

Simulación de Sistemas

Práctica 10

Algoritmo Genético
1455175: Ángel Moreno

16 de octubre de 2018

1. Introducción

Se considera el problema de la mochila de programación entera, el cual consiste en que dado un conjunto N de objetos cada uno con un beneficio y peso, y una mochila con una capacidad, se desea seleccionar los objetos que maximizen el beneficio sujeto a no exceder la capacidad de la mochila.

Se implementa un algoritmo genético el cual sirven en casos donde no existe ningún algoritmo exacto eficiente. Un algoritmo genético representa posibles soluciones a un problema en términos de un genoma o cromosoma, estos algoritmos imitan la evolución humana con una población inicial (en nuestro caso soluciones aleatorias) con dos mecanismos (1) **reproducción** el cual su finalidad principal es crear descendencia con características o aptitudes de dos padres, en este caso de dos soluciones; (2) la **mutación** que su función es alterar un genoma o solución para diversificar las soluciones.

1.1. Tarea

Paraleliza el algoritmo genético y estudia los efectos en su tiempo de ejecución con pruebas estadísticas y visualizaciones, variando el número de objetos en la instancia. Determina con qué tamaño de instancia el algoritmo genético es mejor que el algoritmo exacto en términos de valor total obtenido por segundo de ejecución.

2. Simulación

Se ejecuto simulaciones distintas con diferentes parámetros los cuales se especifican en cada uno de los resultados.

2.1. Resultados

La primera simulación para revisar la eficiencia del genético en paralelo contra el secuencial se utiliza de parámetros una generación de 200, probabilidad de mutación de un 5 %, con 50 reproducciones y un máximo de iteraciones de 50 en el algoritmo. Se varia la cantidad de objetos en una cantidad exponencial con exponentes de 4 a 7 de la base e redondeado. La figura 1 muestra los resultados.

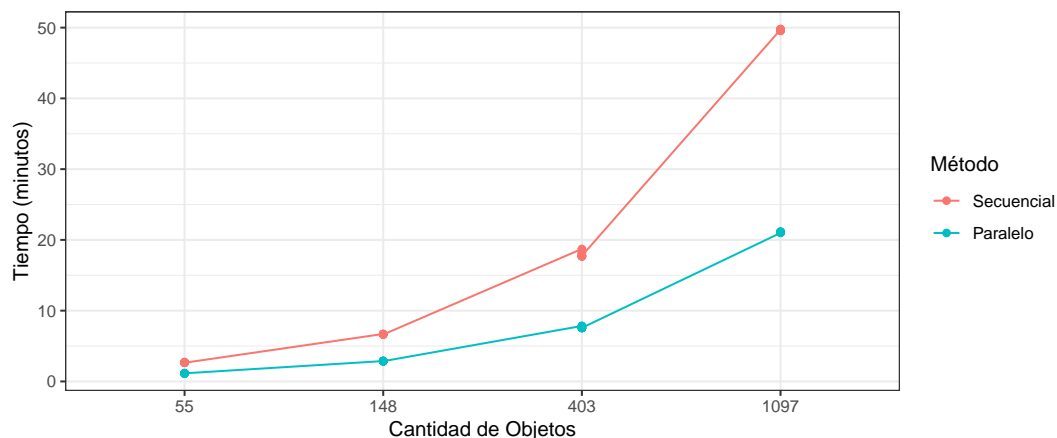


Figura 1: Resultados de los tiempos contra cantidad de objetos de 30 réplicas.

se observa la clara ventaja del genético en paralelo en eficiencia de tiempo de ejecución.

El algoritmo genético tiene muchos parámetros el cual puede hacer hacer el eficiente el algoritmo, se simula el algoritmo genético con distintos parámetros, tales como la cantidad de reproducción (50, 75, 150), probabilidad de mutación(0.05, 0.10, 0.20), cantidad de generaciones(200, 300, 500) y se fijo la cantidad de iteraciones en 50 y en una instancia con 100 objetos. La figura 2 muestra los resultados.

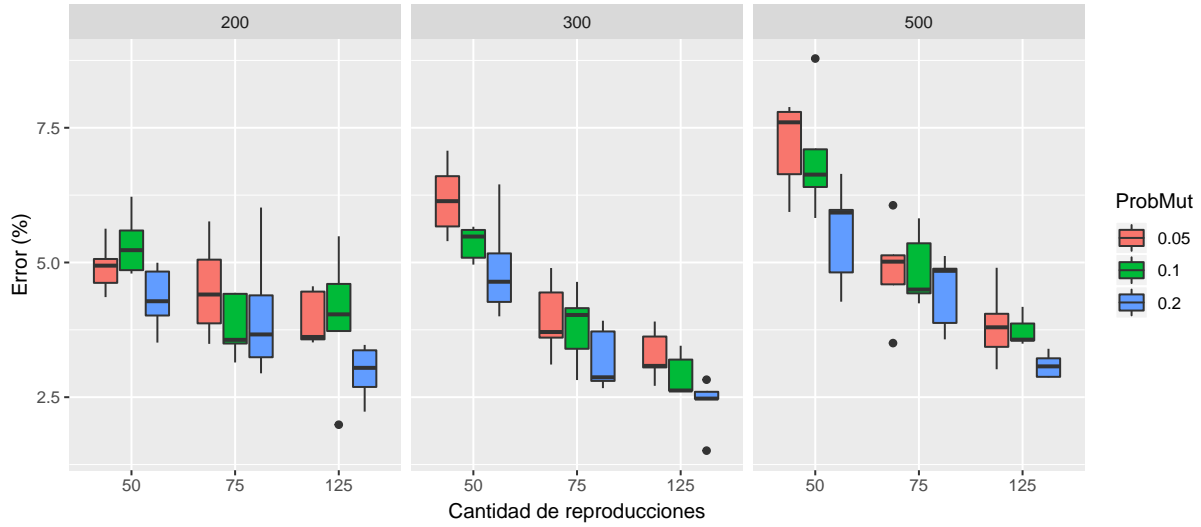


Figura 2: Resultados de los errores contra cantidad de reproducciones.

Esto se hace para tratar de calibrar el algoritmo. El error es calculado con respecto al óptimo del algoritmo exacto (que tan alejado está).

2.2. Reto 1 y 2

El primer reto es cambiar la selección de padres para reproducción a que use selección de ruleta: cada solución se selecciona como padre con una probabilidad que es proporcional a su valor de función objetivo y a su factibilidad, combinando los dos a alguna función que parezca conveniente. Estudia si este cambio produce una mejora estadísticamente significativa en la calidad de la solución, incluyendo visualizaciones informativas.

El segundo reto es extender la selección de ruleta a la fase de supervivencia: en vez de quedarnos con las mejores soluciones, cada solución tiene una probabilidad de entrar a la siguiente generación que es proporcional a su valor de la función objetivo, incorporando el sí o no es factible la solución en cuestión, permitiendo que los k mejores entre las factibles entren siempre (donde k es un parámetro). Estudia nuevamente el efecto de este cambio en la calidad de la solución.

Se utiliza un diseño factorial de experimentos 2^2 donde se varia el método de selección (aleatorio, ruleta), generaciones (mejores, supervivencia), variando la cantidad de objetos en (55, 148). Se utiliza de parámetros una generación de 200, probabilidad de mutación de un 5 %, con 50 reproducciones y un máximo de iteraciones de 50. La figura 3 muestra los resultados.

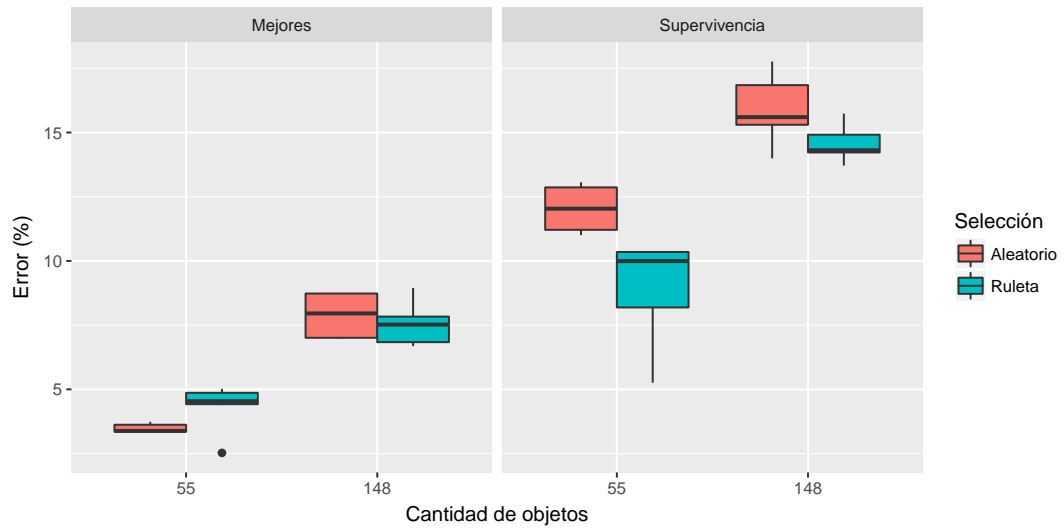


Figura 3: Resultados de error contra cantidad de objetos con 5 réplicas.

Se observa que con mini-instancias la combinación de aleatorio con la generación de los mejores es mas eficiente, pero conforme aumenta la cantidad de objetos se ve que la combinación de la ruleta con cualquier generación es mas eficiente.

Referencias

- [1] SCHAEFFER E. *R paralelo: simulación y análisis de datos*, 2018.
<https://elisa.dyndns-web.com/teaching/comp/par/>
- [2] VALDES E. *Repository of Github*, 2017.
<https://github.com/eduardovaldesga/SimulacionSistemas>
- [3] SAUS L. *Repository of Github*, 2018.
<https://github.com/pejli/simulacion>