

Algoritmos Evolutivos pre-Informe Proyecto Final

1st Guillermo Toyos

5.139.879-9

Facultad de Ingeniería

Montevideo, Uruguay

guillermo.toyos@fing.edu.uy

2nd Federico Vallcorba

5.102.078-4

Facultad de Ingeniería

Montevideo, Uruguay

federico.vallcorba@fing.edu.uy

I. INTRODUCCIÓN

En un mundo plagado de basura, donde prácticamente todo lo que nos rodea tarde o temprano se transformará en un residuo, resulta crucial disponer de un proceso de recolección de basura exitoso. Además del reciclaje y reuso de los desechos, que son fundamentales para reducir la presión sobre el sistema de recolección (más del 30 % de los residuos que se generan en Montevideo se pueden reciclar), es de vital importancia evitar al máximo la acumulación de basura en los centros poblados.

La acumulación de desechos provoca una cantidad importantísima de problemas para la salud humana y el medio ambiente. Desde el punto de vista sanitario, predisposición a infecciones y enfermedades gastrointestinales. En lo ambiental, la basura es el principal causante de contaminación del agua y el aire, la degradación de los suelos y la alteración de los ecosistemas. En las zonas más descuidadas se produce un aumento de plagas, además del taponamiento de las bocas de tormenta que puede producir inundaciones.

En Montevideo los hogares generan unas 1200 toneladas de basura a diario. Es fundamental disponer de un sistema de recolección de residuos que permita recolectarla de forma correcta, cuidando la salud de todos los ciudadanos. Para esto, la Intendencia de Montevideo dispone de una flota de camiones que trabaja las 24 horas, y más de 13000 contenedores de residuos. [1]

Adicionalmente, resulta relevante minimizar la distancia y tiempo que recorren los camiones de basura diariamente con el afán de reducir las emisiones de CO_2 , ya que la flota de camiones de Montevideo, como en gran parte del mundo, utiliza combustibles fósiles.

II. PROBLEMA A RESOLVER

El artículo a continuación trata sobre la optimización del recorrido que realiza la flota de camiones recolectores de basura en distintas instancias de la realidad, centrándose en Montevideo. Cada instancia se encuentra determinada por la cantidad de camiones recolectores, la cantidad de contenedores de residuos, su ubicación y el número máximo de días que pueden pasar sin ser levantados por un camión. También se conoce la ubicación del vertedero, desde donde comienza y finaliza el recorrido de cada camión. Se considerará que los camiones realizan dos turnos diarios de 8 horas cada uno (uno matutino y uno nocturno), por lo que cada recorrido no puede durar más de 8 horas. Además se sabe que los camiones

no pueden levantar más de 100 contenedores, por lo que si algún camión llega a dicha cifra, debe volver al vertedero para vaciarse antes de seguir levantando contenedores.

El problema en concreto es, sabiendo cuantos días pasaron desde la última vez que cada contenedor fue levantado, encontrar la planificación de los recorridos de la flota de camiones que minimiza la distancia que recorren, sujeto a que ningún contenedor supera el número máximo de días que puede pasar sin ser levantado.

III. JUSTIFICACIÓN DEL USO DE AE

El problema planteado es un problema de optimización, ya que se busca minimizar la distancia y tiempo de recorrido en los itinerarios de los camiones mientras se cumplen ciertas restricciones. Esto es, para cada camión asignarle los contenedores que debe levantar cada día. Este problema puede entenderse como una instancia del Vehicle Routing Problem (VRP) con restricciones inherentes de la realidad, el cual cae dentro de los problemas NP-difíciles, con espacio de soluciones $2^{n \times k}$. Teniendo esto en cuenta, se sabe que realizar una búsqueda exhaustiva de la solución óptima no es posible, al menos en un tiempo razonable. Por esto, es de interés el uso de algoritmos evolutivos para lograr una solución próxima a la óptima, en un tiempo viable.

IV. MODELADO DEL PROBLEMA

Se considerará una instancia genérica en la cual se dispone de k camiones de basura y n contenedores. Para cada contenedor n se conoce su valor de fr_n , el cual indica la cantidad máxima de días que pueden pasar sin que sea levantado el n -ésimo contenedor.

Sea $c_d \in \mathbb{R}^n$, un vector que indica cuantos días pasaron desde que cada contenedor fue vaciado para el día d . El mismo se encuentra definido por: $c_0 = c(0)$, $c_{d+1} = f(c_d) + \mathbb{1}$. Donde $f(c_d) \in \mathbb{R}^n$ vale 0 en las entradas de los contenedores que fueron vaciados, y lo mismo que en c_d en el resto, y $\mathbb{1}$ es una columna de unos.

Por lo tanto, el problema se resume a dado un $c(0)$, hallar la planificación de la flota de camiones de los días siguientes, optimizando el recorrido de los mismos, y cuidando que ningún contenedor se desborde. Es importante dar la planificación de los P días siguientes, siendo $P = \max\{fr_n\}$, debido a que de otra manera no se estarían contemplando problemas que pudieran surgir más adelante con otros contenedores.

La planificación de los recorridos de la flota queda determinada por los itinerarios que se le definen a cada camión recolector de basura. A cada camión se le define un itinerario por turno (dos por día), que indica que contenedores debe levantar el mismo en dicho turno. Cada itinerario se define como una matriz $M \in \{0, 1\}^{k \times n}$, donde cada índice indica si el k -ésimo camión levanta el n -ésimo contenedor o no. Para diferenciar los itinerarios de los diferentes turnos y días, se denomina M_{md} al itinerario del turno matutino del día d , y M_{nd} al itinerario del turno nocturno del día d .

Para calcular el tiempo y la distancia que implica cada recorrido de los camiones se dispone de las matrices $D, T \in \mathbb{R}^{n \times n}$ que indican la distancia y el tiempo respectivamente que le lleva a un camión ir de cada contenedor al resto. Además se conoce la distancia y el tiempo que le lleva a un camión ir desde el vertedero a cada contenedor, y viceversa. Adicionalmente se utiliza un algoritmo que, dado una lista de contenedores, minimiza la distancia que implica el recorrido de los mismos (TSP).

V. ESTRATEGIA DE RESOLUCIÓN

V-A. Representación de las soluciones

Dado que las soluciones van a ser las matrices binarias que representan los contenedores que levantará cada camión en cada itinerario, se utilizará una representación binaria donde se concatenarán las matrices representándolas como una tira.

V-B. Función de Fitness

La calidad de una solución está dada por el hecho de que se cumple la frecuencia de recolección, no se excede el horario (8hs) de cada turno, cuantos de estos se utilizan y la distancia que estos recorren. Es decir, el objetivo es minimizar la distancia y cantidad de camiones. A su vez, es preferible que los camiones junten la basura durante el turno nocturno que el matutino para minimizar su impacto en el tráfico de la ciudad.

Tomando lo anterior en consideración, y sea $S(x)$ el conjunto de matrices de itinerarios M que codifica x , se define el fitness de x como:

$$f(x) = \sum_{M \in S(x)} t(M) \sum_{j=0}^n \alpha \times distancia(M_j) + base(M_j)$$

Si la solución es válida, si no lo es porque no respeta las frecuencia de recolección,

$$f(x) = -(\sum_{M \in S(x)} desborde(M))^2$$

Donde $distancia(M)$ es la distancia recorrida por los camiones para ese itinerario, base representa el costo base de utilizar el camión en ese día (si no se utiliza (i.e toda la columna vale 0) $base(M_j) = 0$), α es un factor para la distancia y $t(M)$ es otro factor que depende si M corresponde a un turno matutino o nocturno (Es preferible que los camiones circulen por la noche). $desborde(M)$ devolverá la cantidad de contenedores para los cuales no se respetó la frecuencia horaria correspondiente (notar que un itinerario M tiene asociado un día).

V-C. Operadores evolutivos

Proponemos utilizar los dos operadores del algoritmo genético simple [2]: cruzamiento en un punto y mutación de un bit como los dos operadores evolutivos principales del algoritmo. Adicionalmente, pensamos utilizar un operador de reparación cuando al aplicar los operadores anteriores se generen soluciones que no cumplan la restricción de tiempo. Se les asignara en 0 entradas positivas de forma aleatoria hasta que la solución sea valida. Para incrementar la diversidad, pensamos agregar un operador de migración en nuestro algoritmo paralelo para comunicar el material genético entre las islas”.

V-D. Técnicas Avanzadas

En cuanto a técnicas avanzadas pensamos aplicar fitness sharing y un algoritmo evolutivo paralelo. La función de distancia que utilizará fitness sharing será la Distancia de Hamming, la cual indica la cantidad de posiciones donde hay símbolos diferentes. Por otro lado, planeamos que nuestro algoritmo evolutivo sea paralelo utilizando el método de subpoblaciones distribuidas que van a comunicarse únicamente con el operador de migración descrito en la sección anterior.

VI. EVALUACIÓN EXPERIMENTAL

Para la evaluación experimental se definirán como instancias de evaluación toda la ciudad de Montevideo, e instancias aleatorias de las matrices D y T de distintas dimensiones para probar la ductilidad del algoritmo.

Los parámetros que se deberán ajustar son las probabilidades de los operadores evolutivos (mutación, crossover y migración), y la cantidad de islas y la población de las mismas. Este ajuste se realizará sobre matrices D y T aleatorias de dimensiones más pequeñas para poder contemplar todas las posibles combinaciones de los parámetros considerados en un tiempo razonable.

Además de estudiar la eficiencia computacional de nuestro algoritmo paralelo, realizaremos tests estadísticos para determinar la calidad de las soluciones y sus métricas.

A su vez, pensamos construir un algoritmo greedy y realizar un análisis estadístico para probar que los resultados de nuestro algoritmo son superiores a este. El algoritmo greedy seguirá una heurística de: ”Dado un camión que pasa por el contenedor c , el siguiente contenedor por el que pasará será el que sea más cercano al contenedor c ”.

Finalmente, pensamos comparar el itinerario generado para la ciudad de Montevideo con el itinerario real que realizan los camiones comparando la cantidad de camiones que utilizan y tiempo que recorren. Esta información está disponible de forma abierta en [3]

REFERENCIAS

- [1] <https://montevideo.gub.uy/areas-tematicas/gestion-de-residuos/mas-transparencia-de-la-gestion>
- [2] D. Goldberg, ”Genetic Algorithms in Search, Optimization & Machine Learning”, Adison-Wesley, 1989.
- [3] Catalogo de datos del gobierno: <https://catalogodatos.gub.uy/>