



Facultad de Ciencias
Licenciatura en
Ciencias de la Computación

Cómputo Evolutivo

.....

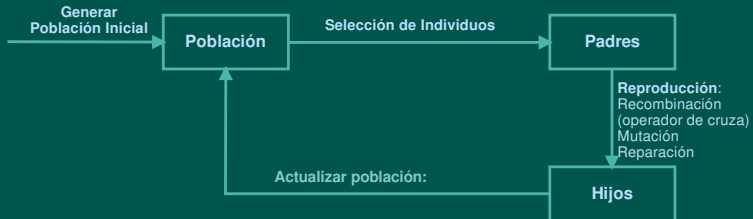
Operadores Algoritmo Evolutivo

M. en C. Oscar Hernández Constantino
(constantino92@ciencias.unam.mx)

Contenido de la Presentación

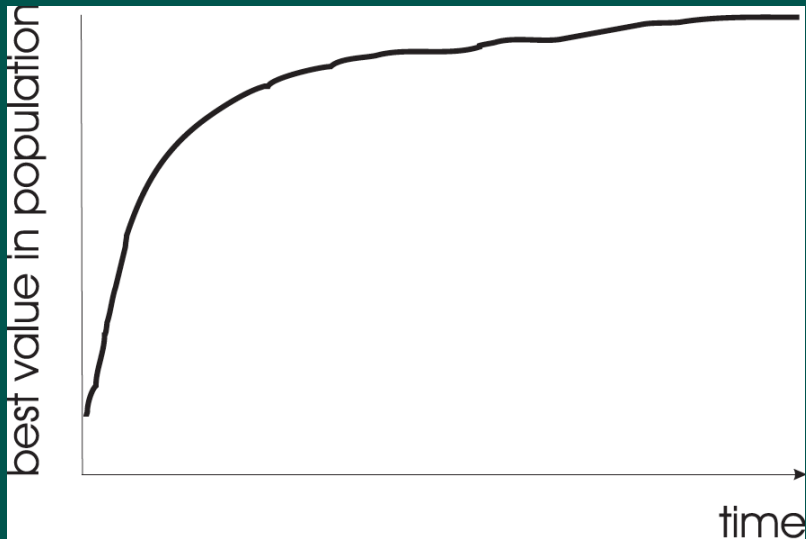
1. Esquema General
2. Inicialización
3. Operadores de Cruce
4. Métodos de Selección
5. Reemplazamiento

Esquema General

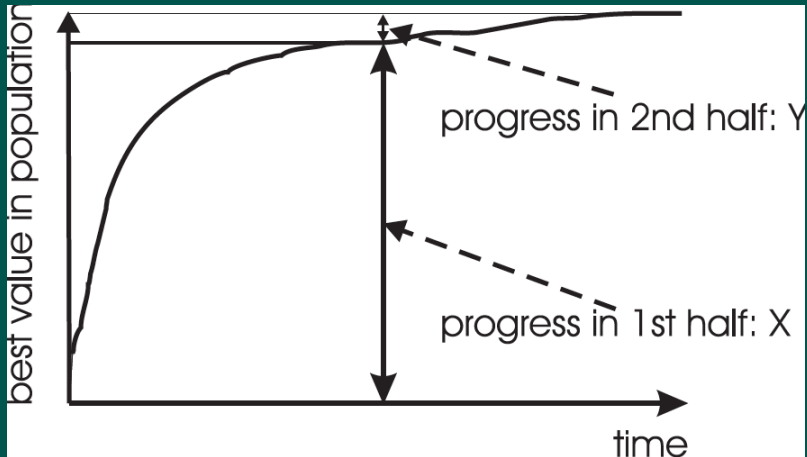


Inicialización

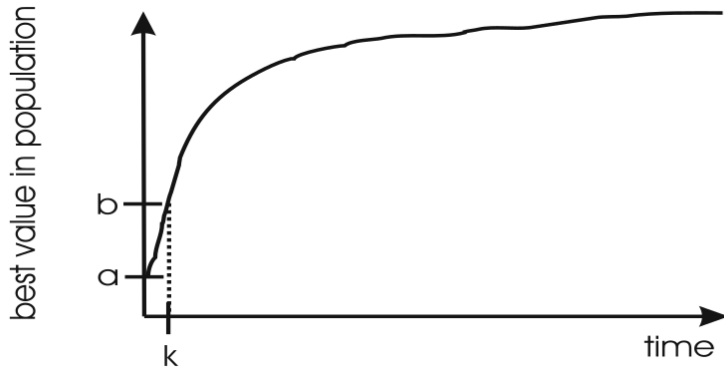
Comportamiento típico



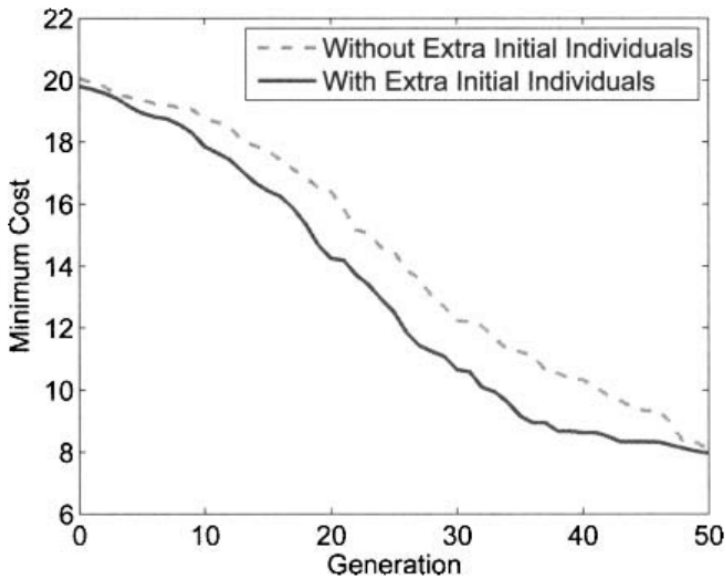
Comportamiento típico



Inicio con soluciones mejoradas

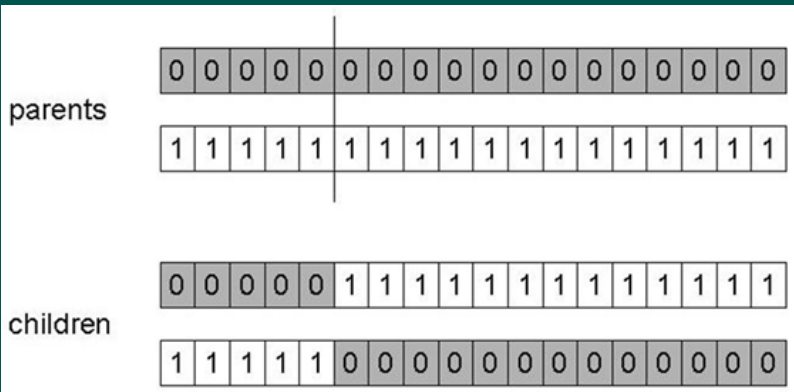


Inicio con soluciones mejoradas



Operadores de Cruce

Cruza de 1 punto



Alternativas

¿Por qué se requieren otros operadores de cruce?

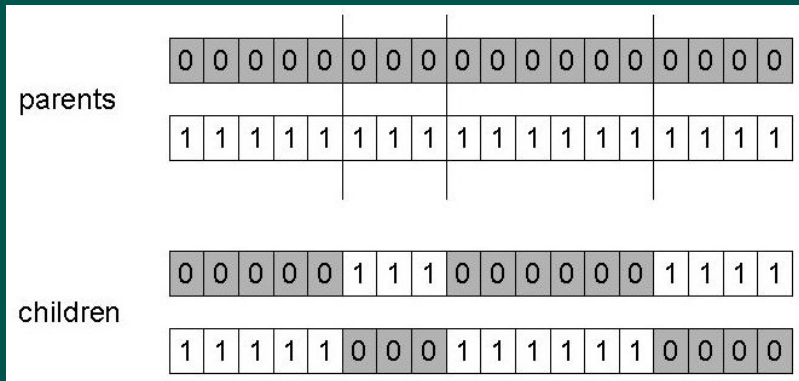
- La **cruza de 1 pto** tiene una dependencia en el orden en que las variables aparecen en la representación.

Sesgo Posicional:

- Genes cercanos se pueden heredar con alta probabilidad
 - Genes alejados tienen una baja probabilidad de heredarse.
- Se puede explotar esta propiedad si se tiene información de la estructura del problema.

Cruza de n puntos

- Genalización de la Cruza de un punto



Cruza Uniforme

parents

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

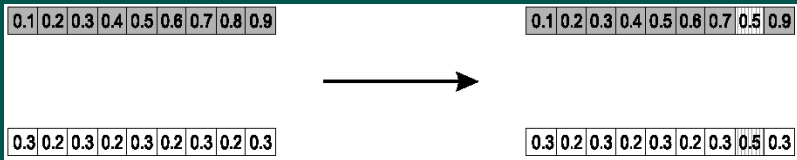
children

0	1	0	0	1	0	1	1	0	0	0	1	0	1	1	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

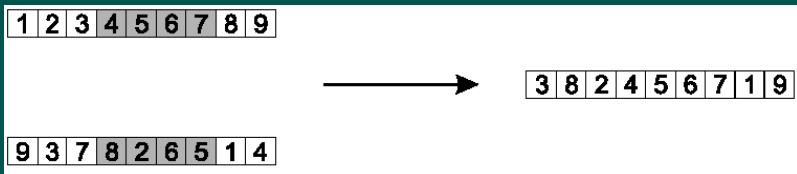
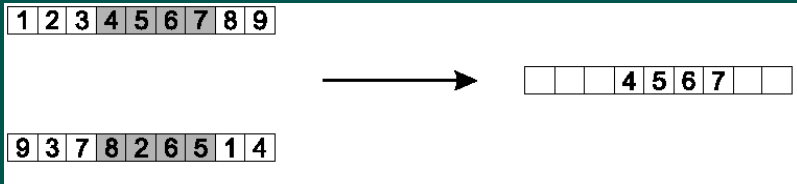
1	0	1	1	0	0	0	0	1	1	1	0	1	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

- La herencia es independiente de la posición

Cruza para números reales



Cruza para permutaciones



Cruza múltiples padres

Parent 1	3	5	9	2	1	8	4	6	7
Parent 2	9	1	5	7	3	4	6	2	8
Parent 3	5	6	3	1	7	2	9	4	8
Parent 4	7	2	9	4	3	1	5	8	6
↓									
	-	-	9	2	1	-	-	-	-

Cruza vs Mutación

- ¿Qué es mejor?
 - Depende del problema
 - Es recomendable tener ambos
 - Es posible tener un algoritmo evolutivo que solo use mutación
 - Tener un algoritmo evolutivo solamente con cruza podría no trabajar bien
- Cruza es un operador n-ario
- Solamente la mutación puede introducir nueva información (alelos)
- La cruza no cambia la frecuencia de los alelos en la población

Exploración vs Explotación

- Se tiene un esquema de **cooperación y competencia**
 - **Exploración**
Descubrir zonas primosorias en el espacio de búsqueda
 - **Explotación**
Optimizar en una zona promisorio
- Típicamente, Cruza es un operador exploratorio
- Mutación es un operador explotativo

Métodos de Selección

Competencia entre Individuos

- Los operadores responsables de que se de la competencia entre individuos trabajan considerando la aptitud de los individuos.
- La competencia ocurre en dos etapas:
 - **Selección** de los padres que son considerados para la reproducción.
 - **Reemplazo** de individuos que son considerados para la siguiente generación.
- Estos operadores (selección y reemplazo) son independientes de la representación.
- Los métodos de selección pueden aplicarse también como métodos de reemplazo.
- **Operadores:** definen una probabilidad de selección

Selección Proporcional (Fitness-Proportionate Selection)

- La probabilidad de selección de un individuo es:

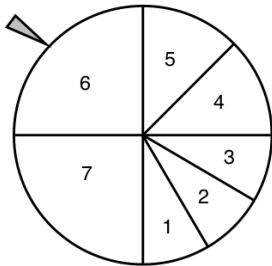
$$P_{FPS}(i) = \frac{f_i}{\sum_{j=1}^{\mu} f_j}$$

- La fórmula está expresada para problemas de maximización
- Cuando se tienen valores negativos de la función de aptitud se debe realizar un ajuste (escalamiento)
- Se tiene una alta **presión de selección**, por lo que suele provoca **convergencia prematura**
- Cuando la mayoría de los individuos tiene una aptitud similar la presión de selección disminuye

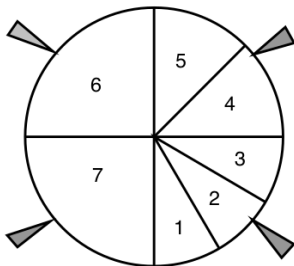
Muestreo Universal Estocástico

Individuals:	1	2	3	4	5	6	7
Fitness:	1	1	1	1.5	1.5	3	3

Outer



Roulette selection



Stochastic universal sampling

Selección Aleatoria (Uniforme)

- La probabilidad de selección de un individuo es:

$$P_{uniforme} = \frac{1}{\mu}$$

- Se realiza una selección de manera uniformemente aleatoria
- No está sesgada, cada individuo tiene la misma probabilidad de ser seleccionado
- Se puede usar en diferentes escenarios, dependiendo del problema y de los otros componentes del algoritmo

Selección basada en Posición (Rank-based Selection)

- La probabilidad de selección de un individuo es:

$$P_{lin-rank}(i) = \frac{(2-s)}{\mu} + \frac{2i(s-1)}{\mu(\mu-1)}$$

- $1 \leq s \leq 2$, controla la ventaja del mejor individuo.
- Se intenta solventar las desventajas de selección proporcional, considerando una aptitud relativa, más que una absoluta.
- Se ordena la población de acuerdo a su aptitud y se asigna la probabilidad de selección dependiendo de la posición.

Selección basada en Posición - Ejemplo

Individuals	A	B	C
Fitness f	1	5	4

Rank r	1	3	2
----------	---	---	---

Probability	0	0.67	0.33
-------------	---	------	------

s (Selection pressure) = 2

Probability	0.1	0.5	0.4
-------------	-----	-----	-----

Roulette selection using fitness

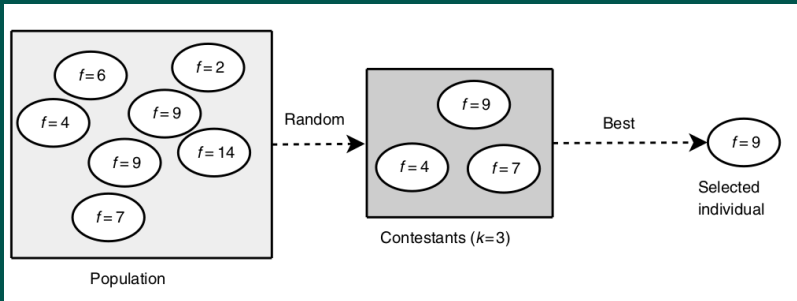
Probability	0.167	0.5	0.33
-------------	-------	-----	------

s (Selection pressure) = 1.5

Selección por Torneo

- Los métodos anteriores consideran la población completa para determinar la probabilidad de selección.
- Es posible utilizar solo información local:
 - Seleccionar k individuos de manera aleatoria y elegir al mejor;
 - Repetir para seleccionar a más individuos.
- La probabilidad de selección dependerá de:
 - Valor de aptitud del individuo
 - k , Tamaño del torneo
A mayor tamaño, mayor presión de selección.
 - Selección con o sin reemplazo
La selección sin reemplazo incrementa la presión de selección.

Selección por Torneo - Ejemplo



Reemplazamiento

Modelos Poblacionales

- **Generacional**

- En cada generación empezamos con una población de μ individuos, desde el cual se selecciona el conjunto de padres.
- λ individuos (hijos) son creados, aplicando operadores de variación a los padres, y evaluados.
- Al final de la iteración (generación), la población es reemplazada por μ individuos seleccionados de los hijos.

- **Estado estacionario (steady-state)**

- La población entera no se cambia en cada iteración, solo una parte. En este caso, λ individuos son reemplazados por los hijos.
- El modelo de estado estacionario ha sido ampliamente estudiado y aplicado, frecuentemente con $\lambda = 1$.

Tipos de reemplazo

- **Basado en la Edad**

- No toma en cuenta la aptitud de los individuos.
- Considera el número de generaciones que un individuo a permanecido en la población.
- Es la estrategia utilizada en el Algoritmo Genético simple, $\mu = \lambda$; también puede aplicarse en el modelo de estado estacionario, $\lambda < \mu$, típicamente con un FIFO.
- Puede implementarse como un reemplazo aleatorio.

- **Basado en la Aptitud**

- **Basado en la Diversidad**

Reemplazo basado en Aptitud

- **Elitismo**

- El mejor individuo siempre se mantiene en la población.
- Utilizado en combinación con otros esquemas (basados en edad, aptitud o diversidad).

- **Reemplazo de los peores**

- Suele provocar una rápida mejora en el promedio de la aptitud de la población.
- Se tiene convergencia prematura.
- Usualmente se utiliza en conjunto con una política para evitar individuos duplicados en la población.

- **Reemplazo con torneos**

Reemplazo basado en Aptitud - II

- **Selección ($\mu + \lambda$)**

- Generalización de los métodos anteriores.
- Los hijos y los padres son mezclados y ordenados para elegir a los mejores.
- Usualmente se tiene $\mu > \lambda$

- **Selección (μ, λ)**

- Típicamente se tiene $\lambda > \mu$.
- Es una combinación de métodos basados en edad y aptitud.
- Todos los padres son descartados.
- Se eligen a los mejores μ hijos para la siguiente generación.

Preservación de Diversidad

- Enfoques explícitos
 - Individuos similares compiten entre sí por la supervivencia
- Enfoques implícitos
 - Utilizan una analogía semejante a la separación geográfica

Preservación de Diversidad - Espacios

- Genotipo
Conjunto de soluciones representables
- Fenotipo
El resultado final. La vecindad podría tener poca relación con el genotipo.
- Espacio del Algoritmo
Estructura de la población de soluciones candidatas.

Compartición de Aptitud (Fitness Sharing)

- Se limita el número de individuos en un nicho (zona) de acuerdo a la aptitud.
- Se asignan individuos a los nichos en proporción de la aptitud en el nicho.

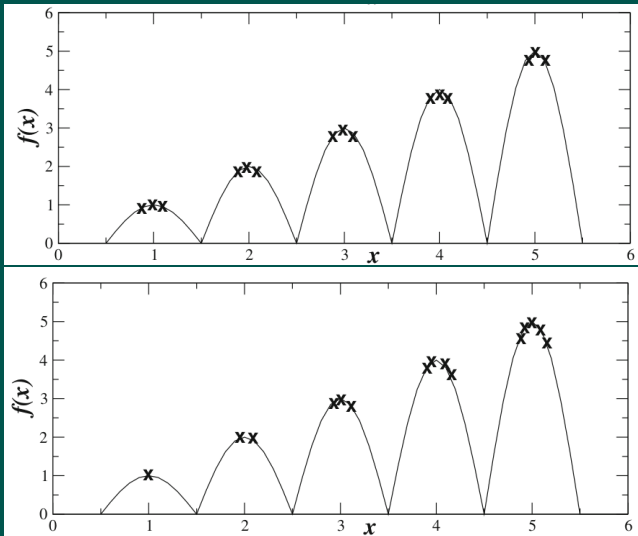
$$f'(i) = \frac{f(i)}{\sum_{j=1}^{\mu} sh(d(i, j))}$$

$$sh(d) = \begin{cases} (1 - d)^{\alpha/\sigma} & \text{si } d \leq \sigma \\ 0 & \text{en otro caso} \end{cases}$$

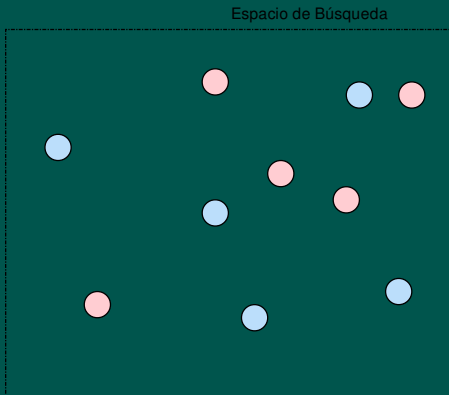
Agrupamiento (Crowding)

- Intenta distribuir a los individuos de manera uniforme en los nichos.
- Se basa en el supuesto de que los hijos estarán cerca de los padres.
- La propuesta original trabaja con un esquema de estado estacionario.
- Cuando un individuo se inserta en la población, se eligen un número CF (Crowding Factor) de padres y el hijo reemplaza al padre más que es más similar (genotipo).

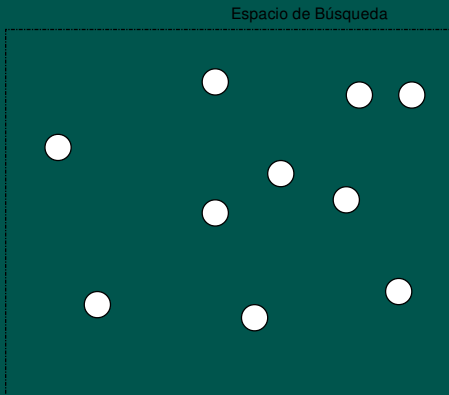
Preservación de Diversidad - Ejemplo



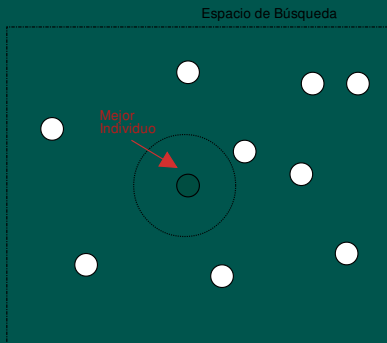
Mejor No Penalizado (BNP, Best Non Penalized)



Mejor No Penalizado (BNP, Best Non Penalized)

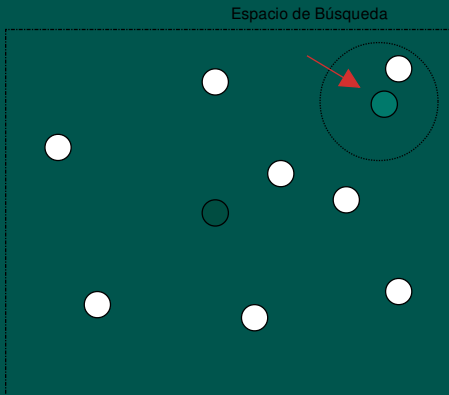


Mejor No Penalizado (BNP, Best Non Penalized)

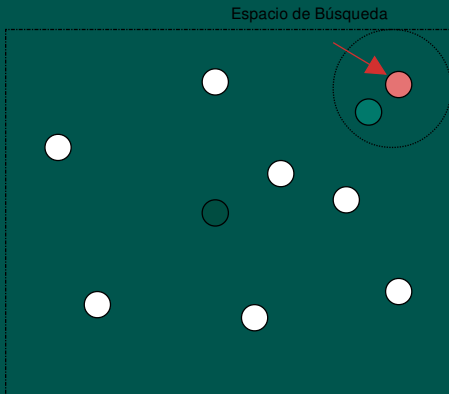


a) Selección del Mejor

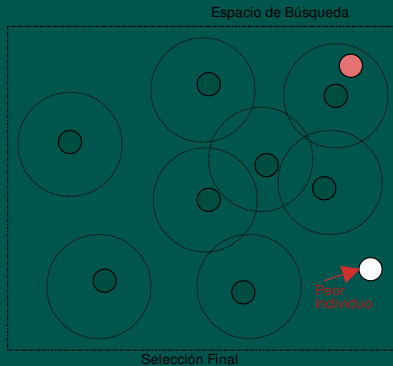
Mejor No Penalizado (BNP, Best Non Penalized)



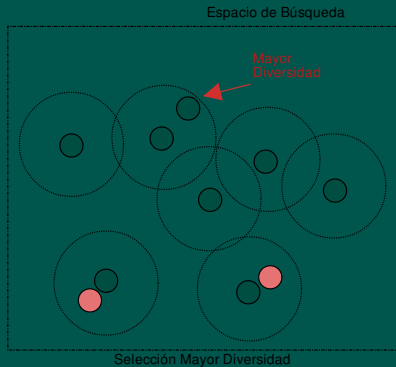
Mejor No Penalizado (BNP, Best Non Penalized)



Mejor No Penalizado (BNP, Best Non Penalized)



Mejor No Penalizado (BNP, Best Non Penalized)



¿ Preguntas ?