

Modelos Matemáticos para o Problema de Compartilhamento de Veículos Elétricos

Welverton Rodrigues da Silva

Instituto de Computação, Universidade Estadual de Campinas
Av. Albert Einstein 1251, Campinas, SP, 13083-852, Brasil
welverton.silva@ic.unicamp.br

Rafael Crivellari Saliba Schouery

Instituto de Computação, Universidade Estadual de Campinas
Av. Albert Einstein 1251, Campinas, SP, 13083-852, Brasil
rafael@ic.unicamp.br

Fábio Luiz Usberti

Instituto de Computação, Universidade Estadual de Campinas
Av. Albert Einstein 1251, Campinas, SP, 13083-852, Brasil
fusberti@ic.unicamp.br

RESUMO

Neste trabalho apresentamos o problema de compartilhamento de veículos elétricos, um novo problema de otimização combinatória motivado por sistemas de aluguel de carros compartilhados, conhecido por *car-sharing*. O objetivo desse problema é atribuir aos veículos da frota existente uma sequência de demandas (solicitações de aluguel) a serem atendidas que maximize o tempo total do aluguel dos veículos. Para a solução do problema, este trabalho propõe dois modelos de programação linear inteira mista baseados em uma formulação do problema como uma rede de espaço-tempo. Ambos os modelos propostos foram avaliados por meio de experimentos computacionais realizados em instâncias com até 50 clientes, geradas de forma aleatória.

PALAVRAS CHAVE. Compartilhamento de Veículos, Carros Elétricos, Programação Linear Inteira.

Tópicos: Otimização combinatória, Programação matemática.

ABSTRACT

In this paper, we present the electric vehicle sharing problem, a new combinatorial optimization problem motivated by car-sharing systems. The goal of this problem is to assign a sequence of driving demands to be fulfilled by the existing fleet of vehicles to maximize the total rental time of the vehicles. To solve the problem, this work proposes two mixed integer linear programming models, based on a space-time network formulation. Both models were evaluated through computational experiments carried out on randomly generated instances with up to 50 clients.

KEYWORDS. Car-sharing, Electric Car, Integer Linear Programming.

Paper topics: Combinatorial optimization, Mathematical programming.

1. Introdução

O compartilhamento de carros, ou *car-sharing*, é uma das tendências mundiais de economia compartilhada, com base no conceito de que clientes alugam carros (sem intervenção de um agente da empresa) por curtos períodos de tempo, para atenderem às suas necessidades de locomoção em grandes centros urbanos [Ding et al., 2019]. Muitos dos sistemas de carros compartilhados existentes são baseados em estações fixas, i.e., possuem vagas de estacionamento dedicadas, e também podem permitir a reserva antecipada [Ströhle et al., 2019]. Ao fim da jornada, o cliente devolve o carro alugado à sua estação de origem, ou em outra estação pré-determinada, para que seja disponibilizado a outro cliente.

Em Böhmová et al. [2016] é estudado um problema de otimização combinatória motivado por sistemas de *car-sharing* com trajeto ponto a ponto (modalidade conhecida como *one-way*), denominado *transfers for commuting*. Em sua definição, duas estações denominadas A e B são consideradas e todo cliente apresenta um par de demandas (solicitações de aluguel), em direções opostas, uma no sentido de A para B e outra no sentido de B para A, mas não necessariamente nessa ordem. Cada demanda também especifica os instantes de tempo de início e de fim do período de locação. Um conjunto de veículos indistinguíveis é considerado, inicialmente distribuídos entre as duas estações, e o número de vagas de estacionamento em cada estação é suficiente para cobrir todos os veículos do sistema.

Esse problema visa atribuir os veículos aos clientes, ao longo do tempo, para atender o maior número possível de clientes, de modo que se uma das demandas de um cliente é atendida, então a outra demanda também é atendida. Especificamente, ou o cliente viaja entre as duas estações no formato de ida e volta ou não usa o sistema. O problema pode ser enunciado, em suma, como: dado um conjunto C de clientes, encontrar um subconjunto $C' \subseteq C$ de clientes a serem atendidos pela frota de veículos existentes, tal que a cardinalidade de C' seja máxima.

O trabalho de Böhmová et al. [2016] demonstra que esse problema pertence à classe de problemas NP-difíceis, inclusive APX-difíceis, mesmo quando existe apenas um único veículo disponível e todos os períodos de locação têm a mesma duração.

Luo et al. [2018a,b,c] consideram variantes do *transfers for commuting* em que todos os clientes apresentam apenas uma única demanda e, posteriormente, em Luo et al. [2020], consideram a sua versão original (i.e., em que cada cliente apresenta duas demandas em direções opostas). Nestes trabalhos, os autores investigam algoritmos *online*, abordagem com tomada de decisão imediata ou de curto prazo para responder às novas demandas que surgem. As suas contribuições são demonstrações de limites superiores e inferiores para a razão de competitividade dos algoritmos.

Silva e Schouery [2020] estudaram formas de resolução exata e heurística para o *transfers for commuting*. Em seu trabalho, os autores propõem um modelo de programação linear inteira mista, com base na modelagem do problema em rede de fluxos, conforme descrito por Böhmová et al. [2016]; e a adaptação das meta-heurísticas GRASP, VNS e Busca Tabu para obtenção de soluções para instâncias geradas de forma aleatória, utilizadas em experimentos computacionais. Além disso, os autores também propõem um procedimento de pré-processamento para reduzir o tamanho das instâncias do problema a partir da sua modelagem em redes de fluxos.

Neste trabalho apresentamos uma variante para o *transfers for commuting*, referenciada por EVSP, em que cada cliente apresenta múltiplas reservas (e.g., ida e volta entre uma área residencial e um shopping center próximo) de uso de veículos elétricos ao longo de um dado horizonte de planejamento. O objetivo do EVSP é maximizar o tempo total do aluguel dos veículos, sujeito a que todo cliente atendido tenha todas as suas demandas atendidas. Para essa variante propomos dois modelos de programação linear inteira mista e apresentamos alguns resultados computacionais

para instâncias de pequeno porte, geradas de forma aleatória.

O restante deste artigo está organizado como segue. A Seção 2 apresenta a definição do problema. A Seção 3 descreve os modelos de programação linear inteira mista propostos. A Seção 4 apresenta e discute os resultados obtidos com os experimentos computacionais. Por fim, a Seção 5 contém as nossas considerações finais sobre este estudo.

2. Definição do Problema

Por tratar-se de uma variante do problema de Böhmová et al. [2016], as seguintes suposições são consideradas. Cada estação possui um número limitado de vagas de estacionamento e um número possivelmente menor de carregadores de baterias dos veículos. Além disso, as estações são livres de operadores (funcionários), logo cada veículo elétrico (EV, de *electric vehicle*) só pode ser conectado a um carregador por meio do cliente que o esteja devolvendo em uma estação e, neste caso, deve permanecer conectado até ser utilizado pelo próximo cliente.

Consideramos que a frota de EVs do sistema é homogênea e que a velocidade de locomoção é constante. Desta forma, consideramos que o consumo da bateria é linear em função do tempo, onde denotamos por $\lambda \in \mathbb{Q}_{>0}$ a potência elétrica, em watts. Inicialmente todos os EVs estão com a bateria totalmente carregada, sendo L a capacidade da bateria. Inclusive, em cada estação, os EVs estão inicialmente distribuídos de tal modo que o número de vagas de estacionamento vazias e com carregadores seja a maior possível.

Também consideramos que todas as estações possuem a mesma tecnologia de recarga e, desta maneira, qualquer carregador permite a recarga completa de uma bateria dentro de um mesmo período de tempo. Ademais, consideramos que o processo de recarga das baterias trata-se de uma função linear no tempo, onde denotamos por $\mu \in \mathbb{Q}_{>0}$ a potência elétrica fornecida pelo carregador, em watts.

Formalmente, o **problema de compartilhamento de veículos elétricos**, ou *electric vehicle sharing problem* (EVSP) em inglês, é definido como segue.

Seja $T = \{t_1, \dots, t_m\}$ um conjunto de instantes de tempo (em minutos) dentro de um horizonte de planejamento (e.g., um dia inteiro ou uma parte do dia em que há um número de solicitações acima da capacidade do sistema). Sejam C e S , respectivamente, os conjuntos de clientes e estações.

Para cada cliente $c \in C$ existe um conjunto D_c de demandas. Uma demanda é descrita por uma quintupla da forma $(\vec{s}, t_i, \vec{s}, t_j, \varepsilon) \in (S \times T)^2 \times \mathbb{Q}_{>0}$, onde: $\vec{s}$ representa a estação de origem¹; t_i indica o início do período de locação; $\vec{s}$ representa a estação de destino; t_j indica o fim do período de locação; e ε é o tempo de condução de um EV durante o período de locação (definido a partir da distância percorrida e da velocidade de locomoção), tal que $\varepsilon \leq t_j - t_i$ e $t_i < t_j$. Seja $D = \bigcup_{c \in C} D_c$ o conjunto de todas as demandas.

Para cada estação $s \in S$ existe um conjunto V_s de EVs, que indica quais são os veículos inicialmente disponíveis em s para locação. Cada estação s possui uma capacidade total (i.e., vagas de estacionamento), denotada por C_s , e um número de carregadores, denotado por R_s , tal que $R_s \leq C_s$ e $|V_s| \leq C_s$. Seja $V = \bigcup_{s \in S} V_s$ o conjunto de todos os EVs.

O objetivo do EVSP é atribuir aos EVs da frota existente uma sequência de demandas a serem atendidas, associada a uma rota válida, que maximize o tempo total do aluguel dos veículos, de modo que se uma das demandas de um cliente é atendida, então todas as suas demandas são atendidas.

¹O símbolo de sobrescrito “ $\vec{\cdot}$ ” é utilizado no decorrer de o todo texto, para transmitir a noção de saída (e.g., estação, vaga de estacionamento, etc.); enquanto que, de forma inversa, o sobrescrito “ $\cdot$ ” é utilizado para transmitir a noção de entrada.

Dizemos que uma sequência de demandas a serem atendidas fazem de parte uma *rota válida* para $v \in V$ quando satisfaz as seguintes restrições. Para cada demanda $d = (\vec{s}, t_i, \vec{s}, t_j, \varepsilon)$ dessa sequência, d é atendida somente se v estiver na estação $\vec{s}$ no instante t_i ; se a energia da bateria de v for maior ou igual a $\lambda\varepsilon$ (consumo total de energia da bateria durante a locação); e se existir uma vaga de estacionamento em $\vec{s}$, que deve estar vazia no instante t_j , para receber v . Uma rota deve determinar se e quando v será conectado a um carregador para a recarga da bateria (somente se a vaga de estacionamento selecionada para v conter um carregador). Cabe ressaltar que se uma demanda é atendida, então v estará à disposição dos clientes a partir do instante t_j na estação $\vec{s}$.

3. Formulações

Para indicar a distribuição inicial dos EVs em cada estação, conforme a consideração apontada na definição do problema, considere o seguinte. Seja $Z_s \subseteq V_s$, para cada $s \in S$, o conjunto de EVs da estação s inicialmente distribuídos em vagas de estacionamento com carregadores. Para simplificar a notação utilizada, considere também o seguinte. Seja $T_0 = T \cup \{t_0\}$ uma extensão dos instantes de tempo, onde t_0 é usado apenas para indicar a configuração inicial do sistema, e seja $I = \{0, \dots, m\}$ o conjunto ordenado de índices que representam os instantes de tempo.

Omitiremos da notação de uma demanda o tempo de condução ε sempre quando não for necessário para o contexto, denotando apenas como uma quádrupla $(\vec{s}, t_i, \vec{s}, t_j) \in D$.

3.1. Modelo de fluxos em redes homogêneas

Esta primeira formulação para o EVSP consiste em considerar o problema em uma rede de espaço-tempo, com o eixo das ordenadas representando o espaço e o eixo das abscissas representando o tempo (Figura 1), e de múltiplos fluxos; em que cada fluxo representa uma rota. A rede é definida como segue.

O conjunto de nós da rede é definido por uma coleção de conjuntos N_s de nós, para cada $s \in S$. Em cada N_s existe um nó s_i para cada $i \in I$. Ou seja, existe um nó s_i que corresponde a estação s no instante $t_i \in T_0$. De modo a não sobrecarregar a notação utilizada, caso não haja ambiguidade, escrevemos apenas s_i sem indicar o conjunto ao qual pertence.

O conjunto de arcos é definido por uma coleção de conjuntos E_c e E_s de arcos, para cada $c \in C$ e cada $s \in S$. Em cada E_c existe um arco e de $\vec{s}_i$ para $\vec{s}_j$, onde denotamos o consumo de energia da bateria por $E_e^\lambda = \lambda\varepsilon$, para cada demanda $(\vec{s}, t_i, \vec{s}, t_j, \varepsilon) \in D_c$. Enquanto que em cada E_s existe um arco e de s_{i-1} para s_i , onde denotamos a energia de recarga da bateria por $E_e^\mu = \mu(t_i - t_{i-1})$, para todo $i \in I \setminus \{0\}$. Se um arco e pertence a algum E_c , então dizemos que e é do tipo *demanda*; caso contrário (e pertence a algum E_s), dizemos que e é do tipo *conexão*.

Considere também as definições e notações, apresentadas a seguir, para os arcos que entram e saem de um nó da rede. Para cada $s_i \in N_s$, $s \in S$, definimos $\vec{E}(s_i) = \{(s_j, s_i) \in E_c, \forall c \in C\}$ e, analogamente, $\vec{E}(s_i) = \{(s_i, s_j) \in E_c, \forall c \in C\}$ como sendo, respectivamente, os conjuntos de todos os arcos do tipo demanda que entram s_i e de todos os arcos do tipo demanda que saem de s_i . Em particular aos arcos do tipo conexão, por serem conjuntos unitários, denotamos apenas por $\vec{e}(s_i) = (s_{i-1}, s_i) \in E_s$ e por $\vec{e}(s_i) = (s_i, s_{i+1}) \in E_s$ os arcos de entrada e de saída, respectivamente.

Para formularmos o problema a partir desta rede, definimos três conjuntos de variáveis binárias de fluxo para cada $v \in V$ — que equivale a considerar $|V|$ redes idênticas. Um conjunto é definido pelas variáveis de decisão x_e^v , para cada arco e do tipo demanda, de modo que assume valor 1 se e somente se a demanda associada a e é atendida por v . Os outros dois conjuntos são definidos respectivamente pelas variáveis de decisão y_e^v e z_e^v , para cada arco e do tipo conexão. Cada par de variáveis y_e^v e z_e^v está sujeito a restrição de que apenas uma delas assumo o valor 1, quando v

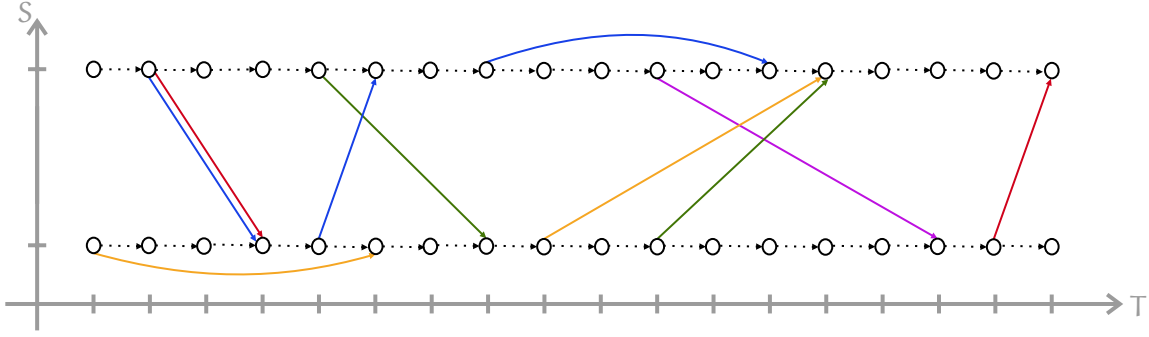


Figura 1: Exemplo de instância do EVSP modelada como uma rede de espaço-tempo. Neste exemplo existem duas estações, de modo que os nós de cada estação são agrupados horizontalmente. Os arcos do tipo conexão são indicados por arcos de linha tracejada; enquanto que arcos do tipo demanda são indicados por arcos de linha sólida. Arcos de mesma cor correspondem às demandas de um mesmo cliente.

está estacionado na estação associada a e durante seu intervalo de tempo. Em especial, z_e^v indica também que v está conectado a um carregador.

Para indicar a energia residual da bateria de $v \in V$ a cada instante de tempo, definimos um conjunto de variáveis contínuas ℓ_i^v , para todo $i \in I$; e para indicar quais clientes são atendidos, definimos um conjunto de variáveis binárias w_c , para cada $c \in C$, onde w_c assume valor 1 se e somente se o cliente c é atendido através de um conjunto de rotas viáveis para os EVs. Desta forma, a função objetivo a ser maximizada, é descrita por um somatório em função das variáveis w_c .

Com isto, o EVSP é formulado como o problema de programação linear inteira mista apresentado a seguir.

(EVSP1)

$$\max \sum_{c \in C} \left(\sum_{(\vec{s}, t_i, \vec{s}, t_j) \in D_c} t_j - t_i \right) w_c \quad (1)$$

s.t.

$$\sum_{v \in V} x_e^v \leq 1 \quad \forall c \in C, \forall e \in E_c \quad (2)$$

$$\sum_{v \in V} \sum_{e \in E_c} x_e^v = |D_c| w_c \quad \forall c \in C \quad (3)$$

$$\sum_{v \in V} \sum_{e \in \tilde{E}(s_i)} x_e^v + \sum_{v \in V} \left(y_{\vec{e}(s_i)}^v + z_{\vec{e}(s_i)}^v \right) \leq C_s \quad \forall s \in S, \forall i \in I : \exists e \in \tilde{E}(s_i) \quad (4)$$

$$z_{\vec{e}(s_i)}^v \leq R_s - \sum_{v' \in V \setminus \{v\}} z_{\vec{e}(s_i)}^{v'} \quad \forall s \in S, \forall v \in V, \forall i \in I : \exists e \in \tilde{E}(s_i) \quad (5)$$

$$\sum_{v \in V} y_{\vec{e}(s_i)}^v \leq C_s - R_s \quad \forall s \in S, \forall i \in I : \exists e \in \tilde{E}(s_i) \quad (6)$$

$$\sum_{v \in V} z_{\vec{e}(s_i)}^v \leq R_s \quad \forall s \in S, \forall i \in I : \exists e \in \tilde{E}(s_i) \quad (7)$$

$$\ell_i^v \leq \ell_{i-1}^v + \sum_{s \in S} \left(E_{\vec{e}(s_i)}^\mu z_{\vec{e}(s_i)}^v - \sum_{e \in \tilde{E}(s_i)} E_e^\lambda x_e^v \right) \quad \forall v \in V, \forall i \in I \setminus \{0\} \quad (8)$$

$$\ell_i^v \leq L \quad \forall v \in V, \forall i \in I \setminus \{0\} \quad (9)$$

$$\ell_i^v \geq E_e^\lambda x_e^v \quad \forall v \in V, \forall e = (s_i, s_j) \in E_c, \forall c \in C \quad (10)$$

$$\ell_0^v = L \quad \forall v \in V \quad (11)$$

$$y_{\vec{e}(s_i)}^v \leq y_{\vec{e}(s_i)}^v + \sum_{e \in \vec{E}(s_i)} x_e^v \quad \forall v \in V, \forall s \in S, \forall i \in I \setminus \{0, m\} \quad (12)$$

$$z_{\vec{e}(s_i)}^v \leq z_{\vec{e}(s_i)}^v + \sum_{e \in \vec{E}(s_i)} x_e^v \quad \forall v \in V, \forall s \in S, \forall i \in I \setminus \{0, m\} \quad (13)$$

$$y_{\vec{e}(s_i)}^v + z_{\vec{e}(s_i)}^v + \sum_{e \in \vec{E}(s_i)} x_e^v = y_{\vec{e}(s_i)}^v + z_{\vec{e}(s_i)}^v + \sum_{e \in \vec{E}(s_i)} x_e^v \quad \forall v \in V, \forall s \in S, \forall i \in I \setminus \{0, m\} \quad (14)$$

$$y_{\vec{e}(s_0)}^v = 1, z_{\vec{e}(s_0)}^v = 0 \quad \forall s \in S, \forall v \in V_s \setminus Z_s \quad (15)$$

$$y_{\vec{e}(s_0)}^v = 0, z_{\vec{e}(s_0)}^v = 1 \quad \forall s \in S, \forall v \in Z_s \quad (16)$$

$$y_{\vec{e}(s_0)}^v = 0, z_{\vec{e}(s_0)}^v = 0 \quad \forall s \in S, \forall v \in V \setminus V_s \quad (17)$$

$$\ell_i^v \in \mathbb{R}_{\geq 0} \quad \forall v \in V, \forall i \in I \quad (18)$$

$$x_e^v \in \{0, 1\} \quad \forall v \in V, \forall e \in \bigcup_{c \in C} E_c \quad (19)$$

$$y_e^v \in \{0, 1\}, z_e^v \in \{0, 1\} \quad \forall v \in V, \forall e \in \bigcup_{s \in S} E_s \quad (20)$$

$$w_c \in \{0, 1\} \quad \forall c \in C \quad (21)$$

A função objetivo (1) maximiza o tempo total do aluguel dos veículos. As restrições definidas pelas inequações (2) e (3) desempenham papel quanto ao atendimento das demandas de um cliente. As inequações (4) garantem que a capacidade total da estação não seja excedida e também impõem a condição de que para entrar em uma estação deve existir previamente alguma vaga de estacionamento vazia. Em especial, as inequações (5) impõem estritamente que as vagas de estacionamento com carregadores estejam vazias antes de serem ocupadas. As inequações (6) e (7) limitam a quantidade de EVs em vagas de estacionamentos, respectivamente, sem e com carregadores.

Já as restrições de (8) a (11) são referentes à conservação de energia das baterias. As inequações (12) e (13) garantem que cada EV permaneça estacionado em uma vaga de estacionamento (e recarregando) até ser utilizado por um cliente. As equações (14) asseguram a determinação de rotas para os EVs (i.e., restrições de conservação de fluxo através dos nós). As equações (15) a (17) fixam uma configuração inicial do sistema, de modo que represente a distribuição dos EVs em cada estação. Por fim, as restrições de (18) a (21) definem o domínio das variáveis de decisão.

Um pré-processamento pode ser aplicado a fim de reduzir significativamente o número de variáveis, resultando também na redução do número de restrições. A princípio, duas operações podem ser utilizadas: a remoção de nós e a junção de arcos. Basicamente, para todo nó $s_i \in N_s$, para $s \in S$, tal que existam somente os arcos do tipo conexão $\vec{e}(s_i)$ e $\vec{e}(s_i)$, ou seja, o seu grau de saída e de entrada são ambos iguais a 1, podemos remover s_i e fazer a junção de $\vec{e}(s_i)$ e $\vec{e}(s_i)$. A remoção de s_i consiste em sua retirada e de seus arcos incidentes da rede; enquanto que a junção de $\vec{e}(s_i) = (s_{i-1}, s_i)$ e $\vec{e}(s_i) = (s_i, s_{i+1})$ consiste na inclusão de um novo arco (s_{i-1}, s_{i+1}) à rede.

Uma outra melhoria pode ser obtida com a fixação de variáveis, a partir da observação de que, para cada $v \in V$, podem existir demandas para as quais conseguimos assegurar que v nunca irá atendê-las em alguma rota. Neste sentido, as variáveis associadas a estas demandas podem ser fixadas em zero ou, em outras palavras, podem ser removidas do modelo.

A formulação apresentada a seguir originou-se a partir da combinação dessas ideias.

3.2. Modelo de fluxos em redes heterogêneas

Uma maneira para se obter modelos menores (no sentido de reduzir o número de variáveis) consiste em considerar uma rede para cada EV. Para este fim, é preciso determinar inicialmente os conjuntos de demandas alcançáveis, para todo $v \in V$. Dizemos que uma demanda $d = (\vec{s}, t_i, \vec{s}, t_j)$ de D é *alcançável* por v se, e somente se, existir alguma rota válida para v em que a demanda d é atendida.

Da mesma forma como fizemos para a primeira formulação, apresentamos a seguir apenas o necessário para a modelagem da rede e fazemos algumas considerações acerca da notação utilizada, visando não sobrecarregá-la. Com isto posto, seguimos com a descrição da notação adotada na construção da rede de fluxo para cada $v \in V$.

Seja $D_c^v \subseteq D_c$, para cada $c \in C$, o conjunto de todas as demandas de c alcançáveis por $v \in V$. O conjunto D_c^v pode ser facilmente obtido por meio de um algoritmo de exploração de caminhos em grafos, lembrando que é preciso considerar as possíveis recargas da bateria. Também seja $I_s^v = \{i \in I : \exists (s, i) \in \Pi\} \cup \{0\}$ o conjunto de índices, para $s \in S$, onde Π é a união de todos os conjuntos $\{(\vec{s}, i), (\vec{s}, j)\}$, para $(\vec{s}, t_i, \vec{s}, t_j) \in \bigcup_{c \in C} D_c^v$; e seja $I^v = \bigcup_{s \in S} I_s^v$ o conjunto de todos os índices.

Para cada $v \in V$, o conjunto de nós de sua rede é definido por uma coleção de conjuntos N_s^v de nós, para cada $s \in S$; enquanto que o conjunto de arcos é definido por uma coleção de conjuntos E_c^v e E_s^v de arcos, para cada $c \in C$ e cada $s \in S$. Em cada N_s^v existe um nó s_i para cada $i \in I_s^v \cup \{m\}$. Novamente, por simplicidade, escrevemos apenas s_i sem explicitar o conjunto a qual pertence. Em cada E_c^v existe um arco e do tipo demanda de $\vec{s}_i$ para $\vec{s}_j$ para cada $(\vec{s}, t_i, \vec{s}, t_j) \in D_c^v$ e, por fim, em cada E_s^v existe um arco e do tipo conexão de $s_{i'}$ para s_i , para $i \in I_s^v \setminus \{0\} \cup \{m\}$ e $i' = \max\{i' \in I_s^v : i' < i\}$.

O índice m é usado para assegurar a criação de nós sorvedouros. Diferente da formulação anterior, estamos reduzindo a quantidade de nós, o que implica na redução de arcos. Se deixarmos de considerar tais nós como sorvedouros, somado ao fato das redes serem heterogêneas, então não seria possível atestar a quantidade de EVs estacionados em uma estação até o instante t_m . Dito de outra forma, determinar a quantidade total de EVs em uma estação s no instante t_i pode incluir um arco do tipo conexão (s_j, s_m) , tal que $j < i \leq m$.

Como já apresentamos na subseção anterior, considere as notações estabelecidas para a energia de consumo e de recarga da bateria quando associadas aos arcos de uma rede, bem como as notações para os arcos que entram e saem a partir de um nó, de modo que já incorporam a modelagem apresentada acima. Considere também o conjunto $\tilde{I}_s = \bigcup_{v \in V} \{i \in I_s^v : \exists e \in \tilde{E}(s_i)\}$ de índices com entrada de EVs na estação s , para cada $s \in S$.

Prosseguimos agora para a definição dos conjuntos de variáveis de decisão. Aqui, estamos dando um novo significado para as variáveis associadas as vagas de estacionamento e ao estado do EV estacionado, como carregando ou não.

Para cada $v \in V$, definimos dois conjuntos de variáveis binárias de fluxo. Um conjunto é definido pelas variáveis de decisão x_e^v , para cada arco e do tipo demanda de E_c^v , para $c \in C$, de modo que assume valor 1 se e somente se a demanda associada a e é atendida por v . O segundo é definido pelas variáveis de decisão y_e^v , para cada arco e do tipo conexão de E_s^v , para $s \in S$, onde

assume valor 1 se e somente se v está estacionado na estação associada a e durante seu intervalo de tempo. Ademais, definimos um outro conjunto de variáveis binárias z_e^v , para cada arco e do tipo conexão de E_s^v , para $s \in S$, para indicar que v deve ser conectado a um carregador, tal que a inequação $z_e^v \leq y_e^v$ é válida.

Para indicar a energia residual da bateria de $v \in V$ a cada instante de tempo, mais uma vez definimos um conjunto de variáveis contínuas ℓ_i^v , para todo $i \in I^v$; e para indicar a quantidade de clientes atendidos, mantemos a definição de um conjunto de variáveis binárias w_c , para cada $c \in C$.

Usando esta notação, o EVSP é reformulado como o seguinte problema de programação linear inteira mista a seguir.

(EVSP2)

$$\max \sum_{c \in C} \left(\sum_{(\vec{s}, t_i, \vec{s}, t_j) \in D_c} t_j - t_i \right) w_c \quad (22)$$

s.t.

$$\sum_{v \in V: \exists e = (\vec{s}_i, \vec{s}_j) \in E_c^v} x_e^v \leq 1 \quad \forall c \in C, \forall (\vec{s}, t_i, \vec{s}, t_j) \in D_c \quad (23)$$

$$\sum_{v \in V} \sum_{e \in E_c^v} x_e^v = |D_c| w_c \quad \forall c \in C \quad (24)$$

$$\sum_{v \in V} \sum_{e \in \tilde{E}(s_i)} x_e^v + \sum_{v \in V: \exists (s_j, s_k) \in E_s^v, j < i \leq k} y_{\vec{e}(s_k)}^v \leq C_s \quad \forall s \in S, \forall i \in \tilde{I}_s \quad (25)$$

$$z_{\vec{e}(s_j)}^v \leq R_s - \sum_{v' \in V \setminus \{v\}: \exists (s_j, s_k) \in E_s^{v'}, j < i \leq k} z_{\vec{e}(s_k)}^{v'} \quad \forall s \in S, \forall i \in \tilde{I}_s, \forall v \in V: \exists (s_j, s_k) \in E_s^v \text{ e } j \leq i < k \quad (26)$$

$$\sum_{v \in V: \exists (s_j, s_k) \in E_s^v, j \leq i < k} (y_{\vec{e}(s_j)}^v - z_{\vec{e}(s_j)}^v) \leq C_s - R_s \quad \forall s \in S, \forall i \in \tilde{I}_s \quad (27)$$

$$\sum_{v \in V: \exists (s_j, s_k) \in E_s^v, j \leq i < k} z_{\vec{e}(s_j)}^v \leq R_s \quad \forall s \in S, \forall i \in \tilde{I}_s \quad (28)$$

$$\ell_i^v \leq \ell_{i'}^v + \sum_{s \in S} \left(E_{\vec{e}(s_i)}^\mu z_{\vec{e}(s_i)}^v - \sum_{e \in \tilde{E}(s_i)} E_e^\lambda x_e^v \right) \quad \forall v \in V, \forall i \in I^v \setminus \{0\}, i' = \max\{i' \in I^v: i' < i\} \quad (29)$$

$$\ell_i^v \leq L \quad \forall v \in V, \forall i \in I^v \setminus \{0\} \quad (30)$$

$$\ell_i^v \geq E_e^\lambda x_e^v \quad \forall v \in V, \forall c \in C, \forall e = (s_i, s_j) \in E_c^v \quad (31)$$

$$\ell_0^v = L \quad \forall v \in V \quad (32)$$

$$2z_{\vec{e}(s_i)}^v \leq y_{\vec{e}(s_i)}^v + z_{\vec{e}(s_i)}^v + \sum_{e \in \tilde{E}(s_i)} x_e^v \quad \forall v \in V, \forall s \in S, \forall i \in I^v \setminus \{0, m\} \quad (33)$$

$$z_{\vec{e}(s_i)}^v + \sum_{e \in \tilde{E}(s_i)} x_e^v \geq z_{\vec{e}(s_i)}^v \quad \forall v \in V, \forall s \in S, \forall i \in I^v \setminus \{0, m\} \quad (34)$$

$$y_{\vec{e}(s_i)}^v + \sum_{e \in \tilde{E}(s_i)} x_e^v = y_{\vec{e}(s_i)}^v + \sum_{e \in \tilde{E}(s_i)} x_e^v \quad \forall v \in V, \forall s \in S, \forall i \in I^v \setminus \{0, m\} \quad (35)$$

$$\sum_{v \in V_s} y_{\vec{e}(s_0)}^v = |V_s| \quad \forall s \in S \quad (36)$$

$$\sum_{v \in V_s} z_{\bar{e}(s_0)}^v = |Z_s| \quad \forall s \in S \quad (37)$$

$$\ell_i^v \in \mathbb{R}_{\geq 0} \quad \forall v \in V, \forall i \in I^v \quad (38)$$

$$x_e^v \in \{0, 1\} \quad \forall v \in V, \forall e \in \bigcup_{c \in C} E_c^v \quad (39)$$

$$y_e^v \in \{0, 1\}, z_e^v \in \{0, 1\} \quad \forall v \in V, \forall e \in \bigcup_{s \in S} E_s^v \quad (40)$$

$$w_c \in \{0, 1\} \quad \forall c \in C \quad (41)$$

A função objetivo (22) e as restrições de (23) a (32) são análogas as da primeira formulação. As restrições dadas pelas inequações (33) e (34) asseguram que cada EV permaneça estacionado e conectado a um carregador, caso esteja carregando a sua bateria, em uma mesma vaga até ser utilizado por um cliente. Em particular, as inequações (33) também asseguram que cada EV seja conectado a um carregador somente se estiver estacionado, por se tratar da combinação das inequações:

$$z_{\bar{e}(s_i)}^v \leq y_{\bar{e}(s_i)}^v, \quad \forall s \in S, \forall v \in V, \forall i \in I^v \setminus \{0\} \quad (42)$$

$$z_{\bar{e}(s_i)}^v \leq z_{\bar{e}(s_i)}^v + \sum_{e \in \bar{E}(s_i)} x_e^v, \quad \forall s \in S, \forall v \in V, \forall i \in I^v \setminus \{0, m\} \quad (43)$$

Note que, devido às inequações (42) e (43) possuírem conjuntos de índices diferentes, as inequações (33) acabam abrindo mão de soluções factíveis nas quais os EVs podem ser conectados a carregadores após a sua última demanda ser atendida. Para que isto não ocorra, as inequações $z_{\bar{e}(s_m)}^v \leq y_{\bar{e}(s_m)}^v$, para cada $s \in S$ e cada $v \in V$, devem fazer parte do modelo (no caso da nossa implementação, estas inequações estão presentes). Isto não compromete o problema em si, visto que o EV, após a sua última demanda ser atendida, não precisa ser conectado ao carregador da vaga estacionado.

Finalmente, para concluir essa segunda formulação, as equações (35) impõem a conservação de fluxo e as equações (36) e (37) forçam a configuração inicial dos EVs em vagas de estacionamento. Os domínios das variáveis de decisão são definidos pelas restrições de (38) a (41) (análogas às restrições da primeira formulação).

4. Resultados Computacionais e Discussão

Todos os testes computacionais a seguir foram realizados em um computador com processador Intel Core i7 com 4 núcleos de 1.80 GHz cada e com 16 GB de RAM, sob sistema operacional Ubuntu 20.04 LTS. Os modelos foram implementados utilizando-se a linguagem C++, usando a biblioteca do resolvidor de programação linear inteira Gurobi Optimizer versão 9.0.3, compilado com a *flag* de otimização -O3 usando GCC versão 7. Utilizamos um limite de tempo de 3 horas de computação para cada execução para cada instância e desativamos o paralelismo do resolvidor.

Para este trabalho utilizamos um conjunto de teste com 50 instâncias de pequeno porte, dividido em 5 grupos de 10 instâncias cada, contendo respectivamente 10, 20, ..., 50 clientes, onde cada cliente possui no máximo quatro demandas. Além disso, cada instância contém exatamente 3 estações, onde cada estação dispõe de 10 a 20 vagas de estacionamento e uma quantidade de carregadores variando entre 15% a 50% das vagas de estacionamento da estação. Inicialmente, cada estação dispõe de uma quantidade de EVs entre 30% a 75% da sua capacidade total. Todas as

instâncias foram geradas de forma aleatória, a partir de uma adaptação do procedimento de geração de instâncias descrita no trabalho de Xu e Meng [2019].

O desempenho de cada um dos modelos propostos para o EVSP pode ser observado no gráfico da Figura 2. Este gráfico indica a quantidade de instâncias resolvidas na otimalidade em relação ao tempo de computação. A partir do gráfico podemos observar que (EVSP2) é o modelo que apresentou o melhor desempenho, tanto no tempo de computação como no número total de instâncias resolvidas.

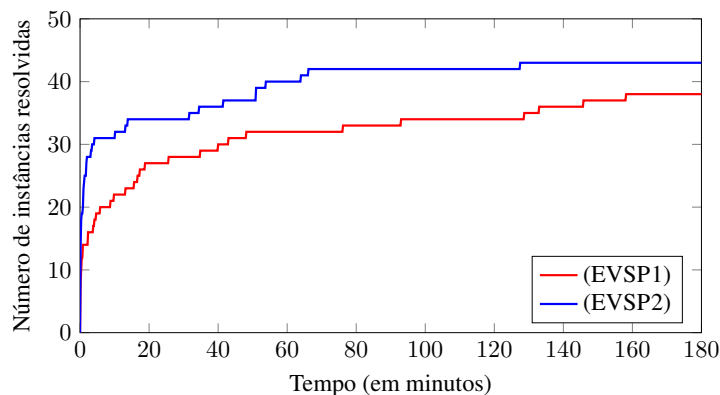


Figura 2: Gráfico de desempenho dos modelos (EVSP1) e (EVSP2) para encontrar uma solução ótima dentro do mesmo limite de tempo.

Os resultados dos experimentos computacionais são reportados nas Tabelas 1 e 2. Em ambas as tabelas, as colunas são organizadas em segmentos para cada grupo de instâncias. Cada segmento reporta os valores do limite inferior (LB), do limite superior (UB) e do tempo computacional (em segundos) — LB é o valor da melhor solução viável obtida e UB é o valor obtido pela relaxação linear (maior limitante superior dentre todos os nós ativos). Os valores LB em negrito correspondem ao valor ótimo conhecido. Cada linha corresponde a uma instância, onde a primeira linha está associada à primeira instância do grupo, e assim por diante.

Tabela 1: Resultados obtidos com o modelo (EVSP1) para cada um dos 5 grupos de instâncias.

Instâncias 1-10			Instâncias 11-20			Instâncias 21-30			Instâncias 31-40			Instâncias 41-50		
LB	UB	Tempo (s)	LB	UB	Tempo (s)	LB	UB	Tempo (s)	LB	UB	Tempo (s)	LB	UB	Tempo (s)
1228	1228	5,37	1730	1730	130,43	2964	2964	780,40	4047	4161	10800,03	5510	10800,00	
989	989	1,33	2186	2186	132,91	2959	2959	996,76	4075	4075	7715,34	4332	4369	10800,01
824	824	2,10	2384	2384	342,64	3739	3739	8749,42	2045	2045	1030,57	3853	10800,03	
872	872	6,00	1434	1434	16,51	3788	3788	4562,75	891	3923	10800,01	5097	10800,00	
755	755	8,43	1761	1761	26,40	2771	2771	2086,62	3225	3225	2575,42	5086	10800,00	
1008	1008	21,75	1975	1975	227,07	2930	2930	1539,11	3902	3957	10800,01	5433	10800,00	
913	913	3,51	1839	1839	42,03	3293	3293	2395,30	2142	2142	521,08	2794	2794	939,17
1253	1253	11,51	1785	1785	276,57	2705	2705	584,64	4177	4177	9482,21	4805	10800,01	
685	685	1,43	1532	1532	41,90	3115	3405	10800,00	2661	2661	5577,28	5394	10800,01	
905	905	3,48	1870	1870	1120,29	1442	1442	240,46	2953	2953	2887,16	4135	4135	7973,84

Na Tabela 1 podemos observar uma grande dificuldade para encontrar uma solução ótima para as instâncias do grupo 41-50, a ponto de não encontrar uma solução viável. Isto decorre do fato de que para o modelo (EVSP1), o tempo de computação necessário para resolver a relaxação linear em cada nó da árvore de enumeração é alto. Em algumas instâncias com 50 clientes, o tempo de computação da relaxação linear no nó raiz é de quase uma hora. A partir da Tabela 2 podemos observar que, embora o modelo (EVSP2) não tenha conseguido encontrar uma solução ótima para todas as instâncias, esse modelo obteve alguma solução viável para as instâncias com 50 clientes, exceto para uma instância.

Tabela 2: Resultados obtidos com o modelo (EVSP2) para cada um dos 5 grupos de instâncias.

Instâncias 1-10			Instâncias 11-20			Instâncias 21-30			Instâncias 31-40			Instâncias 41-50		
LB	UB	Tempo (s)	LB	UB	Tempo (s)	LB	UB	Tempo (s)	LB	UB	Tempo (s)	LB	UB	Tempo (s)
1228	1228	0,48	1730	1730	10,92	2964	2964	104,52	4161	4161	823,19	1175	5510	10800,00
989	989	0,40	2186	2186	6,86	2959	2959	117,82	4075	4075	3054,84	4369	4369	7640,36
824	824	0,22	2384	2384	14,62	3739	3739	2060,39	2045	2045	181,03	3853	3853	2480,22
872	872	0,35	1434	1434	4,20	3788	3788	1893,79	1725	3923	10800,06	4677	4940	10800,00
755	755	1,04	1761	1761	4,27	2771	2771	249,20	3225	3225	609,09		5086	10800,00
1008	1008	1,21	1975	1975	22,97	2930	2930	58,79	3957	3957	3833,16	3234	5433	10800,00
913	913	0,64	1839	1839	8,88	3293	3293	74,87	2142	2142	60,03	2794	2794	204,14
1253	1253	1,20	1785	1785	19,02	2705	2705	59,93	4177	4177	3966,81	4805	4805	3226,39
685	685	0,45	1532	1532	15,47	3004	3294	10800,01	2661	2661	103,88	5342	5394	10800,00
905	905	0,51	1870	1870	45,98	1442	1442	52,63	2953	2953	785,03	4135	4135	3057,82

É possível notar a partir dos valores das médias de cada grupo de instâncias, apresentados na Tabela 3, que um dos fatores que contribuem para o modelo (EVSP2) apresentar melhor desempenho está relacionado ao número de variáveis binárias e de restrições. Outro fator que pode estar contribuindo para o modelo (EVSP2) apresentar melhor desempenho é que o valor da sua relaxação linear é melhor ou igual que o valor da relaxação linear associado ao modelo (EVSP1) para este conjunto de instâncias. Em comparação com o modelo (EVSP1), o segundo modelo apresentou uma redução de no mínimo 57.49% do número de variáveis binárias e de no mínimo 42.45% do número de restrições para cada um dos 5 grupos de instâncias de teste.

Tabela 3: Média e desvio padrão do número de variáveis binárias e de restrições para ambos os modelos.

Grupo de instâncias	Modelo (EVSP1)				Modelo (EVSP2)			
	Número de variáveis binárias		Número de restrições		Número de variáveis binárias		Número de restrições	
	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão
1-10	5537,80	989,57	9394,70	1675,97	1868,10	446,22	4557,50	1019,87
11-20	10902,60	2529,68	18564,70	4278,70	4020,30	1044,39	9659,50	2453,66
21-30	15542,80	3421,26	26489,40	5798,66	6023,30	1444,03	14277,50	3278,28
31-40	21451,80	3686,18	36532,00	6227,24	8770,10	1510,56	20443,90	3468,59
41-50	23579,40	3761,20	40206,20	6358,97	10023,40	1621,33	23137,70	3664,27

*Os valores utilizados na elaboração desta tabela foram obtidos antes da etapa de *presolve* do Gurobi Optimizer.

5. Considerações Finais

Neste trabalho apresentamos o problema de compartilhamento de veículos elétricos (EVSP), um novo problema de otimização combinatória motivado por sistemas de compartilhamento de carros elétricos. Este problema é uma variante do *transfers for commuting*, problema proposto e estudado por Böhmová et al. [2016]. Para o EVSP, propomos dois modelos de programação linear inteira mista, ambos baseados na modelagem de instâncias do problema como rede (ou redes) de espaço-tempo e de fluxo.

Os resultados obtidos a partir da realização de experimentos computacionais apontam que o segundo modelo, referenciado por (EVSP2), é mais eficiente que o primeiro modelo, referenciado por (EVSP1), para se obter uma solução ótima. Os modelos foram capazes de resolver quase todas as instâncias com até 40 clientes, apresentando dificuldade em resolver as instâncias do grupo com 50 clientes. De fato, o EVSP possui alta dificuldade de resolução por meio desses modelos, visto que, como inicialmente as baterias dos veículos estão completamente carregadas, o conjunto de demandas que podem ser atendidas por um EV, também podem ser atendidas por qualquer outro que esteja localizado inicialmente em uma mesma estação. Isto configura simetrias que são prejudiciais para um bom desempenho de ambos os modelos.

Trabalhos futuros com o EVSP podem ter diferentes focos. Como, por exemplo, investigar melhores formulações de programação linear inteira mista, o que também pode incluir técnicas de *warm-start*, heurísticas primais e duais, etc. Outro foco seria investigar métodos heurísticos para a obtenção de soluções viáveis para instâncias do EVSP difíceis e de grande porte.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, da Fundação de Amparo à Pesquisa do Estado de São Paulo - Brasil (FAPESP) processo 2015/11937-9 e do Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil (CNPq) processos 425340/2016-3, 314384/2018-9, 425806/2018-9, 435520/2018-0 e 311039/2020-0.

Referências

- Böhmová, K., Disser, Y., Mihalák, M., e Šrámek, R. (2016). Scheduling transfers of resources over time: Towards car-sharing with flexible drop-offs. In Kranakis, E., Navarro, G., e Chávez, E., editors, *LATIN 2016: Theoretical Informatics*, p. 220–234, Berlin, Heidelberg. Springer Berlin Heidelberg. ISBN 978-3-662-49529-2.
- Ding, N., Pan, J., Zhang, Z., e Yang, J. (2019). Life cycle assessment of car sharing models and the effect on GWP of urban transportation: A case study of Beijing. *Science of The Total Environment*, 688:1137–1144. ISSN 0048-9697. URL <https://www.sciencedirect.com/science/article/pii/S0048969719326798>.
- Luo, K., Erlebach, T., e Xu, Y. (2018a). Car-sharing between two locations: Online scheduling with flexible advance bookings. In Wang, L. e Zhu, D., editors, *Computing and Combinatorics*, p. 242–254, Cham. Springer International Publishing. ISBN 978-3-319-94776-1.
- Luo, K., Erlebach, T., e Xu, Y. (2018b). Car-sharing between two locations: Online scheduling with two servers. In *43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)*, volume 117 of *Leibniz International Proceedings in Informatics (LIPIcs)*, p. 50:1–50:14, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-086-6.
- Luo, K., Erlebach, T., e Xu, Y. (2018c). Online scheduling of car-sharing requests between two locations with many cars and flexible advance bookings. In *29th International Symposium on Algorithms and Computation (ISAAC 2018)*, volume 123 of *Leibniz International Proceedings in Informatics (LIPIcs)*, p. 64:1–64:13, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-094-1.
- Luo, K., Xu, Y., e Liu, H. (2020). Online scheduling of car-sharing request pairs between two locations. *Journal of Combinatorial Optimization*, 40:1–24. ISSN 1573-2886. URL <https://doi.org/10.1007/s10878-020-00635-8>.
- Silva, W. e Schouery, R. (2020). Maximum customers' satisfaction in one-way car-sharing: Modeling, exact and heuristic solving. URL <https://arxiv.org/abs/2008.07372>.
- Ströhle, P., Flath, C., e Gärttner, J. (2019). Leveraging customer flexibility for car-sharing fleet optimization. *Transportation Science*, 53(1):42–61. URL <https://doi.org/10.1287/trsc.2017.0813>.
- Xu, M. e Meng, Q. (2019). Fleet sizing for one-way electric carsharing services considering dynamic vehicle relocation and nonlinear charging profile. *Transportation Research Part B: Methodological*, 128:23–49. ISSN 0191-2615. URL <https://www.sciencedirect.com/science/article/pii/S0191261518309822>.