A Bellman-Ford Approach to Energy Efficient Routing of Electric Vehicles

Rami Abousleiman and Osamah Rawashdeh
Department of Electrical and Computer Engineering
Oakland University
Rochester, Michigan, 48326, USA
{abousleiman.rami@gmail.com and rawashd2@oakland.edu}

Abstract –Most experts foresee more demand for electric and plug-in electric vehicles. This demand is triggered by environmental concerns, energy dependency, and unstable fuel prices. Available vehicle routing algorithms are designed for fossil-fuelled vehicles. These algorithms optimize for the shortest distance or the shortest travel time between 2 points. Dijkstra or Dijkstra-like algorithms are mostly used for solving such optimization problems. Energy-efficient routing for electric vehicles, on the other hand, requires different approaches as it cannot be solved using Dijkstra or Dijkstra-like algorithms. Negative path costs generated by regenerative braking, battery power and energy limits, and vehicle parameters that are only available at query time, make the task of electric vehicle energy-efficient routing a challenging problem. In this paper, we present a solution approach to the electric vehicle energy efficient routing problem using Bellman-Ford. Bellman-Ford is a deterministic optimization method that is capable of solving routes with negative paths. A model representing electric vehicles is presented. Bellman-Ford search is then applied on the model and is used to find the most energy efficient route. The generated solution is then used to guide the electric vehicle through the desired path. The performance of the Bellman – Ford algorithm is then studied by applying the implemented algorithm on different map sizes.

Index Terms – Determinist Optimization, Electric Vehicles, Vehicle Routing.

I. Introduction

Automotive vehicle drivers usually have hundreds or even thousands of route options to pick from when travelling from a start to a destination point on a predefined map. The routes that are mostly assumed practical are the ones that yield the least travel time or the shortest travel distance. Sometimes a compromise between these 2 constraints is considered a viable solution, as optimizing for both at the same time might not be feasible. In general, optimizing for either the travel time or the travel distance will indirectly lead to energy efficient route selection or a "close enough" solution. This is true because the less time traveled or the shorter distance traveled will yield less engine run time. The less time the engine is run, the less fuel it will consume.

This is not necessarily true for an electric vehicle (EV) mainly because they are capable of regenerating energy. EVs are powered by high-efficient electric motors that can function as generators when the EV is in coast mode, travelling downhill, or when the driver steps on the brake pedal to decelerate the vehicle. The kinetic and/or the potential energies are recovered and stored back onboard the vehicle in the high-voltage battery pack [1]. This stored energy can be used at a later time to propel the vehicle. The range of an EV can increase by 20% or more due to regenerative braking [2].

Some studies have tackled the energy efficient routing problem for electric vehicles. These studies can be split into 2 categories: deterministic and stochastic optimization methods. Previously we have done some work in the stochastic field [3]. The energy efficient routing problem was solved using Particle Swarm Optimization [4], Ant Colony Optimization [5], and Tabu Search [6]. Other work tackled the problem from the deterministic side. Artmeier et al. [7] studied energy-efficient routing for an EV from a graph theory context. The authors developed a generic shortest path algorithm using potential shifting to discard the negative edge costs and to solve the EV routing problem. They analysed the shortest path problem and adapted the method to fit specific EV characteristics. Furthermore, they identified hard constraints which restrict the battery from discharging below certain limits and soft constraints which impose that the battery cannot store more energy than its maximum capacity. The developed algorithm had a worst case complexity of $O(n^3)$. Sachenbacher et al. [8] presented an A* search that considers regenerative energy and the battery parameters which yields a consistent heuristic function of $O(n^2)$ complexity problem. The authors developed the routing algorithm claiming that it expands each node at most once. The presented algorithm adjusted for the battery limits by modifying the weight function that the authors presented. The weight function is generated by the potential energy and the propulsion system inefficiencies. In [9], Baum et al. presented a fast energy efficient routing algorithm for an EV and they tested their results on European and Japanese road networks. Although the model was based on an actual vehicle (Peugeot iOn) the authors also did not prove their results on an actual EV. Furthermore, their developed algorithm does not handle negative cycles. The authors also did not consider traffic conditions. The effects of traffic can be substantial as a road segment might be completely blocked. That will substantially delay the developed algorithm's processing time by a factor of thousands. Severe traffic might completely block a road segment thus yielding to a road topology change. Finally, the authors did not consider the vehicle travel time. Vehicle traffic time has a lot of impact not only on the passengers' comfort but also on the EV energy efficient routing problem. Ambient

978-1-4673-6741-7/15/$31.00 ©2015 IEEE

temperatures might yield high loads on the EV's battery thus substantially affecting the routing problem.

This paper will address the EV energy efficient routing problem in its simplest and most straightforward form. Bellman-Ford approach is applied to an EV model. The generated results are compared to the shortest routes and the routes with the least travel time.

This paper is organized as follows: the next section will discuss the Bellman-Ford algorithm, followed by a section that describes the EV model. The developed algorithm will then be presented and analysed. The paper is then ended with the results' discussions and the conclusion sections.

II. BELLMAN-FORD ALGORITHM

There are numerous deterministic methods that are used to solve for the shortest path problem. Table I lists some of these algorithms along with their complexity, negative edge support, and the year they were introduced. One of the most significant Dynamic Programming methods is the Bellman-Ford algorithm [10, 11]. It is mostly used to solve for problems with a single source vertex that might include negative routes This algorithm can detect negative nodes but it cannot find the shortest path if a negative cycle is reachable from the source. Furthermore, it is slower than Dijkstra's algorithm and it is mostly used in applications where the target to be sought is not at a negative cycle, this is true because of the fact that it terminates upon finding a negative cycle.

TABLE I
Major developed shortest path algorithms

Year	Algorithm	Complexity	Negative Edge Support
1957	Bellman, et al.	$O(V^3)$	✓
1958	Dijkstra	$O(V^2)$	✗
1962	Floyd-Warshall	$O(V^3)$	✓
1964	Williams	$O(E\ \log_2 V)$	✗
1974	Suurballe	$O(E + V \log_2 V)$	✗
1977	Johnson	$O(V^2 \log V + \lvert V \rvert \lvert E \rvert)$	✓
1984	Fredman, et al.	$O(E + \mathrm{V} \log V)$	✗

The pseudo code that describes how the Bellman - Ford algorithm operates is shown in Fig. 1. The key idea of the Bellman - Ford algorithm is the relaxation step or operation. This operation takes 3 inputs which include 2 nodes (A, B) and the edge length connecting the 2 nodes (A) and (B). If the distance between the source and the first node (A) plus the edge length is less than the distance to the second node (B), then the first node is marked as the predecessor for the second node (B) (line 11). In this case, the distance from the source to the second node (B) is calculated according to (1). The ||…|| throughout the paper refers to the signed distance or length between 2 points. If the distance between the source and the first node (A) plus the edge length is greater than the distance to the second node (B), then no changes are applied.

$$Dist_{OB} = \|OA\| + \|edge_{AB}\| \tag{1}$$

Bellman - Ford Algorithm

```
01. function (predecessor) BellmanFord(vertices, edges, source)
02.
03. for  i = 1 to total number of vertices do
04.      distance[i]       = +∞
05.      predecessor[i] = null
06. end
07. distance[source] = 0
08.
09. for  i = 1 to length(vertices) – 1 do
10.      for each edge e do
11.          if distance[e_from] + length(e) < length(e_to) do
12.              distance[e_to] = distance[e_from] + length(e)
13.              predecessor[e_to] = e_from
14.          end
15.      end
16. end
17. for each edge e do
18.      if distance[e_from] + length(e)  < distance[e_to]  do
19.          return ERROR
20.      end
21. end
22.
23. return predecessor
```

Fig 1: Bellman - Ford Algorithm Pseudo Code

III. ELECTRIC VEHICLE MODEL

For an EV the theoretical driveline energy consumption is an integration of the power output at the battery terminals. The power output (in watts) can be expressed as shown in (2). Where M is the vehicle mass in kilograms, g is the gravitational acceleration, $\mathrm{f_r}$ is the tire rolling resistance coefficient, $\mathrm{p_a}$ is the air mass density, $\mathrm{C_D}$ is the aerodynamic drag coefficient, $\mathrm{A_f}$ is the frontal area of the vehicle in $\mathrm{m^2}$, V is the vehicle speed in $\mathrm{m/s}$, δ is the rotational inertia factor, i is the road grade, and a is the vehicle acceleration in $\mathrm{m/s^2}$. Equation (3) shows the regenerative braking power at the battery terminals, where α $(0 < \alpha < 1)$ is the percentage of the total brake energy that can be regenerated by the electric motor [12]. The total energy shown in Equation (4) is the net energy consumption at the battery terminals. The goal of the proposed optimization process is to minimize (4), given that the electric vehicle has to be driven between 2 points.

$$\mathrm{P_{out}} = \mathrm{V}\left(\mathrm{Mg}(\mathrm{f_r} + \mathrm{i}) + \tfrac{1}{2}\rho_\mathrm{a}\mathrm{C_D}\mathrm{A_f}\mathrm{V}^2 + \mathrm{M}\delta a\right) \tag{2}$$

$$\mathrm{P_{in}} = \alpha\mathrm{V}(\mathrm{Mg}(\mathrm{f_r} + \mathrm{i}) + \tfrac{1}{2}\rho_\mathrm{a}\mathrm{C_D}\mathrm{A_f}\mathrm{V}^2 + \mathrm{M}\delta a) \tag{3}$$

$$\mathrm{Energy_{TOT}} = \int \mathrm{P_{out}}\mathrm{dt} + \int \mathrm{P_{in}}\mathrm{dt} \tag{4}$$

In general, the terms in (2) and (3) can take values as shown in Table II. Furthermore, these equations can be further simplified to make the calculations less tedious. Equation (5) shows a simpler form representing (2). K, A, B, and C are all constants.

$$P_{out} = V(K + A\,.\,i + B.\,V^2 + C.\,a) \qquad (5)$$

Where:

$K = M.\,g.\,f_r$
$A = M.\,g$
$B = 0.5\,.\,\rho_a.\,C_D.\,A_f$
$C = M.\,\delta$

Similarly, (3) can be expressed in the same way as shown in (6).

$$P_{in} = \alpha.\,V(K + A\,.\,i + B.\,V^2 + C\,.\,a) \qquad (6)$$

TABLE II
Typical Vehicle parameters

Description	Symbol	Value	Unit
Aerodynamic Drag Coefficient	C_D	0.25	–
Frontal Area	A_f	2.25	m^2
Rolling Resistant Coefficient	f_r	0.005	–
Vehicle Mass	M	1,500	Kg
Gravitational Acceleration	G	9.8	m/s^2
Air Mass Density	ρ_a	1.275	Kg/m^3
Rotational Inertia Factor (mass factor)	δ	1	–
Regenerative Braking Factor	α	0.4-0.6	–

IV. BELLMAN-FORD APPROACH IMPLEMENTATION

To be able to implement the Bellman-Ford search algorithm on a source to destination problem the energy consumption between the nodes needs to be calculated. This calculated energy might be negative. To better illustrate the process a simple routing example is presented. Fig. 2 shows a simplified map with 7 nodes and 10 edges. The variables between the nodes marked as $[d_x, s_x]$ represent the distance and the speed limit respectively on that specific edge. Table III shows the values of these variables. Furthermore Table IV shows the potential energy level of each node i.e. the relative elevation. This is needed to calculate the i term in (5) and (6).

Using (5) and (6) along with the parameters listed in Tables II, III, and IV a new map is generated. This new map is shown in Fig. 3. The numbers listed on the path between the nodes represents the energy consumption needed for the EV to travel between these nodes in the direction of the arrow. Observing Fig. 2 the shortest route between A and G is: A to C to F to G with a total distance of 1740 meters. The route that will lead the fastest travel time between A and G is: A to B to E to G with a total travel time of 101.2 seconds this yields an average driving speed of ~80 km/h or ~49 mph. Applying the Bellman-Ford Algorithm onto Fig. 3, the most energy efficient route between A and G is: A to C to D to G.

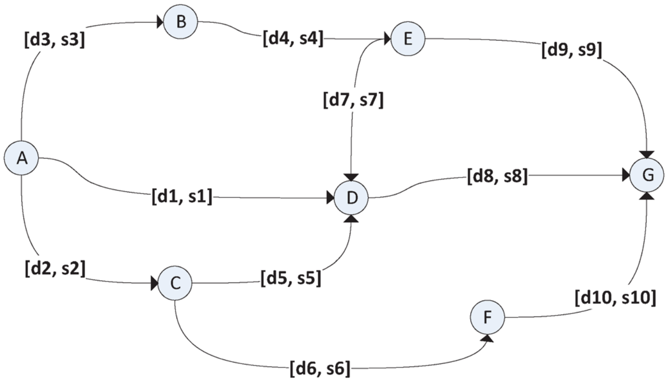

Fig. 2: Simple Map used to Illustrate the Bellman-Ford Implementation (Map not to scale)

TABLE III
Edge Properties

Edge	Distance (meters)	Speed Limit (Km/Hr)
[d1, s1]	d1 =2500	s1 = 120
[d2, s2]	d2 =750	s2 = 50
[d3, s3]	d3 =500	s3 = 50
[d4, s4]	d4 =650	s4 =75
[d5, s5]	d5 =100	s5 =10
[d6, s6]	d6 =650	s6 =50
[d7, s7]	d7 =500	s7 =50
[d8, s8]	d8 =900	s8 =100
[d9, s9]	d9 =1100	s9 =120
[d10, s10]	d10 =340	s10 =40

TABLE IV
Node Elevation

Node	Elevation
A	500
B	550
C	350
D	600
E	550
F	450
G	400

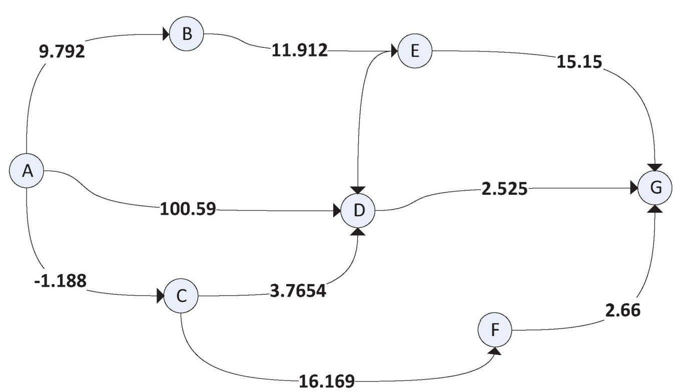

Fig. 3: Energy Consumption ($\times 10^6$) in Joules Needed to Move the EV between the Nodes

V. DISCUSSIONS

It can be observed from the results that neither the shortest route nor the route with the least travel time yielded the most energy efficient route. This concludes that applying the optimization process on either the distance or the travel time will not yield optimal results.

The Bellman-Ford algorithm was applied to different map sizes. The simulation was run using a quad core desktop computer running Matlab R2009b at 2.27 GHz of processing speed. Table V shows different results using different map sizes. These results show that the implementation is practical and can be used to generate real-time results if the number of vertices and edges is reasonable. Whenever the number of vertices and edges become high Bellman – Ford becomes impractical and cannot be used to solve for the EV routing problem. Other suggested techniques include preprocessing of the map, but that might be also impractical because of the dynamic characteristics of the battery and the routes. Another approach is to use metaheuristic methods as we used in [3 – 6]. These optimization techniques are slower for smaller maps, but for more complex maps where Bellman – Ford starts lacking, they produce very promising results as they can be used in real time routing problems.

Table V
Bellman-Ford Computational Time to Generate a Solution

Number of Vertices	Number of Edges	Time to Generate Solution
20	37	0.054 sec
29	63	0.138 sec
294	1,152	25.80 sec
988	10,026	774.8 sec
16,340	270,780	203 hours

V. CONCLUSION

In this paper, we presented a Bellman-Ford solution to the energy efficient routing problem for electric vehicles. An EV model was presented and the algorithm was applied on the model. The developed algorithm proved to run fast and can be used in real-time to provide drivers the most energy efficient route.

REFERENCES

[1] R. Abousleiman, A. Al-Refai, and O. Rawashdeh, "Charge Capacity Versus Charge Time in CC-CV and Pulse Charging of Li-Ion Batteries," SAE Technical Paper 2013-01-1546, 2013, doi:10.4271/2013-01-1546.

[2] A. Artmeier, J. Haselmayr, M. Leucker and M. Sachenbacher, The optimal routing problem in the context of battery-powered electric vehicles, Workshop: CROCS at CPAIOR-10, Second International Workshop on Constraint Reasoning and Optimization for Computational Sustainability, Bologna, Italy, 2010.

[3] R. Abousleiman and O. Rawashdeh. "Energy-efficient routing for electric vehicles using metaheuristic optimization frameworks", Mediterranean Electrotechnical Conference (MELECON), 2014 17th IEEE

[4] R. Abousleiman and O. Rawashdeh, "Energy Efficient Routing for Electric Vehicles using Particle Swarm Optimization", SAE Technical Paper 2014-01-0054, 2014.

[5] R. Abousleiman and O. Rawashdeh, "An Application of Ant Colony Optimization to Energy Efficient Routing for Electric Vehicles", SAE Technical Paper 2013-01-0337, 2013.

[6] R. Abousleiman and O. Rawashdeh, "Tabu Search based Solution to the Electric Vehicle Energy Efficient Routing Problem", ITEC Dearborn, MI, 2014.

[7] A. Artmeier, J. Haselmayr, M. Leucker, and M. Sachenbacher, "The shortest path problem revisited: Optimal routing for electric vehicles," In 33rd Annual German Conference on Artificial Intelligence (KI-2010).

[8] M. Sachenbacher, M. Leucker, A. Artmeier, and J. Haselmayr, "Efficient energy-optimal routing for electric vehicles," In Twenty-Fifth AAAI Conference on Artificial Intelligence, 2011.

[9] M. Baum, et al., "Energy-Optimal Routes for Electric Vehicles" Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2013.

[10] R. Bellman, On a routing problem. No. RAND-P-1000. RAND CORP Santa Monica CA, 1956.

[11] R. Ford, and F. Delbert. A simple algorithm for finding maximal network flows and an application to the Hitchcock problem. Rand Corporation, 1955.

[12] Ehsani, Mehrdad, Yimin Gao, and Ali Emadi. Modern electric, hybrid electric, and fuel cell vehicles: fundamentals, theory, and design. CRC, 2009.