

# Algoritmo NGSA2 e VEGSO para Otimização Multiobjetivo

Tiago Santos Ferreira

PECS – UEMA

2020

# Introdução

- Uma grande variedade de problemas em engenharia, indústria e muitos outros campos envolvem a otimização simultânea de vários objetivos. Em muitos casos, os objetivos são definidos em unidades incomparáveis e apresentam algum **grau de conflito** entre eles (ou seja, um objetivo não pode ser melhorado sem a deterioração de pelo menos outro objetivo).

# Introdução

O método mais comumente adotado na otimização multiobjetivo para comparar soluções é o denominado relação de dominância de Pareto que, ao invés de uma única solução ótima, leva a um conjunto de alternativas com diferentes trade-offs entre os objetivos.

Essas soluções são chamadas de soluções ótimas de Pareto ou soluções não dominadas (non-dominated).

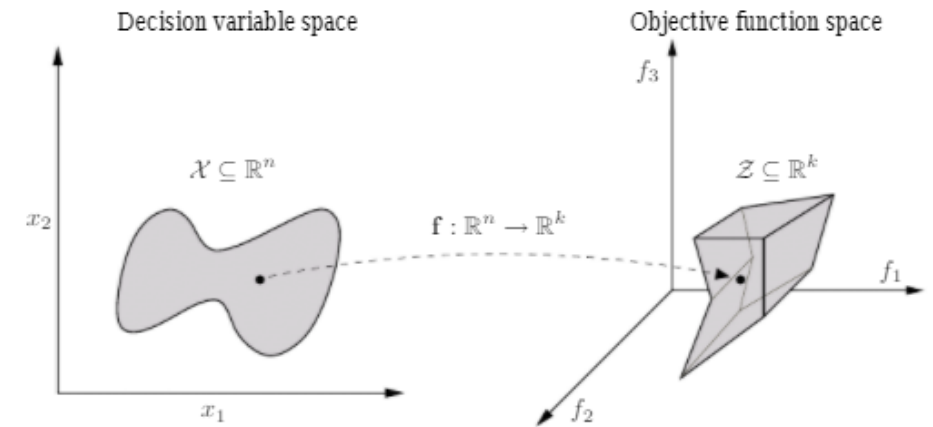


Figura 1. Espaços de busca em problemas de otimização multiobjetivo.

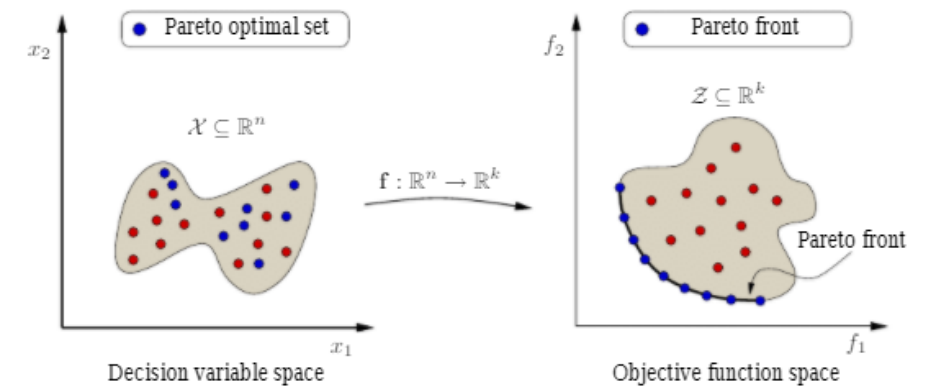


Figura 2. Fronteira/Frente de Pareto (pareto-front).

# Non Dominated Sorting Genetic Algorithm 2 (NSGA2)

- o NSGA-II trabalha com o mesmo fluxo de um algoritmo genético padrão.
- A diferença é que, ao final de cada iteração, uma nova população para a próxima iteração é escolhida usando o método de classificação não dominado (Non Dominated Sorting).
- As soluções são divididas em várias frentes com base em seu status de dominação umas contra as outras. Se tivermos o tamanho inicial da população de 10, os 10 primeiros indivíduos da população de pais e filhos da **melhor frente** (elitismo) serão escolhidos para a próxima iteração. Além disso, para controlar a diversidade de soluções, calcula-se o crowding distance, que corresponde à distância de Manhattan de duas soluções vizinhas para dois objetivos. Aqueles com a maior distância dos vizinhos são finalmente selecionados.

# Vector Evaluated Glowworm Swarm Optimization (VEGSO)

- O VEGSO apresenta o mesmo princípio do algoritmo VEGA (vector Evaluated Genetic Algorithm), a ideia principal é que uma parte do enxame concentra-se na função 1, e a outra parte na função 2.
- Para o Algoritmo GSO essa diferenciação está localizada na atualização do brilho (luciferin) de cada indivíduo da população.

```
#update luciferin
for i in glowworms[0:particle_size//2]:

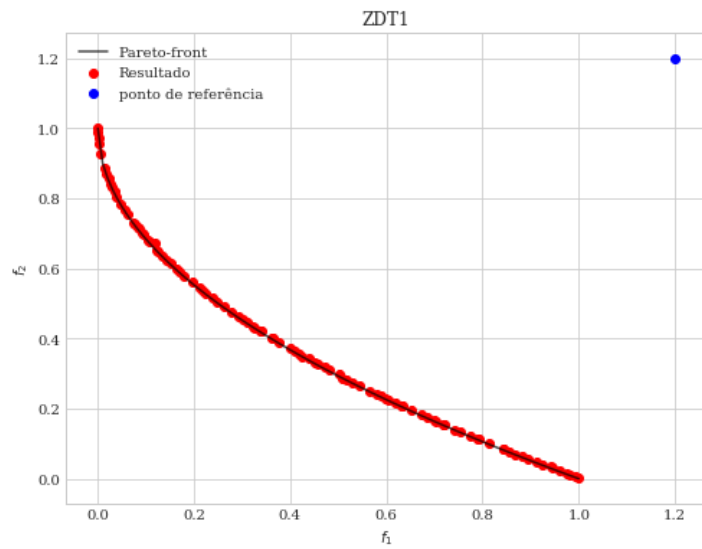
    i.luciferin = (1-rho)*i.luciferin - gamma*ZDT_f1(i.posX, i.posY)

    #i.luciferin = (1-rho)*i.luciferin + gamma*objective_function(i.posX, i.posY)
for i in glowworms[((particle_size//2)+1):particle_size]:

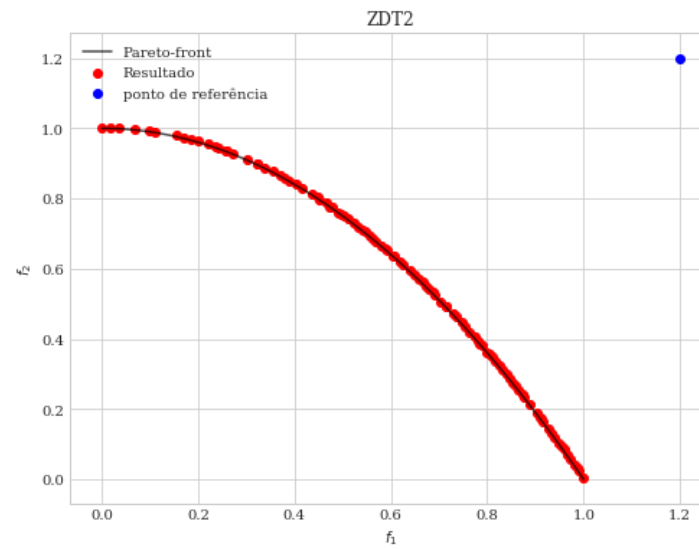
    i.luciferin = (1-rho)*i.luciferin - gamma*ZDT_f2(i.posX, i.posY)

    #i.luciferin = (1-rho)*i.luciferin + gamma*objective_function(i.posX, i.posY)
```

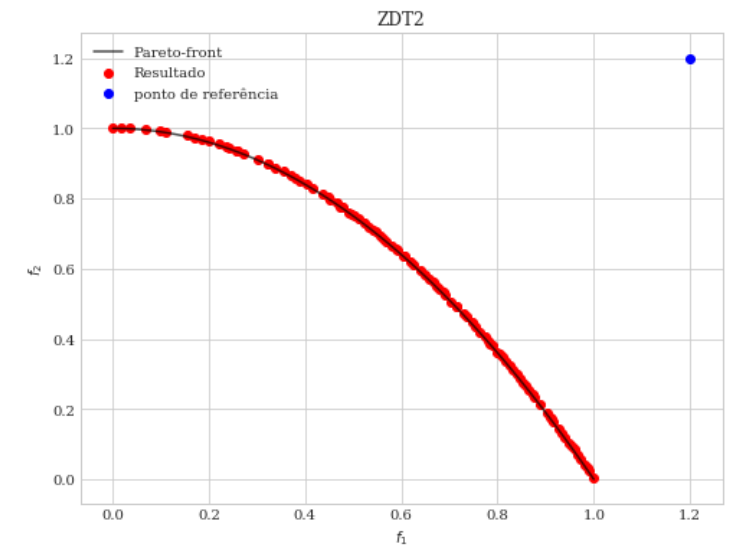
# Resultados NSGA2



a)



b)



c)

Figura 3. a) Resultados NSGA2 para ZDT1 b) Resultados NSGA2 para ZDT2 c) Resultados NSGA2 para ZDT3

# Resultados VEGSO

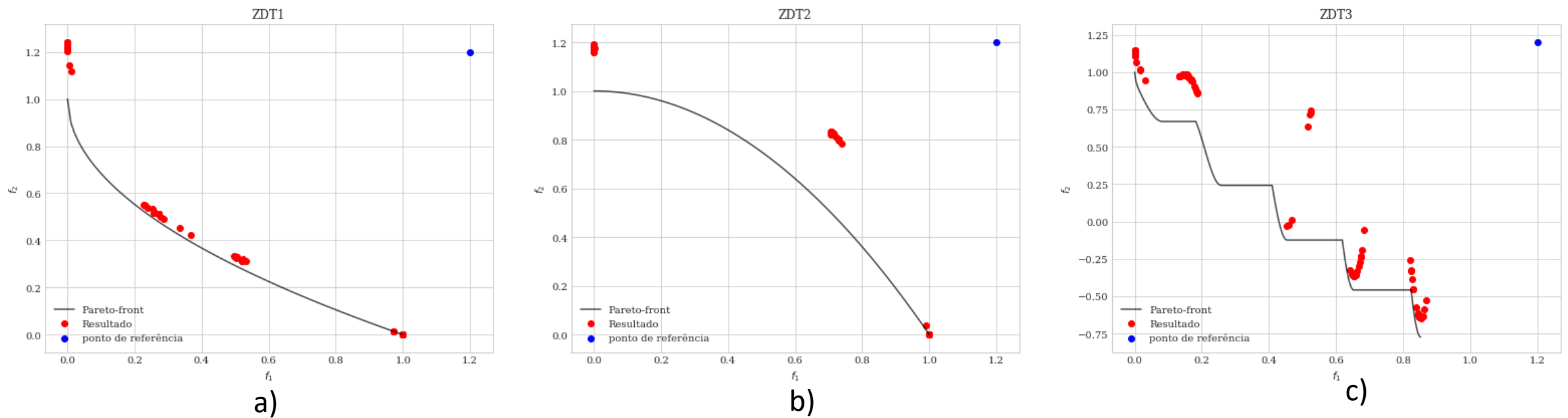


Figura 4. a) Resultados VEGSO para ZDT1 b) Resultados VEGSO para ZDT2 c) Resultados VEGSO para ZDT3

# Resultados

Um resultado de 'rejeitar = Verdadeiro' significa que uma diferença significativa foi observada.

Logo o Algoritmo NSGA2 mostrou-se melhor para solução dos problemas propostos

```
Teste Tukey para ZDT1:      Multiple Comparison of Means - Tukey HSD, FWER=0.05
=====
  group1      group2  meandiff p-adj  lower  upper  reject
-----
outNSGA2_ZDT1 outVEGSO_ZDT1  0.0965 0.0013 0.0505 0.1426  True
-----

Teste Tukey para ZDT2:      Multiple Comparison of Means - Tukey HSD, FWER=0.05
=====
  group1      group2  meandiff p-adj  lower  upper  reject
-----
outNSGA2_ZDT2 outVEGSO_ZDT2 -0.1065 0.0128 -0.1835 -0.0295  True
-----

Teste Tukey para ZDT3:      Multiple Comparison of Means - Tukey HSD, FWER=0.05
=====
  group1      group2  meandiff p-adj  lower  upper  reject
-----
outNSGA2_ZDT3 outVEGSO_ZDT3 -0.0573 0.009  -0.0959 -0.0187  True
-----
```

Figura 5. Test-Tukey HSD para VEGSO e NSGA2 utilizando as funções ZDT1, ZDT2 e ZDT3



# Conclusão

- Os Algoritmos demonstram-se ferramentas uteis para solucionar problemas de otimização multiobjetivo. O algoritmo VEGSO ainda apresenta certas falhas, como o problema de crowding e tempo de processamento, mas demonstra ser uma modificação com potencial do algoritmo original GSO.

# Referências

- JAIMES, Antonio López; MARTINEZ, Saúl Zapotecas; COELLO, Carlos A. Coello. An introduction to multiobjective optimization techniques. **Optimization in Polymer Processing**, p. 29-57, 2009.
- ZHANG, Xingyi et al. An efficient approach to nondominated sorting for evolutionary multiobjective optimization. **IEEE Transactions on Evolutionary Computation**, v. 19, n. 2, p. 201-213, 2014.