Akademie věd České republiky
Ústav teorie informace a automatizace, v.v.i.

Academy of Sciences of the Czech Republic
Institute of Information Theory and Automation

RESEARCH REPORT

Ondřej Sedláček and Evženie Uglickich

Calculating Electric Vehicle Range Using Dynamic Programming with Battery Current and Velocity Setpoints

2398 November 2024

Otevřená věda 2024
Project 101096884, Listen2Future
Project 9A22004 of MEYS

ÚTIA AV ČR, P.O.Box 18, 182 08 Prague, Czech Republic
Tel: (+420)266052422, Fax: (+420)286890378, Url: http://www.utia.cas.cz, E-mail: utia@utia.cas.cz

Any opinions and conclusions expressed in this report are those of the authors and do not necessarily represent the views of the involved institutions.

Contents

1 Introduction

This work was made during an internship at Institute of Information Theory and Automation organized by Otevřená věda. The internship was supervised by Evženie Uglickich.

The effort put into this internship resulted in a program simulating and predicting the range of a BMW i3 60 Ah electric car using Bayesian estimation of regression model [1] and optimal control with dynamic programming [2, 3]. After a dataset is entered, the program trains models to be used for the simulation and then allows you to set setpoints to a vehicle speed or battery current. Once the simulation with the setpoints is completed, it will return the speed at which the car was traveling, how much current was drawn from the battery, and the range of the electric vehicle (EV) traveling at that speed. In addition, the graph of the ride of the simulated car is shown after the computation.

We have used the driving data of the BMW i3 60 Ah electric car from publicly available datasets, accessible at [4] under a Creative Commons Attribution license.

The layout of the work is as follows: Section 2 provides a manual for users interested in trying out the program. Then, in Section 3, the models are introduced and it is specified which data features were used during training the models. Section 4 demonstrates results of experiments run on the simulated vehicles. Conclusions in Section 5 close the work.

2 Manual to the program

The program can be downloaded from `https://github.com/ZDVokoun/ev-range-modelling`.

The program consists of several Python scripts, from which we will run `main.py`. This program has dependencies written in the `requirements.txt` file (NumPy, skikit-learn, matplotlib, Pandas). We recommend installing the exact versions listed in this file so as to be sure that every feature works. We also recommend installing these dependencies in the virtual environment. You can do this by opening a terminal in the folder containing the code and then entering (the commands may differ depending on your OS, tested on Debian with Python 3.12.6):

```
python -m venv .
source ./bin/activate
pip install -r requirements.txt
```

Then you run the program using these commands:

```
source ./bin/activate
python main.py
```

When you run `main.py`, you will get a command prompt. The first thing you always need to do is to type `load {}`, where instead of `{}` you put the code in `A01` format. To do this, you must download datasets in the `dataKaggle` folder first and have dataset files named `TripA01.csv`, `TripA02.csv`, ..., `TripB01.csv`, etc. When this command is entered, model training for current and speed prediction is started. For example, when you enter `load A01` you will see this:

```
>>> load A01
Training current model...
RMSE: 10.570
R2: 0.757602
Coef: [-0.28145583  1.74261843  0.18230414 -0.76709843  0.06121821  0.85283665]
Training velocity model...
RMSE: 6.088
R2: 0.897811
Coef: [ 3.89797385e-04  1.83349615e+00  5.67938707e-03 -8.36048233e-01
 -3.33939910e-03]
```

Then you can enter optimal control commands, namely `velocity_optimal`, `throttle_optimal`, `current_optimal` and their variants, which are used to calculate multiple values at the same time. For this command, you specify the setpoint you want the program to follow. After the calculation, it returns the range of the EV driving according to the setpoint and opens a graph of the simulated EV driving. The examples are as follows:

```
>>> velocity_optimal 50
Range with -37.60 A and 50.00 km/h: 76.87 km
>>> velocity_optimal_range 40 60 5
Range with -23.11 A and 40.00 km/h: 95.77 km
Range with -30.35 A and 45.00 km/h: 82.84 km
Range with -37.60 A and 50.00 km/h: 74.79 km
Range with -44.84 A and 55.00 km/h: 69.29 km
>>> current_optimal_range -100 -50 10
Range with -100 A and 93.08 km/h: 53.53 km
Range with -90 A and 86.17 km/h: 55.06 km
Range with -80 A and 79.27 km/h: 56.98 km
Range with -70 A and 72.37 km/h: 59.45 km
Range with -60 A and 65.46 km/h: 62.74 km
```

For more information, you can always type `help` to get a short help.

3 Specification of models and algorithms for optimal control

We use a linear regression model in the form

$$y_t = b_0 u_t + a_1 y_{t-1} + b_1 u_{t-1} + a_2 y_{t-2} + b_2 u_{t-2} + k + e_t, \tag{1}$$

where y_t is the modeled output variable, and u_t is the control input variable, both measured at discrete time instants t; $\{b_0, a_1, b_1, a_2, b_2, k\}$ is a set of unknown regression coefficients, and e_t is a normal white noise with zero mean and an unknown fixed variance r.

In the program, we use two types of models (1):

- the first describes velocity as the output y_t,
- and the second uses battery current as the output y_t,

both with throttle as the control input u_t. The regression coefficients and noise variance for both types of models are estimated using the recursive Bayesian estimation algorithm [1], applied to the steering data. Its results are close to those obtained from least squares estimation but allow us to compute predictions at each time step of the trip. After the models are trained, we are then able to compute the EV range for given setpoints of velocity and current.

For optimization, using the throttle as the control input and tracking the setpoints for velocity and battery current, we will employ an optimal control algorithm with substituted point estimates of the model parameters. This approach leverages dynamic programming to minimize the expected value of the penalization function over the entire control interval; see, e.g., [2, 3, 5], among others. The following quadratic penalization function is used:

$$J_t = (y_t - y_{ref;t})^2 \omega + (u_t - u_{t-1})^2 \lambda, \tag{2}$$

where $y_{ref;t}$ is the setpoint at time t. The penalization weights used in this program are $\omega = 0.1$ and $\lambda = 1$. After the computation of the optimal throttle, the only step left is to predict the current from the optimal throttle when given a velocity setpoint and vice versa. At this point, we have all the information needed to use the formula for computing the EV range.

The computation of EV range is obtained through the following derivations: We can suppose that when the driver is going to ride the vehicle the same way as he did during the measured period, the ratio between vehicle range s_{range} and travelled distance s is the same as the ratio between the battery capacity Q and the charge lost during the trip Q_{lost}, hence:

$$\frac{s_{\text{range}}}{s} = \frac{Q}{Q_{\text{lost}}}. \tag{3}$$

Because the battery capacity is usually measured in Ah, we will change the equation a little bit:

$$\frac{s_{\text{range}}}{s} = \frac{3600Q}{Q_{\text{lost}}}. \tag{4}$$

The current in the datasets used is measured in 0.1 s long intervals, therefore:

$$Q_{\text{lost}} = \left|\sum_{t=1}^{n} I_t \cdot 0.1\right| = 0.1 \cdot \left|\sum_{t=1}^{n} I_t\right|, \tag{5}$$

where n is the number of data points, and I_t denotes the current. Since positive values of battery current in the dataset indicate that the vehicle is recuperating energy, the sum is expressed as an absolute value function to prevent the vehicle range from taking on negative values. After substitution, we obtain the final form of the formula:

$$s_{\text{range}} = s\frac{3600Q}{0.1 \cdot |\sum_{t=1}^{n} I_t|}. \tag{6}$$

4 Experiments

This section shows examples of using the program with two selected datasets from [4], B01 and B03.

4.1 Models trained on dataset B01

Here, we show results of running experiments on the `TripB01.csv` dataset from [4]. The output of training the models is as follows:

```
>>> load B01
Training current model...
RMSE: 36.364
R2: 0.743192
Coef: [-0.44129143  1.82030291  0.37597297 -0.82871325  0.04495354  0.52506564]
Training velocity model...
RMSE: 18.328
R2: 0.728929
Coef: [ 1.35615266e-03  1.90492241e+00  5.59607076e-03 -9.05704465e-01
 -5.74119631e-03]
```

4.1.1 Impact of velocity on vehicle range

In these experiments, we tried setting the velocity setpoint to values between 40 km/h and 125 km/h. The output is as follows:

```
>>> velocity_optimal_range 40 130 5
Range with -0.12 A and 40.00 km/h: 681.71 km
Range with -7.93 A and 45.00 km/h: 218.83 km
Range with -15.75 A and 50.00 km/h: 144.96 km
Range with -23.57 A and 55.00 km/h: 113.61 km
Range with -31.39 A and 60.00 km/h: 96.28 km
Range with -39.21 A and 65.00 km/h: 85.27 km
Range with -47.03 A and 70.00 km/h: 77.67 km
Range with -54.85 A and 75.00 km/h: 72.10 km
Range with -62.66 A and 80.00 km/h: 67.85 km
Range with -70.48 A and 85.00 km/h: 64.49 km
Range with -78.30 A and 90.00 km/h: 61.78 km
Range with -86.12 A and 95.00 km/h: 59.54 km
Range with -93.94 A and 100.00 km/h: 57.66 km
Range with -101.76 A and 105.00 km/h: 56.06 km
Range with -109.57 A and 110.00 km/h: 54.69 km
Range with -117.39 A and 115.00 km/h: 53.49 km
Range with -125.21 A and 120.00 km/h: 52.44 km
Range with -133.03 A and 125.00 km/h: 51.52 km
```

Figure 1 shows a plot of the car riding with the entered setpoints. The results suggest that a change in speed during faster driving (e.g. on a motorway) has less impact on the simulated car than during slower driving (e.g. through a village), but in general the range increases with lower speed. Thus, the range of the simulated vehicle can be most affected when driving slower.

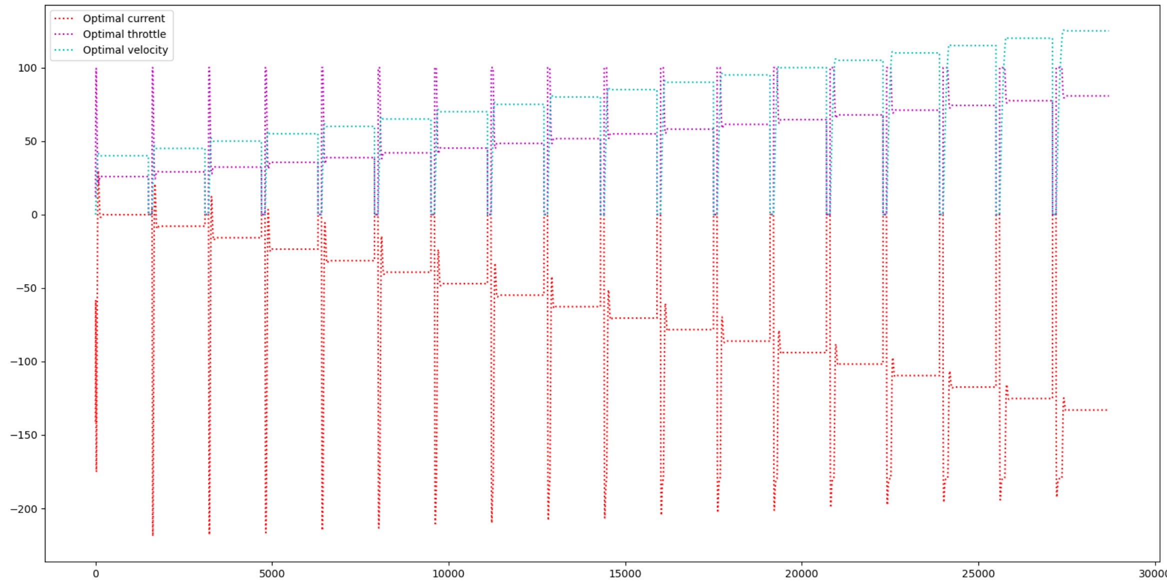

Figure 1: Graph of optimal control with velocity setpoint

4.1.2 Impact of current on vehicle range

In these experiments we tried to set the setpoint current to values from -180 A to -30 A. The output is as follows:

```
>>> current_optimal_range -180 -20 10
Range with -180 A and 154.85 km/h: 47.71 km
Range with -170 A and 148.64 km/h: 48.41 km
Range with -160 A and 142.25 km/h: 49.23 km
Range with -150 A and 135.85 km/h: 50.16 km
Range with -140 A and 129.46 km/h: 51.21 km
Range with -130 A and 123.06 km/h: 52.43 km
Range with -120 A and 116.67 km/h: 53.85 km
Range with -110 A and 110.27 km/h: 55.54 km
Range with -100 A and 103.88 km/h: 57.55 km
Range with -90 A and 97.48 km/h: 60.02 km
Range with -80 A and 91.09 km/h: 63.09 km
Range with -70 A and 84.69 km/h: 67.05 km
Range with -60 A and 78.30 km/h: 72.32 km
Range with -50 A and 71.90 km/h: 79.69 km
Range with -40 A and 65.51 km/h: 90.74 km
Range with -30 A and 59.11 km/h: 109.11 km
```

Figure 2 shows a plot of the car riding with the entered setpoints. Here, in the data we see similar results to speed, or at lower speeds it costs less energy to increase the speed of the simulated vehicle.

4.1.3 Impact of throttle on vehicle range

In these experiments, we tried setting the fixed throttle values from 40 % to 100 %. The output is as follows:

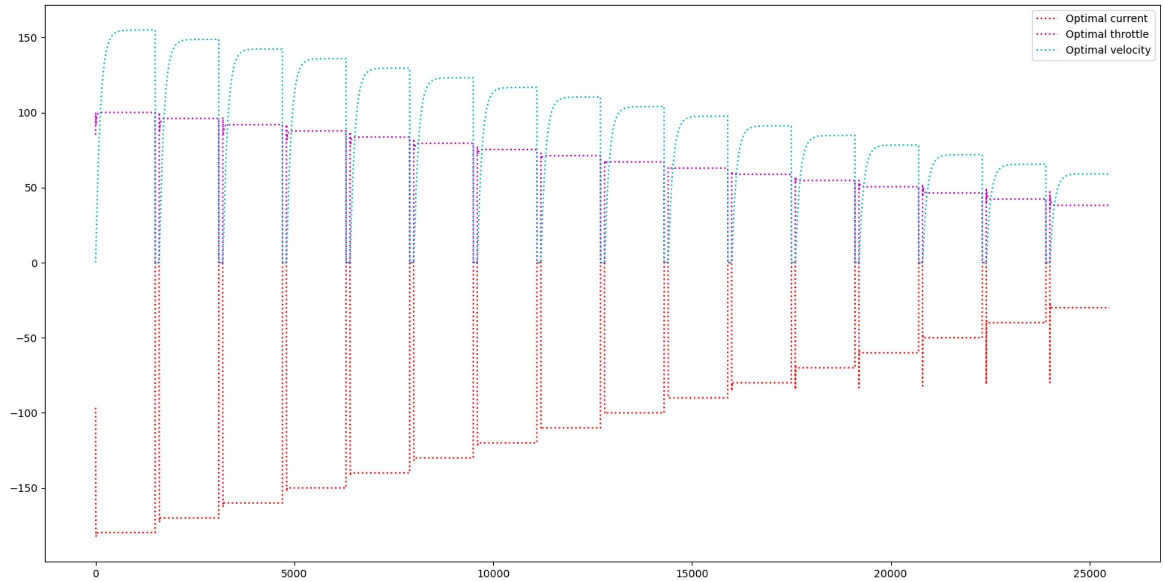

Figure 2: Graph of optimal control with current setpoint

```
>>> throttle_optimal_range 40 101 10
Range with -34.43 A and 61.94 km/h: 100.53 km
Range with -58.64 A and 77.43 km/h: 73.09 km
Range with -82.85 A and 92.91 km/h: 62.10 km
Range with -107.07 A and 108.40 km/h: 56.07 km
Range with -131.28 A and 123.88 km/h: 52.27 km
Range with -155.50 A and 139.37 km/h: 49.65 km
Range with -179.71 A and 154.85 km/h: 47.73 km
```

Figure 3 shows a plot of the car riding with the entered throttle values. It can be seen that as the throttle is increased, the current increases and so does the speed, which implies a reduction in range.

4.2 Models trained on dataset B03

The models we will now run experiments on to confirm the results of the previous experiments are trained on the `TripB03.csv` dataset. The output of training the models is as follows:

```
>>> load B03
Training current model...
RMSE: 35.292
R2: 0.779033
Coef: [-0.49804703  1.80147833  0.41714241 -0.81003718  0.05763614  0.59328676]
Training velocity model...
RMSE: 19.985
R2: 0.740818
Coef: [ 7.28864042e-04  1.89326407e+00  5.91067799e-03 -8.94335424e-01
 -4.99650761e-03]
```

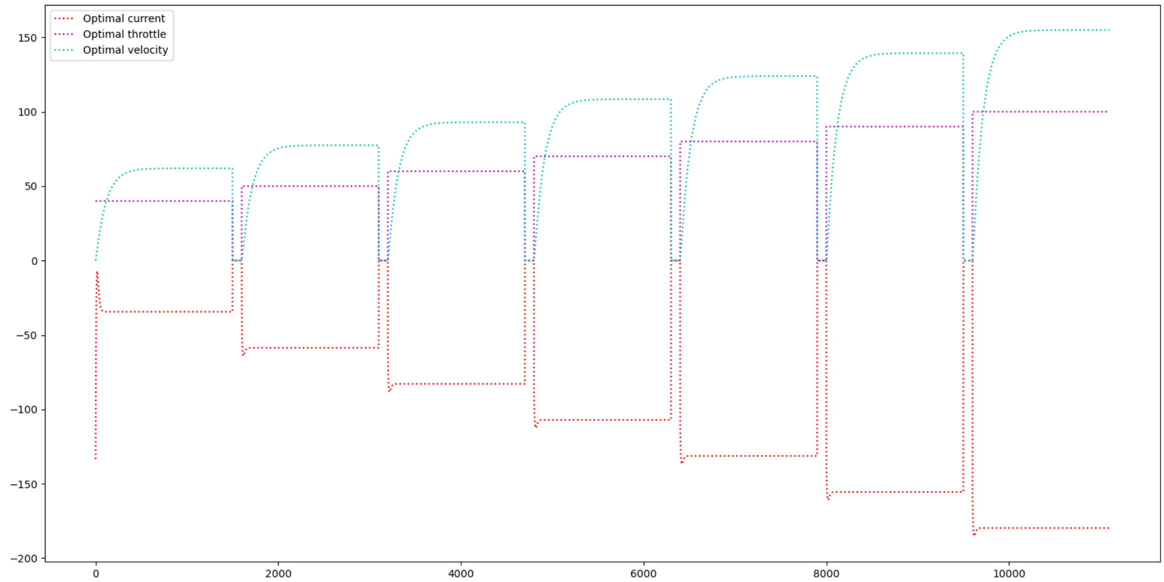

Figure 3: Graph of optimal control with throttle setpoint

4.2.1 Impact of velocity on vehicle range

In these experiments, we tried setting the velocity setpoint to values between 40 km/h and 125 km/h with the different dataset used. The output is as follows:

```
>>> velocity_optimal_range 40 130 5
Range with -1.59 A and 40.00 km/h: 484.83 km
Range with -10.45 A and 45.00 km/h: 186.32 km
Range with -19.32 A and 50.00 km/h: 125.70 km
Range with -28.18 A and 55.00 km/h: 99.30 km
Range with -37.04 A and 60.00 km/h: 84.52 km
Range with -45.91 A and 65.00 km/h: 75.07 km
Range with -54.77 A and 70.00 km/h: 68.51 km
Range with -63.63 A and 75.00 km/h: 63.69 km
Range with -72.50 A and 80.00 km/h: 60.00 km
Range with -81.36 A and 85.00 km/h: 57.08 km
Range with -90.22 A and 90.00 km/h: 54.72 km
Range with -99.09 A and 95.00 km/h: 52.76 km
Range with -107.95 A and 100.00 km/h: 51.12 km
Range with -116.82 A and 105.00 km/h: 49.73 km
Range with -125.68 A and 110.00 km/h: 48.52 km
Range with -134.54 A and 115.00 km/h: 47.47 km
Range with -143.41 A and 120.00 km/h: 46.56 km
Range with -152.27 A and 125.00 km/h: 45.74 km
```

Figure 4 shows a plot of the car riding with the entered setpoints. This result confirms the result of the speed-range impact experiment with models trained on the `TripB01.csv` dataset and clearly shows that changing the speed of the simulated vehicle has a much greater effect on the range when driving slowly than when driving faster. Consequently, it is worth sticking to lower speeds when driving through the village.

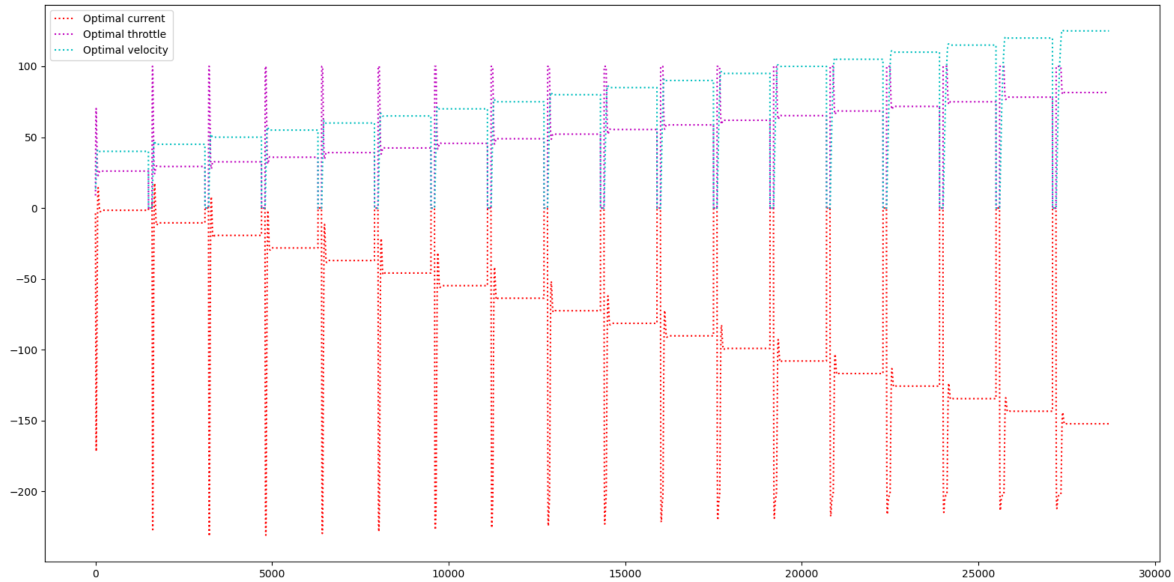

Figure 4: Graph of optimal control with velocity setpoint (dataset B03)

5 Conclusion

The program described in this paper can help in getting an insight into the problem of range optimization of electric cars. It uses recursive Bayesian theory for estimation of regression models to simulate electric vehicle and compute vehicle range based on the parameters given. Furthermore, the applied Bayesian approach allows us to train the model online instead of retraining it completely.

However, by using this approach, we have made many assumptions on the data which may not reflect reality. Moreover, the models may not take many factors affecting vehicle range into account. Thus, the computations done by the program may not be precise enough.

Despite the imperfections listed, this program may be useful for researchers working in the field. In addition, we could use different datasets or the model with other variables involved with the help of relatively slight modifications of the program.

Acknowledgements

European Funding: Under grant 101096884, Listen2Future is co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or Chips Joint Undertaking. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the CHIPS JU and its members (including top-up funding by Austria, Belgium, Czech Republic, Germany, Netherlands, Norway and Spain.
National Funding: This project has also received national funding from the Ministry of Education, Youth and Sports of the Czech Republic (MEYS) under grant agreement No 9A22004.

References

[1] Peterka, V., 1981. Bayesian system identification, in Eykhoff, P. (Ed.), Trends and Progress in System Identification. Oxford, Pergamon Press, pp. 239-304.

[2] Feldbaum, A.A., 1961. Theory of dual control. Autom. Rem. Control 22 (2).

[3] Bellman, R., 2003. Dynamic Programming. Princeton University Press (Dover paperback edition ISBN 0-486-42809-5).

[4] Steinstraeter, M., Buberger, J., Trifonov, D., 2020. Battery and Heating Data in Real Driving Cycles. IEEE Dataport. https://dx.doi.org/10.21227/6jr9-5235

[5] Suzdaleva, E., Nagy, I., 2014. Data-based speed-limit-respecting eco-driving system, Transportation Research Part C: Emerging Technologies, vol. 44, p. 253-264, https://doi.org/10.1016/j.trc.2014.04.009.