



## Fault Detection for Shipboard Monitoring and Decision Support Systems

**Lajic, Zoran; Nielsen, Ulrik Dam**

*Published in:*

28th International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2009)

*Publication date:*

2009

*Document Version*

Early version, also known as pre-print

[Link back to DTU Orbit](#)

*Citation (APA):*

Lajic, Z., & Nielsen, U. D. (2009). Fault Detection for Shipboard Monitoring and Decision Support Systems. In 28th International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2009) (28th ed., pp. OMAE2009-79367). Honolulu: American Society of Mechanical Engineers.

---

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

OMAE2009-79367

## FAULT DETECTION FOR SHIPBOARD MONITORING AND DECISION SUPPORT SYSTEMS

Zoran Lajic

Technical University of Denmark,  
Department of Mechanical Engineering,  
Section of Coastal, Maritime and Structural Eng.  
Build. 403, 2800-Kgs. Lyngby, Denmark  
{zl@mek.dtu.dk}

Ulrik Dam Nielsen

Technical University of Denmark,  
Department of Mechanical Engineering,  
Section of Coastal, Maritime and Structural Eng.  
Build. 403, 2800-Kgs. Lyngby, Denmark  
{udn@mek.dtu.dk}

### ABSTRACT

In this paper a basic idea of a fault-tolerant monitoring and decision support system will be explained. Fault detection is an important part of the fault-tolerant design for in-service monitoring and decision support systems for ships. In the paper, a virtual example of fault detection will be presented for a containership with a real decision support system onboard. All possible faults can be simulated and detected using residuals and the generalized likelihood ratio (GLR) algorithm.

### INTRODUCTION

The SeaSense system (Nielsen et al., 2006) has been installed on several containerships and navy vessels. The system provides an estimation of the actual sea state, information about the longitudinal hull-girder loading, sea-keeping performance of the ship, and decision support on how to operate the ship within acceptable limits. It is able to identify critical forthcoming events and to give advice regarding speed and course changes to decrease the wave-induced loads. The system, sketched in Figure 1, includes a number of sensors all of which are used to estimate hull girder responses and predict wave loads, with the purpose of avoiding critical levels of hull stresses and ship motion(s). In this paper, it is suggested to improve the system by converting the system into a fault-tolerant monitoring and decision support system. All the changes are on the software side, but the hardware remains the same. The wave elevation could be obtained using SeaSense system by combining the relative wave height and the vertical acceleration. The wave elevation has been – artificially – included in the sensor fault detection procedure as a virtual (non-existent) sensor. The sensor fault detection is performed using several measurements: vertical acceleration, heave, pitch,

wave elevation and relative wave height (distance between the deck and the water surface).

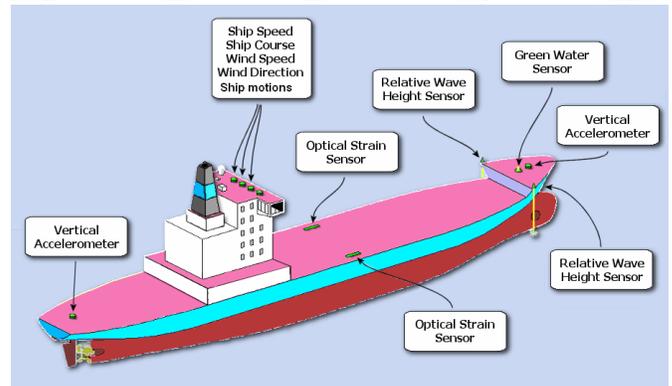


Fig. 1: Onboard sensor arrangement

### FAULT-TOLERANT MONITORING AND DECISION SUPPORT SYSTEM

A fault is something that changes the system behaviour in a manner so that the system does no longer satisfy its purpose. The fault is the primary cause of system performance degradation or even loss of the system function. Therefore, it is very important to find the faults as quickly as possible and to make decisions that stop the propagation of their effects. The aim of these measures is to make the system fault-tolerant, such that the system functions after the appearance of a fault. A fault-tolerant monitoring and decision support system has the ability to react on the existence of the fault by adjusting its activities to the faulty behaviour of the system. The procedure to make a monitoring and decision

support system fault-tolerant consists of two steps, similar as in case of the fault-tolerant control system (Blanke et al., 2006):

**Fault diagnosis:** The existence of faults has to be detected and the faults have to be identified.

**Re-design:** A monitoring and decision support system has to be adapted to the faulty situation so that the overall system continues to satisfy its goal.

Let us consider the system with the input  $u(t)$  and the output  $y(t)$ . The pair  $(u, y)$  is called input/output pair and the set of all possible pairs that may occur for a given system define the behaviour  $\mathbf{B}$ . The behaviour  $\mathbf{B}$  can be defined as a subset of the space  $u \times y$  of all possible combinations of input and output signals. In Figure 2 is given a graphical interpretation of the system behaviour. The point A represents a specific input/output pair that may occur for the given system and the point B represents a pair which is inconsistent with the system dynamics. The point B represents an input/output pair in the presence of fault. Actually, in the faulty case the system behaviour will move from the set  $\mathbf{B}_0$  towards the set  $\mathbf{B}_f$ . In the case of common input  $u$ , both systems, faultless ( $\mathbf{B}_0$ ) and faulty ( $\mathbf{B}_f$ ) will give different outputs  $y_A$  and  $y_B$ , respectively. The change in the system behaviour makes the detection and isolation of the fault possible. However, the measurement information  $(u, y)$  alone is not sufficient, but a model, which describes the nominal system behaviour, is necessary. The model is a representation of the system behaviour.

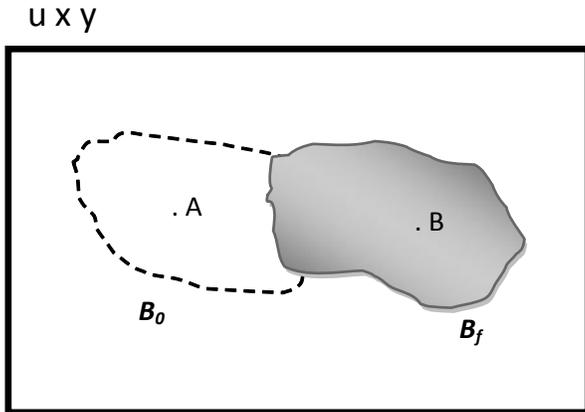


Fig. 2: System behaviour.

The models describe the behaviour of the faultless and the faulty system. They restrict the possible input/output pairs to those that appear in the behaviour  $\mathbf{B}_0$  or  $\mathbf{B}_f$ . With other words, models are constraints on the signals  $u$  and  $y$  that appear in the system. The model of the system in the presence of a specific fault shows how the system output  $y$  is affected by this fault. In fault diagnosis both the input  $u$  and the output  $y$  are known and it is checked whether the input/output pair belongs to the specific behaviour. In summary, the diagnostic principle can be described as follows (Blanke et al., 2006):

For given models that describe the behaviour  $\mathbf{B}_f$  of the system subjected to the faults  $f \in \mathcal{F}$ , test whether the input/output pair  $(U, Y)$  satisfies the relation

$$(U, Y) \in \mathbf{B}_f.$$

**Fault detection:** If the input/output pair is inconsistent with the behaviour  $\mathbf{B}_0$  of the faultless system

$$(U, Y) \notin \mathbf{B}_0$$

Then a fault is known to have occurred.

**Fault isolation and identification:** If the input/output pair is consistent with the behaviour  $\mathbf{B}_f$ ,

$$(U, Y) \in \mathbf{B}_f.$$

Then a fault  $f$  may have occurred.

We can notice that fault detection is possible without any information about the behaviour of the faulty system. Fault detection uses only a model of the nominal system. It is sufficient to identify deviations of the current system behaviour from the nominal behaviour. On the other hand, fault isolation and identification are not possible without information about the faulty system. Therefore, the fault model has to be known. The fault diagnosis can be divided in three steps on the basis of accuracy as mentioned before:

**Fault detection:** Decide whether or not a fault has occurred. This step determines the time at which the system is subjected to some fault.

**Fault isolation:** Find in which component a fault has occurred. This step determines the location of the fault.

**Fault identification and fault estimation:** Identify the fault and estimate its magnitude. This step determines kind of fault and its severity.

In the fault diagnosis the sensor measurement  $y$  is compared with the analytically computed value  $\hat{y}$ . Therefore, the consistency of the system with the model can be tested at every time  $t$  by:

$$r(t) = y(t) - \hat{y}(t)$$

Where  $r(t)$  is called a residual. In the faultless case, the residual is close to zero. It is not exactly zero, due to measurement noise and model uncertainties. In the presence of fault, the residual has some specific, non-vanishing value.

Diagnostic algorithms for continuous-variable systems generally consist of two components; cf. Figure 3 (Blanke et al., 2006):

**Residual generation:** The model and the input/output pair are used to determine residuals, which describe the degree of consistency between the system and the model behaviour.

**Residual evaluation:** The residual is evaluated in order to detect, isolate and identify faults.

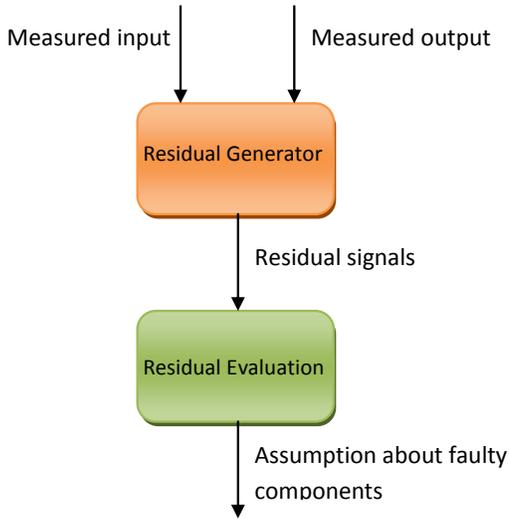


Fig. 3: Fault detection and isolation

The re-design considers the problem of changing the system structure after a fault has occurred in the plant. The aim is to provide the capability of the system to satisfy its purpose after occurrence of the fault. Two principal ways of re-design have to be distinguished, fault accommodation and reconfiguration:

**Fault accommodation** (Off-line solution): Fault accommodation means to adapt the parameters to the dynamical properties of the faulty system. The input and output of the plant remain the same as for faultless case. The fault accommodation is based on predesigned systems, each of which has been selected off-line for a specific fault. This method is fast and can meet strong real-time constraints. However, re-design has to be made for all possible faults before the system is operative.

**Reconfiguration** (On-line solution): If fault accommodation is impossible, the complete monitoring and decision support system has to be reconfigured. In this case we have a new system configuration where alternative input and output signals are used. Those signals are selected according to the faults. A new system has to be designed on-line. Reconfiguration is necessary in case of sensor failure.

### EXAMPLE: A CONTAINERSHIP WITH DECISION SUPPORT SYSTEM

For sensor fault detection, there is as mentioned a need to find physical relations between measured values. The SeaSense system has at its disposal several measurements: vertical acceleration, heave, pitch, wave elevation and relative wave height. The basic ship motions are shown in Figure 4.

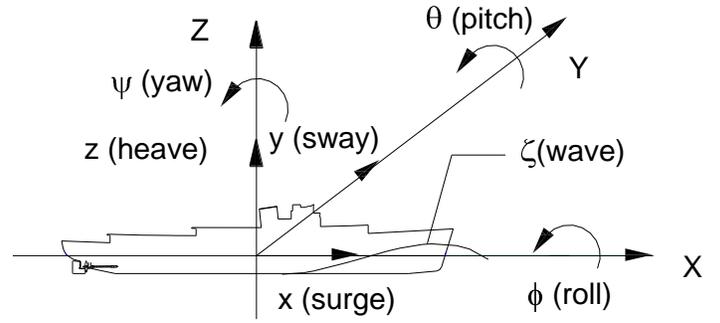


Fig. 4: Basic ship motions

For the actual sea state estimation (Nielsen et al., 2006), it is sufficient to use three different ship motions. Therefore, the fault-tolerant approach is convenient for implementation in our case.

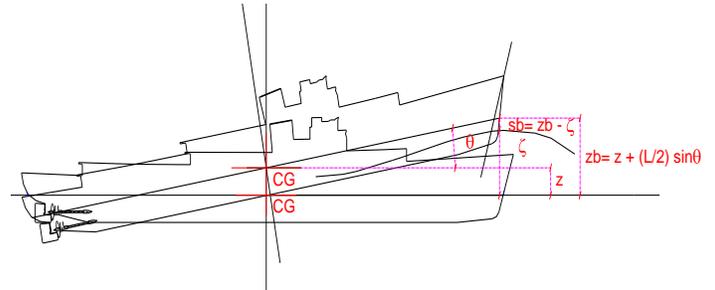


Fig. 5: Heave and pitch motion

The relation between different ship motions are given by the following set of constraints (see Figure 5):

$$\begin{aligned}
 c_1: z_b &= z + \frac{L}{2} \sin \theta \\
 c_2: \dot{z}_b &= \dot{z} + \frac{L}{2} \frac{d}{dt} (\sin \theta) = \dot{z} + \frac{L}{2} \cos \theta \cdot \dot{\theta} \\
 c_3: \ddot{z}_b &= \ddot{z} + \frac{L}{2} \frac{d^2}{dt^2} (\sin \theta) = \ddot{z} + \frac{L}{2} \cos \theta \cdot \ddot{\theta} - \frac{L}{2} \sin \theta \cdot \dot{\theta}^2 \\
 c_4: s_b &= z_b - \zeta \\
 d_1: \dot{z} &= \frac{dz}{dt} \\
 d_2: \ddot{z} &= \frac{d\dot{z}}{dt} \\
 d_3: \dot{z}_b &= \frac{dz_b}{dt} \\
 d_4: \ddot{z}_b &= \frac{d\dot{z}_b}{dt} \\
 d_5: \dot{\theta} &= \frac{d\theta}{dt} \\
 d_6: \ddot{\theta} &= \frac{d\dot{\theta}}{dt} \\
 m_1: y_1 &= \dot{z}_b \\
 m_2: y_2 &= z \\
 m_3: y_3 &= \theta
 \end{aligned}$$

$$m_4: y_4 = s_b$$

$$m_5: y_5 = \zeta$$

Where:

$\ddot{z}_b$	Vertical acceleration on the bow
$z$	Heave
$\theta$	Pitch
$s_b$	Relative bow motion (distance between the deck on the bow and the wave)
$\zeta$	Wave elevation
$L$	Length of the ship (it is assumed that centre of gravity is on the midship section).

Symbols  $c_1\dots c_4$  denote constraints,  $d_1\dots d_6$  derivatives and  $m_1\dots m_5$  measurements.

For more information about constraints, see (Bhattacharyya, 1978). The set of variables in a system are separated into the sets  $X$  (unknown) and  $K$  (known).

Known variables:

$$K = \{y_1, y_2, y_3, y_4, y_5\}$$

Unknown variables:

$$X = \{z_b, \dot{z}_b, \ddot{z}_b, z, \dot{z}, \ddot{z}, \theta, \dot{\theta}, \ddot{\theta}, s_b, \zeta\}$$

In order to extract the residuals, available for fault diagnosis, one may apply an analysis of the structure of the constraints and the unknown variables. While there are several ways to deduct analytical redundancy relations for nonlinear systems, analysis is particularly simple. Algorithms exist for matching unknown variables to constraints and find the non-matched constraints that are the basis for minimal structurally overdetermined sets that are used to determine the possible diagnosis algorithms (Krysander et al., 2008). The constraints may be matched to the unknown variables as follows:

$$m_1(y_1) \rightarrow \ddot{z}_b$$

$$m_2(y_2) \rightarrow z$$

$$d_1(z) = d_1(m_2(y_2)) \rightarrow \dot{z}$$

$$d_2(\dot{z}) = d_2(d_1(m_2(y_2))) \rightarrow \ddot{z}$$

$$m_3(y_3) \rightarrow \theta$$

$$d_5(\theta) = d_5(m_3(y_3)) \rightarrow \dot{\theta}$$

$$d_6(\dot{\theta}) = d_6(d_5(m_3(y_3))) \rightarrow \ddot{\theta}$$

$$m_4(y_4) \rightarrow s_b$$

$$m_5(y_5) \rightarrow \zeta$$

$$c_1(z, \theta) = c_1(m_2(y_2), m_3(y_3)) \rightarrow z_b$$

$$d_3(z_b) = d_3(c_1(m_2(y_2), m_3(y_3))) \rightarrow \dot{z}_b$$

It is given one complete matching on the unknown variables, three constraints remains unmatched. Each of these gives basis for a residual generator that can check the consistency of the constraints. If deviation from normal behavior should occur, this will result in one or more of the residuals being different from zero. The residuals (in schematic form) are:

$r_1: c_3(\ddot{z}_b, \ddot{z}, \ddot{\theta}, \dot{\theta}, \theta) \rightarrow 0$ , that is, after substitution of the unknown variables

$$r_1: c_3\left(m_1(y_1), d_2\left(d_1\left(m_2(y_2)\right)\right), d_6\left(d_5\left(m_3(y_3)\right)\right), d_5\left(m_3(y_3)\right), m_3(y_3)\right) \rightarrow 0$$

$r_2: c_4(s_b, z_b, \zeta) \rightarrow 0$ , that is, after eliminating unknown variables

$$r_2: c_4(m_4(y_4), c_1(m_2(y_2), m_3(y_3)), m_5(y_5)) \rightarrow 0$$

$r_3: c_2(\dot{z}_b, \dot{z}, \theta, \dot{\theta}) \rightarrow 0$ , that is

$$r_3: c_2\left(d_3\left(c_1\left(m_2(y_2), m_3(y_3)\right)\right), d_1\left(m_2(y_2)\right), m_3(y_3), d_5\left(m_3(y_3)\right)\right) \rightarrow 0$$

It is easier to match the unknown variables using a matching table (Blanke et al., 2006). The matching table for the specific system is shown in Table 1.

$i$	$c_1$	$c_2$	$c_3$	$c_4$	$d_1$	$d_2$	$d_3$	$d_4$	$d_5$	$d_6$	$m_1$	$m_2$	$m_3$	$m_4$	$m_5$
$y_1$	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
$y_2$	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
$y_3$	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
$y_4$	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
$y_5$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
$z_b$	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0
$\dot{z}_b$	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0
$\ddot{z}_b$	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0
$z$	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0
$\dot{z}$	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0
$\ddot{z}$	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
$\theta$	1	1	1	0	0	0	0	0	1	0	0	0	1	0	0
$\dot{\theta}$	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0
$\ddot{\theta}$	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
$s_b$	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
$\zeta$	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
zero		-0	-0	-0											

Table 1: Matching table

In the paper, residuals  $r_1$  and  $r_2$  will be discussed only, because from the third residual it is not possible to obtain new information about the system.

The system contains five sensors. Therefore, it is possible to have the following faults:

- $f_1$ - vertical acceleration,
- $f_2$ - heave,
- $f_3$ - pitch,
- $f_4$ - wave elevation and
- $f_5$ - relative wave height.

The fault signatures are shown in Table 2. The symbol **i** denotes a fault isolation and the symbol **d** denotes a fault detection.

/	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$
$r_1$	x	x	x		
$r_2$		x	x	x	x
	1	3	3	2	2
	<b>i</b>	<b>d</b>	<b>d</b>	<b>d</b>	<b>d</b>

Table 2: Fault signatures

All five column vectors are different from zero, and thus all faults are detectable. The fault  $f_1$  has a unique signature and therefore this fault is isolable. As a result of the analysis, the following set has been obtained:

**Detectable faults:**  $f_2, f_3, f_4, f_5$

**Undetectable faults:** -

**Isolable faults:**  $f_1$

The residuals can be expressed in the analytic form as:

$$r_1: y_1 - \ddot{y}_2 - \frac{L}{2} \frac{d^2}{dt^2} (\sin y_3) = 0$$

$$r_2: y_4 + y_5 - y_2 - \frac{L}{2} \sin y_3 = 0$$

In the following, simulations of ship motions in waves and the residuals  $r_1$  and  $r_2$  are presented. The simulations of ship motions in waves are similar to (Nielsen, 2007). The referred ship is a containership (length  $L= 275$  m, breath  $B= 40$  m, draught  $T= 12$  m), which sails at speed of 10 m/s. Residuals have been simulated using Simulink® (Dabney et al., 2004). The simulations are performed during a time period of 1000 seconds. The simulations of pitch, heave, vertical acceleration and wave elevation are shown in figures 6-9.

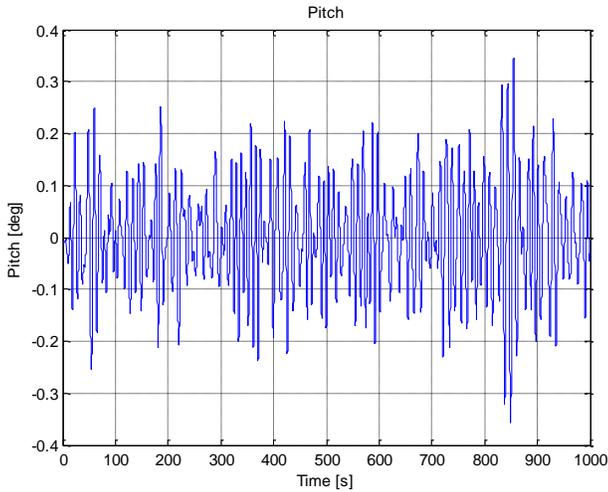


Fig. 6: Pitch

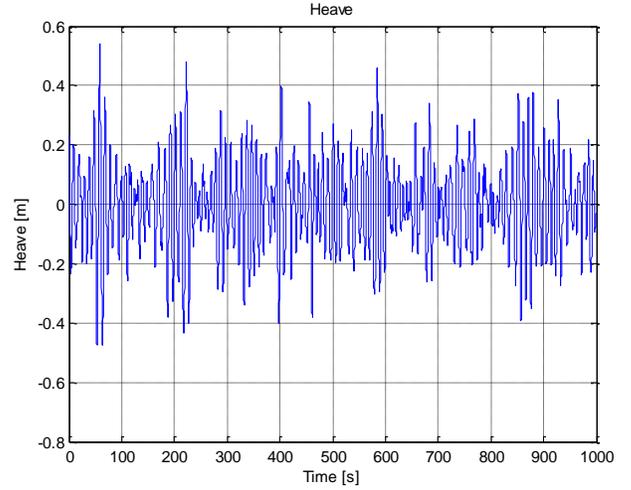


Fig. 7: Heave

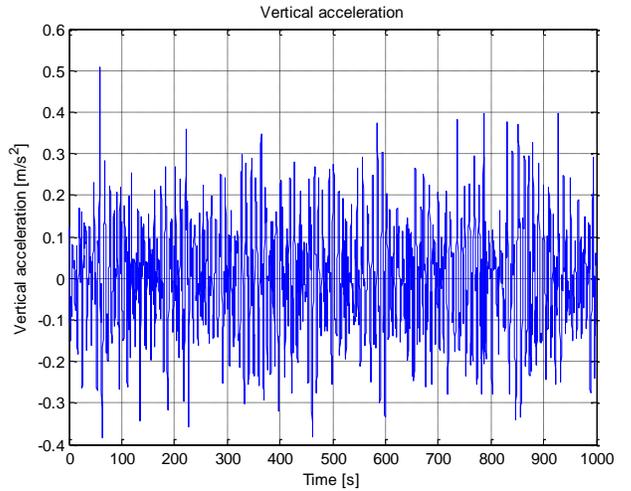


Fig. 8: Vertical acceleration

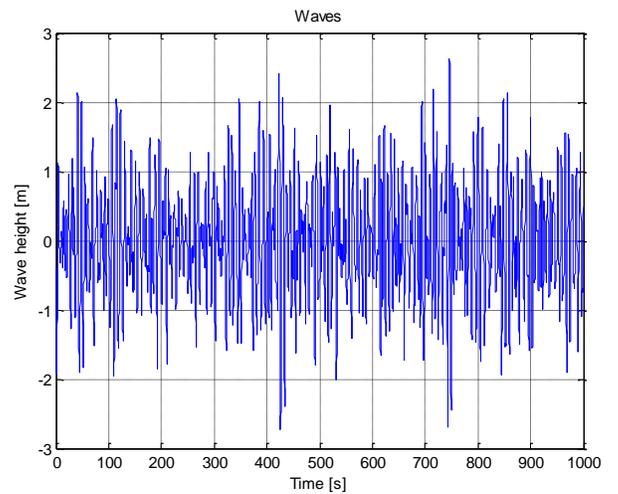


Fig. 9: Wave elevation (midship section)

Residuals  $r_1$  and  $r_2$  in case of no faults are shown in Figures 10 and 11. Both residuals indicate the absence of the fault.

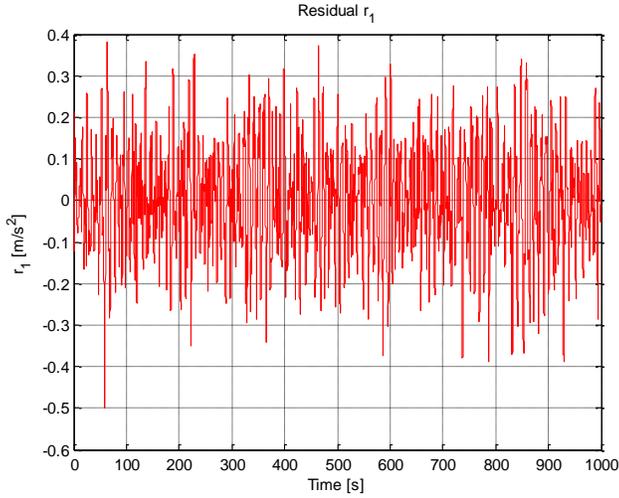


Fig. 10: Residual  $r_1$  (no faults).

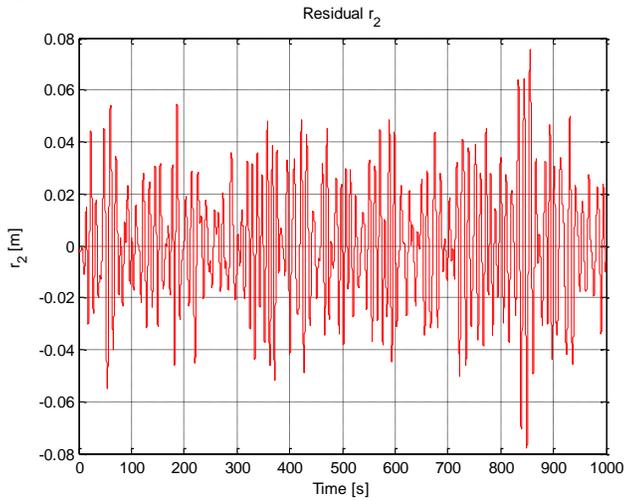


Fig. 11: Residual  $r_2$  (no faults).

In the following, the residuals in the presence of faults will be presented and discussed. As examples of fault scenarios, the faults will be created by adding an extra signal to the sensor output during time interval between 100 seconds and 500 seconds. After 500 seconds, extra signals are removed. The configuration is summarized in Table 3.

Fault	Type	Units	Value	Time interval [s]
$f_1$	vertical acceleration	$m/s^2$	0.5	100-500
$f_2$	heave	m	0.5	100-500
$f_3$	pitch	rad	0.05	100-500
$f_4$	wave elevation	m	0.5	100-500
$f_5$	relative wave height	m	0.5	100-500

Table 3: Simulated faults.

Residuals can be directly evaluated by the GLR (generalized likelihood ratio) test. The GLR test is a useful tool in fault detection. The GLR algorithm relies on the log-likelihood ratio of an observation  $z$ . Important steps in GLR algorithm are selection of threshold  $h$ , decision function  $g$  calculation and the construction of the alarm diagram using selected threshold and calculated decision function. The GLR decision function can be calculated (Blanke et al., 2006):

$$g(k) = \frac{1}{2\sigma^2} \max_{k-M+1 \leq j \leq k} \frac{1}{k-j+1} \left[ \sum_{i=j}^k (z(i) - \mu_0) \right]^2$$

Where:

$z$ - observation,

$k$ - present time instant,

$j$ - time instant of the fault occurrence (hypothetical),

$M$ - time horizon,

$\sigma$ - variance,

$\mu_0$ - mean.

The main advantage of the GLR test is that it links the test threshold with the probability of false alarm and the probability of correct detection. Another advantage of this technique is that it can estimate the jump amplitude and the fault appearing time (Peng et al., 1997). More information regarding GLR algorithm and an alternative approach, which is less time-consuming, can be found in (Nikiforov, 2001).

The residual  $r_1$  in the presence of the fault  $f_1$  and results of the GLR test, e.g. decision function and alarm are given in Figure 12.

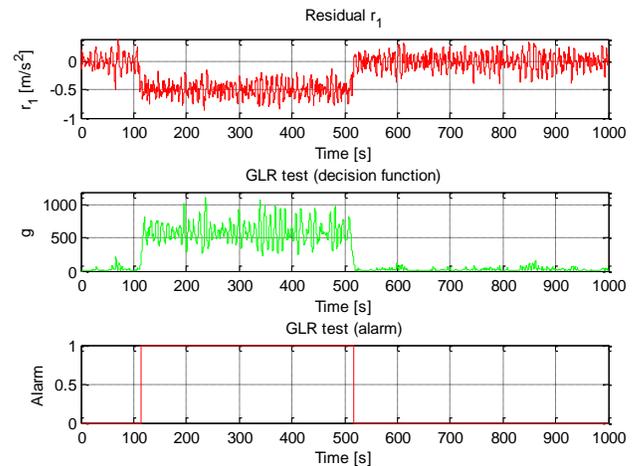


Fig. 12: Residual  $r_1$ , fault  $f_1 = 0.5 m/s^2$  and GLR test (decision function and alarm).

By using the GLR algorithm it is possible to detect a fault  $f_1$ . The alarm diagram is constructed using a decision function and an appropriate threshold. It is possible to notice small delays in the fault detection. The delays in detecting the beginning and the end of the interval of the presence of fault are due to the relatively big noise. Therefore, it has been necessary to

implement a relatively high horizon  $M$  which produces the detection delay. It can be handled by implementing a filter.

In case of the fault  $f_2$ , we have both residuals at our disposal. However, the best result has been obtained using residual  $r_2$  (Figure 13). The fault  $f_2$  can be detected using the decision function and the alarm diagram. It is easy to notice that the system is in an alarm condition in the time interval between 100 seconds and 500 seconds, which is exactly the interval of the presence of a fault.

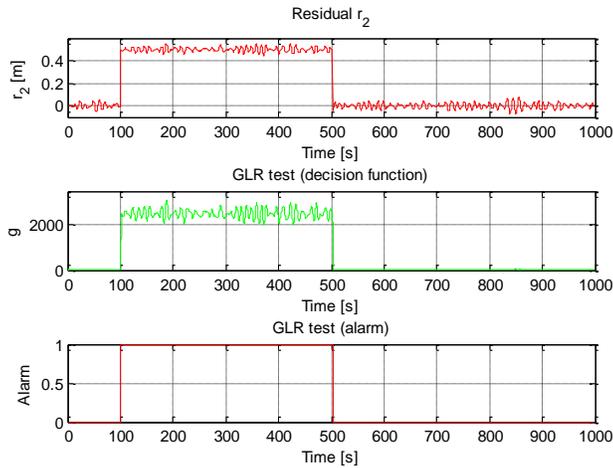


Fig. 13: Residual  $r_2$ , fault  $f_2= 0.5$  m and GLR test (decision function and alarm).

Also, in case of the fault  $f_3$  it is possible to use both residuals, but like in the case of the fault  $f_2$  better results are obtained using the second residual. The residual in the presence of a fault  $f_3$  and the results of the GLR test are shown in Figure 14. The fault  $f_3$  is detected.

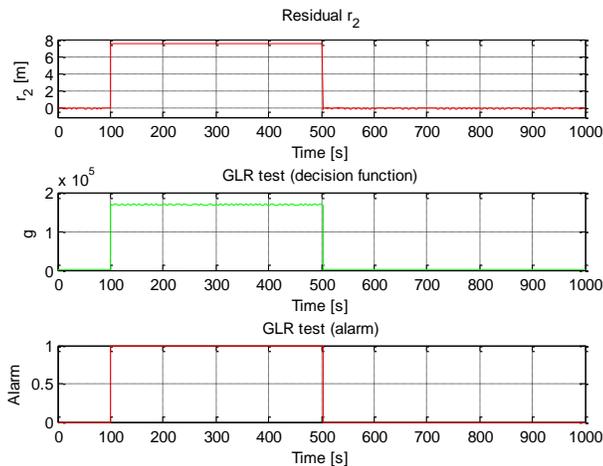


Fig. 14: Residual  $r_2$ , fault  $f_3= 0.05$  rad and GLR test (decision function and alarm).

From the analytical expression of the residual  $r_2$  it can be seen that the faults  $f_4$  and  $f_5$  will create similar effects on the residual. That is, the residual  $r_2$  and the results of the GLR test will be the same in both cases. Therefore, the results for the fault  $f_4$  will be presented only.

The residual  $r_2$  in the presence of a fault  $f_4$  and the results of the GLR test are presented in Figure 15.

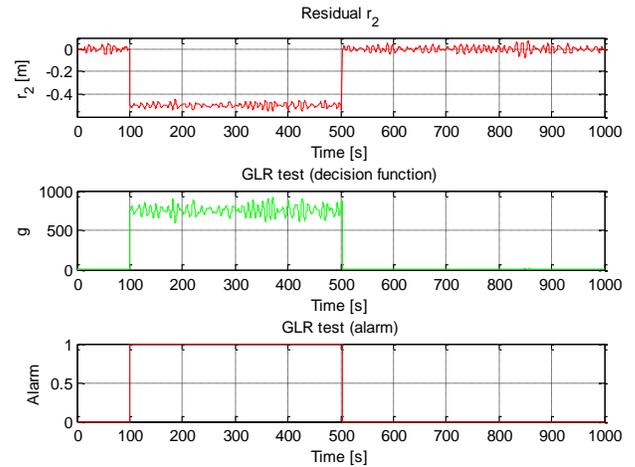


Fig. 15: Residual  $r_2$ , fault  $f_4= 0.5$  m and GLR test (decision function and alarm).

The residual  $r_2$  is very sensitive to faults of sensors  $y_4$  and  $y_5$ . The fault presence is obvious and the GLR test has also easily detected the fault.

## CONCLUSIONS

In this paper the basic idea of a fault-tolerant monitoring and decision support system has been presented. The onboard sensor fault detection for monitoring and decision support systems has been demonstrated by a practical example. The faults have been created by adding an extra signal to the sensor output and the sensitivity of the residuals to the sensor fault has been investigated. The residuals have been evaluated by GLR test. All the faults have been detected and the implementation of the fault-tolerant design has been justified.

## ACKNOWLEDGMENTS

The inspiring discussions with Professor Jørgen Juncher Jensen and Professor Mogens Blanke during the writing of this paper are highly appreciated.

## REFERENCES

- Bhattacharyya, R (1978). *Dynamics of Marine Vehicles*. John Wiley & Sons, New York.
- Blanke, M., M. Kinnaert, J. Lunze, M. Staroswiecki (2006). *Diagnosis and Fault Tolerant Control*, Springer - Verlag, Berlin Heidelberg.
- Dabney, J.B., T.L. Harman (2004). *Mastering Simulink*, Pearson Prentice Hall, New Jersey.

Krysander, M., J. Åslund and M. Nyberg (2008). An efficient algorithm for finding minimal over-constrained subsystems for model-based diagnosis. *IEEE trans. on Systems, Man and Cybernetics - Part A: Systems and Humans*, 38(1).

Nielsen, J.K., N.H. Pedersen, J. Michelesen, U.D. Nielsen, J. Baatrup, J.J. Jensen, E.S. Petersen(2006). SeaSense – Real-time Onboard Decision Support, *Proceedings of WMTC2006*, London, UK.

Nielsen, U.D (2007). Response-based estimation of sea state parameters- influence of filtering, *Ocean Engineering*, **Vol. 34**, **Issue. 13**, 1797-1810.

Nikiforov, I.V (2001). A simple change detection scheme, *Signal Processing*, **Vol. 81**, 149-172.

Peng, Y., A. Youssouf, , Ph. Arte, M. Kinnaert (1997). A complete procedure for residual generation and evaluation with application to heat exchanger. *IEEE Transaction*, CST-5, 542-555.