

Resolución del problema de recolección de residuos utilizando Algoritmos Genéticos

Santiago Castro y Matías Mansilla
Grupo 15

Resumen—El problema de la recolección de residuos es común y muy conocido en muchas ciudades del mundo: se tienen los contenedores ya dispuestos en la ciudad, se conoce su demanda y lo que se desea es encontrar la forma de recolectar su basura minimizando el consumo de recursos. Al igual que el Problema del Viajante (TSP), es un problema de la clase NP-duro, por lo tanto se desconoce una forma eficiente de hallar el óptimo para instancias de tamaño grande. En el presente artículo se presenta una implementación basada en Algoritmos Genéticos que no asegura devolver el óptimo, pero sí soluciones de buena calidad. Se trabajan con datos reales y públicos de la ciudad de Montevideo, Uruguay. Los resultados experimentales muestran que el algoritmo propuesto tiene mejores resultados que una variación de una heurística común utilizada para TSP.

Palabras clave—algoritmos evolutivos, recolección de residuos, recolección de contenedores.

I. INTRODUCCIÓN

EN los últimos años el manejo de los residuos a nivel mundial ha significado uno de los principales problemas; el crecimiento demográfico y los avances en la industria provocan que la cantidad de residuos sea cada vez mayor. Es por esto que las grandes ciudades han tenido que tomar medidas para combatir dicho problema.

En Uruguay, a fines del año 2003 fue introducido por la Intendencia de Montevideo [9] el sistema de recolección de basura mediante contenedores. Algunos de los beneficios son: evitar que la basura sea dejada en la puerta de cada hogar produciendo entre otras cosas mal olor y basura desperdigada por la ciudad, permitir a los habitantes sacar la basura y dejarla en el contenedor en el horario que deseen, entre otras.

La recolección de los contenedores está a cargo de camiones recolectores-compactadores automatizados de gran eficiencia, los cuales vacían cada contenedor en menos de un minuto. Además un segundo operario verifica previo al vaciado el estado de los residuos y mantiene la limpieza entorno al contenedor.

En el presente trabajo se utiliza entonces a Montevideo como ciudad para obtener los datos y evaluar la solución propuesta al problema presentado.

Este algoritmo evolutivo optimiza el recorrido de los camiones recolectores. Es decir, intenta minimizar la cantidad de tiempo que un camión se encuentra en la calle, con la restricción de satisfacer la demanda de todos los contenedores de una zona y no pasarse de la capacidad del camión para alojar basura.

Esto, entre otras cosas, permite un ahorro sustantivo de combustible y disminuye el tráfico de los camiones en la

ciudad, los cuales a pesar de funcionar en general en horario nocturno, dificultan el tráfico del resto de los vehículos, entre otros motivos.

En la Figura 1 se aprecia un ejemplo de distribución de contenedores para una parte del barrio La Comercial, Montevideo.

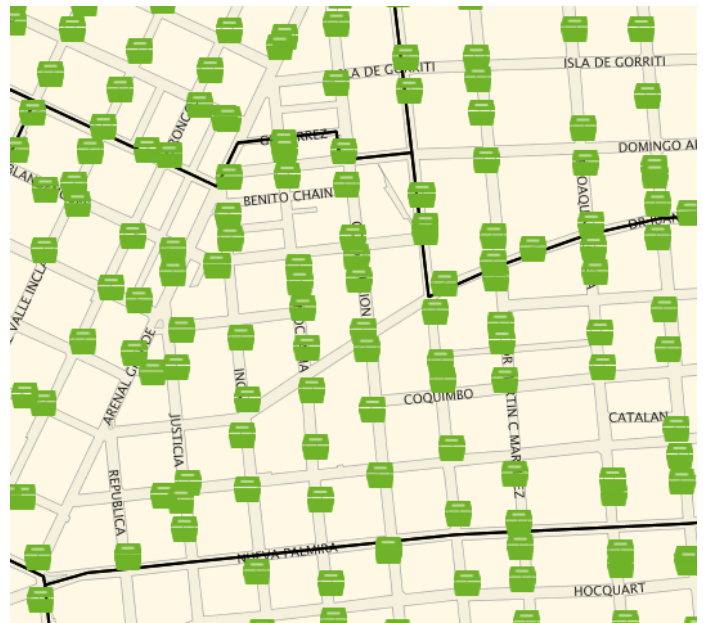


Figura 1. Imagen de la distribución de los contenedores en La Comercial, Montevideo [7]. Las líneas negras delimitan recorridos distintos.

II. JUSTIFICACIÓN DE USAR ALGORITMOS EVOLUTIVOS

Si se encontrara un algoritmo exacto (es decir que encuentra el óptimo) de tiempo de ejecución polinomial según la cantidad total de contenedores para este problema, también se lo encontraría para TSP (Travelling Salesman Problem), ya que es un caso particular de éste. Por ejemplo, si se toma que el tiempo en que se demora en llenarse cada contenedor es de 24 horas y el camión tiene capacidad infinita, cada día deberían recorrerse todos los contenedores y por tanto se convierte cada recorrido en un problema TSP, y se estaría teniendo por ende un algoritmo exacto de complejidad polinomial en el tiempo para TSP, cuando en realidad no se conoce ninguno aún, si es que existe. Por lo tanto se tiene un problema perteneciente a la clase NP-duro, ya que TSP está en dicha clase y es reducible a este problema.

Es por eso que elegimos este problema a ser resuelto con algoritmos evolutivos como metaheurística, ya que se pueden obtener soluciones de buena calidad con un uso eficiente del recurso tiempo.

III. RELEVAMIENTO DE DATOS

Tengamos en cuenta que los datos que se necesitan para poder realizar este trabajo son: la demanda de los contenedores (cada cuántos días se llenan), su capacidad, la capacidad del camión recolector-compactador, el tiempo en que se tarda en recolectar cada contenedor y la matriz de tiempos en que se demora de ir desde un punto a otro en la ciudad.

La primer tarea realizada fue encontrar el *Mapa de contenedores de residuos domiciliarios* [7], y a partir de ahí, llegar a la página *GeoServer* de la Intendencia de Montevideo [6] y encontrar un archivo con la información de todos los contenedores de la ciudad. Éste contiene datos como su ubicación exacta, su capacidad, el Centro Comunal Zonal al que pertenece, su código de recorrido, etc.

Para poder conocer la demanda de los contenedores se utiliza la información censal del año 2004 de la Instituto Nacional de Estadística (INE) [4] para obtener la información sobre la población por cada manzana de Montevideo. Se la trata con el software QGIS de QGIS Development Team [13] y con el complemento *MMQGIS* de Minn [11] para poder relacionar cada polígono del archivo de información geográfica con su información relacionada. Una vez hecho esto, se tiene cada polígono representado como una lista ordenada de vértices. Se quiere conocer las manzanas (representadas por polígonos) más cercanas a un contenedor dado, por lo tanto es de interés saber para cada polígono donde está su centro, y a partir de él poder hallar la distancia al contenedor.

Se escribió un archivo de comandos en el *Lenguaje de Programación Ruby* [10] para poder obtener los centros de cada polígono. Además se utilizó la página web de MyGeo-data Portal [12] para obtener las coordenadas en el sistema convencional (WGS 84, con SRID 4326) a partir del sistema en el que estaban los datos, EPSG:32721, WGS 84, UTM zone 21S (WGS 84, UTM zone 21S, con SRID 32721). Una vez hecho esto, se procedió a utilizar otro archivo de comandos para obtener la demanda de cada contenedor calculando sus cuatro manzanas más cercanas y asignándoles a cada una de ellas el inverso del cuadrado de la distancia al mismo como peso. A partir de una normalización, se le asignó a cada contenedor una parte de la población de las manzanas. Es interesante observar los datos de cuántas personas tienen asignados los contenedores. En la Figura 2 se puede ver el histograma.

Siguiendo lo sugerido por la asociación civil uruguaya CEMPRE (Compromiso Empresarial Para el Reciclaje) [2] que promueve la reducción de residuos y el reciclaje, una persona residente de un país en vías de desarrollo produce, promedialmente, entre 400 y 700 gramos de basura diaria. Teniendo en cuenta que Montevideo es la parte más metropolitana de Uruguay, se tomó el mayor valor de este intervalo como referencia. Utilizando este dato y además el dato de densidad de masa de la basura antes de ser compactada

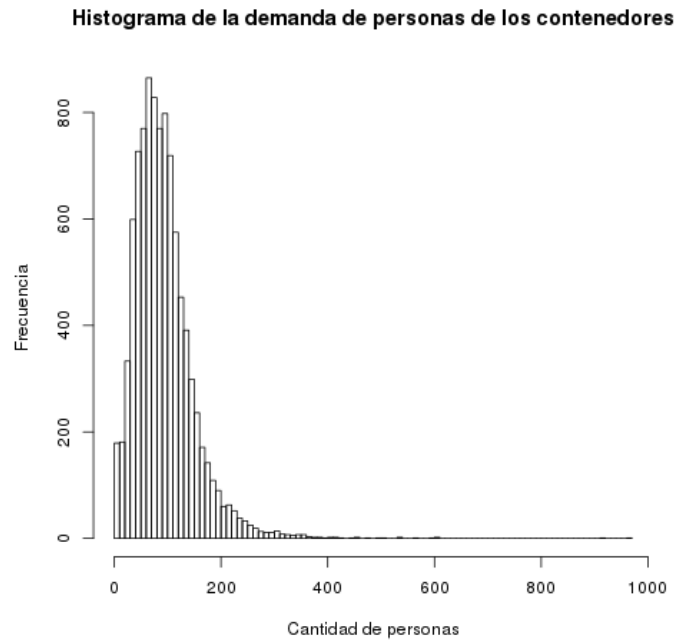


Figura 2. Histograma (en percentiles) de las personas asignadas a los contenedores. La distribución está muy concentrada, siendo el promedio 92,75 y la desviación estándar 56,62.

sugerida por Sakurai [14], que es de 200 kg/m^3 , se pudo obtener la información sobre cuánto tarda en llenarse cada contenedor.

A partir de la *API de rutas de Google* [3] se obtuvo cuánto se tarda en ir desde y hasta cada par de contenedores de las zonas de interés. También se obtuvo cuánto se tarda en ir desde el origen de los camiones, que se encuentra en la Avenida General Rivera 3881, y a su vez cuánto se tarda en ir desde cada contenedor hasta el lugar de destino, las Usinas de Disposición Final de Camino Felipe Cardozo 3220 esquina Cepeda. Cabe destacar que el servicio tiene en cuenta las flechadas de las calles, la información de tránsito, entre otra información.

Conociendo además que un camión tarda aproximadamente 45 segundos en procesar un contenedor, y que éste tiene 108.000 litros de capacidad para basura compactada (la compacta antes de guardarla), se pudieron relevar todos los datos necesarios para trabajar con el problema.

De esta manera, teniendo en cuenta la información de los *Municipios de Montevideo* [8] y los *Centros Comunales Zonales* [5] relacionados a los contenedores, se relevaron datos de los barrios Cerro, La Blanqueada y Pocitos, que tienen densidades de población baja, media y alta, respectivamente.

IV. ESTRATEGIA DE RESOLUCIÓN

Como se ha mencionado se utilizan Algoritmos Evolutivos, en particular Algoritmos Genéticos.

IV-A. Modelo matemático

El problema refiere a intentar minimizar el costo, que en este caso es el consumo de combustible por parte del camión.

Como se tiene la suposición de que es siempre el mismo mientras el camión está en marcha, a modo de simplificación se tiene en cuenta el tiempo en que tarda en hacer todos el recorridos como función objetivo, ya que minimizar esto es equivalente a minimizar el consumo de combustible. También se puede suponer que el tiempo de carga de un contenedor es constante, de parámetro T . Entonces los tiempos serán los de traslado entre contenedores, que se tomará en función de la distancia recorrida entre los ellos y la velocidad promedio, y los de cargarlos.

Se tiene que la municipalidad está dispuesta a realizar hasta un recorrido por día [7], en horario nocturno. Se quiere planificar el recorrido para una semana modelo. Basta con determinar cuáles contenedores serán recorridos en cada día, y en qué orden. Hay que recordar que cada contenedor tiene una capacidad máxima de almacenamiento, la cual se satisface según la demanda en su entorno. Cuánto tarda en llenarse depende de la población de sus manzanas adyacentes. Existe entonces la restricción de que no debe dejarse que los contenedores se llenen por completo, ya que esto implicaría que luego la gente no tenga dónde dejar su basura, y hasta inclusive a veces la dejan al costado de los contenedores sobre la vereda o la calle, ensuciando la ciudad. También se tiene la restricción de que los camiones tienen una capacidad máxima de carga, que en un recorrido no pueden sobrepasarla.

Se sabe que los camiones salen de un depósito dado y al finalizar el recorrido se dirigirán a un vertedero determinado. Además, cabe destacar que en una solución candidata, en un día determinado si no hay por lo menos un contenedor a recoger ni vale la pena hacer el recorrido ese día y por lo tanto no se paga el costo fijo de salir a recorrer.

Se tiene entonces que la función objetivo es la siguiente:

$$\min \sum_{d=1}^7 \mathbb{1}_{N_d > 0} \times (N_d T + \sum_{i=0}^{N_d} \text{dist_tiempo}(x_{d,i}, x_{d,i+1}))$$

Esta función expresa el tiempo del recorrido de cada día, si es que se hace, siendo $x_{d,i}$ el número de contenedor (están numerados de 1 a K) visitado el día d en el lugar (en el orden) i , N_d la cantidad de contenedores visitados el día d y T el tiempo en recoger la basura de un contenedor. $x_{d,0}$ y x_{d,N_d+1} representan el lugar de origen y de destino respectivamente, para todo día d .

Las restricciones se pueden expresar de la siguiente forma, siendo $y_{d,c}$ la variable que indica si el contenedor c es recogido el día d :

$$\begin{aligned} \forall c \in \{1, \dots, K\}, d_0 \in \{1, \dots, 7\} \\ \rightarrow \exists d \in \{d_0, \dots, d_0 + \lfloor D_c \rfloor - 1\} / y_{d,c} = 1 \end{aligned}$$

Es decir que para todo contenedor y para todo intervalo de días que tarda justo antes de llenarse (según lo que demora en llenarse, D_c), debe haber al menos uno de los días en el intervalo cuyo recorrido pase por él. Se tiene en cuenta que los días son circulares respecto a las semanas, y entonces que el final de la semana se “engancha” con el principio de la misma. Considerar que la semana modelada es una semana

representativa, y entonces que el estado de los contenedores la primera vez que se pone en práctica una solución no se tiene en cuenta, sino que se trata al problema como si fuera una semana más del año.

También se tiene que tener en cuenta la capacidad del camión:

$$\forall d \in \{1, \dots, 7\}, \sum_{c=1}^K y_{d,c} \times m_c \times \frac{d - \text{ant}(d, c)}{D_c} \leq M$$

Esto es que la suma, para cada contenedor c , de la capacidad m_c del contenedor, multiplicado por qué tan lleno estaba al recogerlo, sea menor o igual a la capacidad del camión M . Para saber qué tan lleno estaba, se mira la diferencia de días con el día de la última recogida del contenedor y se lo divide entre lo que se tarda en llenar. Éste se define así:

$$\text{ant}(d, c) = \max_a a < d / y_{a,c} = 1$$

Cabe aclarar que lo que se demora en desplazarse hasta un contenedor una vez que se empieza el recorrido no es tenido en cuenta en el tiempo de su demanda, es decir que si por ejemplo un contenedor se llena en 24 horas y ese tiempo termina al comenzar el recorrido, en nuestro caso se estaría cumpliendo con la demanda igualmente. Este tiempo es despreciable teniendo en cuenta lo que se espera que demoren los recorridos. Si se mira el peor caso, que sería ir un día primero a un contenedor particular que demora 24 horas en llenarse, y al siguiente día visitarlo en lugar último, la diferencia sería de unas pocas horas, además agregando que a las horas que se recorren los contenedores hay muy poca actividad de arrojar residuos en ellos.

Una solución candidata es entonces un conjunto de recorridos, uno por cada día de la semana.

IV-B. Representación

La representación es de largo variable. Es un vector de 7 elementos, uno por cada día de la semana, en donde cada uno adentro tiene una lista de contenedores, que indica el orden que deben ser recorridos. Dentro de la lista de contenedores a recorrer de un día particular no debe haber repetidos. Un individuo a su vez debe cumplir las restricciones anteriormente planteadas.

La decodificación es parecida a la usada en la representación con permutaciones del problema TSP, se calcula de antemano los caminos y sus distancias entre todo par de puntos a visitar, y entonces se puede armar fácilmente el camino del recorrido una vez que se sabe qué contenedores son visitados y en qué orden.

IV-C. Inicialización de la población

Para inicializar a la población se procede de la siguiente manera: mientras haya contenedores cuya demanda no es satisfecha (se llenan antes de que pase un camión), se toma uno de ellos y se elige un día al azar para pasar a recogerlo. Luego de esta etapa, se procede a elegir un orden aleatorio de recorrida, para cada día. La idea es tratar de ser lo

más exploratorio posible, para no sesgarse en el espacio de soluciones.

IV-D. Función de fitness

La función de fitness (o de costo) es entonces la siguiente:

$$f(x) = \sum_{d=1}^7 \mathbb{1}_{N_d > 0} \times (N_d T + \sum_{i=0}^{N_d} \text{dist_tiempo}(x_{d,i}, x_{d,i+1}))$$

Pasando a maximización queda:

$$f(x) = - \sum_{d=1}^7 \mathbb{1}_{N_d > 0} \times (N_d T + \sum_{i=0}^{N_d} \text{dist_tiempo}(x_{d,i}, x_{d,i+1}))$$

En donde se tienen en cuenta por lo tanto los tiempos de los caminos entre un contenedor y el siguiente en el fenotipo.

IV-E. Selección

Como método de selección se utiliza la selección proporcional ya que se desea que sea proporcional al fitness. Este método selecciona a un individuo con la probabilidad de ser elegido según su fitness, comparado con el fitness de toda la población.

IV-F. Operador de cruzamiento

Se procede de forma parecida a SPX (Single Point Crossover). Se toma al azar un día de corte y luego para cada padre un lugar en ese día de corte, pudiendo caer el punto de corte en cualquier punto de un recorrido, inclusive donde termina un día y comienza otro.

Luego de aplicar esto se hacen correcciones para satisfacer las restricciones. Si llegan a quedar, dentro del día de punto de corte, contenedores a visitar repetidos, se sacarán los repetidos, eligiendo al azar uno de cada par de ellos. Si se deja de satisfacer la demanda de llenado de un contenedor, se toma un día de los intervalos para los cuales no se cumple y se lo agrega al recorrido, en algún lugar al azar, de forma de mantener diversidad de las soluciones. Luego, si se está superando la capacidad de carga del camión en alguno de los días, se le quitarán contenedores al azar de su recorrido hasta que se cumpla la restricción, pero eligiendo aquellos contenedores que no comprometan la demanda de llenado. De no ser posible no quedará otra opción más que descartar al individuo.

Buenas características de los individuos pueden ser mantenidas de esta manera, ya que recorridos de días enteros se mantendrán entre soluciones, y para el día en donde se corta se mantienen partes del recorrido.

IV-G. Operador de mutación

Primero, se toma al individuo como una lista sola, de forma parecida al operador de cruzamiento. Se recorre cada posición y con cierta probabilidad se puede o bien agregar un contenedor a un recorrido (que no esté presente ya), o quitar uno de él, o intercambiarlo con otro al azar (posiblemente en otro recorrido). Si esto hace que no se verifiquen restricciones,

va a tener que descartarse. De esta manera, se logra introducir variedad a los individuos, explorando el espacio de soluciones.

Abdoun, Tajani y Abouchabaka [1] abarcan el estudio de un operador especial de cruzamiento, HPRM (Hybridizing PSM and RSM Operator) que es un híbrido de PSM (Partial Shuffle Mutation) y RSM (Reverse Sequence Mutation), el cual da mejores resultados para problemas de tipo TSP.

PSM propone para cada posición de la representación, bajo la probabilidad de mutación, intercambiar su valor con otro del vector, al azar. RSM, en cambio, procede eligiendo dos posiciones y luego invirtiendo la sublista determinada por ellas. HPRM lo que establece es entonces escoger dos posiciones, y para cada posición entre ellas ir permutándolas, y bajo la probabilidad de mutación, intercambiar la segunda de ellas con una posición al azar del vector.

Nuestro problema tiene mucho parecido con TSP en cuanto a tratar de conocer en qué orden visitar los puntos, lo cual implica que seguimos la sugerencia de los autores. Se decidió entonces que, con probabilidad 50 % se aplique el primer operador de mutación mencionado, o se aplique el algoritmo sugerido para cada día particular.

Por motivos de completitud, se le asigna la mitad de la probabilidad de mutación a la primera parte mencionada, y la otra mitad a la segunda parte.

IV-H. Reemplazo

Los hijos compiten con los padres, quedando aquellos de mejor fitness.

IV-I. Criterio de parada

Como criterio de parada se eligió 10.000 generaciones.

IV-J. Ambiente de desarrollo

Se utilizó C++ como lenguaje de programación y la biblioteca *Malva* [15] por su extensibilidad y cantidad de características que soporta.

V. EVALUACIÓN EXPERIMENTAL

V-A. Instancias del problema

Se recabaron datos para armar seis instancias del problema, dos para cada uno de los barrios Cerro (que está en el municipio A), La Blanqueada (del municipio C) y Pocitos (del municipio CH), como se mencionó anteriormente, de densidades de población baja, media y alta, respectivamente. De cada par, una es una instancia de tamaño pequeño (la segunda) y otra de tamaño real (la primera).

V-B. Plataforma de ejecución

El análisis experimental se llevó a cabo con las siguientes especificaciones de computadora: procesador Intel Core i3 3,07Ghz, 4 GB de RAM y sistema operativo Fedora 19.

V-C. Ajuste de parámetros

Se tomó una de las tres instancias de tamaño pequeño, la de densidad media de población, para determinar qué combinación de los valores de los parámetros tamaño de la población, probabilidad de cruzamiento y probabilidad de mutación consigue los mejores valores. Se consideraron como valores candidatos tamaños de población de 50, 100 y 150, probabilidades de cruzamiento de 60 %, 70 %, 80 %, 90 % y 95 % y probabilidades de mutación de 0,1 %, 1 % y 10 %.

Se realizaron cuarenta ejecuciones independientes de cada una de las tres instancias. Consideramos que esta cantidad es suficiente para asegurar reducir la varianza de los resultados. En la Tabla II se puede ver los valores obtenidos para las diferentes configuraciones. La Figura 3, la Figura 4 y la Figura 5 muestran cómo es el fitness promedio para valores de población 50, 100 y 150 respectivamente. Recordemos que el problema es de minimización al mirar las gráficas.

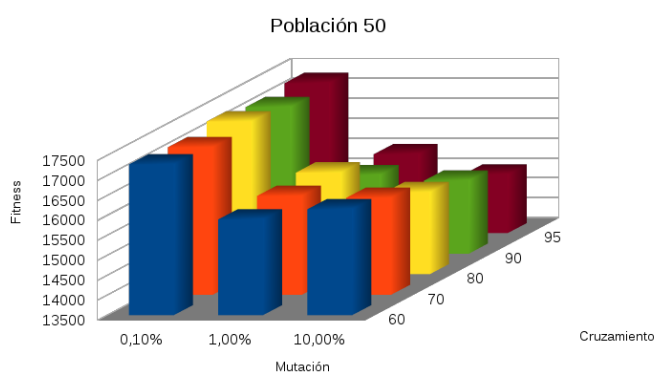


Figura 3. Promedio de fitness para cada instancia de configuración con población 50.

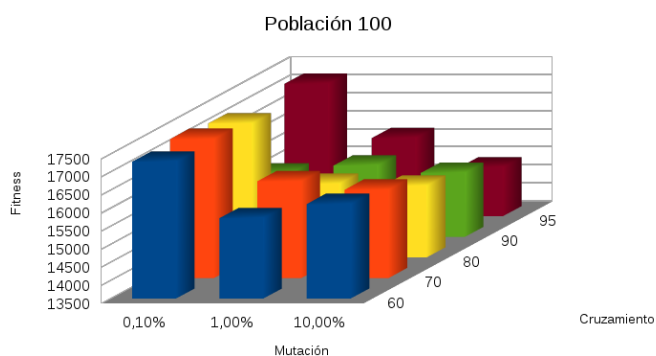


Figura 4. Promedio de fitness para cada instancia de configuración con población 100.

Los mejores resultados fueron obtenidos con población 100, probabilidad de cruzamiento 95 % y probabilidad de mutación 10 %.

Se puede observar que el tamaño de la población no hace variar significativamente los resultados. Se obtienen mejores resultados cuando se aumenta la probabilidad de mutación, pero también acompañada con la de cruzamiento, como se puede observar en las gráficas.

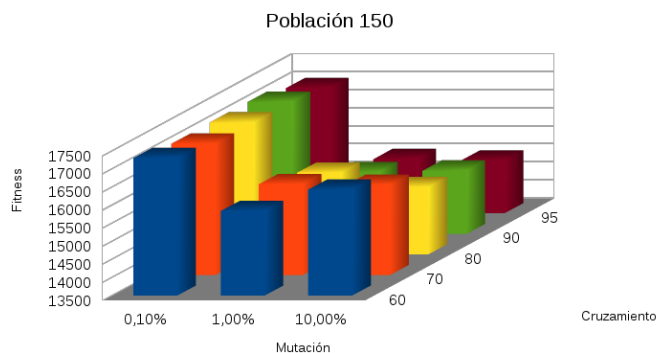


Figura 5. Promedio de fitness para cada instancia de configuración con población 150.

V-D. Evaluación

Se realizaron cuarenta ejecuciones independientes de las otras cinco instancias del problema, es decir, todas menos la usada para ajustes paramétricos.

A continuación y a modo de ejemplo se muestra la mejor solución encontrada para la instancia A01, una zona del barrio Cerro. Los números identifican a los distintos contenedores.

Lunes : 68 → 83 → 51 → 91 → 61 → 74 → 15 → 69
→ 17 → 65 → 29 → 8 → 1 → 58 → 14 → 63
→ 32 → 82 → 26

Martes : 57 → 12 → 21 → 71 → 60 → 47 → 82 → 85
→ 13 → 38 → 53 → 30 → 27 → 55 → 39
→ 24 → 6 → 43 → 4 → 44 → 88 → 42
→ 3 → 56 → 77 → 41 → 16 → 65 → 75
→ 78 → 1 → 59 → 45 → 5 → 11 → 69
→ 66 → 86 → 70 → 92 → 31 → 18

Miércoles :

Jueves : 76 → 67 → 23 → 27 → 68 → 49 → 34
→ 50 → 82 → 79 → 25 → 89 → 15 → 37
→ 81 → 0 → 80 → 46 → 64 → 29 → 8
→ 88 → 87 → 7 → 90 → 84 → 52 → 10
→ 22 → 71 → 47 → 26 → 28

Viernes : 20 → 17 → 65 → 75 → 9 → 14 → 40
→ 59 → 5 → 1 → 16 → 77 → 15 → 56
→ 13 → 55 → 49 → 36 → 3 → 85 → 89
→ 25 → 35 → 33 → 32 → 73 → 38 → 72
→ 54 → 62 → 44 → 24 → 74 → 69 → 11
→ 41 → 67 → 93 → 57 → 12 → 53 → 18
→ 19 → 22 → 48 → 60

Sábado : 71 → 30 → 23 → 66 → 82 → 2 → 60

Domingo :

Se observa que para este ejemplo el algoritmo encontró que no salir a recorrer para los días miércoles y domingos

representa un costo fijo que no se tuvo que realizar, obteniendo una mejora significativa al poder acomodar los contenedores en los demás días.

De manera ilustrativa se muestra en la Figura 6 el recorrido del día sábado para esta solución, y en la Figura 7 el recorrido del día jueves para la segunda instancia del barrio Pocitos. Como se puede observar, el orden del recorrido es coherente. Este problema, al igual que TSP, debe resolver el orden a recorrer, pero además elegir qué día visitar los contenedores.

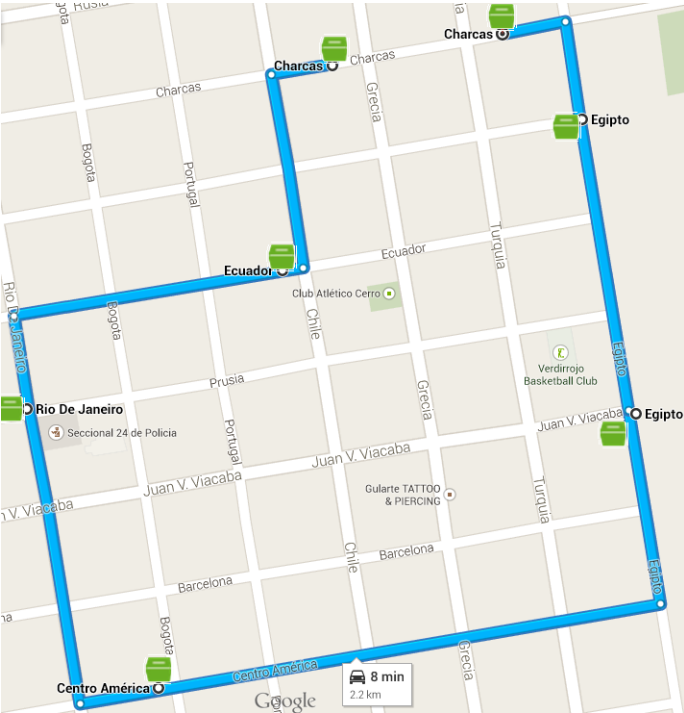


Figura 6. Recorrido del día sábado de la solución de la primera instancia del barrio Cerro. El recorrido empieza por, de los dos extremos, el de la izquierda.

Para el análisis se obtuvieron en cuenta datos como el fitness promedio, la desviación estándar del fitness, la mejor solución obtenida, el tiempo ocurrido hasta la mejor solución obtenida, la cantidad de generaciones hasta la mejor solución obtenida, entre otros.

En las Tablas III a VII en el Apéndice se muestran los resultados de las distintas ejecuciones. Se puede observar que los mejores resultados se obtienen sobre las últimas generaciones, indicando que tal vez se podría no haber parado y sí haber continuado con la ejecución del algoritmo. Tal vez se podrían haber especificado más generaciones, o inclusive no trabajar con un esfuerzo prefijado y que la condición de parada sea dinámica, como por ejemplo que ocurra un cierto número de generaciones sin que cambie el resultado para detenerse. En la Figura 8 se muestra la evolución del mejor fitness con el avance generacional para un caso representativo de la instancia grande del barrio Cerro. Como se puede observar, el fitness sigue bajando luego de mucho tiempo de empezar.

Como buen atributo se puede decir que el algoritmo no queda fácilmente atrapado en óptimos locales, sino que con el tiempo sigue avanzando a mejores soluciones. El uso de mutaciones específicas para recorridos pudo haber sido una



Figura 7. Recorrido del día jueves de la solución de la segunda instancia del barrio Pocitos.

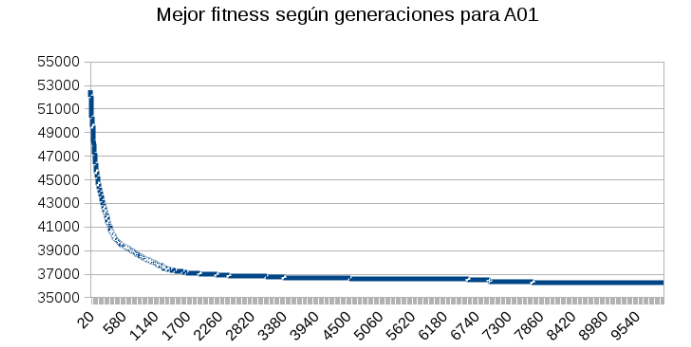


Figura 8. Caso representativo de la evolución del mejor fitness encontrado hasta el momento según el avance de las generaciones, para la primera instancia del barrio Cerro.

buena decisión para esto, ya que probar cambiar ligeramente el recorrido en ciertas partes puede brindar a la población cierta variedad que haga no quedar atrapado en óptimos no globales y dar la oportunidad a cruzarse para obtener mejores valores.

Se realizaron Test de Normalidad utilizando el Test de hipótesis de Lilliefors, arrojando que los datos devueltos para las instancias A02, CH14 y C04 pasan el test, mientras que A01 no lo pasa, con un p-valor de 0,0713, ni tampoco la

instancia CH13, con un p-valor de 0,1865. Como el algoritmo utilizado fue siempre el mismo, consideramos entonces que no se pasa el test, teniendo en cuenta además que pasarlo significa que no hay evidencia para que no sea Normal, mientras que no pasarlo es que la haya. Creemos que un motivo fuerte para que no se comporte como una normal es el hecho de que si en una ejecución puede pasar un día menos, el fitness pega un salto de valor ya que no se está pagando el costo de salir a recorrer ese día. En las Figuras 9 a 13 en el Apéndice se muestran los histogramas del mejor fitness para las ejecuciones para cada instancia, comparados con distribuciones normales de mismo promedio y varianza. Como se puede apreciar a simple vista en las gráficas, varias no se comportan intuitivamente como “Campanas de Gauss”. También en ellas se puede apreciar los “saltos” mencionados.

Cabe la observación de que las mejores soluciones de cada instancia son aproximadamente un 50 % mejores que las peores de cada una.

V-E. Comparación con algoritmo Greedy

Nos comparamos con un algoritmo Greedy que funciona de la siguiente manera: toma cada contenedor, y para los días que no cumple la demanda, toma de ellos aquel que menos ocupada la capacidad del camión tiene. Luego, una vez asignados los contenedores de cada día, se procede a elegir la forma de recorrerlos, utilizando una heurística clásica en TSP: ir recorriendo el lugar más cercano al actual hasta cubrir todos los puntos deseados.

La Tabla I muestra los resultados obtenidos para el algoritmo propuesto y para el que utiliza Greedy. Se puede observar a simple vista que el algoritmo propuesto obtiene soluciones considerablemente mejores que las conseguidas mediante Greedy. Éste último tiene la ventaja de ser mucho más rápido, dando en menos de 1 segundo una solución candidata. Sin embargo, cabe destacar que, con nuestro algoritmo, en menos de 15 minutos, para las instancias grandes, se tiene una solución de mucho mejor calidad.

Inst.	Algoritmo Genético					Greedy		GAP
	Prom.	T. prom.	Dev. est.	Mejor	T. mejor	Mejor	T. mejor	
A01	37.073,575	193,99	2.781,78	31.693	439,696	40.636	0,012	-22,00 %
A02	22.037,100	27,64	2363,42	18.836	848,230	31.325	0,016	-39,96 %
C13	33.019,050	305,19	1.707,90	29.374	448,491	32.016	0,016	-8,25 %
CH03	23.836,625	823,46	156,866	20.889	124,787	27.910	0,020	-25,16 %
CH04	16.675,900	77,178	659,727	15.582	33.229	21.600	0,008	-27,86 %

Tabla I
RESULTADOS COMPARADOS CON GREEDY

Se llevo a cabo con los datos de las distintas instancias el Test U no paramétrico de Mann y Whitney. El resultado arrojó con muy alta probabilidad que el comportamiento de ambos algoritmos son distintos (siendo mejor el del algoritmo propuesto).

VI. CONCLUSIONES Y TRABAJO FUTURO

Hemos realizado un algoritmo que muestra tener buena calidad en sus resultados y mejora ampliamente a la versión Greedy utilizada. Además muestra tener muy buena eficiencia computacional, dada la complejidad del problema, arrojando buenos resultados en considerablemente poco tiempo. Usar algoritmos genéticos resultó ser un buen enfoque. Es positivo notar que nuestro algoritmo supera en promedio un 20 % al Greedy, llegando en una instancia hasta casi un 40 %.

Aunque el algoritmo mostró no parar de mejorar con el paso de las generaciones, muchas veces no pudo acomodar los contenedores para lograr pasar menos días y bajar considerablemente el fitness, quedando atrapado en óptimos locales. Cabe destacar que en algunas ejecuciones sí pudo salir de ellos.

Como trabajo futuro sería deseable tener en cuenta los siguientes puntos:

- Como se mencionó, se podría elegir una condición de parada más adecuada, ya que el algoritmo mostró seguir mejorando aunque se decidió terminar la ejecución. Podrían estudiarse distintas propuestas de condiciones, compararlas y evaluarlas.
- Sería deseable también, para mejorar la eficiencia computacional, poder aprovechar las facilidades que brinda Malva para el paralelismo. El algoritmo mostró tener problemas de concurrencia, lo cual impidió que pueda ser ejecutado en varios procesos y computadoras distintas. Esto, además, permitiría aplicar otras técnicas como usar Islas y Migración.
- Aunque sea mucho más complejo, tal vez pueda ser buena idea (o al menos experimentarlo) hacer un mes modelo, en lugar de una semana modelo, y de esa forma lograr que los efectos de los “bordes” pesen menos sobre las soluciones. Cabe aclarar que la complejidad del problema aumenta en gran medida ya que hay muchos más días a considerar.

REFERENCIAS

- [1] Otman Abdoun, Chakir Tajani y Jaafar Abouchabaka. «Hybridizing PSM and RSM Operator for Solving NP-Complete Problems: Application to Travelling Salesman Problem». En: *International Journal of Computer Science Issues* 9.1 (enero de 2012), págs. 374-378.
- [2] CEMPRE (Compromiso EMpresarial Para el REciclaje). *Preguntas frecuentes*. URL: http://www.cempre.org.uy/index.php?option=com_wrapper&view=wrapper&Itemid=86 (accedido en agosto de 2014).
- [3] Google. *API de rutas de Google*. URL: <https://developers.google.com/maps/documentation/directions/> (accedido en agosto de 2014).
- [4] Instituto Nacional de Estadística (INE). *Población por zona censal en Montevideo*. 2004. URL: <https://catalogodatos.gub.uy/dataset/poblacion-por-zona-censal-en-montevideo> (accedido en agosto de 2014).
- [5] Intendencia de Montevideo. *Centros Comunes Zonales*. URL: <http://www.montevideo.gub.uy/ciudadania/centros-comunes> (accedido en agosto de 2014).

- [6] Intendencia de Montevideo. *GeoServer*. URL: <http://www.montevideo.gub.uy/geoserver/web/> (accedido en agosto de 2014).
- [7] Intendencia de Montevideo. *Mapa de contenedores de residuos domiciliarios*. URL: <http://www.montevideo.gub.uy/ciudadania/ desarrollo - ambiental / limpieza / disposicion - de - residuos / contenedores - de - residuos - domiciliario> (accedido en agosto de 2014).
- [8] Intendencia de Montevideo. *Municipios de Montevideo*. URL: <http://municipios.montevideo.gub.uy/> (accedido en agosto de 2014).
- [9] Intendencia de Montevideo. *Sistema de recolección de residuos domiciliarios con contenedores*. 11 de septiembre de 2013. URL: <http://www.montevideo.gub.uy/ciudadania/ desarrollo - ambiental / limpieza / disposicion - de - residuos / sistema - de - recoleccion - de - residuos -> (accedido en agosto de 2014).
- [10] Yukihiro Matsumoto. *Lenguaje de Programación Ruby*. URL: <https://www.ruby-lang.org/es/> (accedido en agosto de 2014).
- [11] Michael Minn. *MMQGIS*. URL: <http://michaelminn.com/linux/mmqgis/> (accedido en agosto de 2014).
- [12] MyGeodata Portal. *Coordinates transformation on-line*. URL: <http://cs2cs.mygeodata.eu/> (accedido en agosto de 2014).
- [13] QGIS Development Team. *QGIS Geographic Information System*. Open Source Geospatial Foundation. URL: <http://www.qgis.org> (accedido en agosto de 2014).
- [14] Kunitoshi Sakurai. *Hojas de divulgación 17: método sencillo de análisis de residuos sólidos*. URL: <http://www.bvsde.paho.org/eswww/proyecto/repidisc/publica/hdt/hdt017.html> (accedido en agosto de 2014).
- [15] The Malva Project. *Malva*. URL: <https://themalvaproject.github.io/> (accedido en agosto de 2014).

APÉNDICE

Histograma del mejor fitness de cada ejecución para A01

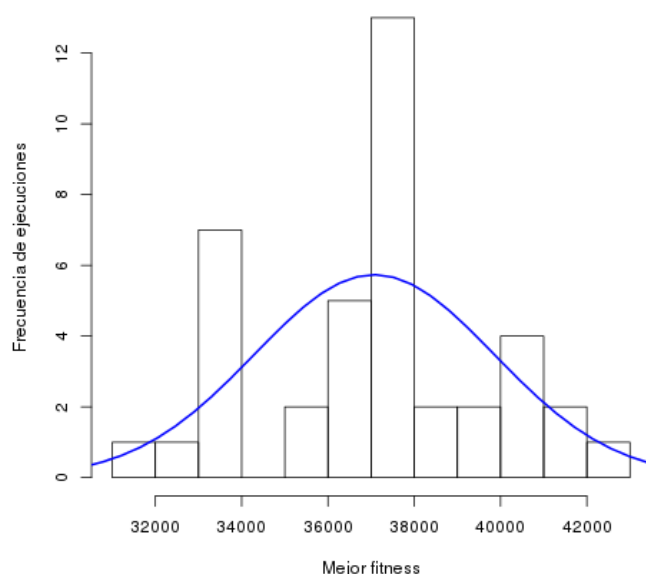


Figura 9. Histograma del mejor fitness de cada ejecución de la instancia de tamaño grande del barrio Cerro, y en azul una distribución normal de mismo promedio y varianza.

Histograma del mejor fitness de cada ejecución para A02

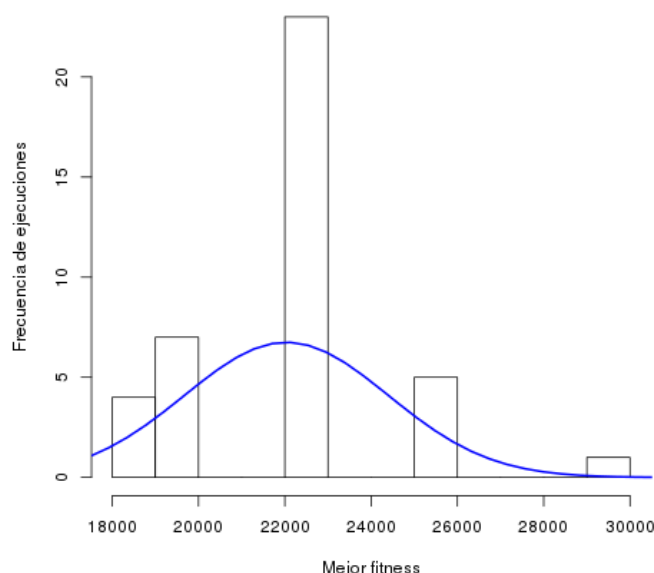


Figura 10. Histograma del mejor fitness de cada ejecución de la instancia de tamaño pequeño del barrio Cerro, y en azul una distribución normal de mismo promedio y varianza.

Población	Cruzamiento	Mutación	Fitness promedio
50	60	0,10 %	17279,275
50	60	1,00 %	15904,825
50	60	10,00 %	16154,925
50	70	0,10 %	17199,050
50	70	1,00 %	15976,600
50	70	10,00 %	15928,800
50	80	0,10 %	17326,475
50	80	1,00 %	16047,750
50	80	10,00 %	15556,225
50	90	0,10 %	17174,000
50	90	1,00 %	15465,400
50	90	10,00 %	15330,525
50	95	0,10 %	17259,000
50	95	1,00 %	15476,850
50	95	10,00 %	14966,325
100	60	0,10 %	17285,150
100	60	1,00 %	15724,850
100	60	10,00 %	16118,400
100	70	0,10 %	17350,825
100	70	1,00 %	16183,125
100	70	10,00 %	15946,125
100	80	0,10 %	17206,600
100	80	1,00 %	15539,175
100	80	10,00 %	15484,475
100	90	0,10 %	15250,350
100	90	1,00 %	15478,250
100	90	10,00 %	15271,175
100	95	0,10 %	17164,875
100	95	1,00 %	15668,900
100	95	10,00 %	14854,100
150	60	0,10 %	17304,400
150	60	1,00 %	15837,850
150	60	10,00 %	16430,975
150	70	0,10 %	17131,275
150	70	1,00 %	16009,950
150	70	10,00 %	15995,425
150	80	0,10 %	17152,450
150	80	1,00 %	15763,575
150	80	10,00 %	15365,700
150	90	0,10 %	17153,250
150	90	1,00 %	15206,225
150	90	10,00 %	15250,350
150	95	0,10 %	16961,750
150	95	1,00 %	14999,700
150	95	10,00 %	14960,950

Tabla II

RESULTADOS PARA LAS INSTANCIAS DE AJUSTES PARAMÉTRICOS

A01						
Ejec.	Mejor	Peor	Evals. mejor	Gen. mejor	T. mejor	Tiempo
1	36.697	57.397	992.900	9.928	170,448	171,678
2	39.923	57.242	998.000	9.979	484,520	485,840
3	37.347	57.454	979.300	9.792	184,340	188,320
4	37.666	57.677	972.700	9.726	169,469	174,186
5	37.747	57.007	975.100	9.750	168,611	172,865
6	33.020	57.968	997.500	9.974	168,781	169,209
7	37.910	57.538	989.200	9.891	170,852	172,693
8	37.735	57.712	942.000	9.419	163,479	173,386
9	41.273	57.589	990.100	9.900	173,945	175,687
10	36.611	58.734	938.900	9.388	157,422	167,557
11	41.899	57.585	997.800	9.977	179,831	180,236
12	36.300	57.472	974.500	9.744	169,481	173,744
13	33.188	57.454	968.700	9.686	164,922	170,179
14	37.454	57.074	949.200	9.491	169,943	179,187
15	40.910	57.361	960.100	9.600	167,383	174,164
16	39.636	57.073	998.700	9.986	179,152	179,415
17	37.266	57.461	990.800	9.907	176,841	178,453
18	40.484	57.771	967.100	9.670	170,954	176,578
19	36.639	57.263	977.100	9.770	171,763	175,731
20	33.009	57.329	982.000	9.819	169,815	172,933
21	37.085	58.090	951.400	9.513	165,636	174,179
22	37.549	57.241	962.700	9.626	164,803	171,124
23	35.653	57.603	985.500	9.854	170,790	173,286
24	31.693	57.330	993.600	9.935	439,696	442,618
25	37.116	57.358	954.200	9.541	170,687	179,061
26	33.035	57.417	983.400	9.833	169,520	172,330
27	32.927	58.061	996.500	9.964	167,713	168,293
28	35.860	57.841	924.700	9.246	527,234	611,946
29	36.186	57.475	978.900	9.788	168,997	172,810
30	42.086	56.908	986.400	9.863	176,815	179,228
31	37.921	56.759	981.400	9.813	169,273	172,462
32	33.156	57.988	992.100	9.920	164,405	165,695
33	38.192	57.627	996.700	9.966	173,877	174,447
34	40.986	58.004	989.200	9.891	179,193	181,101
35	38.538	57.037	997.700	9.976	174,229	174,637
36	33.155	57.989	981.300	9.812	169,834	172,961
37	37.351	57.434	999.700	9.996	176,336	176,404
38	40.530	57.875	999.300	9.992	174,603	174,736
39	33.380	58.781	956.800	9.567	160,553	167,589
40	37.830	58.258	926.700	9.266	163,538	176,612

Tabla III

RESULTADOS PARA LA INSTANCIA GRANDE DEL BARRIO CERRO

A02						
Ejec.	Mejor	Peor	Evals. mejor	Gen. mejor	T. mejor	Tiempo
1	19.127	39.572	792.400	7.923	726,236	1.023,16
2	18.963	38.446	942.200	9.421	228,856	243,724
3	18.836	38.748	983.400	9.833	848,227	869,069
4	22.558	38.847	991.700	9.916	261,648	263,769
5	19.106	39.289	934.700	9.346	284,935	307,045
6	22.609	38.618	688.800	6.887	189,788	263,419
7	22.383	38.835	910.900	9.108	216,894	237,788
8	22.258	38.549	826.000	8.259	500,776	567,697
9	25.590	38.830	983.800	9.837	229,148	232,967
10	22.533	38.989	806.800	8.067	189,897	235,468
11	18.968	39.063	818.400	8.183	198,716	241,569
12	22.426	38.779	993.900	9.938	232,194	233,645
13	25.518	38.674	992.000	9.919	244,331	246,289
14	22.351	38.767	981.100	9.810	228,693	233,099
15	22.209	38.662	967.100	9.670	230,380	238,028
16	18.996	38.833	932.400	9.323	380,949	399,929
17	22.248	38.541	959.900	9.598	302,286	320,098
18	19.011	38.723	956.600	9.565	265,491	276,127
19	22.346	38.841	984.400	9.843	244,230	248,220
20	22.432	38.698	964.000	9.639	249,700	258,695
21	19.094	38.609	417.600	4.175	146,077	358,982
22	19.066	38.703	992.600	9.925	281,672	283,528
23	22.238	39.015	878.000	8.779	213,928	242,599
24	22.497	38.429	992.600	9.925	240,177	241,947
25	25.770	38.463	931.500	9.314	224,962	241,660
26	19.119	38.861	999.300	9.992	250,865	251,061
27	22.447	39.106	849.500	8.494	201,135	237,185
28	22.208	38.929	937.600	9.375	216,230	230,732
29	22.480	38.620	984.900	9.848	360,180	365,891
30	22.521	39.186	914.600	9.145	214,826	234,799
31	22.153	38.726	955.600	9.555	226,541	237,349
32	22.439	38.770	808.300	8.082	192,508	238,577
33	29.189	38.581	967.000	9.669	242,427	250,773
34	25.670	38.664	941.100	9.410	233,834	247,559
35	22.241	38.255	982.700	9.826	254,998	259,123
36	22.504	38.493	993.600	9.935	245,096	246,708
37	25.625	38.705	977.500	9.774	252,457	258,127
38	22.343	38.943	972.200	9.721	235,656	242,449
39	22.259	39.448	966.300	9.662	240,654	248,870
40	19.153	38.875	965.900	9.658	372,330	389,210

Tabla IV
RESULTADOS PARA LA INSTANCIA CHICA DEL BARRIO CERRO

C13						
Ejec.	Mejor	Peor	Evals. mejor	Gen. mejor	T. mejor	Tiempo
1	35.944	50.486	804.700	8.046	160,750	198,834
2	32.352	51.077	960.600	9.605	584,072	610,691
3	33.525	51.210	918.700	9.186	187,911	204,985
4	33.866	51.115	905.900	9.058	179,360	198,330
5	33.955	51.741	958.000	9.579	192,573	201,297
6	31.780	50.529	990.900	9.908	198,002	199,803
7	35.832	50.567	934.700	9.346	192,185	205,464
8	36.516	50.874	974.700	9.746	202,145	207,325
9	32.946	50.893	806.300	8.062	158,933	196,329
10	33.368	51.463	970.400	9.703	203,293	209,405
11	33.113	50.952	705.500	7.054	143,664	203,474
12	33.407	50.682	998.800	9.987	205,017	205,281
13	35.788	50.995	919.600	9.195	185,502	201,618
14	32.493	50.981	989.800	9.897	533,643	539,676
15	35.101	51.589	904.600	9.045	185,477	205,529
16	32.300	51.169	999.700	9.996	467,697	467,893
17	35.123	50.568	956.500	9.564	200,227	209,228
18	35.186	50.993	958.700	9.586	204,452	213,041
19	33.159	51.468	929.000	9.289	193,700	208,451
20	33.728	51.387	980.400	9.803	202,361	206,357
21	32.661	50.451	984.600	9.845	542,356	551,400
22	29.374	51.684	990.400	9.903	448,491	452,966
23	35.149	51.256	979.200	9.791	255,277	261,454
24	35.502	51.223	974.800	9.747	200,328	205,506
25	31.846	51.046	975.500	9.754	878,298	910,333
26	34.243	50.744	936.200	9.361	193,752	207,030
27	31.229	51.124	998.900	9.988	207,496	207,738
28	32.839	50.712	844.400	8.443	200,349	237,315
29	31.040	51.109	992.500	9.924	194,990	196,430
30	29.801	51.694	998.400	9.983	428,980	430,508
31	36.160	50.632	895.100	8.950	188,219	209,952
32	34.123	50.935	965.900	9.658	702,779	726,233
33	33.477	50.593	957.100	9.570	810,056	866,437
34	33.703	50.803	995.200	9.951	204,670	205,655
35	34.323	51.278	973.200	9.731	653,370	675,567
36	35.991	51.806	959.400	9.593	212,049	221,582
37	33.973	51.171	968.900	9.688	200,855	207,267
38	35.247	50.960	847.800	8.477	180,293	211,452
39	35.339	50.886	994.500	9.944	203,369	204,487
40	33.260	51.602	998.100	9.980	520,803	521,663

Tabla V
RESULTADOS PARA LA INSTANCIA GRANDE DEL BARRIO LA BLANQUEADA

CH03						
Ejec.	Mejor	Peor	Evals. mejor	Gen. mejor	T. mejor	Tiempo
1	22.478	41.548	904.660	9.046	133,940	153,433
2	24.021	42.350	949.060	9.490	116,282	122,549
3	23.730	41.557	988.560	9.885	119,162	120,522
4	24.075	42.389	974.760	9.747	195,541	199,420
5	24.405	41.934	801.960	8.019	97,190	121,041
6	24.403	41.904	963.060	9.630	116,553	120,935
7	24.368	41.976	975.060	9.750	121,955	125,009
8	23.125	42.292	932.860	9.328	116,870	125,973
9	24.381	41.596	929.160	9.291	378,795	413,896
10	24.108	41.559	994.360	9.943	135,110	135,812
11	22.793	41.390	922.060	9.220	111,344	120,808
12	20.889	42.025	983.560	9.835	124,787	126,770
13	23.872	41.605	978.860	9.788	118,544	121,071
14	24.506	41.580	888.260	8.882	127,069	143,895
15	22.421	41.500	992.960	9.929	120,159	121,015
16	24.583	42.563	843.760	8.437	117,207	140,712
17	24.460	42.400	993.860	9.938	677,632	683,056
18	23.917	42.775	945.260	9.452	115,645	122,406
19	24.372	42.708	994.160	9.941	122,541	123,272
20	22.509	42.075	997.960	9.979	148,598	148,932
21	24.429	41.660	992.960	9.929	124,447	125,529
22	24.346	41.618	942.260	9.422	114,297	121,256
23	24.439	41.937	989.360	9.893	118,294	119,556
24	24.030	42.348	974.660	9.746	116,712	119,735
25	24.488	41.727	987.660	9.876	120,293	121,844
26	22.270	41.751	995.760	9.957	133,177	133,716
27	24.123	41.929	996.760	9.967	256,799	257,988
28	22.646	41.822	971.760	9.717	118,603	122,039
29	23.392	41.300	949.360	9.493	123,649	138,675
30	24.540	41.917	932.560	9.325	182,781	198,557
31	24.233	41.497	997.960	9.979	121,266	121,524
32	23.850	42.016	891.260	8.912	281,251	309,603
33	24.467	41.704	998.660	9.986	129,051	129,229
34	23.795	41.874	994.160	9.941	119,737	120,450
35	24.116	41.823	859.360	8.593	102,819	119,658
36	23.850	41.814	973.060	9.730	117,523	120,760
37	24.257	41.868	993.360	9.933	118,769	119,584
38	24.293	41.925	994.560	9.945	124,387	125,051
39	24.119	42.289	977.560	9.775	118,058	120,852
40	24.366	42.104	901.460	9.014	317,814	352,444

Tabla VI

RESULTADOS PARA LA INSTANCIA GRANDE DEL BARRIO POCITOS

CH04						
Ejec.	Mejor	Peor	Evals. mejor	Gen. mejor	T. mejor	Tiempo
1	16.848	28.572	542.160	5.421	40,430	73,694
2	17.284	27.770	956.660	9.566	94,283	98,799
3	16.081	28.327	966.760	9.667	72,552	75,141
4	17.168	27.676	819.060	8.190	103,078	127,331
5	17.075	28.087	900.560	9.005	75,001	83,281
6	15.997	28.000	743.260	7.432	53,792	72,269
7	15.695	28.441	746.260	7.462	63,224	86,123
8	17.201	28.515	920.060	9.200	82,304	90,807
9	15.582	28.212	422.860	4.228	33,229	76,427
10	16.079	28.263	935.660	9.356	74,386	79,388
11	17.071	27.883	955.560	9.555	69,617	72,846
12	15.806	28.178	748.460	7.484	60,831	87,599
13	15.682	28.210	971.360	9.713	112,761	116,316
14	16.034	27.920	938.460	9.384	87,842	92,750
15	16.986	28.508	993.060	9.930	77,734	78,270
16	15.873	28.356	949.660	9.496	127,723	136,086
17	17.471	27.994	520.760	5.207	41,763	82,152
18	17.690	28.002	825.560	8.255	88,779	104,884
19	17.294	28.718	993.560	9.935	141,404	141,945
20	17.225	27.964	622.860	6.228	62,377	111,498
21	17.056	27.589	646.760	6.467	67,712	99,844
22	16.942	28.007	842.460	8.424	65,339	78,759
23	15.888	28.141	917.060	9.170	68,1024	74,060
24	17.314	28.088	767.460	7.674	63,175	83,396
25	17.324	28.109	852.460	8.524	63,873	74,792
26	15.945	28.142	547.960	5.479	51,944	85,659
27	17.224	27.695	503.960	5.039	50,882	114,266
28	15.828	28.576	987.260	9.872	123,599	125,423
29	17.114	28.088	875.460	8.754	92,562	103,387
30	17.219	28.077	757.360	7.573	56,836	74,730
31	17.244	28.207	978.460	9.784	89,998	91,990
32	17.338	28.211	493.860	4.938	38,391	80,966
33	16.034	27.905	994.760	9.947	73,062	73,469
34	17.289	28.093	912.760	9.127	76,063	83,764
35	17.075	27.623	856.960	8.569	70,386	81,929
36	16.096	28.564	902.660	9.026	128,098	141,454
37	17.056	28.028	930.560	9.305	69,496	74,688
38	17.008	27.802	950.760	9.507	78,176	82,074
39	15.712	28.044	803.460	8.034	96,761	116,395
40	16.188	28.215	787.360	7.873	99,552	127,666

Tabla VII

RESULTADOS PARA LA INSTANCIA CHICA DEL BARRIO POCITOS

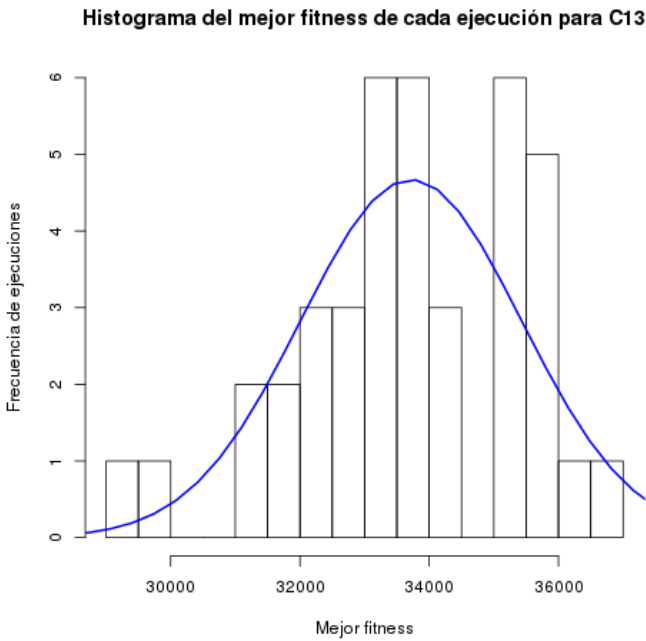


Figura 11. Histograma del mejor fitness de cada ejecución de la la instancia de tamaño grande del barrio La Blanqueada, y en azul una distribución normal de mismo promedio y varianza.

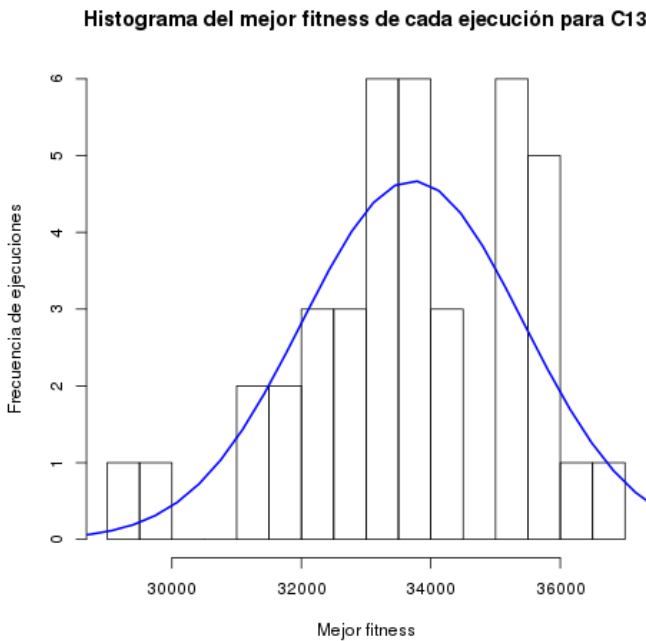


Figura 13. Histograma del mejor fitness de cada ejecución de la la instancia de tamaño pequeño del barrio Pocitos, y en azul una distribución normal de mismo promedio y varianza.

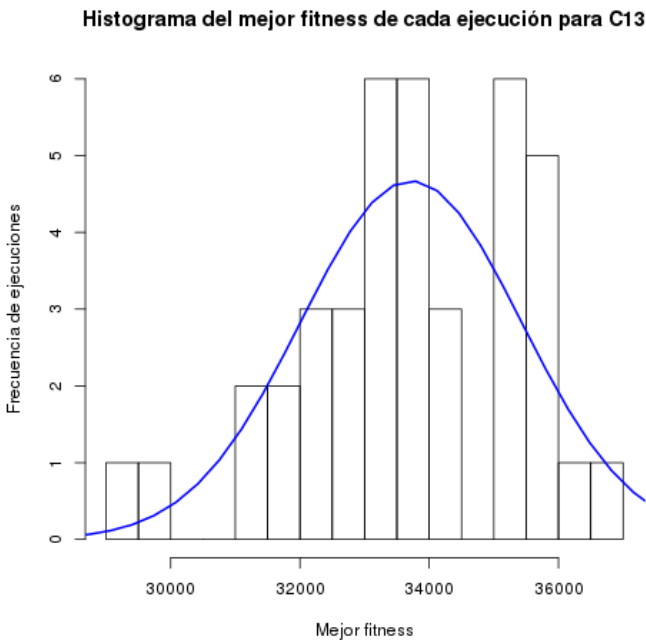


Figura 12. Histograma del mejor fitness de cada ejecución de la la instancia de tamaño grande del barrio Pocitos, y en azul una distribución normal de mismo promedio y varianza.