Logistical Planning for Electric Vehicles Under Time-Dependent Stochastic Traffic

Xiaowen Bi and Wallace K. S. Tang, *Senior Member, IEEE*

***Abstract*— For the benefit of global environmental preservation, electric vehicles (EVs) have been gradually accepted by people in the past few years. However, the technical problem of limited drivable range and long charging duration is still a major hurdle for the popularization of EVs, especially for commercial usage. In this paper, a dynamic electric vehicle routing problem (D-EVRP) model is designed for planning the itinerary for goods delivery by the utilization of EVs in logistics industry. To reflect the real situation, the D-EVRP considers a time-dependent stochastic traffic condition and captures the discharging/charging pattern of an EV using an analytical battery model. Its aim is to minimize the overall service duration, subject to a variety of the state-of-art constraints common in EV routing problems. Furthermore, to address the D-EVRP, a hybrid rollout algorithm (HRA), which incorporates a dedicated pre-planning strategy and a rollout algorithm, is also proposed. The effectiveness of the HRA and benefits of incorporating the analytical battery model are justified by extensive simulations using the real-world D-EVRP instances.**

***Index Terms*— Logistical planning, electric vehicles (EVs), time-dependent stochastic traffic, approximate dynamic programming.**

NOMENCLATURE

Notations for Vehicle's Dynamics

Parameters:

Q_p	Cargo capacity of EV
m_g	Gross weight of EV
g_a	Gravitational acceleration
c_r	Rolling friction coefficient
θ	Road gradient
ρ	Air density
A_f	EV's frontal area
c_d	Aerodynamic drag coefficient
f_r	Mass factor of EV's rotating parts
α	Recuperation efficiency
η	Energy efficiency of the electric motor
P_A	Power demand for EV's auxiliary components

Variables:

u, $\dot{u}$	Velocity and acceleration of EV
m	EV's total mass, summing up m_g and cargo weight
F_{roll}	Rolling friction between EV's pneumatic tires and road surface, where $F_{roll} = mg_a c_r \cos(\theta)$
F_{grade}	Grade resistance, where $F_{grade} = mg_a \sin(\theta)$
F_{air}	Aerodynamic resistance, where $F_{air} = \frac{1}{2}\rho A_f c_d u^2$
F_{acc}	Inertia of EV, where $F_{acc} = mf_r\dot{u}$
P_T	Mechanical power for moving EV
P_E	Electrical power drawn from battery pack

Notations for Battery Modeling

Parameters:

N_p	Number of parallel serial connections of cells
N_s	Number of cells in each serial connection
R	Constant resistance of electrochemical cell
E_0	Battery constant voltage
K	Polarization voltage
A	Exponential zone amplitude
B	Time constant inverse
Q_c	Maximum battery capacity
I_{cc}	Constant current for charging at CC phase
U_{cv}	Constant voltage for charging at CV phase
$\text{SoC}_{\max}$	SoC of a fully-charged battery

Variables:

$U_{oc}(t)$	Controlled open-circuit voltage source of battery cell
$I(t)$	Current flowing through battery cell
$U(t)$	Terminal voltage of battery cell
$\text{SoC}(t)$	Battery's state-of-charge, giving the ratio of present charge to Q_c
P_c	Electrical power drawn from each battery cell
$\text{SoC}(0)$	Initial SoC for recharging
t_s	Time index at which the charger switches from CC to CV phase

Notations for D-EVRP and HRA

Parameters:

$\text{SoC}_{\min}$	Minimum allowable SoC
ω	$\omega \in \mathbb{N}^+$, $2\omega+1$ is the maximal realization number of $T_{ij}(t)$
β	Threshold parameter that is close to 100%
γ	Discount factor
λ	Number of Monte-Carlo simulation episodes

Manuscript received April 14, 2018; revised October 3, 2018; accepted November 14, 2018. Date of publication December 12, 2018; date of current version October 2, 2019. The Associate Editor for this paper was Y. Lv. *(Corresponding author: Xiaowen Bi.)*

The authors are with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: xiaowenbi2-c@my.cityu.edu.hk; eekstang@cityu.edu.hk).

Digital Object Identifier 10.1109/TITS.2018.2883791

1524-9050 © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Variables:

d_{ij}	Distance for traveling edge (i, j)
$\tau_{ij}(t)$	Duration for traveling edge (i, j), which is a realization of random variable $T_{ij}(t) \sim N(\mu_{ij}(t), \sigma_{ij}(t))$
k	Discrete decision stage of D-EVRP
t_k	Discrete system time index, spaced with a constant interval σ
$v(t_k)$	EV's location at t_k
$C(t_k)$	EV's cargo weight at t_k
$P(s'\|s, a)$	Transition probability from state s to s', given control action a
$g(s, a, s')$	Transition cost from state s to s', given control action a
ϵ	Discretization precision, determined by ω and β

Tuple and Sets:

$\mathcal{V}_c$	Set of customers
$\mathcal{V}_s$	Set of charging stations
$\mathcal{V}$	Set of nodes, where $V = \{v_d\} \cup \mathcal{V}_c \cup \mathcal{V}_s$, with v_d being the depot
$\mathcal{E}$	Set of edges
$\mathcal{A}$	Set of weights for the edges
$\mathcal{S}$	State space of D-EVRP
s_k	System state $(t_k, v(t_k), \mathrm{SoC}(t_k), C(t_k), \mathcal{D}(t_k))$ at stage k
$\mathcal{D}(t_k)$	Customers' demands at t_k, where $\mathcal{D}(t_k) = \{D_k^1, D_k^2, ..., D_k^{\|\mathcal{V}_c\|}\}$
$\mathcal{D}_{\mathrm{ini}}$	Set of customers' initial demands
a_k	Control action in a form of $(v(t_{k+1}), \tau_c)$, where $v(t_{k+1})$ is EV's location at stage $k+1$ and τ_c is the discrete charging duration, taking values from Ψ
$\zeta(s_k)$	Set of feasible control actions given state s_k
Ψ	Set of charging duration options
Π, Π^*	Set of control policies $\pi(\cdot)$ and the optimal ones $\pi^*(\cdot)$

I. Introduction

ELECTRIC vehicle (EV) is one of the promising initiatives for environmental protection, to combat the deteriorating air pollution and global climate change. Recently, countries around the world [1] have put tremendous effort to promote its development and popularization. As compared to conventional fossil fuel-powered vehicles, EVs could be completely emission free by utilizing electricity generated from renewable energy resources, such as tide, sunlight and wind [2]. However, due to restrictions in current battery technology, EVs have quite limited driving range and considerably long recharging time. As a result, EV drivers are likely to suffer from range anxiety, fearing the power depletion before arriving at the destination. To alleviate the range anxiety and facilitate large-scale deployment of EVs, extensive research efforts have been made. For instance, [3]–[7] improve the utility of charging facilities by allocating them explicitly; [8]–[11] develop routing strategies to search for cost-effective paths; [12]–[15] focus on the control of EVs' internal energy storage system; [16]–[18] introduce efficient and economical controls of EVs' charging behavior.

It is a general belief that the upcoming transition from fuel-powered vehicles to EVs will lead to dramatic changes with respect to road transportation systems. As one of the major industries heavily relied on transportation, the distribution logistics would confront significant challenges in providing high-quality delivery service with EVs. Therefore, EV routing problem (E-VRP) and its variants are extensively studied in operational research community to expedite the profitable employment of EVs [19]. E-VRP takes the specifics of EVs into account, and aims to minimize the overall transportation cost through itinerary planning for a fleet of EVs to traverse a given set of customers. Some recent contributions are briefly summarized as follows. Reference [20] tackles the fixed route vehicle charging problem (FRVCP) with non-linear charging functions, which is derived from piecewise linear approximation. Given the visiting sequence of customers, the goal of FRVCP is to determine the occasion and quantity of battery charging. Reference [21] addresses the electric location routing problem with time windows and partial recharging (ELRP-TWPR). Its novelty lies in the simultaneous decision making in vehicle routing and charging infrastructure siting. Reference [22] considers the electric fleet size and mixed vehicle routing problem with time windows and recharging stations (E-FSMFTW), managing the composition of EV fleet together with vehicle routing.

It should be emphasized that, though existing literatures cover different practical constraints and objectives in the context of E-VRP, they rely on some implicit assumptions and simplifications. It is commonly assumed that the road traffic is static, meaning that the properties associated with the edges of the road network, such as traveling distance and duration, are stationary [20]–[24]. With a fixed vehicle route, static traveling distance is a plausible simplification. However, the traveling duration is indeed highly dynamic in reality due to fluctuating traffic conditions [25], [26]. As for the battery operations, in particular of the charging process, almost all the previous E-VRP models assume a linear relationship between the charging duration and state of charge (SoC) [21], [22], [24], [27], [28], except [20] in which piecewise linear functions are used to approximate the charging pattern. To the best of our knowledge, the incorporation of analytical battery modeling in EV routing is still in vain.

Evidently, assuming static traffic and/or neglecting battery characteristics would lead to inaccurate modeling for the real-world problems. Thus, in this paper, we propose a dynamic electric vehicle routing problem (D-EVRP) model for the logistical planning of a single EV based on finite-horizon Markov decision process (MDP) formulation. In particular, D-EVRP considers time-dependent stochastic traffic, which necessitates the sequential decision-making under uncertainty. Unlike the existing E-VRP models for which solutions are usually static routes, D-EVRP expects for a control policy, prescribing the optimal action to be taken for a given system state. It should be noted that such control policy not only

determines the successive path of route, but also the recharging duration for the EV if required. In addition, an analytical battery model [29], [30] is incorporated in D-EVRP to better capture the discharging and charging pattern of EV. The state-of-art constraints in E-VRP, including maximum cargo capacity and deterministic customer demand, are also included.

Since the state space cardinality of D-EVRP can be extremely large, the usage of stochastic dynamic programming (SDP) approach [31] on D-EVRP would induce a huge demand in computational resources. To mitigate the burden, a hybrid rollout algorithm (HRA) is proposed and designed. HRA combines a dedicated pre-planning strategy with a rollout algorithm. Based on historical traffic data, the pre-planning strategy is able to reduce the policy space by pre-selecting the path and also secure the itinerary from running out of power halfway. Rollout algorithm [32], as an approximate dynamic programming (ADP) approach [31], would derive much cheaper but sufficiently good solution policy, by approximating the cost-to-go function using simulation methods.

For demonstrative purpose, the logistics company data [26] and the charging network data [33], respectively indicating the location of customers and public charging stations in Singapore, are utilized to generate multiple D-EVRP instances. The Nissan's E-NV200 [34] electric van is assumed to be employed and historical traffic data are collected via Google Map Distance Matrix API [35]. As shown in our simulations, the proposed algorithm can effectively solve D-EVRP instances by acquiring decent solution policies within a reasonable time frame, and such decision-making benefits from the incorporation of analytical battery model.

The remainder of the paper is organized as follows. In Sect. II, major elements in D-EVRP are described and the D-EVRP is formulated. The proposed HRA is then explained in Sect. III. To demonstrate the effectiveness of HRA, extensive simulations are conducted and results are presented in Sect. IV. Finally, conclusion is given in Sect. V.

II. Problem Formulation

A. Problem Description

In D-EVRP, the commodity delivery from depot v_d to a set of customers $\mathcal{V}_c$ is performed by a single EV. If necessary, EV can recharge its battery at a charging station $v_s \in \mathcal{V}_s$, which is assumed to be publicly available and able to serve any incoming EVs simultaneously with DC fast charger. Based on the locations of involved parties and traffic conditions, D-EVRP can be defined over a complete digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, where $\mathcal{V} = \{v_d\} \cup \mathcal{V}_c \cup \mathcal{V}_s$ is the set of nodes, $\mathcal{E} = \{(i, j) | i, j \in \mathcal{V}, i \neq j\}$ is the set of edges and $\mathcal{A} = \{(d_{ij}, \tau_{ij}(t)) | (i, j) \in \mathcal{E}\}$ is the set of weights associated to edges, with d_{ij} and $\tau_{ij}(t)$ being the traveling distance and duration from i to j, respectively. $\tau_{ij}(t)$ is a realization of a time-dependent random variable $T_{ij}(t)$, which is assumed to follow a normal distribution, i.e., $T_{ij}(t) \sim N(\mu_{ij}(t), \sigma_{ij}(t))$, and remains unchanged in the next constant time interval σ.

To start, an EV departs from the depot v_d with a fully-charged battery, carrying a full-load of commodity Q_p for its customers. Each customer demands a fixed amount of commodity, which is less than Q_p. The EV will visit a customer only if it has sufficient stock. Whenever the carried stock is depleted, the EV must return to v_d to restore the load up to Q_p and for simplicity, every replenishment is assumed to be completed instantaneously. After delivering goods to all the customers, the EV shall return to v_d.

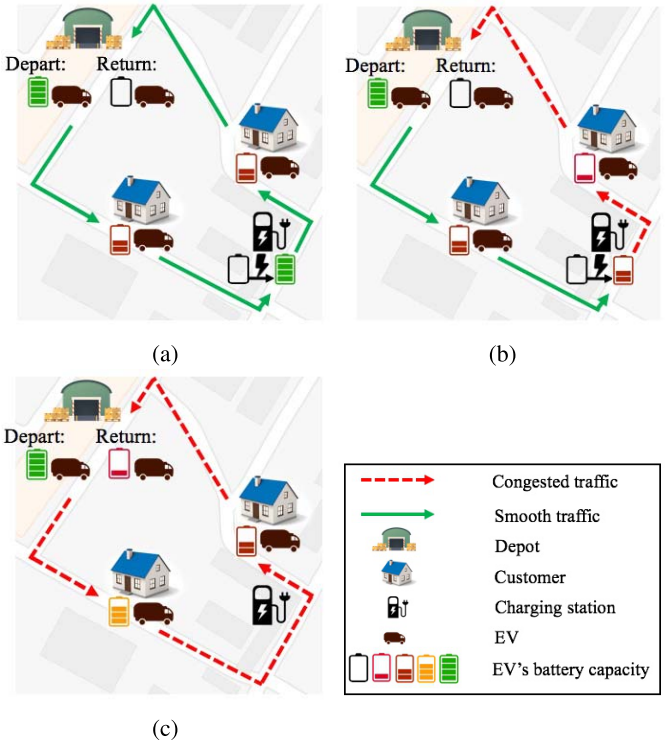


Fig. 1. An example for D-EVRP with 3 different itineraries. (a) Itinerary 1. (b) Itinerary 2. (c) Itinerary 3.

The objective of D-EVRP is to minimize the total duration of the delivery service. In this work, it is assumed that EV always travels as fast as possible under the current traffic condition, which is specified by the traveling duration $T_{ij}(t)$. Therefore, given the traffic condition at time t, the EV's average velocity for traveling (i, j) can be estimated by $u = \frac{d_{ij}}{\tau_{ij}(t)}$. Clearly, an EV traveling at high average velocity will certainly contribute to an efficient commodity distribution. However, as demonstrated in [36], EV in high speed also suffers from higher power consumption and shorter drivable range due to the aerodynamic resistance, which is proportional to the square of an EV's speed (see definition of F_{air} in *Nomenclature*). Such an increase in energy consumption may necessitate a halfway recharging, which can be quite time-consuming even with DC fast chargers. Therefore, minimizing the total duration of delivery signifies an optimal trade-off between traveling and recharging when planning the itinerary for EV. Since the traffic condition is time-dependent, traveling the right path at the right time to minimize the impact of recharging is of vital importance for addressing D-EVRP.

Due to the stochasticity of traveling duration, the decision-making of D-EVRP is sequential, and lies in not only the path selection, but also the determination of charging duration at each charging station. In Fig. 1, a toy D-EVRP instance consisting of two customers and one charging station is considered

and three possible itineraries are presented. For illustrative purpose, the EV consumes 1 unit and 2 units of battery power, respectively when it travels an edge under a congested traffic condition (red dashed edge) and a smooth traffic condition (green edge). The charging duration is determined based on traffic condition, so as to avoid unnecessary time consumption. For example, the traffic in Fig. 1(a) happens to be smooth all along the way, and hence more battery power would be consumed. Therefore, it is necessary for the EV to fully recharge its battery at the charging station so that the remaining paths can be completed. For the itinerary in Fig. 1(b), recharging the battery to half-full is already sufficient for remaining paths because less battery power would be spent when EV's speed is limited by a congested traffic condition. Similarly, the EV in Fig. 1(c) under congested traffic does not need to recharge at all.

B. Energy Consumption

In this section, the dynamics of EV [36] is briefly introduced. To overcome external resistances and drive an EV at speed u, the required mechanical power P_T is given by

$$P_T = (F_{roll} + F_{grade} + F_{air} + F_{acc}) \cdot u \tag{1}$$

where the definitions of F_{roll}, F_{grade}, F_{air} and F_{acc} are given in *Nomenclature*. The mechanical propulsion of an EV relies on its internal electric motor, which converts the electrical energy stored in the battery into mechanical energy. The electrical power P_E drawn from the battery is thus

$$P_E = (1-\alpha)\left(\frac{P_T}{\eta} + P_A\right) \tag{2}$$

where α is the recuperation efficiency, which is defined to reflect the effectiveness of the regenerative braking; η is the energy efficiency of the electric motor; and P_A is the additional power required for EV's auxiliary components.

C. Generic Battery Modeling

In order to tackle with D-EVRP, a generic battery model is adopted to characterize the dynamical behavior of EV's battery. The EV's battery is inherently a pack of electrochemical cells, converting the chemical energy into electrical energy, or vice versa. The cells are usually connected in a N_p-parallel-and-N_s-series structure, where N_p indicates the number of parallel serial connections of cells and N_s is the number of cells in each serial connection. The dynamics of an individual cell can be captured by a simple electrical circuit [29], [30], which includes a controlled open-circuit voltage source $U_{oc}(t)$ in series with a constant resistance R. For both discharging and charging operations, $U_{oc}(t)$ can be approximated by

$$U_{oc}(t) = E_0 - \frac{K}{\mathrm{SoC}(t)} + A \cdot e^{-B \cdot Q_c \cdot (1-\mathrm{SoC}(t))} \tag{3}$$

where Q_c is the maximum battery capacity, $\mathrm{SoC}(t)$ is the state of charge giving the ratio of present charge with respect to Q_c, and the parameters E_0, K, A and B are defined as in *Nomenclature*. The practical method to extract these parameters from battery manufacturer's discharge curve (with constant current) can be found in [30] and is omitted here. Also, for the sake of simplicity, other factors, such as the temperature effect, Peukert effect and self-discharge, are ignored in Eq. (3). In this paper, Lithium-ion (Li-ion) battery, which is commonly used in recent EVs, is focused.

1) Discharging Process: All the cells in a Li-ion battery pack are assumed to be identical and energy is evenly distributed among them. If an overall power P_E is demanded, the electrical power P_c drawn from each cell can be calculated by

$$P_c = \frac{P_E}{N_p \cdot N_s} \tag{4}$$

The changing rate of the instantaneous SoC is derived as [29]

$$\frac{d\mathrm{SoC}(t)}{dt} = -\frac{I(t)}{Q_c} \tag{5}$$

where $I(t)$ is the current flowing through the battery cell, and

$$I(t) = \frac{U_{oc}(t) - \sqrt{U_{oc}^2(t) - 4 \cdot R \cdot P_c}}{2R} \tag{6}$$

2) Charging Process: A Li-ion battery cell is charged with firstly a constant current I_{cc} and then a constant voltage U_{cv}. The $\mathrm{SoC}(t)$ during the so-called CC and CV phases are given by Eq. (7) and Eq. (8) [29], respectively.

$$\mathrm{SoC}(t) = \mathrm{SoC}(0) - \frac{I_{cc} \cdot t}{Q_c}, \quad \forall t \mid U(t) < U_{cv} \tag{7}$$

$$\frac{d\mathrm{SoC}(t)}{dt} = \frac{U_{cv} - U_{oc}(t)}{Q_c \cdot R}, \quad \forall t \geq t_s \tag{8}$$

where $\mathrm{SoC}(0)$ is the initial SoC; $U(t)$ denotes the terminal voltage; and t_s is the time index at which the charger is switched to CV phase. Note that I_{cc} is negative to indicate the direction of current flow (i.e. charging).

D. Finite-Horizon MDP Formulation

In this section, it is to describe how the D-EVRP can be formulated as a finite-horizon MDP model. Let $\mathcal{S}$ be the state space of D-EVRP, the state $s_k \in \mathcal{S}$ is a tuple $(t_k, v(t_k), \mathrm{SoC}(t_k), C(t_k), \mathcal{D}(t_k))$ where the definitions of k, t_k, $v(t_k)$, $C(t_k)$ and $D(t_k)$ are given in *Nomenclature*.

Assuming an initial user demand set $\mathcal{D}_{\mathrm{ini}}$, the initial state of D-EVRP is denoted by s_1, in which $t_1 = 0$, $v(t_1) = v_d$, $\mathrm{SoC}(t_1) = \mathrm{SoC}_{\max}$, $C(t_1) = Q_p$ and $\mathcal{D}(t_1) = \mathcal{D}_{\mathrm{ini}}$. The state s_k is reckoned as the absorbing terminal state s_r, if the following conditions of route accomplishment are satisfied

$$\begin{aligned} D_k^i &= 0, \quad \forall i \in \{1, 2, \cdots, |\mathcal{V}_c|\} \\ v(t_k) &= v_d \\ \mathrm{SoC}(t_k) &\geq \mathrm{SoC}_{\min} \end{aligned} \tag{9}$$

where $\mathrm{SoC}_{\min}$ indicates the minimum allowable SoC. For any state $s_k \in \mathcal{S} \setminus \{s_r\}$, policy $\pi_k(\cdot)$ deterministically associates a control action a_k to s_k, i.e. $a_k = \pi_k(s_k)$. A control action a_k belongs to a feasible control set $\zeta(s_k)$, which is defined as

$$\zeta(s_k) = (\zeta_1(s_k) \cup \zeta_2(s_k)) \cap (\zeta_3(s_k) \cup \zeta_4(s_k)) \tag{10}$$

where

$$\begin{aligned}
\mathcal{V}_p &= \{v_i \in \mathcal{V}_c : i \in [1, |\mathcal{V}_c|] \wedge D_k^i \neq 0\} \\
\zeta_1(s_k) &= \{v(t_{k+1}) | (\forall v(t_k) \in \mathcal{V}_s)[v(t_{k+1}) = v(t_k)]\} \\
&\quad \times \{\tau_c | \tau_c \in \Psi\} \\
\zeta_2(s_k) &= \{v(t_{k+1}) | (\forall v(t_k) \notin \mathcal{V}_s)[v(t_{k+1}) \\
&\quad \in \{v_d \cup \mathcal{V}_p \cup \mathcal{V}_s\} \setminus v(t_k)]\} \times \{\tau_c | \tau_c \in \Psi\} \\
\zeta_3(s_k) &= \{v(t_{k+1}) | v(t_{k+1}) \in \{v_d \cup \mathcal{V}_p\}\} \times \{\tau_c | \tau_c \equiv 0\} \\
\zeta_4(s_k) &= \{v(t_{k+1}) | v(t_{k+1}) \in \mathcal{V}_s\} \times \{\tau_c | \tau_c > 0\}
\end{aligned} \tag{11}$$

The control a_k is in the form of $(v(t_{k+1}), \tau_c)$, where τ_c is the discrete charging duration, taking values from a set of options Ψ. Due to the variety of battery's charging pattern, the elements of Ψ in addition to 0 (i.e. no charging action) shall be determined in accordance with the battery characteristics. Obviously, if Ψ contains more elements, the decision would be more precise, but the evaluation of an enlarged control set $\zeta(s_k)$ would certainly lead to higher computational burden. Note that t_k is specified as the time index at which the EV has already taken the control action a_k at $v(t_k)$.

In Eq. (10), the control set $\zeta(s_k)$ is defined as an intersection of two unions. The union of $\zeta_1(s_k)$ and $\zeta_2(s_k)$ summarizes all the possible control actions given state s_k, while the union of $\zeta_3(s_k)$ and $\zeta_4(s_k)$ gives reasonable constraints. According to Eq. (11), $\zeta_1(s_k)$ prescribes that EV can choose to stay (and charge) at $v(t_k)$ for t_{k+1} only when $v(t_k) \in \mathcal{V}_s$; $\zeta_2(s_k)$ specifies the general control actions when $v(t_k) \notin \mathcal{V}_s$; $\zeta_3(s_k)$ allows no charging options at either the depot or customers' spot; $\zeta_4(s_k)$ stipulates that, the EV has to charge at least a minimum duration once arriving at any charging station.

Given the current state s_k and the control action a_k, the system proceeds to the successive state s_{k+1}, incurring on a transition probability $P(s_{k+1}|s_k, a_k)$ and a transition cost $g(s_k, a_k, s_{k+1})$. The transition probability $P(s_{k+1}|s_k, a_k)$ is given by the non-stationary traffic. As discussed in Section. II-A, each edge $(i, j) \in \mathcal{E}$ is associated with a time-dependent random variable $T_{ij}(t)$ of traveling duration. For any realization $\tau_{ij}(t) \in \{\mu_{ij}(t) \pm 2n\epsilon | n \in [0, \omega]\}$, $T_{ij}(t)$ can be discretized as follows

$$\begin{aligned}
&P(T_{ij}(t) = \tau_{ij}(t)) \\
&= P(\tau_{ij}(t) - \epsilon \leq T_{ij}(t) \leq \tau_{ij}(t) + \epsilon) \\
&= P\left(\frac{\tau_{ij}(t) - \epsilon - \mu_{ij}(t)}{\sigma_{ij}(t)} \leq \frac{T_{ij}(t) - \mu_{ij}(t)}{\sigma_{ij}(t)} \leq \frac{\tau_{ij}(t) + \epsilon - \mu_{ij}(t)}{\sigma_{ij}(t)}\right) \\
&= \Phi\left(\frac{\tau_{ij}(t) + \epsilon - \mu_{ij}(t)}{\sigma_{ij}(t)}\right) - \Phi\left(\frac{\tau_{ij}(t) - \epsilon - \mu_{ij}(t)}{\sigma_{ij}(t)}\right)
\end{aligned} \tag{12}$$

where ϵ is the discretization precision; $\omega \in \mathbb{N}^+$ and $2\omega + 1$ is the maximal number of realizations; $P(T_{ij}(t) = \tau_{ij}(t))$ is the probability of taking a duration of $\tau_{ij}(t)$ to travel (i, j); and $\Phi(\cdot)$ represents the cumulative density function. Given any ω, ϵ is determined based on the following condition

$$\sum_{n=-\omega}^{\omega} P(T_{ij}(t) = \mu_{ij}(t) + 2n\epsilon) \geq \beta \tag{13}$$

where β is a threshold parameter close to 100%, including traveling durations that are most likely to realize. With β fixed, a larger ω would lead to more successive states, which may contribute in a better decision-making. Let $t = t_k$, $i = v(t_k)$ and $j = v(t_{k+1})$, the transition probability $P(s_{k+1}|s_k, a_k)$ is

$$P(s_{k+1}|s_k, a_k) = P(T_{v(t_k)v(t_{k+1})}(t_k) = \tau_{v(t_k)v(t_{k+1})}(t_k)) \tag{14}$$

The time index t_{k+1} of successive state s_{k+1} is thus

$$t_{k+1} = t_k + \lceil g(s_k, a_k, s_{k+1}) \rceil \tag{15}$$

where the transition cost $g(s_k, a_k, s_{k+1})$ is defined by

$$g(s_k, a_k, s_{k+1}) = \frac{1}{\sigma} \cdot (\tau_c + \tau_{v(t_k)v(t_{k+1})}(t_k)) \tag{16}$$

Depending on the control a_k for different possible cases, other components of state s_{k+1} is updated accordingly:

1) $a_k = (v_d, 0)$

$$\begin{aligned}
\mathrm{SoC}(t_{k+1}) &= \mathrm{SoC}(t_k) - M(v(t_k), v_d) \\
C(t_{k+1}) &= Q_p \\
\mathcal{D}(t_{k+1}) &= \mathcal{D}(t_k)
\end{aligned} \tag{17}$$

2) $a_k = (v_i, 0)$ with $v_i \in \mathcal{V}_c$ and $i \in [1, |\mathcal{V}_c|]$

$$\begin{aligned}
\mathrm{SoC}(t_{k+1}) &= \mathrm{SoC}(t_k) - M(v(t_k), v_i) \\
C(t_{k+1}) &= C(t_k) - D_k^i \\
D_{k+1}^i &= 0
\end{aligned} \tag{18}$$

3) $a_k = (v_s, \tau_c)$ with $v_s \in \mathcal{V}_s$

$$\begin{aligned}
\mathrm{SoC}(t_{k+1}) &= \mathrm{SoC}(t_k) - M(v(t_k), v_s) + N(\tau_c) \\
C(t_{k+1}) &= C(t_k) \\
\mathcal{D}(t_{k+1}) &= \mathcal{D}(t_k)
\end{aligned} \tag{19}$$

4) $a_k = (v(t_k), \tau_c)$ with $v(t_{k+1}) = v(t_k)$

$$\begin{aligned}
\mathrm{SoC}(t_{k+1}) &= \mathrm{SoC}(t_k) + N(\tau_c) \\
C(t_{k+1}) &= C(t_k) \\
\mathcal{D}(t_{k+1}) &= \mathcal{D}(t_k)
\end{aligned} \tag{20}$$

where $M(i, j)$ is a function to return the discharged SoC after traveling along the edge (i, j), and $N(\tau_c)$ is another function to give the amount of SoC recovered by charging for τ_c.

Given the aforementioned specifications, D-EVRP is formally defined as a combinatorial optimization problem, minimizing the expected total duration of logistics service:

$$\begin{aligned}
&\min_{\pi_k(\cdot)} J = \mathbb{E}[G(s_k)] = \mathbb{E}\left[\sum\nolimits_{k=1}^{r-1} \gamma^k g(s_k, a_k, s_{k+1})\right] \\
&\text{s.t. } s_k \in \mathcal{S} \setminus \{s_r\} \quad \text{and} \quad a_k \in \zeta(s_k)
\end{aligned} \tag{21}$$

where J is cost-to-go function, which evaluates the expected service duration from now on, averaging over a stochastic distribution of traveling duration; $G(s_k)$ is the cumulative transition cost from state s_k to s_r; and γ is the discount factor.

The minimization of J is achieved by a set of optimal control policy $\Pi^* = \{\pi_1^*(\cdot), \pi_2^*(\cdot), ..., \pi_p^*(\cdot)\}$. In particular,

given any state $s_k \in \mathcal{S} \setminus \{s_r\}$ under a control policy $\pi_k(\cdot)$, $J(s_k)$ is obtained by the Bellman expectation equation

$$J(s_k) = g(s_k, a_k, s_{k+1}) + \gamma \sum_{s_{k+1}} P(s_{k+1}|s_k, a_k) J(s_{k+1}) \quad (22)$$

Based on Bellman's principle of optimality, one has

$$\pi_k^*(s_k) = \arg\min_{a_k \in \zeta(s_k)} \Big\{ g(s_k, a_k, s_{k+1}) + \gamma \sum_{s_{k+1}} P(s_{k+1}|s_k, a_k) J^*(s_{k+1}) \Big\} \quad (23)$$

where $J^*(s_k) = \min J(s_k)$.

III. Methodology

The classical SDP approaches, including policy iteration and value iteration, etc., are guaranteed to find an optimal policy in polynomial time [37] for any finite-horizon MDPs like D-EVRP, when the probability and cost of state transition are fully available. In SDP approaches, policies are evaluated by performing the backup operation, in which the cost-to-go function $J(s_k)$ of each state $s_k \in \mathcal{S} \setminus \{s_r\}$ is computed recursively using Eq. (22). However, when the problem scale increases, the state space $\mathcal{S}$ would grow exponentially, making the backup operation computationally intractable. To provide solution to the curse of dimensionality in large-scale MDPs, rollout algorithm is usually employed. The rollout algorithm is basically a heuristic version of policy iteration, while the optimal cost-to-go function $J^*(s_{k+1})$ in Eq. (23) is replaced with an approximation $\hat{J}(s_{k+1})$, the value of which is estimated using Monte-Carlo simulation (MCS). Although rollout algorithm only obtains good, but not necessarily optimal policies, its computational efficiency is significantly better.

Since the decision-making in D-EVRP comprises the selection of both successive path and charging duration, the cardinality of policy space, $|\Pi|$, can be enormous. Therefore, a hybrid version of rollout algorithm (HRA) is proposed by incorporating a newly devised pre-planning strategy. HRA's procedures (i.e. HybridRA) are described in **Algorithm 1**.

In particular, the pre-planning strategy (PrePlan in line-7 of Algorithm 1 and described in **Algorithm 2**) aims to reduce the cardinality of policy space while maintain the optimality of solution policy as much as possible. It is used in procedure HybridRA to resolve the control set $\zeta(s_k)$. Procedure PrePlan consists of two phases: *Phase I* determines the routing options tentatively according to EV's present load as well as the customers' pending demands, while *Phase II* updates the routing options from *Phase I* decisively based on EV's current SoC. The update in *Phase II* is to secure the itinerary from power depletion in halfway. Using the historical traffic data, a two-step look ahead (TwoStep in line-9 of Algorithm 2 and described in **Algorithm 3**) is carried out to estimate the maximal drop of SoC from any routing option to the charging stations. Procedure TwoStep is feasible because given state s_k, the maximal SoC drop for any successive paths can be estimated by a one-step look ahead (OneStep in line-16 of Algorithm 2 and line-8 of Algorithm 3, described

Algorithm 1 HRA

1: **procedure** HybridRA
2: Initialize set $\Pi, \mathcal{S}_{tp}, \mathcal{S}_{ow} \leftarrow \emptyset$ and $k \leftarrow 1$
3: $s_k \leftarrow (0, v_d, \text{SoC}_{\max}, \mathcal{Q}_p, \mathcal{D}_{\text{ini}})$
4: $\mathcal{S}_{ow} \leftarrow \mathcal{S}_{ow} \cup \{s_k\}$
5: **while** $\mathcal{S}_{ow} \neq \emptyset$ **do**
6: **for all** $s_k \in \mathcal{S}_{ow}$ **do**
7: $\zeta(s_k) \leftarrow$ PrePlan(s_k, Ψ) ▷ Pre-planning
8: **for all** $a_k \in \zeta(s_k)$ **do**
9: **for all** s_{k+1} **do**
10: $\hat{J}(s_{k+1}) \leftarrow$ MCEval$(s_{k+1}, \pi^{bs}(\cdot), \lambda)$
11: $\pi_k(s_k) \leftarrow \arg\min_{a_k \in \zeta(s_k)} \{g(s_k, a_k, s_{k+1}) + \gamma \sum_{s_{k+1}} P(s_{k+1}|s_k, a_k) \hat{J}(s_{k+1})\}$
12: **for all** s_{k+1} given $\pi_k(s_k)$ **do**
13: $\mathcal{S}_{tp} \leftarrow \mathcal{S}_{tp} \cup \{s_{k+1}\}$
14: $\Pi \leftarrow \Pi \cup \{\pi_k(\cdot)\}$
15: $k \leftarrow k + 1$
16: $\mathcal{S}_{ow} \leftarrow \mathcal{S}_{tp}$
return Π

Algorithm 2 A Pre-Planning Strategy

1: **procedure** PrePlan(s_k, Ψ)
2: Initialize set $\mathcal{H}_t, \mathcal{H}_d \leftarrow \emptyset$
3: **for** $i \leftarrow 1$ to $|\mathcal{V}_c|$ **do** ▷ Phase I
4: **if** $D_k^i \leq C(t_k)$ **then**
5: $\mathcal{H}_t \leftarrow \mathcal{H}_t \cup \{v_i\}$
6: **if** $\mathcal{H}_t = \emptyset$ **then**
7: $\mathcal{H}_t \leftarrow \mathcal{H}_t \cup \{v_d\}$
8: **for** $i \leftarrow 1$ to $|\mathcal{H}_t|$ **do** ▷ Phase II
9: **if** TwoStep(s_k, v_i) **then**
10: $\mathcal{H}_d \leftarrow \mathcal{H}_d \cup \{v_i\}$
11: **if** $\mathcal{H}_d = \emptyset$ **then**
12: **if** $v(t_k) \in \mathcal{V}_s$ **then**
13: $\mathcal{H}_d \leftarrow \mathcal{H}_d \cup \{v(t_k)\}$
14: **else if** $v(t_k) \notin \mathcal{V}_s$ **then**
15: **for** $i \leftarrow 1$ to $|\mathcal{V}_s|$ **do**
16: $\text{SoC}_{os} \leftarrow$ OneStep(s_k, v_i)
17: **if** $\text{SoC}_{os} \geq \text{SoC}_{\min}$ **then**
18: $\mathcal{H}_d \leftarrow \mathcal{H}_d \cup \{v_i\}$
return $\zeta(s_k) \leftarrow \mathcal{H}_d \times \Psi$

Algorithm 3 Two-Step Look Ahead

1: **procedure** TwoStep(s_k, v_{os})
2: $\text{SoC}(t_{k+1}) \leftarrow$ OneStep(s_k, v_{os})
3: $\tau_{v(t_k)v_{os}}(t_k) \leftarrow \arg\max M(v(t_k), v_{os})$
4: $t_{k+1} \leftarrow t_k + \lceil \frac{1}{\sigma} \cdot \tau_{v(t_k)v_{os}}(t_k) \rceil$
5: Obtain $C(t_{k+1})$ and $\mathcal{D}(t_{k+1})$ from Eq. (17) or Eq. (18)
6: $s_{k+1} \leftarrow (t_{k+1}, v_{os}, \text{SoC}(t_{k+1}), C(t_{k+1}), \mathcal{D}(t_{k+1}))$
7: **for** $i \leftarrow 1$ to $|\mathcal{V}_s|$ **do**
8: $\text{SoC}(t_{k+2}) \leftarrow$ OneStep(s_{k+1}, v_i)
9: **if** $\text{SoC}(t_{k+2}) \geq \text{SoC}_{\min}$ **then return** true
return false

Algorithm 4 One-Step Look Ahead

1: **procedure** ONESTEP(s_k, υ_{os})
2: $\text{SoC}_{dp} \leftarrow \max M(\upsilon(t_k), \upsilon_{os})$
3: **return** $\text{SoC}_{os} \leftarrow \text{SoC}(t_k) - \text{SoC}_{dp}$

in **Algorithm 4**). If the EV can arrive at any charging stations without power depletion, such a routing option is valid for the present control set. While if none of those options is valid, EV must recharge at the next decision stage. In this case, $\zeta(s_k)$ shall be determined according to the EV's present location. If EV is presently located at a charging station, it would simply stay and charge for more. Otherwise, there must be at least one reachable charging station since the present location was reckoned to be valid in the previous stage.

Given the control set $\zeta(s_k)$, MCS (MCEVAL in line-10 of Algorithm 1 and described in **Algorithm 5**) is then applied to approximate the cost-to-go function $J(s_{k+1})$ for all of the successive states s_{k+1}. Consider the heuristic approach by visiting nearest node and charging for minimum period as the base policy $\pi^{bs}(\cdot)$, MCS generates λ episodes of D-EVRP experiences (from s_{k+1} to s_r), each of which can be deemed as a path in the MDP graph. Note that by the law of large numbers, the larger λ is, the more effective but also time-consuming HRA can be. The approximation $\hat{J}(s_{k+1})$ is calculated by averaging the cumulative transition cost $G(s_k)$ of all the MCS episodes. By further solving Eq. (23) with $\hat{J}(s_{k+1})$, the control policy $\pi_k(s_k)$ can be heuristically improved from the base one.

Algorithm 5 Monte-Carlo Policy Evaluation

1: **procedure** MCEVAL($s_k, \pi^{bs}(\cdot), \lambda$)
2: Initialize $\hat{J}(s_k) \leftarrow 0$
3: **for** $i \leftarrow 1$ to λ **do**
4: Generate an episode using $\pi^{bs}(\cdot)$
5: Let $a_k^{bs} \leftarrow \pi^{bs}(s_k)$
6: $G(s_k) \leftarrow \sum_{k'=0}^{r-1} \gamma \cdot g(s_{k+k'}, a_{k+k'}^{bs}, s_{k+k'+1})$
7: $\hat{J}(s_k) \leftarrow \hat{J}(s_k) + G(s_k)$
return $\hat{J}(s_k) \leftarrow \frac{\hat{J}(s_k)}{\lambda}$

In HRA, the operations of pre-planning, MCS and policy improvement are conducted iteratively, until all the episodes of D-EVRP terminate. As a result, a set of control policies Π will be returned as the solution for D-EVRP.

IV. EXPERIMENTAL RESULTS

A. Case Study: Logistics in Singapore

To evaluate the performance of HRA, a real-world problem confronted by a logistics company in Singapore [26] is considered as a case study. The company intends to distribute commodity to a set of customers widely dispersed around Singapore using EVs. By merging the customers whose actual locations are highly aggregated, the total number of customers to be traversed is reduced to 24. Furthermore, 14 charging stations [33] are also included to allow recharging of EV. The geographic locations of these nodes are as shown in Fig. 2.

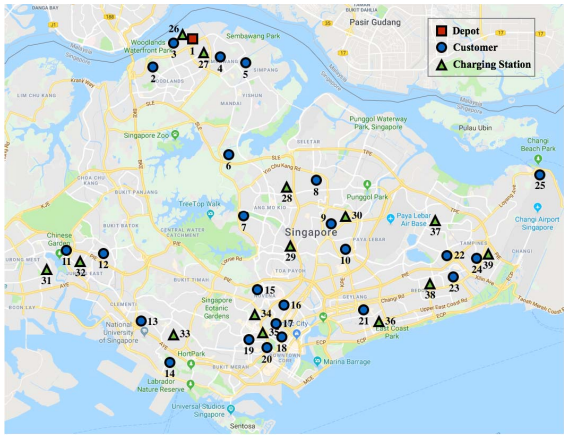


Fig. 2. Geographic locations of nodes in Singapore.

TABLE I
NODE CLUSTERING FOR CASE A AND B

Case	Instance	$\mathcal{V}_c$	$\mathcal{V}_s$	$\lvert\mathcal{V}\rvert$
A	1	2, 3, 4, 5	26, 27	7
	2	6, 7, 8, 9, 10	28, 29, 30	9
	3	11, 12, 13, 14	31, 32, 33	8
	4	15, 16, 17, 18, 19, 20	34, 35	9
	5	21, 22, 23, 24, 25	36, 37, 38, 39	10
B	6	2, 3, 4, 5, 6, 7, 8, 9, 10, 22, 23, 24, 25	26, 27, 28, 29, 30, 36, 37, 38, 39	23
	7	11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21	31, 32, 33, 34, 35, 36	18

Assuming the logistics company owns more than one EV, customers can be partitioned into multiple clusters, with accessible charging stations assigned, each of which is deemed as a D-EVRP instance and addressed independently by HRA. Following the work in [26], customers are partitioned based on the postal code of regions, giving Instance 1 to Instance 5 (i.e. Case A) as shown in Table. I. Since the instances' scales are small enough for SDP approaches, they're ideal for evaluating the quality of HRA's solution policy by comparing with the true optimality. Furthermore, to investigate the performance of HRA for large-scale instances, the nodes are re-partitioned in Case B to form another two instances, Instances 6 and 7, as shown in Table. I. It should be emphasized that, the number of nodes studied in others' work (e.g. [26] only considers 11 customers) is fewer than these two instances. Also, as shown later, the total time to serve such a set of customers is already more than 7 hours, which is considered to be long for a driver.

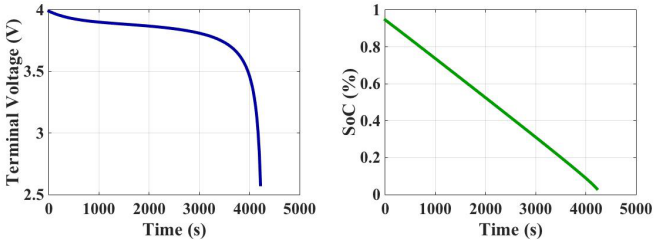
The Google Map Distance Matrix API [35] is adopted to collect the estimated traveling duration of each directed edge, which takes traffic conditions into account. Starting from 11:00 am, the data is enquired for every 300 seconds (i.e. $\sigma = 300$) and in total, 100 samples are acquired per day. The dynamics of traveling duration used in the simulation is then captured using the weekly traffic data.

TABLE II
PARAMETERS OF E-NV200'S BATTERY CELL MODEL

Symbols	R	E_0	K	A	B	Q_c
Values	0.001	3.9747	0.0427	0.15	0.3191	32.5
Units	Ω	V	V	V	Ah^{-1}	Ah

TABLE III
PARAMETERS OF E-NV200'S DYNAMICS

Symbols	Q_p	m_g	g_a	c_r	ρ	A_f	c_d
Values	650	2200	9.81	0.008	1.055	1.51	0.33
Units	kg	kg	ms^{-2}	N/A	kgm^{-3}	m^2	N/A
Symbols	f_r	α	η	P_A			
Values	1	0.1	0.9	400			
Units	N/A	N/A	N/A	W			



(a) (b)

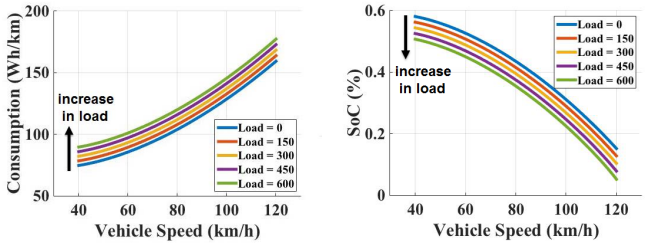
Fig. 3. The discharging pattern of E-NV200's battery pack. (a) Terminal voltage. (b) SoC.

B. E-NV200's Battery Modeling

In this work, Nissan's E-NV200 [34] electric van is assumed to be employed for the delivery tasks. Based on the manufacturer's discharging profile [38], the parameters of the generic model (see Sec. II-C) for a single Li-ion cell of E-NV200's battery pack, are derived as shown in Table. II.

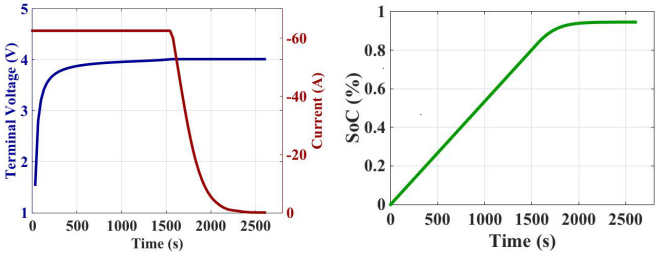
1) Discharging Pattern: The EV's discharging process is closely related to the vehicles' dynamics. Referring to [34] and [39], the parameters of the energy consumption model are listed in Table. III. For simplicity, the gradient θ is assumed to be zero (i.e. flat ground). The electrical power consumption P_E for traveling a particular path is estimated with Eq. (2). Based on the battery model specified in Table. II, the drop in SoC can be calculated with a given initial SoC using Eq. (5).

To evaluate our battery model, Fig. 3 depicts an example of the discharging pattern of a E-NV200's single run. Specifically, the EV starts with a fully-loaded cargo and a fully-charged battery. It travels at an average velocity of 30 ms^{-1}, and finally reaches a maximal drivable range of around 130km in 70 minutes. This distance, 130km, is shorter than 170km which has been reported in [34], because the EV is now heavily loaded and traveling at a sustained high speed (the maximal speed of E-NV200 is about 33.975ms^{-1}). The energy consumption and the SoC versus EV speed are also investigated, and the results are shown in Fig. 4 (a) and (b), respectively. It can be clearly observed that, the EV's energy consumption gradually increases (and so SoC decreases) when either cargo load or traveling velocity increases. Such observation is also consistent with the literatures aforementioned [36], [40].



(a) (b)

Fig. 4. Energy consumption and SoC versus EV's speed. (a) Energy consumption. (b) SoC.



(a) (b)

Fig. 5. The charging pattern of E-NV200's battery pack. (a) Terminal voltage/ current. (b) SoC.

2) Charging Pattern: The battery pack of E-NV200 consists of 48 energy modules connected in series. Since each energy module connects 4 cells in a 2-parallel-and-2-series formation [38], the battery pack has a structure of 2-parallel-and-96-series, i.e. $N_p = 2$ and $N_s = 96$. Assuming the DC fast chargers provide a power of 50kW (125A), under CHAdeMO standard [41], the current flowing through each battery cell during CC phase is calculated as $I_{cc} = -\frac{125\mathrm{A}}{N_p} = -62.5\mathrm{A}$. To avoid over-charging the battery, U_{cv} is arbitrarily determined as 4.015V. Using Eq. (7) and Eq. (8), the charging pattern of the E-NV200's battery pack can be obtained as shown in Fig. 5. With the 50kW DC fast charger, the battery pack takes about 25 minutes to recharge from 0% to 80%, which is close to 30 minutes as stated in [34]. And the SoC of a fully-charged battery is 94.57% (i.e. $\mathrm{SoC}_{\max} = 94.57\%$).

C. Experimental Settings

The general settings for all the D-EVRP instances are given in Table. IV. Since D-EVRP instance may have a variety of solution policies for different customer demands $\mathcal{D}_{\mathrm{ini}}$, the demand of each customer $D_1^i \in \mathcal{D}_{\mathrm{ini}}$ is assumed to follow a discrete uniform distribution $U(100, 400)$, and for each D-EVRP instance, 10 sets of customer demands are randomly generated. Given a specific $\mathcal{D}_{\mathrm{ini}}$, any D-EVRP instance can be solved with a solution policy, the performance of which is justified upon 500 traffic samples that are randomly generated using historical traffic data. Note that the number of MCS episodes is selected as 50 for HRA, i.e. $\lambda = 50$.

TABLE IV
GENERAL SETTINGS OF D-EVRP INSTANCES

Symbols	D_1^i	$\mathrm{SoC}_{\min}$	Ψ	ω	β	γ
Values	$U(100, 400)$	10%	$\{0, 750, 1500\}$	1	99%	1
Units	kg	N/A	sec	N/A	N/A	N/A

For comparison, two counterparts of HRA, hybrid policy iteration (HPI) and hybrid base heuristic (HBH), are devised. For any given D-EVRP instance, HPI searches for the optimal solution policy by combining the pre-planning strategy with classic policy iteration; while HBH determines the corresponding base policy $\pi^{bs}(\cdot)$, which prescribes the EV to always visit the nearest node and charge for minimum period, with control set $\zeta(s_k)$ constrained by the pre-planning strategy. The instances in Case A are solved by all of the three algorithms, whereas those in Case B are tackled by HRA and HBH only, due to the dimensionality problem in HPI.

The cumulative transition cost of initial state $G(s_1)$ is considered as the primary performance metric, and the relative improvement of algorithm x over y is thus defined as

$$L(x, y) = \frac{G_y(s_1) - G_x(s_1)}{G_y(s_1)} \tag{24}$$

where $G_x(s_1)$ and $G_y(s_1)$ denote the average $G(s_1)$ over 500 traffic samples, when the control policies of algorithm x and y are respectively employed.

D. Simulation Results

1) Case A: Fig. 6 shows the performance comparison between HPI, HRA and HBH. As given in Fig. 6 (a), the average improvements of HPI and HRA against HBH, $\bar{L}(\mathrm{HPI}, \mathrm{HBH})$ and $\bar{L}(\mathrm{HRA}, \mathrm{HBH})$, are of about 10%. It is also noticed that the improvement for Instance 4 is much smaller (eg. $\bar{L}(\mathrm{HRA}, \mathrm{HBH}) = 5.64\%$). In this instance, customers' locations are highly aggregated at the downtown area, and hence the distances and traffic conditions of the routes among customers are quite similar. The overall service time for different paths may not introduce significant difference. As a result, the expected total service time is within a small range of variations, even with different solution policies.

The variance of improvements (i.e. $L(\mathrm{HRA}, \mathrm{HBH})$) for all instances of Case A is depicted in Fig. 6 (b). The variances are found to be only slightly larger than $L(\mathrm{HPI}, \mathrm{HBH})$ for most D-EVRP instances, reflecting HRA's robustness in addressing different D-EVRP instances with various customer demands. It is also observed that the variance of $L(\cdot, \mathrm{HBH})$ for Instance 5 is larger than others. According to the simulated results, $L(\mathrm{HRA}, \mathrm{HBH})$ ranges from 0.17% to 28.22% for different $\mathcal{D}_{\mathrm{ini}}$, showing that the improvement heavily depends on customers' demand. It is because, the impact of $\mathcal{D}_{\mathrm{ini}}$ is magnified by the widely dispersed customers, while the preference of visiting nearest node in HBH may perform sufficiently good under some conditions, making $L(\mathrm{HRA}, \mathrm{HBH})$ small.

Fig. 6 (c) shows the CPU time spent in each instance for different algorithms. Obviously, HBH spends very little CPU time (from 10 ms for simple cases to a few seconds for large cases), as the EV always goes to the nearest node, if possible. The CPU time spent in HRA is smaller than that in HPI. In particular for Instances 4 and 5, in which the state space cardinality is larger, HPI consumes excessive time while the CPU time spent in HRA remains at a low level.

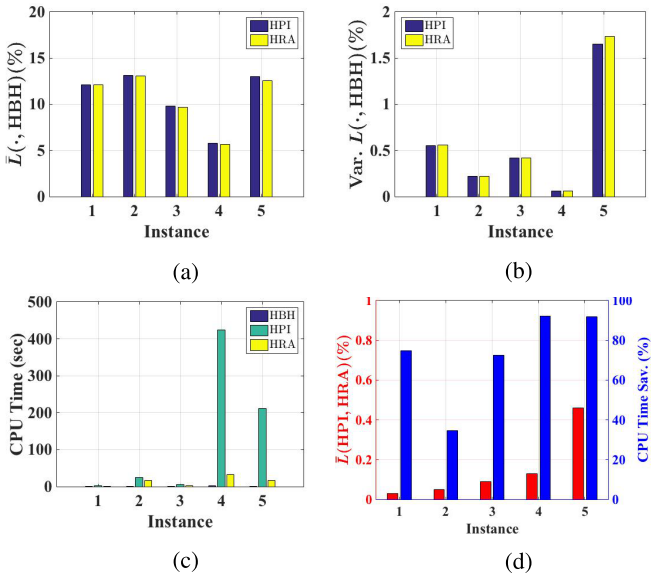
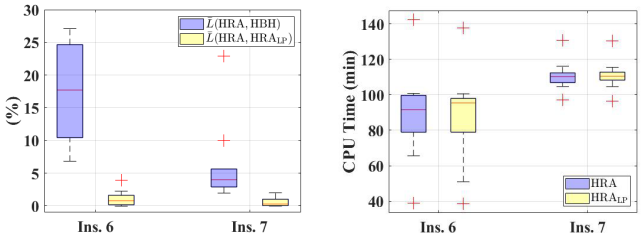


Fig. 6. Performance comparison among algorithms for different instances with 10 set of customer demands. (a) $\bar{L}(\cdot, \mathrm{HBH})$ (b) Variance of $L(\cdot, \mathrm{HBH})$ (c) CPU Time (d) $\bar{L}(\mathrm{HPI}, \mathrm{HRA})$ & Time Sav.

Fig. 6 (d) shows a direct comparison of HRA and HPI. As shown, $\bar{L}(\mathrm{HPI}, \mathrm{HRA})$ is very small, ranging from 0.03% to 0.46%. This confirms that HRA only performs slightly worse than HPI in all instances. On the other hand, HRA saves up more than 90% of CPU time in Instances 4 and 5. It is noted that the time saving is much lower in Instance 2 (which is 34.72%). In this instance, there are a lot of states (up to 1408 states, in average) to be evaluated by MCS, and all the redundant episodes facilitate efficiency degradation. However, the saving in CPU time may be improved by reducing the number of MCS episodes. For simple case like Instance 2, letting $\lambda = 20$, the saving of CPU time increases to 65.82% while the performance improvement is kept almost identical.

2) Case B: Fig. 7 summarizes HRA's overall performance on solving large-scale instances. In Fig. 7 (a), the box plot of $\bar{L}(\mathrm{HRA}, \mathrm{HBH})$ is given. The median of $\bar{L}(\mathrm{HRA}, \mathrm{HBH})$ for Instances 6 and 7 are 17.71% and 4.0%, respectively, confirming the effectiveness of HRA. It is remarked that the customers in Instance 6 are widely dispersed. Therefore, $\bar{L}(\mathrm{HRA}, \mathrm{HBH})$ of Instance 6 has much larger interquartile range as compared to that of Instance 7. The computational time for Instances 6 and 7 are given in Fig. 7 (b). It is interested to point out that, the CPU time for Instance 6 is less in general, even though it has five more nodes than Instance 7. It is because the nodes in Instance 7 are much closer to each other, which increases the eligible routing actions at each decision stage. Accordingly, there are more successive states, while each requires λ episodes of rollout, demanding for more computational time. Moreover, the average $G(s_1)$ are



(a) (b)

Fig. 7. Performance of HRA on large-scale instances. (a) $\bar{L}(\text{HRA}, \cdot)$ (b) CPU Time.

101.92 and 92.68 which are equivalent to 8.49 and 7.72 hours to serve the customers in Instances 6 and 7, respectively. The results justify that they are practically large cases.

3) Comparison of Battery Models: We also studied the effectiveness of the analytical battery model in use as compared to a linear approximation (LP) model, which fits the first and last data points of the nonlinear function. To better illustrate the result, $L(\text{HRA}, \text{HRA}_{\text{LP}})$ is adopted to quantify the extent of performance degradation due to the linear approximation. It is observed that the maximum and the mean of $L(\text{HRA}, \text{HRA}_{\text{LP}})$ for Instance 6 are 3.89% and 1.11%, while those for Instance 7 are 2.01% and 0.63%. Intuitively, the solution policies based on LP model are more conservative since the duration required for charging a battery is overestimated. As a result, the EV intends to charge more to prevent power depletion at each decision stage, which in turns, leads to the observed performance degradation. On the other hand, there is very little difference between the CPU time for HRA and HRA_{LP}. Though the performance degradation caused by the linear approximation of battery model may not be serious, but considering the very little additional time cost, it is still recommended to adopt the analytical battery model to facilitate the search of a better solution policy.

V. Conclusion

To expedite the profitable employment of EV in logistics industry, a dynamic electric vehicle routing problem (D-EVRP) is suggested for the logistical planning of EV. The problem is established based on a finite-horizon Markov decision process (MDP) formulation. In addition to the state-of-art constraints in EV routing problem (E-VRP), D-EVRP further considers time-dependent stochastic traffic, and also incorporates an analytical battery model to better capture the nonlinear dynamics of EV's battery pack. As a MDP, D-EVRP aims to minimize the overall service duration. Its sequential decision-making lies in the selection of both successive path and charging duration, which makes it computationally intractable if the instance scale is large. Therefore, a hybrid rollout algorithm (HRA), composed of a pre-planning strategy and rollout algorithm, is developed to address the D-EVRP. The algorithm is applied to multiple D-EVRP instances based on the delivery task of a logistics company in Singapore using Nissan's E-NV2000 electric van. Simulation results confirm that HRA is not only effective, but also robust in solving the studied instances with near-optimal policies, and the quality of which also evidently benefits from accurate modeling of battery dynamics.

References

[1] *These Countries Want to Ban Gas and Diesel Cars*. Accessed: Jan. 12, 2018. [Online]. Available: http://money.cnn.com/2017/09/11/autos/countries-banning-diesel-gas-cars/index.html

[2] W. Su, H. Eichi, W. Zeng, and M.-Y. Chow, "A survey on the electrification of transportation in a smart grid environment," *IEEE Trans. Ind. Informat.*, vol. 8, no. 1, pp. 1–10, Feb. 2012.

[3] Y. Zheng, Z. Y. Dong, Y. Xu, K. Meng, J. H. Zhao, and J. Qiu, "Electric vehicle battery charging/swap stations in distribution systems: Comparison study and optimal planning," *IEEE Trans. Power Syst.*, vol. 29, no. 1, pp. 221–229, Jan. 2014.

[4] P. Sadeghi-Barzani, A. Rajabi-Ghahnavieh, and H. Kazemi-Karegar, "Optimal fast charging station placing and sizing," *Appl. Energy*, vol. 125, pp. 289–299, Jul. 2014.

[5] X. Wang, C. Yuen, N. U. Hassan, N. An, and W. Wu, "Electric vehicle charging station placement for urban public bus systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 1, pp. 128–139, Jan. 2017.

[6] G. Wang *et al.*, "Robust planning of electric vehicle charging facilities with an advanced evaluation method," *IEEE Trans. Ind. Informat.*, vol. 14, no. 3, pp. 866–876, May 2018.

[7] Y. Xiong, J. Gan, B. An, C. Miao, and A. L. C. Bazzan, "Optimal electric vehicle fast charging station placement based on game theoretical framework," *IEEE Trans. Intell. Transp. Syst.*, no. 99, pp. 1–12, Oct. 2017.

[8] M. Strehler, S. Merting, and C. Schwan, "Energy-efficient shortest routes for electric and hybrid vehicles," *Transp. Res. B, Methodol.*, vol. 103, pp. 111–135, Sep. 2017.

[9] C.-K. Chau, K. Elbassioni, and C.-M. Tseng, "Drive mode optimization and path planning for plug-in hybrid electric vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 12, pp. 3421–3432, Dec. 2017.

[10] A. Bourass, S. Cherkaoui, and L. Khoukhi, "Secure optimal itinerary planning for electric vehicles in the smart grid," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3236–3245, Dec. 2017.

[11] H. Yang, Y. Deng, J. Qiu, M. Li, M. Lai, and Z. Y. Dong, "Electric vehicle route selection and charging navigation strategy based on crowd sensing," *IEEE Trans. Ind. Informat.*, vol. 13, no. 5, pp. 2214–2226, Oct. 2017.

[12] J. Guanetti, S. Formentin, and S. M. Savaresi, "Energy management system for an electric vehicle with a rental range extender: A least costly approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 11, pp. 3022–3034, Nov. 2016.

[13] J. Shen and A. Khaligh, "Design and real-time controller implementation for a battery-ultracapacitor hybrid energy storage system," *IEEE Trans. Ind. Informat.*, vol. 12, no. 5, pp. 1910–1918, Oct. 2016.

[14] X. Qi, G. Wu, K. Boriboonsomsin, and M. J. Barth, "Development and evaluation of an evolutionary algorithm-based online energy management system for plug-in hybrid electric vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2181–2191, Aug. 2017.

[15] Q. Zhang, W. Deng, and G. Li, "Stochastic control of predictive power management for battery/supercapacitor hybrid energy storage systems of electric vehicles," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 3023–3030, Jul. 2018.

[16] S. Huang, L. He, Y. Gu, K. Wood, and S. Benjaafar, "Design of a mobile charging service for electric vehicles in an urban environment," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 787–798, Apr. 2015.

[17] C. D. Korkas, S. Baldi, S. Yuan, and E. B. Kosmatopoulos, "An adaptive learning-based approach for nearly optimal dynamic charging of electric vehicle fleets," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2066–2075, Jul. 2018.

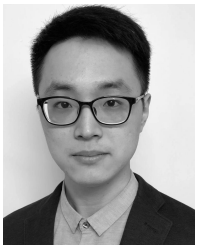
[18] Y. Nie, X. Wang, and K.-W. E. Cheng, "Multi-area self-adaptive pricing control in smart city with EV user participation," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2156–2164, Jul. 2018.

[19] S. Pelletier, O. Jabali, and G. Laporte, "50th anniversary invited article—Goods distribution with electric vehicles: Review and research perspectives," *Transp. Sci.*, vol. 50, no. 1, pp. 3–22, 2016.

[20] A. Montoya and C. Guéret, J. E. Mendoza, and J. G. Villegas, "The electric vehicle routing problem with nonlinear charging function," *Transp. Res. B, Methodol.*, vol. 103, pp. 87–110, Sep. 2017.

[21] M. Schiffer and G. Walther, "The electric location routing problem with time windows and partial recharging," *Eur. J. Oper. Res.*, vol. 260, no. 3, pp. 995–1013, 2017.

[22] G. Hiermann, J. Puchinger, S. Ropke, and R. F. Hartl, "The electric fleet size and mix vehicle routing problem with time windows and recharging stations," *Eur. J. Oper. Res.*, vol. 252, no. 3, pp. 995–1018, 2016.
[23] J. Hof, M. Schneider, and D. Goeke, "Solving the battery swap station location-routing problem with capacitated electric vehicles using an AVNS algorithm for vehicle-routing problems with intermediate stops," *Transp. Res. B, Methodol.*, vol. 97, pp. 102–112, May 2017.
[24] G. Desaulniers, F. Errico, S. Irnich, and M. Schneider, "Exact algorithms for electric vehicle-routing problems with time windows," *Oper. Res.*, vol. 64, no. 6, pp. 1388–1405, 2016.
[25] H. Jula, M. Dessouky, and P. A. Ioannou, "Truck route planning in nonstationary stochastic networks with time windows at customer locations," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 51–62, Mar. 2006.
[26] G. Kim, Y. S. Ong, T. Cheong, and P. S. Tan, "Solving the dynamic vehicle routing problem under traffic congestion," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2367–2380, Aug. 2016.
[27] M. Keskin and B. Çatay, "Partial recharge strategies for the electric vehicle routing problem with time windows," *Transp. Res. C, Emerg. Technol.*, vol. 65, pp. 111–127, Apr. 2016.
[28] Á. Felipe, M. T. Ortuño, G. Righini, and G. Tirado, "A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges," *Transp. Res. E, Logistics Transp. Rev.*, vol. 71, pp. 111–128, Nov. 2014.
[29] S. Pelletier, O. Jabali, G. Laporte, and M. Veneroni, "Battery degradation and behaviour for electric vehicles: Review and numerical analyses of several models," *Transp. Res. B, Methodol.*, vol. 103, pp. 158–187, Sep. 2017.
[30] O. Tremblay, L.-A. Dessaint, and A.-I. Dekkiche, "A generic battery model for the dynamic simulation of hybrid electric vehicles," in *Proc. IEEE Vehicle Power Propuls. Conf. (VPPC)*, Sep. 2007, pp. 284–289.
[31] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Hoboken, NJ, USA: Wiley, 2007.
[32] D. P. Bertsekas, J. N. Tsitsiklis, and C. Wu, "Rollout algorithms for combinatorial optimization," *J. Heuristics*, vol. 3, no. 3, pp. 245–262, 1997.
[33] *ChargeNow Singapore*. Accessed: Jan. 22, 2018. [Online]. Available: http://chargenow.greenlots.com/
[34] *Brochure for Nissan E-NV200*. . Accessed: Jan. 22, 2018. [Online]. Available: https://www.nissan-cdn.net/content/dam/Nissan/gb/brochures/Nissane-NV200van.pdf
[35] *Github—Googlemaps/Google-Maps-Services-Java: Java Client Library for Google Maps Api Web Services*. . Accessed: Jan. 22, 2018. [Online]. Available: https://github.com/googlemaps/google-maps-services-java
[36] G. Wager, J. Whale, and T. Braunl, "Driving electric vehicles at highway speeds: The effect of higher driving speeds on energy consumption and driving range for electric vehicles in Australia," *Renew. Sustain. Energy Rev.*, vol. 63, pp. 158–165, Sep. 2016.
[37] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.
[38] *Cell, Module, and Pack for EV Applications*. Accessed: Feb. 9, 2018. [Online]. Available: http://www.eco-aesclb.com/en/product/liionev/
[39] J. Asamer, A. Graser, B. Heilmann, and M. Ruthmair, "Sensitivity analysis for energy demand estimation of electric vehicles," *Transp. Res. D, Transport Environ.*, vol. 46, pp. 182–199, Jul. 2016.
[40] *Model S Efficiency and Range*. Accessed: Jan. 22, 2018. [Online]. Available: https://www.tesla.com/enHK/blog/models-efficiency-and-range?redirect=no
[41] *Chademo—Wikipedia*. Accessed: Apr. 5, 2018. [Online]. Available: https://en.wikipedia.org/wiki/CHAdeMO



Xiaowen Bi received the B.S. degree from Shandong University, China, in 2014, and the M.S. degree from the City University of Hong Kong, Hong Kong, in 2015, where he is currently pursuing the Ph.D. degree. His research interests include computational intelligence and complex networks.

Wallace K. S. Tang (M'96–SM'09) received the Ph.D. degree from the City University of Hong Kong in 1996. He is currently an Associate Professor with the Department of Electronic Engineering, City University of Hong Kong. He has published over 100 journal papers, eight book chapters, and three books, focusing on nonlinear systems, optimization and complex networks. He has been an Associate Editor of the *International Journal of Bifurcation and Chaos* since 2012. He was an Associate Editor of the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, from 2013 to 2015, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, from 2016 to 2017.