
Algorithm 1 Algoritmo Metaheurístico

```
1:  $V$  /* Conjunto de nodos optimizados */
2:  $S \leftarrow \emptyset$  /* Conjunto de soluciones para cada carro */
3:  $P$  /* Conjunto de pedidos con sus direcciones */ /* Se ordenan los pedidos
   por distancia */
4:  $CM$  /* capacidad máxima de cada carro */
5:  $M$  /* Matriz de tiempos */
6:  $Vol$  /* Matriz de volúmenes */
7:  $T_{max}$  /* Tiempo máximo de cada carro */
8:  $P \leftarrow ordenar(P)$ 
9:  $i \leftarrow 0$ 
10: while  $P \neq \emptyset$  do
11:    $S[i] \leftarrow \emptyset$ 
12:    $S[i] \leftarrow S[i] \cup P[0]$ 
13:    $P \leftarrow P - P[0]$ 
14:    $j \leftarrow 0$ 
15:   /* Se agregan los nodos que no superen la capacidad máxima y el
      tiempo máximo */
16:   while  $j < |P|$  and  $\sum_{k \in S[i]} Vol[k] + Vol[P[j]] \leq CM$  and
       $\sum_{k \in S[i]} M[k][P[j]] + M[P[j]][S[i][0]] \leq T_{max}$  do
17:     /* Se busca un nodo entre el nodo actual y el nodo siguiente para
        pasar por ahí */
18:     if  $\exists k \in S[i] \mid M[k][P[j]] + M[P[j]][S[i][0]] \leq M[k][S[i][0]]$  then
19:        $S[i] \leftarrow S[i] \cup k$ 
20:        $P \leftarrow P - k$ 
21:     end if
22:     /* Se agrega el nodo actual */
23:      $S[i] \leftarrow S[i] \cup P[j]$ 
24:      $P \leftarrow P - P[j]$ 
25:      $j \leftarrow j + 1$ 
26:   end while
27:    $i \leftarrow i + 1$ 
28: end while
```
