Leveraging protein-protein interactions in phenotype prediction through graph neural networks

Battery-Aware Electric Truck Delivery Route Planner

Donkyu Baek, Yukai Chen,
Enrico Macii, Massimo Poncino
Dept. of Control and Computer Enginnering
Politecnico di Torino, Italy
{name.surname@polito.it}

Naehyuck Chang
School of Electrical Engineering
Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea
naehyuck@cad4x.kaist.ac.kr

***Abstract*---Finding the energy-optimal route in the context of parcel delivery with electric vehicles (EVs) is more complicated than for conventional internal combustion engine (ICE) vehicles, where the energy cost of a path is mostly determined by the total traveled distance. In the case of EV delivery, the total energy consumption strongly depends on the order of delivery because the efficiency of the EV is affected by how the transported weight changes over time as it directly affects the battery efficiency. This makes impossible to find an optimal solution using traditional routing algorithms such as the traveling salesman problem (TSP) using a static quantity (e.g., distance) as a metric.**

In this paper, we propose a solution for the least-energy delivery problem using EVs; we implement an electric truck simulator and evaluate different static metrics to assess their quality on small size instances for which the optimal solution can be computed exhaustively. A greedy algorithm using the empirically best metric (namely, distance $\times$ residual weight) provides significant reductions (up to 33%) with respect to a common-sense heaviest first package delivery route determined using a metric suggested by the battery properties, and is sensibly faster than state-of-the-art TSP heuristic algorithms.

***Index Terms*---Electric truck simulator, traveling salesman problem, least-energy routing algorithm, metric evaluation**

I. Introduction

Electric vehicles (EVs) are becoming increasingly popular and are expected to progressively replace traditional internal combustion engine vehicles (ICEVs). EVs have high energy efficiency and do not emit greenhouse gas (GHG); even if when accounting the emission incurred during electricity production for the charge of EVs, their overall GHG emission is up to 58% lower than the emissions of an average mid-size passenger ICEVs [1]. Also, the impact on climate change by the production of electricity and operation of EVs is less than up to 30% compared with ICEVs when considering the average generation of electricity in Europe. Recently, the landscape of EVs is widening and extends to domains such as electric racing cars, electric buses, and electric trucks.

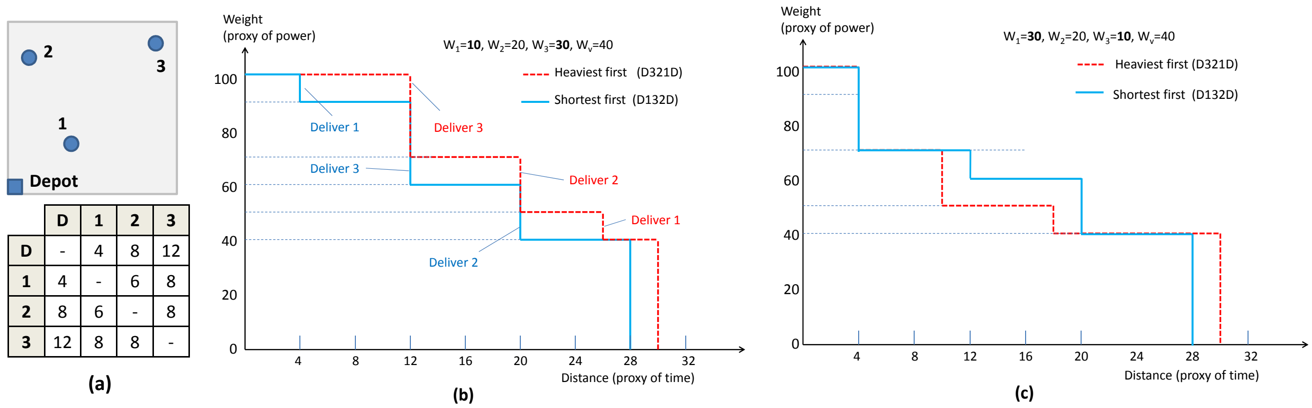
In particular, electric trucks will replace existing ICE trucks in the future as Tesla announced [2]. The electric truck can accelerate faster than conventional diesel trucks because of the characteristic of the electric motor: high torque at low rotations per minute (RPM). In addition, 98% of the kinetic energy can be replaced with electric energy during regenerative braking, which makes the electric truck more energy efficient. According to the announcement by Tesla, electric truck owners can save more than $200,000 over a million miles based on fuel costs alone.

This intrinsic energy efficiency can be further improved by finding the optimal delivery route; an electric truck loads all packages for customers at a depot, visits each customer to deliver their package, and then returns to the depot without payload. For a conventional ICEV, the ''cost'' of a path is mostly driven by the distance (even if weight also matters) and the problem nicely fits into the well-known traveling salesman problem (TSP) using distance as a metric.

However, when considering EVs, the solution is not as straightforward; the total energy consumption strongly depends on the order of delivery as the efficiency of the EV is affected by the total (vehicle + payload) weight. As a matter of fact, one key characteristic of a battery is that it is progressively less efficient in delivering its energy as its state of charge (SoC) decreases [3], [4]. A fully charged battery is more efficient to deliver a high current demand than when it is partially discharged. As the power consumption of the electrical motor depends on the total weight, then apparently if we deliver the heaviest package first, the overall vehicle weight is reduced the most after unloading this package and following such order would be optimal. On the other hand, also distance should be considered; if we deliver the heaviest package first and this corresponds a very long distance from the depot, we will discharge the battery by driving the heaviest weight for a long time.

One first difference with respect to a plain ICEv delivery is therefore in terms of metrics: for EVs, some combination of weight and distance should be considered. But the most significant difference (and complication) lies in the fact that the calculation of the optimal energy path cannot be done *incrementally*, as the energy cost of a path is ''dynamic'', i.e., it depends on the previous choices as a consequence of the dependence on the residual weight.

In this paper, we propose the overall framework for the least-energy electric truck delivery problem. We first implement an electric truck simulator with a powertrain model and a non-linear battery model of the Tesla Semi [5] in order to predict the change of SOC during the package delivery. From the simulator, we show that a conventional metric for TSP,



	D	1	2	3
D	-	4	8	12
1	4	-	6	8
2	8	6	-	8
3	12	8	8	-

Fig. 1: Motivational Example.

total delivery distance, like any other ''static'' metric, does not minimize total energy consumption for an EV delivery. As only an exhaustive exploration of all path guarantees to find the optimal path, we evaluate different static metrics (functions of weight and distance) on small graph instances to assess their quality; then using the best metric derived in this calibration phase we show how a greedy algorithm using that metric provides significant reductions (up to 33%) with respect to the common-sense heaviest first package delivery.

II. Motivation

We have built a small motivating example in order to show how it is not possible to build an energy-optimal delivery schedule using a ''static'' metric. Fig. 1(a) shows a simple 3-destination delivery task from a depot (D) with a rough mapping on the plane, and the distance matrix between any pair of destinations; without loss of generality we have assumed symmetric between node pairs.

In order to assess the energy cost of a delivery path, we will use a time diagram that plots the evolution of the total transported weight over traveled distance. Weight can be used as a proxy of power consumption as the electric motor power is proportional to the total weight of the vehicle plus the payload. This is clearly a simplification and does not take into account all the non-linearities of a battery, but even this approximation helps showing the point we are making.

Distance is used as a proxy of time, assuming a constant speed for the deliveries. Again, this is an approximation of the real setting, where speed can be extremely varying. When using a real battery model in the loop, however, the real speed profile of the vehicle can be accounted for. Therefore, evolution of weight over distance is a proxy of power over time, and the area of one such curve is then an estimate of the energy consumed for that delivery route.

Fig. 1(b) shows two such delivery routes for a case in which the weight are $W_1 = 10, W_2 = 20, W_3 = 30$ and vehicle weight is $W_v = 40$. The dotted red curve represents a route for which packages are delivered in heaviest-first order ($D \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow D$), whereas the solid blue line denotes a route with packages delivered in increasing order of distances ($D \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow D$). In this specific case the ''shortest-first'' policy works best (smaller area under the curve) and, by exhaustive exploration of the 3!=6 combinations of deliveries it can be shown to yield the best value of the metric.

Fig. 1(c) shows two other waveforms for the same delivery task in which the weights are now $W_1 = 30, W_2 = 20, W_3 = 10$, that is, in which the heaviest package corresponds to the closest destination (node 1). In this case the red dotted profile of the ''heaviest-first'' yields the best value of the metric, while the ''shortest-first'' yields a slightly worse value. Notice that the blue solid line corresponds to the same order of delivery (yet with a different cost) as in Fig. 1(b) as the distance has not changed in the two examples.

Notice also that (due the symmetric distances) there are two paths (e.g., $D \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow D$ and $D \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow D$) with the same distance but with different ''energy'' cost. Therefore, an algorithm that picks edge simply based on distance could even get farther from the optimal solution.

This example, yet in the presence of a number of approximations, shows the main two points raised by our work. First, no simple static metric can solve optimally the problem of finding the energy-optimal delivery route.

Secondly, due to the state-dependent characteristic of the cost function, only an exhaustive exploration can find the optimal solution. Since this is not feasible but for very small instances, we need to find a provably good static metric that can be used in a heuristic algorithm. From our motivating example, this static metric should combine weight *and* distance as both affect the energy consumption of the EV.

III. Background and Related Work

A. Battery Model

The model of the battery pack requires then a model for the individual cell, and the model must be able to accurately account for the load current and SOC variations of the usable battery capacity. A single cell in the pack can be modeled with a circuit-equivalent model that models the capacity dependency on current magnitude and dynamics [6], [7]. Fig. 2 depicts the circuit-equivalent model of a battery cell. It consists of a left-hand part for modeling the battery lifetime and a right-hand part represents the transient battery voltage. Notice that the left-part also account for current magnitude and frequency dependency on battery capacity. From this model we built the battery

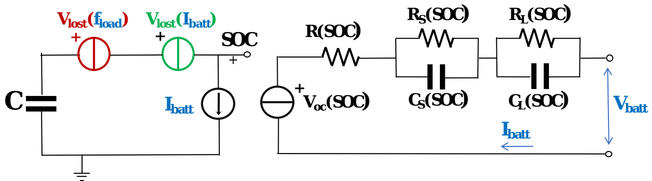


Fig. 2: Adopted circuit-equivalent model for battery cell.

pack model by simply scaling all parameters according to the series/parallel connection; although somehow ideal (e.g., cell mismatches are not considered) this is still more accurate than a linear model that neglects state-dependent battery characteristic. Given this model, we can track the energy consumed by the EV by applying to the model the drawn power (as current and voltage waveforms) corresponding to the electrical motor consumption on a given leg of the route. In the most general case there will a non-ideal power conversion step between the electrical motor and the battery. In this case, it suffices to scale the motor current and voltage according to the converter efficiency $\eta < 1$, which can be any complex function of the motor parameters, i.e., $P_{batt} = P_{motor} \cdot \eta$.

B. Vehicle Routing Problem

The vehicle routing problem is formulated as a graph $G(V, E, C)$ where $V = \{v_0, \cdots, v_N\}$ is the set of vertices including N destinations and a depot, $E = \{e_{ij}|_{i,j \in V}\}$ is the set of edges between two vertices v_i and v_j, and $C = \{c_{ij}|_{i,j \in V}\}$ is the cost related to each edge e_{ij}. Vertex v_0 is the depot, while the remaining vertices in V represent customers that need to be served. The TSP consists in finding a route based at the depot, such that each of the vertices is visited exactly once while minimizing the overall routing cost.

The formulation of the vehicle routing problem is generalized as the TSP. Because TSP is known as NP-hard, several approximation algorithms are proposed during several decades. Christofides designed an approximation algorithm for TSP using the minimum spanning tree (MST) algorithm, which obtains approximated results less than 1.5 times of the optimal solution [8]. From the general TSP, there are several variants of TSP to consider various constraints and delivery requirements. There is a variant of TSP considering a set of potential customers living near secured customers [9]. The salesman finds the shortest path to cover all potential customers within a certain distance from the path. A fleet of delivery vehicles characterized by different capacities and costs is an important variant of TSP [10], [11]. There is a set of customers and a set of different types of vehicles. Each vehicle has different capacity in terms of the number of customers and operation cost; the goal is to find a set of routes for each vehicle minimizing total delivery cost. Authors consider the number of customers that each vehicle should be responsible for, but they do not consider the vehicle weight changing with each delivery.

Recently, the vehicle routing problem with pick-up and delivery considers the situation in which packages have to be picked-up from one of customers and delivered to another location [12], [13]. During the pick-up and delivery process, visiting each pickup and delivery places occurs exactly once and total package weight during the delivery should not exceed the capacity. This problem considers the weight of each package, however, does not consider the energy consumption that changes after unloading each package.

IV. Analysis of Routing Algorithms

The examples of Fig. 1 suggest that a good way to track the actual energy spent on a delivery path (modulo the non-idealities of the battery) would be to use a metric that is correlated to distance $\times$ weight. More precisely, said i and j the two vertices, d_{ij} the distance between them and W_i and W_j the respective weights, the metric should be proportional to $d_{ij} \times (W_{current} - W_i)$, assuming the edge is travelled in the direction $i \rightarrow j$. $W_{current}$ is the current weight of the vehicle when reaching node i. We call the above metric DxW_r (i.e., distance times residual weight).

Thus, intuitively, as the delivery problem is an instance of TSP, one could be tempted to run some TSP heuristic algorithm using the above cost instead of using distance as in traditional TSP instance. Although approximate (even without considering battery non-idealities), this strategy will leverage well-consolidated heuristics for the solving the TSP.

There is a however a subtlety in this argument. State-of-the-art TSP heuristics rely on the calculation of the MST algorithm as a pre-processing. This is essentially because the cost of a MST is the simplest lower bound for the TSP. As a matter of fact, Since removal of one edge from any Hamiltonian cycle (i.e., a solution of the TSP) yields a spanning tree [8]. MST algorithms systematically grow the tree by greedily picking edges in increasing order of the cost function; this implies that the above cost function $d_{ij} \times (W_{current} - W_i)$ cannot be used, for two reasons. First, there is no equivalent of $W_{current}$, since we are not yet building a path; second, MST runs on an undirected graph and there is no intuition about in what direction the edge is traversed.

A possible approximation of the DxW_r metric suitable for a TSP that starts from a MST would be (i) to assume a 50% chance of travelling the edge in each direction and (ii) approximate $W_{current}$ with the total weight (vehicle + payload). This would yield the metric that we call DxW:

$$d_{ij} \times (W_{total} - (W_i + W_j)/2)$$

which allows to be used in a MST-based TSP heuristic.

The computational complexity of the TSP heuristic that provides the best approximation, i.e., Christofides' algorithm, is $O(n^3)$, where n is the number of vertices, assuming that the graph is fully connected [8].

Given the number of approximation a TSP-based solution would incur (intrinsic approximation of the algorithm plus that of the metric, plus the fact that battery properties are not incorporated), even though its complexity is polynomial, it could make sense to devise an alternative and simpler *greedy* algorithm that builds up the cycle *as a path, one edge at a time starting from the depot*. This choice would allow one to use the DxW_r metric that more closely tracks the energy value; by forming a path, in fact, we can calculate the equivalent of

$W_{current}$ for the path being built. Moreover, since we start from the depot node, edges have an implicit direction and DxW_r can be calculated correctly. The approximation lies obviously in the fact that the greedy solution is not optimal, and unlike the TSP heuristic, the approximation cannot be bounded. Christofides' algorithm, for example, can be shown to yield a solution that is no more than 3/2 of the optimal cost. The greedy heuristic would clearly be linear in the number of nodes. Should it be roughly as approximate as the TSP heuristic, it would at least guarantee that it can handle larger problem instances.

In our analysis we will therefore compare three classes of algorithms to solve the optimal routing problem:

1) a set of algorithms based on enumerating all paths (and therefore feasible only for small instances) using different metrics and used for evaluating the quality of the approximations;
2) heuristic TSP algorithms using different metrics
3) heuristic greedy algorithms using different metrics

V. Simulation Results

A. Simulation Setup

1) Powertrain model: We implemented a powertrain model of a Tesla Semi truck from the vehicle specification based on the presentation by Elon Musk; this is currently the only source of information for the specs as Tesla is preparing to release the Semi in 2019 or later [14], [5]. The powertrain consists of four Model 3 electric motors; each motor is 3-phase AC permanent magnet electric motor with maximum power of 192 kW from 4700 to 9000 RPM, and maximum torque is 410 Nm below 4500 RPM, respectively [15], [16]. We estimate curb weight of Semi as the sum of typical weight of class 8 truck and battery pack weight [17].

The powertrain model P_{EV}, a function of torque T and angular speed ω, is defined in [18] to consider both dynamics and loss:

$$P_{dyna} = T\omega = Fds/dt = (F_R + F_G + F_I + F_A)v$$
$$P_{EV} = P_{dyna} + C_0 + C_1 v + C_2 v^2 + C_3 T^2$$
$$P_{regen} = \epsilon T v + \zeta.$$

There are four resistances acting on a vehicle P_{dyna}: rolling resistance F_R, gradient resistance F_G, inertia resistance F_I, and aerodynamic resistance F_A. Coefficients C_0, C_1, C_2, and C_3 are for constant loss, iron and friction loss, drivetrain loss, and copper loss, respectively. Regenerative braking power P_{regen} is a function of negative torque and speed.

We first implemented a vehicle model in ADVISOR (ADvanced VehIcle SimulatOR) [19] by using the above vehicle specification. Then, we extracted the coefficients of EV powertrain model with a number of ADVISOR simulations as described in [20]. Table I summarizes the model coefficients of Tesla Semi. Fig. 3 shows the difference between the estimation of power consumption by the vehicle simulator and the powertrain models; the normalized root-mean-square error is 4.93%.

TABLE I: Model coefficients for Tesla Semi truck.

α	0.098	β	10.1522	γ	1.006	δ	2.5e-5
C_0	10000	C_1	0.03	C_2	0.02598	C_3	1.54e-5
ϵ	0.5912	ζ	0.0				

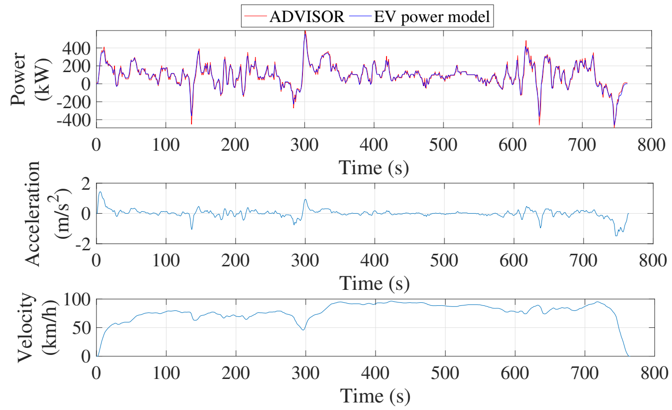


Fig. 3: Powertrain model validation result.

2) Battery pack model: In our experiment, we assume that each electric motor is connected to each battery pack of Model 3. Each battery pack is composed of 4 modules that are connected in series; each module consists of Panasonic NCR18650B $3400mAh$ Lithium battery cells arranged in a 46p24s configuration [21]. Table II summarizes the physical electrical parameters of each cell, each module and the whole battery pack.

TABLE II: Electrical parameters of the battery pack.

Parameters	Cell	Module	Whole Pack
Nominal Capacity	3400 mAh	156.4 Ah	156.4 Ah
Nominal Voltage	3.6 V	86.4 V	345.6 V
Cut-off voltage	2.75 V	66.0 V	264.0 V

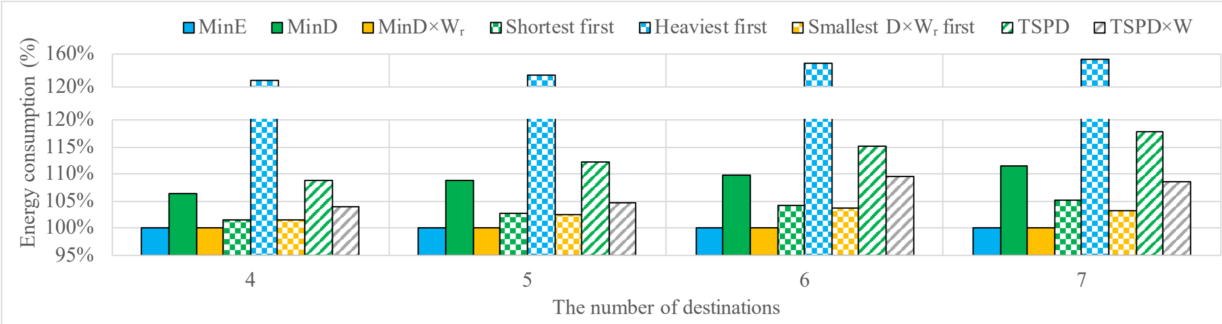
We built our battery single cell model based on the measurement data by adopting the method described in [6]. We assumed such $7,104$ battery cells in the pack to be ideally balanced in the following experiments, then built battery pack model as section III-A indicated. Concerning the regenerative braking phase, we assumed that regenerative charging efficiency is 20% in our simulation, i.e., 20% of the kinetic energy is converted to electric energy and transferred into the battery pack.

B. Simulation Results

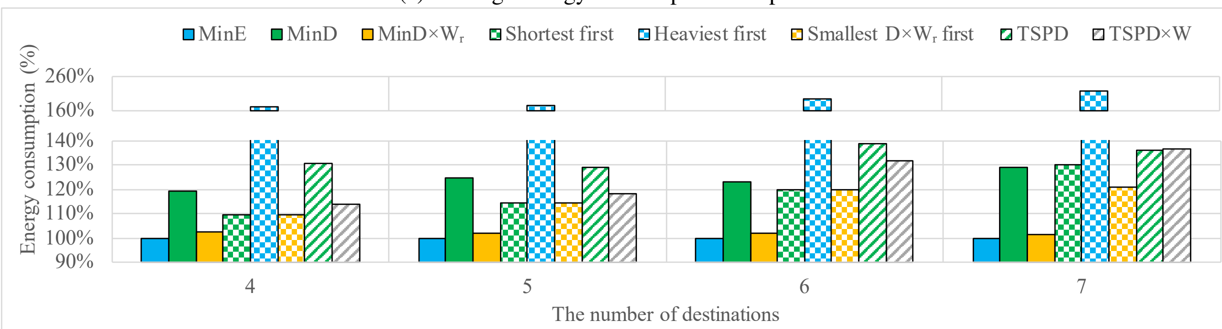
1) Comparison Against Exact Results: In this section, we compare the energy consumption of various delivery strategies based on different policies for a set of small-sized (4, 5, 6, and 7) instances for which an exhaustive exploration of all the possible delivery paths is feasible.

For each number of destinations, we randomly generated 50 instances with different distributions of locations of the depot and of the destinations by uniformly distributing them in a 30x30 km area. We selected the area for the delivery so that all the delivery sequences can be completed without exhausting the battery energy before returning to a depot. For each of the 50 instances, package weights for each destination have been chosen as uniformly distributed from 0.1 ton to 3 ton.

For each problem instance (destination and weight distribution) we calculate by exhaustive exploration the route yielding the smallest value of energy for a number of metrics. Energy has been calculated by applying the battery pack model fed by the power profile generated by the EV model of Section V-A. In this test example, a constant speed of 76 km/h has been



(a) Average energy consumption comparison.



(b) Worst energy consumption comparison.

Fig. 4: Energy consumption for different metrics (exhaustive exploration).

assumed, which is an average truck speed on metropolitan area interstates in the US, 2015 [22].

TABLE III: Exhaustive exploration of paths: List of algorithms.

Name	Description
Min$D$	paths are sorted in order of total length, and the shortest path is selected.
Min$D$xW_r	paths are sorted in order of total $D \times W_r$ (as described in Section IV) and the path with the smallest aggregate value is selected.
Heaviest first	paths are built by greedily choosing vertices in decreasing order of weights.
Shortest first	paths are built by greedily picking edges **starting from the depot node** in increasing order of distance.
Smallest DxW_r first	paths are built by picking edges **starting from the depot node** in increasing order of $D \times W_r$.
TSPD	TSP heuristic algo using distance as a metric.
TSPDxW	TSP heuristic algo using DxW as a metric.

The latter are grouped into two classes separated by the double line in the table; the first set use path-based metrics (i.e., aggregating a given metric along the path) and can be therefore only evaluated using an exhaustive exploration. In the second set the metrics are used to greedily build the path based on the given metric, as explained in Section IV.

Fig. 4(a) shows the energy consumption of the optimal route, averaged over the 50 instances, for problems with 4, 5, 6, and 7 destinations and for the set of algorithms described in Table III. The leftmost blue bars represent the optimal routes yielding the minimum energy consumption among all routes, obtained by computing the actual energy consumption per each segment using the battery model. This is the reference value against which the other results are compared. All the other bars refer to solutions (i.e. routes) returned by the above algorithms and *evaluated using the battery model*. The objective of the simulation is to check how the greedy algorithms (TSP-based or not) differ from the optimal solution and how the error increases with increased problem sizes. Bars in the plot are in the same order as in Table III. For ease of reading, bars referring to path-based algorithms (first part of the table) are shown as solid bars, whereas those referring to greedy or TSP algorithms are shown as patterned bars.

Concerning path-based algorithms, we can notice how the Min$D$xW_r metric (third bar from left) tracks very well the true energy value, much better than distance alone (second bar from left).

Concerning approximation algorithms, as a first general comment we can see that all algorithms do overestimate the actual energy consumption. Then, we immediately observe that weight alone (Heaviest-first) track quite badly the actual consumed energy, somehow contradicting the intuition suggested by the battery property; the distance from the reference is already $> 20\%$ even for the 4-destination instance. Although the actual error may differ depending on the weight distribution (as shown in Fig. 1), the results average 50 different runs so we can safely assume this is not a good metric. Notice also that the tracking error increases with larger instances.

Another observation is that a traditional TSP with distance metric (second bar from the right) performs reasonably only for the smallest instance; that average error increases quickly and is already around 18% for 7 destinations. Therefore, we can also rule out this algorithm from the list.

The remaining ones (Smallest DxW_r first, Shortest first, and TSPDxW) have errors below 10%, with the greedy algorithms being below 5% and scaling better with problem size than

TSPDxW.

Fig. 4(b) shows the worst case error among the 50 instances for the same set of algorithms. Results are consistent with average error, with the maximum error significantly larger than the average one. The greedy algorithms have show again the best results, both in terms of error and scalability. The smallest DxW_r is the only algorithm with worst-case error around 20% (as opposed to about 30%-35% of the others) for the 7-node case.

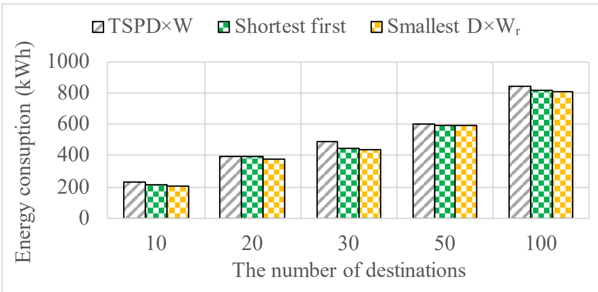


Fig. 5: Greedy and approximate TSP algorithms on large problem instances.

2) Application to Larger-Scale Instances: We generated a number of instances with 10, 20, 30, 50 and 100 destinations; for each problem size we generated 20 random instances and collected the average value of energy and execution time. In all cases, weights have been scaled so that the delivery task could be completed. Fig. 5 compares the absolute energy values for the three competitive algorithms resulting from the previous section: TSPDxW) and the the two greedy heuristics (shortest first and smallest DxW_r first). From Fig. 4(a) we know that all approximations are overestimations, so we can assume that lower values of energy imply higher accuracy. In Fig. 5, the smallest DxW_r first shows the best results: its energy consumption 10% smaller than the TSP heuristic and 4% smaller than the shortest first one, for the larger 100-node instance.

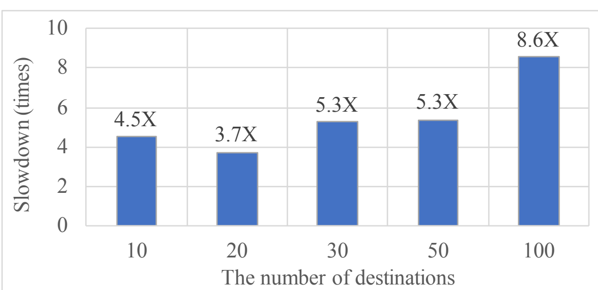


Fig. 6: Slowdown of TSP heuristics vs. greedy algorithm.

Fig. 6 shows the slowdown of the TSP heuristic with respect to the smallest DxW_r first algorithm. The TSP execution time is obviously independent of the metric used (D vs. DxW). The TSP heuristic is significantly slower that than the greedy method; the slowdown increases for increasing problem sizes, reaching 8.6x for the 100 destination case.

VI. Conclusions

The total energy consumption of an electric truck strongly depends on the order of delivery because the efficiency of the electric truck is affected by how the transported weight changes over time as it directly affects the battery efficiency. However, it is impossible to find an optimal solution using traditional routing algorithms using ''static'' metrics such as distance. In this paper, we demonstrate that the functions of weight and distance as metrics provide significant reductions with respect to the traditional routing algorithms, and a greedy algorithm minimizing DxW_r shows almost 10x fast calculation than TSP heuristic method with better results.

References

[1] ''Electric vehicles from life cycle and circular economy perspectives,'' European Environment Agency (EEA), Tech. Rep., 2018.

[2] Tesla Press Information, https://www.tesla.com/presskit#semi.

[3] Y. Chen and et al., ''A systemc-ams framework for the design and simulation of energy management in electric vehicles,'' *IEEE Access*, vol. 7, pp. 25 779--25 791, 2019.

[4] D. Baek and et al., ''Systemc-ams simulation of energy management of electric vehicles,'' in *2018 IEEE International Telecommunications Energy Conference (INTELEC)*, Oct 2018, pp. 1--8.

[5] Tesla Semi Official Website, https://www.tesla.com/semi, Tesla.

[6] Y. Chen, E. Macii, and M. Poncino, ''A circuit-equivalent battery model accounting for the dependency on load frequency,'' in *Proceedings of the Conference on Design, Automation & Test in Europe (DATE)*, 2017.

[7] Y. Chen, D. J. Pagliari, E. Macii, and M. Poncino, ''Battery-aware design exploration of scheduling policies for multi-sensor devices,'' in *ACM/IEEE Great Lakes symposium on VLSI (GLSVLSI)*, 2018.

[8] N. Christofides, ''Worst-case analysis of a new heuristic for the traveling salesman problem,'' *Report 388, Graduate School of Industrial Administration, Carnegie Mellon University*, February 1976.

[9] A. Dumitrescu and J. S. Mitchell, ''Approximation algorithms for tsp with neighborhoods in the plane,'' *Journal of Algorithms*, vol. 48, no. 1, 2003, annual ACM-SIAM Symposium on Discrete Algorithms.

[10] B. R., B. M., and V. D., *Routing a Heterogeneous Fleet of Vehicles*. Springer, vol. 43.

[11] F. Li, B. Golden, and E. Wasil, ''A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem,'' *Computers & Operations Research*, vol. 34, no. 9, pp. 2734 -- 2742, 2007.

[12] G. Desaulniers, J. Desrosiers, A. Erdmann, M. M. Solomon, and F. Soumis, *9. VRP with Pickup and Delivery*, January 2002, pp. 225--242.

[13] M. Gansterer, M. Küçüktepe, and R. F. Hartl, ''The multi-vehicle profitable pickup and delivery problem,'' *OR Spectrum*, vol. 39, no. 1, pp. 303--319, Jan 2017.

[14] Meet the Tesla Semitruck, Elon Musk's most electrifying gamble yet, =https://www.tesla.com/presskit#model3, Wired, Nov. 2017.

[15] Tesla Model3 Technical Specifications and performance figures, http://www.zeperfs.com/en/fiche7083-tesla-model-3-75.htm, Zeperfs.

[16] Tesla Model 3: 2018 Motor Trend Car of The Year Finalist, https://www.motortrend.com/news/tesla-model-3-2018-car-of-the-year-finalist/, MotorTrend.

[17] ''Technologies and approaches to reducing the fuel consumption of medium and heavy-duty vehicles,'' National Academy of Sciences, Tech. Rep., March 2010.

[18] D. Baek and et al., ''Battery-aware operation range estimation for terrestrial and aerial electric vehicles,'' *IEEE Transactions on Vehicular Technology*, 2019.

[19] T. Markel, A. Brooker, T. Hendricks, V. Johnson, K. Kelly, B. Kramer, M. O'Keefe, S. Sprik, and K. Wipke, ''ADVISOR: a systems analysis tool for advanced vehicle modeling,'' *Journal of Power Sources*, vol. 110, no. 2, pp. 255 -- 266, 2002.

[20] D. Baek and N. Chang, ''Runtime power management of battery electric vehicles for extended range with consideration of driving time,'' *IEEE Transactions on Very Large Scale Integration Systems*, vol. 27, 2019.

[21] Tesla Model 3 & Chevy Bolt Battery Packs Examined, https://cleantechnica.com/2018/07/08/tesla-model-3-chevy-bolt-battery-packs-examined/, CleanTechnica.

[22] ''Freight facts and figures,'' U.S. Department of Transportation, 2017.