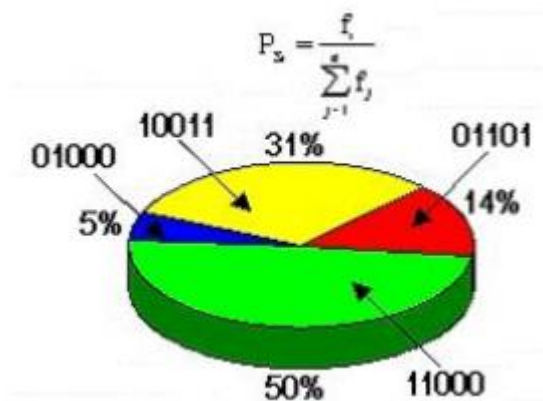


## Operadores genéticos: selección, cruce y mutación

### Selección

El principio detrás del algoritmo genético es esencialmente la selección natural de Darwin. La selección provee la fuerza motora a un algoritmo genético. La presión selectiva es crítica. En un extremo, la búsqueda terminará prematuramente; mientras que en el otro, el progreso será más lento que el necesario. Típicamente, se sugiere una baja presión selectiva al comienzo de los algoritmos genéticos en favor de una amplia exploración del espacio de búsqueda, mientras se recomienda una alta presión selectiva al final, para explotar las regiones más prometedoras del espacio de búsqueda. La selección dirige la búsqueda del GA a través de regiones prometedoras del espacio de búsqueda. Se presentan tres cuestiones básicas relacionadas con la fase de selección: Espacio de muestreo. Mecanismo de muestreo. Probabilidad de selección. Cada una de ellas ejerce una influencia significativa sobre la presión selectiva y por consiguiente en el comportamiento del algoritmo genético.



### Cruce

La implementación del método de cruce sigue el esquema mostrado en el ejemplo del apartado de diseño del cruce. En primer lugar, se crean las listas que contienen los genes, los números de pieza y los operadores de ambos padres. En las listas números de pieza no existen huecos, mientras que en las de los operadores, sí. Posteriormente se crea un diccionario vacío cuyas claves son cadenas y los valores son listas de cadenas. Después, se obtiene de manera aleatoria dos puntos cualesquiera de las listas compactadas (las listas de números de pieza) para concretar los puntos de corte. Una vez se conocen estos puntos de corte, se procede a realizar un recorrido por la lista de operadores del progenitor número uno y se determina qué operadores deben ser cambiados en el cruce. Una vez se

conoce qué operadores deben ser cambiados, se recorren las dos listas de operadores (las correspondientes a ambos progenitores) y se cambian los que estén marcados, de manera ordenada (i.e., el tercer operador del cromosoma del progenitor uno sólo puede ser cambiado por el tercer operador del cromosoma del progenitor dos). Para la parte de números de pieza, en primer lugar se realiza el intercambio directo de los genes que se encuentran dentro del segmento delimitado por los puntos de corte aleatoriamente obtenidos. En segundo lugar, se introducen en el diccionario las distintas relaciones que tienen los genes y se llama a la función `unfold_relationships` para desplegar las relaciones en el diccionario. En tercer lugar, se realiza el recorrido por las listas que determina si faltan números de pieza o existen repetidos y se modifica utilizando las relaciones del diccionario para legalizar los cromosomas. A continuación se muestra en pseudocódigo el funcionamiento del recorrido:

Desde `i=0` hasta `primer_punto_corte - 1`

y desde `i=segundo_punto_corte + 1` hasta FIN:

Desde `j=primer_punto_corte` hasta `segundo_punto_corte` :

Si `lista_progenitor1[i]` y `lista_progenitor1[j]` son iguales:

Para cada elemento relacionado con el elemento `lista_progenitor1[j]`

Si ese elemento no está en el gen `lista_progenitor1`:

Intercambiar `lista_progenitor1[i]` por el elemento

Encontrar el elemento en la parte externa<sup>17</sup> de `lista_progenitor2` e

intercambiar por el elemento que había antes en `lista_progenitor1[i]`

De esta manera, recorriendo únicamente la primera lista de número de piezas, se consigue legalizar ambos cromosomas, ya que los cambios en el primero se pueden aprovechar porque son los cambios inversos que se deben realizar en el segundo

## Mutación

La implementación del método de mutación sigue la línea descrita en el diseño del método. El método implementado `mutation(...)` devuelve un valor booleano que indica si la mutación se ha podido efectuar o no. La mutación no se puede efectuar si para todos los genes que se encuentran entre los dos genes seleccionados, ambos inclusive.

Seleccionar dos genes de manera aleatoria ( $p_1$  y  $p_2$ )

Si los dos genes contienen números de pieza:

Intercambiar los genes

Devolver True

Si el primer gen contiene un operador:

Intercambiar los genes

Devolver True

Si el primer gen contiene un número de pieza y el segundo contiene un operador:

Si cada gen entre  $p_1$  y  $p_2$ , ambos inclusive, cumple que  $N_o \leq N_p - 3$  :

Intercambiar los genes

Devolver True

Si no:

Devolver False

De esa manera, se obtiene un método que intercambia dos genes aleatoriamente en el caso de que se pueda. Para asegurar la mutación para un individuo dado, es necesario realizar un bucle que lance este método hasta que devuelva True, que significa que la mutación se ha efectuado con éxito.