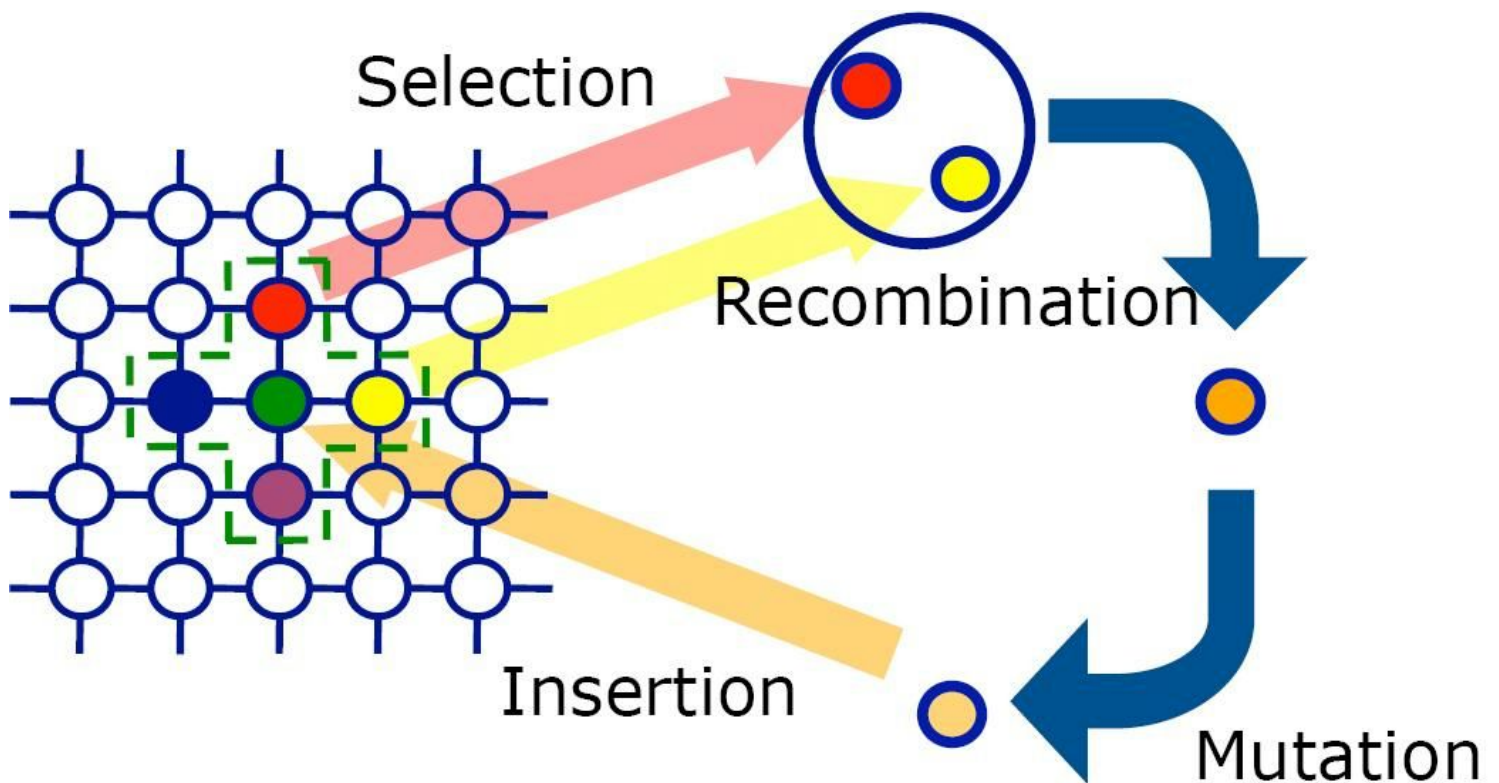


Bloque 1.

Algoritmos Genéticos.

Práctica 2.



Introducción:

El propósito de la práctica es profundizar un poco más en los distintos tipos de operadores de los que se compone el algoritmo genético, visitando las versiones más conocidas de los mismos.

Desarrollo de la práctica:

El problema que debemos resolver en esta práctica es un modelo de simulación de tráfico basado en autómatas celulares.

				1				5						
				2				6						
				3				7						
				4				8						
9	10	11	12	13	14	15	16	17	18	19	20	21	22	
				23				27						
				24				28						
				25				29						
				26				30						
31	32	33	34	35	36	37	38	39	40	41	42	43	44	
				45				49						
				46				50						
				47				51						
				48				52						

Para **representar** el modelo utilizaremos una matriz binaria de 4 filas que representan los 4 semáforos horizontales. Los semáforos verticales serán simplemente el valor contrario a su semáforo más cercano.

Las columnas de dicha matriz dependen del tiempo de simulación que deseemos utilizar, si deseamos simular dos horas (7200 sg) la matriz será de 4x720 ya que los semáforos cambian cada 10 segundos.

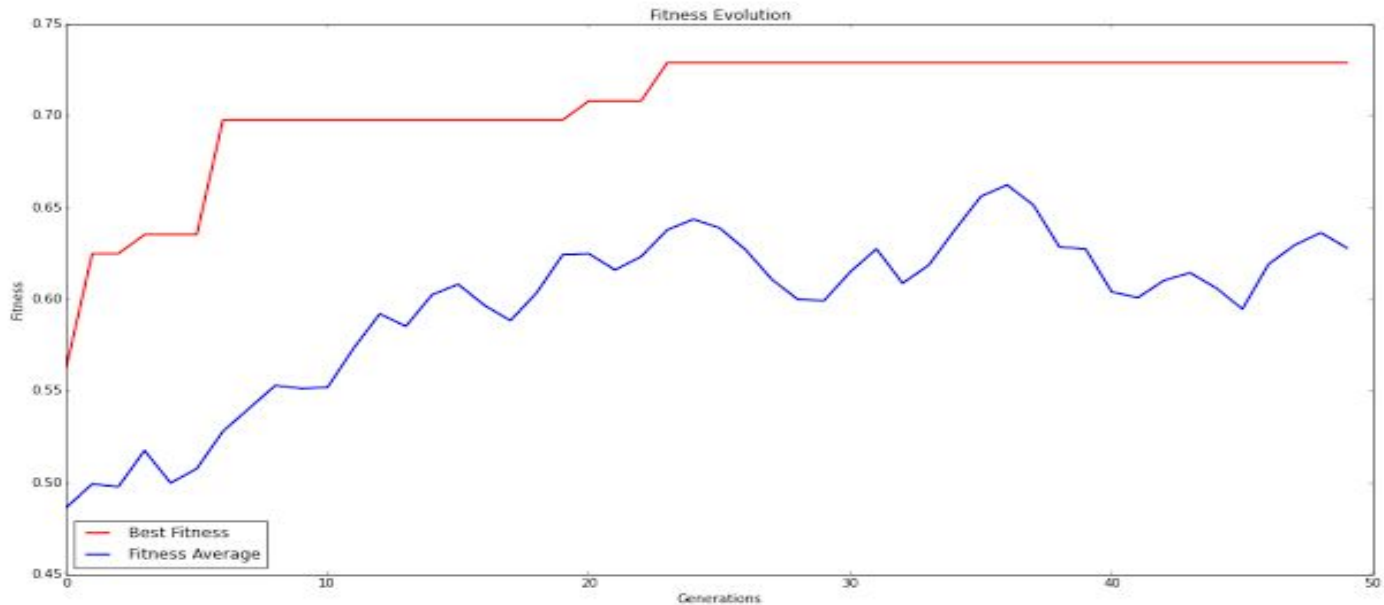
La **función de fitness** para evaluar los cromosomas simplemente se basa en una simulación de tráfico, donde el valor de fitness serán los coches que salen de las calles entre todos los que entran.

Parámetros:

- Probabilidad de mutación: 40%
- Número de generaciones: 50
- Número de individuos seleccionados: 12
- Número de individuos: 20

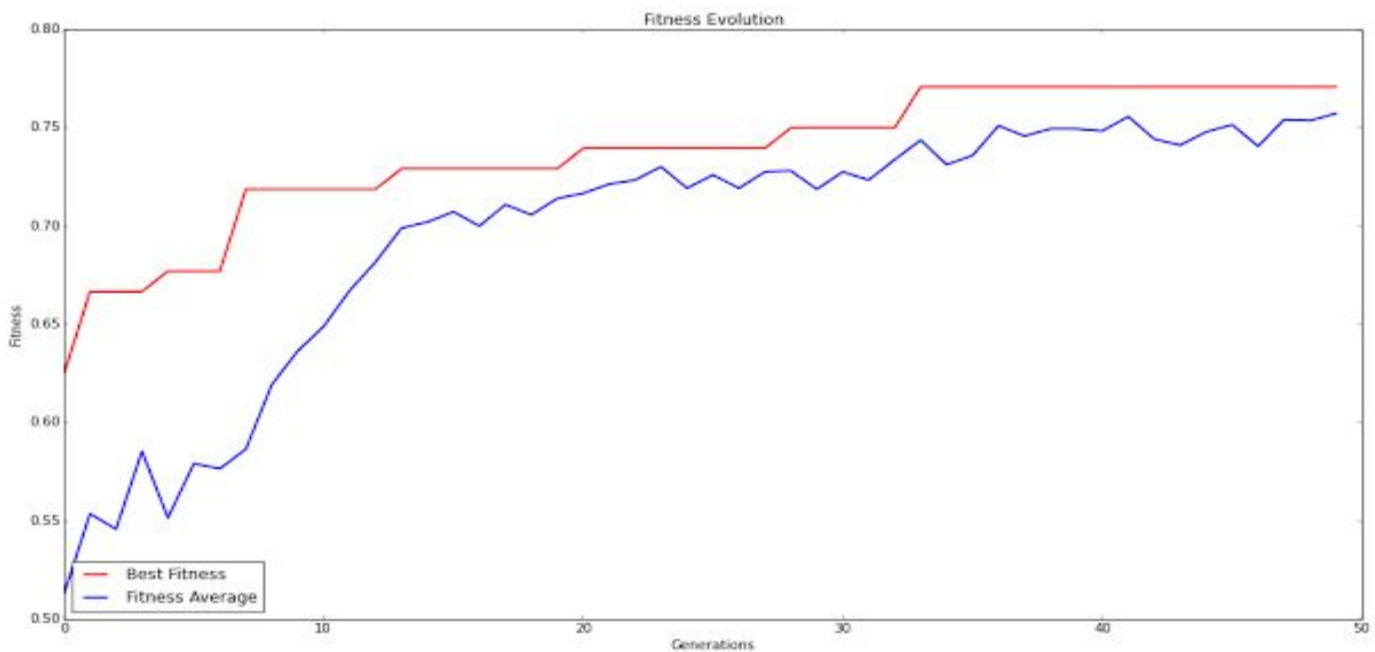
Operador Selección (Selección por truncado vs Selección por torneo):

Torneo: Con este método de selección podemos observar subidas y bajadas en la media de los fitness debido a la posibilidad de contemplar individuos a priori menos aptos. Esto nos permite una mayor variedad en la población y por lo tanto no se homogeneizarán en torno a un solo fitness. Por otro lado pagamos un sobre coste en el número de generaciones que necesitaremos para alcanzar una solución.



Torneo: En la selección por torneo pasa la contrario, al desechar un número fijo de individuos, reducimos la variedad de la población, lo que nos puede llevar a homogeneizar los valores de fitness de toda la población.

La ventaja es la velocidad a la cual podemos alcanzar una solución, siendo esta mucho mas rápida que en la selección por torneo.

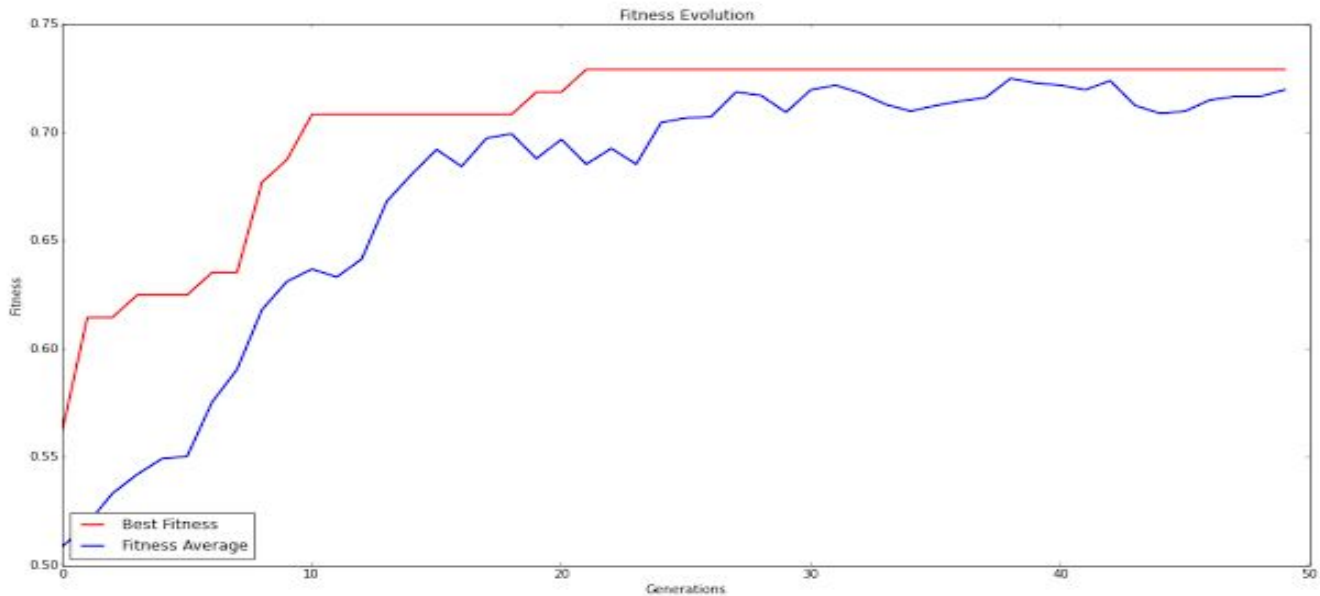


Operador Cruce (Uniforme vs ThreeParentCrossover) [Simulados con selección por torneo]:

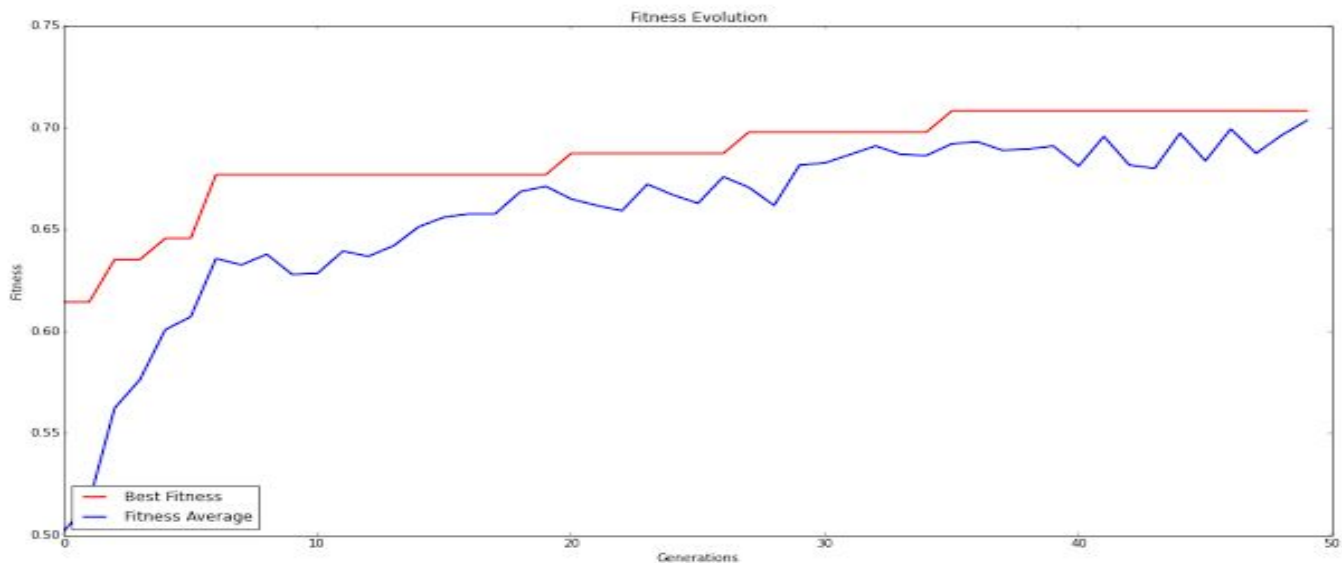
Uniforme: En el cruce uniforme tenemos el 50% de posibilidades de en cada gen coger el del padre o el de la madre, esto nos permite conformar un nuevo individuo con una variación grande con padres.

ThreeParentCrossover: En el ThreeParentCrossover preguntamos gen a gen si son iguales en la madre y en el padre, en caso verdadero, ese sera el gen del nuevo individuo, en caso contrario, se cogerá el gen de un 3 padre.

Uniforme



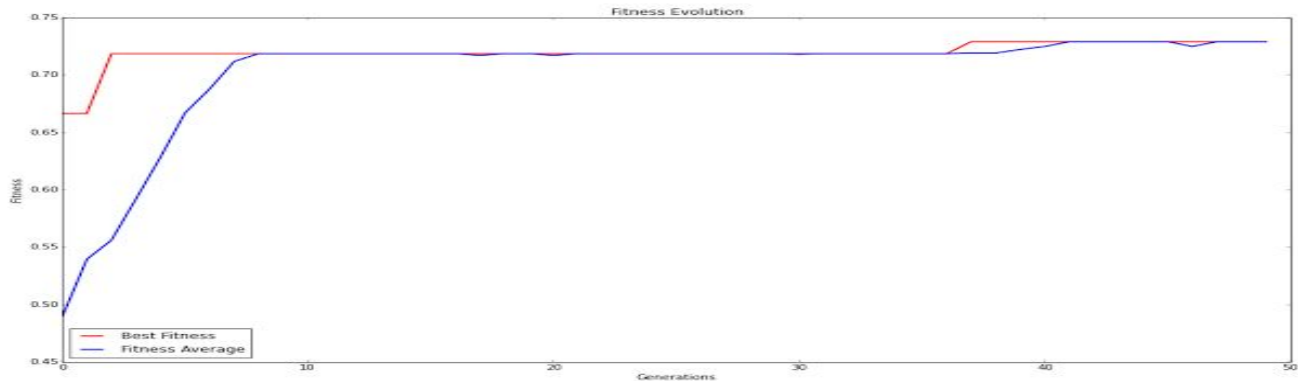
Three parent crossover



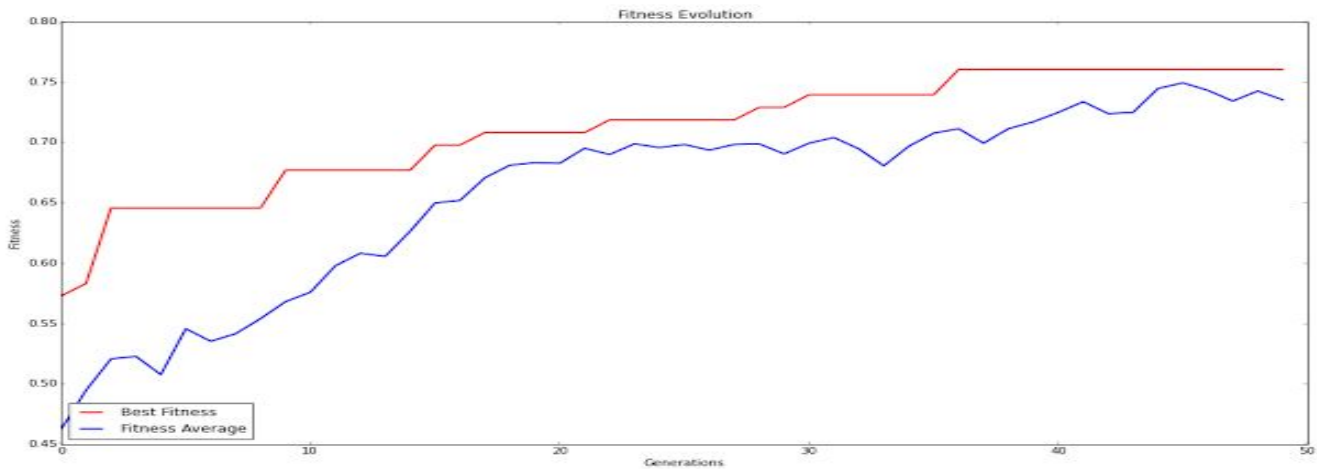
Operador Mutación:

Bit String Mutation: Se cambia el valor en un gen aleatorio según una probabilidad de mutación definida en los parámetros del programa. Si bajamos la probabilidad de mutación a 0 la población evolucionará solo de los valores de cruce de los individuos existentes, es decir, reducimos la diversidad genética. Por el contrario, cuanto mayor es, mas variedad de individuos habrá y más oscilación observaremos en la media de fitness.

Probabilidad de mutación = 0



Probabilidad de mutación = 50



Probabilidad de mutación = 100

