

# **Búsqueda Tabú para la programación de una estación de carga de vehículos eléctricos**

**Lina Reyes <sup>a</sup>, Luis Tarazona <sup>b</sup>**

<sup>a</sup> Universidad de los Andes, Bogotá, Colombia, l.reyesa@uniandes.edu.co

<sup>b</sup> Universidad de los Andes, Bogotá, Colombia, le.tarazona@uniandes.edu.co

## **Resumen**

Esta investigación se realizó con el objetivo de aplicar la metaheurística Búsqueda Tabú para realizar la programación de una estación de carga de vehículos eléctricos, se consideró dos partes, la primera parte con una función objetivo de minimizar el tiempo promedio de espera, considerando que todos los vehículos ingresan al mismo tiempo y los conectores con la misma capacidad de carga, y la segunda parte un caso real donde los vehículos ingresan a una hora definida, los conectores tienen diferentes tipos de carga y existe consideraciones de costos por picos de hora. Además, se consideró la reprogramación a cierto tiempo (TR). Se concluye con buenos resultados obtenidos en un tiempo computacional por debajo de 10 segundos para todos los casos, lo que demuestra la velocidad para tener un resultado.

## **Palabras Clave**

Estación de carga, Vehículos eléctricos, Búsqueda Tabú, Scheduling, Rescheduling

## 1. Introducción

La Agencia Internacional de Energía (IEA), menciona que en el mundo existe ya cerca de 16 millones de vehículos eléctricos e híbridos enchufables en circulación, los cuales han ingresado al mercado vehicular a competir con los vehículos convencionales por el bajo impacto ambiental que tienen ya que generan hasta 30% menos emisión de gases de efecto invernadero y un consumo de 30 TWh por año de electricidad (Gil, 2022). En Colombia la situación con estos vehículos tiene el mismo escenario, para el año 2021 se vendieron 3008 vehículos entre híbridos enchufables y eléctricos (Restrepo, 2022), mientras que el primer semestre del presente año se ha vendido 1823 vehículos (Avendaño, 2022), lo que llega a superar la cantidad de 8200 de estos tipos de vehículos en el país e inclusive supera la meta pactada por el Registro Único Nacional de Tránsito de 6600 vehículos (Vera, 2022).

Si bien estos vehículos presentan grandes ventajas tanto técnicas como ambientales, las desventajas más importantes son la autonomía por el nivel de batería que presentan estos vehículos, ya que pierden en promedio 2.3% de autonomía por año y dependerá del tipo de batería (Viera, 2003), la disponibilidad de puntos de carga, en el país, teniendo en cuenta que en el país existen 210 puntos de carga, haciendo un total de 491 conectores, por lo que se puede asumir que existe una precaria infraestructura de carga (Restrepo, 2022), y el tiempo de carga que puede ser rápida (10-40 minutos) con un promedio de 50 kWh, semi rápida (1.5 – 3 horas) con un promedio de 7.4 kWh y lenta (5-8 horas) con un promedio de 3.7kWh, aunque existen cargas super y ultra rápida que pueden tener un promedio de 130 – 150 kWh (Creara, 2022).

Estas desventajas conllevan netamente a la limitación existente de puntos o estaciones de carga. Ya que según las condiciones de operación y la demanda existente se debe realizar una adecuada distribución de los vehículos que solicitan carga. Estos son factores importantes para determinar el tiempo que estos vehículos deben esperar para poder ser recargados, además de tomar decisiones de programación de vehículos para carga, que, si bien pueden ser programadas considerando una reserva de recargada, también deben ser atendidas según la hora de llegada de estos vehículos a la estación. Es por esta razón que en esta investigación se realiza la aplicación de un algoritmo de búsqueda tabú para solucionar el problema de la programación de carga de vehículos eléctricos viendo este como una programación de vehículos eléctricos, para eso este trabajo se compone preliminarmente de la definición del problema y formulación matemática, la formulación de la metaheurística, la experimentación, resultados y conclusiones.

## 2. Definición del problema y formulación matemática

### 2.1 Definición del problema

El problema se define de la siguiente forma: una estación de carga que atiende en una jornada de tiempo de trabajo  $T$ , tiene  $M$  conectores para atender  $N$  vehículos eléctricos que requieren una cantidad de carga  $Q$ . Algunos autores han estudiado este problema y han acondicionado algunos factores dentro de la evaluación, como es el caso de Athulya (2020) que evaluó la programación en una estación de carga considerando horas de entrada de los vehículos, penalización por superar el límite máximo de demanda y nivel de kW por hora pico y no pico. Deng, Yongxi, Fengji, & Yunfei (2021) realizaron una investigación para planificar una estación de carga mediante la utilización de sistemas de almacenamiento de energía de batería de segunda vida, Hutson, Kumar, & Corzine (2008) que aplicaron un método para programar el uso de la capacidad de almacenamiento de energía disponible de los vehículos eléctricos híbridos enchufables y vehículos eléctricos. Wang, Jochem, & Fichtner (2020), realizaron un estudio donde programaron la carga de vehículos eléctricos bajo incertidumbre de disponibilidad de vehículo y demanda de carga.

En ese contexto, en esta investigación se planteó el problema de la siguiente forma: se tiene una estación de carga, con una jornada de tiempo de trabajo  $T$ , que tiene  $M$  conectores para atender a  $N$  vehículos que requieren una cantidad de carga  $Q$ , se sabe que los vehículos solicitan carga por lo que son registrados y programados considerando que el primero en el registro es el primero que sale (FIFO), además, que se busca reducir el promedio de tiempo de espera global, se asume que los vehículos ingresan a la misma hora, que los conectores tienen el mismo poder de carga y que no existe alguna penalización por superar el límite máximo de demanda.

Este problema se asemeja a una programación de máquinas en paralelo, donde la función objetivo es minimizar el promedio de tiempo de espera global y como restricción principal es que el primero en la lista es el primero que debe salir de la estación de carga (FIFO). La formulación matemática se adaptó de la propuesta por Mena & Daniel (2015).

### 2.2 Formulación matemática

#### Variables

$e_i$ : tiempo de espera de cada vehículo

$$x_i \begin{cases} 1 & \text{si se realiza el trabajo } i \text{ después del trabajo } i' \\ 0 & \text{si no se realiza el trabajo } i \text{ después del trabajo } i' \end{cases}$$

$p_i$ : tiempo de carga de cada vehículo

$q_i$ : carga requerida por cada vehículo

$t_i$ : tiempo de inicio de carga de cada vehículo

$c_i$ : requerido de carga de cada vehículo

$r_i$ : número de registro de cada vehículo

### Conjuntos

N: conjunto de vehículos

M: conjunto de conectores en la estación de carga

L: Número grande

R: Conjunto de registro

### Parámetros

K: Capacidad de carga de cada conector de la estación

### Función objetivo

$$\min \frac{\sum_{i=1}^N e_i}{N} \quad (1)$$

### Restricciones

$$\min \sum_{i=1}^N x_i \quad \forall i = 1, 2, 3, \dots, N \quad (2)$$

$$\sum_{i=1}^N x_i \leq 1 \quad \forall i = 1, 2, 3, \dots, N \quad (3)$$

$$t_i \geq t_{i'} + p_{i'} - (1 - x_i) \quad \forall i = 1, 2, 3, \dots, N; \forall i' = 1, 2, 3, \dots, N \quad (4)$$

$$c_i = t_i + p_i \quad \forall i = 1, 2, 3, \dots, N \quad (5)$$

$$r_i < r_{i'} \quad \forall i = 1, 2, 3, \dots, R \quad (6)$$

$$c_i \leq t_i \quad (7)$$

### Formulaciones

$$p_i = \frac{q_i}{K} \quad (8)$$

$$e_i = c_i - p_i \quad (9)$$

## 2.3 Representación

La fórmula (1) representa la función objetivo de minimizar el promedio del tiempo de espera de los vehículos, la fórmula (2) es la restricción que garantiza que cada trabajo se realice solo una vez, la fórmula (3) es la restricción que obliga a realizar un trabajo en una sola máquina, la fórmula (4) es la restricción que indica que el tiempo de inicio de un trabajo tiene que ser superior o igual a la finalización de un trabajo que se realizó antes de este en la misma máquina, la fórmula (5) es la restricción que calcula el tiempo de finalización de un trabajo, la fórmula (6) garantiza el cumplimiento del FIFO, la fórmula (7) es la restricción que asegura que el tiempo de inicio de un trabajo sea mayor o igual que el tiempo de finalización de un trabajo precedente, la fórmula (8) calcula el tiempo de carga de cada vehículo y la fórmula (9) calcula el tiempo de espera de cada vehículo.

## 3 Metaheurística

La búsqueda tabú es un método de optimización perteneciente a las técnicas de búsqueda local, este tipo de búsqueda aumenta el rendimiento del método de búsqueda local mediante el uso de estructuras de memoria, una vez que una solución es encontrada, se le marca como “tabú” para que el algoritmo no vuelva a buscarla durante un determinado plazo (tenure). Esta metaheurística fue planteada por Fred Glover en 1986 (Glover, 1986).

El procedimiento metaheurístico de la presente investigación estuvo dividido en 2 partes: La primera parte consistió en asumir que todos los vehículos ( $r_i$ ) ingresan en un tiempo igual a cero ( $h_i = 0$ ) con una carga requerida ( $q_i$ ) y que los conectores tienen una misma capacidad de carga ( $w_i$ ), seguidamente se genera la solución inicial tomando en cuenta el FIFO y encontrar un tiempo de espera promedio ( $e$ ), para posteriormente aplicar la Búsqueda Tabú que mejore el tiempo de espera promedio anterior. En la Figura 1 se muestra el pseudocódigo para esta primera parte.

La segunda parte consistió en un caso realista donde los vehículos tienen una hora de entrada definida ( $h_i$ ), con una carga requerida ( $q_i$ ), los conectores tienen diferente capacidad de carga ( $w_i$ ), los costos de energía ( $C_i$ ) dependen de la hora en la que se esté cargando, existe un precio de venta ( $p$ ) y se puede reprogramar los vehículos a determinado tiempo ( $TR$ ), se vuelve a utilizar FIFO para encontrar un tiempo de espera promedio ( $e$ ) y las utilidades ( $U$ ) de la solución inicial. Se aplica la Búsqueda Tabú aplicando a los vehículos con la misma hora de entrada y así buscar una solución que mejore el tiempo promedio de espera y las utilidades. En un determinado tiempo se realiza la reprogramación en la Búsqueda Tabú, se reprograma la carga de vehículos y se vuelve a calcular el tiempo promedio de espera y las utilidades. En la Figura 2 se muestra el pseudocódigo para esta segunda parte. La parte de

reprogramación se muestra en la Figura 3. Cabe aclarar que la reprogramación se realiza a aquellos vehículos que el tiempo de inicio ( $t_i$ ) es menor al TR y que el tiempo de salida ( $s_i$ ) es mayor al TR.

---

	<b>Paso 1: Solución inicial</b>
1	Input $\rightarrow$ Datos ( $q_i, M, r_i$ )
2	Secuencia = R
3	Aplicar FIFO
4	Output $\rightarrow e$
	<b>Paso 2: Búsqueda Tabú</b>
5	SWAP posiciones en Secuencia
6	Guardar SWAP
7	Actualizar Secuencia
8	Aplicar FIFO
9	Almacenar SWAP como Tabú
10	Si tamaño de Tabú = tenure
11	Eliminar SWAP de Tabú
12	Output $\rightarrow e$
13	Si Solución Paso 2 < Solución Paso 1
14	Actualizar secuencia
15	Output $\rightarrow e$
16	Si no Repetir hasta que Iteraciones = Iteraciones
17	Actualizar secuencia
18	Output $\rightarrow e$

---

**Figura 1 Pseudo código de Búsqueda Tabú para la primera parte**

Fuente: Elaboración propia

---

	<b>Paso 1: Solución inicial</b>
1	Input $\rightarrow$ Datos ( $q_i, M, r_i, h_i, w_i, C_i, h_i$ )
2	Secuencia = R
3	Aplicar FIFO
4	Determinar $c_i$ para cada $r_i$
5	Determinar $C_i$ para cada $r_i$
6	Output $\rightarrow e, U$
	<b>Paso 2: Búsqueda Tabú</b>
7	SWAP posiciones en Secuencia
8	Guardar SWAP
9	Actualizar Secuencia
10	Aplicar FIFO
11	Almacenar SWAP como Tabú
12	Si tamaño de Tabú = tenure
13	Eliminar SWAP de Tabú
14	Output $\rightarrow e, U$
15	Si $e$ Paso 2 < $e$ Paso 1 y $U$ Paso 2 > $U$ Paso 1
16	Actualizar secuencia
17	Output $\rightarrow e, U$
18	Si no Repetir hasta que Iteraciones = Iteraciones
19	Actualizar secuencia
20	Output $\rightarrow e, U$

---

**Figura 2 Pseudo código de Búsqueda Tabú para la segunda parte**

Fuente: Elaboración propia

---

<b>Paso 1: Solución inicial</b>	
1	Input $\rightarrow$ Datos ( $q_i, M, r_i, h_i, w_i, C_i, h_i$ )
2	Secuencia = R
3	Aplicar FIFO
4	Determinar $c_i$ para cada $r_i$
5	Determinar $C_i$ para cada $r_i$
6	Output $\rightarrow e, U$
<b>Paso 2: Reprogramación</b>	
7	Input $\rightarrow$ Datos ( $q_i, M, r_i, h_i, w_i, C_i, h_i, TR$ )
8	Si $h_i < TR$ y $s_i > TR$
9	Seleccionar $r_i$
10	Actualizar $h_i = TR$
11	Actualizar Secuencia
12	
13	SWAP posiciones en Secuencia
14	Guardar SWAP
15	Actualizar Secuencia
16	Aplicar FIFO
17	Almacenar SWAP como Tabú
18	Si tamaño de Tabú = tenure
19	Eliminar SWAP de Tabú
20	Output $\rightarrow e, U$
21	Si $e$ Paso 2 $< e$ Paso 1 y $U$ Paso 2 $> U$ Paso 1
22	Actualizar secuencia
23	Output $\rightarrow e, U$
24	Si no Repetir hasta que Iteraciones = Iteraciones
25	Actualizar secuencia
26	Output $\rightarrow e, U$

---

**Figura 3 Pseudo código para la reprogramación**

Fuente: Elaboración propia

## 4 Experimentación

### 4.1 Supuestos de la investigación y parámetros de la estación de carga

Para la primera parte se consideró los siguientes supuestos:

- Todos los vehículos ingresan a una hora de entrada igual a cero.
- No existe un límite de jornada laboral.
- Los conectores de la estación tienen la misma capacidad de carga.
- No existe una hora de salida definida
- Todos los conectores están disponibles en el tiempo igual a cero.
- No se puede cargar otro vehículo hasta terminar un vehículo anterior.

Para la segunda parte se consideró los siguientes supuestos:

- Cada vehículo tiene una hora de entrada definida.
- No existe un límite de jornada laboral. No existe una hora de salida definida

- Todos los conectores están disponibles en el tiempo igual a cero.
- Los conectores tienen diferente capacidad de carga
- El costo de carga dependerá de la hora en la que se esté realizando la carga
- Se puede cargar otro vehículo si un vehículo anterior no se ha terminado de cargar (solo cuando se reprogreme).

Los parámetros de la estación de carga han sido obtenidos de (Athulya, 2020) y son los siguientes:

- Capacidad de carga de los vehículos (kW/h): 12, 15 19
- Precio por hora (\$/kW): 4, 5, 6
- Precio venta (\$/kW): 6,5
- Horas pico (considerando hora inicial 0)
  - Horas bajo pico: 0 a 4 horas
  - Horas medio pico: 4 a 8 horas
  - Horas alto pico: 8 a más horas

## 4.2 Instancias

Se han generado 10 instancias que contienen las horas de entrada (aplicada a la segunda parte) y la carga requerida, no se encontraron instancias para la primera parte por lo que no se pudo hacer una discusión respectiva, mientras que para la segunda parte se han utilizado también las instancias propuestas por (Athulya, 2020) (Instancia 11) y por (Sedighizadeh, 2019) (Instancia 12).

## 4.3 Procedimiento

Se realizó el siguiente procedimiento experimental considerando las siguientes variables:

**Tabla 1. Diseño experimental**

<b>N° de conectores</b>	<b>Iteraciones</b>
3	1000
4 (solo segunda parte)	5000
6	



## 5 Resultados

En la Tabla 2 se muestra los resultados para 3 conectores y 1000/5000 iteraciones para la primera parte, se puede evidenciar variación positiva y significativa para 9 de las 10 instancias, donde el menor valor de variación fue para la instancia 2 con 0% donde no hubo una mejora en el tiempo de espera y un mayor valor de variación fue para la instancia 3 donde hubo una mejora en el tiempo de espera de 25%, todo esto para 1000 iteraciones. Para 5000 iteraciones no se ha mostrado mejoras significativas. Los tiempos computacionales estuvieron por debajo de 2 segundos para 1000 iteraciones y por debajo de 10 segundos para 5000 iteraciones, lo que demuestra que el algoritmo es veloz para mostrar un resultado.

**Tabla 2. Resultados para 3 conectores y 1000/5000 iteraciones primera parte**

				N° iteraciones		1000	N° iteraciones		5000
				Tenure	5		Tenure	10	
N° conectores	3	Inicial		TabuSearch		% variación	TabuSearch		% variación
Inst	N° Vehículos	FO inicial	T <sub>comp</sub>	FO	T <sub>comp</sub>		FO	T <sub>comp</sub>	
1	7	2,6	0,01	2,07	1,27	20,38%	2,07	6,46	20,38%
2	15	4,46	0,08	4,46	1,31	0,00%	4,46	6,99	0,00%
3	20	9,2	0	6,9	1,48	25,00%	6,9	6,93	25,00%
4	25	11,65	0,01	9,38	1,59	19,48%	9,38	7,23	19,48%
5	31	14,6	0	12,22	1,68	16,30%	12,22	7,94	16,30%
6	36	20,16	0,01	16,93	1,66	16,02%	16,93	8,09	16,02%
7	41	21,55	0	17,46	1,7	18,98%	17,46	8,57	18,98%
8	20	10,04	0,01	9,44	1,51	5,98%	9,44	7,06	5,98%
9	46	23,85	0	20,21	1,8	15,26%	20,21	8,42	15,26%
10	51	26,32	0	22,61	1,83	14,10%	22,61	8,73	14,10%

En la Tabla 3 se muestra los resultados para 6 conectores y 1000/5000 iteraciones para la primera parte, se puede evidenciar variación positiva y significativa para 8 de las 10 instancias, donde el menor valor de variación fue para la instancia 1 e instancia 2 con 0% donde no hubo una mejora en el tiempo de espera y un mayor valor de variación fue para la instancia 7 donde hubo una mejora en el tiempo de espera de 21,02%, todo esto para 1000 iteraciones. Para 5000 iteraciones no se ha mostrado mejoras significativas. Los tiempos computacionales estuvieron por debajo de 2 segundos para 1000 iteraciones y por debajo de 10 segundos para 5000 iteraciones, lo que demuestra que el algoritmo es veloz para mostrar un resultado.

**Tabla 3. Resultados para 6 conectores y 1000/5000 iteraciones primera parte**

				N° iteraciones		1000	N° iteraciones		5000
				Tenure	5		Tenure	10	
N° conectores	6	Inicial		TabuSearch		% variación	TabuSearch		% variación
Inst	N° Vehículos	FO inicial	T <sub>comp</sub>	FO	T <sub>comp</sub>		FO	T <sub>comp</sub>	
1	7	0,34	0,01	0,34	1,26	0,00%	0,34	6,44	0,00%
2	15	1,55	0,02	1,55	1,32	0,00%	1,55	6,47	0,00%
3	20	3,29	0,01	2,62	1,46	20,36%	2,62	7,24	20,36%
4	25	4,6	0,01	3,84	1,55	16,52%	3,84	7,4	16,52%
5	31	6,08	0	5,22	1,72	14,14%	5,22	7,76	14,14%
6	36	9,09	0,03	7,46	1,92	17,93%	7,46	8,13	17,93%
7	41	9,85	0,01	7,78	1,88	21,02%	7,78	8,42	21,02%
8	20	3,96	0,01	3,73	1,45	5,81%	3,73	7,33	5,81%
9	46	11,03	0,01	9,11	1,78	17,41%	9,11	8,66	17,41%
10	51	12,2	0,01	10,34	1,89	15,25%	10,34	8,82	15,25%

Debido a los resultados obtenidos para la primera parte, se optó por no realizar más pruebas para 5000 iteraciones, ya que no mejora la solución a 1000 iteraciones.

En la Tabla 4 se muestra los resultados para 4 conectores y 1000 iteraciones para la segunda parte, se puede evidenciar variación positiva y significativa para 7 de las 10 instancias creadas, donde el menor valor obtenido fue de 0% y el mayor valor fue de 41% para el promedio de tiempo de espera (Instancia 3) y 8,45% para la utilidad (Instancia 9). Respecto a la instancia propuesta por (Athulya, 2020) (Instancia 11), no se muestra mayores mejoras debido que la instancia contempla solamente 6 vehículos y 6 conectores además de aplicar FIFO. Respecto a la instancia propuesta por (Sedighzadeh, 2019) (Instancia 12), no se muestran mayores mejoras, en esta instancia contempla 43 vehículos y 5 conectores, además de también aplica FIFO. Además, tener en cuenta que los supuestos han sido específicos para cada investigación incluyendo esta.

Respecto al tiempo computacional, los tiempos computacionales estuvieron por debajo de 10 segundos y en algunas instancias se ha evidenciado que para la Búsqueda Tabú se obtuvo resultados en menor tiempo comparando con el tiempo utilizado para la solución inicial, esto demuestra que el algoritmo es veloz para mostrar un resultado.

**Tabla 4. Resultados para 4 conectores y 1000 iteraciones segunda parte**

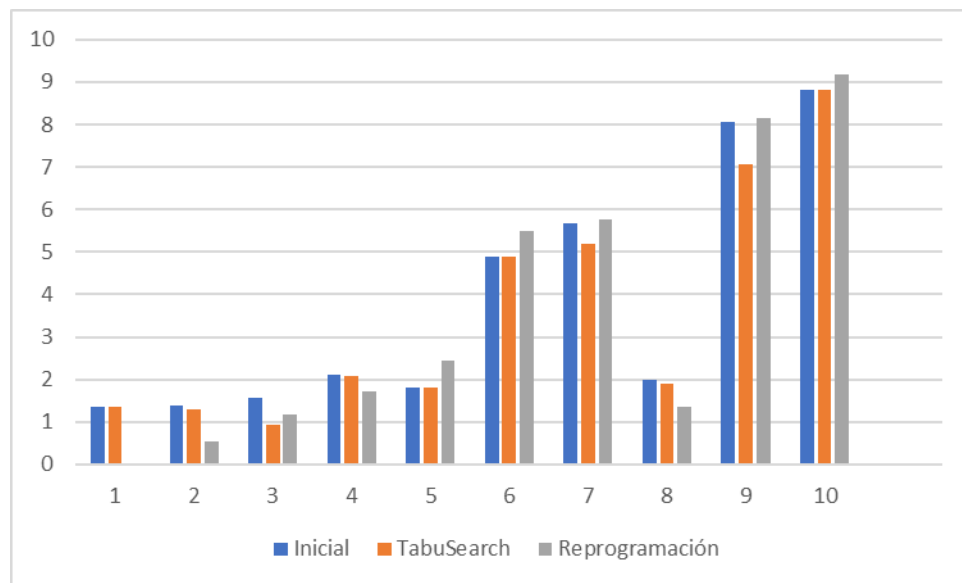
					N° iteraciones		1000		
					Tenure		5		
N° conectores	4	Inicial			TabuSearch			% var <sub>espera</sub>	% var <sub>utilidad</sub>
Instancia	N° Vehículos	Espera <sub>inicial</sub>	Utilidad <sub>inicial</sub>	T <sub>comp</sub>	Espera	Utilidad	T <sub>comp</sub>		
1	7	1,36	477,5	0	1,36	477,5	1,87	0,00%	0,00%
2	15	1,4	651	0	1,29	701	2,39	-7,86%	7,68%
3	20	1,56	1050	0	0,92	1063	3,24	-41,03%	1,24%
4	25	2,1	1131	0	2,07	1155	3,45	-1,43%	2,12%
5	31	1,81	1212,5	0	1,8	1236,5	4,18	-0,55%	1,98%
6	36	4,9	1525	0	4,9	1525	4,38	0,00%	0,00%
7	41	5,68	1496,5	0	5,2	1580,5	4,62	-8,45%	5,61%
8	20	1,99	1221,5	0	1,91	1305	3,02	-4,02%	6,84%
9	46	8,06	1431,5	0	7,05	1552,5	5,42	-12,53%	8,45%
10	51	8,82	1511,5	0	8,82	1511,5	5,62	0,00%	0,00%
11	6	1,21	218	0	1,21	218	1,84	0,00%	0,00%
12	43	10,13	201,6	0	10,13	201,6	5,2	0,00%	0,00%

En la Tabla 5 se muestra los resultados para 6 conectores y 1000 iteraciones para la segunda parte, se puede evidenciar variación positiva y significativa para 5 de las 10 instancias creadas, donde el menor valor obtenido fue de 0% y el mayor valor fue de 20% para el promedio de tiempo de espera (Instancia 7) y 11,61% para la utilidad (Instancia 5). Respecto a la instancia propuesta por (Athulya, 2020) (Instancia 11), no se muestra mayores mejoras debido que la instancia contempla solamente 6 vehículos y 6 conectores además de aplicar FIFO. Lo mismo para la instancia propuesta por (Sedighizadeh, 2019) (Instancia 12) Respecto al tiempo computacional, los tiempos computacionales estuvieron por debajo de 10 segundos y en algunas instancias se ha evidenciado que para la Búsqueda Tabú se obtuvo resultados en menor tiempo comparando con el tiempo utilizado para la solución inicial, esto demuestra que el algoritmo es veloz para mostrar un resultado.

**Tabla 5. Resultados para 6 conectores y 1000 iteraciones segunda parte**

					N° iteraciones			1000	
					Tenure		5		
N° conectores	6	Inicial			TabuSearch			% var <sub>espera</sub>	% var <sub>utilidad</sub>
Instancia	N° Vehículos	Espera <sub>inicial</sub>	Utilidad <sub>inicial</sub>	T <sub>comp</sub>	Espera	Utilidad	T <sub>comp</sub>		
1	7	0,36	554,5	0	0,36	554,5	2,06	0,00%	0,00%
2	15	1,34	806	0	1,34	806	2,8	0,00%	0,00%
3	20	2,01	1052	0	2	1083	3,47	-0,50%	2,95%
4	25	2,03	1133	0,01	2,03	1133	4,2	0,00%	0,00%
5	31	2,01	1214,5	0	1,77	1355,5	4,91	-11,94%	11,61%
6	40	1,05	1953	0	0,89	1976	5,35	-15,24%	1,18%
7	41	1,1	1983,5	0,01	0,88	2063	6,02	-20,00%	4,01%
8	20	1,5	1316,5	0	1,5	1316,5	3,77	0,00%	0,00%
9	46	2,91	1895,5	0,01	2,43	1970	6,84	-16,49%	3,93%
10	51	3,13	1975,5	0	3,13	1975,5	7,26	0,00%	0,00%
11	6	0	218	0	0	218	1,89	0,00%	0,00%
12	43	8,8	206,4	0,02	8,8	206,4	6,25	0,00%	0,00%

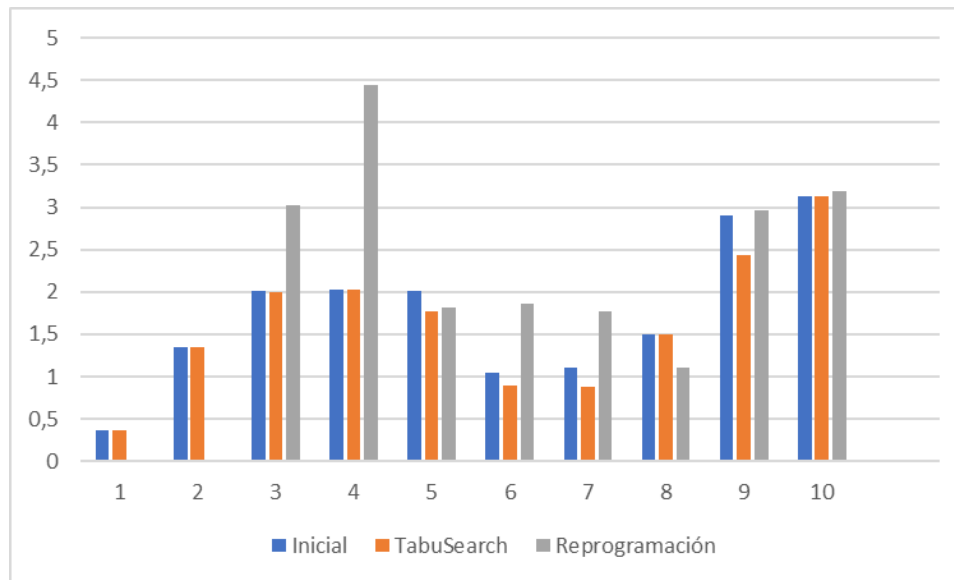
Respecto a la reprogramación, el tiempo de reprogramación fue 7, en la Figura 4 se muestra una comparación del tiempo promedio de espera para la prueba de 4 conectores y 1000 iteraciones de la segunda parte, se puede ver que para la reprogramación para las instancias 5, 6, 7, 9 y 10 es mayor considerando que la cantidad de vehículos es menor, por lo que puede asumir que el reprogramar puede afectar significativamente al tiempo promedio de espera.



**Figura 4 Comparación de tiempo de espera promedio 4 conectores**

Fuente: Elaboración propia

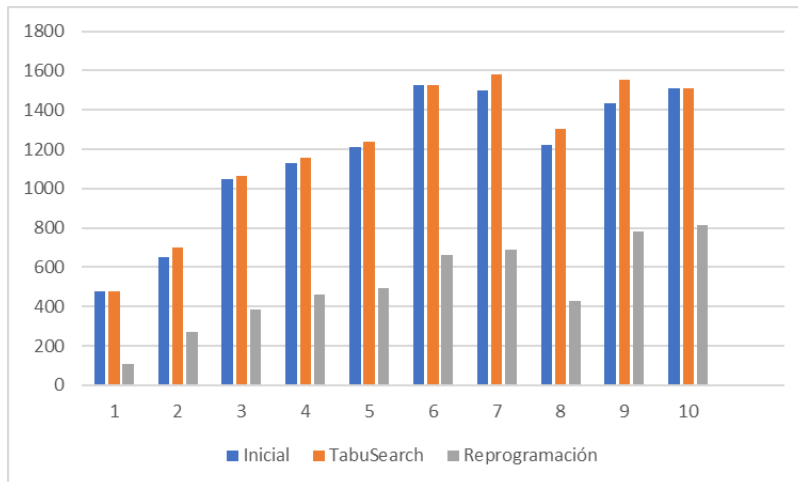
En la Figura 5 se muestra una comparación del tiempo promedio de espera para la prueba de 6 conectores y 1000 iteraciones de la segunda parte, se puede ver que para la reprogramación, la instancia 4 entrega un resultado muy por encima de los resultados de la solución inicial y de la Búsqueda Tabú, en el resto de instancias se evidencia que el tiempo promedio de espera es igual o superior a los resultados de la solución inicial y de la Búsqueda Tabú, por lo que puede asumir que el reprogramar puede afectar significativamente al tiempo promedio de espera.



**Figura 5 Comparación de tiempo de espera promedio 6 conectores**

Fuente: Elaboración propia

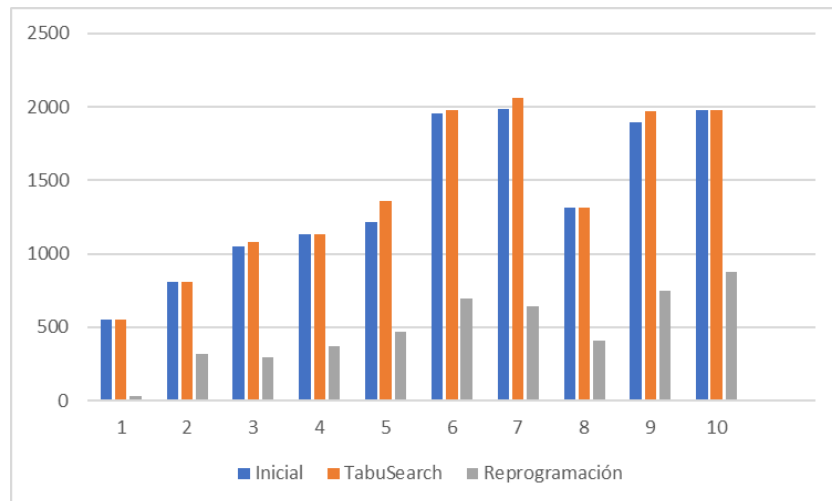
En la Figura 6 se muestra una comparación de las utilidades espera para la prueba de 4 conectores y 1000 iteraciones de la segunda parte, se puede ver un comportamiento interesante para las instancias 6, 7, 9 y 10 donde la utilidad en la reprogramación es la mitad de la solución inicial y de la Búsqueda Tabú, esto se puede asumir como una mejora en la utilidad para la estación de carga. En el resto de instancias se puede evidenciar una utilidad baja, que no significa que no se ajuste a la reprogramación, sino más bien, esto dependerá de la consideración de horas pico, ya que el costo de energía varía dependiendo esas horas y por tanto si la reprogramación se hace en horas de alto pico las utilidades serán menores.



**Figura 6 Comparación de utilidades 4 conectores**

Fuente: Elaboración propia

En la Figura 7 se muestra una comparación de las utilidades espera para la prueba de 6 conectores y 1000 iteraciones de la segunda parte, se puede ver un comportamiento general de la utilidad, donde está es menor ya que se debe a la consideración de horas pico, ya que el costo de energía varía dependiendo esas horas y por tanto si la reprogramación se hace en horas de alto pico las utilidades serán menores, es claro que se presenta el nerviosismo del sistema, y por tanto se altera significativamente las programaciones iniciales.



**Figura 7 Comparación de utilidades 6 conectores**

Fuente: Elaboración propia

## **6 Conclusiones y recomendaciones.**

Se aplicó la metaheurística de Búsqueda Tabú para solucionar el problema de programación de estación de carga de vehículos eléctricos, considerando como función multiobjetivo minimizar el tiempo promedio de espera y maximizar las utilidades de la estación de carga.

Se tomó en cuenta la reprogramación de vehículos a un tiempo de programación (TR) de 7, consiguiendo resultados que dan a entender lo importante de considerar las horas picos para que los costos de energía, y que el nerviosismo del sistema puede afectar significativamente a la utilidad y al tiempo promedio de espera.

Se recomienda ajustar la reprogramación y hacer un análisis de efectividad para reducir el nerviosismo del sistema.

## 7 Referencias

- Athulya, M. S. (2020). *Electric vehicle recharge Scheduling in a Shopping Mall Charging Station*. National Power Systems Conference.
- Avendaño, G. (08 de Agosto de 2022). *Colombia, primera en venta de carros eléctricos en latinamérica*. Obtenido de <https://www.motor.com.co/industria/Colombia-primera-en-venta-de-carros-electricos-en-Latinoamerica-20220808-0008.html>
- Creara. (30 de Noviembre de 2022). *Tipos de recarga de vehículos eléctricos*. Obtenido de <https://www.creara.es/post/tipos-recarga-coche-electrico/>
- Deng, Y., Yongxi, Z., Fengji, L., & Yunfei, M. (2021). *Operational Planning of centralized charging stations utilizing second-life battery energy storage systems*. IEEE Transactions on sustainable energy.
- Gil, M. (22 de Julio de 2022). *Carros eléctricos generan hasta 30% menos emisiones de gases de efecto invernadero*. Obtenido de <https://www.larepublica.co/responsabilidad-social/carros-electricos-generan-hasta-30-menos-emisiones-de-gases-de-efecto-invernadero-3408082#:~:text=Seg%C3%BAn%20la%20Agencia%20Europea%20de,efecto%20invernadero%20respecto%20a%20los>
- Glover, F. (1986). *Future paths for integer programming and link to artificial intelligence*. Computers and operations researchs.
- Hutson, C., Kumar, G., & Corzine, K. (2008). *Intelligent scheduling of hybrid and electric vehicle storage capacity in a parking lot for profit maximization in grid power transactions*. Atlanta: Ieee Energy 2030.
- Melián, B., & Glover, F. (2006). *Introducción a la búsqueda tabú*.
- Mena, & Daniel. (2015). *Programación de trabajos en máquinas en paralelo con relación de precedencia*. Sevilla: Universidad de Sevilla.
- Restrepo, Ó. (30 de Noviembre de 2022). *Estos son los países donde más se vendieron autos eléctricos en 2021*. Obtenido de <https://www.elcarrocolombiano.com/industria/autos-electricos-paises-que-mas-vendieron-en-2021/>



- Sedighzadeh, M. (2019). *A daytime optimal stochastic energy management for EV commercial parking lots by using approximate dynamic programming and hybrid big bang big crunch algorithm*. Elsevier.
- Vera, J. (30 de Noviembre de 2022). *¿Cuántas electrolinerías hay en Colombia?: así avanza la movilidad eléctrica en el país*. Obtenido de <https://www.elcarrocolombiano.com/industria/electrolineras-colombia-asi-avanza-la-movilidad-electrica-en-el-pais/#:~:text=Actualmente%20hay%20m%C3%A1s%20de%208.200,Nacional%20de%20Tr%C3%A1nsito%20para%202022>.
- Viera, J. (2003). *Carga rápida de baterías de Ni-Cd y Ni-MH de media y gran capacidad. Análisis y comparación de nuevos métodos*. Gijón: Universidad de Oviedo.
- Wang, Z., Jochem, P., & Fichtner, W. (2020). *A scenario-based stochastic optimization model for charging scheduling of electric vehicles under uncertainties of vehicle availability and charging demand*. Journal of cleaner production.