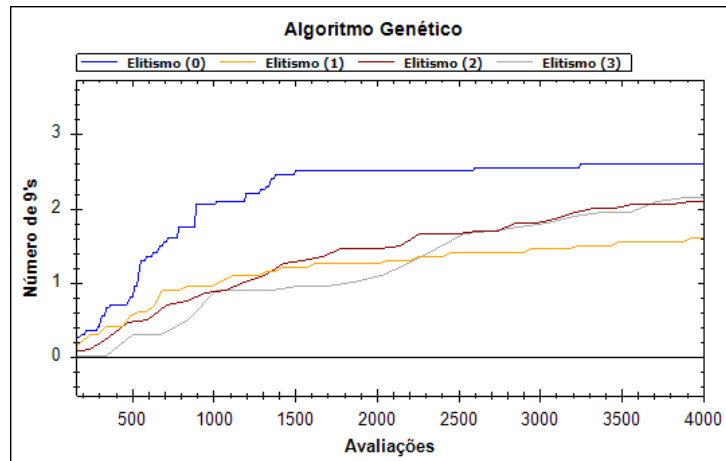
No gráfico, os valores testados foram 1500, 3500, 9500 e 14000, respectivamente.



É possível observar que quanto maior o número de avaliações, maior é a probabilidade de maior qualidade dos valores. O 1º valor não teve um valor satisfatório, mas em compensação o gráfico referente às 14000 avaliações também estagnaram depois de algum tempo, provavelmente devido à aleatoriedade.

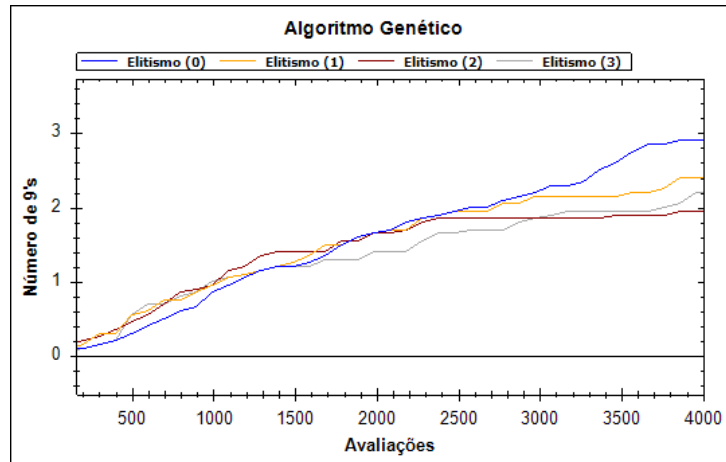
### 8.1.2 Tamanho da População

Para este parâmetro utilizamos os valores 10, 50, 120 e 170, respectivamente.



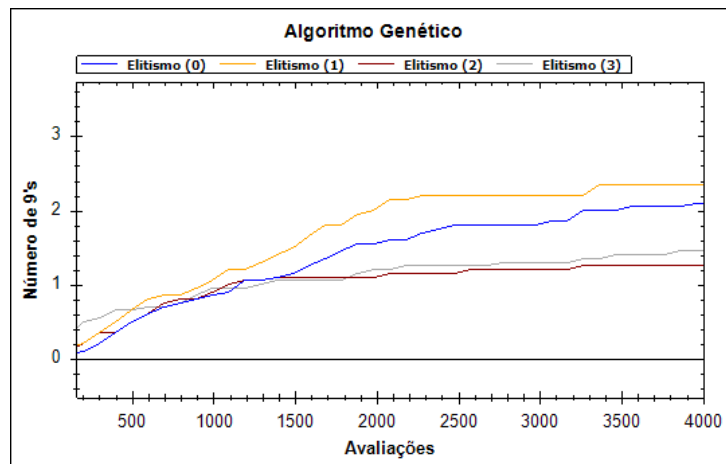
### 8.1.3 Taxa de *Crossover*

Para a taxa de *Crossover* serão utilizados os valores 15, 40, 65, 90, respectivamente.



#### 8.1.4 Taxa de Mutação

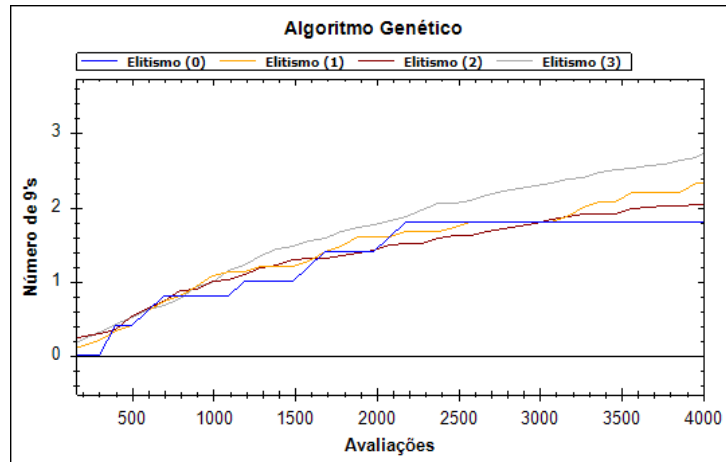
Para taxa de Mutação, serão utilizados os valores 0.2, 2, 20, 70, respectivamente.



A respeito da taxa de mutação, o *elitismo(1)* prevaleceu como melhor resultado, sendo um valor não tão alto nem tão baixo. Quanto menor essa taxa, melhor foi o resultado nos testes.

#### 8.1.5 Taxa de rodadas

Para taxa de rodadas, serão utilizados os valores 5, 15, 45, 75, respectivamente.



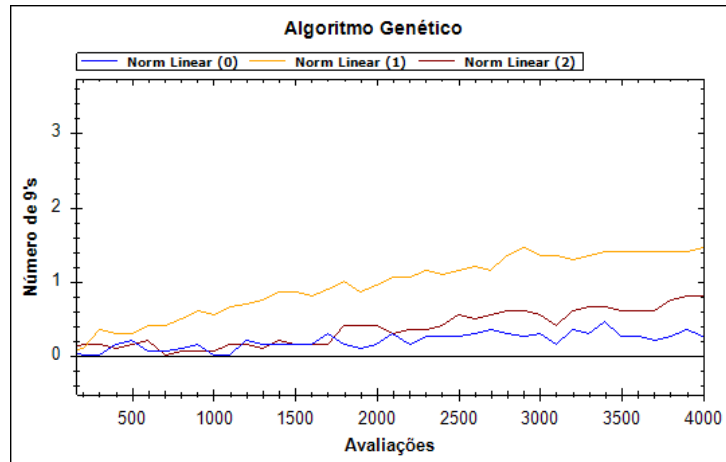
É possível perceber que conforme o valor da taxa de rodadas aumenta, mais constante é a eficiência do algoritmo. O que teve melhor resultado foi o *elitismo(3)*, porém os de número 1 e 2 também obtiveram uma melhora ao longo das avaliações bem considerável. Apenas o *elitismo(0)* que estagnou depois de um tempo.

## 8.2 Características do Algoritmo

Serão alterados apenas as características do algoritmo, as características da execução serão as mesmas da tabela da questão 1, mantendo tipo de *crossover One Point*.

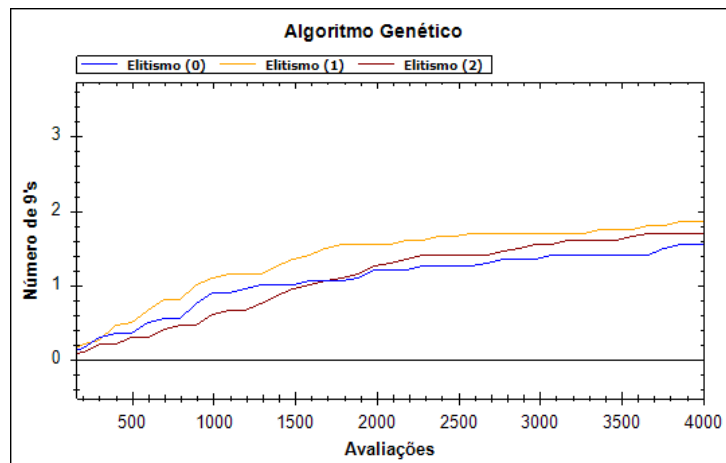
### 8.2.1 Normalização linear

Para o teste deste item será testado primeiro um mínimo alto(60 e 100), depois um máximo baixo(1 e 40) e por fim valores mínimo e alto medianos(25 e 65).



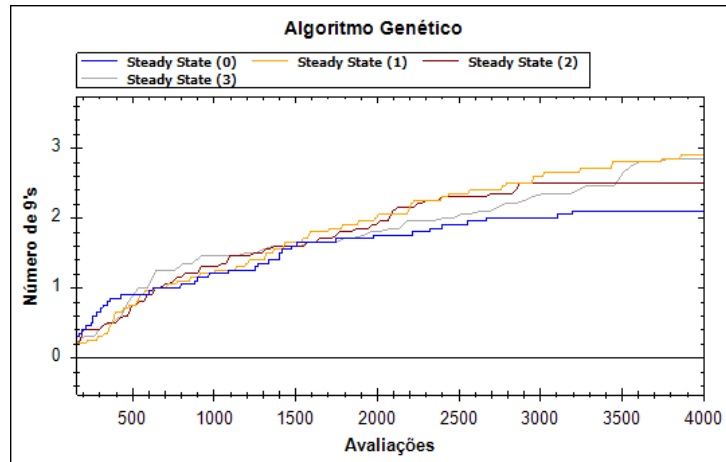
### 8.2.2 Elitismo

Para o teste deste item serão utilizados os mesmo parâmetros do anterior.



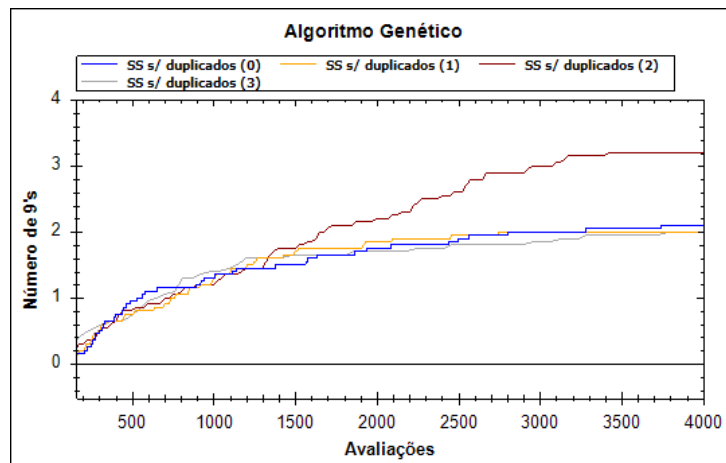
### 8.2.3 *Steady State*

Para este item, os valores mínimo e máximo voltam a ser de 1 a 100. O gap será modificado entre os valores 1, 10, 25, 55.



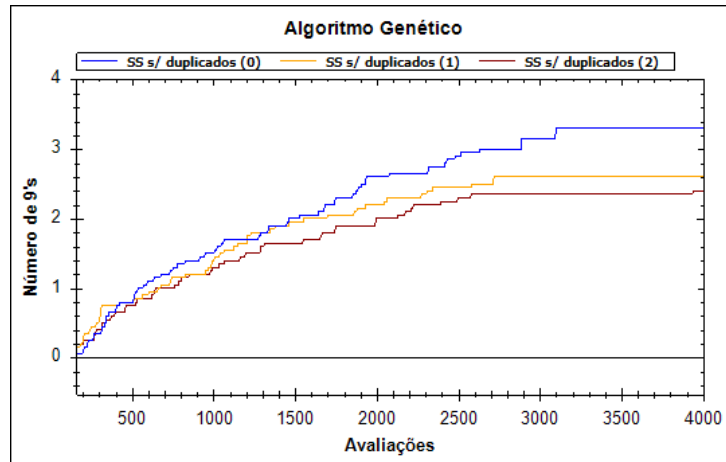
#### 8.2.4 *Steady State* sem duplicados

Para este item serão utilizados os mesmos parâmetros do item anterior.



### 8.3 Tipo de *Crossover*

Para os tipos de *crossover* foram utilizados os parâmetros da questão 1, variando entre *One Point*, *Two Point* e *Uniform*. A característica do algoritmo base utilizada foi *Steady State* sem duplicados.



## 9 Consideração Final

Analisando os resultados obtidos no decorrer do trabalho, pudemos assimilar empiricamente a parte teórica da matéria. Entretanto, pela GA ser um modelo de caráter probabilístico, tiveram momentos em que os resultados oscilaram de maneira significativa, gerando uma certa dúvida na confiabilidade do algoritmo. Porém com o decorrer dos experimentos, entendemos a característica estocástica do processo.