

# Unidad 3



## Inteligencia artificial: *Optimización Multiobjetivo* *NSGA II*

Dra. Soledad Espezua  
[sespezua@pucp.edu.pe](mailto:sespezua@pucp.edu.pe)



# Optimización

El objetivo de un problema de optimización global, es encontrar un conjunto de variables  $x^*$  que maximice o minimice una determinada función  $f(x)$ .

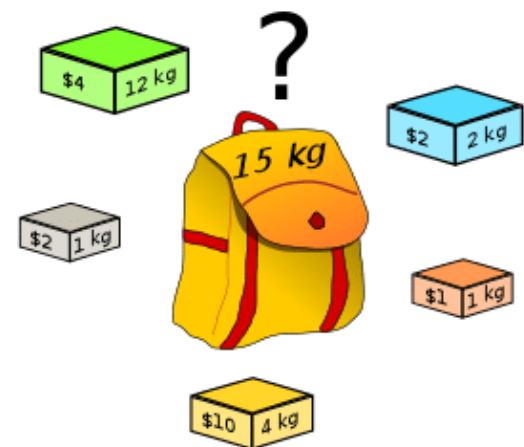
## Optimización Multiobjetivo (MOO)

- Cuando un problema de optimización tiene varias funciones objetivo, la tarea de encontrar una o más soluciones óptimas se denominada optimización multiobjetivo.

### Problema de la mochila

Colocar en una mochila objetos de diferente peso y variada importancia, sin exceder el espacio.

Objetivo: Minimizar el peso de los objetos y maximizar su utilidad sujetos a la capacidad de la mochila



# Optimización Multiobjetivo (MOO)

En este escenario diferentes soluciones pueden generar situaciones conflictivas entre sus objetivos.

- Una solución que es óptima con respecto a un objetivo puede no serlo para el resto.

## Dilema del prisionero

- Dos sospechosos de un crimen son colocados en ambientes separados para obtener pruebas para condenarlos.
- Objetivo: salir de la cárcel, delatando a su compañero o cooperando alegando inocencia (no delatar)

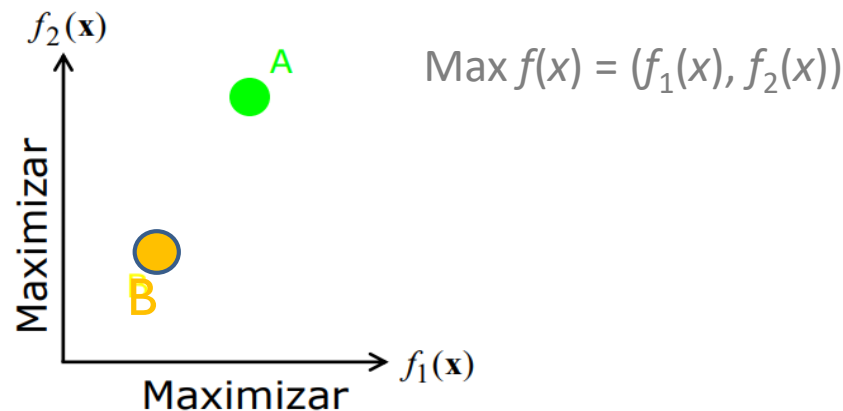
		Prisionero 1	
		No delatar	Delatar
Prisionero 2	No delatar	(1,1)	(7,0)
	Delatar	(0,7)	<u>(3,3)</u>

<https://economipedia.com/definiciones/dilema-del-prisionero.html>

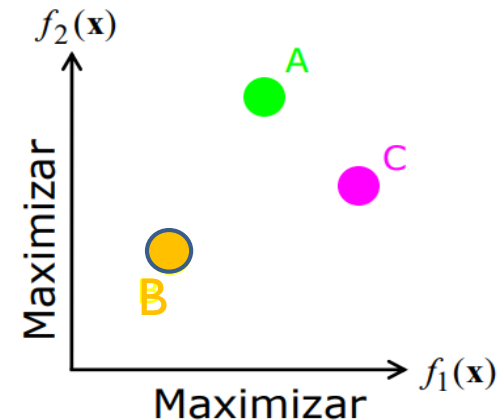
- MOO busca crear una solución de consenso entre los objetivos.

# MOO – Dominancia

- ▶ Se dice que una solución  $x_1$  **domina** a otra solución  $x_2$ , si se cumplen las siguientes condiciones:
  1. La solución  $x_1$  no es peor que  $x_2$  en todos los objetivos.
  2. La solución  $x_1$  es estrictamente mejor que  $x_2$  en por lo menos un objetivo.
- ▶ Si alguna de las condiciones es violada, la solución  $x_1$  **no domina** la solución  $x_2$



A domina a B  
B es dominada por A  
(A es mejor que B)

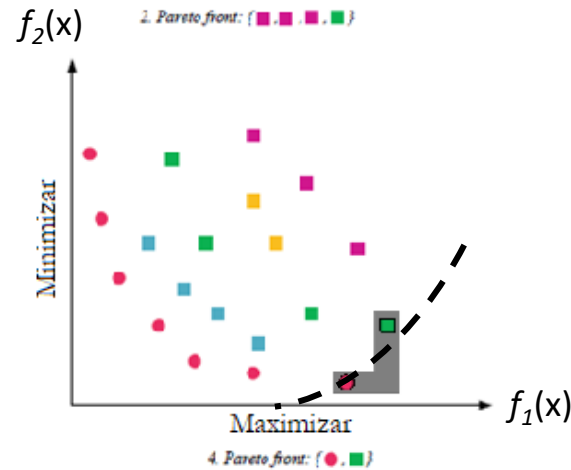
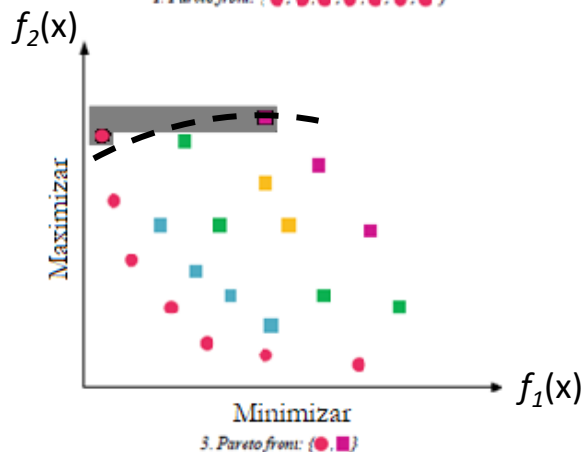
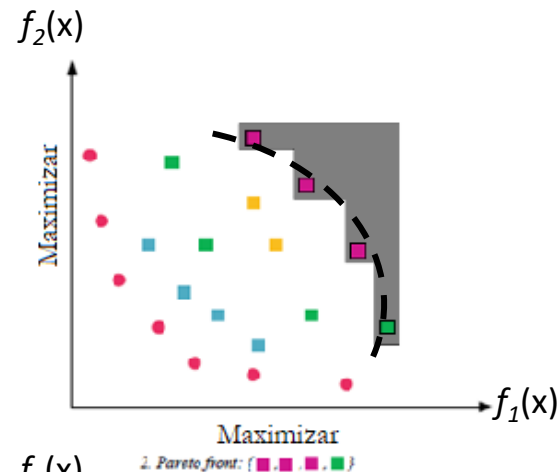
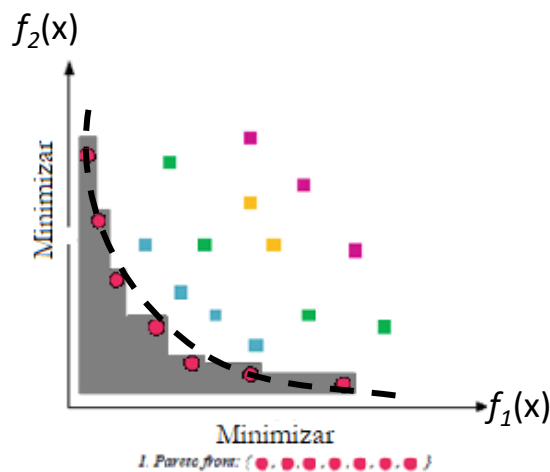


A y C son no dominadas entre sí  
(ninguna domina a la otra)  
Las dos dominan a B

# MOO - Frontera de Pareto

El conjunto de soluciones **no dominado** se denomina **Frontera de Pareto**.

Ejm. Conjunto de frontera de Pareto optima para un Problema bi-objetivo

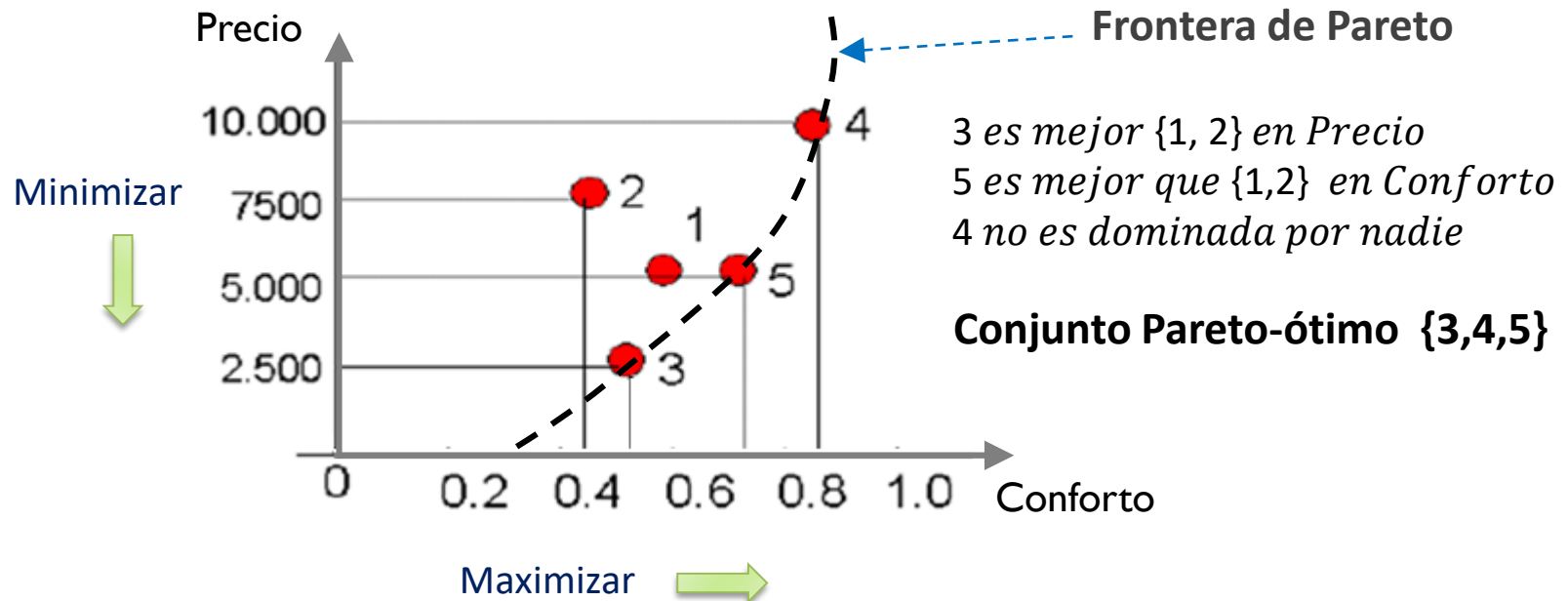


# MOO - Frontera de Pareto

- ▶ Una solución  $x^* \in \mathfrak{S}$  es **Pareto-óptima**, cuando es **no dominada** por ninguna otra solución.
- ▶ Una solución es **no dominada**, cuando al menos es tan buena como las otras en todos sus objetivos y es mejor en al menos uno de ellos.

Ejemplo

Habitaciones de un hotel



# NSGA II

## *Nondominated Sorting Genetic Algorithm (NSGA-II)*

NSGA-II <sup>1</sup>, es un AG elitista propuesto para solucionar problemas MOO.

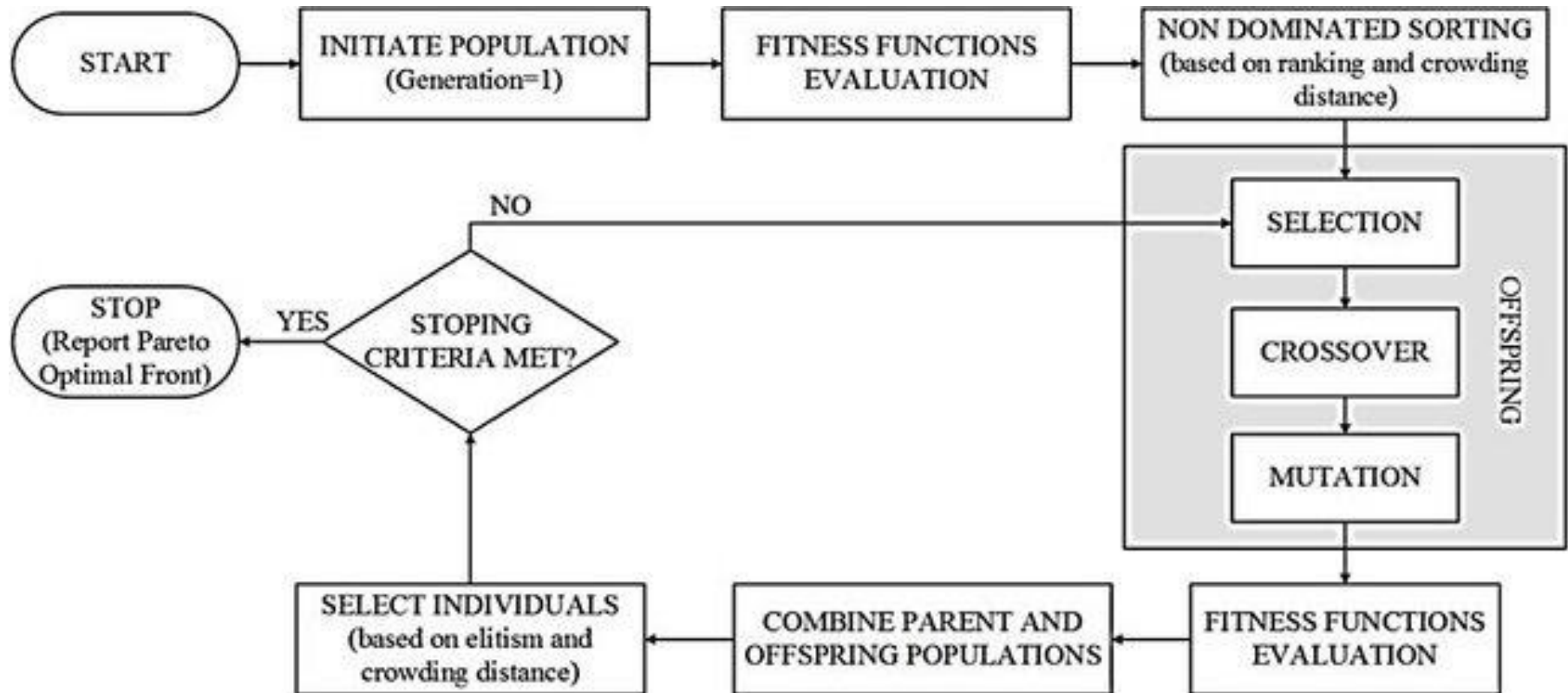
NSGA II fue propuesto por *Kalyanmoy Deb* y es una versión mejorada de NSGA.

Características:

- ▶ Combina la población de padres y hijos, conservando las mejores soluciones de ambos.
- ▶ Usa elitismo, que lo hace mucho más eficiente (computacionalmente) que NSGA (descarta soluciones no-dominadas en la iteración actual).
- ▶ Utiliza un operador de distancia (*crowding distance*) que no requiere parámetros.

1. Deb, k.; Agrawal, s.; Pratab, a.; Meyarivan, t. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. KanGAL report 200001, Indian Institute of Technology, Kanpur, India, 2000.

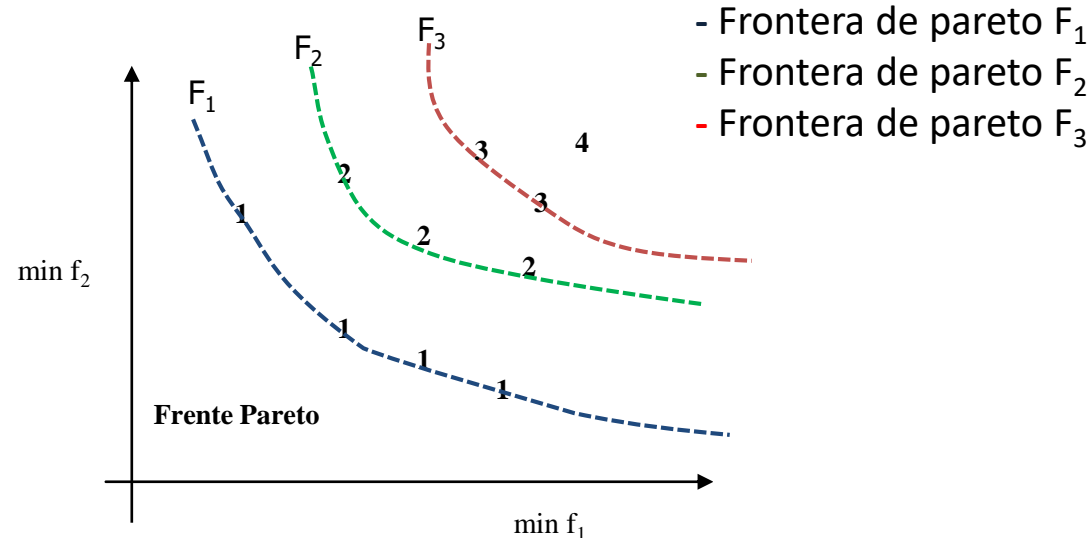
# NSGA II





# NSGA II - Procedimiento

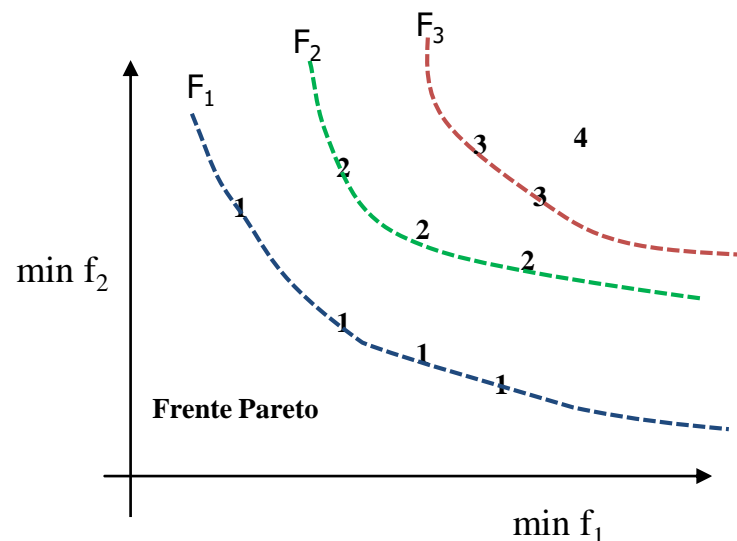
1. Generar aleatoriamente una población  $P_t$  de tamaño  $N$ .
2. Evaluar la población según las funciones objetivo según el concepto de **dominancia**
3. Ordenar la población  $P_t$  por ranking de no dominancia (frontera de Pareto).
  - Objetivo: Ordenar población según el nivel de **no dominancia**, en los diferentes frentes ( $F_1, F_2, F_3, \dots$ , etc, donde 1 es el mejor nivel, luego 2 y así sucesivamente).



# NSGA II - Procedimiento

Pasos para ordenar la población según el ranking de no dominancia:

- Identificar los individuos **no dominados** de una población y colocarles el mismo *fitness*.
  - Esto implica que todas las soluciones en dicha categoría tienen la misma probabilidad de reproducirse.
- Retirar los individuos ya asignados a un categoría (*rank*) e identificar un nuevo conjunto de soluciones no dominadas.
  - A este nuevo grupo se le asigna el mismo *fitness* pero mayor que el rank anterior.
- El proceso continua hasta que todos los individuos de la población son asignados a una categoría.



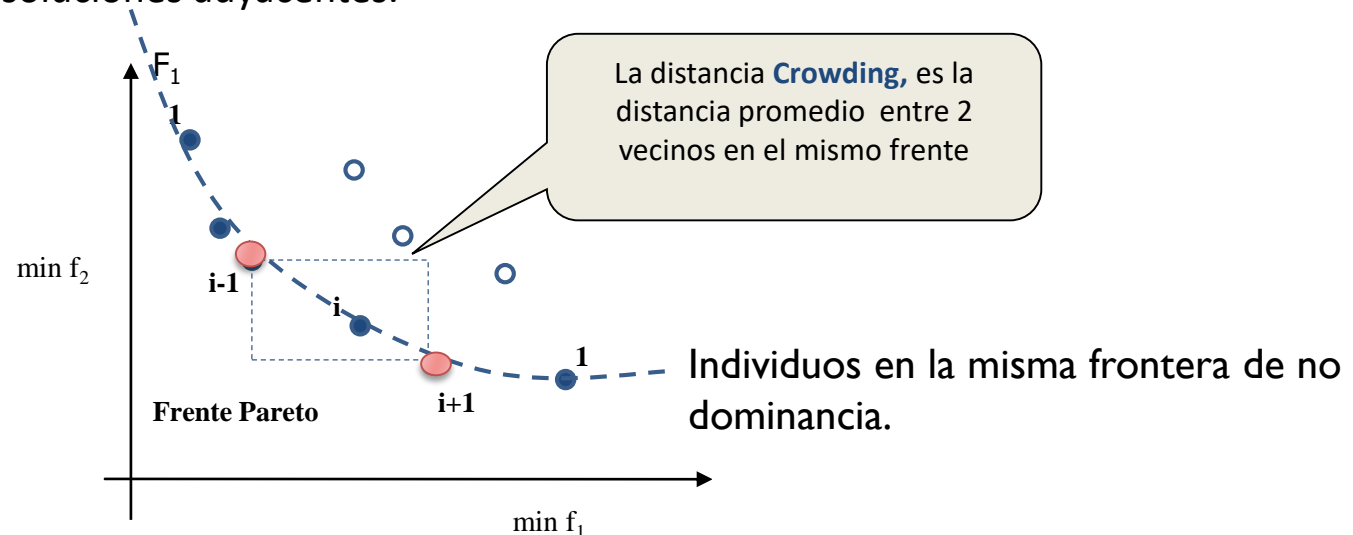
Población ordenada según el valor (*fitness*) en orden ascendente

# NSGA II - Procedimiento

Un individuo tiene 2 valores: un ranking ( $F_1, F_2, F_3, \dots$ ) de no dominancia asociado y una distancia crowding.

## 4. Seleccionar los individuos por torneo de dos.

- Escoger 2 individuos y elegir al mejor con respecto al ranking de no dominancia.
- Si 2 individuos seleccionados están en la misma frontera de no dominancia, usar la distancia crowding para elegir el mejor.
  - La distancia crowding se calcula como la diferencia normalizada absoluta de los valores fitness de dos soluciones adyacentes.

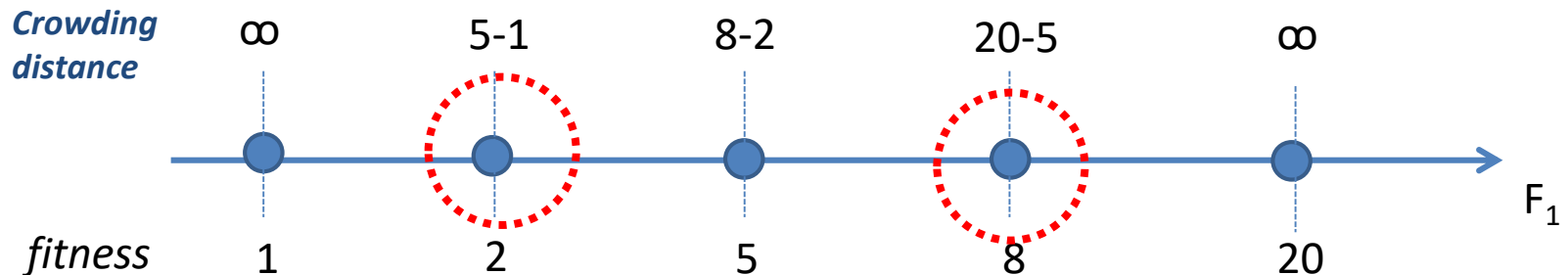


# NSGA II - Procedimiento

Pasos para seleccionar los individuos usando la distancia crowding:

- ▶ Por cada frontera realizar lo siguiente:
  - Asignar a los primeros y últimos individuos en el rank un valor de distancia infinity:  
Distancia **Crowding** = **infinity**
  - Al resto de individuos (soluciones intermedias) asignar un valor usando la distancia *crowding*.
  - Calcular el valor global de la distancia *Crowding* se calcula como la suma de los valores de distancia individuales correspondientes a cada individuo en una frontera.

Ejemplo: Para la frontera  $F_1$



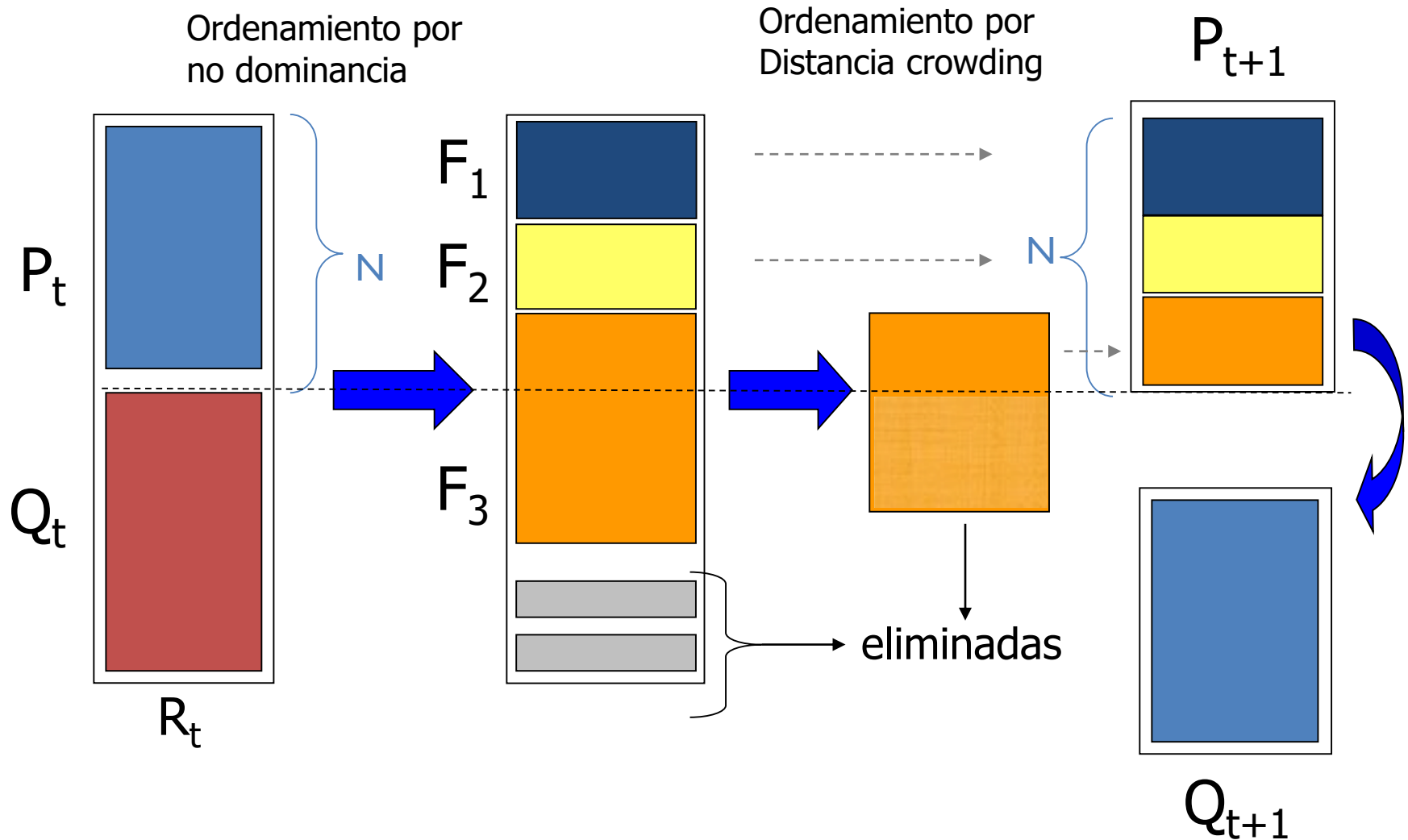
# NSGA II - Procedimiento

5. Usar los operadores de reproducción (cruzamiento y mutación) de igual forma que en un AG, para crear la siguiente generación  $Q_t$  de tamaño N.
6. Evaluar las funciones objetivo de la nueva generación  $Q_t$ .
7. Combinar la población de padres  $P_t$  e hijos  $Q_t$  para formar  $R_t$  de tamaño 2N.

$$R_t = P_t \cup Q_t$$

5. Para seleccionar los individuos de la siguiente generación  $Q_{t+1}$ :
  - Primero ordenar la población  $R_t$  por **ranking de no dominancia de Pareto** (selección de los mejores individuos)
  - Luego ordenar los individuos resultantes por la **distancia crowding**.
  - Finalmente seleccionar los N mejores individuos de la población.
8. Criterio de parada: El procedimiento termina cuando se alcanza el máximo número de generaciones.

# NSGA II- Generación de hijos



# Algoritmos Evolutivos Multiobjetivo

Métodos populares en la literatura:

Sigla	Nombre del Modelo
VEGA	<i>Vector Evaluated Genetic Algorithm</i>
WBGA	<i>Weight Based Genetic Algorithm</i>
MOGA	<i>Multiple Objective Genetic Algorithm</i>
NSGA	<i>Non-Dominated Sorting Genetic Algorithm</i>
NPGA	<i>Niched-Pareto Genetic Algorithm</i>
PPES	<i>Predator-Prey Evolution Strategy</i>
REMOE	<i>A Rudolph's Elitist Multi-Objective Evolutionary Algorithm</i>
NSGA-II	<i>Elitist Non-Dominated Sorting Genetic Algorithm</i>
SPEA, SPEA-2	<i>Strength Pareto Evolutionary Algorithm 1 y 2</i>
TGA T	<i>Thermodynamical Genetic Algorithm</i>
PAES	<i>Pareto-Archived Evolutionary Strategy</i>
MONGA -I, MONGA - II	<i>Multi-Objective Messy Genetic Algorithm</i>
PESA-I, PESA-II	<i>Pareto Envelope-Base Selection Algorithm</i>

# Bibliografía MOO

---

- ▶ Coello, C. A Short Tutorial on Evolutionary Multiobjective Optimization. In: Zitzler, E.; Deb, K.; Thiele, L.; Coello, C. A. C.; Corne, D., eds. First International Conference on Evolutionary Multi-Criterion Optimization, Springer-Verlag., 2001,p. 21–40 (Lecture Notes in Computer Science, v.1993).
- ▶ Deb, K. Multi-objective optimization using evolutionary algorithms. New York: John Wiley & Sons, 2001.
- ▶ Deb, K.; Agrawal, S.; Pratab, A.; Meyarivan, T. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. KanGAL report 200001, Indian Institute of Technology, Kanpur, India, 2000.