M. Blanke
M. Kinnaert
J. Lunze
M. Staroswiecki

Diagnosis and Fault-Tolerant Control

Second Edition

Springer

Chapter 10

Application Examples

Part of the 2nd Edition

(Springer 2006)

Chapter 10

Application examples

This chapter presents five applications that illustrate how the methods developed in the preceeding chapters can be applied under real practical conditions and how they can be combined to get a complete solution of fault-tolerant control problems. A three-tank system, a chemical process, a ship propulsion system, a steam generator and a steering-by-wire system for a warehouse truck are considered, each of which have been investigated in detail including experimental tests.

10.1 Fault-tolerant control of a three-tank system

10.1.1 Control problem

Consider the three coupled tanks depicted in Fig. 10.1. These tanks are connected by pipes which can be controlled by different valves. Water can be filled into the left and right tanks using two identical pumps. Measurements available from the process are the continuous water levels h_i of each tank and, additionally, from tank T_2 discrete signals from two capacitive proximity switches signalling whether the water level in the tank is above or below the position of the sensor.

In the nominal case (Fig. 10.2), only the left tank T_1 and the middle tank T_2 are used. The right tank T_3 and pump P_2 act as redundant hardware. The purpose of the system is to provide a continuous water flow $q_2(t) = q_N$ to a consumer. Therefore, the water level in the middle supply-tank T_2 has to be maintained within the interval $h_{2L} < h_2 < h_{2H}$, i.e. between the two discrete level sensors of tank T_2.

Water flows between the tanks can be controlled by several valves (V_{12L}, V_{12H}, V_{23L}, V_{23H}). All valves can only be completely opened or completely closed (on/off valves). The connection pipes between the tanks are placed at the bottom of the tanks (pipes with valves V_{12L}, V_{23L}) and at a height of h_H (pipes with valves V_{12H},

V_{23H}). One of the considered faults is a leakage in tank T_1 (see below). If such a leakage occurs, there is an additional outflow q_L of tank T_1 (cf. Fig. 10.1).

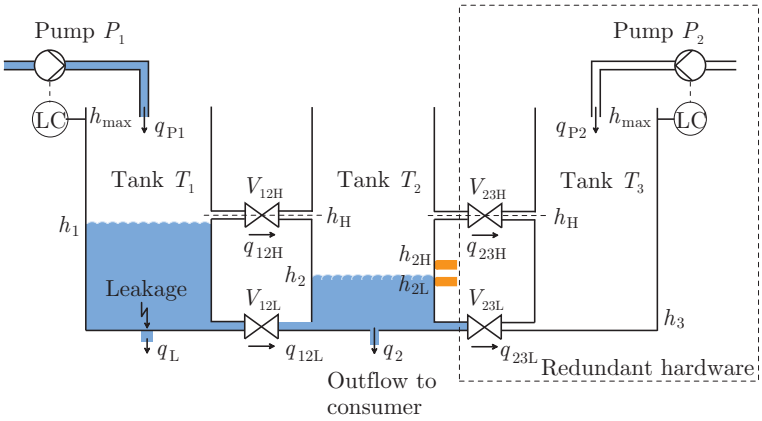

Fig. 10.1. Three-tank system

Dynamical model. Depending on the water levels and the position of the valves, different non-linear state-space models are valid. In general, the water flow q_{ij} from Tank i to Tank j can be calculated using the Toricelli law

$$q_{ij} = c_{ij} \cdot \text{ sign } (h_i - h_j) \cdot \sqrt{|h_i - h_j|} \ ,$$

where c_{ij} is a constant depending on the geometry of the connecting pipe and the valve and h_i, h_j are the water levels. The change of water volume V in a tank is described by

$$\dot{V} = A \cdot \dot{h} = \sum q_{\text{in}} - \sum q_{\text{out}} \ , \tag{10.1}$$

where $\sum q_{\text{in}}$ is the sum over all water inflows and $\sum q_{\text{out}}$ the sum over all water outflows of the tank. In (10.1), A is the cross-section area and h the water level in the cylindric tank. For the three tanks Eq. (10.1) yields:

$$\dot{h}_1 = \frac{1}{A}(q_{P1} - q_{12L} - q_{12H} - q_L) \tag{10.2}$$

$$\dot{h}_2 = \frac{1}{A}(q_{12L} + q_{12H} - q_{23L} - q_{23H} - q_2) \tag{10.3}$$

$$\dot{h}_3 = \frac{1}{A}(q_{P2} + q_{23L} + q_{23H}) \ . \tag{10.4}$$

The flows in Eqs. (10.2) - (10.4) depend on the levels h_1, h_2 and h_3 as well on the position of the valves and the commands u_{P1}, u_{P2} given to the pumps. For example, the existence of the flow q_{12H} depends on the water levels h_1 and h_2 and

the position of the valve V_{12H}. The flow is only nonzero if the valve is open and at least one liquid level exceeds the height h_H of the upper connecting pipe.

More precisely, the following expressions are obtained for the flows, with the parameters given in Table 10.1:

$$q_{P1} = \begin{cases} \text{if } h_1 \le h_{\max} \text{ and } c_{P1}\cdot u_{P1} < q_{P1}^{\max} \\ q_{P1}^{\max} \quad \text{if } h_1 \le h_{\max} \text{ and } c_{P1}\cdot u_{P1} \ge q_{P1}^{\max} \\ 0 \quad \text{otherwise ,} \end{cases}$$

$$q_{P2} = \begin{cases} c_{P2}\cdot u_{P2} \quad \text{if } h_3 \le h_{\max} \text{ and } c_{P2}\cdot u_{P2} < q_{P2}^{\max} \\ q_{P2}^{\max} \quad \text{if } h_3 \le h_{\max} \text{ and } c_{P2}\cdot u_{P2} \ge q_{P2}^{\max} \\ 0 \quad \text{otherwise ,} \end{cases}$$

$$q_{12L} = \begin{cases} c_{12L}\ \operatorname{sign}(h_1-h_2)\sqrt{|h_1-h_2|} & \text{if } V_{12L} \text{ open} \\ 0 & \text{otherwise ,} \end{cases}$$

$$q_{12H} = \begin{cases} c_{12H}\sqrt{|h_1-h_H|} & \text{if } h_1>h_H,\ h_2\le h_H,\ V_{12H} \text{ open} \\ -c_{12H}\sqrt{|h_2-h_H|} & \text{if } h_1\le h_H,\ h_2>h_H,\ V_{12H} \text{ open} \\ c_{12H}\ \operatorname{sign}(h_1-h_2)\sqrt{|h_1-h_2|} & \text{if } h_1>h_H,\ h_2>h_H,\ V_{12H} \text{ open} \\ 0 & \text{otherwise ,} \end{cases}$$

$$q_{23L} = \begin{cases} c_{23L}\ \operatorname{sign}(h_2-h_3)\sqrt{|h_2-h_3|} & \text{if } V_{23L} \text{ open} \\ 0 & \text{otherwise ,} \end{cases}$$

$$q_{23H} = \begin{cases} c_{23H}\sqrt{|h_2-h_H|} & \text{if } h_2>h_H,\ h_3\le h_H,\ V_{23H} \text{ open} \\ -c_{23H}\sqrt{|h_3-h_H|} & \text{if } h_2\le h_H,\ h_3>h_H,\ V_{23H} \text{ open} \\ c_{23H}\ \operatorname{sign}(h_2-h_3)\sqrt{|h_2-h_3|} & \text{if } h_2>h_H,\ h_3>h_H,\ V_{23H} \text{ open} \\ 0 & \text{otherwise ,} \end{cases}$$

$$q_2 = \begin{cases} c_2\sqrt{h_2} \quad \text{if } h_2>0 \\ 0 \quad \text{otherwise ,} \end{cases}$$

$$q_L = \begin{cases} c_L\sqrt{h_1} \quad \text{if } h_1>0 \text{ and leakage in tank 1} \\ 0 \quad \text{otherwise .} \end{cases}$$

Nominal configuration. In the nominal case, valves V_{12L}, V_{23H}, V_{23L} are closed and not in use. Valve V_{12H} is used to control the water level in tank T_2, pump P_1 to control the level in tank T_1. To control the water levels in the reservoir-tank T_1 and the supply-tank T_2, a conventional PI-controller and an discrete (on-off) controller are used (Fig. 10.2):

Table 10.1 Parameters and variables of the three-tank system and the controllers

h_1, h_2, h_3	[m]	Tank levels in meters
q_{P1}, q_{P2}, q_2, q_L	[m^3/s]	Volume flows in cubic metres per second
q_{12L}, q_{12H}	[m^3/s]	Volume flows in cubic metres per second
q_{23L}, q_{23H}	[m^3/s]	Volume flows in cubic metres per second
A	$1.54 \cdot 10^{-2}\mathrm{m}^2$	Cross-section area of both tanks
$h_{\max}$	0.60 m	Height of both tanks
h_H	0.60 m	Height of both tanks
c_{12L}	$1.6 \cdot 10^{-4}\mathrm{m}^{5/2}/\mathrm{s}$	Flow constant of valve V_{12L}
c_{12H}	$1.6 \cdot 10^{-4}\mathrm{m}^{5/2}/\mathrm{s}$	Flow constant of valve V_{12H}
c_{23L}	$1.6 \cdot 10^{-4}\mathrm{m}^{5/2}/\mathrm{s}$	Flow constant of valve V_{23L}
c_{23H}	$1.6 \cdot 10^{-4}\mathrm{m}^{5/2}/\mathrm{s}$	Flow constant of valve V_{23H}
c_2	$1.6 \cdot 10^{-4}\mathrm{m}^{5/2}/\mathrm{s}$	Flow constant of the outlet of tank 2
c_L	$1.6 \cdot 10^{-4}\mathrm{m}^{5/2}/\mathrm{s}$	Flow constant of a leakage in tank 1
c_{P1}	$1.0 \cdot 10^{-4}\mathrm{m}^3/\mathrm{s}$	Flow constant of pump 1
c_{P2}	$1.0 \cdot 10^{-4}\mathrm{m}^3/\mathrm{s}$	Flow constant of pump 2
$q_{P1}^{\max}$	$1.0 \cdot 10^{-4}\mathrm{m}^3/\mathrm{s}$	Maximum flow of pump 1
$q_{P2}^{\max}$	$1.0 \cdot 10^{-4}\mathrm{m}^3/\mathrm{s}$	Maximum flow of pump 1
h_1^{ref}	0.50 m	Set point of PI controller
K_P	10.0 1/m	Proportional gain of PI controller
K_I	$5.0 \cdot 10^{-2}1/\mathrm{ms}$	Integral gain of PI controller
h_{2L}	0.09 m	Position of lower discrete level sensor
h_{2H}	0.11 m	Position of upper discrete level sensor

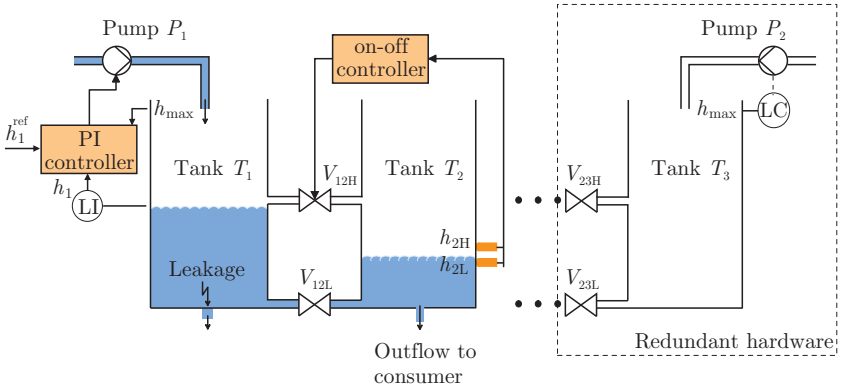

Fig. 10.2. Nominal configuration of the three-tank system

$$\begin{aligned} u_{P1}(t) &= k(h_1(t), h_1^{\mathrm{ref}}) \\ &= K_P \cdot (h_1^{\mathrm{ref}} - h_1(t)) + K_I \cdot \int_0^{'} (h_1^{\mathrm{ref}} - h_1(\tau))d\tau \end{aligned} \tag{10.5}$$

$$V_1 \quad = \quad \begin{cases} \text{open} & : \quad h_2 \leq h_{2L} \\ \text{close} & : \quad h_2 \geq h_{2H} \\ \text{no change} & : \quad h_{2L} < h_2 < h_{2H} \ , \end{cases} \tag{10.6}$$

where K_P and K_I are controller parameters and h_1^{ref} is the set-point for tank T_1. Equation (10.6) describes under what conditions the on-off controller changes the position of the valve from opened to closed or vice-versa. All parameters of the controllers are given in Table 10.1.

In summary, the nominal behaviour is characterised by the following:

- Only the left and middle tank are in use, water level h_2 must be medium, the set-point for h_1 is chosen to h_1^{ref}.
- Valves $V_{12L}, V_{23L}, V_{23H}$ are closed.
- No leakage occurs ($q_L = 0$).
- The PI-controller (10.5) controls the level h_1 of tank T_1 with pump P_1 using a continuous level sensor.
- The on-off controller (10.6) controls the level h_2 of tank T_2 with valve V_1 using discrete level sensors.

Reconfiguration problem. Three different fault scenarios are given:

1. Fault f_1: Valve V_{12H} is closed and blocked.
2. Fault f_2: Valve V_{12H} is opened and blocked.
3. Fault f_3: A leakage in Tank T_1 occurs ($q_L \neq 0$).

The reconfiguration task is to find *automatically* a new control configuration of the three-tank system such that

- the water level h_2 remains between h_{2L} and h_{2H} for all scenarios, i.e. the relation

$$[h_2(k)] = medium \tag{10.7}$$

  should hold for $k \geq \bar{k}$ for a possibly small $\bar{k}$.
- for scenario 3, the loss of water is minimal, i.e.

$$[h_1(k)] = empty \tag{10.8}$$

  should hold for $k \geq \bar{k}$ for a possibly small $\bar{k}$.

The reconfiguration task consists in finding a new control structure by selection of actuators and sensors, new control laws and new set-points for the control loops, such that the control aims above are met. If needed, the use of redundant hardware components is possible. Obviously, the idea of reconfiguration cannot be satisfied by simply changing the parameters K_P or K_I, but a structural change of the system is necessary.

10.1.2 Generic component-based analysis of the three-tank system

This section applied the methods elaborated in Chapter 4 to the three-tank example.

Modelling of the field components. The three tank system is composed of the interconnection of 14 elementary components, namely

- 4 valves: V_1, V_{13}, V_2, V_{23},
- 2 pumps: P_1, P_2,
- 3 tanks: T_1, T_2, T_3,
- 3 level sensors: L_1, L_2, L_3,
- 2 controllers $C_1 C_2$.

The interconnecting pipes might also be considered as components, if the service they deliver was to be analysed.

The generic model of each of these components should include:

1. its use-mode automaton,
2. the list of services associated with each use-mode.

Valves. Consider first the valves V_i, $i = 1, ...4$, and assume they are all described by the same model, with the use-mode set = $\{V_i$_off, V_i_maintenance, V_i_automatic$\}$, associated with the service lists:

- *V_i_off: V_i_to_maintenance, V_i_to_automatic*
- *V_i_maintenance: V_i_open, V_i_close, V_i_open_manual, V_i_close_manual, V_i_to_off, V_i_ to_automatic*
- *V_i_automatic: V_i_open, V_i_close, V_i_to_off, V_i_to_maintenance*

Since only the automatic behaviour will be of interest for the valves as well as for the other components, the partial model associated with the automatic operation mode, whose services are *{V_i_ open, V_i_ close}* is the only one which will be considered.

The definition of these services in terms of consumed, produced variables and procedure is as follows, where q_i is the flow through valve i, Δp_i is the pressure drop between the input and output of the valve, and k_i is a parameter.

Service	Consumed	Produced	Procedure
V_i_open :	Δp_i	q_i	$q_i = k_i \text{ sign } (\Delta p_i)\sqrt{\lvert\Delta p_i\rvert}$
V_i_close :	Δp_i	q_i	$q_i = 0, \forall \Delta p_i$

Pumps. Each pump $P_i, i = 1, 2$ can provide the single service $deliver\_Q_i$ where Q_i is the flow parameter associated with the request for the "deliver" service, and q_i is the flow really delivered.

Service	Consumed	Produced	Procedure
$deliver\_Q_i$	Q_i	q_i	$q_i = Q_i$

Tanks. Each tank $T_i, i = 1, 2, 3$ can provide the service T_i_store, where l_i is the level in tank i , Δq_i is the difference between the input and output flows in the tank, $l_i^{\max}$ and a_i are parameters.

Service	Consumed	Produced	Procedure
T_i_store	Δq_i	l_i	$l_i(t) = \min\left\{\max\left\{0, a_i \int \Delta q_i(t)dt\right\}, l_i^{\max}\right\}$

Sensors. Each sensor $L_i, i = 1, 2, 3$ can provide the service $level\_value_i$, where l_i is the true level in tank i, h_i is its measured value (estimated by the sensor) and g is a given function.

Service	Consumed	Produced	Procedure
$level\_value_i$	l_i	h_i	$h_i = g(l_i)$

Controllers. Controller C_1 produces the Q_i parameters of the pumps. Two services are provided, namely *max_flow* which delivers the control signal for the maximum flow and *regul_flow* which delivers the control signal for a PI regulated flow, where $Q_i^{\max}$ is the maximum value of the flow which can be requested from pump P_i, w_i is the reference level for the PI controller, and K_{Pi}, (respectively K_{Ii}) are the proportional (respectively integral) coefficients of the PI regulator.

Service	Consumed	Produced	Procedure
max_flow$_i$:	$-$	Q_i	$Q_i = Q_i^{\max}$
$regul\_flow_i$:	h_i, w_i	Q_i	$Q_i = K_{Pi}(h_i - w_i) + K_{Ii}\int(h_i - w_i)dt$

Controller C_2 provides the service $control\_V_i$, which is associated with an on/off regulation, and which requests the V_i_open and V_i_close services of the valves, where h_i^-, h_i^+ are the min/max level values associated with the on/off regulator, v_i is the control request to valve i ($v_i \in \{$*open,close*$\}$) .

Service	Consumed	Produced	Procedure
$control\_V_i:$	h_i, h_i^-, h_i^+	v_i	$h_i \leq h_i^- \Longrightarrow v_i = open$ $h_i \geq h_i^+ \Longrightarrow v_i = close$

Use-modes and objectives. The definition of the use-mode set of the overall system, and for each use-mode the definition of its associated objectives directly result from the specification analysis. For the three-tank system, the different objectives are the following:

- Objective 0: No action
- Objective 1: Reach the level set points as fast as possible
- Objective 2: Regulate the levels to the set points
- Objective 3: Completely empty the system
- Objective 4: Protect the environment.

The following table gives the different use-modes and the associated objectives.

Number of use-mode	Name	Objectives
0	No_operation	0
1	Preparation	1, 4
2	Regulation	2, 4
3	End_of_production	3, 4
4	Fall_back	3

The associated use-mode management graph is given on Fig. 10.2.

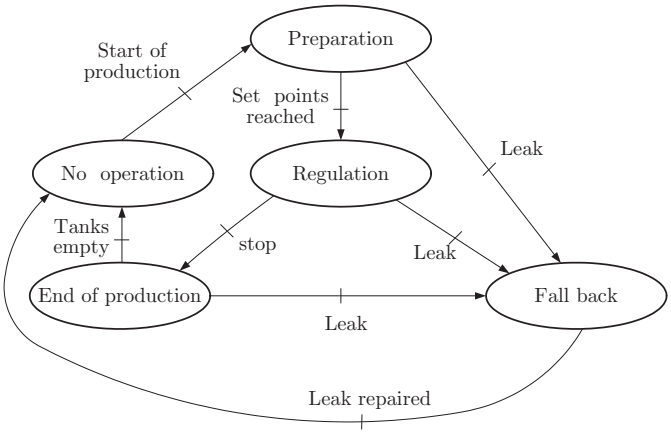

Fig. 10.3. Use-mode management graph

High-level services. Objectives are achieved from the services provided by the system elementary components. Figure 10.4 gives the pyramidal decomposition of the three-tank system, which is decomposed into three subsystems, each of them constituted of one tank and its instrumentation.

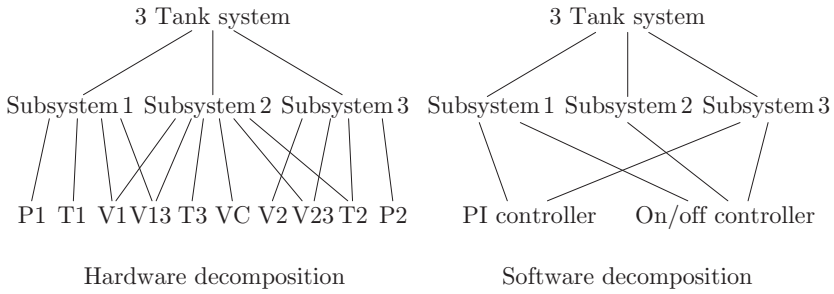

Fig. 10.4. Pyramidal decomposition

The services of the elementary components define "instructions" (a basic vocabulary) with which "programs" (words formed on this vocabulary, using given connectors) can be written to deliver services of higher level. The feasible combinations of elementary services which provide the subsystem 1 (respectively subsystem 3) services are given by the first column of table 1 (respectively table 2) where the writing

$$\text{S1.2} = \{T_1\_store, deliver\_Q_1, V_1\_close, V_{13}\_open\}$$

means that the four elementary services in the brackets are needed to provide the high-level service: "decrease level in Tank T1". Note that, according to the control architecture, this high-level service could be implemented in different ways, for example:

$$T_1\_store \;//\; deliver\_Q_1 \;//\; V_1\_close \;//\; V_{13}\_open$$

(where // means the parallel execution connector) if the pump and the valves are intelligent actuators with their own processing unit, or

$$T_1\_store \;//\; PV$$

where PV is defined by the program:

do

$deliver\_Q_1$
V_1_close
V_{13}_open

end_do

which corresponds to the "word" $deliver\_Q_1/V_1\_close/V_{13}\_open$, where / is the sequencial execution connector, if P_1, V_1, V_{13} are controlled by the same processing unit. The corresponding functional interpretations in the case where the services are

run from the regulation nominal conditions are given by the second column of the tables.

Table 10.2 Service of subsystem 1: A functional interpretation for $h_1 = 0.5$ m, and $h_3 = 0.1$ m

Feasible combination	Functional interpretation
$S1.1 =$ $\{T_1\_store, deliver\_Q_1, V_1\_close, V_{13}\_close\}$	if $Q_1 \neq 0$ increase level 1, else keep level 1 constant
$S1.2 =$ $\{T_1\_store, deliver\_Q_1, V_1\_close, V_{13}\_open\}$	decrease level 1
$S1.3 =$ $\{T_1\_store, deliver\_Q_1, V_1\_open, V_{13}\_close\}$	decrease level 1
$S1.4 =$ $\{T_1\_store, deliver\_Q_1, V_1\_open, V_{13}\_open\}$	decrease level 1

Table 10.3 Service of subsystem 3: A functional interpretation for $h_1 = 0.5$ m, and $h_3 = 0.1$ m

Feasible combination	Functional interpretation
$S3.1 = \{T_3\_store, V_1\_close, V_{13}\_close\}$	decrease level 3
$S3.2 = \{T_3\_store, V_1\_close, V_{13}\_open\}$	increase level 3
$S3.3 = \{T_3\_store, V_1\_open, V_{13}\_close\}$	increase level 3
$S3.4 = \{T_3\_store, V_1\_open, V_{13}\_open\}$	increase level 3

Services of subsystems 1 and 3 can themselves be associated to provide services at the system level. As in the previous step, feasible associations are automatically generated by the exploration of the possible service combinations taking care that the choice of a Table 10.3 service may impose the choice of a Table 10.4 service, since a part of the elementary services implied in the aggregated one are the same. For example, a service which makes use of V_1_open cannot be run simultaneously with a service which makes use of V_1_close. Feasible combinations of group 1 and 3 services to provide system services are given by the first column of Table 10.4. The corresponding functional interpretation in the case where the service is run from the regulation nominal conditions is given by the second column.

Table 10.4 High-level services: A functional interpretation for $h_1 = 0.5$ m and $h_3 = 0.1$ m

High-level service	Functional Interpretation
$S0.1 = \{S1.1, S3.1\}$	increase or keep level 1, decrease level 3
$S0.2 = \{S1.2, S3.2\}$	decrease level 1, increase level 3
$S0.3 = \{S1.3, S3.3\}$	decrease level 1, increase level 3
$S0.4 = \{S1.4, S3.4\}$	decrease level 1, increase level 3

Table 10.4 shows that there is no service which allows to keep Tank 1 and Tank 3 levels constant. Consequently the regulation objective can only be performed by requesting successively level-increasing and level-decreasing services. The regulation objective achievement requires the level set points w_1, h_3^-, h_3^+ as input and can be provided by different versions, which are given by

$$
\begin{aligned}
M2.1 &= \{S0.1, S0.3, level\_value_3, level\_value_1, regul\_flow_1, control\_V_1\} \\
M2.2 &= \{S0.1, S0.2, level\_value_3, level\_value_1, regul\_flow_1, control\_V_{13}\} \\
M2.3 &= \{S0.1, S0.4, level\_value_3, level\_value_1, regul\_flow_1, control\_V_1, \\
&\quad control\_V_{13}\}
\end{aligned}
$$

The algorithm which realises service $M2.1$ could be, for example:

Algorithm 10.1 *M2.1 algorithm*

Inputs: w_1, h_3^-, h_3^+
Do: Until end of regulation service.
1. $regul\_flow\_Q_1$: $Q_1 = f(w_1, h_1)$.
2. $control\_V_1$: $v_1 = f(h_3, h_3^-, h_3^+)$.
3. if $v_1 = open$ then S0.3 else S0.1.

Faults scenarios. When faults occur, some lower level services become unavailable or become permanent in time. The available versions of the high-level services are those which do not require the lost low-level services and in which the permanent low-level services are implied. Let us consider three examples.

Scenario 10.1 The current operation mode is UM2, and the currently used version for achieving objective 2 is the nominal one M2.1. Suppose that Valve V_1 gets blocked closed. The analysis is as follows:

- Service V_1_close gets permanent in time and service V_1_open becomes unavailable. Therefore, services S1.3, S1.4, S3.3, and S3.4 become unavailable, which implies the unavailability of services S0.3 and S0.4.

- There exists one version, namely M2.2, which can be run to achieve the regulation objective, since it does not make use of any unavailable service. The nominal version can no longer be provided but objective 2 can still be achieved, thanks to service reconfiguration.

Scenario 10.2 The current operation mode is UM2, and the currently used version for achieving objective 2 is the nominal one M2.1. Suppose that Valve V_1 gets blocked open. The analysis is as follows:

- Service V_1_open gets permanent in time and service V_1_close becomes unavailable. Therefore, services S1.1, S1.2, S3.1, S3.2 become unavailable, which implies the unavailability of services S0.1, S0.2.
- The nominal version for achieving objective 2 can no longer be provided, and no other version can be performed, since service S0.1 is common to all versions. Level 1 cannot be kept to $0.5\,\mathrm{m}$, and it makes no sense to stay in UM2 any longer. A possible solution is to move to another use-mode whose missions both do not contain the regulation one and can be fulfiled using the reduced set of remaining services. Another possible solution is to change the original mission parameters so as to make success possible, e. g. change level 1 set point from w_1 to w_1^*, where w_1^* is the level of the valve V_1 connecting pipe. With this new objective, Tables 10.2, 10.3, 10.4 becomes Tables 10.5, 10.6, 10.7 (only the functional interpretation changes) and level 3 can be regulated using one of the versions given by

$$M2.1 = \{S0.3, S0.4, level\_value_3, level\_value_1, regul\_flow\_Q_1, control\_V_{13}\}.$$

Table 10.5 Service of subsystem 1: A functional interpretation for $h_1 = w_1^*$ and $h_3 = 0.1\ \mathrm{m}$

Feasible combination	Functional interpretation
$S1.3 =$ $\{T_1\_store, deliver\_Q_1, V_1\_open, V_{13}\_close\}$	if $Q_1 \neq 0$ increase level 1, else keep level 1 constant
$S1.4 =$ $\{T_1\_store, deliver\_Q_1, V_1\_open, V_{13}\_open\}$	decrease level 1

Table 10.6 Service of subsystem 3: A functional interpretation for $h_1 = w_1^*$ and $h_3 = 0.1\ \mathrm{m}$

Feasible combination	Functional interpretation
$S3.3 = \{T_3\_store, V_1\_open, V_{13}\_close\}$	decrease level 3
$S3.4 = \{T_3\_store, V_1\_open, V_{13}\_open\}$	increase level 3

Table 10.7 High-level services: A functional interpretation for $h_1 = w_1^*$ and $h_3 = 0.1\ \mathrm{m}$

Feasible combination	Functional interpretation
$S0.3 = \{S1.3, S3.3\}$	decrease level 1, increase level 3
$S0.4 = \{S1.4, S3.4\}$	decrease level 1, increase level 3

Scenario 10.3 There is a leak in Tank T_1. The corresponding storage service becomes unavailable and the environment protection objective can no longer be fulfilled. The system has to be moved to an use-mode in which this objective does not appear, namely the fall back use-mode. In this use-mode, achieving objective 3 leads to completely empty the tanks.

As a remark, it should be noted that for the combination of Tanks T_2 and T_3 to appear as a redundant hardware allowing further reconfiguration, objective 2 should have been formulated as: "regulate levels 1 and 3 or levels 2 and 3 to the set points".

10.1.3 Solution of the reconfiguration task

The reconfiguration problem of the three-tank system includes discrete decisions that have to be made concerning the choice of the actuators, the sensors, the controller and the set-points. Therefore, it is reasonable to use a representation of the three-tank system which refers directly to these decision variables. The method presented in Section 9.7 will be applied here, where the non-deterministic automaton is abstracted from a discrete-time version of the continuous-variable model (10.2) – (10.4).

Partitioning of the signal spaces. The quantiser of the level h_2 is already given in the problem formulation, where this level is only known to assume one of the three qualitative values *low*, *medium* or *high*. The quantisers for h_1 and h_3 are introduced deliberately, because the decision concerning the reconfiguration of the controller does, in general, not depend on the precise quantitative value $\boldsymbol{x}$ but on a global assessment $[\boldsymbol{x}]$ of the state. Hence, the signal spaces of the tank levels are partitioned into the six intervals described in the following table.

$[h_i] = empty$	$h_i \in [0, 0.09\ \mathrm{m})$
$[h_i] = low$	$h_i \in [0.09\ \mathrm{m},\ 0.11\ \mathrm{m})$
$[h_i] = medium$	$h_i \in [0.11\ \mathrm{m},\ 0.49\ \mathrm{m})$
$[h_i] = high$	$h_i \in [0.49\ \mathrm{m},\ 0.51\ \mathrm{m})$
$[h_i] = full$	$h_i \in [0.51\ \mathrm{m},\ 0.60\ \mathrm{m})$
$[h_i] = overflow$	$h_i \geq 0.6\ \mathrm{m}$

Instead of the tank levels h_i only the qualitative values $[h_i]$ are assumed to be known, which form the vector

$$[\boldsymbol{x}(k)] = ([h_1(k)],\ \ [h_3(k)],\ \ [h_2(k)])'.$$

For the valves, the qualitative values *closed* and *open* correspond to the quantitative values $Pos(V) = 0$ or $Pos(V) = 1$, respectively. The set point h_1^{ref} is assumed to have one of the qualitative values $[h_1]$ of the level of Tank T_1 and the pump P_2 is assumed to have three qualitative values as shown in the following table:

$[V_i] = closed$	$Pos(V_i) = 0$
$[V_i] = open$	$Pos(V_i) = 1$
$[Q_2^{P_2}] = off$	$Q_2^{P_2} = 0$
$[Q_2^{P_2}] = medium$	$Q_2^{P_2} = 0.5\, Q_{\max}$
$[Q_2^{P_2}] = on$	$Q_2^{P_2} = Q_{\max}$

The discrete input vector $[\boldsymbol{u}(k)]$ is composed of the qualitative input values similarly as the qualitative state $[\boldsymbol{x}(k)]$.

Qualitative modelling of the tank system. A model (9.38) of the three-tank system subject to the three faults considered can be obtained by applying the abstraction method developed in Section 9.4.3. For the reconfiguration purposes, a nondeterministic automaton is used rather than a stochastic automaton. The automaton takes into account all three tanks, because the right tank has to be used in case of the fault f_3. As the levels in Tanks T_1 and T_3 are quantised into 6 intervals each and the level of the middle tank into three intervals, the automaton has $6 \cdot 3 \cdot 6 = 102$ states and cannot be shown here.

After the qualitative model has been obtained by the abstraction procedure, the controller has been found by Algorithm 9.3. With the requirements given for the three-tank system, the set $\mathcal{Z}_{\mathrm{Aim}}\,(f)$ of admissible operation points is given by

$$\mathcal{Z}_{\mathrm{Aim}}\,(f) = \{[\boldsymbol{x}] \,:\, \text{Eqs. (10.7) and (10.8) are satisfied}\}.$$

For fault f_1,

$$\mathcal{Z}_{\mathrm{Aim}}\,(f_1) = \{([h_1], medium, [h_3])' \text{ with arbitrary } [h_1], [h_3]\}$$

holds. The function k_q can be represented as a tabular showing the relation among the different faults f, the qualitative state $[\boldsymbol{x}]$ and the qualitative input $[\boldsymbol{u}]$.

Experimental results. Experiments with the three-tank system have been made with sampling time $T_s = 7\,\mathrm{s}$ and with the quantiser described above. $[\boldsymbol{x}(k)]$ is measured by capacity sensors that indicate merely whether the liquid level in the tank is above or below the sensor position. At time $k = 0$ the fault-tolerant control algorithm is informed about the current fault f, which has been applied to the tank system earlier and which has brought the tank levels to "wrong" values. Then the computer selects the control input according to the control law k_q.

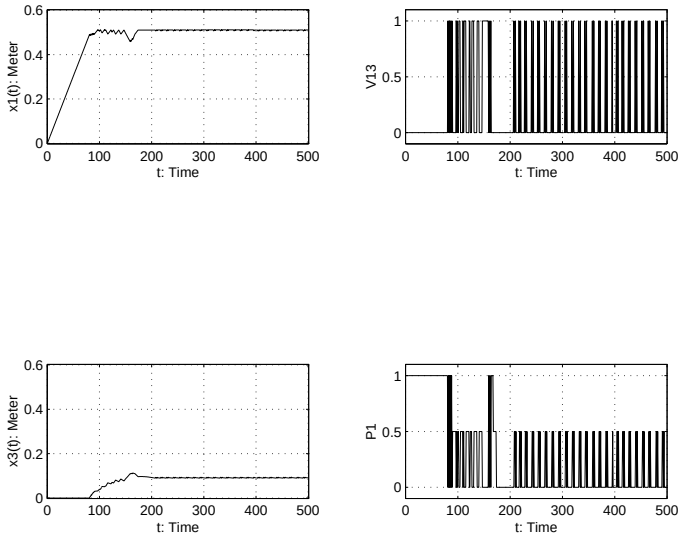

Fig. 10.5. Behaviour of the reconfigured system for fault f_1

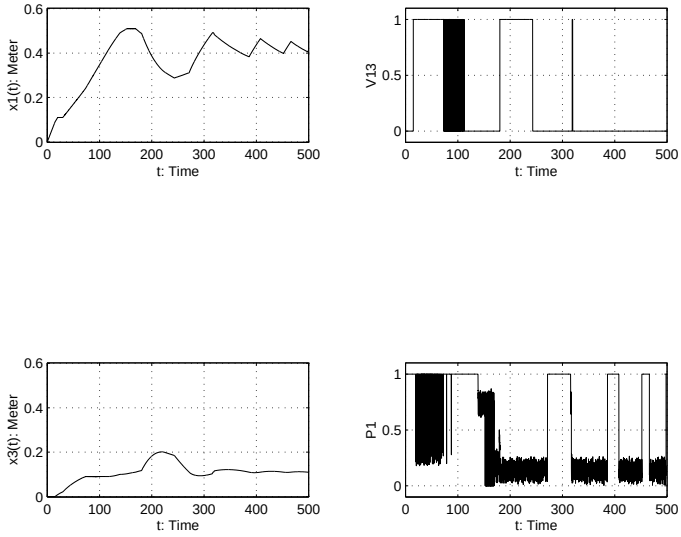

Fig. 10.6. Behaviour of the reconfigured system for fault f_2

The experimental results are shown in Fig. 10.5 for fault f_1 and the initial state $\boldsymbol{x}_0 = \mathbf{0}$, which corresponds to the extreme assumption that after the fault has occurred, the tank system is emptied until the fault has been identified. This extreme assumption is made to show the effect of the discrete controller, which works unter the influence of the fault. The controller uses the valve V_{12L} as new control input and brings the system into the required state within about $170\,\mathrm{s}$. In the experiment shown in Fig. 10.6, Fault f_2 occurred. The controller reduces the set point of the level controller of Tank T_1 to $[h_1^{\mathrm{ref}}] = medium$ and uses again the valve V_{12L} to bring the level of Tank T_2 to the required value $medium$.

10.2 Diagnosis and fault-tolerant control of a chemical process

In this section, fault diagnosis and fault-tolerant control are applied to two chemical processes, where in the first case a fault-tolerant temperature and level controller should be applied whereas in the aim of the second case is to attenuate disturbances concerning the conductivity and temperature of a liquid. Both problems are tackled by means of linearised models. The experiments with industrial equipment show impressive results with the methods developed in this book but also point to the restrictions of the fault tolerance if the process diverges considerably from the operation point and the nonlinearities shift the plant properties from the nominal ones.

10.2.1 Fault diagnosis by means of a discrete-event model

The experimental set-up used for the test of qualitative diagnostic methods is depicted in Figs. 10.7 and 10.8. Although the main modelling problems are posed by the stirred reactor, which is depicted in the middle of Fig. 10.9, the faults affect other parts of the whole system as well.

Figure 10.9 shows the part of the process that is considered further. An inflowing liquid is heated in the stirred tank reactor such that the outflowing liquid has a temperature of approximately $70^\circ\,\mathrm{C}$. The temperature is controlled by a dual-mode controller switching both heating elements simultaneously on or off such that the temperature in the reactor is held between 69 and $71^\circ\,\mathrm{C}$. The liquid level in the tank is controlled by means of a dual-mode controller that opens the outlet valve half or completely such that the level in the reactor varies between 30 and $40\,\mathrm{cm}$. In Fig. 10.13, the faultless behaviour of the process is depicted. The valves V_{20} and V_{90} are additional inputs which are not used under faultless conditions but may be used in fault-tolerant control. With these valves, water of temperature $20^o\,\mathrm{C}$ or $90^o\,\mathrm{C}$, respectively, can be put into the reactor.

Faults. In the following, three faults will be considered, which all block the controllers and, hence, necessitate a reconfiguration of the control algorithm:

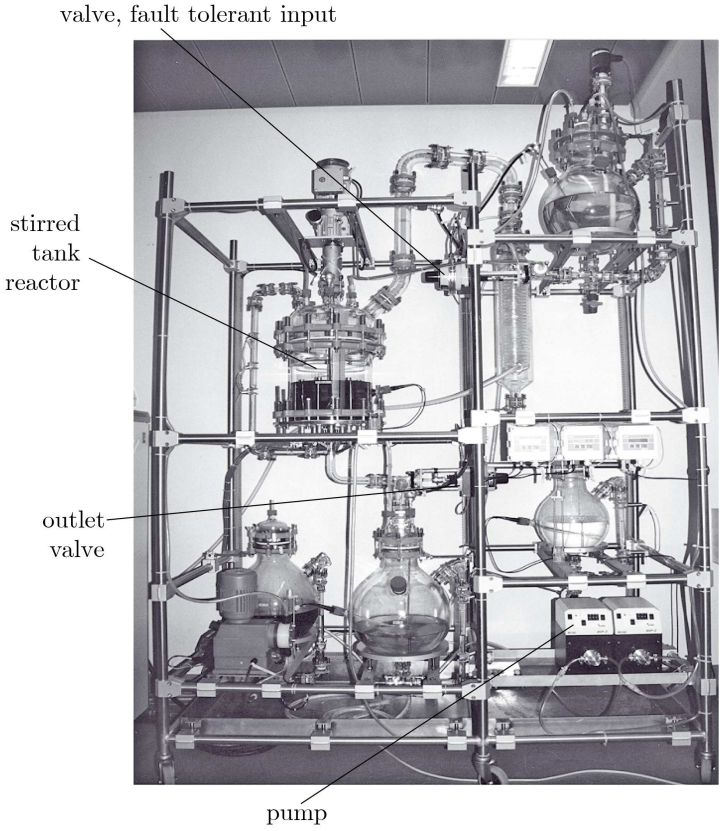

Fig. 10.7. The chemical plant TINA

- **Heating element fault.** If a single heating element ceases to operate, the heating power is insufficient to maintain a reactor temperature of 70° C.

- **Valve fault.** If the outlet valve is stuck in the completely opened position, after some time, the liquid level falls below 20 cm which is below the top of the heating elements and therefore causes a safety mechanism to turn off the heating and to deactivate the temperature control. Fault-tolerant control has to prevent the safety system from shutting off the plant.

- **Cooler fault.** The temperature T_{in} of the inflowing liquid is the output of another process including a cooler, which may fail. Under normal conditions, the temperature is 23° C. In case of the cooler fault, the temperature of the inflowing liquid rises to 90° C such that cooling rather than heating in the stirred reactor is necessary.

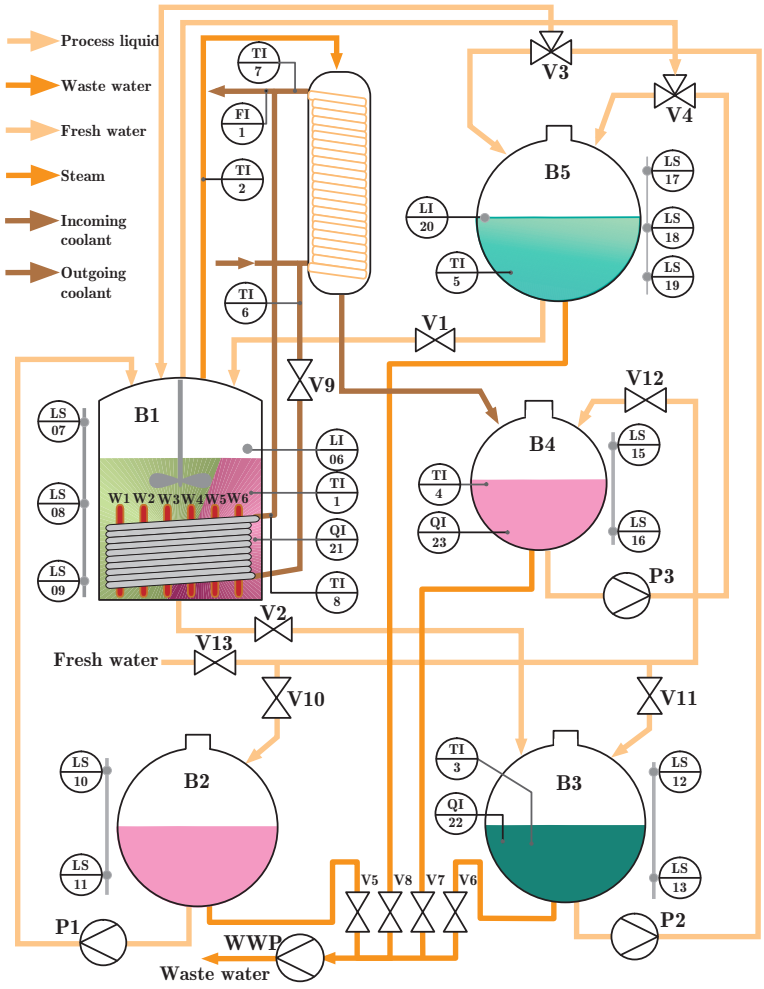

Fig. 10.8. Schematic diagram of the overall process

For the plant considered here, the first two faults are internal faults whereas the third fault is an external fault (cf. Chapter 3).

Plant model for diagnosis. For the demonstration of the diagnostic algorithm, the system under consideration is the stirred tank reactor combined with two dual-mode controllers. As the block diagram depicted in Fig. 10.10 shows, the plant has the four input signals T_{in}, $Pump$, V_{20} and V_{90}, whose interpretations are also given in Fig. 10.9. The dual-mode controller is considered as a part of the system whose faults should be diagnosed. The measured output is the temperature T and

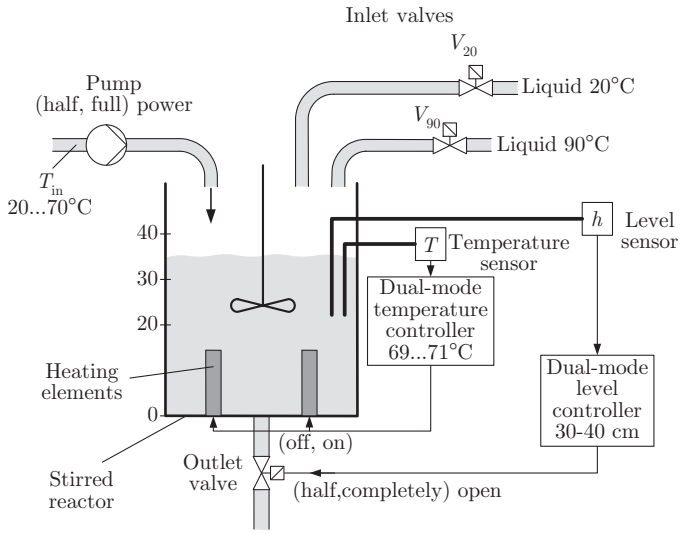

Fig. 10.9. Part of the process used for the experiments

the liquid level h of the reactor. The three faults are considered as additional inputs to the system.

The chemical plant has continuous as well as discrete signals. For example, the temperatures are continuous whereas the input signals generated by the dual-mode controllers are discrete. Therefore, the quantised systems approach explained in Chapter 9 is useful, because all signals are interpreted uniformly as discrete signals.

A state-space model of the reactor can be derived with the liquid level and the temperature as the two state variables. Two differential equations occur, where the first is analogous to the equation used in the tank example throughout this book and the second results from an enthalpy balance. These two differential equations have been combined with the switching conditions of the dual-mode controllers, which results in a hybrid model of the system to be diagnosed. Each dual-mode controller has two discrete state variables, which are considered to be immeasurable and should be observed. The overall state space consists of two continuous and two discrete state variables.

The qualitative model is abstracted from a quantitative model for a sampling time of $T_s = 120\,\mathrm{s}$. The liquid temperature and the level are the output signals. For the qualitative model, the partitioning of the continuous-variable subspace, which is identical to the partitioning of the output space, is depicted in Fig. 10.11. The size of the state sets around the set-point of $70^\circ\mathrm{C}$ is chosen smaller than in the other regions of the state space. Likewise, the input space used by the controller of the faultless system is partitioned. The additional input signals, which may be used by the fault-tolerant controller, are the positions of two valves V_{20} and V_{90}, which can be completely opened or closed, only.

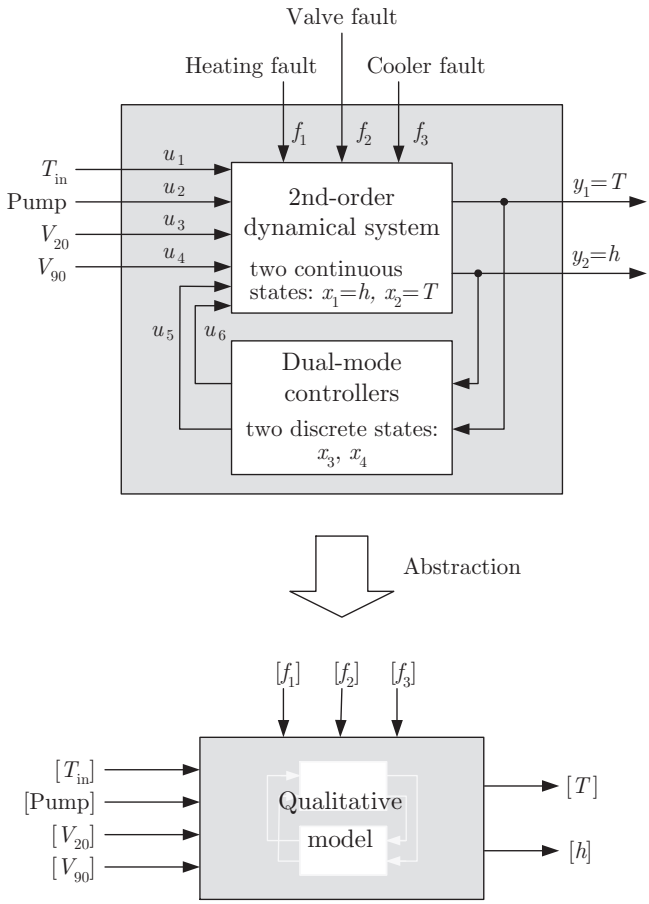

Fig. 10.10. Abstraction of the qualitative model

The result is a stochastic automaton with 400 states.

Experimental results for state observation. The observation algorithm described in Section 9.5.3 is applied to determine the discrete state of the dual-mode controllers. These discrete states coincide with the control input generated by the controllers, which switch the heating on and off and determine the position of the outlet valve. The reactor temperature $T(t)$ and the liquid level $h(t)$ are used as output measurements. The task considered here is to determine the dual-mode controller states from the quantised measurement information.

The input and output signals measured in an experiment with the faultless plant are depicted in Figs. 10.12 and 10.13. The first figure shows the pump power $Pump(t)$ and the input temperature $T_{\text{in}}(t)$. The two other input signals shown in Fig. 10.10 are constant. The measured reactor temperature T and liquid level h are

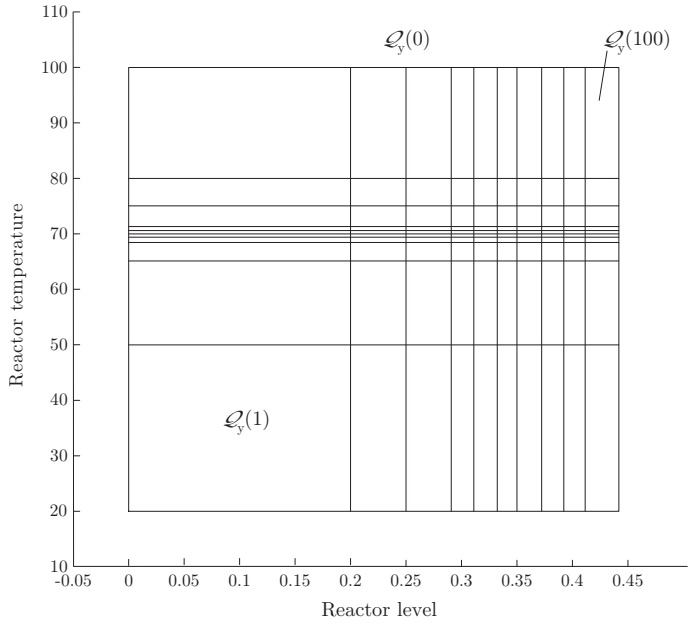

Fig. 10.11. Partitioning of the state space

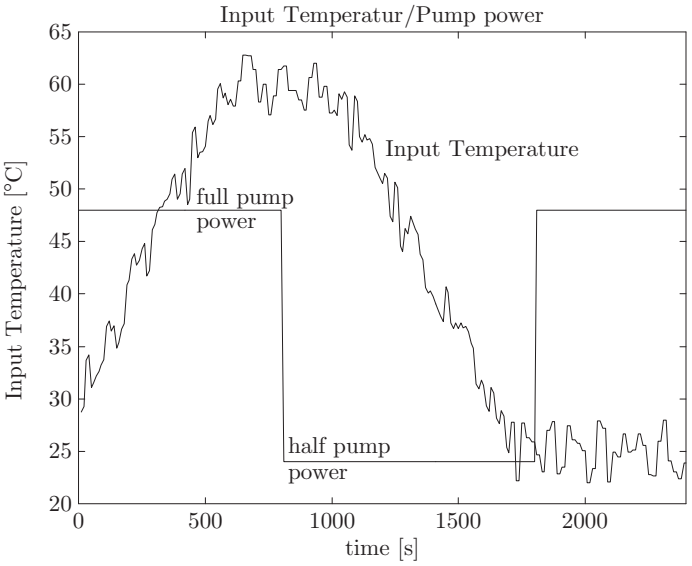

Fig. 10.12. Process input

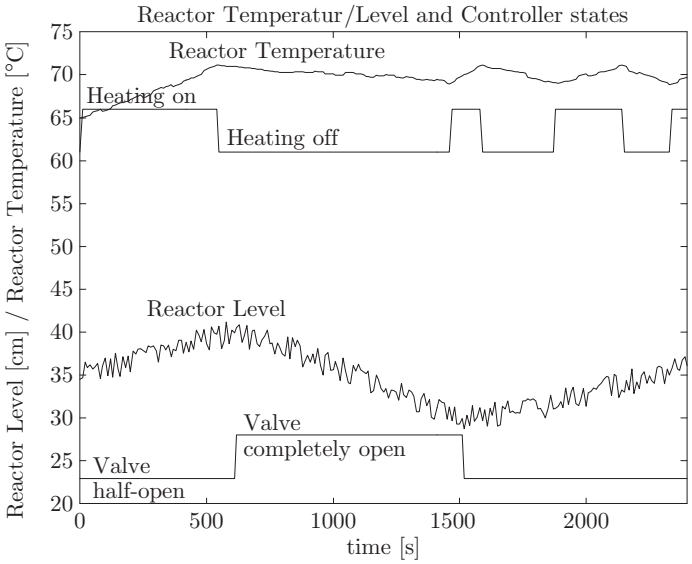

Fig. 10.13. Output measurements

depicted in Fig. 10.13. This figure also shows the true discrete states of the dual-model controllers, which correspond to the heating switching and the valve position. These discrete states have been assumed immeasurable and, therefore, to be reconstructed by the observation algorithm.

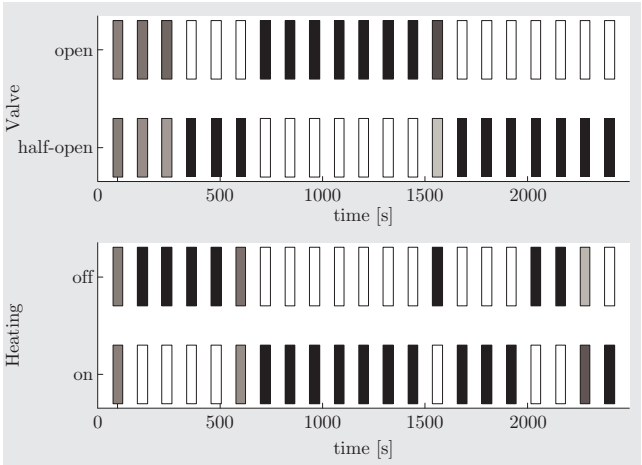

Fig. 10.14. Qualitative observation result

The observation result is shown in Fig. 10.14. The probabilities of the dual-mode controller states are depicted in grey scale. A dark colour represents a high and a light a low probability. All measurement information has been quantised according to the chosen signal space partitions before they are processed by the observation algorithm.

The grey rectangles for the initial time point show that the discrete controller states cannot be determined from the measurement obtained for the first time instant alone. The algorithm has been initialised with a uniform distribution over the qualitative states. At the second time step the heating position can be uniquely determined to be "off", whereas three quantised measurements are necessary to determine the valve position to be "half-open".

The quality of the observation can be evaluated by comparing the observation result with the true results shown in Fig. 10.13. It can be seen, that the observation algorithm provides a good approximation of the discrete controller states. This state observation is incorporated into the diagnostic algorithm described in the following.

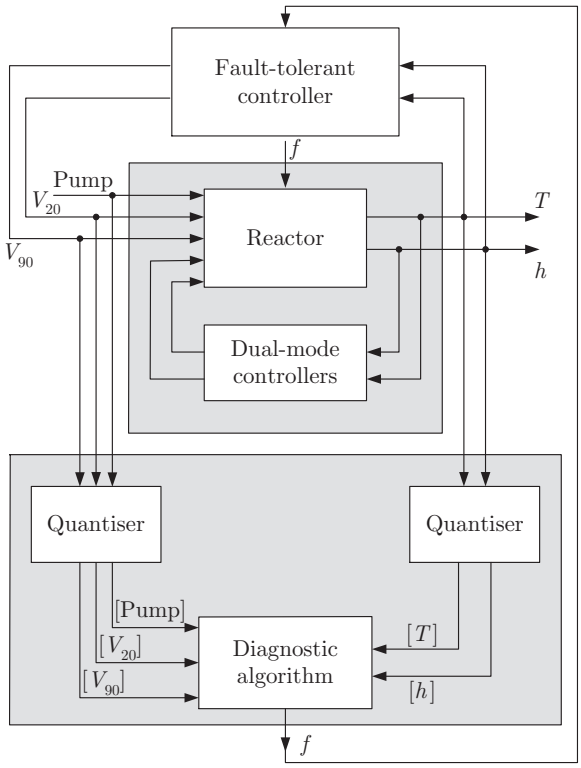

Fig. 10.15. Fault diagnosis of the reactor

Diagnostic results. The diagnostic algorithm developed in Section 9.6 is applied. As a heating fault means that the temperature control loop is opened and, hence, the control aim can no longer be met, the reactor was equipped with a fault-tolerant controller that closes a loop around the system considered so far. It uses the reactor temperature and liquid level as system output and the valves V_{20} and V_{90} as control input (Fig. 10.15). The controller gets the diagnostic result as further information. The additional controller is used here to hold the reactor in a long time interval inside its region of acceptable performance. The experiment is made to demonstrate the diagnostic algorithm.

As the diagnosis also concerns the cooler fault, the input temperature T_{in} is not used as measurement.

Figure 10.16 shows the experimental results. The upper part of the figure includes the immeasurable input temperature T_{in} and the pump power $Pump$ and the middle part depicts the reactor temperature T and the liquid level h together with the input signals V_{20} and V_{90} generated by the fault-tolerant controller, which bring about an additional cold liquid inflow or hot liquid inflow, respectively.

The first fault is a break-down of a heating element at time 800 seconds. As the second fault, at time 2700 seconds, the cooler breaks down, which leads to the increasing temperature depicted in the upper part of the figure.

Figure 10.17 shows the diagnostic result. The algorithm is started in the start-up phase of the process, when the reactor temperature was still too low and the fault-tolerant controller opens the valve V_{90} to let hot liquid into the reactor to increase the temperature. The diagnostic algorithm provides the fault probabilities, which are also depicted in Fig. 10.17. It can be seen, that both faults, which also occur in combination, are detected quickly and uniquely, and the valve fault is explicitly excluded.

After approximately ten minutes, the heating element fault occurs. It can be seen in Fig. 10.17 that the diagnostic algorithm finds this fault at once. To prevent the reactor from leaving its region of acceptable performance, the fault-tolerant controller opens the valve V_{90} to let hot liquid into the reactor in order to stabilise the temperature despite of the reduced heating power. As a result, the experiment can be continued.

After one hour, in addition to the heating element fault, a cooler fault occurs. The input temperature starts to rise to $90°C$. Note that the diagnostic algorithm is not supplied with the depicted information about the temperature, but assumes that the input temperature is low (i.e. between 20 and $30°C$). However, the diagnostic algorithm detects the fault (Fig. 10.17). By using this information, the fault-tolerant controller starts adding cold liquid by opening valve V_{20} and thus returns the process into the region of acceptable performance.

10.2.2 Reconfiguration of a level and temperature control loop

For a demonstration of the control reconfiguration in case of an actuator failure the part of the chemical process shown in Fig. 10.18 is considered. The control

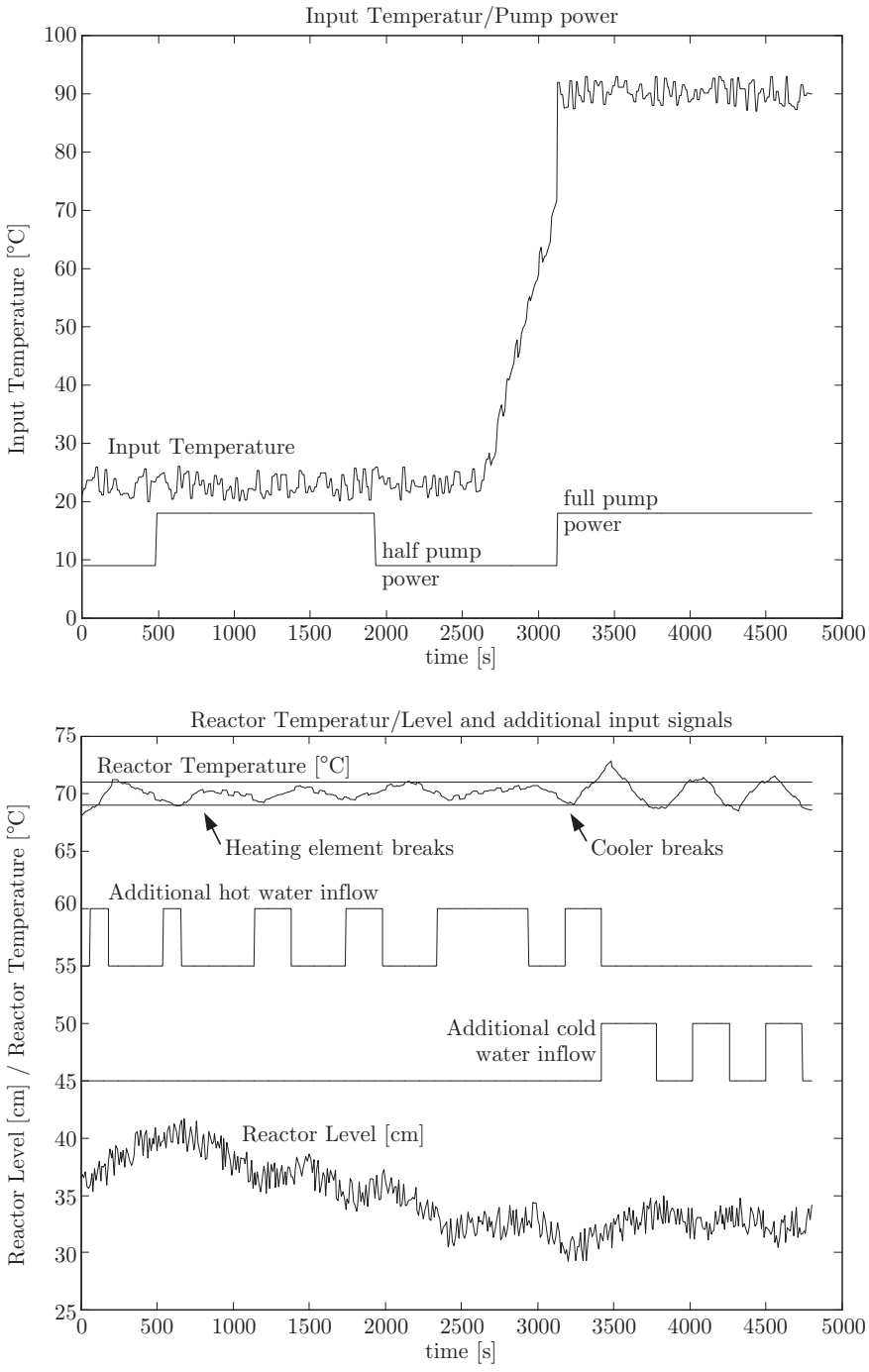

Fig. 10.16. Process input generated by the fault-tolerant controller and output measurements

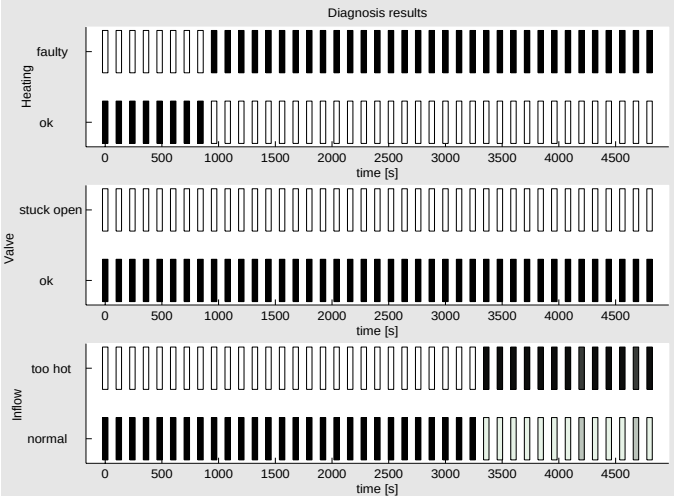

Fig. 10.17. Diagnostic result

objectives are to maintain a constant liquid level and a constant temperature in the reactor tank B_1 and, thus, producing a constant product outflow. To achieve this, hot and cold liquid can be brought into the reactor from Tanks B_2 and B_5. The main reactor B_1 can be heated and cooled.

In the nominal case the liquid level is controlled by adjusting the cold liquid inflow from Tank B_5 and the temperature by means of the heating.

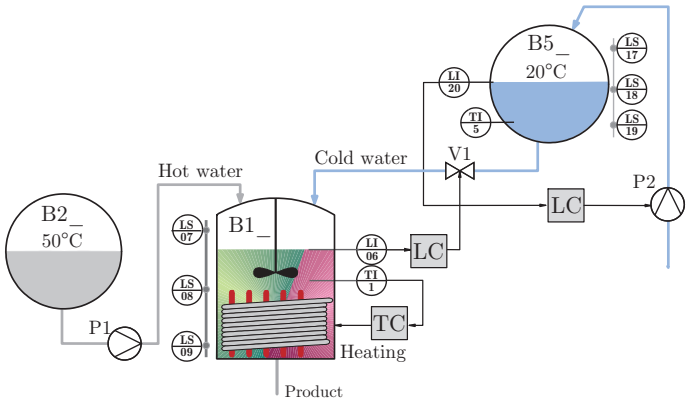

Fig. 10.18. Plant used for control reconfiguration (LC - level control, TC - temperature control

Plant model. The plant model contains three states: the reactor content V_{B1}, the reactor temperature ϑ_{B1} and the content of the cold liquid tank V_{B5}. From a mass balance, the following equations are obtained

$$\begin{aligned}
\dot{V}_{B5}(t) &= k_{P2}u_{P2}(t) - q_{51}(t) \\
\dot{V}_{B1}(t) &= q_{21}(t) + q_{51}(t) - q_{1out}(t) \\
\dot{\vartheta}_{B1}(t) &= (\vartheta_{B2}(t) - \vartheta_{B1}(t))\frac{q_{21}(t)}{V_{B1}(t)} + (\vartheta_{B5}(t) - \vartheta_{B1}(t))\frac{q_{51}(t)}{V_{B1}(t)} \\
&\quad + \frac{u_{heat}(t)k_{heat}}{V_{B1}(t)},
\end{aligned}$$

where for the liquid flows the relations

$$\begin{aligned}
q_{21}(t) &= k_{P1}u_{P1}(t) \\
q_{51}(t) &= k_{V1}\,124.5^{u_{V1}(t)}\sqrt{h_{B5}(t) + 1.07} \\
q_{1out}(t) &= k_{V2}\sqrt{\frac{V_{B1}(t)}{A_{B1}} + 1.4}
\end{aligned}$$

hold. $h_{B5}(t)$ is the liquid level in the spherical tank B_5, $u_{heat}(t)$ the heating power, k_{heat} a heating coefficient, $u_{P1}(t)$, $u_{P2}(t)$ and $u_{V1}(t)$ the control input to the two pumps and to the Valve V_1 and A_{B1} the cross-section area of the Tank B_1. After a linearisation of this nonlinear model around the operating point of $\vartheta_{B1} = 40^o\mathrm{C}$, the following linear model is obtained:

$$\begin{aligned}
\begin{pmatrix} \dot{V}_{B5}(t) \\ \dot{V}_{B1}(t) \\ \dot{\vartheta}_{B1}(t) \end{pmatrix} &= 10^{-3}\begin{pmatrix} -0.46 & 0 & 0 \\ +0.46 & -0.33 & 0 \\ -0.48 & 0.008 & -1.1 \end{pmatrix}\begin{pmatrix} V_{B5}(t) \\ V_{B1}(t) \\ \vartheta_{B1}(t) \end{pmatrix} \\
&\quad + \begin{pmatrix} 0.09 & -0.023 & 0 & 0 \\ 0 & +0.023 & +0.05 & 0 \\ 0 & -0.024 & +0.02 & 0.223 \end{pmatrix}\begin{pmatrix} u_{P2}(t) \\ u_{V1}(t) \\ u_{P1}(t) \\ u_{heat}(t) \end{pmatrix} \\
\boldsymbol{y} &= \begin{pmatrix} h_{B5}(t) \\ h_{B1}(t) \\ \vartheta_{B1}(t) \end{pmatrix}.
\end{aligned}$$

The nominal proportional controllers are defined by:

$$\begin{aligned}
u_{V1}(t) &= -0.5\,V_{B1}(t) \\
u_{heat}(t) &= -0.5\,\vartheta_{B1}(t) \\
u_{P2}(t) &= -1\,V_{B5}(t).
\end{aligned}$$

They can be represented as

$$\boldsymbol{u}(t) = -\boldsymbol{K}\boldsymbol{y}(t)$$

with

$$\boldsymbol{K} = \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}.$$

Note that these controllers do not use the control input u_{P1}, because the matrix $\boldsymbol{K}$ has a vanishing third row.

Faults. Several severe faults can occur that open the control loops. For example, due to a heating failure, the reactor can no longer be heated, or clogging or blockage of Valve V_1 can bring the level controller out of operation. In the following the heating failure and a blockage of Valve V_1 in its nominal position will be considered.

Controller reconfiguration after a heating failure. After a heating failure has occurred, the temperature controller

$$u_{heat}(t) = -0.5\,\vartheta_{B1}(t)$$

has no influence on the process. The system in the nominal and the faulty case has the matrices

$$\begin{aligned} \boldsymbol{B} &= \begin{pmatrix} 0.09 & -0.023 & 0 & 0 \\ 0 & +0.023 & +0.05 & 0 \\ 0 & -0.024 & +0.02 & 0.223 \end{pmatrix} \\ \boldsymbol{B}_f &= \begin{pmatrix} 0.09 & -0.023 & 0 & 0 \\ 0 & +0.023 & +0.05 & 0 \\ 0 & -0.024 & +0.02 & 0 \end{pmatrix}, \end{aligned}$$

which distinguish in the last column. Both matrices have the same rank and can be related to one another by the matrix

$$\boldsymbol{N} = \begin{pmatrix} 1 & 0 & 0 & -1.72 \\ 0 & 1 & 0 & -6.72 \\ 0 & 0 & 1 & 3.09 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

such that the equation

$$\boldsymbol{B}_f \boldsymbol{N} = \boldsymbol{B}$$

holds. Hence a complete reconfiguration is possible by using the third control input, which is not used in the nominal case. The reconfigured controller

$$\boldsymbol{u}(t) = -\boldsymbol{N}\boldsymbol{K}\boldsymbol{y}(t)$$

has the controller matrix

$$\boldsymbol{NK} = \begin{pmatrix} 0.5 & 0 & -0.86 \\ 0 & 1 & -3.36 \\ 0 & 0 & 1.55 \\ 0 & 0 & 0 \end{pmatrix}.$$

Obviously, the fourth actuator is no longer used. The effect of this actuator is distributed among the three remaining actuators, which can be seen in the last column of the new controller matrix. With the reconfigured controller, the behaviour of the nominal system is completely reproduced.

Controller reconfiguration by means of a virtual actuator. The loss of the actuator V_1 does not affect the operation point, but it breaks the level control loop for the reactor B_1. The use of a reduced virtual actuator allows to keep the nominal controller while changing the control structure as little as possible.

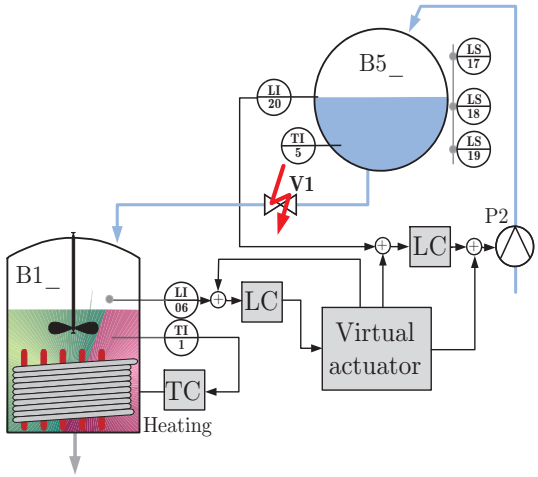

Fig. 10.19. Reconfigured controller including a virtual actuator

In the terminology of Section 7.5.3, the directly influencable part $\boldsymbol{x}_{F1}$ of the plant state is defined by V_{B5} and ϑ_{B1}, while $\boldsymbol{x}_{F2}$ is the single state variable V_{B1}:

$$\boldsymbol{x}_{f1}(t) = \begin{pmatrix} B_{B5}(t) \\ \vartheta_{B5}(t) \end{pmatrix}, \qquad \boldsymbol{x}_{f2}(t) = V_{B1}(t).$$

The $(1,\,2)$-parameter matrix $\boldsymbol{M}$ is determined by pole placement. The element of $\boldsymbol{M}$ that is acting on ϑ_{B1} has no influence on the actuator pole and is, therefore, set to 0. The other value is chosen so that the actuator pole lies at -0.004 in order to make the influence of the virtual actuator on the closed-loop dynamics as small as possible. The application of the method explained in Section 7.5.4 to this example leads to

$$
\begin{aligned}
\dot{\hat{x}}_2(t) &= -0.004\,\hat{x}_2(t) + 0.0229\,u_{V2,R}(t) \\
\hat{\boldsymbol{u}}(t) &= \begin{pmatrix} 0.015 \\ -0.318 \\ 0 \end{pmatrix} \hat{x}_2(t) + \begin{pmatrix} -0.107 \\ 1.78 \\ 0 \end{pmatrix} u_{V2,R}(t) \\
\hat{\boldsymbol{y}}(t) &= \begin{pmatrix} -8 \\ 0 \\ 1 \end{pmatrix} \hat{x}_2(t) \ .
\end{aligned}
$$

The function of the reduced virtual actuator can be described as follows (Fig. 10.19). The input $u_{V1}(t)$ is not available to control the inflow into the main reactor, but this inflow also depends on the level in Tank B_5 and, hence, on V_{B5}. In order to reach the same effect as the broken actuator, $V_{B5}(t)$ is increased or decreased by influencing the Pump P_2 via the input $u_{P2}(t)$. As $V_{B5}(t)$ cannot be changed instantaneously, this "replacement action" is slower than the direct action of the nominal control loop on the valve V_1 and leads to a slower reaction of the system under the influence of the reconfigured controller.

In mathematical terms, the virtual actuator brings about an additional pole which yields the slower dynamics. The difference between the nominal and the new behaviour is determined by the virtual actuator and deducted from the measurements of $V_{B1}(t)$ and $V_{B5}(t)$. In this way, the additional pole remains hidden from the level controller and this controller acts like in the nominal case.

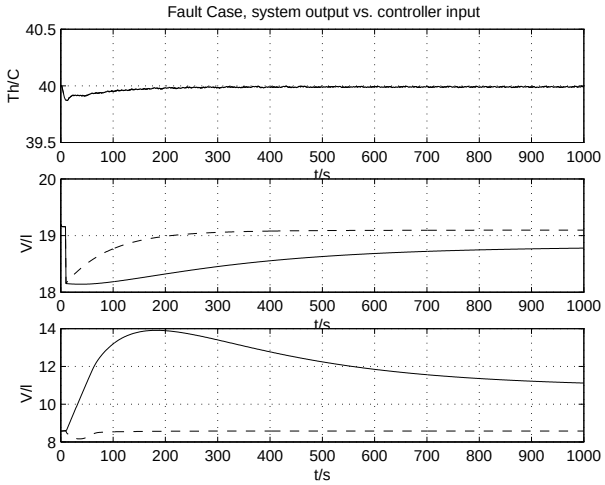

Fig. 10.20. Results of the reconfiguration experiment (Reactor temperature $\vartheta_{B1}(t)$ (top), reactor content $V_{B1}(t)$ (middle) and reactor content $V_{B5}(t)$

The experimental results are shown in Fig. 10.20. The state $V_{B1}(t)$ is disturbed by withdrawing a considerable amount of liquid until time $t = 10\,\mathrm{s}$. The virtual actuator increases the level $V_{B5}(t)$ in Tank B_5 by increasing the pump input $u_{P1}(t)$. The effect of this manipulation and of the fault is "'simulated"' by the virtual actuator, subtracted from the sensors data and, therefore, hidden from the nominal controller. After 180 seconds the tank level $V_{B5}(t)$ reaches its maximum and after another 800 seconds the state deviation has been reasonably compensated. A static deviation remains because of some modelling inaccuracies.

The dashed lines show the behaviour of the faultless closed-loop system. The slower reaction of the level controller results in the slower disturbance attenuation shown in the middle part of the figure, where the nominal system reaches the set-point of $19\,\mathrm{dm}^3$ quicker than the reconfigured system. Hence, the operation of the main reactor can be restored with a minor performance degradation.

In the lower part of the figure the different behaviour of Tank B_5 can be seen. The difference is due to the different functions that this tank has in both situations. In the faultless case the level controller of this tank adjusts the liquid content to the set-point, whereas under faulty conditions this variable is used as a means to control the inflow into Tank B_1 and, thus, to control the contents of B_1.

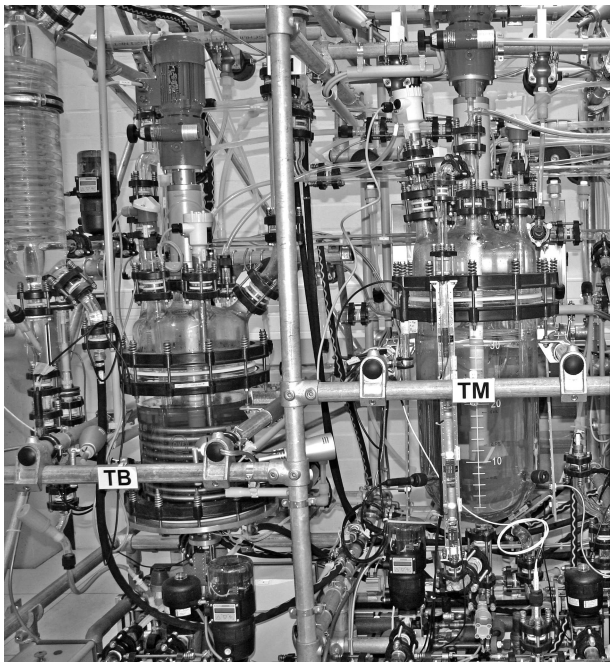

Fig. 10.21. Part of the chemical plant VERA used in the experiment

10.2.3 Reconfiguration of a conductivity control loop

The second application of the reconfiguration method that uses the virtual actuator is the fault-tolerant control of the conductivity of a liquid. Figure 10.21 shows the experimental set-up and Fig. 10.22 the schematic diagram of the three reactors involved in the control loop considered. The sequence of the two Reactors TM and TB with the Reactor TS is used to produce a liquid with prescribed temperature and conductivity. Several control loops have to be used, which are shown in the schematic diagram with the abbreviations LC for level controller, TC for temperature controller and CC for concentration controller. If actuator failures occur, these loops are brought out of operation. Typical failures concern the valves V_{TM} and V_{CW} and the heating P_{el}.

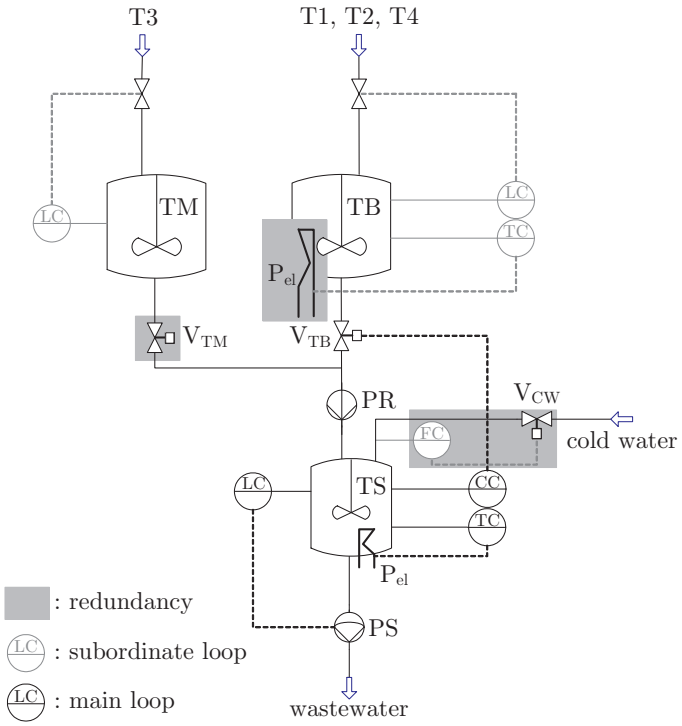

Fig. 10.22. Schematic diagram of the process

The nominal controller uses the inputs u_{PS}, u_{TS} and u_{TB}, which are subject to the three failures. The three variables to be controlled are the temperature ϑ_{TB}, the liquid level l_{TS} in the Reactor TS and the conductivity λ_{TS} of the liquid in the Reactor TS (Fig. 10.23). The block diagram also shows the redundant inputs u_{CW} and u_{TM}, which will be used for the reconfiguration.

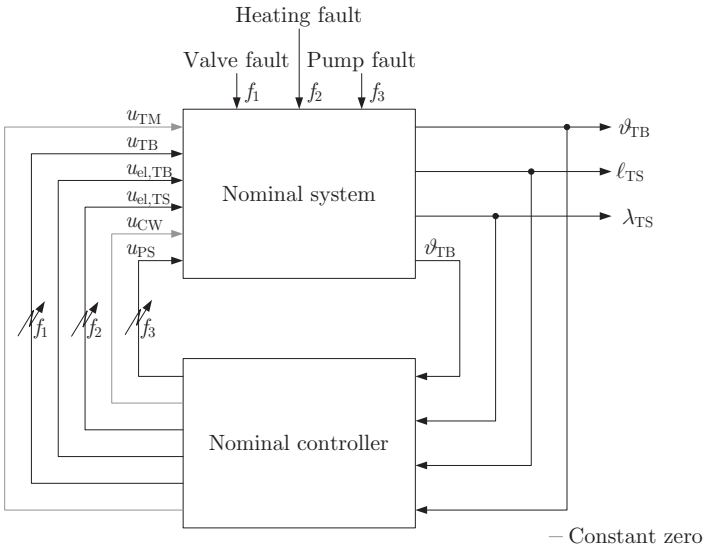

Fig. 10.23. Schematic diagram of the process

Nonlinear model. The following nonlinear model is obtained from balance equations that concern the different components of the plant. To shorten the notation of the equations, the dependency of the signals from the time t is omitted:

• Change of the liquid temperature of Reactor TS:

$$\dot{\vartheta}_{TS} = \frac{1}{A_{TS}\rho l_{TS}}\left\{\frac{P_{el,TS}-\dot{Q}_{PL,TS}}{c_p}+\dot{m}_{TB}(\vartheta_{TB}-\vartheta_{TS})+\dot{m}_{TM}(\vartheta_{TM}-\vartheta_{TS})+\dot{m}_{CW}(\vartheta_{CW}-\vartheta_{TS})\right\}$$

• Change of the liquid volume in Reactor TS:

$$\dot{l}_{TS}(t)=\frac{\dot{m}_{TB}(t)+\dot{m}_{TM}(t)+\dot{m}_{CW}(t)-\dot{m}_{TW}(t)}{A_{TS}\rho}$$

• Change of the concentration in Reactor TS:

$$\dot{c}_{TS}(t) = \frac{\dot{m}_{TB}(t)(c_{TB}-c_{TS}(t))+\dot{m}_{TM}(t)(c_{TM}-c_{TS}(t))-\dot{m}_{CW}(t)c_{TS}(t)}{A_{TS}\rho l_{TS}(t)}$$

• Change of the liquid temperature in Reactor TB:

$$\dot{\vartheta}_{TB}(t) = \frac{1}{A_{TB}\rho l_{TB}}\left\{\frac{P_{el,TB}(t)-\dot{Q}_{PL,TB}(t)}{c_p}+\dot{m}_{T124}(t)(\vartheta_{T124}-\vartheta_{TB}(t))\right\}$$

• Behaviour of the cold water Valve V_{CW}:

$$\begin{aligned} \dot{x}_{CW}(t) = &= -\frac{1}{T_{CW}} x_{CW}(t) + \frac{1}{T_{CW}} u_{CW}(t) \\ \dot{m}_{CW}(t) &= x_{CW}(t) \qquad \text{with } T_{CW} = 3,7\,\mathrm{s} \end{aligned}$$

• Actuator dynamics of the heating of the Reactor TB:

$$\begin{aligned} \dot{x}_{TB}(t) = &= -\frac{1}{T_{el,TB}} x_{TB}(t) + \frac{1}{T_{el,TB}} u_{TB}(t) \\ P_{el,TB}(t) &= k_{TB} x_{TB}(t), \\ &\quad \text{with } T_{el,TB} = 27\,\mathrm{s}, \;\; k_{TB} = 18\,\mathrm{kW} \end{aligned}$$

• Actuator dynamics of the heating of the Reactor TS:

$$\begin{aligned} \dot{x}_{TS}(t) = &= -\frac{1}{T_{el,TS}} x_{TS}(t) + \frac{1}{T_{el,TS}} u_{TS}(t) \\ P_{el,TS}(t) &= k_{TS} x_{TS}(t), \\ &\quad \text{with } T_{el,TS} = 65\,\mathrm{s}, \;\; k_{TS} = 4\,\mathrm{kW} \end{aligned}$$

Besides the state variables ϑ_{TB} and l_{TS}, the conductivity is the third variable to be controlled. This signal is obtained by the following relation:

$$\lambda_{TS}(t) = 0,4469 \frac{\mathrm{mS}}{\mathrm{cm}} + 2047,7 \frac{\mathrm{mS}}{\mathrm{cm}} c_{TS}(t).$$

All these equations use the following mass and heat flows:

• Mass flow from Rector TB towards Reactor TS:

$$\dot{m}_{TB}(t) = \begin{cases} \left(0,019 \frac{\mathrm{kg}}{\mathrm{s}\sqrt{\mathrm{m}}} + 0,727 \frac{\mathrm{kg}}{\mathrm{s}\sqrt{\mathrm{m}}} (u_{TB}(t) - 0,13)\right) \sqrt{l_{TB} + 0,3\,\mathrm{m}}, \\ \qquad\qquad \text{if } u_{TB} \geq 0,13 \\ 0 \frac{\mathrm{kg}}{\mathrm{s}}, \\ \text{else} \end{cases}$$

• Mass flow from Rector TM towards Reactor TS:

$$\dot{m}_{TM}(t) = \begin{cases} \left(0,047 \; \frac{\mathrm{kg}}{\mathrm{s}\sqrt{\mathrm{m}}} + 0{,}605 \; \frac{\mathrm{kg}}{\mathrm{s}\sqrt{\mathrm{m}}} (u_{TM}(t) - 0,04)\right) \sqrt{l_{TM} + 0{,}3\,\mathrm{m}} \\ \qquad\qquad \text{if } u_{TM} \geq 0{,}04 \\ 0 \frac{\mathrm{kg}}{\mathrm{s}} \qquad\qquad \text{else.} \end{cases}$$

• Mass flow out of the Reactor TS:

$$\dot{m}_{PS}(t) = \dot{m}_{TW}(t) = 0,1679 \frac{\mathrm{kg}}{\mathrm{s}\sqrt{\mathrm{m}}} u_{PS}(t) \sqrt{l_{TS}(t) + 0,36\,\mathrm{m}}$$

• Heat balance of the Reactor TS:

$$\dot{Q}_{PL,TS}(\vartheta_{TS}(t)) = \begin{cases} \dot{Q}_{PL,TS,on}(\vartheta_{TS}(t)), & \text{if heating is on} \\ \dot{Q}_{PL,TS,off}(\vartheta_{TS}(t)), & \text{if heating is off} \end{cases}$$

with

$$\dot{Q}_{PL,TS,on}(\vartheta_{TS}(t)) = \begin{cases} 46,9403\frac{\mathrm{W}}{^\circ\mathrm{C}}(\vartheta_{TS}(t) - 22,5\,^\circ\mathrm{C}), & \text{if } \vartheta_{TS} \geq 22,5\,^\circ\mathrm{C} \\ 0\,\mathrm{W}, & \text{if } \vartheta_{TS} < 22,5\,^\circ\mathrm{C} \end{cases}$$

$$\dot{Q}_{PL,TS,off}(\vartheta_{TS}(t)) = \begin{cases} 4,8968\frac{\mathrm{W}}{^\circ\mathrm{C}}(\vartheta_{TS}(t) - 22,5\,^\circ\mathrm{C}), & \text{if } \vartheta_{TS} \geq 22,5\,^\circ\mathrm{C} \\ 0\,\mathrm{W}, & \text{if } \vartheta_{TS} < 22,5\,^\circ\mathrm{C} \end{cases}$$

• Heat balance of the Reactor TB:

$$\dot{Q}_{PL,TB}(\vartheta_{TB}(t)) = \begin{cases} \dot{Q}_{PL,TB,on}(\vartheta_{TB}(t)), & \text{if heating is on} \\ \dot{Q}_{PL,TB,off}(\vartheta_{TB}(t)), & \text{if heating is off} \end{cases}$$

$$\dot{Q}_{PL,TB,on}(\vartheta_{TB}(t)) = \begin{cases} 135,468\frac{\mathrm{W}}{^\circ\mathrm{C}}(\vartheta_{TB}(t) - 22,5\,^\circ\mathrm{C}), & \text{if } \vartheta_{TB} \geq 22,5\,^\circ\mathrm{C} \\ 0\,\mathrm{W}, & \text{if } \vartheta_{TB} < 22,5\,^\circ\mathrm{C}. \end{cases}$$

$$\dot{Q}_{PL,TB,off}(\vartheta_{TB}(t)) = \begin{cases} 4,8968\frac{\mathrm{W}}{^\circ\mathrm{C}}(\vartheta_{TB}(t) - 22,5\,^\circ\mathrm{C}), & \text{if } \vartheta_{TB} \geq 22,5\,^\circ\mathrm{C} \\ 0\,\mathrm{W}, & \text{if } \vartheta_{TB} < 22,5\,^\circ\mathrm{C} \end{cases}$$

The given equations can be lumped together to get a nonlinear state-space model (9.3), (9.4)

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{g}(\boldsymbol{x}(k), \boldsymbol{u}(k)), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(k) &= \boldsymbol{h}(\boldsymbol{x}(k), \boldsymbol{u}(k)) \end{aligned}$$

with the state, input and output vectors

$$\boldsymbol{x}(t) = \begin{pmatrix} \vartheta_{TS}(t) \\ l_{TS}(t) \\ c_{TS}(t) \\ \vartheta_{TB}(t) \\ x_{CW}(t) \\ x_{TB}(t) \\ x_{TS}(t) \end{pmatrix}, \quad \boldsymbol{u}(t) = \begin{pmatrix} u_{TM}(t) \\ u_{TB}(t) \\ u_{TB}(t) \\ u_{TS}(t) \\ u_{CW}(t) \\ u_{PS}(t) \end{pmatrix}, \quad \boldsymbol{y}(t) = \begin{pmatrix} \vartheta_{TS}(t) \\ l_{TS}(t) \\ \lambda_{TS}(t) \\ \vartheta_{TB}(t) \end{pmatrix}.$$

Linearised model. A linearised state-space model

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{aligned}$$

is obtained from the nonlinear model with the following matrices:

$$\boldsymbol{A} = 10^{-3} \cdot \begin{pmatrix} -3,46 & 0 & 0 & 1,46 & -59,12 & 0 & 39,36 \\ 0 & -0,76 & 0 & 0 & 1,41 & 0 & 0 \\ 0 & 0 & -3,15 & 0 & -0,0034 & 0 & 0 \\ 0 & 0 & 0 & -1,34 & 0 & 157,46 & 0 \\ 0 & 0 & 0 & 0 & -270,27 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -37,03 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -15,38 \end{pmatrix}$$

$$\boldsymbol{B} = 10^{-3} \cdot \begin{pmatrix} -10,62 & 0 & 0 & 0 & 0 & 0 \\ 7,11 & 8,49 & 0 & 0 & 0 & -1,98 \\ 0,0249 & 0,0235 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 270,27 & 0 \\ 0 & 0 & 37,03 & 0 & 0 & 0 \\ 0 & 0 & 0 & 15,38 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{C} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2047,7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{D} = \boldsymbol{O}.$$

The set of eigenvalues of the matrix $\boldsymbol{A}$

$$\sigma = \{-0,2703; -0,0370; -0,0154; -0,0035; -0,0032; -0,0013; -0,0008\}$$

gives an impression of the dynamical properties of the plant.

Models of the faulty system. The three actuator failures cause a change of the matrix $\boldsymbol{B}$ of the linearised state-space model:

• Failure f_1 of the Valve V_{TB}, which gets the input signal u_{TB}:

$$\boldsymbol{B}_{f_1} = 10^{-3} \cdot \begin{pmatrix} -10,62 & 0 & 0 & 0 & 0 & 0 \\ 7,11 & 0 & 0 & 0 & 0 & -1,98 \\ 0,0249 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 270,27 & 0 \\ 0 & 0 & 37,03 & 0 & 0 & 0 \\ 0 & 0 & 0 & 15,38 & 0 & 0 \end{pmatrix}$$

• Failure f_2 of the heating of the Reactor TS, which acts according to the control input u_{TS}:

$$\boldsymbol{B}_{f_2} = 10^{-3} \cdot \begin{pmatrix} -10,62 & 0 & 0 & 0 & 0 & 0 \\ 7,11 & 8,49 & 0 & 0 & 0 & -1,98 \\ 0,0249 & 0,0235 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 270,27 & 0 \\ 0 & 0 & 37,03 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

• Failure f_3 of the Pump PS, which runs according to the control input u_{PS}:

$$\boldsymbol{B}_{f_3} = 10^{-3} \cdot \begin{pmatrix} -10,62 & 0 & 0 & 0 & 0 & 0 \\ 7,11 & 8,49 & 0 & 0 & 0 & 0 \\ 0,0249 & 0,0235 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 270,27 & 0 \\ 0 & 0 & 37,03 & 0 & 0 & 0 \\ 0 & 0 & 0 & 15,38 & 0 & 0 \end{pmatrix}.$$

These matrices differ from the matrix $\boldsymbol{B}$ for the nominal model with respect to one column each, which is set to zero for the failed actuator.

Control reconfiguration by a virtual actuator. For all three fault cases, the virtual actuator described in Definition 7.7 is used for the control reconfiguration (Fig 10.24). The scheme is the same in all cases, only the matrix $\boldsymbol{B}_f$, which is a parameter of the virtual actuator, differs. This shows that the control reconfiguration is completely automatic in the sense that a general reconfiguration algorithm can be applied, which adapts the effect of the nominal controller to the failure that has occurred.

The first experiment concerns the reconfiguration with the goal to retain the stability of the closed-loop system. For this task, a virtual actuator with parameter matrix $\boldsymbol{N} = \boldsymbol{O}$ is used.

In case of the failure of the Valve V_{TB}, the virtual actuator has been designed to have the following set of eigenvalues:

$$\begin{aligned} \sigma_{VA} &\stackrel{!}{=} 25\sigma \\ &= \{-6,7568; -0,9259; 0,3846; -0,0866; -0,0790; -0,0335; -0,0190\} \end{aligned} \tag{10.9}$$

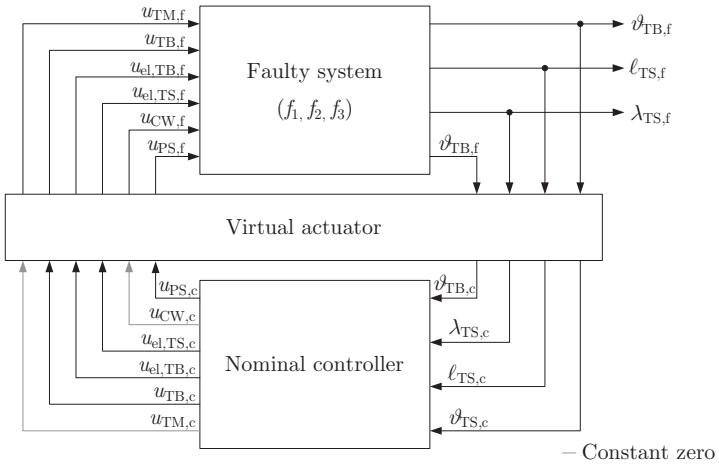

Fig. 10.24. Reconfiguration by means of a virtual actuator

This is accomplished by the feedback matrix

$$\boldsymbol{M} = \begin{pmatrix} -12{,}31 & -16{,}05 & 77{,}63 & 0{,}15 & 5{,}11 & 0{,}40 & -3{,}71 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 13{,}39 & -0{,}01 & 5770 & 90{,}07 & -23{,}71 & 178{,}06 & 15{,}41 \\ 17{,}18 & -0{,}06 & 7332 & 23{,}26 & -31{,}85 & 39{,}14 & 25{,}31 \\ -1{,}48 & -0{,}01 & -642{,}30 & -2{,}04 & 2{,}11 & -3{,}43 & -1{,}81 \\ -130{,}19 & -192{,}04 & 239{,}73 & 0{,}75 & 18{,}61 & 2{,}21 & -12{,}85 \end{pmatrix}.$$

This pole assignment is possible, because the pair $(\boldsymbol{A}, \boldsymbol{B}_{f1})$ is completely controllable. The eigenvalues are chosen with respect to the eigenvalues of the plant. They make the virtual actuator much quicker than the plant. The zero row of the matrix $\boldsymbol{M}$ ensures that the failed valve is no longer used for feedback control. Due to the separation property of the virtual actuator, the overall closed-loop system has the eigenvalues of the nominal closed-loop system and the eigenvalues given in Eq. (10.9) for the virtual actuator. Hence, the reconfigured system is stable.

Figure 10.25 approves this result. The two bars placed at time $t = 350\,\mathrm{s}$ mark the time instant at which the valve is blocked and the controller reconfigured. The temperature ϑ_{TS} and the level l_{TS} remain at the set-points, whereas the conductivity cannot follow precisely the set-point change at time $t = 300\,\mathrm{s}$ marked by the dashed line. This is due to the proportional controller used.

Figure 10.26 shows the six control inputs. After the valve V_{TB} is blocked, the signal u_{TB} shown in the top right corner of the figure does no longer change. The virtual actuator uses the input signals u_{TS}, u_{TB} and u_{PS} which are also used by the nominal controller. In addition to this, the virtual actuator exploits the input u_{CW} to the cold water Valve V_{CW}, whereas the other additional input u_{TM} is not used.

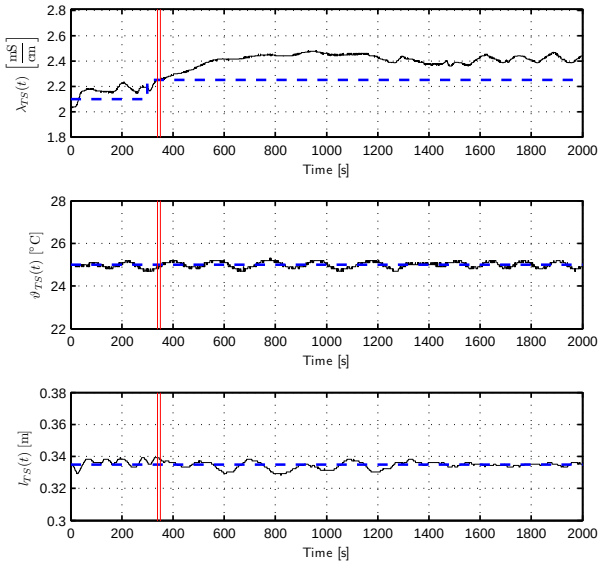

Fig. 10.25. Reconfiguration in case of the valve V_{TB}-failure

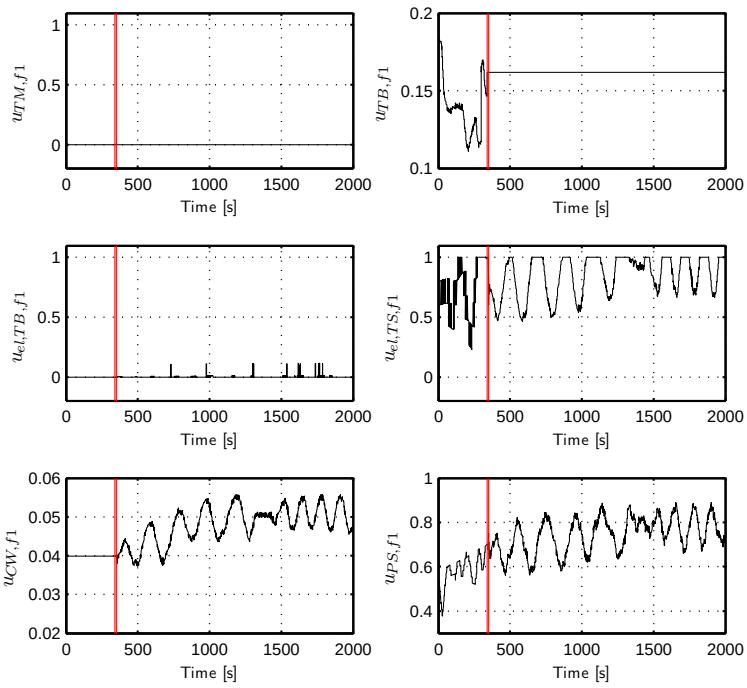

Fig. 10.26. Absolute values of the control input after the reconfiguration in case of the valve V_{TB}-failure

The choice how to distribute the effect of the blocked valve over the remaining actuators is made implicitly by the virtual actuator. No selection procedure, with a possible involvement of a human control designer, is necessary. Therefore, the concept of the virtual actuator can be applied completely automatically.

The second experiment concerns the aim to bring all variables to be controlled back to their set-points. Here the "'complete"' virtual actuator with the two parameter matrices $\boldsymbol{M}$ and $\boldsymbol{N}$ is used. Besides the matrix $\boldsymbol{M}$ given above, the direct feedthrough is chosen as

$$\begin{aligned}\boldsymbol{N} &= \left(\boldsymbol{C}(\boldsymbol{A}-\boldsymbol{B}_f\boldsymbol{M})^{-1}\boldsymbol{B}_f\right)^{-1}\left(\boldsymbol{C}(\boldsymbol{A}-\boldsymbol{B}_f\boldsymbol{M})^{-1}\boldsymbol{B}\right)\\ &= \begin{pmatrix} 1 & 0{,}291 & -0{,}016 & 0{,}053 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ 0 & -0{,}588 & 0{,}031 & -0{,}037 & 1 & 0\\ 0 & -4{,}250 & -0{,}012 & -0{,}004 & 0 & 1 \end{pmatrix},\end{aligned}$$

which ensures set-point following, because the reconfigured closed-loop system has the same static reinforcement as the nominal control loop.

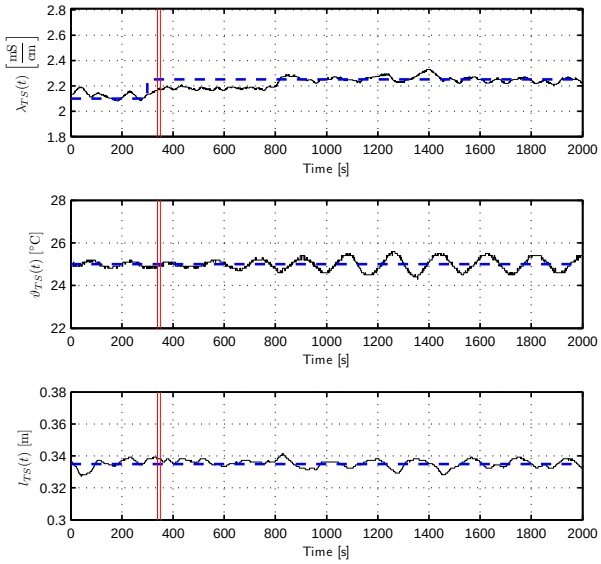

Fig. 10.27. Reconfiguration after valve V_{TB}-failure

The reconfiguration result is depicted in Fig. 10.27. The same experiment has been made as before, but now all three control outputs are moved back to their set-points.

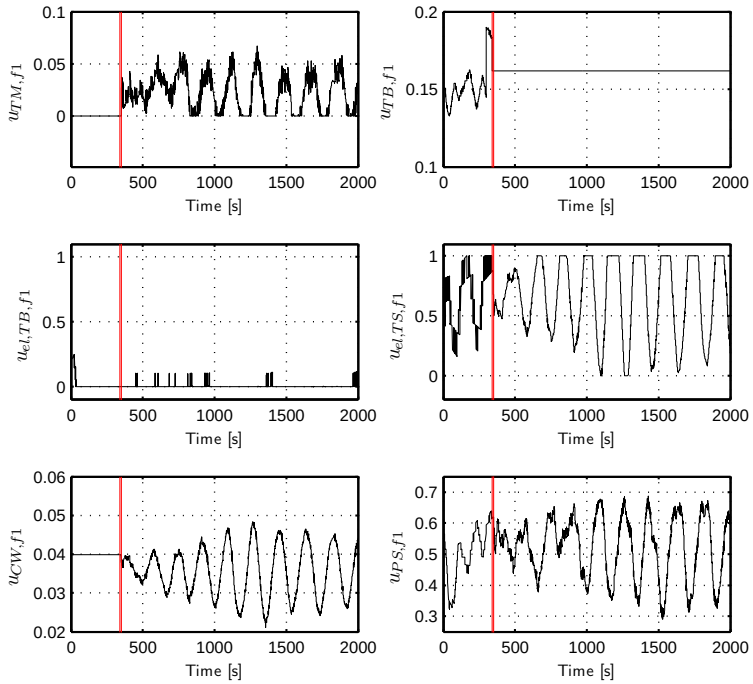

Fig. 10.28. Control input after the reconfiguration for valve V_{TM}-failure

As Fig. 10.28 shows, the virtual actuator uses now the additional inputs u_{CW} and u_{TM}. The reconfiguration is completely successful including the restoration of the set-point.

10.3 Diagnosis and control of a ship propulsion system

10.3.1 Structure of the ship propulsion system

Faults in a ship propulsion system and its associated automation system can cause a dramatic reduction in the ship's ability to propel and manoeuvre itself, and effective means are needed to prevent faults to develop into a failure. Various algorithms and methods from different research areas can be used to analyse the system and subsequently detect, isolate, and accommodate the faults. The ship propulsion system described in this section was presented as an international benchmark and was used as a platform for development of new ideas and comparison of methods.

The topics selected for this section are based on structural analysis. It is shown how residual generators are directly deduced from analysis of structure and how fault-tolerance can be obtained. Diagnostic methods and a supervisor logic to obtain

fault-tolerant control are implemented and tested against recorded time-history data which were manipulated to include well defined faults.

The dynamics of the propulsion system is non-linear. Furthermore, one essential fault is non-additive. The implication is that some residual generators become non-linear. This section illustrates how such real-life phenomena can be handled in the general framework developed in this book and where slight extensions are needed.

The propulsion system example origins from studies and manoeuvering trials with the Danish intercity ferry MF Dr. Ingrid, a 10.000 tons combined passenger and train ferry. Detailed modelling and data recorded from manoeuvering trials with the ferry give a realistic scenario for test of diagnostic methods and techniques to obtain fault tolerance.

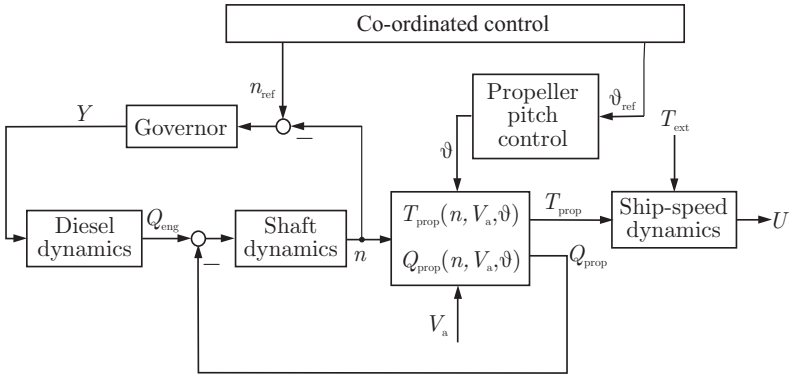

Fig. 10.29. Block diagram of the ship propulsion system

Ship propulsion system. An outline of the propulsion system chosen for the benchmark is shown in Fig. 10.29 (of Table 10.8 for a list of symbols). The main components are described by the following blocks:

- *Diesel dynamics* gives engine torque to drive the propeller shaft.
- *Shaft dynamics* provides shaft speed given diesel and propeller torques.
- *Propeller characteristics* provide propeller thrust and load torque from shaft speed n, propeller pitch ϑ and water speed V_a;
- *Ship speed dynamics* determines ship speed from propeller thrust and external forces.
- *Propeller pitch and shaft speed controllers* (governor) control the propeller pitch and shaft speed.
- *The coordinated control level* calculates set-points for shaft speed and propeller pitch controllers.

Table 10.8 List of symbols used in the ship propulsion system

Symbol	Unit	Explanation
I_t	kgm^2	Total inertia
K_y	Nm	Torque coefficient
$n(t)$	rads^{-1}	Shaft speed
$R(U)$	N	Hull resistance
$T_{prop}(t)$	N	Propeller thrust
$T_{ext}(t)$	N	External force
$1-t_T$	-	Thrust deduction factor
$U(t)$	ms^{-1}	Ship speed
$V_a(t)$	ms^{-1}	Flow at propeller
$1-w$	-	Wake fraction
$Q_{eng}(t)$	Nm	Diesel torque
Q_f	Nm	Shaft friction
$Q_{prop}(t)$	Nm	Propeller torque
$Y_d(t)$	0...1	Fuel index
$\vartheta(t)$	-1...1	Propeller pitch

The coordinated control level is detailed in Fig. 10.30. The following functions are included:

- **Combinator**: Gives a set of command values: $n_{\rm com}(t)$ and $\vartheta_{\rm com}(t)$ as functions of the command handle position.

- **Efficiency optimiser**: A module to optimise propulsion efficiency determines $n_{\rm com}(t)$ and $\vartheta_{\rm com}(t)$, based on measured values of $Y(t)$, $n(t)$, $\vartheta(t)$ and $U(t)$.

- **Speed control**: A ship speed-control module maintains a command value of ship speed $U_{\rm ref}$, using measured values of $Y(t)$, $n(t)$, $\vartheta(t)$ and $U(t)$ as input.

- **Overload control**: Modifies $n_{\rm com}(t)$ and $\vartheta_{\rm com}(t)$ to prohibit the prime mover from reaching its torque limits. The fuel index is used to determine an approaching overload condition.

10.3.2 Models of the propulsion system

The overall function of the propulsion system is to maintain the ship's ability to propel itself and to manoeuvre. Propulsion requires thrust ahead whereas manoeuvres require ahead and astern thrust ability. With a positive shaft speed n, this is

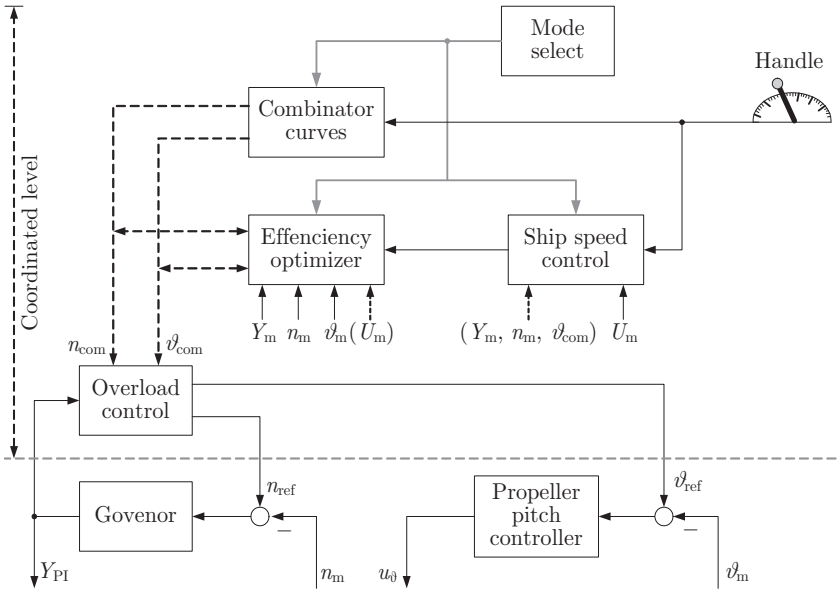

Fig. 10.30. Hierarchy of controllers for the propulsion system. The handle gives input to a combinator, efficiency optimiser, and ship speed control module. Lower level controls are shaft speed (governor), propeller pitch and diesel overload blocks.

obtained by an appropriate change of the propeller pitch ϑ, which is the angle that the propeller blades are twisted.

The component hierarchy is treated as belonging to two levels. Lower level components are the diesel engine with shaft speed controller, the propeller with the pitch controller, and the ship's speed dynamics. The upper level comprises coordinated control for the lower level components and overall command to the propulsion system. Reconfiguration will take place at the upper level, but lower-level controllers should be fault-tolerant, if possible, to maintain their primary services.

Upper-level components. Upper level components are the following:

- **Command handle**: A command handle's position constitutes the main man-machine interface (MMI).

- **Combinator**: Use-modes with different interpretations of handle position are available:
  - **Manoeuvring**: Handle position determines $n(t)$ and $\vartheta(t)$;
  - **Economy**: Handle position determines $n_{\mathrm{com}}(t)$ and $\vartheta_{\mathrm{com}}(t)$;
  - **Set speed**: Maintain a set ship speed using measured ship speed $U(t)$.

- **Efficiency optimiser**: The efficiency optimiser determines the set of $n(t)$ and $\vartheta(t)$ that achieves the desired ship speed $U_{\text{ref}} = f_{sc}(h(t))$ as determined by the handle position, without ship speed feedback.

- **Ship speed control**: Ship speed control aims at maintaining a set ship speed within a narrow margin. This component uses measured ship speed as one of its input variables.

- **Diesel overload control**: Overload is avoided by reducing the propeller pitch if diesel torque is close to maximum at a given shaft speed.

Lower-level components. The lower level consists of the shaft speed and the propeller-pitch controllers and the physical components of the propulsion system. In a component-based analysis, the physical components related to the pitch control function are lumped together to a new entity called *propeller pitch control*.

Propeller pitch control. The pitch control is an aggregated component that comprises a large hydraulic actuator turning the propeller blades, the feedback from a pitch sensor, a controller and the drive electronics. In its original implementation, this component has only one version of the use-mode um_1, which denotes the automatic mode. In order to obtain fault-tolerant properties, other versions are added. The concise definition of the component is given in the following equations:

$$
\begin{aligned}
< \mathbf{Propeller\ pitch\ control} > \quad &::= \\
< M(0,1) > \quad &::= \quad < um_0\ (manual),\ um_1\ (automatic) > \\
\\
< um_o\ (manual) > \quad &::= \quad < s_o > \\
< um_1\ (automatic) > \quad &::= \quad < s_o,\ s_1 > \\
\\
< \mathbf{s_0}\ (\mathbf{3-state}) > \quad &::= \quad < v_0\ (up-down) > \\
\\
< v_0\ (up-down) > \quad &::= \quad < consumed >, < produced >, \\
&\qquad\quad < procedure >, < request >, \\
&\qquad\quad < activation >, < resources > \\
< consumed > \quad &::= \quad < command\ \in [up,\ nil,\ down], > \\
< produced > \quad &::= \quad < pitch\ angle\ of\ blades, > \\
< procedure > \quad &::= \quad < plant\ dynamics \text{ Eq. } (10.12)\ with \\
&\qquad\quad control\ (10.13), > \\
< request > \quad &::= \quad < select\ 3-state, > \\
< activation > \quad &::= \quad < none, > \\
< resources > \quad &::= \quad < hydraulic\ oil\ supply,\ CP\ propeller >
\end{aligned}
$$

$$
\begin{array}{rcl}
< \boldsymbol{s_1}\ (\boldsymbol{continuous}) > & ::= & < v_1\,(normal),\ v_2\,(ftc-a) > \\
 & & \\
< v_1\,(normal) > & ::= & < consumed >, < produced >, \\
 & & < procedure >, < request >, \\
 & & < activation >, < resources > \\
< consumed > & ::= & < pitch\ angle\ command, > \\
< produced > & ::= & < pitch\ angle\ of\ blades, > \\
< procedure > & ::= & < plant\ dynamics\ \text{Eq. (10.12)}\ with\ control \\
 & & (10.14), > \\
< request > & ::= & < select\ automatic,\ normal, > \\
< activation\ condition > & ::= & < hydraulic\ pressure\ present, > \\
< resources > & ::= & < angle\ sensor,\ hydraulic\ oil\ supply, \\
 & & CP\ propeller > \\
 & & \\
< v_1\,(ftc-a) > & ::= & < consumed >, < produced >, \\
 & & < procedure >, < request >, \\
 & & < activation >, < resources > \\
< consumed > & ::= & < pitch\ angle\ command, > \\
< produced > & ::= & < pitch\ angle\ of\ propeller, > \\
< procedure > & ::= & < \text{Eq. (10.12)}\ and\ fault-tolerant\ control \\
 & & results, > \\
< request > & ::= & < select\ automatic,\ ftc-a, > \\
< activation > & ::= & < hydraulic\ pressure\ present, > \\
< resources > & ::= & < hydraulic\ oil\ supply, CP\ propeller > . \\
< FPA\ input \mid um_0 \vee um_1 > & ::= & < sensor\ fault\ f_1 = \Delta\vartheta,\ \text{leak}\ f_2 = \Delta\dot{\vartheta}_{inc}, \\
 & & e_{i1} = \vartheta_{\text{com}} > \\
< FPA\ output \mid um_0 \vee um_1 > & ::= & < e_{o1} = \vartheta L > \\
< FPA\ description \mid um_o > & ::= & < M_{pc-um0} > \\
< FPA\ description \mid um_1 > & ::= & < M_{pc-um1} >
\end{array}
$$

The mathematical model for the physical parts of the component is composed of the following equations:

$$\vartheta_m(t) = \vartheta(t) + \nu_\vartheta(t) + \Delta\vartheta(t) \tag{10.10}$$

$$\dot{\vartheta}(t) = \max\left(\dot{\vartheta}_{\min},\ \min\left(u_{\dot{\vartheta}}(t), \dot{\vartheta}_{\max}(t)\right)\right) + \Delta\dot{\vartheta}_{\text{inc}} \tag{10.11}$$

$$\vartheta(t) = \max\left(\vartheta_{\min},\ \min\left(\vartheta(t),\ \vartheta_{\max}\right)\right). \tag{10.12}$$

The control signal $u_{\dot{\vartheta}}(t)$ is generated according to the version running. In version v_o the control signal is obtained from

$$u_{\dot{\vartheta}}(t) = k_t u_{\text{cmd}}(t), \quad u_{\text{cmd}} \in [-1, 0, 1]. \tag{10.13}$$

In versions v_1 and v_2,

$$u_{\dot{\vartheta}}(t) = k_t\left(\vartheta_{\text{ref}}(t) - \vartheta_m(t)\right) \tag{10.14}$$

holds. Here, $\vartheta_m(t)$ is the measured propeller pitch, $[\dot{\vartheta}_{\min},\ \dot{\vartheta}_{\max}]$ the rate interval set by the hydraulic pump capacity and geometry, and $[\vartheta_{\min},\ \vartheta_{\max}]$ is the physical

interval for propeller-blade travel. $\nu_{\vartheta}(t)$ is the measurement noise. Two faults are included in the model: leakage $\Delta\dot{\vartheta}_{\mathrm{inc}}(t)$, and pitch sensor fault $\Delta\vartheta(t)$. It is noted that the control signal $u_{\dot{\vartheta}}(t)$ is not measured.

With $(e_i) \in [low,\ high,\ fluc,\ undef]$ we get

$$\boldsymbol{M}_{pc-umo} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}'$$

$$\boldsymbol{M}_{pc-um1} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}'.$$

Shaft speed control. The input to the shaft speed controller, which is called the governor, is given by the shaft speed reference $n_{\mathrm{ref}}(t)$ and the measured shaft speed $n_m(t)$. The output is the throttle of the diesel engine, which is proportional to the fuel index $Y(t)$. The governor is a PI controller. Anti-windup is part of the integrating action, and K is the anti-windup gain.

$$\begin{aligned}
&< \textbf{Shaft speed controller} > \quad ::= \\
&\qquad < M(0,1) > \quad ::= \quad < um_1\ (automatic) > \\
&\\
&\qquad < um_1\ (automatic) > \quad ::= \quad < s_o,\ s_1 > \\
&\\
&\qquad < \boldsymbol{s_0}\ (\textbf{constant}) > \quad ::= \quad < v_0\ (constant) > \\
&\qquad < v_0\ (constant) > \quad ::= \quad < consumed >, < produced >, \\
&\qquad\qquad\qquad\qquad\quad < procedure >, < request >, \\
&\qquad\qquad\qquad\qquad\quad < activation >, < resources > \\
&\qquad < consumed > \quad ::= \quad < logic\ command >, \\
&\qquad < produced > \quad ::= \quad < diesel\ index\ Y >, \\
&\qquad < procedure > \quad ::= \quad < Y = Y_{ta}\ time\ of\ activation >, \\
&\qquad < request > \quad ::= \quad < constant >, \\
&\qquad < activation > \quad ::= \quad < none >, \\
&\qquad < resources > \quad ::= \quad < diesel\ engine > \\
&\\
&\qquad < \boldsymbol{s_1}(continuous) > \quad ::= \quad < v_1\ (normal),\ v_2(ftc-n) > \\
&\qquad < v_1\ (normal) > \quad ::= \quad < consumed >, < produced >, \\
&\qquad\qquad\qquad\qquad\quad < procedure >, < request >, \\
&\qquad\qquad\qquad\qquad\quad < activation >, < resources >
\end{aligned}$$

$$
\begin{array}{rcl}
< consumed > & ::= & < n_m,\ n_{\rm com},\ Y_m >, \\
< produced > & ::= & <\text{Y}>, \\
< procedure > & ::= & < \text{Eqs. (10.15) } and \text{ (10.16)} >, \\
< request > & ::= & < automatic >, \\
< activation > & ::= & < none >, \\
< resources > & ::= & < n_m,\ Y_m,\ n_com,\ power, mainengine > \\
 & & \\
< v_2\ (ftc-n) > & ::= & < consumed >, < produced >, \\
 & & < procedure >, < request >, \\
 & & < activation >, < resources > \\
< consumed & ::= & < \hat{n},\ n_{\rm com},\ Y_m >, \\
< produced > & ::= & < Y >, \\
< procedure > & ::= & < \text{Eq. (10.15) } with\ n_m = \hat{n}\ and \\
 & & \text{Eq. (10.16)} >, \\
< request > & ::= & < automatic >, \\
< activation > & ::= & < automatic,\ ftc-n >, \\
< resources > & ::= & < \hat{n},\ n_com,\ Y_m,\ power,\ main\ engine > \\
 & & \\
< FPA\ in\ \mid\ um_1,\ v_0 > & ::= & < n_m,\ n_{\rm com},\ Y_m > \\
< FPA\ in\ \mid\ um_1,\ v_1 > & ::= & < n_m,\ n_{\rm com},\ Y_m > \\
< FPA\ in\ \mid\ um_1,\ v_2 > & ::= & < \hat{n},\ n_{\rm com},\ Y_m > \\
< FPA\ out > & ::= & < Y > \\
< FPA\ in\ \mid\ um_1,\ v_0 > & ::= & < Mat0_{4,12} > \\
< FPA\ out\ \mid\ um_1,\ v_0 > & ::= & < \boldsymbol{M}_{gov-v1} > \\
< FPA\ mat\ \mid\ um_1\ v_0 > & ::= & < \boldsymbol{M}_{gov-v2} >
\end{array}
$$

The controller is given by

$$
\begin{array}{rcl}
n_m(t) & = & n(t) + \nu_n(t) + \Delta n(t) \\
\dot{Y}_i(t) & = & \dfrac{k_r}{\tau_i}\big((n_{\rm ref}(t)\ -\ n_m(T)) - K(Y_{PIb}(t) - Y_{PI}(t))\big) \\
Y_{PIb}(t) & = & Y_i(t)\ +\ k_r.(n_{\rm ref}(t)\ -\ n_m(t)) \\
Y_{PI}(t) & = & \min\left(\max\left(Y_{PIb}(t),\ Y_{lb}\right),\ Y_{ub}\right).
\end{array}
\tag{10.15}
$$

Y_{lb} and Y_{ub} are the lower and upper bounds for the integrator part of the governor, and $\Delta n(t)$ the measurement fault. The governor comprises fuel index limits to keep the diesel engine within its allowed envelope of operation. These limits are given below

$$
y_{\max} = \left\{ \begin{array}{ll}
0.4 & n_m \leq 40\,\% \text{ of } n_{\max,\rm a} \\
1 & n_m \geq 80\,\% \text{ of } n_{\max,\rm a} \\
\dfrac{1.5 n_m}{n_{\max,\,a}} - 0.2 & \text{otherwise}
\end{array} \right\}
\tag{10.16}
$$

$$
Y(t) = \max(0,\ \min(Y_{PI}(t),\ y_{\max})),
$$

where $n_{\max,\mathrm{a}}$ is the maximum generated shaft speed allowed, n_m the measured shaft speed, $y_{\max}$ the hard limit specified for the engine, and $Y \in [0,\, Y_{\max}]$ denotes the command fuel index.

A similar formal description can be made for each physical component, but this is omitted for brevity.

Diesel engine. The diesel engine generates a torque Q_{eng}, which is controlled by its fuel index Y, to drive the shaft. The diesel engine dynamics can be divided into two parts. The first part describes the relation between the generated torque and the fuel index. It is given by the transfer function

$$Q_{\mathrm{eng}}(s) = \frac{K_y + \Delta K_y}{1 + \tau_c\, s} Y(s), \tag{10.17}$$

where K_y is the gain constant and τ_c is the time constant corresponding to torque build-up from cylinder firings.

The second part expresses the torque balance of the shaft:

$$I_m \dot{n}(t) = Q_{\mathrm{eng}}(t) - Q_{\mathrm{prop}}(t) - Q_f. \tag{10.18}$$

$Q_{\mathrm{eng}}(t)$ is the torque developed by the diesel engine, $Q_{\mathrm{prop}}(t)$ is the torque developed from the propeller, and Q_f is the friction torque.

Propeller thrust and torque. A controllable pitch propeller (CP) has blades that can be turned by means of a hydraulic mechanism. The propeller pitch ϑ can be changed from $100\,\%$ (full ahead) to $-100\,\%$ (full astern).

The propeller thrust and torque are determined by the following bilinear relations:

$$T_{\mathrm{prop}}(t) = T_{|n|n}(\vartheta)\,|n(t)|n(t) + T_{|n|V_a}(\vartheta(t))\,|n(t)|V_a(t) \tag{10.19}$$

$$Q_{\mathrm{prop}}(t) = Q_{|n|n}(\vartheta)\,|n(t)|n(t) + Q_{|n|V_a}(\vartheta(t))\,|n(t)|V_a(t). \tag{10.20}$$

V_a is the velocity of the water passing through the propeller disc

$$V_a(t) = (1 - w)U(t),$$

where w is a hull-dependent parameter called the wake fraction. The coefficients $T_{|n|n}$, $T_{|n|V_a}$, $Q_{|n|n}$ and $Q_{|n|V_a}$ are complex functions of the pitch $\vartheta(t)$. T_{prop} and Q_{prop} are calculated by interpolating between tables of data measured in model propeller tests. Figure 10.31 shows graphically the dependencies of T_{prop} and Q_{prop} on n and V_a for different values of the pitch. K_T and K_Q, in these figures denote thrust and torque coefficients

$$T_{\mathrm{prop}} = K_T\,\rho\,D^4\,|n|n \tag{10.21}$$

$$Q_{\mathrm{prop}} = K_Q\,\rho\,D^5\,|n|n,$$

where D is the propeller diameter and ρ the mass density of water.

Ship speed dynamics. The following non-linear differential equation approximates the ship speed dynamics:

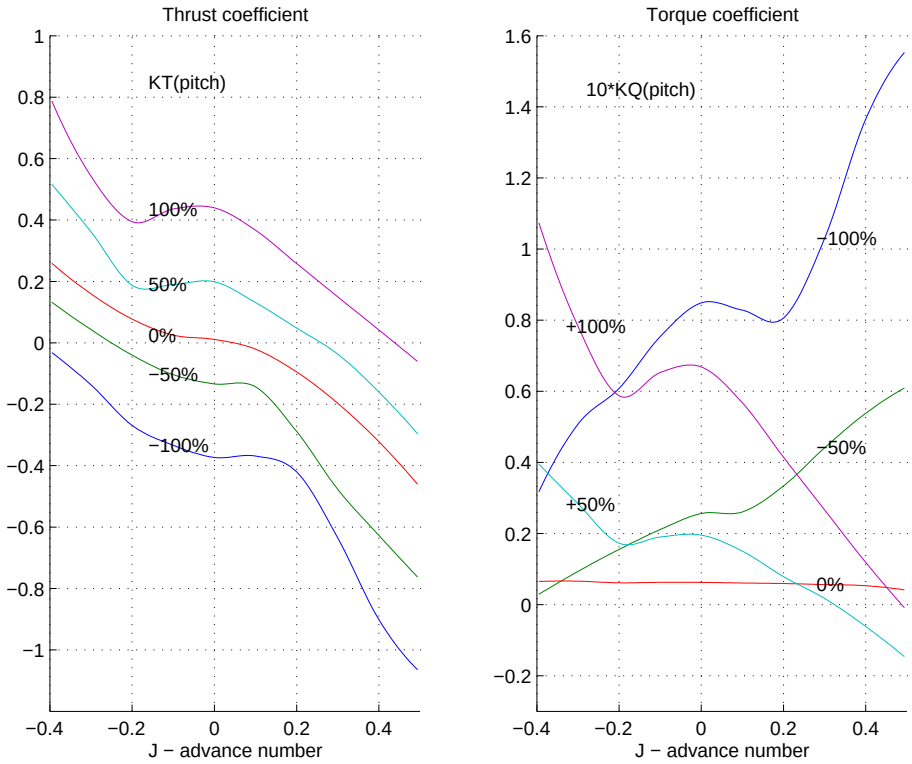

Fig. 10.31. Propeller torque as functions of advance number $J = 2\pi V_a/n$ for different values of pitch

$$\begin{aligned}(m - X_{\dot{U}})\dot{U}(t) &= R(U(t)) + (1 - t_T)T_{\text{prop}}(t) + T_{\text{ext}}(t) \qquad (10.22)\\ U_m(t) &= U(t) + \nu_U(t)\,.\end{aligned}$$

The term $R(U)$ describes the resistance of the ship in the water. It is a negative quantity. Figure 10.32 shows the hull resistance as a function of the speed for two given load conditions. $X_{\dot{U}}$ represents the added mass in surge, which is negative. The thrust deduction $1 - t_T$ represents the net thrust lost due to the propeller-generated flow at the ship's stern. T_{ext} is the external force brought about by the wind and the waves. ν_U is the measurement noise.

10.3.3 Fault scenarios and requirements on the diagnosis

Fault scenario. The faults are summarised in Table 10.9.

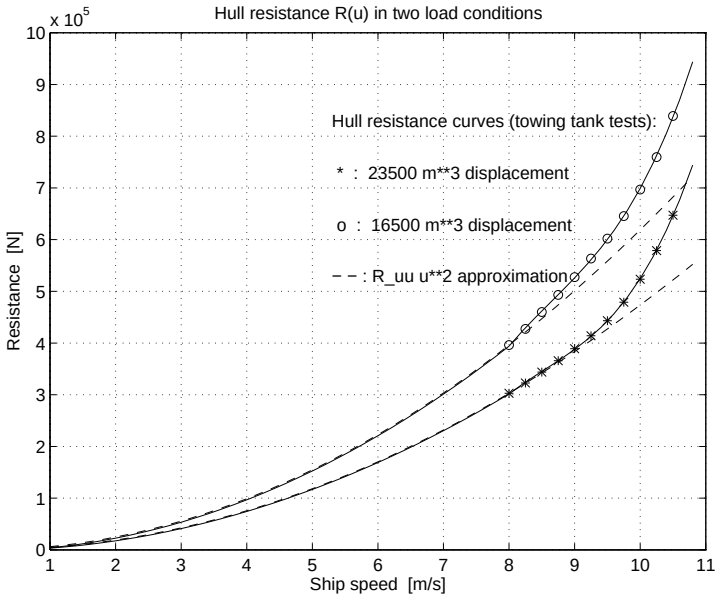

Fig. 10.32. The magnitude of the hull resistance in dependence upon the ship speed and its loading conditions. The commonly adopted square law curve is not valid in the relevant range of operation ($8.5\ \mathrm{m/s}$ to $10.5\ \mathrm{m/s}$)

Table 10.9 Faults considered

Fault	**Symbol**	**Type**
Sensor faults	$\Delta\vartheta$	additive - abrupt
Hydraulic leak	$\Delta\dot{\vartheta}_{\mathrm{inc}}$	additive - incipient
Sensor faults	Δn	additive - abrupt
Diesel fault	ΔK_y	multiplicative - abrupt

A formal analysis of fault propagation shows that they have different degrees of severity. Some are very serious and need rapid fault detection and accommodation to avoid serious accidents if the component failure occurs during a critical manoeuvre. The time to detect and reconfigure is hence essential. Some of the faults described are based on actual events that have caused serious damage due to the lack of fault-tolerant features in existing propulsion control systems. Figure 10.33 locates the generic faults of the benchmark in the system block diagram. The faults are:

1. Faults $\Delta\vartheta$ related to the propeller pitch:
   - $\Delta\vartheta_{\mathrm{high}}$: This fault can occur due to an electrical or mechanical defect in the pitch sensor or its interface.
   - $\Delta\vartheta_{\mathrm{low}}$: This fault can occur due to an electrical or mechanical fault in the pitch sensor or its interface.

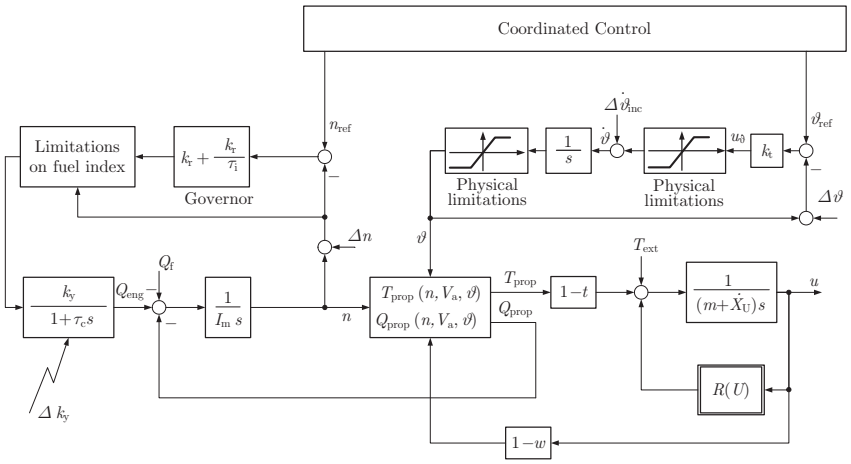

Fig. 10.33. Block diagram of the propulsion system with saturation phenomena shown for shaft speed and pitch controller. The generic faults of the benchmark are indicated.

- $\Delta\dot{\vartheta}_{\mathrm{inc}}$: A leakage can occur in the (hydraulic) actuation part of the control system; in practice, often in an over-pressurised valve.

2. Faults Δn related to shaft speed measurement. A dual pulse pick-up is used for measuring the shaft speed. The followings faults are considered:
   - a maximum signal Δn_{high} (disturbance on one pick-up), and
   - a minimum signal Δn_{low} (loss of both pick-up signals).
3. Faults related to the diesel engine (ΔK_y). The generated shaft torque can be lower than expected for the following reasons: reduced air inlet, reduced fuel oil inlet, or a cylinder is not working.

To determine how faults affect the system operation, the fault-propagation analysis methodology presented in Chapter 4 was employed. This analysis shows the end-effects for each fault. Combining end effects, the severity level for each fault was assessed. The results are shown in Table 10.10.

Table 10.10 Consequences and severity levels for the benchmark faults

Fault	Consequence	Severity
$\Delta\vartheta_{\text{high}}$	$deceleration \Rightarrow manoeuvring\ risk$	high
$\Delta\vartheta_{\text{low}}$	$acceleration \Rightarrow collision\ risk$	very high
$\Delta\dot{\vartheta}_{\text{inc}}$	$gradual\ speed\ change \Rightarrow cost\ increase$	medium
Δn_{high}	$deceleration \Rightarrow manoeuvring\ risk$	high
Δn_{low}	$acceleration \Rightarrow collision\ risk$	very high
ΔK_y	$diesel\ overload \Rightarrow wear, slowdown$	medium

Requirements on the diagnosis. The requirements for our diagnostic methods are as follows, where T_s is the sampling time in the particular control loop:

- *Time-to-detect* (T_d): For the sensor feedback faults ($\Delta\vartheta_{\text{low}}$, $\Delta\vartheta_{\text{high}}$, Δn_{low}, and Δn_{high}): $T_d < 2T_s$, the incipient fault $\Delta\dot{\vartheta}_{\text{inc}}$: $T_d < 100\,T_s$, the gain fault Δk_y : $T_d < 5T_s$.
- *Unknown input*: A time-varying external drag force from weather and shallow water is a potential source of false detection. Diagnosis should be insensitive to this.
- *False detection probability:* $P_f < 0.01$. Fault-free real data, including harbour manoeuvres, are provided to allow realistic testing of algorithms with respect to false detection.
- *Missed-detection probability:* $P_m < 0.001$. Due to the high severity level of faults, which can result in endangering the ship (and its crew), the probability of not detecting them when they do occur should be as low as possible.
- *Robust design:* Several sources of model uncertainty exist: slowly increasing hull resistance $R(U)$ due to growth ($0\,\%$ increasing to $20\,\%$ of $R(U)$), or varying external force from sea and wind ($\pm 10\,\%$ of $R(U)$), *a-priori* uncertainty in propeller thrust and torque ($\pm\,10\,\%$) and engine friction (from $5\,\%$ to $8\,\%$), and general uncertainty on other physical parameters ($\pm\,2\,\%$). Fault diagnosis should be robust to these. *A-posteriori* data for parameters may be identified and used for diagnosis.

Requirements on fault handling. Fault handling should incorporate appropriate steps to accommodate the faults. The remedial actions should primarily use the re-

configuration at the coordination level. Performance in a reconfigured mode can be lower than under faultless conditions. Large transients should be avoided when changing to a reconfigured mode. Bump-less transfer is not required, but is a desirable feature.

Test scenario. The test sequences constitute recordings from manoeuvres with the intercity ferry MF Dr. Ingrid. Faults are superimposed on the recorded data using the high-fidelity simulation model. Time stamps for different fault events are given in Table 10.11. The total simulation time is 3500 seconds. The fault ΔK_y corresponds to a $20\,\%$ drop in the diesel engine gain K_y.

Diagnosis. The main task is to find fault diagnosic algorithms that make it possible to detect and isolate the faults mentioned above. The algorithms should be robust to model uncertainty, load changes, and external forces.

The known and measured variables are propeller pitch set-point $\vartheta_{\mathrm{ref}}(t)$, propeller pitch measurement $\vartheta_m(t)$, shaft speed set-point $n_{\mathrm{ref}}(t)$, shaft speed measurement $n_m(t)$, ship speed $U_m(t)$, and the fuel index $Y_m(t)$. The data are obtained by sampling every 1 second.

Table 10.11 Fault events and their corresponding activation time intervals

Event	Start time	End time
$\Delta\vartheta_{\mathrm{high}}$	$180\,\mathrm{s}$	$210\,\mathrm{s}$
$\Delta\dot{\vartheta}_{\mathrm{inc}}$	$800\,\mathrm{s}$	$1700\,\mathrm{s}$
$\Delta\vartheta_{\mathrm{low}}$	$1890\,\mathrm{s}$	$1920\,\mathrm{s}$
Δn_{high}	$680\,\mathrm{s}$	$710\,\mathrm{s}$
Δn_{low}	$2640\,\mathrm{s}$	$2670\,\mathrm{s}$
ΔK_y	$3000\,\mathrm{s}$	$3500\,\mathrm{s}$

10.3.4 Structural analysis of the propulsion system

After having made an assessment of the set of high severity faults that need to be handled to obtain a fault-tolerant propulsion system, the next step is to provide relations for use in design of residual generators. Structural analysis is employed for this purpose. A prerequisite for structural analysis is that the set of constraints are listed. For the open loop system, involving the shaft and ship dynamics, the related constraints are the following:

$$
\begin{array}{lcll}
c_1 & : & c_1(\vartheta, \vartheta_m) = 0 & : \quad \vartheta = \vartheta_m \\
c_2 & : & c_2(n, n_m) = 0 & : \quad n = n_m \\
c_3 & : & c_3(Y, Y_m) = 0 & : \quad Y = Y_m \\
c_4 & : & c_4(k_y, K_y) = 0 & : \quad k_y = K_y \\
c_5 & : & c_5(Y, k_y, Q_{\text{eng}}) = 0 & : \quad Q_{\text{eng}} + \tau_c \dot{Q}_{eng} = k_y Y \\
c_6 & : & c_6(\dot{Q}_{eng}, Q_{\text{eng}}) = 0 & : \quad \dot{Q}_{eng} = \frac{dQ_{\text{eng}}}{dt} \\
c_7 & : & c_7(Q_{\text{eng}}, Q_{\text{prop}}, n) = 0 & : \quad I_m \dot{n} = Q_{\text{eng}} - Q_{\text{prop}} \\
c_8 & : & c_8(\dot{n}, n) = 0 & : \quad \dot{n} = \frac{dn}{dt} \\
c_9 & : & c_9(n, \vartheta, U, Q_{\text{prop}}) = 0 & : \quad Table^1 \\
c_{10} & : & c_{10}(n, \vartheta, U, T_{\text{prop}}) = 0 & : \quad Table^2 \\
c_{11} & : & c_{11}(\dot{U}, R(U), T_{\text{prop}}) = 0 & : \quad m\dot{U} = R(U) - (1 - t_T) T_{\text{prop}} \\
c_{12} & : & c_{12}(R(U), U) = 0 & : \quad Table^3 \\
c_{13} & : & c_{13}(\dot{U}, U) = 0 & : \quad \dot{U} = \frac{dU}{dt} \\
c_{14} & : & c_{14}(U, U_m) = 0 & : \quad U = U_m
\end{array}
\tag{10.23}
$$

The propeller developed torque $Q_{\text{prop}}(t)$ and trust $T_{\text{prop}}(t)$ are functions of $n(t)$, $\vartheta(t)$, and $U(t)$, and are calculated by interpolating between data that are described in the given tables. For the sake of clarity, the following simplifications are made:

- measurement noises in the system are not represented,
- disturbances, Q_f and T_{ext} are disregarded,
- $X_{\dot{U}}$ is negligible and the trust deduction $1 - t_T$ is known.

Furthermore, the propeller pitch dynamic is described by the following constraints:

$$
\begin{array}{lcll}
c_{15} & : & c_{15}(u_{\dot{\vartheta}}, \vartheta_{\text{ref}}, \vartheta_m) = 0 & : \quad u_{\dot{\vartheta}} = k_t(\vartheta_{\text{ref}} - \vartheta_m) \\
c_{16} & : & c_{16}(u_{\dot{\vartheta}}, \dot{\vartheta}) = 0 & : \quad u_{\dot{\vartheta}} = \dot{\vartheta} \\
c_{17} & : & c_{17}(\dot{\vartheta}, \vartheta) = 0 & : \quad \dot{\vartheta} = \frac{d\vartheta}{dt}
\end{array}
\tag{10.24}
$$

These constraints are valid during normal operational conditions when the control and system's physical limits are not violated.

The system structure is

$$
\begin{aligned}
\mathcal{S} &= \bigcup_{i=1}^{17} c_i, \\
\mathcal{C} &= \{c_1, c_2, \cdots, c_{17}\}, \\
\mathcal{K} &= \{\vartheta_m, n_m, Y_m, U_m, K_y, \vartheta_{\text{ref}}\}, \\
\mathcal{X} &= \{U, \vartheta, n, Y, k_y, \dot{Q}_{\text{eng}}, Q_{\text{eng}}, Q_{\text{prop}}, T_{\text{prop}}, u_{\dot{\vartheta}}, R(U), \dot{U}, \dot{n}, \dot{\vartheta}\}, \\
\mathcal{Z} &= \mathcal{K} \bigcup \mathcal{X}.
\end{aligned}
$$

As the measurement noise is disregarded in analysis of structure, the relations $\vartheta_m = \vartheta$ and $n = n_m$ etc. hold. A bi-partite graph representation of the system structure is depicted in Fig. 10.34.

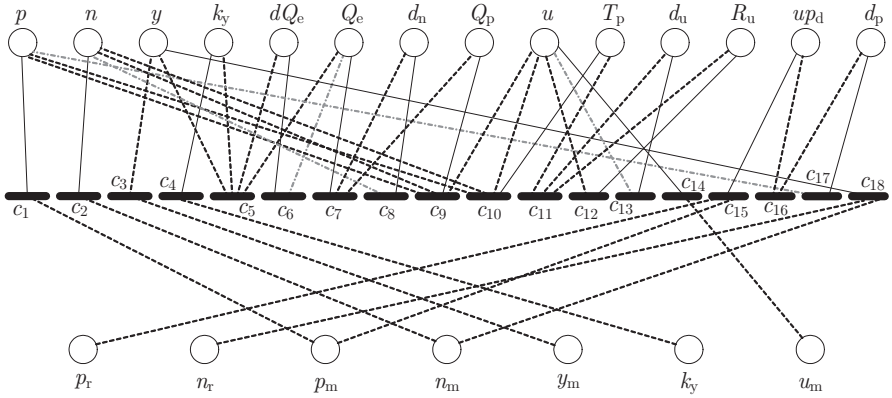

Fig. 10.34. Structural model of the ship propulsion system. The matching is illustrated by the thick lines.

Matching. The edges of a matching are identified by a thick line in the graph representation (Fig. 10.34) and by 'o' in the following incidence matrix.

As it is illustrated, a complete matching with respect to the unknown variables are obtained. There are three unmatched constraints that can be used for detecting the faults. Each unmatched constraint involves a different part of the system as follows:

Subsystem 1: Engine dynamics and propeller torque characteristics. The unmatched constraint c_5 involves engine and shaft dynamics and the propeller torque characteristics. The involved constraints are:

$$
\begin{aligned}
c_1 &: \quad \vartheta(t) = \vartheta_m(t) \\
c_2 &: \quad n(t) = n_m(t) \\
c_3 &: \quad Y(t) = Y_m(t) \\
c_4 &: \quad k_y(t) = K_y(t) \\
c_5 &: \quad Q_{\mathrm{eng}}(t) + \tau_c \dot{Q}_{\mathrm{eng}}(t) = k_y(t) Y(t) \\
c_6 &: \quad \dot{Q}_{\mathrm{eng}}(t) = \frac{dQ_{\mathrm{eng}}(t)}{dt} \\
c_7 &: \quad I_m \dot{n}(t) = Q_{\mathrm{eng}}(t) - Q_{\mathrm{prop}}(t) \\
c_8 &: \quad \dot{n}(t) = \frac{dn(t)}{dt} \\
c_{14} &: \quad U(t) = U_m(t)
\end{aligned}
$$

The faults that affect the dynamics of this subsystem are obviously faults in pitch and shaft measurements as well as the fault in the diesel engine.

Subsystem 2: Ship speed dynamics and propeller thrust characteristics. The unmatched constraint c_{11} involves ship speed dynamics and the propeller thrust

$\leftrightarrow$	ϑ_{m}	n_{m}	Y_{m}	U_{m}	K_{y}	ϑ_{ref}	U	ϑ	n	Y	k_{y}	$\dot{Q}_{\mathrm{eng}}$	Q_{eng}	Q_{prop}	T_{prop}	$R(U)$	$\dot{U}$	$\dot{n}$	u_ϑ	$\dot{\vartheta}$
c_1	1							①												
c_2		1							①											
c_3			1							①										
c_4					1						①									
c_5										1	1	1	1							
c_6												①	x							
c_7													①	1				1		
c_8									x									①		
c_9							1	1	1					①						
c_{10}							1	1	1						①					
c_{11}															1	1	1			
c_{12}							1									①				
c_{13}							x										①			
c_{14}				1			①													
c_{15}	1					1													①	
c_{16}																			1	①
c_{17}								x												1

Fig. 10.35. Incidence matrix of the ship example

characteristics. The involved constraints are:

$$\begin{aligned}
c_1 &: \vartheta(t) = \vartheta_m(t) \\
c_2 &: n(t) = n_m(t) \\
c_{10} &: Table^2 \\
c_{11} &: m\dot{U}(t) = R(U(t)) - (1 - t_T)T_{\text{prop}}(t) \\
c_{13} &: \dot{U}(t) = \frac{dU(t)}{dt} \\
c_{14} &: U(t) = U_m(t)
\end{aligned}$$

The involved dynamics will be affected by fault in shaft speed and pitch measurements.

Subsystem 3: Propeller pitch dynamics. The unmatched constraint c_{17} involves conctraints related to the propeller pitch dynamics. The involved constraints are:

$$\begin{aligned}
c_1 &: \vartheta(t) = \vartheta_m(t) \\
c_{15} &: u_{\dot{\vartheta}}(t) = k_t(\vartheta_{\text{ref}}(t) - \vartheta_m(t)) \\
c_{16} &: u_{\dot{\vartheta}}(t) = \dot{\vartheta}(t) \\
c_{17} &: \dot{\vartheta}(t) = \frac{d\vartheta(t)}{dt}
\end{aligned}$$

This subsystem dynamics will be affected by fault in pitch measurement and the hydraulic fault.

10.3.5 Fault diagnosis using the parity space approach and state observation

Residual generation. As it has been extensively argued in previous chapters, the parity space and observer-based approaches are commonly used to generate residuals. In this section, these approaches are applied to the propulsion system to obtain an expression for residuals.

Subsystem 3. Using the relevant constraints, a parity equation can be set up, which involves only known variables of this subsystem. The obtained parity equation is:

$$\frac{d\vartheta_m(t)}{dt} = k_t(\vartheta_{\text{ref}}(t) - \vartheta_m(t)).$$

Based on this, a residual can be defined as

$$r_\vartheta(t) = \frac{d\vartheta_m(t)}{dt} - k_t(\vartheta_{\text{ref}}(t) - \vartheta_m(t)).$$

The residual should have a vanishing mean value under normal conditions, i.e. when no sensor fault has occurred. The residual's dynamical behaviour shall change in the presence of faults ($\Delta\vartheta_{\text{high}}$, $\Delta\vartheta_{\text{low}}$, and $\Delta\dot{\vartheta}_{\text{inc}}$).

Subsystem 1. Since sensor measurements for propeller pitch, shaft speed and ship speed are available, the value of the propeller torque Q_{prop} can be computed means of $Table^1$. Hence, $Q_{\text{prop}}(\vartheta_m,\, n_m,\, U_m)$ is known. Through constraint c_7 we obtain

$$Q_{\text{eng}}(t) = I_m \dot{n}_m(t) - Q_{\text{prop}}(\vartheta_m(t), n_m(t), U_m(t))$$

Now, it is possible to set up a parity equation by using the constraints c_3, c_4, c_5, and c_6, i.e.

$$\begin{aligned} K_y Y_m(t) &= Q_{\text{eng}}(t) + \tau_c \frac{dQ_{\text{eng}}(t)}{dt} \\ &= I_m \dot{n}_m(t) - Q_{\text{prop}}(\vartheta_m(t), n_m(t), U_m(t)) \\ &+ \tau_c \frac{d(I_m \dot{n(t)}_m - Q_{\text{prop}}(\vartheta_m(t), n_m(t), U_m(t)))}{dt}. \end{aligned}$$

A residual expression can be defined, by using the obtained parity equation, in a straightforward manner:

$$\begin{aligned} r_Q(t) &= I_m \dot{n(t)}_m - Q_{\text{prop}}(\vartheta_m(t), n_m(t), U_m(t)) \\ &+ \tau_c \frac{d(I_m \dot{n(t)}_m - Q_{\text{prop}}(\vartheta_m(t), n_m(t), U_m(t)))}{dt} - K_y Y_m(t). \end{aligned}$$

Actual computation of this residual, in present from, requires employing numerical methods (for instance Euler method or central-difference formula of order 2). The computed residual should have a mean value equal zero under normal conditions. The residual's dynamical behaviour shall change in presence of sensor measurement faults ($\Delta\vartheta_{\text{high}}$, $\Delta\vartheta_{\text{low}}$, Δn_{high}, Δn_{low} and the engine gain fault ΔK_y.

Residual generator obtained by matching. This subsection employs the flexibility of structural analysis to generate a residual with desired properties. It would be desirable if the diesel engine gain fault could be detected independently of a fault in the shaft speed measurement. Since the constraint c_2 is not valid if the shaft speed fault is present, remove this constraint from the original set. Further, make the following simplifications: The diesel engine dynamics is much faster than the shaft speed and ship speed dynamics. This assumption is manifested by removing the constraint c_6 and changing the equation in the constraint c_5 to $Q_{\text{eng}} = k_y Y$.

Example 10.1 *Matching on revised system*

Matching on the open loop system that follows from removing c_2 and c_6 from the set of constraints in (10.23) and c_5 modified, makes us determine the unmatched constraints that are necessary to detect the diesel engine gain fault. This is left as an exercise.

The obtained fault-free subsystem can be written as:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{g}(\boldsymbol{x}(t)) + \boldsymbol{g}_u(\boldsymbol{x}(t))\boldsymbol{u}(t) \\ y(t) &= U(t) \end{aligned}$$

where

$$\boldsymbol{x}(t) = \begin{pmatrix} n(t) \\ U(t) \end{pmatrix}$$

$$
\begin{aligned}
\boldsymbol{g}(\boldsymbol{x}) &= \begin{pmatrix} 0 \\ \frac{1}{m}R(U) + \frac{1-t_T}{m}\,T_{|n|V_a}(1-w)nU \end{pmatrix} \\
\boldsymbol{g}_u(\boldsymbol{x}) &= \begin{pmatrix} \frac{1}{I_m}\,k_y & -\frac{1}{I_m}\left[Q_{|n|n}n^2 + Q_{|n|V_a}(1-w)nU\right] \\ 0 & \frac{1-t_T}{m}\,T_{|n|n}n^2 \end{pmatrix} \\
\boldsymbol{u}(t) &= \begin{pmatrix} Y_m(t) \\ \vartheta_m(t) \end{pmatrix}.
\end{aligned}
$$

The diesel engine dynamics is very fast (small τ_c), i.e. Eq. (10.17) has been used to employ a static relation for engine torque $Q_{eng} = (k_y + \Delta k_y)Y_m$. $Table^1$ and $Table^2$ are represented by their bilinear approximations (10.19) and (10.20). Q_0 is disregarded in the equations. □

Detection of the diesel engine gain fault ΔK_y. To return to the original notation of the ship dynamics the following notation is used for this subsystem

$$
\begin{aligned}
\dot{n}(t) &= \frac{1}{I_m}\,K_y\,Y_m - \frac{1}{I_m}\,\left[Q_{|n|V_a}(1-w)\,n\,U + Q_{|n|n}\,n^2\right]\,\vartheta \\
\dot{U}(t) &= \frac{1}{m}R(U) + \frac{1-t_T}{m}\,T_{|n|V_a}(1-w)\,n\,U + \frac{1-t_T}{m}\,T_{|n|n}\,n^2\,\vartheta \\
y(t) &= U(t)
\end{aligned}
$$

with the fuel index measurement Y_m and the pitch measurement ϑ_m as external input. For this system an observer can be given in the following form:

$$
\begin{aligned}
\dot{\hat{n}}(t) &= \frac{1}{I_m}\,K_y\,Y_m - \frac{1}{I_m}\,\left[Q_{|n|V_a}(1-w)\hat{n}\hat{U} + Q_{|n|n}\hat{n}^2\right]\,\vartheta_m + K^{\hat{n}}_{\Delta k_y}(U_m - \hat{U}) \\
\dot{\hat{U}}(t) &= \frac{1}{m}R(\hat{U}) + \frac{1-t_T}{m}\,[T_{|n|V_a}(1-w)\hat{n}\hat{U} + T_{|n|n}\hat{n}^2\vartheta_m] + K^{\hat{U}}_{\Delta k_y}(U_m - \hat{U}) \\
\hat{y}(t) &= \hat{U}(t).
\end{aligned}
$$

A residual can be obtained by using the output (ship speed estimate) of the observer and the ship speed measurement U_m in the following way:

$$r_{TQ} = U_m - \hat{U}.$$

The residual r_{TQ} is by construction only affected by the gain fault ΔK_y (when considering the pitch signal to be fault-free). As the observer is stable, the residual behaves in the fault-free case such that $r_{TQ} \to 0$ holds for $t \to \infty$. For the estimation errors

$$e^1_n(t) = n(t) - \hat{n}(t) \quad \text{and} \quad e^1_U(t) = U(t) - \hat{U}(t)$$

the following error dynamics can be given (with $\vartheta = \vartheta_m$):

$$
\begin{aligned}
\dot{e}^1_n = \frac{1}{I_m}\,\Delta k_y\,Y - \frac{1}{I_m}\,&\left[Q_{|n|V_a}(1-w)(nU - \hat{n}\hat{U}) + Q_{|n|n}(n^2 - \hat{n}^2)\right]\vartheta_m \\
&- K^{\hat{n}}_{\Delta k_y}(U_m - \hat{U}) \qquad (10.25)
\end{aligned}
$$

$$
\begin{aligned}
\dot{e}^1_U = \dot{r}_1 = \frac{1}{m}(R(U) - R(\hat{U})) + \frac{1-t_T}{m}[\,&T_{|n|V_a}(1-w)(nU - \hat{n}\hat{U}) + \\
&+ T_{|n|n}(n^2 - \hat{n}^2)\vartheta_m] - K^{\hat{U}}_{\Delta k_y}(U_m - \hat{U}). \qquad (10.26)
\end{aligned}
$$

Looking at the estimation error dynamics (10.25) one could think that an occurring gain fault ΔK_y would have a direct impact on $\dot{e}_n^1$ leading to a growing estimation error: $n \neq \hat{n}$. As can be seen from Eq. (10.26) this would then also affect the shaft speed estimate, i.e. $U \neq \hat{U}$. Hence, the first residual would deviate from zero in case of a gain fault: $r_{TQ} \neq 0$. However, with the coupled nonlinear equations at hand this argumentation is not correct. Simulation results given below show that the gain fault ΔK_y does indeed affect the residual r_{TQ}.

The observer offers also a second possibility to generate a residual when using the shaft speed measurement n_m:

$$r_{TQ_2}(t) = n_m(t) - \hat{n}(t).$$

Obviously, this residual is also affected by the shaft speed sensor fault Δn. The residual dynamics can be stated as follows

$$\dot{r}_{TQ_2}(n_m(t), \hat{n}(t)) = \dot{e}_n^1(t) + \dot{\Delta} n(t),$$

where $\dot{e}_n^1$ is described by Eq. (10.25).

Simulation results. The gains and initial conditions for the observer are chosen as follows:

$$\begin{aligned} K_{\Delta k_y}^{\hat{n}} &= 0.001, \\ K_{\Delta k_y}^{\hat{U}} &= 0.01, \\ \hat{n}(t=0) &= 9\ \mathrm{rad/s}, \\ \hat{U}(t=0) &= 0.1\ \mathrm{m/s}. \end{aligned}$$

The simulation result is shown in Fig. 10.36.

The second residual, i.e. r_{TQ_2}, is generated by choosing the gains and the initial conditions of the observer as:

$$\begin{aligned} K_{\Delta k_y}^{\hat{n}} &= 0.001, \\ K_{\Delta k_y}^{\hat{U}} &= 0.01, \\ \hat{n}(t=0) &= 9\ \mathrm{rad/s}, \\ \hat{U}(t=0) &= 0.1\ \mathrm{m/s}. \end{aligned}$$

The simulation result is shown in Fig. 10.37.

Comparing Fig. 10.36 and 10.37, it is clear that residual r_{TQ} is not affected by shaft sensor fault as it was the idea that was described in the start of section 10.3.5, whereas residual r_{TQ_2} is affected by both the shaft speed fault and the engine gain fault. Small deviations (from zero) in both residuals are due to the sudden change in measurement/input signals combined with the nonlinear behaviour of the system and can be handled by choosing an appropriate threshold.

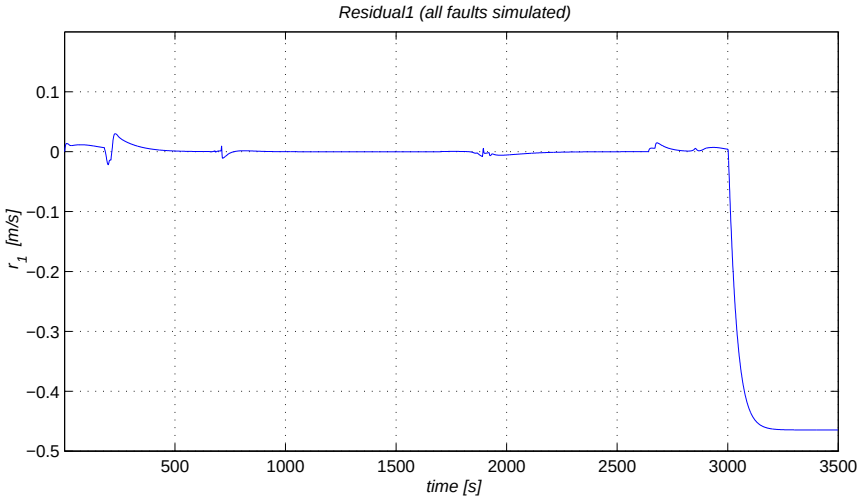

Fig. 10.36. Residual $r_{TQ} = U_m - \hat{U}$. Simulation including all faults and no measurement noise.

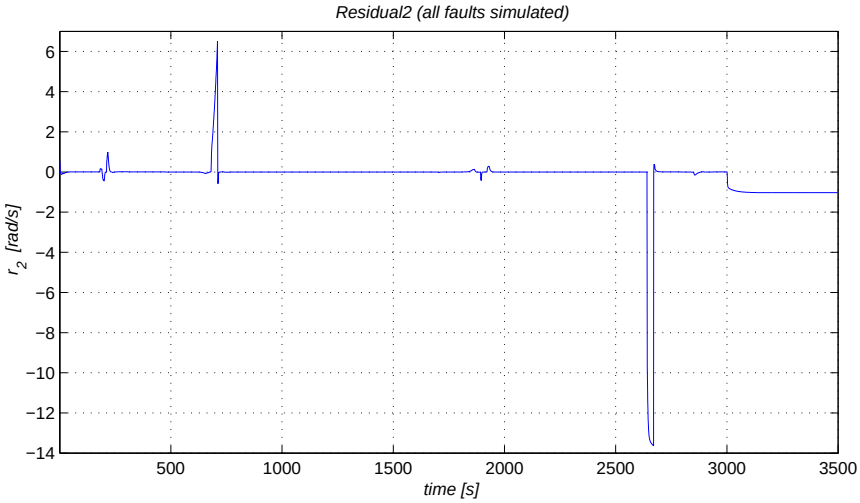

Fig. 10.37. Residual $r_{TQ_2} = n_m - \hat{n}$. Simulation including all faults and no measurement noise.

10.3.6 Quantised systems approach to the diagnosis of the pitch control loop

As an alternative approach to the diagnosis of the ship propulsion system, the quantised systems approach explained in Chapter 9 is applied to the pitch control loop, which is shown in Fig. 10.33 in the right upper corner. The loop is re-drawn as the block diagram shown in Fig. 10.38, where the input θ_{ref} is represented by the left arrow and the measured pitch angle θ_m by the right arrow. The two faults, which affect this loop, are shown by the arrows labeled as $\Delta\theta$ and $\Delta\dot{\theta}_{inc}$ where the first

represents the sensor fault, which may have the two values $\Delta\theta_{low}$ or $\Delta\theta_{high}$, and the second a fault in the hydraulic system.

This figure depicts how the structure proposed in Fig. 9.3 on p. 452 is applied to a part of the ship propulsion problem. The following investigations will show that the faults change the behaviour of the pitch control loop in such a way that they can be detected by means of rough measurement sequences that result from a quantisation of the numeric data that represent the evolution of θ_{ref} and θ_m. This diagnostic method is robust against model uncertainties and measurement noise, because it is based on rough data.

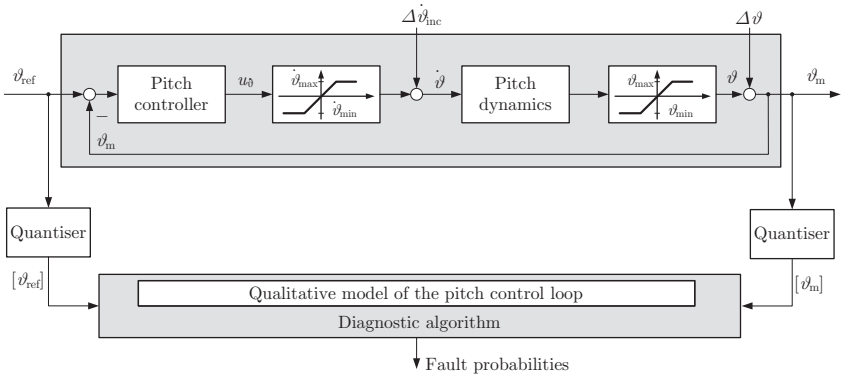

Fig. 10.38. Quantised systems approach to the diagnosis of the pitch control loop

Qualitative modelling of the pitch control loop. According to Section 9.4, the pitch control loop together with the quantisers can be described by a stochastic automaton, which is obtained by the abstraction algorithm. The quantitative model is given by Eqs. (10.12), (10.14). It is used here in a discrete-time version for the sampling time $T_s = 1\,\mathrm{s}$. The quantisers are chosen so as to get, on the one hand, a low number of symbolic input and output values but, on the other hand, to obtain enough information about the performance of the pitch control loop.

Figure 10.39 shows how the quantisation intervals of the measured pitch angle have been chosen. The quantisation of the input, which is the reference input for the measured pitch angle, is chosen accordingly. The size of the intervals are comparable with the size of the measurement noise. Due to this noise, a smaller quantisation resolution is useless. On the other hand, the diagnostic results will show that the resolution used is small enough for the diagnostic purposes, i.e., the quantised input and output sequences provide enough information for detecting and identifying the faults.

With these quantisers, the abstraction algorithm developed in Section 9.4.3 has been applied. The result is a stochastic automaton, which cannot be shown here due

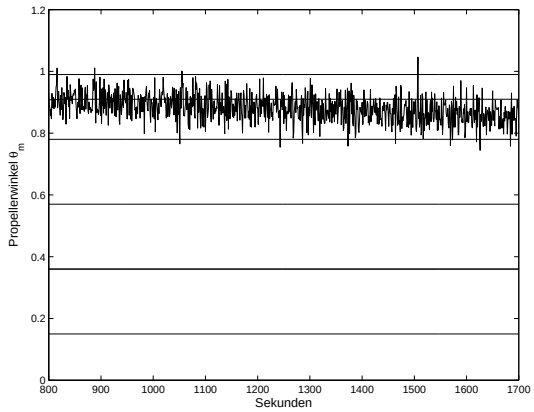

Fig. 10.39. Measured pitch angle in the quantised output space

to its size. This automaton is a complete model of the pitch control loop together with the chosen quantisers.

Diagnostic results. The diagnostic algorithm has been tested for the pitch control loop by using input and output sequences, which have been obtained by quantising input and output signals used in simulation studies of the ship propulsion system. In a long simulation run, the three different fault situations, which concern the pitch control loop, have been simulated during the time intervals given in the table.

Fault symbol	Numerical fault value	Activation time	Final time
$\Delta\theta_{high}$	1	180 s	210 s
$\Delta\theta_{low}$	-0.7	1880 s	1920 s
$\Delta\dot{\theta}_{inc}$	$\frac{0.00001}{s+0.0001}$	800 s	1700 s

Figure 10.40 shows the trajectory of the measured pitch angle θ_m in a time interval, in which the first fault was present. The diagnostic algorithm has no access to this measurement values, but obtains merely the quantised version of them. The figure is used here to explain the diagnostic result.

The diagnostic algorithm finds the fault $\Delta\theta = \Delta\theta_{high}$, which represents a positive measurement error, at once. This result is visible in Fig. 10.41, where the rectangle showing this fault (called “Sensorfault high”) is black from the second time instant shown, which means that the diagnostic result associates with this fault a high probability (which is nearly one over the whole time interval where the fault is present).

At time 210 seconds, the fault disappears. Hence the measured pitch angle is much lower than before (cf. Fig. 10.40), which imitates the contrasting fault $\Delta\theta_{low}$. This behaviour explaines why the diagnostic algorithm associates with this fault $\Delta\theta_{low}$

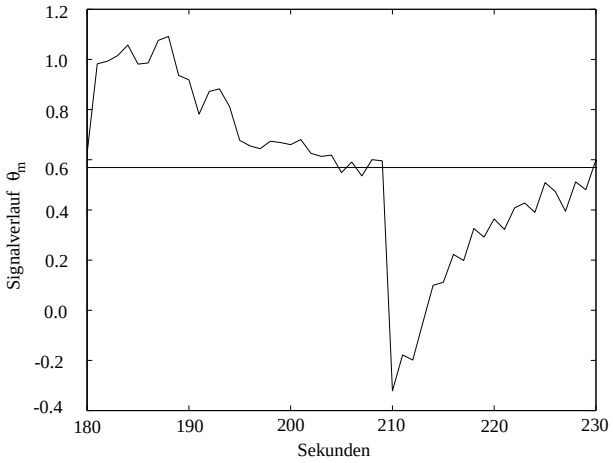

Fig. 10.40. Quantitative trajectory of the measured pitch angle

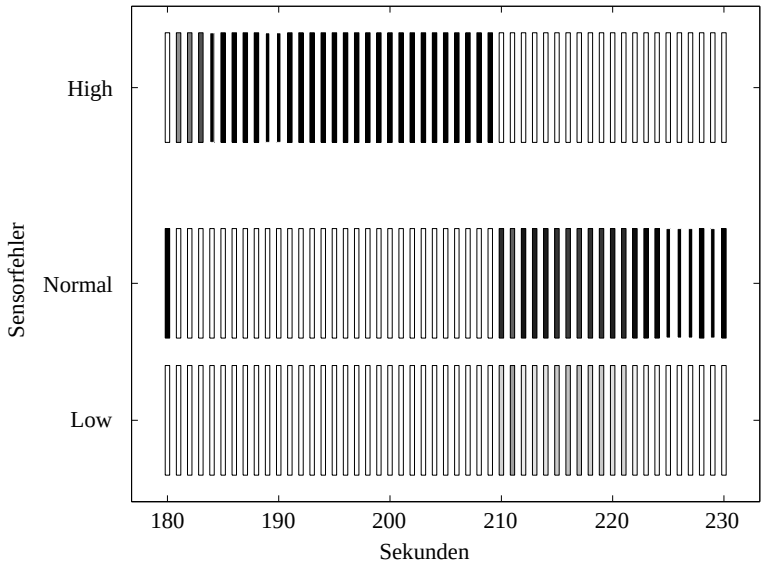

Fig. 10.41. Diagnostic result for sensor fault $\Delta\theta_{high}$ between 180 and 210 seconds

Fig. 10.42. Quantitative trajectory of the measured pitch angle for sensor fault $\Delta\theta_{low}$

a low (but positive) probability at the time instant 211. However, as the diagnostic algorithm does not only use the system output, but checks the consistency of the (quantised) input and output sequences with the stochastich automaton, it finds out that the decrease of the measurement values is the results of a disappearance of the fault $\Delta\theta_{high}$ and not due to the low-measurement fault $\Delta\theta_{low}$. Hence, the diagnostic algorithm gets the right result showing that the system is faultless.

For the second fault $\Delta\theta_{low}$, the pitch angle behaves approximately in the opposite way as before (cf. Fig. 10.42). The diagnostic algorithm finds this fault at once and behaves similarly as in the first fault case, when the second fault disappears at time instant 1920 seconds (Fig. 10.43).

Finally, the diagnosis of the pitch control loop subject to the hydraulic fault is considered. As Fig. 10.44 shows, the measured pitch angle changes very slowly. Hence, the effect of the fault is detectable rather late. Therefore, the figures are drawn with a much higher sampling time as before.

Figure 10.45 shows that the fault is detected at time around 1400 seconds.

The results show that the faults in the pitch control loop can by found by a diagnostic method that uses only quantised measurement signals. Both sensor faults are identified at once, whereas the hydraulic fault, which changes the system behaviour slowly, is identified rather late. If the hydraulic fault appears as an incipient fault (cf. Table 10.9), this method is not quick enough for its identification. However, even with precise numerical measurements, this fault turns out to be hard to detect (cf. Section 10.3.5).

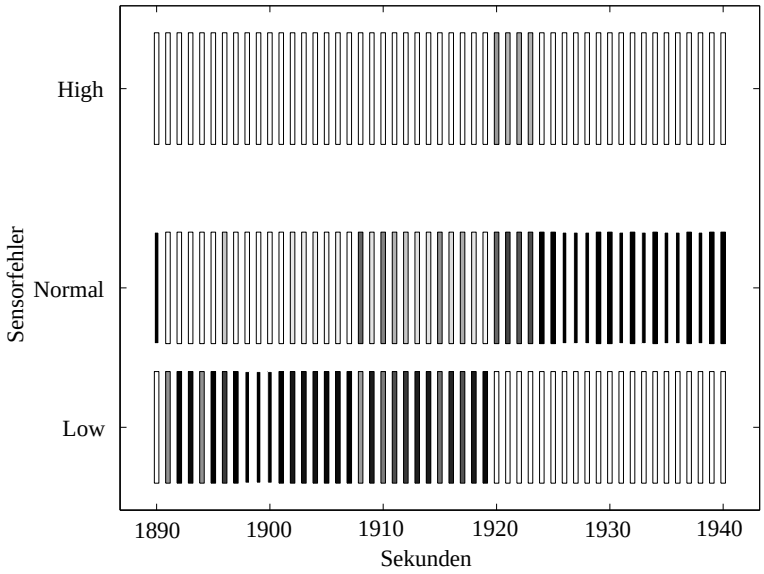

Fig. 10.43. Diagnostic result for sensor fault $\Delta\theta_{low}$ between 1880 and 1920 seconds

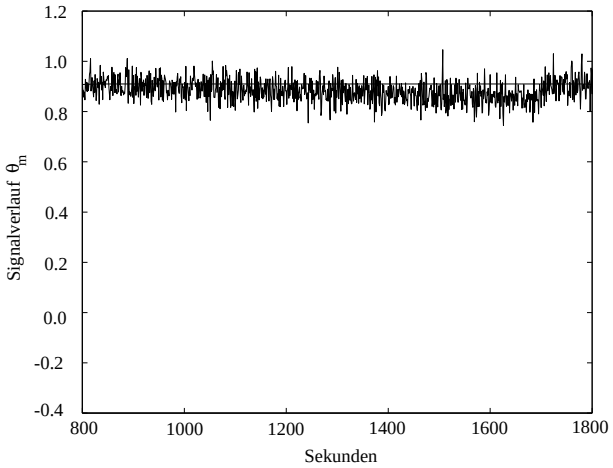

Fig. 10.44. Quantitative trajectory of the measured pitch angle for hydraulic fault $\Delta\dot{\theta}_{inc}$

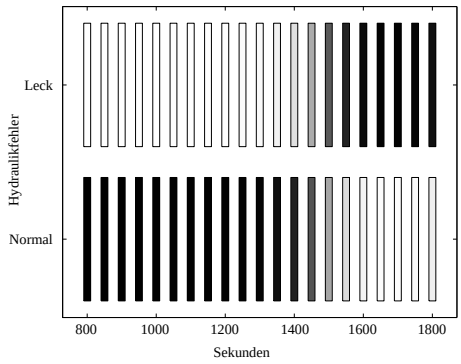

Fig. 10.45. Diagnostic result for hydraulic fault $\Delta\dot{\theta}_{inc}$ between 800 and 1700 seconds

10.3.7 Fault-tolerant propulsion

Active reconfiguration is achieved at the lower level of the ship propulsion system. Figure 10.46 shows how software redundancy, in the case where shaft speed measurement has failed, is implemented. The redundant module consists of the nonlinear observer that uses the fuel index Y_m and the propeller pitch measurement ϑ_m as input and the ship speed measurement, U_m, as the system output. As mentioned earlier, this observer is independent of faults in the shaft speed measurement. However, it can correctly provide an estimate of the shaft speed, which in turn is used as a (software) redundant information when a fault occurs.

The block containing the observer and fault isolation modules in Fig. 10.46 is the fault detection and isolation block and originally contains several residual generation and fault identification modules, which are not all explained here.

A shaft speed sensor fault occurs at time $680\,\mathrm{s}$ Fig 10.47 shows a zoom of the time responses of the shaft speed and the fuel index when the fault occurs at the worst instant in time, during a transient behaviour caused by alteration of the handle command simultaneously with the diesel torque being close to the maximum limit. A statistical fault detection method (CUSUM) detects the sensor failing high at time 681 s and generates a fault event. The state in the decision logic changes to *n_faulty* and the fault is accommodated on 682^{th} s where the effector alters the faulty measurement signal with the estimate generated by the nonlinear observer.

The transient behaviour following the fault causes the overload controller rapidly to reduce the pitch angle. This rapid load reduction makes it difficult for the shaft speed controller to reduce the shaft speed. Even in this extreme case, the overshoot in shaft speed is some $5\,\%$, which is below the critical limit of over-speed shut down (is $9\,\%$) of the main engine. The transient thus has no critical effects and is certainly acceptable compared with the alternative, which would be instant loss of propulsion of the ship. In each of the figures, the curves represent the following cases: normal case (solid line), faulty case (dash dotted line), and re-configured case (dashed line), reference signal (dotted line). The resulting overshoot is now well below the critical

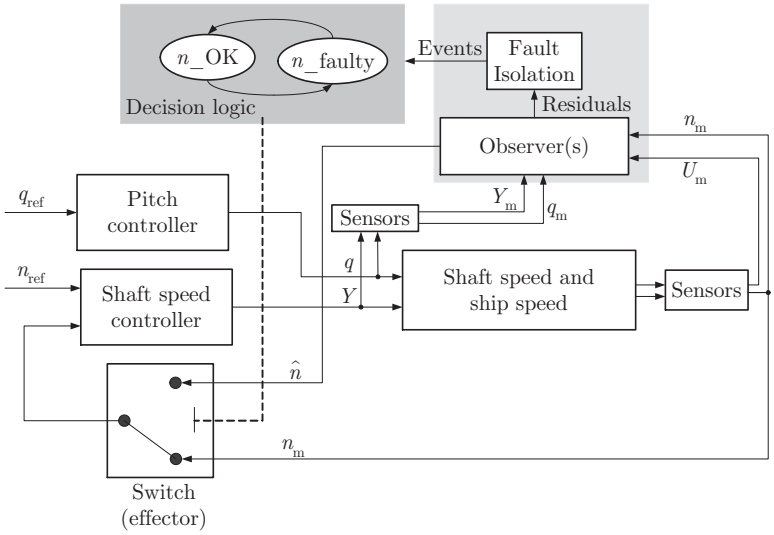

Fig. 10.46. Active reconfiguration scheme for the shaft speed when the non-linear observer is used

over-speed shut down limit and the effort of designing the adaptive observer was well spent.

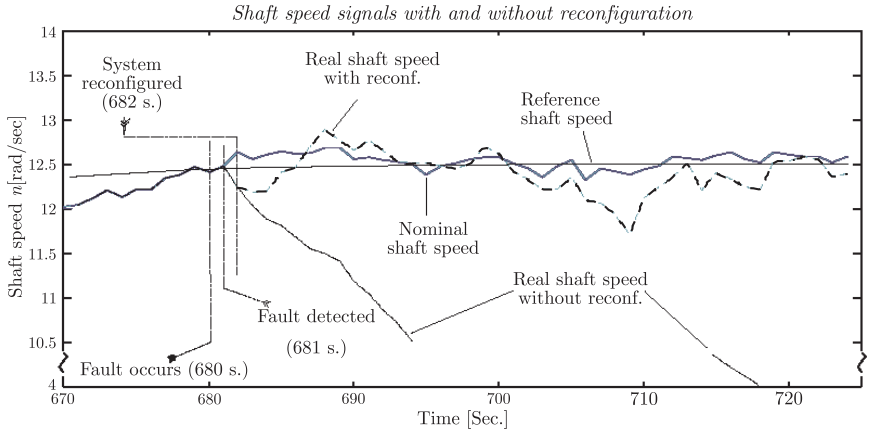

Fig. 10.47. A zoom of the shaft speed and fuel index values in the worst case, using the observer-generated shaft speed (Experiment 1)

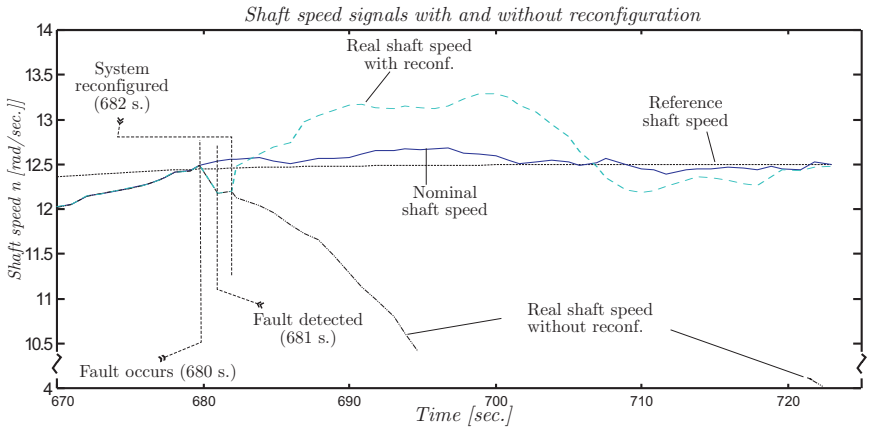

Fig. 10.48. A zoom of the shaft speed and fuel index values in the worst case, using the observer-generated shaft speed (Experiment 2)

10.4 Supervision of a steam generator

10.4.1 Description of the process

This section shows how a diagnostic algorithm can be found for a steam generator by using structural analysis.

The steam generator is a pilot process available at the University of Lille (France). A general view of the installation is given on Fig. 10.49.

Fig. 10.49. General view of the steam generator

The energy of the primary loop is produced by an electric heating, while the energy of the steam (which would be really used in industrial applications, e.g. by providing it to a turbo-alternator in a power station), is here simply dissipated by a cooling system that is composed of a set of modulating valves and a condenser coupled with a heat exchanger.

In the following a part of the installation which is composed of the water feeding circuit and of the 175 litres boiler together with the heating is considered. The technological description is given by Fig. 10.50.

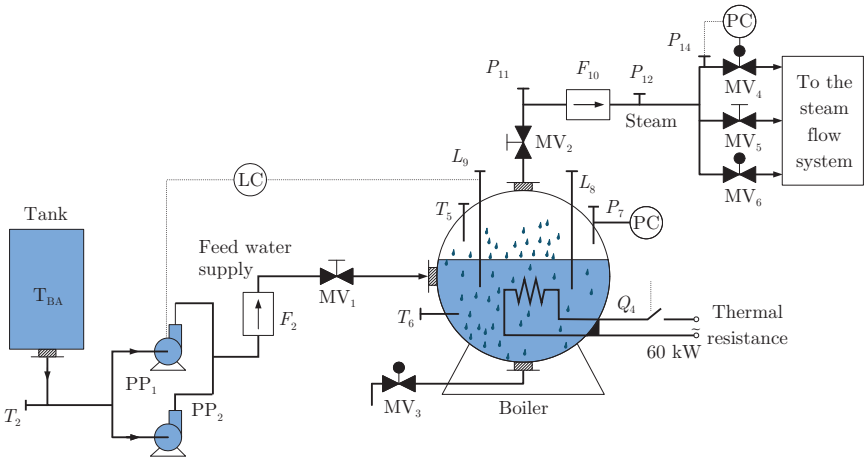

Fig. 10.50. Scheme of the process

The different system variables are labelled as:

F: massic flow (kg/s)
T: temperature ($^\circ C$)
P: pressure (bar)
L: level (litre).

Labels with a numerical index denote a sensor output while labels with a litteral index stand for the model variables. For example, P_7 is the output of the pressure sensor while P_{GV} is the pression as computed from the model. Thermal power (measured in J/s) which is transfered by convection or conduction, respectively, is labelled by $\dot{H}$ and $\dot{Q}$.

Water feeding the steam generator is stored in the tank at temperature T_{BA} and then fed into the boiler through the pump PP_1 at constant speed. Pump PP_2 is in parallel with PP_1 and stands as redundant hardware. The water level L_{GV} and the pressure P_{GV} in the boiler are measured by the sensors L_8, L_9 and P_7. They are regulated by the two on-off regulators PC (pressure control) and LC (level control). These controllers act on the heating power of the thermal resistor $\dot{Q}_{TH}$, which is measured by sensor Q_4, and on the flow of pump F_P which is measured by sensor

F_2. The produced steam, whose nominal flow is $F_{VG} = 60\,\mathrm{kg/h}$ (measured by sensor F_{10}), is subject to an isenthalpic depressurisation , due to a set of modulating valves $MV_{4,}$ MV_5 and MV_6, until the fixed pressure P_{VD} is reached. This pressure, measured by sensors P_{12} and P_{14} just ahead of the modulating valves, is regulated by a pressure controller (PC). The pressure drop between them being neglectible, the two measurements are redundant. The steam and saturated water temperatures in the boiler (both equal to T_{GV}), are measured by the thermocouples T_5 and T_6. The manual valve MV_3 on the emptying pipe of the boiler allows to create a water leakage in the steam generator, while the manual valve MV_2 on the output steam pipe allows to create a pipe clogging. MV_5 is a by-pass valve which also allows to create a steam leakage.

10.4.2 Modeling of the steam generator

The steam generator is a dynamical nonlinear system. It has energy of three domains, namely hydraulic energy (flows of fluids in pipes), electric energy (heating power), thermal energy (production of steam, thermal exchanges) and mechanical energy (pumps, valves). Three subsystems are distinguished:

1. the feedwater circuit (pump, valve, pipe)
2. the steam generation process
3. the thermal resistor .

The modelling hypotheses are as follows:

- Water and steam are saturated. Thermodynamical properties are calculated at equilibrium (this is justified by the assumption that the water-steam mixture is homogeneous).
- The water-steam mixture is at a uniform pressure P_{GV} i.e. the effect of the superficial tension of steam bubbles is neglected.
- More generally, all variables have uniform values, due to the small size of the boiler.
- The steam generator has known thermal capacity and it suffers from thermal losses by conduction to the external environment.
- The liquid in the feeding circuit is uncompressible.
- The produced steam is compressible.

Feedwater circuit. The feedwater circuit is composed of a number of pipes, of a hydraulic restriction, of two parallel pumps, and a manual valve MV_1. Water is pushed by the on-off controlled hydraulic pumps according to the water level L_{GV} in the steam generator.

Hydraulic model. The hydraulic model is intended to determine the water flow F_{AL0} in the boiler feeding pipe. This flow is obtained by the intersection of the

chraracteristics of the pump F_{PA} (given by the provider) and of the pipe F_{AL} (Fig. 10.51).

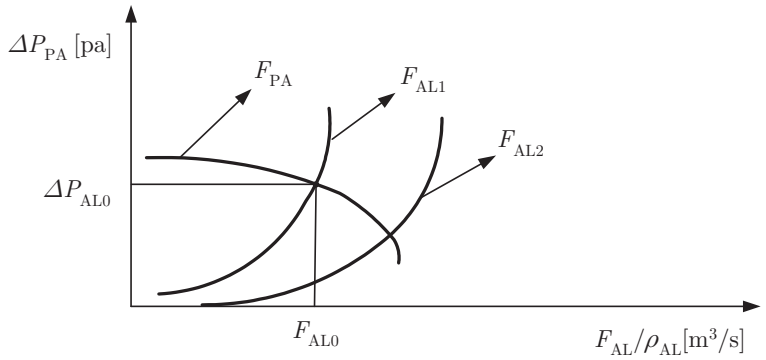

Fig. 10.51. Determination of the feeding flow

The curves F_{AL1} and F_{AL2} represent the pipes characteristics for different values of its hydraulic resistance. The pump characteristics is given as a function which links the pressure difference ΔP_{PA} (pascal) between the input and the output, and the volumic flow F_{PA} ρ_{AL}.

The determination of the massic flow F_{AL} as a function of the output pressure in the Tank P_{SB} and of the pressure in the steam generator P_{GV} just consists in solving the following system of equations:

$$\begin{cases} \frac{F_{AL}}{\rho_{AL}} = \left(-8,4948 \cdot 10^{-10} \Delta P_{PA} + 9,722 \cdot 10^{-4}\right) b_1 \\ \frac{F_{AL}}{\rho_{AL}} = \sqrt{\frac{(P_{PA}-P_{GV})10^5}{K_D(z_{V1})}}. \end{cases}$$

ΔP_{PA} is the pressure drop between the input and the output of the pump:

$$\Delta P_{PA} = 10^5 \left(P_{PA} - P_{SB}\right).$$

P_{SB} is the (known) pressure at the output of the tank and P_{PA} is the pressure at the output of the pump. The pressure drop between the tank and the input of the pump is negligible compared with the drop between the output of the pump and the boiler.

ρ_{AL} is the water density, P_{GV} the pressure inside the boiler and $K_D(z_{V1})$ the pressure drop coefficient associated with the configuration of the pipe (lenght, number of turns, etc). The latter also depends on the opening degree of the manual valve MV_1. b_1 is a Boolean variable associated with the output of the controller, which depends on the level difference between the water level in the steam generator L_{GV} and the reference water level L_{GV_ref} , and on Δ, the dead zone of the relay:

$$b_1 = F(\ \mathrm{sign}\ (L_{GV\_ref} - L_{GV}), \Delta).$$

Thermal model. Water is fed into the boiler through a calorifugated pipe, into which it arrives at ambiant temperature T_{Al}. The operating regime is discontinuous, and thus the thermal losses, by radiation, convection and conduction can be neglected. Therefore, the enthalpy flow $\dot{H}_{AL}$ (J/s) carried by the feedwater flow is equal to that at the output of the tank $\dot{H}_{SB}$:

$$\dot{H}_{AL} = \dot{H}_{SB} = F_{AL} \cdot h_{AL}.$$

The specific enthalpy of the water is h_{AL} (J/kg)

$$h_{AL} = c_{pe} T_{AL},$$

where c_{pe}, the specific heat of the water, is practically constant but depends weakly on the temperature and varies from 4180 to $4200\,\mathrm{J}\cdot\ \mathrm{kg}^{-1}\cdot{}^{\circ}\mathrm{C}^{-1}$ for temperature between 15 and 100°C. Since the feedwater temperature varies between 30 and 60°C, the model is written as

$$\dot{H}_{AL} = 4200 F_{AL} T_{AL}.$$

It should be noticed that the state of the water has to be evaluated with respect to a basic reference temperature, which is taken at $T_0 = 0°C$. The specific enthalpy of liquid water is thus considered to be $h_{AL} = 0$ at $T_{AL} = 0$.

Hydraulic model of steam accumulation. The variation of the mass of the water-steam mixture M_{GV}(kg) in the boiler is

$$\frac{d}{dt}(M_{GV}) = \dot{M}_{GV} = F_{AL} - F_{VG},$$

where F_{VG} represents the massic flow of the steam at the output of the boiler, which flows through the manual valve MV_2.

Many models can be used to compute the massic flow F of a compressible fluid, depending on the operating regime of the physical process. A well known model is given by

$$\begin{aligned} F &= K_D \tfrac{P_1}{\sqrt{T_1}} \;\; if \;\; P_2 < 0.5 P_1 \\ F &= K_{D1} \sqrt{(P_1 - P_2)\,\tfrac{P_2}{T_1}} \qquad if\, P_2 > 0.5 P_1, \end{aligned} \tag{10.27}$$

where K_D is some flow coefficient, and P_1, T_1, T_2 and P_2 are respectively the pressures and temperatures at the input (the output) of the pipe restrictions. It is worth noticing that Bernoulli's law (which links the flow only with the difference of pressures) is valid only for uncompressible fluids.

In the present case, the pressures at the input and output of valve MV_2 are respectively the relative pressure within the steam generator P_{GV} (P_1) and the pressure at the input of the depressurisation valves P_{VD} (P_2). This pressure is measured by sensors P_{12} and P_{14}. Taking into account relation (10.27), it follows

$$F_{VG} = \sqrt{K_D(z_{V2}).\frac{(P_{GV} - P_{VD})\,P_{VD}}{T_{GV}}}. \tag{10.28}$$

In Equation (10.28), $K_D(z_{V2})$ is the pressure drop coefficient associated with the steam output pipe. It depends on the valve MV_2 position and it is experimentally determined.

Thermal model of the boiler. The energy balance equation in the steam generator is given by

$$\frac{d(H_{GV})}{dt} = \dot{H}_{GV} = \dot{Q}_{TH} + \dot{H}_{AL} + \dot{E}_W - \dot{H}_{VG} - \dot{Q}_{PG},$$

where h_{GV} or H_{GV} are, respectively, the specific enthalpy and the total enthalpy in the boiler, and $\dot{Q}_{TH}$ is the thermal heating power

$$\dot{Q}_{TH} = b_2.Q_4 = b_2.60000 \text{ W}.$$

This power is generated by a $60\,\text{kW}$ heating resistor, measured by sensor Q_4, and controlled by the on/off regulator, which acts on the pressure P_{GV} in the steam generator. b_2 (the relay output) is the Boolean control, which depends on the difference between the pressure in the steam generator P_{GV} and its reference value P_{GV_ref}, taking into account the relay's dead zone Δ.

$$b_2 = F(\text{ sign } (P_{GV\_ref} - P_{GV}), \Delta).$$

$\dot{E}_W$ is the power (J/s) provided by the work of the pressure forces, which is computed from the relation

$$\dot{E}_W = V_{GV}\dot{P}_{GV}$$
$$\dot{P}_{GV} = \frac{dP_{GV}}{dt}.$$

V_{GV} (m^3) is the geometric volume of the boiler, namely $0,175\,\text{m}^3$.

The total enthalpy flow carried by the steam is proportional to the specific enthalpy of the steam h_V and to its massic flow F_{VG}:

$$\dot{H}_{VG} = F_{VG} \cdot h_V.$$

$\dot{Q}_{PG}$ is the thermal power dissipated by conduction from the water-steam mixture to the metal body of the boiler. It is calculated below, in the thermal losses.

Modeling the thermal losses. Let c_{MG} be the thermal capacity of the metal body of the boiler. Two cases are in general considered when modelling such systems:

- In the first case, the thermal capacity is taken into account, but the isolation between the metal body and the ambiant atmosphere is considered as perfect, so that the losses are neglected.

- In the second case, radiation/conduction is present between the metal body and the outside world.

In the Lille process, experiments have shown that thermal losses (by radiation and conduction) could not be neglected, since several parts of the pipes, sensors, valves

are not calorifugated. Moreover, the mass of the metal body of the boiler cannot be neglected ($102\,\mathrm{kg}$).

The energy balance associated with the metal body of the boiler, whose volume is V_{MG} and density is ρ_{MG} is given by

$$\rho_{MG} V_{MG} c_{MG} \frac{dT_{MG}}{dT} = \dot{Q}_{PG} - \dot{Q}_{EX},$$

where

$$\dot{Q}_{PG} = K_{GM} \left(T_{GV} - T_{MG} \right).$$

$\dot{Q}_{EX}$ is the dissipated thermal power to the external world, whose temperature T_{EX} is supposed to be constant. Parameters ρ_{MG} and c_{MG} depend on the kind of metal which constitutes the body of the boiler

$$\dot{Q}_{EX} = K_{EX} \left(T_{MG} - T_{EX} \right).$$

K_{GM} and K_{EX} are heat exchange coefficients. K_{GM} depends on the length l_C of the pipe, of its external and internal diameters D_{CE} and D_{CL}, on the volumic mass ρ_V of the steam, on the thermal conductivity coefficient λ_V, on the dynamical viscosity μ_V, and on the radiation coefficient R_{AY}. K_{EX} depends on the ambiant atmosphere properties.

Description of the two-phase water-steam mixture. Two possibilities exist for modelling a two-phase mixture:

1. Write the balance equations for each of the two phases, and write the equations which model the exchanges between them. However, these equations are not well known, and this leads to a very huge number of equations.
2. Consider global equations which apply to the two phases simultaneously. This approach, although not very rigourous, is the only one which allows to obtain practical models, particularly because it involves the so-called *quality of the steam* X, which represents the ratio between the steam and the water in the mixture.

In a water-steam mixture, i.e. in the case of saturated steam, temperature and pressure are not independent. The pressure P_{GV} and the steam quality X are determined by the following mixture equation

$$\begin{cases} h_{GV} &= \frac{H_{GV}}{M_{GV}} = h_V(P_{GV}) \cdot X + h_L(P_{GV}) \cdot (1 - X) \\ \\ \nu_{GV} &= \frac{V_{GV}}{M_{GV}} = \nu_V(P_{GV}) \cdot X + \nu_L(P_{GV}) \cdot (1 - X), \end{cases}$$

where $h_L(P_{GV})$, $h_V(P_{GV})$, $\nu_L(P_{GV})$ and $\nu_V(P_{GV})$ are polynomial thermodynamical fonctions of the pressure P_{GV}, of the specific enthalpies h and of the massic volumes ν of the liquid and of the steam. They are determined (for a given operating regime) by a least squares optimisation approach from the water-steam equilibrium tables:

$$
\begin{aligned}
h_V(P_{GV}) &= -0,74P_{GV}^2 - 17,21P_{GV} + 2680 \\
h_L(P_{GV}) &= -0,0243P_{GV}^4 + 0,8487P_{GV}^3 - 11,9P_{GV}^2 - 99,97P_{GV} + 347 \\
\nu_V(P_{GV}) &= -5,3 \cdot 10^{-5}P_{GV}^5 + 0,00207P_{GV}^4 - 0,032P_{GV}^3 \\
&\quad +0,2498P_{GV}^2 - 1,03P_{GV} + 2,166 \\
\nu_L(P_{GV}) &= -3,59 \cdot 10^{-7}P_{GV}^3 + 1,2456 \cdot 10^{-5}P_{GV}^2 + 1,03 \cdot 10^{-3}
\end{aligned}
$$

This system thus allows to determine the steam quality X, the pressure in the water-steam mixture P_{GV} as well as the temperature T_{GV} using the following thermodynamical function:

$$T_{GV} = -0,4594P_{GV}^2 + 12,7243P_{GV} + 99,003.$$

10.4.3 Design of the diagnostic system

Specifications. Faults can occur in the process itself, or in the sensors and actuators, e.g. as a change in the parameters of some component. The specifications are intended to list those faults which have to be considered. Each fault is then associated with one or several equations of the system model, i.e. with variables or parameters of that model.

The diagnostic system will be associated with given detection and isolation quality levels, evaluated e.g. by the false alarm and missed-detection probabilities, the detection delay, the possibility to find out which fault really occured, among the different possible ones.

The main goal of a steam generator is to deliver a steam flow with given pressure and temperature. The first considered fault is thus associated with the output steam flow, i.e. with the (partial or total) clogging of the output pipe, which can be modelled as a change in the pressure drop coefficient K_D.

In industrial power plants it is important to completely separate the primary and the secondary loops. This means that no leakage is allowed. Therefore, the second considered fault is a leakage in the steam generator. Such a fault will influence the thermal energy H_{GV} and the mass M_{GV} accumulated in the boiler.

The measurements are essential for the control of the process, thus leading to consider sensor faults $\{T_2, F_3, Q_4, T_5, T_6, P_7, F_{10}, P_{11}, P_{12}, P_{14}\}$ in our list.

Design of the analytic redundancy relations. The system model can be decomposed into two subsets of relations. The behaviour of the process is described by the relations of the above model (RM), which obviously concern only letter-indexed variables. The knowledge available in real time about the behaviour is described by the relations RC which are just a way of showing which system variables are measured.

Behaviour model. From the previous equations, the behaviour model is as follows

$$
\begin{aligned}
\boldsymbol{RM1}: \quad & T_{GV} && = -0,4594P_{GV}^2 + 12,7243P_{GV} + 99,003 \\
\boldsymbol{RM2}: \quad & h_{GV} && = h_V X + h_L\,(1-X) \\
\boldsymbol{RM3}: \quad & \nu_{GV} && = \nu_V X + \nu_L(1-X) \\
\boldsymbol{RM4}: \quad & h_L && = -0,0243P_{GV}^4 + 0,8487P_{GV}^3 - 11,9P_{GV}^2 + \\
& && +99,97P_{GV} + 347 \\
\boldsymbol{RM5}: \quad & h_V && = -0,74P_{GV}^2 + 17,21P_{GV} + 2680 \\
\boldsymbol{RM6}: \quad & \nu_L && = -3,59\cdot 10^{-7}P_{GV}^3 + 1,2456\cdot 10^{-5}P_{GV}^2 + 1,03\cdot 10^{-3} \\
\boldsymbol{RM7}: \quad & \nu_V && = -5,3\cdot 10^{-5}P_{GV}^5 + 0,00207P_{GV}^4 - 0,032P_{GV}^3 \\
& && +0,2498P_{GV}^2 - 1,03P_{GV} + 2,166 \\
\boldsymbol{RM8}: \quad & \nu_{GV} && = \tfrac{V_{GV}}{M_{GV}} \\
\boldsymbol{RM9}: \quad & h_{GV} && = \tfrac{H_{GV}}{M_{GV}} \\
\boldsymbol{RM10}: \quad & F_{VG} && = \sqrt{K_D(z_{V2})\cdot \tfrac{(P_{GV}-P_{VD})P_{VD}}{T_{GV}}} \\
\boldsymbol{RM11}: \quad & \dot{M}_{GV} && = \tfrac{dM_{GV}}{dt} \\
\boldsymbol{RM12}: \quad & \dot{M}_{GV} && = F_{AL} - F_{VG} \\
\boldsymbol{RM13}: \quad & \dot{H}_{GV} && = \tfrac{dH_{GV}}{dt} \\
\boldsymbol{RM14}: \quad & \dot{H}_{GV} && = \dot{Q}_{TH} + \dot{H}_{AL} + \dot{E}_W - \dot{H}_{VG} - \dot{Q}_{PG}
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{RM15}: \quad & \dot{H}_{AL} && = 4200F_{AL}T_{AL} \\
\boldsymbol{RM16}: \quad & \dot{H}_{VG} && = F_{VG}.h_V \\
\boldsymbol{RM17}: \quad & \dot{Q}_{PG} && = K_{GM}\,(T_{GV} - T_{MG}) \\
\boldsymbol{RM18}: \quad & \rho_{MG} && V_{MG}c_{MG}\cdot \dot{T}_{MG} + T_{MG}\,(K_{EX} + K_{GM}) \\
& && = K_{GM}T_{GV} + K_{EX}T_{EX} \\
\boldsymbol{RM19}: \quad & \dot{T}_{MG} && = \tfrac{dT_{MG}}{dt} \\
\boldsymbol{RM20}: \quad & \dot{E}_W && = V_{GV}\dot{P}_{GV} \\
\boldsymbol{RM21}: \quad & \dot{P}_{GV} && = \tfrac{dP_{GV}}{dt}
\end{aligned}
$$

This model is a set of 21 constraints (nonlinear algebraic and differential equations), which link 25 unknown variables.

From the physical analysis of the process, the state vector is of dimension three. Two state variables are associated with the accumulation in the boiler, namely the thermal energy H_{GV} and the mass M_{GV}. The third state variable is associated with the accumulation of the thermal energy in the metal body of the boiler. It is here represented by the temperature T_{MG}.

In general, the initial conditions being unknown, only derivative causality can be used for the determination of the analytic redundancy relations. However, in this case, the initial conditions can be derived at each operation of the steam generator. Indeed, the initial mass $M_{GV}(0)$ of the mixture follows from the relation

$$
\begin{aligned}
M_{GV}(0) &= V_L(0)\rho_L(0) + V_V(0)\rho_V(0) \\
&= V_L(0)\rho_L(0) + [V_{GV} - V_L(0)]\rho_V(0),
\end{aligned}
$$

where $V_L(0)$ is the initial volume of the liquid in the boiler, which is fixed (for a given pressure $P_{GV}(0)$) by the reference of the level regulation system ($0.146\,\mathrm{m}^3$ in our application). The volume $V_V(0)$ of the steam then follows from V_{GV}, the total volume of the boiler, given by its geometry. The volumic masses of the steam, $\rho_V(0)$ and of the liquid $\rho_L(0)$ are determined from the thyermodynamical tables at pressure $P_{GV}(0)$.

The total initial enthalpy accumulated in the steam generator is then

$$\begin{aligned} H_{GV}(0) &= M_{GV}(0)h_{GV}(0) \\ &= M_{GV}(0)[h_V(0).X(0) + h_L(0)(1 - X(0))]. \end{aligned}$$

The initial specific enthalpy of the mixture, $h_{GV}(0)$, is determined as a function of the initial steam quality, $X(0)$ which is computed by

$$X(0) = \frac{M_V(0)}{M_V(0) + M_L(0)} = \frac{M_V(0)}{M_{GV}(0)}$$

and of the steam – $h_V(0)$ – and water – $h_L(0)$ – enthalpies, determined from thermodynamical tables. The initial temperature of the boiler body $T_{MG}(0)$ results from the differential equation $RM18$ in static regime, where the initial temperature in the boiler $T_{GV}(0)$ depends on the saturation pressure $P_{GV}(0)$

$$T_{MG}(0) = \frac{K_{GM}T_{GV}(0) + K_{EX}T_{EX}}{K_{GM} + K_{EX}}.$$

Sensors. The constraints associated with the sensors are the following:

$$\begin{array}{rrll} \boldsymbol{RC1}: & T_2 & = T_{AL} & (T_2,\ T_{AL}:\ ^\circ\,\mathrm{C}) \\ \boldsymbol{RC2}: & \frac{F_3}{3600} & = F_{AL} & (F_3:\ \mathrm{kg/h},\ F_{AL}:\ \mathrm{kg/s}) \\ \boldsymbol{RC3}: & Q_4 \cdot 1000 & = \dot{Q}_{TH} & (Q_4:\ \mathrm{kW}, \dot{\mathrm{Q}}_{TH}:\ \mathrm{W}) \\ \boldsymbol{RC4}: & T_5 & = T_{GV} & (T_5,\ T_{GV}:\ ^\circ\,\mathrm{C}) \\ \boldsymbol{RC5}: & T_6 & = T_{GV} & (T_6,\ T_{GV}:\ ^\circ\,\mathrm{C}) \\ \boldsymbol{RC6}: & P_7 & = P_{GV} & (P_7,\ P_{GV}:\mathrm{bar}) \\ \boldsymbol{RC7}: & \frac{F_{10}}{3600} & = F_{VG} & (F_{10}:\ \mathrm{kg/h},\ F_{VG}\ \mathrm{kg/s}) \\ \boldsymbol{RC8}: & P_{11} & = P_{GV} & (P_{11},\ P_{GV}:\ \mathrm{bar}) \\ \boldsymbol{RC9}: & P_{12} & = P_{VD} & (P_{12},\ P_{VD}:\ \mathrm{bar}) \\ \boldsymbol{RC10}: & P_{14} & = P_{VD} & (P_{14},\ P_{VD}:\ \mathrm{bar}). \end{array}$$

10.4.4 Structural analysis

The structural model is intended to analyse the structural properties of the system and to provide the means of creating residuals. Once these residuals have been identified, the behaviour model has to be used to determine the actual functions which have to be implemented for their real-time computation.

	P_{GV}	T_{GV}	h_{GV}	h_v	X	h_L	v_V	v_L	v_{GV}	M_{GV}	P_{VD}	F_{VG}	H_{GV}	$\dot{M}_{GV}$	F_{AL}	$\dot{H}_{GV}$	$\dot{Q}_{TH}$	$\dot{H}_{AL}$	$\dot{Q}_{PG}$	$\dot{H}_{VG}$	T_{AL}	$\dot{T}_{MG}$	T_{MG}	$\dot{E}_W$	$\dot{P}_{GV}$	T_2	F_3	Q_4	T_5	T_6	P_7	F_{10}	P_{11}	P_{12}	P_{14}
RM1	x	1																																
RM2			❶	1	1	1																1												
RM3					❶		1	1	1								**1**																	
RM4	x					❶																												
RM5	x			❶																														
RM6	x							❶																										
RM7	x						❶																											
RM8									❶	1																								
RM9			1							1			1																					
RM10	1	1									1	1																						
RM11										❶				1																				
RM12												1		❶	1																			
RM13													❶			1																		
RM14																❶	**1**	1	**1**	1	1			1										
RM15															1			❶			1													
RM16				1								1								❶														
RM17		1																	❶				1											
RM18		1																				❶	1											
RM19																						1	❶											
RM20																								❶	1									
RM21	1																								❶									
RC1																					1					1								
RC2															❶												1							
RC3																	❶											1						
RC4		1																											1					
RC5		❶																												1				
RC6	❶																														1			
RC7												❶																				1		
RC8	1																																1	
RC9											❶																							1	
RC10											1																								1

Fig. 10.52. Incidence matrix of the system structural graph

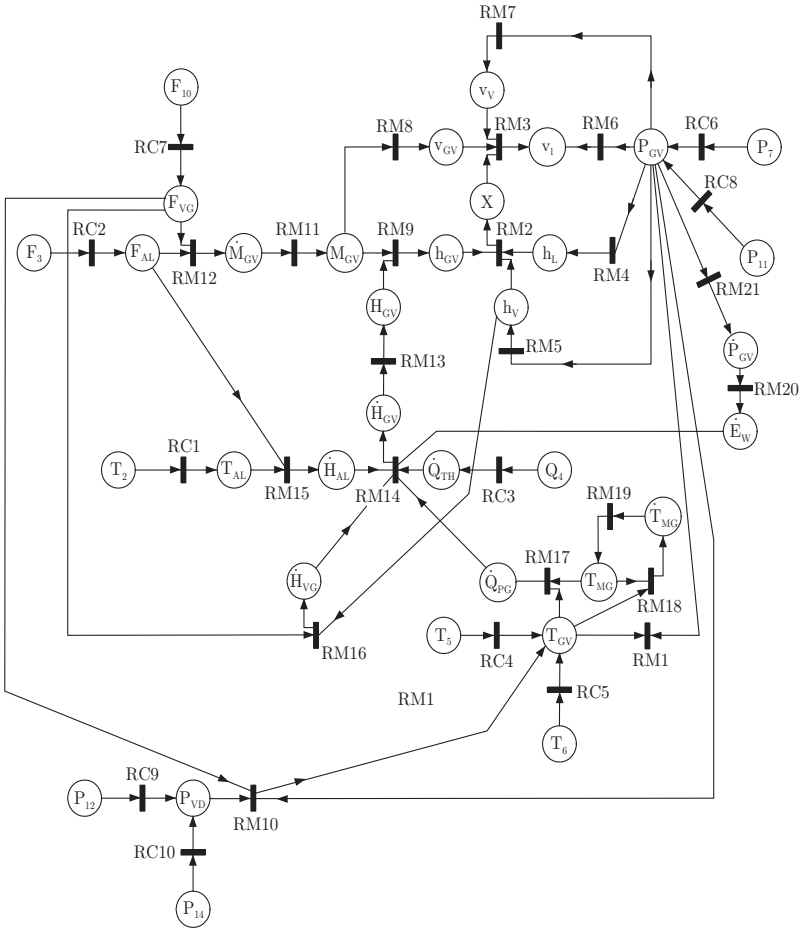

Fig. 10.53. Oriented structure graph of the boiler

The incidence matrix of the system structural graph is given by Fig. 10.53.

The analytic redundancy relations express the compatibility conditions of this system of 31 equations and 25 unknowns. They are exhibited by performing a complete matching with respect to the unknown variables, as shown on the incidence matrix, and illustrated by Fig. 10.53. The non-matched constraints are

$$NMC = \{RM1, RM9, RM10, RC4, RC8, RC10\} .$$

Computing the unknown variables as functions of the known ones by means of the matching, and then putting the result into the constraints of NMC, one obtains six relations which link only known variables, i.e. six analytic redundancy relations.

ARR1. The first $ARR\,(ARR1)$ results from $RM1$, $RC5$ and $RC6$, and the associated alternated chain is shown on Fig. 10.54.

Fig. 10.54. ARR1 subgraph

This leads to the following computations:

$$\begin{aligned}
\boldsymbol{RM1} \quad &\Rightarrow T_{GV} \quad = -0,4594P_{GV}^2 + 12,7243P_{GV} + 99,003\\
\boldsymbol{RC5}: \quad & T_6 \quad = T_{GV}\\
\boldsymbol{RC6}: \quad & P_7 \quad = P_{GV}.
\end{aligned}$$

Replacing P_{GV} and T_{GV} by their measures P_7 and T_6 into expression $RM1$, one obtains $ARR1$

$$T_6 = -0,4594P_7^2 + 12,7243P_7 + 99,003. \tag{10.29}$$

ARR2. The computation form of $ARR2$ is:

$$\begin{aligned}
\boldsymbol{RM10} \quad &\Rightarrow F_{VG} \quad = \sqrt{K_D(z_{V2})\frac{(P_{GV} - P_{VD})\cdot P_{VD}}{T_{GV}}}\\
\boldsymbol{RM1}: \quad & T_{GV} \quad = -0,4594P_{GV}^2 + 12,7243P_{GV} + 99,003\\
\boldsymbol{RC6}: \quad & P_7 \quad = P_{GV}\\
\boldsymbol{RC7}: \quad & \tfrac{F_{10}}{3600} \quad = F_{VG}\\
\boldsymbol{RC9}: \quad & P_{12} \quad = P_{VD}\\
\boldsymbol{RM10} \quad & \Rightarrow \quad \frac{(3600)^2(P_7 - P_{12})\cdot K_D \cdot P_{12}}{-0,4594P_7^2 + 12,7243P_7 + 99,003} - F_{10}^2 = 0\\
& r_2 \quad = F_{10}^2 - \frac{(3600)^2(P_7 - P_{12})\cdot K_D \cdot P_{12}}{-0,4594P_7^2 + 12,7243P_7 + 99,003}.
\end{aligned}$$

r_2 is in $\mathrm{kg}^2/\mathrm{s}^2$, it represents the link between the steam flow F_{VG}, as computed by P_{GV}, P_{VD} and T_{GV}, and F_{10}, the measured value of this flow. It is represented on Fig. 10.55.

ARR3. The successive steps for the computation of $ARR3$ are as follows.

Step 1. From constraints $RC1 - RC10$ and $RM13 - RM17$, an elimination procedure provides the following expression of H_{GV}:

$$H_{GV}^{(1)} = \int (Q_4 + 4200T_2F_3 - K_{GM}(T_5 - T_{MG}) - F_{10}h_V(P_7)dt) + H_{GV}(0).$$

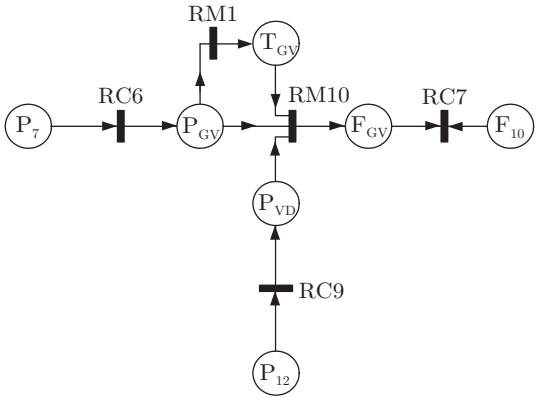

Fig. 10.55. ARR2 subgraph

Step 2. From $RM11 - RM13$, $RC2$ and $RC7$ the following relation is obtained:

$$M_{GV} = \int (F_3 - F_{10})/3600dt + M_{GV}(0).$$

Step 3. The expression of X (taken from $RM3$) is obtained using $RM8$ and the above expression of M_{GV}

$$X = \frac{0,175/\left(\int (F_3 - F_{10})/3600dt + M_{GV}(0)\right) - \nu_L(P_7)}{\nu_V(P_7) - \nu_L(P_7)}.$$

Step 4. Another expression of H_{GV} is obtained by substitution, replacing X and M_{GV} by their respective expressions in constraint $RM9$:

$$H_{GV}^{(2)} =$$

$$\left[\left(\frac{0,175/\left(\int (F_3 - F_{10})/3600dt + M_{GV}(0)\right) - \nu_L(P_7)}{\nu_V(P_7) - \nu_L(P_7)} \cdot (h_V(P_7) - h_L(P_7))\right)\right.$$

$$\left. + h_L(P_7)\right] \cdot \left(\int (F_3 - F_{10})/3600dt + M_{GV}(0)\right).$$

The residual is then

$$r_3 = H_{GV}^{(2)} - H_{GV}^{(1)}$$

$$\boldsymbol{r}_3 = \left[\left(\frac{0,175/\left(\int (F_3 - F_{10})/3600dt + M_{GV}(0)\right) - \nu_L(P_7)}{\nu_V(P_7) - \nu_L(P_7)}\right.\right.$$

$$\left.\left. \cdot\ (h_V(P_7) - h_L(P_7))\right) + h_L(P_7)\right] \cdot \left(\int (F_3 - F_{10})/3600dt + M_{GV}(0)\right)$$

$$-\left(\int (Q_4 + 4200T_2F_3 - K_{GM}(T_5 - T_{MG}) - F_{10}h_V(P_7))dt + H_{GV}(0)\right).$$

r_3 is an energy (in J) whose structure is shown on Fig. 10.56.

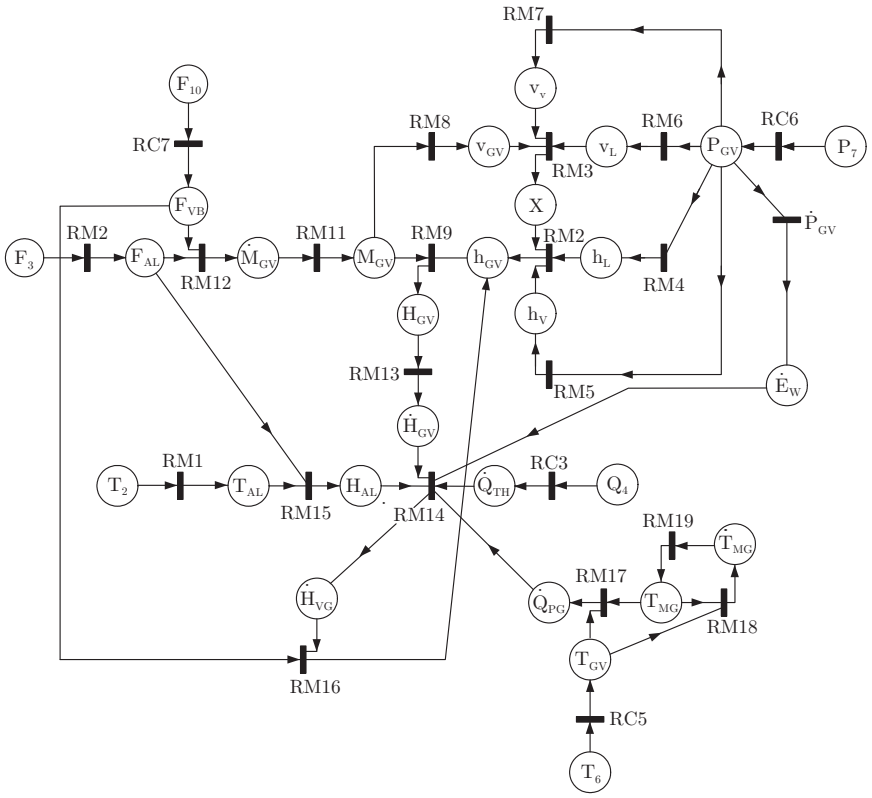

Fig. 10.56. ARR3 subgraph

Some remarks on $ARR3$ follow.

Remark 10.1 *Homogeneity*
For clarity, there is no coefficient in this expression to insure its homogeneity. In fact, the flows should be in kg/s while the sensors F_3 and F_{10} provide measurements in kg/h. The energy should be in J while sensor Q_4 provides measurements in kJ. Besides, $h_V(P_7)$ and $h_L(P_7)$ must be in J/kg. □

Remark 10.2 *Complexity*
The computation form of this residual makes use of many constraints. This follows from the fact that many variables cannot be measured (enthalpy, volumic mass, energy, quality of the steam). □

ARRs 4, 5 and 6. The three last ARRs are associated with hardware redundancy, since several sensors are measuring the same system variables.

Sensors T_5 and T_6 are both located in the boiler, but T_6 is slightly higher than T_5 and thus it measures the temperature of the steam. Under the hypothesis that the mixture is really homogeneous, both sensors should provide the same value

$$r_4 = T_5 - T_6.$$

Sensor P_7 is located in the boiler while sensor P_{11} is located at the output. Neglecting the pressure drop in the (small) pipe, both should provide the same value

$$r_5 = P_7 - P_{11}.$$

Sensors P_{12} and P_{14} are both located before the two parallel modulating valves at the beginning of the depressurisation circuit

$$r_6 = P_{12} - P_{14}.$$

10.4.5 Fault signatures

Remember that the signature of a fault is the subset of the redundancy relations which are influenced by the fault. The faults to be detected and isolated are faults in the sensors $\{T_5, T_6, P_7, P_{11}, P_{12}, P_{14}, F_{10}\}$ and in $\{M_{GV}, K_D\}$. For the system of 6 redundancy relations, the resulting signature table is given in Table 10.12.

Table 10.12 Fault signatures

$\nearrow$	T_2	F_3	Q_4	T_5	T_6	P_7	F_{10}	P_{11}	P_{12}	P_{14}	K_D	M_{GV}
r_1					1	1						
r_2						1	1		1		1	
r_3	1	1	1		1	1	1					1
r_4				1	1							
r_5						1		1				
r_6									1	1		

All twelve column vectors are different from zero, and thus all faults are detectable. Note that faults on sensors T_2, F_3 and Q_4 which were not in the specifications, can also be detected (it will be seen below that this is not exactly true, the actual sensitivity being too low). The physical interpretation of the theoretical signatures is as follows:

- Residual r_1 links the temperature T_{GV} and the pressure P_{GV} of the steam-water mixture. A clogging or a leakage should not act on this residual.
- Residual r_2 links the steam massic flow F_{VG}, the output pressure P_{VD}, the pressure in the boiler P_{GV}, and the temperature T_{GV}. A clogging should show

through this residual since coefficient K_D is explicitely present in the computation form.

- Residual r_3 links the different energies exchanged. It should be sensitive to a water leak because M_{GV} appears explicitely in its expression.
- Residuals r_4, r_5, and r_6 express hardware redundancy and thus are sensitive to the faults in the associated sensors. They should be independent of the clogging or the leakage faults.

Note that it is always possible to obtain new $ARRs$ by combining the existing ones. This can be done by substituting one variable in a relation by its expression obtained from another relation. For example, a new residual r_7 can be created from r_1 and r_5 by eliminating P_7 and, therefore, r_7 would only depend on T_6 and P_{11}.

The decision procedure now works as follows:

1. First compute the residuals and check whether they are or not different from zero (residuals being stochastic variables, this calls for hypotheses testing of change point detection algorithms).
2. Construct the system signature by putting a zero for every zero residual and a 1 for every non-zero residual, and compare the system signature with the theoretical ones given by Table 10.12.

Of course, since decision errors are always possible, the system signature might be different from any theoretical signature in the table. In that case, the nearest one is in general taken. Thus, the detection and isolation procedure will be less sensitive to errors as the distances between the theoretical signature is larger. Table 10.13 gives the Hamming distances between the 12 theoretical signatures.

It can be noticed that the distance between the signatures associated with T_2, F_3, Q_4 and M_{GV} is zero. This means that any of these faults has the same signature, and thus, it is impossible to isolate which one has occured. Namely, a fault of one of the three sensors canot be distinguished from a leakage in the boiler.

Table 10.13 Hamming distances

$\nearrow$	T_2	F_3	Q_4	T_5	T_6	P_7	F_{10}	P_{11}	P_{12}	P_{14}	K_D	M_{GV}
T_2	0	0	0	2	2	3	1	1	3	2	2	0
F_3		0	0	2	2	3	1	1	3	2	2	0
Q_4			0	2	2	3	1	1	3	2	2	0
T_5				0	2	5	3	2	3	2	2	2
T_6					0	3	3	4	5	4	4	2
P_7						0	2	3	4	5	3	3
F_{10}							0	3	2	3	1	1
P_{11}								0	3	2	2	1
P_{12}									0	1	1	3
P_{14}										0	2	2
K_D											0	2
M_{GV}												0

10.4.6 Experimental results

Normal operation. In normal operation, the residuals should be zero in the ideal case (no modelling error, no uncertainties, no measurement noise). This is not the case, since modelling errors and noises are really present. However, taking into account the relative values of the residuals with respect to the corresponding signals shows that the precision is not bad, as illustrated now for residual r_3, whose behaviour in normal operation is shown by Fig. 10.57.

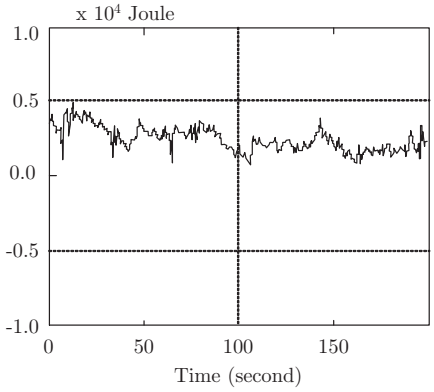

Fig. 10.57. Residual r_3 in normal operation

This residual expresses the difference between the two available ways of computing H_{GV}. Table 10.14 gives the normalised values of r_3 (%). It is seen that this residual, although nonzero in normal operation, can still be used, since its average value $\left(2,58 \cdot 10^3 \text{ J}\right)$ is very small with respect to the average value of H_{GV}, $\left(7,19^5 \text{ J}\right)$, the ratio being $0,36\,\%$.

Table 10.14 Residual r_3 analysis

	Residual r_3 value	Percentage with respect to H_{GV}
Average value	$2,5842 \cdot 10^3$	$0,36$
Standard deviation	$833,0997$	$0,12$
Maximum value	$5,033 \cdot 10^3$	$0,70$
Minimum value	$874,9006$	$0,12$

10.4.7 Fault scenarios

Sensor faults. Sensor faults are created by adding a $15\,\%$ extra signal to their output on the time window [75 s, 150 s]. T_2, F_3 and Q_4 act only on residual r_3 but this one

does not react. The reason is that although it is structurally sensitive to the faults, its actual (numerical) sensitivity is much too low (the energy associated with the faults is neglectible with respect to the energy of the generator). T_5 and T_6 are present in residuals r_1, r_3 and r_4. r_3 does not react to these faults, for the same reason as already explained. But r_1 and r_4 are really sensitive as shown by Fig. 10.58.

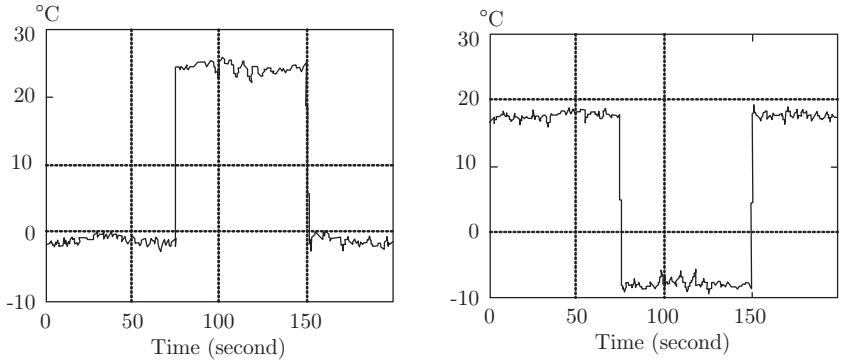

Fig. 10.58. Residual r_1 with $+15\,\%$ on sensor T_6 (left) and residual r_4 with $+415\,\%$ on sensor T_5 (right)

Faults on sensors P_7 and P_{11} affect residuals r_1, r_2 and r_3. Experimental results are given on Fig. 10.59 and 10.60.

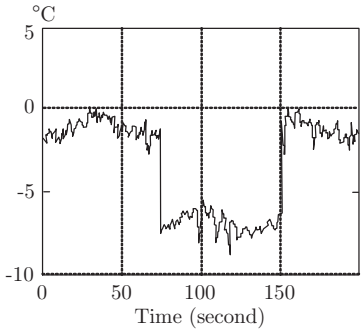

Fig. 10.59. Residual r_1 with $+15\,\%$ on P_7

It is seen that the sensitivity of r_3 to P_7 is real, and thus it reacts to the fault. Similarly, residuals r_2 et r_3 are really sensitive to faults of sensor F_{10} and residuals r_2 , r_6 are really senstive to faults of P_{12} and P_{14}.

Thus, all sensor faults can be detected since at least one residual is really senstivie to each of them.

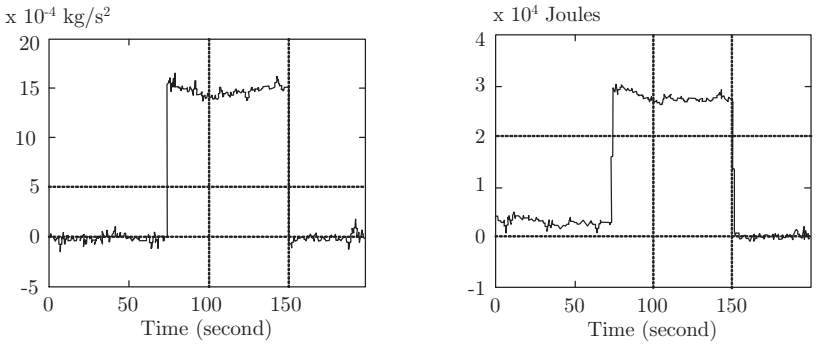

Fig. 10.60. Residual r_2 (left) and r_3 (right) with 15 % on P_7

Process faults. Two experiments are made with the faulty process: the clogging of the output valve (between time 60 s and 100 s), and a water leakage in the boiler (only for 3 seconds, from time 125 s to 128 s, since the experiment is very dangerous).

As expected, residual r_1 is not sensitive to any of these faults. Residual r_2 is sensitive only to the clogging (Fig. 10.61) because it is associated with the steam flow F_{VG} (measured by F_{10}) which is not affected by the fault in the considered operating conditions (the boiler still contains water enough).

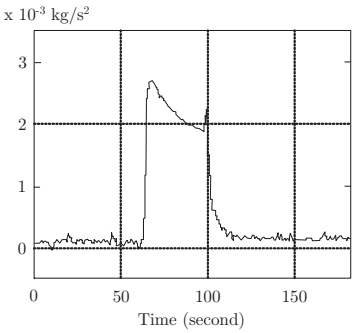

Fig. 10.61. Residual r_2 clogging of the output pipe

Residual r_3 (Fig. 10.62) is only senstive to the leak. The physical interpretation is that since the level regulator tries to compensate the leak, the boiler fills up with more cold water, hence creating a change in its energetic content.

Residuals r_4, r_5 and r_6 confirm their insensitivity to the clogging and to the leak.

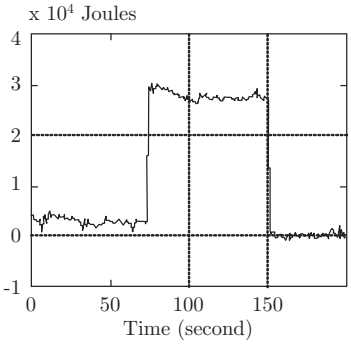

Fig. 10.62. Residual r_3 leak in the boiler

10.4.8 Evaluation of the experimental results

In spite of the simplifications, the model of the steam generator is quite good, as shown by the fact that in normal operation, the residuals are practically zero.

Structural analysis has been applied to design the fault diagnostic system, and has provided residuals which detect and isolate the faults given in the specifications. The consistency between the expected behaviour of the residuals and their actual behaviour has been experimentally shown.

The decision procedure which has been applied is rather simple: The residuals have been compared with a threshold equal to 3 times the standard deviation under the normal operation hypothesis. It has not been necessary to develop more sophisticated approaches, since the results obtained in the nominal operating regime (6-8 bars) are quite satisfactory: For all the experiments which have been performed, the false alarm and the missed-detection rates were zero.

10.5 Fault-tolerant electrical steering of warehouse trucks

This case study deals with electrical steering, a combined hardware-software-control problem that shows how the methods from the theory chapters are applied to this type of embedded system. Being critical to the safety of vehicles, the steering system of a vehicle is required to maintain the vehicles ability to steer until it is brought to halt, should a fault occur. With electrical steering becoming a cost-effective candidate for electrical powered vehicles, a fault-tolerant architecture is needed that meets this requirement. This case study treats the fault-tolerance properties of an electrical steering system. It presents a novel fault-tolerant architecture where a dedicated AC-motor design was used in conjunction with cheap voltage measurements to ensure detection of all relevant faults in the steering system. The study shows how active control reconfiguration can accommodate all critical faults. The fault-tolerant steering system was implemented on the hardware of a warehouse

truck and validations were made with hardware-in-the-loop tests, demonstrating diagnosis of all critical faults and ability to obtain the required fault-tolerant capabilities.

10.5.1 Introduction

A steering system for a vehicle on public road is required to maintain its ability to steer until it can be brought to halt, irrespective of any single fault in the steering system. With electrical steering becoming a cost-effective candidate for steering of electrical powered vehicles, system architecture and underlying interfaces, signal processing and control methods need be dedicated to meet this fundamental requirement.

Fail-operational systems that are able to continue operation with unchanged performance irrespective of any defect within the system itself are common in critical applications such as airplane control. Implemented with triple redundancy or more, these systems are, however, prohibitive for commercial markets. In order to achieve low-cost solutions, ideas from fault-tolerant control could be useful since we could accept degraded performance after a fault has occurred, if the vehicle is still able to be steered until it can be brought to a halt.

This case study suggests a fault-tolerant solution for an electrical steering system, discusses how hardware, software and system functionality should be addressed and presents a fault-tolerant architecture that enables a cheap solution that meets the requirement of authorities for driving on public roads. Analysing the architecture of a steering-by-wire system it is considered how duplicated actuator motors could be avoided. Using the AC motor star-point measurement, the study shows how diagnosis is obtained for all critical single-fault cases in the system. Finally, an extract of systematic tests of hardware and software faults are included as validation of the fault-tolerance abilities of electrical steering system. Hardware in the loop tests were made as validation using the hardware platform from a warehouse truck to document real performance.

10.5.2 Electrical Steering

The architecture of a basic electrical steering system for vehicles is shown in Fig. 10.63. The double-arrows indicate that connected sub-systems affect each other. The steering system contains a steering input system, a computer, a drive system, an induction motor, mechanical link to steering wheel(s), and a battery power supply. The user command comes from a steering-wheel or a joy-stick. The steering input system is a hardware component, which transforms the mechanical input to a reference steering signal for the computer. A control algorithm in the computer generates the actual control command to the drive system, and the drive system converts the voltage of the power supply into a three phase voltage signal for the induction motor. The motor is mechanically linked to the wheel(s) of the vehicle.

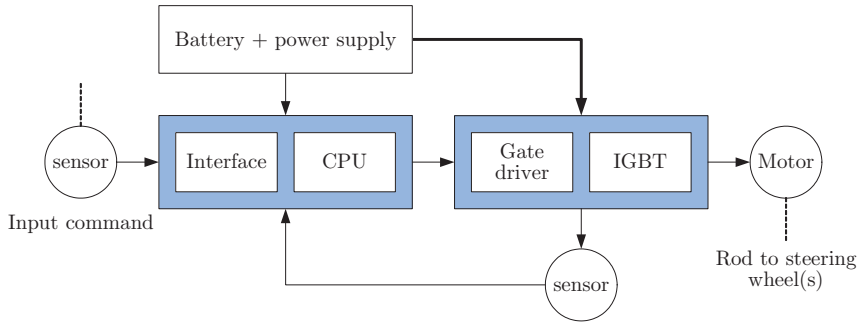

Fig. 10.63. A basic electrical steering system

For warehouse trucks and similar commercial applications, a steering system is needed that is low-cost and fault-tolerant. A basic electric steering system is shown in Fig. 10.63. It is potentially low-cost but is not fault-tolerant. To develop a low-cost and fault-tolerant steering system, all subsystems and their interactions need be carefully considered. With this purpose, it is useful to separate into three subsystems. One is the power supply, the second is the steering input system, the third is the wheel drive actuator, comprising computer, drive electronics, induction motor, wheel, and sensors. This case study considers the latter.

Actuator and sensor system. Fig. 10.64 shows wheel drive actuator system. The drive consists of an inverter and a gate-driver. PWM-signals are calculated by the computer and forwarded to the gate-driver that opens and closes the semiconductor switches (IGBT). The power electronics converts the battery supply voltage to a switched three phase voltage which is applied to the induction motor. Potentially, such system could contain a number of different sensors. However, for the warehouse truck and other low-cost applications, it is possible to use a control principle which uses only the average current i_{DC} for feedback. Figure 10.64 shows three sensors: the wheel demand V_w, current consumed by the power stage i_{DC} and the stator'a star point voltage of the motor, V_{st}.

Critical faults. It is possible to identify a large number of possible faults in the drive system and the induction motor, however, not all faults are relevant. For the warehouse truck, safety is the key issue, and faults that need to be considered are those that have implications on safety.

The steering system for a warehouse truck is as a relative low-torque/low-power application and in such applications, regulations by public authorities list mechanical components which are unlikely to fail if their design follow given standards. For example, breakage of the rotor shaft and breakage of the iron bars in the rotor belong to this category. In contrast, motor bearings become worn and will fail eventually.

All electronic components need to be considered prone to failure. This includes the DC link sensor (the i_{DC} signal), the computer with analog interface and the

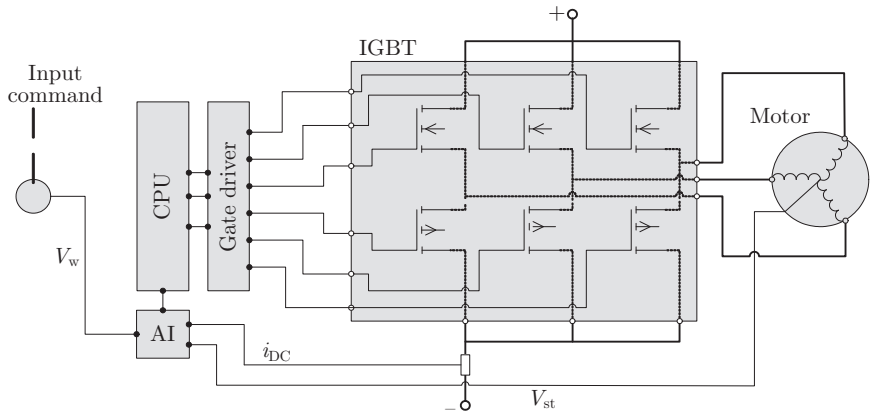

Fig. 10.64. Basic wheel drive system

inverter electronics. The inverter contains gate-driver and the power switches. Each power switch can either be short circuited or be open circuited (disconnected).

Electrical components include harness and induction motor. In the motor, stator windings are subject to be disconnected or short circuited, an internal turn winding short circuit could occur. In the harness or in the motor, a phase could be disconnected (open phase fault). A phase could be short-circuited to chassis but such failure is dealt with by intrinsic design using well established techniques with double-insulation.

Component failures that would cause loss of steering capability with the basic architecture of the electrical steering system are listed in Table 10.1.

Table 10.1. Critical component failures

Subsystem	Component	Failure	No.
Drive	Gate driver	Malfunction	f_1
	Power switch	Short circuit	f_2
	Power switch	Fail open	f_3
Motor	Winding	Open phase	f_4
	Winding	Turn short	f_5
	Harness	Open phase	f_6
	Harness	Phase to chassis	f_7
	Bearing	Stuck	f_8
Sensor	DC link	Malfunction	f_9
Power	Harness	Power loss	f_0

10.5.3 System architecture

The requirement of maintaining ability to steer the vehicle until it can be brought to a halt has some straightforward implications on the architecture of the electrical steering system,

- No single failure of a component may prevent adequate steering ability
- The system shall bypass a faulty component to continue operation or override the effects of a faulty component
- If an actuator fault is present, parallel action should have enough control authority to override the effects of the fault

While good case by case engineering designs could be achieved from such immediate observations, a systematic and rigorous analysis offer important benefits. One is to assure that any component discrepancy from normal is covered by an analysis of fault propagation and the consequences of component failure. Another is a provable correct deduction of fault propagation from basic assumptions is a valuable tool in the quality assurance and systematic validation of a design. The algebraic approach to fault propagation analysis, with the extension to generic component representations provide a useful method for a systematic design.

Component-based analysis. The analysis presented here is based on services offered by components in normal or faulty modes, and the impact the architecture has on the service available from the entire electrical steering system. A complete fault-propagation analysis was carried out in [254].

Subsystem behaviours. The system breakdown in Fig. 10.63 showed the system components and their interaction. With notation for signals shown in Table 10.3, behaviours in normal mode between input and output signals are listed in Table 10.2. The service $S^{(k)}$ offered by a component k is to deliver an output, according to the specified behaviour $S^{(k)}(c_k^{(v)})$ where $v \in \{n, d1, d2, .., o\}$ is a version of the service that follows from the condition of the component (normal, reduced1,..,none). If a component has an internal failure, a version of the service may be available with degraded performance $S^{(k)}(c_k^{(d)})$ or the service may not be available at all $S^{(k)}(c_k^{(d)})$.

System service. The steering service obtained for the system as an entirety is a function of the component architecture *A* and the version vector $\boldsymbol{v}$ for the present condition of components. With m components, the set of available behaviours will be $C_v = (c_1^{v(1)}, c_2^{v(2)}, ..., c_m^{v(m)})$, and the overall system service is $S^{(s)}(A, c_i^{v(i)}) = A(S^{(i)}), i = 1, ..., m$. With a single string architecture as Fig. 10.64, we obtain

Table 10.2. Component services and behaviours

Component	In	Out	Behaviour
Reference	u_{ref}	T_{dem}	$T_{\mathrm{dem}} = c_r(u_{\mathrm{ref}})$
Computer	T_{dem}	d_{com}	$d_{\mathrm{com}} = c_c(T_{\mathrm{dem}})$
AC drive	$\boldsymbol{u}_{\mathrm{com}}$	$\boldsymbol{u}_s$	$u_s = c_d(d_{\mathrm{com}})$
Motor	$\boldsymbol{u}_s$	T_m,$\mathbf{i}_s$	$T_m = c_{mT}(\boldsymbol{u}_s, \mathbf{i}_s, \omega_s)$
			$\mathbf{i}_s = c_{mi}(\boldsymbol{u}_s, T_m, \omega_s)$
i-sensor	i_m	i_{dc}	$i_m = i_{dc}$
Power	V_{bat}	V_{bus}	$V_{\mathrm{bus}} = const.$

Table 10.3. Notation

Variable	Explanation
u_{ref}	Driver's input command
Q_{dem}	Torque demand
$\boldsymbol{u}_{\mathrm{com}}$	Command to inverter
$\boldsymbol{u}_s$; $\boldsymbol{u}_r$	Stator (rotor) voltage
$\mathbf{i}_s$; $\mathbf{i}_r$	Stator (rotor) current
ψ_s; ψ_r	Stator (rotor) magnetic flux linkage
ϕ, θ	Angles stator and rotor fields
T_m	Motor torque
ω_m	Motor-shaft angular velocity
ϕ_m	Motor-shaft angle
i_{dc}	Average motor current
i_m	Measured average motor current
u_n	Starpoint voltage
V_{bus}	Battery voltage
V_{bus}	Bus voltage

$$S^{(s)}_{\mathrm{single}} = S^w \wedge S^p \wedge S^m \wedge S^d \wedge S^i \wedge S^c \wedge S^y, \tag{10.30}$$

where superscript w indicates wheel, p is power supply, m is motor, d is drive, i is current sensor, c is computer and u is voltage sensor.

An alternative could be a hardware configuration with two parallel totally redundant lines with only the wheel in common,

$$\begin{aligned} S^{(s)}_{\mathrm{rhw}} \quad = \quad & S^w \wedge \big(\big(S^{p1} \wedge S^{m1} \wedge S^{d1} \wedge S^{i1} \wedge S^{c1} \wedge S^{y1} \big) \\ & \vee \big(S^{p2} \wedge S^{m2} \wedge S^{d2} \wedge S^{i2} \wedge S^{c2} \wedge S^{y2} \big) \big) \end{aligned} \tag{10.31}$$

This solution is expensive as it requires two motors. Two motors would be allowed to drive a common shaft if it is proved that a healthy motor will be able to have control authority over a faulty one. A cost effective solution would be one single motor that could use dual windings on the stator and divide the power drive output stages between the windings. With such solution, both winding sets and inverter

stages would be used in normal operation while, in case of failure in one stator, the motor would be driven with up to half of maximal (nominal) torque. The fault-tolerant architecture shown in Fig. 10.65 is based on this idea. It was a prerequisite for this solution that the rotor bars of the AC motor are not prone to failure. Present (2006) standards allow a common rotor provided design rules are adhered to, just as is the case of the common rod from the motor to the steering wheel, which is assumed unlikely to fail provided standard design rules are followed.

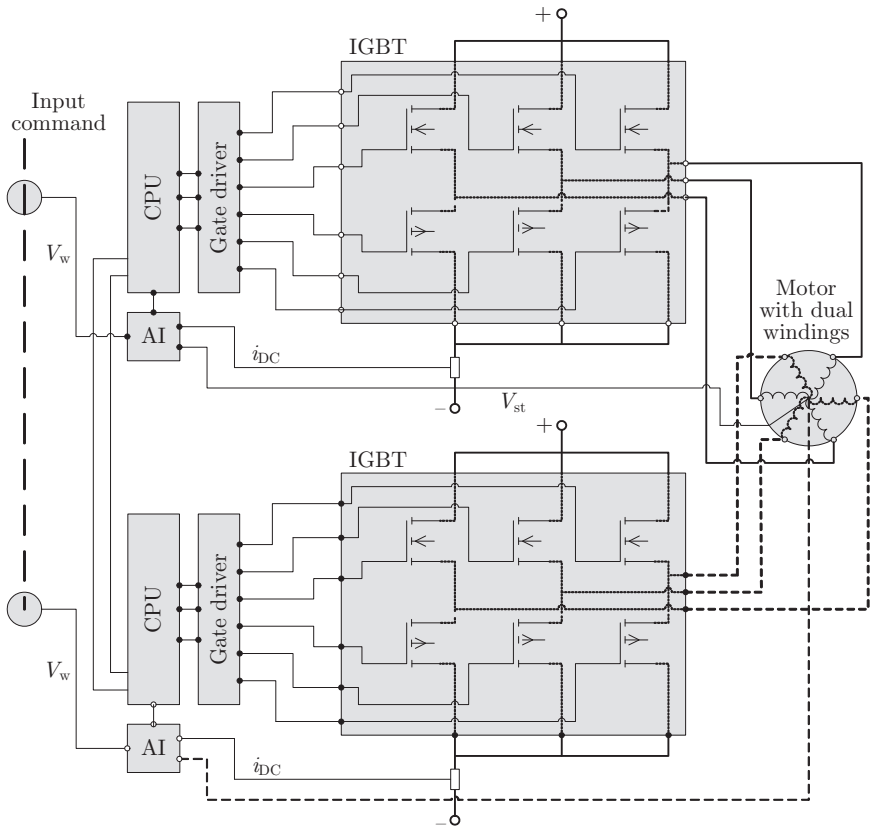

Fig. 10.65. An architecture for fault-tolerant electrical steering

The service at system level is

$$S_{\text{ftc}}^{(s)} = S^w \wedge S^{md} \wedge \left(S^{p1} \vee S^{p2}\right) \wedge \left(S^{i1} \vee S^{i2}\right) \wedge \left(\left(S^{c1} \wedge S^{d1} \wedge S^{y1}\right) \vee \left(S^{c2} \wedge S^{d2} \wedge S^{y2}\right)\right) \tag{10.32}$$

The paradigm in this architecture is that component failures should be detectable and faulty components be bypassed by controlling the signal flow in the software of the system. Using a motor with dual windings requires

- No critical failure in the motor must prevent the development of torque in the non-faulty part.
- A motor bearing fault should be detected before it can turn into a failure that makes the motor shaft unable to turn.

Dual stator AC motor. An AC motor with four windings was proposed earlier in the literature. A simpler and more cost effective solution is to remain with a three phase drive systems since mass produced power stages for inverters (IGBT devices) are available for the six transistor switch bridge sets used by 3-phase drives. Hence, it is worthwhile to investigate AC motor properties for motors with duplicated stator windings. A layout of a dual stator AC motor is shown in Fig. 10.66. A scrutiny of parameters in the dual winding motor showed that one physical motor with dual windings give fault-tolerance properties quite equivalent to two independent motors on a common shaft. The key issue is that the mutual inductance M_{ss} is not so large that a short circuit on one winding will prevent the motor from turning using the other winding for control.

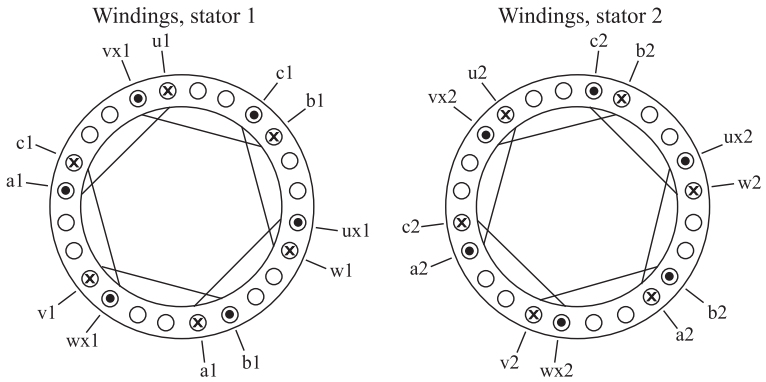

Fig. 10.66. AC motor with dual stator windings (by courtesy of J. S. Thomsen)

Requirements to fault detection and accommodation. The fault-tolerant architecture includes two actuators in parallel consisting of drive system and stator windings. In normal operation both actuators are active and co-operate to rotate the wheel according to control input. If a fault should occur in one of the two actuators it must be detected and accommodated to achieve fault tolerance. All critical faults which are not handled by design as described in Section 10.5.3 must be detected and accommodated. This includes faults f_1, f_2, f_3, f_4, f_5, and f_9, in Table 10.1.

It can be expected that faults in the drive system will propagate to have an effect in the stator windings. Several methods exists for detecting faults in stator windings.

Utilizing a star point sensor is advantageous for a low-cost solution as only two voltage measurements are needed.

Coverage. The paradigm in this architecture is that faults can be detected and fault handling be successfully achieved to bypass malfunctioning components. In theory, the probability of making a successful detection and system reconfiguration (coverage), is not 100%, but it will be shown in the experimental section that for all faults, which impair steering performance, faults can indeed be detected, isolated and the system be reconfigured.

In the following, we focus the discussion to power drive and actuator and limit the treatment to the critical faults, noting that diagnosis of motor bearing wear is not pursued in this context.

10.5.4 Structural analysis

The analysis of structure comprises the elements: formulating constraints, providing a matching on the unknown variables, determine measurements in which all critical faults are detectable and faulty parts are isolable such that correct reconfiguration can be made.

Constraints for the dual stator AC motor. The behaviours of the dual stator AC motor are available through a small extension to the theory for usual electrical motors, by taking account of the mutual inductions within the dual winding motor. Let the terminal voltage be $\boldsymbol{u}_t = (u_{t1}\ u_{t2}\ u_{t3})'$, the current in a stator winding $\mathbf{i}_s = (i_{s1}\ i_{s2}\ i_{s3})'$ and current in the rotor $\mathbf{i}_r = (i_{r1}\ i_{r2}\ i_{r3})'$. The flux in a stator is similarly the vector ψ_s and the flux through the rotor is ψ_r. Parameters are R for resistance, L for inductance, M for mutual inductance, θ for electrical angle between stator 1 and rotor, β the electrical offset angle between two stators and $N_{(3,3)}(\phi)$ a rotation matrix used to express the rotor-stator flux interaction,

$$\boldsymbol{N}(\phi) = \begin{pmatrix} \cos(\phi) & \cos(\phi+\frac{2\pi}{3}) & \cos(\phi+\frac{4\pi}{3}) \\ \cos(\phi+\frac{4\pi}{3}) & \cos(\phi) & \cos(\phi+\frac{2\pi}{3}) \\ \cos(\phi+\frac{2\pi}{3}) & \cos(\phi+\frac{4\pi}{3}) & \cos(\phi) \end{pmatrix}$$

For later use, note that the sum of elements in a column of $\boldsymbol{N}(\phi)$ is zero,

$$\sum_{k=1}^{3} \boldsymbol{N}_{kj}(\phi) = cos(\phi) + cos(\phi+\tfrac{2\pi}{3}) + cos(\phi+\tfrac{4\pi}{3}) = 0. \quad (10.33)$$

The basic electrical equations for the dual winding AC motor are

$$
\begin{array}{lll}
c_1: & \boldsymbol{u}_t^{(1)} & = u_n^{(1)} + R_s \mathbf{i}_s^{(1)} + \frac{d}{dt}\left(\psi_s^{(1)}\right); \\
c_2: & \boldsymbol{u}_t^{(2)} & = u_n^{(2)} + R_s \mathbf{i}_s^{(2)} + \frac{d}{dt}\left(\psi_s^{(2)}\right); \\
c_3: & 0 & = R_r \mathbf{i}_r + \frac{d}{dt}\left(\psi_r\right) \\
c_4: & \psi_s^{(1)} & = L_s \mathbf{i}_s^{(1)} + M_{sr}\boldsymbol{N}(\theta)\mathbf{i}_r + M_{ss}\boldsymbol{N}(-\beta)\mathbf{i}_s^{(2)} \\
c_5: & \psi_s^{(2)} & = L_s \mathbf{i}_s^{(2)} + M_{sr}\boldsymbol{N}(\theta+\beta)\mathbf{i}_r + M_{ss}\boldsymbol{N}(\beta)\mathbf{i}_s^{(1)} \\
c_6: & \psi_r & = L_r \mathbf{i}_r + M_{sr}\boldsymbol{N}'(\theta)\mathbf{i}_s^{(1)} + M_{sr}\boldsymbol{N}'(\theta-\beta)\mathbf{i}_s^{(2)} \\
c_7: & 0 & = \sum_{k=1}^{3} i_{sk}^{(1)} \\
c_8: & 0 & = \sum_{k=1}^{3} i_{sk}^{(2)} \\
c_9: & 0 & = \sum_{k=1}^{3} i_{rk}
\end{array}
\tag{10.34}
$$

where $\boldsymbol{u}_t^{(1)}$ and $\boldsymbol{u}_t^{(2)}$ are the terminal voltage vectors applied on the two stator windings.

The mechanical variables are the angle θ and the angular velocity ω, motor torque T_m and load torque T_l. Total inertia referred to the motor shaft is I_t. The torque balance of the motor then gives

$$
\begin{array}{lll}
c_{10}: & T_m & = \left(\mathbf{i}_s^{(1)}\right)' \frac{\partial}{\partial\theta}\left(M_{sr}(\theta)\right)\mathbf{i}_r \\
 & & \quad + \left(\mathbf{i}_s^{(2)}\right)' \frac{\partial}{\partial\theta}\left(M_{sr}(\theta-\beta)\right)\mathbf{i}_r \\
c_{11}: & \frac{d}{dt}\left(I_t\omega\right) & = T_m - T_l \\
d_1: & \omega & = \frac{d}{dt}\left(\theta\right)
\end{array}
\tag{10.35}
$$

Without loss of generality, several differential constraints have been written implicitly in Equations (10.34) and (10.35) to limit the size of the incidence matrix.

The sets of unknown variables X_m and known variables K_m in Equations (10.34) and (10.35) are

$$
\begin{array}{lcl}
X_m & = & \{u_n^{(1)}, u_n^{(2)}, \mathbf{i}_s^{(1)}, \mathbf{i}_s^{(2)}, \mathbf{i}_r, \psi_s^{(1)}, \psi_s^{(2)}, \psi_r, \theta, \omega, T_m, T_l\} \\
K_m & = & \{\boldsymbol{u}_t^{(1)}, \boldsymbol{u}_t^{(2)}\}.
\end{array}
\tag{10.36}
$$

Viewing Eq. (10.36) as scalar variables, there are 24 unknown and 6 known scalar variables. There are 24 scalar constraints comprised in Equations (10.34) and (10.35). Hence, the system is under-constrained or just constrained on X_m.

The structure graph S_m is shown in the incidence matrix where each of the vector constraints have been split into their scalar parts a, b and c respectively, and the columns refer to the scalar variables in the set X_m.

	$u_n^{(1)}$	$u_n^{(2)}$	$i_{s1}^{(1)}$	$i_{s2}^{(1)}$	$i_{s3}^{(1)}$	$i_{s1}^{(2)}$	$i_{s2}^{(2)}$	$i_{s3}^{(2)}$	i_{r1}	i_{r2}	i_{r3}	$\psi_{s1}^{(1)}$	$\psi_{s2}^{(1)}$	$\psi_{s3}^{(1)}$	$\psi_{s1}^{(2)}$	$\psi_{s2}^{(2)}$	$\psi_{s3}^{(2)}$	ψ_{r1}	ψ_{r2}	ψ_{r3}	θ	ω	T_m	T_l	$u_{ta}^{(1)}$	$u_{tb}^{(1)}$	$u_{tc}^{(1)}$	$u_{ta}^{(2)}$	$u_{tb}^{(2)}$	$u_{tb}^{(3)}$
c_{1a}	1	0	①	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c_{1b}	1	0	0	①	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c_{1c}	①	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c_{2a}	0	1	0	0	0	①	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c_{2b}	0	1	0	0	0	0	①	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c_{2c}	0	①	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
c_{3a}	0	0	0	0	0	0	0	0	①	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
c_{3b}	0	0	0	0	0	0	0	0	0	①	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
c_{3c}	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	①	0	0	0	0	0	0	0	0	0	0
c_{4a}	0	0	1	0	0	1	1	1	1	1	1	①	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
c_{4b}	0	0	0	1	0	1	1	1	1	1	1	0	①	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
c_{4c}	0	0	0	0	1	1	1	1	1	1	1	0	0	①	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
c_{5a}	0	0	1	1	1	1	0	0	1	1	1	0	0	0	①	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
c_{5b}	0	0	1	1	1	0	1	0	1	1	1	0	0	0	0	①	0	0	0	0	1	0	0	0	0	0	0	0	0	0
c_{5c}	0	0	1	1	1	0	0	1	1	1	1	0	0	0	0	0	①	0	0	0	1	0	0	0	0	0	0	0	0	0
c_{6a}	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	①	0	0	1	0	0	0	0	0	0	0	0	0
c_{6b}	0	0	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	①	0	1	0	0	0	0	0	0	0	0	0
c_{6c}	0	0	1	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	1	①	0	0	0	0	0	0	0	0	0
c_7	0	0	1	1	①	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c_8	0	0	0	0	0	1	1	①	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c_9	0	0	0	0	0	0	0	0	1	1	①	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c_{10}	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	①	0	0	0	0	0	0	0
c_{11}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	①	0	0	0	0	0	0
d_4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	①	0	0	0	0	0	0	0	0

A complete matching on X_m is marked in the incidence matrix by ①. The matching is also complete on C_m so driven by the terminal voltages on the two windings, and with unknown load torque, this system is just constrained. The matching could not be found using the simple ranking algorithm due to the closed loops in this graph. The more general algorithms had to be employed.

The existence of a complete matching knowing the terminal voltages on the motor, but not the mechanical load torque, shows that the motor will enter an equilibrium state where motor torque outbalance the load torque. Currents and fluxes within the motor are determined by the solution to the set of nonlinear equations in Equations (10.34) and (10.35). The existence of a symbolic or numeric solution can not be determined from the structure graph alone since the loops in the structure graph comprise nonlinear elements. A scrutiny in electrical machines shows, however, that a solution does exist, which is indeed expected from similarity of this motor with single stator AC motors.

Assured that the solution does exist, it timely to consider which sensors should be made available to meet the fault-tolerance requirements.

Measurements on the motor. Several signals could be monitored in the AC motor and the connected components. From a cost perspective, high-frequency (i.e. 20-200 kHz) pulse width modulated (PWM) voltage signals are difficult and expensive to monitor. In contrast, voltages without high-frequency contents are easily and inexpensively converted to digital signals. Measurement of high currents is certainly possible but require expensive transducers when the currents are above the 5-10 A range. Since $\boldsymbol{u}_t$ is high frequency chopped (PWM) and the 6 components of the vector valued currents $\mathbf{i}_s^{(1)}$, $\mathbf{i}_s^{(2)}$ belong to the high current category, the preferred monitoring possibility for the AC motor are the two star point voltages $v_n^{(1)}$ and $v_n^{(2)}$. Choosing the star-point voltages as measurements means to add two constraints,

$$\begin{array}{lll} m_1: & u_{nm}^{(1)} & = u_n^{(1)} \\ m_2: & u_{nm}^{(2)} & = u_n^{(2)} \end{array} \tag{10.37}$$

Star point voltage. With $u_n^{(1)}$ and $u_n^{(1)}$ related to voltage and current in the two stators, constraints c_1 to c_2 in Eq. (10.34), consider summation of the a, b and c components for each stator. For brevity, we use the notation

$$\sum_{k=1}^{3} u_{tk}^{(1)} := \sum_{3} u_t^{(1)}.$$

Then

$$\sum_{3} u_t^{(1)} \quad = \quad 3u_n^{(1)} + L_s \sum_{3} i_s^{(1)} + \frac{d}{dt} \sum_{3} \psi_s^{(1)} \tag{10.38}$$

and

$$\begin{aligned}\sum_3 \psi_{tk}^{(1)} &= L_s \sum_3 i_s^{(1)} + M_{sr} \sum_3 (\boldsymbol{N}(\theta)\mathbf{i}_r) \\ &+ M_{ss} \sum_3 (\boldsymbol{N}(\theta-\beta)\mathbf{i}_s^{(2)})\end{aligned}$$

Since

$$\begin{aligned}\sum_3 \boldsymbol{N}(\theta)\mathbf{i}_r &= \sum_{k=1}^{3} \boldsymbol{N}_{k1}(\theta) i_{r1} + \sum_{k=1}^{3} \boldsymbol{N}_{k2}(\theta) i_{r2} + \sum_{k=1}^{3} \boldsymbol{N}_{k3}(\theta) i_{r3} \\ &= \mathbf{0} \cdot \mathbf{i}_r\end{aligned}$$

and

$$\sum_3 i_s^{(1)} = 0,$$

Eq. (10.38) is reduced to

$$0 = 3u_n^{(1)} - \sum_3 u_t^{(1)}, \tag{10.39}$$

and a similar result is obtained for stator 2.

This shows that with a symmetric voltage vector applied at the terminals , i.e. $\sum_3 u_t = 0$, the star point voltage is zero when the system acts according to its normal behaviour.

Structural detectability in star-point residuals. Adding the two measured star-point voltages as known variables and the associated constraints from Eq. (10.37) to the structure graph, these two constraints remain unmatched and are hence parity relations, that are used for residual generation. Backtracking through the matching disclose how violation of constraints are detectable in the two residuals,

$$\begin{aligned}r_1(t) &= 3u_n^{(1)}(t) - \sum_3 u_t^{(1)}(t) \\ r_2(t) &= 3u_n^{(2)}(t) - \sum_3 u_t^{(2)}(t),\end{aligned} \tag{10.40}$$

The result is that from the star-point measurements, violation of any constraint in Equations (10.34) and (10.35) are detectable in both of the residuals of Eq. (10.40).

Structural isolability is not achieved for any violation of the primary constraints because both residuals depend structurally on the entire set of the basic constraints. This does not necessary mean that (not structural) isolation is impossible. In order to scrutinize the properties of the residuals in Eq. (10.40).

Extension to other components. So far, only the dual stator AC motor was treated. It remains to formulate the constraints for the remaining components of the electrical steering system.

Each of the power stages (power drives) receive a voltage command $\boldsymbol{u}_{cmd}$ from the microprocessor and deliver the PWM signal $\boldsymbol{u}_t$ to the motor. The power consumption of the drive is P_d and power delivered to a stator is P_s with an efficiency η_d. Then, $(j = 1, 2)$

$$\begin{aligned} pd_{1j}: \quad & \boldsymbol{u}_t^{(j)} && = \boldsymbol{u}_{cmd}^{(j)} \\ pd_{2j}: \quad & P_d^{(j)} && = (\mathbf{i}_{dc}^{(j)}) V_{bus} \\ pd_{3j}: \quad & \eta_d P_d^j && = P_s^j \end{aligned} \tag{10.41}$$

where i_{dc} is the current drawn by the power stage from the DC voltage supply.

For each of the micro processors $(j = 1, 2)$,

$$\begin{aligned} mp_{1j}: \quad & \boldsymbol{u}_{cmd}^j && = c_c(u_{rm}^{(j)}) \\ mp_{2j}: \quad & u_{nm}^{(j1)} && = u_n^{(1)} \\ mp_{3j}: \quad & u_{nm}^{(j2)} && = u_n^{(2)} \\ mp_{4j}: \quad & i_m^{(j1)} && = i_{dc}^{(1)} \\ mp_{5j}: \quad & i_m^{(j2)} && = i_{dc}^{(2)} \\ mp_{6j}: \quad & u_{rm}^{(j)} && = u_{ref} \end{aligned} \tag{10.42}$$

where constraint mp_2 shows that the physical measurement of the star point voltage is done by the microprocessor unit. Similarly, it is the microprocessor that measures command u_{ref} and current consumption i_m for each of the power stages. The processor outputs the command voltage $\boldsymbol{u}_{cmd}$. The measurements of are conducted such that microprocessor associated with the stator (1) line has information about the measurements from the stator (2) line. The cross-measurements are not necessarily analog interface but could be implemented using data-bus communication between the microprocessors, however with a penalty in isolability of faults in interface or sensors.

Having introduced power consumption in the constraints, we revert to the motor equations and express the power balance of the motor using already available variables,

$$\begin{aligned} c_{12}: \quad & P_s^{(1)} && = (\mathbf{i}_s^{(1)})' \boldsymbol{u}_s^{(1)} \\ c_{13}: \quad & P_s^{(2)} && = (\mathbf{i}_s^{(2)})' \boldsymbol{u}_s^{(2)} \\ c_{14}: \quad & P_m && = \eta_m \left(P_s^{(1)} + P_s^{(1)} - P_0 \right) \\ c_{15}: \quad & P_m && = T_m \omega \end{aligned} \tag{10.43}$$

where η_m is a known motor efficiency and P_0 the magnetization loss, which is also known from motor data.

Matching the structure graph resulting from constraints in Equations (10.41), (10.42) and (10.43) leads to further parity relations that makes it possible to isolate faults in either of the computer units, and in each of the combination of drive and stator blocks. The structural analysis that gives these results is straight forward.

10.5.5 Analytical properties of residuals

Continuing with the next level of detail in the design, we now analyze the properties of the analytical parity relations found in the structural analysis. Isolability of faults to one or the other of the stator feed lines (computer - power stage - stator) is the key issue and weak and strong detectability are of faults are essential in this respect.

Star-point residuals. Reverting to $r_1(t)$ from Eq. (10.40),

$$\begin{aligned} r_1(t) &= 3u_n^{(1)}(t) - \sum_3 \boldsymbol{u}_t^{(1)}(t), \\ &= R_s \sum_3 \mathbf{i}_s^{(1)} + \sum_3 \frac{d}{dt}\left(\psi_s^{(1)}\right) \\ &= \sum_3 R_s \mathbf{i}_s^{(1)} + \omega \sum_3 M_{sr} \frac{\partial \boldsymbol{N}(\theta)}{\partial \theta} \mathbf{i}_r + \sum_3 L_s \frac{d}{dt}(\mathbf{i}_s^{(1)}) \\ &\quad + \sum_3 M_{sr} \boldsymbol{N}(\theta) \frac{d}{dt}(\mathbf{i}_r) + \sum_3 M_{ss} \boldsymbol{N}(-\beta) \frac{d}{dt}(\mathbf{i}_s^{(2)}) \end{aligned} \tag{10.44}$$

and a symmetrical result is obtained for $r_2(t)$.

Eq. (10.44) shows that there is strong detectability in r_1 of faults in stator 1 and in the rotor. Faults in faults in stator 2 will be weakly detectable if the resulting derivatives of currents are symmetrical, i.e. the sum of the components remain zero. The implication is that stator 2 faults will be weakly detectable or very small in r_1. With the symmetry of r_1 and r_2, stator 2 and rotor faults will be strongly detectable in r_2 and stator 1 faults will be weakly detectable.

It it noted that also faults in the stator 1 power electronics will be strongly detectable as imbalance in $\boldsymbol{u}_t$ will give rise to imbalance in $\mathbf{i}_s$.

This is a quite fortunate result as it is possible to obtain a unambiguous diagnostic result and the remedial reaction to diagnosed faults is clear,

$$\begin{aligned} H_1(r_1) \wedge H_0(r_2) &\Rightarrow \text{disable(1)} \\ H_0(r_1) \wedge H_1(r_2) &\Rightarrow \text{disable(2)} \\ H_1(r_1) \wedge H_1(r_2) &\Rightarrow \text{reduce power to motor} \end{aligned}$$

It is noted that, according to regulations, by regulatory design of the rotor and the shaft of an AC motor, rotor defects are considered impossible. If, nonetheless, a rotor defect should be developing, the total power is reduced to prevent rotor failure.

Change detection from star point residual. With an unbalance, the star point residual will differ from zero in amplitude. The star point has the fundamental frequency of $\boldsymbol{u}_t$ as a dominant component. Detection of changes in $r(t)$ therefore need be done at the fundamental frequency. Correlation at the fundamental frequency is

done through correlation over a window of N samples of the residual, and with sampling time T_s,

$$r_j(n) \quad = \quad \sum_{k=n-N}^{n} u_{nm}(nT_s)^{(j)} e^{-j\omega_s T_s k}. \tag{10.45}$$

Other residuals. With two current measurements representing electrical power to a drive, additional residuals exist but the load torque on the wheel is unknown and only one additional redundancy relation can be obtained as a reliable measure of discrepancy within the system. This residual is r_3 as measured by CPU 1 and r_4 as measured by CPU2,

$$\begin{aligned} r_3(t) &= i_{m1}^{(1)}(t) - i_{m1}^{(2)}(t) \\ r_4(t) &= i_{m2}^{(1)}(t) - i_{m2}^{(2)}(t) \end{aligned} \tag{10.46}$$

Isolation in case of a DC link sensor fault is possible from the "passive" residuals Eq. (10.46), but certain common mode faults would affect both residuals, which would prevent isolation. As the system is part of an fault-tolerant control system, active isolation can easily be performed. A perturbation signal is added to the command for each drive and the signature in the current signal $i_m^{(1)}$ and $i_m^{(2)}$ will determine which sensor could be defect.

10.5.6 Fault detection and isolation

The critical faults that need be detected and accommodated in the part of the system on which we focus, are f_1 - f_5 and f_9 of Table 10.1. The effects of each fault was investigated and the signature found in the residuals.

$$\begin{aligned} f_1 \quad &\rightarrow \quad \{u_n \text{ balance} \vee u_n \text{ unbalance }\} \\ f_2 \quad &\rightarrow \quad \{u_n \text{ unbalance}\} \\ f_3 \quad &\rightarrow \quad \{u_n \text{ unbalance}\} \\ f_4 \quad &\rightarrow \quad \{u_n \text{ unbalance}\} \\ f_5 \quad &\rightarrow \quad \{u_n \text{ unbalance}\} \\ f_9 \quad &\rightarrow \quad \{\text{DC link incorrect value}\} \end{aligned} \tag{10.47}$$

f_1 is a gate driver malfunction. If a critical fault occurs in the gate-driver it can easily be assumed that the output signals has no resemblance with valid PWM signals, except when all signals are logical zero. Some combinations of signals will enable a circuit from positive to negative supply, either in the inverter or through the stator. In both cases one or more power switches would be destroyed. This would result in stator unbalance similar to an open phase fault. Lack of unbalance only occur in the situation where all output signals are logical zero. This does not leave a signature

in the star point signal but the fault is easily identified in the DC link signal, which will be zero despite a control input is present. f_2 is a power switch short circuit. As destruction of one or more switches can be expected, the voltages applied to the stator will not be balanced. Likewise unbalanced signals are obtained if f_3 (power switch fail open) or f_4 (windings open phase) happen. f_5 is an internal turn fault in the stator. Such fault results in a reduced winding and, thus, an unbalanced stator is obtained. f_9 is a DC link sensor fault. This causes no unbalance in the stator windings but with two DC link sensors, fault isolation is possible.

Change detection and isolation. In conclusion, mean value or combined mean value and variance change detection in the two residuals $r_j(n)$, and in $\epsilon(n)$ was able to detect each of the critical faults. All critical faults were strongly detectable and application of standard CUSUM methods was straightforward. Isolation was also possible in all cases with appropriate means.

10.5.7 Experiments

The fault-tolerant architecture in Section 10.5.3 has been implemented as a laboratory test system using actual hardware. With the test system it is possible to generate selected non-destructive faults; a phase wire can be physically disconnected in system 1, each transistor in the inverter of system 2 can be disabled, and a part of a phase winding in a stator can be short circuited. Using the test system it is possible to experimentally generate faults f_1, f_2, f_3, f_4 and f_5, and detection and isolation was validated.

A number of tests were performed. Three test results (tests 2, 5, and 7) are shown below. Fig. 10.67 shows the case where a phase is physically disconnected in system 1, in Fig. 10.68 one of the inverter transistors is disabled (open) in system 2, and in Fig. 10.69 a part of a phase winding in system 1 is short circuited. The tests show the strong detectability of faults and a time to detect in the 0.1-0.3 s range, which is acceptable.

10.5.8 Evaluation of the results

This industrial case study showed how the systematic approach from component and system structure could be employed to find the properties of the overall fault-tolerant electrical steering system. It was shows how hardware, software and system functionality aspects could be combined to obtain system-wide fault-tolerance. Duplicated motors were avoided and replaced by one double stator induction motor to obtain a low-cost yet fault-tolerant solution. Using the AC motor star-point measurement and a simple measurement of total current to each drive section, the case showed how correct diagnosis was obtained for all single fault cases, both in the motor and in associated power electronics. The case considered how defects in power

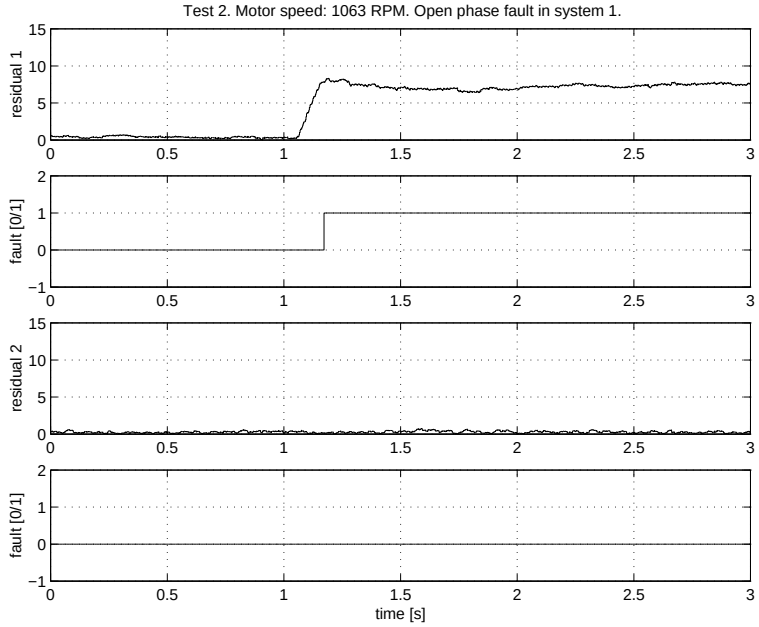

Fig. 10.67. A phase is physically disconnected in system 1 (by courtesy of J. S. Thomsen)

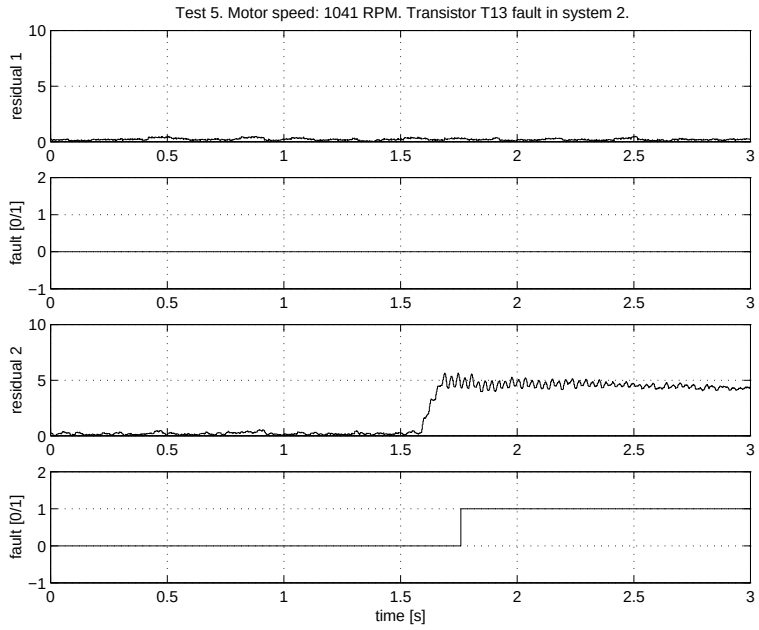

Fig. 10.68. An inverter transistor is disabled (open) in system 2 (by courtesy of J. S. Thomsen)

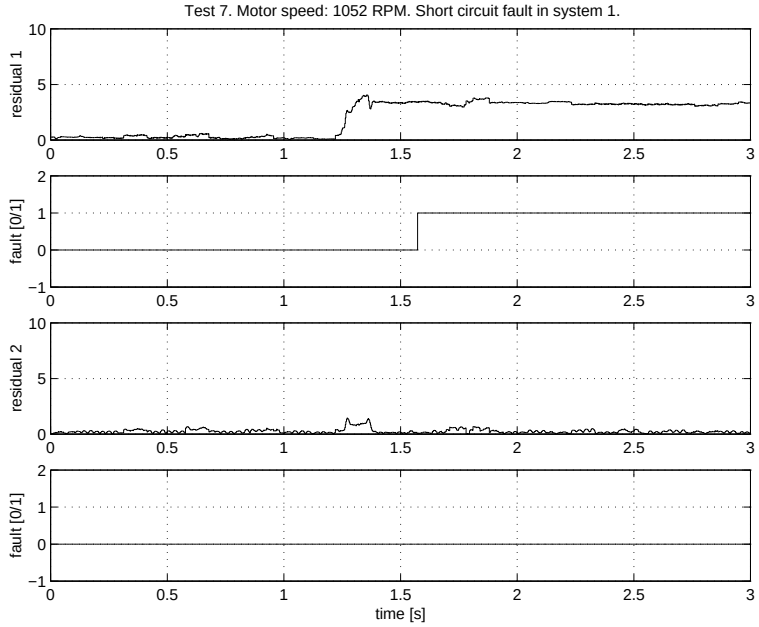

Fig. 10.69. A part of a phase winding in system 1 is short circuited (by courtesy of J. S. Thomsen)

electronics could be detected in time to allow faults to be isolated and how reconfiguration could be obtained. Finally, systematic tests of faults imposed on a warehouse truck platform demonstrated the fault-tolerant abilities of the new steering system.

10.6 Summary: Guidelines for the design of fault-tolerant control

As a summary for further applications, this section treats the architecture for implementation of autonomous supervision and describes the design process needed to achieve a fault-tolerant control algorithm in a documented and reliable way.

10.6.1 Architecture

Autonomous supervision requires development and implementation observing completeness and correctness qualities. It is important that the design of a supervised control system follows a modular approach, where each functionality can be designed, implemented, and tested independently of the remaining system. The algorithms that realise the supervisory functionality constitute themselves an increased

risk for failures in software, so the overall reliability can only be improved if the supervisory level is absolutely trustworthy.

The design of an autonomous supervisor relies heavily on having an appropriate architecture that supports clear allocation of methods to different software tasks. This is crucial for both development and verification. The latter is vital since test of the supervisor functions in an autonomous control system is a daunting task.

Supervisor. The implementation of a supervisory level onto a control system is not a trivial task. The architecture shall accommodate the implementation of diverse functions

- Support of overall coordinated plant control in different phases of the controlled process; start-up, normal operation, batch processing, event triggered operation with different control objectives, close-down.
- Support of all use-modes for normal operation and modes of operation in fault-tolerant control versions of services, for the foreseeable faults.
- Autonomous monitoring of operational status, control errors, process status and conditions.
- Fault diagnosis, accommodation and re-configuration as needed. This is done autonomously, with status information to plant-wide coordinated control.

These functions are adequately implemented in a supervisory structure with three levels in the autonomous controller, and communication to a plant-wide control as the fourth. The autonomous supervision is composed of levels 2 and 3, taking care of fault diagnosis, logic for state control and effectors for activation or calculation of appropriate remedial actions. This is illustrated in Fig. 10.70 that shows:

1. A lower level with input/output and the control loop.
2. A second level with algorithms for fault diagnosis and effectors to fault accommodation.
3. A third level with supervisor logic.
4. A fourth layer with plant-wide control and coordination.

The control level is designed and tested in each individual mode that is specified by different operational phases and different instrumentation configurations. The miscellaneous controller modes are considered separately and it is left to the supervisor design to guarantee selection of the correct mode in different situations.

The detectors are signal processing units that observe the system and compares with the expected system behaviour. An alarm is raised when an anomaly is detected. The effectors execute the remedial actions associated with fault accommodation or reconfiguration.

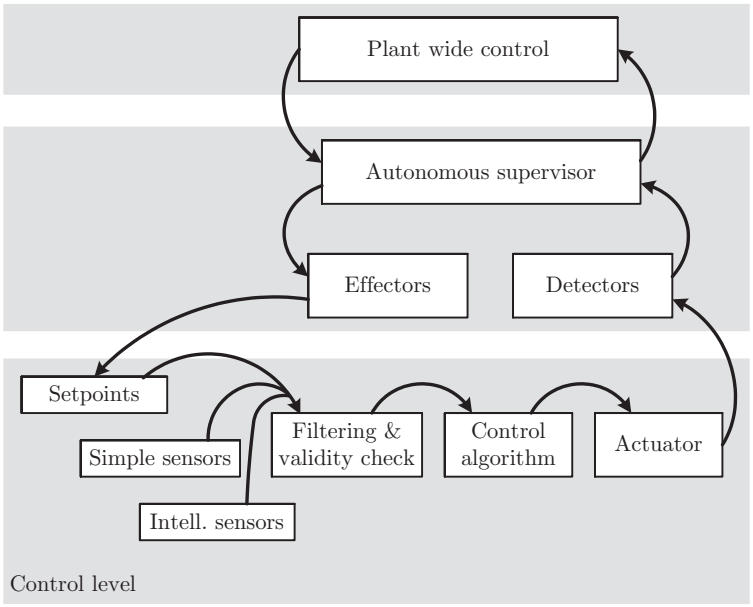

Fig. 10.70. Autonomous supervisor comprises fault diagnosis, supervisor logic and effectors, the latter to carry out the necessary remedial actions when faults are diagnosed. The upper level is plant-wide control and operator supervision.

10.6.2 Design procedure

When the level of autonomy becomes high and thereby demands a high level of reliable operation, it becomes inherently complex for the designer to cover all possible situations and guarantee correct and complete operation.

A systematic design strategy will use the analysis of fault propagation and structure as basic elements to obtain completeness of the analysis and correct operation of the system when implemented.

Component based analysis. Describe components and their interconnections using the generic model and fault propagation analysis introduced in Chapter 4. The result is a list of component related faults/failures that needs to be handled to avoid high severity end effects or critical events.

- **System breakdown:** Make, as the initial step, a top-down breakdown of the system into suitable subsystems. Make a further breakdown of subsystems into components, of types aggregated or simple.

- **Component models:** Describe the services of each component, the use-modes and services associated with each use-mode. List input-output variables associated with each version of a service. Provide fault description, effects on compo-

nent output from possible faults and failure and construct the propagation matrices associated with each version of a service.

- **Fault propagation:** Make a Fault Propagation Analysis of all relevant subsystems and combine into a complete analysis of the controlled system. The end-effects describe consequences at top level. Re-use any available fault propagation matrices for components and accumulating knowledge about component failures.

- **Severity assessment:** Judge top level end-effects for severity. The ones with significant influence on control performance, safety or availability are collected in a list for treatment by the autonomous supervisor.

- **Reverse deduction:** Make a reverse deduction of fault propagation to locate faults that would cause any of the severe end-effects, or combinations thereof, from the list.

- **Result:** The result is a short-list of faults that should be diagnosed and handled.

Structural analysis. Analyse system structure using the methods given in Chapter 5. The result gives a type of information whether sufficient redundancy is available in the system to detect and isolate each of the selected faults, and to handle the faults by a reconfiguration strategy.

- **Constraints:** Deduce an enumerated list of constraints from the set of models of the individual components.There may be different sets of constraints associated with different services.

- **Structure graph:** Use the set of constraints to formulate the system structure graph as explained in Chapter 5.

- **Matching:** Make a complete matching to find a set of unmatched constraints.

- **Constraints for residual generation:** Use unmatched constraints to provide parity equations for residual generation.

- **Ability to diagnose severe faults:** For each fault with severe end effects listed in the fault propagation analysis, verify that the system structure allows the particular fault to be diagnosed.

- **Ability to perform fault handling:** For each of the faults from the list, verify that sufficient redundancy is available in the faulty system to allow handling of the fault. For sensor faults investigate structural observability. For actuator faults, investigate structural controllability. For other faults validate the ability to control the faulty system.

Remedial actions. The possible remedial actions are designed in this step. For each of the short-listed faults, the designed must choose to utilise physical redundancy or analytical redundancy according to the results of the structural analysis. Whether the original control objectives can be met will not be known at this stage of design. The following issues need to be dealt with.

- **Fault-tolerant control version of service with full performance:** If redundant hardware is available, use this to operate with full performance.

- **Fault-tolerant control version of service with degraded performance:** Change to a control scheme that does not require the faulty component or can compensate the fault by estimating its magnitude. A fault-tolerant solution with some performance degradation is mostly an acceptable alternative to part of a plant becoming unavailable. A performance index should be available to guide controller re-design.

- **Predetermined controller re-design:** Predetermined reactions to faults can be obtained in many cases and should be preferred when possible, due to less complexity than the alternative. Predetermined solutions include estimator or controller re-design done at the design stage.

- **Autonomous controller re-design:** When remedial actions depend on the state of the system, online autonomous re-design can be necessary. Autonomous re-design is considered the most challenging of the fault-tolerant control possibilities, with a complexity equivalent to solutions in adaptive control.

- **Fail to safe state:** When appropriate fault handling cannot be achieved, the supervisor should make the system fail to a safe state. When autonomous re-design is relevant, the supervisor should comprise an independent diagnostic module for performance monitoring and must retain the ability to fail to a safe state, should proper performance of the re-design fail to be confirmed.

Fault diagnosis design. The structure information again provides a list of possibilities. The reconfigurability measure for the faulty system indicates how difficult reconstruction will be.

- **Residual generator:** Based on unmatched constraints, formulate the parity equations that can provide the basis for a residual generator.

- **Detailed design for detection:** Make a detailed design for diagnosis following the results in Section 6.2 using the parity equations approach or Section 6.4 using an optimisation-based approach.

- **Detailed design for isolation:** The selected remedial actions determine the requirements for fault isolation. It is not necessary to isolate faults below the level

where the fault propagation can be stopped.

- **Detailed design for estimation:** If the remedial action requires fault estimation, design fault estimation if possible.

Control of the faulty system. Control of the faulty system is by nature of the problem, as difficult as a design of an original control system. However, if the fault is of one of the simpler types in a sensor, a remedial action is sometimes straightforward. This is also the case if the fault can be treated as an incremental change to the system, or the fault is purely additive, then results in Chapter 7 can be applied and show the family of controllers that stabilise the system. In the general case of re-design, the entire range of design methods in feedback control could be employed, however, following the discussion in Chapter 7, a methodology is recommended that is based on the formulation of a clearly defined performance specification.

- **Sensor faults:** Attempt to estimate the faulty measurement, design and employ an observer; design an output feedback without using the faulty sensor; if the type of fault is additive (bias or drift) consider a compensation through fault estimation.

- **Actuator faults:** Consider the controllability without using the faulty actuator. If possible, make a controller re-design for the faulty system using remaining actuators. If the actuator faults is physically additive, fault estimation may make it possible to make a simple compensator.

- **Plant faults:** Consider controllability (stabilisability) of the faulty system. Determine possible performance without re-tuning the controller. Investigate whether a simple re-tuning can be achieved of the faulty system using Youla-Kucera parameterisation.

- **Reconfiguration:** If other options fail, re-design the controller completely, to obtain required performance.

- **Time-to-reconfigure:** If reconfiguration is needed and complete isolation cannot be achieved within the required time to reconfigure, the set (Σ, τ) will need to be selected assuming a worst-case condition in the set of diagnostic results. The worst case fault is one that has the highest degree of severity.

- **Change objective:** When other possibilities are exhausted, relax the performance objectives for the faulty system and design an appropriate controller.

- **Fail to safe:** If the original control objectives cannot be met, handling of the problem by a the supervision function must be considered. The autonomous part

of supervision must always be to offer graceful degradation and close down when this is necessary as fall-back.

Supervisor logic. Supervisor inference rules are designed using the information about which faults/effects are detected and how they are treated. The autonomous supervisor determines the most appropriate action from the present condition and commands. The autonomous supervisor must be designed to treat mode changes of the controlled process and any overall/operator commands. Worst-case conditions and overall safety objectives should have priority when full isolation or controller-re-design cannot be accomplished within the required time to get within control specifications after a fault.

Test. Tests should be complete. The main obstacle is the complexity of the resulting hybrid system consisting of controller and plant. Transient conditions like switching between normal and not-normal controllers - and reverse - should in particular be carefully tested.

The above steps are intended to make the supervisor design cheaper, faster, and better. The fault coverage is then (hopefully) as complete as is possible, because the fault propagation analysis step in principle includes all possible faults. The analysis is modular, because small subsystems are treated individually. Furthermore, the strategy has the advantage that the system is analysed on a logical level as far as possible before the laborious job of mathematical modelling and design is initiated. This should ensure that superfluous analysis and design are avoided.

10.7 Bibliographical notes

The reconfiguration problem for the three-tank system described in Section 10.1 has been tackled as a benchmark problem in the COSY project [91]. Several solutions are described in [3], [151].

The chemical process described in Section 10.2 has been used to test and evaluate different diagnostic methods and fault-tolerant control principles. The results outlined here have been published in [156], [155] and [217]. The virtual sensor and virtual actuator example for reconfiguration is published in [162], [240]. A fault-tolerant control principle, which is based on an on-line optimisation, is described in [212]. The recent experimental results with the conductivity control problem are reported in [215].

The diagnosis and fault-tolerant control problem of the ship propulsion system described in Section 10.3 was presented as an international benchmark [105], [106] and was used as a platform for the comparison of different methods and the development of new ideas. The modelling of the ship propulsion system was described in [14], [18]. The quantised systems approach to the diagnosis of the ship system is presented in detail in [92]. The stability of an observer used in Section 10.3 has been proved in [71].

[19] presented an adaptive observer solution to estimate states and faults and used the same nonlinear observer for reconfiguration. [53] extended used a dedicated sliding mode observer for fault detection on the nonlinear propulsion plant. Two Ph.D. theses used the benchmark as a main example. Supervisor logic design was a main theme in [107]; the nonlinear problem

was the focus for [133]. [28] used a high gain observer for the nonlinear shaft speed dynamics, similar to that of [19], however not adaptive. They applied a static detector to diagnose pitch and engine gain faults. An estimate of the magnitude of sensor faults was used by the control scheme to accommodate faults by setpoint alteration. Reflecting on the nonlinear dynamics, uncertain parameters and complex designs of several earlier methods, [103] suggested a neuro-fuzzy output observer for diagnosis. [270] suggested a Kalman filter solution where non-switching fault accommodation was obtained using sensor fault estimation.

Describing the behaviour of steam generators considered in Section 10.4 results in highly nonlinear models due to the coupling of many physical phenomena of different natures. Taking into account the fact that steam generators are among the most widely spread processes, many works have been devoted to the subject, e.g. [3], [29], [189], [251].

The coefficients of the thermal model described in Section 10.4 are computed by empirical algorithms given in [251].

One of the first applications of fault-tolerant control in an industrial scale was described in [256] for mass-produced inverters for induction motors.

The process of arriving at development methods for fault-tolerant control is an evolution where some areas of application have been explored, but a final form cannot be said to be reached. Steps in the evolution include [22] presenting general ideas, [26] and [25] where experience from applying fault-tolerant methods for the the Ørsted satellite were incorporated, [107] who treat the supervisor-logic level, [257] and [256] aiming at implementation in a large volume industrial product and therefore also includes cost-benefit assessments.

Concerning implementation, a correct and consistent control system analysis should always be followed by equally correct software implementation. This is particularly relevant for the supervisory parts of an fault-tolerant control scheme [107]. Testing of the fault-tolerant control elements is difficult since it is difficult to replicate the real conditions under which faults occur. Well planned software architecture and implementation are thus crucial issues for fault-tolerant control implementation. A study of the use of object-oriented programming architectures was described by [137]. The fault-tolerant control area is hence very wide and involves several areas of system theory. One overview [193] emphasised many algorithmic essentials and the role of fault diagnosis. Another [22] presented an engineering view of the means to obtain fault-tolerant control. Formal definitions were introduced in [21]. Analysis of structure was covered in [76], [107] and [232]. Measures of recoverability were discussed in [67], [76] and [268]. Quantitative techniques to assess the reliability of fault-tolerant control implementations was the subject of [23], [266], [267].

Modeling for detection of faults in individual components have been widely studied: [38] filtered the star-point voltage of an AC motor around the fundamental frequency and used a level test to detect AC motor faults; a model for simulation of turn faults was published in [248]; detection of particular faults occurring in closed-loop AC motor actuators were studied in [249]; [111] used diagnostic methods for centrifugal pumps; [208] analysed ways to detect partial failure in power switch circuits. Performance of components was considered by [239] who analysed a multi-phase induction machine with faults. Analysis of an entire system was treated in [55] who needed a fully hardware redundant solution, with duplicated permanent motors, to cope with component faults. An AC motor with four windings was proposed for fault-tolerant systems in [239]. The case study of electrical steering originated in the methods to obtain system-wide fault-tolerance [22] and specific research results obtained in [254], in the patent [20] and in [255].