

# Chapter 10

## Unmanned Surface Vessels



**Abstract** Autonomous navigation of unmanned surface vessels (USVs) (such as ships, hovercrafts, etc), is a significant topic, since it can find use in both security and defence tasks, as well as in maritime transportation. The problem of control and trajectory tracking for unmanned surface vessels (of the ship or hovercraft type) is non-trivial because the associated dynamic and kinematic models are complex non-linear ones. A first problem that arises in controller design for unmanned surface vessels is that trajectory tracking has to be achieved despite modelling uncertainty and external perturbations and thus the control loop must exhibit sufficient robustness. Another problem that has to be dealt with is that the vessels model is often underactuated (the propulsion system consists of less actuators than the vessel's degrees of freedom). The present chapter treats the problem of control of unmanned surface vessels. Solution to the associated control problem is provided through (i) global linearization methods, (ii) approximate linearization methods and (iii) Lyapunov methods. To solve the control problem for unmanned surface vessels without prior knowledge of the associated dynamic model, elaborated real-time estimation methods are developed. These allow for identifying the unknown dynamic model of the vessel and for implementing an indirect adaptive control scheme. Moreover, for the accurate localization of the vessel and for precise computation of its motion characteristics advanced (and precisely validated) nonlinear filtering and distributed filtering are applied. These enable to perform fusion of the measurements of heterogeneous sensors and of state estimates provided by individual distributed local filters. In particular, the chapter treats the following issues: (a) Nonlinear control and Kalman Filtering for a 3-DOF surface vessel, (b) Flatness-based control for the autonomous hovercraft (c) Nonlinear optimal control for autonomous navigation of unmanned surface vessels, and (d) validation of distributed Kalman Filtering for ship tracking applications.

## 10.1 Chapter Overview

The present chapter treats the following issues: (a) Nonlinear control and Kalman Filtering for a 3-DOF surface vessel, (b) Flatness-based control for the autonomous hovercraft (c) Nonlinear optimal control for autonomous navigation of unmanned surface vessels, and (d) validation of distributed Kalman Filtering for ship tracking applications.

With reference to (a) the chapter examines the problem of dynamic ship positioning with the use of Kalman Filter-based and Particle Filter-based sensor fusion algorithms. The proposed approach enables to estimate accurately the ship's state vector by fusing the vessel's position and heading measurements coming from on-board sensors together with distance measurements coming from sensors located at the coast (e.g. radar). The estimated state vector is used in turn in a control loop, to regulate the horizontal position and heading of the vessel.

With reference to (b) the chapter proposes a nonlinear control approach for the underactuated hovercraft model based on differential flatness theory and uses a new nonlinear state vector and disturbances estimation method under the name of Derivative-free nonlinear Kalman Filter. It is proven that the nonlinear model of the hovercraft is a differentially flat one. It is shown that this model cannot be subject to static feedback linearization, however it admits dynamic feedback linearization which means that the system's state vector is extended by including as additional state variables the control inputs and their derivatives. Next, using the differential flatness properties it is also proven that this model can be subject to input-output linearization and can be transformed to an equivalent canonical (Brunovsky) form. Based on this latter description the design of a state feedback controller is carried out enabling accurate maneuvering and trajectory tracking. Additional problems that are solved in the design of this feedback control scheme are the estimation of the nonmeasurable state variables in the hovercraft's model and the compensation of modeling uncertainties and external perturbations affecting the vessel. To this end, the application of the Derivative-free nonlinear Kalman Filter is proposed. This nonlinear filter consists of the Kalman Filter's recursion on the linearized equivalent model of the vessel and of an inverse nonlinear transformation based on the differential flatness features of the system which enables to compute state estimates for the state variables of the initial nonlinear model. The redesign of the filter as a disturbance observer makes possible the estimation and compensation of additive perturbation terms affecting the hovercraft's model.

With reference to (c) the chapter proposes a new nonlinear optimal control approach for autonomous navigation of unmanned surface vessels. The dynamic model of the surface vessels undergoes approximate linearization round local operating points which are redefined at each iteration of the control algorithm. These temporary equilibria consist of the last value of the vessel's state vector and of the last value of the control signal that was exerted on it. For the approximate linearization of the system's dynamics Taylor series expansion is performed through the computation of the associated Jacobian matrices. The modelling errors are compensated by

the robustness of the control algorithm. Next, for the linearized equivalent model of the vessel an H-infinity feedback controller is designed. This requires the solution of an algebraic Riccati equation at each iteration of the computer control program. It is shown that the control scheme achieves H-infinity tracking performance, which implies maximum robustness to modelling errors and external perturbations. The stability of the control loop is proven through Lyapunov analysis.

With reference to (d) the chapter considers that tracking of ships' motion and monitoring of maritime traffic can be performed with the use of distributed Kalman Filtering. However, some of the local Kalman Filters which constitute distributed estimation schemes may depend on inaccurate models of the vessel's dynamics or kinematics and in such a case the aggregate state estimate provided by the distributed filter is unreliable. To treat this problem the chapter proposes a statistical method of optimized performance for the validation of Fuzzy Kalman Filters used in ship tracking. By showing the equivalence of the local Kalman Filters to ARMAX models, the Fuzzy Kalman Filter is proven to be equivalent to fuzzy weighting of local ARMAX models. Using this equivalent modeling of the Fuzzy Kalman Filter, the local statistical approach to fault diagnosis is applied for validating the accuracy of the distributed filter or in the opposite case for detecting the local Kalman Filter that makes use of an imprecise ship model. By applying the Generalized Likelihood Ratio on the residuals of the Kalman Filtering procedure the proposed validation method finally takes the form of a  $\chi^2$  statistical change detection criterion. This statistical validation test is capable of detecting the faulty local filter within the distributed estimation method, even in the case of small errors in the local model's parameters which do not exceed 1% of the associated nominal values.

## 10.2 Nonlinear Control and Filtering for a 3-DOF Surface Vessel

### 10.2.1 Outline

During the last years, research on marine navigation systems and on autonomous vessels have grown rapidly. Modern marine vessels are equipped with sophisticated motion-control systems which accomplish various control objectives such as position and heading regulation, trajectory tracking, and wave-induced motion compensation [148, 189, 462]. Motion control operates in the three planar degrees of freedom, i.e. surge (forward motion), sway (transverse motion), and yaw (rotation about the vertical axis, also called heading) and is implemented through the feedback of information from position and heading measurements. To estimate accurately the vessel's position, measurements coming from GPS, radar or an IMU can be used, while to estimate the orientation of the vessel, fusion of measurements coming from magnetic compasses and gyroscopes can be performed. The term "Dynamic positioning" describes the use of the propulsion system, in a control loop, to regulate the horizontal position and heading of the vessel. Early Dynamic Positioning Systems (DPS)

were based on three term PID control with a notch filter in order to counteract high frequency motion due to waves. There exist also results on nonlinear filtering for autonomous navigation systems which have been presented in [110, 310, 567, 592]. The present section studies sensor fusion-based dynamic positioning for ships using Kalman and Particle Filtering methods.

Sensor-fusion based motion estimation using probabilistic inference forms a core component in most modern guidance and navigation systems [234]. The estimator fuses observations from multiple sensors with predictions from a dynamic state-space model of the system under control. The most widely used algorithms for multi-sensor fusion are variants of the Kalman Filter (KF), which in the case of nonlinear dynamical models take the form of the Extended Kalman Filter. Currently, Kalman Filtering is a main element in the design of Dynamic Positioning Systems [163, 477, 547]. A basic assumption made by Kalman Filtering is that of Gaussian process/measurement noise. On the other hand, the Extended Kalman Filter is based on the linearization of the system dynamics, and proceeds with the recursive estimation of the standard Kalman Filter [431, 433, 457].

A different approach to filtering and sensor fusion-based state estimation is Particle Filtering. The Particle Filter is a non-parametric state estimator which unlike the KF or the EKF does not make any assumption on the probability density function of the measurements [23, 555]. The concept of particle filtering comes from Monte-Carlo methods. The Particle Filter (PF) can provide optimal estimation in non-Gaussian state-space models. In the case of nonlinear dynamical models the PF avoids also the calculations associated with the Jacobians which appear in the EKF equations [271, 625]. The main stages of the PF are prediction (time update), correction (measurement update) and resampling for substituting the unsuccessful state vector estimates with those particles that have better approximated the real state vector.

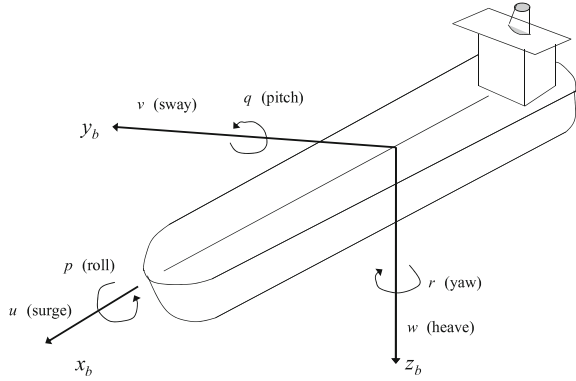
The main developments of the section are outlined in the following: (i) design of a Kalman Filter-based disturbance estimator that enables simultaneous estimation of the ship's state vector and of the vector of external disturbances, through the processing of measurements from various types of sensors, (ii) design of a Particle Filter disturbance estimator that enables simultaneous estimation of the ship's state vector and of the vector of external disturbances, again through the fusion of measurements from various sensors, (iii) implementation of state estimation-based control using these nonlinear filtering methods.

## ***10.2.2 Kinematic and Dynamic Models of Vessels for the Problem of Dynamic Positioning***

### **10.2.2.1 A Generic Kinematic and Dynamic Ship Model**

The motion of a ship is described by two reference frames: (i) a local geographical earth-fixed frame and, (ii) a body-fixed frame denoted as  $X_b Y_b Z_b$  which is attached

**Fig. 10.1** Components of the linear velocity vector of the vessel (surge, sway and heave) and components of the angular velocity of the vessel (roll, pitch and yaw Euler angles)



to the vessel (see Fig. 10.1). The components of the position vector of the vessel are  $[x, y, \psi]^T$  where  $(x, y)$  are the coordinates of the ship’s center of symmetry in a local geographical frame and  $\psi$  is the orientation angle with reference to the  $OX$  axis of the local coordinates frame [148].

The components of the ship’s velocity vector, denoted as  $v = [u, v_s, r]^T$ , are the surge and sway velocities  $(u, v_s)$  and the yaw rate  $r$ . A model for vessel kinematics, relating the ship’s position vector  $\eta$  to the ship velocity vector  $v$ , is

$$\dot{\eta} = R(\psi)v \tag{10.1}$$

The kinematic transformation of Eq. (10.1) relates the body-fixed velocities to the position derivatives in the local geographical frame. The transformation is described by matrix  $R(\psi) \in \mathbb{R}^{3 \times 3}$  which performs a rotation round the  $z$ -axis by an angle  $\psi$ . The equation of the ship dynamics describes the relation between the ship’s velocity and the generalized forces vector (forces and torques  $\tau_{control}$ ,  $\tau_{wind}$  and  $\tau_{waves}$ ) which is applied to the vessel [148, 167].

$$M\dot{v} + C_{RB}(v)v + d(V_{rc}, \gamma_c) = \tau_{control} + \tau_{wind} + \tau_{waves} \tag{10.2}$$

In the above equation,  $C_{RB}(v)v$  denotes Coriolis-centripetal terms while  $d(V_{rc}, \gamma_c)$  denotes disturbance terms (e.g. due to wind and currents). The inertia matrix  $M$  is the sum of two matrices  $M_A$  and  $M_{RB}$ , i.e.  $M = M_A + M_{RB}$  where

$$M_A = \begin{pmatrix} -X_{\ddot{u}} & 0 & 0 \\ 0 & -Y_{\ddot{v}_s} & -Y_{\ddot{r}} \\ 0 & -Y_{\ddot{r}} & -N_{\ddot{r}} \end{pmatrix} \quad M_{RB} = \begin{pmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{pmatrix} \tag{10.3}$$

In the positive-definite hydrodynamic matrix  $M_A$ , the added-mass coefficients  $X_{\ddot{u}}$ ,  $Y_{\ddot{v}_s}$ , and  $N_{\ddot{r}}$  depend on the hull shape and show the change in momentum in the fluid due to the vessel accelerations. On the other hand in the equation of the positive

definite rigid-body mass matrix  $M_{RB}$ , parameter  $x_g$  denotes the longitudinal position of the center of gravity of the vessel relative to the body-fixed frame. The Coriolis-centripetal terms matrix  $C_{RB}$  is given by

$$C_{RB} = \begin{pmatrix} 0 & 0 & -m(x_g r + v_s) \\ 0 & 0 & mu \\ m(x_g r + v_s) & -mu & 0 \end{pmatrix} \quad (10.4)$$

When a vessel operates under positioning control the velocities are small and thus the Coriolis-centripetal terms  $C_{RB}(v)v$  in Eq. (10.2) can be omitted from the ship's dynamic model.

### 10.2.2.2 Ship Model for the Dynamic Positioning Problem

As noted before, the term  $d(V_{rc}, \gamma_c)$  on the left hand side of Eq. (10.2) represents the current and damping forces. The speed of the current is denoted as  $V_{rc}$  while the angle of the current is denoted as  $\gamma_{rc}$  and is defined relative to the bow of the vessel [148]. It is common practice to write the current forces in surge, sway and yaw as functions of non-dimensional current coefficients  $C_{X_c}(\gamma_{rc})$ ,  $C_{Y_c}(\gamma_{rc})$ ,  $C_{N_c}(\gamma_{rc})$  which is

$$d(V_{rc}, \gamma_{rc}) = \frac{1}{2} \rho V_{rc}^2 \begin{pmatrix} A_{F_c} C_{X_c}(\gamma_{rc}) \\ A_{L_c} C_{Y_c}(\gamma_{rc}) \\ A_{L_c} L_{0\alpha} C_{N_c}(\gamma_{rc}) \end{pmatrix} \quad (10.5)$$

where  $\rho$  is the water density,  $A_{F_c}$  and  $A_{L_c}$  are frontal and lateral projected areas of the submerged part of the vessel and  $L_{0\alpha}$  is the length of the ship. However, the current coefficients  $C_{X_c}(\gamma_{rc})$ ,  $C_{Y_c}(\gamma_{rc})$ ,  $C_{N_c}(\gamma_{rc})$  are difficult to estimate with accuracy. In such cases, one can simplify the model of Eq. (10.5), in terms of a linear damping term and a bias term which finally takes the form

$$d(V_{rc}, \gamma_{rc}) \simeq D(v)v - R^T(\psi)d \quad (10.6)$$

$$\text{where } D = D^T = \begin{pmatrix} D_{11} & 0 \\ 0 & D_{22} & D_{23} \\ 0 & D_{32} & D_{33} \end{pmatrix}, \quad d = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} \quad (10.7)$$

The wind forces and moments can be represented in a similar way to the current forces and moments, i.e.

$$\tau_{wind} = \frac{1}{2} \rho_{\alpha} V_{rw}^2 \begin{pmatrix} A_{Fw} C_{Xw}(\gamma_{rw}) \\ A_{Lw} C_{Yw}(\gamma_{rw}) \\ A_{Lw} L_{0\alpha} C_{Nw}(\gamma_{rw}) \end{pmatrix} \quad (10.8)$$

where  $\rho_{\alpha}$  is the air density,  $A_{Fw}$  and  $A_{Lw}$  are the frontal and lateral projected wind areas and  $L_{0\alpha}$  is the vessel's overall length. The wind speed is  $V_{rw}$  and its direction is  $\gamma_{rw}$  in earth-fixed coordinates. The wind model coefficients can be obtained by model tests while with reference to the control problem, obtaining measurements of the wind's speed and direction enables to compensate  $\tau_{wind}$  using a feed-forward term  $\hat{\tau}_{wind}$ . The difference (modeling error) between  $\tau_{wind}$  and  $\hat{\tau}_{wind}$  can be described by a bias term  $R^T(\psi)d$ , as in the case of the current bias term that was given in Eq. (10.6).

Wave forces are usually modeled as the sum of a linear and a nonlinear component, i.e.

$$\tau_{waves} = \tau_{waves}^{lin} + \tau_{waves}^{nlin} \quad (10.9)$$

The low-frequency nonlinear wave forces can be modeled again by a bias term, and considered to be input disturbances. On the other hand the linear wave forces are considered to be output disturbances. Therefore, the observation (measurement) equation of the ship is given by  $z = \eta + n_w + v_1$ , where  $n$  is the vessel's position calculated using the ship's dynamic model of Eqs. (10.1) and (10.2),  $v_1$  is sensor measurement noise and  $n_w$  is the ship's displacement due to the linear wave forces.

Using Eq. (10.6) and the above assumptions about the wind and waves forces, the vessel's kinematic and dynamic model described in Eqs. (10.1) and (10.2) respectively, is given by

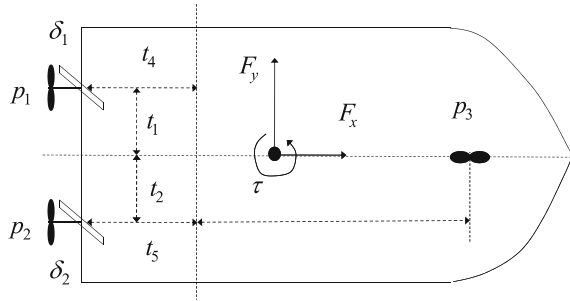
$$\begin{aligned} \dot{\eta} &= R(\psi)v \\ \dot{v} + M^{-1}Dv &= M^{-1}[R^T(\psi)d + \tau_{control}] + w \\ \dot{d} &= w \\ z &= \eta + n_w + v_1 \text{ or } z = n + v \end{aligned} \quad (10.10)$$

The bias is an additive disturbance in the ship's dynamic model which can be estimated with the use of a state observer. Once the bias is accurately estimated it can be compensated by a suitable control term in the right hand side of Eq. (10.10). This additional control term provides the required robustness to compensate for the bias effects.

### 10.2.3 Ship Actuator Model

Without loss of generality the model of a vessel with two propellers and one bow thruster is considered (see Fig. 10.2). The vector of the ship's control forces and torques  $\tau \in R^3$  is related to propeller pitch ratios vector  $u$  (or propeller revolutions

**Fig. 10.2** Model of a vessel with two propellers and a bow thruster



for fixed blade propellers) as follows [171]

$$\tau = T \cdot K(U) \cdot u \tag{10.11}$$

where  $U$  is the magnitude of the ship’s velocity in the  $xy$ -plane i.e.  $U = \sqrt{u^2 + v^2}$  while  $u$  denotes the surge velocity and  $v$  denotes the sway velocity. Vector  $u$  is defined as  $u = [f_1(p_1), f_2(p_2), f_3(p_3), f_4(\delta_1), f_5(\delta_2)]^T$ . For the (fully actuated) ship model of Fig. 10.2 with two propellers  $p_1$  and  $p_2$ , one thruster  $p_3$  and two rudders  $\delta_1$  and  $\delta_2$ , matrix  $T \in R^{3 \times 6}$  depends on the position of the actuators  $p_1, p_2$  and  $p_3$ , while matrix  $K(U) \in R^{6 \times 6}$  depends on the ship’s velocity and the type of the actuators. The coefficients of matrices  $T$  and  $K$  are defined as follows:  $p_i, (i = 1, 2, 3)$  are the propeller pitch ratios (or for fixed-blade propellers are the propeller revolutions),  $\delta_i, (i = 1, 2)$  are the rudder angles,  $t_i, (i = 1, \dots, 5)$  are distances to of the actuators from the ship’s symmetry axes, and  $k_i, (i = 1, \dots, 5)$  are the force coefficients.

### 10.2.4 Feedback Linearization for Ship Dynamic Positioning

#### 10.2.4.1 Nonlinear Positioning Control of the Ship Model

As mentioned above, the kinematic and dynamic model of the ship is given by

$$\begin{aligned} \dot{\eta} &= R \cdot v \\ M \dot{v} + D(v)v - R^T d &= \tau \end{aligned} \tag{10.12}$$

From the previous equation one obtains  $v = R^{-1} \dot{\eta}$ , or since  $R^T = R^{-1}$  it can be written as  $v = R^T \dot{\eta}$ . Similarly one obtains  $\dot{v} = \dot{R}^T \dot{\eta} + R^T \ddot{\eta}$ . Consequently, this gives [140, 227, 462]

$$J(\eta) \ddot{\eta} + C(\eta, \dot{\eta}) \dot{\eta} + F(\eta) \dot{\eta} - d = \tau^* \tag{10.13}$$



where the ship model's parameters are defined as

$$\begin{aligned} J(\eta) &= RMR^T \in \mathbb{R}^{3 \times 3} & C(\eta, \dot{\eta}) &= RMR^T \in \mathbb{R}^{3 \times 3} \\ F(\eta) &= RDR^T \in \mathbb{R}^{3 \times 3} & \tau^* &= R\tau \end{aligned} \quad (10.14)$$

and denoting  $\sin(\psi)$  and  $\cos(\psi)$  as  $S_\psi$  and  $C_\psi$  respectively, while using  $m_{ij}$ ,  $i, j = 1, \dots, 3$  to represent the elements of the inertia matrix and  $d_{ij}$ ,  $i, j = 1, \dots, 3$  to represent the elements of the damping matrix, the terms of the ship's dynamic model are described by [140]

$$J(\eta) = \begin{pmatrix} m_{11}C_\psi^2 + m_{22}S_\psi^2 & (m_{11} - m_{22})S_\psi C_\psi & -m_{23}S_\psi \\ (m_{11} - m_{12})S_\psi C_\psi & m_{11}S_\psi^2 + m_{22}C_\psi^2 & m_{23}C_\psi \\ -m_{23}S_\psi & m_{23}C_\psi & m_{33} \end{pmatrix} \quad (10.15)$$

$$C(\eta, \dot{\eta}) = \begin{pmatrix} \dot{\psi}(m_{22} - m_{11})S_\psi C_\psi & \dot{\psi}(m_{11}C_\psi^2 + m_{22}S_\psi^2) & 0 \\ -\dot{\psi}(m_{11}S_\psi^2 + m_{22}C_\psi^2) & \dot{\psi}(m_{22} - m_{11})S_\psi C_\psi & 0 \\ -\dot{\psi}(m_{23}C_\psi & -\dot{\psi}(m_{23}S_\psi & 0 \end{pmatrix} \quad (10.16)$$

$$F(\eta) = \begin{pmatrix} d_{11}C_\psi^2 + d_{22}S_\psi^2 & (d_{11} - d_{12})S_\psi C_\psi & -d_{23}S_\psi \\ (d_{11} - d_{12})S_\psi C_\psi & d_{11}S_\psi^2 + d_{22}C_\psi^2 & d_{23}C_\psi \\ -d_{32}S_\psi & d_{32}C_\psi & d_{23} \end{pmatrix} \quad (10.17)$$

The control signal is chosen to be

$$\begin{aligned} \tau^* &= J(\eta)[\ddot{\eta}_d + J(\eta)^{-1}C(\eta, \dot{\eta})\dot{\eta} + \\ &+ J(\eta)^{-1}F(\eta)\dot{\eta} - J(\eta)^{-1}d - K_D\dot{\tilde{\eta}} - K_P\tilde{\eta}] \end{aligned} \quad (10.18)$$

where  $\tilde{\eta} = \eta - \eta_d$  is the tracking error, while

$$\begin{aligned} K_D &= \text{diag}[k_{d_1}, k_{d_2}, k_{d_3}] \\ K_P &= \text{diag}[k_{p_1}, k_{p_2}, k_{p_3}] \end{aligned} \quad (10.19)$$

are feedback gain matrices. This finally results into the tracking error dynamics

$$\begin{aligned} \ddot{\eta} - \ddot{\eta}_d + K_D\dot{\tilde{\eta}} + K_P\tilde{\eta} &= 0 \\ \text{or } \ddot{\tilde{\eta}} + K_D\dot{\tilde{\eta}} + K_P\tilde{\eta} &= 0 \end{aligned} \quad (10.20)$$

### 10.2.5 Joint Estimation of the Ship's State Vector and of Unknown Additive Disturbances

The sensor fusion-based estimation procedure for obtaining the ship's state vector is affected by uncertainties characterizing the ship's dynamic model. Such uncertainties can be due to parametric variations in the model of Eqs. (10.13) and (10.14) or due to external disturbances, e.g. additive input disturbances as shown in Eqs. (10.2) and (10.10). Simultaneous estimation of a dynamical system's state vector and of the disturbances vector can be achieved using disturbance observers [82, 87, 105, 106, 180, 256, 341, 428, 623].

In the case of a surface vessel, defining the generalized state vector  $x = [\eta, d, \dot{\eta}, \dot{d}]^T$  and considering invariance of the disturbance  $d$  for specific time periods, one obtains the generalized ship state-space model

$$\begin{aligned} \ddot{\eta} + J(\eta)^{-1}[C(\eta, \dot{\eta}) + F(\eta)]\dot{\eta} - J^{-1}(\eta)d &= J^{-1}(\eta)\tau \\ \ddot{d} &= 0. \end{aligned} \quad (10.21)$$

Setting  $x_1 = \eta$ ,  $x_2 = d$ ,  $x_3 = \dot{\eta}$ ,  $x_4 = \dot{d}$  and taking into account the existence of process and measurement noise one obtains a ship's model of the form

$$\begin{aligned} \dot{x} &= Ax + Bu + w \\ z &= \gamma(x) + v \end{aligned} \quad (10.22)$$

where matrices  $A$  and  $B$  are given by

$$\begin{aligned} A &= \begin{pmatrix} 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} \\ 0_{3 \times 3} & J^{-1}(x) & -J^{-1}(x)[C(x, \dot{x}) + F(x)] & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{pmatrix} \\ B &= (0_{3 \times 3} \ 0_{3 \times 3} \ J^{-1}(x) \ 0_{3 \times 3})^T \end{aligned} \quad (10.23)$$

The extended state vector is  $x = [x_1, x_2, x_3, x_4]^T$  with  $x_i \in R^{3 \times 1}$ ,  $i = 1, 2, 3, 4$ . The control input is  $\tau \in R^{3 \times 1}$ . The measurement vector of the ship's model is given by  $z = [x, y, \psi, d^1]^T$ , where  $x, y$  are measurements of the ship's cartesian coordinates,  $\psi$  is a measurement of the ship's orientation and  $d^1$  is a measurement of the ship's distance from the coast, provided by a coastal sensor (e.g. radar). The vectors of process and measurement noises are denoted as  $w$  and  $v$ , respectively. Using the above state-space representation, state vector  $x$  can be estimated by processing a sequence of output measurements  $y$  with the use of a state observer or Kalman Filtering [46, 149, 631].

It is noted that disturbance terms affecting the ship's model, as shown in Eq. (10.10), can be identified with the use of disturbance observers were initially

conceived with a static observer gain [82, 87, 180, 256, 341, 623]. However they can be suitably modified so as to be based on dynamic adaptation of the observer gain through the Kalman Filter recursion [105, 428]. Once the disturbances vector is estimated, a supervisory control term can be introduced in the control law so as to annihilate the disturbances effects. A common technique is the Unknown Input Observer which estimates both the states of the system and the disturbance by augmenting a linear design model with a linear disturbance model. Another approach is based on the Extended State Observer. This has the state and disturbance estimation power of an Unknown Input Observer while being also simpler in tuning. Another solution to the problem of simultaneous state and disturbance estimation comes from the Perturbation Observer. The Perturbation observer is suitable not only for estimation of additive disturbances but also for estimation of unmodeled variations of the monitored system. In place of static observer gain for the aforementioned observers one can consider on-line adaptation of the observer's gain through the Kalman Filter recursion. Therefore it is possible to design Kalman Filter-based disturbance observers exhibiting the advantages of Kalman Filter estimation such as minimization of the estimation error and smoother convergence of the estimated state variables towards the real state variables.

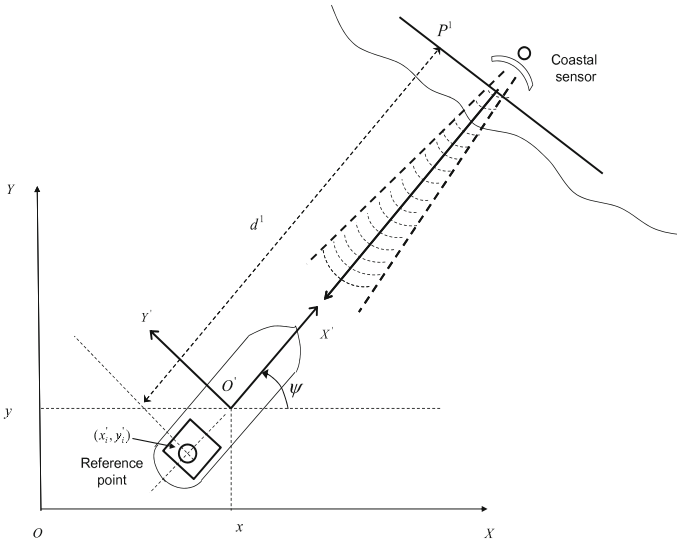
### 10.2.6 Sensor Fusion for the Surface Vessel Using Kalman Filtering

The application of EKF to the fusion of data that come from different sensors of the monitored surface vessel is examined first. The ship's kinematic and dynamic model is considered again. It is assumed that at each time instant measurements of the ship's cartesian coordinates  $(x, y)$  as well as of the ship's heading  $\psi$  are available. Moreover the distance of the ship from the coast is provided by a coastal sensor (e.g. radar). Fusing the aforementioned measurements with the use of a stochastic estimation algorithm, such as the Kalman Filter, can provide an accurate estimate of the ship's state vector.

The coordinates of the center of symmetry of the ship with respect to  $OXY$  (inertial coordinates system) are  $(x, y)$ , while the coordinates of a reference point  $i$  of the ship (e.g. bridge), with respect to  $O'X'Y'$  (body-fixed coordinates system) are  $x'_i, y'_i$  (Fig. 10.3). The orientation of the ship's reference point with respect to  $O'X'Y'$  is  $\psi'_i$ .

Thus the coordinates of the reference point  $i$  with respect to  $OXY$  are  $(x_i, y_i)$  and its orientation is  $\psi_i$ , and are given by

$$\begin{aligned} x_i(k) &= x(k) + x'_i \sin(\psi(k)) + y'_i \cos(\psi(k)) \\ y_i(k) &= y(k) - x'_i \cos(\psi(k)) + y'_i \sin(\psi(k)) \\ \psi_i(k) &= \psi(k) + \psi_i \end{aligned} \quad (10.24)$$



**Fig. 10.3** Estimation of the ship’s state vector by fusing the measurements of its position and orientation with the measurement of its distance from the coast provided by a coastal sensor (e.g. radar)

Each reference plane  $P^j$  on the coast can be represented by  $P_r^j$  and  $P_n^j$  (Fig. 10.3), where (i)  $P_r^j$  is the normal distance of the plane from the origin  $O$ , (ii)  $P_n^j$  is the angle between the normal line to the plane and the  $x$ -direction. Using the above notation, the distance of the ship’s reference point  $i$  (e.g. bridge), from the reference plane  $P^j$  on the coast depends on  $P_r^j$ ,  $P_n^j$  (see Fig. 10.3) [433]:

$$d^j(k) = P_r^j - x_i(k)\cos(P_n^j) - y_i(k)\sin(P_n^j). \tag{10.25}$$

By definition of the measurement vector one has that the output function  $\gamma(x(k))$  is given by

$$\gamma(x(k)) = [x(k), y(k), \psi(k), d^j(k)]^T \tag{10.26}$$

To obtain the Extended Kalman Filter (EKF), the model of the ship is linearized about the estimates  $\hat{x}(k)$  and  $\hat{x}^-(k)$  as described in the previous subsection. The process noise covariance matrix  $Q(k) \in R^{12 \times 12}$  and the measurement noise matrix  $R \in R^{4 \times 4}$  are taken to be diagonal. The Kalman Filter gain is  $K \in R^{12 \times 4}$ . For matrix  $\gamma$  appearing in the ship’s output equation it holds

$$\gamma(\hat{x}(k)) = [\hat{x}(k), \hat{y}(k), \hat{\psi}(k), P_r^j - x_i(k)\cos(P_n^j) - y_i(k)\sin(P_n^j)]^T \tag{10.27}$$

The Jacobian of the ship model’s output  $\gamma$  with respect to the state vector  $x(k)$  is thus,

$$J_y^T(\hat{x}^-(k)) = \begin{pmatrix} 1 & 0 & 0 & 0_{1 \times 9} \\ 0 & 1 & 0 & 0_{1 \times 9} \\ 0 & 0 & 1 & 0_{1 \times 9} \\ \alpha_{41} & \alpha_{42} & \alpha_{43} & 0_{1 \times 9} \end{pmatrix} \quad (10.28)$$

where  $\alpha_{41} = -\cos(P_n^j)$ ,  $\alpha_{42} = -\sin(P_n^j)$  and  $\alpha_{43} = \{x'_i \cos(\psi - P_n^j) - y'_i \sin(\psi - P_n^j)\}$ . As analyzed in Sect. 10.2.4, the ship can be steered along the reference trajectory using the estimated state vector and control based on feedback linearization of the ship's dynamic model. Alternatively nonlinear backstepping control can be used [462].

For the dynamic model of the vessel that is described by Eqs. (10.161) and (10.162) sensor fusion-based state estimation can be performed using Kalman Filtering. As shown for instance in Eqs. (4.88) and (4.89) the implementation stages of the Kalman Filter comprise a measurement-update part and a time-update part.

### 10.2.7 Particle Filter-Based Sensor Fusion for Estimating the Ship's Motion and Disturbances

As in the KF case, the Particles Filter consists also of the measurement update (correction stage) and the time update (prediction stage) [271, 555, 625]. The prediction stage calculates

$$p(x(k)|Z^-) \text{ where } Z^- = \{z(1), z(2), \dots, z(k-1)\} \quad (10.29)$$

are output measurements up to time instant  $k-1$ . It holds that

$$p(x(k-1)|Z^-) = \sum_{i=1}^N w_{k-1}^i \delta_{\xi_{k-1}^i}(x(k-1)) \quad (10.30)$$

while from Bayes formula it holds

$$p(x(k)|Z^-) = \int p(x(k)|x(k-1))p(x(k-1)|Z^-)dx \quad (10.31)$$

From the above one finally obtains:

$$p(x(k)|Z^-) = \sum_{i=1}^N w_{k-1}^i \delta_{\xi_{k-}^i}(x(k)) \quad (10.32)$$

with  $\xi_{k-}^i \sim p(x(k)|x(k-1)) = \xi_{k-1}^i$

The previous relation means that the state equation of the system is executed  $N$  times, starting from the  $N$  previous values of the state vectors  $x(k-1) = \xi_{k-1}^i$ . Consequently, the value of the state vector which is calculated in the prediction stage is the result of the weighted averaging of the state vectors which were computed after running the state equation, starting from the  $N$  previous values of the state vectors  $\xi_{k-1}^i$ .

The a-posteriori probability density is found as follows: a new position measurement  $z(k)$  is obtained and the objective is to calculate the corrected probability density

$$p(x(k)|Z) \text{ where } Z = \{z(1), z(2), \dots, z(k)\} \quad (10.33)$$

From Bayes law it holds that

$$p(x(k)|Z) = \frac{p(Z|x(k))p(x(k))}{p(Z)} \quad (10.34)$$

which can be also written as

$$p(x(k)|Z) = \frac{p(z(k)|x(k))p(x(k)|Z^-)}{\int p(z(k)|x(k), Z^-)p(x(k)|Z^-)dx} \quad (10.35)$$

After intermediate calculations one finally obtains

$$p(x(k)|Z) = \sum_{i=1}^N w_k^i \delta_{\xi_{k-}^i}(x(k))$$

$$\text{where } w_k^i = \frac{w_{k-}^i p(z(k)|x(k) = \xi_{k-}^i)}{\sum_{j=1}^N w_{k-}^j p(z(k)|x(k) = \xi_{k-}^j)} \quad (10.36)$$

The previous equation denotes the corrected value for the state vector. The recursion of the PF proceeds in a way similar to the update of the Kalman Filter or the Extended Kalman Filter [450, 555].

*Measurement update:* Acquire  $z(k)$  and compute the new value of the state vector

$$p(x(k)|Z) = \sum_{i=1}^N w_k^i \delta_{\xi_{k-}^i}(x(k))$$

$$\text{with corrected weights } w_k^i = \frac{w_{k-}^i p(z(k)|x(k) = \xi_{k-}^i)}{\sum_{j=1}^N w_{k-}^j p(z(k)|x(k) = \xi_{k-}^j)} \quad (10.37)$$

$$\text{and } \xi_k^i = \xi_{k-}^i$$

*Resampling:* Substitute the degenerated particles. The particles of low weight factors are removed and their place is occupied by duplicates of the particles with high weight factors.

*Time update:* compute state vector  $x(k + 1)$  according to the pdf

$$p(x(k + 1)|Z) = \sum_{i=1}^N w_k^i \delta_{\xi_k^i}(x(k)) \quad (10.38)$$

where  $\xi_k^i \sim p(x(k + 1)|x(k) = \xi_k^i)$

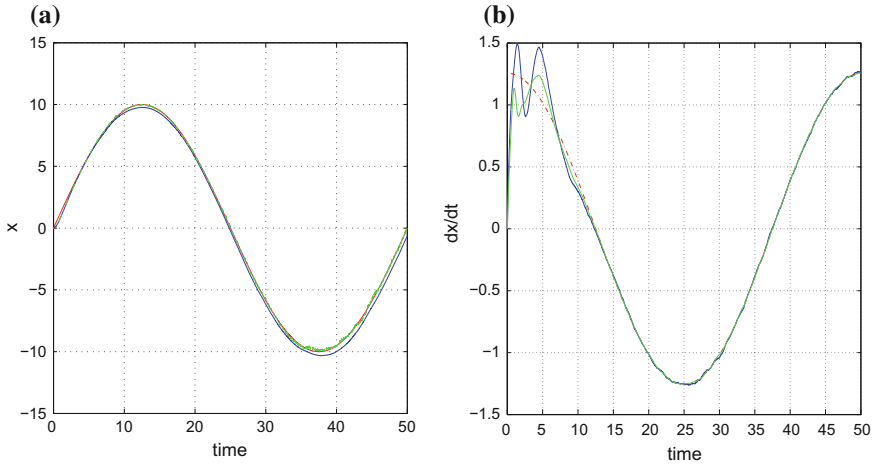
Knowing the measured value of the ship's position and orientation  $[x, y, \psi]$ , one can assign a weight to each particle (estimate of the state vector  $[\hat{x}, \hat{y}, \hat{\psi}]_i$ ), according to how closely the particle approaches the measured state vector. Similarly, knowing the distance  $d^1$  from the coastal reference surface, and calculating an estimation of this distance  $\hat{d}^1$  for every particle  $[\hat{x}, \hat{y}, \hat{\psi}]_i$ , one can assign a weight to the particle according to the accuracy of estimation of the distance  $d^1$ . Further averaging of these two weight values associated with each particle provides the aggregate particle's weight which is used in the Particle Filter's iteration.

## 10.2.8 Simulation Tests

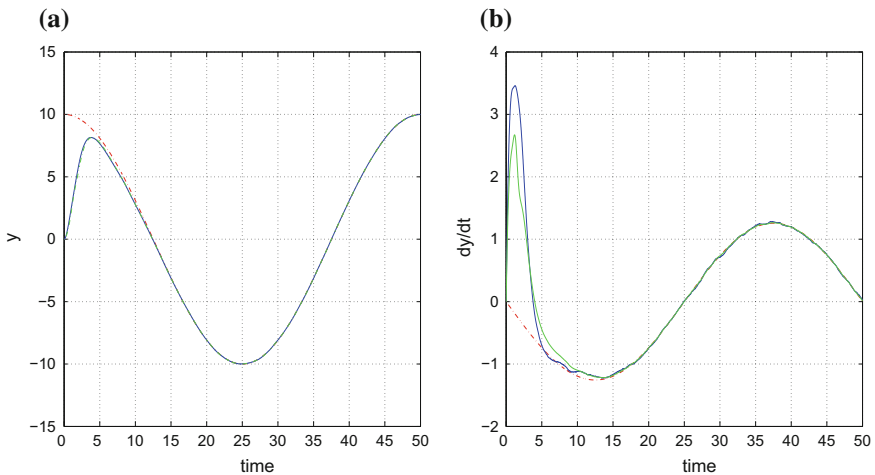
### 10.2.8.1 Dynamic Ship Positioning with the Use of Kalman Filtering

The use of Kalman and Particle Filtering for fusing the data that come from the ship's navigation instruments with the measurements that come from coastal sensors provides an estimation of the state vector  $[x(t), y(t), \psi(t)]$  and enables the successful application of nonlinear steering control. In the implementation of the Kalman Filter the process noise covariance matrix  $Q \in R^{12 \times 12}$  and the measurement noise covariance matrix  $R \in R^{4 \times 4}$  were taken to be diagonal with nonzero elements equal to  $10e^{-3}$ . The number of particles used by the PF was  $N = 1000$ . From the simulation experiments it can be observed that the KF and the PF provide accurate estimations of the external disturbances. Thus, an auxiliary control term based on the disturbances estimation can be included in the right hand side of Eq. (10.18), and can compensate for the disturbances' effects.

The following cases were examined: (i) Kalman Filtering-based ship's Dynamic Positioning (DP) through tracking of a circular trajectory. The associated results are shown in Figs. 10.4, 10.5, 10.6, 10.7 and 10.8 (ii) Kalman Filtering-based ship's DP through tracking of an eight-shaped trajectory. The associated results are shown in Figs. 10.9, 10.10, 10.11, 10.12 and 10.13 (iii) Kalman Filtering-based ship's DP through tracking of a complex curved trajectory. The associated results are shown in Figs. 10.14, 10.15, 10.16, 10.17 and 10.18 (iv) Particle Filtering-based ship's DP through the tracking of a circular trajectory. The associated results are shown in Figs. 10.19, 10.20, 10.21, 10.22 and 10.23 (v) Particle Filtering-based ship's DP through the tracking of an eight-shaped trajectory. The associated results are shown in Figs. 10.24, 10.25, 10.26, 10.27 and 10.28 (vi) Particle Filtering-based ship's DP through the tracking of a complex curved trajectory. The associated results are shown in Figs. 10.29, 10.30, 10.31, 10.32 and 10.33.

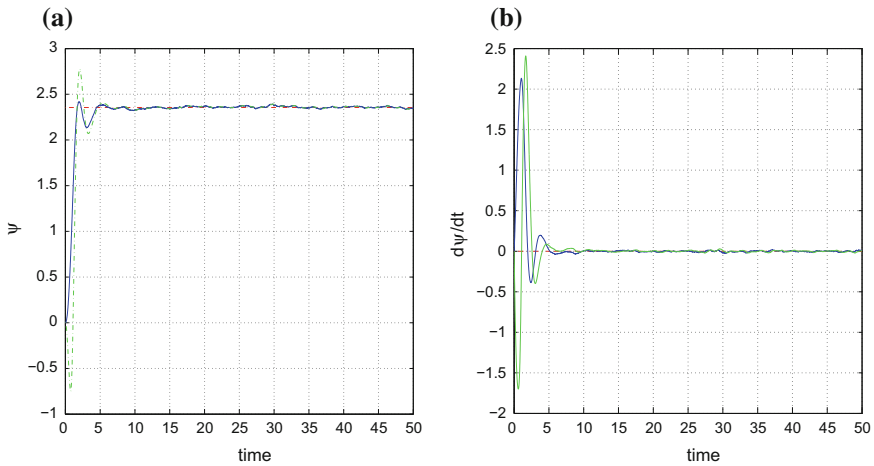


**Fig. 10.4** Tracking of a circular trajectory: **a** KF-based estimation of the ship's position along the  $x$ -axis (green line) and desirable  $x$ -axis position (red line), **b** KF-based estimation of the ship's velocity along the  $x$ -axis (green line) and desirable  $x$ -axis velocity (red line)

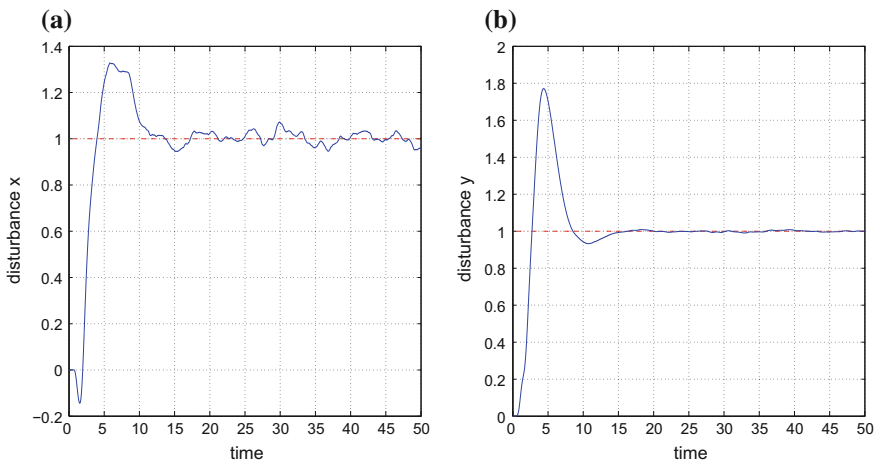


**Fig. 10.5** Tracking of a circular trajectory: **a** KF-based estimation of the ship's position along the  $y$ -axis (green line) and desirable  $y$ -axis position (red line), **b** KF-based estimation of the ship's velocity along the  $y$ -axis (green line) and desirable  $y$ -axis velocity (red line)

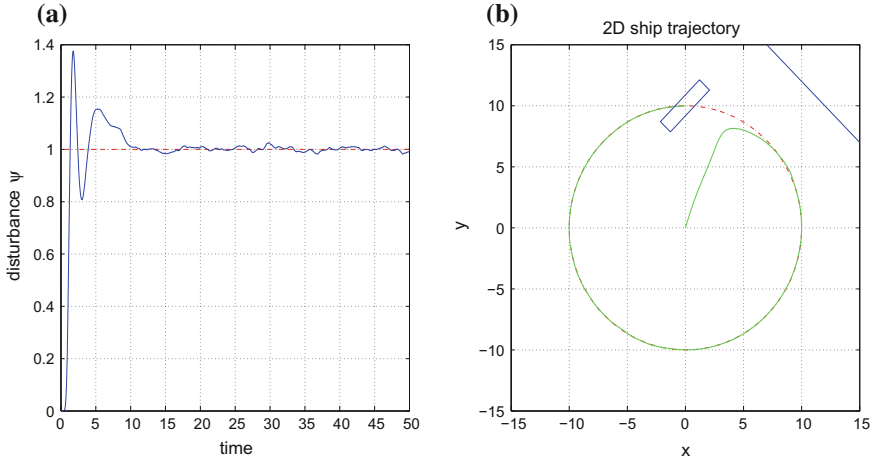




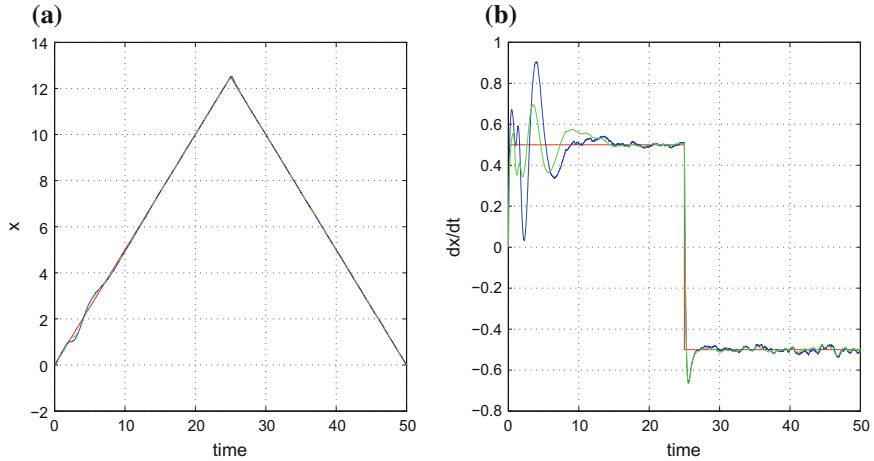
**Fig. 10.6** Tracking of a circular trajectory: **a** KF-based estimation of the ship's angle round the  $z$ -axis (green line) and desirable  $z$ -axis rotation angle (red line), **b** KF-based estimation of the ship's angular velocity round the  $z$ -axis (green line) and desirable angular velocity (red line)



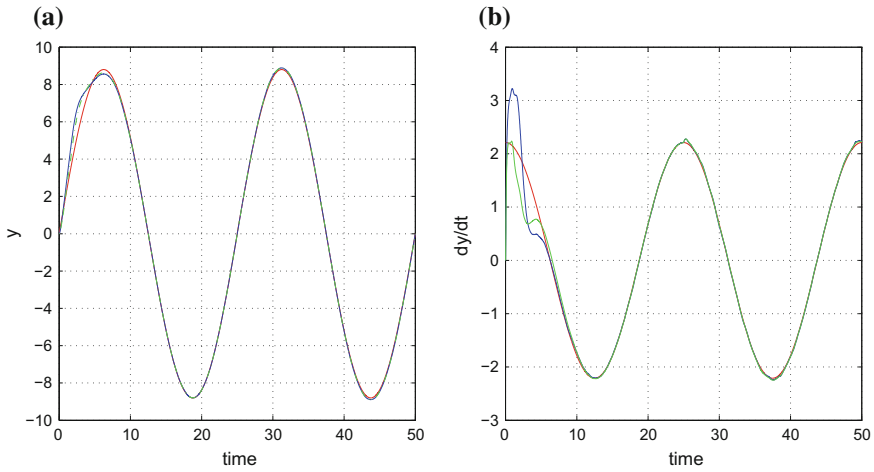
**Fig. 10.7** **a** Tracking of a circular trajectory: KF-based estimation of the disturbance along the  $x$ -axis (blue line) and real value of the  $x$ -axis disturbance (red line), **b** KF-based estimation of the disturbance along the  $y$ -axis (blue line) and real value of the  $y$ -axis disturbance (red line)



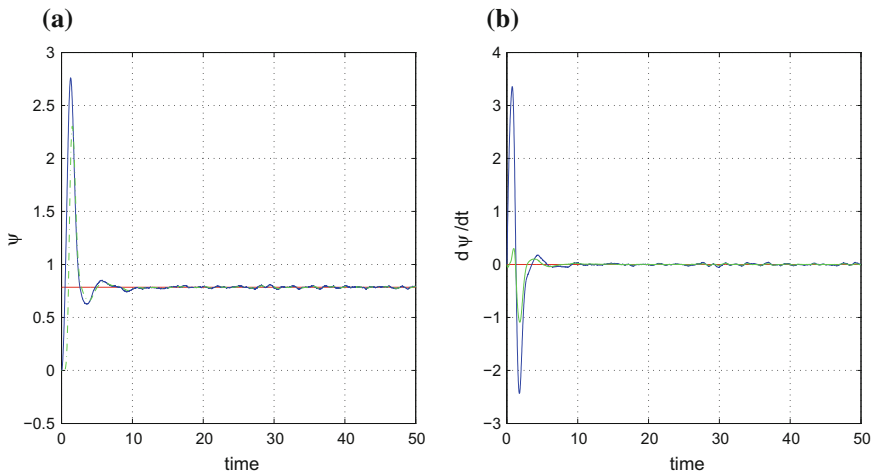
**Fig. 10.8** Tracking of a circular trajectory: **a** KF-based estimation of the disturbance torque round the  $z$ -axis (blue line) and real value of the  $z$ -axis disturbance torque (red line), **b** Trajectory of the ship on the  $xy$ -plane (green line) and desirable ship trajectory (red line) in the case of KF-based state estimation



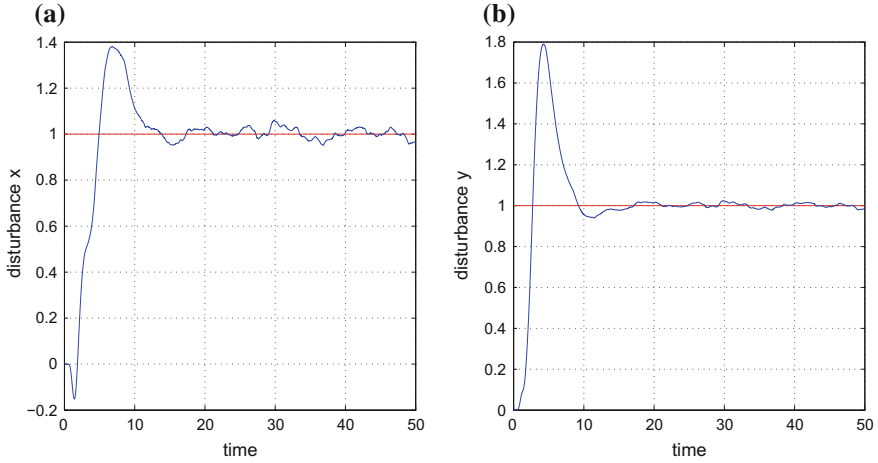
**Fig. 10.9** Tracking of an eight-shaped trajectory: **a** KF-based estimation of the ship's position along the  $x$ -axis (green line) and desirable  $x$ -axis position (red line), **b** KF-based estimation of the ship's velocity along the  $x$ -axis (green line) and desirable  $x$ -axis velocity (red line)



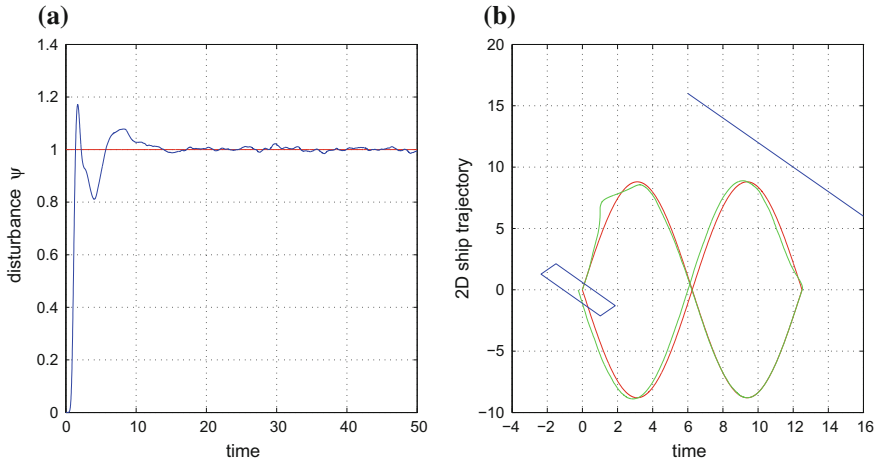
**Fig. 10.10** Tracking of an eight-shaped trajectory: **a** KF-based estimation of the ship's position along the  $y$ -axis (green line) and desirable  $y$ -axis position (red line), **b** KF-based estimation of the ship's velocity along the  $y$ -axis (green line) and desirable  $y$ -axis velocity (red line)



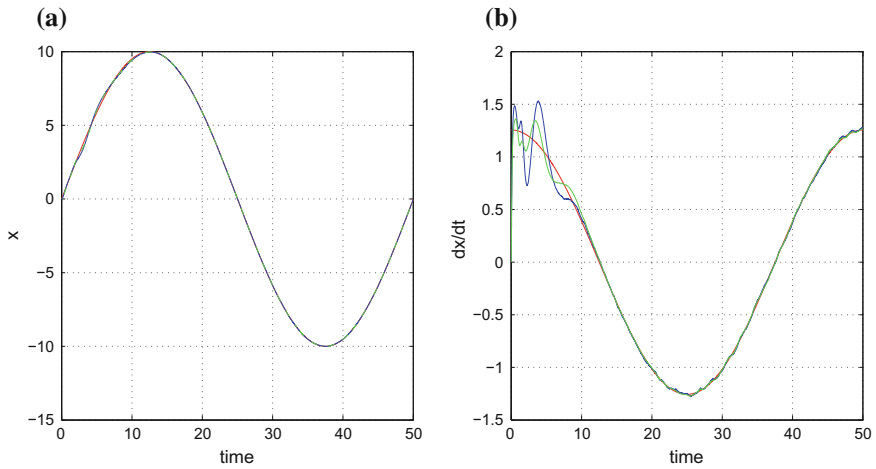
**Fig. 10.11** Tracking of an eight-shaped trajectory: **a** KF-based estimation of the ship's angle round the  $z$ -axis (green line) and desirable  $z$ -axis rotation angle (red line), **b** KF-based estimation of the ship's angular velocity round the  $z$ -axis (green line) and desirable angular velocity (red line)



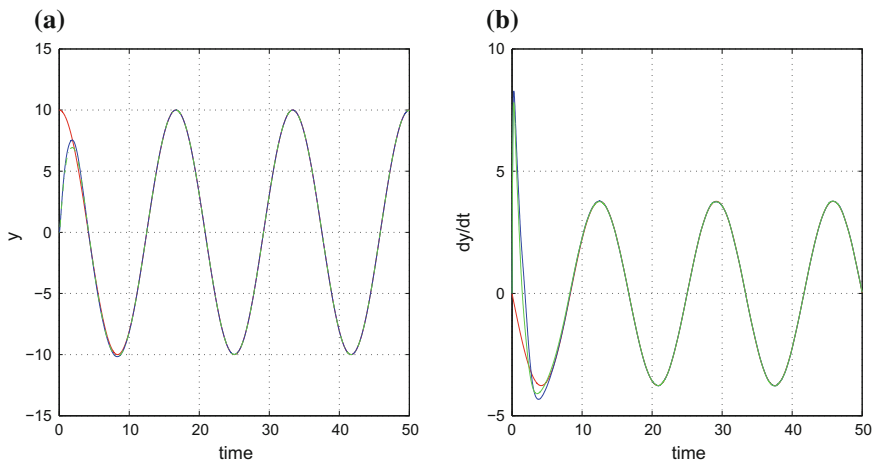
**Fig. 10.12** Tracking of an eight-shaped trajectory: **a** KF-based estimation of the disturbance along the  $x$ -axis (blue line) and real value of the  $x$ -axis disturbance (red line), **b** KF-based estimation of the disturbance along the  $y$ -axis (blue line) and real value of the  $y$ -axis disturbance (red line)



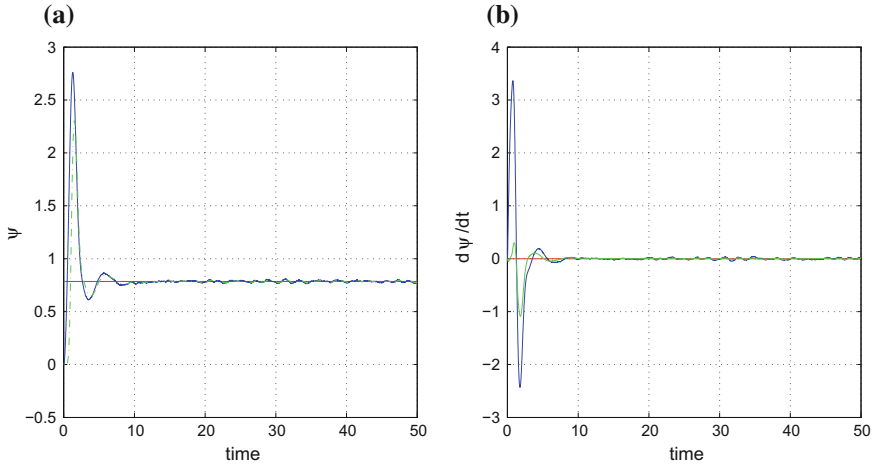
**Fig. 10.13** Tracking of an eight-shaped trajectory: **a** KF-based estimation of the disturbance torque round the  $z$ -axis (blue line) and real value of the  $z$ -axis disturbance torque (red line), **b** Trajectory of the ship on the  $xy$ -plane (green line) and desirable ship trajectory (red line) in the case of KF-based state estimation



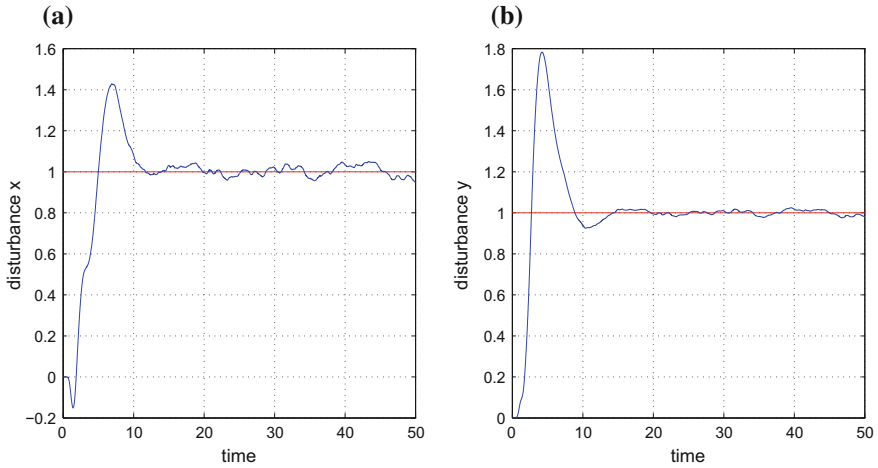
**Fig. 10.14** Tracking of a complex-curved trajectory: **a** KF-based estimation of the ship’s position along the  $x$ -axis (green line) and desirable  $x$ -axis position (red line), **b** KF-based estimation of the ship’s velocity along the  $x$ -axis (green line) and desirable  $x$ -axis velocity (red line)



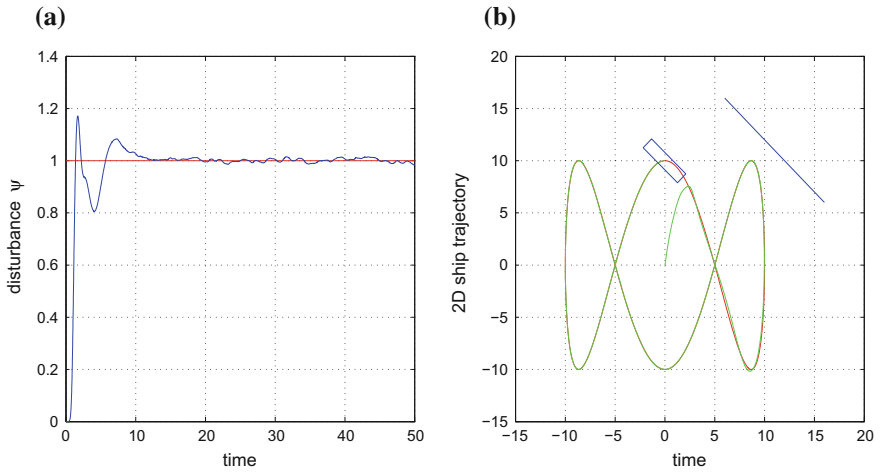
**Fig. 10.15** Tracking of a complex-curved trajectory: **a** KF-based estimation of the ship’s position along the  $y$ -axis (green line) and desirable  $y$ -axis position (red line), **b** KF-based estimation of the ship’s velocity along the  $y$ -axis (green line) and desirable  $y$ -axis velocity (red line)



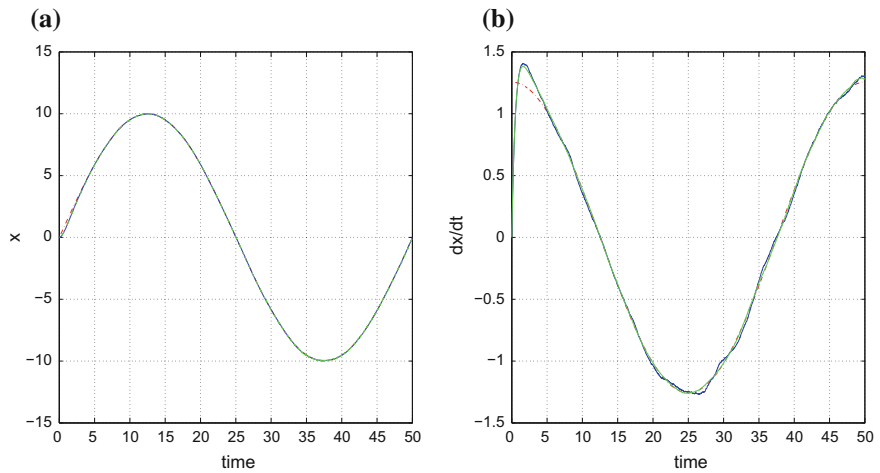
**Fig. 10.16** Tracking of a complex-curved trajectory: **a** KF-based estimation of the ship's angle round the  $z$ -axis (green line) and desirable  $z$ -axis rotation angle (red line), **b** KF-based estimation of the ship's angular velocity round the  $z$ -axis (green line) and desirable angular velocity (red line)



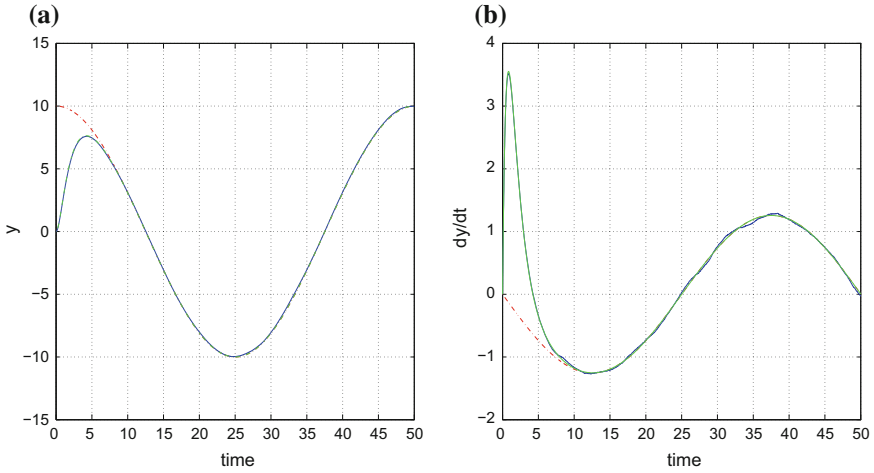
**Fig. 10.17** Tracking of a complex-curved trajectory: **a** KF-based estimation of the disturbance along the  $x$ -axis (blue line) and real value of the  $x$ -axis disturbance (red line), **b** KF-based estimation of the disturbance along the  $y$ -axis (blue line) and real value of the  $y$ -axis disturbance (red line)



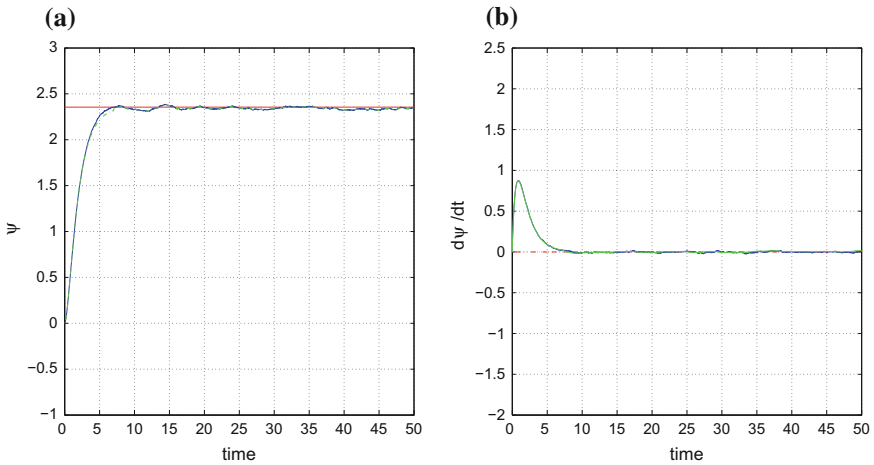
**Fig. 10.18** Tracking of a complex-curved trajectory: **a** KF-based estimation of the disturbance torque round the  $z$ -axis (blue line) and real value of the  $z$ -axis disturbance torque (red line), **b** Trajectory of the ship on the  $xy$ -plane (green line) and desirable ship trajectory (red line) in the case of KF-based state estimation



**Fig. 10.19** Tracking of a circular trajectory: **a** PF-based estimation of the ship's position along the  $x$ -axis (green line) and desirable  $x$ -axis position (red line), **b** PF-based estimation of the ship's velocity along the  $x$ -axis (green line) and desirable  $x$ -axis velocity (red line)

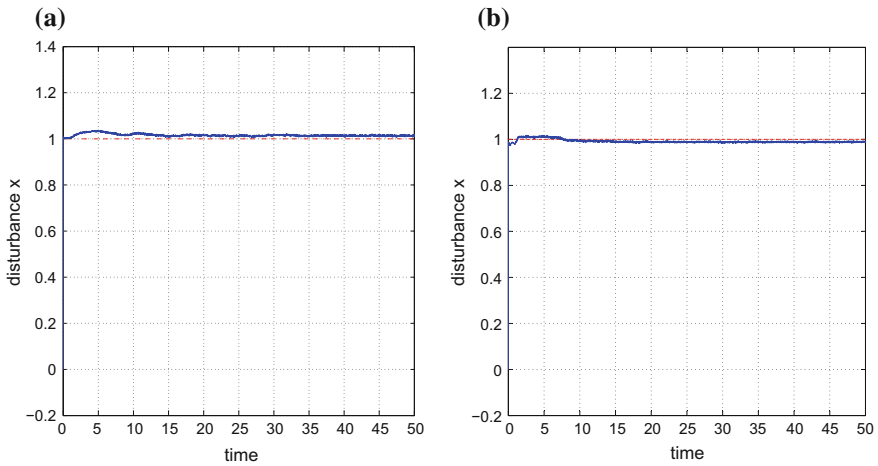


**Fig. 10.20** Tracking of a circular trajectory: **a** PF-based estimation of the ship’s position along the y-axis (green line) and desirable y-axis position (red line), **b** PF-based estimation of the ship’s velocity along the y-axis (green line) and desirable y-axis velocity (red line)

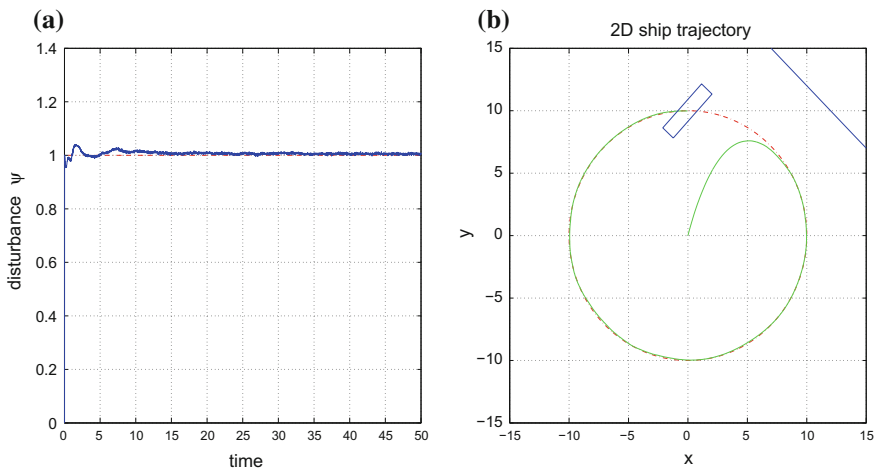


**Fig. 10.21** Tracking of a circular trajectory: **a** PF-based estimation of the ship’s angle round the z-axis (green line) and desirable z-axis rotation angle (red line), **b** PF-based estimation of the ship’s angular velocity round the z-axis (green line) and desirable angular velocity (red line)

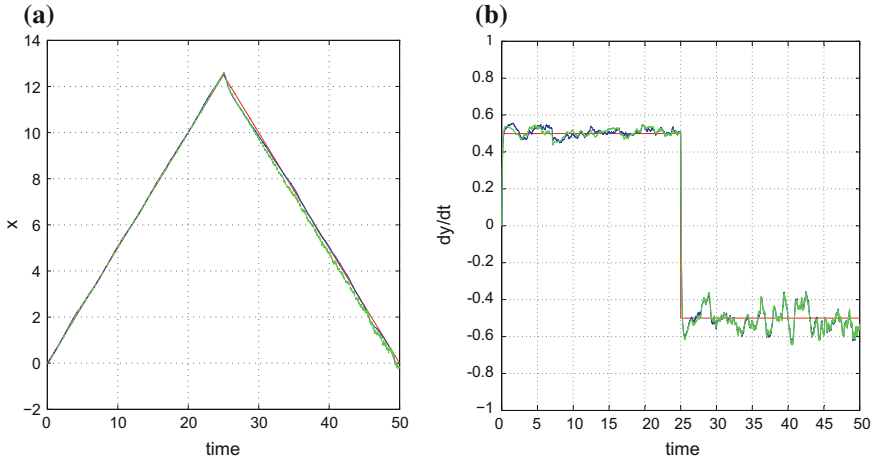




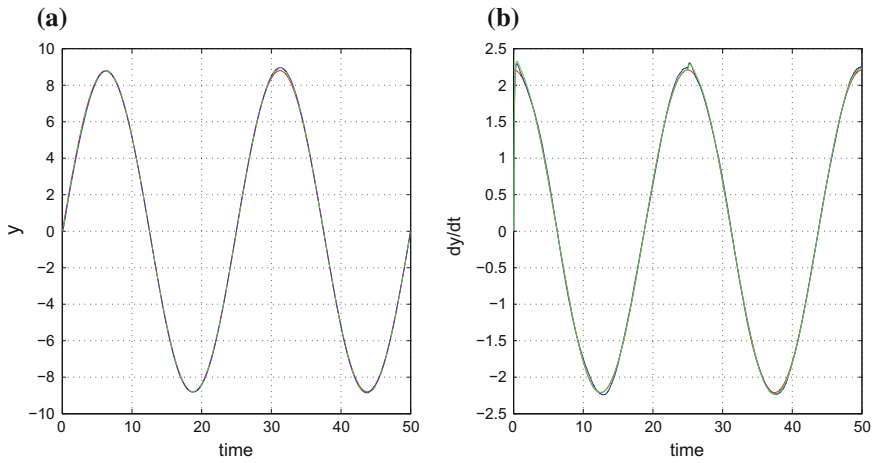
**Fig. 10.22** Tracking of a circular trajectory: **a** PF-based estimation of the disturbance along the  $x$ -axis (blue line) and real value of the  $x$ -axis disturbance (red line), **b** PF-based estimation of the disturbance along the  $y$ -axis (blue line) and real value of the  $y$ -axis disturbance (red line)



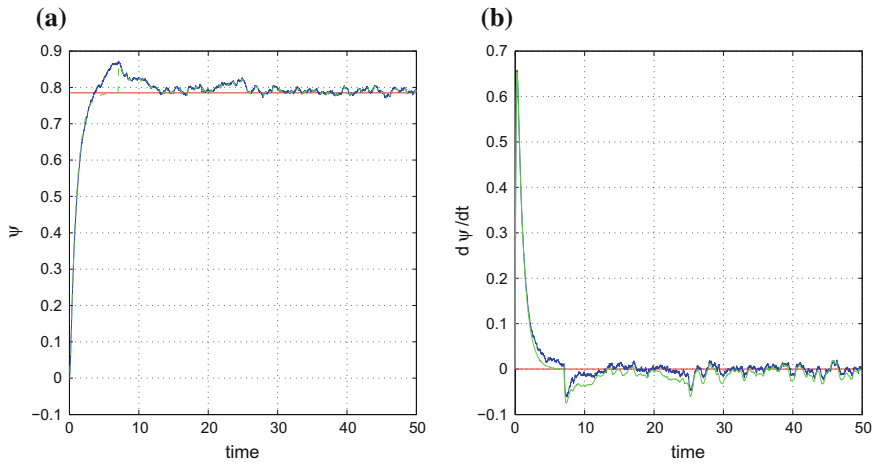
**Fig. 10.23** Tracking of a circular trajectory: **a** PF-based estimation of the disturbance torque round the  $z$ -axis (blue line) and real value of the  $z$ -axis disturbance torque (red line), **b** Trajectory of the ship on the  $xy$ -plane (green line) and desirable ship trajectory (red line) in the case of PF-based state estimation



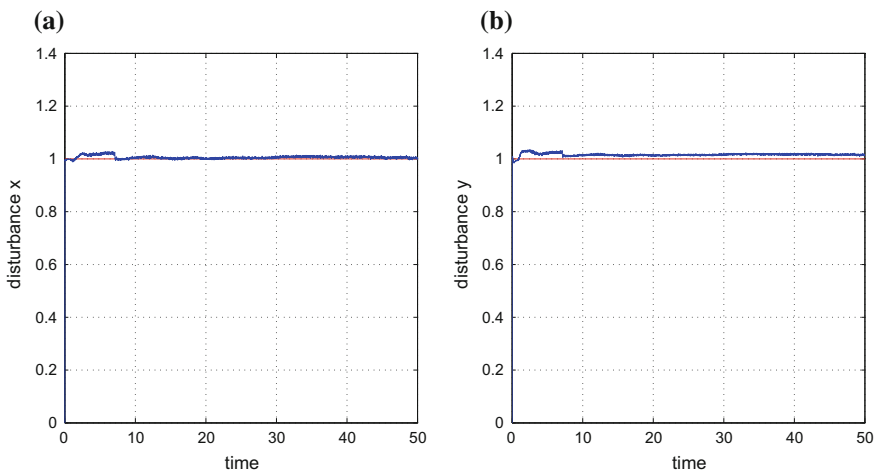
**Fig. 10.24** Tracking of an eight-shaped trajectory: **a** PF-based estimation of the ship’s position along the  $x$ -axis (green line) and desirable  $x$ -axis position (blue line), **b** PF-based estimation of the ship’s velocity along the  $x$ -axis (green line) and desirable  $x$ -axis velocity (red line)



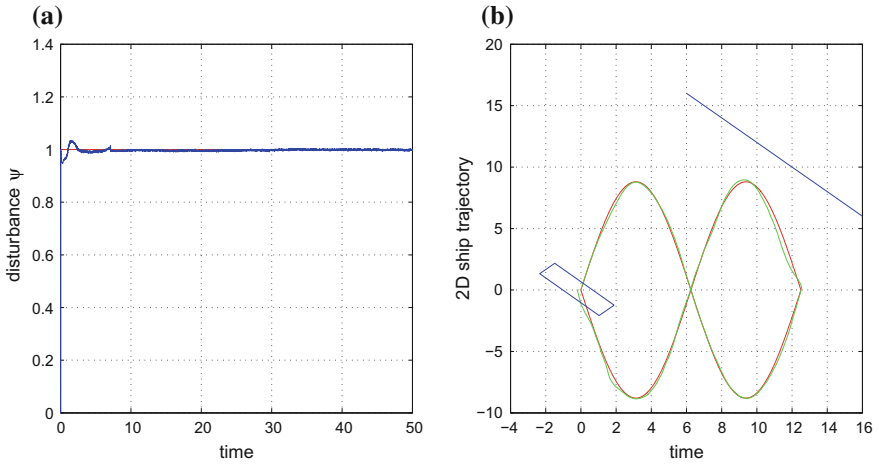
**Fig. 10.25** Tracking of an eight-shaped trajectory: **a** PF-based estimation of the ship’s position along the  $y$ -axis (green line) and desirable  $y$ -axis position (red line), **b** PF-based estimation of the ship’s velocity along the  $y$ -axis (green line) and desirable  $y$ -axis velocity (red line)



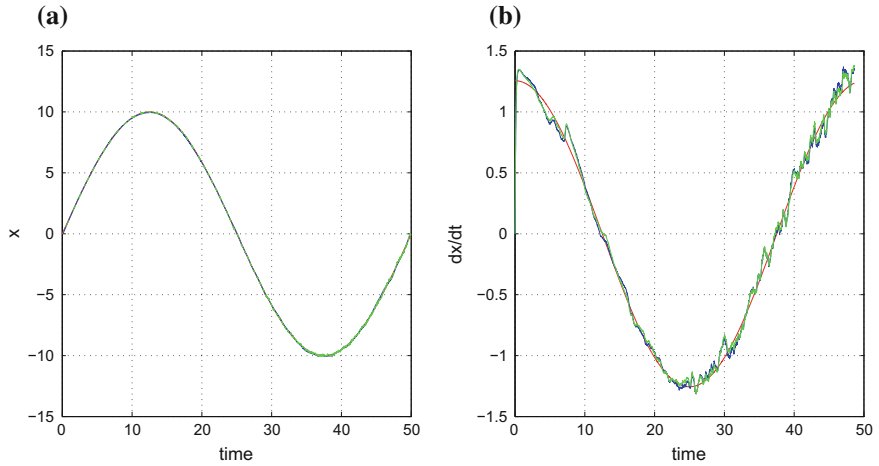
**Fig. 10.26** Tracking of an eight-shaped trajectory: **a** PF-based estimation of the ship’s angle round the  $z$ -axis (blue line) and desirable  $z$ -axis rotation angle (red line), **b** PF-based estimation of the ship’s angular velocity round the  $z$ -axis (blue line) and desirable angular velocity (red line)



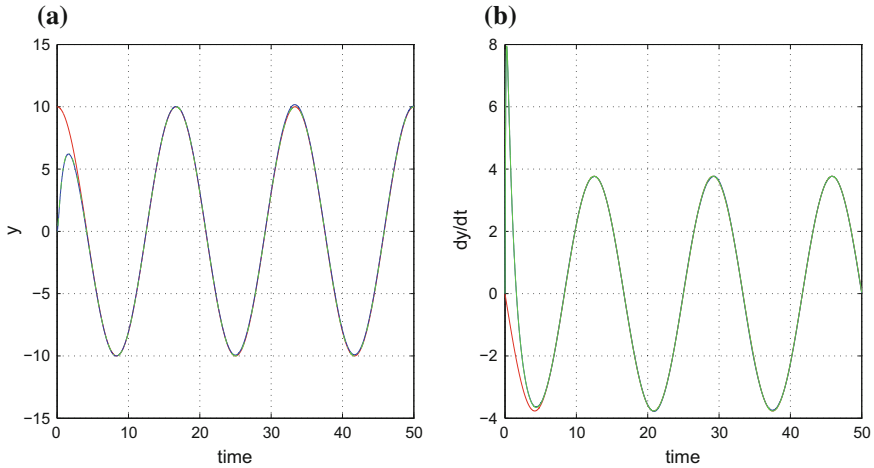
**Fig. 10.27** Tracking of an eight-shaped trajectory: **a** PF-based estimation of the disturbance along the  $x$ -axis (green line) and real value of the  $x$ -axis disturbance (red line), **b** PF-based estimation of the disturbance along the  $y$ -axis (green line) and real value of the  $y$ -axis disturbance (red line)



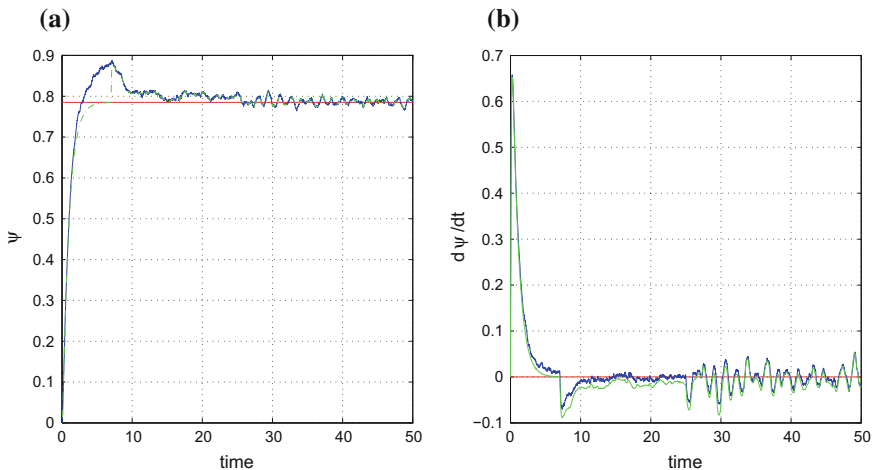
**Fig. 10.28** Tracking of an eight-shaped trajectory: **a** PF-based estimation of the disturbance torque round the  $z$ -axis (blue line) and real value of the  $z$ -axis disturbance torque (red line), **b** Trajectory of the ship on the  $xy$ -plane (green line) and desirable ship trajectory (red line) in the case of PF-based state estimation



**Fig. 10.29** Tracking of a complex-curved trajectory: **a** PF-based estimation of the ship's position along the  $x$ -axis (green line) and desirable  $x$ -axis position (red line), **b** PF-based estimation of the ship's velocity along the  $x$ -axis (green line) and desirable  $x$ -axis velocity (red line)



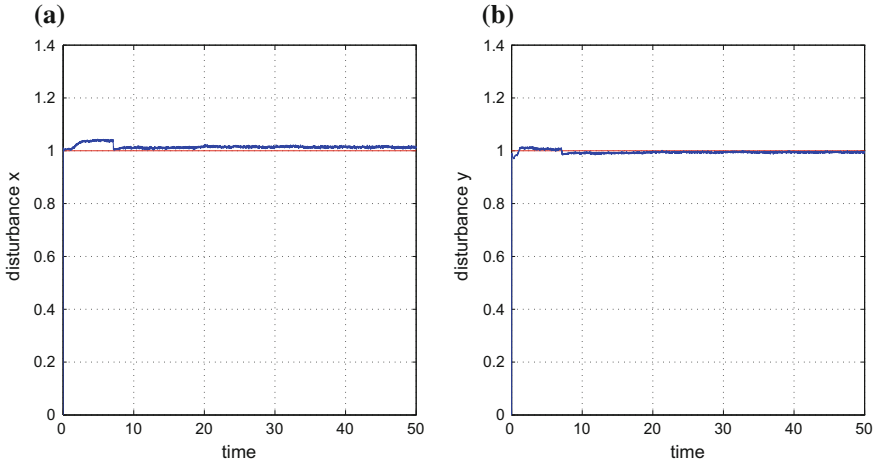
**Fig. 10.30** Tracking of a complex-curved trajectory: **a** PF-based estimation of the ship’s position along the  $y$ -axis (green line) and desirable  $y$ -axis position (red line), **b** PF-based estimation of the ship’s velocity along the  $y$ -axis (green line) and desirable  $y$ -axis velocity (red line)



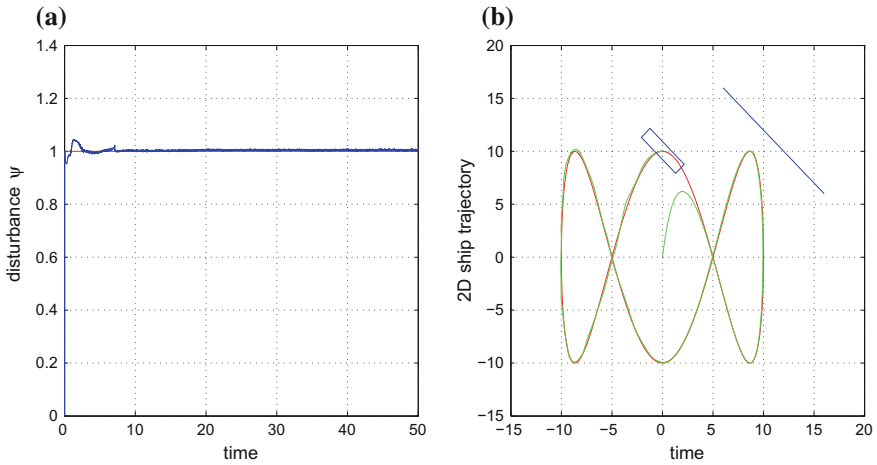
**Fig. 10.31** Tracking of a complex-curved trajectory: **a** PF-based estimation of the ship’s angle round the  $z$ -axis (green line) and desirable  $z$ -axis rotation angle (red line), **b** PF-based estimation of the ship’s angular velocity round the  $z$ -axis (green line) and desirable angular velocity (red line)

### 10.2.8.2 Evaluation of the Particle Filter-Based State Estimation and Control

From the simulation experiments it can be also noticed that the Particle Filter performs equally well to the Kalman Filter and that it provides accurate estimates of the vessel’s



**Fig. 10.32** Tracking of a complex-curved trajectory: **a** PF-based estimation of the disturbance along the  $x$ -axis (blue line) and real value of the  $x$ -axis disturbance (red line), **b** PF-based estimation of the disturbance along the  $y$ -axis (blue line) and real value of the  $y$ -axis disturbance (red line)



**Fig. 10.33** Tracking of a complex-curved trajectory: **a** PF-based estimation of the disturbance torque round the  $z$ -axis (blue line) and real value of the  $z$ -axis disturbance torque (red line), **b** Trajectory of the ship on the  $xy$ -plane (green line) and desirable ship trajectory (red line) in the case of PF-based state estimation

**Table 10.1** Variance using KF and PF for  $N = 1000$ 

State variable	$x$	$y$	$\psi$
KF	$5.24 \cdot 10^{-2}$	$8.52 \cdot 10^{-2}$	$1.94 \cdot 10^{-1}$
PF	$7.03 \cdot 10^{-2}$	$6.80 \cdot 10^{-2}$	$7.47 \cdot 10^{-3}$

**Table 10.2** Particle number, simulation time and variance of  $\hat{\psi}$ 

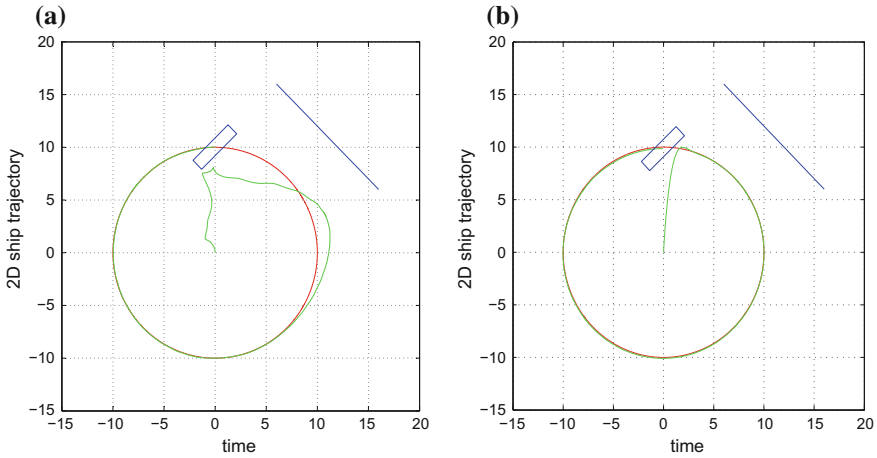
No particles	800	1000	1300	1600	1900	2200
PF cycle time (s)	0.355	0.463	0.577	0.775	0.975	1.097
Variance	$2.733 \cdot 10^{-1}$	$7.473 \cdot 10^{-3}$	$1.153 \cdot 10^{-2}$	$1.647 \cdot 10^{-3}$	$8.905 \cdot 10^{-2}$	$7.817 \cdot 10^{-3}$

state vector. Besides, the Particle Filter is a nonparametric estimator, therefore its application is not constrained by the assumption about Gaussian measurements noise made in the case of the Kalman Filter. Table 10.1 presents results on the variance of the state vector estimates, when considering equal noise levels for the EKF and the PF simulation, and assuming that the number of particles used by the PF was  $N = 1000$ . The measure used for evaluating the accuracy of the estimation performed by the nonlinear filters, as well as the accuracy of tracking of the state estimation-based control loops was the Root Mean Square Error (RMSE). Alternatively, the Cramer–Rao Lower Bound (CRLB) could have been considered [129, 181, 639].

The cycle time (runtime) of the Particle Filter with respect to the number of particles, using the Matlab platform on a PC with a 2GHz Intel Core Duo processor, is depicted in Table 10.2. Optimization of code of the resampling procedure can improve to some extent the speed of the algorithm. When it is necessary to use more particles, improved hardware and parallel processing available to embedded systems enable real-time implementation of the PF algorithm [53, 340, 611].

When sorting of particles is not performed in the resampling procedure the runtime of Particle Filtering increases linearly with respect to the number of particles [493].

Finally, the accuracy of tracking of the previously described reference trajectories was examined under progressively increasing disturbance magnitude. It was observed that tracking of these trajectories was possible even when the magnitude of the disturbance became several times larger than the initial disturbance of Figs. 10.7, 10.8, 10.9, 10.10, 10.11, 10.12, 10.13, 10.14, 10.15, 10.16, 10.17, 10.18, 10.19, 10.20, 10.21, 10.22, 10.23, 10.24, 10.25, 10.26, 10.27, 10.28, 10.29, 10.30, 10.31, 10.32 and 10.33. Indicative results for disturbance  $d_i = 3$ ,  $i = 1, \dots, 3$  are presented in Fig. 10.34.



**Fig. 10.34** Tracking of a circular trajectory under raised additive disturbances: **a** Trajectory of the ship on the  $xy$ -plane (continuous line) and desirable ship trajectory (red line) in the case of KF-based state estimation, **b** Trajectory of the ship on the  $xy$ -plane (red line) and desirable ship trajectory (dashed line) in the case of PF-based state estimation

## 10.3 Flatness-Based Control for the Autonomous Hovercraft

### 10.3.1 Outline

Autonomous navigation of unmanned surface vessels (USVs) (such as hovercrafts), is a significant topic, since it can find use in both security purposes and passenger transportation [345, 452, 517, 518, 545]. The problem of control and trajectory tracking for unmanned surface vessels of the hovercraft type is non-trivial because the associated kinematic model is a complex nonlinear one [20, 128, 179, 522, 566]. A first problem that arises in controller design for unmanned vessels is that trajectory tracking has to be achieved despite modelling uncertainty and external perturbations and thus the control loop must exhibit sufficient robustness [297, 498]. Another problem that has to be dealt with is that the vessels' model can be underactuated [36, 47, 99, 127, 167, 187, 375, 498, 505, 505, 526, 627]. Indicative results on control of underactuated dynamical systems can be found in [416].

As previously noted, the problem of autonomous navigation of unmanned surface vessels has received particular attention, since it can find use in both security purposes and passenger transportation [2, 111, 452, 517, 518, 545]. In particular, the problem of control and trajectory tracking for unmanned surface vessels of the hovercraft type is non-trivial because the associated kinematic model is a complex nonlinear one [20, 128, 179, 522, 566]. Another problem that has to be dealt with is that the hovercraft's model is underactuated [36, 47, 187, 375, 505, 526]. Indicative results on control of underactuated dynamical systems can be found in [402, 404, 427].



Moreover, the hovercraft's model cannot be subjected to undergo static feedback linearization, but admits only dynamic feedback linearization. This means that to achieve linearization, the state-space description of the system has to be augmented by considering as additional state variables the control inputs and their derivatives. Thus, finally the control input that is applied to the vessel contains integral terms of the tracking error. The present section proposes a solution to the control problem of hovercrafts with the use of differential flatness theory and of a nonlinear filtering method, the so-called Derivative-free nonlinear Kalman Filter.

First it is shown that the hovercraft's model is differentially flat. This means that all its state variables and the control inputs can be written as differential functions of one single algebraic variable which is the flat output [57, 145, 254, 267, 322, 450, 472, 476, 519]. By exploiting the differential flatness properties it is shown that the system can be transformed into the linear canonical form, through dynamic feedback linearization. To achieve this, dynamic extension is performed which means that the state-vector's dimension is increased by considering as additional state variables certain control inputs and their derivatives. For the linearized equivalent model of the system the design of a state feedback controller is possible, through the use of pole placement techniques. Next, to estimate the nonmeasurable state variables of the surface vessel and to identify additive disturbance terms that affect the system, the Derivative-free nonlinear Kalman Filter is redesigned as a disturbance observer [33, 421, 431, 438, 443, 457, 463]. This estimation algorithm consists of the standard Kalman Filter recursion applied on the linearized equivalent model of the surface vessel and of an inverse transformation that makes use of differential flatness theory, which permits to compute estimates of the state variables of the initial nonlinear system.

Comparing to approximate linearization methods [79, 99, 205, 564], nonlinear feedback control approaches which are based on exact feedback linearization of the vessel's model, are assessed as follows: (i) they avoid cumulative numerical errors which are due to the approximate linearization of the system dynamics coming from the application of Taylor series expansion, (ii) the generated control input compensates exclusively for the effects of external perturbations whereas in approximate linearization methods the control input has to compensate both for internal modelling errors and for exogenous disturbances, (iii) they require a smaller number of real-time computations for generating the control inputs, because unlike the approximate linearization methods a large part of the controller's design (e.g. computation of the linearized equivalent model of the system) is performed out of the loop.

## ***10.3.2 State-Space Description of the Underactuated Hovercraft***

### **10.3.2.1 State-Space Equation of the Underactuated Hovercraft**

The hovercraft's dynamic and kinematic model stems from the generic ship's model, after setting specific values for the elements of the inertia and Coriolis matrix and

after reducing the number of the available control inputs [452, 517, 518]. The state-space equation of the nonlinear underactuated hovercraft model (Fig. 10.35) is given by

$$\begin{aligned}
 \dot{x} &= u\cos(\psi) - v\sin(\psi) \\
 \dot{y} &= u\sin(\psi) + v\cos(\psi) \\
 \dot{\psi} &= r \\
 \dot{u} &= v \cdot r + \tau_u \\
 \dot{v} &= -u \cdot r - \beta v \\
 \dot{r} &= \tau_r
 \end{aligned} \tag{10.39}$$

where  $x$  and  $y$  are the cartesian coordinates of the vessel,  $\psi$  is the orientation angle,  $u$  is the surge velocity,  $v$  is the sway velocity and  $r$  is the yaw rate. Coefficient  $\beta$  is a function of elements of the inertia matrix and hydrodynamic damping matrix of the vessel. The control inputs are the surge force  $\tau_u$  and the yaw torque  $\tau_r$ . The hovercraft's model is also written in the matrix form:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} u\cos(\psi) - v\sin(\psi) \\ u\sin(\psi) + v\cos(\psi) \\ r \\ vr \\ -ur - \beta v \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \tau_u \\ \tau_r \end{pmatrix} \tag{10.40}$$

or equivalently, one has the description

$$\dot{\tilde{x}} = \tilde{f}(\tilde{x}) + \tilde{g}(\tilde{x})\tilde{v} \tag{10.41}$$

The system's state vector is denoted as  $\tilde{x} = [x, y, \psi, u, v, r]^T$ ,  $f(\tilde{x}) \in R^{6 \times 1}$ , and  $\tilde{g}(\tilde{x}) = [\tilde{g}_a, \tilde{g}_b] \in R^{6 \times 2}$ , while the control input is the vector  $\tilde{v} = [\tau_u, \tau_r]^T$ .

The system's state vector can be extended by including as additional state variables the control input  $\tau_u$  and its first derivative  $\dot{\tau}_u$ . These are denoted as  $z_1 = \tau_u$  and  $z_2 = \dot{\tau}_u$ . The extended state-space description of the hovercraft becomes

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{r} \\ \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} = \begin{pmatrix} u\cos(\psi) - v\sin(\psi) \\ u\sin(\psi) + v\cos(\psi) \\ r \\ vr + z_1 \\ -ur - \beta v \\ 0 \\ z_2 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \dot{\tau}_u \\ \tau_r \end{pmatrix} \tag{10.42}$$

or equivalently, one has the description

$$\dot{z} = f(z) + g(z)\tilde{v} \tag{10.43}$$

The extended system’s state vector is denoted as  $z = [x, y, \psi, u, v, r, z_1, z_2]^T$ . Moreover, one has  $f(z) \in \mathbb{R}^{8 \times 1}$ , and  $g(z) = [g_a, g_b] \in \mathbb{R}^{8 \times 2}$ , while the control input is the vector  $\tilde{v} = [\tilde{\tau}_u, \tilde{\tau}_r]^T$ .

From the previous state-space description it can be noticed that the hovercraft’s model is an underactuated one. Underactuation in the considered hovercraft’s model means that, to control the vehicle’s motion and orientation, one has fewer control inputs than the degrees of freedom. Thus considering the capability of the vessel to move on the  $xy$  plane and also to rotate round its  $z$  axis by a yaw angle that is denoted by  $\psi$  one has three degrees of freedom. On the other hand the vessel is supplied with only two control inputs which are the surge force  $\tau_u$  and the yaw torque  $\tau_r$ . In practice, the hovercraft is supplied with a number of fans that inject air mass backwards and which are installed at its rear part, while being also symmetrically placed with respect to the vessel’s longitudinal axis (Fig. 10.35). Without loss of generality one can consider the propulsion scheme of Fig. 10.35. The aggregate surge force  $\tau_u$  is the sum of the two propulsion forces  $F_L$  and  $F_R$ , which are generated by the left and right fan respectively. The yaw torque  $\tau_r$  is generated by altering the force produced by the two fans according to the sign of the desirable turn angle. Thus, to turn left  $F_L$  is set to be smaller than  $F_R$ . On the other hand, to turn right  $F_R$  is set to be smaller than  $F_L$ . In this manner the steering of the vessel (turn by a specific yaw

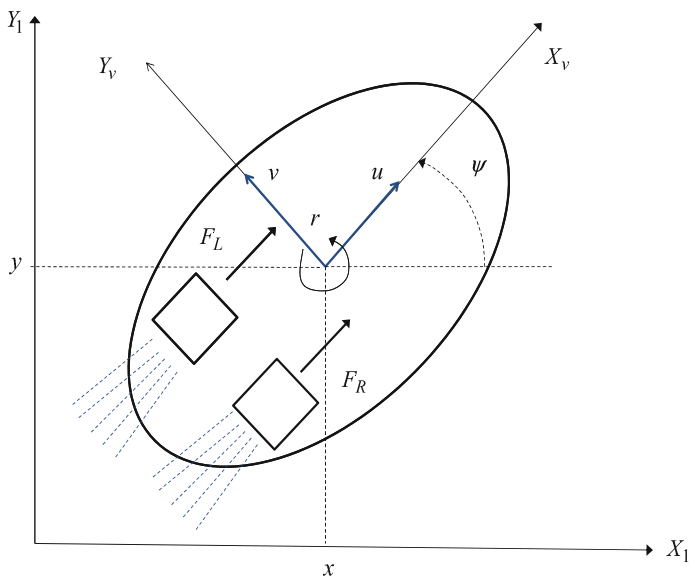


Fig. 10.35 Diagram of the underactuated hovercraft’s kinematic model

angle) is achieved without the use of a rudder. Succeeding control of the hovercraft's motion and orientation in case of underactuation is very important, for assuring the efficient navigation of the vessel.

### 10.3.2.2 Lie Algebra-Based Control of the Underactuated Hovercraft

It will be shown that Lie algebra-based control through the computation of Lie derivatives is an approach equivalent to differential flatness theory-based control for the model of the underactuated hovercraft. The state-space model of the hovercraft that was described in Sect. 10.3.2, and particularly Eq. (10.85), is considered again. The following linearizing outputs of the system are defined

$$z_{1,1} = x \quad z_{2,1} = y \quad (10.44)$$

Moreover, the new state variables are defined

$$\begin{aligned} z_{1,2} &= L_f z_{1,1} & z_{2,2} &= L_f z_{2,1} \\ z_{1,3} &= L_f^2 z_{1,1} & z_{2,3} &= L_f^2 z_{2,1} \\ z_{1,4} &= L_f^3 z_{1,1} & z_{2,4} &= L_f^3 z_{2,1} \end{aligned} \quad (10.45)$$

The system will be brought to a linearized input-output form using

$$\begin{aligned} \dot{z}_{1,4} &= L_f^4 z_{1,1} + L_{g_a} L_f^3 z_{1,1} \ddot{u} + L_{g_b} L_f^3 z_{1,1} \tau_r \\ \dot{z}_{2,4} &= L_f^4 z_{2,1} + L_{g_a} L_f^3 z_{2,1} \ddot{u} + L_{g_b} L_f^3 z_{2,1} \tau_r \end{aligned} \quad (10.46)$$

It holds that  $z_{1,1} = x$ . Thus one has

$$z_{1,2} = L_f z_{1,1} \Rightarrow z_{1,2} = \frac{\partial z_{1,1}}{\partial x} f_1 + \frac{\partial z_{1,1}}{\partial y} f_2 + \frac{\partial z_{1,1}}{\partial \psi} f_3 + \frac{\partial z_{1,1}}{\partial u} f_4 + \frac{\partial z_{1,1}}{\partial v} f_5 + \frac{\partial z_{1,1}}{\partial r} f_6 + \frac{\partial z_{1,1}}{\partial z_1} f_7 + \frac{\partial z_{1,1}}{\partial z_2} f_8 \Rightarrow z_{1,2} = 1 \cdot f_1 \Rightarrow z_{1,2} = u \cos(\psi) - v \sin(\psi).$$

Similarly, one obtains

$$\begin{aligned} z_{1,3} &= L_f^2 z_{1,1} \Rightarrow z_{1,3} = \frac{\partial z_{1,2}}{\partial x} f_1 + \frac{\partial z_{1,2}}{\partial y} f_2 + \frac{\partial z_{1,2}}{\partial \psi} f_3 + \frac{\partial z_{1,2}}{\partial u} f_4 + \frac{\partial z_{1,2}}{\partial v} f_5 + \frac{\partial z_{1,2}}{\partial r} f_6 + \\ &\frac{\partial z_{1,2}}{\partial z_1} f_7 + \frac{\partial z_{1,2}}{\partial z_2} f_8 \Rightarrow z_{1,3} = (-u \sin(\psi) - v \cos(\psi)) f_3 + \cos(\psi) f_4 - \sin(\psi) f_5 \Rightarrow z_{1,3} = \\ &(-u \sin(\psi) - v \cos(\psi)) r + \cos(\psi) (v r + z_1) - \sin(\psi) (-u r - \beta v) \Rightarrow z_{1,3} = \tau_u \cos(\psi) + \\ &\beta v \sin(\psi). \end{aligned}$$

Equivalently, it holds that

$$z_{1,4} = L_f^3 z_{1,1} \Rightarrow z_{1,4} = L_f z_{1,3} \Rightarrow z_{1,4} = \frac{\partial z_{1,3}}{\partial x} f_1 + \frac{\partial z_{1,3}}{\partial y} f_2 + \frac{\partial z_{1,3}}{\partial \psi} f_3 + \frac{\partial z_{1,3}}{\partial u} f_4 + \frac{\partial z_{1,3}}{\partial v} f_5 + \frac{\partial z_{1,3}}{\partial r} f_6 + \frac{\partial z_{1,3}}{\partial z_1} f_7 + \frac{\partial z_{1,3}}{\partial z_2} f_8,$$

while after intermediate operations one obtains

$$z_{1,4} = (-\tau_u \sin(\psi) + \beta v \cos(\psi)) f_3 + \beta \sin(\psi) f_5 + \cos(\psi) f_7 \Rightarrow z_{1,4} = (-\tau_u \sin(\psi) + \beta \cos(\psi)) r + \beta \sin(\psi) (-ur + \beta v) + \cos(\psi) z_2 \Rightarrow z_{1,4} = (\tau_u \sin(\psi) + \beta v \cos(\psi)) r + \beta \sin(\psi) (-ur - \beta v) + \cos(\psi) z_2$$

or, using the extended state vector variables notation one has

$$z_{1,4} = z_2 \cos(\psi) - z_1 \sin(\psi) r - \beta u r \sin(\psi) - \beta^2 v \sin(\psi) + \beta v \cos(\psi) r \quad (10.47)$$

It also holds that

$$\begin{aligned} \dot{z}_{1,4} &= L_f^4 z_{1,1} + L_{g_a} L_f^3 z_{1,1} \ddot{u} + L_{g_b} L_f^3 z_{1,1} \ddot{r} \Rightarrow \\ \dot{z}_{1,4} &= L_f z_{1,4} + L_{g_a} z_{1,4} \ddot{u} + L_{g_b} z_{1,4} \ddot{r} \end{aligned} \quad (10.48)$$

where

$$\begin{aligned} L_f z_{1,4} &= \frac{\partial z_{1,4}}{\partial x} f_1 + \frac{\partial z_{1,4}}{\partial y} f_2 + \frac{\partial z_{1,4}}{\partial \psi} f_3 + \frac{\partial z_{1,4}}{\partial u} f_4 + \\ &+ \frac{\partial z_{1,4}}{\partial v} f_5 + \frac{\partial z_{1,4}}{\partial r} f_6 + \frac{\partial z_{1,4}}{\partial z_1} f_7 + \frac{\partial z_{1,4}}{\partial z_2} f_8 \end{aligned} \quad (10.49)$$

which gives

$$\begin{aligned} L_f z_{1,4} &= (-z_2 \sin(\psi) - z_1 \cos(\psi) r - \beta u r \cos(\psi) - \beta^2 v \cos(\psi) - \beta v \sin(\psi) r) r + \\ &(-\beta r \sin(\psi))(vr + z_1) + (-\beta^2 \sin(\psi) + \beta \cos(\psi) r)(-ur + \beta v) + (-z_1 \sin(\psi) - \\ &\beta v \sin(\psi) + \beta v \cos(\psi)) 0 + (-\sin(\psi) r) z_2 \end{aligned}$$

while after some intermediate computations one obtains

$$\begin{aligned} L_f z_{1,4} &= -2z_2 \sin(\psi) r - z_1 \cos(\psi) r^2 - \\ &-\beta v r^2 \sin(\psi) - \beta z_1 r \sin(\psi) - \\ &-\beta u r^2 \cos(\psi) + \beta^2 u r \sin(\psi) - \\ &-\beta^3 v \sin(\psi) - \beta^2 v r \cos(\psi) - \beta u r^2 \cos^2(\psi) + \beta^2 v r \cos(\psi) \\ &-\beta v r^2 \sin(\psi) \end{aligned} \quad (10.50)$$

In a similar manner one computes

$$\begin{aligned} L_{g_a} z_{1,4} &= \frac{\partial z_{1,4}}{\partial x} g_{a1} + \frac{\partial z_{1,4}}{\partial y} g_{a2} + \frac{\partial z_{1,4}}{\partial \psi} g_{a3} + \frac{\partial z_{1,4}}{\partial u} g_{a4} + \frac{\partial z_{1,4}}{\partial v} g_{a5} + \frac{\partial z_{1,4}}{\partial r} g_{a6} + \frac{\partial z_{1,4}}{\partial z_1} g_{a7} + \\ \frac{\partial z_{1,4}}{\partial z_2} g_{a8} L_{g_a} z_{1,4} &= \frac{\partial z_{1,4}}{\partial z_2} \Rightarrow L_{g_a} z_{1,4} = \cos(\psi) \end{aligned}$$

and also

$$\begin{aligned} L_{g_b} z_{1,4} &= \frac{\partial z_{1,4}}{\partial x} g_{b1} + \frac{\partial z_{1,4}}{\partial y} g_{b2} + \frac{\partial z_{1,4}}{\partial \psi} g_{b3} + \frac{\partial z_{1,4}}{\partial u} g_{b4} + \frac{\partial z_{1,4}}{\partial v} g_{b5} + \frac{\partial z_{1,4}}{\partial r} g_{b6} + \frac{\partial z_{1,4}}{\partial z_1} g_{b7} + \\ \frac{\partial z_{1,4}}{\partial z_2} g_{b8} L_{g_b} z_{1,4} &= \frac{\partial z_{1,4}}{\partial r} \Rightarrow L_{g_b} z_{1,4} = -z_1 \sin(\psi) - \beta u \sin(\psi) + \beta v \cos(\psi) \end{aligned}$$

In an equivalent way, and using that  $z_{2,1} = y_2 = y$  one can compute

$$z_{2,2} = L_f z_{2,1} \Rightarrow z_{2,2} = \frac{\partial z_{2,1}}{\partial x} f_1 + \frac{\partial z_{2,1}}{\partial y} f_2 + \frac{\partial z_{2,1}}{\partial \psi} f_3 + \frac{\partial z_{2,1}}{\partial u} f_4 + \frac{\partial z_{2,1}}{\partial v} f_5 + \frac{\partial z_{2,1}}{\partial r} f_6 + \frac{\partial z_{2,1}}{\partial z_1} f_7 + \frac{\partial z_{2,1}}{\partial z_2} f_8 \Rightarrow z_{2,2} = 1 \cdot f_2 \Rightarrow z_{2,2} = \sin(\psi) + v \cos(\psi)$$

Equivalently, one has

$$z_{2,3} = L_f^2 z_{2,1} \Rightarrow z_{2,3} = L_f z_{2,2} \Rightarrow z_{2,3} = \frac{\partial z_{2,2}}{\partial x} f_1 + \frac{\partial z_{2,2}}{\partial y} f_2 + \frac{\partial z_{2,2}}{\partial \psi} f_3 + \frac{\partial z_{2,2}}{\partial u} f_4 + \frac{\partial z_{2,2}}{\partial v} f_5 + \frac{\partial z_{2,2}}{\partial r} f_6 + \frac{\partial z_{2,2}}{\partial z_1} f_7 + \frac{\partial z_{2,2}}{\partial z_2} f_8 \Rightarrow z_{2,3} = (u \cos(\psi) - v \sin(\psi))r + \sin(\psi)(vr + z_1) + \cos(\psi)(-ur - \beta v) \Rightarrow z_{2,3} = z_1 \sin(\psi) + \beta v \cos(\psi)$$

In an equivalent manner one obtains

$$z_{2,4} = L_f^3 z_{2,1} \Rightarrow z_{3,3} = L_f z_{2,3} \Rightarrow z_{2,4} = \frac{\partial z_{2,3}}{\partial x} f_1 + \frac{\partial z_{2,3}}{\partial y} f_2 + \frac{\partial z_{2,3}}{\partial \psi} f_3 + \frac{\partial z_{2,3}}{\partial u} f_4 + \frac{\partial z_{2,3}}{\partial v} f_5 + \frac{\partial z_{2,3}}{\partial r} f_6 + \frac{\partial z_{2,3}}{\partial z_1} f_7 + \frac{\partial z_{2,3}}{\partial z_2} f_8 \Rightarrow z_{2,4} = (\cos(\psi) - \beta v \sin(\psi))f_3 + \beta \cos(\psi)f_5 + \sin(\psi)f_7 \Rightarrow z_{2,4} = (z_1 \cos(\psi) - \beta v \sin(\psi))r + (\beta \cos(\psi)(-ur - \beta v) + \sin(\psi)z_2) \Rightarrow z_{2,4} = z_1 \cos(\psi)r - \beta v r \sin(\psi) - \beta u r \cos(\psi) + \beta^2 v \cos(\psi) + z_2 \sin(\psi)$$

Moreover, it holds that

$$\dot{z}_{2,4} = L_f^4 z_{2,1} + L_{g_a} L_f^3 z_{2,1} \ddot{u} + L_{g_b} L_f^3 z_{2,1} \tau_r \quad (10.51)$$

where

$$L_f^4 z_{2,1} = L_f z_{2,4} \Rightarrow L_f^4 z_{2,1} = \frac{\partial z_{2,4}}{\partial x} f_1 + \frac{\partial z_{2,4}}{\partial y} f_2 + \frac{\partial z_{2,4}}{\partial \psi} f_3 + \frac{\partial z_{2,4}}{\partial u} f_4 + \frac{\partial z_{2,4}}{\partial v} f_5 + \frac{\partial z_{2,4}}{\partial r} f_6 + \frac{\partial z_{2,4}}{\partial z_1} f_7 + \frac{\partial z_{2,4}}{\partial z_2} f_8 \Rightarrow \quad (10.52)$$

which gives

$$L_f^4 z_{2,1} = [-z_1 \sin(\psi)r - \beta v r \cos(\psi) + \beta u r \sin(\psi) - \beta^2 v \sin(\psi) + z_2 \cos(\psi)]r + [-\beta r \cos(\psi)](vr + z_1) + [-\beta r \sin(\psi) + \beta^2 \cos(\psi)](-ur - \beta v) + [z_1 \cos(\psi) - \beta v \sin(\psi) - \beta u r \cos(\psi)]0 + [\cos(\psi)r]z_2 + [\sin(\psi)]0$$

and after additional computations one arrives at the form

$$L_f^4 z_{2,1} = -z_1 r^2 \sin(\psi) - \beta v r^2 \cos(\psi) + \beta u r^2 \sin(\psi) - \beta^2 v r \sin(\psi) + z_2 r \cos(\psi) - \beta v r^2 \cos(\psi) - \beta r z_1 \cos(\psi) + \beta u r^2 \sin(\psi) - \beta^2 r v \sin(\psi) - \beta^2 u r \cos(\psi) + \beta^2 v \cos(\psi) + z_2 r \cos(\psi)$$

Proceeding as before, one computes

$$L_{g_a} L_f^3 z_{2,1} = L_{g_a} z_{2,4} \Rightarrow L_{g_a} L_f^3 z_{2,1} = \frac{\partial z_{2,4}}{\partial x} g_{a1} + \frac{\partial z_{2,4}}{\partial y} g_{a2} + \frac{\partial z_{2,4}}{\partial \psi} g_{a3} + \frac{\partial z_{2,4}}{\partial u} g_{a4} + \frac{\partial z_{2,4}}{\partial v} g_{a5} + \frac{\partial z_{2,4}}{\partial r} g_{a6} + \frac{\partial z_{2,4}}{\partial z_1} g_{a7} + \frac{\partial z_{2,4}}{\partial z_2} g_{a8} \Rightarrow L_{g_a} L_f^3 z_{2,1} = \frac{\partial z_{2,4}}{\partial z_2} \Rightarrow L_{g_a} L_f^3 z_{2,1} = \sin(\psi)$$

Equivalently, one computes

$$L_{g_b} L_f^3 z_{2,1} = L_{g_b} z_{2,4} \Rightarrow L_{g_b} L_f^3 z_{2,1} = \frac{\partial z_{2,4}}{\partial x} g_{b1} + \frac{\partial z_{2,4}}{\partial y} g_{b2} + \frac{\partial z_{2,4}}{\partial \psi} g_{b3} + \frac{\partial z_{2,4}}{\partial u} g_{b4} + \frac{\partial z_{2,4}}{\partial v} g_{b5} + \frac{\partial z_{2,4}}{\partial r} g_{b6} + \frac{\partial z_{2,4}}{\partial z_1} g_{b7} + \frac{\partial z_{2,4}}{\partial z_2} g_{b8} \Rightarrow L_{g_b} L_f^3 z_{2,1} = \frac{\partial z_{2,4}}{\partial v} \Rightarrow L_{g_b} L_f^3 z_{2,1} = z_1 \cos(\psi) = \beta v \sin(\psi) - \beta u \cos(\psi)$$

The aggregate dynamics of the input-output linearized system is

$$\begin{aligned} x^{(4)} &= L_f^4 z_{1,1} + L_{g_a} L_f^3 z_{1,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{1,1} \tau_r \\ y^{(4)} &= L_f^4 z_{2,1} + L_{g_a} L_f^3 z_{2,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{2,1} \tau_r \end{aligned} \quad (10.53)$$

By defining the new control inputs

$$\begin{aligned} v_1 &= L_f^4 z_{1,1} + L_{g_a} L_f^3 z_{1,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{1,1} \tau_r \\ v_2 &= L_f^4 z_{2,1} + L_{g_a} L_f^3 z_{2,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{2,1} \tau_r \end{aligned} \quad (10.54)$$

one arrives at the following description for the input-output linearized system

$$\begin{aligned} x^{(4)} &= v_1 \\ y^{(4)} &= v_2 \end{aligned} \quad (10.55)$$

which can be also written in the state-space form

$$\begin{pmatrix} \dot{z}_{1,1} \\ \dot{z}_{1,2} \\ \dot{z}_{1,3} \\ \dot{z}_{1,4} \\ \dot{z}_{2,1} \\ \dot{z}_{2,2} \\ \dot{z}_{2,3} \\ \dot{z}_{2,4} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_{1,1} \\ z_{1,2} \\ z_{1,3} \\ z_{1,4} \\ z_{2,1} \\ z_{2,2} \\ z_{2,3} \\ z_{2,4} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (10.56)$$

while the associated measurement equation is

$$\begin{pmatrix} z_1^m \\ z_2^m \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_{1,1} \\ z_{1,2} \\ z_{1,3} \\ z_{1,4} \\ z_{2,1} \\ z_{2,2} \\ z_{2,3} \\ z_{2,4} \end{pmatrix} \quad (10.57)$$

A suitable feedback control law for the linearized system is

$$v_1 = x_d^{(4)} - k_1^1(x^{(3)} - x_d^{(3)}) - k_2^1(\ddot{x} - \ddot{x}_d) - k_3^1(\dot{x} - \dot{x}_d) - k_4^1(x - x_d), \text{ and } v_2 = y_d^{(4)} - k_1^2(y^{(3)} - y_d^{(3)}) - k_2^2(\ddot{y} - \ddot{y}_d) - k_3^2(\dot{y} - \dot{y}_d) - k_4^2(y - y_d).$$

One can also compute the control input that is finally applied to the hovercraft model. It holds that

$$\bar{v} = \tilde{f} + \tilde{M}\tilde{v} \quad (10.58)$$

where matrices and vectors  $\bar{v}$ ,  $\tilde{f}$ ,  $\tilde{M}$  and  $\tilde{v}$  are defined as

$$\begin{aligned} \bar{v} &= \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} & \tilde{f} &= \begin{pmatrix} L_f^4 z_{1,1} \\ L_f^4 z_{2,1} \end{pmatrix} \\ \tilde{M} &= \begin{pmatrix} L_{g,a} L_f^3 z_{1,1} & L_{g,b} L_f^3 z_{1,1} \\ L_{g,a} L_f^3 z_{2,1} & L_{g,b} L_f^3 z_{2,1} \end{pmatrix} & \tilde{v} &= \begin{pmatrix} \ddot{u} \\ \tau_r \end{pmatrix} \end{aligned} \quad (10.59)$$

The above equation can be solved with respect to  $\tilde{v}$ , which finally gives

$$\tilde{v} = \tilde{M}^{-1}(\bar{v} - \tilde{f}) \quad (10.60)$$

The vector  $\tilde{u}$  is the control input that is finally applied to the system, which finally means that the control signal contains integrals of the tracking error.

### 10.3.3 Differential Flatness Properties of the Hovercraft's Model

It can be proven that the model of the underactuated vessel (hovercraft) given in Eq. (10.39) is a differentially flat one. This means that all its state variables and the control inputs can be written as functions of a single variable, which is the flat output. In the hovercraft's case the flat output is the vector of the vessel's cartesian coordinates, that is

$$\tilde{y} = [y_1, y_2] = [x, y] \quad (10.61)$$

It holds that

$$\begin{aligned} \ddot{x} &= \dot{u}\cos(\psi) - u\sin(\psi)\cdot\dot{\psi} - \dot{v}\sin(\psi) - v\cos(\psi)\dot{\psi} \\ \ddot{y} &= \dot{u}\sin(\psi) + u\cos(\psi)\cdot\dot{\psi} + \dot{v}\cos(\psi) - v\sin(\psi)\dot{\psi} \end{aligned} \quad (10.62)$$

Moreover, it holds that

$$\begin{aligned} \ddot{x} + \beta\dot{x} &= \cos(\psi)(\dot{u} - v\dot{\psi} + \beta u) + \sin(\psi)(-u\dot{\psi} - \dot{v} - \beta v) \\ \ddot{y} + \beta\dot{y} &= \cos(\psi)(\dot{v} + u\dot{\psi} + \beta v) + \sin(\psi)(-v\dot{\psi} + \dot{u} + \beta u) \end{aligned} \quad (10.63)$$



Using Eqs. (10.63) and (10.39), and after computing that

$$\begin{aligned} u\dot{\psi} + \dot{v} + \beta v &= u \cdot r - ur - \beta v + \beta v = 0 \\ \dot{u} - v\dot{\psi} + \beta u &= vr + \tau_u - vr + \beta u = \tau_u + \beta u \end{aligned} \quad (10.64)$$

one obtains

$$\frac{\ddot{y} + \beta \dot{y}}{\ddot{x} + \beta \dot{x}} = \frac{\cos(\psi)0 + \sin(\psi)(\tau_u + \beta u)}{\cos(\psi)(\tau_u + \beta u) - \sin(\psi)0} \Rightarrow \quad (10.65)$$

$$\frac{\ddot{y} + \beta \dot{y}}{\ddot{x} + \beta \dot{x}} = \tan(\psi) \Rightarrow \psi = \text{atan}^{-1}\left(\frac{\ddot{y} + \beta \dot{y}}{\ddot{x} + \beta \dot{x}}\right)$$

Thus, through Eq. (10.65) it is proven that the state variable  $\psi$  (heading angle of the vessel) is a function of the flat output and of its derivatives.

From Eq. (10.63) one also has that

$$(\ddot{x} + \beta \dot{x})^2 + (\ddot{y} + \beta \dot{y})^2 = (\tau_u + \beta u)^2 \quad (10.66)$$

Moreover, it holds that

$$\begin{aligned} \dot{x}(\ddot{x} + \beta \dot{x}) &= (u\cos(\psi) - v\sin(\psi))\cos(\psi)(\tau_u + \beta u) \\ \dot{y}(\ddot{y} + \beta \dot{y}) &= (u\sin(\psi) + v\cos(\psi))\sin(\psi)(\tau_u + \beta u) \end{aligned} \quad (10.67)$$

while using Eq. (10.66) and after intermediate computations one finally obtains

$$\dot{x}(\ddot{x} + \beta \dot{x}) + \dot{y}(\ddot{y} + \beta \dot{y}) = u(\tau_u + \beta u) \quad (10.68)$$

Dividing Eq. (10.68) with the square root of Eq. (10.66) one obtains

$$\frac{\dot{x}(\ddot{x} + \beta \dot{x}) + \dot{y}(\ddot{y} + \beta \dot{y})}{\sqrt{(\ddot{x} + \beta \dot{x})^2 + (\ddot{y} + \beta \dot{y})^2}} = \frac{u(\tau_u + \beta u)}{(\tau_u + \beta u)} \quad (10.69)$$

which finally give

$$u = \frac{\dot{x}(\ddot{x} + \beta \dot{x}) + \dot{y}(\ddot{y} + \beta \dot{y})}{\sqrt{(\ddot{x} + \beta \dot{x})^2 + (\ddot{y} + \beta \dot{y})^2}} \quad (10.70)$$

It also holds that

$$\begin{aligned} \dot{y}\ddot{x} - \dot{x}\ddot{y} &= (u\sin(\psi) + v\cos(\psi))(\dot{u}\cos(\psi) - u\sin(\psi)\dot{\psi} - \dot{v}\sin(\psi) - v\cos(\psi)\dot{\psi}) - \\ &\quad -(u\cos(\psi) - v\sin(\psi))(\dot{u}\sin(\psi) + u\cos(\psi)\dot{\psi} + \dot{v}\cos(\psi) - v\sin(\psi)\dot{\psi}) \end{aligned}$$

which after intermediate computations and substitution of the derivative variables from Eq. (10.39) give

$$\dot{y}\ddot{x} - \dot{x}\ddot{y} = v(\beta u + \tau_u) \quad (10.71)$$

From Eqs. (10.71) and (10.66) one obtains

$$v = \frac{\dot{y}\ddot{x} - \dot{x}\ddot{y}}{\sqrt{(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2}} \quad (10.72)$$

From the state-space equations it holds that

$$r = \dot{\psi} \quad (10.73)$$

where from Eq. (10.65) one has that

$$\psi = \text{atan}^{-1} \left( \frac{\ddot{y} + \beta\dot{y}}{\ddot{x} + \beta\dot{x}} \right) \quad (10.74)$$

which means that  $r$  is also a function of the flat output and of its derivatives. This can be also confirmed analytically. Indeed from Eq. (10.74) it holds that

$$\frac{\cos^2(\psi)\dot{\psi} + \sin^2(\psi)\ddot{\psi}}{\cos^2(\psi)} = \frac{(y^{(3)} + \beta\ddot{\psi})(\ddot{x} + \beta\dot{x}) - (\ddot{y} + \beta\dot{y})(x^{(3)} + \beta\ddot{x})}{(\ddot{x} + \beta\dot{x})^2} \quad (10.75)$$

which also gives

$$\frac{\dot{\psi}}{\cos^2(\psi)} = \frac{(y^{(3)} + \beta\ddot{\psi})(\ddot{x} + \beta\dot{x}) - (\ddot{y} + \beta\dot{y})(x^{(3)} + \beta\ddot{x})}{(\ddot{x} + \beta\dot{x})^2} \quad (10.76)$$

while using that

$$\frac{1}{\cos^2 \psi} = \tan^2(\psi) + 1 \quad (10.77)$$

one obtains that

$$\cos^2 \psi = \frac{(\ddot{x} + \beta\dot{x})^2}{(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2} \quad (10.78)$$

Thus, from Eqs. (10.76) and (10.73) one has that

$$r = \dot{\psi} \Rightarrow r = \cos^2(\psi) \frac{(y^{(3)} + \beta\ddot{\psi})(\ddot{x} + \beta\dot{x}) - (\ddot{y} + \beta\dot{y})(x^{(3)} + \beta\ddot{x})}{(\ddot{x} + \beta\dot{x})^2} \quad (10.79)$$

which after intermediate operations gives

$$r = \frac{y^{(3)}(\ddot{x} + \beta\dot{x}) - x^{(3)}(\ddot{y} + \beta\dot{y}) - \beta^2(\ddot{x}\dot{y} - \ddot{y}\dot{x})}{(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2} \quad (10.80)$$

Equivalently, from the state-space equations one has that

$$\tau_u = \dot{u} - v \cdot r \Rightarrow \tau_u = \frac{d}{dt} \left\{ \frac{\dot{x}(\ddot{x} + \beta\dot{x}) + \dot{y}(\ddot{y} + \beta\dot{y})}{\sqrt{(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2}} \right\} -$$

$$- \frac{\dot{y}\ddot{x} - \dot{x}\ddot{y}}{\sqrt{(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2}} \cdot \frac{y^{(3)}(\ddot{x} + \beta\dot{x}) - x^{(3)}(\ddot{y} + \beta\dot{y}) - \beta^2(\ddot{x}\dot{y} - \dot{y}\ddot{x})}{(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2} \quad (10.81)$$

which after intermediate operations gives

$$\tau_u = \frac{\ddot{x}(\ddot{x} + \beta\dot{x}) + \ddot{y}(\ddot{y} + \beta\dot{y})}{\sqrt{(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2}} \quad (10.82)$$

Finally, for the control input  $\tau_r$  it holds that  $\tau_r = \dot{r}$  and using Eq. (10.80) this implies that  $\tau_r$  is also a function of the flat output and of its derivatives. This can be also shown analytically according to the following:

$$\tau_r = \dot{r} \Rightarrow \tau_r =$$

$$\frac{y^{(4)}(\ddot{x} + \beta\dot{x}) - x^{(4)}(\ddot{y} + \beta\dot{y}) + \beta(y^{(3)}\ddot{x} - x^{(3)}\ddot{y}) - \beta^2(x^{(3)}\dot{y} - y^{(3)}\dot{x})}{[(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2]} \cdot$$

$$- 2 \frac{[y^{(3)}(\ddot{x} + \beta\dot{x}) - x^3(\ddot{y} + \beta\dot{y}) - \beta^2(\ddot{x}\dot{y} - \dot{y}\ddot{x})]}{[(\ddot{x} + \beta\dot{x})^2 + (\ddot{y} + \beta\dot{y})^2]^2} \cdot$$

$$\{(\ddot{x} + \beta\dot{x})(x^{(3)} + \beta\ddot{x}) + (\ddot{y} + \beta\dot{y})(y^{(3)} + \beta\ddot{y})\} \quad (10.83)$$

Through Eq. (10.83) it is confirmed that that all state variables and the control input of the hovercraft's model can be written as functions of the flat output and of its derivatives. Consequently, the vessel's model is a differential flat one.

### 10.3.4 Flatness-Based Control of the Hovercraft's Model

Next, it will be shown that a flatness-based controller can be developed for the hovercraft's model. It has been shown that it holds

$$\ddot{x} = \dot{u}\cos(\psi) - u\sin(\psi)\dot{\psi} - \dot{v}\sin(\psi) - v\cos(\psi)\dot{\psi} \Rightarrow \ddot{x} = (vr + \tau_u)\cos(\psi) - u\sin(\psi)r - (-ur - \beta v)\sin(\psi) - v\cos(\psi)r \Rightarrow \ddot{x} = \tau_u\cos(\psi) + \beta v\sin(\psi)$$

By differentiating once more with respect to time and after intermediate operations one finally obtains

$$x^{(3)} = \dot{\tau}_u\cos(\psi) - \tau_u\sin(\psi)r +$$

$$+ \beta(-ur - \beta v)\sin(\psi) + \beta v\cos(\psi)r \quad (10.84)$$

Similarly one has

$$\ddot{y} = \dot{u}\sin(\psi) + u\cos(\psi)\dot{\psi} + \dot{v}\cos(\psi) - v\sin(\psi)\dot{\psi} \Rightarrow \ddot{y} = (vr + \tau_u)\sin(\psi) + u\cos(\psi)r + (-ur - \beta v)\cos(\psi) - v\sin(\psi)r \Rightarrow \ddot{y} = \tau_u\sin(\psi) - \beta v\cos(\psi)$$

As in Sect. 10.3.2, the state vector of the system is extended so as to include as new state variables the control input  $\tau_u$  and its first derivative  $\dot{\tau}_u$ . The new state variables are denoted as  $z_1 = \tau_u$  and  $z_2 = \dot{\tau}_u$ . The extended state-space description of the system becomes

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{r} \\ \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} = \begin{pmatrix} u\cos(\psi) - v\sin(\psi) \\ u\sin(\psi) + v\cos(\psi) \\ r \\ vr + z_1 \\ -ur - \beta v \\ 0 \\ z_2 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \begin{pmatrix} \ddot{\tau}_u \\ \tau_r \end{pmatrix} \quad (10.85)$$

or equivalently, one has the description

$$\dot{z} = f(z) + g(z)\tilde{v} \quad (10.86)$$

The system's state vector is again denoted as  $z = [x, y, \psi, u, v, r, z_1, z_2]^T$ ,  $f(z) \in \mathbb{R}^{8 \times 1}$ , and  $g(z) = [g_a, g_b] \in \mathbb{R}^{8 \times 2}$ , while the control input is the vector  $\tilde{v} = [\ddot{\tau}_u, \tau_r]^T$ .

The extended state-space description of the system given in Eq. (10.85) or in its compact form described by Eq. (10.86), is used. By differentiating once more with respect to time and after intermediate operations one finally obtains

$$\begin{aligned} y^{(3)} &= z_2\sin(\psi) + z_1\cos(\psi)r + \beta ur\cos(\psi) + \\ &+ \beta^2 v\cos(\psi) + \beta v\sin(\psi)r \end{aligned} \quad (10.87)$$

It can be noticed that in the equations of the third order derivatives for both  $x$  and  $y$  only the control input  $\tau_u$  and its derivative  $\dot{\tau}_u$  appear, while the control input  $\tau_r$  is missing. Therefore, differentiation of  $x^{(3)}$  once more with respect to time is performed. This gives

$$\begin{aligned} x^{(4)} &= \ddot{\tau}_u\cos(\psi) - 2z_2\sin(\psi)r - z_1\cos(\psi)r^2 - z_1\sin(\psi)\tau_r - \beta vr^2\sin(\psi) - \\ &\beta z_1 r\sin(\psi) - \beta u\tau_r\sin(\psi) - \beta ur^2\cos(\psi) + \beta^2 ur\sin(\psi) - \beta^3 v\sin(\psi) - \beta^2 vr\cos(\psi) - \\ &\beta ur^2\cos(\psi) + \beta^2 vr\cos(\psi) - \beta vr^2\sin(\psi) + \beta v\cos(\psi)\tau_r \end{aligned}$$

while after substituting the time derivative according to Eq. (10.39) and after regrouping terms one obtains a description of the form

$$\begin{aligned}
x^{(4)} = & [-2z_2 \sin(\psi)r - z_1 \cos(\psi)r^2 - \beta vr^2 \sin(\psi) - \beta z_1 r \sin(\psi) - \\
& -\beta ur^2 \cos(\psi) + \beta^2 ursin(\psi) - \beta^3 v \sin(\psi) - \beta^2 vrcos(\psi) - \\
& -\beta ur^2 \cos(\psi) + \beta^2 vrcos(\psi) - \beta vr^2 \sin\psi] + [\cos(\psi)]\ddot{\tau}_u + \\
& + [-z_1 \sin(\psi) - \beta u \sin(\psi) + \beta v \cos(\psi)]\tau_r
\end{aligned}$$

Consequently, the fourth derivative of  $x$  is finally written in the form

$$x^{(4)} = L_f^4 y_1 + L_{g_a} L_f^3 y_1 \ddot{\tau}_u + L_{g_b} L_f^3 y_1 \tau_r \quad (10.88)$$

where

$$\begin{aligned}
L_f^4 y_1 = & -2z_2 \sin(\psi)r - z_1 \cos(\psi)r - \beta vr^2 \sin(\psi) - \beta z_1 r \sin(\psi) - \\
& -\beta ur^2 \cos(\psi) + \beta^2 ursin(\psi) - \beta^3 v \sin(\psi) - \beta^2 vrcos(\psi) - \beta ur^2 \cos(\psi) + \\
& \beta^2 vrcos(\psi) - \beta vr^2 \sin\psi
\end{aligned}$$

$$L_{g_a} L_f^3 y_1 = \cos(\psi) \quad (10.89)$$

$$L_{g_b} L_f^3 y_1 = -z_1 \sin(\psi) - \beta u \sin(\psi) + \beta v \cos(\psi) \quad (10.90)$$

In a similar manner, differentiating once more with respect to time the expression about  $y^{(3)}$  one gets

$$\begin{aligned}
y^{(4)} = & \dot{z}_1 \cos(\psi)r - z_1 \sin(\psi)\dot{\psi}r + z_1 \cos(\psi)\dot{r} - \\
& -\beta \dot{v}r \sin(\psi) - \beta v \dot{r} \sin(\psi) - \beta vr \cos(\psi)\dot{\psi} - \\
& -\beta \dot{u}r \cos(\psi) - \beta u \dot{r} \cos(\psi) + \beta u r \sin(\psi)\dot{\psi} + \\
& + \beta^2 \dot{v} \cos(\psi) - \beta^2 v \sin\psi \dot{\psi} + \\
& + \dot{z}_2 \sin(\psi) + z_2 \cos(\psi)\dot{\psi}
\end{aligned} \quad (10.91)$$

while after substituting the time derivative according to Eq. (10.39) and after regrouping terms one obtains a description of the form

$$\begin{aligned}
y^{(4)} = & [z_2 r \cos(\psi) - z_1 r^2 \sin(\psi) + \beta ur^2 \sin(\psi) - \beta^2 vr \sin(\psi) - \beta vr^2 \cos(\psi)] \\
& -\beta vr^2 \cos(\psi) - \beta z_1 r \cos(\psi) + \beta ur^2 \sin(\psi) - \\
& -\beta ur \cos(\psi) + \beta^2 v \cos(\psi) - \beta^2 vr \sin(\psi) + z_2 r \cos(\psi)] + \\
& + [\sin(\psi)]\ddot{\tau}_u + [z_1 \cos(\psi) - \beta v \sin(\psi) - \beta u \cos(\psi)]\tau_r
\end{aligned}$$

Thus  $y^{(4)}$  can be also written in the form

$$y^{(4)} = L_f^4 y_2 + L_{g_a} L_f^3 y_2 \ddot{\tau}_u + L_{g_b} L_f^3 y_2 \tau_r \quad (10.92)$$

where  $L_f^4 y_2 = [z_2 r \cos(\psi) - z_1 r^2 \sin(\psi) + \beta u r^2 \sin(\psi) - \beta^2 v r \sin(\psi) - \beta v r^2 \cos(\psi) - \beta v r^2 \cos(\psi) - \beta z_1 r \cos(\psi) + \beta u r^2 \sin(\psi) - \beta u r \cos(\psi) + \beta^2 v \cos(\psi) - \beta^2 v r \sin(\psi) + z_2 r \cos(\psi)]$ , and

$$L_{g_a} L_f^3 y_2 = \sin(\psi) \quad (10.93)$$

$$L_{g_b} L_f^3 y_2 = z_1 \cos(\psi) - \beta v \sin(\psi) - \beta u \cos(\psi) \quad (10.94)$$

Consequently, the aggregate input-output linearized description of the system becomes

$$\begin{aligned} x^{(4)} &= L_f^4 y_1 + L_{g_a} L_f^3 y_1 \ddot{\tau}_u + L_{g_b} L_f^3 y_1 \tau_r \\ y^{(4)} &= L_f^4 y_2 + L_{g_a} L_f^3 y_2 \ddot{\tau}_u + L_{g_b} L_f^3 y_2 \tau_r \end{aligned} \quad (10.95)$$

while by defining the new control inputs

$$\begin{aligned} v_1 &= L_f^4 y_1 + L_{g_a} L_f^3 y_1 \ddot{\tau}_u + L_{g_b} L_f^3 y_1 \tau_r \\ v_2 &= L_f^4 y_2 + L_{g_a} L_f^3 y_2 \ddot{\tau}_u + L_{g_b} L_f^3 y_2 \tau_r \end{aligned} \quad (10.96)$$

the following description for the input-output linearized hovercraft model is obtained

$$\begin{aligned} x^{(4)} &= v_1 \\ y^{(4)} &= v_2 \end{aligned} \quad (10.97)$$

For the dynamics of the linearized equivalent model of the system the following new state variables can be defined

$$\begin{aligned} z_{1,1} &= x \quad z_{1,2} = \dot{x} \quad z_{1,3} = \ddot{x} \quad z_{1,4} = x^{(3)} \\ z_{2,1} &= y \quad z_{2,2} = \dot{y} \quad z_{2,3} = \ddot{y} \quad z_{2,4} = y^{(3)} \end{aligned} \quad (10.98)$$

and the state-space description of the system becomes

$$\begin{aligned} \dot{z} &= Az + Bv \\ z^m &= Cz \end{aligned} \quad (10.99)$$

or equivalently

$$\begin{pmatrix} \dot{z}_{1,1} \\ \dot{z}_{1,2} \\ \dot{z}_{1,3} \\ \dot{z}_{1,4} \\ \dot{z}_{2,1} \\ \dot{z}_{2,2} \\ \dot{z}_{2,3} \\ \dot{z}_{2,4} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_{1,1} \\ z_{1,2} \\ z_{1,3} \\ z_{1,4} \\ z_{2,1} \\ z_{2,2} \\ z_{2,3} \\ z_{2,4} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (10.100)$$

while the associated measurement equation is

$$\begin{pmatrix} z_1^m \\ z_2^m \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_{1,1} \\ z_{1,2} \\ z_{1,3} \\ z_{1,4} \\ z_{2,1} \\ z_{2,2} \\ z_{2,3} \\ z_{2,4} \end{pmatrix} \quad (10.101)$$

A suitable feedback control law for the linearized system is

$$v_1 = x_d^{(4)} - k_1^1(x^{(3)} - x_d^{(3)}) - k_2^1(\ddot{x} - \ddot{x}_d) - k_3^1(\dot{x} - \dot{x}_d) - k_4^1(x - x_d), \text{ and } v_2 = y_d^{(4)} - k_1^2(y^{(3)} - y_d^{(3)}) - k_2^2(\ddot{y} - \ddot{y}_d) - k_3^2(\dot{y} - \dot{y}_d) - k_4^2(y - y_d)$$

One can compute again the control input that is finally applied to the hovercraft model. It holds that

$$\bar{v} = \tilde{f} + \tilde{M}\tilde{v} \quad (10.102)$$

where matrices and vectors  $\bar{v}$ ,  $\tilde{f}$ ,  $\tilde{M}$  and  $\tilde{v}$  are defined as

$$\bar{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad \tilde{f} = \begin{pmatrix} L_f^4 z_{1,1} \\ L_f^4 z_{2,1} \end{pmatrix} \quad (10.103)$$

$$\tilde{M} = \begin{pmatrix} L_{g,a} L_f^3 z_{1,1} & L_{g,b} L_f^3 z_{1,1} \\ L_{g,a} L_f^3 z_{2,1} & L_{g,b} L_f^3 z_{2,1} \end{pmatrix} \quad \tilde{v} = \begin{pmatrix} \ddot{u} \\ \tau_r \end{pmatrix}$$

The above equation can be solved with respect to  $\tilde{u}$ , which finally gives

$$\tilde{v} = \tilde{M}^{-1}(\bar{v} - \tilde{f}) \quad (10.104)$$

The vector  $\tilde{v}$  is the control input that is finally applied to the system, which finally means that the control signal contains integrals of the tracking error.

For the linearized equivalent model of the system it is possible to perform state estimation using the Derivative-free nonlinear Kalman Filter. Before computing the Kalman Filter stages, the previously defined matrices  $A$ ,  $B$  and  $C$  are substituted by their discrete-time equivalents  $A_d$ ,  $B_d$  and  $C_d$ . This is done through common discretization methods. The recursion of the filter's algorithm consists of two stages:

*Measurement update:*

$$\begin{aligned} K(k) &= P^- C_d^T [P^- C_d^T P + R]^{-1} \\ \hat{z}(k) &= \hat{z}^-(k) - K(k) [C_d z(k) - C_d \hat{z}^-(k)] \\ P(k) &= P^-(k) - K(k) C_d P^-(k) \end{aligned} \quad (10.105)$$

*Time update:*

$$\begin{aligned} P^-(k+1) &= A_d^T P(k) A_d + Q(k) \\ \hat{z}^-(k+1) &= A_d \hat{z}(k) + B_d u(k) \end{aligned} \quad (10.106)$$

Moreover, by using the nonlinear transformations which are provided by differential flatness theory according to Eqs. (10.70), (10.72), (10.74) and (10.79) one can obtain estimates of the state variables of the initial nonlinear hovercraft model.

### 10.3.5 Disturbances' Compensation with the Use of the Derivative-Free Nonlinear Kalman Filter

Next, a Kalman Filtering method will be developed for estimating at the same time: (i) the non-measurable state vector elements of the hovercraft and (ii) the external perturbations that affect the vessel's model. It is assumed that the input-output linearized equivalent model of the system, is subject to disturbance terms which express the effects of both modeling uncertainty and of external perturbations. Thus one has

$$\begin{aligned} x^{(4)} &= v_1 + \tilde{d}_1 \\ y^{(4)} &= v_2 + \tilde{d}_2 \end{aligned} \quad (10.107)$$

It is considered that the disturbance signals are equivalently represented by their time derivatives (up to order  $n$ ) and by the associated initial conditions (however, since disturbances are estimated with the use of the Kalman Filter, finally the dependence on knowledge of initial conditions becomes obsolete). It holds that

$$\tilde{d}_1^{(n)} = f_{d_1} \tilde{d}_2^{(n)} = f_{d_2} \quad (10.108)$$

The system's state vector is extended by including as additional state variables the disturbances' derivatives. Thus, taking that the derivative's order is  $n = 2$  one has





Again the Kalman Filter recursion provides joint estimation of the non-measurable state vector elements, of the disturbances' inputs and of their derivatives. Prior to computing the Kalman Filter stages, the previously defined matrices  $A$ ,  $B$  and  $C$  are substituted by their discrete-time equivalents  $A_{e_d}$ ,  $B_{e_d}$  and  $C_{e_d}$ . This is done through common discretization methods. The recursion of the filter's algorithm consists of two stages. Thus, one has

*Measurement update:*

$$\begin{aligned} K(k) &= P_e^- C_{e_d}^T [P_e^- C_{e_d}^T P_e + R_e]^{-1} \\ \hat{z}_e(k) &= \hat{z}_e^-(k) - K(k) [C_{e_d} z_e(k) - C_{e_d} \hat{z}_e^-(k)] \\ P_e(k) &= P_e^-(k) - K(k) C_{e_d} P_e^-(k) \end{aligned} \quad (10.115)$$

*Time update:*

$$\begin{aligned} P_e^-(k+1) &= A_{e_d}^T P_e(k) A_{e_d} + Q_e(k) \\ \hat{z}_e^-(k+1) &= A_{e_d} \hat{z}_e(k) + B_{e_d} v_e(k) \end{aligned} \quad (10.116)$$

For compensating the disturbances' effects, the modified control input that is applied to the system is

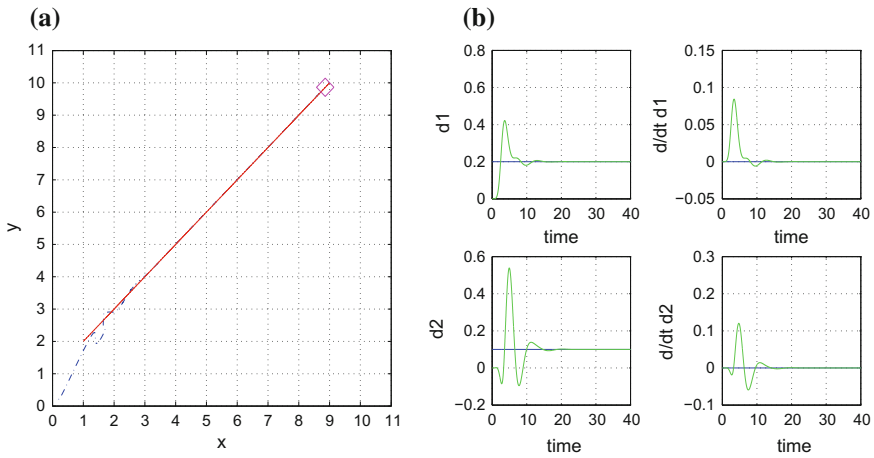
$$\begin{aligned} v_1 &= x_d^{(4)} - k_1^1(x^{(3)} - x_d^{(3)}) - k_2^1(\ddot{x} - \ddot{x}_d) - k_3^1(\dot{x} - \dot{x}_d) - k_4^1(x - x_d) - \hat{z}_{d,1} \text{ and} \\ v_2 &= y_d^{(4)} - k_1^2(y^{(3)} - y_d^{(3)}) - k_2^2(\ddot{y} - \ddot{y}_d) - k_3^2(\dot{y} - \dot{y}_d) - k_4^2(y - y_d) - \hat{z}_{d,2}. \end{aligned}$$

### 10.3.6 Simulation Tests

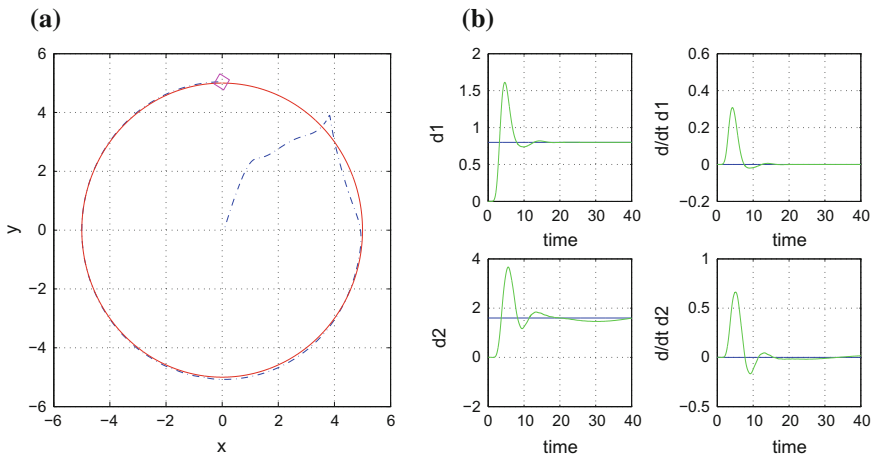
The performance of the flatness-based control method for the underactuated hovercraft was evaluated in the case of several reference setpoints. The associated results are presented in Figs. 10.36, 10.37, 10.38, 10.39 and 10.40. It can be observed that in all cases the nonlinear feedback controller achieved fast and accurate tracking of the reference setpoints. The Derivative-free nonlinear Kalman Filter enabled estimation of the nonmeasurable variables of the system's state-vector which were needed for the implementation of the feedback control scheme. Moreover, by using the inverse transformation that was provided by differential flatness theory it was possible to obtain estimates of the state variables of the initial nonlinear system.

The convergence of the state variables of the hovercraft (position  $x$ ,  $y$  to the desirable setpoints is shown in Figs. 10.36a, 10.37a, 10.38a, 10.39a and 10.40a. The estimation of the disturbance terms that were applied to the hovercraft model are depicted in Figs. 10.36b, 10.37b, 10.38b, 10.39b and 10.40b, respectively. It can be noticed again that the proposed feedback nonlinear control scheme achieved fast and accurate tracking to these setpoints.

For the underactuated hovercraft one can succeed exactly the same motion and orientation control as in the case of the fully actuated vessel. Therefore, it is possible

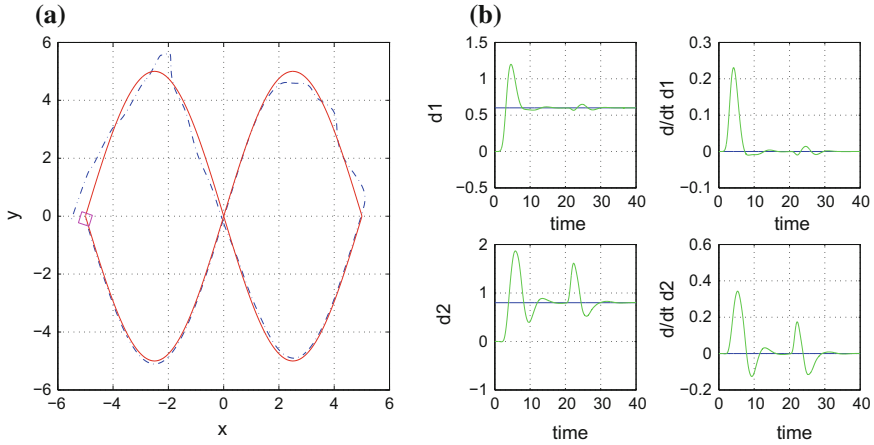


**Fig. 10.36** Reference path 1 **a** Trajectory tracking for states  $x, y$  of the underactuated hovercraft, **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

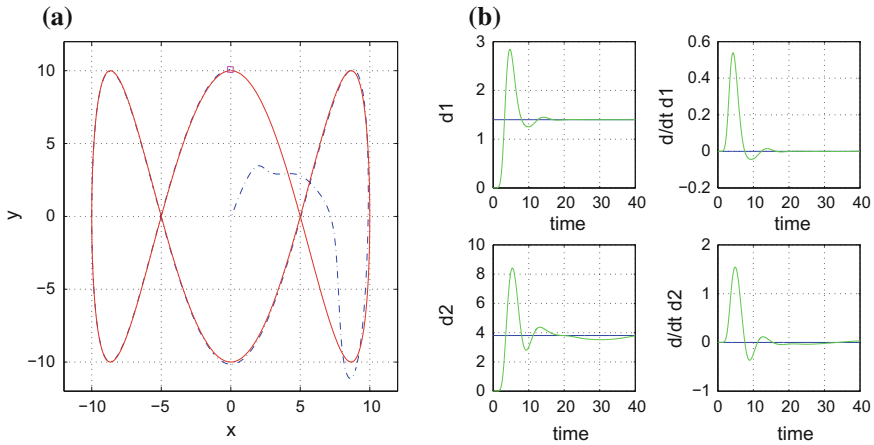


**Fig. 10.37** Reference path 2 **a** Trajectory tracking for states  $x, y$  of the underactuated hovercraft, **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

for the hovercraft to track complicated reference paths with excellent accuracy while keeping also the desirable velocity. This has been demonstrated through a series of examples, in the simulation tests section of the manuscript (Figs. 10.36, 10.37, 10.38, 10.39 and 10.40). It is noteworthy that the dynamic feedback linearization procedure which has been implemented on the hovercraft’s model, results in the canonical form description of Eqs. (10.100) and (10.101) which is confirmed to



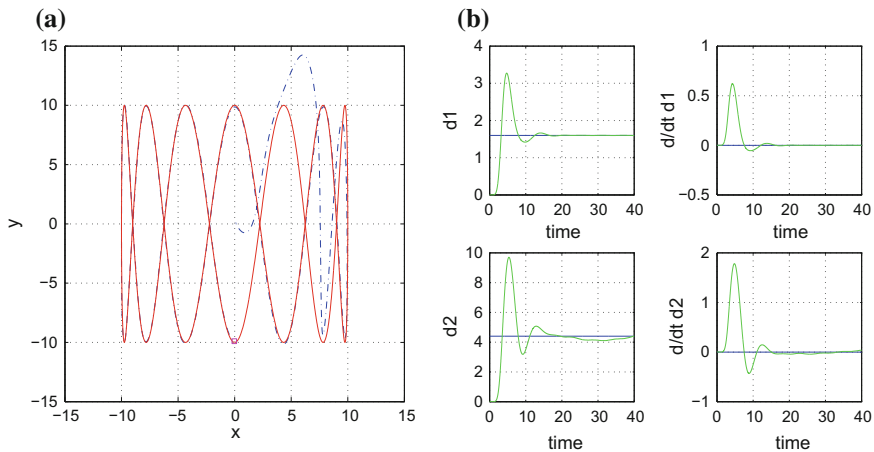
**Fig. 10.38** Reference path 3 **a** Trajectory tracking for states  $x$ ,  $y$  of the underactuated hovercraft, **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter



**Fig. 10.39** Reference path 4 **a** Trajectory tracking for states  $x$ ,  $y$  of the underactuated hovercraft, **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

be controllable. Practically, this means that under the proposed control scheme the vessel can reach any point in its motion plane and can track any reference path.

The possibility for the appearance of singularities in the computation of the control signal is present in all static or dynamic feedback linearization algorithms which arrive at a transformed control input of the form  $v = f(x, t) + g(x, t)u$ , that is  $u = g(x, t)^{-1}[v - f(x, t)]$ . There are two cases: (i) due to the inherent model of  $g(x, t)$  its inverse never becomes 0. In such a case the singularity problem is avoided,



**Fig. 10.40** Reference path **a** Trajectory tracking for states  $x$ ,  $y$  of the underactuated hovercraft, **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

(ii) for certain areas of the state vector space  $x \in R^n$  the zeroing of  $g(x, t)^{-1}$  cannot be excluded. For the latter case the avoidance of singularities can be succeeded by a state variables transformation into a new state-space representation which does not include any points of singularity.

The presented simulation experiments have been performed under the assumption that the hovercraft was subject to external disturbances such as wind or current. The proposed control scheme is robust to modeling uncertainties and external perturbations. This is a prerequisite in the design of control systems for underactuated surface vessels [127, 167, 297, 345, 498, 627]. First, it has been proven that the feedback control applied on the input-output linearized model of the hovercraft achieved the placement of all poles of the control loop in the left complex semiplane. Next, it can be confirmed that the extended state-space model of the hovercraft, which contains disturbances as additional state variables, has multiple poles at the origin (multiple poles at zero). This means that by output feedback one can achieve infinite gain margin and a sufficiently large phase margin. With the use of the Derivative-free nonlinear Kalman Filter it became possible to identify the perturbation and modeling uncertainty terms in real-time and subsequently to compensate for them through the inclusion of an additional term in the control signal. This amendment in the feedback control scheme provided the control loop with elevated robustness. Finally, it is worth mentioning that the proposed control scheme had an excellent performance although it was not possible to measure directly all elements of the state vector (only the cartesian coordinates of the vessel could be measured) and several state variables had to be estimated with the use of filtering.

The model of the disturbances considered in this section is quite realistic. The disturbance inputs can be represented equally well if their analytical function is known

or if their  $n$ th order derivative and the associated initial conditions are known. However, in the latter case and with the Kalman filtering approach followed in the present section, the knowledge of initial conditions becomes obsolete since the Kalman Filter can reconstruct the disturbance inputs and their derivatives without dependence on initial conditions. In conclusion, the numerical simulation performed in this section estimates quite well the disturbances' effects. The disturbance inputs have a clear physical meaning since they represent the effects of wind forces or wave forces exerted on the vessel.

The control method which has been implemented in the present section is a global linearization one. This means that the vessel's dynamic model is transformed through a change of variables into an equivalent linear description for which the design of the feedback controller becomes easier. Moreover, this linearization is an exact one because it does not introduce any numerical errors due to truncation of terms in the linearization procedure. Prior to this transformation the so-called dynamic extension is performed that is the vessel's state-space model is extended by considering as additional state variables the control inputs and their derivatives [382, 634]. A second major class of solutions for the problem of autonomous navigation of underactuated vessels uses methods that asymptotically linearize the vessel's dynamics. This holds for instance in the case of H-infinity control or local models fuzzy control. The vessel's model can be linearized round local operating points. Next, for the linearized model of the vessel a feedback controller is designed, taking also into account that robustness should be exhibited against both approximate linearization errors and external perturbations [79, 99, 205, 564]. A third class of possible solutions comprises Lyapunov methods in which the stabilizing control of the underactuated vessels is obtained from the procedure of minimization of a suitably chosen Lyapunov function. Such solutions can be model-based which means that prior knowledge about the vessel's dynamic model is available and is used by the control algorithm. They can also be model-free taking the form of adaptive control. In the latter case the vessel's dynamic model is considered to be completely unknown and is identified online by an adaptation scheme during the execution of the control algorithm [126, 127, 251, 279, 398]. The comparison of the aforementioned approaches shows that the differential flatness theory-based method for the underactuated vessel is conceptually simpler and straightforward to implement, while also avoiding linearization approximations, numerical errors and constraining assumptions about the structure of the controlled vessel's model.

## **10.4 A Nonlinear H-Infinity Control Approach for Underactuated Surface Vessels**

### **10.4.1 Outline**

In this section control of unmanned surface vessels (USVs) is based on a local linearization approach. The linearization takes place round the USV's local operating

point which is defined at each time instant by the present value of the state vector and the last value of the control inputs vector [461]. The linearization makes use of Taylor series expansion and of the computation of Jacobian matrices [33, 431, 463, 564]. The modelling error, due to truncation of higher order terms in the Taylor series, is considered as perturbation which is compensated by the robustness of the control algorithm. For the linearized model of the USV an H-infinity feedback controller is designed. A cost function is introduced comprising the weighted square of the error of the system's state vector (distance of the state vector from the reference setpoints).

This control method represents a differential game taking place between the control input which tries to minimize the above cost function and between the disturbances which try to maximize this objective function. The computation of the feedback control gain requires the solution of an algebraic Riccati equation, which takes place at each iteration of the control algorithm. The solution of the Riccati equation provides a positive definite symmetric matrix which is used as a weighting coefficient in the computation of the controller's feedback gain. The known robustness features of H-infinity control assure the elimination of perturbation effects, which in turn implies compensation of model uncertainty terms, external disturbance inputs as well as of measurement noises. The stability properties of the control scheme are assured by Lyapunov analysis. It is shown that the proposed feedback control law for USVs results in H-infinity tracking performance which means robustness against modeling uncertainty and external perturbations. Under moderate conditions it is also proven that the control loop is also globally asymptotically stable. The tracking accuracy and the smooth transients in the proposed USV control method are also confirmed through simulation experiments.

Yet computationally simple, the proposed  $H_\infty$  control scheme has an excellent performance. Comparing to the control of underactuated vessels that is based on global linearization methods (see [416, 450, 452, 457, 460]), the following features can be attributed to the presented nonlinear H-infinity control scheme (i) it is applied directly on the nonlinear dynamical model of the underactuated vessel and does not require the computation of diffeomorphisms (change of variables) that will bring the system into an equivalent linearized form, (ii) the computation of the feedback control signal follows an optimal control concept and requires the solution of an algebraic Riccati equation at each iteration of the control algorithm, (iii) the method retains the advantages of typical optimal control, that is fast and accurate tracking of the reference trajectories under moderate variations of the control inputs.

### ***10.4.2 Approximate Linearization of the Underactuated Vessel***

In the previous section it was shown that the joint kinematic and dynamic model of the underactuated vessel is

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} u\cos(\psi) - v\sin(\psi) \\ u\sin(\psi) + v\cos(\psi) \\ r \\ v \cdot r \\ -ur - \beta v \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \tau_u \\ \tau_r \end{pmatrix} \tag{10.117}$$

where  $(x, y)$  are the cartesian coordinates if the vessel in the inertial reference frame,  $\psi$  is the heading angle of the vessel in the inertial reference frame,  $u$  is the surge velocity,  $v$  is the sway velocity,  $r$  is the yaw rate,  $\beta$  is a function of the elements of the inertia matrix of the vessel.

Linearization of the vessel’s kinematic and dynamic model will be performed round a local operating point (equilibrium)  $(x^*, u^*)$ . To this end, the joint kinematics and dynamics model of Eq. (10.117) is written in the form:

$$\dot{x} = f(x) + g(x)u \tag{10.118}$$

where the state vector is  $x = [x_1, x_2, x_3, x_4, x_5, x_6]^T = [x, y, \psi, u, v, r]^T$  and

$$f(x) = \begin{pmatrix} x_4\cos(x_3) - x_5\sin(x_3) \\ x_4\sin(x_3) + x_5\cos(x_3) \\ x_6 \\ x_5x_6 \\ -x_4x_6 - \beta x_5 \\ 0 \end{pmatrix} \quad g(x) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \tag{10.119}$$

The linearization of the vessel’s model round the temporary equilibrium gives

$$\dot{x} = Ax + Bu \tag{10.120}$$

where

$$A = \nabla_x[f(x) + g(x)u] |_{(x^*,u^*)} \Rightarrow A = \nabla_x f(x) |_{(x^*,u^*)} \tag{10.121}$$

$$B = \nabla_u[f(x) + g(x)u] |_{(x^*,u^*)} \Rightarrow B = g(x) |_{(x^*,u^*)} \tag{10.122}$$

For the Jacobian matrix  $A = \nabla_x[f(x) + g(x)u] |_{(x^*,u^*)}$ , it holds that

$$A = \nabla_x[f(x) + g(x)u] |_{(x^*,u^*)} \Rightarrow$$



$$A = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & \dots & \frac{\partial f_1}{\partial x_6} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \dots & \frac{\partial f_2}{\partial x_6} \\ \dots & \dots & \dots & \dots & \dots \\ \frac{\partial f_6}{\partial x_1} & \frac{\partial f_6}{\partial x_2} & \frac{\partial f_6}{\partial x_3} & \dots & \frac{\partial f_6}{\partial x_6} \end{pmatrix} \quad (10.123)$$

For the first row of the aforementioned Jacobian matrix one has:

$$\frac{\partial f_1}{\partial x_1} = 0, \frac{\partial f_1}{\partial x_2} = 0, \frac{\partial f_1}{\partial x_3} = -x_4 \sin(x_3) - x_5 \cos(x_3), \frac{\partial f_1}{\partial x_4} = \cos(x_3), \frac{\partial f_1}{\partial x_5} = -\sin(x_3), \frac{\partial f_1}{\partial x_6} = 0.$$

For the second row of the aforementioned Jacobian matrix one has:

$$\frac{\partial f_2}{\partial x_1} = 0, \frac{\partial f_2}{\partial x_2} = 0, \frac{\partial f_2}{\partial x_3} = x_4 \cos(x_3) - x_5 \sin(x_3), \frac{\partial f_2}{\partial x_4} = \sin(x_3), \frac{\partial f_2}{\partial x_5} = \cos(x_3), \frac{\partial f_2}{\partial x_6} = 0.$$

For the third row of the aforementioned Jacobian matrix one has:

$$\frac{\partial f_3}{\partial x_1} = 0, \frac{\partial f_3}{\partial x_2} = 0, \frac{\partial f_3}{\partial x_3} = 0, \frac{\partial f_3}{\partial x_4} = 0, \frac{\partial f_3}{\partial x_5} = 0, \frac{\partial f_3}{\partial x_6} = 1.$$

For the fourth row of the aforementioned Jacobian matrix one has:

$$\frac{\partial f_4}{\partial x_1} = 0, \frac{\partial f_4}{\partial x_2} = 0, \frac{\partial f_4}{\partial x_3} = 0, \frac{\partial f_4}{\partial x_4} = 0, \frac{\partial f_4}{\partial x_5} = x_6, \frac{\partial f_4}{\partial x_6} = x_5.$$

For the fifth row of the aforementioned Jacobian matrix one has:

$$\frac{\partial f_5}{\partial x_1} = 0, \frac{\partial f_5}{\partial x_2} = 0, \frac{\partial f_5}{\partial x_3} = 0, \frac{\partial f_5}{\partial x_4} = -x_6, \frac{\partial f_5}{\partial x_5} = -\beta, \frac{\partial f_5}{\partial x_6} = -x_4.$$

For the sixth row of the aforementioned Jacobian matrix one has:

$$\frac{\partial f_6}{\partial x_1} = 0, \frac{\partial f_6}{\partial x_2} = 0, \frac{\partial f_6}{\partial x_3} = 0, \frac{\partial f_6}{\partial x_4} = 0, \frac{\partial f_6}{\partial x_5} = 0, \frac{\partial f_6}{\partial x_6} = 0.$$

### 10.4.3 Design of an H-Infinity Nonlinear Feedback Controller

#### 10.4.3.1 Equivalent Linearized Dynamics of the Vessel

After linearization round its current operating point, the USV's dynamic model is written as

$$\dot{x} = Ax + Bu + d_1 \quad (10.124)$$

Parameter  $d_1$  stands for the linearization error in the USV's dynamic model appearing in Eq. (10.124). The reference setpoints for USV's state vector are denoted by  $\mathbf{x}_d = [x_1^d, \dots, x_6^d]$ . Tracking of this trajectory is achieved after applying the control input  $\tilde{u}^*$ . At every time instant the control input  $\tilde{u}^*$  is assumed to differ from the control input  $u$  appearing in Eq. (10.124) by an amount equal to  $\Delta u$ , that is  $\tilde{u}^* = u + \Delta u$

$$\dot{x}_d = Ax_d + Bu^* + d_2 \tag{10.125}$$

The dynamics of the controlled system described in Eq. (10.124) can be also written as

$$\dot{x} = Ax + Bu + Bu^* - Bu^* + d_1 \tag{10.126}$$

and by denoting  $d_3 = -Bu^* + d_1$  as an aggregate disturbance term one obtains

$$\dot{x} = Ax + Bu + B\tilde{u}^* + d_3 \tag{10.127}$$

By subtracting Eq. (10.125) from (10.127) one has

$$\dot{x} - \dot{x}_d = A(x - x_d) + Bu + d_3 - d_2 \tag{10.128}$$

By denoting the tracking error as  $e = x - x_d$  and the aggregate disturbance term as  $\tilde{d} = d_3 - d_2$ , the tracking error dynamics becomes

$$\dot{e} = Ae + Bu + \tilde{d} \tag{10.129}$$

The above linearized form of the USV's model can be efficiently controlled after applying an H-infinity feedback control scheme.

### 10.4.3.2 The Nonlinear H-Infinity Control

The initial nonlinear model of the unmanned surface vessel is in the form

$$\dot{x} = \tilde{f}(x, u) \quad x \in R^n, \quad u \in R^m \tag{10.130}$$

Linearization of the model of the unmanned surface vessel is performed at each iteration of the control algorithm round its present operating point  $(x^*, u^*) = (x(t), u(t - T_s))$ , where  $T_s$  is the sampling period. The linearized equivalent of the system is described by

$$\dot{x} = Ax + Bu + L\tilde{d} \quad x \in R^n, \quad u \in R^m, \quad \tilde{d} \in R^q \tag{10.131}$$

where matrices  $A$  and  $B$  are obtained from the computation of the Jacobians

$$A = \left( \begin{array}{ccc} \frac{\partial \tilde{f}_1}{\partial x_1} & \frac{\partial \tilde{f}_1}{\partial x_2} & \dots & \frac{\partial \tilde{f}_1}{\partial x_n} \\ \frac{\partial \tilde{f}_2}{\partial x_1} & \frac{\partial \tilde{f}_2}{\partial x_2} & \dots & \frac{\partial \tilde{f}_2}{\partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial \tilde{f}_n}{\partial x_1} & \frac{\partial \tilde{f}_n}{\partial x_2} & \dots & \frac{\partial \tilde{f}_n}{\partial x_n} \end{array} \right) \Big|_{(x^*, u^*)} \quad B = \left( \begin{array}{ccc} \frac{\partial \tilde{f}_1}{\partial u_1} & \frac{\partial \tilde{f}_1}{\partial u_2} & \dots & \frac{\partial \tilde{f}_1}{\partial u_m} \\ \frac{\partial \tilde{f}_2}{\partial u_1} & \frac{\partial \tilde{f}_2}{\partial u_2} & \dots & \frac{\partial \tilde{f}_2}{\partial u_m} \\ \dots & \dots & \dots & \dots \\ \frac{\partial \tilde{f}_n}{\partial u_1} & \frac{\partial \tilde{f}_n}{\partial u_2} & \dots & \frac{\partial \tilde{f}_n}{\partial u_m} \end{array} \right) \Big|_{(x^*, u^*)} \tag{10.132}$$

and vector  $\tilde{d}$  denotes disturbance terms due to linearization errors. As already analyzed, the problem of disturbance rejection for the linearized model that is described by

$$\begin{aligned}\dot{x} &= Ax + Bu + L\tilde{d} \\ y &= Cx\end{aligned}\tag{10.133}$$

where  $x \in R^n$ ,  $u \in R^m$ ,  $\tilde{d} \in R^q$  and  $y \in R^p$ , cannot be handled efficiently if the classical LQR control scheme is applied. This is because of the existence of the perturbation term  $\tilde{d}$ . The disturbance term  $\tilde{d}$  apart from modeling (parametric) uncertainty and external perturbation terms can also represent noise terms of any distribution.

It has been already noted in the previous application examples of the  $H_\infty$  control approach, that a feedback control scheme is designed for trajectory tracking by the system's state vector and simultaneous disturbance rejection, considering that the disturbance affects the system in the worst possible manner. The disturbances' effect are incorporated in the following quadratic cost function:

$$J(t) = \frac{1}{2} \int_0^T [y^T(t)y(t) + ru^T(t)u(t) - \rho^2 \tilde{d}^T(t)\tilde{d}(t)] dt, \quad r, \rho > 0 \tag{10.134}$$

It has been pointed out that the meaning of the negative sign in the cost function's term that is associated with the perturbation variable  $\tilde{d}(t)$  is that the disturbance tries to maximize the cost function  $J(t)$  while the control signal  $u(t)$  tries to minimize it. The physical meaning of the relation given above is that the control signal and the disturbances compete to each other within a min-max differential game. This problem of min-max optimization can be written as

$$\min_u \max_{\tilde{d}} J(u, \tilde{d}) \tag{10.135}$$

As previously explained, the objective of the optimization procedure is to compute a control signal  $u(t)$  which can compensate for the worst possible disturbance, that is externally imposed to the system. However, the solution to the min-max optimization problem is directly related to the value of parameter  $\rho$ . This means that there is an upper bound in the disturbances magnitude that can be annihilated by the control signal.

### 10.4.3.3 Computation of the Feedback Control Gains

For the linearized system given by Eq. (10.133) the cost function of Eq. (10.134) is defined, where the coefficient  $r$  determines the penalization of the control input and the weight coefficient  $\rho$  determines the reward of the disturbances' effects. It is assumed that (i) The energy that is transferred from the disturbances signal  $\tilde{d}(t)$  is bounded, that is  $\int_0^\infty \tilde{d}^T(t)\tilde{d}(t)dt < \infty$ , (ii) the matrices  $[A, B]$  and  $[A, L]$  are stabilizable, (iii) the matrix  $[A, C]$  is detectable. Then, the optimal feedback control

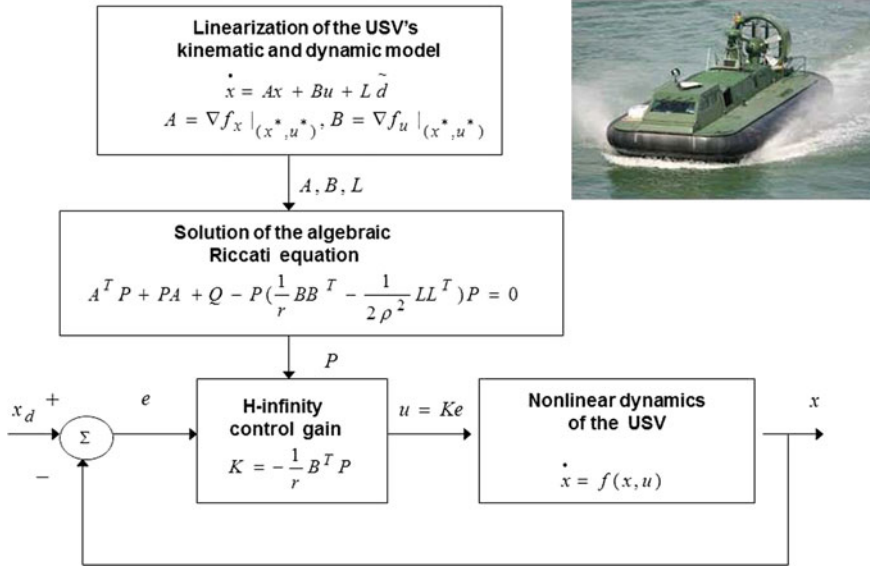


Fig. 10.41 Diagram of the nonlinear H-infinity control scheme for the unmanned surface vessel

law is given by

$$u(t) = -Kx(t) \tag{10.136}$$

with

$$K = \frac{1}{r} B^T P \tag{10.137}$$

where  $P$  is a positive semi-definite symmetric matrix which is obtained from the solution of the Riccati equation

$$A^T P + PA + Q - P \left( \frac{1}{r} BB^T - \frac{1}{2\rho^2} LL^T \right) P = 0 \tag{10.138}$$

where  $Q$  is also a positive definite symmetric matrix. The worst case disturbance is given by  $\tilde{d}(t) = \frac{1}{\rho^2} L^T P x(t)$ . The diagram of the considered control loop is depicted in Fig. 10.41.

### 10.4.4 Lyapunov Stability Analysis

Through Lyapunov stability analysis it will be shown that the proposed nonlinear control scheme assures  $H_\infty$  tracking performance for the USV, and that in case of bounded disturbance terms asymptotic convergence to the reference setpoints is

achieved. The tracking error dynamics for the unmanned surface vessel is written in the form

$$\dot{e} = Ae + Bu + L\tilde{d} \quad (10.139)$$

where in the USV's case  $L = I \in R^{6 \times 6}$  with  $I$  being the identity matrix. Variable  $\tilde{d}$  denotes model uncertainties and external disturbances of the USV's model. The following Lyapunov function is considered

$$V = \frac{1}{2}e^T P e \quad (10.140)$$

where  $e = x - x_d$  is the tracking error. By differentiating with respect to time one obtains

$$\begin{aligned} \dot{V} &= \frac{1}{2}\dot{e}^T P e + \frac{1}{2}e^T P \dot{e} \Rightarrow \\ \dot{V} &= \frac{1}{2}[Ae + Bu + L\tilde{d}]^T P e + \frac{1}{2}e^T P [Ae + Bu + L\tilde{d}] \Rightarrow \end{aligned} \quad (10.141)$$

$$\begin{aligned} \dot{V} &= \frac{1}{2}[e^T A^T + u^T B^T + \tilde{d}^T L^T] P e + \\ &+ \frac{1}{2}e^T P [Ae + Bu + L\tilde{d}] \Rightarrow \end{aligned} \quad (10.142)$$

$$\begin{aligned} \dot{V} &= \frac{1}{2}e^T A^T P e + \frac{1}{2}u^T B^T P e + \frac{1}{2}\tilde{d}^T L^T P e + \\ &\frac{1}{2}e^T P A e + \frac{1}{2}e^T P B u + \frac{1}{2}e^T P L \tilde{d} \end{aligned} \quad (10.143)$$

The previous equation is rewritten as

$$\begin{aligned} \dot{V} &= \frac{1}{2}e^T (A^T P + P A) e + (\frac{1}{2}u^T B^T P e + \frac{1}{2}e^T P B u) + \\ &+ (\frac{1}{2}\tilde{d}^T L^T P e + \frac{1}{2}e^T P L \tilde{d}) \end{aligned} \quad (10.144)$$

*Assumption:* For given positive definite matrix  $Q$  and coefficients  $r$  and  $\rho$  there exists a positive definite matrix  $P$ , which is the solution of the following matrix equation

$$A^T P + P A = -Q + P \left( \frac{2}{r} B B^T - \frac{1}{\rho^2} L L^T \right) P \quad (10.145)$$

Moreover, the following feedback control law is applied to the system

$$u = -\frac{1}{r} B^T P e \quad (10.146)$$

By substituting Eqs. (10.145) and (10.146) one obtains

$$\begin{aligned} \dot{V} &= \frac{1}{2}e^T [-Q + P(\frac{2}{r} B B^T - \frac{1}{\rho^2} L L^T) P] e + \\ &+ e^T P B (-\frac{1}{r} B^T P e) + e^T P L \tilde{d} \Rightarrow \end{aligned} \quad (10.147)$$

$$\begin{aligned} \dot{V} = & -\frac{1}{2}e^T Qe + \frac{1}{r}e^T PBB^T Pe - \frac{1}{2\rho^2}e^T PLL^T Pe \\ & -\frac{1}{r}e^T PBB^T Pe + e^T PL\tilde{d} \end{aligned} \quad (10.148)$$

which after intermediate operations gives

$$\dot{V} = -\frac{1}{2}e^T Qe - \frac{1}{2\rho^2}e^T PLL^T Pe + e^T PL\tilde{d} \quad (10.149)$$

or, equivalently

$$\begin{aligned} \dot{V} = & -\frac{1}{2}e^T Qe - \frac{1}{2\rho^2}e^T PLL^T Pe + \\ & +\frac{1}{2}e^T PL\tilde{d} + \frac{1}{2}\tilde{d}^T L^T Pe \end{aligned} \quad (10.150)$$

*Lemma:* The following inequality holds

$$\frac{1}{2}e^T PL\tilde{d} + \frac{1}{2}\tilde{d}^T L^T Pe - \frac{1}{2\rho^2}e^T PLL^T Pe \leq \frac{1}{2}\rho^2\tilde{d}^T\tilde{d} \quad (10.151)$$

*Proof:* The binomial  $(\rho\alpha - \frac{1}{\rho}b)^2$  is considered. Expanding the left part of the above inequality one gets

$$\begin{aligned} \rho^2 a^2 + \frac{1}{\rho^2} b^2 - 2ab \geq 0 & \Rightarrow \frac{1}{2}\rho^2 a^2 + \frac{1}{2\rho^2} b^2 - ab \geq 0 \Rightarrow \\ ab - \frac{1}{2\rho^2} b^2 \leq \frac{1}{2}\rho^2 a^2 & \Rightarrow \frac{1}{2}ab + \frac{1}{2}ab - \frac{1}{2\rho^2} b^2 \leq \frac{1}{2}\rho^2 a^2 \end{aligned} \quad (10.152)$$

The following substitutions are carried out:  $a = \tilde{d}$  and  $b = e^T PL$  and the previous relation becomes

$$\frac{1}{2}\tilde{d}^T L^T Pe + \frac{1}{2}e^T PL\tilde{d} - \frac{1}{2\rho^2}e^T PLL^T Pe \leq \frac{1}{2}\rho^2\tilde{d}^T\tilde{d} \quad (10.153)$$

Equation (10.153) is substituted in Eq. (10.150) and the inequality is enforced, thus giving

$$\dot{V} \leq -\frac{1}{2}e^T Qe + \frac{1}{2}\rho^2\tilde{d}^T\tilde{d} \quad (10.154)$$

Equation (10.154) shows that the  $H_\infty$  tracking performance criterion is satisfied. The integration of  $\dot{V}$  from 0 to  $T$  gives

$$\begin{aligned} \int_0^T \dot{V}(t)dt & \leq -\frac{1}{2}\int_0^T \|e\|_Q^2 dt + \frac{1}{2}\rho^2 \int_0^T \|\tilde{d}\|^2 dt \Rightarrow \\ 2V(T) + \int_0^T \|e\|_Q^2 dt & \leq 2V(0) + \rho^2 \int_0^T \|\tilde{d}\|^2 dt \end{aligned} \quad (10.155)$$

Moreover, if there exists a positive constant  $M_d > 0$  such that

$$\int_0^{\infty} \|\tilde{d}\|^2 dt \leq M_d \quad (10.156)$$

then one gets

$$\int_0^{\infty} \|e\|_Q^2 dt \leq 2V(0) + \rho^2 M_d \quad (10.157)$$

Thus, the integral  $\int_0^{\infty} \|e\|_Q^2 dt$  is bounded. Moreover,  $V(T)$  is bounded and from the definition of the Lyapunov function  $V$  in Eq. (10.140) it becomes clear that  $e(t)$  will be also bounded since  $e(t) \in \Omega_e = \{e | e^T P e \leq 2V(0) + \rho^2 M_d\}$ . According to the above and with the use of Barbalat's Lemma one obtains  $\lim_{t \rightarrow \infty} e(t) = 0$ .

Elaborating on the above, it can be noted that the proof of global asymptotic stability for the control loop of the unmanned surface vessel relies on Eq. (10.154) and on the application of Barbalat's Lemma. It uses the condition of Eq. (10.156) about the boundedness of the square of the aggregate disturbance and modelling error term  $\tilde{d}$  that affects the model. However, as explained above the proof of global asymptotic stability is not restricted by this condition. By selecting the attenuation coefficient  $\rho$  to be sufficiently small and in particular to satisfy  $\rho^2 < \|e\|_Q^2 / \|\tilde{d}\|^2$  one has that the first derivative of the Lyapunov function is upper bounded by 0. Therefore for the  $i$ th time interval it is proven that the Lyapunov function defined in Eq. (10.140) is a decreasing one. This also assures that the Lyapunov function of the system defined in Eq. (10.140) will always have a negative first-order derivative.

### 10.4.5 Robust State Estimation with the Use of the H-Infinity Kalman Filter

The control loop can be implemented with the use of information provided by a small number of sensors and by processing only a small number of state variables. To reconstruct the missing information about the state vector of the hovercraft it is proposed to use a filtering scheme and based on it to apply state estimation-based control [457]. As previously explained, the recursion of the  $H_{\infty}$  Kalman Filter, for the model of the USV, can be formulated in terms of a *measurement update* and a *time update* part.

*Measurement update:*

$$\begin{aligned} D(k) &= [I - \theta W(k)P^-(k) + C^T(k)R(k)^{-1}C(k)P^-(k)]^{-1} \\ K(k) &= P^-(k)D(k)C^T(k)R(k)^{-1} \\ \hat{x}(k) &= \hat{x}^-(k) + K(k)[y(k) - C\hat{x}^-(k)] \end{aligned} \quad (10.158)$$

*Time update:*

$$\begin{aligned}\hat{x}^-(k+1) &= A(k)x(k) + B(k)u(k) \\ P^-(k+1) &= A(k)P^-(k)D(k)A^T(k) + Q(k)\end{aligned}\tag{10.159}$$

where it is assumed that parameter  $\theta$  is sufficiently small to assure that the covariance matrix  $P^-(k)^{-1} - \theta W(k) + C^T(k)R(k)^{-1}C(k)$  will be positive definite. When  $\theta = 0$  the  $H_\infty$  Kalman Filter becomes equivalent to the standard Kalman Filter. One can measure only a part of the state vector of the USV, and estimate through filtering the rest of the state vector elements. Moreover, the proposed Kalman filtering method can be used for sensor fusion purposes.

### 10.4.6 Simulation Tests

The efficiency of the nonlinear H-infinity control method for the problem of autonomous navigation of the underactuated unmanned surface vessels, was tested in the case of tracking of several reference trajectories. The setpoints have been chosen by taking into account the differential flatness properties of the vessel which have been explained in previous sections. In the autonomous vessel's case the flat outputs of the model are the vessel's cartesian coordinates  $(x, y)$ .

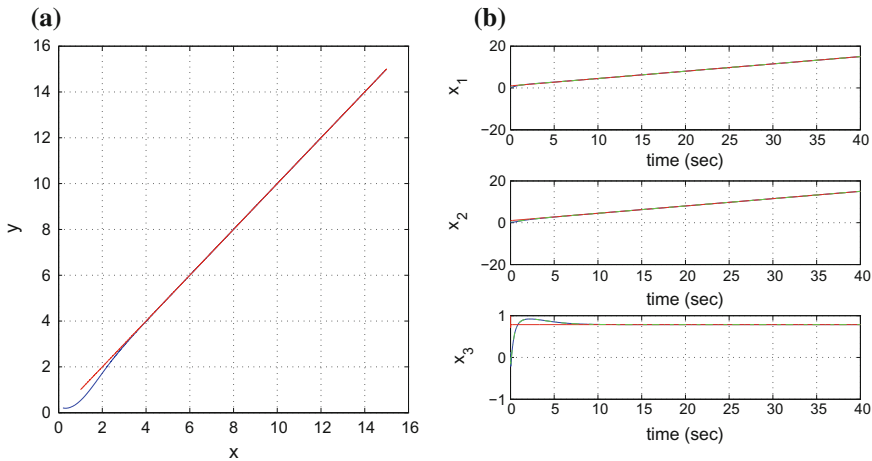
In the presented simulation experiments state estimation-based control has been implemented. Out of the 6 state variables of the USV only the cartesian coordinates of the vessel  $(x, y)$  were considered to be measurable. The rest of the state variables were indirectly estimated with the use of the H-infinity Kalman Filter. The real value of each state variable has been plotted in blue, the estimated value has been plotted in green, while the associated reference setpoint has been plotted in red.

The obtained results are presented in Figs. 10.42, 10.43, 10.44, 10.45, 10.46, 10.47, 10.48, 10.49, 10.50 and 10.51. The state variables of the model have been measured in SI units. It has been confirmed that the proposed control method resulted in fast and accurate tracking of the reference paths. The H-infinity controller assured fast elimination of the tracking error for all state variables of the USV while the variation of the control inputs was smooth and remained within moderate ranges. The method exhibited significant robustness both to the modelling error that was due to the approximate linearization of the USV's dynamics and to parametric changes.

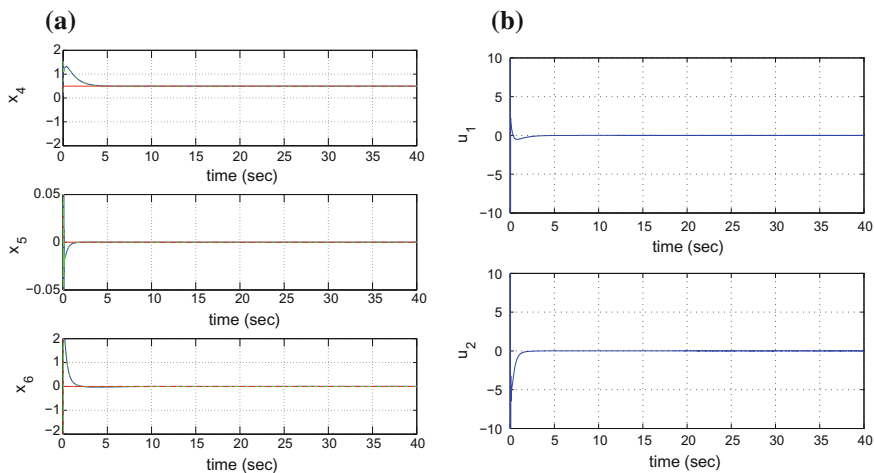
The tracking performance of the nonlinear H-infinity control method for the model of the underactuated vessel (in its state estimation-based implementation) is outlined in Table 10.3:

Apart from remarkable tracking accuracy, the proposed control method exhibits also significant robustness. Even if it is considered that the controller is designed under uncertainty about parameter  $\beta$  of Eq. (10.39) and deviation from the parameter's nominal value ( $\beta = 15$ ), the state variables of the hovercraft converged accurately to the reference setpoint and the tracking error was negligible. The robustness of the control method is defined by the attenuation coefficient  $\rho$ . Actually, maximum

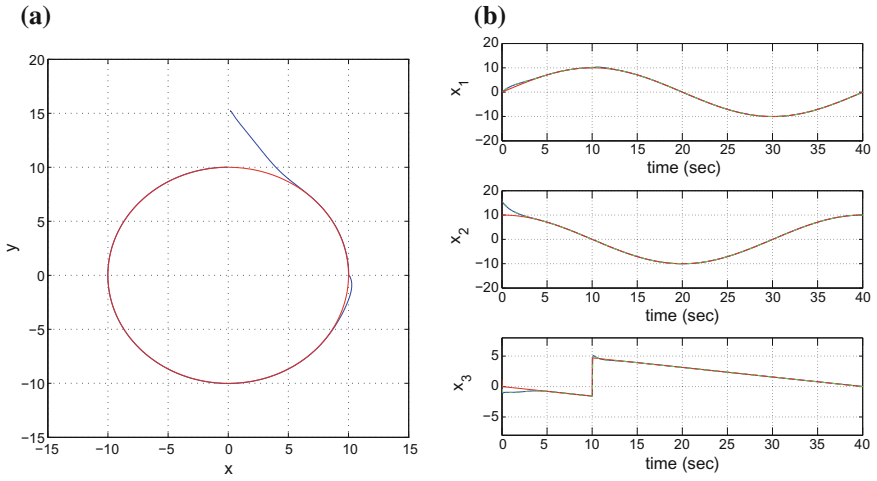




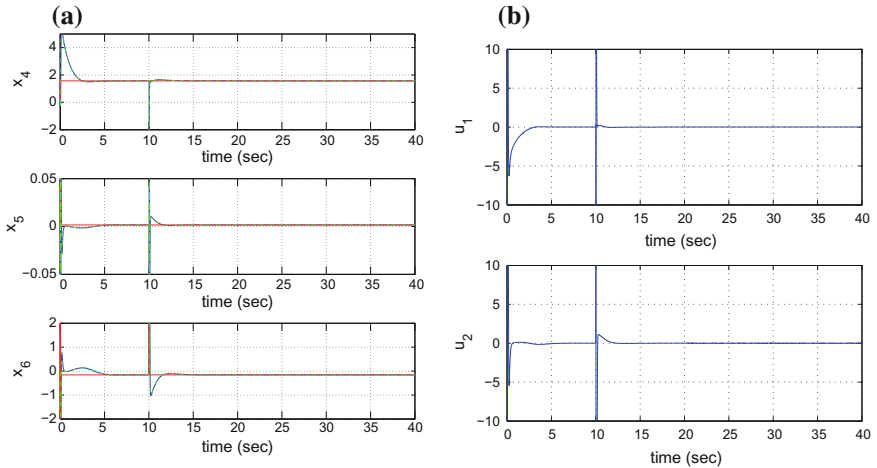
**Fig. 10.42** Reference path 1: **a** Tracking of the reference trajectory (red line) in the  $x - y$  plane by the unmanned surface vessel (blue line), **b** Convergence of the state variables of the vessel  $x_1 = x$ ,  $x_2 = y$  and  $x_3 = \psi$  (blue line) to the associated reference values (red line)



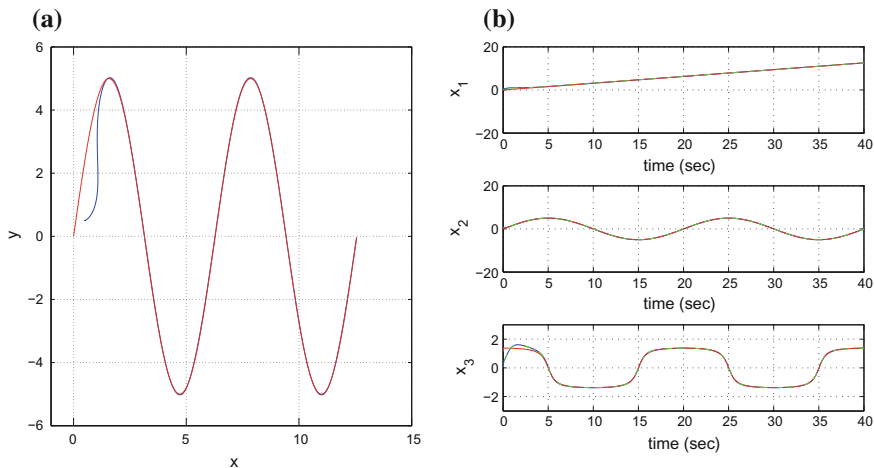
**Fig. 10.43** Reference path 1: **a** Convergence of the state variables of the vessel  $x_4 = u$ ,  $x_5 = v$  and  $x_6 = r$  (blue line) to the associated reference values (red line) **b** Control inputs  $u_1 = \tau_u$  and  $u_2 = \tau_r$  exerted on vessel



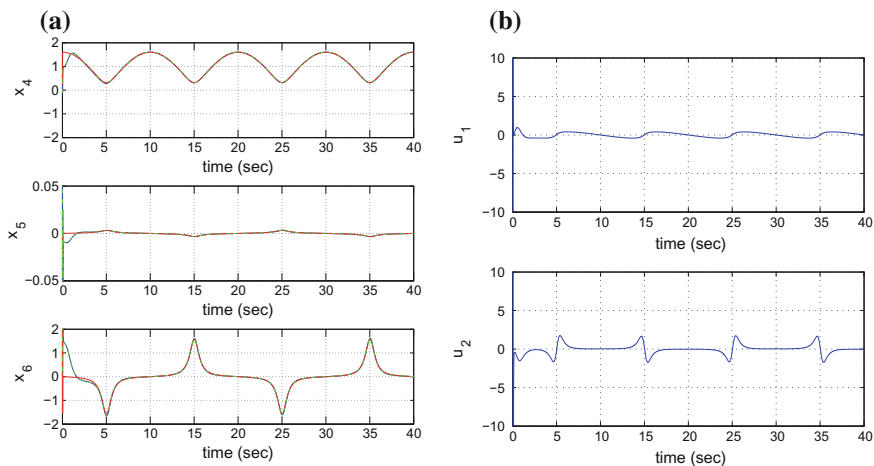
**Fig. 10.44** Reference path 2: **a** Tracking of the reference trajectory (red line) in the  $x - y$  plane by the unmanned surface vessel (blue line), **b** Convergence of the state variables of the vessel  $x_1 = x$ ,  $x_2 = y$  and  $x_3 = \psi$  (blue line) to the associated reference values (red line)



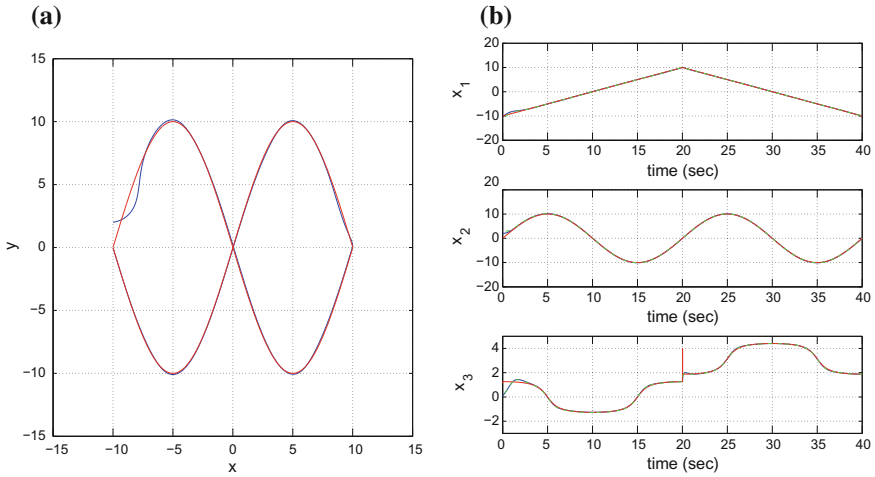
**Fig. 10.45** Reference path 2: **a** Convergence of the state variables of the vessel  $x_4 = u$ ,  $x_5 = v$  and  $x_6 = r$  (blue line) to the associated reference values (red line) **b** Control inputs  $u_1 = \tau_u$  and  $u_2 = \tau_r$  exerted on vessel



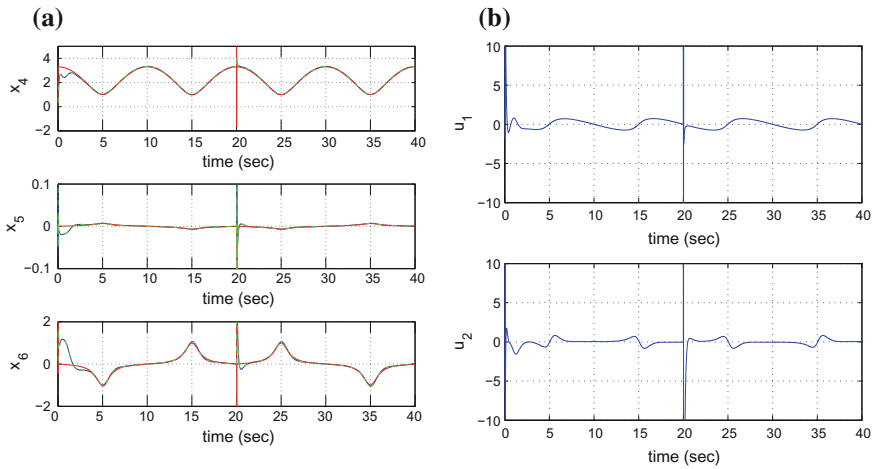
**Fig. 10.46** Reference path 3: **a** Tracking of the reference trajectory (red line) in the  $x - y$  plane by the unmanned surface vessel (blue line), **b** Convergence of the state variables of the vessel  $x_1 = x$ ,  $x_2 = y$  and  $x_3 = \psi$  (blue line) to the associated reference values (red line)



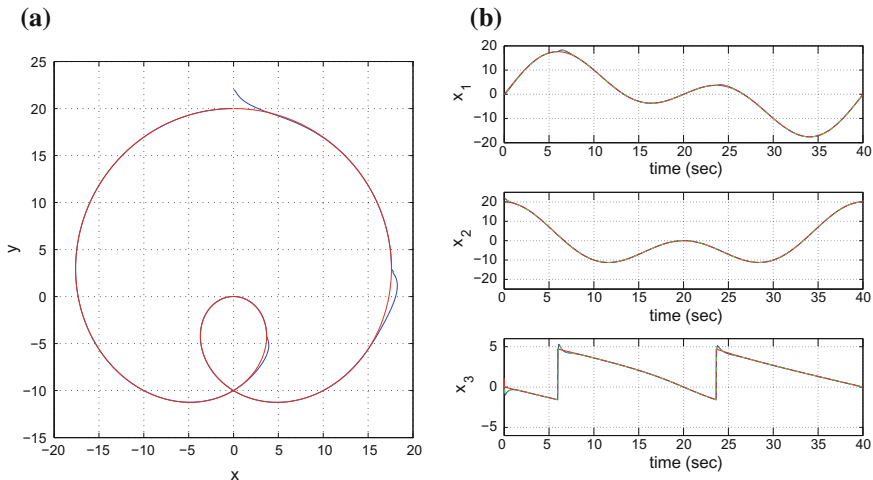
**Fig. 10.47** Reference path 3: **a** Convergence of the state variables of the vessel  $x_4 = u$ ,  $x_5 = v$  and  $x_6 = r$  (blue line) to the associated reference values (red line) **b** Control inputs  $u_1 = \tau_u$  and  $u_2 = \tau_r$  exerted on vessel



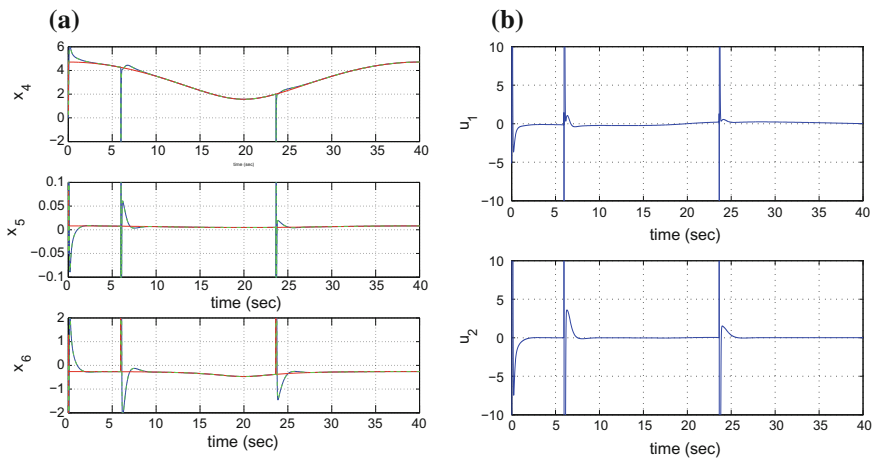
**Fig. 10.48** Reference path 4: **a** Tracking of the reference trajectory (red line) in the  $x - y$  plane by the unmanned surface vessel (blue line), **b** Convergence of the state variables of the vessel  $x_1 = x$ ,  $x_2 = y$  and  $x_3 = \psi$  (blue line) to the associated reference values (red line)



**Fig. 10.49** Reference path 4: **a** Convergence of the state variables of the vessel  $x_4 = u$ ,  $x_5 = v$  and  $x_6 = r$  (blue line) to the associated reference values (red line) **b** Control inputs  $u_1 = \tau_u$  and  $u_2 = \tau_r$  exerted on vessel



**Fig. 10.50** Reference path 5: **a** Tracking of the reference trajectory (red line) in the  $x - y$  plane by the unmanned surface vessel (blue line), **b** Convergence of the state variables of the vessel  $x_1 = x$ ,  $x_2 = y$  and  $x_3 = \psi$  (blue line) to the associated reference values (red line)



**Fig. 10.51** Reference path 5: **a** Convergence of the state variables of the vessel  $x_4 = u$ ,  $x_5 = v$  and  $x_6 = r$  (blue line) to the associated reference values (red line) **b** Control inputs  $u_1 = \tau_u$  and  $u_2 = \tau_r$  exerted on vessel

**Table 10.3** RMSE of the USV's state variables

<i>Parameter</i>	<i>x</i>	<i>y</i>	$\psi$	<i>u</i>	<i>v</i>	<i>r</i>
$RMSE_1$	$0.677 \cdot 10^{-4}$	$0.499 \cdot 10^{-4}$	$0.355 \cdot 10^{-4}$	$0.001 \cdot 10^{-4}$	$0.001 \cdot 10^{-4}$	$0.002 \cdot 10^{-4}$
$RMSE_2$	$2.412 \cdot 10^{-4}$	$4.845 \cdot 10^{-4}$	$0.418 \cdot 10^{-4}$	$0.856 \cdot 10^{-4}$	$0.001 \cdot 10^{-4}$	$0.001 \cdot 10^{-4}$
$RMSE_3$	$4.556 \cdot 10^{-4}$	$5.851 \cdot 10^{-4}$	$5.945 \cdot 10^{-4}$	$2.493 \cdot 10^{-4}$	$0.038 \cdot 10^{-4}$	$9.418 \cdot 10^{-4}$
$RMSE_4$	$3.993 \cdot 10^{-4}$	$8.129 \cdot 10^{-4}$	$3.710 \cdot 10^{-4}$	$3.450 \cdot 10^{-4}$	$0.047 \cdot 10^{-4}$	$4.528 \cdot 10^{-4}$
$RMSE_5$	$2.554 \cdot 10^{-4}$	$1.549 \cdot 10^{-4}$	$0.679 \cdot 10^{-4}$	$1.280 \cdot 10^{-4}$	$0.003 \cdot 10^{-4}$	$0.020 \cdot 10^{-4}$

robustness is achieved for the smallest value of  $\rho$  for which the algebraic Riccati equation given in Eq. (10.145) can be solved.

As previously noted, the joint kinematic-dynamic model of the surface vessel comes primarily from [517, 518]. The considered model is related to a specific type of surface vessels that is hovercrafts. The model of hovercrafts, is obtained from the generic ship model under specific assumptions about the vessel's parameters and this issue has been analyzed in Ref. [416]. However, the section's results can be applied to more types of surface vessels.

Comparing the H-infinity control approach for the underactuated model of the hovercraft against other control methods such as PID control the following conclusion can be reached: PID control is of questionable performance and unsuitable for the problem of autonomous navigation of unmanned surface vessels. PID controllers are usually tuned round local operating points and assuming a linear dynamics for the controlled system. In the case of the unmanned surface vessel the condition about linearity of the vessel's kinematic-dynamic model does not hold. Moreover, in the problem of autonomous navigation the reference setpoints change continuously, therefore one cannot assume that the operating points of the control loop remain unchanged. Consequently, one cannot select gains of the PID controller that assure the reliable functioning of the vessel's control loop, in terms of global asymptotic stability. Additionally, the PID control is vulnerable to external perturbations. In conclusion PID control is not computationally simpler than the proposed control nonlinear H-infinity control method. Besides it cannot assure the stability of the control loop.

## 10.5 Validation of Distributed Kalman Filtering for Ship Tracking Applications

### 10.5.1 Outline

As it has been already pointed out, filtering and control methods of improved accuracy are necessary for developing safe autonomous surface or underwater vessels and reliable maritime transportation systems [222, 271, 555, 625, 631]. By estimating the motion characteristics of a ship through filtering procedures it becomes possible

to implement feedback control for precise trajectory tracking and for avoidance of collisions with nearby vessels [148, 227, 450, 457, 462]. Distributed Kalman filtering is often used for the localization and motion characteristics estimation of ships [189, 234, 309, 329]. Therefore, it is important to develop statistical validation methods for confirming the accuracy of such filtering schemes and for initiating parameters update and corrections in the associated algorithms [30, 33, 58, 107, 170, 221, 430, 652]. To this end, in this section a statistical method is applied for validating the precision of Fuzzy Kalman Filtering, and aiming at a more efficient tracking of the motion of marine vessels.

Fuzzy Kalman Filtering (FKF) is a distributed filtering approach in which a global state estimate is obtained after making use of fuzzy weighting of local estimates provided by distributed Kalman Filters [189, 234, 309, 329]. In the case of autonomous vessels and maritime traffic monitoring, this means that the motion plane of the vessels on the sea surface is covered by spatially distributed Kalman Filters, and that the aggregate estimate of a ship's motion characteristics is computed through a fuzzy averaging procedure. The estimate provided by each local Kalman Filter is attributed a weight which denotes the proximity of a vessel to the center of the area covered by this specific Kalman Filter. To compute an estimate of the ship's state vector each local Kalman Filter makes use of a model of the ship's dynamics. However, the parameters of certain local models may differ from the nominal values of the parameters constituting the real ship dynamics. In such a case inaccurate state estimates are produced, first at a local level, while at a second stage this erroneous state estimates are reflected in the aggregate outcome of the distributed filtering procedure.

The purpose of statistical validation of the Fuzzy Kalman Filter is to detect if the filter provides reliable and precise state estimates about the ship's motion. Moreover, the statistical validation test should detect the local filters which make use of the inaccurate dynamic model of the vessel, thus enabling the update of these models and the removal of errors from the filtering procedure. The Fuzzy Kalman Filter validation method developed in this section is based on the *local statistical approach* to fault diagnosis. To apply the method, it is first shown that local Kalman Filters are equivalent to ARMAX models and next that the Fuzzy Kalman Filter is equivalent to a set of fuzzy weighted ARMAX models [76, 190, 211, 632]. A key element of the proposed validation approach is the Generalized Likelihood Ratio, computed through the processing of the residuals of the estimation procedure (that is the differences between the real and the estimated outputs of the ship's dynamics) [34, 42, 463, 464, 633]. This finally results in the  $\chi^2$  change detection test and enables to define an optimal threshold beyond which the distributed filtering procedure and the indicated local dynamical models of the vessels are considered to be unreliable [32, 35, 624]. The efficiency of the proposed validation scheme for the Fuzzy Kalman Filter is confirmed through simulation experiments, making use of a 6th order model of a surface vessel dynamics. It is shown that the statistical validation test is capable of detecting the faulty local filter, even under small errors in the local model's parameters which do not exceed 1% of the associated nominal values.

## 10.5.2 Dynamic Model of Surface Vessels

### 10.5.2.1 Estimation of the Ship's State Vector

The sensor fusion-based estimation procedure for obtaining the ship's state vector is affected by uncertainties characterizing the ship's dynamic model. Such uncertainties can be due to parametric variations in the model of Eqs. (10.13) and (10.14) or due to external disturbances, e.g. additive input disturbances as shown in Eqs. (10.2) and (10.10).

In the case of a surface vessel, defining the generalized state vector  $x = [\eta, \dot{\eta}]^T$  and considering invariance of the disturbance  $d$  for specific time periods, one obtains the generalized ship state-space model

$$\ddot{\eta} + J(\eta)^{-1}[C(\eta, \dot{\eta}) + F(\eta)]\dot{\eta} - J^{-1}(\eta)d = J^{-1}(\eta)\tau \quad (10.160)$$

Setting  $x_1 = \eta$ ,  $x_2 = d$ ,  $x_3 = \dot{\eta}$ ,  $x_4 = \dot{d}$  and taking into account the existence of process and measurement noise one obtains a ship's model of the form

$$\begin{aligned} \dot{x} &= Ax + Bu + w \\ z &= \gamma(x) + v \end{aligned} \quad (10.161)$$

where matrices  $A$  and  $B$  are given by

$$\begin{aligned} A &= \begin{pmatrix} 0_{3 \times 3} & I_{3 \times 3} \\ 0_{3 \times 3} & -J^{-1}(x)[C(x, \dot{x}) + F(x)] \end{pmatrix} \\ B &= (0_{3 \times 3} \ J^{-1}(x))^T \end{aligned} \quad (10.162)$$

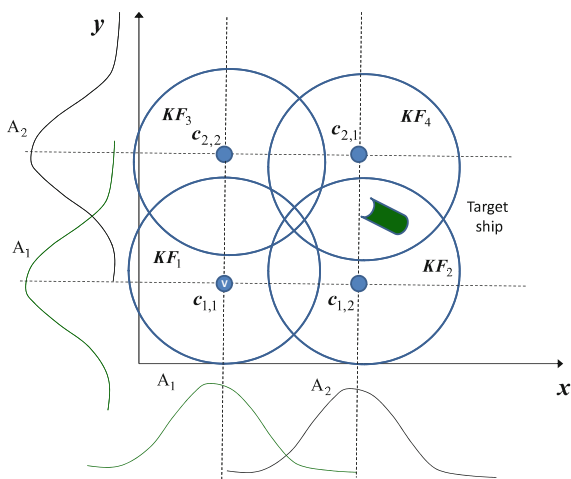
The extended state vector is  $x = [x_1, x_2, x_3, x_4]^T$  with  $x_i \in \mathbb{R}^{3 \times 1}$ ,  $i = 1, 2, 3, 4$ . The control input is  $\tau \in \mathbb{R}^{3 \times 1}$ . The measurement vector of the ship's model is given by  $z = [x, y, \psi]^T$ , where  $x, y$  are measurements of the ship's cartesian coordinates, and  $\psi$  is a measurement of the ship's orientation. The vectors of process and measurement noises are denoted as  $w$  and  $v$ , respectively. Using the above state-space representation, state vector  $x$  can be estimated by processing a sequence of output measurements  $y$  with the use of a state observer or Kalman Filtering [222, 555].

### 10.5.3 Fuzzy Kalman Filtering for Ship Motion Estimation

Fuzzy Kalman Filtering is a distributed filtering approach in which the aggregate state estimate is provided by fuzzy weighting of the estimates generated by local and



**Fig. 10.52** Fuzzy Kalman Filtering for ship tracking applications



spatially distributed Kalman Filters [189, 234, 309, 329]. Here, the sea surface is partitioned into local areas, each one monitored by a different Kalman Filter. The area that each Kalman Filter covers is described by fuzzy rules  $R^l$  of the form:

$$R^l \text{ IF } x \text{ is } A_i \text{ AND } y \text{ is } A_j \text{ THEN } KF^l \text{ estimates } \hat{x}^l \quad (10.163)$$

where  $i = 1, 2, \dots, n, j = 1, 2, \dots, n$  and  $l = 1, 2, \dots, n \times n$ .

Next, it is assumed that the partitioning of the sea surface is as depicted in Fig. 10.52. Then the associated Fuzzy Kalman Filter is described by the following fuzzy rules:

$$\begin{aligned} &\text{IF } x \text{ is } A_1 \text{ AND } y \text{ is } A_1 \text{ THEN } KF_1 \\ &\text{IF } x \text{ is } A_1 \text{ AND } y \text{ is } A_2 \text{ THEN } KF_2 \\ &\text{IF } x \text{ is } A_2 \text{ AND } y \text{ is } A_1 \text{ THEN } KF_3 \\ &\text{IF } x \text{ is } A_2 \text{ AND } y \text{ is } A_2 \text{ THEN } KF_4 \\ &\dots \dots \dots \\ &\text{IF } x \text{ is } A_n \text{ AND } y \text{ is } A_{n-1} \text{ THEN } KF_{n \times n-1} \\ &\text{IF } x \text{ is } A_n \text{ AND } y \text{ is } A_n \text{ THEN } KF_{n \times n} \end{aligned} \quad (10.164)$$

Next, the number of the fuzzy rules is denoted as  $M = n \times n$ . The aggregate estimate that is provided by the fuzzy Kalman Filter is of the form

$$\hat{x} = \sum_{l=1}^M \frac{\prod_{i=1}^N A_i^l}{\sum_{j=1}^M \prod_{i=1}^N A_i^j} \hat{x}_l \Rightarrow \hat{x} = \sum_{l=1}^M w_l \hat{x}_l \quad (10.165)$$

According to the above the ship's model was written in the linear state-space form

$$\begin{aligned}\dot{x} &= Ax + Bu + w \\ y &= Cx + v\end{aligned}\quad (10.166)$$

while after discretization, the discrete-time description of the linearized ship dynamics, in discrete-time, is obtained

$$\begin{aligned}x(k+1) &= A_d x(k) + B_d u(k) + w(k) \\ y(k) &= C_d x(k) + v(k)\end{aligned}\quad (10.167)$$

The  $i$ th Kalman Filter, which is associated with the  $l$ th fuzzy rule is given by

$$\hat{x}^l(k+1) = A_d^l \hat{x}^l(k) + B_d^l u(k) + K_f^l C_d^l (x^l(k)) - \hat{x}^l(k) \quad (10.168)$$

The difference  $\varepsilon(k) = C_d^l (x^l(k)) - \hat{x}^l(k)$  between the real and the estimated output of the vessel's dynamic model, is the residual and follows a zero-mean Gaussian distribution. By applying the  $z$  transformation the equivalent description of the Kalman Filter in the  $z$ -frequency domain can be obtained, which has a MIMO transfer function form

$$\begin{pmatrix} Y_1(z) \\ Y_2(z) \\ Y_3(z) \end{pmatrix} = \begin{pmatrix} H_{11}^A & H_{12}^A & H_{13}^A \\ H_{21}^A & H_{22}^A & H_{23}^A \\ H_{31}^A & H_{32}^A & H_{33}^A \end{pmatrix} \begin{pmatrix} U_1(z) \\ U_2(z) \\ U_3(z) \end{pmatrix} + \begin{pmatrix} H_{11}^B & H_{12}^B & H_{13}^B \\ H_{21}^B & H_{22}^B & H_{23}^B \\ H_{31}^B & H_{32}^B & H_{33}^B \end{pmatrix} \begin{pmatrix} E_1(z) \\ E_2(z) \\ E_3(z) \end{pmatrix} \quad (10.169)$$

Next, examining for instance the subsystem

$$\begin{aligned}Y_1(z)H_{11}^A U_1(z) + H_{12}^A U_2(z) + H_{13}^A U_3(z) + \\ + H_{11}^B E_1(z) + H_{12}^B E_2(z) + H_{13}^B E_3(z)\end{aligned}\quad (10.170)$$

where each one of the transfer functions  $H_{1j}$   $j = 1, 2, 3$  included in the above description has in its denominator the system's characteristic polynomial given by the determinant  $|zI - A_d|$ . Taking into account that this characteristic polynomial is of 6th order one gets the equivalent ARMAX description

$$\begin{aligned}y_1(k) &= \sum_{i=1}^6 a_i y(k-i) + \sum_{j_1=1}^5 b_{j_1} u_1(k-j_1) + \\ &+ \sum_{j_2=1}^5 c_{j_2} u_2(k-j_2) + \sum_{j_3=1}^5 c_{j_3} u_3(k-j_3) + \\ &+ \sum_{m_1=1}^5 p_{m_1} \varepsilon_1(k-m_1) + \sum_{m_2=1}^5 q_{m_2} \varepsilon_2(k-m_2) + \\ &+ \sum_{m_3=1}^5 r_{m_3} \varepsilon_3(k-m_3)\end{aligned}\quad (10.171)$$

where coefficients  $a_i$   $i = 1, 2, 3$  are obtained from the system's characteristic polynomial  $\det|zI - A|$  in descending order. Next, the aforementioned characteristic polynomial is computed. It holds that

$$A = \begin{pmatrix} 0_{3 \times 3} & I_{3 \times 3} \\ 0_{3 \times 3} - J^{-1}x & [C(x, \dot{x}) + F(x)] \end{pmatrix} \quad (10.172)$$

Furthermore, using the definition of matrices  $J$ ,  $C$  and  $F$  given in Eqs. (10.15), (10.16) and (10.17) respectively, and considering that the yaw angle of the ship is 0, that is  $\psi = 0$  and  $\dot{\psi} = 0$  one obtains  $C = 0_{3 \times 3}$  while

$$J = \begin{pmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & m_{23} & 0 \end{pmatrix} \quad F = \begin{pmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & d_{23} \\ 0 & d_{32} & d_{23} \end{pmatrix} \quad (10.173)$$

To compute the system's description in MIMO transfer function form, one has to calculate first the inverse matrix  $(zI - A_d)^{-1}$ . It holds that

$$Q = (zI - A_d)^{-1} = \frac{1}{|zI - A_d|} \begin{pmatrix} Q_{11} & -Q_{21} & Q_{31} & Q_{41} & -Q_{51} & Q_{61} \\ -Q_{12} & Q_{22} & -Q_{32} & -Q_{42} & Q_{52} & -Q_{62} \\ Q_{13} & -Q_{23} & Q_{33} & Q_{43} & -Q_{53} & Q_{63} \\ -Q_{14} & Q_{24} & -Q_{34} & -Q_{44} & Q_{54} & -Q_{64} \\ Q_{15} & -Q_{25} & Q_{35} & Q_{45} & -Q_{55} & Q_{65} \\ -Q_{16} & Q_{26} & -Q_{36} & Q_{46} & -Q_{56} & Q_{66} \end{pmatrix} \quad (10.174)$$

where the elements of the adjoint matrix are

$$Q_{11} = (z-1)^2(z-1-T_s \frac{d_{11}}{m_{11}})[(z-1-T_s v_{55})(z-1-T_s v_{66}) - (-1-T_s v_{65})(-1-T_s v_{56})], \quad Q_{12} = 0, \quad Q_{13} = 0, \quad Q_{14} = 0, \quad Q_{15} = 0 \text{ and } Q_{16} = 0.$$

$$Q_{21} = 0, \quad Q_{22} = (z-1)^2(z-1-T_s \frac{d_{11}}{m_{11}})[(z-1-T_s v_{65})(z-1-T_s v_{66}) - (-1-T_s v_{65})(-1-T_s v_{56})], \quad Q_{23} = 0, \quad Q_{24} = 0, \quad Q_{25} = 0, \text{ and } Q_{26} = 0.$$

$$Q_{31} = 0, \quad Q_{32} = 0, \quad Q_{33} = (z-1)^2(z-1-T_s \frac{d_{11}}{m_{11}})[(z-1-T_s v_{55})(z-1-T_s v_{66}) - (-1-T_s v_{65})(-1-T_s v_{56})], \quad Q_{34} = 0, \quad Q_{35} \text{ and } Q_{36} = 0.$$

$$Q_{41} = (z-1)^2(-T_s)[(z-1-T_s v_{65})(z-1-T_s v_{66}) - (-1-T_s v_{65})(-1-T_s v_{56})], \quad Q_{42} = 0, \quad Q_{43} = 0, \quad Q_{44} = (z-1)^2[(z-1-T_s v_{55})(z-1-T_s v_{66}) - (-1-T_s v_{65})(-1-T_s v_{56})], \quad Q_{45} = 0, \text{ and } Q_{46} = 0.$$

$$Q_{51} = 0, \quad Q_{52} = -T_s(z-1)^2(z-1-T_s \frac{d_{11}}{m_{11}})(z-1-T_s v_{66}), \quad Q_{53} = T_s(z-1)^2(z-1-T_s \frac{d_{11}}{m_{11}})(-1-T_s v_{65}), \quad Q_{54} = (z-1)^3(z-1-T_s \frac{d_{11}}{m_{11}})(z-1-T_s v_{66}), \quad Q_{55} = (z-1)^3(z-1-T_s \frac{d_{11}}{m_{11}})(z-1-T_s v_{66}), \text{ and } Q_{56} = (z-1)^3(z-1-T_s \frac{d_{11}}{m_{11}})(-1-T_s v_{65})$$

$$Q_{61} = 0, \quad Q_{62} = (z-1)^2(z-1-T_s \frac{d_{11}}{m_{11}})(-T_s)(-1-T_s v_{56}), \quad Q_{63} = -(z-1)^2(z-1-T_s \frac{d_{11}}{m_{11}})T_s(z-1-T_s v_{55}), \quad Q_{64} = 0, \quad Q_{65} = (z-1)^3(z-1-T_s \frac{d_{11}}{m_{11}})(-1-T_s v_{56}), \quad Q_{66} = (z-1)^3(z-1-T_s \frac{d_{11}}{m_{11}})(z-1-T_s v_{55}).$$

The computation of the associated characteristic polynomial gives:

$$\begin{aligned}
 \det(zI - A) &= z^6 + [(-2 - T_s v_{66}) - T_s v_{55} + (-4 + T_s \frac{d_{11}}{m_{11}})]z^5 + \\
 &[(T_s v_{66} + T_s v_{65} + T_s^2 v_{55} v_{66} - T_s v_{56} - T_s v_{65} - T_s^2 v_{65} v_{56}) + \\
 &+ (-4 + T_s \frac{d_{11}}{m_{11}})(-2 - T_s v_{66} - T_s v_{55}) + (6 + 3T_s \frac{d_{11}}{m_{11}})]z^4 + \\
 &[(-4 + T_s \frac{d_{11}}{m_{11}})(T_s v_{66} + T_s v_{65} + T_s^2 v_{55} v_{66} - T_s v_{56} - T_s v_{65} - T_s^2 v_{65} v_{56}) + \\
 &+ (6 + 3T_s \frac{d_{11}}{m_{11}})(-2 - T_s v_{66} - T_s v_{55}) + (-4 - 3T_s \frac{d_{11}}{m_{11}})]z^3 + \quad (10.175) \\
 &[(6 + 3T_s \frac{d_{11}}{m_{11}})(T_s v_{66} + T_s v_{65} + T_s^2 v_{55} v_{66} - T_s v_{56} - T_s v_{65} - T_s^2 v_{65} v_{56}) + \\
 &+ (-4 - 3T_s \frac{d_{11}}{m_{11}})(-2 - T_s v_{66} - T_s v_{55}) + (1 + T_s \frac{d_{11}}{m_{11}})]z^2 + \\
 &[(-4 - 3T_s \frac{d_{11}}{m_{11}})(T_s v_{66} + T_s v_{65} + T_s^2 v_{55} v_{66} - T_s v_{56} - T_s v_{65} - T_s^2 v_{65} v_{56}) + \\
 &+ (1 + T_s \frac{d_{11}}{m_{11}})(-2 - T_s v_{66} - T_s v_{55})]z + \\
 &[(1 + T_s \frac{d_{11}}{m_{11}})(T_s v_{66} + T_s v_{65} + T_s^2 v_{55} v_{66} - T_s v_{56} - T_s v_{65} - T_s^2 v_{65} v_{56})]
 \end{aligned}$$

where  $T_s$  is the sampling period and

$$\begin{aligned}
 v_{55} &= \frac{d_{22}m_{11}m_{33} - d_{33}m_{11}m_{23}}{m_{11}(m_{22}m_{33} - m_{23}^2)} & v_{56} &= \frac{d_{23}m_{11}m_{33} + d_{33}m_{11}m_{23}}{m_{11}(m_{22}m_{33} - m_{23}^2)} \\
 v_{65} &= \frac{d_{22}m_{11}m_{23} + d_{33}m_{11}m_{22}}{m_{11}(m_{22}m_{33} - m_{23}^2)} & v_{66} &= \frac{-d_{23}m_{11}m_{23} + d_{33}m_{11}m_{22}}{m_{11}(m_{22}m_{33} - m_{23}^2)}
 \end{aligned} \quad (10.176)$$

Next, to avoid extended computations in the section's example, the ARMAX model of Eq. (10.171) is simplified into the form

$$\begin{aligned}
 y^l(k+1) &= a_1^l y^l(k) + a_2^l y^l(k-1) + a_3^l y^l(k-2) + \\
 &+ b_1^l u_1^l(k) + b_2^l u_2^l(k) + b_3^l u_3^l(k) + \\
 &+ c_1^l \varepsilon_1^l(k)
 \end{aligned} \quad (10.177)$$

where  $l = 1, 2, 3, 4$  is the  $l$ th local model and

$$\begin{aligned}
 a_1^l &= -[(-2 - T_s v_{66}) - T_s v_{55} + (-4 + T_s \frac{d_{11}}{m_{11}})] \\
 a_2^l &= -[(T_s v_{66} + T_s v_{65} + T_s^2 v_{55} v_{66} - T_s v_{56} - T_s v_{65} - T_s^2 v_{65} v_{56}) + \\
 &\quad + (-4 + T_s \frac{d_{11}}{m_{11}})(-2 - T_s v_{66} - T_s v_{55}) + (6 + 3T_s \frac{d_{11}}{m_{11}})] \\
 a_3^l &= -[(-4 + T_s \frac{d_{11}}{m_{11}})(T_s v_{66} + T_s v_{65} + T_s^2 v_{55} v_{66} - T_s v_{56} - T_s v_{65} - T_s^2 v_{65} v_{56}) + \\
 &\quad + (6 + 3T_s \frac{d_{11}}{m_{11}})(-2 - T_s v_{66} - T_s v_{55}) + (-4 - 3T_s \frac{d_{11}}{m_{11}})]
 \end{aligned} \tag{10.178}$$

Moreover, taking into account that the transfer function matrices given in Eq. (10.169), are  $H^A(z) = C_d(zI - A_d)^{-1}B$  and  $H^B(z) = C_d(zI - A_d)^{-1}K_f$  (where  $K_f$  is the gain of the Kalman Filter), and that the measurement matrix for the ship's model is  $C = [I_{3 \times 3} \ 0_{3 \times 3}]$ , it holds

$$\begin{aligned}
 H^A(z) &= \frac{1}{\det(zI - A_d)} \begin{pmatrix} Q_{11} & -Q_{21} & Q_{31} & -Q_{41} & Q_{51} & -Q_{61} \\ -Q_{12} & Q_{22} & -Q_{32} & Q_{42} & -Q_{52} & Q_{62} \\ Q_{13} & -Q_{23} & Q_{33} & -Q_{43} & Q_{53} & -Q_{63} \end{pmatrix} \cdot \\
 &\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -\frac{T_s}{m_{11}} & 0 & 0 \\ 0 & -\frac{T_s m_{33}}{m_{22} m_{33} - m_{23}^2} & \frac{T_s m_{23}}{m_{22} m_{33} - m_{23}^2} \\ 0 & \frac{T_s m_{23}}{m_{22} m_{33} - m_{23}^2} & -\frac{T_s m_{22}}{m_{22} m_{33} - m_{23}^2} \end{pmatrix}
 \end{aligned} \tag{10.179}$$

$$\begin{aligned}
 H^B(z) &= \frac{1}{\det(zI - A_d)} \begin{pmatrix} Q_{11} & -Q_{21} & Q_{31} & -Q_{41} & Q_{51} & -Q_{61} \\ -Q_{12} & Q_{22} & -Q_{32} & Q_{42} & -Q_{52} & Q_{62} \\ Q_{13} & -Q_{23} & Q_{33} & -Q_{43} & Q_{53} & -Q_{63} \end{pmatrix} \begin{pmatrix} K_{f11} & K_{f12} & K_{f13} \\ K_{f21} & K_{f22} & K_{f23} \\ K_{f31} & K_{f32} & K_{f33} \\ K_{f41} & K_{f42} & K_{f43} \\ K_{f51} & K_{f52} & K_{f53} \\ K_{f61} & K_{f62} & K_{f63} \end{pmatrix}
 \end{aligned} \tag{10.180}$$

Then for the rest of the parameters of the local ARMAX model of the Kalman Filter one has that:

$b_1$  is the coefficient multiplying the highest order term of the polynomial  $[T_s Q_{41}]/m_{11}$ ,  $b_2$  is the coefficient multiplying the highest order term of the polynomial

$[-T_3 m_{33} Q_{51} - T_3 m_{23} Q_{61}]/(m_{22} m_{33} - m_{23}^2)$ ,  $b_3$  is the coefficient multiplying the highest order term of the polynomial  $[T_3 m_{23} Q_{51} + T_3 m_{22} Q_{61}]/(m_{22} m_{33} - m_{23}^2)$ , and  $c_1$  is the coefficient multiplying the highest order term of the polynomial  $K_{f_{11}} Q_{11} - K_{f_{21}} Q_{21} + K_{f_{31}} Q_{31} - K_{f_{41}} Q_{41} + K_{f_{51}} Q_{51} - K_{f_{61}} Q_{61}$ .

Using the above and Fig. 10.52 the fuzzy Kalman Filter for the ship tracking problem is described by the following fuzzy rule base:

$$\begin{aligned}
 &\text{IF } x \text{ is } A_1 \text{ AND } y \text{ is } A_1 \text{ THEN} \\
 &\hat{y}^1(k+1) = a_1^1 \hat{y}^1(k) + a_2^1 \hat{y}^1(k-1) + a_3^1 \hat{y}^1(k-2) + \\
 &+ b_1^1 u_1^1(k) + b_2^1 u_2^1(k) + b_3^1 u_3^1(k) + c_1^1 \varepsilon_1^1(k) \\
 &\text{IF } x \text{ is } A_1 \text{ AND } y \text{ is } A_2 \text{ THEN} \\
 &\hat{y}^2(k+1) = a_1^1 \hat{y}^2(k) + a_2^1 \hat{y}^2(k-1) + a_3^2 \hat{y}^2(k-2) + \\
 &+ b_1^1 u_1^2(k) + b_2^2 u_2^2(k) + b_3^2 u_3^2(k) + c_1^2 \varepsilon_1^2(k) \\
 &\text{IF } x \text{ is } A_2 \text{ AND } y \text{ is } A_1 \text{ THEN} \\
 &\hat{y}^3(k+1) = a_1^3 \hat{y}^3(k) + a_2^3 \hat{y}^3(k-1) + a_3^3 \hat{y}^3(k-2) + \\
 &+ b_1^3 u_1^3(k) + b_2^3 u_2^3(k) + b_3^3 u_3^3(k) + c_1^3 \varepsilon_1^3(k) \\
 &\text{IF } x \text{ is } A_2 \text{ AND } y \text{ is } A_2 \text{ THEN} \\
 &\hat{y}^4(k+1) = a_1^4 \hat{y}^4(k) + a_2^4 \hat{y}^4(k-1) + a_3^4 \hat{y}^4(k-2) + \\
 &+ b_1^4 u_1^4(k) + b_2^4 u_2^4(k) + b_3^4 u_3^4(k) + c_1^4 \varepsilon_1^4(k)
 \end{aligned} \tag{10.181}$$

For a properly functioning fuzzy Kalman Filter it should hold  $a_j^1 = a_j^2 = a_j^3 = a_j^4$   $j = 1, 2, 3$  and similarly  $b_j^1 = b_j^2 = b_j^3 = b_j^4$   $j = 1, 2, 3$ , and finally  $c_j^1 = c_j^2 = c_j^3 = c_j^4$   $j = 1$ . If the above condition does not hold then for at least one local Kalman Filter the parameters of the ship model used in the estimation procedure are incorrect. The statistical change detection test which is proposed in this section is capable of detecting the inconsistent local Kalman Filter.

### 10.5.4 Consistency of the Kalman Filter

To obtain accurate estimates with the Kalman Filter, previously a tuning process is required. A question that arises is about which state estimates can be considered as reliable. There is need for systematic methods showing when the Kalman Filter is not performing optimally and when its retuning, either in terms of the used dynamic or kinematic model or in terms of the covariance matrices, should be performed. Several methods can be applied to test the consistency of the Kalman Filter, from the desired characteristics of the measurement residuals. These include the normalized error square (NES) test, the autocorrelation test, and the normalized mean error (NME) test and have been analyzed in [30, 107].

(i) It is assumed that a discrete error process  $e_k$  with dimension  $m \times 1$  is a zero-mean Gaussian white-noise process with covariance given by  $E_k$ . This process can be the Kalman Filter's residual associated to the state estimation error or the residual associated to the measurement estimation error. Then, the following *normalized error square* (NES) is defined

$$\varepsilon_k = e_k^T E_k^{-1} e_k \quad (10.182)$$

The normalized error square follows a  $\chi^2$  distribution. An appropriate test for the normalized error sum is to numerically show that the following condition is met within a level of confidence (according to the properties of the  $\chi^2$  distribution)

$$E\{\varepsilon_k\} = m \quad (10.183)$$

This can be achieved using statistical hypothesis testing, which are associated with confidence intervals. A 95% confidence interval is frequently applied, which is specified using  $100(1 - a)$  with  $a = 0.05$ . Actually, a two-sided probability region is considered cutting-off two end tails of 2.5% each. For  $M$  runs of Monte-Carlo experiments the normalized error square that is obtained is given by

$$\bar{\varepsilon}_k = \frac{1}{M} \sum_{i=1}^M \varepsilon_k(i) = \frac{1}{M} \sum_{i=1}^M e_k^T(i) E_k^{-1}(i) e_k(i) \quad (10.184)$$

where  $\varepsilon_i$  stands for the  $i$ th run at time  $t_k$ . Then  $M\bar{\varepsilon}_k$  will follow a  $\chi^2$  density with  $Mm$  degrees of freedom. This condition can be checked using a  $\chi^2$  test. The hypothesis holds true if the following condition is satisfied

$$\bar{\varepsilon}_k \in [\zeta_1, \zeta_2] \quad (10.185)$$

where  $\zeta_1$  and  $\zeta_2$  are derived from the tail probabilities of the  $\chi^2$  density. For example, for  $m = 2$  and  $M = 100$  one has  $\chi_{Mm}^2(0.025) = 162$  and  $\chi_{Mm}^2(0.975) = 241$ . Using that  $M = 100$  one obtains  $\zeta_1 = \chi_{Mm}^2(0.025)/M = 1.62$  and  $\zeta_2 = \chi_{Mm}^2(0.975)/M = 2.41$ .

(ii) Another consistency checking method is the *test for whiteness*. This is obtained by using the following sample autocorrelation:

$$\bar{\rho}_{k,j} = \frac{1}{\sqrt{M}} \sum_{i=1}^M e_k^T(i) \left[ \sum_{i=1}^M e_k(i) e_k^T(i) \sum_{i=1}^M e_j(i) e_j^T(i) \right]^{-1/2} e_j(i) \quad (10.186)$$

For a sufficiently large value of  $M$ , variable  $\bar{\rho}_{j,k}$  for  $k \neq j$  is zero mean with variance given by  $1/M$ . Next the application of the central limit theorem provides a normal approximation, and considering a 95% confidence interval one finally obtains

$$\bar{\rho}_{j,k} \in \left[ -\frac{1.96}{M}, +\frac{1.96}{M} \right] \quad (10.187)$$

(iii) An additional consistency test is based on the normalized mean error (NME) for the  $j$ th element of  $e_k$

$$[\bar{\mu}_k]_j = \frac{1}{M} \sum_{j=1}^M \frac{[e_k]_j}{\sqrt{[E_k]_{jj}}}, \quad j = 1, 2, \dots, M \quad (10.188)$$

Then, since the variance of  $[\bar{\mu}_k]_j$  is  $\frac{1}{M}$  for a 95% acceptance interval one has

$$[\bar{\mu}_k]_j \in \left[ -\frac{1.96}{\sqrt{M}}, +\frac{1.96}{\sqrt{M}} \right] \quad (10.189)$$

The hypothesis holds true, if Eq. (10.189) is satisfied. The NES, NME and autocorrelation consistency tests can be all performed with a single run using  $N$  data points. Using a time-averaging approach one obtains a low variability test statistic, which can be executed in real-time. In the latter case the time-average NES is given by

$$\bar{\varepsilon} = \frac{1}{N} \sum_{k=1}^N e_k^T E_k^{-1} e_k \quad (10.190)$$

Considering that  $e_k$  is a zero mean, white-noise process, then  $N\bar{\varepsilon}$  follows a  $\chi^2$  density distribution with  $Nm$  degrees of freedom. Through the computation of the time-average auto-correlation the whiteness test for  $e_k$  is

$$\bar{\rho}_j = \frac{1}{\sqrt{N}} \sum_{k=1}^N e_k^T e_{k+j} \left[ \sum_{k=1}^N e_k^T e_k \sum_{k=1}^N e_{k+j}^T e_{k+j} \right]^{-\frac{1}{2}} \quad (10.191)$$

For  $N$  sufficiently large,  $\bar{\rho}_j$  has zero mean and variance given by  $1/N$ . With a 95% acceptance interval one has

$$\bar{\rho}_j \in \left[ -\frac{1.96}{\sqrt{N}}, +\frac{1.96}{\sqrt{N}} \right] \quad (10.192)$$

The hypothesis is accepted if Eq. (10.192) is satisfied. The aforementioned tests can be applied to the residuals of the Kalman Filter or to the Kalman Filter state errors for checking the consistency of the obtained estimation and for checking the necessary consistency for filter optimality. If the tests are not satisfied then this means that the Kalman Filter is not running optimally, and the filter has to be retuned, or the filter's design has to be reconsidered.



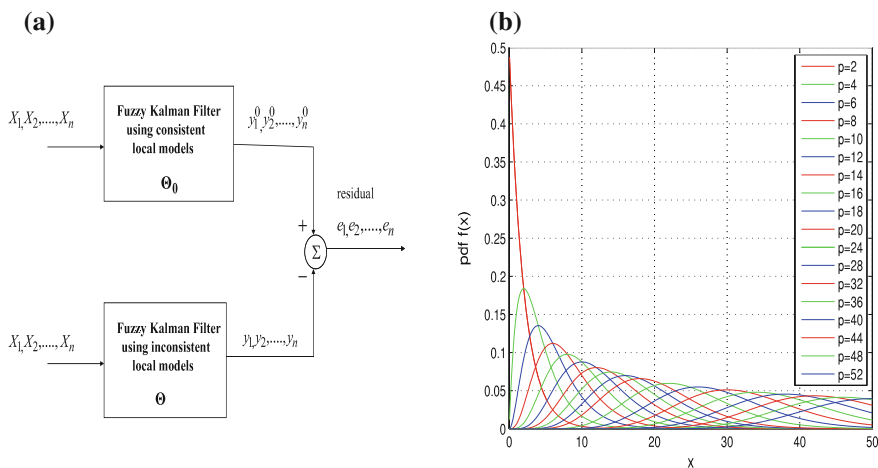
In this section a systematic method, the *local statistical approach to fault diagnosis*, will be introduced for checking the consistency of Fuzzy Kalman Filtering. It will be shown that the method is capable of identifying the elements responsible for the filter’s failure, in the dynamic or kinematic model associated with the estimation performed by the local Kalman filters.

### 10.5.5 Change Detection with the Local Statistical Approach

#### 10.5.5.1 The Global $\chi^2$ Test for Change Detection

The *local statistical approach* to fault diagnosis is a statistical method of fault diagnosis which can be used for consistency checking of the Fuzzy Kalman Filter. Based on a small parametric disturbance assumption, the proposed FDI method aims at transforming complex detection problems concerning a parameterized stochastic process into the problem of monitoring the mean of a Gaussian vector. The local statistical approach consists of two stages: (i) the global test which indicates the existence of a change in some parameters of the fuzzy model, (ii) the diagnostics tests (sensitivity or min–max) which isolate the parameter affected by the change. The method’s stages are analyzed first, following closely the method presented in [33, 633].

As shown in Fig. 10.53 the proposed method relies on the definition of the residual  $e_i$  described as the difference between the output from the nonlinear ARMAX model of the Fuzzy Kalman Filter obtained with the use of the changed dynamics or kinematics of the system and the output of the nonlinear ARMAX model of the Fuzzy



**Fig. 10.53** **a** Residual between the Fuzzy Kalman Filter that uses consistent local models and the Fuzzy Kalman Filter that uses inconsistent (distorted) local models, **b** Probability density function of the  $\chi^2$  distribution, for various degrees of freedom  $p$

Kalman Filter obtained with the use of the unchanged dynamics or kinematics. The nonlinear ARMAX model is actually a neuro-fuzzy model of the Takagi–Sugeno type that is based on the system’s dynamics or kinematics model in an undistorted (fault-free) mode.

The concept of this FDI technique is as follows: there is a nonlinear ARMAX model that represents the unchanged system dynamics. At each time instant the output of the aforementioned reference nonlinear ARMAX model is compared to the output of the nonlinear ARMAX model that represents the changed system dynamics. The difference between these two output measurements is called residual. The statistical processing of a sufficiently large number of residuals through an FDI method provides an index-variable that is compared against a fault threshold’ and which can give early indication about deviation of the model used by the Kalman Filter from the real system dynamics or kinematics. Under certain conditions (detectability of changes) the proposed FDI method enables also fault isolation, i.e. to identify the source of fault within the model used by the Fuzzy Kalman Filter. In practical terms this means that the proposed change detection method can find out the  $i$ th local Kalman Filter (out of the  $N$  local Kalman Filters that constitute the Fuzzy Kalman Filter) which makes use of an inconsistent model of the monitored vessel’s dynamics or kinematics.

Considering the representation of the Fuzzy Kalman Filter as a neuro-fuzzy model of the Takagi–Sugeno type, the partial derivative of the residual square is:

$$H(\theta, y_i) = \frac{1}{2} \frac{\partial e_i^2}{\partial \theta} = e_i \frac{\partial \hat{y}_i}{\partial \theta} \quad (10.193)$$

where  $\theta$  is the vector of model’s parameters. The vector  $H$  having as elements the above  $H(\theta, y_i)$  is called primary residual. Since the nonlinear ARMAX model is a neuro-fuzzy model, the gradient of the output with respect to the consequent parameters  $c_{fi}^l$  is given by

$$\frac{\partial \hat{y}}{\partial c_{fi}^l} = \frac{x_i \mu_{R^l}(x)}{\sum_{l=1}^L \mu_{R^l}(x)} \quad (10.194)$$

The gradient with respect to the center  $c_i^l$  has been given in Eq. (10.217) while the gradient with respect to the spread  $v_i^l$  has been given in Eq. (10.218).

Next, having calculated the partial derivatives of Eqs. (10.216)–(10.218), the rows of the Jacobian matrix  $J$  are found by

$$J(\theta_0, y_k) = \left. \frac{\partial \hat{y}_k(\theta)}{\partial \theta} \right|_{\theta=\theta_0} \quad (10.195)$$

where  $\theta_0$  represents the nominal value of the parameters. The problem of change detection with the  $\chi^2$  test consists of monitoring a change in the mean of the Gaussian variable which for the one-dimensional parameter vector  $\theta$  is formulated as

$$X = \frac{1}{\sqrt{N}} \sum_{i=1}^N e_k \frac{\partial \hat{y}_k}{\partial \theta} \sim N(\mu, \sigma^2) \tag{10.196}$$

where  $\hat{y}_k$  is the output of the neural model generated by the input pattern  $x_k$ ,  $e_k$  is the associated residual and  $\theta$  is the vector of the model's parameters. It is noted that  $X$  is the monitored parameter for the FDI test, which means that when the mean value of  $X$  is 0 the system is in the fault-free condition, while when the mean value of  $X$  has moved away from 0 the system (Kalman Filter) is in a faulty condition. For a multivariable parameter vector  $\theta$  should hold  $X \sim N(M\delta\theta, S)$ , where  $S$  denotes the covariance matrix of  $X$ . In order to decide if the system (Kalman Filter) is in fault-free operating conditions, given a set of data of  $N$  measurements, let  $\theta_*$  be the value of the parameters vector  $\mu$  minimizing the RMSE. The notation is introduced only for the convenience of problem formulation, and its actual value does not need to be known. Then the model validation problem amounts to make a decision between the two hypotheses [33, 633]:

$$\begin{aligned} H_0 : \theta_* &= \theta_0 \\ H_1 : \theta_* &= \theta_0 + \frac{1}{\sqrt{N}}\delta\theta \end{aligned} \tag{10.197}$$

where  $\delta\theta \neq 0$ . It is known from the central limit theorem that for a large data sample, the normalized residual given by Eq. (10.196) asymptotically follows a Gaussian distribution when  $N \rightarrow \infty$  [33, 34, 42]. More specifically, the hypothesis that has to be tested is:

$$\begin{aligned} H_0 : X &\sim N(0, S) \\ H_1 : X &\sim N(M\delta\theta, S) \end{aligned}$$

where  $M$  is the sensitivity matrix (see Eq. (10.198)),  $\delta\theta$  is the change in the parameters' vector and  $S$  is the covariance matrix (see Eq. (10.199)). The product  $M\delta\theta$  denotes the new center of the monitored Gaussian variable  $X$ , after a change on the system's parameter  $\theta$ . The sensitivity matrix  $M$  of  $\frac{1}{\sqrt{N}}X$  is defined as the mean value of the partial derivative with respect to  $\theta$  of the primary residual defined in Eq. (10.215), i.e.  $E\{\frac{\partial}{\partial\theta} H(\theta, y_k)\}$  and is approximated by [33, 35, 624, 633]:

$$M(\theta_0) \simeq \frac{\partial}{\partial\theta} \frac{1}{N} \sum_{k=1}^N H(\theta_0, y_k) \simeq \frac{1}{N} J^T J \tag{10.198}$$

The covariance matrix  $S$  is defined as  $E\{H(\theta, y_k)H^T(\theta, y_{k+m})\}$ ,  $m = 0, \pm 1, \dots$  and is approximated by [32]:

$$\begin{aligned} S &\simeq \sum_{k=1}^N [H(\theta_0, y_k)H^T(\theta_0, y_k)] + \\ &+ \sum_{m=1}^I \frac{1}{N-m} \sum_{k=1}^{N-m} [H(\theta_0, y_k)H^T(\theta_0, y_{k+m}) + \\ &+ H(\theta_0, y_{k+m})H^T(\theta_0, y_k)] \end{aligned} \tag{10.199}$$

where an acceptable value for  $I$  is 3. The decision tool is the likelihood ratio  $s(X) = \ln \frac{p_{\theta_1}(X)}{p_{\theta_0}(X)}$ , where  $p_{\theta_1}(X) = e^{[X - \mu(X)]^T S^{-1} [X - \mu(X)]}$  and  $p_{\theta_0}(X) = e^{X^T S^{-1} X}$  [633]. The center of the Gaussian distribution of the changed system is denoted as  $\mu(X) = M \delta \theta$  where  $\delta \theta$  is the change in the parameters vector. The *Generalized Likelihood Ratio* (GLR) is calculated by maximizing the likelihood ratio with respect to  $\delta \theta$  [32]. This means that the most likely case of parameter change is taken into account. This gives the global  $\chi^2$  test  $t$ :

$$t = X^T S^{-1} M (M^T S^{-1} M)^{-1} M^T S^{-1} X \quad (10.200)$$

Since  $X$  asymptotically follows a Gaussian distribution, the statistics defined in Eq. (10.200) follows a  $\chi^2$  distribution with  $n$  degrees of freedom. Mapping the change detection problem to this  $\chi^2$  distribution enables the choice of the change threshold. Assume that the desired probability of false alarm is  $\alpha$  then the change threshold  $\lambda$  should be chosen from the relation [33, 633]

$$\int_{\lambda}^{\infty} \chi_n^2(s) ds = \alpha, \quad (10.201)$$

where  $\chi_n^2(s)$  is the probability density function (p.d.f.) of a variable that follows the  $\chi^2$  distribution with  $n$  degrees of freedom.

### 10.5.5.2 Statistical Fault Isolation with the Sensitivity Test

Fault isolation is needed to identify the source of faults in the dynamic or kinematic model of the system used by the Fuzzy Kalman Filter. This means that the fault diagnosis method should also be able to find out (among the  $N$  local Kalman Filters that constitute the Fuzzy Kalman Filter) which is the local Kalman Filter that makes use of an inconsistent model. A first approach to change isolation is to focus only on a subset of the parameters while considering that the rest of the parameters unchanged [32]. The parameters vector  $\eta$  can be written as  $\eta = [\phi, \psi]^T$ , where  $\phi$  contains those parameters to be subject to the isolation test, while  $\psi$  contains those parameters to be excluded from the isolation test.  $M_{\phi}$  contains the columns of the sensitivity matrix  $M$  which are associated with the parameters subject to the isolation test. Similarly  $M_{\psi}$  contains the columns of  $M$  that are associated with the parameters to be excluded from the sensitivity test.

Assume that among the parameters  $\eta$ , it is only the subset  $\phi$  that is suspected to have undergone a change. Thus  $\eta$  is restricted to  $\eta = [\phi, 0]^T$ . The associated columns of the sensitivity matrix are given by  $M_{\phi}$  and the mean of the Gaussian to be monitored is  $\mu = M_{\phi} \phi$ , i.e.

$$\mu = M A \phi, \quad A = [0, I]^T \quad (10.202)$$

Matrix  $A$  is used to select the parameters that will be subject to the fault isolation test. The rows of  $A$  correspond to the total set of parameters while the columns of  $A$  correspond only to the parameters selected for the test. Thus the fault diagnosis  $\chi^2$  test (sensitivity test) of Eq. (10.200) can be restated as [33, 633]:

$$t_\phi = X^T S^{-1} M_\phi (M_\phi^T S^{-1} M_\phi)^{-1} M_\phi^T S^{-1} X \quad (10.203)$$

### 10.5.5.3 Statistical Fault Isolation with the Min–Max Test

In this approach the aim is to find a statistic that will be able to detect a change on the part  $\phi$  of the parameters vector  $\eta$  and which will be robust to a change in the non observed part  $\psi$  [32]. Assume the vector partition  $\eta = [\phi, \psi]^T$ . The following notation is used:

$$M^T S^{-1} M = \begin{pmatrix} I_{\phi\phi} & I_{\phi\psi} \\ I_{\psi\phi} & I_{\psi\psi} \end{pmatrix} \quad (10.204)$$

$$\gamma = \begin{pmatrix} \phi \\ \psi \end{pmatrix}^T \cdot \begin{pmatrix} I_{\phi\phi} & I_{\phi\psi} \\ I_{\psi\phi} & I_{\psi\psi} \end{pmatrix} \cdot \begin{pmatrix} \phi \\ \psi \end{pmatrix} \quad (10.205)$$

where  $S$  is the previously defined covariance matrix. The min–max test aims to minimize the non-centrality parameter  $\gamma$  with respect to the parameters that are not suspected for change. The minimum of  $\gamma$  with respect to  $\psi$  is given for [33, 463, 633]:

$$\psi^* = \arg \min_{\psi} \gamma = \varphi^T (I_{\phi\phi} - I_{\phi\psi} I_{\psi\psi}^{-1} I_{\psi\phi}) \varphi \quad (10.206)$$

and is found to be

$$\begin{aligned} \gamma^* &= \min_{\psi} \gamma = \varphi^T (I_{\phi\phi} - I_{\phi\psi} I_{\psi\psi}^{-1} I_{\psi\phi}) \varphi = \\ &= \begin{pmatrix} \varphi \\ -I_{\psi\psi}^{-1} I_{\psi\phi} \varphi \end{pmatrix}^T \begin{pmatrix} I_{\phi\phi} & I_{\phi\psi} \\ I_{\psi\phi} & I_{\psi\psi} \end{pmatrix} \begin{pmatrix} \varphi \\ -I_{\psi\psi}^{-1} I_{\psi\phi} \varphi \end{pmatrix} \end{aligned} \quad (10.207)$$

which results in

$$\gamma^* = \varphi^T \{ [I, -I_{\phi\psi} I_{\psi\psi}^{-1}] M^T \Sigma^{-1} \} \Sigma^{-1} \{ \Sigma^{-1} M [I, -I_{\phi\psi} I_{\psi\psi}^{-1}] \} \varphi \quad (10.208)$$

The following linear transformation of the observations is considered:

$$X_\phi^* = [I, -I_{\phi\psi} I_{\psi\psi}^{-1}] M^T \Sigma^{-1} X \quad (10.209)$$

The transformed variable  $X_\phi^*$  follows a Gaussian distribution  $N(\mu_\phi^*, I_\phi^*)$  with mean:

**Table 10.4** Stages of the local statistical approach for FDI

1.	Generate the residuals partial derivative given by Eq. (10.215)
2.	Calculate the Jacobian matrix J given by Eq. (10.195)
3.	Calculate the sensitivity matrix M given by Eq. (10.198)
4.	Calculate the covariance matrix S given by Eq. (10.199)
5.	Apply the $\chi^2$ test for change detection of Eq. (10.200)
6.	Apply the change isolation tests of Eq. (10.203) or (10.212)

$$\mu_\varphi^* = I_\varphi^* \varphi \tag{10.210}$$

and with covariance:

$$I_\varphi^* = I_{\varphi\varphi} - I_{\varphi\psi} I_{\psi\psi}^{-1} I_{\psi\varphi} \tag{10.211}$$

The min–max test decides between the hypotheses:

$$H_0^* : \mu^* = 0$$

$$H_1^* : \mu^* = I_\varphi^* \varphi$$

and is described by:

$$\tau_\varphi^* = X_\varphi^{*T} I_\varphi^{*-1} X_\varphi^* \tag{10.212}$$

The stages of fault detection and isolation (FDI) with the use of the local statistical approach are summarized in Table 10.4.

### 10.5.5.4 Sensitivity of the Fuzzy Kalman Filter to Parametric Changes

It was shown that the Fuzzy Kalman Filter can be represented as a fuzzy weighting of ARMAX models, which is actually the so-called Takagi–Sugeno fuzzy model. These are written as:

$$R_l : IF x_1 \text{ is } A_1^l \text{ AND } x_2 \text{ is } A_2^l \text{ AND } \dots \text{ AND } x_n \text{ is } A_n^l$$

$$THEN \bar{y}^l = \sum_{i=1}^n c_{f_i}^l x_i \quad l = 1, 2, \dots, L \tag{10.213}$$

where  $R^l$  is the  $l$ th rule,  $x = [x_1, x_2, \dots, x_n]^T$  is the input (antecedent) variable,  $\bar{y}^l$  is the output (consequent) variable, and  $w_i^l, b^l$  are the parameters of the local linear models. The output of the Takagi–Sugeno model is given by the weighted average of the rules consequents [211, 463]:

$$\hat{y} = \frac{\sum_{l=1}^L \bar{y}^l \prod_{i=1}^n \mu_{A_i^l}(x_i)}{\sum_{l=1}^L \prod_{i=1}^n \mu_{A_i^l}(x_i)} \quad (10.214)$$

where  $\mu_{A_i^l}(x_i) : R \rightarrow [0, 1]$  is the membership function of the fuzzy set  $A_i^l$  in the antecedent part of the rule  $R^l$ . The output of the  $l$ th local model is given by  $\bar{y}^l = \sum_{i=1}^n c_{f_i}^l x_i$  [211, 463].

First, the residual  $e_i$  is defined as the difference between the fuzzy model output  $\hat{y}_i$  and the physical system output  $y_i$ , i.e.  $e_i = \hat{y}_i - y_i$ . It is also acceptable to define the residual as the difference between the fuzzy model output and the exact model output, where the exact model replaces the physical system and has the same number of parameters as the fuzzy model (see Fig. 10.53). The partial derivative of the residual square is:

$$H(\theta, y_i) = \frac{1}{2} \frac{\partial e_i^2}{\partial \theta} = e_i \frac{\partial y_i}{\partial \theta} \quad (10.215)$$

The vector  $H$  having as elements the above  $H(\theta, y_i)$  is called primary residual. Next, the gradients of the output with respect to the model's parameters are computed [463]. In the case of fuzzy models the gradient of the output with respect to the consequent parameters  $w_i^l$  is given by

$$\frac{\partial \hat{y}}{\partial w_i^l} = \frac{x_i \mu_{R^l}(x)}{\sum_{l=1}^L \mu_{R^l}(x)} \quad (10.216)$$

The gradient with respect to the center  $c_i^l$  is

$$\frac{\partial \hat{y}}{\partial c_i^l} = \sum_{l=1}^L \frac{\bar{y}^l \frac{2(x_i - c_i^l)}{v_i^l} \mu_{R^l}(x_i) [\sum_{j=1}^L \mu_{R^j}(x_i) - \mu_{R^l}(x_i)]}{[\sum_{l=1}^L \mu_{R^l}(x_i)]^2} \quad (10.217)$$

The gradient with respect to the spread  $v_i^l$  is

$$\frac{\partial \hat{y}}{\partial v_i^l} = \sum_{l=1}^L \frac{\bar{y}^l \frac{2(x_i - c_i^l)^2}{v_i^{l3}} \mu_{R^l}(x_i) [\sum_{j=1}^L \mu_{R^j}(x_i) - \mu_{R^l}(x_i)]}{[\sum_{l=1}^L \mu_{R^l}(x_i)]^2} \quad (10.218)$$

It is noted that the equivalence between the fuzzy Kalman filter and a Takagi–Sugeno neurofuzzy model enables to exploit previous results on fault detection and isolation for non-parametric estimators, such as neurofuzzy networks, by making use the *local statistical approach* to fault diagnosis. By describing the Fuzzy Kalman filter in the form of a Takagi–Sugeno neurofuzzy model it becomes easy to complete the intermediate stages for the application of the change detection method, which are described in Table 10.4 [32, 463].

### 10.5.6 Simulation Tests

The motion of the ship was considered to be monitored by the Fuzzy Kalman Filter of Fig. 10.52. The fuzzy Kalman Filter through local linear models was taken to consist of the following rules:

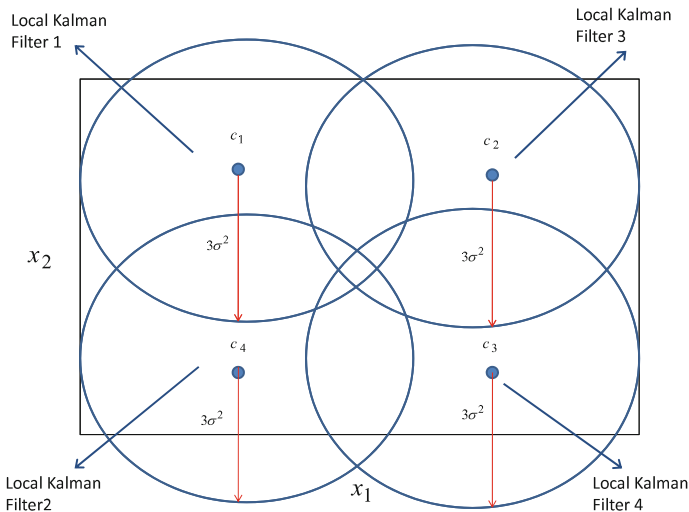
$$\begin{aligned}
 R^{(1)} &: \text{IF } x_1 \text{ is } (c_1^{(1)}, v) \text{ AND } x_2 \text{ is } (c_2^{(1)}, v) \text{ AND } \dots \\
 &\quad \text{AND } x_n \text{ is } (c_n^{(1)}, v) \dots \text{ THEN } \hat{y} = \underline{c}_f^{(1)} \underline{x}^T \\
 R^{(2)} &: \text{IF } x_1 \text{ is } (c_1^{(2)}, v) \text{ AND } x_2 \text{ is } (c_2^{(2)}, v) \text{ AND } \dots \\
 &\quad \text{AND } x_n \text{ is } (c_n^{(2)}, v) \dots \text{ THEN } \hat{y} = \underline{c}_f^{(2)} \underline{x}^T \\
 R^{(3)} &: \text{IF } x_1 \text{ is } (c_1^{(3)}, v) \text{ AND } x_2 \text{ is } (c_2^{(3)}, v) \text{ AND } \dots \\
 &\quad \text{AND } x_n \text{ is } (c_n^{(3)}, v) \dots \text{ THEN } \hat{y} = \underline{c}_f^{(3)} \underline{x}^T \\
 R^{(4)} &: \text{IF } x_1 \text{ is } (c_1^{(4)}, v) \text{ AND } x_2 \text{ is } (c_2^{(4)}, v) \text{ AND } \dots \\
 &\quad \text{AND } x_n \text{ is } (c_n^{(4)}, v) \dots \text{ THEN } \hat{y} = \underline{c}_f^{(4)} \underline{x}^T
 \end{aligned}$$

According to Sect. 10.5.3, the regressor's vector appearing in the consequent part of the previous fuzzy rules is  $x^T = [\hat{y}(k), \hat{y}(k-1), \hat{y}(k-2), u_1(k), u_2(k), u_3(k), \varepsilon_1(k)]^T$ , while the parameters' vector is  $c_f = [a_1, a_2, a_3, b_1, b_2, b_3, c_1]$ . The above model implies fusion of local estimates from 4 sub-models. The spread of the membership functions is denoted by  $v$ . A 2D projection of the input space partition is demonstrated in Fig. 10.54.

As mentioned before, to reduce the number of parameters in the statistical validation test, only the first three variables were maintained in the AR part of the local ARMAX models, that is  $y(k-1)$ ,  $y(k-2)$  and  $y(k-3)$ . Thus, the parameters set in the new TSK fuzzy model consisted of  $4 \times 7 + 4 \times 3 = 40$  parameters (28 linear parameters which were the output layer weights and 12 nonlinear parameters which were the centers of the fuzzy sets in the antecedent part of the rules). This means that by applying the local statistical approach to FDI and the  $\chi^2$  change detection test to the considered model, the fault threshold should be equal to 40.

The numerical tests confirmed theory. In case that no fault was assumed for the monitored system the mean value of the  $\chi^2$  test over a number of trials was found to be close to the threshold value 40. Such a value was anticipated according to the theoretical analysis of the  $\chi^2$  test. For slight deviations of the parameters of the fuzzy Kalman Filter from their nominal (fault-free) values, the global  $\chi^2$  test was capable of giving a clear indication about the existence of a fault. Thus for changes which varied between 0.1% and 1% of the nominal parameter's value the score of the  $\chi^2$  test deviated significantly from the fault threshold (which as mentioned before was set equal to 40).



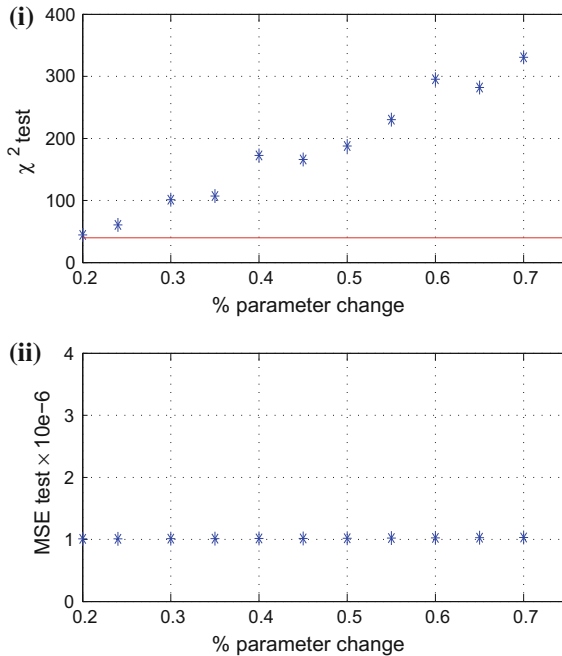


**Fig. 10.54** Fuzzy rule base generated with input space partition

**Table 10.5** Comparison between  $\chi^2$  and MSE tests

% change	$\chi^2$	MSE
0.20	44.45	$1.01 \cdot 10^{-6}$
0.24	60.54	$1.01 \cdot 10^{-6}$
0.30	101.21	$1.01 \cdot 10^{-6}$
0.35	106.99	$1.01 \cdot 10^{-6}$
0.40	172.49	$1.01 \cdot 10^{-6}$
0.45	165.97	$1.02 \cdot 10^{-6}$
0.50	187.87	$1.02 \cdot 10^{-6}$
0.55	230.14	$1.02 \cdot 10^{-6}$
0.60	295.33	$1.02 \cdot 10^{-6}$
0.65	282.00	$1.03 \cdot 10^{-6}$
0.70	330.59	$1.03 \cdot 10^{-6}$
0.75	365.88	$1.03 \cdot 10^{-6}$

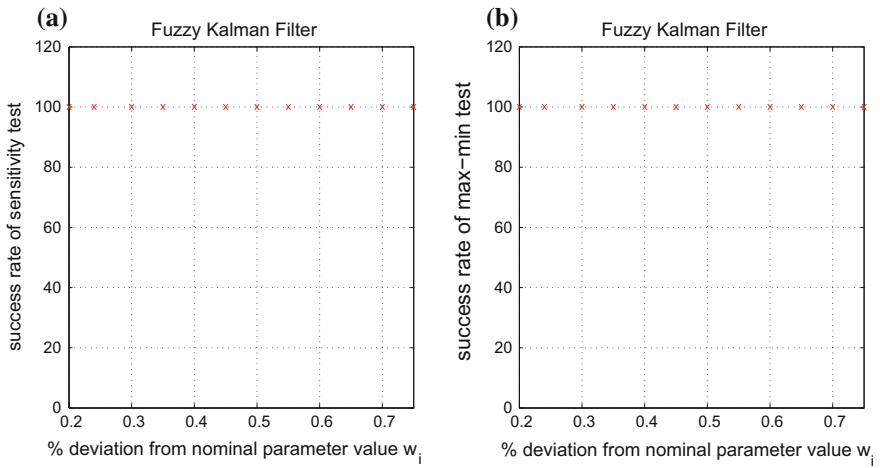
A comparison between (i) the proposed  $\chi^2$  change detection test based on the local statistical approach and the Generalized Likelihood ratio and (ii) the mean square error (MSE) test, for detecting model inconsistencies in the distributed/fuzzy Kalman Filter is given in Table 10.5 and in Fig. 10.55. It can be clearly noticed that for small parametric changes in the ship’s local models used by the fuzzy Kalman Filter, the MSE test gives the erroneous conclusion that the functioning of the Kalman Filter remains accurate. Actually it is observed that there is no change in the MSE value despite changes in the parameters of the model used by the Kalman Filter, and the MSE value remains low as in the case of fault-free operation. Besides in the MSE test



**Fig. 10.55** Comparison between **i** the proposed  $\chi^2$  test based on the local statistical approach and the Generalized Likelihood ratio and **ii** of the mean square error (MSE) test, for detecting model inconsistencies in the distributed/Fuzzy Kalman Filter

the fault threshold is defined in an ad-hoc manner and this is another reason for the low credibility of this test. On the other hand the proposed  $\chi^2$  test based on the local statistical approach and the Generalized Likelihood ratio provides a clear indication about inconsistencies between the models used by the fuzzy Kalman Filter and the dynamics of the real system. Despite the small magnitude of parametric changes, the output of the  $\chi^2$  test based on the local statistical approach becomes several times larger than the fault threshold (that is 40). Thus a clear indication is provided about the need to correct the parameters of the local models used by the Fuzzy Kalman Filter.

As far as fault isolation is concerned, the numerical results showed that the sensitivity method for fault isolation was very efficient in distinguishing the parameter subject to fault among all parameters in the fuzzy Kalman Filter's model. The sensitivity fault isolation test and the min-max fault isolation test was performed for the parameters (weight  $w_i$ ) of the local Kalman Filter. As it can be observed from the test's success rate depicted in Fig. 10.56 the proposed fault isolation methods can detect the local Kalman Filter, that uses an inconsistent model with reference to the



**Fig. 10.56** **a** Success rate of the fault isolation test (sensitivity method) for changes in a parameter of the first local Kalman Filter, ranging between 0.1 and 1.0% of the nominal value **b** Success rate of the fault isolation test (max–min method) for changes in a parameter of the first local Kalman Filter, ranging between 0.1 and 1.0% of the nominal value

real system’s model. Thus correction of the parameters of this particular filter can be carried out instead of redesign of all local Kalman Filters constituting the Fuzzy Kalman Filter.