

Práctica 10

Jose David Martinez Ruiz

17 de octubre de 2017

1. Introducción

En esta práctica se trabajo con el algoritmo genético en un problema de mochila. El objetivo de esta práctica es, a partir de un código ya proporcionado, encontrar lugares en donde se pudiera implementar la paralelización, para de esta forma ahorrar tiempo de ejecución. Se compararon los tiempos obtenidos en la implementación en paralelo y la secuencial para así saber cual de estas implementaciones es la mas rápida.

Para el primer reto se trabajo con la versión paralelizada del problema y se busca encontrar mejores soluciones. para esto se utilizo la selección de la ruleta a la hora de seleccionar a los padres que se reproducirán para producir nuevas soluciones, para esto se calculo una probabilidad para cada solución que es directamente proporcional a el valor que le dan a la función objetivo, de manera que aquellas soluciones que den un resultado mejor, por así decirlo, con una mayor probabilidad serán seleccionados para que se reproduzcan.

En el segundo reto se hizo algo similar a lo que se hizo en el primer reto, pero ahora en lugar de utilizar la selección de ruleta en la fase de reproducción, se utilizó en la fase de selección de población para la siguiente implementación del algoritmo genético, ademas de eso se fijó el numero de elementos de la población que se van a "salvar", de manera que se algunas soluciones no factibles se tomaran en cuenta para la población del siguiente paso.

2. Desarrollo

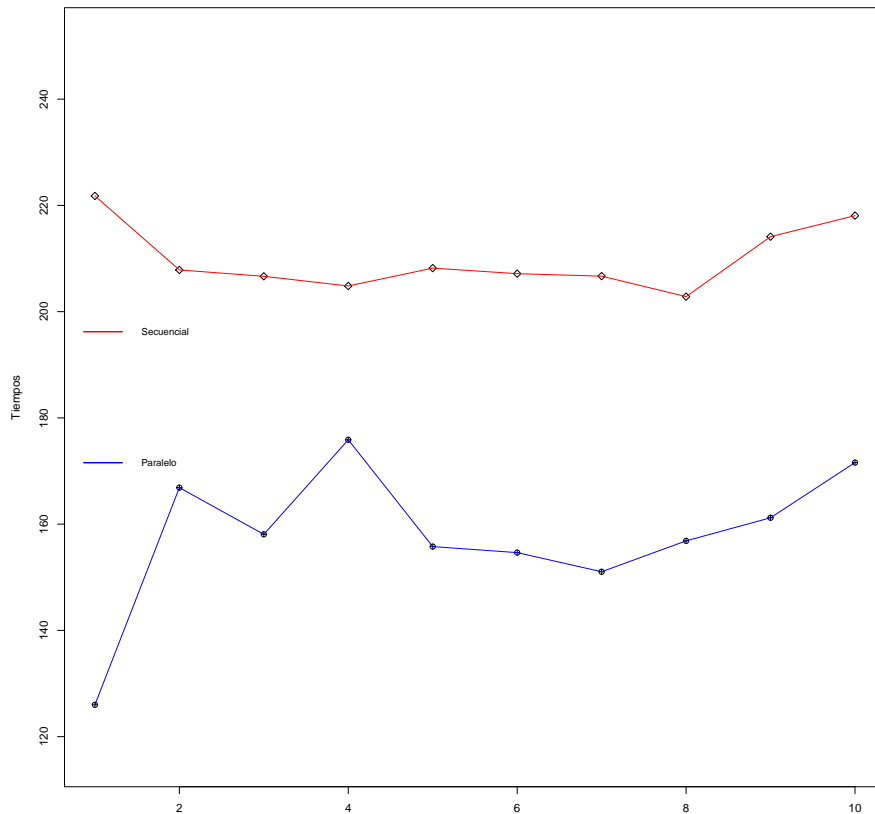
Primero se identifico del código los lugares del código donde se pudiera implementar la paralelización, los lugares que se encontraron fueron en la fase de mutación de las soluciones, la fase de reproducción de las soluciones y también el procedimiento de obtención de el valor de la función objetivo de cada solución y también la verificación de factibilidad de las soluciones obtenidas. Para realizar esto se crearon funciones nuevas a partir de las funciones ya existentes. La función de mutación toma una solución y modifica uno de sus componentes para así generar una solución distinta. La función de reproducción toma un par de soluciones y genera dos hijos a partir de ellos, el primero tiene la primera mitad del padre y la segunda mitad de la madre, y el otro hijo al revés, la primera mitad de la madre y la segunda mitad del padre, esta función regresa a ambos hijos como soluciones nuevas. La función que evalúa las soluciones obtenidas para obtener el valor de función objetivo, esta función toma una solución y calcula que tan beneficioso es llevar esos objetos en la mochila. Por ultimo la función de factibilidad, toma una solución, calcula el peso total de llevar esos objetos en la mochila y verifica si es posible llevar esos objetos, regresa un verdadero o un falso dependiendo si es factible o no. Ahora, se creó una funcion para cada una de las anterior mencionadas de manera que al paralelizar utilizando parSapply puedan funcionar de manera adecuada, se pudiera ejecutar en paralelo exitosamente.

El programa funciona de la siguiente manera, primero se generan los pesos de los objetos y también el valor de cada uno. La capacidad de la mochila esta dada por el sesenta y cinco por ciento de la suma del valor de todos los objetos. Se genera también una población inicial de doscientas soluciones, cada solución esta formada por cincuenta (el numero total de objetos que se pueden llevar) variables binarias, donde un uno representa que se lleva y un cero representa que no se lleva. Se fijó una probabilidad de mutación, con la cual cada elementos de la población tiene esa probabilidad de mutar, también se fijó un numero de reproducciones que se harán y también cuantas

veces se realizara el algoritmo genético. El algoritmo genético funciona de manera que a partir de una población inicial, ciertos miembros de la población se reproducirán y otros mas presentaran mutaciones (en esta práctica tanto los hijos como los mutados son soluciones a parte, es decir, no se modifican ni se eliminan las soluciones con las cual fueron creados), posterior a eso se ordenan los datos de manera que estén ordenados de factibles a no factibles y a la vez de mayor a menor. Una vez ordenados se salvaran como "nueva generación" los primeros elementos de la población tal que tengamos la misma cantidad de gente que cuando iniciamos. Todo esto se repetirá una cierta cantidad de veces de manera que así vaya obteniendo mejores soluciones hasta que llegue o se acerque lo más cercano al óptimo real del problema.

Las funciones que se mencionaron antes se paralelizaron debido a que como cada operación se realiza de manera independiente con las soluciones, es decir, un resultado futuro no depende de ningún resultado anterior, fue por lo cual se decidió paralelizar estas funciones.

Ahora, nos interesa saber si implementar la paralelización resulta en una mejoría del tiempo de ejecución, para esto se tomaron diez muestras del tiempo de ejercicio de la implementación secuencial y la implementación en paralelo, los resultados se muestran en la siguiente gráfica.



Como se puede observar, la implementación en paralelo es más rápida que la implementación secuencial, ahora vamos a corroborar si dicha diferencia es significativa, para eso se hizo una prueba de Wilcoxon para diferencia de medias. Al realizar la prueba, como era de esperarse, se obtuvo un p-valor muy pequeño, por lo que se concluye que si hay una diferencia significativa entre los tiempos de ambas implementaciones a favor de la implementación en paralelo.

3. Reto 1

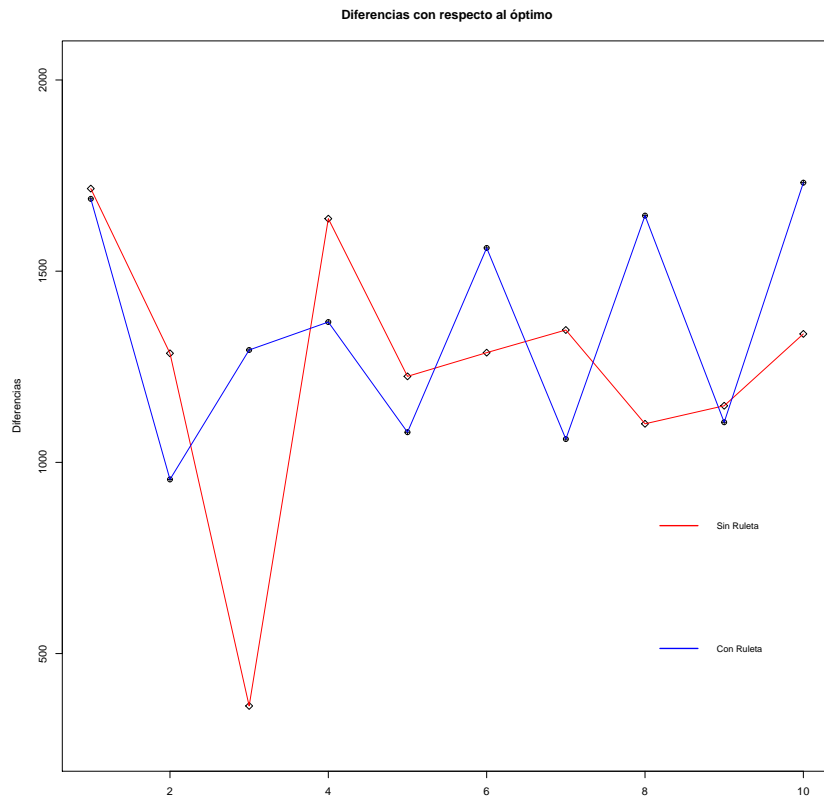
Para este primer reto se trabajó con la versión paralelizada anterior, y se modificó de manera que se seleccionaran con el método de la ruleta los padres que se van a reproducir, esto en función de su valor en la función objetivo.

Para hacer esto, la función reproducir que se paralelizó, se le agregó una variable extra de probabilidad que se agregó a la hora de seleccionar a los padres. Este valor de probabilidad está dado dependiendo de su valor en la función objetivo, justo después de que la población mutara, se calculó provisionalmente el valor de la función objetivo para cada miembro de la población en ese punto de la simulación. Una vez obtenidos estos valores, se modificaron de manera que estos valores estuvieran entre cero y uno. Esto último se hizo para poder parametrizar estos valores en un rango de probabilidad que va de 0.1 a 0.9. De este modo, la solución que de el valor de la función objetivo más pequeño tendrá una probabilidad del diez por ciento de ser seleccionado para reproducirse, mientras que la solución que de el valor de la función objetivo más grande tendrá una probabilidad del noventa por ciento de ser seleccionado para reproducirse.

La idea de hacer esto es para que a partir de los padres que generan mejores soluciones crear hijos que también den buenas soluciones o inclusive mejores, para así poder llegar al valor óptimo del problema con una mayor rapidez.

El resto del programa funciona de la misma manera que en la tarea base, ese fue el cambio que se hizo y en este caso nos interesa saber que tan buenas soluciones nos da hacer la selección de los padres de esta manera. Para esto se tomaron diez muestras de la diferencia que hubo entre el óptimo real del problema y el mejor valor que el algoritmo genético pudo encontrar. Estas muestras fueron tomadas para esta nueva implementación y la implementación en paralelo anterior para así saber cual de estas dos implementaciones da una mejor solución.

Los resultados obtenidos de verificar lo anterior se muestran en la siguiente gráfica.



Como se puede observar los resultados se ven muy variopintos y no se ve muy claro cual de estas implementaciones da un mejor resultado, por lo que se realizo una prueba estadística de Wilcoxon para poderlo verificar de mejor manera. Al realizar esta prueba nos arroja un p-valor muy superior al valor de referencia del cinco por ciento, por lo que no hay diferencia significativa entre ambas implementaciones.

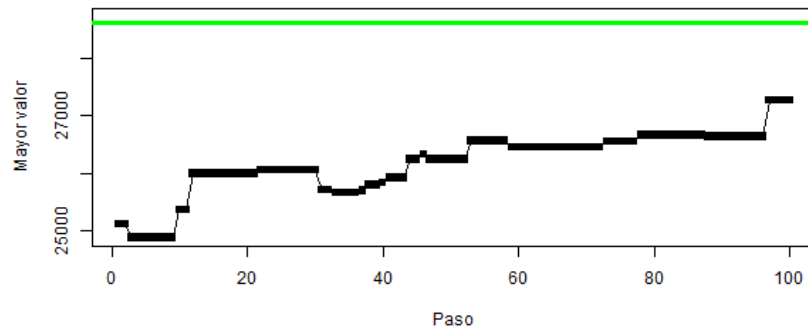
4. Reto 2

Del segundo reto, se hizo algo similar a lo que se hizo del primer reto, esto es la selección de la ruleta, sólo que esta vez en lugar de aplicarlo en la fase de reproducción, se utilizó en la selección de la población de la siguiente generación.

Justo después de haber calculado el valor de la función objetivo con cada una de las soluciones y también si dichas soluciones son factibles o no, se hizo algo similar al primer reto. De los valores obtenidos de la función objetivo para cada solución y a partir de ella se le asignó una probabilidad, sólo que en este caso, debido a la gran cantidad de soluciones que se obtienen que no son factibles, estas soluciones tuvieron una penalización a la hora de calcular su probabilidad, mientras que las soluciones factibles no se vieron afectadas a la hora de calcular sus probabilidades de manera similar a la forma en que se calcularon en el primer reto.

La idea de hacer esto es para diversificar las generaciones y así tener resultados que aunque a primera vista no nos sirvan, nos puedan ayudar a obtener valores que nos acerquen más rápido al óptimo del problema.

Se tomó una población bastante grande para poder aplicar el método de una mejor manera. Se tomaron en cuenta muchas posibilidades pero en general el comportamiento del algoritmo genético con el método de la ruleta en la selección de la generación, da resultados muy similares a lo siguiente.



5. Conclusiones

De la tarea base se concluye que, en efecto, la paralelización ayuda bastante para ahorrar tiempos de ejecución, ya que hubo una diferencia significativa entre los tiempos de ambas implementaciones, por lo que en este caso que requiere de una gran cantidad de operaciones, utilizar la paralelización es la mejor opción.

Del primer reto se buscó mejorar los resultados obtenidos de la tarea base al implementar el método de la ruleta a la hora de la reproducción. Los resultados obtenidos mostraron que en la implementación de la tarea base y del método de la ruleta no hay diferencia significativa, esto puede deberse a que los hijos generados por los padres "élite" no son factibles y aunque den una mejor solución, no es posible tomarlos en cuenta como una solución real.

Del segundo reto la implementación del método de la ruleta para la selección de la siguiente

generación, de manera similar al primer reto no arrojó mejores valores con respecto a la implementación paralela original, al igual que en el primer reto seguramente por la demasiada diversificación y sobretodo por el exceso de valores no factibles que dan valores de la función objetivo muy buenos que no da oportunidad de soluciones factibles de ir abarcando el camino correcto. Por lo que aunque la diversificación ayuda, demasiada no ayuda a encontrar mejores valores.