Vehicle State Estimation in Automotive Electromobility

Implementation of a grey-box recursive least squares (RLS) algorithm, which can be used for estimation of key electric vehicle (EV) parameters, such as battery condition and vehicle mass, in real time. The method contains least squares estimation with forgetting factor for faster adaptation to changing conditions

2.1. Introduction

The transition from vehicles powered by fossil fuels to electric vehicles (EV) not only brought challenges in developing battery cells and new powertrains but also the need to estimate vehicle parameters during their operation [1]. It is essential to know the current vehicle status to apply operating strategies and control of an electric vehicle. Of particular interest for modern EVs are especially the battery condition and the vehicle mass, which can change depending on the load [2], [3]. Recursive least squares estimators are a computationally efficient method for determining the vehicle's state [3]. Black-box or grey-box modeling is required to estimate these unmeasured states. In comparison, Black-box models estimate the unmeasured states only from input signals [4]. They are independent of physical relations (e.g. neuronal networks), while grey-box modeling estimates system parameters based on their physical relations [5]. Recursive least squares estimators are a computationally efficient method for determining the vehicle's state within a grey box model as an extension to the classic least squares method [6], [7]. In [2], this online parameter identification approach using least squares is used, estimating the capacity and the maximum available output power in EV batterie management systems. The state determination can be made adaptive by extending the method to include exponential forgetting [3], [7]. Based on new sensor data, the recursive least squares update their solution continuously, which makes real-time vehicle state estimations possible.
This report discusses the grey-box model of a recursive least squares algorithm estimation based on the classic least squares method from the lecture for estimating EV parameters. Additionally, a forgetting factor will be introduced, which enables faster adjustment of changing parameters. In conclusion, the presented theory is applied and analyzed based on the difference between a noisy and not noisy system output for recursive least squares estimation with exponential forgetting, using Matlab.

2.2. Methods

The general nonlinear or linear physical relation between system inputs and -outputs must be known to estimate the vehicle parameters (grey box model). In this report, only linear physical relations are assumed, which can be described in an n-dimensional **Multiple-Input-Single-Output (MISO)** model

$$y = \boldsymbol{\phi}^T \hat{\boldsymbol{\theta}} + \Delta y = \begin{bmatrix} \phi_1 & \dots & \phi_n \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} + \Delta y, \tag{1}$$

where y is the known system output (scalar), ϕ is the known system input-vector, Δy (unknown) measurement error, and $\hat{\boldsymbol{\theta}}$ the searched for parameter-vector [7].

Applying a **least squares estimation (LS)** requires N independent measurements of equation 1, which results in the model equation

$$\boldsymbol{y} = \begin{bmatrix} y[1] \\ \vdots \\ y[N] \end{bmatrix} = \begin{bmatrix} \phi^\top[1] \\ \vdots \\ \phi^\top[N] \end{bmatrix} \hat{\boldsymbol{\theta}} + \begin{bmatrix} \Delta y[1] \\ \vdots \\ \Delta y[N] \end{bmatrix} = \boldsymbol{\Phi}\hat{\boldsymbol{\theta}} + \boldsymbol{\Delta}\mathbf{y}. \tag{2}$$

From the lecture theory, the estimation of the searched parameter vector $\hat{\boldsymbol{\theta}}$ can be calculated to

$$\hat{\boldsymbol{\theta}} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{y} = \boldsymbol{\Phi}^+\boldsymbol{y}, \tag{3}$$

where the sum of error squares of $e = y - \Delta y = y - \Phi\hat{\theta}$ is minimized [8]. The input information matrix $\boldsymbol{\Phi^T\Phi = P^{-1}}$ must be invertible. This condition is referred to **persistent excitation** [9] (page.64).

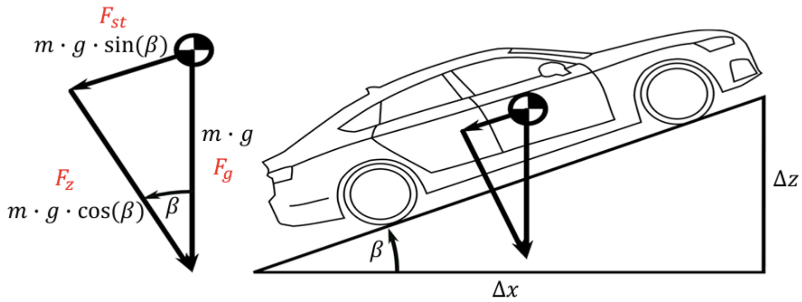

Figure 1: Uphill driving Vehicle with the acceleration of gravity g and the road pitch β

Vehicle Longitudinal Dynamics Model [10]

The vehicle's longitudinal dynamics can be derived using the neton law with the F_A driving force, road pitch resistance F_{ST}, wheel resistance F_W and air resistance F_L [10]:

$$m\dot{v}_x = \sum_i F_i = F_A - F_{ST} - F_W - F_L \tag{4}$$

$$\iff m\dot{v}_x = F_A + mg\beta - c_{ro}mg - \frac{1}{2}\rho_L v_x^2 c_w A_{Veh} \tag{5}$$

with the airtight ρ_L, the vehicle mass m, the rolling resistance c_{ro}, the vehicle front area A_{Veh}, the air resistance c_w and the vehicle velocity v_x. Therefore, resolving according to the unknown parameters for θ, the MISMO model derived:

$$F_A = \underbrace{\begin{bmatrix}\dot{v}_x - g\beta & g & \frac{1}{2}\rho_L v_x^2\end{bmatrix}}_{\phi^T} \underbrace{\begin{bmatrix} m \\ mc_{ro} \\ c_w A_{Veh}\end{bmatrix}}_{\theta} \tag{6}$$

In application to the introduced vehicle longitudinal dynamics model for estimation of the vehicle mass, an real world example for no persistent excitation is given in the following. The case of no persistent excitation is present for driving with constant velocity (v_x = constant) and acceleration ($\dot{v}_x = 0$) if at the same time the road pitch is zero ($\beta = 0$). This results in the system input vector being

$$\phi[k] = \begin{bmatrix}0 & g & \frac{1}{2}\rho_L v_x^2\end{bmatrix}^T,$$

having rank$\{\boldsymbol{\Phi^T\Phi}\} = 2$, not being full rank. As a result, this matrix cannot be inverted. An example is driving on the highway with active cruise control.

To use least squares estimation, the measurements must have been recorded in advance to determine the estimate offline. This method does not make sense for real-time (online) applications, as the memory requirements and the computational effort for the matrix inverse calculation increase with each time step. Therefore, a **recursive method (RLS)** for least squares estimation is introduced in [6] [7], in which not all measurements are stored, but the current estimate

$$\hat{\boldsymbol{\theta}}[k] = \boldsymbol{P}[k]\Phi^\top[k]y[k]$$

is based on the estimate of the previous time step

$$\hat{\boldsymbol{\theta}}[k-1] = \boldsymbol{P}[k-1]\Phi^\top[k-1]y[k-1]$$

including the covariance matrix

$$\boldsymbol{P}[k] = (\Phi^\top[k]\Phi[k])^{-1}.$$

In [7] (page 271-272) the exact derivation of the recursive least squares method is described, resulting in the new estimation

$$\underbrace{\hat{\boldsymbol{\theta}}[k]}_{\substack{\text{New}\\ \text{Estimation}}} = \underbrace{\hat{\boldsymbol{\theta}}[k-1]}_{\substack{\text{Old}\\ \text{Estimation}}} + \underbrace{\boldsymbol{P}[k]\phi[k]}_{\substack{\text{Correction-}\\ \text{Vector}}} \underbrace{(y[k]}_{\substack{\text{New}\\ \text{Measurement}}} - \underbrace{\phi^T[k]\hat{\boldsymbol{\theta}}[k-1])}_{\substack{\text{Predicted}\\ \text{Measurement}}}. \tag{7}$$

The recursive least squares method is also described as

$$\hat{\boldsymbol{\theta}}[k] = \hat{\boldsymbol{\theta}}[k-1] + \boldsymbol{\gamma}[k]e[k], \tag{8}$$

describing the new estimation $\hat{\boldsymbol{\theta}}[k]$ as the correction $\gamma[k]$ of the model error $e[k]$ from the previous estimation $\hat{\boldsymbol{\theta}}[k-1]$. The model error is described as

$$e[k] = y[k] - \boldsymbol{\phi}^\top[k]\hat{\boldsymbol{\theta}}[k-1], \tag{9}$$

multiplied by the correction factor

$$\gamma[k] = \boldsymbol{P}[k]\boldsymbol{\phi}^\top[k] = \frac{\boldsymbol{P}[k-1]\boldsymbol{\phi}[k]}{\boldsymbol{\phi}^\top[k]\boldsymbol{P}[k-1]\boldsymbol{\phi}[k]+1}, \quad \text{with} \quad \boldsymbol{P}[k] = (\boldsymbol{I} - \boldsymbol{\gamma}[k]\boldsymbol{\phi}^T[k])\boldsymbol{P}[k-1]. \tag{10}$$

In Fig. 2 the online vehicle parameter estimation is illustrated, with using recursive least squares method. Therefore, the parameter estimation is calculated by the residual e between the output of the real system y and the defined physical model $\hat{y}$, given the system input ϕ [11].

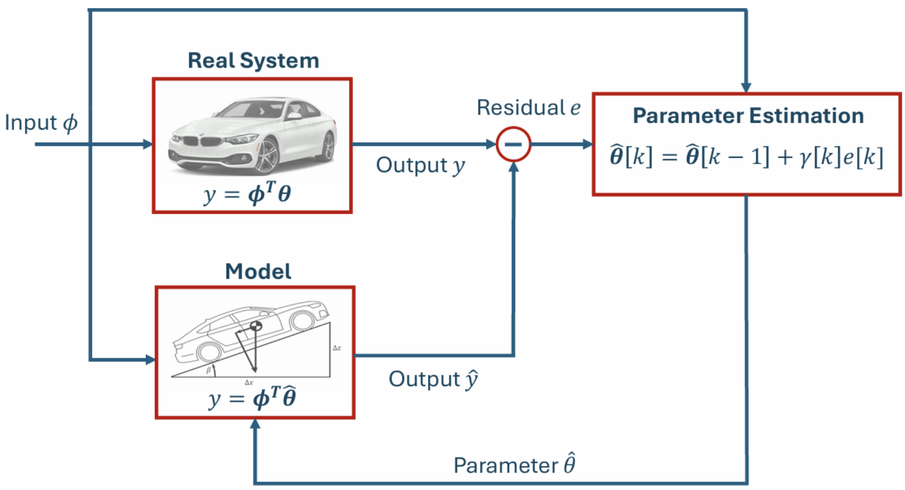

Figure 2: Vehicle Block Diagram: Recursive Least Squares [11]

As the RLS method averages the result across all measurements, Parameter changes are adapted very slowly [7]. In real applications, the estimator should consider current measurements more than those taken a long time ago. This means that the measurement data memory of the estimation algorithm should be forgotten over time with λ as forgetting factor [3], [7] (page 281-283). Therefore, the residuals are weighted with

$$w[k] = \lambda^{N-k} \text{ ,with } 0 < \lambda \leq 1.$$

Following, current measurement is weighted with $w[k] = \lambda^{N-N} = 1$, and the older measurements weights are decreasing exponentially.

$$\hat{\boldsymbol{\theta}} = \left(\boldsymbol{\Phi}^T \boldsymbol{\Lambda} \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^T \boldsymbol{\Lambda} \boldsymbol{y} \quad \text{with} \quad \mathbf{W} = \boldsymbol{\Lambda} = \begin{bmatrix} \lambda^N & & & & & \\ & \lambda^{N-1} & & & & \\ & & \ddots & & & \\ & & & \lambda^2 & & \\ & & & & \lambda & \\ & & & & & 1 \end{bmatrix}$$

Based on the weighted LS, the recursive least squares method with the forgetting factor was derived in [7] (page 281-283), resulting in the correction factor

$$\gamma[k] = \frac{\boldsymbol{P}[k-1]\boldsymbol{\phi}[k]}{\boldsymbol{\phi}^\top[k]\boldsymbol{P}[k-1]\boldsymbol{\phi}[k] + \lambda} \quad \text{with} \quad \boldsymbol{P}[k] = \frac{1}{\lambda}(\boldsymbol{I} - \boldsymbol{\gamma}[k]\boldsymbol{\phi}^T[k])\boldsymbol{P}[k-1]. \tag{11}$$

2.3. Results and Discussion

From the derivation of [7] (page 281-283), a corresponding algorithm of a recursive least square estimation with exponential forgetting factor is illustrated in the appendix 1. Based on this algorithm structure, the corresponding Matlab function **Parameter_Estimation.m** is created, estimating $\hat{\theta}$ with recursive least squares. Therefore, 2 cases are simulated:

- 1. Case: Without Noise on System Output y (see Fig. 3)
- 2. Case: With Noise on System Output y (see Fig. 4)

For the simulation the real θ is defined to be $\theta = 0.5$. The initial estimated parameters are defined to $\hat{\theta}[1] = 0$ and $P[1] = 0$. For each Case, $\hat{\theta}$ is estimated with a forgetting factor $\lambda = 1$ and $\lambda = 0.95$. In Fig. 3, the result of case 1 is plotted. It can be observed that the estimation the estimate converges faster to the actual value for $\lambda = 0.95$, while $\lambda = 1$ converges slower. Since in real world scenarios, such as the Vehicle Longitudinal Dynamics Model from Fig. 1 for vehicle mass estimation, the real theta changes continuously in very short periods of time, it is possible that for $\lambda = 1$, the convergence time is not sufficient to reach the actual value. A forgetting factor is useful for this application [3].

Since the assumption without noise never exists in real-world applications, the simulation results are shown in Fig. 4 with the addition of noisy output signal y of the system. It can be observed that the estimation for a no forgetting $\lambda = 1$ performs significantly better than with a forgetting factor of $\lambda = 0.95$. If the forgetting factor is too small, the recursive least squares model forgets the old data very quickly. This means the estimator trusts the previous estimate less than the new measured values. If the measured values are noisy, the noise is mapped more strongly to the estimation. Therefore, the smoothing effect of the high forgetting factor no longer applies [7] (chapter 12).

As a consequence, for real scenarios such as the estimation of the vehicle mass in Fig. 1, the

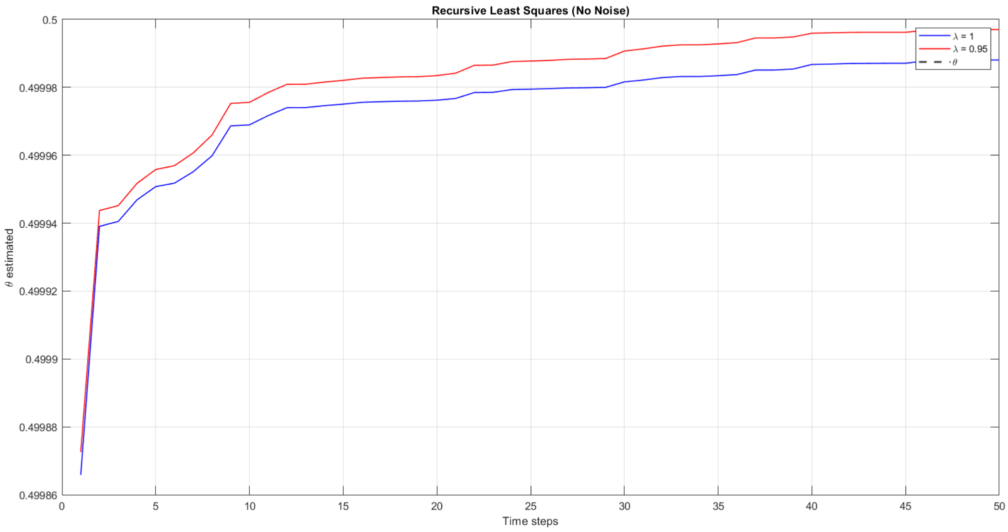

Figure 3: Estimation of $\hat{\theta}$ without Noise on System Output y

selection of the forgetting factor is of crucial importance for the estimation accuracy. Adding a forgetting factor to the RL estimation makes it possible to converge faster toward the real value in fast-changing environmental conditions. However, λ should not be chosen too small, as the system output y contains noise, which the parameter estimation can adapt on. In [7] (page 283) a range of λ between $0.9 < \lambda < 0.995$ for practical applications is suggested.

2.4. Conclusion

This report discusses the application of grey-box models for vehicle parameter estimation in automotive electromobility. In real-time, advanced control and optimization strategies in electric vehicles can be controlled and operated by estimating the vehicle parameters, such as mass or battery characteristics. Therefore, the recursive least squares method with a forgetting factor (RLSeF) for vehicle parameter estimation is introduced, which exponentially weights current measurements more than measurements taken in the past.

Based on the theory, a corresponding Matlab function was developed, simulating recursive least squares with exponential forgetting for parameter estimations
(see Parameter_Estimation.m).
Two cases are compared:

- Estimation with Noisy System Output Signal
- Estimation without Noisy on System Output Signal

The result showed that introducing a forgetting factor improves the estimation quality for no noisy system output, which is essential for real-world applications with fast changing parameters, e.g. vehicle mass or battery capacity. However, in a noisy system environment, an estimation model with a small defined λ can result in amplifying the noise impact on the estimation. In contrast, setting $\lambda = 1$ results in more stable but slower converging estimates, which may not adapt quickly enough to dynamic changes. Therefore, a trade-off must be made when choosing the forgetting factor. The estimation has to be robust (λ not too small) and converge fast ($\lambda < 1$). A range of λ, for practical applications, is suggested to be between between $0.9 < \lambda < 0.995$ [7] (page 283).

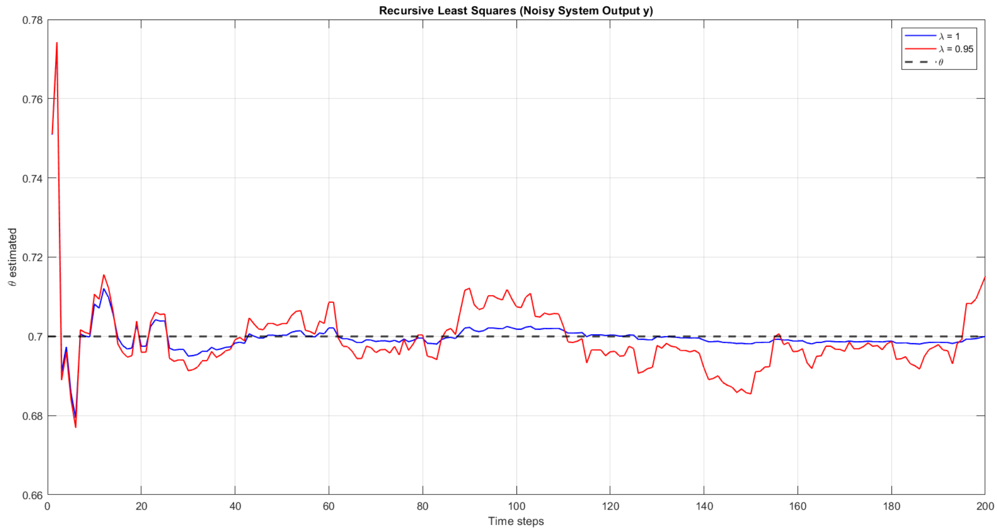

Figure 4: Estimation of $\hat{\theta}$ with Noise on System Output y

In conclusion, the recursive least squares method with exponential forgetting provides an efficient application for parameter estimation in electric vehicles for real-time modeling of dynamic systems.

2.5. Appendix

Algorithm 1 Recursive-Least-Squares with Exponential Forgetting (RLSeF), derived from [7] (page 281-283)

1: **Input:** $y[k]$, $\boldsymbol{\phi}[k]$, $\hat{\boldsymbol{\theta}}[k-1]$, $\boldsymbol{P}[k-1]$, λ

2: $e[k] = y[k] - \boldsymbol{\phi}^T[k]\hat{\boldsymbol{\theta}}[k-1]$

3: $\boldsymbol{\gamma}[k] = \frac{\boldsymbol{P}[k-1]\boldsymbol{\phi}[k]}{\boldsymbol{\phi}^T[k]\boldsymbol{P}[k-1]\phi[k]+1}$

4: $\hat{\boldsymbol{\theta}}[k] = \hat{\boldsymbol{\theta}}[k-1] + \boldsymbol{\gamma}[k]e[k]$

5: $\boldsymbol{P}[k] = (\boldsymbol{I} - \gamma[k]\boldsymbol{\phi}^T[k])\boldsymbol{P}[k-1]$

6: **Output:** $\hat{\boldsymbol{\theta}}[k]$, $\boldsymbol{P}[k]$

References

[1] B. McAulay, B. Li, P. Commins, and H. Du, "Chapter 11 - state and parameter estimation of evs," in *Modeling, Dynamics and Control of Electrified Vehicles* (H. Zhang, D. Cao, and H. Du, eds.), pp. 369–407, Woodhead Publishing, 2018.

[2] P. Shen, M. Ouyang, L. Lu, J. Li, and X. Feng, "The co-estimation of state of charge, state of health, and state of function for lithium-ion batteries in electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 92–103, 2018.

[3] A. Vahidi, A. Stefanopoulou, and H. P. and, "Recursive least squares with forgetting for online estimation of vehicle mass and road grade: theory and experiments," *Vehicle System Dynamics*, vol. 43, no. 1, pp. 31–55, 2005.

[4] A. Costa, E. D. Buccio, M. Melucci, and G. Nannicini, "Efficient parameter estimation for information retrieval using black-box optimization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 7, pp. 1240–1253, 2018.

[5] L. Gianone, J. Bokor, and K. Hangos, "Grey box identification of vehicle dynamics," *IFAC Proceedings Volumes*, vol. 27, no. 8, pp. 1187–1192, 1994. IFAC Symposium on System Identification (SYSID'94), Copenhagen, Denmark, 4-6 July.

[6] S. Rhode and F. Gauterin, "Online estimation of vehicle driving resistance parameters with recursive least squares and recursive total least squares," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 269–276, 2013.

[7] R. Isermann and M. Münchhof, *Identification of dynamic systems: an introduction with applications*, vol. 85. Springer, 2011.

[8] M. T. Heath, *Scientific computing : an introductory survey / Michael T. Heath.* London: McGraw-Hill, 2nd ed ed., 2002.

[9] K. Åström and B. Wittenmark, *Adaptive Control: Second Edition.* Dover Books on Electrical Engineering, Dover Publications, 2013.

[10] N. Lin, C. Zong, and S. Shi, "The method of mass estimation considering system error in vehicle longitudinal dynamics," *Energies*, vol. 12, no. 1, 2019.

[11] A. Chi, J. Wei, T. Wei, G. Song, L. Qin, M. Wei, Y. Gao, and L. Lu, "Hierarchical estimation of vehicle key parameters using kf-rls method," in *2024 8th International Conference on Electrical, Mechanical and Computer Engineering (ICEMCE)*, pp. 1025–1029, 2024.