Universidad Simón Bolívar
Departamento de Computación y Tecnología de la Información
CI-5651 - Diseño de Algoritmos I
Trimestre Enero-Marzo 2017

Proyecto I: Una heurística para el PRPP

1. Introducción

Los Problemas de Enrutamiento por Arcos con Beneficios y Costos (PARPs por sus siglas en inglés, *Prize-Collecting Arc Routing Problems*), fueron introducidos por Aráoz, Fernández y Zoltán [2] y ellos son una generalización de los Problemas de Enrutamiento por Arcos (ARPs por sus siglas en inglés, *Arc Routing Problems*). Los Problemas de Enrutamientos por Arcos consisten en encontrar el menor costo o mayor beneficio de atravesar algunos lados o arcos de un grafo, estando sujeto a una serie de restricciones. Los principales Problemas de Enrutamientos por Arcos son el Problema del Cartero Chino y el Problema del Cartero Rural. El libro editado por Dror [5] es la principal referencia sobre los ARPs. El PARP es definido sobre un grafo en donde hay un vértice d llamado depósito. Cada lado posee una función de beneficio y otra de costo. La función de beneficio sólo es tomada en cuenta la primera vez que se atraviesa el lado. El objetivo del PARP es encontrar un ciclo de máximo beneficio que pase por d. Aráoz et al. [2] muestran que los PARPs son NP-hard y que constituyen una generalización de la mayoría de los Problemas de Enrutamientos por Arcos y del Problema del Agente Viajero. El PARP básico es el Problema del Cartero Rural con Beneficios y Costos (PRPP, por sus siglas en inglés *Prize-Collecting Rural Postman Problem*). Los otros PARPs son definidos imponiendo condiciones adicionales.

EL PRPP consiste en encontrar el ciclo de máximo beneficio que pase por el depósito en un grafo no dirigido. En el PRPP la demanda de servicio se encuentra en los lados del grafo. Cuando en el PRPP se da un servicio a un lado no sólo se incurre en un costo sino que se recibe un beneficio. El costo está asociado con el hecho de atravesar el lado y la ganancia se debe al servicio que se presta al lado. Cuando se construye una ruta se toma en cuenta el costo de atravesar cada uno de los lados que la componen. Es posible que un lado sea atravesado más de una vez. Solamente cuando se atraviesa un lado por primera vez en una ruta, se obtiene una recompensa. El objetivo del PRPP es encontrar una ruta, que parta y termine en el depósito, que maximice el beneficio total obtenido por servir a los lados menos el costo de atravesarlos.

En la Figura 1 se observa un ejemplo de un grafo que es una instancia del PRPP, en donde el vértice que corresponde al depósito está denominado como d. En la Figura 2 se muestra el ciclo que es la solución óptima del problema de la Figura 1.

Aráoz, Fernández y Meza [1] son quienes propusieron la primera solución algorítmica para el PRPP con un algoritmo de dos fases que usa diferentes mecanismos de resolución (*solvers*) en cada una de ellas. En la primera fase se obtiene una solución aproximada del problema por medio de la aplicación una relajación del sistema entero de desigualdades lineales que modelan el problema junto con una heurística, la cual es una adaptación de la heurística 3T [7]. En la segunda fase se aprovecha la solución resultante de la primera fase para obtener una solución óptima del problema.

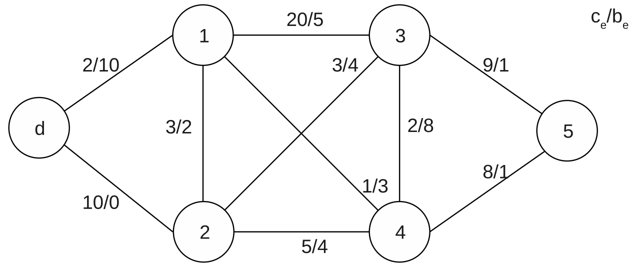

Figura 1: Ejemplo de una instancia del PRPP, c_e/b_e indica los valores costo/beneficio de un lado e

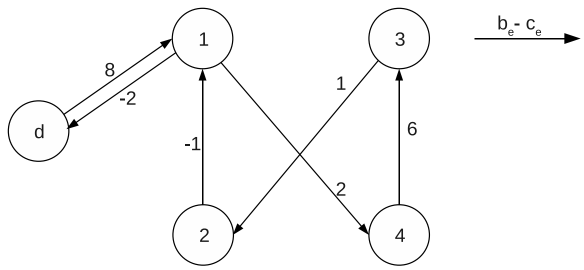

Figura 2: Solución óptima de la instancia de la Figura 1. El ciclo es d-1-4-3-2-1-d con un beneficio igual a 14

Hay potenciales aplicaciones prácticas para el PRPP. Las aplicaciones de los Problemas de Enrutamientos por Lados tratan de minimizar costos. Se ha decidido *a priori*, que la demanda de servicio se encuentra exactamente en ciertos sitios. Debido a esto la meta no es decidir cuáles lados van a ser atendidos, sino determinar el menor costo de atravesarlos. Con el PRPP se puede modelar problemas prácticos más complejos. Una empresa que desea maximizar sus ganancias, puede decidir que la demanda de un lado no será atendida, a menos que de ella se derive un beneficio para la compañía. Un ejemplo en donde se aplica este principio, es en la recolección de papeleras de reciclaje por parte de empresas privadas. Otro ejemplo es el servicio de correo manejado por empresas privadas, las cuales pueden decidir los distritos a los que les prestarán servicios.

2. Problema del Cartero Rural con Beneficios y Costos

En esta sección definimos formalmente el PRPP y presentamos algunas notaciones y definiciones que se van a usar en este trabajo.

Definición 1. *Sea $G(V, E)$ un grafo no dirigido con conjunto de vértices V y un conjunto de lados E, el cual posee un vértice distinguible d al cual llamamos depósito y sean dos funciones de E en $\mathbb{R}^+$, la función de ganancia b y la función de costo c. El PRPP consiste en encontrar un ciclo $\mathcal{C}^*$ que maximice el valor de*

$$\sum_{e \in E(\mathcal{C})} (b_e - t_e c_e) \tag{1}$$

donde $\mathcal{C}$ es un ciclo que pasa por d, el cual no es necesariamente simple puede tener lados repetidos. Tenemos que t_e es el número de veces que el lado e es atravesado en el ciclo $\mathcal{C}$ y $E(\mathcal{C})$ es el conjunto de lados del ciclo $\mathcal{C}$.

A continuación las notaciones a usar y otras definiciones

$V(H)$: Conjunto de vértices de un subgrafo H. Si $H = G$ entonces $V(G) = V$.

$E(H)$: Conjunto de lados de un subgrafo H. Si $H = G$ entonces $E(G) = E$.

$\gamma(S)$: Dado el conjunto S tal que $\emptyset \subset S \subseteq V$ se tiene que $\gamma(S) = \{e \in E \mid e = \{u, v\}, u, v \in S\}$ son el conjunto de lados con ambos vértices en S.

Para el conjunto singleton no se usan llaves. Por ejemplo la siguiente expresión es válida $\gamma(v) \equiv \gamma(\{v\})$.

Las siguientes funciones se definen sobre los lados e de un grafo G.

- $\varphi_e = b_e - c_e$. Se tiene que φ_e es el beneficio que se obtiene cuando se cruza un lado por primera vez
- $\psi_e = b_e - 2c_e$. Se tiene que ψ_e es el beneficio que se obtiene cuando se cruza un lado dos veces

Se divide el conjunto de lados E en tres conjuntos:

$$P = \{e \in E \mid \varphi_e < 0\} \qquad R = \{e \in E \mid \psi_e \geq 0\} \qquad Q = \{e \in E \backslash R \mid \varphi_e \geq 0\}$$

Es de observar que cuando se atraviesa un lado $e \in E$ por primera vez se obtiene un beneficio φ_e. En las siguientes veces que se atraviesa ese lado e, el beneficio que se obtiene cada vez es $-c_e$. El conjunto de lados R juega un rol importante en el PRPP, parecido al conjunto de lados requeridos en el RPP

Definición 2 (Grafo Euleriano)**.** *Un grafo es euleriano si posee un ciclo, tal que cada uno de los elementos del conjuntos de lados E aparece sólo una vez en el ciclo.*

Definición 3 (Grafo Par)**.** *Un Grafo Par es un grafo en el cual todos los vértices tienen grado par.*

Teorema 1 (Por Euler en [6])**.** *Sea G un grafo conexo con al menos dos vértices: G es euleriano si solo sí cada uno de sus vértices tiene grado par.*

Denotaremos una instancia del PRPP que se aplica a un grafo G, con un depósito d, un función de beneficio b y una función de costo c, como (G, d, b, c). De las anteriores definiciones, tenemos que el grafo inducido por una solución factible de una instancia PRPP (G, d, b, c) es un grafo euleriano que pasa por d.

G_R: Es el subgrafo $G_R \equiv (V(R) \cup d, R)$ que se obtiene con el conjunto de lados R y el depósito d. Donde $V(R)$ es el conjunto de vértices incidentes en lados de R.

C_k: Son las componentes conexas del grafo G_R, donde $k \in \{0, \ldots, n\}$. Se asume que $d \in C_0$

$V(C_k) = V_k$: Conjunto de vértices de la componente conexa C_k

$\gamma_R(V_k)$ Se tiene que con $\gamma(V_k)$ se obtienen un conjunto de lados que corresponden al grafo original. Mientras que con $\gamma_R(V_k)$ se obtienen un conjunto de lados que corresponden al grafo G_R. Es decir, $\gamma_R(V_k) = \gamma(V_k) \cap R$.

A continuación se presentan algunas propiedades que poseen las soluciones óptimas del PRPP. Estas fueron deducidas y probadas por por Aráoz, Fernández y Zoltán [2]. Se tiene que $\mathscr{C}^*$ representa a un ciclo que es solución óptima del PRPP.

Dominancia 1. *Ninguno de los lados* $e \in E$ *se encuentra presente más de dos veces en* $\mathscr{C}^*$.

Dominancia 2. *Si un lado* $e \in E$ *es atravesado con un valor* c_e *en* $\mathscr{C}^*$, *entonces es también atravesado con valor* φ_e *en* $\mathscr{C}^*$.

Dominancia 3. *Sea* $e \in \mathscr{C}^*$, *si para alguna de las componentes conexas* C_k *de* G_R *se tiene que* $V(e) \cap V_k \neq \emptyset$, *entonces todos los lados de* $\gamma_R(V_k)$ *están en* $\mathscr{C}^*$.

Observación 1. *La Dominancia 3 implica que los lados en* $\gamma_R(V_k)$, *o se encuentran todos en la solución óptima* $\mathscr{C}^*$ *o ninguno de ellos está en* $\mathscr{C}^*$. *Para cada* $k \in P$ *se tiene que* e_k^R *es elegido para ser uno de los lados en* C_k *con mayor beneficio* φ_e. *La Dominancia 3 también trae como consecuencia que si cualquier lado que no pertenezca al conjunto* R *y se encuentre en el ciclo* $\mathscr{C}^*$ *es incidente con cualquier vértice de* V_k, *entonces implica que todos los lados* $\gamma_R(V_k)$ *están en* $\mathscr{C}^*$.

Lema 1. *En cualquier solución óptima* $\mathscr{C}^*$ *los lados* $e \in P \cup Q$, *son atravesados dos veces sólo para asegurar conectividad.*

Preprocesamiento 1. *Sea* C_k *una componente conexa de* G_R *y sea* $e \in \gamma(V_k) \setminus R$ *para cualquier* k. *Se tiene que el lado* e *es atravesado a lo sumo una vez en solución óptima* $\mathscr{C}^*$

Observación 2. *Se tiene que la Dominancia 3 y el Preprocesamiento 1 implican que si un lado* $e \in \gamma(V_k) \setminus R$ *se encuentra en* $\mathscr{C}^*$, *entonces también todos los lados* $\gamma_R(V_k)$ *están en* $\mathscr{C}^*$.

Preprocesamiento 2. $\gamma_R(V_0) \subseteq \mathscr{C}^*$. *Esto es que todos los lados de la componente conexa de* G_R *que contenga al depósito están en* $\mathscr{C}^*$.

3. Actividades a realizar

Se quiere que diseñe un algoritmo que obtenga soluciones factibles aproximadas al PRPP. El algoritmo a diseñar debe ser un *Algoritmo Ávido* o hacer uso de la técnica *Divide and Conquer*. También es válido diseñar un algoritmo que haga uso de las dos técnicas mencionadas anteriormente.

Las implementaciones de sus algoritmos deben ser probadas sobre un conjunto de 118 instancias del PRPP que fueron formuladas por Aráoz, Fernández y Meza [1]. Estas instancias fueron generadas de un conjunto de instancias del Problema del Cartero Rural RPP. En el apéndice A se muestra en detalle las características de cada una de las instancias. Las 118 instancias se dividen en 5 grupos: ALBAIDA, CHRISTOFIDES, DEGREE, GRID y RANDOM. En estas instancias los nodos están identificados con números enteros y el depósito siempre es el vértice identificado con el número 1.

Los lenguajes de la programación con los que se puede implementar su solución son: `C`, `C++`, `Java` y `Go`. Su código debe venir acompañado con un archivo `Léeme.txt` en donde se explique como compilar y ejecutar su aplicación.

Se define el porcentaje de desviación estándar como: $100 * \frac{\text{Valor óptimo} - \text{Valor solución heurística}}{\text{Valor óptimo}}$. donde *Valor óptimo* es la ganancia de la solución óptima para una instancia dada, y *Valor solución heurística* es la ganancia que obtiene la implementación de un algoritmo heurístico.

Los resultados deben presentarse en 5 tablas, una tabla por cada grupo de instancias. Cada tabla de resultado debe tener el siguiente formato:

Instancia	**Vo**	**%dHeurPlanos**	**%dHeur**	**tHeur** (seg)
Instancia-1	*	*	*	*
Instancia-2	*	*	*	*
⋮	*	*	*	*
Instancia-N	*	*	*	*
Totales	-	*	*	*

Donde:

Vo: es la ganancia de la solución óptima,

%dHeurPlanos: es el porcentaje de desviación estándar de la heurística de planos de cortes [1],

%dHeur: es el porcentaje de desviación estándar de la heurística diseñada por usted,

tHeur: es el tiempo de ejecución de la heurística diseñada por usted.

Totales: es la sumatoria de los valores de la columna.

Dada una instancia *nombre-instancia*, la salida de su aplicación debe crear un archivo llamado *nombre-instancia*`-salida.txt`, que debe contener en su primera línea la ganancia de la instancia *nombre-instancia*, y en la segunda línea debe tener el ciclo encontrado con el orden de visita de los vértices, separados por un espacio. Por ejemplo, el ciclo de la Figura 3, debería generar un archivo con el siguiente contenido:

```
14
d 1 4 3 2 1 d
```

Debe hacer un breve informe con dos secciones:

1. **Descripción de la solución propuesta:** una descripción y justificación de la solución propuesta y el algoritmo diseñado.

2. **Resultados experimentales y análisis:** las tablas con los resultados experimentales y un análisis de los mismos.

Los proyectos con los mejores resultados tendrán puntos extras.

En este tipo de problemas de enrutamiento, además de los algoritmos que construyen soluciones factibles, existen otro tipos de algoritmos que mejoran soluciones factibles. Estos algoritmos se conocen como algoritmos de mejoras de soluciones, y reciben como entrada una solución factible, y por medio de una serie de procesos intenta mejorar el valor de la solución de entrada. Los algoritmos retornan la misma solución de entrada, en caso de no haberla podido mejorar, o retornan una solución factible con mejor valor de la función objetivo. Existen varias algoritmos de mejoras de soluciones para el PRPP, en el apéndice B se muestran varios algoritmos de mejoras de soluciones. De forma opcional puede implementar algún algoritmo de mejora de soluciones, como los indicados en el apéndice B o uno diseñado por usted. Se dará puntos extras por la implementación de un algoritmo de mejora de soluciones.

La cantidad máxima de puntos extras que se otorgarán son 6.

4. Condiciones de entrega

La entrega de este proyecto la debe realizar por email, antes del miércoles de la semana 7 hasta las 11:59 pm.

Referencias

[1] Aráoz, J., Fernández, E., and Meza, O. Solving the prize-collecting rural postman problem. *European Journal of Operational Research 196*, 3 (2009), 886–896.

[2] Aráoz, J., Fernández, E., and Zoltan, C. Privatized rural postman problems. *Computers & Operations Research 33*, 12 (2006), 3432–3449.

[3] Christofides, N., Campos, V., Corberan, A., and Mota, E. An algorithm for the rural postman problem. *Imperial College Report IC-OR-81-5 81* (1981).

[4] Corberán, A., and Sanchis, J. A polyhedral approach to the rural postman problem. *European Journal of Operational Research 79*, 1 (1994), 95–114.

[5] Dror, M. *ARC Routing: Theory, Solutions and Applications.* Kluwer Academic Publishers, 2000.

[6] Euler, L. Solutio Problematis ad Geometriam Situs Pertinentes. *Commentarii Academiae Petropolitannae 8* (1736), 128–140.

[7] Fernández, E., Meza, O., Garfinkel, R., and Ortega, M. On the Undirected Rural Postman Problem: Tight Bounds Based on a New Formulation. *Operations Research 51*, 2 (2003), 281–291.

[8] Garfinkel, R., and Webb, I. On crossings, the crossing postman problem, and the rural postman problem. *Networks 34*, 3 (1999), 173–180.

[9] Hertz, A., Laporte, G., and Nanchen, H. Improvement procedures for the undirected rural postman problem. *INFORMS Journal on Computing 11*, 1 (1999), 62.

Guillermo Palma / gvpalma@usb.ve / Enero de 2017

A. Información sobre las Instancias del PRPP

Las 118 instancias se dividen en 5 grupos. El primer grupo lo llamamos ALBAIDA y contiene dos instancias ALBAIDAA y ALBAIDAB. Fue generado de las instancias de Coberán y Sanchis [4], las cuales fueron obtenidas de un grafo de la ciudad Albaida en España. muestran en la Tabla 2. El segundo grupo lo nombramos CHRISTOFIDES, contiene 24 instancias las cuales fueron generadas a partir de las instancias de Christofides et al. [3], se muestran en la Tabla 3. Los tres últimos grupos fueron obtenidos de las instancias del RPP de propuestas por Hertz et al [9]. El tercer grupo es denominado como DEGREE y esta compuesto de 36 instancias formada por grafos de grado 4 y se muestra en la Tabla 4. El cuarto grupo lo componen 36 instancias que corresponden a grafos tipo *grid*, al cual llamamos GRID y las instancias se muestran en la Tabla 5. El último grupo lo llamamos RANDOM y lo integran 20 instancias generada aleatoriamente. En la Tabla 6 se muestran las características de las instancias RANDOM.

En las Tablas 2, 3, 4, 6 y 5, se muestran las características de las instancias. El significado del encabezado de las columnas de las tablas de las instancias, se presenta en la Tabla 1.

Columna	**Significado**
$\|V\|$	Número de vértices del grafo que corresponde a la instancia
$\|E\|$	Número de lados del grafo que corresponde a la instancia
$\|R \cup Q\|$	Número de lados del tipo R y del tipo Q en la instancia
$\#comp.R \cup Q$	Número de componentes conexas del grafo $G_{R\cup Q}$ inducido por los lados R y Q
$\|R\|$	Número de lados del tipo R en la instancia
$\#comp.R$	Número de componentes conexas del grafo G_R inducido por los lados R
Vo	Valor o ganancia de la solución óptima, es decir, la máxima ganancia de la instancia.
Vhp	Valor de la heurística de planos de cortes de Aráoz, Fernández y Meza [1].

Tabla 1: Significado del encabezado de las columnas de las Tablas de las instancias

Instancia	$\|V\|$	$\|E\|$	$\|R \cup Q\|$	$\#comp.R \cup Q$	$\|R\|$	$\#comp.R$	**Vo**	**Vhp**
ALBAIDAA	102	160	99	10	51	17	6266	6247
ALBAIDAB	90	144	88	11	46	23	4372	4372

Tabla 2: Instancias ALBAIDA

Instancia	$\|V\|$	$\|E\|$	$\|R \cup Q\|$	$\#comp.R \cup Q$	$\|R\|$	$\#comp.R$	**Vo**	**Vhp**
P01	11	13	7	4	5	3	3	3
P02	14	33	12	4	7	4	66	66
P03	28	58	25	5	16	5	56	56
P04	17	35	22	3	13	3	45	45
P05	20	35	16	5	8	5	35	35
P06	24	46	20	7	10	6	60	60
P07	23	47	24	3	14	4	89	89
P08	17	40	24	2	15	3	90	90
P09	14	26	14	3	8	3	46	45
P10	12	20	10	4	6	4	41	41
P11	9	14	7	3	5	4	9	9
P12	7	18	5	3	3	2	10	10
P13	7	10	4	3	3	3	5	5
P14	28	79	31	6	18	6	128	128
P15	26	37	19	8	9	8	43	39
P16	31	94	34	7	20	8	113	106
P17	19	44	17	5	8	5	42	39
P18	23	37	16	8	8	5	21	17
P19	33	54	29	7	16	9	90	90
P20	50	98	63	7	38	11	246	246
P21	49	110	67	6	40	9	258	252
P22	50	184	74	6	43	8	474	474
P23	50	158	78	6	48	7	360	360
P24	41	125	55	7	33	8	237	230

Tabla 3: Instancias CRISTOFIDES

Instancia	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert R\cup Q\rvert$	$\#comp.R\cup Q$	$\lvert R\rvert$	$\#comp.R$	**Vo**	**Vhp**
D0	16	32	3	2	2	2	109	109
D1	16	31	6	3	3	3	115	115
D2	16	31	9	4	5	5	274	274
D3	16	32	8	3	5	3	172	172
D4	16	31	8	4	4	4	210	210
D5	16	31	12	4	6	3	313	313
D6	16	32	11	5	7	4	166	166
D7	16	31	12	4	6	4	260	260
D8	16	31	16	4	7	4	457	457
D9	36	72	12	8	6	5	160	160
D10	36	72	10	7	5	5	0	0
D11	36	72	17	11	7	6	398	398
D12	36	72	17	8	7	6	280	280
D13	36	72	22	5	11	6	717	662
D14	36	72	30	9	15	11	810	810
D15	36	72	32	6	15	8	702	702
D16	36	72	34	4	17	9	980	965
D17	36	72	38	6	22	11	1115	1070
D18	64	128	28	15	14	9	515	515
D19	64	128	29	11	14	10	509	509
D20	64	128	27	11	15	10	457	457
D21	64	128	47	10	26	13	1000	976
D22	64	128	47	12	26	17	989	989
D23	64	128	51	9	28	14	968	968
D24	64	128	68	6	38	12	1463	1463
D25	64	128	62	9	34	14	1317	1317
D26	64	128	75	5	40	14	1625	1610
D27	100	200	50	22	28	19	549	501
D28	100	200	55	19	30	17	814	786
D29	100	200	51	19	28	15	555	517
D30	100	200	86	13	44	21	1378	1378
D31	100	200	90	10	47	17	1503	1503
D32	100	200	82	16	46	17	1066	1028
D33	100	200	121	9	67	22	2074	2074
D34	100	200	118	9	62	18	1806	1806
D35	100	200	117	9	62	19	1901	1820

Tabla 4: Instancias DEGREE

Instancia	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert R\cup Q\rvert$	$\#comp.R\cup Q$	$\lvert R\rvert$	$\#comp.R$	**Vo**	**Vhp**
G0	16	24	3	3	2	2	0	0
G1	16	24	5	5	4	4	0	0
G2	16	24	4	4	3	3	0	0
G3	16	24	8	5	6	4	2	2
G4	16	24	7	5	6	5	0	0
G5	16	24	7	3	6	3	4	4
G6	16	24	13	4	9	4	9	9
G7	16	24	8	4	6	4	1	1
G8	16	24	9	4	7	4	4	4
G9	36	60	11	7	8	6	2	2
G10	36	60	13	9	9	7	0	0
G11	36	60	15	9	10	6	4	4
G12	36	60	26	7	19	10	15	15
G13	36	60	23	6	16	8	11	11
G14	36	60	25	7	18	7	14	14
G15	36	60	35	5	24	5	26	26
G16	36	60	30	7	21	7	20	20
G17	36	60	34	5	23	7	24	24
G18	64	112	24	10	17	9	8	8
G19	64	112	27	12	19	14	6	6
G20	64	112	27	14	19	10	9	9
G21	64	112	46	8	33	12	32	32
G22	64	112	47	8	34	15	32	31
G23	64	112	50	11	35	15	33	33
G24	64	112	68	4	47	10	57	57
G25	64	112	61	5	41	11	46	46
G26	64	112	66	6	45	12	57	57
G27	100	180	41	19	29	18	14	14
G28	100	180	49	20	35	19	23	22
G29	100	180	44	20	31	19	14	13
G30	100	180	73	13	51	19	50	50
G31	100	180	77	18	53	21	54	54
G32	100	180	82	11	56	19	57	55
G33	100	180	113	4	76	16	92	92
G34	100	180	107	9	72	17	86	86
G35	100	180	109	6	73	17	88	87

Tabla 5: Instancias GRID

Instancia	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert R\cup Q\rvert$	$\#comp.R\cup Q$	$\lvert R\rvert$	$\#comp.R$	**Vo**	**Vhp**
R0	20	37	3	3	1	1	1742	1742
R1	20	47	4	4	2	2	4253	3853
R2	20	47	4	3	2	2	5638	5638
R3	20	75	7	4	4	4	18453	18453
R4	20	60	6	3	3	2	17316	17316
R5	30	70	7	4	4	3	298	298
R6	30	111	10	4	5	4	12478	12478
R7	30	70	7	5	4	4	9405	9405
R8	30	111	11	5	6	5	14847	14847
R9	30	111	11	6	6	6	17523	17523
R10	40	130	13	8	6	5	17405	17405
R11	40	103	10	6	5	4	7125	7125
R12	40	82	8	5	4	4	1493	1493
R13	40	203	18	9	7	6	32453	30464
R14	40	203	18	7	7	6	30732	30732
R15	50	203	20	8	9	6	27791	27791
R16	50	162	15	12	7	7	10533	10533
R17	50	130	13	6	6	5	4276	4001
R18	50	203	19	7	8	7	28462	28462
R19	50	203	19	8	8	7	26873	26873

Tabla 6: Instancias RANDOM

B. Algoritmos de Mejoras de Soluciones

Los algoritmos de mejora de soluciones tienen como fin recibir una solución factible del PRPP y mejorar el beneficio de la misma por medio del estudio de su estructura. En esta sección se introducen tres algoritmos de mejoras de soluciones para el PRPP, los cuales son Eliminación de ciclos negativos, Sustitución Euleriana y Eliminación Euleriana.

B.1. Eliminación de ciclos negativos

El algoritmo 1 *Eliminación de ciclos negativos* recorre una un solución factible del PRPP en busca de ciclos. Cuando se encuentra un ciclo se determina el beneficio que aporta a la solución. Si el ciclo produce pérdidas a la solución entonces se elimina de la misma. Esto trae como consecuencia un aumento del beneficio de la solución.

B.2. Eliminación y Sustitución Euleriana

La Eliminación y Sustitución Euleriana son procesos de mejoras de soluciones factibles del Problema del Cartero Rural (*Rural Postman Problem*, RPP) y del *Crossing Postman Problem* (XPP). Estos procesos son originalmente llamados como *Euler-deletion* y *Euler-replacement*. La Eliminación y Sustitución Euleriana fueron introducidas por Garfinkel y Webb [8] y han sido utilizados con éxito por Fernández et al. [7] en un algoritmo heurístico para el Problema del Cartero Rural. En este trabajo se presenta dos algoritmos de mejoras de soluciones del PRPP basados en La Eliminación y Sustitución Euleriana.

Para poder explicar en que consiste la Eliminación y Sustitución Euleriana es necesario presentar la definición del Problema del Cartero Rural realizada por Garfinkel y Webb [8].

Algoritmo 1: Eliminación de ciclos negativos

```
Input: Un ciclo 𝒞 solución factible de PRPP
Output: Un ciclo 𝒞 solución factible de PRPP
begin
    foreach lado e1 = (v_{i-1}, v_i) ∈ 𝒞 do
        e_ini ← el siguiente lado adyacente de e1 en 𝒞
        foreach lado e2 = (v_{j-1}, v_j) ∈ 𝒞 comenzando desde e_ini do
            if v_i = v_j then
                Existe un ciclo que llamamos C_i. Sea
                𝒞 = (d = v_1, ..., v_{i-1}, v_i, ..., v_j = v_i, v_{j+1}, ..., v_n = d) la solución
                PRPP actual y sea C_i = (v_i, ..., v_j) el ciclo de 𝒞 encontrado.
                if al evaluar C_i se tiene que es un ciclo que no produce beneficio then
                    // Se elimina el ciclo de la solución actual
                    𝒞 ← (d = v_1, ..., v_{i-1}, v_i, v_{j+1}, ..., v_n = d)
    return 𝒞
```

Dado un grafo no dirigido y no necesariamente conectado $G_R = (V_R, E_R)$, donde cada componente conexa tiene al menos un lado y Sea E_D un conjunto de lados, tal que $E_D = \{e = (i,j) \mid i,j \in V_R \wedge i \neq j\}$. Se tiene que el costo de cada lado $e = (i,j)$ se denota como d_{ij}. El RPP consiste en encontrar un multiconjunto (no todos los lados son necesariamente diferentes) de lados $E_p \subseteq E_D$, de costo total mínimo, tal que el grafo $G_{R\cup P} = (V_R, E_R \cup E_P)$ es euleriano, es decir, el grafo es conectado y cada vértice tiene grado par.

Algunas observaciones. Como $G_{R\cup P}$ es un grafo euleriano, entonces tiene un ciclo euleriano que atraviesa cada uno de los lados del conjunto $E_R \cup E_P$ exactamente una vez. Como E_P es un multiconjunto se tiene que un lado $e \in E_D$, puede encontrase más de un vez en el conjunto E_P. El conjunto E_R corresponde al conjunto de lados requeridos, mientras que el costo d_{ij} de cada lado $e = (i,j) \in E_D$, corresponde al camino de costo mínimo entre los vértices $i, j \in V$, del grafo $G = (V, E)$ de la definición inicialmente dada.

La Eliminación y Sustitución Euleriana para el RPP se definen como sigue.

Dado un grafo que corresponde a una solución del Problema del Cartero Rural, $G_{sol} = (V_R, E_R \cup F)$ donde $F \subseteq E_D$, la Eliminación Euleriana consiste en encontrar dos lados duplicados $e_1 = (v_1, w_1)$ y $e_2 = (v_2, w_2)$ con $v_1 = v_2$ y $w_1 = w_2$, que pertenezcan a F ($e_1, e_2 \in F$), tal que si son eliminados de G_{sol} este sigue siendo un grafo euleriano. Es decir, se tiene que el grafo $G_{sol} = (V_R, E_R \cup F \setminus \{e_1, e_2\})$, es conectado y par. Al eliminar lados duplicados G_{sol}, el ciclo euleriano que se obtiene de ese grafo debe tener un costo menor o igual al grafo con los lados duplicados.

Dado un grafo que corresponde a una solución del Problema del Cartero Rural, $G_{sol} = (V_R, E_R \cup F)$ donde $F \subseteq E_D$, la Sustitución Euleriana consiste en encontrar dos lados adyacentes $e_1 = (i,j)$ y $e_2 = (j,k)$ que pertenezcan a F ($e_1, e_2 \in F$) que puedan ser reemplazados por un lado $e_3 = (j,k)$ en el grafo G_{sol}, el cual sigue siendo un grafo euleriano. Se tiene que el grafo $G_{sol} = (V_R, E_R \cup F \cup \{e_3\} \setminus \{e_1, e_2\})$, es conectado y par. Al eliminar los lados e_1 y e_2 y agregar el lado e_3 a G_{sol}, el ciclo euleriano que se obtiene de ese grafo debe tener un costo menor al grafo G_{sol} original. La justificación de la la Sustitución Euleriana es que siempre se cumple la desigualdad triangular asociada con la métrica euclidiana. Debido a esto se tiene que la suma de los costos de los lados e_1 y e_2 es mayor que el costo e_3, es decir $d_{e_1} + d_{e_2} > d_{e_3}$.

B.2.1. Sustitución Euleriana para el PRPP

En el algoritmo 2 se presenta la Sustitución Euleriana para el PRPP. Dado un ciclo $\mathscr{C}$ solución del PRPP, para cada uno de los lados adyacentes $e_1 = (i, j)$ y $e_2 = (j, k)$ del ciclo, se calcula el camino del costo mínimo entre los vértices i y k. Si al sustituir los lados e_1 y e_2 por ese camino de costo mínimo en la solución $\mathscr{C}$, se tiene que hay un aumento de beneficio de la solución, entonces el cambio se lleva a cabo y se obtiene un nueva solución del PRPP. La función (2) es la función que usa el algoritmo 2 Sustitución Euleriana para calcular los caminos de costo mínimo

$$c(e) = \begin{cases} c_e, & \text{si } e \in \mathscr{C} \\ -\varphi_e, & \text{si } e \in P \\ 0, & \text{de lo contrario (El lado pertenece al conjunto } R \cup Q\text{).} \end{cases} \tag{2}$$

Algoritmo 2: Sustitución Euleriana para el PRPP

Input: Un ciclo $\mathscr{C}$ solución factible de PRPP y un grafo $G = (V, E)$
Output: Un ciclo $\mathscr{C}$ solución factible de PRPP
begin
  foreach *par de lados adyacentes* $e_1 = (i, j)$ *y* $e_2 = (j, k)$ *en* $\mathscr{C}$ **do**
    Calcular el camino de costo mínimo CCM_{ik} entre i y k en el grafo G
    $\mathscr{C}' \leftarrow$ sustituir los lados e_1 y e_2 por el camino CCM_{ik} en $\mathscr{C}$
    if `beneficio`($\mathscr{C}$) $<$ `beneficio`($\mathscr{C}'$) **then**
      $\mathscr{C} \leftarrow \mathscr{C}'$
  return $\mathscr{C}$

B.2.2. Eliminación Euleriana con Eliminación de Componente Conexa

En el algoritmo 3 se muestra la *Eliminación Euleriana con Eliminación de Componente Conexa* para el PRPP. Este algoritmo se compone de dos partes. En la primera parte se efectúa una eliminación euleriana equivalente a la que se presentó para el RPP. Dada una solución factible C del PRPP se crea un grafo con los vértices y lados de C. Este grafo es euleriano, por lo que es conectado y cada uno de sus vértices tiene grado par. Luego para cada par de lados repetidos del grafo, se verifica si al ser eliminados del grafo no lo desconectan, en cuyo caso procede la eliminación si aumenta el beneficio. Del grafo resultante se obtiene un ciclo euleriano que va a ser la nueva solución del PRPP que va a tener un beneficio mayor o igual que el de la solución inicial.

La segunda parte consiste en utilizar la eliminación euleriana para eliminar una componente conexa que aporta pérdidas a la solución. La idea es la siguiente. De la primera parte se obtiene un grafo al que se le aplicó una eliminación euleriana. Se tiene que pueden existir pares de lados repetidos en ese grafo, que al eliminarlos del grafo hacen que este sea desconectado. Específicamente, se tiene que se forman dos componentes conexas. Una de las componentes conexas incluye al depósito. En la Figura 3 se muestra un ejemplo de este caso. Sea el grafo A de la Figura 3 un grafo euleriano que se obtiene de la primera parte. El lado $(2, 5)$, está duplicado. De este grafo euleriano se obtiene el ciclo $d - 1 - 2 - 5 - 4 - 6 - 5 - 2 - 3 - d$ que es una solución del PRPP con beneficio 10. Si a este grafo se eliminan los lados duplicados $(2, 5)$ se obtiene el grafo B de la Figura 3. Este grafo es desconectado y posee dos componentes conexas. Una de las componentes conexas contiene al depósito. Siempre las componentes conexas resultantes tendrán grado par y esto se debe a que se eliminan un par de lados que las unían. Como la componente conexa que contiene al depósito tiene grado par, con sus lados y vértices se puede construir un grafo euleriano. De este grafo euleriano se puede obtener un ciclo euleriano que es una solución del PRPP. Si esta solución tiene un beneficio mayor que el de la mejor solución actual, entonces nos quedamos con esta solución. Esto lo podemos ver en el grafo B de la Figura 3. De la componente conexa que contiene al depósito podemos obtener un grafo euleriano y por consiguiente una nueva solución del PRPP, que en este caso sería el ciclo $d-1-2-3-d$ con beneficio 14. Esta solución con beneficio 14 mejora el beneficio la solución inicial que correspondía al grafo A de la Figura 3, el cual era 10. En consecuencia esta nueva solución pasa a ser la mejor solución actual. Los detalles de este procedimiento en el que se puede eliminar una componente conexa, se muestran algoritmo 3 en la parte 2.

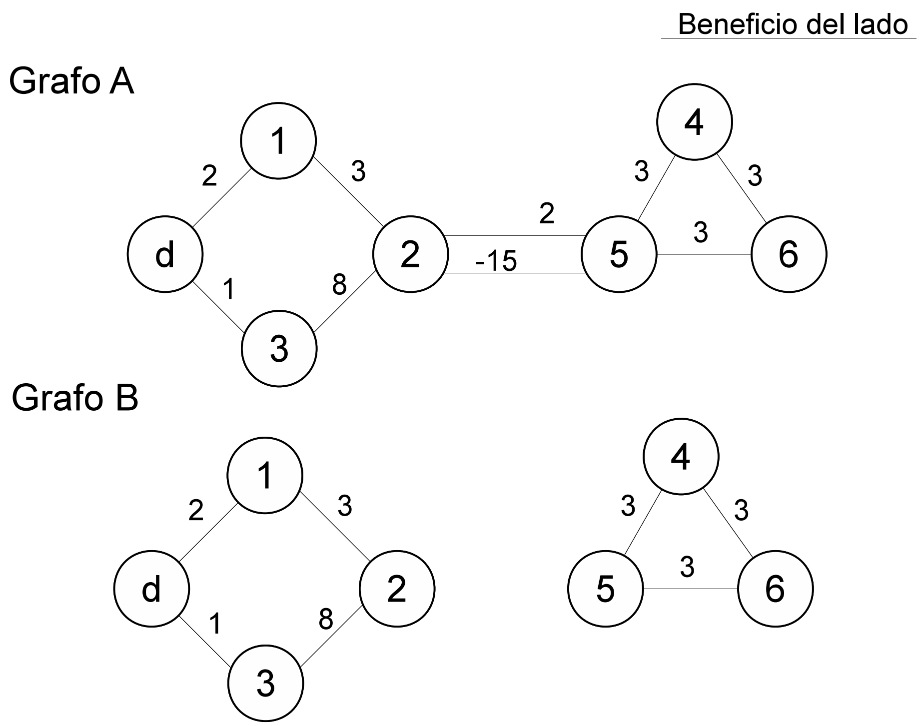

Figura 3: Ejemplo de eliminación de una componente conexa. Del grafo euleriano A se obtiene el ciclo $d-1-2-5-4-6-5-2-3-d$ con beneficio 10. De la componente conexa que contiene al depósito en el grafo B, se puede obtener un ciclo $d-1-2-3-d$ con beneficio 14

Algoritmo 3: Eliminación Euleriana con eliminación de componente conexa

```
Input: Un ciclo 𝒞 solución factible de PRPP
Output: Un ciclo 𝒞 solución factible de PRPP
begin
    // Parte 1: Eliminación Euleriana
    G_sol ← Crear un grafo con los vértices y lados de 𝒞
    foreach par de lados duplicados e_1 = (i, j) y e_2 = (i, j) en G_sol do
        G'_sol ← Eliminar los lados e_1 y e_2 de G_sol
        if G'_sol es un grafo euleriano con mayor beneficio then
            // G'_sol es conexo y todos sus vértices tienen grado par
            G_sol ← G'_sol
            𝒞 ← Obtener un ciclo euleriano de G_sol

    // Parte 2: Eliminación de Componente Conexa
    foreach par de lados duplicados e_1 = (i, j) y e_2 = (i, j) en G_sol do
        G'_sol ← Eliminar los lados e_1 y e_2 de G_sol
        if G'_sol es un grafo desconectado then
            G_cc ← Se crea un grafo con la componente conexa de G'_sol que contiene al
            depósito
            𝒞_cc ← obtener un ciclo euleriano de G_cc
            if beneficio(𝒞_cc) > beneficio(𝒞) then
                𝒞 ← 𝒞_cc
                G_sol ← G_cc
    return 𝒞
```