State/Parameter Estimation of a Small-scale CHP model

Juan I. Videla
Telemark University College
juan.videla@hit.no

Bernt Lie
Telemark University College
bernt.lie@hit.no

Abstract

The state/parameter estimation problem is studied for a small-scale ICE CHP model. Three main groups of estimators with significant performance and complexity differences are analyzed: the Extended Kalman Filter (EKF) as an extension of the classical Kalman filter, the generalized unscented Kalman filter (UKF) that uses the unscented transformation, and particle filtering like the particle filter with resampling (PFr) and the Ensemble Kalman Filter (EnKF)

The internal combustion engine is modeled as a mean-value engine model connected with a static generator model and the heat recovery circuit is modeled with two lumped heat exchanger models, one for the coolant circuit and the other for the exhaust gases. The coolant circuit is connected with the engine through a lumped inner engine thermal model.

Experimental data sets are artificially generated to test the different estimators. Dynamic parameters of the mean-value engine model are identify when the CHP model is simulated in open loop. Additionally, relevant heat transfer coefficients of the heat recovery circuit are monitored when the model is simulated in closed loop.

1. Introduction

There is a growing interest in the use of small-scale and micro cogeneration systems in the residential sector to produce both useful thermal energy and electricity. A review and comparison of these systems presented in [1] indicates that internal combustion engine (ICE) based combined heat and power (CHP) units are the most suitable products available today in the market for the residential sector. However, they do need regular maintenance and servicing, and most of them operate with simple high level control policies (e.g. on/off). Advanced controllers (e.g. optimization based techniques) and better monitoring techniques can be used with an online state/parameter estimator to reduce the operating costs.

It is of interest to study the state/parameter estimation problem of a small-scale CHP unit with the purposes of control, maintenance, and identification.

Online parameter identification of a CHP unit can be achieved using recursive state/parameter estimators. Fast and slow varying parameters tracking is of interest [2] not only to identify the physical parameters but also to detect failures and components degradation. Additional difficulties for the estimation problem include: nonlinearities, a limited number of noisy measurements, unmodeled dynamics, and simplifying assumptions.

For linear systems with normally distributed process and measurement noise the optimal recursive estimator is the Kalman filter [3]. State estimation for nonlinear systems is considerably more difficult and admits a wider variety of suboptimal solutions [4]. Three main groups of techniques with significant performance and complexity differences are analyzed: classical nonlinear extensions of the Kalman filter like the Extended Kalman Filter (EKF), unscented filtering with the generalized Unscented Kalman Filter (UKF), and particle filtering like the particle filter with resampling (PFr) and the Ensemble Kalman Filter (EnKF). The parameters are directly estimated using the parameter state-augmented approach and the discrete version of the estimators is implemented.

A simple CHP model structure of a small-scale spark ignition (SI) ICE based cogeneration system is taken [5]. The main subsystems are: the SI ICE engine, the generator, and the outer loops heat recovery circuits.

The paper is organized as follows: first a description of the CHP model including its main assumptions is given. Then, the recursive nonlinear state/parameter estimators algorithms and their tuning parameters are described. In section 4, the implementation details and the estimation results are presented. Finally, the conclusions with a general comparison of the nonlinear estimators for this problem are given.

2. Model description

In [5,6] a small-scale SI ICE based cogeneration model with a simple dynamic engine model is presented. The CHP main subsystems are the reciprocating spark ignition (SI) internal combustion engine (ICE), the generator, and the outer loop heat recovery circuits.

Table 1. MVEM and generator models

Throttle Valve:

$$A_{\mathrm{th}} = \frac{\pi d_{\mathrm{th}}^2}{4}\left(1 - \frac{\cos\left(\alpha_{\mathrm{th},0} + \left(\frac{\pi}{2} - \alpha_{\mathrm{th},0}\right)u_{\mathrm{th}}\right)}{\cos\alpha_{\mathrm{th},0}}\right) + A_{\mathrm{th},0}$$

$$\text{if } \frac{p_{\mathrm{ab}}}{p_{\mathrm{mn}}} < \frac{1}{2} \quad \dot{m}_{\mathrm{th}} = C_{\mathrm{th}} A_{\mathrm{th}} \frac{p_{\mathrm{ab}}}{\sqrt{R_\beta \vartheta_{\mathrm{ab}}}} \frac{1}{\sqrt{2}}$$

$$\text{else } \quad \dot{m}_{\mathrm{th}} = C_{\mathrm{th}} A_{\mathrm{th}} \frac{p_{\mathrm{ab}}}{\sqrt{R_\beta \vartheta_{\mathrm{mn}}}} \left[2\frac{p_{\mathrm{mn}}}{p_{\mathrm{ab}}}\left(1 - \frac{p_{\mathrm{mn}}}{p_{\mathrm{ab}}}\right)\right]^{\frac{1}{2}}$$

Intake Manifold:

$$\vartheta_{\mathrm{mn}} = \vartheta_{\mathrm{ab}} + \Delta\vartheta_{\mathrm{mn}}$$

$$\frac{d}{dt}p_{\mathrm{mn}} = \frac{R_\beta \vartheta_{\mathrm{mn}}}{V_{\mathrm{mn}}}\left[\dot{m}_{\mathrm{th}} - \dot{m}_\beta\right]$$

Cylinders air mass flows:

$$\dot{m}_{\mathrm{ex}} = \dot{m}_\beta + \dot{m}_\varphi$$

$$\dot{m}_{\mathrm{ex}} = \frac{p_{\mathrm{mn}}}{R_\beta \vartheta_{\mathrm{mn}}} \frac{V_d}{4\pi} \eta_{\mathrm{vl}} \omega_{\mathrm{en}}$$

$$\dot{m}_\beta = \lambda \sigma_0 \dot{m}_\varphi$$

$$\eta_{\mathrm{vl}} = \eta_{\mathrm{vl},1}(\omega_{\mathrm{en}}) \eta_{\mathrm{vl},2}(p_{\mathrm{mn}})$$

$$\eta_{\mathrm{vl},1}(\omega_{\mathrm{en}}) = \left(\gamma_0 + \gamma_1 \omega_{\mathrm{en}} + \gamma_2 \omega_{\mathrm{en}}^2\right)$$

$$\eta_{\mathrm{vl},2}(p_{\mathrm{mn}}) = \frac{V_c + V_d}{V_d} - \frac{V_c}{V_d}\left(\frac{p_{\mathrm{ex}}}{p_{\mathrm{mn}}}\right)^{\frac{1}{\kappa_\beta}}$$

Cylinders torque generation:

$$T_{\mathrm{en}} = p_{\mathrm{meb}} \frac{V_d}{4\pi}$$

$$p_{\mathrm{meb}} = \eta_{\mathrm{td}}\left(\omega_{\mathrm{en}}\right) p_{\mathrm{me}\,\varphi} - p_{\mathrm{mef}}\left(\omega_{\mathrm{en}}\right) - p_{\mathrm{mepg}}$$

$$\eta_{\mathrm{td}} p_{\mathrm{me}\,\varphi} = \left(\eta_0 + \eta_1 \omega_{\mathrm{en}} + \eta_2 \omega_{\mathrm{en}}^2\right) H_l \frac{\dot{m}_\varphi 4\pi}{\omega_{\mathrm{en}} V_d}$$

$$p_{\mathrm{mef}} = \left(\beta_0 + \beta_1 \omega_{\mathrm{en}} + \beta_2 \omega_{\mathrm{en}}^2\right) \frac{4\pi}{V_d}$$

$$p_{\mathrm{mepg}} = p_{\mathrm{ex}} - p_{\mathrm{mn}}$$

Cylinders energy balance:

$$\dot{H}_{\mathrm{ex}} = \dot{H}_\beta + \dot{H}_{\Delta c} - \dot{W}_{\mathrm{en}} - \dot{W}_{\mathrm{pg}} - \dot{Q}_f - \dot{Q}_{\mathrm{cg,cw}}$$

$$\dot{Q}_{\mathrm{cg,cw}} = \left(\delta_0 + \delta_1 p_{\mathrm{meb}} + \delta_2 p_{\mathrm{meb}}^2 + \delta_3 p_{\mathrm{meb}}^3\right) \frac{\omega_{\mathrm{en}}}{4\pi}$$

$$\dot{W}_{\mathrm{en}} = \omega_{\mathrm{en}} T_{\mathrm{en}}$$

$$\dot{W}_{\mathrm{pg}} = \omega_{\mathrm{en}} p_{\mathrm{mepg}} \frac{V_d}{4\pi}$$

$$\dot{H}_{\Delta c} = H_l \frac{\dot{m}_\varphi 4\pi}{\omega_{\mathrm{en}} V_d}$$

Cylinders angular velocity:

$$\frac{d}{dt}\omega_{\mathrm{en}} = \frac{1}{I_{\mathrm{en}}}\left[T_{\mathrm{en}} - T_{\mathrm{ld}}\right]$$

Generator:

$$T_{\mathrm{ld}} = \frac{P_{\mathrm{ec}}}{\omega_{\mathrm{en}} \eta_{\mathrm{ec}}} P_{\mathrm{ec}}$$

The SI ICE is modeled as a lumped parameter mean-value engine model [7–9] and the generator is considered as a simple static model (see Fig. 1). It is assumed that the engine is controlled to operate at its stochiometric ratio with a fixed ignition angle equal to the value of the engine maps. The volumetric and thermodynamic efficiencies are approximated using second order polynomials of the angular velocity and the intake manifold pressure (for details see [10]). The heat flow rate transferred from the cylinder gases into the cylinder walls, $\dot{Q}_{\mathrm{cg,cw}}$, is also approximated using a second order polynomial of the current mean effective pressure (i.e. load) and the angular velocity.

The heat recovery system (see Fig. 2) connects the engine inner thermal model with a secondary circuit (e.g. a buffer tank). The engine cooling and the exhaust gases heat recovery circuits are coupled with the secondary system using two simple heat exchanger models. The pipes connecting the different components are assumed adiabatic and the pumps are operated at constant mass flow rates, that is, $du_{\mathrm{p1}}/dt = 0$ and $du_{\mathrm{p2}}/dt = 0$.

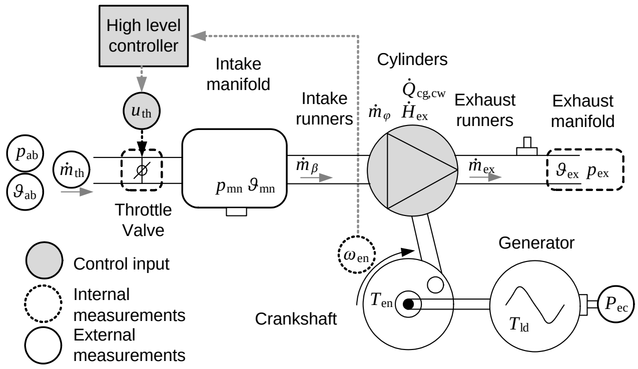

Figure 1. MVEM and electrical generator

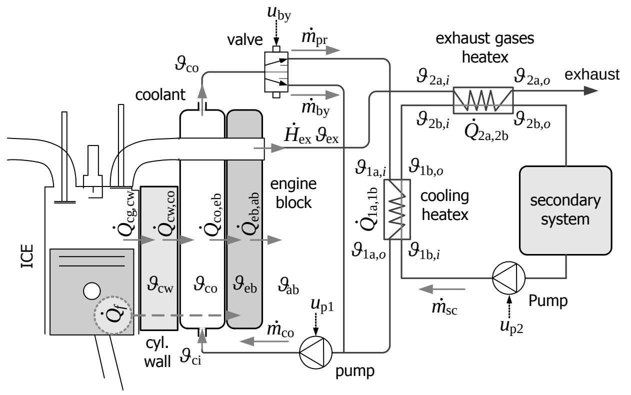

Figure 2. HRC and engine inner thermal model

A parameterized models of the CHP unit can be seen in table 1 for the MVEM and the electrical generator, and in table 2 for the heat recovery circuit and the engine inner thermal model. The model notation is summarized in table 8.

3. Nonlinear estimation

3.1 Model equations

The CHP model can be written in the general discrete nonlinear state space form:

$$\begin{aligned} x_k &= f_{k-1}\left(x_{k-1}, u_{k-1}, w_{k-1}\right) \\ y_k &= h_k\left(x_k, v_k\right) \end{aligned} \quad (1)$$

where $f_{k-1} : \mathbb{R}^{n_x+n_u+n_w} \mapsto \mathbb{R}^{n_x \times 1}$ is the discrete state function, $x_k \in \mathbb{R}^{n_x \times 1}$ is the discrete state vector,

Table 2. HRC models

Inner Engine Thermal Model:
$\frac{d}{dt}\vartheta_{\mathrm{cw}} = \frac{1}{c_{\mathrm{cw}}m_{\mathrm{cw}}}(\dot{Q}_{\mathrm{cg,cw}} - \dot{Q}_{\mathrm{cw,co}})$
$\frac{d}{dt}\vartheta_{\mathrm{co}} = \frac{1}{c_{\mathrm{co}}m_{\mathrm{co}}}[c_{\mathrm{co}}\dot{m}_{\mathrm{co}}(\vartheta_{\mathrm{ci}} - \vartheta_{\mathrm{co}}) + \dot{Q}_{\mathrm{cw,co}} - \dot{Q}_{\mathrm{co,eb}}]$
$\frac{d}{dt}\vartheta_{\mathrm{eb}} = \frac{1}{c_{\mathrm{eb}}m_{\mathrm{eb}}+c_{\mathrm{eo}}m_{\mathrm{eo}}}(\dot{Q}_f + \dot{Q}_{\mathrm{co,eb}} - \dot{Q}_{\mathrm{eb,ab}})$
$\dot{Q}_{\mathrm{cw,co}} = \alpha_{\mathrm{cw,co}}A_{\mathrm{cw,co}}[\vartheta_{\mathrm{cw}} - \frac{1}{2}(\vartheta_{\mathrm{ci}} + \vartheta_{\mathrm{co}})]$
$\dot{Q}_{\mathrm{co,eb}} = \alpha_{\mathrm{co,eb}}A_{\mathrm{co,eb}}[\frac{1}{2}(\vartheta_{\mathrm{ci}} + \vartheta_{\mathrm{co}}) - \vartheta_{\mathrm{eb}}]$
$\dot{Q}_{\mathrm{eb,ab}} = \alpha_{\mathrm{eb,ab}}A_{\mathrm{eb,ab}}(\vartheta_{\mathrm{eb}} - \vartheta_{\mathrm{ab}})$
$\dot{Q}_f = \frac{1}{4\pi}p_{\mathrm{mef}}V_d\omega_{\mathrm{en}}$
Cooling circuit:
$\dot{m}_{\mathrm{co}} = \dot{m}_{\mathrm{by}} + \dot{m}_{\mathrm{pr}}$
$\dot{m}_{\mathrm{by}} = \varpi_0 u_{\mathrm{by}}$
$\dot{m}_{\mathrm{co}} = \xi_{\mathrm{p1},0} + \xi_{\mathrm{p1},1}u_{\mathrm{p1}} + \xi_{\mathrm{p1},2}u_{\mathrm{p1}}^2$
$\frac{d}{dt}\vartheta_{\mathrm{1a}} = \frac{1}{c_{\mathrm{1a}}m_{\mathrm{1a}}}[c_{\mathrm{1a}}\dot{m}_{\mathrm{1a}}(\vartheta_{\mathrm{1a},i} - \vartheta_{\mathrm{1a},o}) - \dot{Q}_{\mathrm{1a,1b}}]$
$\frac{d}{dt}\vartheta_{\mathrm{1b}} = \frac{1}{c_{\mathrm{1b}}m_{\mathrm{1b}}}[c_{\mathrm{1b}}\dot{m}_{\mathrm{1b}}(\vartheta_{\mathrm{1b},i} - \vartheta_{\mathrm{1b},o}) + \dot{Q}_{\mathrm{1a,1b}}]$
$\dot{Q}_{\mathrm{1a,1b}} = \frac{1}{2}\alpha_{\mathrm{1a,1b}}A_{\mathrm{1a,1b}}[(\vartheta_{\mathrm{1a},i} + \vartheta_{\mathrm{1a},o}) - (\vartheta_{\mathrm{1b},i} + \vartheta_{\mathrm{1b},o})]$
Exhaust circuit:
$\frac{d}{dt}\vartheta_{\mathrm{2a}} = \frac{1}{c_{\mathrm{2a}}m_{\mathrm{2a}}}[c_{\mathrm{2a}}\dot{m}_{\mathrm{2a}}(\vartheta_{\mathrm{2a},i} - \vartheta_{\mathrm{2a},o}) - \dot{Q}_{\mathrm{2a,2b}}]$
$\frac{d}{dt}\vartheta_{\mathrm{2b}} = \frac{1}{c_{\mathrm{2b}}m_{\mathrm{2b}}}[c_{\mathrm{2b}}\dot{m}_{\mathrm{2b}}(\vartheta_{\mathrm{2b},i} - \vartheta_{\mathrm{2b},o}) + \dot{Q}_{\mathrm{2a,2b}}]$
$\dot{Q}_{\mathrm{2a,2b}} = \frac{1}{2}\alpha_{\mathrm{2a,2\,b}}A_{\mathrm{2a,2b}}[(\vartheta_{\mathrm{2a},i} + \vartheta_{\mathrm{2a},o}) - (\vartheta_{\mathrm{2b},i} + \vartheta_{\mathrm{2b},o})]$
Secondary Circuit:
$\dot{m}_{\mathrm{sc}} = \xi_{\mathrm{p2},0} + \xi_{\mathrm{p2},1}u_{\mathrm{p2}} + \xi_{\mathrm{p2},2}u_{\mathrm{p2}}^2$

$u_{k-1} \in \mathbb{R}^{n_u\times 1}$ is the discrete input, $w_{k-1} \in \mathbb{R}^{n_w\times 1}$ is the discrete process noise vector, $h_k : \mathbb{R}^{n_x+n_v} \mapsto \mathbb{R}^{n_x\times 1}$ is the discrete output function, $v_k \in \mathbb{R}^{n_v\times 1}$ is the discrete measurement noise vector, $y_k \in \mathbb{R}^{n_y\times 1}$ is the output vector, and k is the time index. The noise vector sequences $\{w_{k-1}\}$ and $\{v_k\}$ are assumed Gaussian, white, zero-mean, uncorrelated, and have the known covariance matrices $Q_k \in \mathbb{R}^{n_x\times n_w}$ and $R_k \in \mathbb{R}^{n_y\times n_v}$.

3.2 Augmented states

The augmented state space approach can be directly used to simultaneously solve the state and the parameter estimation problem (e.g. see [11]). An augmented state space representation is formulated by adding the vector of parameters to be estimated $\theta_k \in \mathbb{R}^{n_\theta\times 1}$ as new states:

$$\begin{aligned}\begin{bmatrix} x_k \\ \theta_k \end{bmatrix} &= \begin{bmatrix} f_{k-1}\left(x_{k-1}, u_{k-1}, w_{k-1}^{(x)}\right) \\ \theta_{k-1} + T_s w_{k-1}^{(\theta)} \end{bmatrix} \qquad (2)\\ y_k &= h_k\left(x_k, v_k\right) \qquad (3)\end{aligned}$$

where T_s is the sampling time step, $w_{k-1}^{(x)} \in \mathbb{R}^{n_w^{(x)}\times 1}$ is the process noise vector that affects the original states, and $w_{k-1}^{(\theta)} \in \mathbb{R}^{n_w^{(\theta)}\times 1}$ is the process noise vector that affects the added parameter states. The noise vector sequences $\{w_{k-1}\}$ and $\{v_k\}$ are assumed Gaussian, white, zero-mean, uncorrelated, and have the known covariance matrices $Q_k = \mathrm{blkdg}(Q_k^{(x)}, Q_k^{(\theta)}) \in \mathbb{R}^{(n_x+n_\theta)\times(n_w^{(x)}+n_w^{(\theta)})}$ and $R_k \in \mathbb{R}^{n_y\times n_v}$.

$$\begin{aligned} w_k &\sim \mathcal{N}(0, \mathrm{blkdg}(Q_k^{(x)}, Q_k^{(\theta)})) \\ v_k &\sim \mathcal{N}(0, R_k)\end{aligned}$$

During the propagation step the augmented states corresponding to parameters θ_k are considered equal to the previous time step θ_{k-1} with some additive process noise $w_{k-1}^{(\theta)}$. If it is assumed that the parameters do not change at all, then there is no process noise vector $w_{k-1}^{(\theta)}$, but for the more general case of time-varying parameters (or where component degradation may happen), the value of $Q_k^{(\theta)}$ will be given by the admissible range of variation of θ_k. During the measurement update step the parameter values are corrected.

For notation simplicity, in the estimators algorithms that follow, the augmented state vector is referred as x_k, the state augmented function (2) is referred as f_{k-1}, and the augmented process noise vector is referred as w_{k-1}.

3.3 Nonlinear Recursive Estimators

The nonlinear estimation problem can be formulated as a recursive Bayesian estimation problem with a propagation and a measurement update steps. This is the optimal way of predicting a state probability density function (pdf) $p(x_k)$ for any system in state space representation with process and measurement noise[1].

Assuming that the initial state pdf $p(x_0)$, the process noise pdf $p(w_{k-1})$, and the measurement noise pdf $p(v_k)$ are known, a recursive solution of the estimation problem can be found using first the Chapman-Kolmogorov equation to calculate the a priori pdf for the state x_k based on the last measurement y_{k-1} (propagation step)

$$p(x_k|y_{k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{k-1})dx_{k-1} \qquad (4)$$

where $p(x_k|x_{k-1})$ can be calculated from the state function f_{k-1} and the pdf of the process noise w_k.

And then, the Bayes rule to update the pdf of the state x_k with the new measurement y_k (measurement update)

$$p(x_k|y_k) = \frac{p(y_k|x_k)\,p(x_k|y_{k-1})}{\int p(y_k|x_k)\,p(x_k|y_{k-1})\,dx_k} \qquad (5)$$

where $p(y_k|x_k)$ is available from our knowledge of the output function h_k and the pdf of v_k, $p(x_k|y_{k-1})$ is known from (4). Although the initial state pdf

[1] Markov process of order one

$p(x_0)$, the process noise pdf $p(w_{k-1})$, and the measurement noise pdf $p(v_k)$ are needed to solve the recursive Bayesian estimation, no specific statistical distribution is required.

The recursive relations (4) and (5) used to calculate the posterior pdf $p(x_k|y_k)$ are a conceptual solution and only for very specific cases can be solved analytically. In general, approximations are required for practical problems. Three main groups of suboptimal techniques with significant performance and computational cost differences are used to approximate the recursive Bayesian estimation problem: the classical nonlinear extensions of the Kalman filter, the unscented filtering techniques, and the particle filtering approaches.

3.3.1 Extended Kalman Filters

Table 3. EKF algorithm

Initialization:

$\hat{x}_{0|0} = E(x_0)$

$P_{0|0} = E[(x_0 - \hat{x}_{0|0})(x_0 - \hat{x}_{0|0}^T)]$

for $k = 1, 2, \ldots$

Propagation step:

(a priori covariance estimate)

$F_{k-1} = \left.\frac{\partial f_{k-1}}{\partial x_{k-1}}\right|_{\hat{x}_{k-1|k-1}} \quad L_{k-1} = \left.\frac{\partial f_{k-1}}{\partial w_{k-1}}\right|_{\hat{x}_{k-1|k-1}}$

$P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^T + L_{k-1} Q_{k-1} L_{k-1}^T$

(a priori state-output estimate)

$\hat{x}_{k|k-1} = f_{k-1}(\hat{x}_{k-1|k-1}, u_{k-1}, 0)$

$\hat{y}_{k|k-1} = h_k(\hat{x}_{k|k-1}, 0)$

Measurement update:

(Kalman gain calculation)

$H_k = \left.\frac{\partial h_k}{\partial u_k}\right|_{\hat{x}_{k|k-1}} \quad M_k = \left.\frac{\partial h_k}{\partial v_k}\right|_{\hat{x}_{k|k-1}}$

$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + M_k R_k M_k^T)^{-1}$

(a posteriori state-covariance estimate)

$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - \hat{y}_{k|k-1})$

$P_{k|k} = (I - K_k H_k) P_{k|k-1}$

The discrete EKF is probably the most used sequential nonlinear estimator nowadays. It was originally developed as a nonlinear extension by Schmidt [12] upon the seminal work of Kalman [3] Based on the Kalman filter, it assumes that the statistical distribution of the state vector remains Gaussian after every time step[2] so it is only necessary to propagate and update the mean and covariance of the state random variable x_k. The main concept is that the estimated state (i.e. estimated mean of x_k) is sufficiently close to the true state (i.e. true mean of x_k) so the nonlinear state/output model equations can be linearized by a truncated first-order Taylor series expansion around the last estimated state.

The discrete algorithm can be seen in table 3. In general, it works for many practical problems but no general convergence or stability conditions can be established[3] and its final performance will depend on the specific case study. For highly nonlinear models with unknown initial conditions the EKF assumptions may prove to be poor and the filter may fail or have a poor performance. The main tunning parameters are the estimator covariance matrices Q_k and R_k.

3.3.2 Unscented Kalman Filters

Table 4. UKF algorithm

Initialization:

$L = n_x + n_w + n_v$, $\lambda = \alpha^2(L + \kappa) - L$

$\gamma = \sqrt[2]{L + \lambda}$, $\Lambda_m^0 = \lambda/(\lambda + L)$

$\Lambda_c^0 = \lambda/(\lambda + L) + (1 - \alpha^2 + \beta)$

for $i = 1, 2, \ldots, 2L$

$\Lambda_m^i = (2(\lambda + L))^{-1}$, $\Lambda_c^i = \Lambda_m^i$

$\hat{x}_{0|0} = E(x_0)$

$P_{0|0} = E[(x_0 - \hat{x}_{0|0})(x_0 - \hat{x}_{0|0})]$

for $k = 1, 2, \ldots$

Propagation step:

(sigma points propagation)

$\tilde{P}_{k-1|k-1} = \text{blkdiag}(P_{k-1|k-1}, Q_k, R_k)$

$\tilde{x}_{k-1|k-1}^0 = [(\hat{x}_{k-1|k-1})^T, 0_{1\times n_W}, 0_{1\times n_v}]^T$

for $i = 1, 2, \ldots, L$

$\tilde{x}_{k-1|k-1}^i = \tilde{x}_{k-1|k-1}^0 + \gamma\,\text{chol}(\tilde{P}_{k-1|k-1}, i)$

$\tilde{x}_{k-1|k-1}^{i+L} = \tilde{x}_{k-1|k-1}^0 - \gamma\,\text{chol}(\tilde{P}_{k-1|k-1}, i + L)$

$\tilde{x}_{k|k-1}^{(x)i} = f_{k-1}(\tilde{x}_{k-1|k-1}^{(x)i}, u_{k-1}, \tilde{x}_{k-1|k-1}^{(w)i})$

$\tilde{y}_{k|k-1}^i = h_k(\tilde{x}_{k|k-1}^{(x)i}, \tilde{x}_{k-1|k-1}^{(v)i})$

(a priori state-output estimate)

$\hat{x}_{k|k-1} = \sum_{i=0}^{2L} \Lambda_m^i \tilde{x}_{k|k-1}^{(x)i}$

$\hat{y}_{k|k-1} = \sum_{i=0}^{2L} \Lambda_m^i \tilde{y}_{k|k-1}^i$

(a priori state covariance estimate)

$\tilde{e}_{x,k|k-1}^i = (\tilde{x}_{k|k-1}^{(x)i} - \hat{x}_{k|k-1})$

$P_{k|k-1} = \sum_{i=0}^{2L} \Lambda_c^i (\tilde{e}_{x,k|k-1}^i)(\tilde{e}_{x,k|k-1}^i)^T$

Measurement update:

(Kalman gain calculation)

$\tilde{e}_{y,k|k-1}^i = (\tilde{y}_{k|k-1}^i - \hat{y}_{k|k-1})$

$P_y = \sum_{i=0}^{2L} \Lambda_c^i (\tilde{e}_{y,k|k-1}^i)(\tilde{e}_{y,k|k-1}^i)^T$

$P_{xy} = \sum_{i=0}^{2L} \Lambda_c^i (\tilde{e}_{x,k|k-1}^i)(\tilde{e}_{y,k|k-1}^i)^T$

$K_k = P_{xy} P_y^{-1}$

(a posteriori state-covariance estimate)

$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - \hat{y}_{k|k-1})$

$P_{k|k} = P_{k|k-1} - K_k P_y K_k^T$

The unscented Kalman filter was originally developed

[2] this assumption is in general not true for nonlinear systems

[3] except for some special cases [13]

by Julier and Uhlman [14–17].
In the unscented Kalman filters instead of approximating the nonlinear state/output functions, it is the probability distribution what is approximated. Basically, a set of points, called sigma points, are generated to match the state mean and state covariance of the probability distribution of the last estimated state, then they are propagated through the nonlinear function. The projected points are used to approximate the first two moments (i.e. the a priori estimated state and state covariance) that are necessary during the measurement update step. This filter normally outperforms the previously presented EKF. Its more general form has a higher computational cost but it does not require the calculation of any Jacobian matrices (i.e. derivatives). The algorithm can be seen in table 4.
The tunning parameters of the UKF are also the estimator process and measurement noise covariance matrices and the scalar parameters $\{\alpha, \kappa, \beta\}$: α determines the spread of the sigma points around the last estimate, and β value depends on the type of distribution assumed (for more details about their values see [17])

Table 5. PF algorithm

Initialization:
(initial particles set)
for $i = 1, 2, \ldots, N$
$x^i_{0\|0} \sim \mathcal{N}(x_0, P_0)$
for $k = 1, 2, \ldots$
Propagation step:
(particles state-output propagation)
for $i = 1, 2, \ldots, N$
$x^i_{k\|k-1} = f_{k-1}(x^i_{k-1\|k-1}, u_{k-1}, w^i_{k-1})$
$y^i_{k\|k-1} = h_k(x^i_{k\|k-1}, v^i_{k-1})$
Measurement update:
(relative likelihood for each particle)
$q_i = p(y_k\|x^i_{k\|k-1})$
$\tilde{q}_i = q_i / \sum_{i=1}^N q_i$
(resampling with replacement step)
$\Pr\left(x^j_{k\|k} = x^i_{k\|k}\right) = \tilde{q}^i$
(a posteriori state-covariance estimates)
$\hat{x}_{k\|k} = (N)^{-1} \sum_{i=1}^N \tilde{q}_i x^i_{k\|k}$
$P_{k\|k} = \sum_{i=1}^N \tilde{q}_i (x^i_{k\|k} - \hat{x}_{k\|k})(x^i_{k\|k} - \hat{x}_{k\|k})^T$

3.3.3 Particle Filters

The particle filter was first proposed by Metropolis and Wiener in the 40's but until the the 80's its implementation was not practical [13]. The particle filter is a statistical, brute-force approach to estimation that works by approximating the recursive Bayesian estimation problem. Its fundamental idea is to represent a pdf by a set of samples called particles.
During the propagation step, the particles that approximates $p(x_{k-1}|y_{k-1})$ from the last iteration are projected through the known dynamic model to obtain an approximation of $p(x_k|y_{k-1})$. Then during the measurement update, each particle likelihood q_i is calculated using the measurement y_k, the known output function and the measurement noise known pdf to approximate the likelihood $p(y_k|x_k)$. If the measurement noise pdf $p(v_k)$ is normal, then the likelihood function for each particle is proportional to

$$\begin{aligned} q_i &= p\left(y_k|x^i_{k|k-1}\right) &(6)\\ e^i_k &= y_k - h_k\left(x^i_{k|k-1}\right) \\ &\propto \frac{1}{2\pi^{m/2}|R|^{1/2}} \exp(-0.5(e^i_k)R^{-1}(e^i_k)^T) &(7)\end{aligned}$$

Then this expression is normalized to assure that the sum of all likelihoods is equal to one

$$\tilde{q}_i = \frac{q_i}{\sum_{j=1}^N p\left(y_k|x^j_{k|k-1}\right)} \qquad (8)$$

Using the resampling with replacement step a new particle set is generated from the old one likelihood weights. The new particle set approximates the $p(x_k|y_k)$. From this pdf the two first central moments are computed [4]. The new particle set is used in the next iteration. For details about the algorithm see table 5.
It is clear that as the number of particles goes to infinity, the quality of the approximations is increased and the approximated pdf approaches the real distribution $p(x_k|y_k)$. The particle filter has several advantages: it can be applied to any nonlinear non-gaussian estimation problem, it approximates the whole pdf and not only a few estimators, it easy to implement and it normally works. Its main disadvantage is clearly its high computational cost. For more details see [18,19].
The EnKF can be considered also a particle filter. It uses an ensemble (i.e. particle set) during the propagation step but the classical Kalman measurement update equations (instead of using the resampling with replacement approach) during the measurement update step. The covariances matrices P_{xy} and P_y obtained from the propagation of the ensemble elements through the nonlinear state-space are used to calculate the Kalman gain K_k. The a posteriori ensemble is calculated from the Kalman gain matrix and an artificially generated measurement particle set that is normally distributed with mean equal to the last measurement y_k and covariance equal to R_k. The a posteriori ensemble is used to calculate the a posteriori state

[4] to be consistent with the other implemented filters.

Table 6. EnKF algorithm

Initialization:
(initial ensemble)
for $i = 1, 2, \dots, N$
$\quad x^i_{0|0} \sim \mathcal{N}(x_0, P_0)$
for $k = 1, 2, \dots$
Propagation step:
(ensemble propagation)
for $i = 1, 2, \dots, N$
$\quad x^i_{k|k-1} = f_{k-1}(x^i_{k-1|k-1}, u_{k-1}, w^i_{k-1})$
$\quad y^i_{k|k-1} = h_k(x^i_{k|k-1}, v^i_{k-1})$
(estimated state-output propagation)
$\hat{x}_{k|k-1} = (N)^{-1} \sum_{i=1}^{N} x^i_{k|k-1}$
$\hat{y}_{k|k-1} = (N)^{-1} \sum_{i=1}^{N} y^i_{k|k-1}$
(covariance calculation)
$e^i_{x,k|k-1} = (x^i_{k|k-1} - \hat{x}_{k|k-1})$
$P_{k|k-1} = (N-1)^{-1} \sum_{i=1}^{N} (e^i_{x,k|k-1})(e^i_{x,k|k-1})^T$
Measurement update:
(Kalman gain calculation)
$e^i_{y,k|k-1} = (y^i_{k|k-1} - \hat{y}_{k|k-1})$
$P_y = (N-1)^{-1} \sum_{i=0}^{N} (e^i_{y,k|k-1})(e^i_{y,k|k-1})^T$
$P_{xy} = (N-1)^{-1} \sum_{i=0}^{N} (e^i_{x,k|k-1})(e^i_{y,k|k-1})^T$
$K_k = P_{xy} {P_y}^{-1}$
(state-out-covariance update)
$x^i_{k|k} = x^i_{k|k-1} + K_k((y_k + v^i_k) - y^i_{k|k-1})$
$\hat{x}_{k|k} = (N)^{-1} \sum_{i=1}^{N} x^i_{k|k}$
$P_{k|k} = P_{k|k-1} - K_k P_y K_k^T$

and covariance estimate, and it is used for the next filter iteration of the algorithm. For details about the algorithm see table 6. The EnKF was originally developed by [20] to overcome the curse of dimensionality in large scale problems (i.e. weather data assimilation). It is suggested in the literature [21] that ensembles (i.e. particle sets) of 50 to 100 are often adequate for systems with thousands of states but no conclusive work has been done on this direction.
Beside the estimator process and measuremnet noise covariance matrices the other tunning parameter for this two filters is the number of particles/ensemble elements.

4. Results

The CHP model is written in Modelica and compiled in Dymola into a stand-alone executable file called Dymosim. The different estimators are implemented in Matlab from where Dymosim is sequentially called during the propagation step to project the state vector in the estimators algorithms. The parameter state vector θ_k is directly propagated within the Matlab code so the original CHP model does not need to be modified to include the parameter dynamic equations.

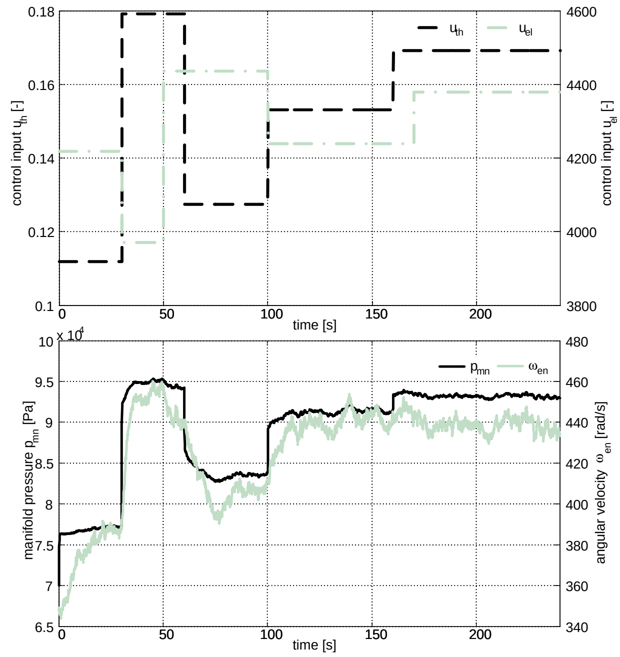

Figure 3. MVEM throttle and electrical load inputs (above) and measured outputs (below).

Because the HRC does not affect the performance of the MVEM model wrt. the mechanical work generated[5] the estimation problem can be divided into two estimation problems for the MVEM and the HRC subsystems. For the two cases it is assumed that the inputs of the system are perfectly known. Due to the lack of experimental measurements, simulated data sets are generated using normally distributed additive white process and measurement noise.

For the MVEM model the parameters to be estimated are $\{V_{\mathrm{mn}}, I_{\mathrm{en}}\}$. The measured outputs are the manifold pressure p_{mn} and the angular velocity ω_{en}. The MVEM is operated in open loop, the inputs are the throttle control input u_{th} and the electrical load control input u_{el} (see Fig. 3). Other model parameters are assumed to be known.

The estimators are simulated for 240 s with a sampling time of 0.1 s. Because the transient response is relevant to identify these parameters, step-like input sequences with high frequency content are used. The same estimator initial state, measurement and process noise covariance matrices are used for all the filters.
The discrete Jacobian matrix is approximated from the jacobian given by Dymosim during its linearization. The UKF parameters are $\{\alpha, \kappa, \beta\} = \{1\,\mathrm{E}-2, 0, 2\}$, the EnKF is simulated for $N = \{30, 50, 100\}$ and the PF for $N = \{100, 200\}$. The estimated parameters

[5] This is true for the simplified model presented.

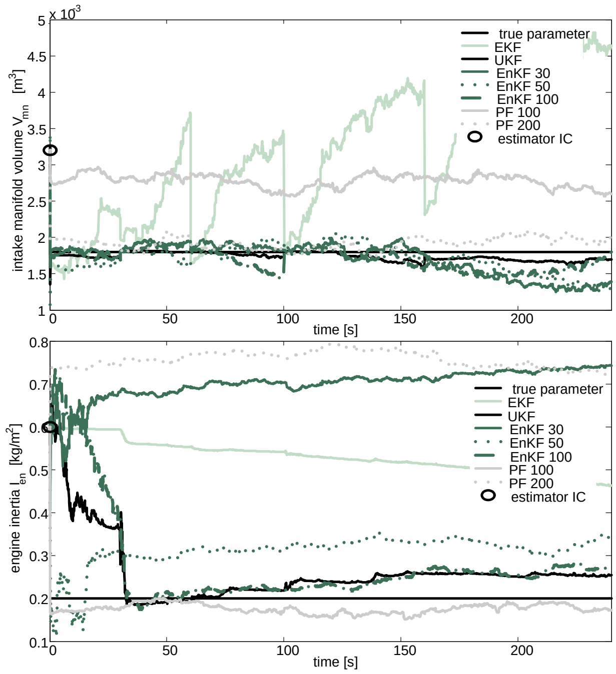

Figure 4. Estimated intake manifold volume V_{mn} (above) and engine inertia I_{en} (below)

$$x_0 = [p_{\mathrm{mn}}, \omega_{\mathrm{en}}, V_{\mathrm{mn}}, I_{\mathrm{en}}]^T = [7\,\mathrm{E}\,4, 350, 1.8\,\mathrm{E}-2, 0.2]^T$$
$$x_{0|0} = [7.9\,\mathrm{E}\,4, 470, 3.2\,\mathrm{E}-2, 0.6]^T$$
$$Q_k = \mathrm{diagblk}\,[\mathrm{diag}([3\,\mathrm{E}\,3, 1\,\mathrm{E}\,2]), \mathrm{diag}([1\,\mathrm{E}-8, 1\,\mathrm{E}-4])]$$
$$R_k = \mathrm{diag}([6\,\mathrm{E}\,3, 2\,\mathrm{E}\,2])$$

for each filter are shown in Fig. 4. The EKF has a poor performance with a very noisy parameter estimation. A zig-zag behavior is seen for the V_{mn} parameter when the inputs change abruptly, this is a clear consequence and limitation of its derivative based formulation.The UKF performs very well staying around the value of the true parameters. The EnKFs as expected, increase their accuracy as the number of elements in the ensemble is increased. The PF has suprisingly a poor performance, an increased number of particles improves the estimation of one parameter but degrades the estimation of the other one. Probably due to the sampling empoverishment the filter stays trapped and an increased number of particles do little to remedy this situation, a roughening algorithm should probably solve this problem.

For the HRC models the parameters to be estimated are $\{\alpha_{\mathrm{cw,co}}, G_1 = \alpha_{\mathrm{1a,1b}} A_{\mathrm{1a,1b}}, G_2 = \alpha_{\mathrm{2a,2b}} A_{\mathrm{2a,2b}}\}$. Other HRC model parameters are assumed known. The measured outputs are $\{\vartheta_{\mathrm{co}}, \vartheta_{\mathrm{2a}}, \vartheta_{\mathrm{2b}}, \vartheta_{\mathrm{1b}}\}$. The HRC is operated in closed loop with the thermal control loop regulating the coolant ouput temperature at 95 °C. The MVEM model is simulated operating

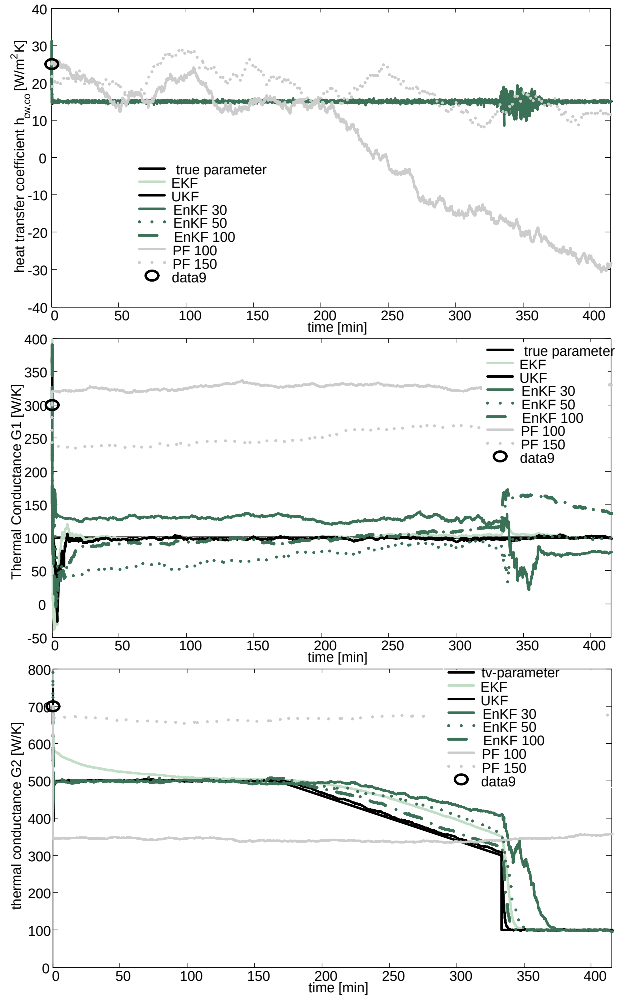

Figure 5. HRC estimated parameters: heat transfer coefficient $\alpha_{\mathrm{cw,co}}$ (above), G_1 (middle), and G_2 (below).

under closed loop wrt. angular velocity with the MVEM parameters are assumed known. The control input is the electrical load u_{el}. Fouling and a sudden change in G_1 is artificially generated to see the tracking capabilities of the filters.
The estimators are simulated for 6.9 h with a sampling time of 60 s. The same estimator initial state, measurement and process noise covariance matrices are used for all the filters.

$$x_0 = [\vartheta_{\mathrm{cw}}, \vartheta_{\mathrm{co}}, \vartheta_{\mathrm{eb}}, \vartheta_{\mathrm{2a}}, \vartheta_{\mathrm{2b}}, \vartheta_{\mathrm{1a}}, \vartheta_{\mathrm{1b}}, \alpha_{\mathrm{cw,co}}, G_1, G_2]^T =$$
$$=[573, 293, 293, 323, 323, 323, 323, 15, 100, 500]^T$$
$$x_{0|0} = [353, 363, 313, 333, 303, 273, 273, 25, 300, 700]^T$$
$$Q_k = \mathrm{dgblk}\,[\mathrm{eye}(7) * 5\,\mathrm{E}-3, \mathrm{eye}(7) * 1\,\mathrm{E}-3]$$
$$R_k = \mathrm{diag}(\mathrm{eye}(4) * 5\,\mathrm{E}-2)$$

The UKF parameters are $\{\alpha, \kappa, \beta\} = \{1\,\mathrm{E}-2, 0, 2\}$, the EnKF is simulated for $N = \{30, 50, 100\}$ and the PF is simulated for $N = \{100, 150\}$ particles. The estimated parameters for each filter are shown in Fig. 5. The EKF has a good performance with a similar response to the UKF for the first two parameters $\{\alpha_{\text{cw,co}}, G_1\}$. For the third parameter G_2 the EKF converges slower than the UKF to the true time-varying parameter. But it outperforms the EnKF with ensembles of size less than 100 elements. The EnKF 100 and the UKF show a comparable performance for all the estimated parameters. The PF again shows a poor performance that can again be explained by the sampling empoverishment.

5. Conclusions

The recursive parameter estimation problem is analyzed for a general CHP model. Some relevant model parameters are estimated using simulated data sets. The best estimator performance is achieved by the UKF for the presented cases. The EKF performs well but some undesirable effects appear as a consequence of the use of the derivatives in its formulation. The EnKF performs similarly to the EKF for moderate size ensembles (i.e. less than 100) reaching a performance similar to the UKF for ensembles larger than 100 elements. The PF performs poorly but this can be probably fix by adding a roughnening algorithm in its formulation [19].

The computational cost of the estimators increases considerably from the EKF to the PF because of the number of projection required for every estimator iteration. Additionally, the CHP model is run from a Dymosim executable file and this slows down significantly the computational performance of the estimators (i.e. the computational time required for every estimator interation). Dymosim uses a slow file input/output interface and consequently exhaustive Montecarlo simulation analysis of the performance of the different estimators are prohibitively slow.

Many aspects concerning the online parameter estimation problem depends on the many characteristics of the real system: the number of uncertain state/parameters, the magnitude of uncertainty, the functional dependence of outputs on the uncertain state/parameters, the quality of output measurements, and the knowledge of system inputs [22]. So our future work will be focus on the identification of the CHP model from real experimental data. Data sets with a limited number of measurements from a CHP unit manufacturer may be available in a short time.

It is also of future interest to run the estimators using a the CHP model implemented in Matlab to speed up the iteration time and to run exhaustive analysis of the estimators.

Table 7. Estimators Notation

symbol	description
(Common notation)	
$k \in \mathbb{N}^+$	time index
$\hat{x}_{k\vert k-1} \in \mathbb{R}^{n_x \times 1}$	a priori x estimate
$\hat{x}_{k\vert k} \in \mathbb{R}^{n_x \times 1}$	a post. x estimate
$\hat{y}_{k\vert k-1} \in \mathbb{R}^{n_y \times 1}$	a priori y estimate
$\hat{y}_{k\vert k} \in \mathbb{R}^{n_y \times 1}$	a post. y estimate
$P_{k\vert k-1} \in \mathbb{R}^{n_x \times n_x}$	a priori P estimate
$P_{k\vert k} \in \mathbb{R}^{n_x \times n_x}$	a post. P estimate
$K_k \in \mathbb{R}^{n_x \times n}$	kalman gain
$\hat{x}_{0\vert 0} \in \mathbb{R}^{n_x \times 1}$	initial x estimate
$P_{0\vert 0} \in \mathbb{R}^{n_x \times n_x}$	initial P estimate
(extended Kalman filters)	
$F_{k-1} \in \mathbb{R}^{n_x \times n_x}$	jacob. f_{k-1} wrt. x_{k-1}
$L_{k-1} \in \mathbb{R}^{n_x \times n_w}$	jacob. f_{k-1} wrt. w_{k-1}
$H_k \in \mathbb{R}^{n_y \times n_x}$	jacob. h_k wrt. x_k
$M_k \in \mathbb{R}^{n_y \times n_v}$	jacob. h_k wrt. v_k
(unscented Kalman filters)	
$L \in \mathbb{N}^+$	num. aug. states
$\alpha, \beta, \kappa, \gamma, \lambda \in \mathbb{R}$	filter parameters
$\Lambda_m^i \in \mathbb{R}$	ith σ-pt. mean weight
$\Lambda_c^i \in \mathbb{R}$	ith σ-pt cov. weight
$\tilde{x}_{k\vert k}^i \in \mathbb{R}^{Lx1}$	ith σ-pt
$\tilde{x}_{k\vert k}^{(x)i} \in \mathbb{R}^{n_x \times 1}$	ith σ-pt x elements
$\tilde{x}_{k\vert k}^{(w)i} \in \mathbb{R}^{n_w \times 1}$	ith σ-pt w elements
$\tilde{x}_{k\vert k}^{(v)i} \in \mathbb{R}^{n_v \times 1}$	ith σ-pt v elements
$\tilde{P}_{k\vert k} \in \mathbb{R}^{LxL}$	a post. aug. P estimate
$P_y \in \mathbb{R}^{n_y \times n_y}$	cov. y vector
$P_{xy} \in \mathbb{R}^{n_x \times n_y}$	crosscov. xy vectors
(particle filters and ensemble KF)	
$N \in \mathbb{N}$	num. particles
$x_{k\vert k-1}^i$	ith a prior. particle
q^i	ith particle likelihood
$\tilde{q}^i$	normalized q^i

References

[1] H. I. Onovwiona and V. I. Ugursal, "Residential cogeneration systems: review of the current technology," *Renewable and Sustainable Energy Reviews*, pp. 1–43, 2004.

[2] G. E. Hovland, T. P. von Hoff, E. A. Gallestey, M. Antoine, D. Farruggio, and A. D. B. Paice, "Nonlinear estimation methods for parameter tracking in power plants," *Control Engineering Practice*, vol. 13, pp. 1341–1355, 2005.

[3] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

Table 8. CHP model notation

$u_{x=\{\text{th,by,p1,p2}\}}$	x control signal $[-]$
$p_{x=\{\text{ab,mn,ex}\}}$	x pressure $[\mathrm{N/m^2}]$
$\vartheta_{x=\{\text{ab,mn,ex}\}}$	x temperature $[\mathrm{K}]$
$\dot{m}_{x=\{\text{th,en},\beta,\varphi,\text{by,co,pr,sc}\}}$	x mass flow rate $[\mathrm{kg/s}]$
$p_{me[x]=\{\text{pg},b,\varphi,f\}}$	x mean effec.press.$[\mathrm{N/m}]$
$T_{x=\{\text{en,ld}\}}$	x torque $[\mathrm{N\,m}]$
$\dot{H}_{x=\{\text{en,ex}\}}$	x enthalpy flow rate $[\mathrm{W}]$
$\dot{Q}_{[x],[y]}$ $x=\{\text{cg,cw,co,eb,1a,2a}\}$ $y=\{\text{cw,co,eb,2a,2b,ab}\}$	heat flow rate from x to y $[\mathrm{W}]$
$\dot{W}_{\text{en}}$	work rate $[\mathrm{W}]$
P_{en}	electrical power $[\mathrm{W}]$
ω_{en}	angular velocity $[\mathrm{rad/s}]$
$V_{x=\{\text{mn},d,c\}}$	x volume in $[\mathrm{m^3}]$
$\eta_{x=\{\text{td,ec,vl}\}}$	x efficiency $[-]$
$c_{x=\{\text{cw,co,eb,oi,1a,1b,2a,2b}\}}$	x spc.heat cap.$[\mathrm{J/kg\,K}]$
$A_{x=\{\text{cw,co,eb,oi,1a,1b,2a,2b}\}}$	x area $[\mathrm{m^2}]$
$m_{x=\{\text{cw,co,eb,oi,1a,1b,2a,2b}\}}$	x mass $[\mathrm{kg}]$
$\alpha_{x=\{\text{cw,co,eb,oi,1a,1b,2a,2b}\}}$	x heat-tf.coef.$[\mathrm{W/m^2\,K}]$
$\beta_i, \delta_i, \gamma_i$	$i$ths $p_{\text{mef}}, \dot{Q}_{\text{cg,cw}}, \eta_{\text{vl}}$ coef.
I_{en}	flywheel inertia $[\mathrm{kg/m^2}]$
$A_{\text{th}}, \alpha_{\text{th}}, d_{\text{th}}, C_{\text{th}}, \Delta\vartheta_{\text{mn}}$	throttle valve parameters
$R_\beta, \kappa_\beta, \lambda, \sigma, H_l$	air-fuel parameters

subscript notation:

1a	hx1 side a	ec	electrical	td	thermdyn.
1b	hx1 side b	en	engine	th	throttle
2a	hx2 side a	eo	eng. oil	vl	volumet.
2b	hx2 side b	ex	exhaust	β	air
ab	ambient	mn	manifold	φ	fuel
by	bypass	ld	load	b	brake
cg	cyl. gas	p1	pump 1	c	compress
ci	coolant in	p2	pump 2	d	displmnt.
co	coolant out	pg	gass loss	f	friction
cw	cyl. wall	pr	primary	l	lower
eb	eng. block	sc	second.		

[4] A. Gelb, J. F. Kasper, R. A. Nash, C. F. Price, and A. A. Sutherland, *Applied Optimal Estimation.* The M.I.T. Press, 1974.

[5] J. I. Videla and B. Lie, "Simulation of a small scale si ice based cogeneration system in modelica/dymola," in *SIMS conference*, Helsinki, Finland, September 2006.

[6] ——, "Simulation of a bio-ethanol ice chp system," in *3rd International Green Energy Conference*, Sweden, 2007.

[7] D. Cho and J. K. Hedrick, "Automotive powertrain modeling for control," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 111, no. 4, pp. pp.586–576, 1989.

[8] E. Hendricks and S. C. Sorenson, "Mean value modeling of spark ignition engines," *SAE Technical Paper Series*, no. 900616, 1990.

[9] E. Hendricks and T. Vesterholm, "Analysis of mean value si engine models," *SAE Technical Paper Series*, no. 920682, pp. 1–19, 1992.

[10] L. Guzzella and C. H. Onder, *Introduction to Modeling and Control of Internal Combustion Engine Systems.* Springer, 2004.

[11] J. L. Crassidis and J. L. Junkins, *Optimal estimation of dynamic systems*, ser. CRC applied mathematics and nonlinear science series. Chapman & Hall, 2000.

[12] S. F. Schmidt, *Application of State-Space Methods to Navigation Problems*, c.t. leondes ed. Academic Press, New York, San Francisco, London, 1966, vol. 3, pp. 293–340.

[13] D. Simon, *Optimal State Estimation – Kalman, H_∞, and Nonlinear Approaches.* Hoboken, New Jersey: John Wiley & Sons, Inc., 2006.

[14] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proceedings of the 1995 American Control Conference*, Seattle, WA, 1995, pp. 1628–1632.

[15] S. Julier and J. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," tech. rep., RRG, Dept. of Engineering Science, University of Oxford, Nov 1996, Tech. Rep., 1996.

[16] ——, "A new extension of the Kalman filter to nonlinear systems," in *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls, Orlando, FL*, 1997.

[17] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation (invited paper)," in *Proceedings of the IEEE*, vol. 92(3). IEEE Institute of Electrical and Electronics, 2004, pp. 401–422.

[18] T. B. Schön, "Estimation of nonlinear dynamic systems, theory and applications," Ph.D. dissertation, Linköpings Studies in Science and Technology, Sweden, 2006.

[19] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter– Particle Filters and Tracking Applications.* Artech House, 2004.

[20] G. Evensen, "The ensemble kalman filter: Theoretical formulation and practical implementation," *Ocean Dynamics*, vol. 53, pp. 343–367, 2003.

[21] S. Gillijns, O. B. Mendoza, J. Chandrasekar, B. L. R. D. Moor, D. S. Bernstein, and A. Ridley, "What is the ensemble kalman filter and how well does it work?" in *Proceedings of the 2006 American Control Conference*, Minneapolis, Minnesota, USA, June 2006.

[22] R. F. Stengel, *Optimal Control And Estimation.* John Wiley & Sons, Inc., 1986.