

COMPUTACION INTELIGENTE (Computación Evolutiva)

Jose Aguilar
Cemisid, Facultad de Ingeniería
Universidad de los Andes
Mérida, Venezuela
aguilar@ula.ve

COMPUTACION EVOLUTIVA

- OPTIMIZACION
- ADAPTACION
- MAQUINAS INTELIGENTES
- BIOLOGIA
- CONSTRUCCION DE PATRONES

COMPUTACION EVOLUTIVA

- ENFOQUES ALTERNATIVOS DE BUSQUEDA Y APRENDIZAJE BASADOS EN MODELOS COMPUTACIONALES DE PROCESOS EVOLUTIVOS
- *IDEA:* BUSQUEDA ESTOCASTICA EVOLUCIONANDO A UN CONJUNTO DE ESTRUCTURAS Y SELECCIONANDO DE MODO ITERATIVO A LAS MAS APTAS

FINALIDAD: SUPERVIVENCIA DEL MAS APTO

MODO: ADAPTACION AL ENTORNO

COMPUTACION EVOLUTIVA

- Los esfuerzos de la CE han consistido en relacionar algunos conceptos de la teoría de la evolución con un conjunto de técnicas computacionales
- El principal aporte de la CE ha sido el uso de mecanismos de selección de soluciones potenciales y de construcción de nuevas candidatas por recombinación de características de otras ya existentes.

COMPUTACION EVOLUTIVA

- EMULACION DE PROCESOS EVOLUTIVOS:
 - POBLACION DE POSIBLES SOLUCIONES => INDIVIDUOS
 - PROCESO DE SELECCIÓN => APTITUD DE LOS INDIVIDUOS
 - PROCESO DE TRANSFORMACION => GENERACION DE NUEVOS INDIVIDUOS

COMPUTACION EVOLUTIVA

- FENOMENOS NATURALES SIMULADOS:
 - HERENCIA GENETICA
 - LUCHA POR LA SUPERVIVENCIA
- INSPIRACIONES TEORICAS:
 - EVOLUCION DE DARWIN
 - SELECCIÓN DE WEISMANN
 - GENETICA DE MENDELL

COMPUTACION EVOLUTIVA

- EVOLUCION DE DARWIN:

Darwin propone la selección natural como uno de los mecanismos fundamentales capaces de explicar la génesis de nuevos tipos de individuos.

La teoría NeoDarwinista, combina la teoría de Darwin con los aportes de la Genética, para postular el principio,
la unidad sobre la que actúa la evolución no es el individuo, sino otra de orden superior: la población.

COMPUTACION EVOLUTIVA

- EVOLUCION DE DARWIN:
 - No todos los individuos de una misma especie son iguales.
 - Algunas de estas variaciones podrán resultar favorables en determinados ambientes.
 - Todas las poblaciones tienden a aumentar su numero geométricamente, sin embargo, en condiciones reales, en cada generación el numero permanece aproximadamente constante.

COMPUTACION EVOLUTIVA

- EVOLUCION DE DARWIN:
 - Los individuos mejor adaptados tendrán mas posibilidades de dejar descendientes a los cuales transmitirán sus características.
 - Como las características favorables se van acumulando (agregando), con el tiempo surgen grandes diferencias entre el grupo original y los individuos que se den dichas características.

COMPUTACION EVOLUTIVA

- Paradigmas Neo-Darwinianos
 - Individuos son el objeto de selección
 - Procesos estocásticos juegan un rol importante en la selección.
 - Variación genotípica es un producto de la recombinación
 - Evolución es un cambio en adaptación y diversidad
 - Selección es probabilista y no determinista

COMPUTACION EVOLUTIVA

- SELECCIÓN DE WEISMANN

“Demostró que los cambios en el cuerpo, debido al uso o el desuso de partes del mismo, no se reflejan en el plasma del germen, y por consiguiente, no pueden ser heredados”.

COMPUTACION EVOLUTIVA

- GENETICA DE MENDELL
 - En cada organismo existe un par de factores que regulan la aparición de una determinada característica (genes).
 - El organismo obtiene tales factores de sus padres.
 - Cada uno de estos factores se transmite como una unidad discreta.

COMPUTACION EVOLUTIVA

- GENETICA DE MENDELL
 - Cuando las células reproductivas (espermatozoides u óvulos) están formadas, los factores se separan y se distribuyen a los gametos en forma de unidades independientes (primera ley de Mendel, o ley de la segregación).
 - Si un organismo posee dos factores diferentes para una característica dada, uno de ellos debe expresarse y excluir totalmente al otro.

Aspectos generales de la computación evolutiva

El propósito genérico de los algoritmos en la CE es guiar una búsqueda estocástica haciendo evolucionar un conjunto de estructuras, seleccionando de modo iterativo las mas adecuadas.

Características de la CE

- Se trabaja con la codificación de un conjunto de parámetros (o un subconjunto de ellos) y no con los parámetros en si.
- Se usa una función objetivo o función de evaluación para definir lo que se desea optimizar. Esta función permite evaluar la calidad de los individuos en cada generación => función de aptitud

Características de la CE

- Se trabaja con una población del problema, no con un punto simple.
- Se usan reglas de transición probabilísticas, no reglas deterministas

COMPUTACION EVOLUTIVA

- OBJETIVOS CONFLICTIVOS QUE SE SIGUEN:
 - EXPLOTAR LAS MEJORES SOLUCIONES
 - EXPLORAR EL ESPACIO DE BUSQUEDA
- PARADIGMAS:
 - ALGORITMOS GENETICOS (HOLLAND)
 - PROGRAMAS EVOLUTIVOS (MICHALEWICZ)
 - PROGRAMACION GENETICA (KOZA)
 - ESTRATEGIAS EVOLUTIVAS (RECHENBERG SCHWEFEL)
 - PROGRAMACION EVOLUTIVA (FOGEL)

COMPUTACION EVOLUTIVA

- NIVEL DE ABSTRACCION:
 - GENOTIPO => ESTRUCTURAS SON CROMOSOMAS Y GENES. OPERADORES GENETICOS CAMBIAN ESTRUCTURAS (AG)
 - FENOTIPO => ESTRUCTURAS SON COMPORTAMIENTOS. OPERADOR MUTACION CAMBIA ESTRUCTURA (EE,PE).



UNIVERSIDAD
DE LOS ANDES
MERIDA VENEZUELA

Algoritmos Evolutivos

COMPUTACION EVOLUTIVA

- MACROALGORITMO:

- 1.- POBLACION INICIAL

- 2.- EVALUACION DE LOS INDIVIDUOS

- 3.- REPRODUCCION INICIAL

- 4.- REEMPLAZO

- 5.- CONDICION DE FINALIZACION O
REGRESA A PASO 2

COMPUTACION EVOLUTIVA

- ASPECTOS DE IMPLEMENTACION:
 - REPRESENTACION DE LOS INDIVIDUOS
 - MEDIDAS DE ADAPTACION (FUNCION ADAPTATIVA)
 - MECANISMOS DE SELECCIÓN Y REMPLAZO
 - INTERPRETACION DE LOS RESULTADOS
 - PARAMETROS Y VARIABLES QUE CONTROLAN EL PROCESO
 - OPERADORES GENETICOS A UTILIZAR

Representación de los Individuos

- Generalidades
- Tipos
 - Cadenas binarias $\Rightarrow 2^l$ soluciones en vector de longitud l
 - Vector de valores reales: incluye vector de desviación standard
 - Elementos del problema
 - Estados Finitos
 - Árboles

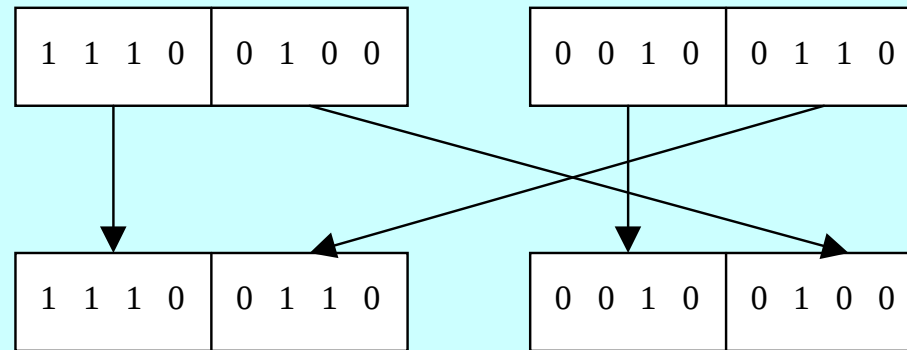
Operadores Genéticos

- Mecanismos de transformación que se utilizan para recorrer el espacio de las soluciones de un problema.
 - realizan los cambios de la población durante la ejecución de un algoritmo evolutivo.
- En general, se basan en los operadores genéticos biológicos.
- Operadores genéticos clásicos:
 - Mutación
 - Cruce

Operador Cruce

- Operador, normalmente binario,
- Permite representar el proceso de apareamiento natural
- Intercambia pedazos de informacion entre diversos individuos para producir dos nuevos individuos.
- El punto de corte es escogido aleatoriamente.

Operador Cruce

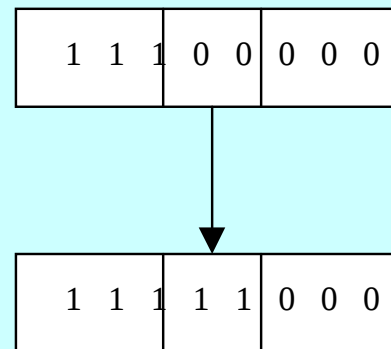


- Tipos:
 - Cruce multipunto
 - Cruce segmentado
 - Cruce uniforme

Operador de Mutación

- Operador unario que simula el proceso evolutivo que ocurre en los individuos cuando cambian su estructura genética.
- Se seleccionan aleatoriamente componentes (genes) de un individuo para ser modificados, también de forma aleatoria.
- A través del operador de mutación se puedan producir cambios en la estructura de un individuo

Operador de Mutación



- Tipos
 - Mutaciones sobre genes
 - Mutaciones no estacionarias
 - Mutaciones no uniformes

Operadores Avanzados

- Dominación
- Segregación
- Inversión
- Duplicación

Mecanismos de funcionamiento de los algoritmos evolutivos

- Estrategias de selección de los individuos susceptibles a reproducirse.
- Estrategia de apareamiento de los individuos en la fase de reproducción.
- Estrategia de reemplazo.

Mecanismos de Selección

- Políticas empleadas para la conformación de los progenitores.
- Tipos:
 - elitesca
 - aleatoria
 - Por sorteo
 - universal (por ruleta)
 - Por torneos

Muestreo por sorteo

$$p_i = \frac{f_i}{\sum_{j=1}^n f_j}$$

- Se calculan las probabilidades acumulativas (q_i):
 - $q_0 = 0$.
 - $q_i = q_{i-1} + p_i$.
- Se genera un número aleatorio α entre 0 y 1.
- Se elige el individuo i que cumpla con la condición:

$$q_{i-1} < \alpha < q_i.$$

Muestreo universal

Se genera un numero aleatorio α y se utiliza este para crear los números:

$$\alpha_i = \frac{\alpha + i - 1}{k} \quad \forall i = 1, k$$

Una vez obtenidos estos números, se aplica k-veces el paso 3 del muestreo por sorteo.

Muestreo por torneos

Se toman al azar m individuos de la población, y el mejor de estos m individuos formara parte de la muestra.

Mecanismos de Reemplazo

- Tipos:
 - Padre por hijo, o Reemplazo directo
 - Por similitud
 - Por inserción (,), p.e., peores por mejores
 - Por inclusión (+)

Estrategia de Apareamiento

- Que individuos con que individuos aparear en la fase de reproduccion
- Tipos:
 - Aleatorio,
 - Autofertilizacion,
 - etc.

ALGUNOS PARAMETROS Y VARIABLES QUE CONTROLAN EL PROCESO

- Probabilidades de uso de los operadores
- Criterios de parada
 - Por ejemplo, Número de generaciones
- Tamaño y diversidad de la población

Revisión Procedimiento general de los algoritmos evolutivos

- Generar una población inicial.
- Evaluar los individuos.
- **Seleccionar**, a través de algún mecanismo, ciertos individuos de la población.
- **Modificar** los genes de los padres seleccionados usando los operadores genéticos.
- Evaluar los nuevos individuos.
- Generar una nueva población con la existente y los individuos generados en el paso 4.
- Verificar el **criterio de convergencia**, o regresar al paso 3.

Aplicaciones

- **Planeación:** enrutamiento, planificación, enpaquetamiento
- **Diseño:** sistemas digitales y electronicos, redes neuronales, reactores, estructuras, aviones, compt. paralela
- **Simulación e Identificación:** modelo-comportamiento, determinar modelos
- **Control:** controlador, optimizador,
- **Clasificación:** economía, biología, procesamiento de imágenes, juegos



UNIVERSIDAD
DE LOS ANDES
MERIDA VENEZUELA

Algoritmos Genéticos

Algoritmos Genéticos (AGs)

METODO ESTOCASTICO DE BUSQUEDA INFORMADA DE SOLUCIONES CUASI-OPTIMAS. SE MANTIENE UNA POBLACION QUE REPRESENTA POSIBLES SOLUCIONES, LA CUAL ES SOMETIDA A CIERTAS TRANSFORMACIONES PARA OBTENER NUEVOS CANDIDATOS, Y A UN PROCESO DE SELECCIÓN SESGADO A FAVOR DE LAS MEJORES SOLUCIONES

Algoritmos Genéticos (AGs)

- Algoritmos de búsqueda que emulan la evolución biológica en el computador, siguiendo un proceso evolutivo inteligente sobre individuos,
- Fueron propuestos por John Holland a mediados de los años 70
- El objetivo final es encontrar la mejor solución posible, partiendo de soluciones escogidas aleatoriamente.

Algoritmos Genéticos (AGs)

- Dicha técnica es la más conocida.
- Razones:
 - Reúnen, de manera natural, todas las ideas fundamentales de la CE.
 - Poseen la mayor base teórica, la cual es sencilla y con mayores posibilidades de ampliación.
 - Requieren menor conocimiento específico del problema, lo que les da mayor versatilidad.

Características de los AGs

- Búsqueda Informada.
- Búsqueda Basada en Población.
- Operadores Aleatorios.

Generalidades de los AGs

- El procedimiento general consiste en mantener una población de potenciales soluciones a través de las generaciones.
- Los individuos de esa población son cadenas de caracteres.
- Se utilizan operadores evolutivos como mecanismos de búsqueda.
- El procedimiento continua hasta cumplirse algún criterio de convergencia o parada.
- Son el paradigma mas usado de la CE, y junto con las Redes Neuronales Artificiales, los mas usados de toda el área de Computación Inteligente.

Implementación de los AGs

- Seleccionar la Codificación: Si suponemos que un individuo Y posee m genes se requieren para la codificación L bits

$$L = \sum_{i=0}^m L_i$$

$$L_i = \log_2 a_i$$

gen i tiene a_i posibles valores (alelos).

- Seleccionar la población inicial y su tamaño.

Lo mas variada posible.

Implementación de los AGs

- Definir criterio de parada

- *numero de iteraciones* (generaciones).
- *similitud de las estructuras*
- *similitud de contenido.*

- Funcion de Evaluación

De manera ideal, la función de evaluación coincide con el objetivo a maximizar o minimizar

- Manejar los individuos no factibles

- *penalización,*
- *reparación o corrección,*
- *decodificación*

Implementación de los AGs

- Seleccionar los operadores genéticos

La influencia de los operadores genéticos es fundamental en el funcionamiento de los AGs; por lo tanto, es importante hacer una buena selección de los mismos. Habitualmente, se utilizan dos operadores, el de cruce y el de mutación. No obstante, cuando se tiene conocimiento específico del problema es posible incorporar otros operadores genéticos, o modificar los operadores ya existentes.

Fundamentos Matemáticos de los AGs

Los mecanismos de reproducción y el uso de los operadores genéticos poseen razones matemáticas para converger a buenas soluciones

Fundamentos Matemáticos de los AGs

- Normalmente, los AGs codifican los individuos como cadenas binarias,
 $\Rightarrow$ alfabeto $V = \{0, 1\}$
- Extendemos el alfabeto V con la agregación del símbolo $*$,
 $\Rightarrow$ alfabeto $V^+ = \{0, 1, *\}$,
* representa el símbolo comodín o carácter no importa, por lo cual puede valer 0 o 1.

Fundamentos Matemáticos de los AGs

- Patrón H: secuencia de símbolos del alfabeto V^+
- *Orden* $[o(H)]$ de un patrón H, es el numero de posiciones del patrón que están ocupadas por un 1 o por un 0.
- *Longitud* $[\Delta(H)]$, representa la distancia entre el primer y ultimo carácter de la cadena con valor de 0 o 1.

Efectos de la reproducción

Definición 1

Para valores altos del promedio de aptitud de un patrón, el numero de instancias de dicho patrón en la población crecerá; mientras que para valores bajos disminuirá.

Efectos de la reproducción

- H es un determinado patrón, $A(t)$ es la población en un tiempo t y, $m(H,t)$ es el numero de instancia del patrón H en A en el momento t .
- En el proceso de reproducción, una cadena A_i participara (formara parte de los padres) con una probabilidad p_i , definida por la ecuación:

$$p_i = \frac{f_i}{\sum_{j=1}^n f_j}$$

Efectos de la reproducción

- Luego de la reproducción y el reemplazo, en una población n ,

$m(H, t+1)$ (la cantidad de instancia del patrón H en el tiempo $t+1$)

$$m(H, t+1) = m(H, t)(f(H)/f')$$

$f(H)$ = la aptitud promedio de las cadenas que son instancias de H en el tiempo t .

Efectos de la reproducción

- Si suponemos que para los patrones que tienen valores promedios de aptitud alto, tal promedio es superior al de la población en una cantidad cf' , siendo c una constante positiva, podemos reescribir dicha ecuación

$$m(H,t+1) = m(H,t)(f' + cf')/f' = (1 + c).m(H,t)$$

$$\Rightarrow m(H,t) = m(H,0)(1 + c)^t$$

incrementa exponencialmente el número de instancias de esos patrones.

Efectos de los operadores genéticos

- El numero de instancias de un patrón en la población, cuando se usa el operador de cruce, es directamente proporcional a la probabilidad de supervivencia p_s del patrón.
- La probabilidad de supervivencia de un patrón H con longitud $\Delta(H)$,

$$p_s = 1 - \frac{\Delta(H)}{L-1}$$

- L es la longitud del código binario utilizado.
- $L-1$ representa las posibles selecciones para el punto de cruce.

Efectos de los operadores genéticos

- Si el operador de cruce es usado con una probabilidad p_c , entonces:

$$p_s \leq 1 - p_c \frac{\Delta(H)}{L-1}$$

- Efecto combinado de la reproducción y el cruce sobre el numero de instancias de un patrón:

$$m(H, t+1) \geq m(H, t) \frac{f(H)}{f'} \left[1 - p_c \frac{\Delta(H)}{L-1} \right]$$

Efectos de los operadores genéticos

- Para que un patrón H sobreviva, en el caso de usarse el operador de mutación, todas las posiciones que son 1 o 0 en H deben sobrevivir.
- Si suponemos que un bit (alelo) es mutado con probabilidad p_m ; entonces podremos decir que la probabilidad de sobrevivencia de un alelo es $1-p_m$;
- La probabilidad de sobrevivencia de un patrón usando el operador de mutación viene dada como $o(H)$ veces el producto de $1-p_m$, esto es:

$$(1-p_m)^{o(H)}$$

- Dicha probabilidad puede ser aproximada al valor $1-o(H).p_m$, para valores pequeños de p_m ($p_m \ll 1$).

Efectos de los operadores genéticos

Efecto combinado de la reproducción, el cruce, y la mutación,

$$m(H, t + 1) \geq m(H, t) \frac{f(H)}{f'} \left[1 - p_c \frac{\Delta(H)}{L - 1} - o(H) \cdot p_m \right]$$

Ejemplos de Aplicación

Búsqueda en un espacio de soluciones

- Supongamos que se desea maximizar la función: $F(x) = \text{Sen}(x\pi/256)$ en el intervalo $[0,255]$.
- *Codificación.* Como las soluciones varían entre 0 y 255, se pueden utilizar cadenas binarias de longitud 8. Por ejemplo, 00010100.

Ejemplos de Aplicación

- *Función de evaluación.* La función de evaluación es la misma función dada, y se utilizará como función de aptitud la diferencia entre el valor máximo (1) y el valor para la función dada: $g(x)=1-F(x)$.
- *Población inicial.*

Individuo	x	F(x)
10111101	189	0.733
11011000	216	0.471
01100011	99	0.937
11101100	236	0.243

Ejemplos de Aplicación

Problema de optimización combinatoria

- El *Agente Viajero*: Dadas N ciudades, el viajero de comercio debe visitar cada ciudad una vez teniendo en cuenta que el costo total del recorrido debe ser mínimo.

$G=(N, A)$

$N=\{1, \dots, n\}$: conjunto de n nodos (vértices)

$A=\{a_{ij}\}$: matriz de adyacencia.

$$d_{ij} = \begin{cases} \infty & \text{Si } a_{ij} = 0 \\ L_{ij} & \text{Si } a_{ij} = 1 \end{cases}$$

L_{ij} : distancia entre las ciudades i y j .

Ejemplos de Aplicación

- Suponiendo que las ciudades son numeradas desde 1 hasta n , una solución al problema puede expresarse a través de una matriz de estado E

$$e_{ij} = \begin{cases} 1 & \text{Si la ciudad } j \text{ fue la } i^{\text{ésima}} \text{ ciudad visitada} \\ 0 & \text{En otro caso} \end{cases}$$

- La matriz E permite definir un arreglo unidimensional V de dimensión n ;

$$v_{ij} = i \quad \text{Si la ciudad } i \text{ fue la } j^{\text{ésima}} \text{ ciudad visitada}$$

Ejemplos de Aplicación

- Función objetivo

$$F1 = \sum_{i=1}^n \sum_{k=1}^n \sum_{j=1}^n L_{ik} e_{ij} e_{kj+1}$$

$$F2 = C \left(\left| \sum_{i=1}^n \sum_{j=1}^n e_{ij} - n \right| + \sum_{i=1}^n \left| \sum_{j=1}^n e_{ij} - 1 \right| + \sum_{j=1}^n \left| \sum_{i=1}^n e_{ij} - 1 \right| \right)$$

$$FC = F1 + F2$$

$C = n * \text{Max}(L_{ik})$ factor de penalización.

Ejemplos de Aplicación

- Codificación de los individuos:
 - Una solución viene dada como un recorrido que cumple con las restricciones del problema.
 - Se puede realizar por medio del arreglo unidimensional V de longitud n
- Función Objetivo: F1 o FC
- Operadores genéticos:
 - Si es F1 => Inversión.

LIMITACIONES

- FUNCION DE APTITUD SIEMPRE POSITIVA
- NUMERO DE GENERACIONES FINITO
- CODIFICACION BINARIA ES LA COMUN
- NO CONSIDERA RESTRICCIONES
- PROBLEMAS DE DIVERSIDAD:
SUPERINDIVIDUOS, CONVERGENCIA
PREMATURA, PRESION SELECTIVA, MECANISMOS
DE ESCISION, CONTROLANDO EDADES, ETC.
- PROBLEMAS DE REPRESENTACION:
COMPLETITUD, COHERENCIA, UNIFORMIDAD, ETC.

AVANCES

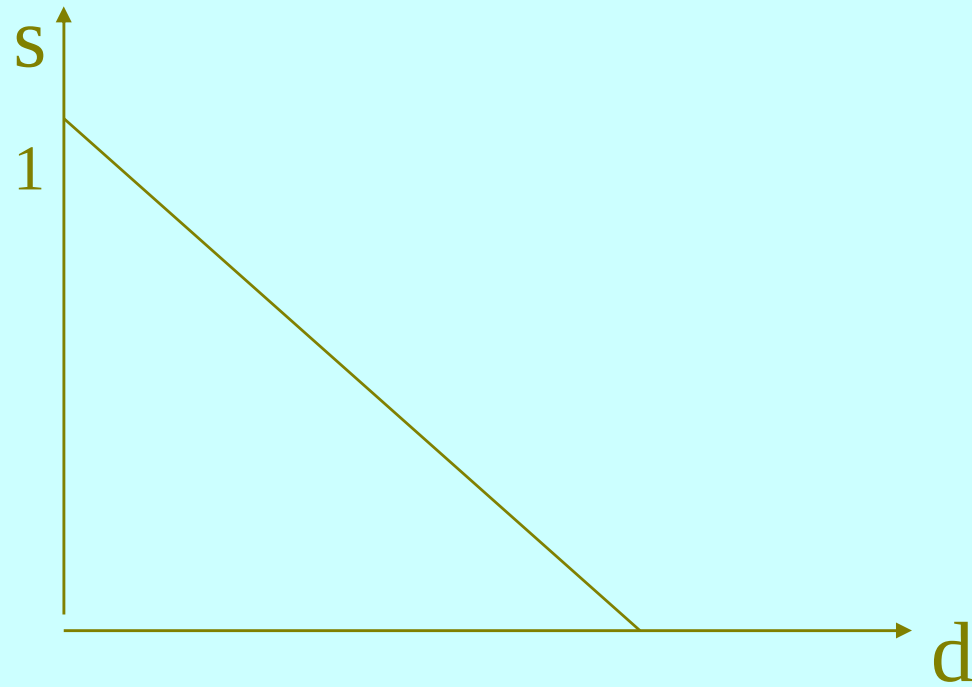
- FUNCIONES MULTIOBJETIVOS
=>conjunto de Paretos
- AG PARALELOS
- META-ALGORITMOS GENETICOS
=>APRENDIZAJE
- NICHOS Y ESPECIES

NICHOS Y ESPECIES

- Diferencia entre ambiente y organismo
- Estables subpoblaciones: especies
- Sirven a diferentes subdominios de una función: nichos
- Metafora de compartir: vecinos

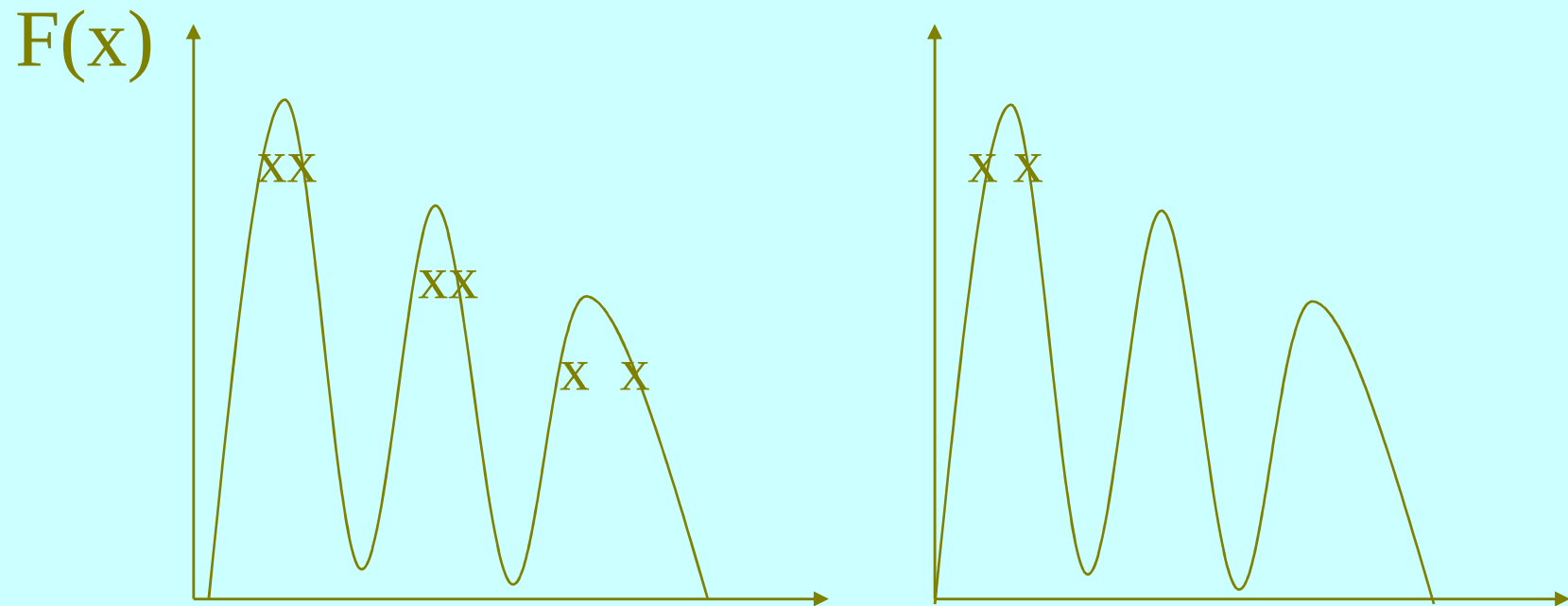
$$Fs(x_i) = F(x_i) / \sum_j s(d(x_i, x_j))$$

NICHOS Y ESPECIES



- Cruce entre individuos semejantes

NICHOS Y ESPECIES



PROBLEMAS MULTIOBJETIVOS

- Muchos problemas de optimización se reducen a la consideración de un solo criterio, mientras que en otros problemas es necesario evaluar varios criterios.

problemas multicriterios o multiobjetivos

- Las técnicas de optimización convencional, tales como los métodos basados en gradiente, y unos menos convencionales, tales como recocido simulado, son difíciles de extender a casos verdaderamente multiobjetivos, ya que estos no fueron diseñados para evaluar esto.

PROBLEMAS MULTIOBJETIVOS

En el mundo real, la mayor parte de los problemas tienen varios objetivos (posiblemente en conflicto entre sí) que se desea sean satisfechos de manera simultánea. Muchos de estos problemas suelen convertirse a mono-objetivo transformando todos los objetivos originales, menos uno, en restricciones adicionales.

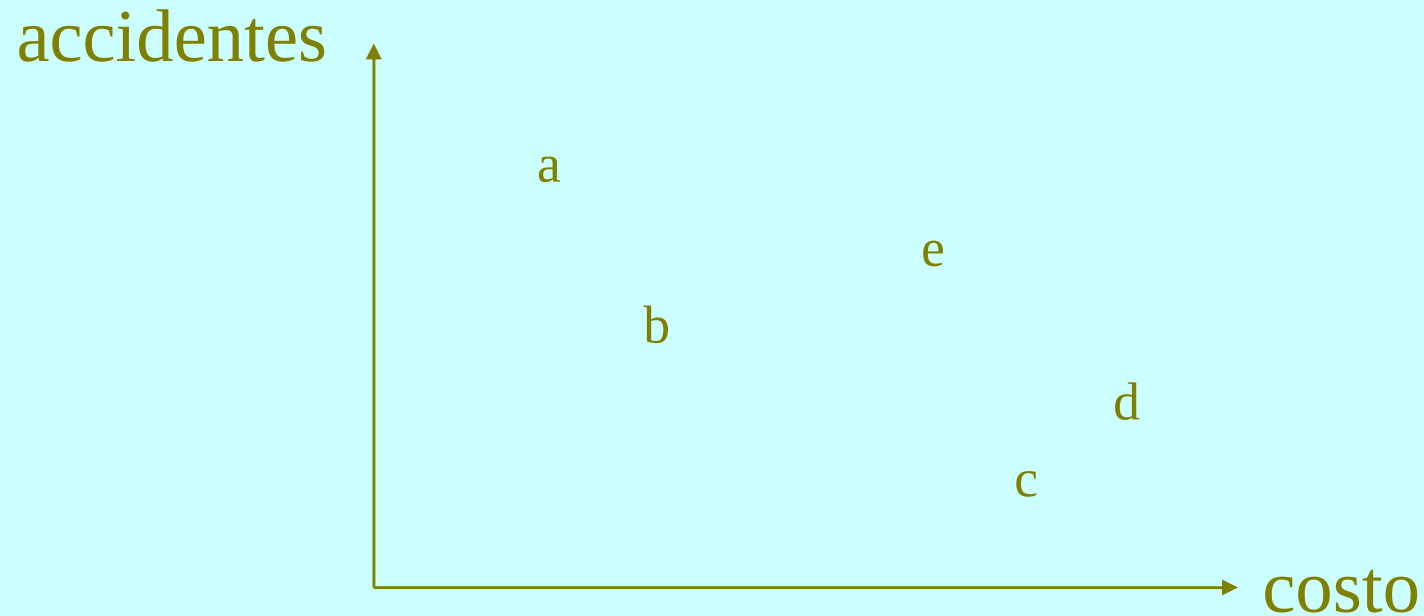
Hay tres tipos de situaciones que pueden presentarse en un problema multiobjetivo:

- Minimizar todas las funciones objetivo.
- Maximizar todas las funciones objetivo.
- Minimizar algunas funciones y maximizar otras.

PROBLEMAS MULTIOBJETIVOS

- Ejemplo: fabrica de cauchos que deben minimizar dos costos:
 - numero de accidentes y
 - costo de producir cauchos
- Escenarios:
 - $a = (2,10)$
 - $b = (4,6)$
 - $c = (8,4)$
 - $d = (9,5)$
 - $e = (7,8)$

PROBLEMAS MULTIOBJETIVOS



Individuos no dominados: a, b, c!!!

Problemas de Localización Multiobjetivo

DEFINICIÓN:

Escoger un conjunto de lugares para situar varias instalaciones que prestarán su servicio a un conjunto de clientes con el objetivo de optimizar uno o más criterios, económicos o de otra índole.

- La función objetivo es generalmente el costo de operación (fijo y variable)
- En otros contextos es necesario considerar otras funciones objetivo: distancia, cobertura, tiempo de respuesta, equidad, etc.
- Criterios en conflicto
 - Costo
 - Cobertura

Problemas de Localización Multiobjetivo

Información para el modelo

- I, i : Conjunto lugares de localización de las instalaciones
m instalaciones
- J, j : Conjunto e índice de los lugares de demanda, n clientes
- f_i : Costo fijo de operar una instalación en el lugar i.
- c_{ij} : Costo de atender toda la demanda del lugar j desde la instalación i.
- d_j : Demanda del cliente j.
- h_{ij} : Distancia entre i y j

COMPONENTES DEL PROBLEMA DE COBERTURA

- D_{max} : Distancia máxima de cobertura.
- Q_j : $\{i: h_{ij} \leq D_{max}\}$, conjunto de instalaciones que pueden atender la demanda del nodo j cumpliendo con la distancia máxima de cobertura.

Problemas de Localización Multiobjetivo

Variables de decisión

$$y_i = \begin{cases} 1, & \text{si se abre la bodega } i, \\ 0, & \text{de lo contrario.} \end{cases}$$

$$x_{ij} = \begin{cases} 1, & \text{si el cliente } j \text{ es atendido por la bodega } i, \\ 0, & \text{de lo contrario.} \end{cases}$$

Problemas de Localización Multiobjetivo

Función objetivo y restricciones

$$\text{mín} \quad z_1 = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} c_{ij} x_{ij} + \sum_{i \in \mathcal{I}} f_i y_i$$

$$\text{máx} \quad z_2 = \sum_{j \in \mathcal{J}} d_j \sum_{i \in Q_j} x_{ij}$$

$$\sum_{i \in \mathcal{I}} x_{ij} = 1 \quad , \quad j \in \mathcal{J}$$

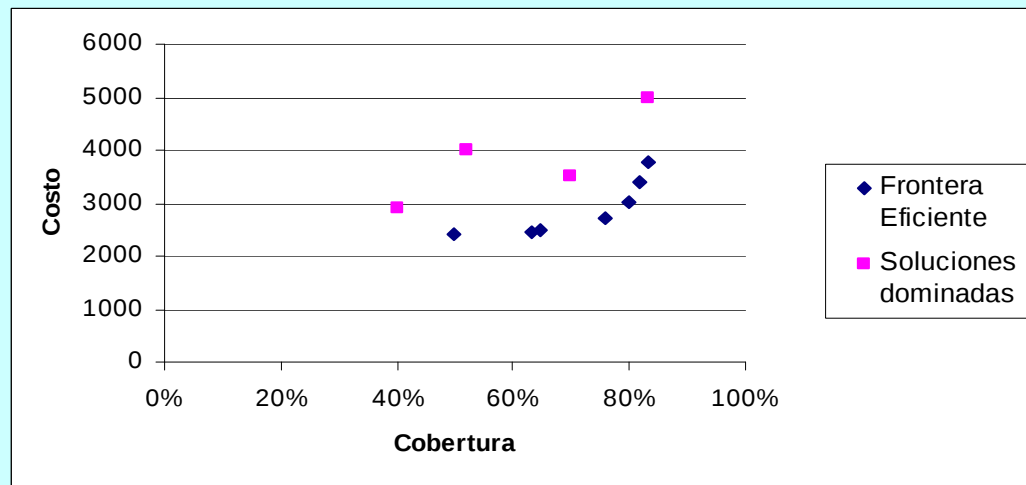
$$x_{ij} \leq y_i \quad , \quad j \in \mathcal{J}, \quad i \in \mathcal{I}$$

$$x_{ij} \in \{0, 1\}, \quad j \in \mathcal{J}, \quad i \in \mathcal{I}$$

$$y_i \in \{0, 1\}, \quad i \in \mathcal{I}$$

Generalidades Optimización Multiobjetivo

- Los objetivos considerados están en conflicto.
- Se busca un conjunto de soluciones eficientes, en las cuales no sea posible mejorar el valor de una de las funciones objetivo sin deteriorar el desempeño de la otra.



PROBLEMAS MULTIOBJETIVOS

TEORIA DE PARETO

- A diferencia de la optimización de un único objetivo, la solución a un problema multiobjetivo no es un único punto, sino una familia de puntos conocida como el "**Conjunto Optimo de Pareto**".
- Cada punto en esa superficie es óptimo en el sentido de que no pueden realizarse mejoras en un componente del vector de costo que no conduzca a la degradación en al menos uno de los componentes restantes.
- La noción de optimalidad de Pareto es sólo un primer paso hacia la solución práctica de un problema multiobjetivos; también envuelve la **escogencia de una única solución** compromiso del "Conjunto Optimo de Pareto" de acuerdo a alguna información de preferencia.

PROBLEMAS MULTIOBJETIVOS

TEORIA DE PARETO

Decimos que un vector de variables de decisión $x^* \in \alpha$ (α es la zona factible) es un óptimo de Pareto si no existe otro $x \in \alpha$ tal que $f_i(x) \leq f_i(x^*)$ para toda $i = 1, \dots, k$

x^* es un óptimo de Pareto si no existe ningún vector factible de variables de decisión $x^* \in \alpha$ que decremente algún criterio sin causar un incremento simultáneo en al menos un criterio. Desafortunadamente, este concepto casi siempre produce no una solución única sino un conjunto de ellas a las que se les llama conjunto de Pareto.

Los vectores x^* correspondientes a las soluciones incluidas en el conjunto de óptimos de Pareto son llamados no dominados. La gráfica de las funciones objetivo cuyos vectores no dominados se encuentran en el conjunto de óptimos de Pareto se denominan frente de Pareto.

PROBLEMAS MULTIOBJETIVOS

TEORIA DE PARETO

"Conjunto Optimo de Pareto" esta compuesto por elementos que son soluciones no-inferiores (no-dominados) para el problema multiobjetivos:

Concepto de Inferioridad: Un vector $\mathbf{u} = (u_1, \dots, u_n)$ se dice es inferior a $\mathbf{v} = (v_1, \dots, v_n)$ si $\mathbf{v}$ es parcialmente menor que $\mathbf{u}$ ($\mathbf{v} p < \mathbf{u}$); es decir:

$$i = 1, \dots, n, v_i \leq u_i \quad i = 1, \dots, n : v_i < u_i \quad (1)$$

Concepto de Superioridad: Un vector $\mathbf{u} = (u_1, \dots, u_n)$ se dice es superior a $\mathbf{v} = (v_1, \dots, v_n)$ si $\mathbf{v}$ es inferior a $\mathbf{u}$.

ESQUEMAS RESOLUCION PROBLEMAS MULTIOBJETIVOS

- No basadas en Pareto
 - Ordenamiento lexicográfico
 - Suma lineal de pesos
 - VEGA
 - Método ε -constraint
 - Teoría de juegos
- Basadas en Pareto
 - Jerarquización de Pareto “pura”
 - MOGA
 - NSGA y NSGA II

PROBLEMAS MULTIOBJETIVOS

ENFOQUES BASADOS EN LA DIVISIÓN DE LA POBLACIÓN

1.- Dividir población en n subpoblaciones $S_1, \dots, S_n$.

% se va a iterar un AG con una FO distinta en c/u.

2.- Definir vector de prioridad para las FOs.

3.- Repetir Hasta (Evaluar todas las FOs)

3.1 Para cada S_i , $\forall i = 1, \dots, n$, usar la FO correspondiente según el vector de prioridad.

3.1.1 Iterar el AG para esa FO_i según un número y dado de iteraciones en cada S_i .

3.1.2 Reemplazar X % de individuos en cada población S_i que cumplen mal la FO actual.

PROBLEMAS MULTIOBJETIVOS

ENFOQUES BASADOS EN LA DIVISIÓN DE LA POBLACIÓN

- 1.- Dividir población en n subpoblaciones $S_1, \dots, S_n$.
- 2.- Iterar un AG con una FO distinta en c/u.
- 3.- Unir las subpoblaciones S_i , $i = 1, \dots, n$.
- 4.- Repetir Hasta (# Iteraciones dada o Población Converge)
 - 4.1 Seleccionar FO_i según probabilidad de escogencia.
 - 4.2 Disminuir probabilidad de escogencia de la FO_i seleccionada.
 - 4.3 Aplicar AG a población global usando FO_i seleccionada. Hacer un reemplazo parcial.

PROBLEMAS MULTIOBJETIVOS

ENFOQUES BASADOS EN POBLACIÓN TOTAL

- 1.- Inicio.
- 2.- Repetir Hasta (Que c/FO sea escogida un mínimo número de veces ó población converja).
 - 2.1 Escoger una FO (aleatoriamente o cíclicamente).
 - 2.2 Iterar el AG en la población usando la FO escogida.

ϵ -restricciones

- Formular un problema sin restricciones con una sola función objetivo (minimizar o maximizar) e incluir la otra función objetivo como restricción adicional y parametrizar sobre dicha restricción.
- Las soluciones encontradas se encuentran todas en la frontera eficiente real pero no se puede garantizar que se encuentran todas las soluciones

Enfoque ε -restricciones: caso localización

Formulación: Minimizar costo, garantizando cierta cobertura

$$\text{mín} \quad z_1 = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} c_{ij} x_{ij} + \sum_{i \in \mathcal{I}} f_i y_i$$

Sujeto a,

$$\sum_{i \in \mathcal{I}} x_{ij} = 1 \quad , \quad j \in \mathcal{J}$$

$$x_{ij} \leq y_i \quad , \quad j \in \mathcal{J}, \quad i \in \mathcal{I}$$

$$\sum_{j \in \mathcal{J}} d_j \sum_{i \in \mathcal{Q}_j} x_{ij} \geq v \sum_{j \in \mathcal{J}} d_j \quad , \quad j \in \mathcal{J}, \quad i \in \mathcal{I}$$

$$x_{ij} \in \{0, 1\} \quad , \quad j \in \mathcal{J}, \quad i \in \mathcal{I}$$

$$y_i \in \{0, 1\} \quad , \quad i \in \mathcal{I}$$

Enfoque ε -restricciones: caso localización

Formulación: Maximizar cobertura, con cierto presupuesto.

$$\text{mín} \quad z_1 = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} c_{ij} x_{ij} + \sum_{i \in \mathcal{I}} f_i y_i$$

Sujeto a,

$$\sum_{i \in \mathcal{I}} x_{ij} = 1, \quad j \in \mathcal{J}$$

$$x_{ij} \leq y_i, \quad j \in \mathcal{J}, \quad i \in \mathcal{I}$$

$$\sum_{j \in \mathcal{J}} d_j \sum_{i \in \mathcal{Q}_j} x_{ij} \geq v \sum_{j \in \mathcal{J}} d_j, \quad j \in \mathcal{J}, \quad i \in \mathcal{I}$$

$$x_{ij} \in \{0, 1\}, \quad j \in \mathcal{J}, \quad i \in \mathcal{I}$$

$$y_i \in \{0, 1\}, \quad i \in \mathcal{I}$$

PROBLEMAS MULTIOBJETIVOS

ENFOQUES BASADOS EN LA TEORIA DE PARETO

Para obtener soluciones no-inferiores se pueden usar los AGs:

- manteniendo una población de soluciones, los AGs pueden buscar muchas soluciones no-inferiores en paralelo.

ENFOQUES BASADOS EN LA TEORIA DE PARETO

Enfoque en dos fases: en la primera se clasifican los individuos en no-inferiores o no (no-dominados o dominados), para cada una de las FOs que conforman la FMO. En la segunda, se utilizan los individuos no-dominados como los padrotes para la fase de reproducción.

Fase de Clasificación :

- 1.- Generar población inicial.
- 2.- Evaluar los individuos para c/FO y establecer su clasificación en c/FO.
- 3.- Seleccionar individuos no-dominados (Según c/u de las FOs).
- 4.- Asignar rango 1 a los individuos no-dominados y remover de la disputa, luego se encuentra un nuevo conjunto de individuos no-dominados con rango 2, y así sucesivamente. Esto se hace para c/FOs.

ENFOQUES BASADOS EN LA TEORIA DE PARETO

Fase de Optimización :

- 1.- Seleccionar los individuos para reproducción por algún mecanismo clásico (ruleta, elitesco).
- 2.- Reproducir con operadores clásicos, usando los individuos no-dominados como los padrotes.

Macro Algoritmo:

- 1.- Generar población.
- 2.- Repetir Hasta que población generada sea igual a la anterior o sea homogénea.
 - 2.1.- Fase de Clasificación: Agrupar individuos en dominados y no-dominados para c/FO.
 - 2.2.- Fase de Optimización: Generar nueva población según un AG clásico.

Paralelismo Implícito

- La eficacia a los AGs se basa en que, aunque el AG sólo procesa n estructuras en cada generación, se puede probar que, bajo hipótesis muy generales, se procesan de modo útil al menos n^3 esquemas.
- Este **paralelismo implícito** se consigue sin ningún dispositivo o memoria adicionales, sólo con la propia población.
- A pesar de la ruptura de los esquemas largos de orden alto por los operadores de cruce y mutación, los algoritmos genéticos procesan inherentemente una gran cantidad de esquemas mientras procesan una cantidad relativamente pequeña de cadenas.

Algoritmos genéticos paralelos

- Paralelismo inherente proporcional al tamaño de la población
- Sugiere que se puede incrementar el tamaño de la población sin afectar la performance
- Problemas en performance generados por sincronización y envío de mensajes
- Grandes poblaciones convergen mas lentamente

Modelos isla

- 6400 individuos en 64 procesadores, por ejemplo
 - Dividir la población total en subpoblaciones de 100 individuos cada una
- Cada subpoblación ejecuta un algoritmo genético
- Cada x cantidad de generaciones, las subpoblaciones intercambian algunos individuos intercambiando material genético
- Si un gran número de individuos migra en cada generación, se pierden las diferencias entre las islas
- Si la migración es poco frecuente, podría llevar a que cada población converja prematuramente

Algoritmos genéticos celulares

- 2500 procesadores dispuestos en una grilla de 50 x 50
- Los procesadores solo se comunican con sus vecinos inmediatos
- Cada string (cada procesador) se fija en sus vecinos inmediatos y elige el mejor individuo que encuentra
- Recombina su individuo con el elegido del vecino
- Luego de algunas generaciones hay algunos focos conteniendo individuos similares

Paralelismo en AG

Repetir

- Para cada individuo i

 - Evaluar $f(i)$

 - Transmitir $f(i)$ para todos los individuos j en el vecindario

 - Elegir un individuo j para combinar basado en fitness

 - Pedir el individuo j

 - Reproducir usando los individuos i y j

Hasta que la variación en la población es pequeña