

# Self-Adaptive Air-path Health Management for a Heavy Duty-Diesel Engine

Tomáš Polóni<sup>1</sup>, Paul Dickinson<sup>2</sup>, Jianrui Zhang<sup>3</sup>, Peng Zhou<sup>4</sup>

<sup>1,2</sup> *Garrett Motion - Connected Vehicle, V Parku 2326/18, Prague, 148 00 Czech Republic*

*tomas.poloni@garrettmotion.com*

*paul.dickinson@garrettmotion.com*

<sup>3,4</sup> *FAW JIEFANG AUTOMOTIVE - Engine Electronic Control and Calibration, Changchun, Jilin, 130011, China*

*zhangjianrui@rdc.faw.com.cn*

*zhoupeng@rdc.faw.com.cn*

## ABSTRACT

This paper presents an air-path health management strategy with the ability to estimate the mass flows and mitigate (adapt to) the air-path faults in a heavy-duty diesel combustion engine, equipped with a twin-scroll turbine. Based on the engine component models applied in the quasi-steady-state mass-balancing approach, two main engine mass-flow quantities are estimated: the Air mass flow (AMF) and the Exhaust gas recirculation (EGR) mass flow. The health management system is monitoring for three kinds of air-path faults that can occur through the engine operation. These are related to either the after-treatment system, EGR valve, or to the turbine balance valve hardware. For each fault, a fault-mitigation strategy is proposed. This is based on in-observer-reconfigurable mass-balance equations with an excluded faulty component model utilizing an exhaust pressure sensor. The observer uses the iterated Kalman filter (IKF) as the core fault mitigating solver for the quasi-steady-state mass-balancing problem. It is further demonstrated how the individual faults are robustly isolated using the Sequential Probability Ratio Test (SPRT). The strategy and results are validated using the test cycle driving data.

## 1. INTRODUCTION

Sustainable mobility requires a diverse portfolio of propulsion systems to ensure the right technologies for the right applications, which besides other emerging technologies (fuel cells, battery-electric, and hybrid-driven) spans over the internal combustion (IC) engine innovations (Reitz et al., 2020). IC engines have developed into complex systems with an increasing number of components, design alternatives, and im-

proved control, monitoring and troubleshooting functionalities (Isermann, 2017). An engine operation with fault diagnosis, including fault detection and isolation (FDI) algorithms (Chen & Patton, 1999), has become an important part of the engine control software. The engine mass-flow observer technology enhances accurate emission controls and power demands (Dahl et al., 2018; Wassén et al., 2019), typically accounting for the sensor faults and model uncertainties (Polóni et al., 2014; Gutiérrez León et al., 2018). The fault isolation methods of air and fuel paths, due to the aging of engine components, have been studied in Nyberg & Stutte (2004) and Schilling et al. (2008). Nonlinear controllers attenuating air-path faults are proposed by Ahmed Ali et al. (2015) and Zhang et al. (2021), where the information redundancy approach for fault-tolerant air-fuel ratio controller is proposed by Amin & ul Hasan (2019). The above mass-flow estimation-related publications are focusing either on mass-flow estimation under nominal system conditions or fault isolation methods. The mass-flow observer publications focusing on mitigation of the air-path faults are practically missing in the literature.

The main motivation for this paper is to demonstrate a novel systematic method that isolates and at the same time mitigates and accounts for aftertreatment, EGR, or turbine balance valve faults inside the mass-flow observer, in a self-adaptive and self-restructuring fashion. The observer conception is based on a new quasi-steady-state formulation that in combination with a stochastic solver, the iterated Kalman filter (IKF), brings sufficient robustness and performance for the mass-flow estimation, even for the faulty air-path system. The performance metric and performance evaluation of the proposed method is given.

Tomáš Polóni et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

<https://doi.org/10.36001/IJPHM.2023.v14i3.3118>

## 2. MODEL SCOPE AND MODELLING APPROACH

The heavy-duty diesel engine with a twin-scroll turbine used in this study is schematically shown in Figure 1 where the rounded edge rectangle denotes its model scope. The production sensors required for the observer are shown in green: ambient pressure,  $p_0$ , ambient temperature  $T_0$ , intake manifold pressure,  $p_2$ , intake manifold temperature  $T_2$ , engine speed  $N_e$ , fuel mass flow  $m_F$ , exhaust lambda,  $\lambda$ , and the EGR and balance valve positions,  $u_{EGR}$ , and  $u_{TRB}$ , respectively. The charge mass flow,  $m_{Ch}$  comprises the fresh air flow,  $m_A$ , and EGR flow,  $m_{EGR}$ . The flow leaving the engine is assumed to be equal to both exhaust branches; EGR is routed from the high pressure branch which feeds the smaller turbine scroll, with a mass flow  $m_{T1}$ . The second exhaust branch directly feeds scroll 2, with a mass flow  $m_{T2}$ , the two exhaust branches are connected with a balancing valve which controls the mass flow between the two exhaust branches,  $m_{BAL}$ . The post turbine flow is denoted  $m_{AFT}$ . The variables shown in red represent the unmeasured states: exhaust temperatures in branches 1 and 2 are  $T_{3a}$  and  $T_{3b}$  respectively, the pressure in the scroll 2 inlet volume is  $p_{3b}$ , the post turbine pressure and temperature are  $p_4$  and  $T_4$  respectively. There are 4 control volumes used to compute the mass balances: the intake manifold,  $i$ , the exhaust branch which feeds the EGR path, the balance valve and scroll 1,  $k_1$ , the second exhaust branch which feeds the larger turbine scroll 2,  $k_2$  and the post-turbine volume,  $k_3$ , which is the entry to the aftertreatment system. The internal pressures and temperatures have to be estimated using the engine component models with a later introduced iterative algorithm solving the mass-balancing problem. Following are listed the engine component models that are involved in the model scope: Charge flow model, EGR valve flow model, Turbine scroll 1 flow model, Turbine scroll 2 flow model, Turbine balancing flow model, Aftertreatment flow model, Exhaust manifold temperature model, Turbine outlet temperature model, Burned fraction model, and Phi model.

The models used by observers and estimators are commonly physics-based using the mean value engine model principles that may become difficult to simulate due to model stiffness (Dahl et al., 2018). In the following modeling approach, the nonlinear functions in the air-path model are defined as multivariate rational functions, see Dahl et al. (2018) and references therein. Many physics-based component models are native to this form (ideal gas equations, stoichiometric equations), whereas others can be approximated by rational functions to a user-defined degree of accuracy. A general form of a system flow component in this framework can be represented as

$$m_* = \frac{m_*^N(x_1, x_2, \dots, x_n, u_1, \dots, u_m)}{m_*^D(x_1, x_2, \dots, x_n, u_1, \dots, u_m)} \quad (1)$$

where  $m_*^N$  is the numerator,  $m_*^D$  is the denominator with state  $x$  and input  $u$  variables (\* stands for a particular component

subscript); functions  $m_*^N$  and  $m_*^D$  are multivariate polynomials. The summary of mass-flow models and their multivariate polynomial approximations can be found in Appendix A.

## 3. MODE-BASED ADAPTIVE OBSERVER

The architecture of mode-based adaptive observer for the air mass-flow, EGR mass-flow, and burn fractions, also known in its 1st generation as the VEGRO - Virtual EGR Observer (Dahl et al., 2018) is updated with the informational as well as the structural redundancy (Baramov et al., 2021). The informational redundancy in the following observer is provided by an exhaust pressure sensor, that is under the nominal conditions (no faults present at the engine), not used for estimation purposes but is used for the monitoring of estimation accuracy. The structural redundancy is in the observer used to switch the observer equations based on the system mode defined either by the nominal no-fault mode or by the three different faults expected to occur on the air path of the engine, resulting in four possible operational modes.

The mode of operation is determined by the Monitor which is an algorithm responsible for the fault isolation where to each fault one mode is assigned. Whenever the Monitor reports a change of operation from the nominal mode to one of the faulty modes, the observer is reconfigured/switched. The nominal mode observer is shown in Figure 2 The nominal model observer consists of five main computational blocks: (1) Balance flow equations solver, (2) Charge flow model, (3) Air-flow model, (4)  $\lambda$ -based air-flow filter, and (5) Air-flow fusion block. The ECU data are fed to the iterative Balance flow equation solver solving simultaneously for the unknown pressures  $p_{3a}$ ,  $p_{3b}$ , and  $p_4$ . These pressures are used for computing the air-flow estimate based on either the EGR valve model (and the charge flow model), turbine or aftertreatment flow models. This flow is denoted as  $m_A^{T,EGR,Aft}$ . The charge flow is computed using the well-known speed density equation. It is based on determining the gas density at the intake manifold using the intake manifold pressure and temperature and using an empirical polynomial model of volumetric efficiency. This empirical model of volumetric efficiency is based on engine speed, intake manifold pressure, and injected fuel mass. The alternative air mass-flow estimate is computed based on the injected fuel quantity and lambda measured by the exhaust sensor. Instead of using  $\lambda$  which is unbounded for a lean-burn engine, its inverse value is used, also known as the equivalence ratio  $\phi = 1/\lambda$ . For the lean operation of the Diesel engine, this limits the value between 0 and 1. The air flow based on the  $\phi$  information is given by

$$m_A^\lambda = \frac{AFR_{stoc} m_F}{(1 - S_h) \phi_F} \quad (2)$$

where  $S_h$  denotes the specific humidity (vapor mass fraction in the air) and  $\phi_F$  is the filtered equivalence ratio. The specific humidity is not measured in the vehicle and hence its value is



of exhaust branch 1 as

$$\begin{aligned} \bar{R}_1(m_{T1}^N, m_{T1}^D, m_{T2}^N, m_{T2}^D, m_{AFT}^N, m_{AFT}^D) = \\ = m_{T1}^N m_{T2}^D m_{AFT}^D + m_{T1}^D m_{T2}^N m_{AFT}^D - m_{T1}^D m_{T2}^D m_{AFT}^N = 0 \end{aligned} \quad (7)$$

$$\begin{aligned} \bar{R}_2(m_{T2}^N, m_{T2}^D, m_{BAL}^N, m_{BAL}^D, m_{Ex}^N, m_{Ex}^D) = \\ = m_{T2}^N m_{BAL}^D m_{Ex}^D + m_{T2}^D m_{BAL}^N m_{Ex}^D - m_{T2}^D m_{BAL}^D m_{Ex}^N = 0 \end{aligned} \quad (8)$$

$$\begin{aligned} \bar{R}_3(m_{T1}^N, m_{T1}^D, m_{BAL}^N, m_{BAL}^D, m_{EGR}^N, m_{EGR}^D, m_{Ex}^N, m_{Ex}^D) = \\ = m_{T1}^N m_{BAL}^D m_{EGR}^D m_{Ex}^D + m_{T1}^D m_{BAL}^N m_{EGR}^D m_{Ex}^D + \\ + m_{T1}^D m_{BAL}^D m_{EGR}^N m_{Ex}^D - m_{T1}^D m_{BAL}^D m_{EGR}^D m_{Ex}^N = 0 \end{aligned} \quad (9)$$

$$\bar{R}_4 = p_{3a} - p_{3a,snr} = 0 \quad (10)$$

where for the sake of shorter notation, the following expression is used

$$m_{Ex}(p_2, m_F, N_E, T_2) = \frac{1}{2}(m_{CH}(p_2, m_F, T_2, N_e) + m_F) \quad (11)$$

Following vector variables are defined for the next explanations and discussions,  $x = [p_{3a}, p_{3b}, p_4]^T$ ,  $\bar{R} = [\bar{R}_1, \bar{R}_2, \bar{R}_3, \bar{R}_4]^T$ , and  $S(x) = [m_{T1}^N, m_{T1}^D, \dots, m_{Ex}^N, m_{Ex}^D]^T$ .

### 3.2. Balance flow equations solver - The Iterated Kalman Filter

Solving the residual equations  $\bar{R}$  for the unknown vector  $x$  is the core task of the applied solver, the iterated Kalman filter (IKF) (Simon, 2006, p. 410). The discrete-time models embedded in the IKF are in a general form written as

$$x_k = f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}), \quad (12)$$

$$y_k = h_k(x_k, u_k, v_k), \quad (13)$$

where  $f_k(\cdot)$  represents the model of nonlinear dynamics being a function of the state vector  $x$ , input vector  $u$ , and process noise  $w$ . The function  $h_k(\cdot)$  represents a nonlinear reference measurement model with the state  $x$ , input  $u$ , and measurement noise  $v$  variables. In the following section, the observer-specific system equations and measurement equations are given.

The system equations  $f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1})$  consist of the unknown pressures, each defined as a constant between two sampling intervals augmented with the process noise term  $w$

$$x_{1,k} = x_{1,k-1} + w_{1,k-1} \quad (14)$$

$$x_{2,k} = x_{2,k-1} + w_{2,k-1} \quad (15)$$

$$x_{3,k} = x_{3,k-1} + w_{3,k-1} \quad (16)$$

where  $x_{1,k} = p_{3a,k}$ ,  $x_{2,k} = p_{3b,k}$  and  $x_{3,k} = p_{4,k}$ . The measurement equations  $h_k(x_k, v_k)$  consist of the residual equations  $\bar{R}$ , where each equation is augmented with the measurement

equation noise term  $v$

$$y_{1,k} = \bar{R}_{1,k}(m_{T1}^N, m_{T1}^D, m_{T2}^N, m_{T2}^D, m_{AFT}^N, m_{AFT}^D) + v_{1,k} \quad (17)$$

$$y_{2,k} = \bar{R}_{2,k}(m_{T2}^N, m_{T2}^D, m_{BAL}^N, m_{BAL}^D, m_{Ex}^N, m_{Ex}^D) + v_{2,k} \quad (18)$$

$$y_{3,k} = \bar{R}_{3,k}(m_{T1}^N, m_{T1}^D, m_{BAL}^N, m_{BAL}^D, m_{EGR}^N, m_{EGR}^D, m_{Ex}^N, m_{Ex}^D) + v_{3,k} \quad (19)$$

$$y_{4,k} = \bar{R}_{4,k}(p_{3a}, p_{3a,snr}) + v_{4,k} \quad (20)$$

The actual measurements of  $y_k$  represent known values of the balance flow errors (the residuals) that are forced to be zeros.

The system sensitivity for the estimated vector is given as

$$F_{k-1} = \left. \frac{\partial f_{k-1}}{\partial x} \right|_{\hat{x}_{k-1}^+} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (21)$$

The system sensitivity concerning the process noise vector is given as

$$L_{k-1} = \left. \frac{\partial f_{k-1}}{\partial w} \right|_{\hat{x}_{k-1}^+} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (22)$$

The measurement sensitivity for the measurement noise vector is given as

$$M_{k,i} = \left. \frac{\partial h}{\partial v} \right|_{\hat{x}_{k,i}^+} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (23)$$

The measurement sensitivity is given by the chain rule due to the temperature  $T_4$  model dependency on the state vector

$$H_{k,i} = \left. \frac{\partial h}{\partial x} \right|_{\hat{x}_{k,i}^+} = \left[ \frac{\partial \bar{R}}{\partial S} \frac{\partial S}{\partial x} + \frac{\partial \bar{R}}{\partial S} \frac{\partial S}{\partial T_4} \frac{\partial T_4}{\partial x} \right]_{\hat{x}_{k,i}^+} \quad (24)$$

**Note on applied assumptions:** The assumptions applied are the designer's choice to simplify a certain system model behavior. Whether the assumptions are sufficiently valid is ultimately verified by the estimation performance - the assumptions are justified by the results. The choice to neglect the manifold filling dynamics is not motivated by the fact of whether the intake (exhaust) volumes are big or small but by the impact on estimation accuracy of computed mass flows if we do so. That also means that the pressure transients are not neglected (the other measured inputs are transient) and are given as a solution to the quasi-steady-state problem given by Eqs. 3 to 6 - the neglected volume size affects the overall inaccuracy/error of each estimated pressure during transients. These equations must hold even during transient operation if the assumption about the steady state nature is combined with another assumption that steady-state flow residual equations do not need to be equal to zero exactly, hence the residuals formulation via Eqs. 17 to 20. It is one of the assumptions

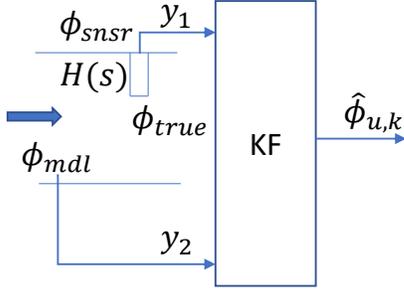


Figure 3. Equivalence ratio filtering scheme.

that the residual error is bounded and can be represented by the measurement noise  $v$  standard deviation parameter  $\sigma_v$ . It is also assumed that the inaccuracy of the pressures during transients is (small and) bounded and within the magnitude of defined process noise  $w$  standard deviation parameter  $\sigma_w$ .

### 3.3. Phi filter

The role of this filter is to compensate for the time lag of the lambda sensor that provides the measured engine out lambda information and fuses it with the virtual sensor. The signal that the lambda sensor provides is not the true lambda - it is time delayed and filtered by the sensor dynamics. To get the estimate of "true lambda", we use the steady-state Kalman filter for unknown input estimation with the reference measurements of the virtual (model-based) lambda and the sensor measured lambda. The filter is a dynamic fusion algorithm when at the beginning the model-based lambda is more trusted, but towards the steady state, the influence of the sensor signal grows - at the steady state the sensor and filter output co-inside. Instead of lambda, its inverse value called the equivalence ratio is used for numerical reasons, where for the normal operating range of the Diesel engine, this value is bounded in the (0, 1) interval. In Figure 3, we use the measured equivalence ratio  $\phi_{snsr}$  and model estimated  $\phi_{mdl}$  to estimate the true equivalence ratio  $\phi_{true}$ . The estimate of the  $\phi_{true}$  represents the unknown input  $\phi_u$  into the sensor dynamic model. The virtual sensor  $\phi_{mdl}$  is providing the information about the equivalence ratio where its measurement model consists of the summation of the estimate of  $\phi_{true}$  and unknown bias  $b_{mdl}$ . The model of continuous lambda sensor dynamics is  $H(s) = 1/\tau s + 1$  where its zero-order hold (ZOH) discrete model is given as

$$\phi_y(k+1) = (1-a)\phi_u(k) + a\phi_y(k) \quad (25)$$

In this equation, the parameter  $a = e^{-Ts/\tau}$ , where  $Ts$  is the sampling interval and  $\tau$  is the sensor's time constant. From the filtering point of view, the following system and measurement variables are defined:  $x_1 = \phi_u$  sensor model input (state variable),  $x_2 = \phi_y$  sensor model output (state variable),  $x_3 = b_{mdl}$  virtual sensor bias (state/parameter variable),  $y_1 = \phi_{snsr}$

measurement variable,  $y_2 = \phi_{mdl}$  measurement variable. The state space model is defined as

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1-a & a & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} w_1(k) \\ w_2(k) \\ w_3(k) \end{bmatrix} \quad (26)$$

$$x(k+1) = F_\phi x(k) + w(k) \quad (27)$$

The measurement model is given as

$$\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} v_1(k) \\ v_2(k) \end{bmatrix} \quad (28)$$

$$y(k) = H_\phi x(k) + v(k) \quad (29)$$

The state equations are augmented with process noise vector  $w \sim (0, Q_\phi)$  and the measurement equations are augmented with the measurement noise vector  $v \sim (0, R_\phi)$ , where  $Q_\phi$  is the process noise error covariance matrix and  $R_\phi$  is the measurement noise error covariance matrix. The noises  $w_1, w_2, w_3, v_1, v_2$  are zero-mean, white Gaussian, and their standard deviations represent Kalman's filter tuning parameters. The noise matrices are given as

$$Q_\phi = \begin{bmatrix} \sigma_{w1}^2 & 0 & 0 \\ 0 & \sigma_{w2}^2 & 0 \\ 0 & 0 & \sigma_{w3}^2 \end{bmatrix} \quad (30)$$

$$R_\phi = \begin{bmatrix} \sigma_{v1}^2 & 0 \\ 0 & \sigma_{v2}^2 \end{bmatrix} \quad (31)$$

where  $\sigma_w$  is the standard deviation of process noise error and  $\sigma_v$  is the standard deviation of measurement noise error.

The steady-state Kalman filter solution, (Simon, 2006, p. 193-194), is defined as

$$\hat{x}_k^+ = A_F \hat{x}_{k-1}^+ + B_F y_k \quad (32)$$

where  $A_F = (I - K_\infty H_\phi) F_\phi$ ,  $I$  is the identity matrix, and  $B_F = K_\infty$  is the steady-state Kalman gain<sup>2</sup>. The estimate of the true equivalence ration  $\hat{\phi}_u$  is represented by the first element of  $\hat{x}_k^+$  vector as

$$\hat{\phi}_{u,k} = C_e \hat{x}_k^+ \quad (33)$$

where  $C_e = [1, 0, 0]$  is the state observation matrix. The following parameter values are set to compute the steady-state Kalman filter:  $T_s = 0.02(s)$  - sampling interval;  $\tau = 0.8(s)$  - the time constant of lambda sensor;  $\sigma_{w1} = 500$ ;  $\sigma_{w2} = 0.01$ ;  $\sigma_{w3} = 0.005$ ;  $\sigma_{v1} = 0.25$ ;  $\sigma_{v2} = 0.1$ . The influence of tuning can be interpreted as weights posed on the measurement and model that are regime dependent. The filter is tuned such that during the transient regime the measurement weight is less dominant while the model weight is more dominant.

<sup>2</sup>The steady-state Kalman gain is given by solving the Discrete Algebraic Riccati Equation (DARE) with MATLAB command `[~,~,K] = dare(F', H', Q, R)`, where  $F = F_\phi$ ,  $H = H_\phi$ ,  $Q = Q_\phi$ ,  $R = R_\phi$ , and  $K_\infty = K^*$ .

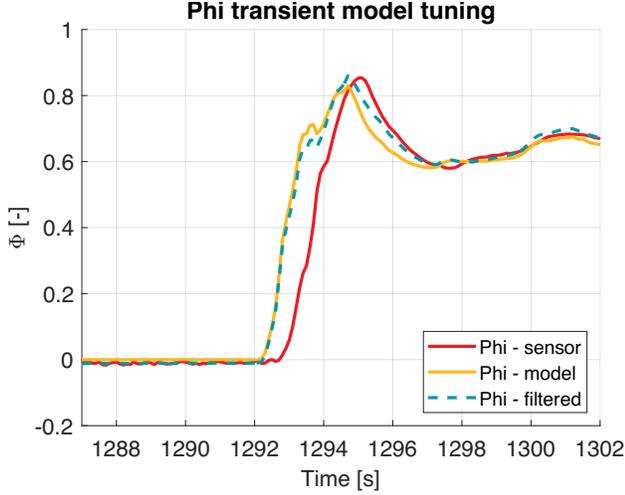


Figure 4. Phi-filter response.

The dominance of measurement and model weights inverses as the steady-state regime is present. The filter response is shown in Figure 4.

### 3.4. Model fusion

The model fusion block performs the least-squares best compromise between two sources of air mass-flow estimates:

- lambda sensor-based estimate (delay compensated in the phi filter)
- turbine model, EGR valve or aftertreatment model-based estimate  $m_A^{T,EGR,Aft}$  from the solver.

The fusion block automatically discounts the source which is less reliable at the moment. This is done using the weighted least squares formulated as follows

$$\min_{m_A} \left\| \begin{matrix} w_1(m_A^\lambda - m_A) \\ w_2(m_A^{T,EGR,Aft} - m_A) \end{matrix} \right\|_2^2 \quad (34)$$

The baseline weight  $w_1$  is determined by a lookup table based on the magnitude of the exhaust O2 concentration. When the concentration is greater than 10%, on a volume basis, the weight is set to 1; the weight linearly increases to 2 when the concentration is 0%. This relationship can be calibrated but was selected based on the physical working principle of the sensor –the sensor has the highest accuracy when measuring stoichiometric compositions. Furthermore, if the sensor status is troublesome, this weight is switched to 0. The second weight is  $w_2 = w_{20}m_1m_2$ . The baseline weight  $w_{20}$  is set to 0.5. The multiplier  $m_1$  equals 0.85 if the balancing valve position exceeds a threshold of 10%, otherwise equals 1. The second multiplier  $m_2$  equals to 0.25 if  $p_{3a} - p_{3b}$  is less than threshold 5 kPa and is 1 otherwise. The weights are used to represent the confidence in the fusion output for various scenarios.

Mode	$\sigma_{v1}$	$\sigma_{v2}$	$\sigma_{v3}$	$\sigma_{v4}$
1/ No fault	$10^{-2}$	$10^{-2}$	$10^{-2}$	$10^2$
2/ Fault A	$6 * 10^2$	$10^{-3}$	$10^{-3}$	$10^{-2}$
3/ Fault B	$10^{-2}$	$10^{-2}$	$6 * 10^2$	$10^{-3}$
4/ Fault C	$10^{-2}$	$10^{-2}$	$10^{-2}$	$10^2$

Table 1. Fault mitigation tuning -  $R$  matrix; Fault A - Aftertreatment fault, Fault B - EGR valve fault, Fault C - Balance valve fault

## 4. MONITOR

The main task of the monitor is to evaluate if the prime state-observer algorithm is performing with models reflecting the nominal health status of hardware components. In a case of model mismatch, biased estimates of air mass-flow, EGR mass-flow (and burned fractions) may result, leading to inaccurately controlled air path. The model mismatch can be induced by sudden faults (e.g. stuck valves) or by slowly degrading hardware components.

### 4.1. Fault mitigation strategy

The Fault mitigation strategy (FMS) is based on the adaptation of VEGRO solver settings, specifically, it is the mode dependent update of the measurement error covariance matrix  $R$  (see the solver algorithm details in Appendix B) (Baramov et al., 2021). Through the  $R$  matrix we define which equation of residuals is irrelevant for the mass-flow estimation due to hardware component/model mismatch.

The mode dependant tuning is realized through the IKF tuning matrices  $Q$  and  $R$  that have the following structure

$$Q = \begin{bmatrix} \sigma_{w1}^2 & 0 & 0 \\ 0 & \sigma_{w2}^2 & 0 \\ 0 & 0 & \sigma_{w3}^2 \end{bmatrix} \quad (35)$$

$$R = \begin{bmatrix} \sigma_{v1}^2 & 0 & 0 & 0 \\ 0 & \sigma_{v2}^2 & 0 & 0 \\ 0 & 0 & \sigma_{v3}^2 & 0 \\ 0 & 0 & 0 & \sigma_{v4}^2 \end{bmatrix} \quad (36)$$

where  $\sigma_{w1}$  (bar) is the standard deviation of  $p_{3a}$  process noise,  $\sigma_{w2}$  (bar) is the standard deviation of  $p_{3b}$  process noise,  $\sigma_{w3}$  (bar) is the standard deviation of  $p_4$  process noise,  $\sigma_{v1}$  (kg/s) is the standard deviation of first residual equation  $y_1$  error,  $\sigma_{v2}$  (kg/s) is the standard deviation of second residual equation  $y_2$  error,  $\sigma_{v3}$  (kg/s) is the standard deviation of third residual equation  $y_3$  error, and  $\sigma_{v4}$  (bar) is the standard deviation of fourth residual equation  $y_4$  error.

The accepted tuning of the  $R$  matrix for each mode is displayed in Table 1. The  $Q$  matrix is a constant matrix that is invariant to the isolated mode where its value is set to  $Q = \text{diag}[(10^{-1})^2, (10^{-2})^2, (10^{-3})^2]$ . The  $R$  matrix changes diagonal gains in dependence of isolated mode. The highest gains represent immense uncertainty of the unwanted residual

Mode	nominal		mitigated	
	$m_A$ (%)	$m_{EGR}$ (%)	$m_A$ (%)	$m_{EGR}$ (%)
1/ No fault	98	75	n/a	n/a
2/ Fault A	98	62	98	69
3/ Fault B	98	30	99	62
4/ Fault C	96	62	98	72

Table 2. Summary of performance - the percentage of estimated air flow and EGR flow meeting the accuracy bounds; Fault A - Aftertreatment fault, Fault B - EGR valve fault, Fault C - Balance valve fault

equation. The effect of tuning shown in Table 1 is demonstrated in the following figures. The nominal tuning (no fault induced) is displayed in Figure 5 and in Figure 6. Note that only EGR flows greater than 0.03 kg/s, corresponding to 30% of the maximum EGR flow, are used in the computation of the accuracy metric since we are evaluating the relative error.

The impact of an aftertreatment fault with nominal solver settings is shown in Figure 7 and Figure 8. The mitigated solver settings (Mode 2, Table 1) are plotted in Figure 9 and in Figure 10. An aftertreatment fault has the main impact on the estimated EGR mass-flow displayed in Figure 8 where the applied mitigation strategy can improve the estimation accuracy by 7%, shown in Figure 10.

The EGR fault result under the nominal (Mode 1, Table 1) solver setting is shown in Figure 11 and in Figure 12. Figure 12 shows a severe impact on EGR mass-flow accuracy. Omitting the EGR model by using Mode 3, Table 5 setting, leads to an improved air mass-flow estimate shown in Figure 13 and a much improved EGR mass-flow estimate documented in Figure 14.

The impact of a turbine balance valve fault can be seen in Figure 15 and Figure 16. The mitigation strategy consists of bypassing the prime VEGRO algorithm and estimating the air mass-flow as a difference between the charge flow model and the EGR mass-flow computed via the EGR valve model, see Figure 17 and Figure 18. The summary of performance indicators under nominal and fault mitigated observer conditions is shown in Table 2.

From the results discussed above, it can be concluded that the mitigation strategy is suitable for all the faults. The sensitivity to simulated faulty components (note below) however differs. The EGR flow rate has the highest correction ratio after the mitigation, it jumps from 30% to 62% showing the highest mitigation efficiency.

**Note and assumption on simulation scenario of applied faults:** All faults in this study are simulated by altering a model with the greater mass flow against the nominal data. This represents an equivalent way of fault present in the system. The blocking issue is in simulation represented by mod-

eled mass-flow (of a given faulty component) being doubled compared to the nominal component mass-flow. This way the reference data shows a smaller flow than the expected faulty component mass flow.

## 4.2. Fault isolation strategy

The VEGRO fault isolation strategy can be summarized in Table 3, where three faults A., B., and C. are expected (Polóni et al., 2021). Each fault has its unique logic related to several indication signals and each fault has an assigned action. Each indicative logical relation from Table 3, e.g. for fault A.  $m_{A,EGRv} = m_{A,\lambda}$ ,  $p_{3a} = p_{3a,snr}$  and  $p_4 > p_{4,nom}$  can be transformed into inequality logical condition. The tested inequality logical conditions are provided as the outcome of the fault detection subsystem based on the means of statistical hypothesis testing, see section 4.3 for more details. The variables in Table 3 have the following meaning:  $m_{A,EGRv}$  is the airflow computed as the charge flow minus the EGR flow given by the EGR valve equation,  $m_{A,\lambda}$  is the airflow given by the lambda model,  $p_{3a,snr}$  is the measured exhaust pressure by the pressure sensor,  $p_{3a}$  is the solved exhaust pressure from the balance flow,  $p_{4,nom}$  is the virtual sensor value for the downstream turbine pressure,  $p_4$  is the solved downstream turbine pressure from the balance flow.

The summary of inequality logic conditions with expected outputs for each isolated mode is given by the reasoning table shown in Table 4.

## 4.3. Fault detection and calibration of the SPRT algorithm

The logical conditions from Table 4 are tested using the principles of Sequential Probability Ratio Test (SPRT) (Basseville & Nikiforov, 1998, p. 37-38). The idea behind the SPRT is the usage of repeated testing of two simple hypotheses about the parameter  $\theta$

$$H_0 : \theta = \theta_0 \quad (37)$$

$$H_1 : \theta = \theta_1 \quad (38)$$

If the decision is taken in favor of  $H_0$ , the sampling and test continue. The test may be stopped after the first sample of observations for which the decision is taken in favor of  $H_1$ . We introduce the following notation. The decision rule is written in a recursive manner as

$$d = \begin{cases} 0 & \text{if } S_{k+1} \geq 0 \\ 1 & \text{if } S_{k+1} = \varepsilon \end{cases} \quad (39)$$

where the recursive term  $S_{k+1}$  is defined as

$$S_{k+1} = \begin{cases} \min(\varepsilon, S_k + s_k) & \text{if } S_k + s_k > 0 \\ \max(0, S_k + s_k) & \text{if } S_k + s_k \leq 0 \end{cases} \quad (40)$$

<sup>3</sup>To isolate the EGR fault uniquely, the  $m_{A,EGRv} \neq m_{A,\lambda}$  condition test needs to return logical one as the first and the  $p_{3a} \neq p_{3a,snr}$  condition test needs to return logical one as the second. Otherwise mode 3 can be falsely identified as mode 4 during a period when  $p_{3a} \neq p_{3a,snr}$  condition test returns logical one but  $m_{A,EGRv} \neq m_{A,\lambda}$  condition test has not finished yet.

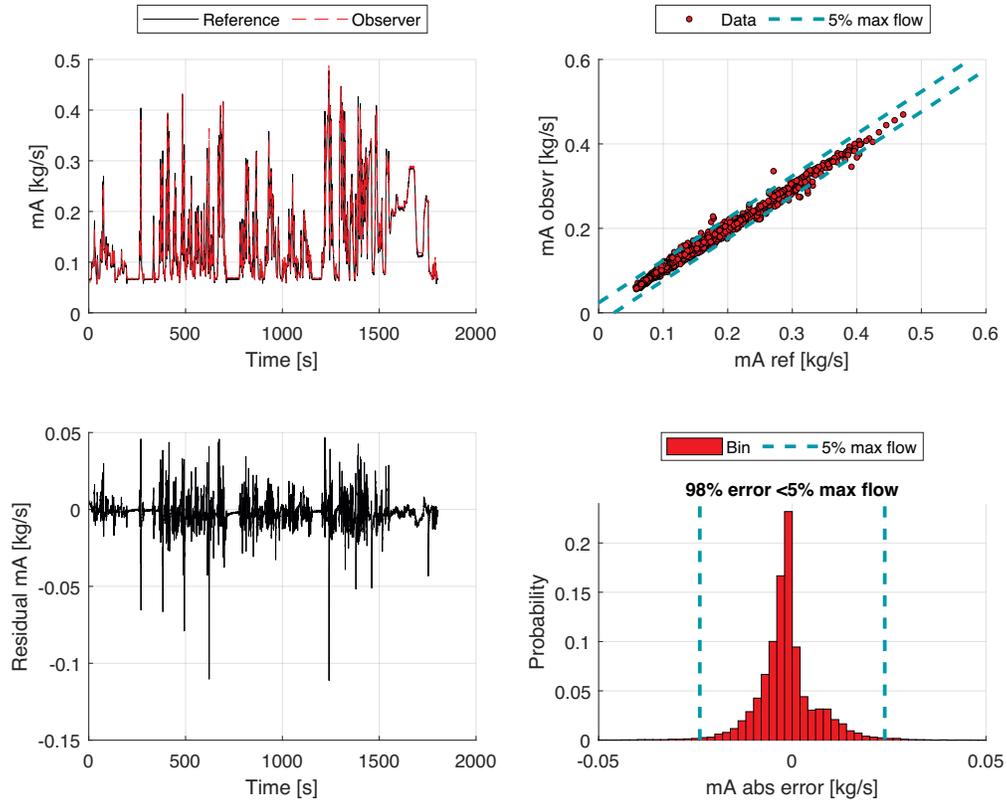


Figure 5. Nominal solver setting - air mass-flow

Fault	Indication	Action
A. Aftertreatment blocked	$m_{A,EGRv} = m_{A,\lambda}$ $p_{3a} = p_{3a,snsr}$ $p_4 > p_{4,nom}$	Section 3.1, use Eq.: (4), (5), (6)
B. EGR blocked	$m_{A,EGRv} < m_{A,\lambda}$ $p_{3a} < p_{3a,snsr}$ $p_4 = p_{4,nom}$	Section 3.1, use Eq.: (3), (4), (6)
C. Balance valve blocked	$m_{A,EGRv} = m_{A,\lambda}$ $p_{3a} < p_{3a,snsr}$ $p_4 = p_{4,nom}$	Use the charge flow model minus the EGR valve model flow (both fed by direct measurement of $p_2$ and $p_{3a}$ )

Table 3. Fault isolation strategy

$m_{A,EGRv} \neq m_{A,\lambda}$ SPRT I	$p_{3a} \neq p_{3a,snsr}$ SPRT II	$p_4 \neq p_{4,nom}$ SPRT III	Isolated mode
0	0	0	No fault, mode 1
0	0	1	Aftertreatment fault, mode 2
1(1st)	1(2nd)	0	EGR fault <sup>3</sup> , mode 3
0	1	0	Balance valve fault, mode 4

Table 4. Reasoning table

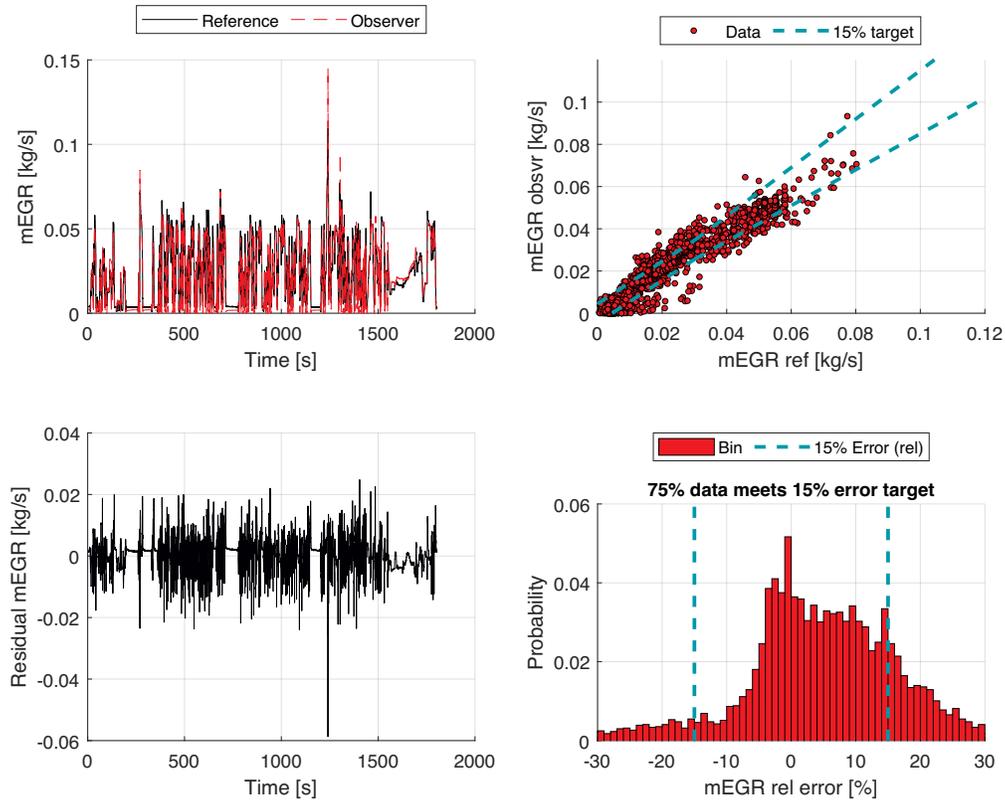


Figure 6. Nominal solver setting - EGR mass-flow

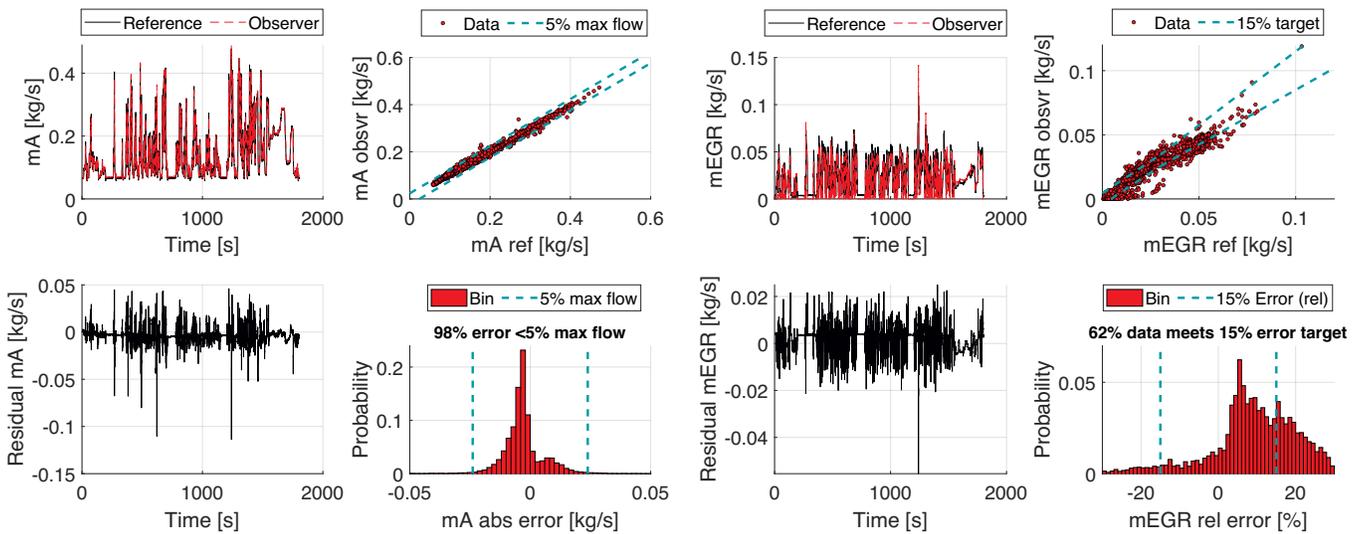


Figure 7. Aftertreatment fault on with nominal solver setting - air mass-flow

Figure 8. Aftertreatment fault on with nominal solver setting - EGR mass-flow

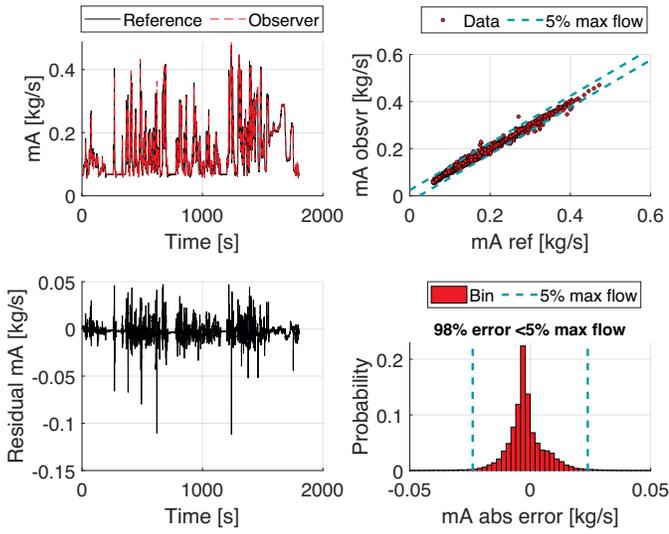


Figure 9. Aftertreatment fault on with mitigated solver setting - air mass-flow

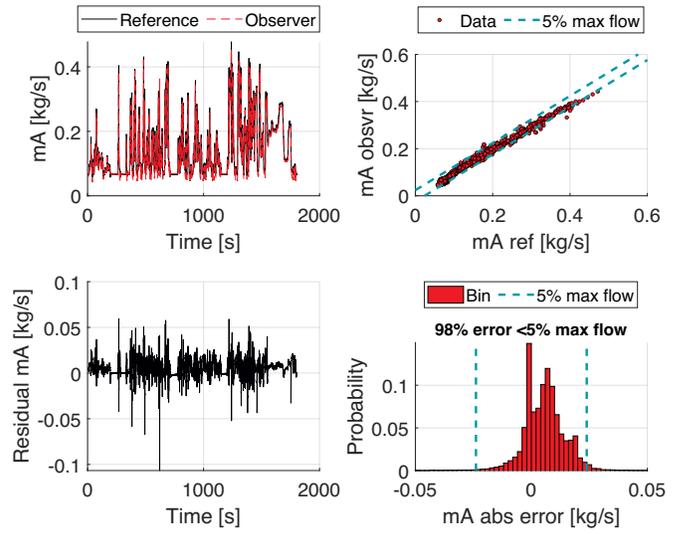


Figure 11. EGR fault on with nominal solver setting - air mass-flow

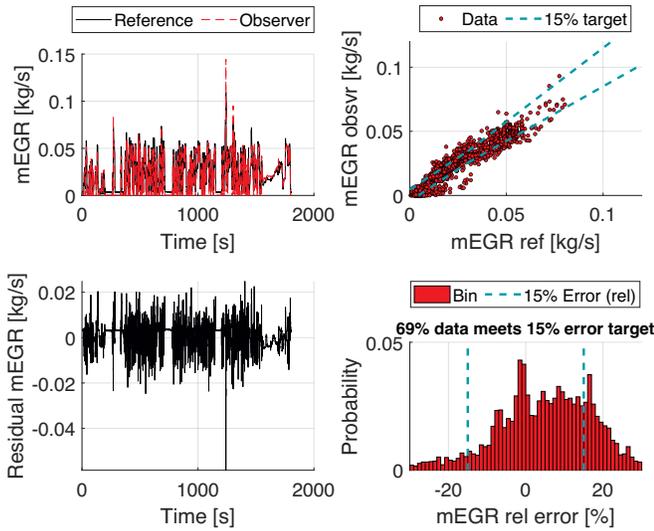


Figure 10. Aftertreatment fault on with mitigated solver setting - EGR mass-flow

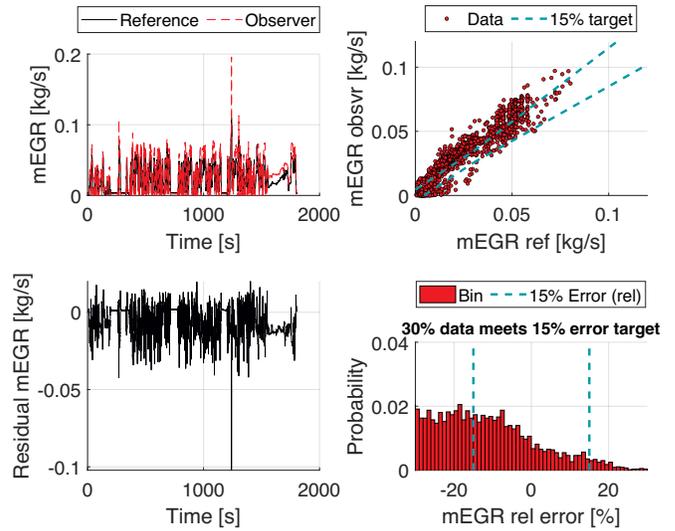


Figure 12. EGR fault on with nominal solver setting - EGR mass-flow

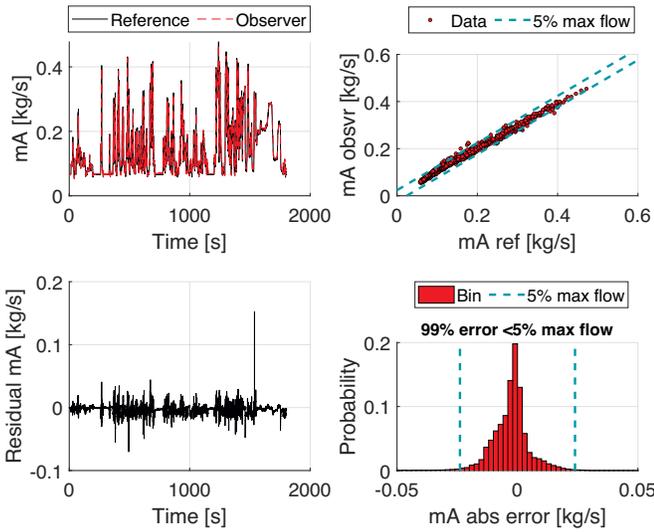


Figure 13. EGR fault on with mitigated solver setting - air mass-flow

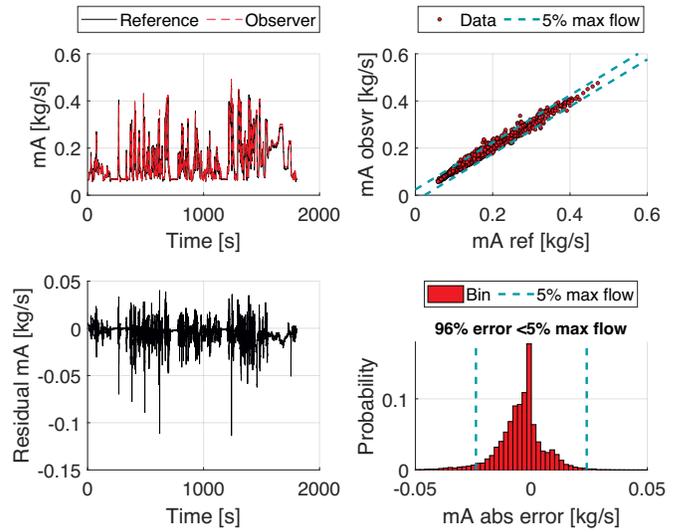


Figure 15. Balance valve fault on with nominal solver setting - air mass-flow

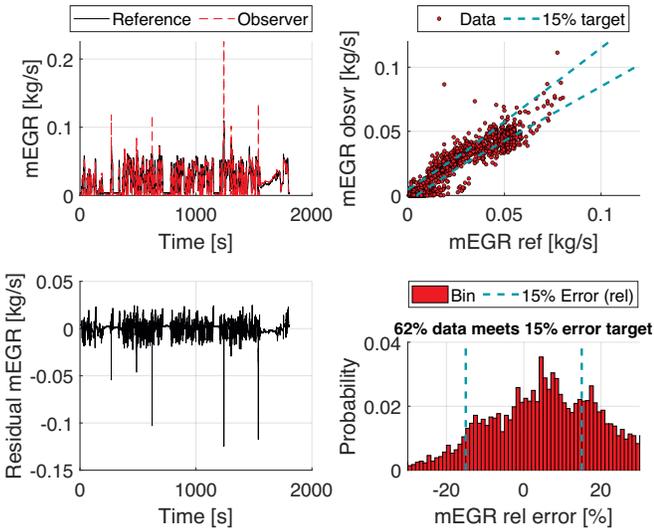


Figure 14. EGR fault on with mitigated solver setting - EGR mass-flow

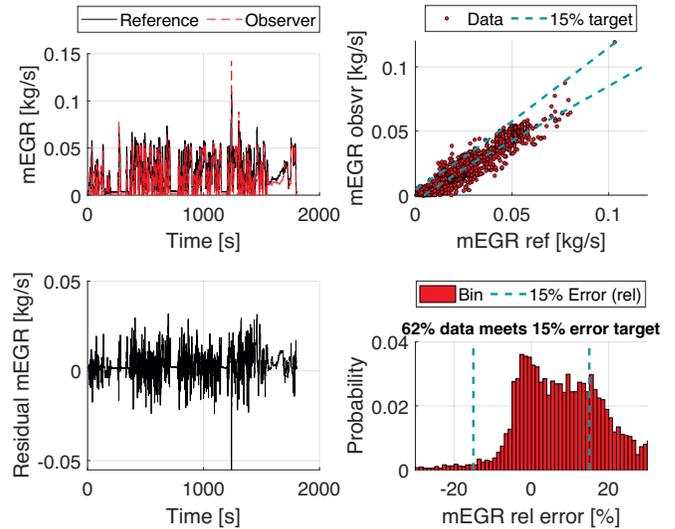


Figure 16. Balance valve fault on with nominal solver setting - EGR mass-flow

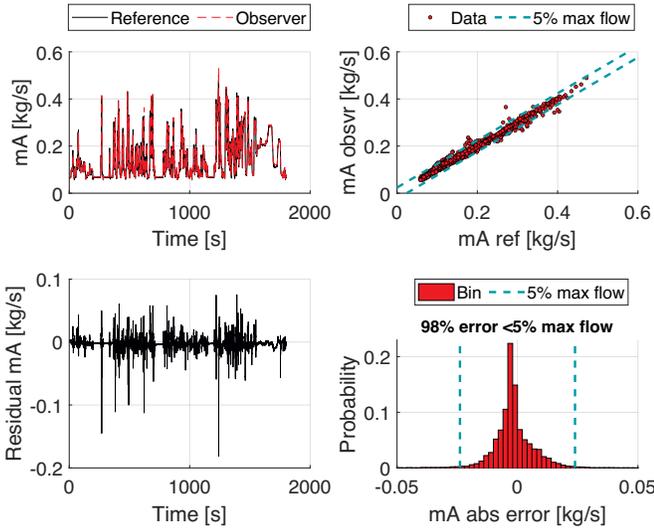


Figure 17. Balance valve fault on with bypassed solver mitigation strategy - air mass-flow

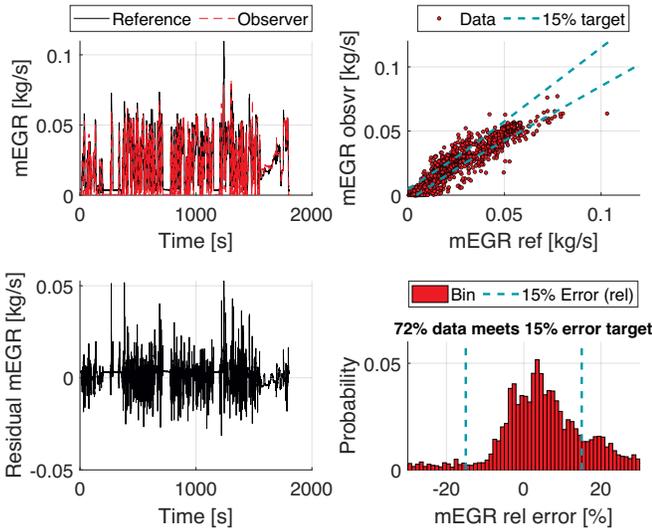


Figure 18. Balance valve fault on with bypassed solver mitigation strategy - EGR mass-flow

In the above expression,  $s_k$  is the log-likelihood ratio, and  $\varepsilon$  is the threshold alarm value. The log-likelihood ratio is defined as

$$s_k = \ln \frac{p(g|\theta_1)}{p(g|\theta_0)} \quad (41)$$

where  $p(\cdot)$  is the probability density function (PDF). It is common for the PDFs to be parametrized where one of the most common models for a real-valued random variable is the normal distribution. The normal distribution is parametrized in terms of the mean  $\mu$  and standard deviation  $\sigma$  that form parameter vector  $\theta_0 = [\mu, \sigma]$  for the null hypothesis  $H_0$ . The alternative hypothesis  $H_1$  is parametrized as  $\theta_1 = [\mu_a, \sigma_a]$ . The  $H_0$  PDF when normalized has the parameters  $\bar{\mu} = 0$ ,  $\bar{\sigma} = 1$  and is given as

$$p(g|\mu, \sigma) = \frac{1}{\sqrt{2\pi}} \exp \left[ -\frac{1}{2} x^2 \right] \quad (42)$$

where  $x$  is the vector of realizations. The expression above is also referred to as the standard normal distribution. The  $H_1$  PDF when normalized has parameters  $\mu_a$ ,  $\sigma_a$  and is given as

$$p(g|\mu_a, \sigma_a) = \frac{1}{\sigma_a \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{x - \mu_a}{\sigma_a} \right)^2 \right] \quad (43)$$

Before any testing takes place, the SPRT algorithm for each expected fault needs to be calibrated. The calibration procedure is as follows (example given for the SPRT I; other tunings for SPRT II and SPRT III are similar):

1. Generate  $H_0$  nominal error signal  $x_0$  that meets the SPRT I entering conditions (see section 4.4)

$$x_0 = m_{A,EGRv}|_{Fault\ off} - m_{A,\lambda}|_{Fault\ off} \quad (44)$$

2. Generate  $H_1$  error signal  $x_1$  in sequence for faults (A., B., C., see Table 3) that meet SPRT I entering conditions (see section 4.4)

$$x_{1,\{A\}} = m_{A,EGRv}|_{\{A\}Fault\ on} - m_{A,\lambda}|_{\{A\}Fault\ on} \quad (45)$$

$$x_{1,\{B\}} = m_{A,EGRv}|_{\{B\}Fault\ on} - m_{A,\lambda}|_{\{B\}Fault\ on} \quad (46)$$

$$x_{1,\{C\}} = m_{A,EGRv}|_{\{C\}Fault\ on} - m_{A,\lambda}|_{\{C\}Fault\ on} \quad (47)$$

3. Compute mean  $\mu$  and standard deviation  $\sigma$  of  $x_0$  and normalize  $x_0$  such that  $\bar{x}_0 = \frac{x_0 - \mu}{\sigma}$
4. Normalize  $x_{1,\{A\}}, x_{1,\{B\}}, x_{1,\{C\}}$  such that  $\bar{x}_{1,\{A\}} = \frac{x_{1,\{A\}} - \mu}{\sigma}$ ,  $\bar{x}_{1,\{B\}} = \frac{x_{1,\{B\}} - \mu}{\sigma}$ , and  $\bar{x}_{1,\{C\}} = \frac{x_{1,\{C\}} - \mu}{\sigma}$
5. Compute mean  $\mu_a$  and standard deviation  $\sigma_a$  for each faulty normalized realization  $\bar{x}_{1,\{A\}}, \bar{x}_{1,\{B\}}, \bar{x}_{1,\{C\}}$ :  $\Delta = \{\mu_{a\{A\}}, \mu_{a\{B\}}, \mu_{a\{C\}}\}$  and  $\Sigma = \{\sigma_{a\{A\}}, \sigma_{a\{B\}}, \sigma_{a\{C\}}\}$
6. Compute the PDFs  $p_0(g|0, 1)$  and  $p_1(g|\mu_a, \sigma_a)$  for the nominal realization  $\bar{x}_0$  and recursively compute  $S_{k+1}$
7. Compute the PDFs  $p_0(g|0, 1)$  and  $p_1(g|\mu_a, \sigma_a)$  for the faulty realizations  $\bar{x}_{1,\{A\}}, \bar{x}_{1,\{B\}}, \bar{x}_{1,\{C\}}$  and recursively

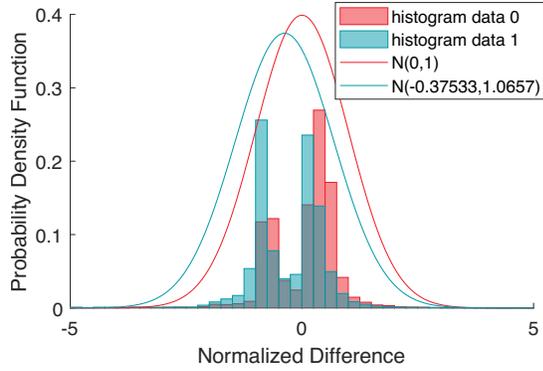


Figure 19. SPRT I: Probability density functions of the null and alternative hypothesis under EGR fault conditions (accepted tuning).

SPRT	$\mu$	$\sigma$	$\mu_a$	$\sigma_a$	$\epsilon$
SPRT I	0.0513	0.083	-0.375	1.066	1000
SPRT II	0.256	5.086	0.400	0.908	7000
SPRT III	-0.231	0.355	1.968	1.949	4000

Table 5. SPRT calibration

compute  $S_{k+1}$  for each

8. Choose the threshold alarm value  $\epsilon$  such that the logical states for SPRT I from Table 4 are achieved
9. Choose one global tuning A, B, or C of pair parameters from sets  $\Delta$ ,  $\Sigma$  for the alternative hypothesis  $H_1$  for which the alarm value is achieved with cumulated sum  $S_{k+1}$
10. Evaluate a chosen tuning for other faults, e.g. if the tuning under fault B is chosen  $\mu_a = \mu_{a\{B\}}$  and  $\sigma_a = \sigma_{a\{B\}}$ , evaluate the tuning for A and C faults such that the logical states for the SPRT I from Table 4 are confirmed.

The tuning of SPRT I is shown in Figure 19 where according to the above calibration procedure in Step 9., tuning B is accepted for the alternative hypothesis  $H_1$ . Similarly, the tuning of SPRT II accepts tuning C and SPRT III accepts tuning A for the alternative hypothesis  $H_1$ . The logical state from Table 4 is the decision variable  $d$ , given by Eq. 39. A summary of tuning parameters for the SPRT I, SPRT II, and SPRT III is displayed in Table 5.

#### 4.4. Entering conditions

The entering conditions serve as an enabler for individual SPRTs to run. If the conditions are met the SPRT subsystem will process its inputs and output the result of the detection. Each SPRT subsystem has its own entering condition logic block that outputs logical zero or logical one. The summary of all entering conditions for individual SPRT subsystems is shown Table 6. As can be seen in Table 6, the SPRT I is defined by a threshold on the estimated EGR flow variable while the SPRT II and SPRT III have the entering logic defined by the pressure error bound of the exhaust pressure and

SPRT	Entering conditions
SPRT I $m_{A,EGRv} \neq m_{A,\lambda}$	<ol style="list-style-type: none"> <li>1. <math>m_{EGR} \geq 0.03</math> (kg/s)</li> <li>2. Lambda sensor is on</li> </ol>
SPRT II $p_{3a} \neq p_{3a,snsr}$	$ p_{3a} - p_{3a,snsr}  \leq 20$ (kPa)
SPRT III $p_4 \neq p_{4,nom}$	$ p_{4,nom} - p_4  \leq 2.5$ (kPa)

Table 6. Entering conditions

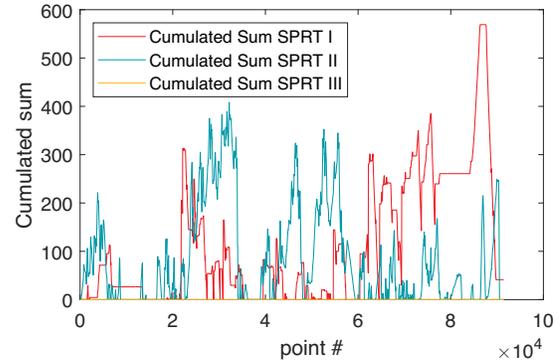


Figure 20. Cumulative sums during the nominal run (no fault induced).

the turbine outflow pressure respectively.

#### 4.5. Detection results

The goal of fault detection and fault isolation strategy is to robustly estimate a fault signature given by the Reasoning table, see Table 4. It is demonstrated in the following figures how the individual faults, when triggered at the beginning of a driving cycle are detected and isolated. The output of individual cumulative sums of individual SPRT blocks for the nominal (no-fault) state is shown in Figure 20. In this case, none of the cumulative sums reaches its threshold value  $\epsilon$ , see Table 5, meaning no fault - Mode 1 is detected. The detection of the aftertreatment fault is given through the fast-integrated cumulative sum of SPRT III displayed in Figure 21. As expected from Table 4, only the SPRT III detects a change that results in the isolation of Mode 2. As given in Table 4, the triggered EGR fault is successfully isolated when the SPRT I and the SPRT II both detect a change in the given order, see Footnote 3. This is shown by two cumulative sums reaching their alarm thresholds. The SPRT I reaches its threshold (1000) first followed by the SPRT II reaching the threshold (7000). The SPRT III stays silent. The balance valve fault is isolated when the SPRT II detects a change and the other SPRTs stay silent as shown in Figure 23. The overall performance of fault detection and fault isolation is working as designed/intended on presented simulated faults.

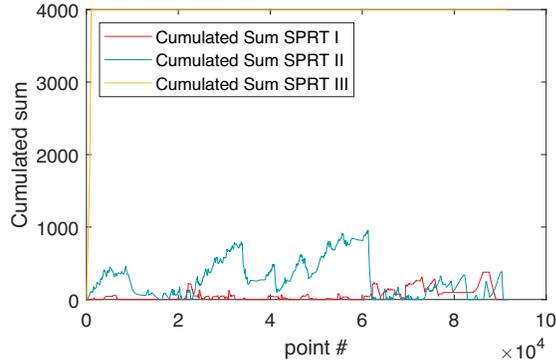


Figure 21. Cumulative sums during the Aftertreatment fault.

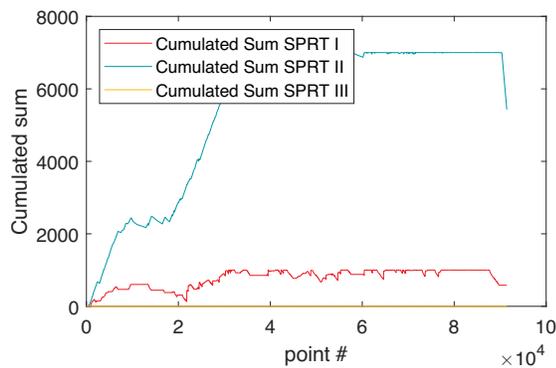


Figure 22. Cumulative sums during the EGR fault.

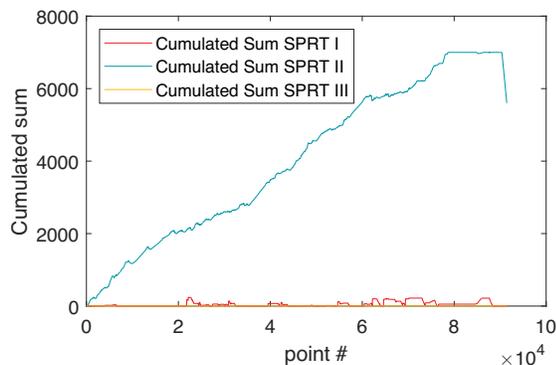


Figure 23. Cumulative sums during the Balance valve fault.

## 5. CONCLUSION

The engine flow observer discussed, is based on solving three polynomial balance flow equations, consisting of six component models, for three unknown pressures: pressures at two turbine inlets and the turbine outlet. A further pressure error equation provides redundancy for solving the flows in the case of a system fault, such as a blockage or stuck valve. The solver, based on the iterated Kalman filter method, is implemented in Simulink, using only elementary Simulink library blocks. The equation solver is a model fusion of the EGR valve, twin-scroll turbine, aftertreatment, balance valve, and charge flow model. The solution is robust, providing accurate estimate of  $p_{3a}$ ,  $p_{3b}$ ,  $p_4$ . Over- (under-) estimation of  $p_{3a}$  pressure would result in over- (under-) estimating both EGR and turbine flows. The sum of the flows must equal the charge flow (relatively insensitive to the  $p_{3a}$  pressure) plus fuel flow to satisfy the flow balance equation. The EGR flow estimate obtained may be, at some points, better than the one based on the valve model only with measured  $p_{3a}$  – this concerns cases when  $p_{3a}$  is close to  $p_2$ . On the other hand, the estimate of the backpressure  $p_4$  is affected by uncertainties in the aftertreatment flow resistance model. As delta pressure on the DPF may be available, it can be used for improving the estimate of the turbine backpressure. Fortunately, turbine flows are not so much sensitive to the back-pressure, if the expansion ratio is not close to 1. Fusing the lambda sensor-based air-flow estimate with the balanced flow estimated airflow adds some robustness to the VEGRO output but not a significant improvement in accuracy, since it is difficult to compensate for the significant time constant of the sensor. The VEGRO observer is being monitored by an independent algorithm – the VEGRO monitor that checks if the VEGRO models reflect the flows of real hardware components. The air path is considering three faults: an aftertreatment blockage, EGR valve blockage, and turbine balance valve blockage that can prevent the VEGRO observer from accurately estimating its outputs. In the case of detected and isolated faults, the fault mitigation strategy can be triggered in a form of rescheduling the IKF's measurement error covariance matrix. It has been demonstrated that the presented fault mitigation strategy is robust to simulated faults, efficient, and simple to implement. Calibration for real-world usage, including thresholds and enabling conditions would need to be done with OBD tolerance parts.

## REFERENCES

- Ahmed Ali, S., Guermouche, M., & Langlois, N. (2015). Fault-tolerant control based super-twisting algorithm for the diesel engine air path subject to loss-of-effectiveness and additive actuator faults. *Applied Mathematical Modelling*, 39(15), 4309-4329. doi: <https://doi.org/10.1016/j.apm.2014.12.047>

- Amin, A. A., & ul Hasan, K. M. (2019). Hybrid fault tolerant control for air–fuel ratio control of internal combustion gasoline engine using Kalman filters with advanced redundancy. *Measurement and Control*, 52(5-6), 473-492. doi: 10.1177/0020294019842593
- Baramov, L., Pekař, J., Dickinson, P., & Polóni, T. (2021, May). *Engine mass flow observer with fault mitigation*. (US Patent Application No. 17/327,066)
- Basseville, M., & Nikiforov, I. V. (1998). *Detection of abrupt changes: Theory and application*. Retrieved from <https://people.irisa.fr/Michele.Basseville/kniga/>
- Chen, J., & Patton, R. (1999). *Robust model-based fault diagnosis for dynamic systems*. Springer, Boston, MA.
- Dahl, J., Wassén, H., Idelchi, A., Baramov, L., Pachner, D., Šantin, O., ... Lánský, L. (2018). Real-time EGR and air flows estimation for heavy-duty engine. In *Symposium for combustion control* (pp. 1–8). Aachen, Germany: RWTH Aachen University.
- Gutiérrez León, P., García-Morales, J., Escobar-Jiménez, R., Gómez-Aguilar, J., López-López, G., & Torres, L. (2018). Implementation of a fault tolerant system for the internal combustion engine's MAF sensor. *Measurement*, 122, 91-99. doi: <https://doi.org/10.1016/j.measurement.2018.03.006>
- Isermann, R. (2017). *Combustion engine diagnosis*. Springer Vieweg.
- Nyberg, M., & Stutte, T. (2004). Model based diagnosis of the air path of an automotive diesel engine. *Control Engineering Practice*, 12(5), 513-525. (Fuzzy System Applications in Control) doi: [https://doi.org/10.1016/S0967-0661\(03\)00120-5](https://doi.org/10.1016/S0967-0661(03)00120-5)
- Pachner, D., Lansky, L., Germann, D., & Eigenmann, M. (2015, April). Fitting turbocharger maps with multidimensional rational functions. In *SAE 2015 world congress & exhibition*. SAE International. doi: <https://doi.org/10.4271/2015-01-1719>
- Polóni, T., Dickinson, P., & Pekař, J. (2021, August). *Methods of health degradation estimation and fault isolation for system health monitoring*. (US Patent Application No. 17/407,047)
- Polóni, T., Rohal'Ilkiv, B., & Arne Johansen, T. (2014). Mass flow estimation with model bias correction for a turbocharged diesel engine. *Control Engineering Practice*, 23, 22-31. doi: <https://doi.org/10.1016/j.conengprac.2013.10.011>
- Reitz, R. D., Ogawa, H., Payri, R., Fansler, T., Kokjohn, S., Moriyoshi, Y., ... Zhao, H. (2020). IJER editorial: The future of the internal combustion engine. *International Journal of Engine Research*, 21(1), 3-10. doi: 10.1177/1468087419877990
- Schilling, A., Amstutz, A., & Guzzella, L. (2008). Model-based detection and isolation of faults due to ageing in the air and fuel paths of common-rail direct injection diesel engines equipped with a  $\lambda$  and a nitrogen oxides sensor. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 222(1), 101-117. doi: 10.1243/09544070JAUTO659
- Simon, D. (2006). *Optimal state estimation*. Wiley.
- Wassén, H., Dahl, J., & Idelchi, A. (2019). Holistic diesel engine and exhaust after-treatment model predictive control. *IFAC-PapersOnLine*, 52(5), 347-352. (9th IFAC Symposium on Advances in Automotive Control AAC 2019) doi: <https://doi.org/10.1016/j.ifacol.2019.09.056>
- Zhang, J., Zhao, H., Feng, Z., & Liu, L. (2021). Fault-tolerant control for turbocharged diesel engine air path via disturbance observer. *International Journal of Systems Science*, 52(7), 1329-1345. doi: 10.1080/00207721.2020.1856449

## APPENDIX A - MASS-FLOW MODELS

The mass-flow models of the balance valve, EGR valve, and aftertreatment system rely on a valve flow model that is typically described by the orifice equation (isentropic expansion model) parametrized with different inputs for each component. It also depends on the flow function  $\Psi$  that is defined as

$$\Psi(\Pi) = \begin{cases} \sqrt{\gamma \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}, & \Pi < \frac{2}{(\gamma+1)^{\frac{\gamma}{1-\gamma}}} \\ \Pi^{\frac{1}{\gamma}} \sqrt{\frac{2\gamma}{\gamma-1} \left(1 - \Pi^{\frac{\gamma-1}{\gamma}}\right)}, & \text{otherwise} \end{cases} \quad (48)$$

where  $\Pi$  is the pressure ratio: upstream pressure over downstream pressure and  $\gamma$  is the specific heat ratio. When the pressure ratio exceeds the critical value for which the flow velocity equals the speed of sound, the flow becomes choked.

### Balance flow

The balance flow is the flow between two banks, schematically displayed in Figure 24. This flow is assumed to consist of two components

- Flow through the balance valve
- Flow in the turbine inlet housing before turbine wheel

The balancing flow model is based on the orifice flow equation which comprises turbulent cross flows in the turbine housing and flow through the balance valve

$$m_{BAL} = A_{eff}(u_{TRB}, p_{3a}, p_{3b}) \Psi \left( \frac{p_{3a}}{p_{3b}} \right) \frac{p_{3a}}{R\sqrt{T_{3a}}} \quad (49)$$

where the constant  $R$  is the specific gas constant of air. The effective area  $A_{eff}$  is a function of valve position and the pressures of the two banks where it captures the communication

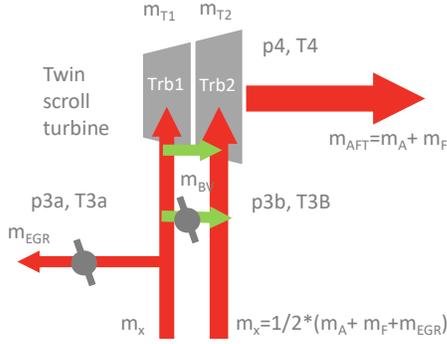


Figure 24. Turbine flow balance. Green arrows indicate the combined "balancing flow".

of flows occurring between the volutes and it is fitted to experimental data. The flow function is approximated by a low complexity rational function as well as  $1/\sqrt{T_{3a}}$  term.

### EGR flow

The EGR flow model uses an orifice model where the effective area is parameterized from data based on the input position. The EGR valve flow model is thus given by

$$m_{EGR} = A.C_{d,EGR}(u_{EGR})\Psi\left(\frac{p_{3a}}{p_2}\right)\frac{p_{3a}}{R\sqrt{T_{3a}}} \quad (50)$$

The term  $A.C_{d,EGR}(u_{EGR})$  is computed using a rational 1D curve with two constrained points.

### Aftertreatment flow

The aftertreatment flow model can be considered as an isothermal restriction with a fixed effective area

$$m_{AFT} = A.C_{d,AFT}\Psi\left(\frac{p_4}{p_0}\right)\frac{p_4}{R\sqrt{T_4}} \quad (51)$$

This restriction model operates on relatively low-pressure ratios.

### Charge flow

The engine charge flow is given as

$$m_{CH} = \frac{V_D p_2}{RT_2} \frac{N_e}{120} \eta_{vol}(N_e, p_2, m_F) \quad (52)$$

where  $V_D$  is the displacement volume,  $T_2$  is the intake manifold temperature,  $p_2$  is the intake manifold pressure,  $N_e$  is the engine speed,  $\eta_{vol}(\cdot)$  is the volumetric efficiency, and  $m_F$  is the amount of injected fuel. The parameterization of the volumetric efficiency map additionally with the fuel quantity is

due to the engine's fuel shot-off events.

### Turbine flow

The turbine flow model makes use of the manufacturer's corrected turbine map  $F_{T1}(\Pi_1)$ , being a function of the pressure expansion ratio where the expansion ratio is  $\Pi_1 = p_{3a}/p_4$ . The turbine flow model is the scaled corrected turbine map given as

$$m_{T1} = \frac{F_{T1}(\Pi_1)}{\sqrt{\frac{T_{3a}}{T_{ref}} \frac{p_{ref}}{p_{3a}}}} \quad (53)$$

where  $p_{ref} = 101325$  Pa and  $T_{ref} = 288$  K. The property of  $F_{T1}(1) = 0$  holds where the function is non-decreasing with a growing expansion ratio. Using these structural assumptions, we approximate the corrected turbine flow and the  $\sqrt{\frac{T_{3a}}{T_{ref}}}$  function by rational functions on a dense grid of points. More details on fitting the turbine map to rational polynomial models can be found in Pachner et al. (2015).

### Multivariate rational polynomial models

$$m_{AFT}^N = 19.82663p_4^2 - 19.82663p_4p_0 + 3.03216T_4p_4^2 - 3.03216T_4p_4p_0 \quad (54)$$

$$m_{AFT}^D = 0.6928695p_4 - 0.6019256p_0 + T_4p_4 - 0.8687431T_4p_0 \quad (55)$$

$$m_{EGR}^N = 1.315198p_{3a}^2u_{EGR} - 1.315198p_{3a}p_2u_{EGR} + 0.2011381T_{3a}p_{3a}^2u_{EGR} - 0.2011381T_{3a}p_{3a}p_2u_{EGR} + 15.28352p_{3a}^2u_{EGR}^2 - 15.28352p_{3a}p_2u_{EGR}^2 + 2.337366T_{3a}p_{3a}^2u_{EGR}^2 - 2.337366T_{3a}p_{3a}p_2u_{EGR}^2 \quad (56)$$

$$m_{EGR}^D = 0.7164297p_{3a} - 0.6579359p_2 + 1.034004T_{3a}p_{3a} - 0.9495812T_{3a}p_2 - 0.2924857p_{3a}u_{EGR} + 0.2686053p_2u_{EGR} - 0.4221368T_{3a}p_{3a}u_{EGR} + 0.3876709T_{3a}p_2u_{EGR} + 4.254647p_{3a}u_{EGR}^2 - 3.907271p_2u_{EGR}^2 + 6.140619T_{3a}p_{3a}u_{EGR}^2 - 5.63926T_{3a}p_2u_{EGR}^2 \quad (57)$$

$$\begin{aligned}
m_{BAL}^N &= 0.2664645p_{3a}^2 - 0.2664645p_{3a}p_{3b} \\
&+ 0.0407514T_{3a}p_{3a}^2 - 0.0407514T_{3a}p_{3a}p_{3b} \\
&- 0.2670565p_{3a}^3 + 0.936663p_{3a}^2p_{3b} \\
&+ 0.06086311p_{3a}^2u_{TRB} - 0.6696065p_{3a}p_{3b}^2 \\
&- 0.06086311p_{3a}p_{3b}u_{TRB} - 0.04084195T_{3a}p_{3a}^3 \\
&+ 0.1432473T_{3a}p_{3a}^2p_{3b} + 0.009308022T_{3a}p_{3a}^2u_{TRB} \\
&- 0.1024054T_{3a}p_{3a}p_{3b}^2 - 0.009308022T_{3a}p_{3a}p_{3b}u_{TRB} \\
&+ 0.1879963p_{3a}^4 - 0.68958p_{3a}^3p_{3b} \\
&+ 0.108605p_{3a}^3u_{TRB} + 0.7892367p_{3a}^2p_{3b}^2 \\
&- 0.3396391p_{3a}^2p_{3b}u_{TRB} + 0.881318p_{3a}^2u_{TRB}^2 \\
&- 0.287653p_{3a}p_{3b}^3 + 0.2310342p_{3a}p_{3b}^2u_{TRB} \\
&- 0.881318p_{3a}p_{3b}u_{TRB}^2 + 0.02875097T_{3a}p_{3a}^4 \\
&- 0.10546T_{3a}p_{3a}^3p_{3b} + 0.01660936T_{3a}p_{3a}^3u_{TRB} \\
&+ 0.1207009T_{3a}p_{3a}^2p_{3b}^2 - 0.05194227T_{3a}p_{3a}^2p_{3b}u_{TRB} \\
&+ 0.1347832T_{3a}p_{3a}^2u_{TRB}^2 - 0.04399184T_{3a}p_{3a}p_{3b}^3 \\
&+ 0.03533291T_{3a}p_{3a}p_{3b}^2u_{TRB} - 0.1347832T_{3a}p_{3a}p_{3b}u_{TRB}^2 \quad (58)
\end{aligned}$$

$$\begin{aligned}
m_{BAL}^D &= 0.6928695p_{3a} - 0.6362993p_{3b} + T_{3a}p_{3a} \\
&- 0.9183537T_{3a}p_{3b} \quad (59)
\end{aligned}$$

$$\begin{aligned}
m_{CH}^N &= 0.8177518N_e p_2 - 0.1037118N_e^2 p_2 \\
&+ 1.458435N_e m_F p_2 + 0.363473N_e p_2^2 \\
&+ 0.1823196N_e^3 p_2 - 2.574592N_e^2 m_F p_2 \\
&- 0.09281738N_e^2 p_2^2 + 77.27074N_e m_F^2 p_2 \\
&- 6.740158N_e m_F p_2^2 + 0.02263237N_e p_2^3 \\
&- 0.005316643N_e^4 p_2 + 1.616709N_e^3 m_F p_2 \\
&- 0.1391244N_e^3 p_2^2 - 23.11955N_e^2 m_F^2 p_2 \\
&+ 0.60167N_e^2 m_F p_2^2 + 0.1204946N_e^2 p_2^3 \\
&- 208.4496N_e m_F^3 p_2 + 16.7372N_e m_F^2 p_2^2 \\
&+ 0.3330827N_e m_F p_2^3 - 0.03489976N_e p_2^4 \quad (60)
\end{aligned}$$

$$m_{CH}^D = 0.3319612T_2 \quad (61)$$

$$\begin{aligned}
m_{T1}^N &= 1.546778p_4^2 p_{3a} - 1.577428p_4 p_{3a}^2 \\
&+ 0.03064934p_{3a}^3 + 0.09704072T_{3a}p_4 p_{3a} \\
&- 0.09896357T_{3a}p_4 p_{3a}^2 + 0.001922857T_{3a}p_{3a}^3 \quad (62)
\end{aligned}$$

$$\begin{aligned}
m_{T1}^D &= 0.7854754p_4^2 - p_4 p_{3a} \\
&+ 0.4707823T_{3a}p_4^2 - 0.5993596T_{3a}p_4 p_{3a} \quad (63)
\end{aligned}$$

$$\begin{aligned}
m_{T2}^N &= 4.825973p_4^2 p_{3b} - 4.848827p_4 p_{3b}^2 \\
&+ 0.02285446p_{3b}^3 + 0.3027686T_{3b}p_4^2 p_{3b} \\
&- 0.3042024T_{3b}p_4 p_{3b}^2 + 0.001433827T_{3b}p_{3b}^3 \quad (64)
\end{aligned}$$

$$\begin{aligned}
m_{T2}^D &= 0.8010185p_4^2 - p_4 p_{3b} \\
&+ 0.4800981T_{3b}p_4^2 - 0.5993596T_{3b}p_4 p_{3b} \quad (65)
\end{aligned}$$

## APPENDIX B - THE ITERATED KALMAN FILTER

The iterated Kalman filter (IKF), (Simon, 2006, p. 410) is summarized in the following steps:

1. The nonlinear system and measurement equations are given as follows

$$x_k = f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}) \quad (66)$$

$$y_k = h_k(x_k, v_k) \quad (67)$$

$$w_k \sim (0, Q_k) \quad (68)$$

$$v_k \sim (0, R_k) \quad (69)$$

2. Initialize the filter as follows

$$\hat{x}_0^+ = E(x_0) \quad (70)$$

$$P_0^+ = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] \quad (71)$$

3. For  $k = 1, 2, \dots$  execute steps a, b and c.

- (a) Perform the following time-update equations

$$P_k^- = F_{k-1}P_{k-1}^+ F_{k-1}^T + L_{k-1}Q_{k-1}L_{k-1}^T \quad (72)$$

$$\hat{x}_k^- = f_{k-1}(\hat{x}_{k-1}^+, u_{k-1}, 0) \quad (73)$$

where the partial derivative matrices  $F_{k-1}$  and  $L_{k-1}$  are defined as follows

$$F_{k-1} = \left. \frac{\partial f_{k-1}}{\partial x} \right|_{\hat{x}_{k-1}^+} \quad (74)$$

$$L_{k-1} = \left. \frac{\partial f_{k-1}}{\partial w} \right|_{\hat{x}_{k-1}^+} \quad (75)$$

Until here the iterated Kalman filter is the same as the standard discrete-time extended Kalman filter (EKF).

- (b) Perform the measurement update by initializing the iterated EKF estimate to the standard EKF estimate

$$\hat{x}_{k,0}^+ = \hat{x}_k^- \quad (76)$$

$$P_{k,0}^+ = P_k^- \quad (77)$$

For  $i = 0, 1, \dots, N$ , evaluate the following equations (where  $N$  is the desired number of measurement-update iterations)

$$H_{k,i} = \left. \frac{\partial h}{\partial x} \right|_{\hat{x}_{k,i}^+} \quad (78)$$

$$M_{k,i} = \left. \frac{\partial h}{\partial v} \right|_{\hat{x}_{k,i}^+} \quad (79)$$

$$K_{k,i} = P_k^- H_{k,i}^T (H_{k,i} P_k^- H_{k,i}^T + M_{k,i} R_k M_{k,i}^T)^{-1} \quad (80)$$

$$P_{k,i+1}^+ = (I - K_{k,i} H_{k,i}) P_k^- \quad (81)$$

$$\hat{x}_{k,i+1}^+ = \hat{x}_k^- + K_{k,i} \left[ y_k - h(\hat{x}_{k,i}^+) - H_{k,i} (\hat{x}_k^- - \hat{x}_{k,i}^+) \right] \quad (82)$$

(c) The final a posteriori state estimate and estimation-error covariance are given as follows

$$\hat{x}_k^+ = \hat{x}_{k,N+1}^+ \quad (83)$$

$$P_k^+ = P_{k,N+1}^+ \quad (84)$$