# RECREATING AN ENGINE EFFICIENCY CURVE

K. SEBESTA

## 1. Introduction

This short paper describes how the author built a sensor suite that returns an efficiency map for an arbitrary car. In previous studies, authors were limited only to efficiency models were available ahead of time.[**?**, **?**, **?**]

Unfortunately, this is not the case in reality. Car manufacturers are notoriously tight-lipped and guard their data jealously. Furthermore, engine data alone does not tell the whole story, as it is ideal data from an ideal engine under ideal conditions. As any driver knows, the manufacturer reported fuel economy is terribly imprecise in real-world situations.

For an optimality study, it is essential to know the true system efficiency, which cannot be predicted from engine data alone. Many other car-specific factors come into play, such as transmission losses, rolling resistance, air resistance, etc... Even when these values can be found, they are oftentimes unreliable (e.g. coefficients of air resistance) and have limited usefulness.

A better approach would take into account the real-world parameters of the individual vehicle in question. This approach should be able to adapt to changing circumstances, such as mass change when a passenger gets out of a car or an air resistance change when a convertible top being is put down. This approach also would allow monitoring motor efficiency with time, opening the door to early fault warning. (A tire low-pressure warning comes to mind.)

The results indicate that this approach can be extended far beyond engine efficiency. Indeed, depending on available information, degenerate versions have already been used to estimate road grade[**?**], vehicle mass[**?**], etc...

## 2. Modeling

Begin by establishing the system model:

**Note 1.** *A table with all model variables is included in the appendix.*

### 2.1. Reference frames.

Two reference frames are used.

- An inertial reference frame $\widehat{ijk}$ linked to the car, where $+\hat{i}$ extends in the direction of forward movement, $+\hat{j}$ points 90 degrees left, and $+\hat{k}$ points straight up from the car.

- A global reference frame $NED$ (North-East-Down) which is the local mercator projection of geoidetic latitude-longitude-altitude coordinates. Note that in order to maintain right-handed orthogonality, *Down* represents negative altitude. Also, note that proper care must be taken while projecting geodetic coordinates into a mercator projection. [**?**] provides a good explanation of the process, along with Matlab code.

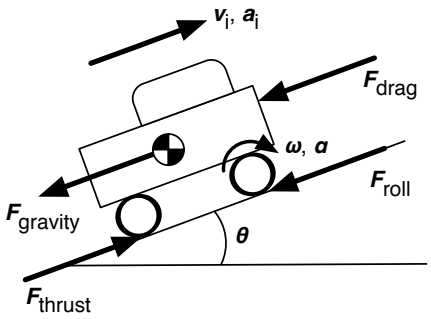


FIGURE 1. Free-body car diagram

2.2. **Kinematics.** Net acceleration is proportional to the sum of forces along the road surface. See Fig. 1. (These kinematic equations can be found in literature, [**?**], but are repeated here for thoroughness.)

$$M_a \dot{v_i} = \Sigma F = F_t - F_{drag} - F_{roll} - F_{gravity} \tag{1}$$

$$F_{drag} = -sign(v_i)\frac{1}{2}\rho C_d A v_i^2 \tag{2}$$

$$F_{roll} = -sign(v_i)C_{rr}(Mg\cos(\theta) + S_k v_i^2) \tag{3}$$

$$F_{gravity} = Mg\sin(\theta) \tag{4}$$

where,

$$M_a = \frac{J}{r_w^2} + M$$

$M_a$ is the equivalent *accelerated* mass, and takes into account the angular momentum of the various spinning parts of the car, where $J$ is the sum of all polar moment of inertias, and $r_w$ the wheel radius.

Note that this model does not separate braking forces (both engine braking and friction brakes) from thrust forces. This is an unnecessary distinction, as only data gathered while the engine is propelling the car forward is interesting.

The car is assumed fixed to the road, and there is a no-slip condition in both longitudinal and latitudinal directions. Furthermore, it is assumed that the car undergoes no rolling rotation ($\phi, \dot{\phi} \equiv 0$). Thus, $v_j, v_k$=0 and any $\Sigma a_j$ or $\Sigma a_k$ represents a centripetal acceleration, i.e. the car is on a curving path:

$$\begin{aligned} a_{j,b} &= S_j v_i^2 \\ a_{k,b} &= S_k v_i^2 \end{aligned} \tag{5}$$

**Note 2.** *$S_j$ and $S_k$, the road straightness coefficients, represent the inverse of road curvature. Straightness coefficients are preferable to road curvature since one does not have to divide by infinity while the car is on a straight section of road. Realistic values for $S_j$ seem to lie on the interval [-0.05, 0.05] and for $S_k$ in [-0.005, 0.005]*

Likewise, the rate of change of $\psi$ and $\theta$ is dependent on the road straightness coefficients and forward velocity:

$$\begin{aligned} \dot{\psi} &= S_j v_i \\ \dot{\theta} &= S_k v_i \end{aligned} \tag{6}$$

**Note 3.** *In the $\widehat{NED}$ coordinate system, $+\psi$ represents a yaw to the left, and $+\theta$ represents a pitch upwards.*

The $N, E, D$ dynamic equations are written in spherical coordinates centered about the body reference frame:

$$\begin{aligned} \dot{N} &= \cos(\psi)\cos(\theta)v_i \\ \dot{E} &= \sin(\psi)\cos(\theta)v_i \\ \dot{D} &= -\sin(\theta)v_i \end{aligned} \tag{7}$$

The engine turns at a fixed rate with respect to the wheels– neutral gear and differential withstanding. The ratio is variable between gears but otherwise constant.

$$\dot{\omega_e} = v_i N_{spd} \tag{8}$$

As we cannot directly measure $N_spd$, we assume that at all times $N_{spd} \equiv 0$ which yields very satisfactory results.

Lastly, the measured acceleration will have slight biases in each direction, due to imperfections in fabrication and in mounting. Note also that an accelerometer in a gravity field measures the pull of gravity along with the net acceleration being applied to the body.

$$\begin{array}{rcl} a_{i,m} & = & a_{i,b} + \frac{M}{M_a} g \sin(\theta) + b_i \\ a_{j,m} & = & a_{j,b} + b_j \\ a_{k,m} & = & a_{k,b} - g \cos(\theta) + b_k \end{array} \tag{9}$$

2.3. **Energy analysis.** Instantaneous power-in is the volume of fuel consumed per second times the amount of energy per volume.

$$P_{in} = k\dot{V}_{fuel} \tag{10}$$

Instantaneous power-out is the thrust force times velocity:

$$P_{out} = F_t v_i \tag{11}$$

Efficiency is

$$\eta = \frac{P_{out}}{P_{in}} \tag{12}$$

Power-out is also equal to torque times angular velocity:

$$P_{out} = T_e \omega_e \tag{13}$$

Thus, by rearranging eqns. (11) and (13) the torque is

$$T_e = F_t \frac{v_i}{\omega_e} \tag{14}$$

**Note 4.** *This result is completely independent of transmission gear ratios.*

## 3. Observer

The model is of the nonlinear form

$$\begin{cases} \dot{\mathbf{x}} & = & A(\mathbf{x}) \\ \mathbf{y}_{t_k} & = & H_k(\mathbf{x}) \end{cases} \tag{15}$$

The Extended Kalman Filter (EKF) is a tried-and-true approach used extensively in science and engineering. While there are other filters available, such as the Unscented Kalman Filter [**?**], the DD1 filter [**?**], the High Gain Kalman Filter [**?**], etc..., the EKF has proven to be a very suitable approach for a wide class of problems, in specific that of vehicle localization[**?**].

3.1. **Continuous-Discrete EKF.** The CD-EKF [**?**] is repeated here for completeness.

For a system with $n_x$ states and $n_y$ measurements:

Prediction:

$$\begin{aligned} \hat{\mathbf{x}}_k^- &= \hat{\mathbf{x}}_{k-1} + \int_{t_{k-1}}^{t_k} \frac{d\hat{\mathbf{x}}}{dt} dt \\ \hat{\mathbf{P}}_k^- &= \hat{\mathbf{P}}_{k-1} + \int_{t_{k-1}}^{t_k} \frac{d\hat{\mathbf{P}}}{dt} dt \end{aligned} \tag{16}$$

where $\frac{d\hat{\mathbf{x}}}{dt} = A(\hat{\mathbf{x}})$ and

$$\frac{d\hat{\mathbf{P}}}{dt} = (\mathbf{A}_c\hat{\mathbf{P}} + \hat{\mathbf{P}}\mathbf{A}_c^T + \mathbf{Q}_c) \quad \Big| \quad \hat{\mathbf{P}}|_{t_{k-1}} = \hat{\mathbf{P}}_{k-1} \tag{17}$$

and $\mathbf{Q}_c$ is a positive semi-definite matrix in $\mathbb{R}^{n_x \times n_x}$

Correction:

$$\begin{aligned} \mathbf{K}_k &= \hat{\mathbf{P}}_k^- \mathbf{H}_k^T (\mathbf{H}_k \hat{\mathbf{P}}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \\ \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{y}_{t_k} - H(\hat{\mathbf{x}}_k^-)) \\ \hat{\mathbf{P}}_k &= (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\hat{\mathbf{P}}_k^- \end{aligned} \tag{18}$$

where $\mathbf{R}_k$ is a positive-definite matrix in $\mathbb{R}^{n_y \times n_y}$

3.2. **Data fusion.** One of the particularities of this problem is the fact that data arrives asynchronously and is in some cases redundant (engine speed is a linear function of car speed, and GPS and DEM both give altitude data).

3.2.1. *Redundancy.* The redundant case is easy to treat. A measurement noise covariance value is assigned that reflects confidence in each particular value, exactly as would be done if the data were not redundant. For instance, the GPS is not very accurate in altitude terms, and tends to get its bias "stuck", so it gets a relatively high covariance value. On the other hand, the DEM is often accurate to sub-cm resolution, so it is assigned a relatively low value.

The advantage of using both is that oftentimes one does not have access to quality DEM data, whereas the GPS gives an output every 200ms.

3.2.2. *Asynchronous data.* The asynchronous case takes more work to treat. In this example, some measurements arrive as a function of time (accelerometer and GPS), some as a function of distance (odometer and engine angular distance), and some as events (fuel input and DEM data).

The asynchronous data necessitates an approach with a variable time step, where a new update/correction is triggered at each time $t_k$ when there is a new data value. This implies a constantly changing measurement matrix $\mathbf{H}_k$.

As shown in Algorithm 1, the differential system equations are integrated until an event is encountered at time $t_k$. At this moment, one must first determine what measurements are present and stack them into the output vector, $\mathbf{y}_k$. Then the measurement Jacobian matrix $\mathbf{H}_k$ must be assembled in the same fashion, again by stacking the individual row vectors associated with the measurements.

Once $\mathbf{H}_k$ is prepared, the discrete update/correction of the EKF proceeds as normal.[1]

Additionally, the measurement covariance matrix $\mathbf{R}_k$ must change as a function of the time step and of the available data. It is always of dimension $\mathbb{R}^{n_y \times n_y}$, where

$$n_y = n_{\mathbf{y}_{t_k}}$$

3.3. **Observer psuedo-code.** The pseudo-code shows the advantage of the CD-EKF in a realtime setting. The integrator continuously integrates until a new measurement arrives. Thus, the most up-to-date estimat ion is always available and the processor can immediately correct the state estimate.

The continuous model and covariance differential equation system is integrated with a Runge-Kutta-4. Note, however, that the interval is $\Delta t = t_k - t_{k-1}$, and as such changes every iteration.

Due to the rapid nature of the measurements in this case (never slower than 40Hz), there is little to be gained by using a variable step integrator.

## 4. Measurement hardware

The sensor suite was designed to be cheap and easy to produce. Ideally, it uses information that is already available to the car's on-board management computer. The one-off cost for the measurement hardware was $50 without GPS, $100 with. If it were to be manufactured in bulk, its cost would fall well under $10. Indeed,

[1]If the discrete-discrete approach had be used instead of the CD-EKF, $\mathbf{Q}$ would have varied at each iteration as a function of time since the covariance of a gaussian noise of covariance $q$ after a time $\Delta t$ is $q\Delta t$. I.e., the longer the observer waits without any measurement data, the higher the uncertainty, which is represented by a growing $q$. Thus in the discrete-discrete case:

$$\mathbf{Q}_d = \mathbf{Q}_c \Delta t_{t_k} \tag{19}$$

**Algorithm 1:** Observer loop

**while** *car is on* **do**

  **while** *no new measurement* **do**

    integrate $\frac{d\hat{\mathbf{x}}(t)}{dt}$ and $\frac{d\hat{\mathbf{P}}}{dt}$

  **end**

  $\hat{\mathbf{x}}_k^- \leftarrow \hat{\mathbf{x}}(t_k)$

  $\hat{\mathbf{P}}_k^- \leftarrow \hat{\mathbf{P}}(t_k)$

  `Measurement`

  $\mathbf{y}_{t_k} \leftarrow$ event(s) at time $t_k$

  $\Delta t \leftarrow t_k - t_{k-1}$

  `Correction`

  construct new measurement Jacobian, $\mathbf{H}_k$

  construct new measurement covariance, $\mathbf{R}_k$

  $\mathbf{K}_k \leftarrow \hat{\mathbf{P}}_k^- \mathbf{H}_k^T (\mathbf{H}_k \hat{\mathbf{P}}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$

  $\hat{\mathbf{P}}_k \leftarrow (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \hat{\mathbf{P}}_k^-$

  $\hat{\mathbf{x}}_k \leftarrow \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_{t_k} - H(\hat{\mathbf{x}}_k^-)$

  `Update loop`

  $\hat{\mathbf{x}}(t_k) \leftarrow \hat{\mathbf{x}}_k$

  $\hat{\mathbf{P}}(t_k) \leftarrow \hat{\mathbf{P}}_k$

  $k \leftarrow k + 1$

**end**

with recent advances in microcontrollers and prices, it should be easy to redesign and manufacture the package on a one-off basis for under $20.

4.1. **Microcontroller.** The hardware is built on an AVR ATMega324p microcontroller, similar to the AVR ATMega168 that powers the MPGuino, that timestamps and logs all information to an SD card. The microcontroller clock is driven by an internal RC oscillator, but it is kept in sync by the GPS's pulse-per-second (PPS) signal, guaranteed to an accuracy of at least $1\mu$s.

4.2. **Odometer.** Distance is measured by the on-board Vss, a circuit that sends a pulse triggered from having rolled a certain distance. As each pulse represents a certain distance, it is trivial to log when each pulse arrives and increment the driven distance in the observer.

4.3. **GPS.** The GPS is a San Jose FV-M8. It runs at 5Hz and communicates over the serial port.

4.4. **Accelerometer.** The accelerometer is the 3-axis STMicro LIS3LV02DL.

4.5. **Fuel flow meter.** The sensor records fuel injector pulse width, and thus measures how long the injector was open and infers how much gasoline was injected. This method is accurate to ±0.1%.[**?**]

4.6. **Digital Elevation Map.** While the NASA developed SRTM data [**?**] is widely available, it is of limited resolution and, by its nature, misses nuances such as the difference between bridges and tunnels.

As free, high-quality mapping data [**?**] becomes more and more available, it can be used to better the model.

## 5. Data gathering

Data was gathered through normal car driving. Not requiring special actions on the part of drivers is an essential part to the approach.

If it is necessary, one can eliminate these two sensors, reducing hardware costs and simplifying the programming task. However, despite the robustness of the observer, it is still preferable to use GPS and DEM when possible.

It must be noted that the efficiency values are globally too high. We suspect that there is a poor estimation of force.

It is interesting to note in these plots that observability is clearly confirmed and the observer performs to satisfaction. Even in the more complicated urban cycle, the error remains sufficiently small.

5.1. **Efficiency map.** An engine efficiency map, Figure 2, is made by averaging together efficiency values as functions of rpm and torque.

Note that the efficiency graph shows a continuous and smooth progression from low to high efficiency, and shows a sharp drop off to the right as the rpm reach the redline.

## 6. Validation

We testerd the on a dynamometer in order to validate the proposed method. The test was performed by putting the dynamometer into constant speed operation, and varying the torque produced by the Smart's engine.

The constant speed steps were taken in units of 100rpm, over a range from 2500rpm to 6000rpm. The torque was varied between 25Nm and 80Nm.

As can be seen in Figure 4, the curve fitted results correspond nicely to the observed results in Figure 3. While there is a scaling effect versus the experimental data results, this is easily corrected by calculating some of the parameters rather than using the reported manufacturer values, e.g. $C_{rr}$, $C_d$, fuel injection calibration, etc...

What is more important is that the maximum efficiency occurs at roughly the same point in the two graphs, and the overall form is identical.

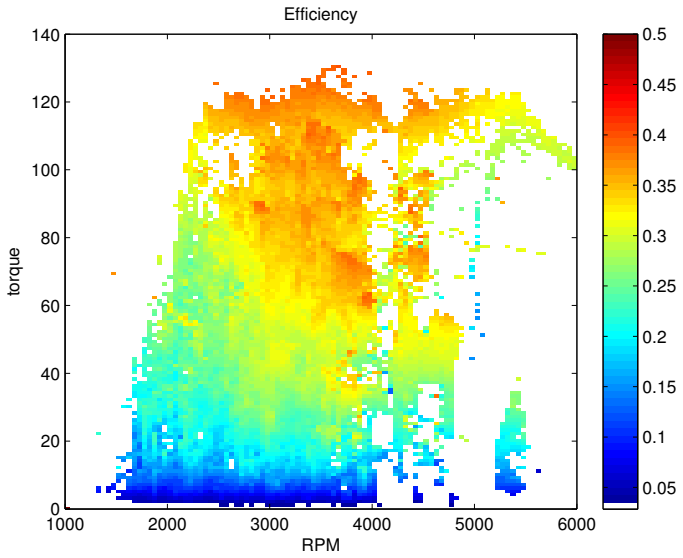


FIGURE 2. Efficiency map

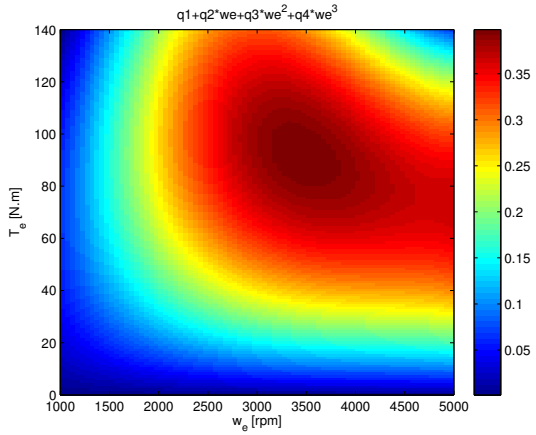


FIGURE 3. Efficiency map as fitted curve, calculated from driving data. Peak efficiency occurs at 3500rpm, 90Nm

## APPENDIX: MODEL VARIABLES

| Variable | Description | Units |
|---|---|---|
| $x_i$ | car displacement in the $\hat{i}$ axis | $[m]$ |
| $v_i$ | car speed in the $\hat{i}$ axis | $[\frac{m}{s}]$ |
| $a_i, a_j, a_k$ | car acceleration in the $\widehat{ijk}$ axes | $[\frac{m}{s^2}]$ |
| $F_t$ | thrust force along $\hat{i}$ axis | $[N]$ |
| $\psi$ | car heading | $[rad]$ |
| $\theta$ | car pitch | $[rad]$ |
| $S_j$ | road straightness coefficient (horizontal plane) | $[\frac{rad}{m}]$ |
| $S_k$ | road straightness coefficient (vertical plane) | $[\frac{rad}{m}]$ |
| $N, E, D$ | Northing, Easting, and Down (local projection plane) | $[m]$ |
| $E_{in}$ | fuel energy content in | $[J]$ |
| $P_{in}$ | fuel power in | $[W]$ |
| $b_i, b_j, b_k$ | accelerometer biases in the $\widehat{ijk}$ axes | $[\frac{m}{s^2}]$ |
| $\theta_e$ | engine angular displacement | $[rad]$ |
| $\omega_e$ | engine angular speed | $[\frac{rad}{s}]$ |
| $C_{rr}$ | rolling resistance coeffecient | [-] |

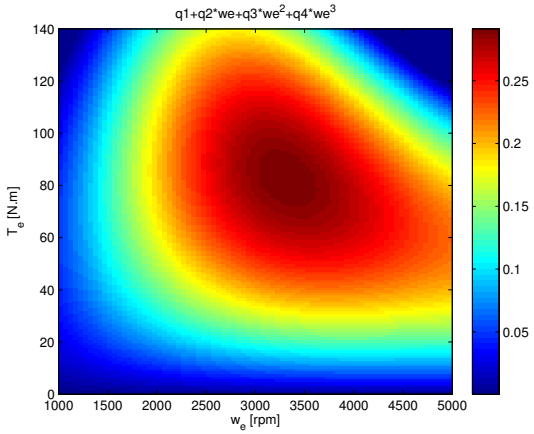


FIGURE 4. Efficiency map as fitted curve, calculated from dynamometer data. Peak efficiency occurs at 3300rpm, 85Nm

## 7. References

### References

[Collaborative effort(2010a)] Collaborative effort, 2010a. MPGuino.
URL `http://ecomodder.com/wiki/index.php/MPGuino`

[Collaborative effort(2010b)] Collaborative effort, 2010b. Open Street Map.
URL `www.openstreetmap.org`

[Gauthier and Kupka(2001)] Gauthier, J., Kupka, I., 2001. Deterministic Observation Theory and Applications. Cambridge University Press.

[Ivarsson et al.(2009)Ivarsson, Aslund, and Nielsen] Ivarsson, M., Aslund, J., Nielsen, L., 2009. Look-ahead control for heavy trucks to minimize trip time and fuel consumption. Control Engineering Practice 17, 245–254.

[Julier and Uhlmann(1997)] Julier, S., Uhlmann, J., 1997. A new extension of the kalman filter to nonlinear systems. Vol. 3068. pp. 182–193.

[Mourllion et al.(2005)Mourllion, Gruyer, Lambert, and Glaser] Mourllion, B., Gruyer, D., Lambert, A., Glaser, S., 2005. Kalman filters predictive steps comparison for vehicle localization. IEEE/RSJ International Conference on Intelligent Robots and Systems.

[NASA(2010)] NASA, 2010. SRTM.
URL `http://www2.jpl.nasa.gov/srtm/`

[Nørgaard et al.(2000)Nørgaard, Poulsen, and Ravn] Nørgaard, M., Poulsen, N., Ravn, O., April 2000. Advances in derivative-free state estimation for nonlinear systems. Tech. Rep. (IMM-REP-1998-15), Technical University of Denmark.

[Rousseau(2008)] Rousseau, G., December 2008. Véhicule hybride et commande optimale. Ph.D. thesis, Ecole Nationale Superieur des Mines de Paris.

[Sahlholm et al.(1997)Sahlholm, Jansson, Kozica, and Johansson] Sahlholm, P., Jansson, H., Kozica, E., Johansson, K., 1997. A sensor and data fusion algorithm for road grade estimation. In: 5th IFAC Symposium on Advances in Automotive Control. Monterey, CA, USA.

[Vahidi et al.(2003)Vahidi, Stefanopoulou, and Peng] Vahidi, A., Stefanopoulou, A., Peng, H., 2003. Experiments for online estimation of heavy vehicle's mass and time-varying road

grade. In: Proceedings of the 2003 ASME International Mechanical Engineering Congress and Exposition.

[van Keulen et al.(2009)van Keulen, de Jager, Serrarens, and Steinbuch] van Keulen, T., de Jager, B., Serrarens, A., Steinbuch, M., 2009. Optimal energy management in hybrid electric trucks using route information.

[Wasmeier(2010)] Wasmeier, P., 2010. Geodetic transformation.