Methodologies for Solving Complex Multi-Objective Combinatorial Problems in Engineering: An Evolutionary Approach Applied to Computer Networks

Prof. Yezid Donoso, Ph.D.

e-mail : ydonoso@uniandes.edu.co

2020 8th International Conference on Computers Communications and Control (ICCCC)

Outline

1. Single-objective vs Multi-Objective Optimization
2. Traditional Methods
3. Multi-Objective Evolutionary Algorithms
4. Conclusion

Outline

1. Single-objective vs Multi-Objective Optimization
2. Traditional Methods
3. Multi-Objective Evolutionary Algorithms
4. Conclusion

Single-objective vs Multi-Objective Optimization

- In real problems in Engineering, solving a problem is not enough; the solution of the problem must be the best solution possible. In other words, it is necessary to find the optimal solution.

- The solution is the best possible solution because in the real world, this problem may have certain constraints by which the solutions found may be feasible, that is, they can be implemented in practice and, unfeasible or that they cannot be implemented.

- Some of these problems in engineering can be MOP (Multi-Objective Optimization Problem)..

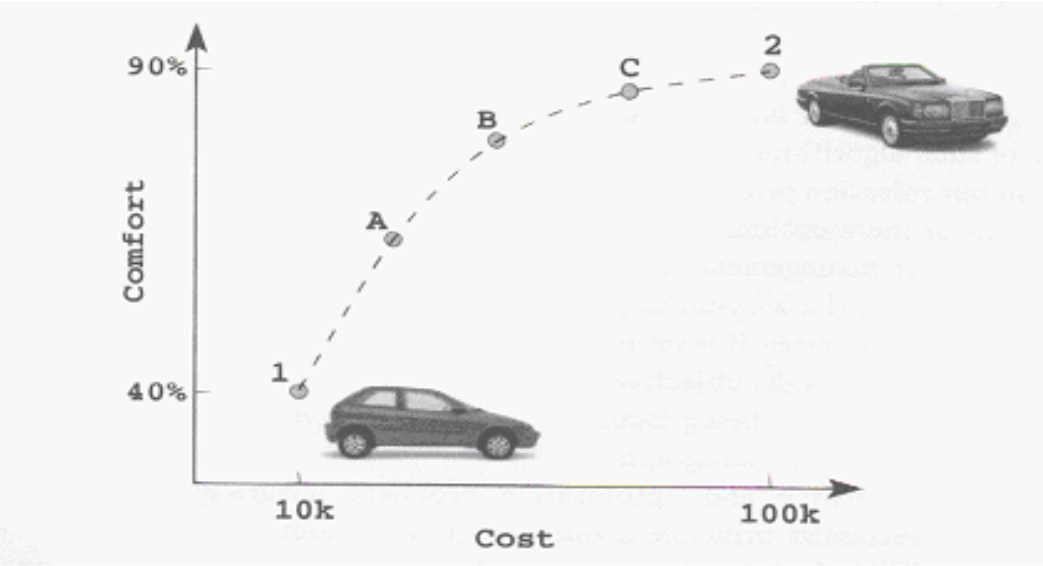
Single-objective vs Multi-Objective Optimization

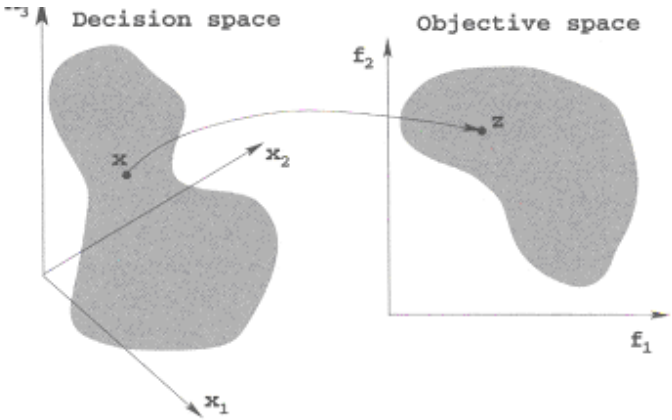
- A general MOP includes a set of n parameters (decision variables), a set of k objective functions and a set of m restrictions.

- Then the MOP can be expressed as:

Optimize	$Y = F(X) = (f_1(X), f_2(X), \ldots , f_k(X))$
Subject to	$E(X) = (e_1(X), e_2(X), \ldots , e_m(X)) \leq 0$
Where	$X = (x_1, x_2, \ldots , x_n) \in X$
	$Y = (y_1, y_2, \ldots , y_k) \in Y$

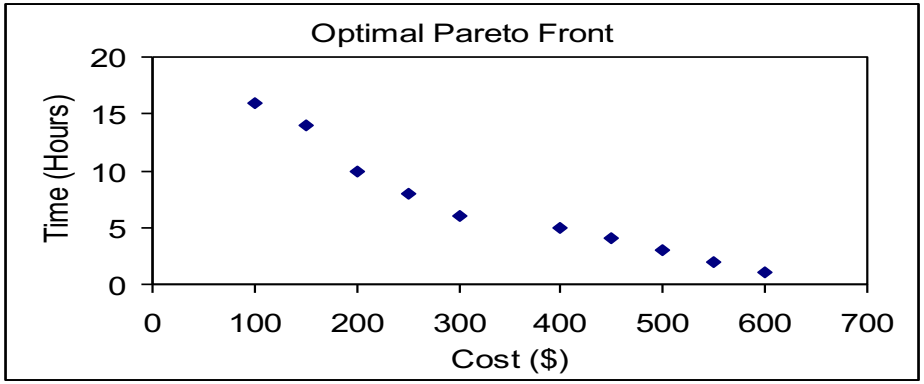
Single-objective vs Multi-Objective Optimization



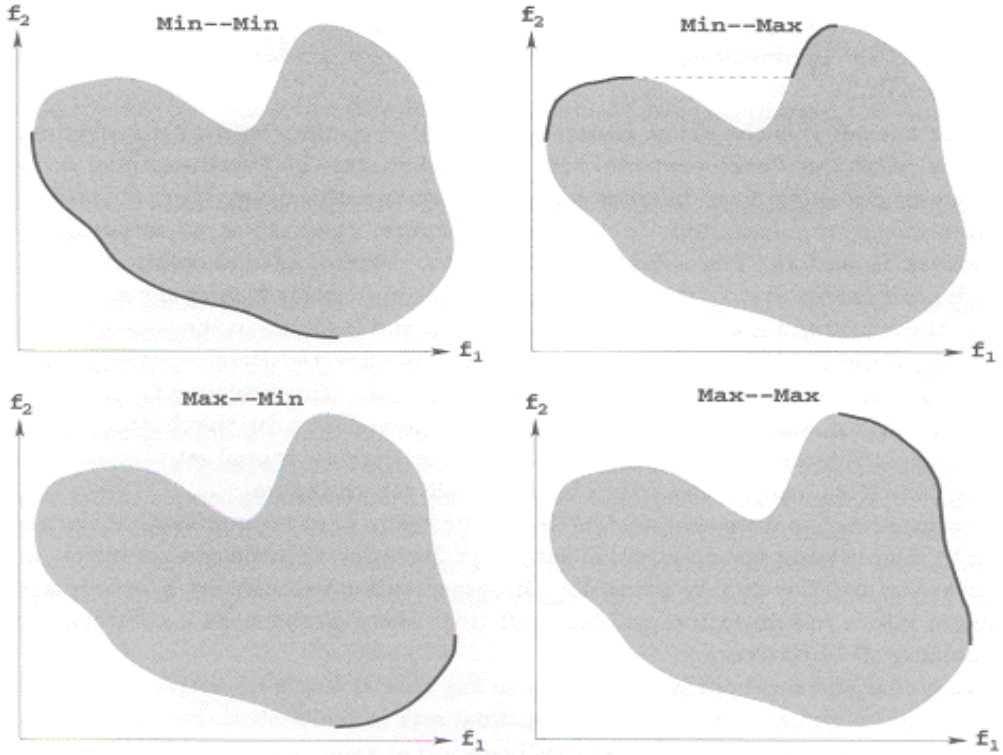


Single-objective vs Multi-Objective Optimization

Opción	1	2	3	4	5	6	7	8	9	10
Cost ($)	600	550	500	450	400	300	250	200	150	100
Time (Hours)	1	2	3	4	5	6	8	10	14	16



Single-objective vs Multi-Objective Optimization

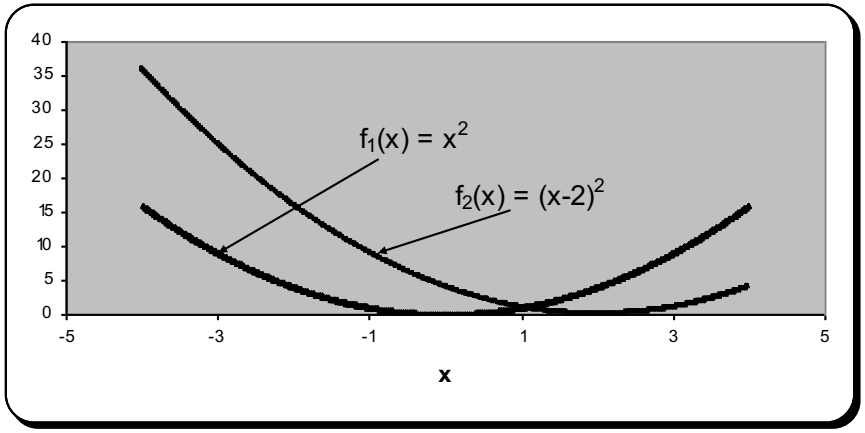


Single-objective vs Multi-Objective Optimization

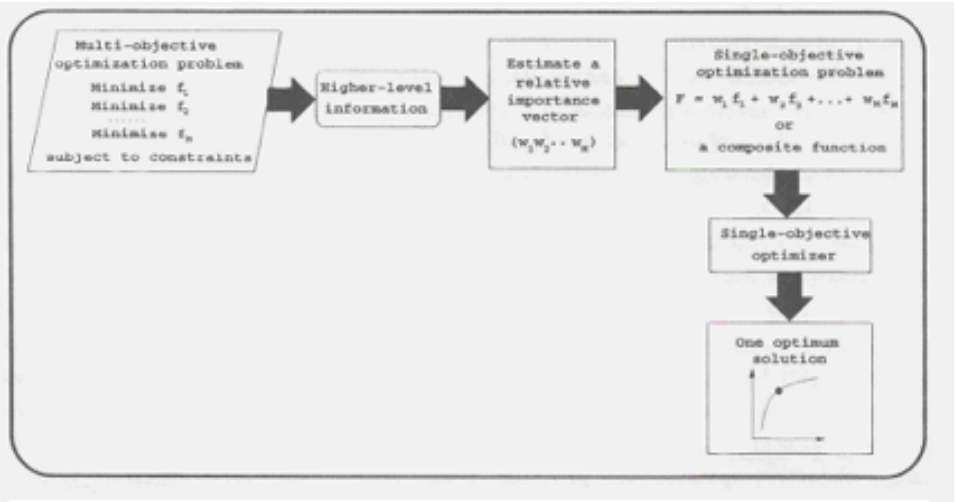
Minimize $f(x) = (f_1(x), f_2(x))$

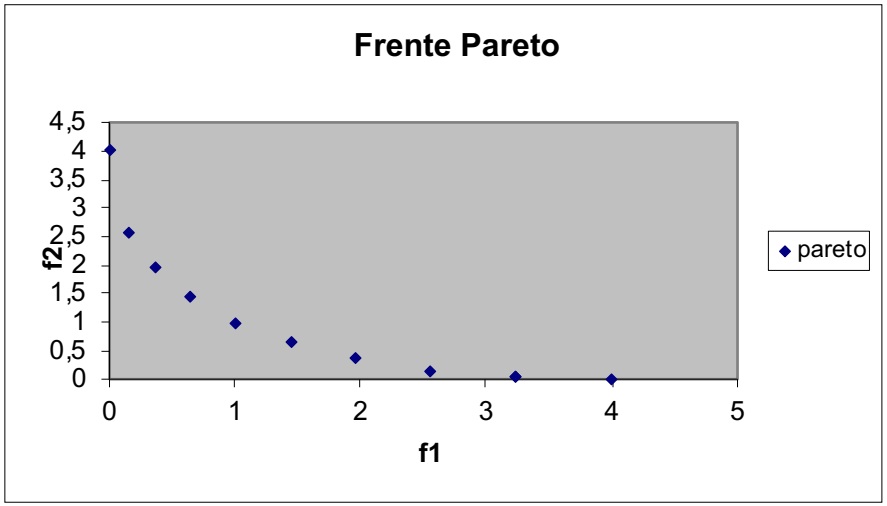
Suject to $f_1(x) = x^2$,

$f_2(x) = (x - 2)^2$



Single-objective vs Multi-Objective Optimization





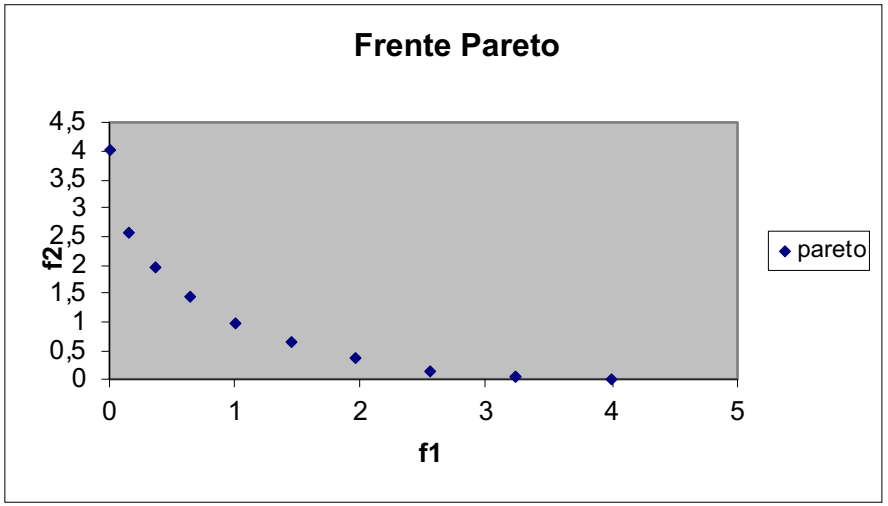
Single-objective vs Multi-Objective Optimization

Minimize $f(x) = (f_1(x), f_2(x))$

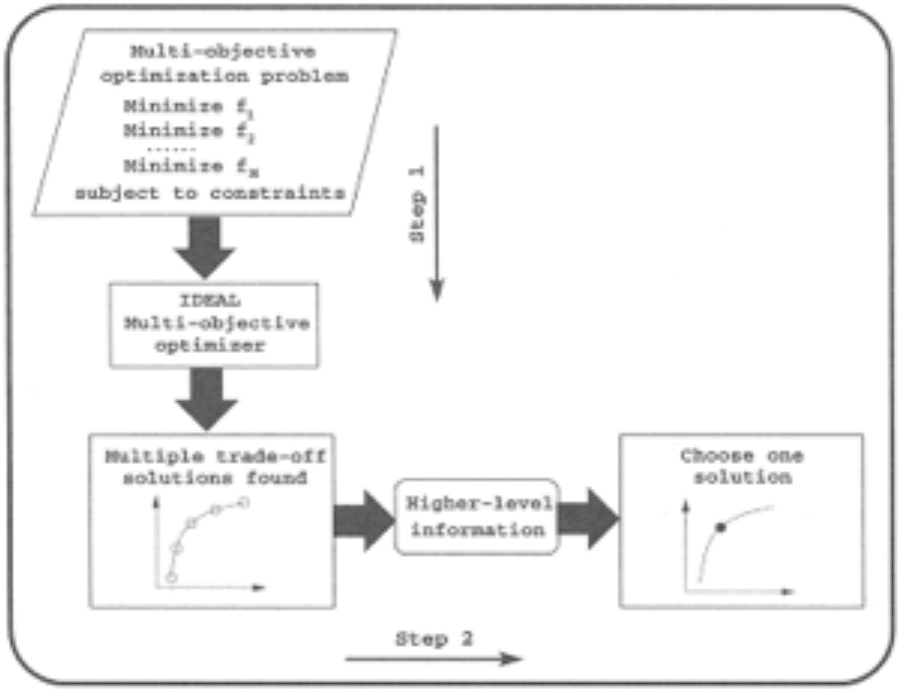
Suject to $f_1(x) = x^2$,

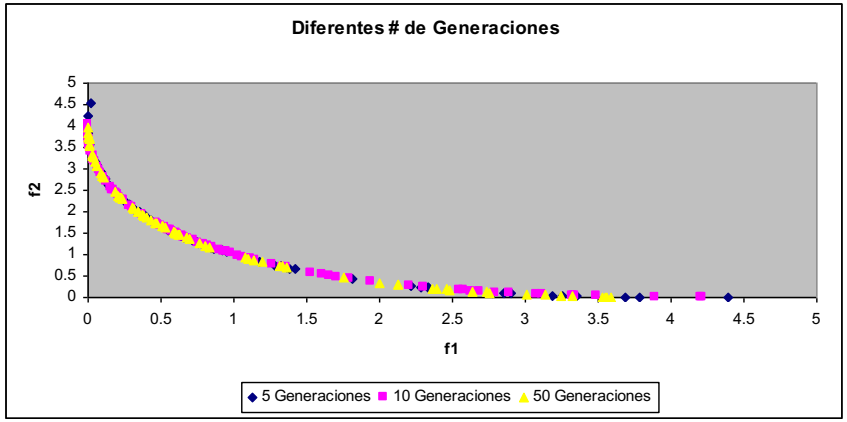
$f_2(x) = (x - 2)^2$

Datos a través del Modelo Analítico mediante pesos					
Valores Optimos					
x	w1	w2	f1	f2	f
0	0	1	4	0	0
1,8	0,1	0,9	3,24	0,04	0,36
1,6	0,2	0,8	2,56	0,16	0,64
1,4	0,3	0,7	1,96	0,36	0,84
1,2	0,4	0,6	1,44	0,64	0,96
1	0,5	0,5	1	1	1
1,2	0,6	0,4	0,64	1,44	0,96
1,4	0,7	0,3	0,36	1,96	0,84
1,6	0,8	0,2	0,16	2,56	0,64
1,8	0,9	0,1	3,24	0,04	2,92
2	1	0	0	4	0



Single-objective vs Multi-Objective Optimization





Single-objective vs Multi-Objective Optimization

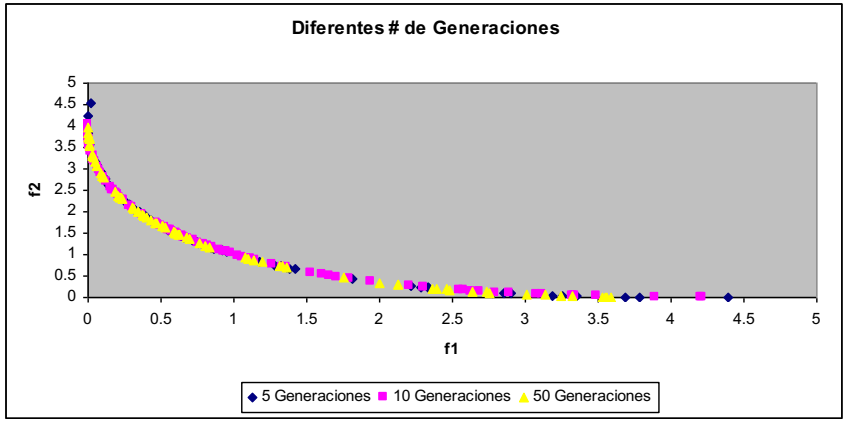
Minimize $f(x) = (f_1(x), f_2(x))$

Suject to $f_1(x) = x^2$,

$f_2(x) = (x - 2)^2$

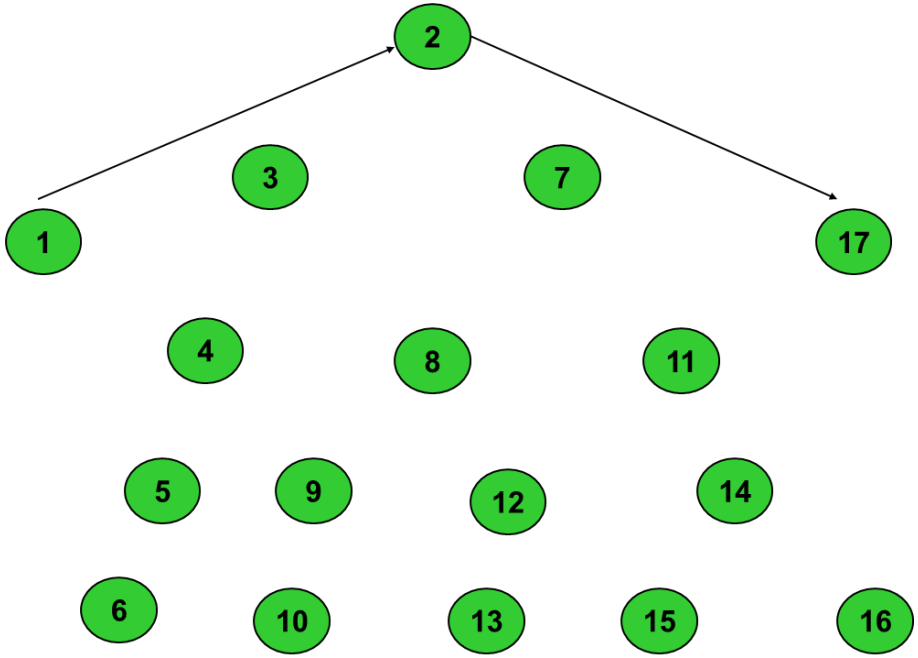
```
Procedimiento GA( )
Inicio
t = 0 /* empezar con un tiempo inicial */
InicializarPoblacion P (t) /* inicializar una población de individuos
generados al azar */
Evaluate P (t) /* evaluar el fitness de todos los individuos de la
población inicial */
Mientras no se cumpla el criterio de parada
        t = t + 1 /* incrementar el contador de tiempo */
        P' = SeleccionarPadres P (t) /* seleccionar una
            poblacion para generar  descendientes */
        Recombinar P' (t) /* recombinar los "genes" del
            grupo de padres seleccionados */
        Mutar P' (t) /* perturbar la población generada de
            manera estocástica */
        Evaluar P' (t) /* calcular fitness de la población
            recién creada */
        P = Sobrevivientes P, P' (t)/* Seleccionar
            sobrevivientes para la siguiente generación */
Fin Mientras
Fin
```

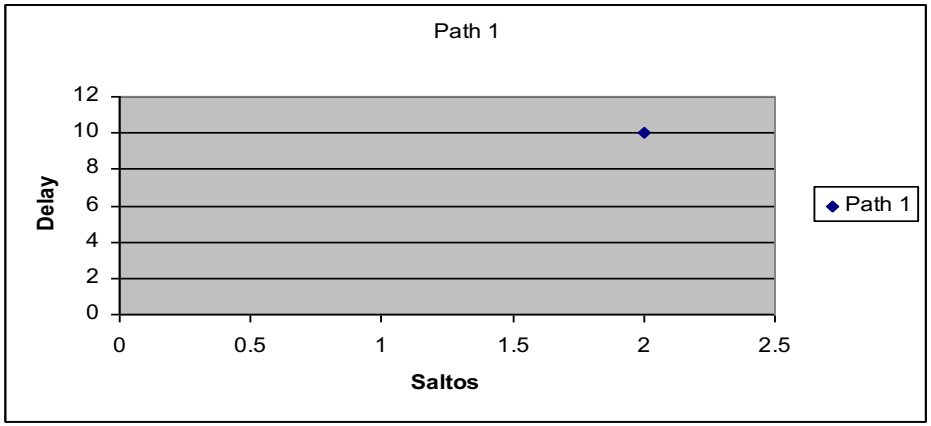
Algoritmo genético básico



Single-objective vs Multi-Objective Optimization

Path 1 ➔ Hops = 2 & Delay = 10

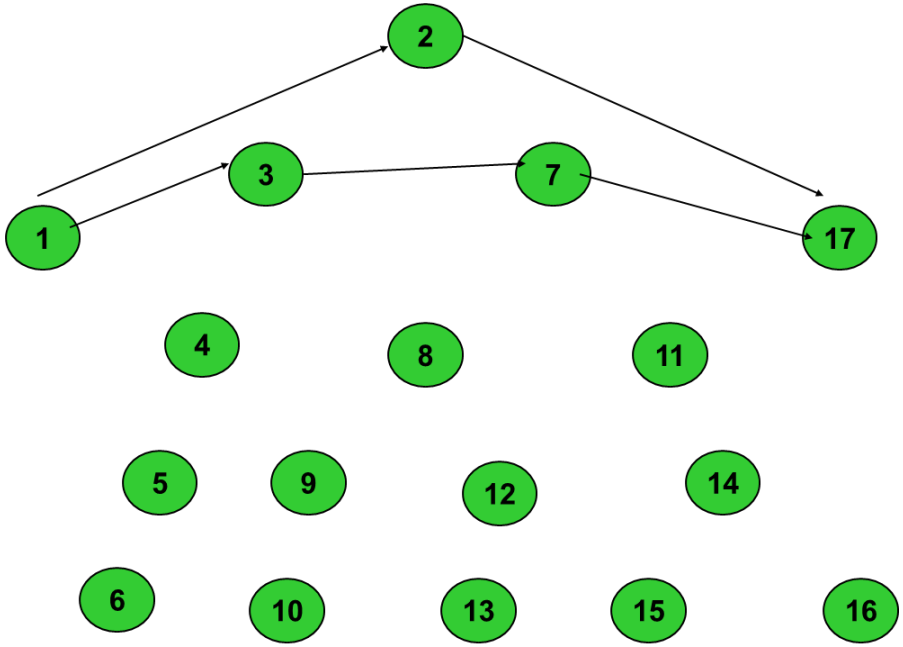


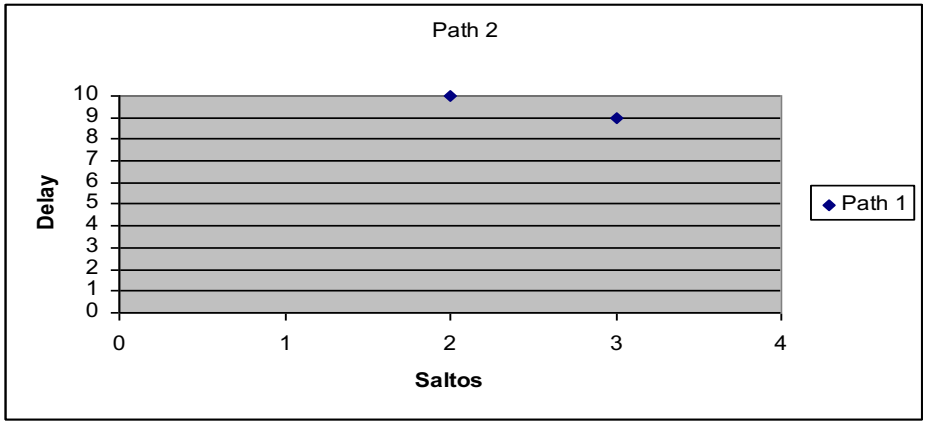


Single-objective vs Multi-Objective Optimization

Path 1 ➔ Hops = 2 & Delay = 10

Path 2 ➔ Hops = 3 & Delay = 9



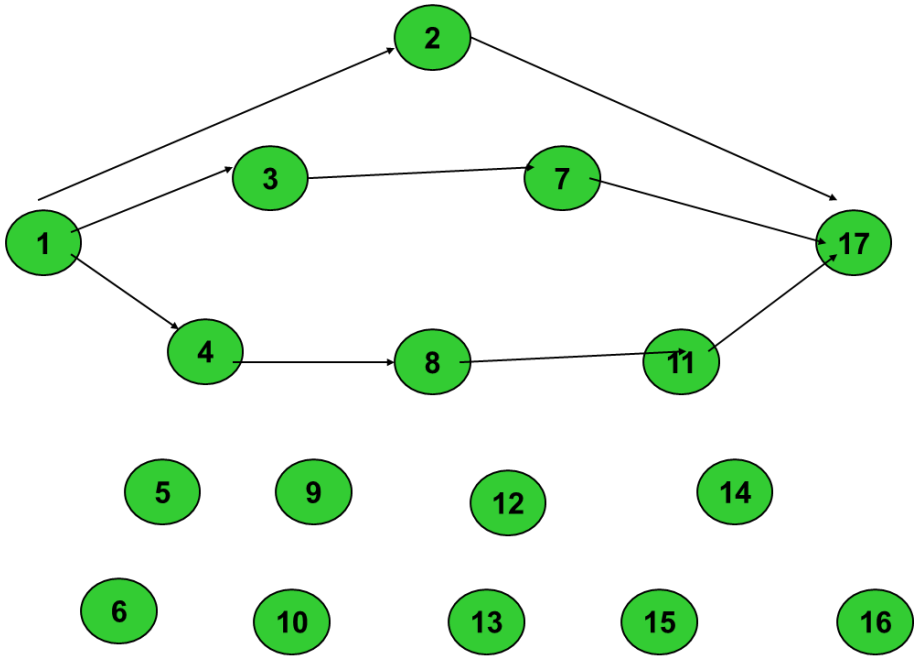


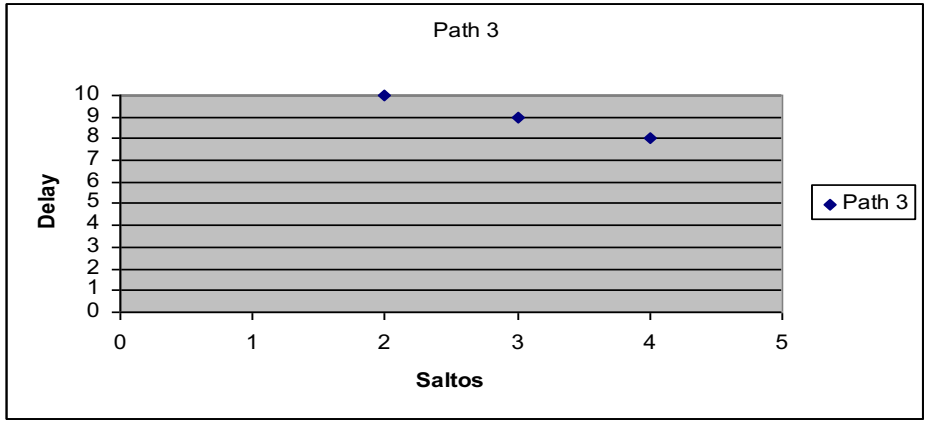
Single-objective vs Multi-Objective Optimization

Path 1 ➔ Hops = 2 & Delay = 10

Path 2 ➔ Hops = 3 & Delay = 9

Path 3 ➔ Hops = 4 & Delay = 8





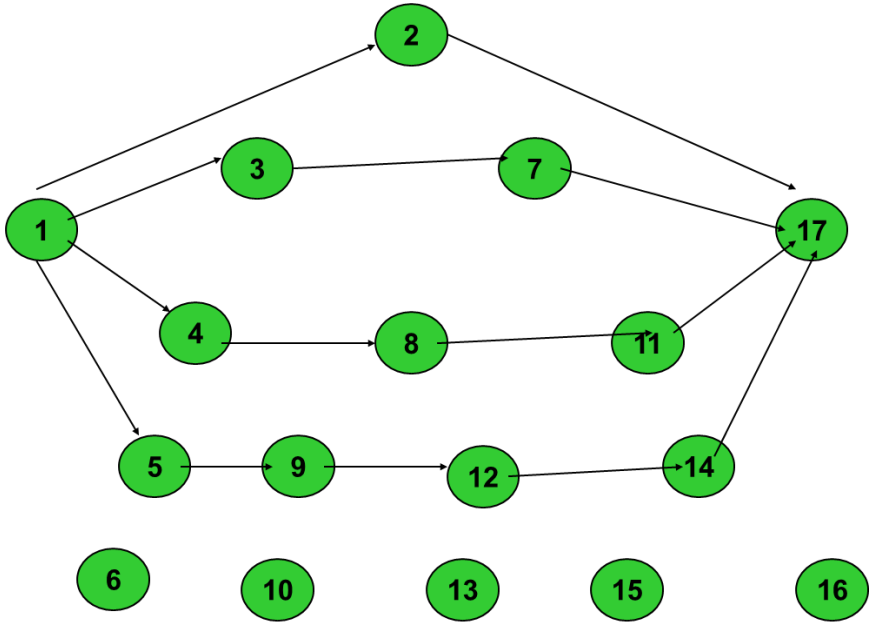
Single-objective vs Multi-Objective Optimization

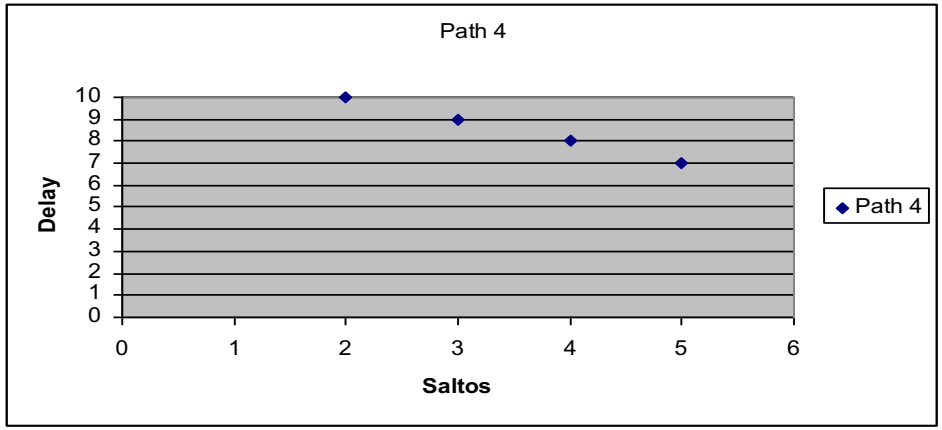
Path 1 ➔ Hops = 2 & Delay = 10

Path 2 ➔ Hops = 3 & Delay = 9

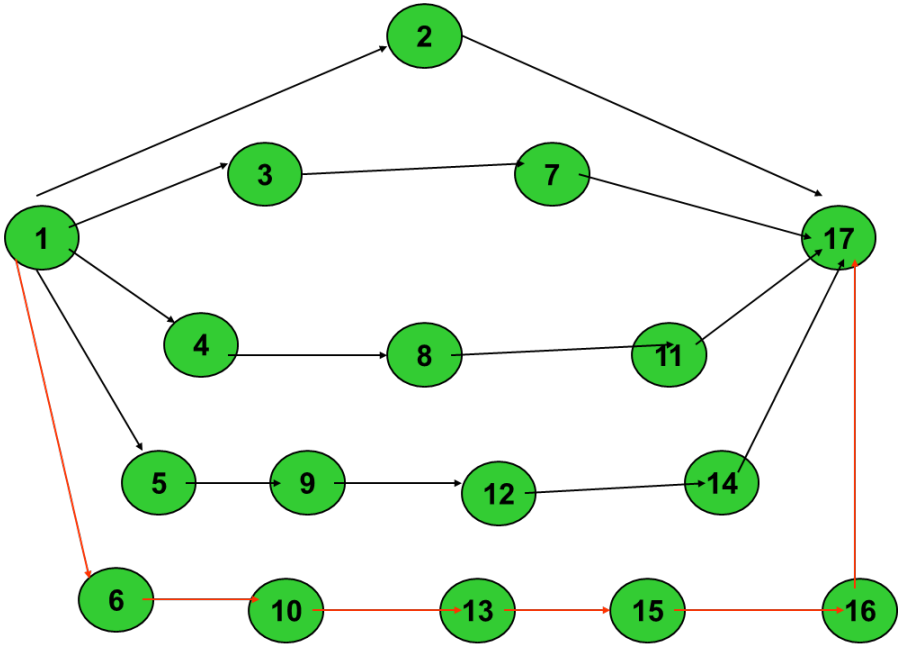
Path 3 ➔ Hops = 4 & Delay = 8

Path 4 ➔ Hops = 5 & Delay = 7





Single-objective vs Multi-Objective Optimization



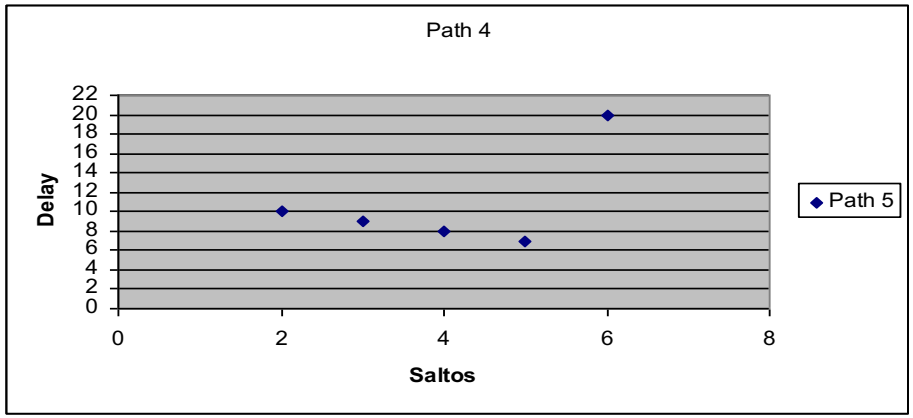
Path 1 ➔ Hops = 2 & Delay = 10

Path 2 ➔ Hops = 3 & Delay = 9

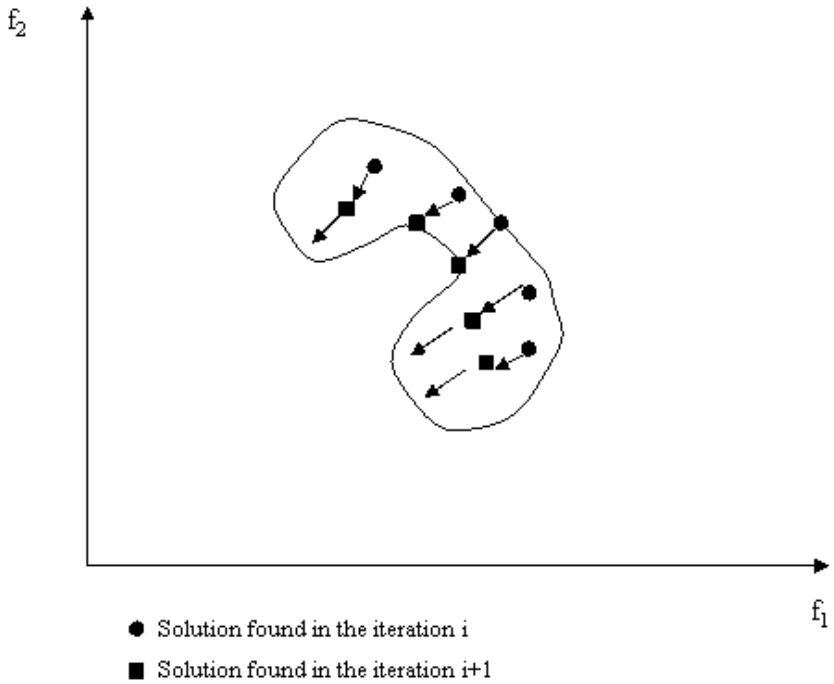
Path 3 ➔ Hops = 4 & Delay = 8

Path 4 ➔ Hops = 5 & Delay = 7

Path 5 ➔ Hops = 6 & Delay = 20

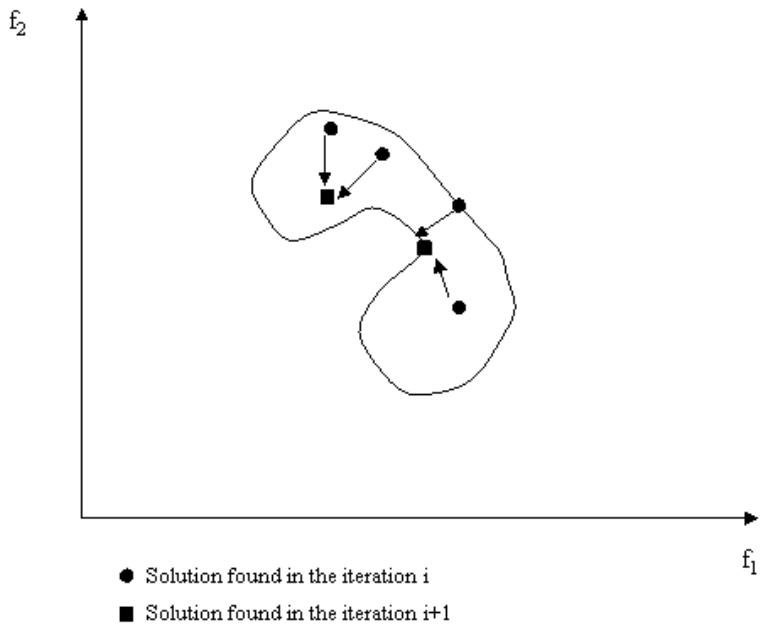


Single-objective vs Multi-Objective Optimization



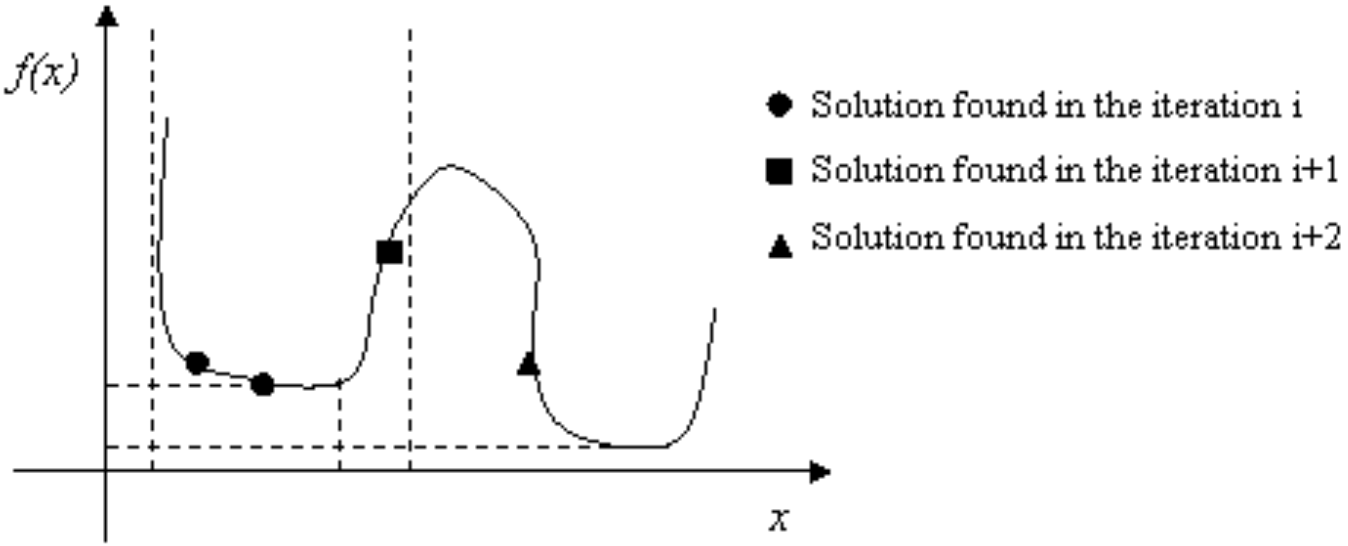
Convergence towards optimal

Single-objective vs Multi-Objective Optimization



Solution in Non-Convexity

Single-objective vs Multi-Objective Optimization



Global Optimum

Outline

1. Single-objective vs Multi-Objective Optimization
2. Traditional Methods
3. Multi-Objective Evolutionary Algorithms
4. Conclusion

Traditional Methods - Weighted Sum

Optimize [Minimize / Maximize]

$$F'(X) = \sum_{i=1}^{n} r_i * f_i(X)$$

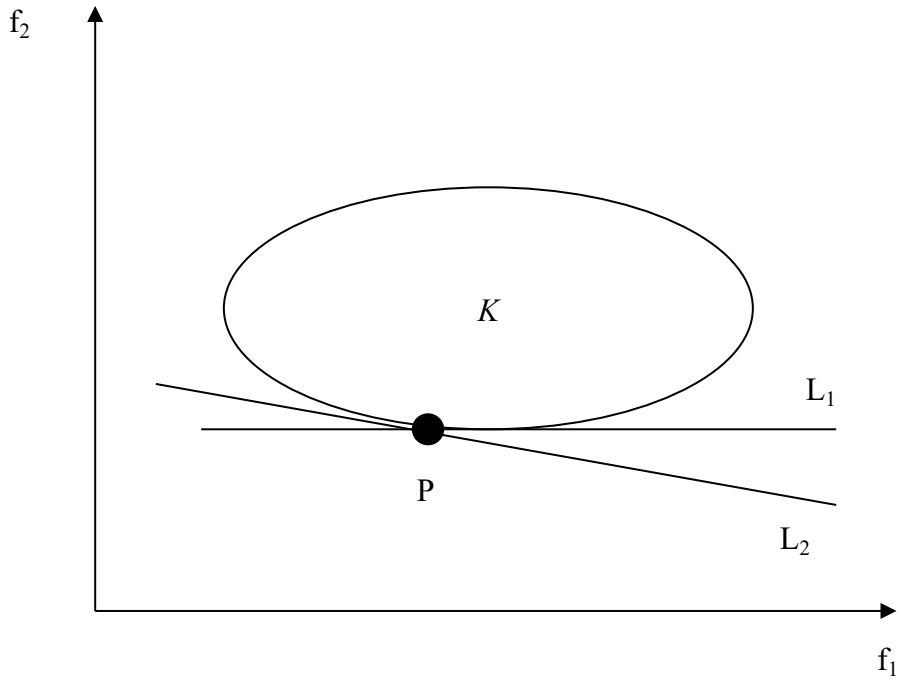
Subject to

$H(X) = 0$

$G(X) \geq 0$

$0 \leq r_i \leq 1,\ i = \{1, .., n\}$

$$\sum_{i=1}^{n} r_i = 1$$



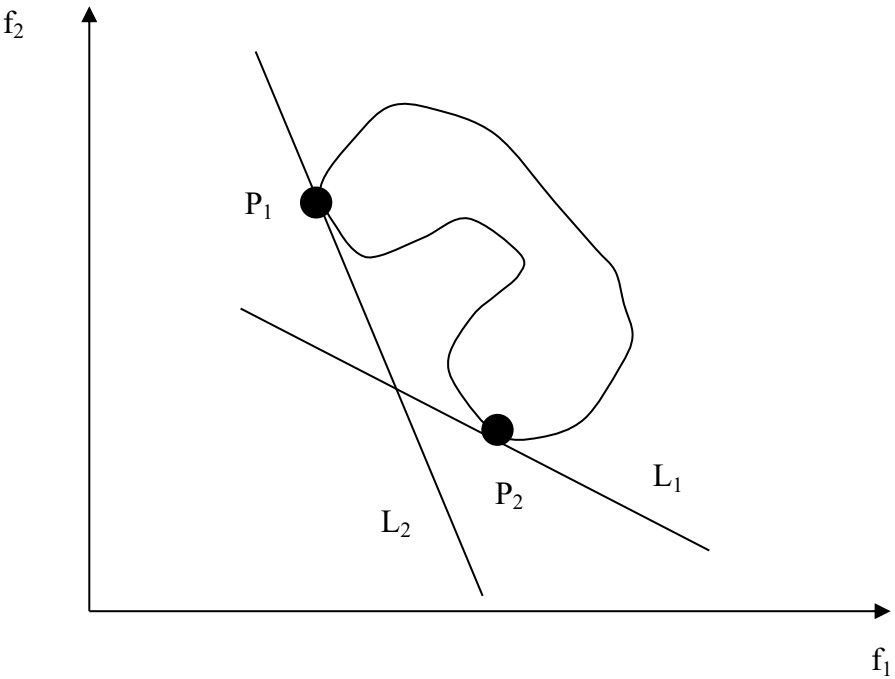
Traditional Methods - Weighted Sum

Optimize [Minimize / Maximize]

$$F'(X) = \sum_{i=1}^{n} r_i * f_i(X)$$

Subject to

$$H(X) = 0$$
$$G(X) \geq 0$$
$$0 \leq r_i \leq 1,\ i = \{1, .., n\}$$
$$\sum_{i=1}^{n} r_i = 1$$



Traditional Methods - Weighted Sum

$$F\left(X_{ij}^{f}\right)=\left[\sum_{f\in F}\sum_{(i,j)\in E}X_{ij}^{f}\ ,\ \sum_{f\in F}\sum_{(i,j)\in E}d_{ij}.X_{ij}^{f}\right]$$

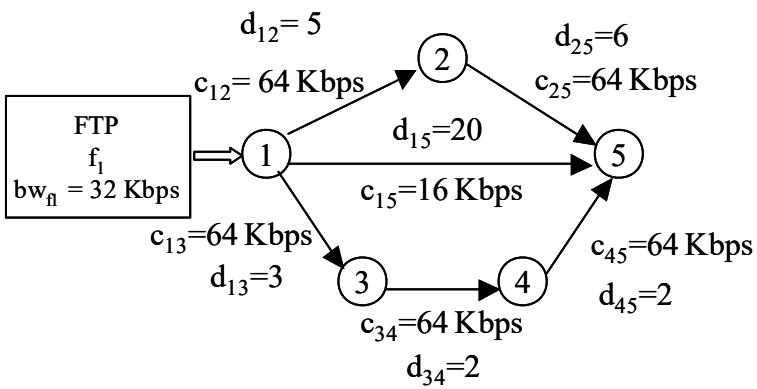
$$\sum_{(i,j)\in E}X_{ij}^{f}=1\quad ,f\in F,i=s$$

$$\sum_{(j,i)\in E}X_{ji}^{f}=-1\quad ,i=t,f\in F$$

$$\sum_{(i,j)\in E}X_{ij}^{f}-\sum_{(j,i)\in E}X_{ji}^{f}=0\quad ,f\in F,i\neq s,i\neq t$$

$$\sum_{f\in F}bw_{f}.X_{ij}^{f}\leq c_{ij}\quad ,(i,j)\in E$$

$$X_{ij}^{f}=\{0,1\}$$



Traditional Methods - Weighted Sum

$$\min Z = r_1 . \sum_{f \in F} \sum_{(i,j) \in E} X_{ij}^f + r_2 . \sum_{f \in F} \sum_{(i,j) \in E} d_{ij} . X_{ij}^f$$

Subject to

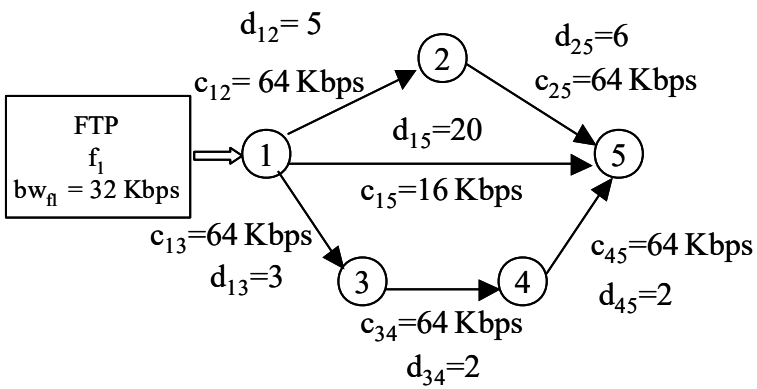
$$\sum_{(i,j) \in E} X_{ij}^f = 1 \quad , f \in F, i = s$$

$$\sum_{(j,i) \in E} X_{ji}^f = -1 \quad , i = t, f \in F$$

$$\sum_{(i,j) \in E} X_{ij}^f - \sum_{(j,i) \in E} X_{ji}^f = 0 \quad , f \in F, i \neq s, i \neq t$$

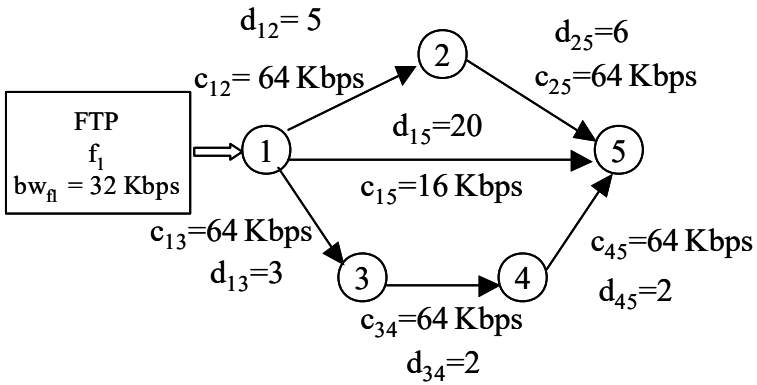
$$\sum_{f \in F} bw_f . X_{ij}^f \leq c_{ij} \quad , (i,j) \in E$$

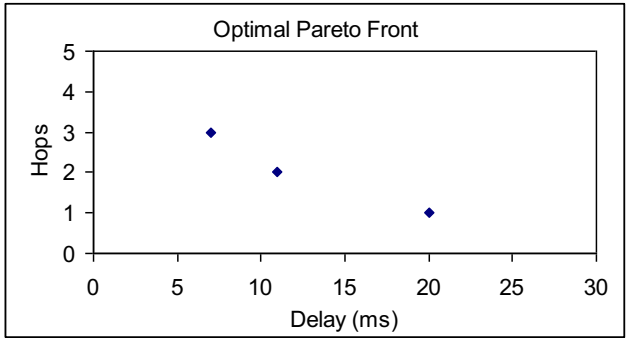
$$X_{ij}^f = \{0,1\}, 0 \leq r_i \leq 1, \sum_{i=1}^{2} r_i = 1$$



Traditional Methods - Weighted Sum

Sol	r1	r2	F1 (hops)	f2 (ms)	f4 (Kbps)	Path
1	1	0	2	11	64	{(1,2),(2,5)}
2	0	1	3	7	96	{(1,3),(3,4),(4,5)}
3	0.5	0.5	3	7	96	{(1,3),(3,4),(4,5)}
4	0.1	0.9	3	7	96	{(1,3),(3,4),(4,5)}
5	0.9	0.1	2	11	64	{(1,2),(2,5)}
6	0.8	0.2	2	11	64	{(1,2),(2,5)}
7	0.7	0.3	3	7	96	{(1,3),(3,4),(4,5)}





Traditional Methods - ε-Constraint

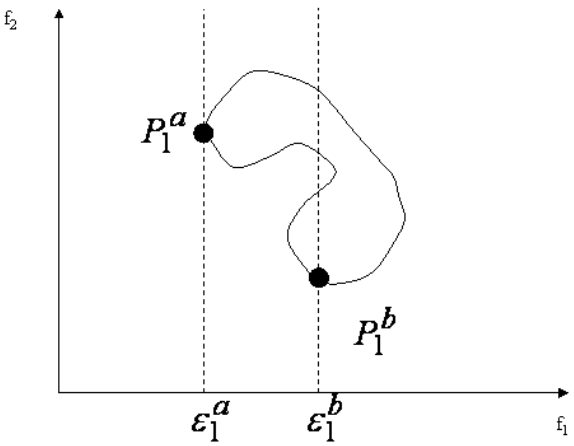
Minimize $f_h(x)$

Suject to $f_j(x) \leq \epsilon_j$

$g_i(x) \geq 0$

$h_k(x) = 0$

$x \in X_f$



Traditional Methods - Weighted Metrics

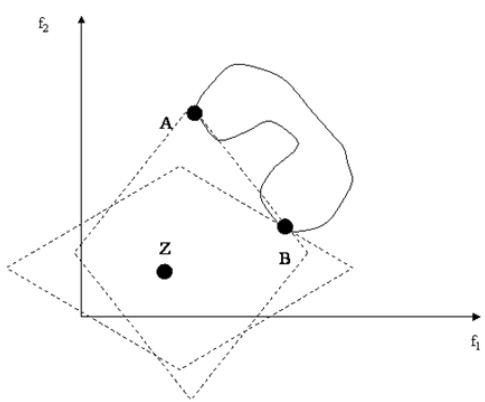
$$Min \quad l_p(x) = \left(\sum_{m=1}^{M} w_m \left| f_m(x) - z_m^* \right|^p \right)^{1/p}$$

$$SA$$

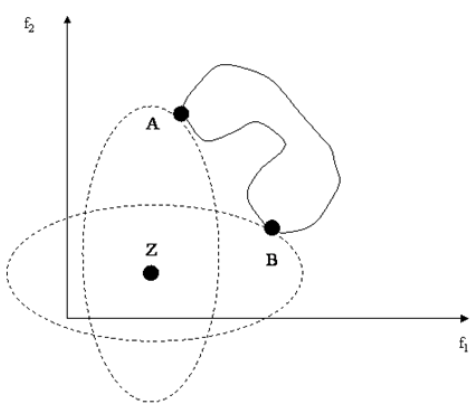
$$g_j(x) \geq 0$$

$$h_k(x) = 0$$

p = 1



p = 2



Traditional Methods - Benson

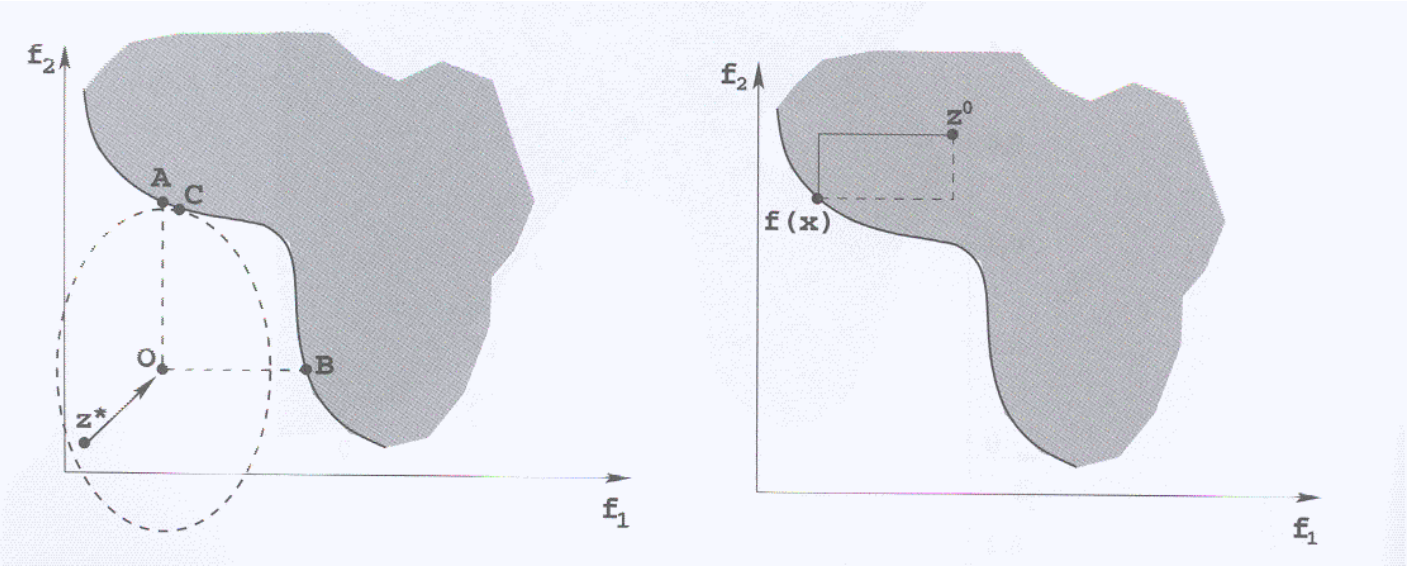
$$Max \quad \sum_{m=1}^{M} \max\left(0, \left(z_m^0 - f_m(x)\right)\right)$$

$$SA$$

$$f_m(x) \le z_m^0$$

$$g_j(x) \ge 0$$

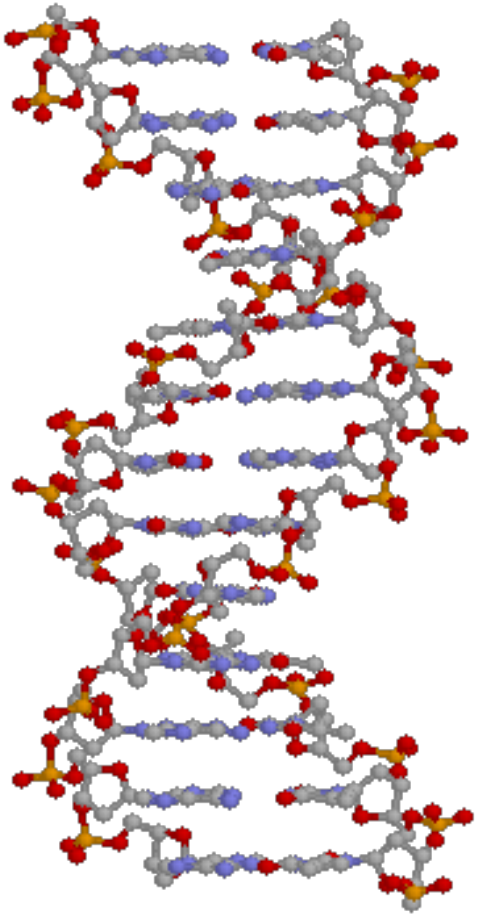
$$h_k(x) \ge 0$$



Outline

1. Single-objective vs Multi-Objective Optimization
2. Traditional Methods
3. Multi-Objective Evolutionary Algorithms
4. Conclusion

Multi-Objective Evolutionary Algorithms



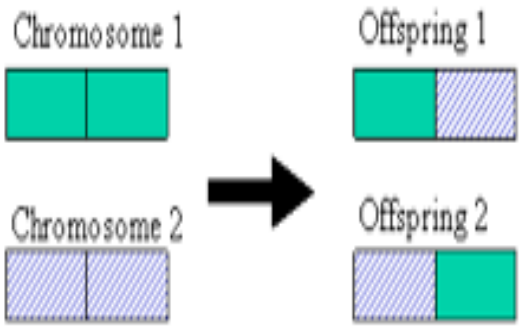
Multi-Objective Evolutionary Algorithms

EVOLUTIONARY ALGORITHM

```
Generate a random population M
while not max number of generation do
    Evaluate population according objective function
    Calculate the fitness of each of the individuals
    Classification Population based on fitness (M,M′)
    Generate New Population Mt+1
    Apply Binary Tournament Selection
    Apply Crossover Operator
    Mutation Operator
end while
Find Pareto Optimal Set.
```

Multi-Objective Evolutionary Algorithms

Cross Over Operator

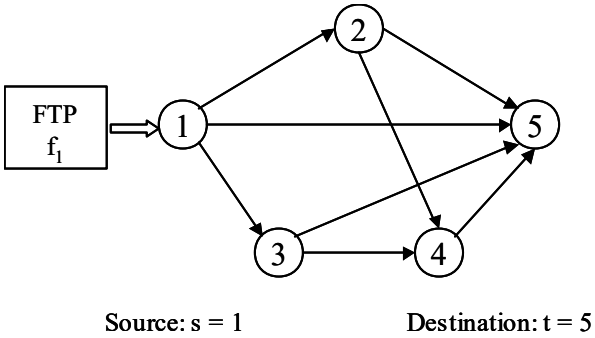


Multi-Objective Evolutionary Algorithms

Mutation Operator

Multi-Objective Evolutionary Algorithms

Cromosome

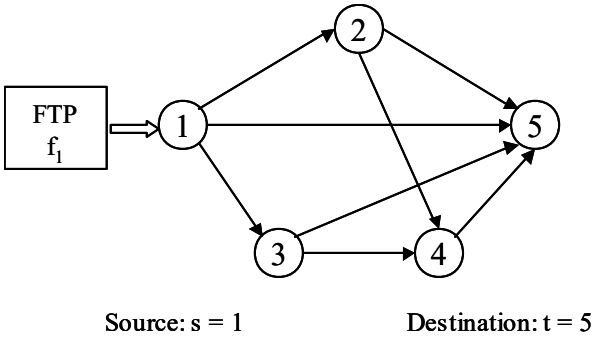


Path

s	n_1	n_2	...	n_i	...	t

Multi-Objective Evolutionary Algorithms

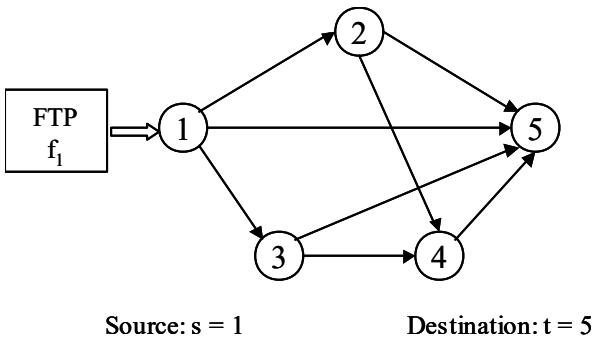
Cromosome

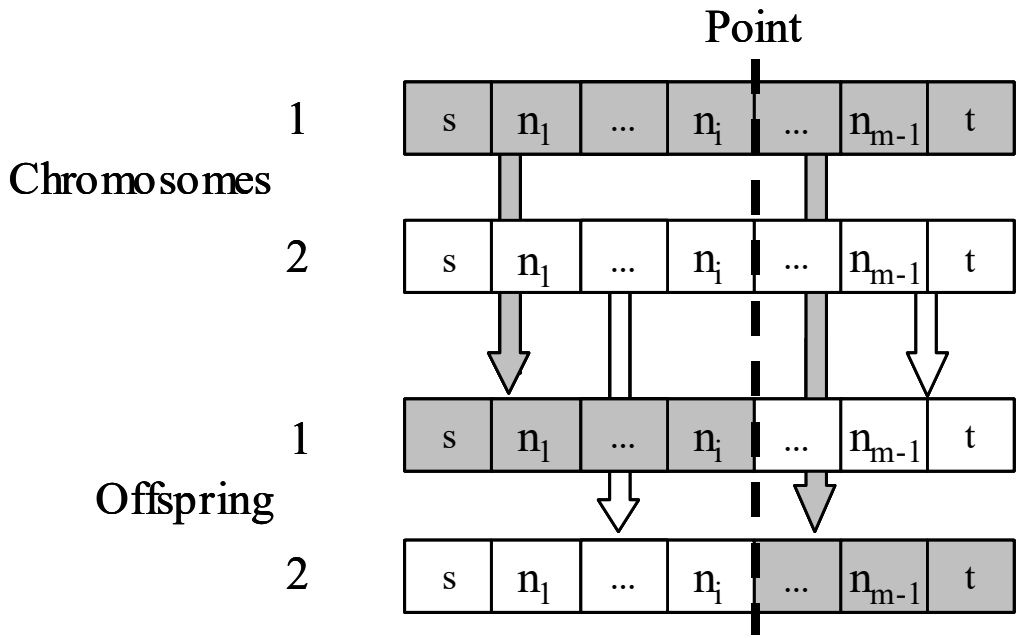


Path 1	1	2	5	
Path 2	1	5		
Path 3	1	3	4	5
Path 4	1	2	4	5
Path 5	1	3	5	

Multi-Objective Evolutionary Algorithms

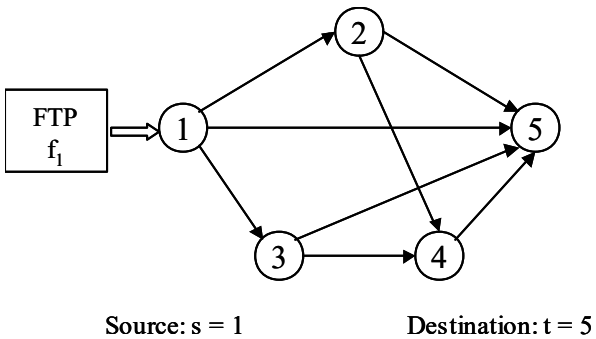
Cross Over Operator

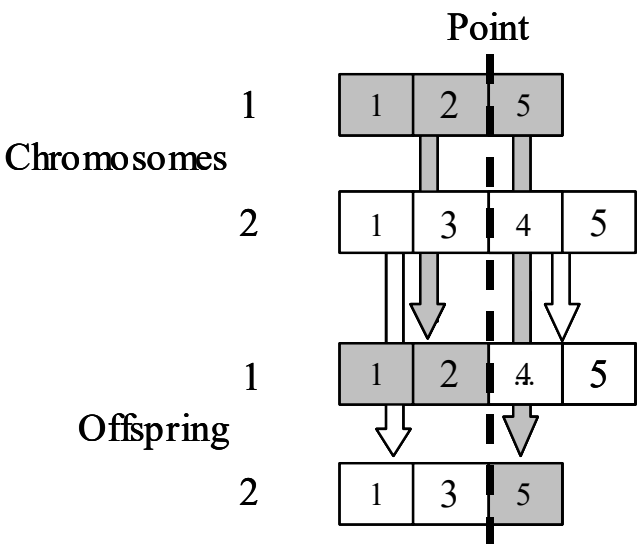




Multi-Objective Evolutionary Algorithms

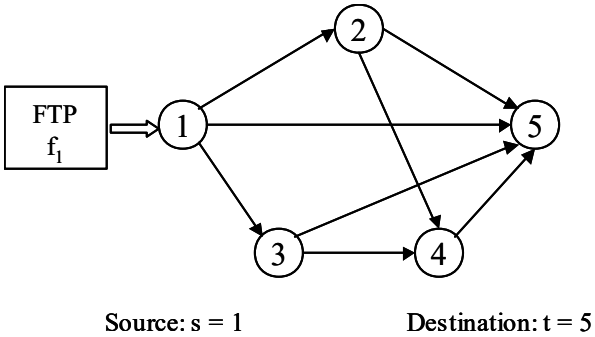
Cross Over Operator

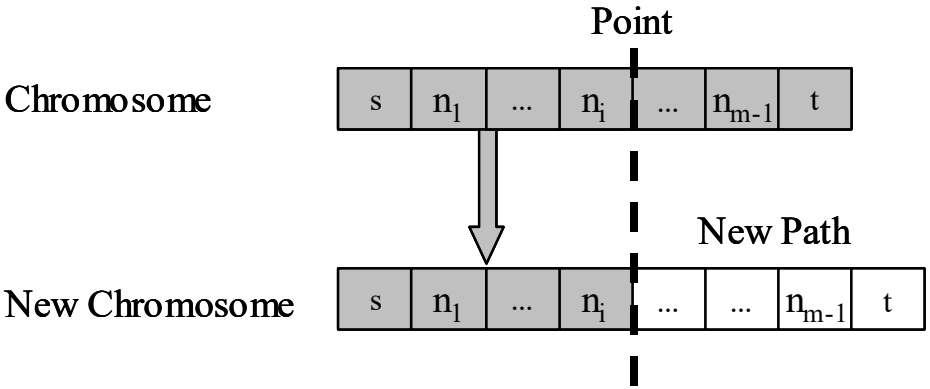




Multi-Objective Evolutionary Algorithms

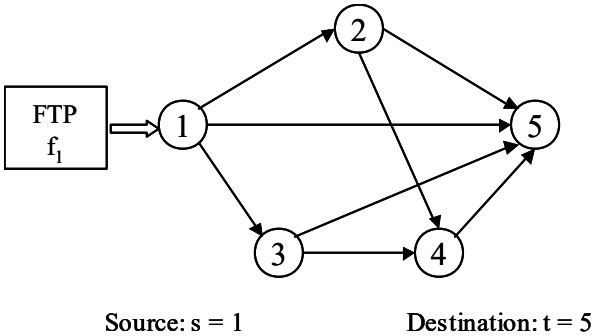
Mutation Operator

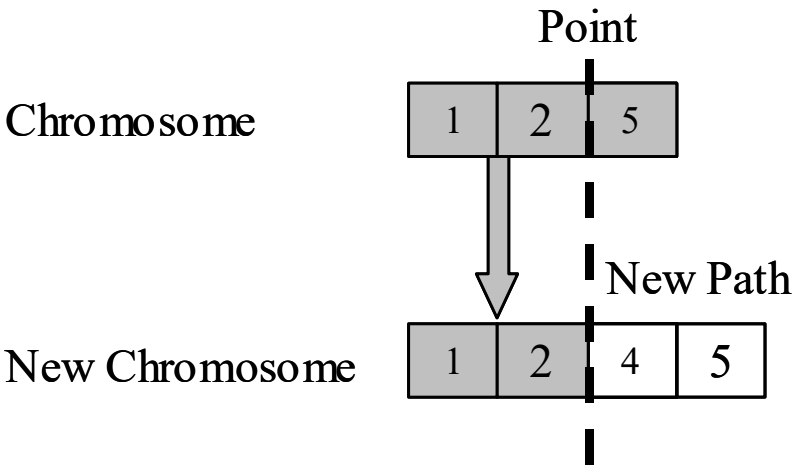




Multi-Objective Evolutionary Algorithms

Mutation Operator





Multi-Objective Evolutionary Algorithms

Algorithms

MOEA No Elitimns

- VEGA – Vector Evaluated Genetic Algorithm
- VOES – Vector-Optimized Evolution Strategy
- WBGA – Weight-Based Genetic Algorithm
- RWGA – Random Weighted Genetic Algorithm
- MOGA – Multiple Objective Genetic Algorithm
- NSGA – Non-Dominated Sorting Genetic Algorithm
- NPGA – Niched-Pareto Genetic Algorithm
- Predator-Prey Evolution Strategy

Multi-Objective Evolutionary Algorithms

Algorithms

Elitims MOEA

- Rudolph's Elitist Multi-Objective Evolutionary Algorithm
- NSGA-II – Elitist Non-Dominated Soroting Genetic Algorithm
- DPGA – Distance-Based Pareto Genetic Algorithm
- SPEA – Strength Pareto Evolutionary Algorithm
- Thermodynamical Genetic Algorithm
- PAES – Pareto-Archived Evolution Strategy
- MOMGA – Multi-Objective Messy Genetic Algorithm

Multi-Objective Evolutionary Algorithms

Algorithms

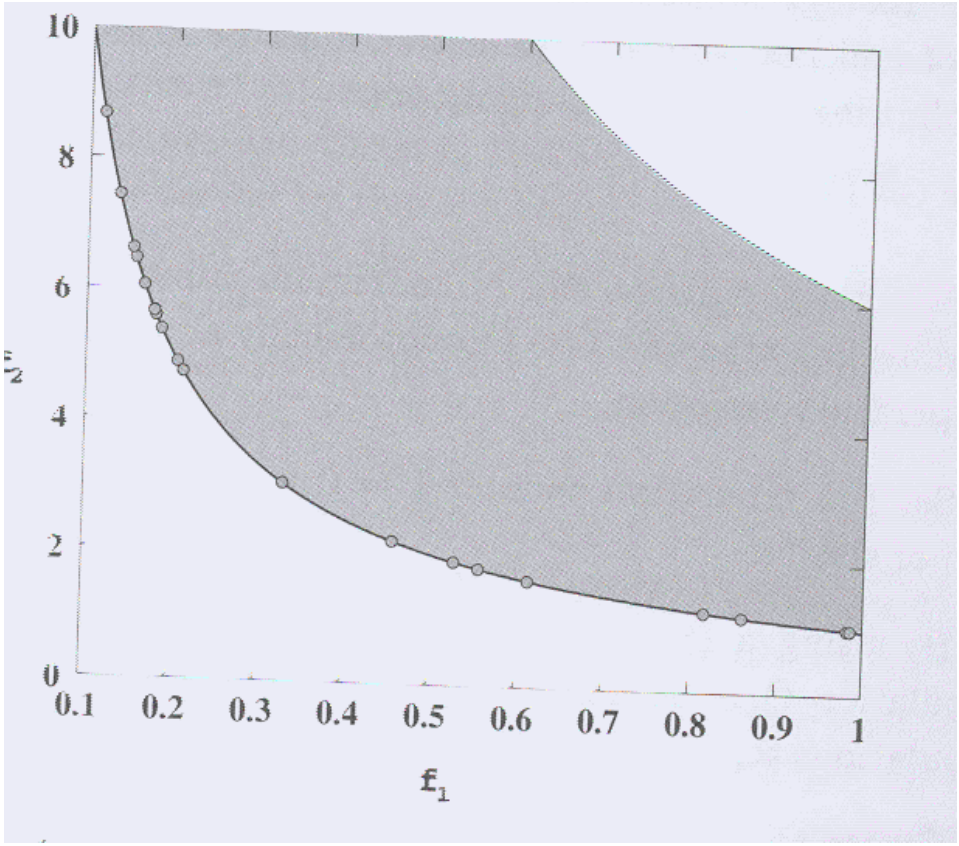
SPEA

Begin
Generate initial population P randomly. Make P feasible according to constrains
Generation $G = 1$
WHILE $G < G_{max}$
Calculate objective values for every solution in P
Find non-dominated solutions in P
Incorporate non-dominated solutions of P to P'
IF number of solutions in P' exceeds maximum number allowed
Purge P' using *clustering*
End IF
Calculate *fitness* for every solution in P and P'
Apply *selection* to $P + P'$
Apply *cross over* and *mutation* to generate new population P
Make P feasible according to constrains
$G = G + 1$
End WHILE
Print results in output file
End

Multi-Objective Evolutionary Algorithms

Algorithms

SPEA



Multi-Objective Evolutionary Algorithms

Algorithms

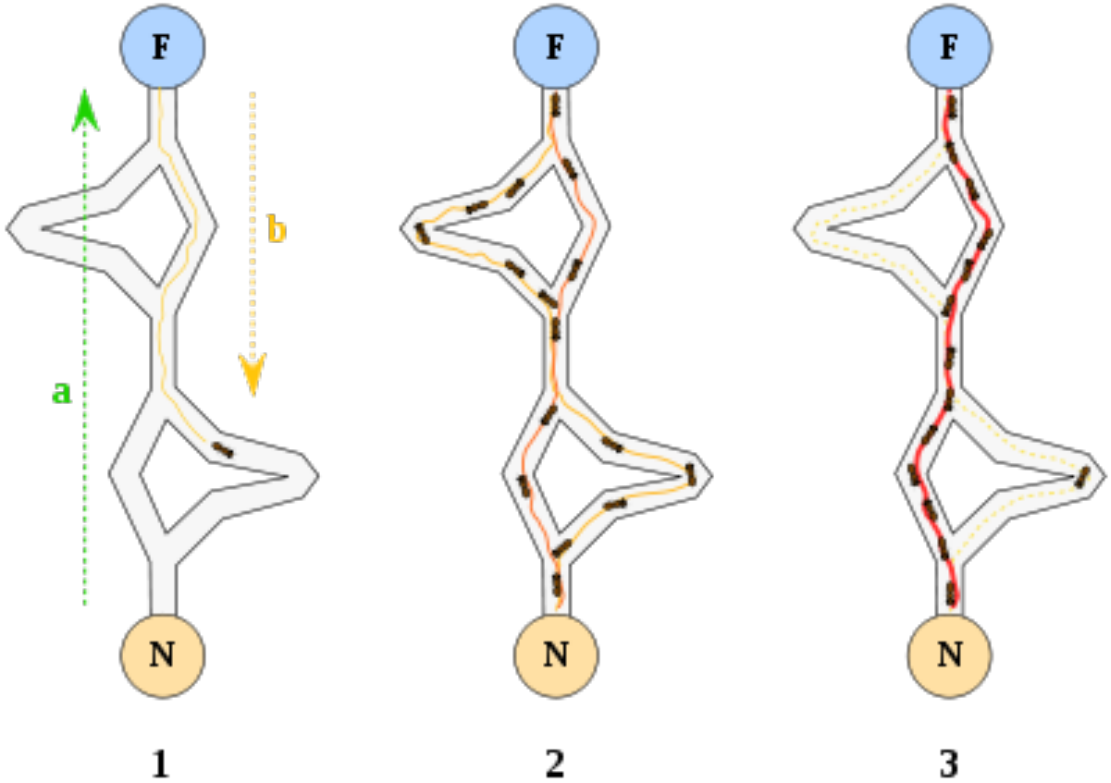
Simulated Annealing

```
Simulated_Annealing(T0,α,L,Tf)
  T <- T0
  Sact <- Generar_solución_inicial()
  While T ≥ Tf
      For cont: 1 to L(T)
        Scand <- Seleccionar_solución_N(Sact)
        δ <- costo(Scand)-costo(Sact)
        If (U(0,1) < e^(-δ/T)) or (δ < 0) Then
          Scand <- Sact
        End_If
      End_For
      T <- α(T)
  End_While
```

Multi-Objective Evolutionary Algorithms

Algorithms

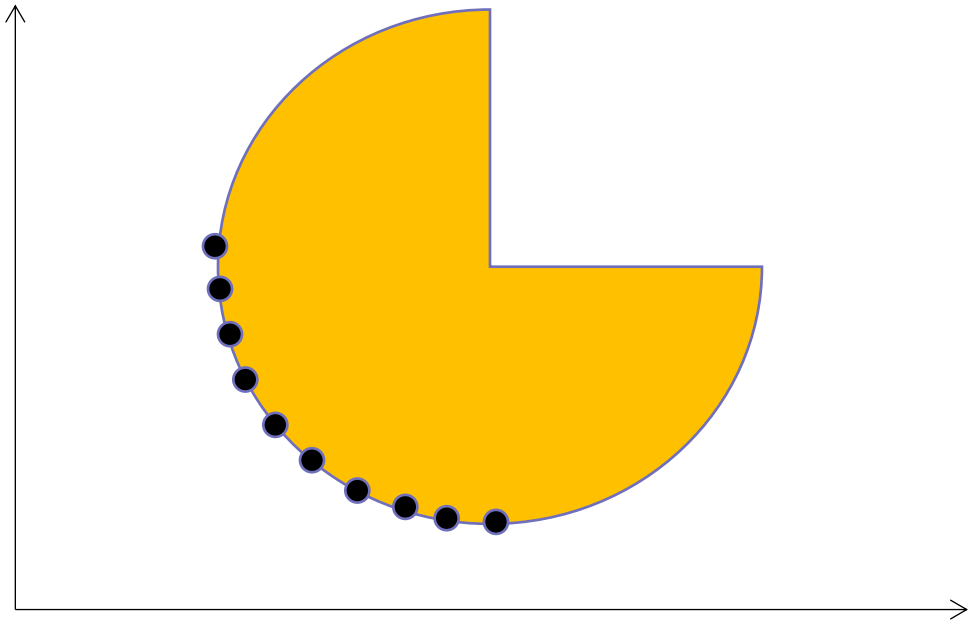
Ant Colony



Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

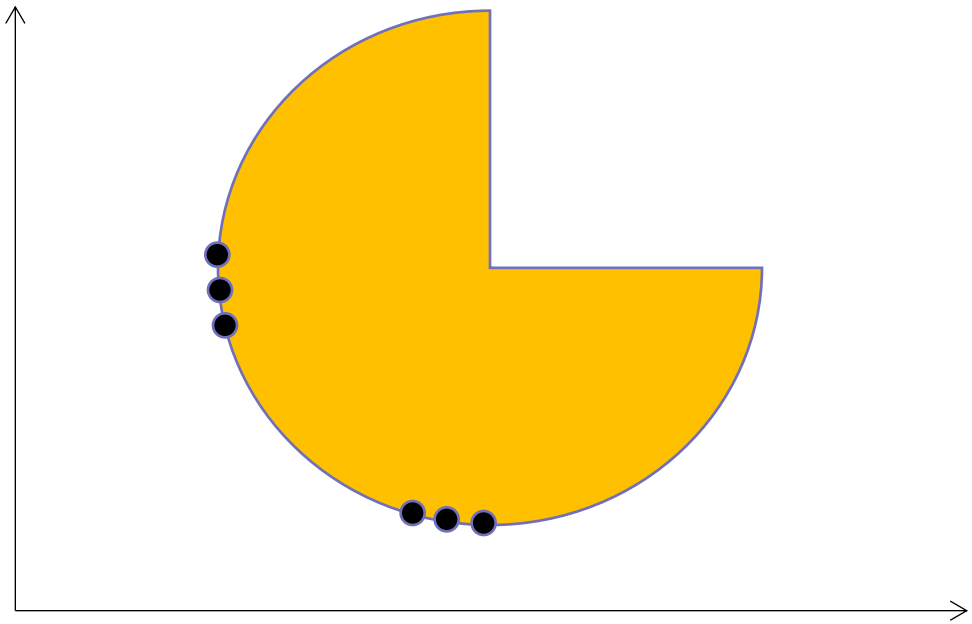
- **Ideal set of non-dominated Solutions**



Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

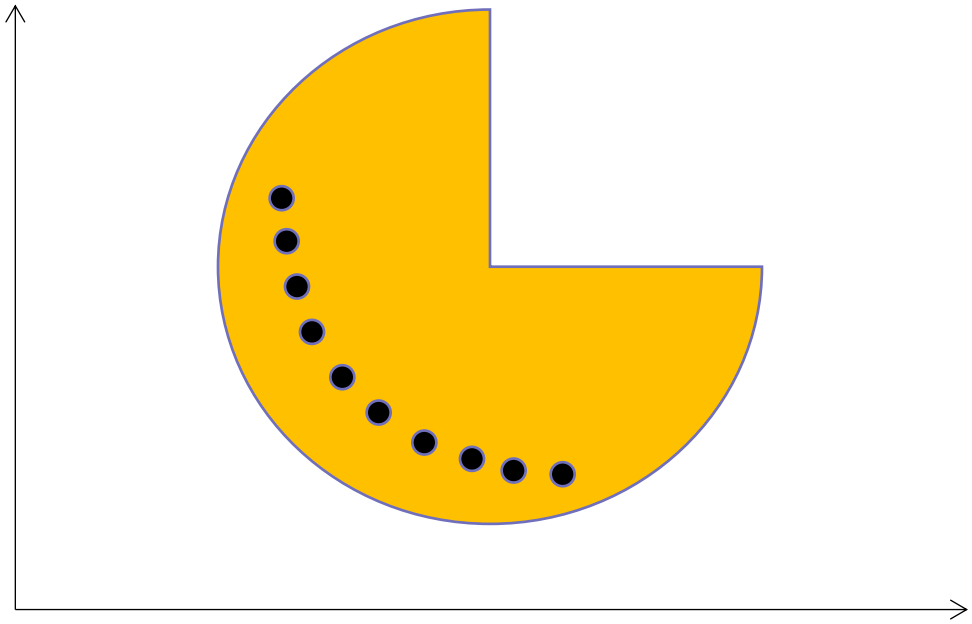
- Good convergence but poor distribution



Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

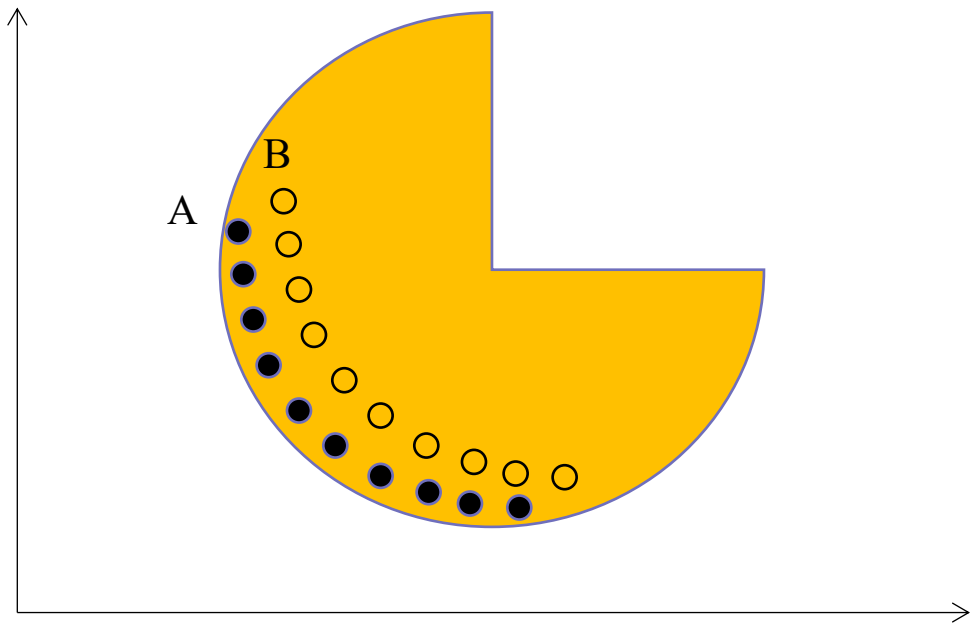
- Convergence poor but good distribution



Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

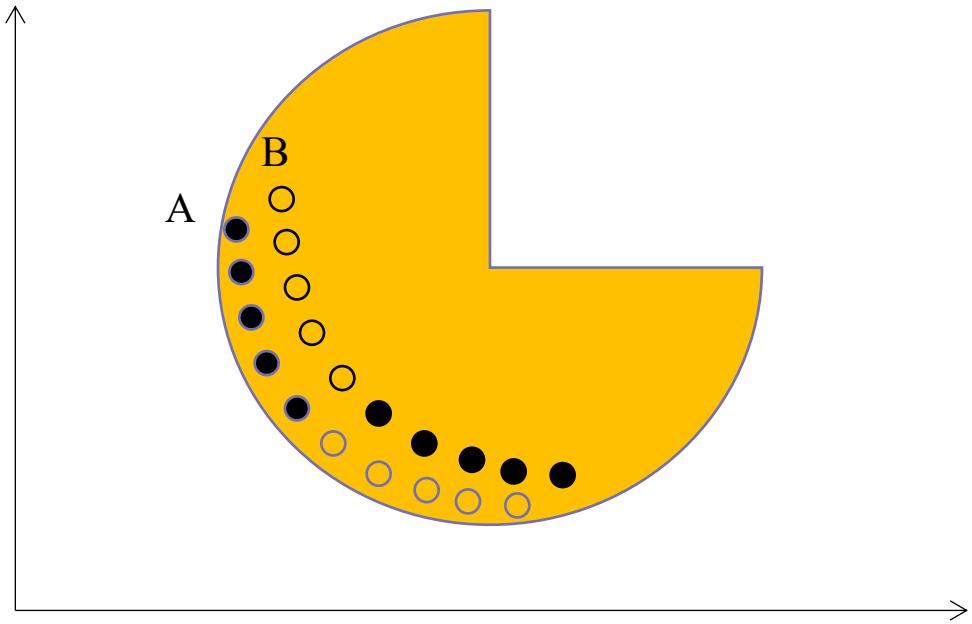
- **Algorithm A with better solutions than B**



Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

- Algorithms A and B are difficult to compare

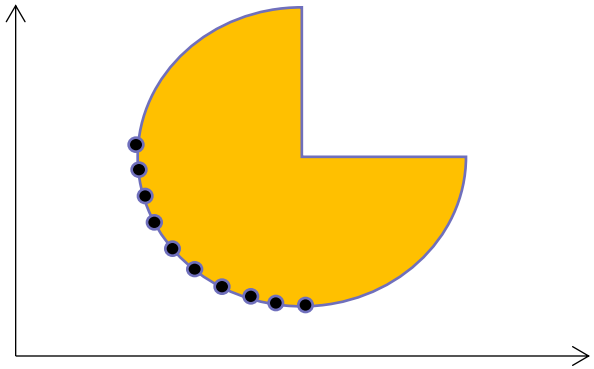


Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

- **Generation Vector Non-Dominated (*GVND*)**

$$GVND \overset{\Delta}{=} | Y_{known} |_c$$

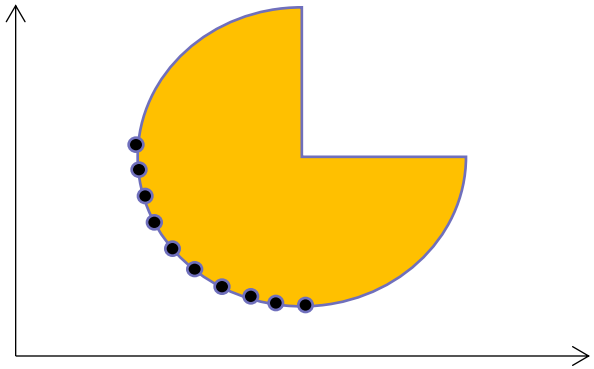


Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

- **Ratio Generation Vector Non-Dominated (*RGVND*)**

$$RGVND \stackrel{\Delta}{=} \frac{GVND}{|Y_{true}|_c}$$

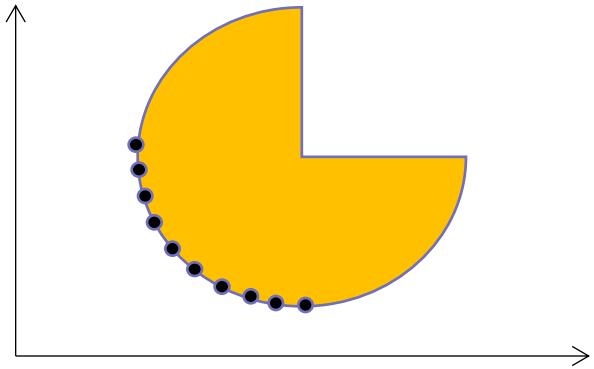


Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

- **Generation Real of Vector Non-Dominated (*GRVND*)**

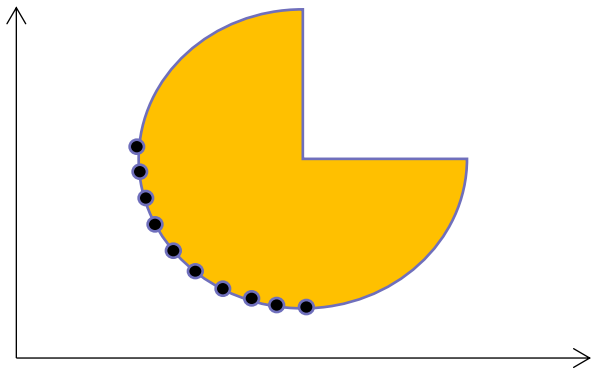
$$GRVND \stackrel{\Delta}{=} \left| \left\{ y \mid y \in Y_{known} \quad \wedge \quad y \in Y_{true} \right\} \right|_c$$



Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

- **Error Ratio (*E*)**



$$E \overset{\Delta}{=} \frac{\sum_{i=1}^{N} e_i}{GVND}$$

$$e_i \begin{cases} 0 & si \text{ un vector en } Y_{known} \text{ esta también en } Y_{true} \\ 1 & de\ otro\ modo \end{cases}$$

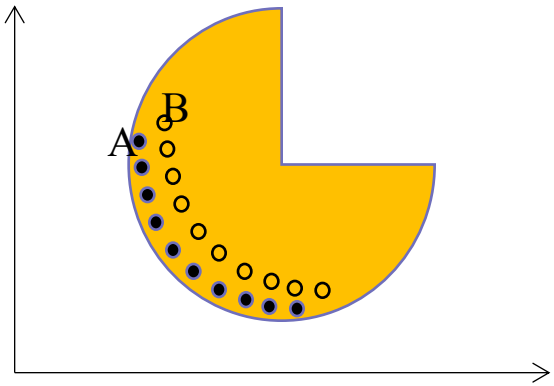
$$E = \frac{GVND - GRVND}{GVND}$$

Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

- **Generational Distance (*G*)**

$$G \overset{\Delta}{=} \frac{\left(\sum_{i=1}^{N} d_i^2\right)^{1/2}}{GVND}$$



$$G \overset{\Delta}{=} \frac{\left(\sum_{i=1}^{N} d_i^p\right)^{1/p}}{GVND}$$

Multi-Objective Evolutionary Algorithms

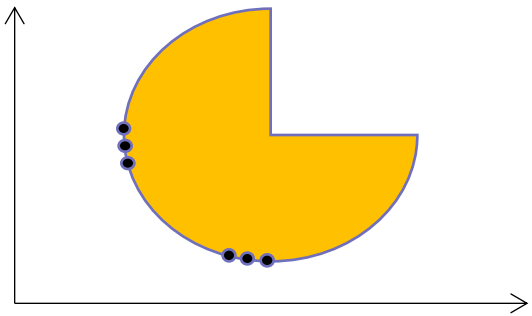
MultiObjective Metrics

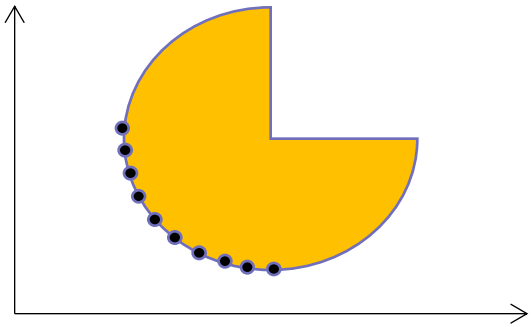
- **Spacing (S)**

$$S = \sqrt{\frac{1}{|Q|} \sum_{i=1}^{|Q|} \left(d_i - \bar{d} \right)^2}$$

$$d_i = \min_{k \in Q \wedge k \neq i} \sum_{m=1}^{M} \left| f_m^i - f_m^k \right|$$

$$\bar{d} = \sum_{i=1}^{|Q|} d_i \Big/ |Q|$$



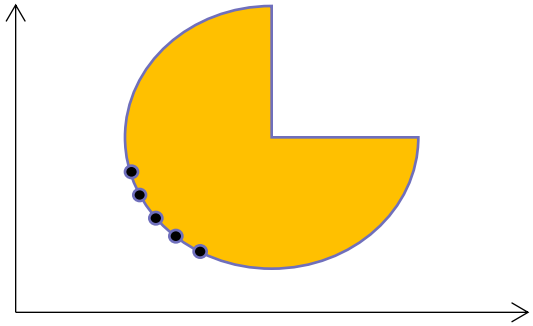


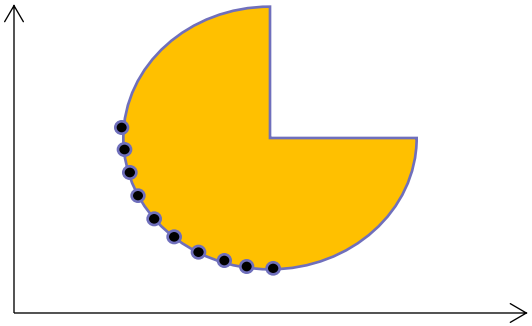
Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

- **Spread (Δ)**

$$\Delta = \frac{\sum_{m=1}^{M} d_m^e + \sum_{i=1}^{|Q|} \left| d_i - \bar{d} \right|}{\sum_{m=1}^{M} d_m^e + |Q| \bar{d}}$$





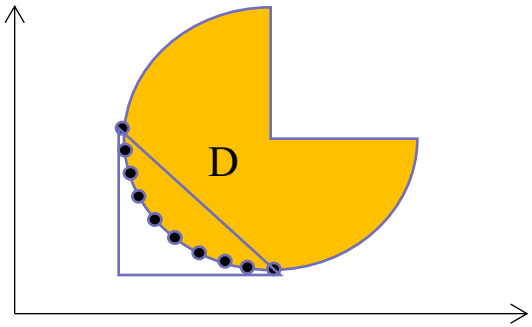
Multi-Objective Evolutionary Algorithms

MultiObjective Metrics

- **Maximum Spread (D)**

$$D = \sqrt{\sum_{m=1}^{M} \left(\max_{i=1}^{|Q|} f_m^i - \min_{i=1}^{|Q|} f_m^i \right)^2}$$

$$\overline{D} = \sqrt{\frac{1}{M} \sum_{m=1}^{M} \left(\frac{\max_{i=1}^{|Q|} f_m^i - \min_{i=1}^{|Q|} f_m^i}{\max_m F - \min_m F} \right)^2}$$



Outline

1. Single-objective vs Multi-Objective Optimization
2. Traditional Methods
3. Multi-Objective Evolutionary Algorithms
4. Conclusions

Conclusions

- Many real-world problems in engineering are multi-objective. On the other hand, there are various algorithmic techniques to solve these problems.
- In this Keynote, I presented both approach the problem in mathematical form and mathematical and numerical techniques to solve them and finally
- With evolutionary algorithms to optimally solve these problems in polynomial time complexity.

References

[1] Donoso, Yezid, Fabregat, Ramon, “Multi-Objective Optimization in Computer Networks Using Metaheuristics”.1 ed. Boca Raton : Auerbach Publications, USA. 2007, v.1. p.449

[2] K. Deb. “Multi-Objective Optimization using Evolutionary Algorithms”. John Wiley & Sons. 2001.

[3] Lozano C, Camelo M, Vila P, Donoso Y. A Multi-objective Routing Algorithm for Wireless Mesh Network in a Smart Cities Environment. Journal of Networks , ISSN 1796-2056. Volume 10, Number 1. 2015.

[4] F.A. Maldonado-Lopez, Yezid Donoso. Reliable Critical Infrastructure: Multiple Failures for Multicast Using Multi-Objective Approach. International Journal of Computers Communications & Control, ISSN 1841-9836. Volume 8, Number 1, 2013.

Thank you!!!

Prof. Yezid E. Donoso Meisel, Ph.D.
e-mail : ydonoso@uniandes.edu.co

2020 8th International Conference on Computers Communications and Control (ICCCC)