

# Reporte práctica diez: Algoritmos genéticos

José Anastacio Hernández Saldaña

Posgrado de Ingeniería de Sistemas

1186622

jose.hernandezsld@uanl.edu.mx

17 de octubre de 2017

## Resumen

Este es un reporte sobre la práctica diez con respecto al tema de algoritmos genéticos que se realizó en la clase de Simulación de sistemas, cómputo paralelo en R.

## 1. Tarea: Paralelización del algoritmo genético

Para esta práctica se realizó una implementación de un algoritmo genético. Los algoritmos genéticos imitan la evolución biológica para resolver problemas de búsqueda y optimización. Trabajan con una población de individuos, cada uno de los cuales representa una solución factible para un problema determinado.

Se simuló su aplicación en un problema de la mochila, conocido en inglés como *Knapsack problem*, donde se tiene un contenedor de capacidad  $C$  limitada y se tienen  $N$  elementos que se pueden meter en el contenedor, cada elemento tiene un valor de beneficio  $b_i$  y un valor de capacidad  $c_i$  que ocupan en el contenedor, se busca tener en el contenedor aquellos elementos que nos maximizan el beneficio  $\max(\sum_{i \in N} x_i b_i)$ , donde  $x_i$  es la decisión de tener o no el elemento en el contenedor sin exceder la capacidad  $C$ .

Los algoritmos genéticos cuentan principalmente con 4 parámetros a tomar en cuenta para su implementación:

- Tamaño de la población  $T$
- Cantidad de Generaciones  $G$
- Probabilidad de Cruce  $P_c$
- Probabilidad de Mutación  $P_m$

El modelo estudiado para esta práctica, el cual está disponible en la página del curso, tiene definidos los valores de parámetros siguientes: tamaño de la población fijo en  $T = 200$ , cantidad de generaciones fija en  $G = 50$ , probabilidad de cruce distribuida de manera uniforme  $P_c = \frac{1}{T}$  y la probabilidad de mutación fija  $P_m = 0.05$ . Esta implementación está hecha de manera secuencial, por lo que se modificará para tener una versión paralelizada.

### 1.1. Diseño del Experimento

Dentro del código, se encontraron los siguientes procedimientos del algoritmo que son candidatos a ser paralelizados:

1. El procedimiento de selección de cruce y el cruce.
2. El procedimiento de selección de mutación y la mutación.
3. El procedimiento de cálculo de la función objetivo y factibilidad.

Se optó por paralelizar los procedimientos 1 y 2. En el procedimiento 1 se optimizó la selección del los elementos a cruzar por medio de vectores y se paralelizó el procedimiento de cruce, para el procedimiento 2 se realizó tanto la selección de los elementos a mutar y el procedimiento de mutación en la misma paralelización. Para los procedimientos de 3 se decidió utilizarlos de manera secuencial.

Como esta paralelización no modifica el algoritmo genético de tal manera que de mejores resultados, sino que podrá hacer el mismo procedimiento en menos tiempo, se espera que el tiempo sea menor pero que no haya una diferencia significativa en el valor de las soluciones óptimas encontradas. Para medir la calidad de las soluciones entregadas por el algoritmo genético se utilizó una función que calcula el valor óptimo  $vo$  para el problema y se calculó el porcentaje de separación de las soluciones actuales  $va_i$  a ese valor de la siguiente manera  $g_i = \frac{vo - va_i}{vo}$ .

Buscando estos resultados se realizó el experimento con 30 replicas con los parámetros ya mencionados, se utilizó una computadora con las siguientes especificaciones: Procesador Intel Core i7-4790 CPU @ 3.6GHz  $\times$  8 y memoria RAM de 24 GB utilizando solamente los cuatro núcleos físicos disponibles. El código esta disponible en el repositorio git del curso bajo el nombre de Tarea10.r.

## 1.2. Resultados

Los primeros resultados a revisar fueron los resultados del porcentaje de separación y así tener la certeza de que el código realizado no afectó el resultado del algoritmo. En la figura 1 podemos ver la diferencia en el porcentaje de separación para los dos tipos de algoritmos.

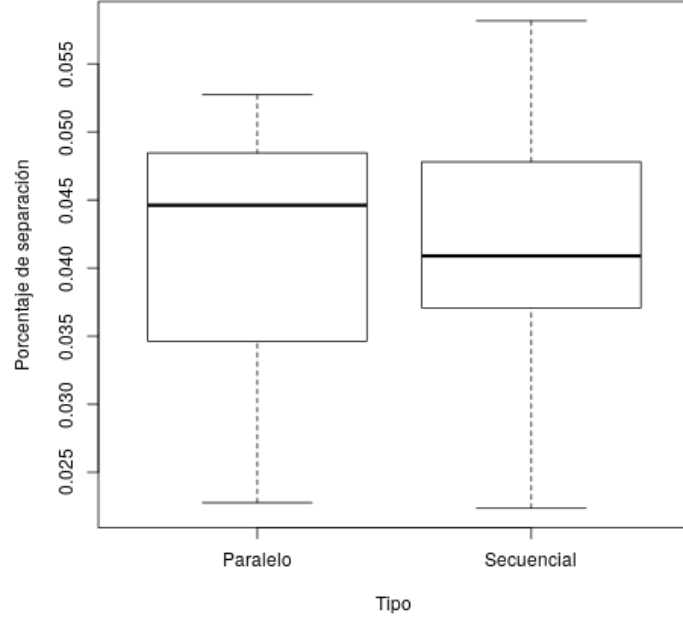


Figura 1: Comparación entre el porcentaje de separación y el tipo de algoritmo.

Se realizó una análisis de la variación de la media para corroborar que las medias son iguales, primero se comprobó por medio de una prueba de Shapiro-Wilk que los datos seguían una distribución normal 2, de donde se obtuvo el valor  $p = 0.17$  por lo que se aceptó la hipótesis nula de que seguían una distribución normal, con esto se realizó una prueba de análisis de varianza, donde se obtuvo el valor  $p = 0.756$  aceptando la hipótesis nula de que las media eran iguales, con esto se comprobó que los procedimientos daban resultados similares y no se afectó su funcionamiento.

El resultado de la comparación del tiempo se puede observar en la figura 3 donde se puede apreciar una diferencia significativa en el tiempo que toma en ejecutarse el algoritmo de poco más de la mitad del tiempo ahorrado.

Con estos resultados se obtuvieron soluciones con una separación similar al valor óptimo en menos tiempo que era lo esperado para este experimento.

## 2. Extra Uno: Agregar procedimiento de selección por ruleta para la reproducción

Tomando como base el experimento anterior, se modificó el procedimiento de selección por ruleta para la reproducción, ya que en la sección anterior se realizó el cruce de las soluciones al azar a partir de una distribución uniforme. Ahora se tomará en cuenta el valor de su función objetivo  $va$  para asignar su probabilidad de selección, esto se realizó sumando los valores de las soluciones y dividiendo cada valor de solución entre el valor de la suma  $p_c = \frac{va}{\sum_{i \in T} (va_i)}$ .

Se utilizaron los mismos parámetros del experimento anterior en el mismo equipo, solo se agregó una métrica más para medir la estabilidad al acercarse al mejor valor, tomando el número de la primer generación  $ge$  donde el valor de la separación al ultimo valor obtenido por el algoritmo es menor al

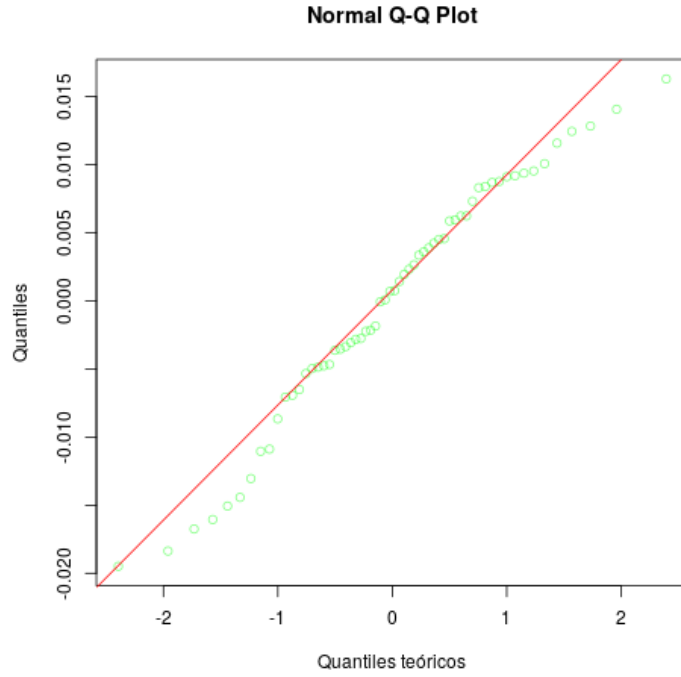


Figura 2: Comparación de los residuales del porcentaje de separación y el tipo de algoritmo.

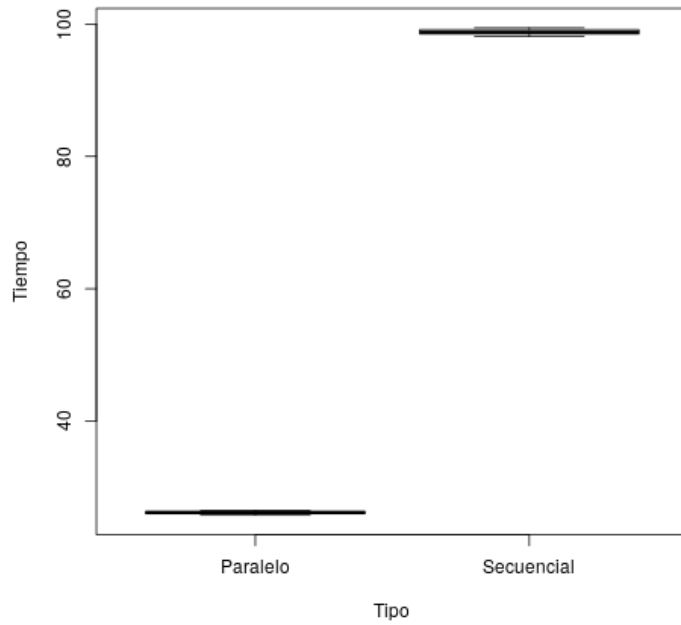


Figura 3: Comparación el tiempo de ejecución y el tipo de algoritmo.

último valor más la desviación estándar  $\min_{j \in G}(g_j) - sd_{j \in G}(g_j)$ , donde  $g_j$  es el menor porcentaje de separación obtenido en cada generación. Este valor nos indica cuando el proceso se estabilizó con respecto al último valor obtenido y así medir en que generación se encuentra cerca del mejor valor que va a obtener.

## 2.1. Resultados

Ahora se comenzó comparando los tiempos obtenidos en función del tipo de algoritmo utilizado como se puede ver en la figura 4. Se aprecia que si hay un incremento en el tiempo de ejecución del algoritmo con selección de ruleta, esto debido al cálculo de las probabilidades de selección.

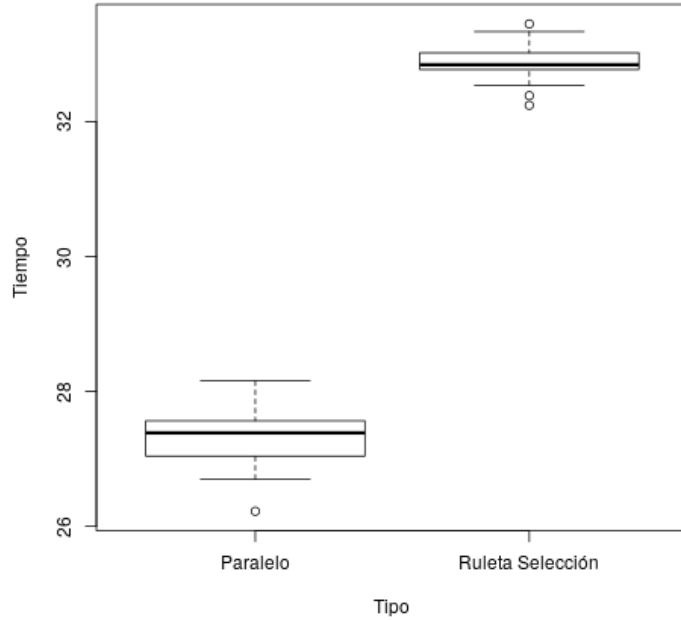


Figura 4: Comparación el tiempo de ejecución y el tipo de algoritmo

El siguiente resultado fue la comparación entre el porcentaje de separación y el tipo de algoritmo usado, como se puede apreciar en la figura 5, no se ve una diferencia significativa en las medias de los valores.

Para corroborar el resultado anterior se realizó un análisis de la variación de la media, primero revisando si los residuos siguen una distribución normal 6, al realizar la prueba de Shapiro-Wilk se obtuvo el valor  $p = 0.796$  por lo que se aceptó la hipótesis nula de que los residuales siguen una distribución normal. Realizando por consiguiente un análisis de varianza, de donde se obtuvo un valor  $p = 0.791$  aceptando la hipótesis nula de que las medias son iguales. Con esto se puede establecer que no hay un efecto significativo en el porcentaje de separación alcanzado por el algoritmo al utilizar el procedimiento de selección por ruleta.

Por último, el resultado de la generación donde queda cerca del mejor valor obtenido. En la figura 7 se puede ver como se va acercando el valor de la separación entre el valor obtenido y el valor óptimo a cero, y algunos valores que están cercanos y marcan la estabilización del valor de la solución.

En la figura 8 vemos la comparación entre los 2 algoritmos, y tampoco se encontró diferencia significativa en los valores de las medias.

De igual manera se realizó un análisis de la variación de la media, primero revisando si los residuos siguen una distribución normal 9, al realizar la prueba de Shapiro-Wilk se obtuvo el valor  $p = 0.0149$  por lo que se rechazó la hipótesis nula de que los residuales siguen una distribución normal. Así que se realizaron pruebas no paramétrica para comparar los valores de las medias por medio de la prueba Kruskal-Wallis, de donde se obtuvo el valor  $p = 0.276$  con que se aceptó la hipótesis nula de que las medias son iguales, corroborando lo que se vio en la figura 8 de que tampoco hay un cambio en el tiempo al que comienza a estabilizarse el valor resultado del algoritmo.

## 3. Conclusiones

Se pudo observar el beneficio de la paralelización del algoritmo genético en comparación a una implementación secuencial y se comparó también con una versión de selección por ruleta, se pudo observar el impacto en el tiempo de ejecución al agregar la selección y que para nuestros parámetros no haya mostrado una diferencia significativa, quedando por explorar los efectos de variar los parámetros del algoritmo para calibrarlos y alcanzar mejores soluciones.

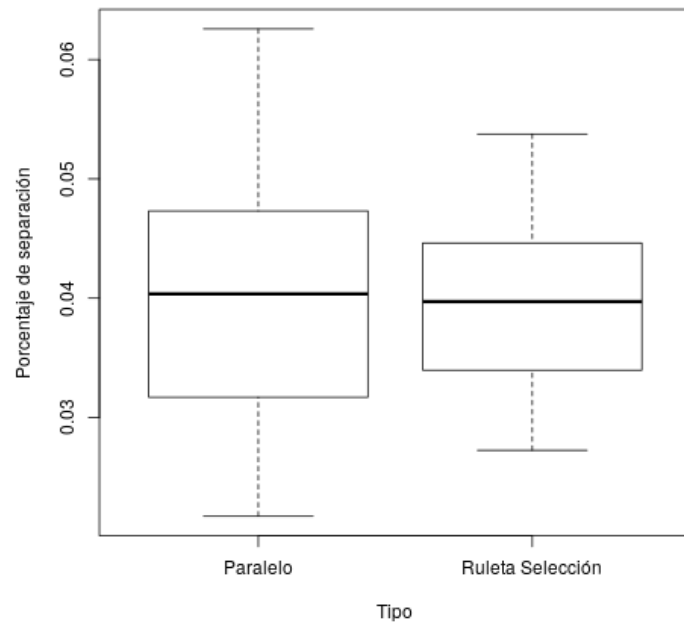


Figura 5: Comparación entre el porcentaje de separación y el tipo de algoritmo.

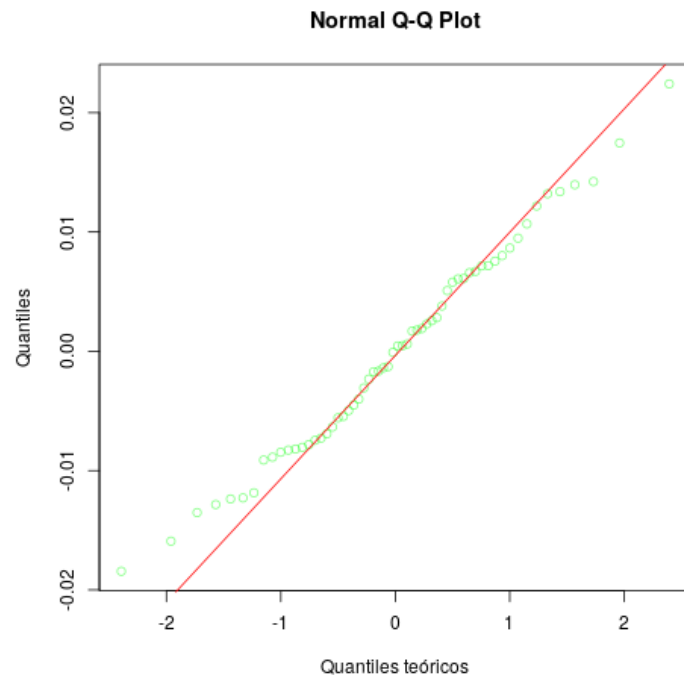


Figura 6: Comparación de los residuales del porcentaje de separación y el tipo de algoritmo.

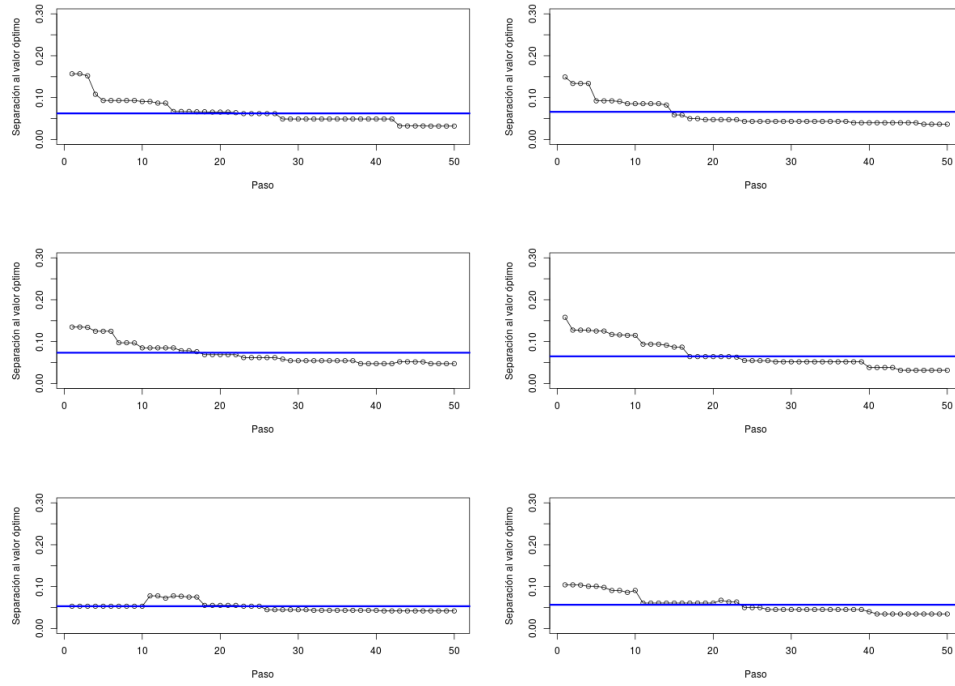


Figura 7: Líneas de seguimiento de la separación al valor óptimo, la línea azul señala los valores menores a un valor de desviación estándar del mejor valor obtenido las gráficas de la izquierda son los de tipo paralelo sin ruleta, del lado derecho son usando ruleta de selección.

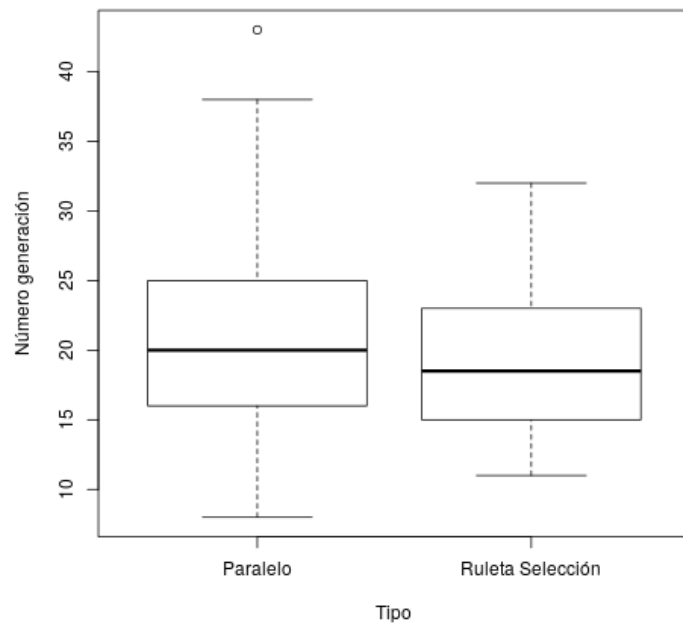


Figura 8: Comparación entre la generación de estabilización y el tipo de algoritmo.

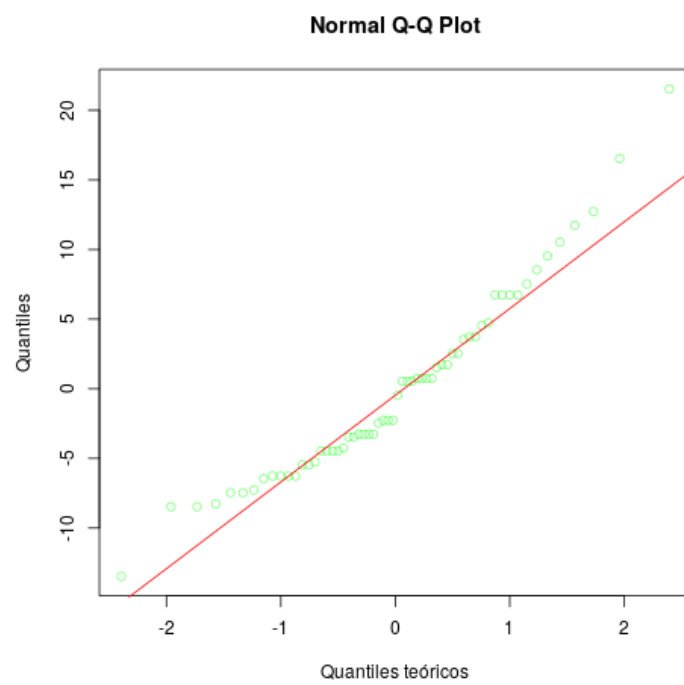


Figura 9: Comparación de los residuales la generación de estabilización y el tipo de algoritmo.