A Platform for Overall Monitoring and Diagnosis for Hybrid Vehicles

Christofer Sundström, Erik Frisk, and Lars Nielsen

Vehicular systems
Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden
www.vehicular.isy.liu.se
E-mail: {csu, frisk, lars}@isy.liu.se
Report: LiTH-ISY-R-2925

April 15, 2010

Abstract

Compared with conventional vehicles, designing hybrid electric vehicles includes new features, such as energy management and monitoring of the electrical components. To be able to investigate such issues a simulation platform of a hybrid vehicle, driver, and diagnosis system is developed based on the `CAPSim` model library. The simulation platform is component based, and is able to handle different powertrain configurations. In this investigation a parallel hybrid is modeled and parameterized to represent a long haulage truck. To be able to easily change a model of a component in the vehicle model, every model of a specific component use the same sets of input and output signals. The vehicle model is based on dynamic equations and in general simple models of the components, since the interplay of the components is of major interest in this investigation.

Three model based diagnosis systems are developed and implemented in the platform with a twofolded purpose. The first purpose is to demonstrate the feasibility of the platform. The second purpose is to investigate issues when designing diagnosis systems on vehicle level of a hybrid vehicle powertrain. New features, for example mode switches in the system and a freedom in choosing operating points of the components via the energy management, affect the diagnosis system. The influence of these issues on the performance of the diagnosis system is investigated by design and implementation of three diagnosis systems on a vehicle level. The diagnosis systems are based on three sensor configurations. Two of these consist of several sensors and one system uses few sensors. In one of the systems using information from several sensors, the sensors are placed close to the components that are to be monitored, while the sensors in the other system is based on a different sensor configuration. All three diagnosis systems detect specific faults, here specifically faults in the electrical components in a hybrid vehicle powertrain, but the methodology is generic. It is shown that there is a connection between the design of the energy management and the three diagnosis systems, and that this interplay is of special relevance when models of components are valid only in some operating modes. The diagnosis system based on few sensors is more complex and includes a larger part of the vehicle model than the system based on several sensors placed close to the components to be monitored.

Contents

Chapter 1

Introduction and Outline

In order to study overall monitoring and diagnosis for hybrid vehicles a simulation platform has been developed. The platform contains a driver model, environment, vehicle, controller and energy management, faults, fault detection, and isolation of the faults. These parts of the platform are interacting according to Figure 1.1.

In this study a truck is modeled. In Chapters 2 and 3 the models of the components used in the environment, vehicle driver, controller and energy management, and vehicle are described. In Chapter 2 there is a summary of components implemented in `CAPSim` [1]. Not all of these components are used in the final model for the truck. The modifications to the `CAPSim` models that are used in the vehicle are described in Chapter 3. In Chapter 4 three diagnosis systems for the truck are developed and compared.

1.1 Aim

The aim of this work is twofold:

- Produce a flexible simulation environment for studies of principle and structures of diagnosis applied to hybrid electric vehicles. Initially models for a truck are considered.
- Implement a set of vehicle level diagnosis systems, that are able to detect whether a component is broken. To detect what part of a component that is broken is not of highest priority at this stage. Different sensor configurations are studied and their consequences for test design, isolability and sensibility.

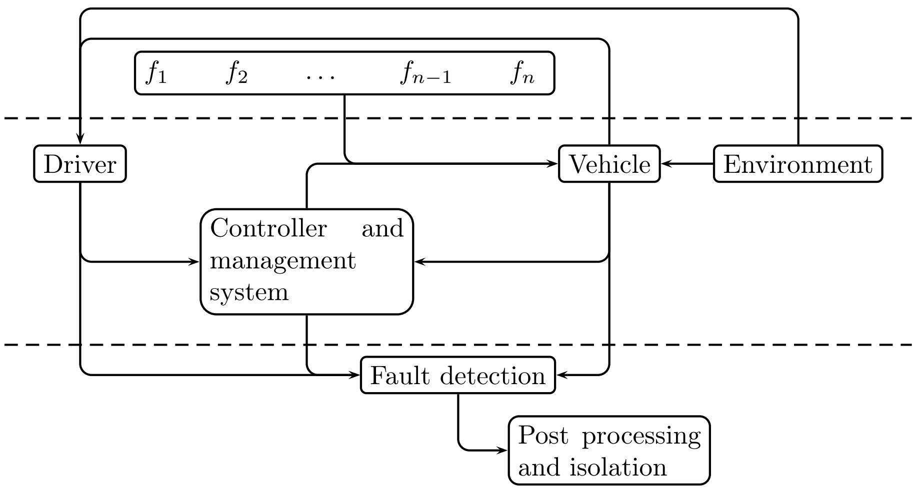

Figure 1.1: The structure of the implemented platform. Above the first horizontal line are the faults induced in the model. Between the lines are the vehicle, driver model, controller and environment. This level includes the information needed to carry out an ordinary simulation. The lowest level includes the diagnosis system.

Chapter 2

Vehicle models from CAPSim

This simulation platform aims to be in conformity with other work, and in this respect utilizes models from the national collaboration `CAPSim` [1]. This chapter recalls some of the models used in `CAPSim`, that are of interest modeling a truck or SEP. For some components several models are described to investigate the possibilities to chose a suitable model for the actual component. The original documentation of the models can be found in the library of `CAPSim`. This chapter describes the models in a slightly different way, but the content is the same. The models used in the specified vehicle are described in Chapter 3, where the modifications made to the original models are presented.

2.1 Vehicle concept

The vehicle concept describes how the different components are connected to each other, e.g. if the vehicle modeled is a conventional vehicle, series or parallel hybrid. The concept gives information about the interaction between the components between the dashed lines in Figure 1.1. The main difference between the concepts is how the different components in the vehicle are connected.

In this section two different parallel hybrids and one series hybrid are described. The difference between the parallel hybrid concepts are where the electric machine is connected to the conventional driveline. In all concepts the inertias in the components are summed and used in the expression for the vehicle acceleration in the chassis.

2.1.1 Concept_parallel_mild1_std

The concept `concept_parallel_mild1_std` includes a fuel tank, internal combustion engine, clutch, fixed stepped gearbox and chassis. In parallel to the combustion engine an electric machine and an energy buffer is connected. This concept represents a vehicle with an integrated starter alternator, or a pre clutch parallel hybrid electric vehicle. The electric motor and the combustion engine are joined together before the clutch. Energy can be regenerated by braking using the electric machine, though the combustion engine has to be connected to the wheels for this to be possible.

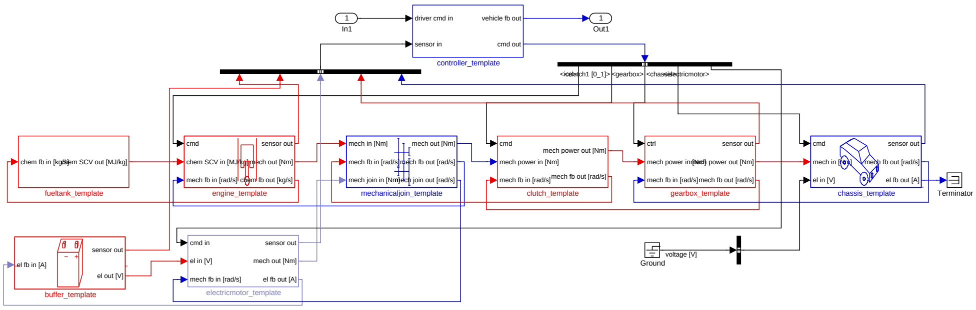

Figure 2.1: The concept_parallel_mild1 model.

2.1.2 Concept_parallel_mild2_std

The concept `concept_parallel_mild2_std` consists of the same components as concept_parallel_mild1_std. The difference compared to the previous model is that the electric machine is connected to the combustion engine after the clutch. Energy can therefore be regenerated when braking using the electric machine, even when the clutch is disengaged. A gear has to be selected during regenerative braking.

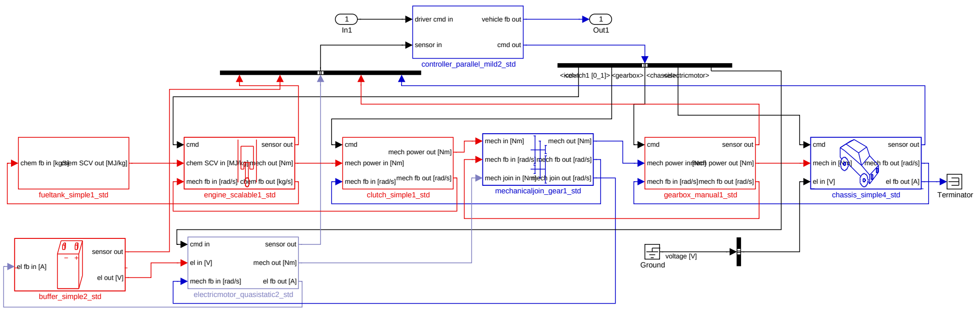

Figure 2.2: The concept_parallel_mild2 model. The difference compared to concept_parallel_mild1 is that the ICE and EM are connected after the clutch in this model.

2.1.3 Concept_series1_std

The last concept described is `concept_series1_std`. It consists of an internal combustion engine that is connected to a generator via a shaft called `inertia`. In the inertia the speed of the ICE and generator is estimated given the torques from the input and output of the shaft, and the inertia of the rotating components. The generator converts the mechanical torque to a current. The electrical part of the generator is connected to the electric split, that also is connected to the electric machine and electric buffer. The electric machine generates a driving, or braking, torque that via the chassis propels the vehicle. The difference in current between the generator and electric machine is stored or taken from the electric buffer.

2.2 Vehicle Driver

The model representing the driver is described in the vehicle driver. Signals as the accelerator position, brake pedal position and gear selection are set in this component.

2.2.1 Vehicledriver_simple1

The model `vehicledriver_simple1` follows a predefined driving cycle. The velocity, v, of the vehicle is compared to the reference speed, v_{ref}. This results in pedal positions for the accelerator and brake pedal, that is sent to the control unit in the vehicle. A simple PI-regulator is used:

$$
\begin{aligned}
e &= v_{ref} - v \\
PI &= \begin{cases} -1, & K_p e + K_i \int edt < -1 \\ K_p e + K_i \int edt, & -1 \leq K_p e + K_i \int edt < 1 \\ 1, & K_p e + K_i \int edt \geq 1 \end{cases} \\
\text{acc } [0..1] &= \max\{PI, 0\} \\
\text{brake } [0..1] &= \min\{PI, 0\} \\
\text{gearSelector } [0..1] &= f(v, v_{ref}) \\
\text{clutch } [0..1] &= f(\text{gearSelector})
\end{aligned}
$$

The clutch pedal is pressed down for a predefined time during a gear shift. No anti-wind-up is implemented in the regulator.

2.3 Controller and energy management

The controller sets the reference torques and speeds for different components. These signals are based on information from sensors and outputs from the vehicle driver. Most of this component is modified in the model used in the truck. Therefore no deeper investigation of the component implemented in CAPSim is of interest in this report.

2.4 Environment

The model of the environment sets parameters such as ambient pressure and temperature.

2.4.1 Environment_simple1

The model used in `CAPSim` called `environment_simple1` sets values to the following parameters:

- Reference speed (driving cycle)
- Gear (not used)
- Slope (both longitudinal and lateral)
- Tire-road friction coefficient (not used)
- Steering wheel position (not connected to the model of the chassis)
- p_{amb}
- T_{amb}

The parameters are defined either by time or distance.

2.5 Buffer

2.5.1 Buffer_simple1

The model `buffer_simple1` is simple and describes the buffer as an equivalent circuit including a voltage source and a resistance, see Figure 2.3. The voltage from the battery is in the implemented model proportional to SoC. Therefore the model most likely represents a super capacitor and not a battery. Observe that SoC not is estimated by integrating the current, instead the power is integrated.

$$
\begin{aligned}
SoC &= SoC_{init} - \frac{1}{E_{max}} \int \left(R_i I^2 + uI\right) dt \\
U &= K_{voltage} SoC
\end{aligned}
$$

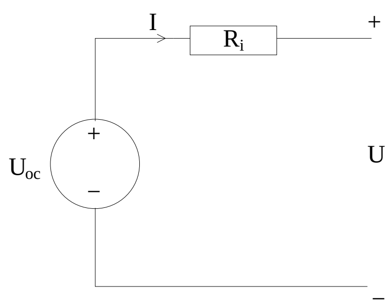

Figure 2.3: The equivalence circuit used in buffer_simple1 model.

2.5.2 Buffer_simple2

In `buffer_simple2` the estimation of SoC is based on the current and not the power as is the case in `Buffer_simple1`. The model uses the equivalent circuit described in Figure 2.3, and $U_{oc}(SoC)$ for one cell is given in Figure 2.4.

$$
\begin{aligned}
U &= U_{oc} - R_i I \\
SoC &= SoC_{init} - \frac{1}{BattCapacity} \int I dt \\
U_{oc} &= f(SoC)
\end{aligned}
$$

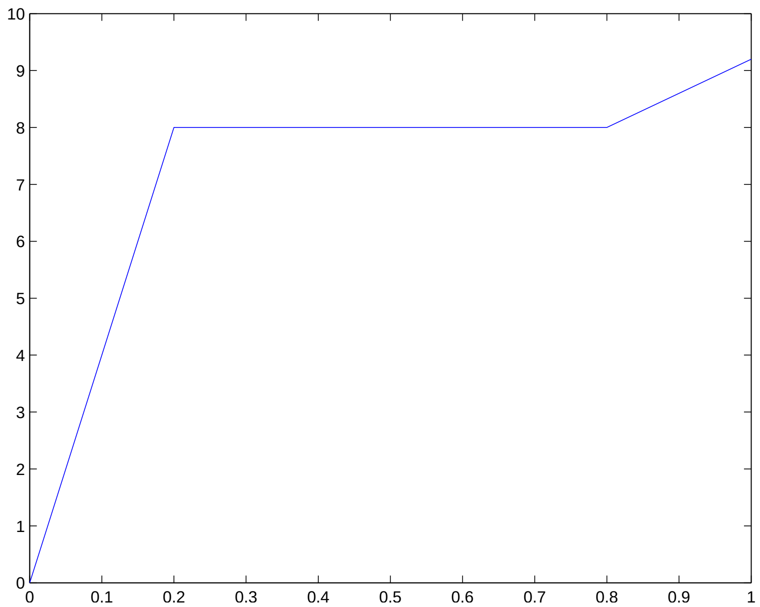

Figure 2.4: $U_{oc}(SoC)$ for one cell in buffer_simple2.

2.5.3 Buffer_rint1

There is one model that is more advanced since R_i and U_{oc} are dependent on SoC and the temperature, T, that is named `buffer_rint1`. When the battery is charged, the current is reduced taking the Coulombic efficiency under consideration. The same equivalent circuit is used as in `buffer_simple1` and 2, and is shown in Figure 2.3. SoC is based on the current and not the power. The model is described in equations:

$$
\begin{aligned}
I_{eff} &= \max(I,0) + \min(I,0)C_{eff} \\
R_i &= f_R(SoC, T) \\
U_{oc} &= f_e(SoC, T) \\
U &= U_{oc} - R_i I_{eff} \\
SoC &= \int I_{eff} dt
\end{aligned}
$$

The value of the Coulombic efficiency is 90.5%. As seen in Figure 2.5, the temperature dependency has only two points in the grid that defines the parameters. This is of course a weakness in the parametrization of the model, especially since the temperatures used are 0°C and 25°C. In the model implemented in `CAPSim`, the temperature is assumed to be constant. It is preferable to add a temperature model for the battery and extend the maps of U_{oc} and R_i.

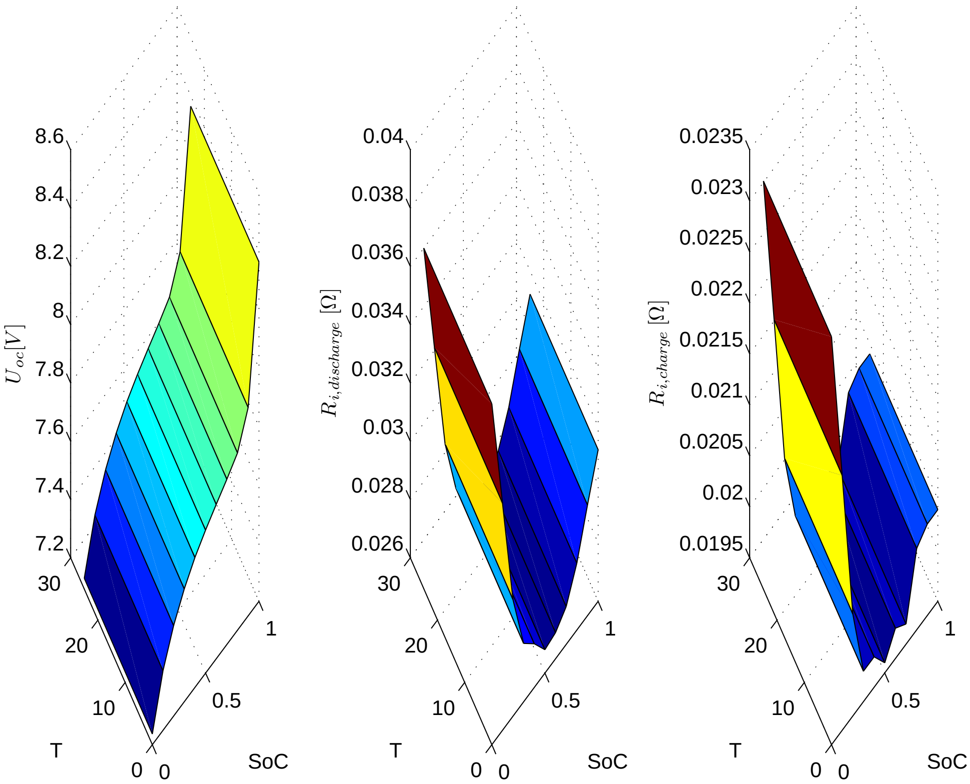

Figure 2.5: The values of the parameters in the model.

2.6 Electric machine

An electric machine is able to operate in all 4 quadrants. This means that the machine is able to reverse in addition to forward operation, and deliver both positive and negative torques. Three models of direct current machines and one alternating current machine are presented in this section.

2.6.1 Electricmotor_quasistatic1

The basic idea in the model `electricmotor_quasistatic1` is that the torque T_{em} is proportional to the current I_{em} and the motor speed ω_{em} is strongly connected to the voltage U_{em}.

$$\left.\begin{aligned} T_{em} &= kI_{em} \\ U_{em} &= k\omega_{em} + R_{em}I_{em} \end{aligned}\right\} \Longrightarrow T_{em} = \frac{k}{R_{em}}U_{em} - \frac{k^2}{R_{em}}\omega_{em} \tag{2.1}$$

The controller of the machine compares either ω_{em} with a reference speed or the voltage needed to achieve a certain torque given the speed of the motor.

In this model the parameter k represents both the torque constant [Nm/A] and speed constant [Vs/rad]. The parameter k is defined by $k = L_m I_{em}$ where L_m is the field mutual inductance, see Guzzella [5].

The following expressions are implemented (it only differs to equation (2.1) when U_{em} or T_{em} is negative):

$$\begin{aligned} U_{em} &= U_{em,control} \\ T_{em} &= \frac{k}{R_{em}}U_{em} - \frac{k^2}{R_{em}}\omega_{em}\,\text{sign}(U_{em}) \\ I_{batt} &= \frac{T_{em}}{k}\frac{|U_{em}|}{U_{batt}} \end{aligned}$$

2.6.2 Electricmotor_quasistatic2

`Electricmotor_quasistatic2` is very similar to `electricmotor_quasistatic1`. The differences between the models are:

- The input signal from the intern controller (the voltage over the motor) is low pass filtered with $\tau = 0.01s$. This is to decrease the stiffness

- The parameter k that is used in equation (2.1) is modeled as two constants in this model. The torque constant used in $T_{em} = k_i I_{em}$ is slightly smaller than the speed constant, k_a. This is one way to model the efficiency of the machine.

$$U_{filt} = \frac{1}{0.01s+1}U \tag{2.2}$$

$$T_{em} = \frac{U_{filt}k_i}{R_{em}} - \frac{\omega_{em}k_a k_i}{R_{em}} \tag{2.3}$$

$$I_{batt} = \underbrace{\frac{T_{em}}{k_i}}_{I_{em}} \frac{|U_{filt}|}{\max\{0.1, U_{batt}\}} \tag{2.4}$$

A drawback with this model is that I_{batt} is limited to $|I_{batt}| \leq 300$ A in a non physical way. The reason for this is that I_{em} and T_{em} not are limited even when $|I_{batt}| > 300$ A. When this occurs all energy the electric machine consumes is not taken from the battery, since the power from the battery is reduced. A better way to solve this problem is to reduce $U_{em,control}$ in the local controller of the electric machine if $|I_{batt}|$ is too large.

The absolute value of U_{filt} in (2.4) is non physical, but since the model is valid in a large part of the operating range for realistic parameter values of k_a and R_{em}, the model is used unchanged to keep consistency with `CAPSim`. This model assumption has some implications on the developed diagnosis system, and comments on this are included in Section 4.5.3.

2.6.3 Electricmotor_simple1

The basic equations used in the model `electricmotor_simple1` are:

$$U - R_{em}I - L\frac{dI}{dt} - k_e\frac{d\Theta}{dt} = 0$$
$$k_m I - T = 0$$

Where the inductance term models the inertia of the magnetic field in the machine. The last term in the equation above is the back EMF, and it is this term that drives the engine.

In the implemented model in `CAPSim` there is a look-up table. This table reduces the requested torque when low torques are requested. In the model the torque is reduced with a factor given in Figure 2.6, when the magnitude of $T_{em,requested}$ is smaller than 0.3 Nm.

2.6.4 Electricmotor_pmsm1

Compared to induction machines, permanent magnet synchronous machines (pmsm) have higher efficiency. `Electricmotor_pmsm1` is a model of a pmsm. The disadvantage with this type of machine is the higher cost that is related to the permanent magnets that are used.

A pmsm consists of a stator that has a winding and a rotor that has permanent magnets mounted on either the outside or included inside [2]. By applying a voltage that results in a current in the stator, the rotor starts to move.

This is an AC machine and the physics of the machine is complicated. There is for example a transformation called Park transformation used in the implementation. The transformation is described in the documentation to the model in `CAPSim` [1].

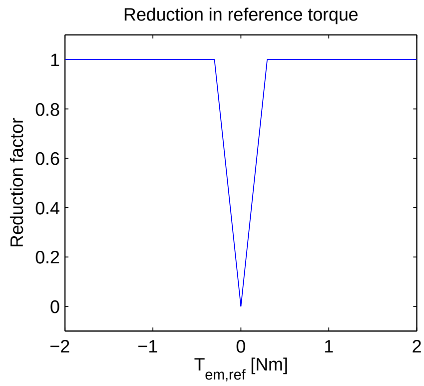

Figure 2.6: Different torques and SoC when the truck is driving `FTP75` with the parameters and models described in this chapter.

2.7 Generator

The generator is a component that only can transform mechanical power to electrical power. Therefore it is a subset of an electric machine.

2.7.1 Generator_quasistatic1_std

The model of the generator named `generator_quasistatic1_std` is very similar to `electricmotor_quasistatic1` (section 2.6.1) and uses the relationships in equation (2.1). The major difference is that the current is a feedback signal in the electric motor and a forward signal in the generator. The torque used by the generator is sent back to the inertia, but sent to the chassis in the electric motor case. When the component acts as a generator, the current is by definition negative.

2.8 Inertia_simple1

The model `inertia_simple1` simulates a shaft that connects two components, e.g. the engine and the generator. The angular speed of the shaft is estimated using the following equations:

$$\begin{aligned}
T_{net} &= T_{ice} - T_{gen} \\
J_{tot} &= J_{ice} + J_{gen} + J_{shaft} \\
\dot{\omega}_{inertia} &= \frac{T_{net}}{J_{tot}} \\
\omega_{inertia} &= \int \dot{\omega}_{inertia} dt
\end{aligned}$$

This speed is equal to the speed of the engine and the generator in the series hybrid concept.

2.9 Fueltank_simple1

The model fueltank_simple1 basically integrates $\dot{m}_f$ that is the fuel mass flow to the engine. The weight reduction of the vehicle when fuel is consumed is also calculated in this model.

$$
\begin{aligned}
m_{fueltank} &= \int -\max\{0, \dot{m}_f\} dt \\
m_{f,reduction} &= \int \max\{0, \dot{m}_f\} dt
\end{aligned}
$$

2.10 Engine

2.10.1 Engine_gasoline1

The modeled engine in engine_gasoline1 is port injected. This model is a mean value engine model (MVEM) that does not model the variations in e.g. torque during the cycle. The air mass-flow through the throttle is modeled as a first order system and $\dot{m}_{ac}$ is estimated using η_{vol}. The injected fuel, $\dot{m}_{fi}$ is calculated using the estimated $\dot{m}_{ac}$ and λ. The amount of air, m_{ac} and fuel, m_{fi} can easily be expressed in $\dot{m}_{ac}$ and $\dot{m}_{fi}$. The model also includes fuel puddles, resulting in a non-stoichiometric combustion during transients. The torque on the crank shaft is estimated using the following approximation:

$$T = a_3 m_{ac}^3 + a_2 m_{ac}^2 + a_1 m_{ac} + a_0$$

The lambda sensor is modeled as a first order system

$$\frac{d}{dt}\lambda_s(t) = \frac{1}{\tau_\lambda}(\lambda(t - \tau_d) - \lambda_s(t))$$

where λ_s is the output from the sensor and λ the actual air to fuel ratio.

2.10.2 Engine_simplemap1

The model engine_simplemap1 is based on two look-up maps:

- $T = f(\omega_e, accpedal)$
- sfc $= f(\omega_e, T)$

First the torque delivered on the crank shaft is estimated using a map. The fuel consumption is then calculated from the specific fuel consumption, sfc. Except from the two maps there are basically only scaling factors in the model.

The disadvantage using a map-based model is that there are many parameters to tune if a new engine is to be implemented.

2.10.3 Engine_scalable1

`Engine_scalable1` is based on a model in QSS [4], with the main difference that this model is a forward facing model and the original model is backward facing. The model estimates the mean effective pressure of the engine to calculate the torque delivered by the engine. The mean effective pressure is defined as

$$p_{me} = \frac{4\pi T}{V_d}$$

p_{me} can be calculated using: $p_{me} = e p_{m\phi} - p_{me0}$, where e is the thermodynamic efficiency of the engine, p_{me0} the pumping and friction losses ($p_{me0} = p_{me0,f} + p_{me0,g}$) and $p_{m\phi}$ the fuel mean effective pressure. The gas exchange term is assumed to be constant while the friction term is estimated using the ETH friction model described by:

$$p_{me0,f} = k_1(k_2 + k_3 S^2 \omega^2)\Pi_{max}\sqrt{\frac{k_4}{B}}$$

2.10.4 Engine_scalable2

`Engine_scalable2` is similar to `engine_scalable1`. To represent the dynamics in the torque delivered by the engine the following expression is added to engine_scalable1:

$$\ddot{T}_{out} = c_1 \left(T - T_{out}\right) \omega^2 - c_2 \omega \dot{T}$$

where T is the demanded torque (the torque delivered from the engine in `engine_scalable1`). The constants c_1 and c_2 are designed with the approximation that it takes about two crank shafts for a four stroke engine to reach stationarity. In the implemented model the values of the parameters are 0.24 and 0.882.

2.11 Clutch_simple1

The clutch is difficult to model. The model in `Clutch_simple1` is based on the control signal `ctrl` that is 0 when the clutch pedal is pressed down and the clutch is disengaged. A flywheel is included in the model and the slip between the flywheel and the outgoing shaft is estimated. In the model, the delivered torque is set to a constant value, that changes sign depending on the sign of $\Delta\omega$, when $|\Delta\omega| > 1\,\mathrm{rad/s}$ and disengaged $= 1$ in equation (2.5). The torque delivered from the clutch when it is disengaged is given by:

$$T_{out} = T_{clutch} = T_{max}\text{ctrl} \cdot \text{sign}\underbrace{\left(\omega_{clutch} - \omega_{out}\right)}_{\Delta\omega} \cdot \text{disengaged} \qquad (2.5)$$

2.12 Mechanicaljoin_gear1

The model of the component that joins the conventional and electric machine parts of the driveline together is `Mechanicaljoin_gear1`. In this component a gear ratio, u_{em}, could be used between the electric motor and the other two shafts. The default value of u_{em} is 1.

$$T_{out} = T_{ice} + T_{em} u_{em}$$

$$\omega_{out} = \omega_{ice} = \frac{\omega_{em}}{u_{em}}$$

$$J_{out} = ice = J_{ice} + J_{em} u_{em}^2$$

2.13 Electricsplit_simple1

The model of the component that connects the generator and the electric motor to the battery is `Electricsplit_simple1`. The component is assumed to be ideal and the current to the battery is simply expressed by:

$$J_{batt} = Jgen + Jem$$

2.14 Gearbox_manual1

`Gearbox_manual1` is a model of a regular gearbox. The used gear is sent to the gearbox as a control signal. From this signal the gear ratio is achieved. The losses in the gearbox are modeled as one constant term and one proportional term that is multiplied with the torque from the gearbox. An expression for this is

$$T_w = \begin{cases} (T_{clutch} - T_{loss})\,\eta_{gear} & T_w > 0 \\ (T_{clutch} - T_{loss})\,\frac{1}{\eta_{gear}} & T_w < 0 \end{cases}$$

where η_{gear} can be implemented to depend on the selected gear and T_{loss} depends on the speed on the clutch and the selected gear. The inertia from the input shaft is compensated for the gear ratio.

2.15 Chassis

In the chassis, the input shaft is connected to the final gear and finally to the wheels. The torque is then transformed to a force acting on the road. The losses according to drag, rolling resistance etc are also estimated.

2.15.1 Chassis_simple1

The first model of the chassis described in this text is `Chassis_simple1` and estimates the drag and rolling resistance by:

$$B = \underbrace{C_r mg \left(1 - \frac{1}{2.81^{(0.5v)}}\right)}_{roll} + \underbrace{\frac{1}{2}\rho C_d A v^2}_{drag}$$

To calculate the speed of the vehicle the following equation is used:

$$v = v_{init} + \frac{1}{m} \int (T_{in} u - T_{brake}) \frac{1}{r} - Bv \; dt$$

The model for the rolling resistance has the benefit of being 0 when $v = 0$.

The chassis model includes functionality to handle the slip between the tires and the road and the user is able to choose if this feature is to be included in the calculations or not.

2.15.2 Chassis_simple2

In the model `Chassis_simple2` the air drag is modeled as in `chassis_simple1` and the rolling resistance $F_r = mg\cos(\alpha)(C_{r,0} + C_{r,1}v)$. $C_{r,1}$ is often 0 and α is the slope of the road.

The model sums the torques for the different contributions and a net torque is calculated. The slope of the road is included in the model. Also the inertia is used to estimate the acceleration of the vehicle accordingly to $\dot{\omega} = \frac{T_{net}}{J}$. This model is able to handle negative velocities.

2.15.3 Chassis_simple4

The inertia of the vehicle is included in `chassis_simple4`. The road slope is only used to estimate the change in potential energy and not used in the expression for the rolling resistance. To be able to handle low velocities and even stand still, T_{roll} is proportional to ω_w at low speeds. There is a block that makes T_{roll} to change sign if the vehicle is reversing.

The equations used in the model of the chassis are presented below

$$
\begin{aligned}
m &= m_{init} - m_{fueltankreduction} \\
T_{drag} &= \frac{1}{2}\rho C_d A_f \omega^2 r^3 \\
T_{roll} &= \begin{cases} mgC_r r, & 1000\omega > mgC_r r \\ 1000\omega, & -mgC_r r \leq 1000\omega < mgC_r r \\ -mgC_r r, & 1000\omega \leq -mgC_r r \end{cases} \\
T_{slope} &= mg\sin\alpha \\
T_{net} &= T_{gb,tot}u_{final} - T_{drag} - T_{brake} - T_{roll} - T_{slope} \\
m &= m_{init} - m_{fueltankReduction} \\
\dot{\omega}_w &= \frac{T_{net}}{J_{gb,tot}{u_{final}}^2 + mr^2} \\
v &= \omega_w r \\
\Theta &= \int \omega_w dt \\
s &= \Theta r \\
\omega_{gb} &= \omega_w u_{final}
\end{aligned}
$$

The implementation of the chassis does not fully support negative velocities. The rolling resistance can change sign, but the torque applied by the mechanical brakes does not change sign. This could lead to problems at stand still, since if the vehicle is slightly reversing nothing is forcing the vehicle to speed zero.

Chapter 3

Truck

A model of a truck is implemented using component models. The modifications to the models presented in Chapter 2 are described in this chapter. Some of the parameters used are also presented.

The implemented truck is a 40 tons vehicle is assumed to be a long haulage vehicle. The added components compared to a regular vehicle are an electric machine and a battery package. As mentioned earlier, there is no component for the power electronics in the model, instead this functionality is included in the model for the electric machine. The configuration of the hybridization is a parallel hybrid.

3.1 Concept - concept_parallel_mild2_std

The model for the concept in the truck is `concept_parallel_mild2_std`. In this model the electric machine is connected to the mechanical powertrain between the clutch and the gearbox. No changes are made in the concept compared to the model included in `CAPSim`.

3.2 Environment - environment_simple1

The model for the environment in the Truck model is `environment_simple1`. No changes are made in the environment model.

3.3 Vehicle driver - vehicledriver_simple1

`vehicledriver_simple1` is used to model the driver. This model is slightly modified to be able to handle a 12 speed gearbox. The different speeds are engaged at the following speeds:

gear selection	change down speed m/s	change down speed m/s
1	eps	0
2	1.5	0.5
3	2.5	1.5
4	4	3
5	6	4
6	8	6
7	10.5	8
8	13	10
9	15	12
10	17	14
11	19	16
12	22	20

The selection is based on the actual speed of the vehicle except the first gear that is based on the reference speed. This is to be able to select the first gear at stand still when the reference speed not is zero.

`Vehicledriver_simple1` is extended with the functionality to disengage the clutch if the engine is running in idle. This functionality is not used in the simulations.

3.4 Buffer - buffer_simple2

In the vehicle model `buffer_simple2` is used to model the battery. The advantage of this model compared to `buffer_rint1` is that there are less parameters to tune. The disadvantage is that the inner resistance and voltage not are dependent on the temperature, as they are in `buffer_rint1`.

The capacity of each cell in the battery is changed from 5.8 Ah to 34.8 Ah. The voltage of each cell is unchanged and is presented in Figure 2.4. The weight of each cell is scaled proportional to the increase in the capacity to 6 kg from 1 kg. There are 32 cells connected in series in the battery, which results in a total weight of 192 kg and a storage capacity of approximately 9 kWh.

The model used in the vehicle model is extended with a simple temperature model. In the model it is assumed that the battery only consists of lithium. This assumption is used when estimating the raise in temperature in the battery, and the power that is added to the lithium is the losses due to the inner resistance. The battery is cooled by assuming that the convection and radiation is small, resulting in that only the conduction is of relevance. The thermal conductivity is estimated to get a reasonable value of the cooling of the battery, according to:

$$P_{cooling} = -\frac{kA}{d}\Delta T$$

where k is the thermal conductivity, A the surface area, d the distance between the lithium and the surrounding air, and ΔT the temperature difference between the lithium and the air. It is assumed that the battery is contained in a box. Furthermore all lithium is assumed to have the same temperature.

To summarize, the equation used to estimate the temperature in the battery is

$$\dot{T} = \frac{P_{loss} - P_{cool}}{m_{batt}nc}$$

where c is the specific heat capacity of lithium and n is the number of cells in the battery. At this stage, the temperature model is not used in the rest of the model, such as the controller, fault detection or temperature dependency in parameters.

3.5 Electric motor - electricmotor_quasistatic2

The model `electricmotor_quasistatic2` is a model of a DC-machine and is used in the model of the truck. The model is unchanged as well as the parameters except from the time constant in the filter on the voltage signal that is changed to 0.1 seconds from 0.01 seconds. The machine has a maximum continuous power of 33 kW and 200 Nm. No functionality for peak power is implemented in the current model.

3.6 Engine - engine_scalable1

`Engine_scalable1` is used to model the engine. The model is based on Willans approximation that is described in [5]. Only the parameters are changed in this model except from the local controller. The parameters are based on Volvo's new D16 that produces 700 hp. More general parameters in the Willans approximation such as the thermodynamic efficiency are the same that are used for a diesel engine in QSS. The following key parameters are used in the model:

Number of cylinders	6	[-]
Stroke	0.165	[m]
Bore	0.144	[m]
Thermodynamic efficiency	0.50	[-]
max torque (speed)	3150 (1250)	[Nm (rpm)]
max power (speed)	515 (1700)	[kW (rpm)]
mass	800	[kg]

Functionality is added in the local controller for the engine to handle idling compared to the original model. The idle control is simple and operates when $\omega_{ice} < (\omega_{idle} + \omega_{idle,offset})$.

$$\begin{aligned}
&\text{if } \omega_{ice} < \omega_{idle} + \omega_{idle,offset} \\
&\quad T_{idle} = T_{ice}(\omega_{ice} = \omega_{idle}) \\
&\text{else} \\
&\quad T_{idle} \\
&\text{end}
\end{aligned}$$

$$T_{ice,req} = \max\{T_{ice,req,controller}, T_{idle}\}$$

where $T_{ice,req,controller}$ is the requested torque from the global controller. T_{idle} is set to a constant value that represents the torque that is achieved to run the engine at a constant speed of 50 $^{rad}/_{s}$ with no load. This controller could be implemented in a better way used e.g. a PID for controlling the engine speed. The implemented strategy is though enough robust for the purpose of this platform.

3.7 Clutch - clutch_simple1

The model of the clutch is `clutch_simple1`. The maximum torque the clutch is able to transfer is increased to 5000 Nm and the weight is set to 100 kg. When the idle control in the engine is used the clutch should be disconnected to avoid the engine is trying to increase ω_{ice} at the same time as $v > v_{ref}$. This would lead to that the controller tries to brake the vehicle. This is only an issue at low engine speeds and is handled in `vehicledriver`.

3.8 Mechanicaljoin - mechanicaljoin_gear1

`Mechanicaljoin_gear1` is used to model the connection between the electric machine and the mechanical driveline. The model is unchanged compared to the model included in `CAPSim`.

3.9 Gearbox - gearbox_manual1

The gearbox used in the model is `gearbox_manual1` and is supposed to represent Volvo's Ishift. The gearbox is modeled as a conventional manual gearbox using 12 gears. The gear ratios differ from 11.73 (1st gear) to 0.78 (12th gear) and the weight of the gearbox is 277 kg. The efficiency of the gearbox is increased to 0.975 in the truck. The inertia depends on the gear selected and no data of this is found. These parameters are most likely needed to be changed in order to achieve an accurate model of the vehicle. For the purposes of this platform this is not an issue at this stage.

3.10 Chassis - chassis_simple4

The model of the chassis is `chassis_simple4`. The parameters are changed to the following values:

Vehicle total mass	40000	[kg]
Tire specification	315/80R22.5	[-]
Rolling resistance	0.007	[-]
Drag coefficient	0.8	[-]
Vehicle frontal area	10	[m^2]
Final gear	3.21	[-]

The total mass of the vehicle is used instead of the sums of the masses of the components, which is the case in the original model.

3.11 Controller - controller_parallel_mild2

The energy management is modified and does not allow the SoC of the battery to decrease below a certain level, SoC_{ref}. When energy is recovered during retarding, the energy stored in the battery is increased. It is not possible to increase SoC above a predefined value, $SoCUpperLimit$, in order to not wear the battery. When $SoC > SoC_{ref}$ energy is primarily taken from the battery. If

$SoC < SoC_{ref}$ the electric motor will never be part of the propulsion of the vehicle. To describe the controller more in detail, the m-code that is implemented in the controller is included below.

```
if Tdemand < 0
    if soc > socUpperLimit
        Tem = 0;
        Tbrake = Tdemand;
    else
        if -maxEMBrakeTorque < Tdemand
            if gear == 0 % EM not connected to the wheels
                Tem=0;
                Tbrake = Tdemand*Gr; % Tdemand is on the ICE.
            else
                Tem = Tdemand/uem;
                Tbrake = 0;
            end
        else
            if gear == 0
                Tem = 0;
                Tbrake = Tdemand;
            else
                Tem = -maxEMBrakeTorque;
                Tbrake = (Tdemand + maxEMBrakeTorque)*Gr;
            end
        end
    end

else
    if socDiff > 0
        if socDiff < 0.02   % To get a smother mode shift
            maxEMTorqueLocal = 50*(socDiff)*maxEMTorque;
        else
            maxEMTorqueLocal = maxEMTorque;
        end
    else
        maxEMTorqueLocal = 0; % Do not use the em at all
    end

    if connected == 0  % If the ICE not is connected to the wheels
        if gear == 0  % Selected gear
            Tem = 0;
            Tice = 0;
        else
            if Tdemand < maxEMTorqueLocal
                Tem = Tdemand;
                Tice = 0;
            else
                Tem = maxEMTorqueLocal;
                Tice = 0;
            end
        end
    else
        if gear == 0 % Will most likely never be in this mode...
            Tem = 0;
            Tice = 0;
        else
            if Tdemand < 0.7*maxEMTorqueLocal % 70% of Tem to save
                battery
                Tem = Tdemand*1/uem;
                Tice = 0;
```

```
            else
                Tem = 0.7*maxEMTorqueLocal;
                Tice = Tdemand - Tem*uem;
            end
        end
    end
end
```

maxEMTorqueLocal: is a parameter that includes information about the maximum torque the electric motor is allowed to deliver and is dependent on SoC accordingly to Figure 3.1.

socDiff: $SoC - SoC_{ref}$

When SoC_{ref} is chosen to 0.50, $maxEMTorqueLocal$ is shown in Figure 3.1.

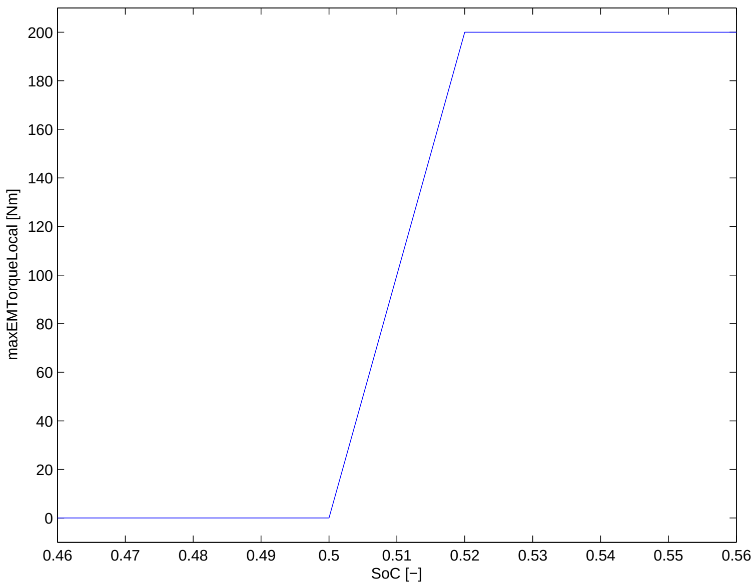

Figure 3.1: The maximum torque the electric motor is allowed to deliver as a function of SoC.

When the required torque is positive, it is checked if the torque the electric motor is able to deliver is enough to fulfill the driving cycle. If not, the ICE is started and delivers the torque the electric motor not was able to deliver. In order to not add tension to the battery, the maximum torque delivered by the electric motor (see Figure 3.1) is multiplied by 0.7. This results in that the maximum torque the electric motor is able to deliver is 140 Nm. During regenerative braking the electric machine is able to apply a negative torque of 200 Nm. In the current model, the gear ratio, u_{em}, between the electric motor and the ICE is set to 1.

3.12 Model equations

In this section the equations used in the models for the vehicle driver, control and energy management and vehicle are tabulated. The blocks in Figure 1.1 concerning the diagnosis system are described in Chapter 4. Some of the equations are already given in either Chapter 2 or earlier in this chapter.

3.12.1 Vehicle driver

$$
\begin{aligned}
e &= v_{ref} - v \\
PI &= \begin{cases} -1, & K_p e + K_i \int edt < -1 \\ K_p e + K_i \int edt, & -1 \leq K_p e + K_i \int edt < 1 \\ 1, & K_p e + K_i \int edt \geq 1 \end{cases} \\
\text{acc } [0..1] &= \max\{PI, 0\} \\
\text{brake } [0..1] &= -\min\{PI, 0\} \\
\text{gearSelector } [0..1] &= f(v, v_{ref}) \\
\text{clutch } [0..1] &= f(\text{gearSelector})
\end{aligned}
$$

3.12.2 Control and energy management

$$
\begin{aligned}
T_{brake} &= f(\text{brake}, \omega_{ice,sens}) \\
T_{acc} &= f(\text{acc}, \omega_{ice,sens}) \\
T_{demand} &= T_{acc} - T_{brake}
\end{aligned}
$$

T_{ice}, T_{em} och T_{brake} is determined by:

```
if Tdemand < 0
    if soc > socUpperLimit
        Tem = 0;
        Tbrake = Tdemand;
    else
        if -maxEMBrakeTorque < Tdemand
            if gear == 0 % EM not connected to the wheels
                Tem=0;
                Tbrake = Tdemand*Gr; % Tdemand is on the ICE.
            else
                Tem = Tdemand/uem;
                Tbrake = 0;
            end
        else
            if gear == 0
                Tem = 0;
                Tbrake = Tdemand;
            else
                Tem = -maxEMBrakeTorque;
                Tbrake = (Tdemand + maxEMBrakeTorque)*Gr;
            end
        end
    end
```

```
else
    if socDiff > 0
        if socDiff < 0.02   % To get a smother mode shift
            maxEMTorqueLocal = 50*(socDiff)*maxEMTorque;
        else
            maxEMTorqueLocal = maxEMTorque;
        end
    else
        maxEMTorqueLocal = 0; % Do not use the em at all
    end

    if connected == 0  % If the ICE not is connected to the wheels
        if gear == 0  % Selected gear
            Tem = 0;
            Tice = 0;
        else
            if Tdemand < maxEMTorqueLocal
                Tem = Tdemand;
                Tice = 0;
            else
                Tem = maxEMTorqueLocal;
                Tice = 0;
            end
        end
    else
        if gear == 0 % Will most likely never be in this mode...
            Tem = 0;
            Tice = 0;
        else
            if Tdemand < 0.7*maxEMTorqueLocal % 70% of Tem to save
                battery
                Tem = Tdemand*1/uem;
                Tice = 0;
            else
                Tem = 0.7*maxEMTorqueLocal;
                Tice = Tdemand - Tem*uem;
            end
        end
    end
end
```

$$T_{ice,req} = \frac{1}{0.1s+1}T_{ice}$$
$$T_{em,req} = \frac{1}{0.1s+1}T_{em}$$
$$T_{brake,req} = \frac{1}{0.1s+1}T_{brake}$$

3.12.3 Vehicle

Fuel tank

$$m_f = \int -\max\{0, \dot{m}_f\}dt$$
$$m_{f,reduction} = \int \max\{0, \dot{m}_f\}dt$$

ICE
Local controller:

$$\begin{array}{ll} \text{if} & \omega_{ice} < \omega_{idle} + \omega_{idle,offset} \\ & T_{idle} = T_{ice}(\omega_{ice} = \omega_{idle}) \\ \text{else} & \\ & T_{idle} \\ \text{end} & \\ T_{ice,req} & = \max\{T_{ice,req,controller}, T_{idle}\} \end{array}$$

$$\begin{aligned} \text{ice}_{controller} = p_{mf} V_d &= \underbrace{\left(T_{ice,req} \frac{16}{SB^2 N_{cyl}} + p_{me0,f} + p_{me0,g} \right)}_{p_{mf}\eta_{term}} \frac{N_{cyl}\pi SB^2}{4\eta_{term}} \\ p_{me0,f} &= k_1 \left(k_2 + k_3 S^2 \omega_{ice,sens}{}^2 \right) \Pi \sqrt{\frac{k_4}{B}} \end{aligned}$$

engine:

$$\begin{aligned} \dot{m}_f &= \text{ice}_{controller} \frac{\omega_{ice}}{4\pi H} \\ T_{ice} &= \left(\text{ice}_{controller} \frac{4\eta_{term}}{N_{cyl}\pi SB^2} - p_{me0,f} - p_{me0,g} \right) N_{cyl} \frac{SB^2}{16} \end{aligned}$$

Buffer
Local controller:

$$SoC_{est} = \int -\frac{I_{batt,sens}}{\text{BattCap\_d} \cdot 3600} dt$$

Buffer:

$$\begin{aligned} SoC &= \int -\frac{I_{batt}}{\text{BattCap} \cdot 3600} dt \\ U_{oc} &= f(SoC) \\ U_{batt} &= \max\{0, nU_{oc} - nR_i I_{batt}\} \end{aligned}$$

Electric Machine
Local controller:

$$U_{limit,upper} = \min\{U_{batt,sens}, \left(P_{max} + \frac{k_{a,c}k_{i,c}}{R_{em,c}} \omega_{em,sens}{}^2 \right) \frac{R_{em,c}}{k_{i,c}\omega_{em,sens}}\} \quad (3.1)$$

$$U_{limit,lower} = \max\{-U_{batt,sens}, -\left(P_{max} + \frac{k_{a,c}k_{i,c}}{R_{em,c}} \omega_{em,sens}{}^2 \right) \frac{R_{em,c}}{k_{i,c}\omega_{em,sens}}\} \quad (3.2)$$

$$U_{temp} = T_{em,req} \frac{R_{em,c}}{k_{a,c}} + \frac{k_{a,c}}{R_{em,c}} \omega_{em,sens} \quad (3.3)$$

$$U_{em,control} = \begin{cases} U_{limit,lower}, & U_{temp} < U_{limit,lower} \\ U_{temp}, & U_{limit,lower} \le U_{temp} < U_{limit,upper} \\ U_{limit,upper}, & U_{temp} \ge U_{limit,upper} \end{cases} \quad (3.4)$$

Remark: There is a bug in the `CAPSim` implementation of equations (3.1) and (3.2) and the equations above are the corrected ones.

Electric machine:

$$
\begin{aligned}
U_{filt} &= \frac{1}{0.1s+1} U_{em,control} \\
T_{em} &= \frac{U_{filt} k_i}{R_{em}} - \frac{\omega_{em} k_a k_i}{R_{em}} \\
I_{batt} &= \underbrace{\frac{T_{em}}{k_i}}_{I_{em}} \frac{|U_{filt}|}{\max\{0.1, U_{batt}\}} \\
|I_{batt}| &\leq 300\text{A}
\end{aligned}
\tag{3.5}
$$

Clutch

$$T_{clutch} = T_{ice}, \quad \text{clutch} == 1 \ \& \ \Delta\omega \text{ small}$$

Mechanical join

$$
\begin{aligned}
T_{mj} &= T_{em} u_{em} + T_{clutch} \\
J_{mj} &= J_{em} u_{em}^2 + J_{clutch} + J_{ice}
\end{aligned}
$$

Gear box

$$
\begin{aligned}
J_{gb} &= f(\text{gear}) \\
u_{gb} &= f(\text{gear}) \\
\eta_{gb} &= \begin{cases} \eta_{pos}, & T_{mj} > T_{gb,loss} \\ \eta_{neg}, & T_{mj} \leq T_{gb,loss} \end{cases} \\
T_{gb,loss} &= f(\text{gear}, \omega_{ice}) \\
T_{gb} &= (T_{mj} - T_{gb,loss})\, \eta_{gb} u_{gb} \\
J_{gb,tot} &= (J_{gb} + J_{mj,tot})\, u_{gb}^2
\end{aligned}
$$

Chassis

$$
\begin{aligned}
m_{tot} &= m_{init} - m_{fueltankreduction} \\
T_{drag} &= \frac{1}{2} \rho C_d A_f \omega_w^2 r_w^3 \\
T_{roll} &= \begin{cases} mgC_r r_w, & 1000\omega_w > mgC_r r_w \\ 1000\omega_w, & -mgC_r r_w \leq 1000\omega_w < mgC_r r_w \\ -mgC_r r_w, & 1000\omega_w \leq -mgC_r r_w \end{cases} \\
T_{slope} &= m_{tot} g \sin\alpha \\
T_{brake} &= T_{brake,control} \\
T_{net} &= T_{gb} u_{final} - T_{drag} - T_{brake} - T_{roll} - T_{slope} \\
\dot{\omega}_w &= \frac{T_{net}}{J_{gb,tot} u_{final}^2 + m_{tot} r_w^2} \\
v &= \omega_w r_w \\
\Theta &= \int \omega_w dt \\
s &= \Theta r_w \\
\omega_{gb} &= \omega_w u_{final}
\end{aligned}
$$

3.13 Simulation results

When simulating the vehicle described above, using the `FTP75` driving cycle, the fuel consumption is 103 $^{l}/_{100km}$. The variations in different torques and SoC are shown in Figure 3.2 and the speed profile in Figure 3.3. When using `EUDC`, the fuel consumption is 68 $^{l}/_{100km}$. Both these cycles, especially `FTP75`, include lots of start and stops. However, the fuel consumption is probably too high in the implemented model, but as already mentioned the model parameters are not necessarily realistic. For the principle comparative studies performed in this work this is not a problem.

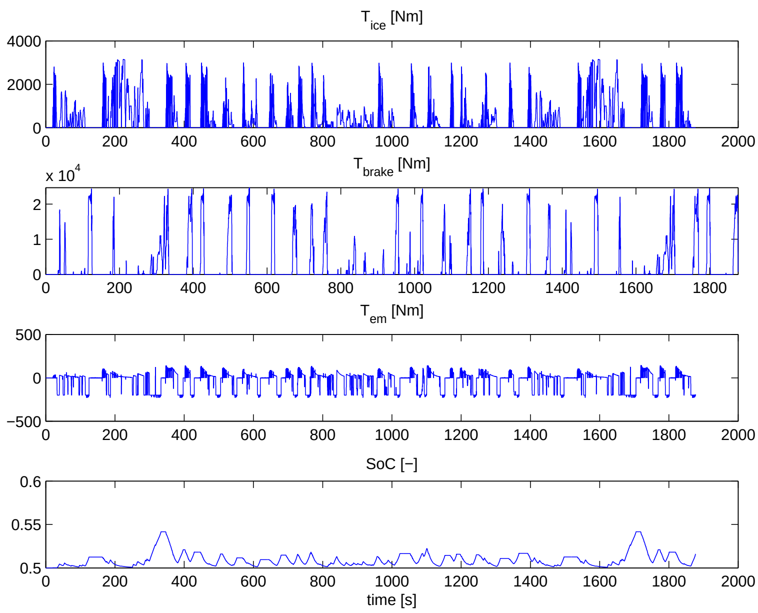

Figure 3.2: Different torques and SoC when the truck is driving `FTP75` with the parameters and models described in this chapter.

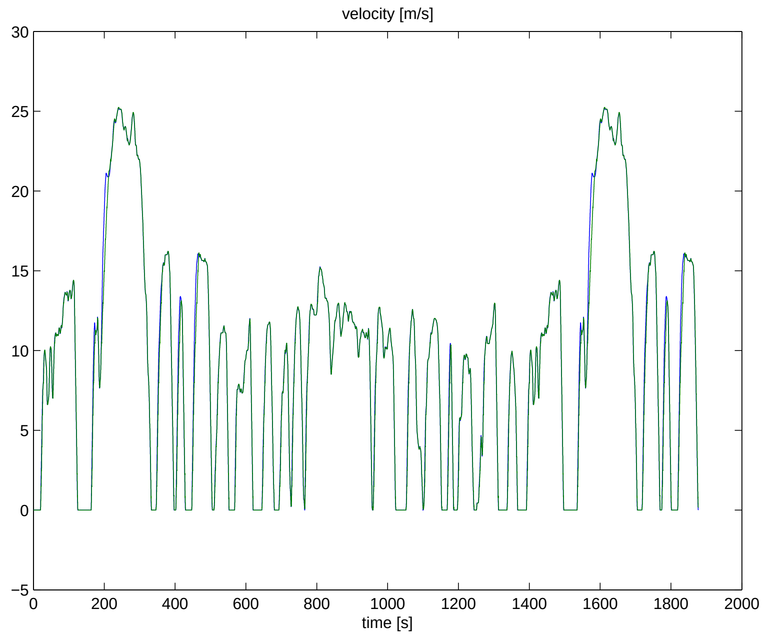

Figure 3.3: The velocity of the truck when it tries to follow the drive cycle, `FTP75`.

Chapter 4

Diagnosis of the truck based on models for correct behavior

The purpose of diagnosis is to be able to detect faults and decide which components that are broken. This functionality is included in the platform according to Figure 1.1. The platform is general and can incorporate any diagnosis system. Here three diagnosis systems for the truck are derived, evaluated and compared with a twofold purpose. The first purpose is to demonstrate the feasibility of the platform and its use. The second purpose is a bit more ambitious. It aims at vehicle level diagnosis, and even though it is applied to a specific truck model, the scope is generic for parallel hybrids. This means that several diagnosis systems are developed, implemented and analyzed. The different diagnostic systems use different sensor configurations to analyze the implication of this. Further, it has been chosen to be general in the sense that no information about the faults' impact on the supervised component is used in the diagnosis systems in the platform.

4.1 Components to monitor

The components that are monitored are the electrical components in the vehicle, i.e. the electrical machine, power electronics, and the battery. The power electronics is not modeled as a separate component in the vehicle, but is part of the electric machine where U_{batt} is transformed to U_{em}. A fault in the power electronics is assumed to result in that $U_{em} \neq U_{em,control}$ (see equation (3.5)). Except from that, it is not known how the fault will affect U_{em}.

In some cases it is advantageous to detect what part of a component that is broken. To represent this both the inner resistance and the torque constant could change values due to faults that occur. The component is broken if at least one of these parameters has drifted from its nominal value.

The fault in the battery that is to be monitored is a short circuit. This will affect the voltage of the battery since the number of cells that is used in the battery is reduced. In the diagnosis system, the knowledge that the number of

Sensors	Additive noise power	Proportional noise power
$\omega_{ice,sens}$	0.01	0
$U_{batt,sens}$	0.01	0
$I_{batt,sens}$	0.01	0
$I_{em,sens}$	0.01	0
$U_{em,sens}$	0.01	0
$\omega_{em,sens}$	0.01	0
$\omega_{gb,sens}$	0.01	0
$\omega_{w,sens}$	0.01	0

Table 4.1: The intensity of the noise added to the sensor signals in the model. The noise is updated at 80 Hz.

cells that are used never can be more than there are physical cells, is never used.

There is a possibility to add the functionality to monitor the sensors used in the model. This means that a fault in the sensor should be detected and isolated by the diagnosis system. The sensors that are monitored differ from each system and are stated in the description of the diagnosis systems.

4.2 Induced faults

The induced faults in the model affects the same parameters that are assumed to be changed when a fault occurs. These are:

- The torque/current constant, k_i, and the resistance, R_{em}, in the electric machine.
- The voltage applied to the electric machine is not the requested voltage. When this occurs the power electronics is faulty.
- The number of cells in the battery are reduced to represent a short circuit.

4.3 Sensor noise and sample frequency

In order to get a more realistic simulation, noise is added to the sensor signals. The intensity of the added noise to the signals can be both proportional to the magnitude of the signal and added as a constant. The block used in `Simulink` to produce the noise signals is called `Band-Limited White Noise` and the frequency for when to recalculate the noise is set to 80 Hz. The power of the noise added to the different sensors used in the model is given in Table 4.1. The noisy measurement signal y_{noise} is given by the sensor signal y added with the noise:

$$y_{noise} = y + y \cdot \texttt{Proportional noise power} + \texttt{Additive noise power}$$

The sensors measures the signals at 80 Hz and the diagnosis system is updated at the same frequency, while the vehicle model is simulated using a variable step length solver.

4.4 Sensor configurations

One way to investigate the properties of the diagnosis systems regarding detectability and isolability is to use structural analysis. The basic idea is to tabulate the unknowns each equation in the model is dependent on. Also all faults and known variables are to be tabulated. The result from the analysis depends on what sensors that are used in the diagnosis system. It is important to state that the results from this analysis does not give any information about e.g. robustness to noise and model uncertainties. To investigate these issues the diagnosis system needs to be implemented.

The outcome from the analysis are examples of how to combine parts of the model to achieve over determined parts that are used in the diagnosis. Some of these parts are used to produce tests that alarms if the over determined equation system not has any solution. The tests are designed to achieve as good isolability as possible.

Using structural analysis it is also possible to detect what sensors that are needed to detect and isolate the specified faults. The idea is to find over determined parts of the model using known signals such as control and sensor signals. These parts are sensitive for different faults in the system and sets of overdetermined parts are created that achieve as high isolability of the faults as possible. There may be several sets leaving the user to choose which one to use. Furthermore it is no guarantee that every fault is detectable in all driving modes. A fault in the electric machine that only influences the system when there is a voltage applied on the component will for example not be detected if the vehicle do not use the electric machine.

The three diagnosis systems use different sensor configurations. One system has a sensor configuration resulting a diagnosis system with small tests. This system uses a torque sensor in the electric machine that is not assumed to be available in the second system. The last system is based on a sensor configuration using as few sensors as possible to achieve full isolability structurally. Each system is described below. There are several terms specific for diagnosis and for an explanation of these [6] is recommended.

4.4.1 Diagnosis system 1

In the first diagnosis system the following sensors are used:

- $T_{em,sens}$ - torque from the electric machine, not common in production
- $\omega_{gb,sens}$ - outgoing speed of gear box
- $I_{batt,sens}$ - current to battery
- $I_{em,sens}$ - current in electric machine
- $U_{em,sens}$ - voltage of electric machine

The structural table for diagnosis system 1 is shown in Figure 4.1. In the figure the unknowns are in the left part, the faults in the middle, and the known variables at the right. On the vertical axis all relations in the model are given, that in this example are 48 relations. The theory used in this report to produce these figures as well as the structural analysis is described in [7] and [3].

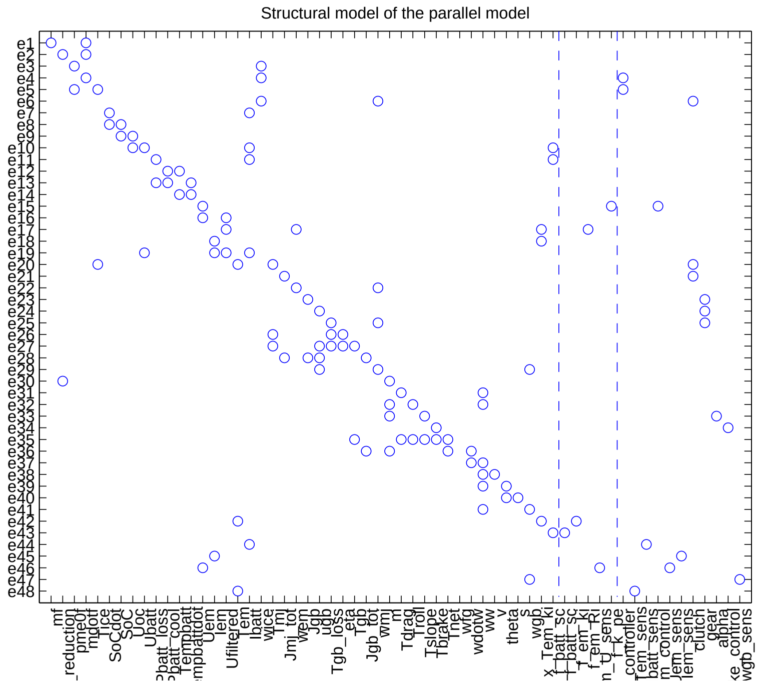

Figure 4.1: The structural relationships for the truck model and sensor placement given in Section 4.5.1 referring to diagnosis system 1.

The number of equations each test consists of is shown in Figure 4.2. The number of over determined parts (or tests) increases when more sensors are used, since there are more possibilities to create such sets of equations. Observe that not all tests are implemented, the figure only gives an idea of how many equations that are included in each test. In general, many equations in a test leads to that higher computational effort is required, but also larger sensitivity for noise and model uncertainties. Some of the equations are trivial relationships such as sensor equations. An example of a sensor equation is

$$\omega_{sens} = \omega + f_{\omega,sens}$$

where the sensor can break down. If this not is the case, $f_{\omega,sens}$ is not included in the expression.

4.4.2 Diagnosis system 2

In this diagnosis system the sensors used are similar to the previous system. The only difference is that the torque sensor in the electric machine is replaced by a rotational speed sensor for the engine. This results in that the following sensors are used in this system:

- $\omega_{ice,sens}$ - speed of engine
- $\omega_{gb,sens}$ - outgoing speed of gear box

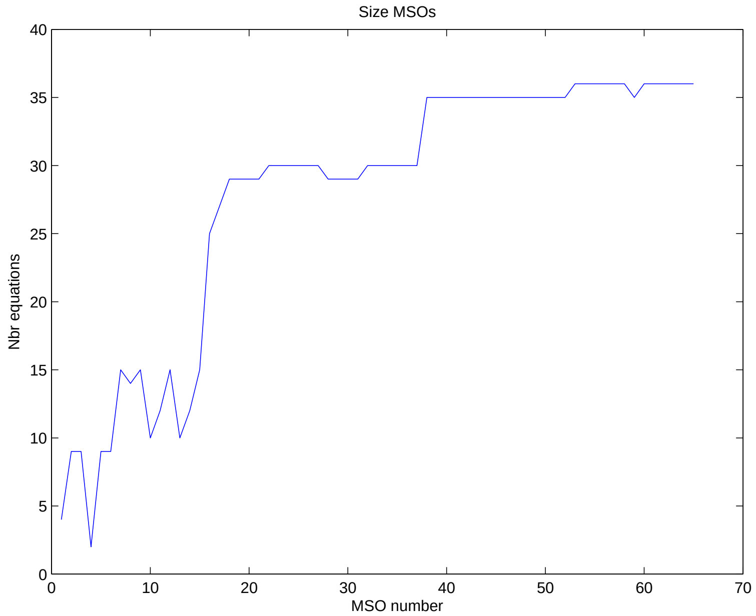

Figure 4.2: The number of equations used in each test for the sensor placement in diagnosis system 1. As seen in the figure there are a a few tests based on few equations, but most of the tests are based on a larger part of the entire model that is 48 equations.

- $I_{batt,sens}$ - current to battery
- $I_{em,sens}$ - current to electric machine
- $U_{em,sens}$ - voltage of battery

The structural table for diagnosis system 2 is shown in Figure 4.3 and the number of equations included in the MSOs are shown in Figure 4.4. There is no major difference in the size of the tests in diagnosis systems 1 and 2.

4.4.3 Diagnosis system 3

In this diagnosis system a minimal set of sensors are used to still achieve full isolability in the detection of the faults. To achieve this the structural analysis is used to propose sensor configurations that fulfills the demand on full isolability of the faults in the system. This shows that two sensors are sufficient to detect the faults in the model. To be able to also isolate faults from all sensors in the system three sensors are required. There are several sensor configurations that manage this, but in this specific system the following sensor configuration is used:

- $\omega_{gb,sens}$ - outgoing speed of gearbox
- $U_{batt,sens,a}$ - voltage of battery

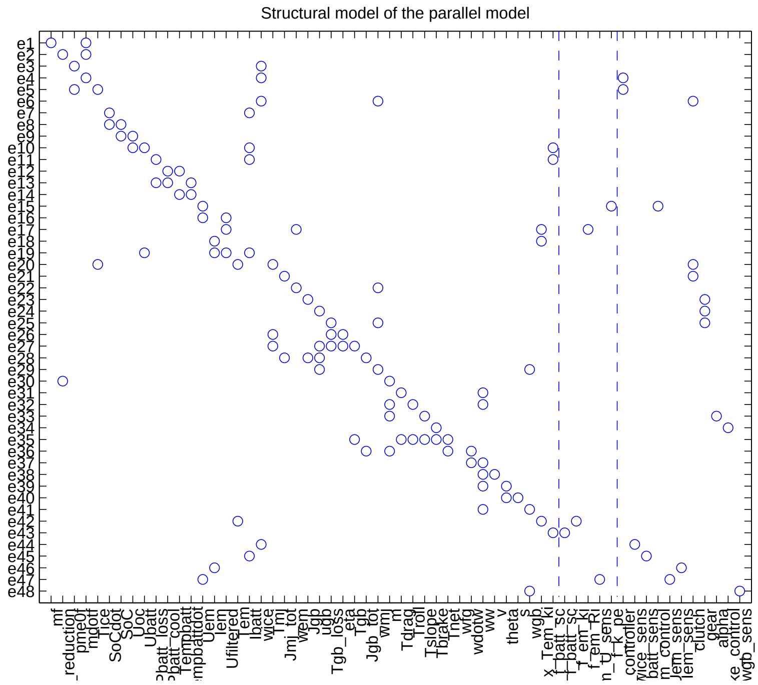

Figure 4.3: The structural relationships for the truck model and sensor placement given in Section 4.5.2 referring to diagnosis system 2.

- $U_{batt,sens,b}$ - voltage of battery

In this case an additional voltage sensor is needed to fully isolate a fault in the original voltage sensor.

The structural table for diagnosis system 3 is shown in Figure 4.5 and the number of equations included in the MSOs are shown in Figure 4.6. There is not as many over determined parts using this sensor configuration compared to diagnosis systems 1 and 2, due to that fewer sensors are used in this system. The number of tests based on smaller parts of the model are also fewer in this system.

4.5 Design and evaluation

In this section the implementation of the diagnosis systems are described. The results from the simulations are also presented.

The faults listed in Section 4.2 are induced one by one at 400 seconds in the simulation. The parameter or signal that is modified is halved except from when a fault in a sensor occurs. If this is the case an offset is added to the actual value with a magnitude specified in each diagnosis system.

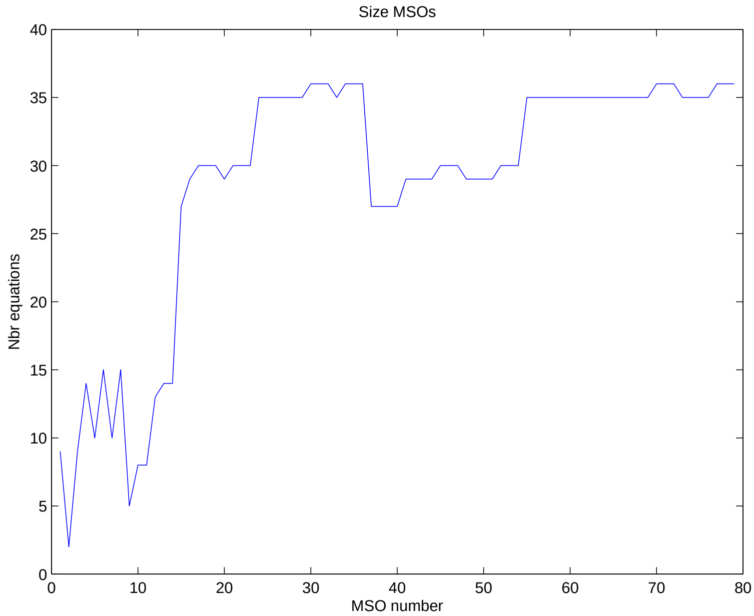

Figure 4.4: The number of equations used in each test for the sensor placement in diagnosis system 2.

4.5.1 Diagnosis system 1

The first diagnosis system is in some sense unrealistic since T_{em} is assumed to be a measured signal, and production vehicles are not equipped with torque sensors.

In addition to the faults listed in Section 4.2, the voltage sensor in the electric machine is monitored in this system. When a fault is induced in the sensor model, 20 V is added to the sensor output compared to the correct value.

There are four tests that use subsets of the sensor and model equations in order to only react to some of the faults. This makes it possible to isolate the fault using a decision structure. This structure contains information about what faults that may cause a test to react. From the structural analysis full isolability is achieved using these four tests that are described later in this section. In some cases the tests do not react on the faults as they would according to the structural analysis. One reason for this is that the fault cancels out when the test is constructed. This is investigated by implementing the tests in the model. Different diagnosis systems have different performance that is possible to analyze by implementing the systems. In some cases this can also be done analytically [8]. The tests that are included in this system are described below.

Test 1

The first test in this diagnosis system uses parts of the electric machine model to detect a fault in this component. The following sensor is used in Test 1:

- $U_{em,sens}$

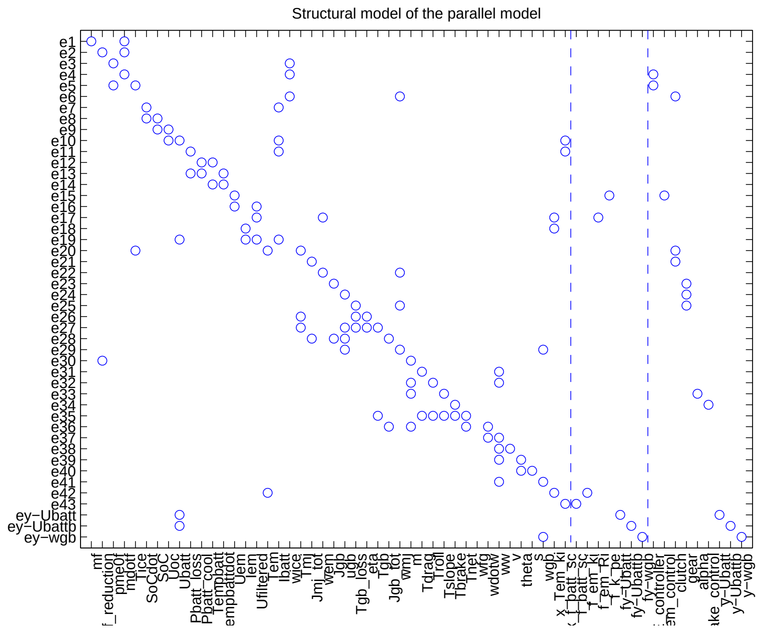

Figure 4.5: The structural relationships for the truck model and sensor placement given in Section 4.5.3 referring to diagnosis system 3.

and the faults the test react on according to the structural analysis are:

- f_{pe}
- $f_{em,U,sens}$

The residual generator is given below and will ideally also react on these faults.

$$r = U_{em,sens} - U_{em} \tag{4.1}$$

where

$$U_{em} = U_{em,control}$$

Test 2

Test 2 also includes parts of the model of the electric machine, but this test detects both faults in the machine. To estimate the angular velocity of the machine, the gear ratio is used as well as the angular velocity sensor in the gear box. The sensors used in the test are:

- $T_{em,sens}$
- $\omega_{gb,sens}$

and the faults the test react on according to the structural analysis are:

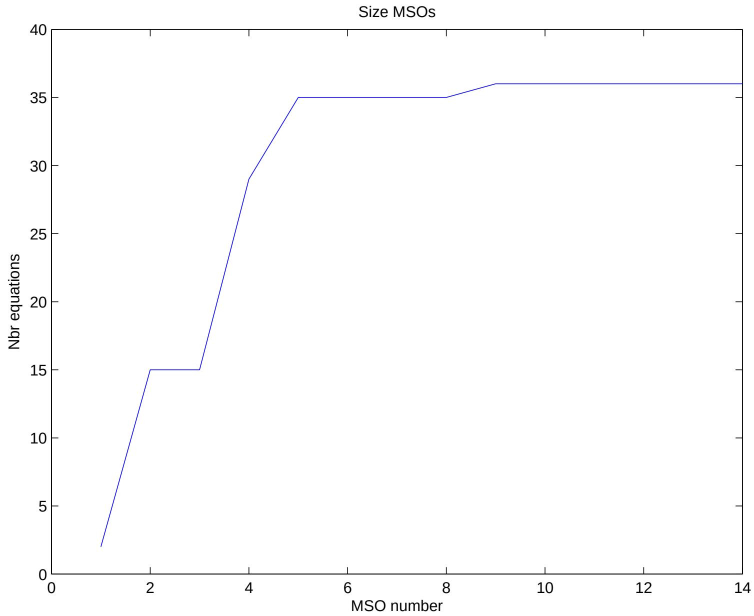

Figure 4.6: The number of equations used in each test for the sensor placement in diagnosis system 3. There is not as many MSOs that include few equations as in the previous diagnosis systems described. There are also less minimal structural overdetermined parts in the system since there are less sensors used.

- $f_{em,ki}$
- $f_{em,R}$
- f_{pe}

The residual generator is:

$$r = T_{em,sens} - T_{em}$$

where

$$\begin{aligned}
T_{em} &= \frac{U_{filt} k_i}{R_{em}} - \frac{\omega_{em} k_a k_i}{R_{em}} \\
U_{filt} &= \frac{1}{0.1s + 1} U_{em} \\
U_{em} &= U_{em,control} \\
\omega_{em} &= u_{em} \omega_{mj} \\
\omega_{mj} &= u_{gb} \omega_{gb} \\
u_{gb} &= f(\text{gear}) \\
\omega_{gb} &= \omega_{gb,sens}
\end{aligned}$$

Test 3

Test 3 monitors the power electronics and parts of the electric machine using the following sensors:

- $I_{batt,sens}$
- $I_{em,sens}$
- $\omega_{gb,sens}$

and the faults the test react on according to the structural analysis are:

- $f_{em,R}$
- f_{sc}

The residual generator is:

$$r = I_{batt} - I_{batt,sens}$$

where

$$\begin{aligned}
I_{batt} &= I_{em}\frac{|U_{filt}|}{U_{batt}} \\
U_{filt} &= I_{em}R_{em} + \omega_{em}k_a \\
\omega_{em} &= u_{em}\omega_{mj} \\
\omega_{mj} &= u_{gb}\omega_{gb} \\
u_{gb} &= f(\text{gear}) \\
U_{batt} &= nU_{oc} - nR_iI_{batt} \\
U_{oc} &= f(SoC) \\
SoC &= \int S\dot{o}C \; dt \\
S\dot{o}C &= -\frac{I_{batt}}{\text{battCap} \cdot 3600} \\
I_{em} &= I_{em,sens} \\
\omega_{gb} &= \omega_{gb,sens}
\end{aligned}$$

Test 4

The sensors used in Test 4 are:

- $I_{batt,sens}$
- $U_{em,sens}$
- $T_{em,sens}$

and the faults the test reacts on according to the structural analysis are:

- $f_{em,ki}$
- f_{sc}
- $f_{U,em,sens}$

The residual generator is:

$$r = T_{em}|U_{filt}| - I_{batt}U_{batt}k_i$$

where

$$\begin{aligned}
U_{filt} &= \frac{1}{0.1s+1}U_{em} \\
U_{batt} &= nU_{oc} - nR_iI_{batt} \\
U_{oc} &= f(SoC) \\
SoC &= \int SoC\ dt \\
S\dot{o}C &= -\frac{I_{batt}}{battCap \cdot 3600} \\
I_{batt} &= I_{batt,sens} \\
U_{em} &= U_{em,sens} \\
T_{em} &= T_{em,sens}
\end{aligned}$$

This test is not used if $|I_{batt,sens}| > 290$ A. The reason for this is that the calculation for U_{filt} not is valid when $|I_{batt}| > 300$ A, see the model description for the electric machine in Section 2.6.2. The limit in this test is set with some margin to the maximum current of the battery to avoid problems related to sensor noise.

Thresholds in tests

To handle the noise in the residuals, r, post processing is necessary. This noise is due to the noise in sensor signals, but also, to some extent, model uncertainties. One way of processing the residuals is to construct a signal, s, that has a negative expectation value in a fault free case and positive when a fault has occurred. The trend of the cumulative sum, g, of s will then contain information about the status of the monitored system. The test quantity is calculated below

$$\begin{aligned}
s(t) &= |r(t)| - \nu \\
g(t+1) &= g(t) + s(t)
\end{aligned}$$

introducing the test quantity, $T(t)$

$$T(t) = g(t) - \min_{0 \le i < t} g(i)$$

where ν is the offset to ensure that $E\{s(t)\} < 0$ in the fault free case. The system alarms if $T > J$, where J is a threshold that is a design parameter.

Table 4.2: Offsets and thresholds in the tests to diagnosis system 1 used in CUSUM.

	offset	threshold
$T1$	2	100
$T2$	10	100
$T3$	10	100
$T4$	400	5000

The algorithm described above is called CUSUM [10] and is well known. In Table 4.2 the parameters used in the algorithm is shown for this diagnosis system. The parameters used in CUSUM are designed to avoid false alarms. The first step is to study the residual in the fault free case that the test is based on. The offset parameter ν is set to be large enough to ensure that the demand $E\{s(t)\} < 0$ is fulfilled. If ν is set to a large value the fault detection will take longer time and smaller faults in the vehicle will maybe not be detected. This has to be considered in the design of the parameters in the algorithm. After the offset is decided, T is studied. The threshold, J, is set to achieve a system that does not alarm when the system is fault free.

Tests are carried out where faults are induced in the system. If the diagnosis system lacks in performance, the parameters described above are modified. The fault free case is then analyzed again till a wanted performance is achieved.

Decision structure

Full isolability is achieved as long as the tests reacts on the faults as given in the structural analysis. This can be seen in the decision structure in Table 4.3. The table shows what tests that will react when a specific fault occurs. If one test alarms it could be any of the faults the test reacts on. If two or more tests alarm, the possible faults to explain this behavior are the faults that get all tests that are triggered to alarm, i.e. the intersection of the sets since only single faults are handled. The columns in the table differs, which indicates that the faults are separable from each other. This is since different tests reacts on different faults. A more formal description of this is carried out in [7].

Full isolability in this systems means that there is one unique set of tests that alarm when a fault occurs. Therefore the diagnosis system is able to decide which fault that has occurred and not only detect that the system is faulty.

Table 4.3: Decision structure for diagnosis system 1.

	$f_{em,ki}$	$f_{em,R}$	f_{pe}	$f_{batt,sc}$	$f_{em,U,sens}$
$T1$			X		X
$T2$	X	X	X		
$T3$		X		X	
$T4$	X			X	X

Results from simulations

Figures 4.7 - 4.9 shows the results in a fault free case. No test alarms and no test is close to the normalized threshold (see Figure 4.9). The magnitude of the residuals vary a lot between the different tests. One reason for this is that the sensor noises are amplified different in different tests, but the main reason is that the magnitude of the two signals that are compared in the residual differ between the tests.

The results from the simulations where the faults are induced one by one at 400 seconds are presented in Figures 4.10 - 4.24. All tests are detected and isolated in a few seconds, see the figures for details. This diagnosis system clearly operates properly.

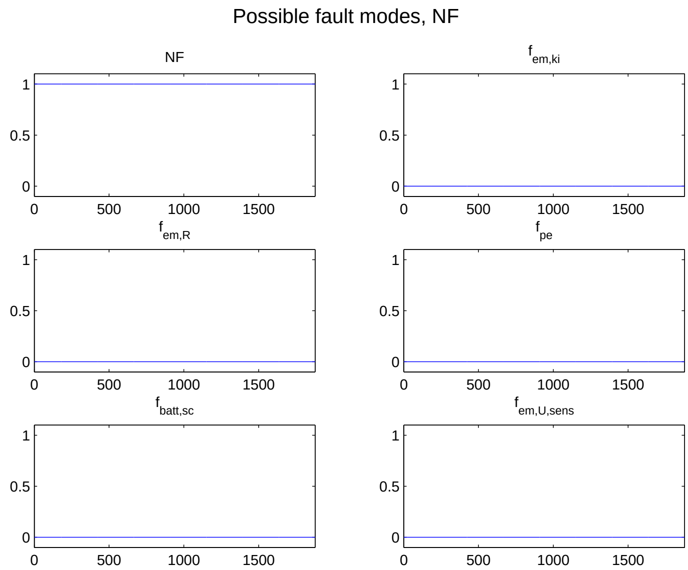

Figure 4.7: No other possibility than that the system is fault free is possible in the fault free case.

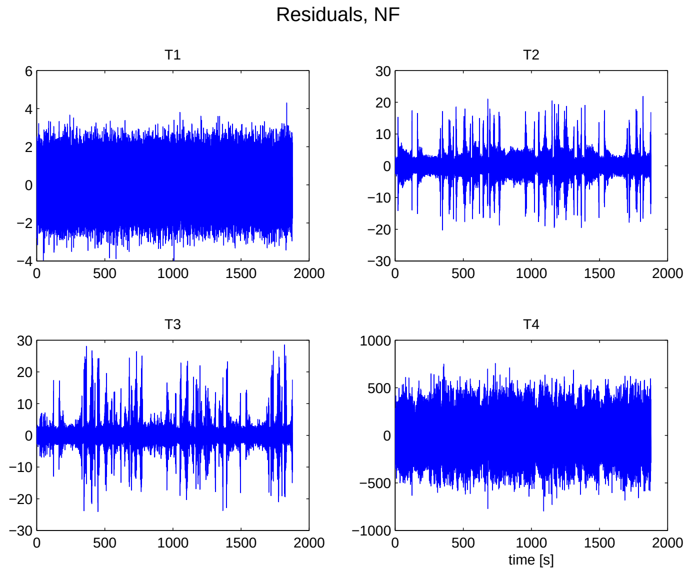

Figure 4.8: The residuals in a fault free case. The magnitude of the residuals in test 2 and 3 varies depending on the operating point of the system. The residuals are largest during transients. One reason for this is that the delays in the model and diagnosis system are not fully synchronized. This results in that different signals are compared from different times in the residual generator.

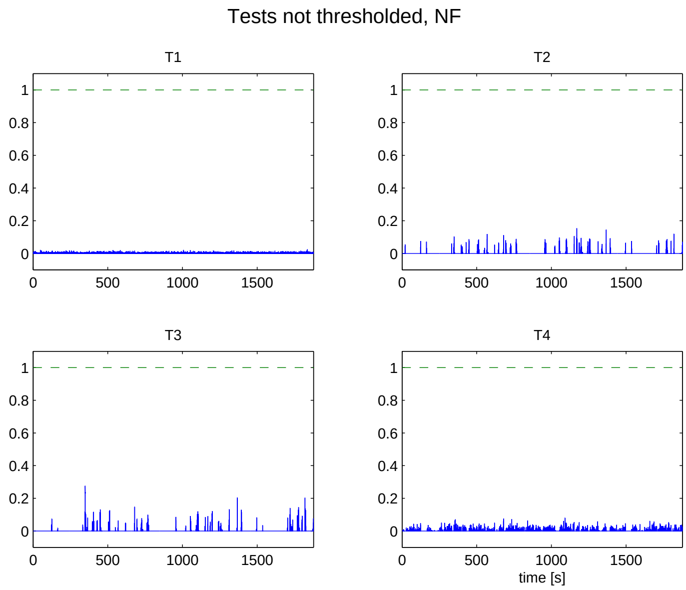

Figure 4.9: Normalized tests before threshold in a fault free case. The test alarms if the test quantity is larger than 1.

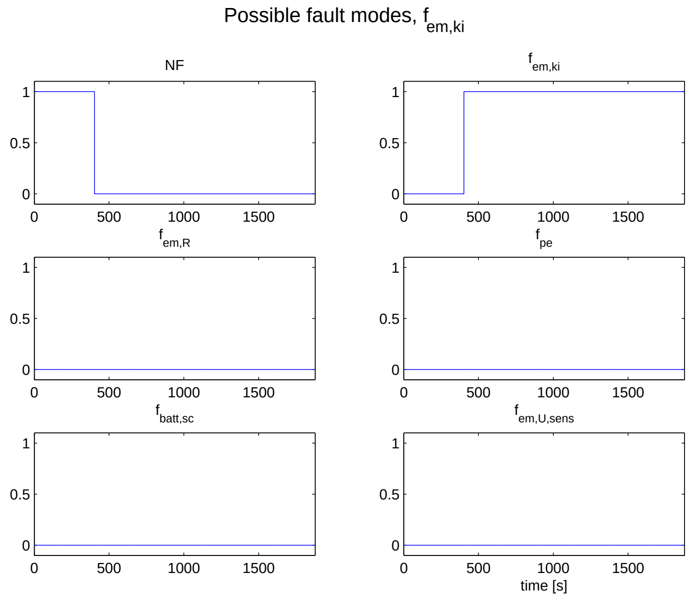

Figure 4.10: Possible fault modes when k_i is halved at 400 seconds. The fault is detected and isolated in 2 seconds after it occurs.

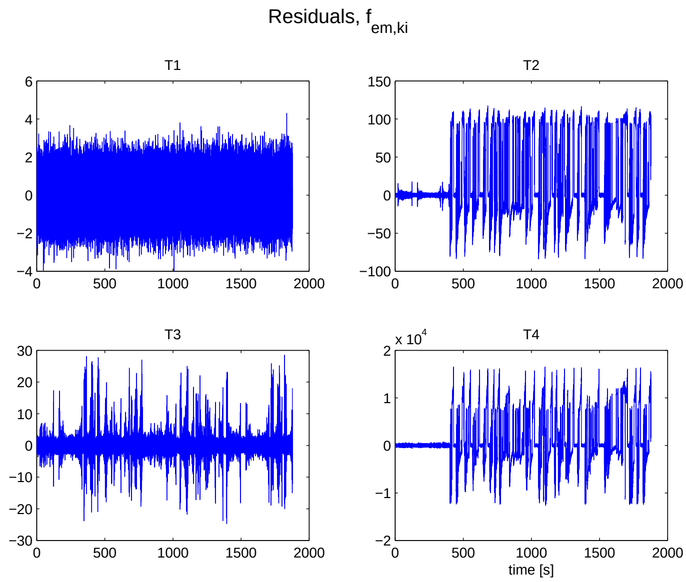

Figure 4.11: The residuals when k_i is halved at 400 seconds. In this simulation it is clear that the residuals to tests 2 and 4 reacts on the change in the parameter k_i.

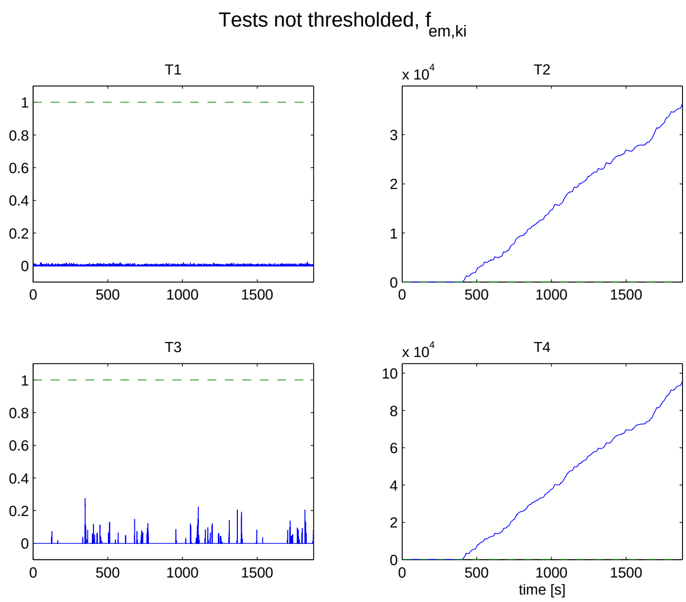

Figure 4.12: Normalized tests before threshold when k_i is halved at 400 seconds. Tests 2 and 4 reacts fast and there is no doubt that the system is faulty.

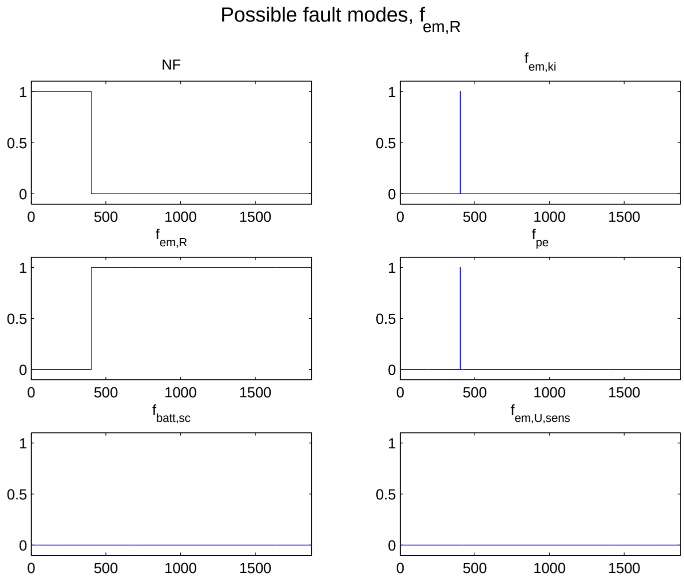

Figure 4.13: Possible fault modes when the resistance in the electric machine is halved at 400 seconds. The fault is detected and isolated in 3 seconds after it occurs. The faults f_{pe} and $f_{em,ki}$ are possible faults for a short period. The reason for this is that when only test 2 reacts on the fault, possible fault modes are $f_{em,R}$, $f_{em,ki}$ or f_{pe} (see Table 4.3). When also test 3 reacts,the only explanation is that there is a fault in the resistance in the electric machine.

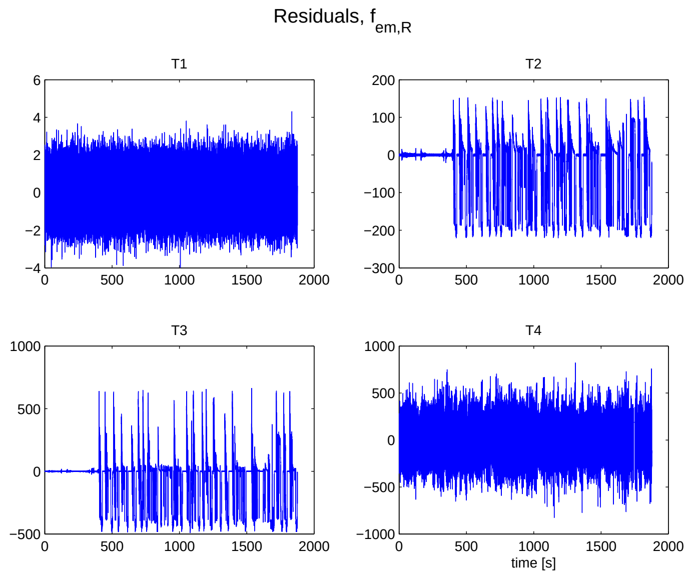

Figure 4.14: The residuals when R_{em} is halved at 400 seconds. It is clearly shown that the residuals for tests 2 and 3 changes magnitude when the fault is induced in the system.

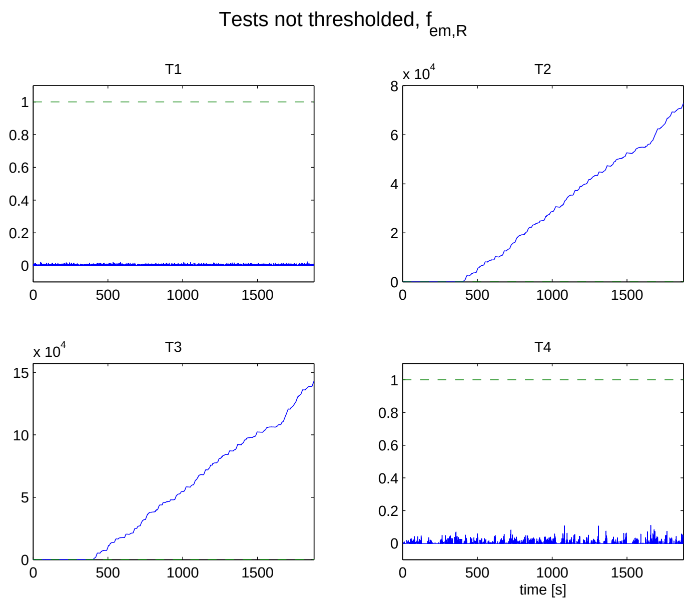

Figure 4.15: Normalized tests before threshold when R_{em} is halved at 400 seconds. There is no doubt which tests that reacts in this case.

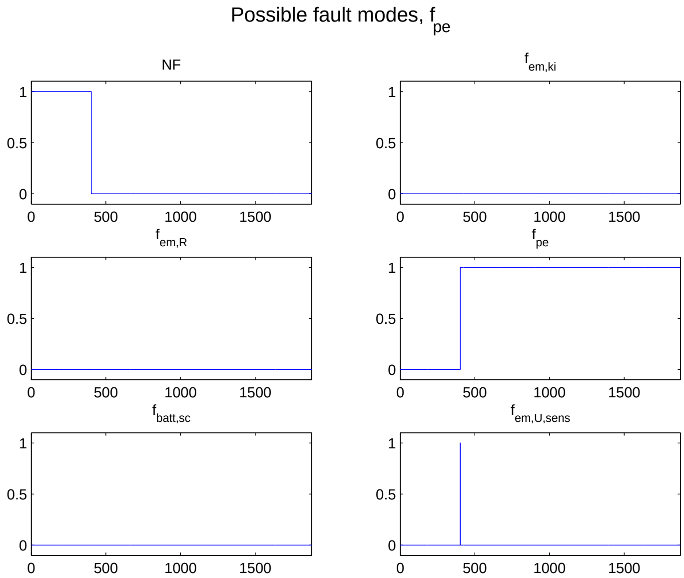

Figure 4.16: Possible fault modes when k_{pe} is halved at 400 seconds. This fault is also detected and isolated in 3 seconds after it occurs.

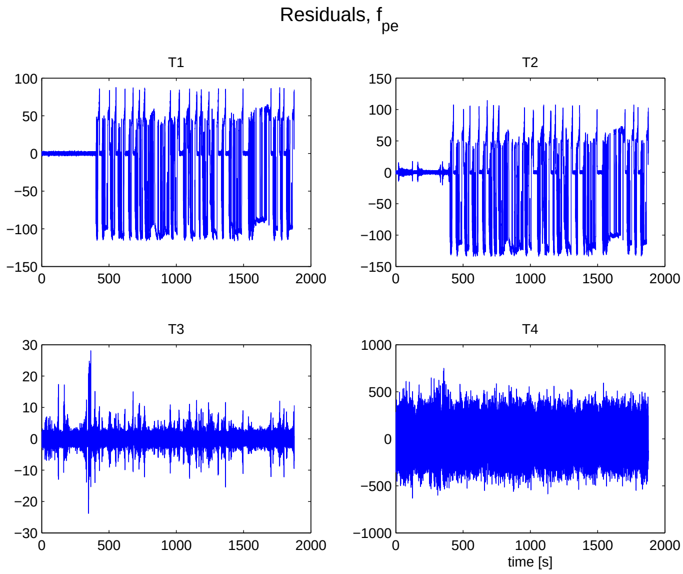

Figure 4.17: The residuals when k_{pe} is halved at 400 seconds. It is obvious that test 1 and 2 should react, which they also do (see Figure 4.18).

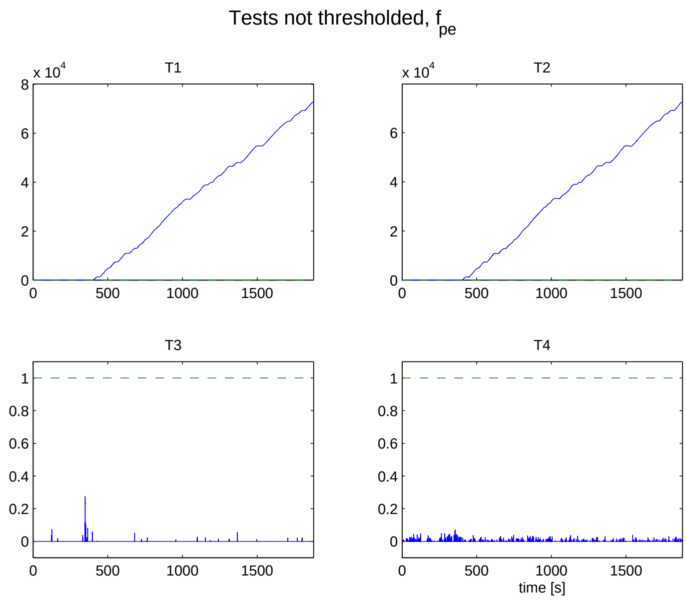

Figure 4.18: Normalized tests before threshold when k_{pe} is halved at 400 seconds. The magnitudes of the test quantities in the tests that react on the fault are large.

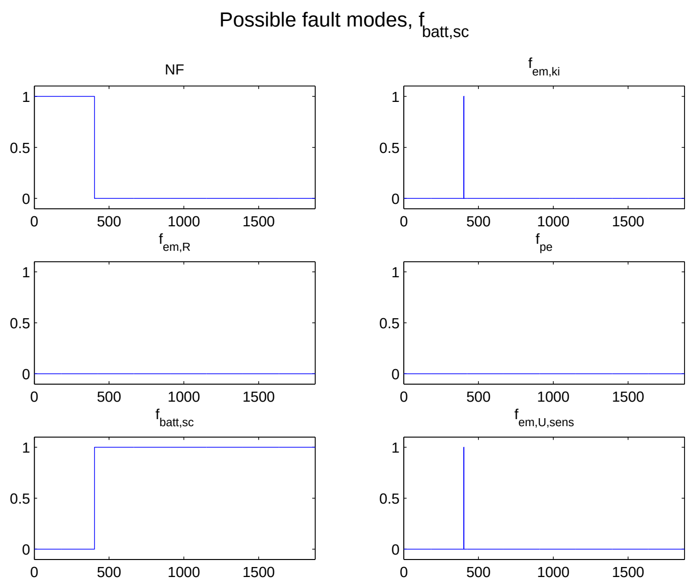

Figure 4.19: Possible fault modes detected when a short circuit in the battery occurs at 400 seconds. Half of the cells are only used after this time. The fault is detected and isolated in 3 seconds.

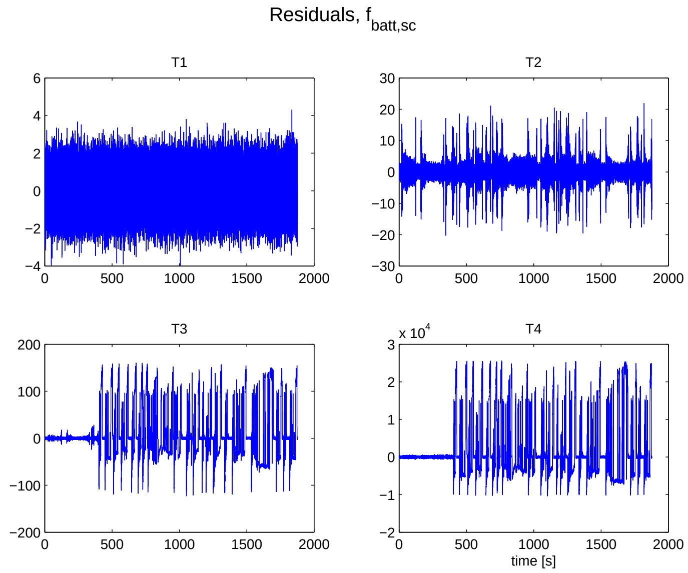

Figure 4.20: The residuals when k_{sc} is halved at 400 seconds.

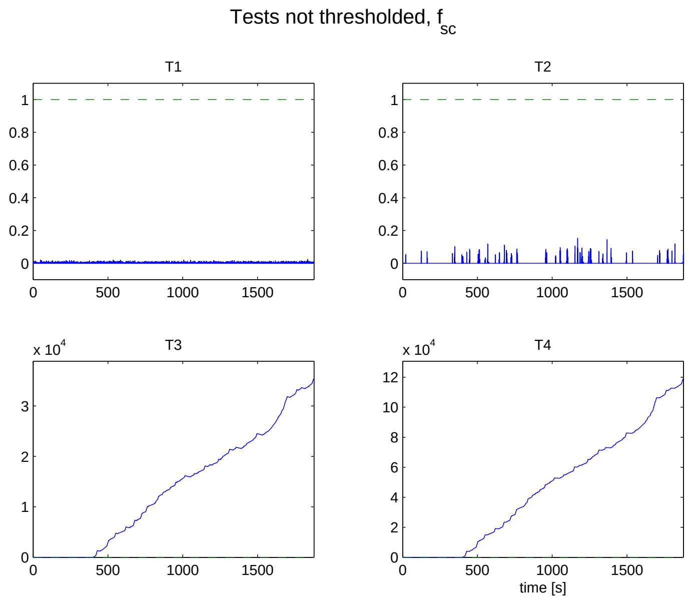

Figure 4.21: Normalized tests before threshold when k_{sc} is halved at 400 seconds. The tests that are supposed to react, do so with large margins.

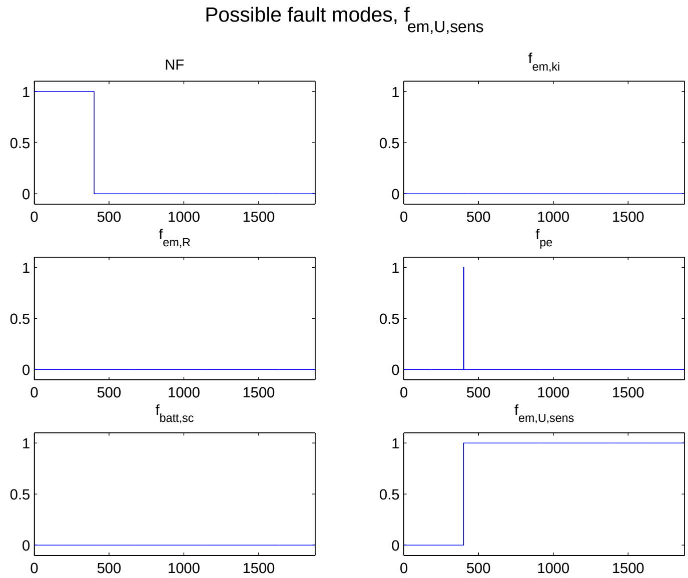

Figure 4.22: The faults detected when an offset of 20 V is added to the sensor signal for the voltage in the electric machine compared to the true value at 400 seconds. The fault is detected in less than one tenth of a second and isolated in 3 seconds.

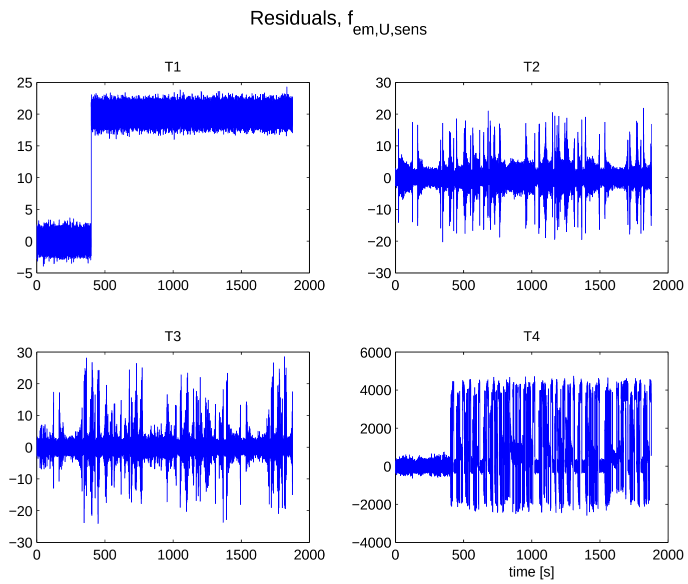

Figure 4.23: The residuals when $U_{em,sens}$ breaks down at 400 seconds. The residual used in the first test has mean value of approximately 20 V. This is due to that the offset in the sensor has a direct impact on the residual given in equation (4.1).

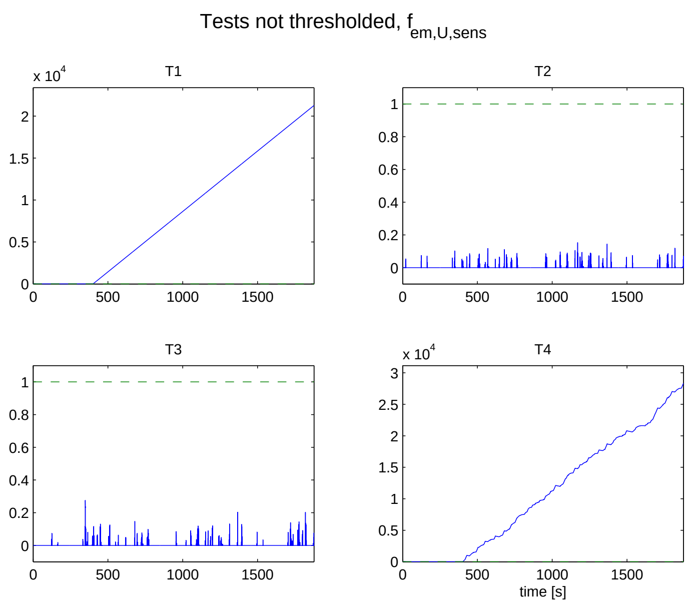

Figure 4.24: Normalized tests before threshold when $U_{em,sens}$ breaks down at 400 seconds. Both tests 1 and 4 reacts as expected.

4.5.2 Diagnosis system 2

The faults that are to be detected are the same as in the previous test, which means that $U_{em,sens}$ are monitored in addition to the parameters given in Section 4.1. In this diagnosis system the offset noise power for ω_{gb} is halved to 0.05, compare to Table 4.1 due to that this diagnosis system is more noise sensitive.

In this system there are four tests, which achieves full isolability of the faults. In this diagnosis system a model for the clutch is needed during gear shift and take off, because in some of the tests, sensors placed on both sides of the clutch are used. Modeling the clutch accurately is a well known problem and the model of the clutch used in the vehicle model is assumed to not be known in the diagnosis systems. The model of the clutch used in the diagnosis system is only valid when the clutch is fully disengaged or engaged, and is assumed to be an ideal component. When there is a slip in the clutch the model is not valid and the test quantities in the diagnosis systems are not updated.

The four tests implemented in this diagnosis system are described below.

Test 1

The first test is the same as the first test in Diagnosis system 1 and the voltage sensor in the electric machine is used in this test:

- $U_{em,sens}$

and the faults the test reacts on according to the structural analysis are:

- $f_{k,pe}$
- $f_{em,U,sens}$

The residual generator is:

$$r = U_{em} - U_{em,sens}$$

where

$$U_{em} = U_{em,control}$$

Test 2

The second test consists of parts of the models of the electric machine, battery and gear box. The sensors used are:

- $I_{batt,sens}$
- $I_{em,sens}$
- $\omega_{gb,sens}$

and the faults the test reacts on according to the structural analysis are:

- $f_{batt,sc}$
- $f_{em,R}$

The residual generator is:

$$r = I_{batt,sens} - I_{batt}$$

where

$$\begin{aligned}
I_{batt} &= I_{em} \frac{|U_{filt}|}{nU_{oc} - nR_i I_{batt}} \\
U_{filt} &= I_{em} R_{em} + \omega_{em} k_a \\
\omega_{em} &= u_{em} \omega_{mj} \\
\omega_{mj} &= u_{gb} \omega_{gb} \\
u_{gb} &= f(\text{gear}) \\
U_{oc} &= f(SoC) \\
SoC &= \int \dot{SoC}\, dt \\
\dot{SoC} &= -\frac{I_{batt}}{BattCap \cdot 3600} \\
I_{batt} &= I_{batt,sens} \\
I_{em} &= I_{em,sens} \\
\omega_{gb} &= \omega_{gb,sens}
\end{aligned}$$

This expression contains an integration to estimate SoC. In most cases it is troublesome to integrate signals, since a small bias in the model causes drift in the integrated signal. In this case the state of charge only is used to calculate U_{oc}, that is modeled to be constant for large variations in SoC (see Figure 2.4). In the diagnosis systems that not are updated at all times, drift is though a problem if longer driving cycles are used. This is not the case with the actual parameter configuration of the vehicle and `FTP75`. This issue is not handled in the platform at this stage.

Test 3

Test 3 uses most of the model of the vehicle and the following sensors are used in the test:

- $\omega_{ice,sens}$
- $\omega_{gb,sens}$

and the faults the test reacts on according to the structural analysis are:

- $f_{k,pe}$
- $f_{em,R}$
- $f_{em,ki}$

The residual generator is:

$$\begin{aligned}\tilde{r} \quad &= \underbrace{\frac{1}{u_{em}}\left(J_{gb,tot} + \frac{1}{u_{final}^2} m r_w^2\right)}_{a}\dot{\omega}_{gb} + \\ &+ \underbrace{\frac{1}{u_{em}u_{final}}\left(T_{drag} + T_{roll} + T_{brake}\right) + \frac{u_{gb}\eta_{gb}}{u_{em}}\left(T_{gb,loss} - T_{ice}\right) - \left(u_{gb}\eta_{gb}\right) T_{em}}_{b}\end{aligned} \tag{4.2}$$

where

$$\begin{aligned}
u_{gb} &= f(\text{gear}) \\
\dot{m}_f &= \text{ice}_{controller}\frac{\omega_{ice}}{4\pi H} \\
m_{f,reduction} &= \int \dot{m}_f \; dt \\
m &= m_{init} - m_{f,reduction} \\
\eta_{gb} &= \begin{cases} \eta_{pos}, & T_{mj} > T_{gb,loss} \\ \eta_{neg}, & T_{mj} \leq T_{gb,loss} \end{cases} \\
T_{mj} &= u_{em}T_{em} + T_{clutch} \\
T_{clutch} &= T_{ice}, \text{ when clutch engaged} \\
J_{gb,tot} &= \left(J_{gb} + J_{mj,tot}\right) u_{gb}^2 \\
J_{gb} &= f(\text{gear}) \\
J_{mj,tot} &= J_{em}u_{em}^2 + J_{clutch} + J_{ice} \\
T_{drag} &= \frac{1}{2}\rho C_d A_f {\omega^2}_w {r^3}_w \\
T_{roll} &= \begin{cases} mgC_r r_w, & 1000\omega_w > mgC_r r_w \\ 1000\omega_w, & -mgC_r r_w \leq 1000\omega_w < mgC_r r_w \\ -mgC_r r_w, & 1000\omega_w \leq -mgC_r r_w \end{cases} \\
T_{brake} &= T_{brake,control} \\
T_{gb,loss} &= f(\text{gear}, \omega_{ice}) \\
\omega_w &= \frac{\omega_{gb}}{u_{final}} \\
T_{ice} &= \left(\text{ice}_{controller}\frac{4\eta_{term}}{N_{cyl}\pi S B^2} - p_{me0,f} - p_{me0,g}\right) N_{cyl}\frac{SB^2}{16} \\
p_{me0,f} &= k_1\left(k_2 + k_3 S^2 {\omega_{ice}}^2\right)\Pi\sqrt{\frac{k_4}{B}} \\
T_{slope} &= mg\sin\alpha \\
T_{em} &= \frac{U_{filt}k_i}{R_{em}} - \frac{\omega_{em}k_a k_i}{R_{em}} \\
U_{filt} &= \frac{1}{0.1s+1}U_{em} \\
U_{em} &= U_{em,control} \\
\omega_{ice} &= \omega_{ice,sens} \\
\omega_{gb} &= \omega_{gb,sens}
\end{aligned}$$

In the expression for $\tilde{r}$ above, $\dot{\omega}_{gb}$ is needed. The signal ω_{gb} is a sensor signal including noise and it is thereby unwanted to differentiate this signal. One solution to the problem is to integrate equation (4.2). The problem doing this is that a small modeling fault results in drift in the integrated signal. If a in the residual generator is constant it is possible to do a variable transformation at the same time as the residual is filtered (see [9]). The benefit of this is that $\dot{\omega}_{gb}$ not has to be estimated. In the filter, that will be presented later, a has to be constant. The mass of the vehicle is changing when fuel is consumed, but this is assumed to not influence the solution since the change is slow. The inertia is dependent on selected gear, but this is not a problem since the model only is valid when the clutch is engaged, and therefor the inertia is constant when the model is valid.

The variable transformation is given below, note that ω is used as a transformation variable and not an angular velocity:

$$r = \frac{\alpha}{p+\alpha} \underbrace{\left(a\dot{\omega}_{gb} + b\right)}_{\tilde{r}} \tag{4.3}$$

using the state

$$\omega = r - \alpha a \omega_{gb} \tag{4.4}$$

we obtain

$$\dot{\omega} = \alpha\left(r - \alpha a \omega_{gb}\right) + \alpha b \tag{4.5}$$

$$r = \omega + \alpha a \omega_{gb} \tag{4.6}$$

The filter parameter α can be modified in the design of the low pass filters of the residuals, where a smaller α filters the signal more. The disadvantage with this is that if there is an error in the initialization of the signal, it will take longer time before the error has faded out. On the other hand, it might be difficult to detect faults on a less filtered signal.

When no gear is selected or the clutch is disengaged, the residual is not updated in this test. Instead the last value of the residual is sent to the “fault detection” block. This is not necessary, since the algorithm that uses the residual not is updated when the residual not is estimated correctly.

When the model in the diagnosis system is getting valid, ω is reinitialized. This is needed since the state probably has drifted during the time the model was invalid. When ω is initialized, it is always assumed that the vehicle is fault free. The expression for ω when the model is getting valid at time t_0 is therefore

$$\omega(t_0) = -\alpha a \omega_{gb}(t_0) \tag{4.7}$$

Using this unfiltered expression is not good since it is very sensitive to noise in $\omega_{gb,sens}$ at one sample. This can lead to a significant offset in the residual before this impact has faded out. To make this problem smaller, the right hand side of equation (4.7) is low pass filtered with different time constants. The results from this is shown in Figure 4.25. It is possible to filter the sensor signal since $\omega_{gb} = \omega_{gb,sens}$ is valid even when the model not is valid. This solution of the problem reduces the above described problem. The signal is still noisy, but if it is more low pass filtered there are significant errors during transients. The chosen time constant for all tests in this platform using this filter is 10 seconds.

One way to possibly increase the performance in the initialization is to use an observer instead of the filter of $-\alpha a \omega_{gb}(t_0)$. It is also possible to use the last valid value of the residual when reinitializing ω. This makes it possible to detect faults even when there are e.g. many start and stops, and gear changes making the diagnosis system valid only for short periods.

To further reduce the problems when reinitializing the state in the residual generator, the CUSUM algorithm is not updated for the first 10 seconds after the model is getting valid. This is to get the error in the initialization of ω a chance to fade out. The delay is in this case the same as the time constant of the filter used.

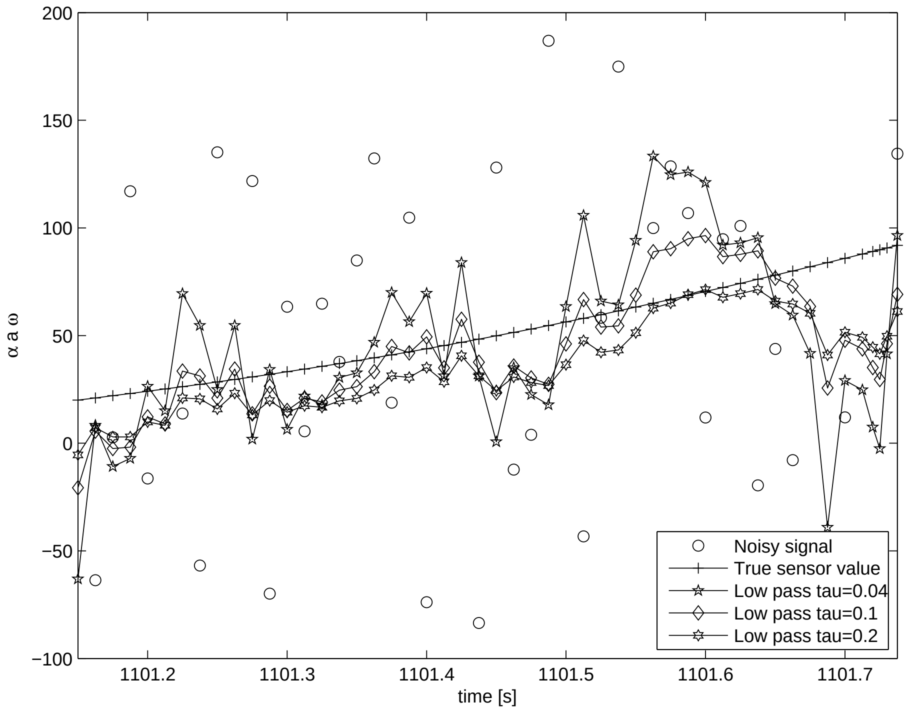

Figure 4.25: Three time constants in the filter of $\alpha a \omega$ is used. Also the unfiltered signal and the sensor signal are included in the figure. As can be seen the filter with low time constant is more noisy than the others, while the filter with high time constant has a time delay during the transient. In the diagnosis systems the time constant is set to 0.1 seconds.

Test 4

This test is similar to the previous test. The same consistency relation is used in both tests, but some of the variables are estimated in different ways. For example is the battery model included in this test, but not in test 3. The following four sensors are used in this test:

- $\omega_{ice,sens}$

- $\omega_{gb,sens}$
- $U_{em,sens}$
- $I_{batt,sens}$

and the faults the test reacts on according to the structural analysis are:

- $f_{em,ki}$
- $f_{batt,sc}$
- $f_{U,em,sens}$

The residual generator is:

$$\begin{aligned}\tilde{r} \quad &= \underbrace{\frac{1}{u_{em}}\left(J_{gb,tot} + \frac{1}{u_{final}^2} m r_w^2\right)}_{a}\dot{\omega}_{gb} + \\ &+ \underbrace{\frac{1}{u_{em}u_{final}}\left(T_{drag} + T_{roll} + T_{brake}\right) + \frac{u_{gb}\eta_{gb}}{u_{em}}\left(T_{gb,loss} - T_{ice}\right) - \left(u_{gb}\eta_{gb}\right) T_{em}}_{b}\end{aligned}$$

where

$$
\begin{aligned}
u_{gb} &= f(\text{gear}) \\
\dot{m}_f &= \text{ice}_{controller} \frac{\omega_{ice}}{4\pi H} \\
m_{f,reduction} &= \int \dot{m}_f \; dt \\
m &= m_{init} - m_{f,reduction} \\
\eta_{gb} &= \begin{cases} \eta_{pos}, & T_{mj} > T_{gb,loss} \\ \eta_{neg}, & T_{mj} \leq T_{gb,loss} \end{cases} \\
T_{mj} &= u_{em} T_{em} + T_{clutch} \\
T_{clutch} &= T_{ice}, \text{ when clutch engaged} \\
J_{gb,tot} &= \left(J_{gb} + J_{mj,tot}\right) u_{gb}^2 \\
J_{gb} &= f(\text{gear}) \\
J_{mj,tot} &= J_{em} u_{em}^2 + J_{clutch} + J_{ice} \\
T_{drag} &= \frac{1}{2} \rho C_d A_f {\omega^2}_w {r^3}_w \\
T_{roll} &= \begin{cases} mgC_r r_w, & 1000\omega_w > mgC_r r_w \\ 1000\omega_w, & -mgC_r r_w \leq 1000\omega_w < mgC_r r_w \\ -mgC_r r_w, & 1000\omega_w \leq -mgC_r r_w \end{cases} \\
T_{brake} &= T_{brake,control} \\
\omega_w &= \frac{\omega_{gb}}{u_{final}} \\
T_{ice} &= \left(\text{ice}_{controller} \frac{4\eta_{term}}{N_{cyl} \pi S B^2} - p_{me0,f} - p_{me0,g} \right) N_{cyl} \frac{S B^2}{16} \\
p_{me0,f} &= k_1 \left(k_2 + k_3 S^2 {\omega_{ice}}^2 \right) \Pi \sqrt{\frac{k_4}{B}} \\
T_{slope} &= mg \sin \alpha \\
T_{em} &= I_{em} k_i \\
I_{em} &= \frac{I_{batt} U_{batt}}{|U_{filt}|} \\
U_{filt} &= \frac{1}{0.1s + 1} U_{em} \\
U_{batt} &= n U_{oc} - n R_i I_{batt} \\
U_{oc} &= f(SoC) \\
SoC &= \int \dot{SoC} \; dt \\
\dot{SoC} &= -\frac{I_{batt}}{\text{BattCap} \cdot 3600} \\
\omega_{ice} &= \omega_{ice,sens} \\
\omega_{gb} &= \omega_{gb,sens} \\
U_{em} &= U_{em,sens} \\
I_{batt} &= I_{batt,sens}
\end{aligned}
$$

When $|U_{filt}| < 1$ V, 1 V is used instead of $|U_{filt}|$ in the expression for I_{em} to avoid division by zero. Therefore, when $|U_{filt}| < 1$ V the test above is not valid and the test is not reacting on faults. This is also the case when the clutch is engaged or no gear is selected. The residual generator starts to operate three seconds after the clutch has been disengaged. When at least one of these cases occurs, the residual and the test is not updated.

The same transformation using ω is used in this test as is described in the previous test.

Thresholds in tests

Table 4.4 shows the parameter values used in the CUSUM algorithm for this system.

Table 4.4: Thresholds in the tests to diagnosis system 2 used in CUSUM.

	offset	threshold	α
$T1$	2	500	
$T2$	8	1000	
$T3$	65	8000	0.1
$T4$	65	8000	0.1

Decision structure

In Table 4.5 the decision structure for this diagnosis system is shown. Full isolability should be possible to achieve in this system.

Table 4.5: Decision structure for diagnosis system 2.

	$f_{em,ki}$	$f_{em,R}$	f_{pe}	$f_{batt,sc}$	$f_{em,U,sens}$
$T1$			X		X
$T2$		X		X	
$T3$	X	X	X		
$T4$	X			X	X

Results from simulations

In this section simulations are carried out and one fault at the time is induced in the model. In all simulations sensor noise is assumed with properties according to Table 4.1 except the offset noise power for $\omega_{gb,sens}$ that is changed to 0.05. The parameter α is set to 0.1 in both tests where it is used, and the offsets and thresholds used in CUSUM are tabulated in Table 4.4.

Figures 4.26 to 4.29 are results from a simulation of a fault free case. As seen in Figure 4.28, the tests are well below the threshold in the fault free case.

The results from the simulations where the faults are induced one by one at 400 seconds in the simulations are presented in Figures 4.30 - 4.44. In general the faults are detected fast, but it takes several hundreds of seconds to isolate the fault. It is test 3 and test 4 that takes time to alarm. The residual generators to these tests include $\dot{\omega}_{gb}$ and are filtered accordingly to equations (4.3) - (4.6).

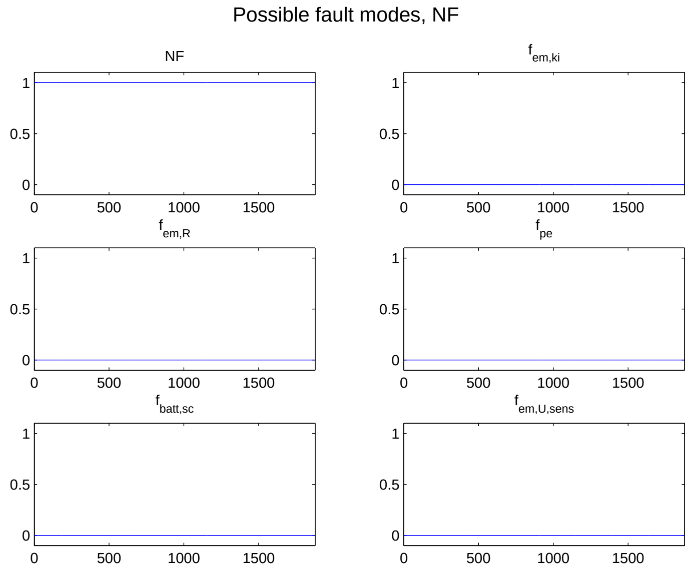

Figure 4.26: No faults are detected in the fault free case.

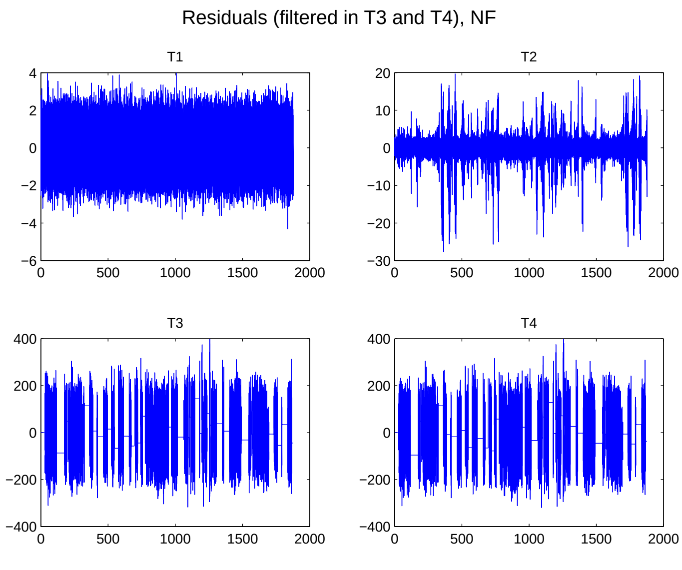

Figure 4.27: The residuals in the fault free case. The residuals used in tests 3 and 4 are not updated at all times. This depends on several conditions explained in each test above. Note that the tests are not being updated the first 10 seconds after the residuals have been valid again.

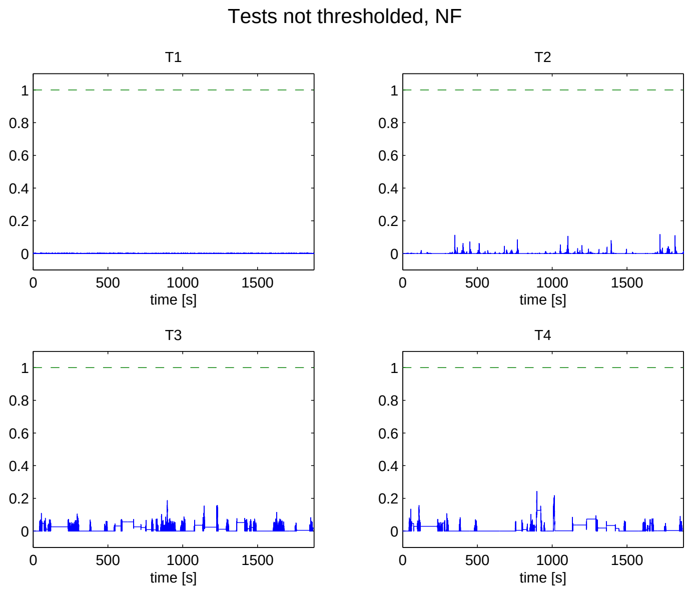

Figure 4.28: Normalized tests before threshold. There is no tendency for false alarms.

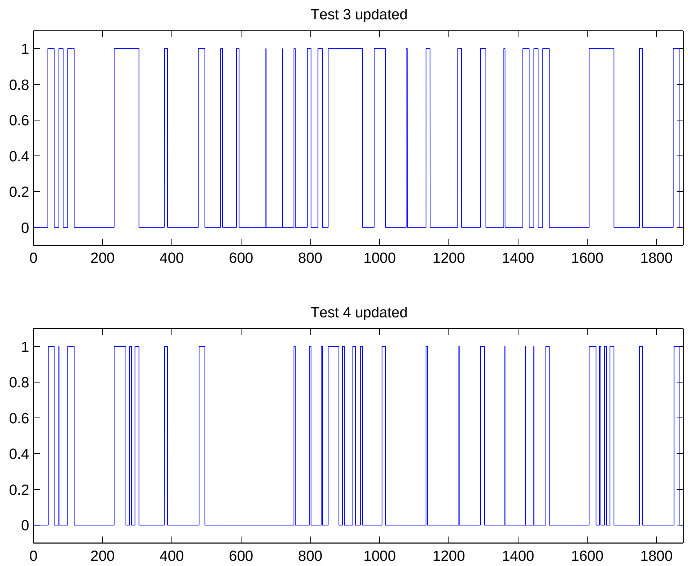

Figure 4.29: This figure includes information about when the tests are updated. When the signal is 1 the test is updated and the conditions for when the tests are updated are stated in each test above. This signal is 0 during the first 10 seconds after the conditions set up in the tests for the system to be valid are fulfilled.

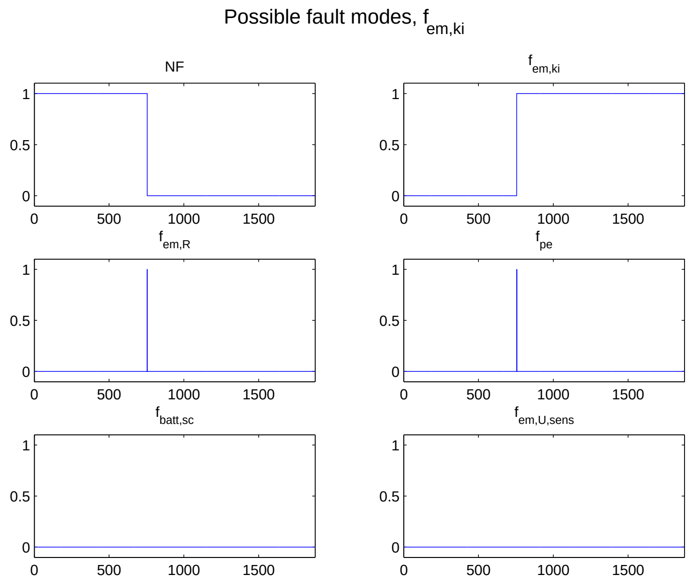

Figure 4.30: The faults detected when k_i is halved at 400 seconds. The fault is detected and isolated, but it takes about 350 seconds to achieve this. One of the reasons for the long detection time is that the tests 3 and 4 that are supposed to react not are updated for long periods, see Figure 4.29.

.

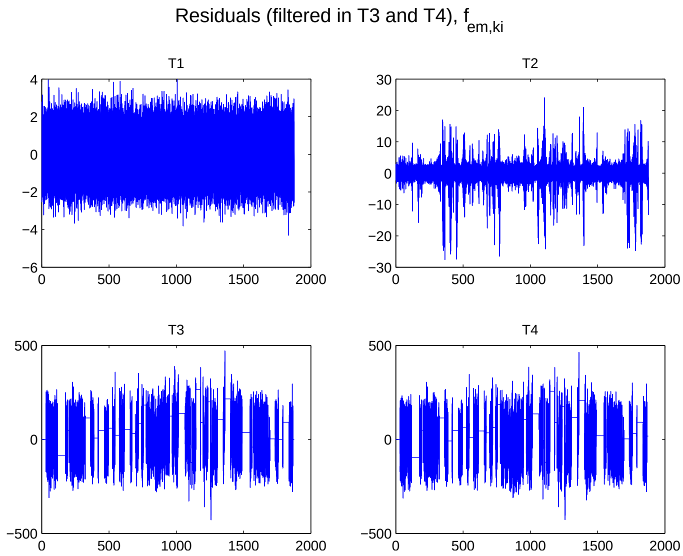

Figure 4.31: The residuals when k_i is halved at 400 seconds. In tests 3 and 4 there are periods where the residual not is updated. This is when the model used in the diagnosis system is not valid and therefore the residual is not updated.

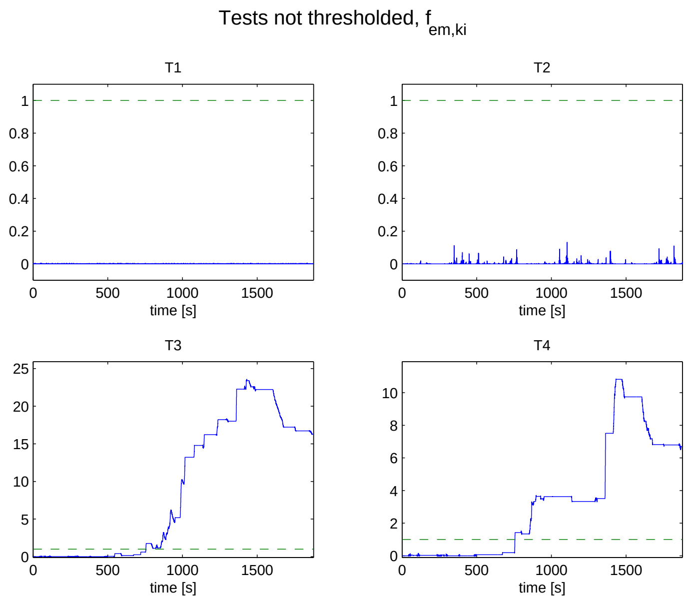

Figure 4.32: Normalized tests before threshold when k_i is halved at 400 seconds. Both tests that react on this fault have the worrying trend of that the test quantity decreases at the end of the simulation.

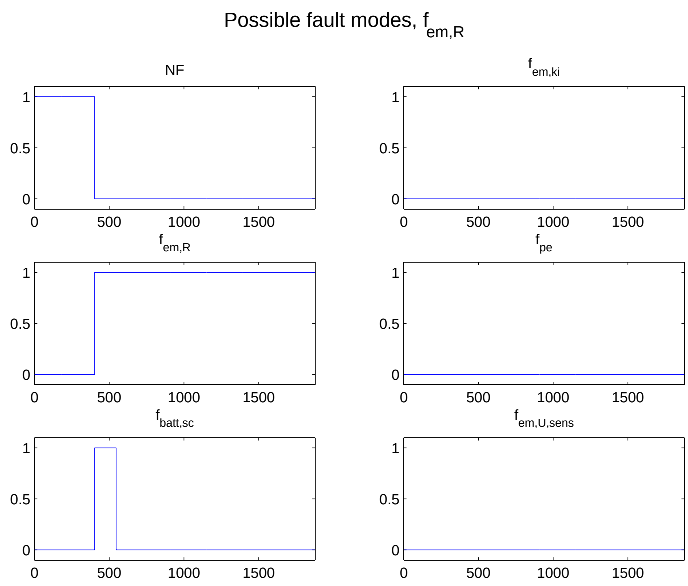

Figure 4.33: The possible faults when R_{em} is halved at 400 seconds. The fault is detected after a few seconds, but it takes about 150 seconds to isolate the fault since this is the time it takes for test 3 to react.

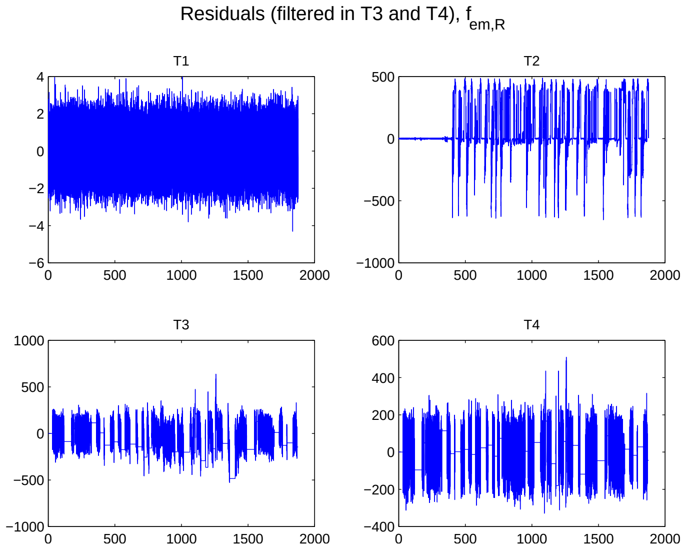

Figure 4.34: The residuals when R_{em} is halved at 400 seconds. It is obvious that the residual in test 2 changes characteristics after 400 seconds when the fault is induced, but it is not in the residual used in test 3.

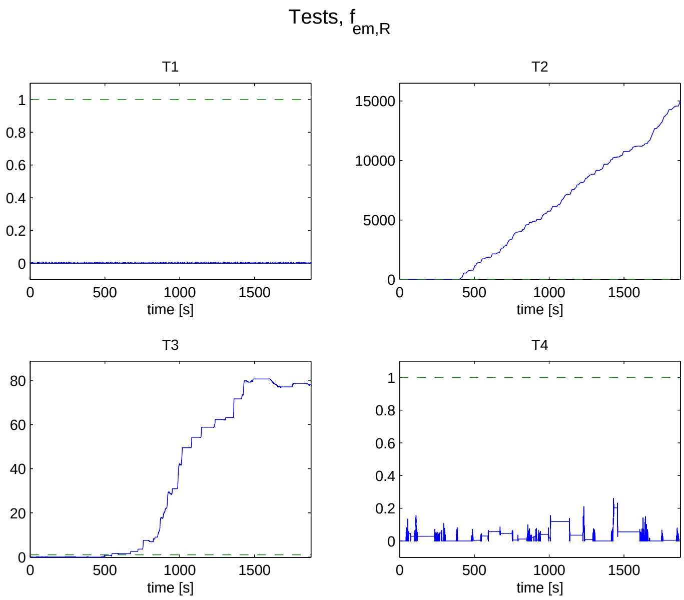

Figure 4.35: Normalized tests before threshold when R_{em} is halved at 400 seconds. The relative magnitude of the test quantity used in test 2 is more than 100 times larger than corresponding value for test 3. One reason for this is that test 3 not is updated at all times, but this does not explain the entire difference.

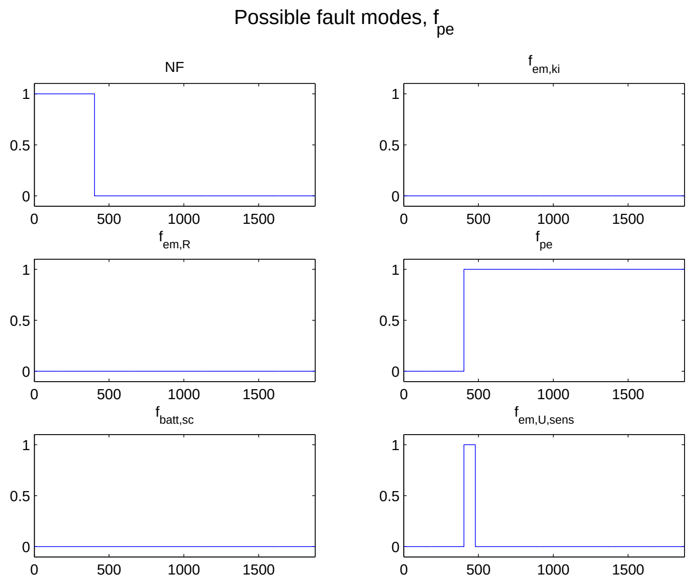

Figure 4.36: Possible fault modes when k_{pe} is halved at 400 seconds. The fault is detected almost immediately after the fault is induced in the model, while it takes about 80 seconds to achieve full isolability. Test 3 reacts in a few seconds after the test is valid.

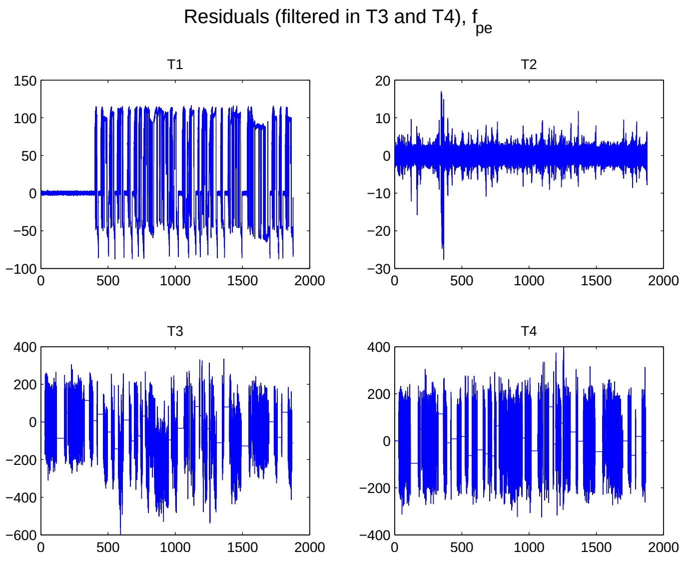

Figure 4.37: The residuals when k_{pe} is halved at 400 seconds. The large variations in the residual used in test 2 that is shown e.g. at about 350 seconds occurs at gearshifts. At these times the test quantity is not updated and therefore it does not affect the performance of the diagnosis system.

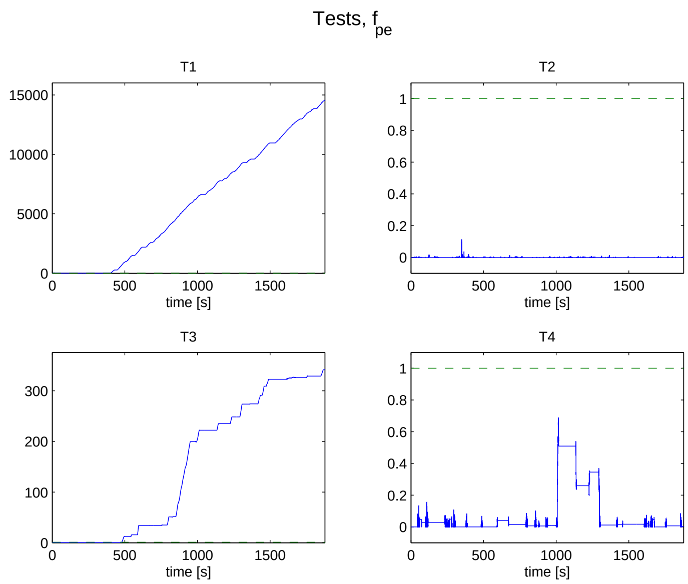

Figure 4.38: Normalized tests before threshold when k_{pe} is halved at 400 seconds. The most interesting in this figure is that test 4 seems to be affected of the fault in the power electronics. This should not be the case since the relationship $U_{em} = U_{em,control}$ not is used in this test. When simulating the system without measurement noise, test 4 is not affected of the fault any more. Therefore one possible explanation to this behavior is that the vehicle is running in a different operating point when the fault has occurred, resulting in that the noise is excited in a different way.

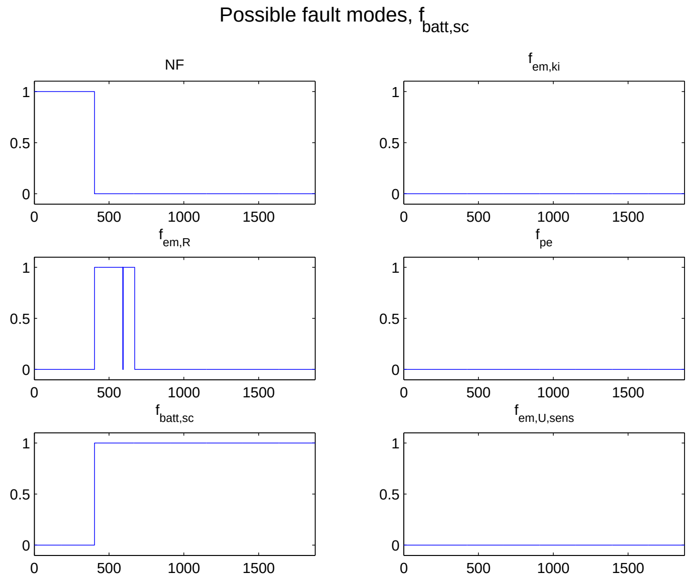

Figure 4.39: Possible fault modes when a short circuit in the battery occurs at 400 seconds resulting in that only half of the cells are only used after this time. The fault is detected in a few seconds after it is induced in the model, but It takes almost 300 seconds to achieve isolability of the fault since this is the time it takes for test 4 to react.

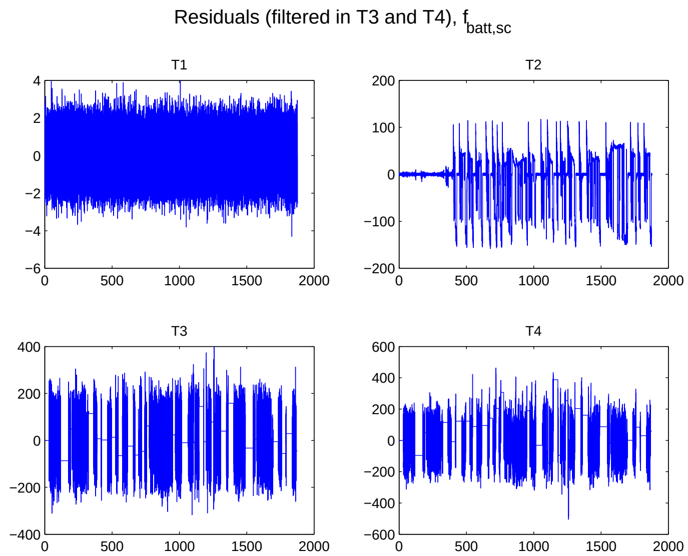

Figure 4.40: The residuals when k_{sc} is halved at 400 seconds. The residual regarding test 2 clearly changes mode after the fault has occurred. In the residual in test 4 it is more difficult to identify if a fault has occurred or not.

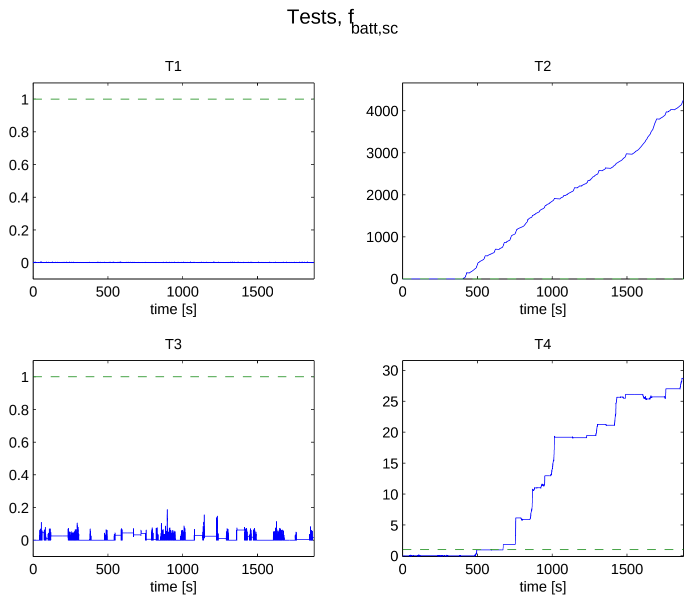

Figure 4.41: Normalized test before threshold when k_{sc} is halved at 400 seconds. Compared to the other faults test 4 reacts on, the magnitude of the test quantity is larger. It has neither the trend of decreasing at the end of the simulation as is the case when $f_{em,ki}$ is detected in Figure 4.32.

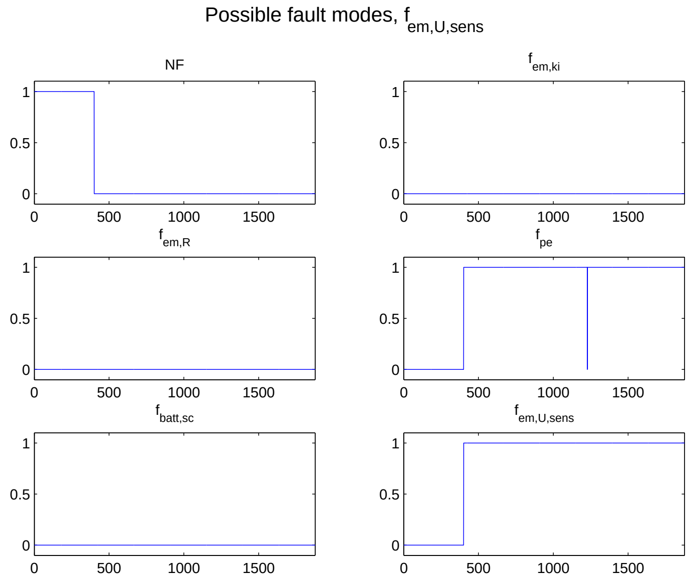

Figure 4.42: Possible fault modes when an offset of 20 V is added to the sensor signal for the voltage in the electric machine at 400 seconds. Also this fault is detected fast, but it takes 350 seconds to achieve isolability.

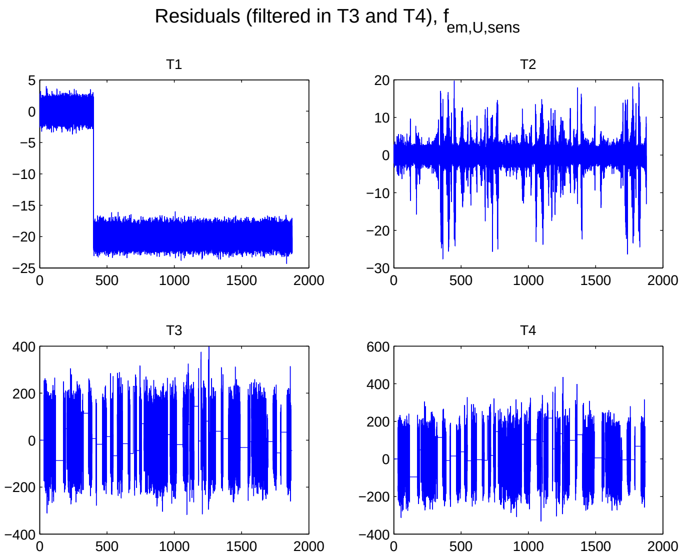

Figure 4.43: The residuals when $U_{em,sens}$ breaks down at 400 seconds.

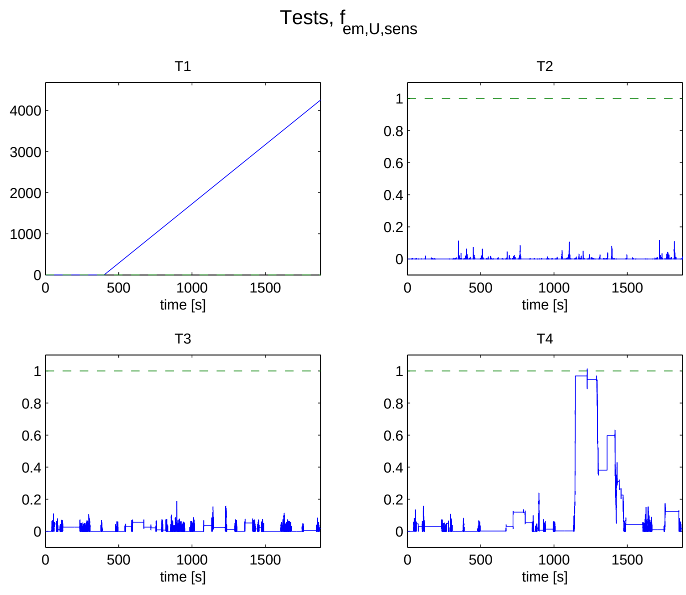

Figure 4.44: Normalized test before threshold when $U_{em,sens}$ breaks down at 400 seconds. The fault is detected fast since test 1 reacts. Test 4 only reacts for a short period (less than a second). When a larger fault in the sensor is induced in the model also test 4 reacts. This test could most likely detect this fault for a longer time if the parameters in CUSUM are modified. Though, this is not easy since the test quantity of test 4 also increases when a fault in the power electronics occurs, see Figure 4.28 and 4.38.

4.5.3 Diagnosis system 3

In addition to the faults given in Section 4.1 all sensors are monitored in the third diagnosis system. If the sensors not are monitored, only two sensors manage the task detecting and isolating the given faults. To be able to isolate the faults in the sensors, two sensors measure the same signal in this case.

Tests 3-6 uses the methodology described in equations (4.3) - (4.6) since $\dot{\omega}_{gb}$ is used in consistency relations. The residuals used in tests 2-6 are not updated when no gear is selected or the clutch is disengaged. The system starts to calculate the residuals 3 seconds after the clutch has been engaged. In tests 5 and 6 the residuals are not updated when $|U_{filt}| < 1$ V to avoid division by a small number. After a test including $\dot{\omega}_{gb}$ not has been updated, the CUSUM algorithm waits 10 seconds before it starts to update the test quantity after the test has been valid. This is to decrease the problems when ω in equations (4.3) - (4.6) is reinitialized, as described in the previous diagnosis system.

There are 6 tests used in this system and each test is described below.

Test 1

The first test in this diagnostic system compares the sensors measuring the voltage in the battery. The following sensors are thereby used:

- $U_{batt,sens,a}$
- $U_{batt,sens,b}$

and the fault the test reacts on according to the structural analysis are:

- $f_{batt,U,sens,a}$
- $f_{batt,U,sens,b}$

The residual generator is:

$$r = U_{batt,sens,a} - U_{batt,sens,b}$$

Test 2

Test 2 uses models of the battery, electric machine and the gear box. The following sensors are used:

- $\omega_{gb,sens}$
- $U_{batt,sens,b}$

and the fault the test reacts on according to the structural analysis are:

- $f_{em,R}$
- f_{pe}
- $f_{batt,sc}$
- $f_{\omega,gb,sens}$
- $f_{batt,U,sens,b}$

The residual generator is:

$$U_{batt} = U_{batt,sens,b}$$

where

$$\begin{aligned}
U_{batt} &= nU_{oc} - nR_i I_{batt} \\
U_{oc} &= f(SoC) \\
SoC &= \int \dot{SoC} dt = \int -\frac{I_{batt}}{BattCap \cdot 3600} \\
I_{batt} &= I_{em} \frac{|U_{filt}|}{U_{batt,sens,b}} \\
I_{em} &= \frac{U_{filt} - \omega_{em} k_a}{R_{em}} \\
U_{filt} &= \frac{1}{0.1s + 1} U_{em} \\
\omega_{em} &= u_{em} \omega_{mj} \\
\omega_{mj} &= u_{gb} \omega_{gb,sens} \\
u_{gb} &= f(\text{gear}) \\
U_{em} &= U_{em,control} \\
\omega_{gb} &= \omega_{gb,sens}
\end{aligned}$$

Test 3

Test 3 includes most parts of the model of the vehicle. The sensor used is:

- $\omega_{gb,sens}$

and the fault the test reacts on according to the structural analysis are:

- $f_{em,ki}$
- $f_{em,R}$
- f_{pe}
- $f_{\omega,gb,sens}$

The residual generator is:

$$\begin{aligned}
\tilde{r} &= \underbrace{\frac{1}{u_{em}} \left(J_{gb,tot} + \frac{1}{u_{final}^2} m r_w^2 \right)}_{a} \dot{\omega}_{gb} + \\
&+ \underbrace{\frac{1}{u_{em} u_{final}} (T_{drag} + T_{roll} + T_{brake}) + \frac{u_{gb}\eta_{gb}}{u_{em}} (T_{gb,loss} - T_{ice}) - (u_{gb}\eta_{gb}) T_{em}}_{b}
\end{aligned}$$

where

$$
\begin{aligned}
u_{gb} &= f(\text{gear}) \\
m &= m_{init} - m_{f,reduction} \\
m_{f,reduction} &= \int \dot{m}_f dt \\
\dot{m}_f &= \text{ice}_{controller} \frac{\omega_{ice}}{4\pi H} \\
\eta_{gb} &= \begin{cases} \eta_{pos}, & T_{mj} > T_{gb,loss} \\ \eta_{neg}, & T_{mj} \leq T_{gb,loss} \end{cases} \\
J_{gb,tot} &= \left(J_{gb} + J_{mj,tot}\right) u_{gb}^2 \\
J_{gb} &= f(\text{gear}) \\
J_{mj,tot} &= J_{em} u_{em}^2 + J_{clutch} + J_{ice} \\
T_{mj} &= u_{em} T_{em} + T_{clutch} \\
T_{clutch} &= T_{ice}, \text{ when clutch engaged} \\
T_{drag} &= \frac{1}{2} \rho C_d A_f \omega_w^2 r_w^3 \\
T_{roll} &= \begin{cases} mgC_r r_w, & 1000\omega_w > mgC_r r_w \\ 1000\omega_w, & -mgC_r r_w \leq 1000\omega_w < mgC_r r_w \\ -mgC_r r_w, & 1000\omega_w \leq -mgC_r r_w \end{cases} \\
T_{brake} &= T_{brake,control} \\
T_{gb,loss} &= f(\text{gear}, \omega_{ice}) \\
\omega_w &= \frac{\omega_{gb}}{u_{final}} \\
T_{ice} &= \left(\text{ice}_{controller} \frac{4\eta_{term}}{N_{cyl} \pi S B^2} - p_{me0,f} - p_{me0,g} \right) N_{cyl} \frac{SB^2}{16} \\
p_{me0,f} &= k_1 \left(k_2 + k_3 S^2 \omega_{ice}^2\right) \Pi \sqrt{\frac{k_4}{B}} \\
\omega_{ice} &= \omega_{mj} \\
\omega_{mj} &= u_{gb} \omega_{gb} \\
T_{em} &= \frac{U_{filt} k_i - \omega_{em} k_a k_i}{R_{em}} \\
U_{filt} &= \frac{1}{0.1s + 1} U_{em} \\
U_{em} &= U_{em,control} \\
\omega_{gb} &= \omega_{gb,sens}
\end{aligned}
$$

Test 4

The sensors used in test 4 are:

- $\omega_{gb,sens}$
- $U_{batt,sens,a}$

and the fault the test reacts on according to the structural analysis are:

- $f_{em,ki}$
- $f_{em,R}$
- $f_{batt,sc}$
- $f_{batt,U,sens,a}$
- $f_{\omega,gb,sens}$

The residual generator is:

$$\begin{aligned}\tilde{r} &= \underbrace{\frac{1}{u_{em}}\left(J_{gb,tot} + \frac{1}{u_{final}^2} m r_w^2\right)}_{a} \dot{\omega}_{gb} + \\ &+ \underbrace{\frac{1}{u_{em} u_{final}}\left(T_{drag} + T_{roll} + T_{brake}\right) + \frac{u_{gb}\eta_{gb}}{u_{em}}\left(T_{gb,loss} - T_{ice}\right) - \left(u_{gb}\eta_{gb}\right) T_{em}}_{b}\end{aligned}$$

where

$$
\begin{aligned}
u_{gb} &= f(\text{gear}) \\
m &= m_{init} - m_{f,reduction} \\
m_{f,reduction} &= \int \dot{m}_f dt \\
\dot{m}_f &= \text{ice}_{controller} \frac{\omega_{ice}}{4\pi H} \\
\eta_{gb} &= \begin{cases} \eta_{pos}, & T_{mj} > T_{gb,loss} \\ \eta_{neg}, & T_{mj} \leq T_{gb,loss} \end{cases} \\
J_{gb,tot} &= \left(J_{gb} + J_{mj,tot}\right) u_{gb}^2 \\
J_{gb} &= f(\text{gear}) \\
J_{mj,tot} &= J_{em} u_{em}^2 + J_{clutch} + J_{ice} \\
T_{mj} &= u_{em} T_{em} + T_{clutch} \\
T_{clutch} &= T_{ice}, \text{ when clutch engaged} \\
T_{drag} &= \frac{1}{2} \rho C_d A_f \omega_w^2 r_w^3 \\
T_{roll} &= \begin{cases} mgC_r r_w, & 1000\omega_w > mgC_r r_w \\ 1000\omega_w, & -mgC_r r_w \leq 1000\omega_w < mgC_r r_w \\ -mgC_r r_w, & 1000\omega_w \leq -mgC_r r_w \end{cases} \\
T_{brake} &= T_{brake,control} \\
\omega_w &= \frac{\omega_{gb,sens}}{u_{final}} \\
T_{ice} &= \left(\text{ice}_{controller} \frac{4\eta_{term}}{N_{cyl}\pi SB^2} - p_{me0,f} - p_{me0,g}\right) N_{cyl} \frac{SB^2}{16} \\
p_{me0,f} &= k_1 \left(k_2 + k_3 S^2 \omega_{ice}^2\right) \Pi \sqrt{\frac{k_4}{B}} \\
\omega_{ice} &= \omega_{mj} \\
\omega_{mj} &= u_{gb} \omega_{gb} \\
T_{em} &= I_{em} k_i \\
I_{em} &= \frac{U_{filt} - \omega_{em} k_a}{R_{em}} \\
|U_{filt}| &= \frac{I_{batt} U_{batt}}{I_{em}} \\
I_{batt} &= \frac{1}{R_i} \left(U_{oc} - \frac{U_{batt}}{n}\right) \\
U_{oc} &= f(SoC) \\
SoC &= \int \dot{SoC} \; dt \\
\dot{SoC} &= -\frac{I_{batt}}{\text{BattCap} \cdot 3600} \\
\omega_{em} &= u_{em} \omega_{mj} \\
U_{em} &= U_{em,sens,a} \\
\omega_{gb} &= \omega_{gb,sens}
\end{aligned}
$$

The expressions for I_{em} and $|U_{filt}|$ above form an equation system that has to be solved. There are four solutions for I_{em} since the equations form a second order equation. Furthermore, in the expression for $|U_{filt}|$ there is no information about the sign. The following expression for I_{em} is found:

$$I_{em} = -\frac{\omega_{em} k_a}{2R_{em}} \pm \sqrt{\left(\frac{\omega_{em} k_a}{2R_{em}}\right)^2 \pm \frac{I_{batt} U_{batt}}{R_{em}}} \tag{4.8}$$

As noted in Section 2.6.2, the absolute value operation is non-physical. Using a more physical model gives that the $\pm$ under the square root is replaced by a $+$. However, this does not change the fact that there is no unique solution for the current I_{em}. For this reason, the `CAPSim` model is used unmodified.

It is possible to reduce the number of valid solutions using the sign of $I_{batt}U_{batt}$. Using this information, in some cases a unique solution can be found, but in other cases there might be several solutions for I_{em}. A disadvantage using only the valid solution is that it is dependent on $\omega_{ice,sens}$ and therefore the noise in the sensor might affect the solution of I_{em}. This is also the case for $U_{batt,sens}$, and therefore this information is not used solving I_{em}. Due to this, residuals based on all four currents are calculated and the residual with the smallest amplitude is used as the residual in the test. If a solution of the current includes imaginary parts, the solution is not valid.

Each residual is low pass filtered accordingly to equations (4.3) - (4.6). This is made for all four residuals calculated and before one residual is chosen in the test. The disadvantage doing this is that the solution is based on a faulty estimation of I_{em} after a mode change in the solution of I_{em} occurs. The alternative to this is to filter the residual after the residual with smallest magnitude is chosen.

Test 5

The sensors used in test 5 are:

- $\omega_{gb,sens}$
- $U_{batt,sens,b}$

and the fault the test reacts on according to the structural analysis are:

- $f_{em,ki}$
- f_{pe}
- $f_{batt,sc}$
- $f_{batt,U,sens,b}$
- $f_{\omega,gb,sens}$

The residual generator is:

$$\begin{aligned} \tilde{r} \ &= \underbrace{\frac{1}{u_{em}}\left(J_{gb,tot} + \frac{1}{u_{final}^2} m r_w^2\right)}_{a} \dot{\omega}_{gb} + \\ &+ \underbrace{\frac{1}{u_{em} u_{final}} \left(T_{drag} + T_{roll} + T_{brake}\right) + \frac{u_{gb}\eta_{gb}}{u_{em}} \left(T_{gb,loss} - T_{ice}\right) - \left(u_{gb}\eta_{gb}\right) T_{em}}_{b} \end{aligned}$$

where

$$
\begin{aligned}
u_{gb} &= f(\text{gear}) \\
m &= m_{init} - m_{f,reduction} \\
m_{f,reduction} &= \int \dot{m}_f dt \\
\dot{m}_f &= \text{ice}_{controller} \frac{\omega_{ice}}{4\pi H} \\
\eta_{gb} &= \begin{cases} \eta_{pos}, & T_{mj} > T_{gb,loss} \\ \eta_{neg}, & T_{mj} \leq T_{gb,loss} \end{cases} \\
J_{gb,tot} &= \left(J_{gb} + J_{mj,tot}\right) u_{gb}^2 \\
J_{gb} &= f(\text{gear}) \\
J_{mj,tot} &= J_{em} u_{em}^2 + J_{clutch} + J_{ice} \\
T_{mj} &= u_{em} T_{em} + T_{clutch} \\
T_{clutch} &= T_{ice}, \text{ when clutch engaged} \\
T_{drag} &= \frac{1}{2} \rho C_d A_f \omega_w^2 r_w^3 \\
T_{roll} &= \begin{cases} mgC_r r_w, & 1000\omega_w > mgC_r r_w \\ 1000\omega_w, & -mgC_r r_w \leq 1000\omega_w < mgC_r r_w \\ -mgC_r r_w, & 1000\omega_w \leq -mgC_r r_w \end{cases} \\
T_{brake} &= T_{brake,control} \\
\omega_w &= \frac{\omega_{gb}}{u_{final}} \\
T_{ice} &= \left(\text{ice}_{controller} \frac{4\eta_{term}}{N_{cyl}\pi SB^2} - p_{me0,f} - p_{me0,g} \right) N_{cyl} \frac{SB^2}{16} \\
p_{me0,f} &= k_1 \left(k_2 + k_3 S^2 \omega_{ice}^2\right) \Pi \sqrt{\frac{k_4}{B}} \\
\omega_{ice} &= \omega_{mj} \\
\omega_{mj} &= u_{gb} \omega_{gb} \\
T_{em} &= I_{em} k_i \\
I_{em} &= \frac{I_{batt} U_{batt}}{|U_{filt}|} \\
I_{batt} &= \frac{nU_{oc} - U_{batt}}{nR_i} \\
U_{oc} &= f(SoC) \\
SoC &= \int \dot{SoC}\ dt \\
\dot{SoC} &= -\frac{I_{batt}}{\text{BattCap} \cdot 3600} \\
U_{filt} &= \frac{1}{0.1s + 1} U_{em} \\
U_{em} &= U_{em,control} \\
U_{batt} &= U_{batt,sens,b} \\
\omega_{gb} &= \omega_{gb,sens}
\end{aligned}
$$

Test 6

The sensor used in test 6 is:

- $U_{batt,sens,b}$

and the faults the test reacts on according to the structural analysis are:

- $f_{em,ki}$
- $f_{em,R}$
- f_{pe}
- f_{sc}
- $f_{batt,U,sens,b}$

The residual generator is:

$$\begin{aligned}\tilde{r} &= \underbrace{\frac{1}{u_{em}}\left(J_{gb,tot} + \frac{1}{u_{final}^2} m r_w^2\right)}_{a} \dot{\omega}_{gb} + \\ &+ \underbrace{\frac{1}{u_{em} u_{final}}\left(T_{drag} + T_{roll} + T_{brake}\right) + \frac{u_{gb}\eta_{gb}}{u_{em}}\left(T_{gb,loss} - T_{ice}\right) - \left(u_{gb}\eta_{gb}\right) T_{em}}_{b}\end{aligned}$$

where

$$
\begin{aligned}
u_{gb} &= f(\text{gear}) \\
m &= m_{init} - m_{f,reduction} \\
m_{f,reduction} &= \int \dot{m}_f dt \\
\dot{m}_f &= \text{ice}_{controller} \frac{\omega_{ice}}{4\pi H} \\
\eta_{gb} &= \begin{cases} \eta_{pos}, & T_{mj} > T_{gb,loss} \\ \eta_{neg}, & T_{mj} \leq T_{gb,loss} \end{cases} \\
J_{gb,tot} &= \left(J_{gb} + J_{mj,tot}\right) u_{gb}^2 \\
J_{gb} &= f(\text{gear}) \\
J_{mj,tot} &= J_{em} u_{em}^2 + J_{clutch} + J_{ice} \\
T_{mj} &= u_{em} T_{em} + T_{clutch} \\
T_{clutch} &= T_{ice}, \text{ when clutch engaged} \\
T_{drag} &= \frac{1}{2} \rho C_d A_f \omega_w^2 r_w^3 \\
T_{roll} &= \begin{cases} mgC_r r_w, & 1000\omega_w > mgC_r r_w \\ 1000\omega_w, & -mgC_r r_w \leq 1000\omega_w < mgC_r r_w \\ -mgC_r r_w, & 1000\omega_w \leq -mgC_r r_w \end{cases} \\
T_{brake} &= T_{brake,control} \\
T_{ice} &= \left(\text{ice}_{controller} \frac{4\eta_{term}}{N_{cyl} \pi S B^2} - p_{me0,f} - p_{me0,g} \right) N_{cyl} \frac{SB^2}{16} \\
p_{me0,f} &= k_1 \left(k_2 + k_3 S^2 \omega_{ice}^2\right) \Pi \sqrt{\frac{k_4}{B}} \\
\omega_w &= \frac{\omega_{gb}}{u_{final}} \\
\omega_{gb} &= \frac{\omega_{mj}}{u_{gb}} \\
\omega_{mj} &= \frac{\omega_{em}}{u_{em}} \\
\omega_{em} &= \frac{U_{filt} - I_{em} R_{em}}{k_a} \\
\omega_{ice} &= \omega_{mj} \\
U_{filt} &= \frac{1}{0.1s + 1} U_{em} \\
U_{em} &= U_{em,control} \\
I_{em} &= \frac{U_{batt} I_{batt}}{|U_{filt}|} \\
I_{batt} &= \frac{n U_{oc} - U_{batt}}{n R_i} \\
U_{oc} &= f(SoC) \\
SoC &= \int \dot{SoC} \; dt \\
\dot{SoC} &= -\frac{I_{batt}}{\text{BattCap} \cdot 3600} \\
T_{em} &= I_{em} k_i \\
U_{batt} &= U_{batt,sens,b}
\end{aligned}
$$

In this test and in test 5 there is an equation system including I_{batt} and U_{batt}. This has been solved by using the previous sample of I_{batt} in the expression for $\dot{SoC}$. This has little influence on the solution since SoC only is used to estimate U_{oc}, which is constant given the SoC of the battery during the simulation.

Thresholds in tests

Table 4.6 shows the parameter values used in the CUSUM algorithm for this system.

Table 4.6: Offsets and thresholds used in CUSUM for diagnosis system 3.

	offset	threshold
$T1$	2	30
$T2$	4	50
$T3$	70	3000
$T4$	55	18000
$T5$	65	4000
$T6$	900	180000

The parameter α that is used in the filter described in equations (4.3) - (4.6) is set to 0.1 in all test that use this filter, i.e. tests 3-6.

Decision structure

In Table 4.7 the decision structure for this diagnosis system is shown. Full isolability is possible to achieve in this system according to the structural analysis. To isolate every fault, four tests have to react except from when $U_{batt,sens,a}$ breaks down and only two tests are supposed to be affected. This can be explained that most of the vehicle model is used in every test and thereby more tests react on faults in this system compared to diagnosis systems 1 and 2. This results in that if one test not is working properly in this case, the probability for detecting the fault is increased since more tests are sensitive for the fault. The probability for full isolability decreases though when the number of tests that are expected to alarm when a fault occurs increases.

Table 4.7: Decision structure for diagnosis system2.

	$f_{em,ki}$	$f_{em,R}$	f_{pe}	$f_{batt,sc}$	$f_{batt,U,sens,a}$	$f_{batt,U,sens,b}$	$f_{\omega,gb,sens}$
$T1$					X	X	
$T2$		X	X	X		X	X
$T3$	X	X	X				X
$T4$	X	X		X	X		X
$T5$	X		X	X		X	X
$T6$	X	X	X	X		X	

Results from simulations

In this section simulations are carried out and one fault at the time is induced in the model at 400 seconds. In all simulations sensor noise is assumed with properties according to Table 4.1 except from ω_{gb} that is halved to 0.05, as described in diagnosis system 2. The parameter α is set to 0.1 in tests 3-6 and the thresholds used are tabulated in Table 4.6.

In general the faults in the electric machine is more difficult to monitor compared to the other faults using this diagnosis system. Despite this the system performs well, keeping in mind there are only three sensors monitoring seven faults.

The results from a simulation of a fault free case are presented Figures 4.45-4.47. As seen in Figure 4.47 all tests are below the threshold and therefore no false alarm occurs in the fault free case. In Figure 4.48 it is shown when the system not is updated due to slip in the clutch or that $|U_{filt}| < 1$ V.

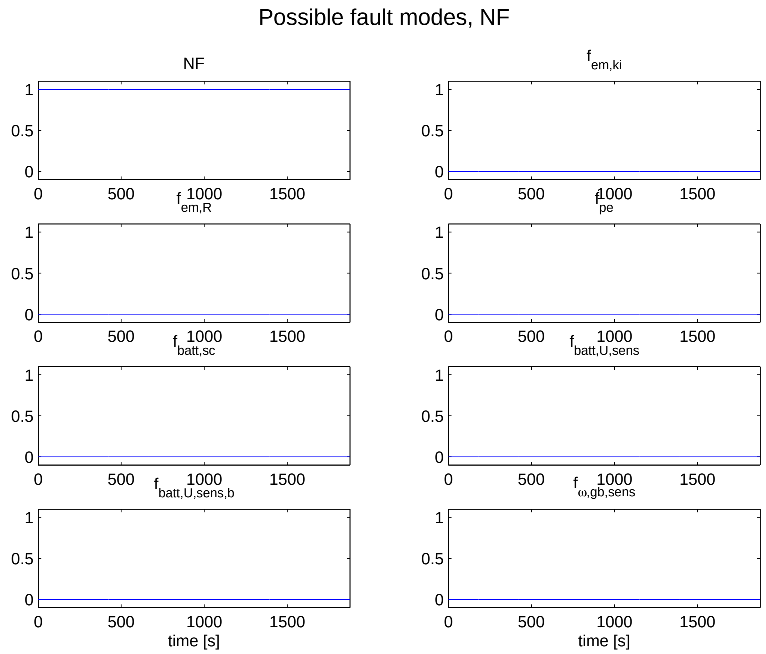

Figure 4.45: No faults are detected in a fault free case.

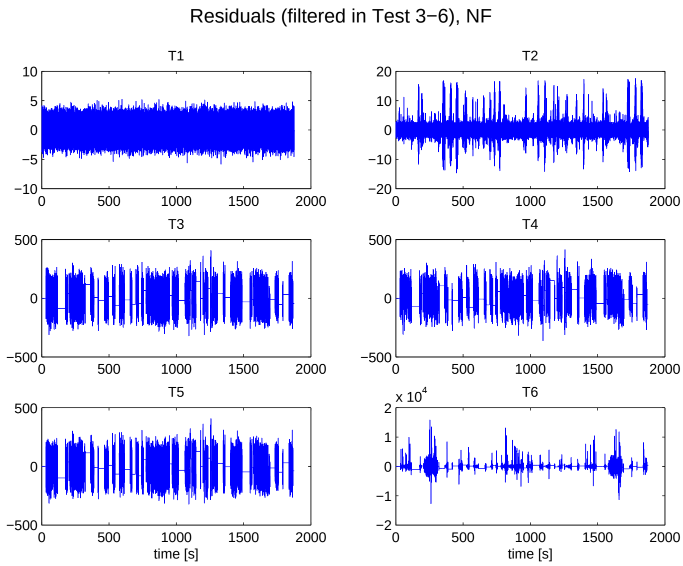

Figure 4.46: The residuals in a fault free case. The magnitude of the residual used in test 6 is larger than the other residuals. This is due to the calculation of the current in the battery that uses $I_{batt} = \frac{nU_{oc}-U_{batt}}{nR_i}$. This estimation of I_{batt} is sensitive for noise in $U_{batt,sens,b}$ when U_{batt} is close to nU_{oc}. This has a significant importance of the estimated torque, T_{em} that in some cases differ up to 100 Nm compared to the true value.

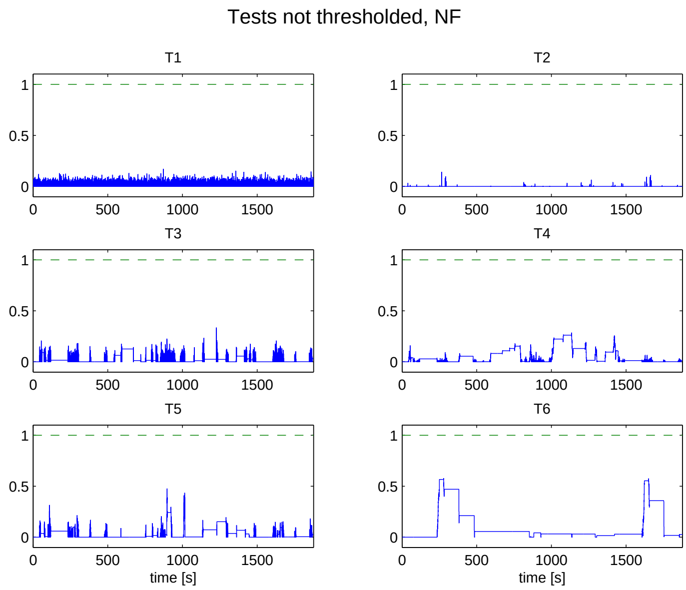

Figure 4.47: Normalized tests before threshold.

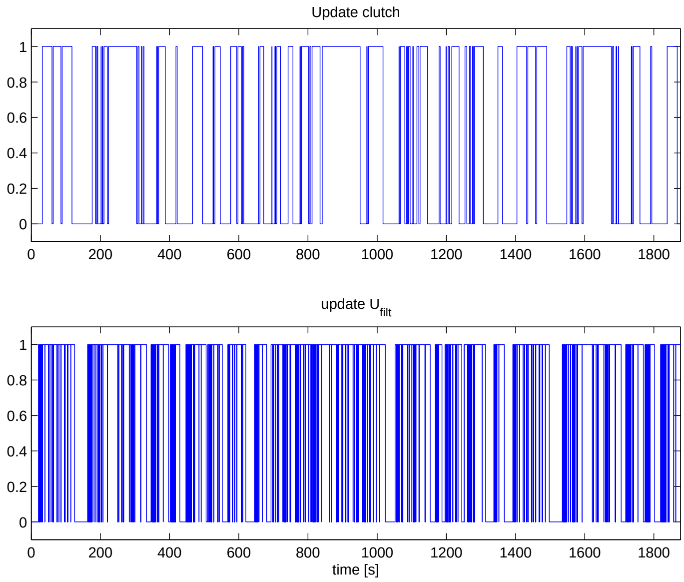

Figure 4.48: The tests are updated when all conditions required for the actual test are fulfilled. Test 1 is for example valid all the times, but test 6 only when both the clutch is engaged and $|U_{filt}| > 1$ V. When different limitations are used are shown in this figure.

The figures corresponding to the simulation where the fault k_i in the electric machine is induced, are Figures 4.49-4.51. Studying the decision structure, the fault in k_i should trigger test 3-6. Test 6 never alarms, and test 4 and 5 only partially detects the fault. Also the test quantity of test 3 has the worrying trend of being decreased after about 1400 seconds. The reason for that test 6 does not alarm is that when the parameter k_i is halved, T_{em} is halved. This has not a major impact on the overall residual, since T_{ice} is much larger than T_{em}. The noise in the system makes it difficult to find a fault that has such small impact on the residual.

Due to that test 6 does not alarm and that the tests 3-5 only do so during some periods, results in bad isolability for this fault in this diagnosis system. It is mainly $f_{em,ki}$ and $f_{\omega,gb,sens}$ that are difficult to separate from each other. This is a problem since the faults relates to two different components and further investigations has to be made to decide which component that is broken and needs to be replaced.

Though, the fault is detected most of the times since test 3 is working relatively well. Observe that it takes more than 300 seconds after the fault is induced till it is detected. The reason for this is not investigated at this stage, but one reason may be that the driving cycle in combination with the controller strategy not uses the electric machine in a way that makes the fault appear as clearly in the diagnosis system. Another reason is that the tests are not updated due to limitations when they are valid as described above (see Figure 4.48.)

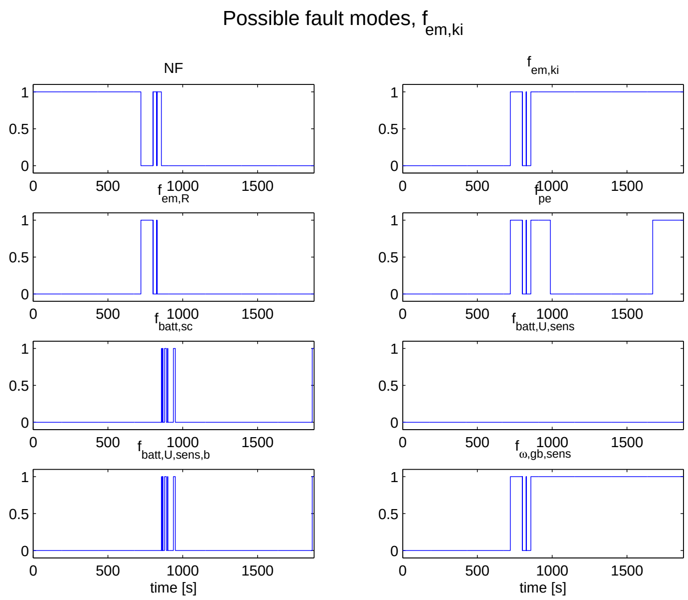

Figure 4.49: Possible fault modes when k_i is halved at 400 seconds. This fault is difficult to separate from $f_{\omega,gb,sens}$.

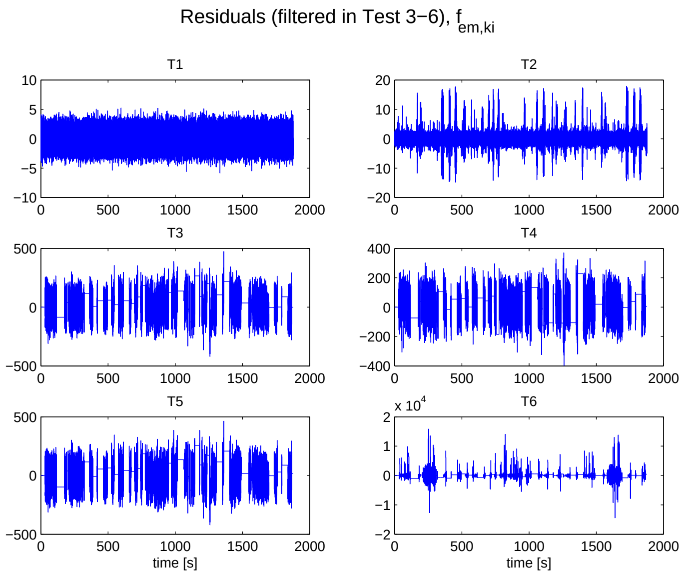

Figure 4.50: The residuals when k_i is halved at 400 seconds. Note that even when the residuals are calculated it is not sure that the test quantity is updated.

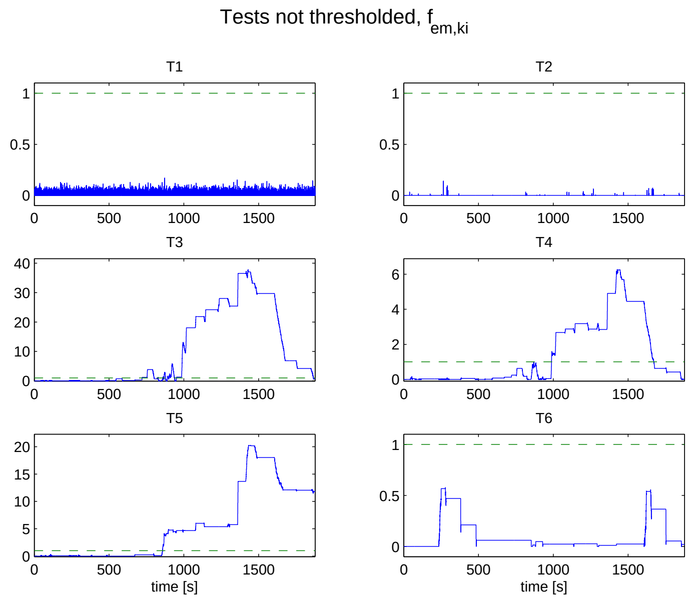

Figure 4.51: Normalized tests before threshold when k_i is halved at 400 seconds. Test 6 should alarm in this fault mode, but there is nothing in the test quantity that indicates this. Therefore it is impossible for the system to decide whether it is a fault in the electric machine or the speed sensor in the gear box that is broken. This is in analogy with the decision structure in Table 4.7.

The results from the diagnosis system when $f_{em,R}$ occurs are given in Figures 4.52-4.54. In this case all tests except test 6 do what is expected, resulting in that it is difficult to separate the fault from $f_{\omega,gb,sens}$. Test 4 is not as robust as the other tests, but the test is triggered most of the time. $f_{em,R}$ is detected after a few seconds when test 2 alarms.

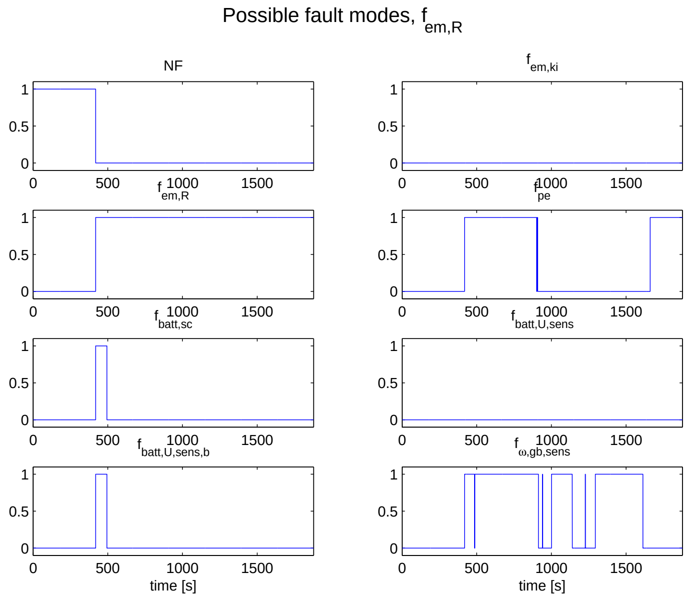

Figure 4.52: Possible fault modes when R_{em} is halved at 400 seconds. The fault is detected fast, but it is more difficult to isolate the fault.

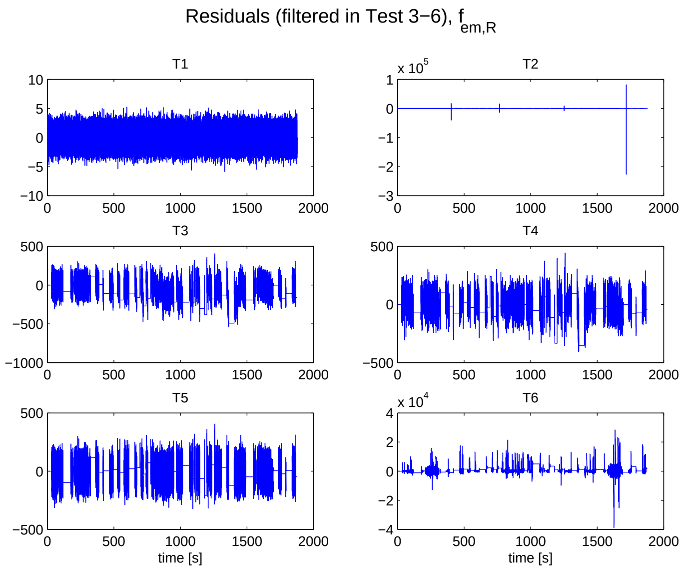

Figure 4.53: The residuals when the resistance in the electric machine is halved at 400 seconds.

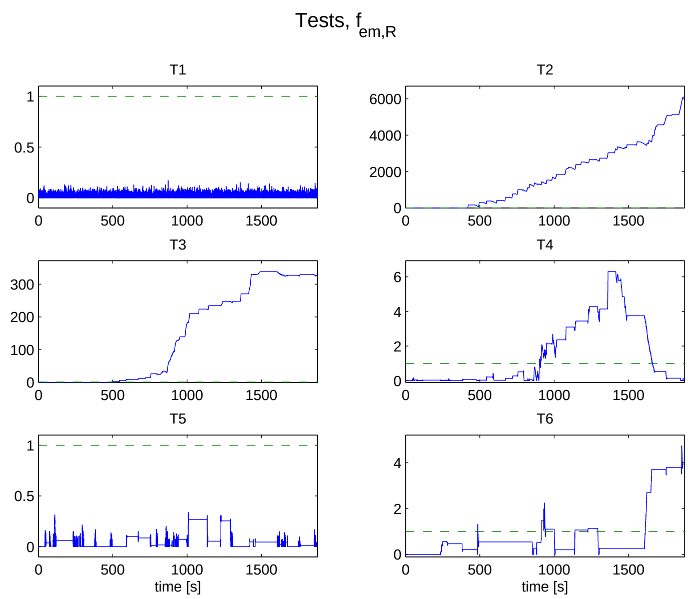

Figure 4.54: Normalized tests before threshold when R_{em} is halved at 400 seconds. It is mainly test 6 that causes some problems since the test only alarms during short periods.

The figures corresponding the case where the fault in the power electronics is induced in the model are given in Figures 4.55- 4.57. When this fault occurs all tests seem to work properly, though it takes long time for test 5 to react. One interesting issue is that the test quantity in Test 4 has increased compared to the case where no fault is induced, see Figure 4.47). One theory that could explain this is that when the vehicle not is fault free, it is operating in different modes. This influences which solution to I_{em} (equation (4.8)) that results in the smallest value of the residual. Since the four candidates to the residual to be used in the calculation of the test quantity are filtered before the selection, the time for mode shifts in the system may influence the result of the test quantity.

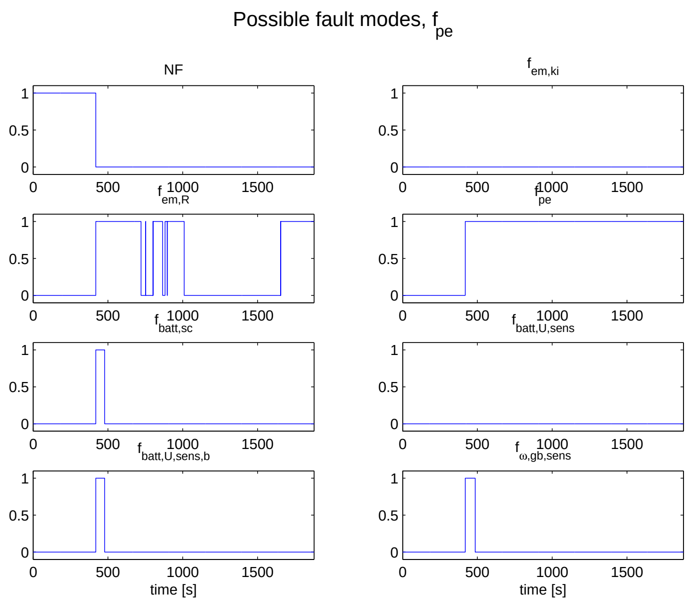

Figure 4.55: Possible fault modes when k_{pe} is halved at 400 seconds. The fault is detected fast and full isolability is achieved for some time.

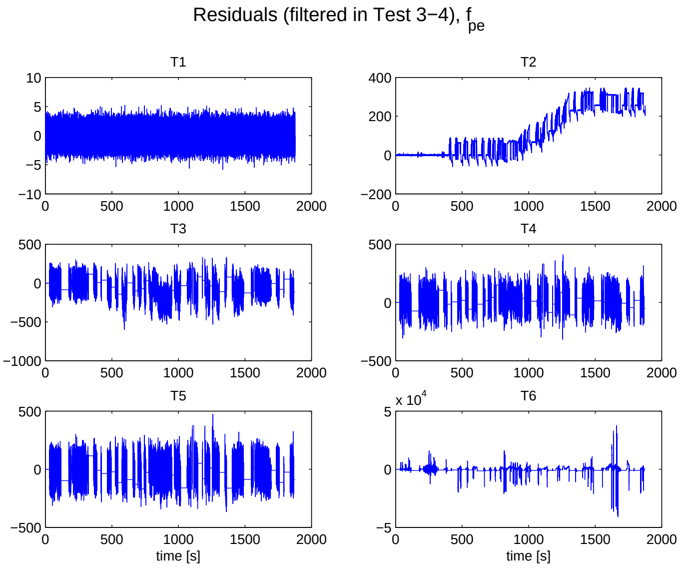

Figure 4.56: The residuals when k_{pe} is halved at 400 seconds.

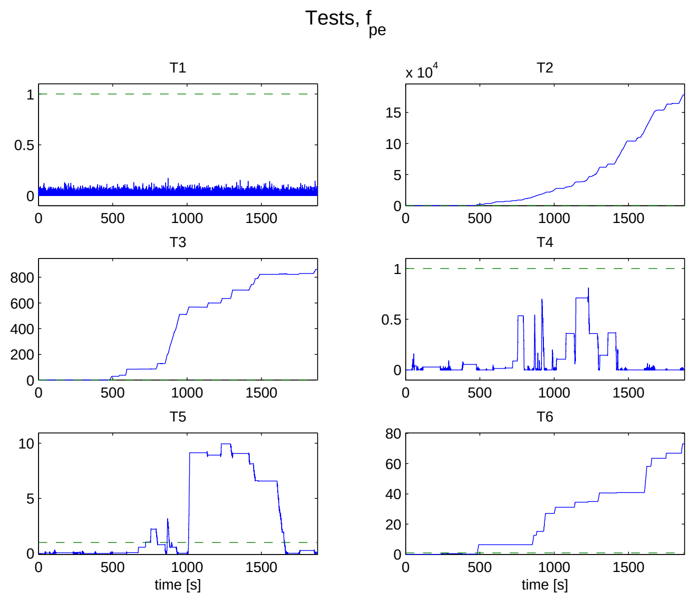

Figure 4.57: Normalized tests before threshold when k_{pe} is halved at 400 seconds. The most interesting in this figure is that the test quantity in test 4 is larger compared to the fault free case, even though the test shouldn't react according to the design of the test.

The results of the simulation where a short circuit in the battery occurs are given in Figures 4.58 - 4.60. This fault is detected and isolated in a good way.

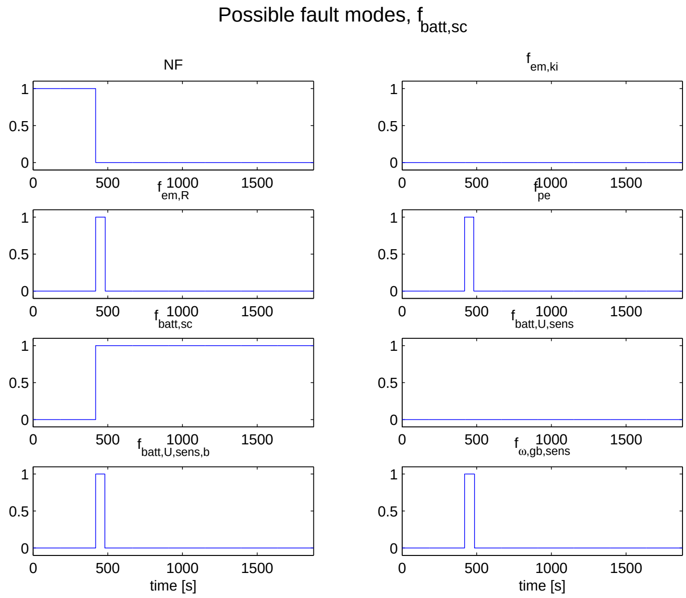

Figure 4.58: The possible fault modes when a short circuit in the battery occurs at 400 seconds and only half of the cells are used after this time. The fault is detected and isolated in less than 100 seconds.

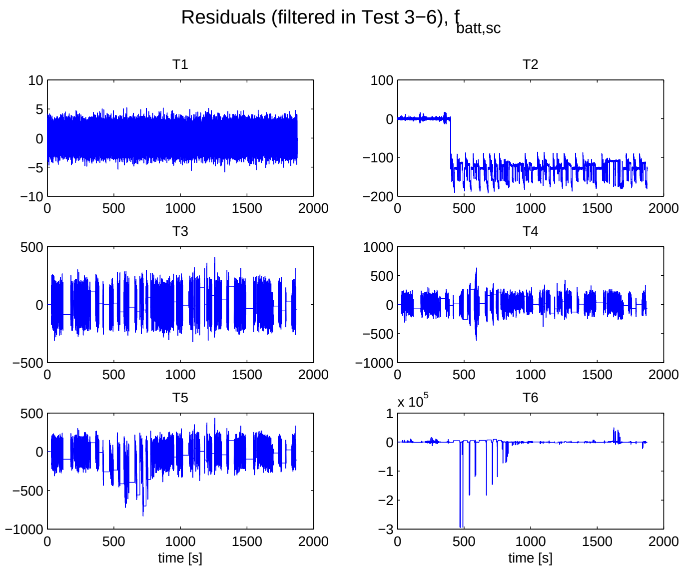

Figure 4.59: The residuals when k_{sc} is halved at 400 seconds.

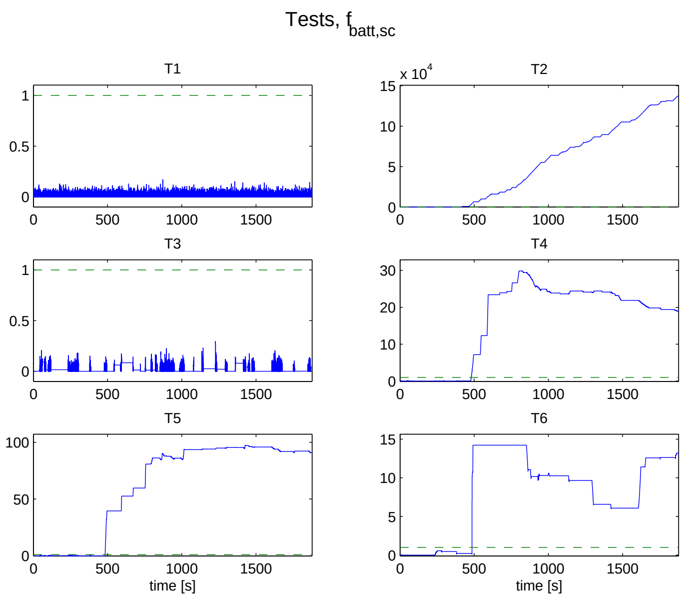

Figure 4.60: Normalized tests before threshold when k_{sc} is halved at 400 seconds. Tests 4 triggers on the fault during the first period after the fault has occurred. At the end of the simulation the test quantity decreases instead of increase as is preferred. The reason for this could be that the vehicle is driven in a different mode, which not excite the faults in the diagnosis system.

The performance of the diagnosis system when one of the two voltage sensors in the battery is simulated to brake down are given in Figures 4.61 - 4.63. This fault is detected fast, but it takes more than 100 seconds to detect which sensor for U_{batt} that is broken.

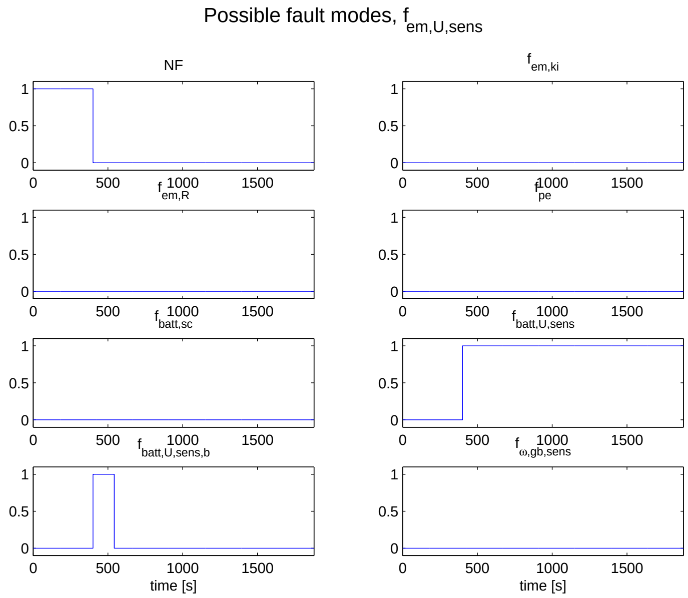

Figure 4.61: Possible fault modes when an offset of 20 V is added to one of the sensor signals for the voltage in the battery compared to the true value at 400 seconds.

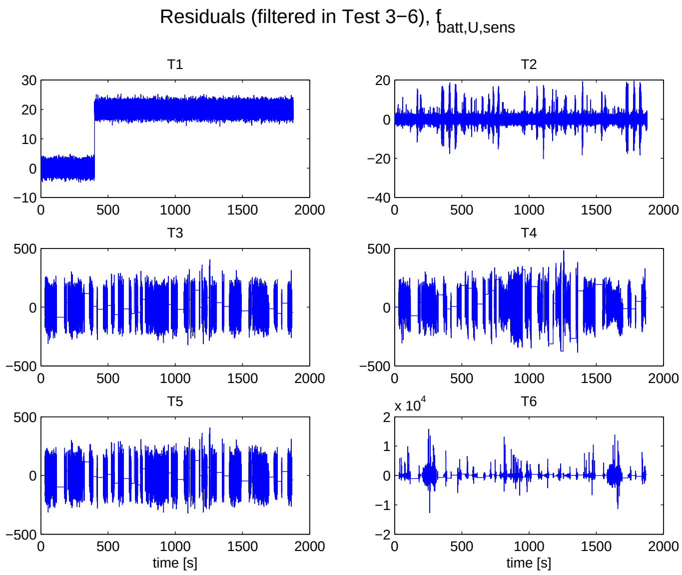

Figure 4.62: The residuals when $U_{batt,sens,a}$ breaks down at 400 seconds. In the first residual the offset is, as expected, 20 V after the fault is induced.

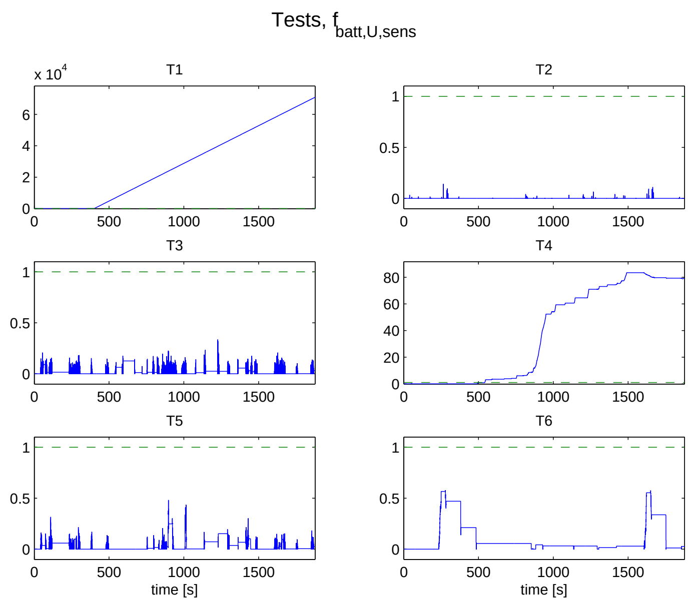

Figure 4.63: Normalized tests before threshold when $U_{batt,sens}$ breaks down at 400 seconds. Test 1 and 4 trigger as expected from the decision structure.

The result from the simulation where the second voltage sensor gets broken is shown in Figures 4.64-4.66. As the other sensor for the voltage in the battery, this sensor is monitored fast and accurately. The fault is isolated in a few seconds. Tests 1 and 2 reacts faster than the other tests. There are probably several reasons for this, but one is that test 3- 6 not are updated in 13 seconds after the clutch is used. In combination with a restriction in $|U_{filt}|$ this results in that these tests not are updated as much as test 1 and 2. Another reason for that test 3-6 are slower is that the when the transformation ω is reinitialized after the test is valid again, it is assumed the system is fault free.

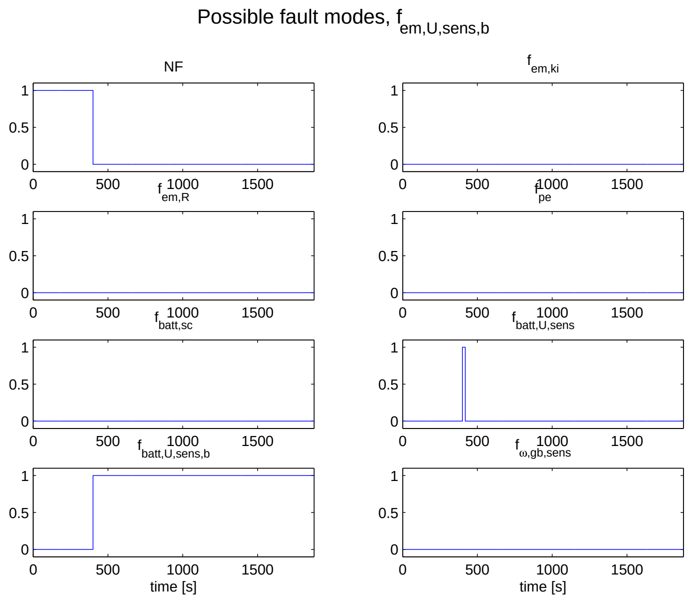

Figure 4.64: Possible fault modes when an offset of 20 V is added to the second sensor signal for the voltage in the battery compared to the true value at 400 seconds.

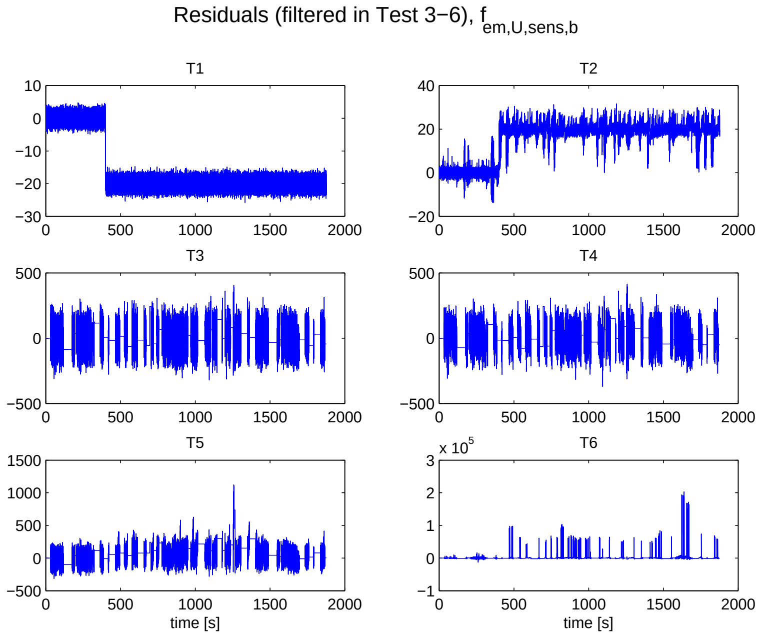

Figure 4.65: The residuals when $U_{batt,sens,b}$ breaks down at 400 seconds. In the residuals used in test 1, 2 and 6 it is clear that a fault has occurred in the system. In test 5 it is not as easy.

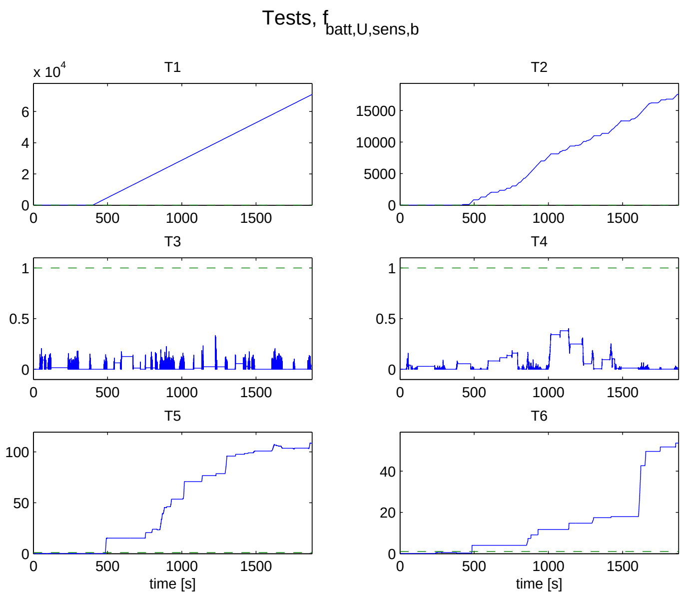

Figure 4.66: Normalized tests before threshold when $U_{batt,sens,b}$ breaks down at 400 seconds. All tests are working properly.

The performance of the diagnosis system when a fault is induced in the speed sensor in the gear box is shown in Figures 4.67- 4.69. The fault in this sensor is detected in less than 100 seconds.

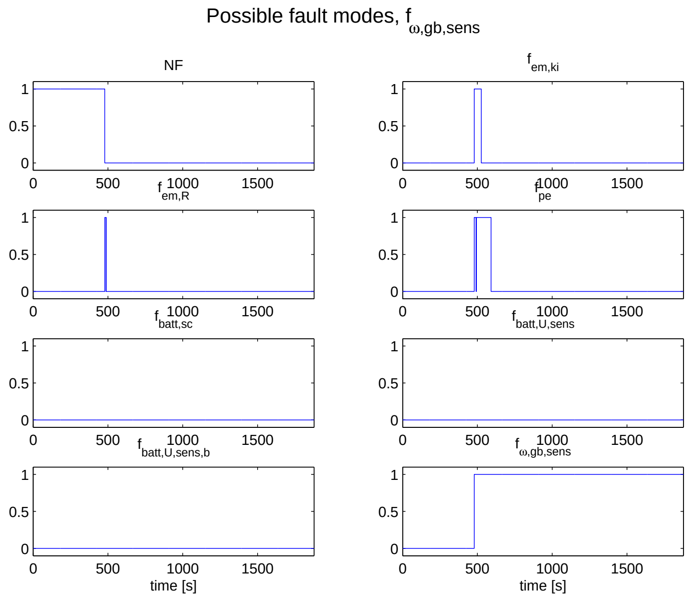

Figure 4.67: Possible fault modes when an offset of 10 rad/s is added to the sensor signal for the angular velocity at the outgoing shaft of the gear box compared to the true value at 400 seconds.

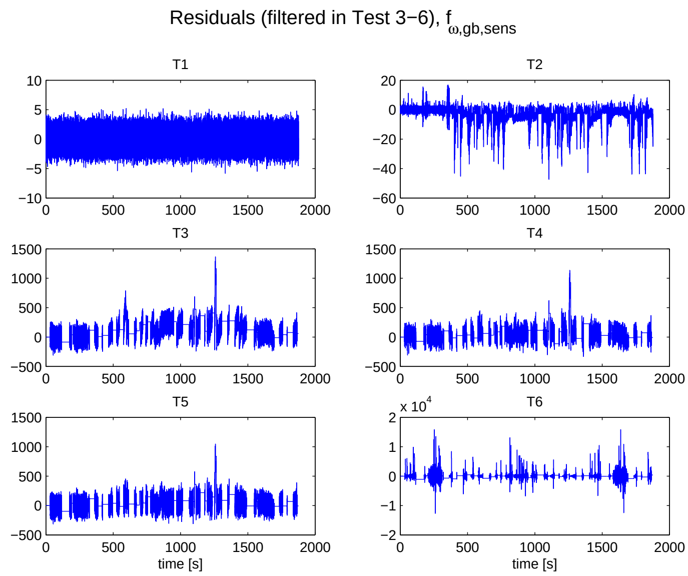

Figure 4.68: The residuals when $\omega_{gb,sens}$ breaks down at 400 seconds.

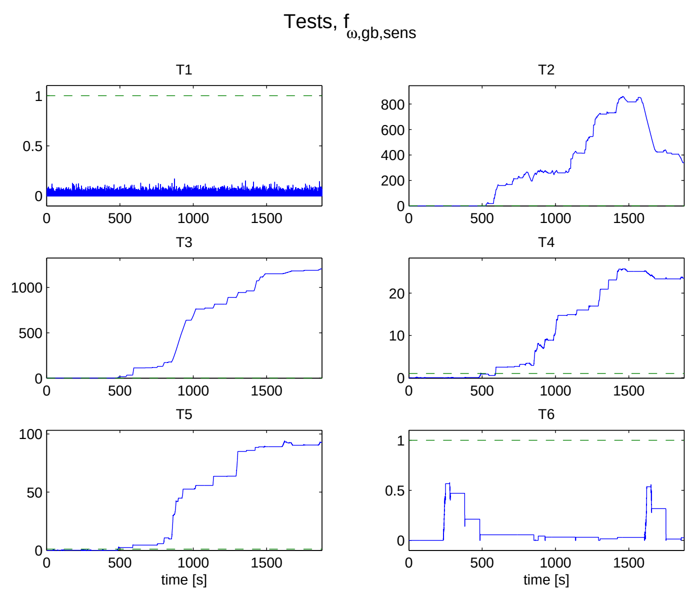

Figure 4.69: Normalized tests before threshold when $\omega_{gb,sens}$ breaks down at 400 seconds. Tests 2-5 trigger as expected.

4.6 Comparison of the diagnosis systems

Three diagnosis systems are designed and compared using the simulation platform and the difference between the diagnosis systems is the sensor configuration. The main objective of this discussion is to analyze the implications of the different sensor configurations with respect to complexity in design of the diagnosis systems, run-time computational complexity, performance, and the influence of the energy management strategy.

4.6.1 Design and run-time complexity

Each residual in the designed diagnosis systems is based on a set of equations and the size of this set is related to the complexity of designing the residual generator. One could expect that increased size of the set means increased design complexity. This is not true in general since the same principles can be used in the design regardless of the size, so a more important property of the equations is the characteristics of the equations that are included. A set of linear equations are easy to solve, almost regardless of the size of the system of equations. On the other hand, equations with strong non-linearities can be impossible to solve analytically and also difficult to solve numerically.

Dynamics in the set of equations that the residual is based on influences both the design and run-time complexity of the residual generator. In the design step, differentiated measurement signals appear in the consistency relation which has to be handled, for example with respect to measurement noise. Run-time complexity increases with the order of the dynamics, i.e. number of states.

In the first diagnosis system, analysis showed that for full isolability four residuals were needed. Each residual is based on a relatively small set of equations, primarily because enough sensors were positioned close to the supervised components. The resulting four tests are all static and requires very little computational effort. In the second system, the torque sensor is replaced by a speed sensor in the combustion engine. A consequence of this is that the, uncertain and incomplete, model of the clutch has to be included when designing tests. This also results in larger sets of equations that has to be handled in each residual generator design. The resulting diagnosis system consists of, again, four tests but now two of the tests contain dynamics. The complexity of design is similar to the first case, with the additional task of taking care of dynamic equations. The third system has a minimal number of sensors that still achieves full fault isolability. Then, for full isolability, six residual generators are required. Design complexity is similar as in the second system, but with some additional complications. In particular, in one of the dynamic residuals it is not possible, given the set of equations and with the consistency relation chosen in Section 4.5.3, to uniquely compute the current I_{em}. Instead, there are multiple possible candidate solutions that all have to be considered. With the `CAPSim` model there are four possible solutions and with the more physical oriented model there are two solutions. In effect, this means that four residuals, one for each possible solution, has to be computed to generate the final residual. Thus, a significant increase in both design and run-time complexity appears in diagnosis System 3.

4.6.2 Diagnosis systems performance

The ability to detect small changes, in the presence of uncertainties such as model inaccuracies and sensor noise, is a fundamental performance property of the diagnosis system. In the diagnosis Systems 2 and 3, non-linearities and the incomplete clutch model, make some residuals not valid in some operating points. Of course, in such regions the overall performance of the diagnosis system is degraded. The operating point of the individual components affects the activation and de-activation of the residuals. Since a hybrid driveline has some freedom in choosing the component operating points, the energy management has a direct impact on the operating points and therefore also on the diagnosis performance.

Activation of dynamic residual generators has to be done with care since there is always an initial transient due to unknown initial conditions. The suggested approach includes filtering and also a time window, directly after activation, where the corresponding test quantity is not updated. This directly affects the detection performance, in particular if de-activation and activation is frequent, which according to Figure 4.48 clearly is the case using the energy management in the platform in combination with the driving cycle.

In the tests including dynamics a resulting torque on the wheels is estimated. If for example a fault in the electric machine has occurred, this fault leads to a small relative change in the overall torque of the vehicle if the combustion engine delivers a much higher torque than the electric machine. To still be able to detect the fault, high model accuracy and robustness to measurement noise of the model estimating the torque from the combustion engine in such operating point is required. If this is considered when designing the energy management, it is possible to increase the performance of the diagnosis systems.

One interesting observation is that when dynamic residual generators are used, the test quantities in at least two cases (Figures 4.18 and 4.57) are affected even when they should not. In both these occasions a fault in the power electronics is induced, leading to that the operating points of the electric machine is modified. This may influence the impact of the noise in the system. Another possibility explaining why Test 4 in System 3 is affected when a fault in the power electronics is induced, is that the times when the mode shifts in the residual generator occurs are dependent on the operating modes of the vehicle. There are four possible solutions to I_{em} in the test, and the valid solution depends on the operating point of the electric machine. Based on these solutions for I_{em}, four residuals are calculated and filtered before the selection of the smallest residual. Due to that the filters are updated even when the solutions to the equation system not is valid, this will influence the value of the residual when it is selected. The time for these mode shifts may lead to different faults in the filter, leading to that the test quantity may be affected even when it should not be according to the analytical expression of the residual generator.

To conclude, the performance of the first system is better than the second system, that is better than the third system. The first system isolates the faults fast and distinct, is easy to design and implement, and requires low computational effort. The disadvantage, due to cost, with the first system is that three sensors are used in the electric machine. The performance of the second system is not as robust as the first system. It takes in general hundreds of seconds to isolate a fault and it is difficult to isolate the fault in the voltage sensor at all. Tests 1 and 2 in the system perform well, while the performance in Tests 3 and 4, that are based on dynamic residual generators, varies more depending on the fault in the system. The third system includes 4 dynamic tests of the same character as tests 3 and 4 in system 2. This leads to that the system based on only 3 sensors performs less regarding isolability compared to the other systems. Many of the tests in this diagnosis system are not valid at all times (Figure 4.48), due to that the model of the clutch during slip is not used in the diagnosis system, or to avoid division by zero when the voltage in the electric machine is close to 0 V.

Bibliography

[1] CAPSim. http://www.capsim.se, 2009.

[2] K.T. Chau, C.C. Chan, and Chunhua Liu. Overview of permanent-magnet brushless drives for electric and hybrid electric vehicles. *Industrial Electronics, IEEE Transactions on*, 55(6):2246 –2257, june 2008.

[3] Dilek Dustegör, Erik Frisk, Vincent Coquempot, Mattias Krysander, and Marcel Staroswiecki. Structural analysis of fault isolability in the DAMADICS benchmark. *Control Engineering Practice*, 14(6):597–608, 2006.

[4] Lino Guzzella and Alois Amstutz. Cae tools for quasi-static modeling and optimization of hybrid powertrains. *IEEE Transactions on Vehicular Technology*, 48(6):1762–1769, 1999.

[5] Lino Guzzella and Antonio Sciarretta. *Vehicle Populsion System, Introduction to Modelling and Optimization*. Springer Verlag, Zürich, 2 edition, 2007.

[6] Mattias Krysander, Jan Åslund, and Mattias Nyberg. An efficient algorithm for finding minimal over-constrained sub-systems for model-based diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 38(1), 2008.

[7] Mattias Krysander and Erik Frisk. Sensor placement for fault diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 38(6):1398–1410, 2008.

[8] Mattias Krysander and Mattias Nyberg. Statistical properties and design criterions for fault isolation in noisy systems. 19th International Workshop on Principles of Diagnosis (DX-08), Sydney, Australia, 2008.

[9] Mattias Nyberg and Erik Frisk. *Model Based Diagnosis of Technical Processes*. Linköping, 1 edition, 2008.

[10] E.S. Page. Continous inspection schemes. *Biometrika*, 41:100–115, 1954.