Chapter 6
Differential Flatness Theory in Mobile Robotics and Autonomous Vehicles

6.1 Outline

The chapter proposes differential flatness theory-based methods of filtering and control for MIMO nonlinear dynamical systems, such as unmanned vehicles. These can be of different types such as Unmanned Ground Vehicles (UGVs), Unmanned Surface Vessels (USVs), Autonomous Underwater Vessels (AUVs), and Unmanned Aerial Vehicles (UAVs). The considered nonlinear filtering schemes which are based on differential flatness theory can be applied to autonomous vehicle models without the need for calculation of Jacobian matrices. Nonlinear systems such as unmanned ground vehicles, satisfying the differential flatness property, can be written in the Brunovsky (canonical) form via a transformation of their state variables and control inputs. After transforming the nonlinear system to the canonical form, it is straightforward to apply for it the standard Kalman Filter recursion. The performance of the proposed derivative-free nonlinear filtering scheme is tested through simulation experiments on the problem of state estimation-based control for autonomous navigation of unmanned ground vehicles.

First, the chapter proposes state estimation-based control for cooperating mobile robots, which have the kinematic model of a unicycle. To estimate with accuracy the position of these robotic vehicles as well as their motion characteristics, fusion of estimates from multiple sensors is performed with the use of the Derivative-free distributed nonlinear Kalman Filter. The proposed derivative-free nonlinear filtering method enables distributed state estimation, by substituting the Extended Information Filter with the standard Information Filter recursion. This filtering approach has significant advantages because, unlike the Extended Information Filter, it is not based on local linearization of the nonlinear dynamics and computation of Jacobian matrices. The proposed nonlinear control is in accordance with the principles of differential flatness theory. The performance of the considered distributed filtering-based control is tested through simulation experiments on the problem of autonomous navigation of automatic ground vehicles (AGVs) under a master-slave scheme.

G.G. Rigatos, *Nonlinear Control and Filtering Using Differential Flatness Approaches*, Studies in Systems, Decision and Control 25,
DOI 10.1007/978-3-319-16420-5_6

Additionally, the chapter proposes the Derivative-free distributed nonlinear Kalman Filter, for integrity monitoring of navigation sensors in automatic ground vehicles (AGV). Unlike the Extended Information Filter (EIF), the proposed filter avoids approximation errors caused by the linearization of the AGV kinematic model and does not require the computation of Jacobians. The use of a statistical fault detection and isolation algorithm for processing the residuals generated by the proposed filtering method, can provide an indication about the condition of the navigation sensors and about failures that may have appeared. As an application example, the chapter considers failure diagnosis for wheel encoders or IMU devices of an AGV.

Next, the chapter explains how controller design for autonomous 4-wheeled ground vehicles can be performed using differential flatness theory. Using a 3-DOF nonlinear model of the vehicle's dynamics and through the application of differential flatness theory, an equivalent model in linear canonical (Brunovksy) form is obtained. For the latter model, a state feedback controller is developed that enables accurate tracking of velocity setpoints. Moreover, it is shown that with the use of Kalman Filtering it is possible to dynamically estimate the disturbances due to unknown forces and torques exerted on the vehicle. The processing of velocity measurements (provided by a small number of onboard sensors) through a Kalman Filter which has been redesigned in the form of a disturbance observer results in accurate identification of external disturbances affecting the vehicle's dynamic model. By including in the vehicle's controller an additional term that compensates for the estimated disturbance forces, the vehicle's motion characteristics remain unchanged. Numerical simulation confirms the efficiency of both the proposed controller and of the disturbance forces estimator.

Moreover, in this chapter a solution to the problem of active control and disturbances compensation in vehicles' suspensions is proposed. It is shown that the suspension model satisfies differential flatness properties and the associated flat output is a weighted sum of the system's state vector elements. Differential flatness of the suspension's model enables transformation into a linear canonical form for which it is possible to design a state feedback controller. Kalman filtering is used for estimating the nonmeasurable elements of the suspension's transformed state vector through the processing of measurements provided by a small number of onboard sensors. Moreover, by reformulating the Kalman Filter as a disturbance observer it is possible to simultaneously estimate the external disturbances and the system's transformed state vector. The inclusion of an additional control term based on the disturbances estimation enables to compensate for the disturbances' effects and to attenuate vibrations. The performance of the proposed Kalman Filter-based active suspension control scheme has been tested through numerical simulation experiments.

Another topic treated by the chapter is the use of the Derivative-free nonlinear Kalman Filter for developing a robust controller which can be applied to quadropters. The control problem for quadropters is nontrivial and becomes further complicated if this robotic model is subjected to uncertainties and external disturbances. Using differential flatness theory, it is shown that the model of a quadropter can be transformed to linear canonical form. For the linearized equivalent of the quadropter it is shown that a state feedback controller can be designed. Since certain elements of the state vector of the linearized system cannot be measured directly, it is proposed to estimate them with the use of a novel filtering method, the so-called Derivative-free nonlinear Kalman Filter. Moreover, by redesigning the Kalman Filter as a disturbance observer, it is shown that one can estimate simultaneously external disturbances terms that affect the quadropter or disturbance terms which are associated with parametric uncertainty. The efficiency of the proposed control scheme is confirmed through simulation experiments.

Finally, the chapter proposes a nonlinear control approach for an underactuated vessel model based on differential flatness theory and using the Derivative-free nonlinear Kalman Filter as a state and disturbance estimator. It is proven that the sixth-order nonlinear model of the hovercraft is a differentially flat one. It is shown that this model cannot be subjected 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 subjected 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 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 of the vessel and of an inverse nonlinear transformation based on the differential flatness features of the system which enables to compute 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. The efficiency of the proposed nonlinear control and state estimation scheme is shown again through simulation experiments.

6.2 State Estimation-Based Control of Autonomous Vehicles

6.2.1 Localization and Autonomous Navigation of Ground Vehicles

Filtering-based state estimation for unmanned ground vehicles (UGVs) is a significant topic because it enables their accurate localization and autonomous navigation [39]. For nonlinear systems such as UGVs, and under the assumption of Gaussian noise, the Extended Kalman Filter (EKF) is frequently applied for estimating the nonmeasurable state variables through the processing of input and output sequences or for obtaining estimates of the state vector through the fusion of measurements coming from various sensors [31, 68, 192, 229, 372]. The Extended Kalman Filter is based on linearization of the system's dynamics using a first-order Taylor expansion [244, 405, 408, 411, 412, 421, 573]. Although EKF is efficient in several estimation and fusion problems, it is characterized by cumulative errors due to the local linearization assumption and this may affect the accuracy of the UGV's motion estimation or even risk the stability of the UGV state estimation-based control loop.

First, in this section, and using differential flatness theory, the nonlinear system is subjected to a linearization transformation. This makes possible (i) to design an efficient control law for trajectories tracking, and (ii) to apply to the nonlinear system a filtering method that it is based on the standard Kalman Filter recursion. Unlike the Extended Kalman Filter (EKF), the proposed filtering method provides estimates of the state vector of the UGV without the need for derivatives and Jacobians calculation. By avoiding linearization approximations, the proposed derivative-free nonlinear Kalman filtering method improves the accuracy of estimation of the system state variables, and results in smooth control signal variations and in minimization of the tracking error of the associated control loop.

Next, the section extends the results of [424, 429] toward nonlinear dynamical systems, such as UGVs, which are described by multi-input multi-output (MIMO) models. Actually, the section's results are applicable to differentially flat MIMO nonlinear dynamical systems which after applying the differential flatness theory can be written in the Brunovksy (canonical) form [263, 340]. Simulation experiments on the problem of autonomous navigation of a ground vehicle are provided to test the performance of the proposed derivative-free Kalman Filter.

6.2.2 Application of Derivative-Free Kalman Filtering to MIMO UGV Models

6.2.2.1 Kinematic Models for Autonomous Vehicles

The proposed method of derivative-free Kalman Filtering for MIMO nonlinear systems will be analyzed through an application example. The kinematic model of a UGV (robotic unicycle) is considered. This is given by

$$\begin{aligned}\dot{x} &= vcos(\theta)\\ \dot{y} &= vsin(\theta)\\ \dot{\theta} = \omega &= \tfrac{v}{L}tan(\phi)\end{aligned} \tag{6.1}$$

where $v(t)$ is the velocity of the vehicle, L is the distance between the front and the rear wheel axis of the vehicle, θ is the angle between the transversal axis of the vehicle and axis OX, and ϕ is the angle of the steering wheel with respect to the transversal axis of the vehicle. The position of such a vehicle is described by the coordinates (x, y) of the center of its rear axis and its orientation is given by the angle θ between the x-axis and the axis of the direction of the vehicle. The steering angle ϕ and the speed v are considered to be the inputs of the system.

The problem of control of autonomous ground vehicles (AGVs) is first considered. The position of such a vehicle is described by the coordinates (x, y) of the center of its rear axis and its orientation is given by the angle θ between the x-axis and the axis of the direction of the vehicle. The steering angle ϕ and the speed v are considered to be the inputs of the system. The kinematic model of autonomous vehicles can be expressed in the general form [429]

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} v(t) \\ v(t)\rho(t) \end{pmatrix} \tag{6.2}$$

where (x, y) are the coordinates of the center of the vehicle's rear wheels axis, $v(t)$ is the velocity of the vehicle, and θ is the angle between the transversal axis of the vehicle and axis OX. The autonomous vehicle is a nonholonomic system. Nonholonomic systems are characterized by nonintegrable differential expressions, such as

$$\sum_{i=i}^{n} f_{ij}(q_1, q_2, \ldots, q_n, t)\dot{q}_i = 0, \ \ j = 1, 2, \ldots, m \tag{6.3}$$

where $\dot{q}_i$ represents the nth generalized coordinate (state variable), m is the number of equations defining the nonholonomic constraints, $\dot{q}_i$ represents the generalized speed, and f_{ij} are nonlinear functions of q_i at time t. For the kinematic model of Eq. (6.2), the following nonholonomic constraint exists:

$$\dot{x}sin(\theta) - \dot{y}cos(\theta) = 0 \tag{6.4}$$

The curvature radius for any path can be written as

$$R(t) = \frac{1}{\rho(t)} = \frac{L}{tan(\phi)} \tag{6.5}$$

where L is the distance between the front and the back wheels, and ϕ (namely the steering angle) is the angle defined by the main axis of the vehicle and the velocity vector of the front wheel (for cart-like vehicles as shown in Fig. 6.1), or the central front point (for car-like vehicles as shown in Fig. 6.1). The value of $R(t)$ is usually bounded by R_{min}, the minimum curvature radius (Fig. 6.2).

Fig. 6.1 The model of the autonomous robotic vehicle (cart-like vehicle)

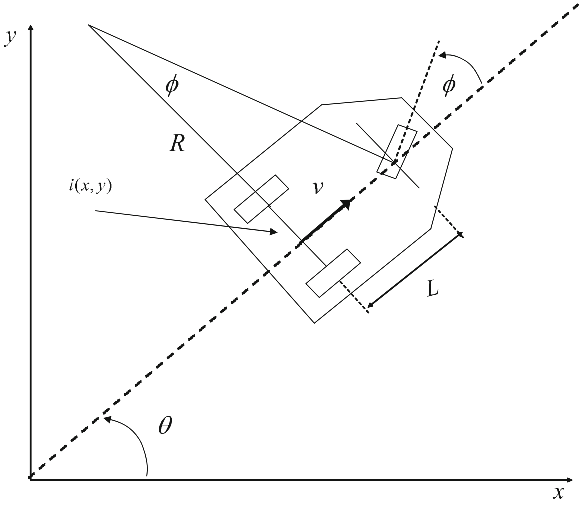
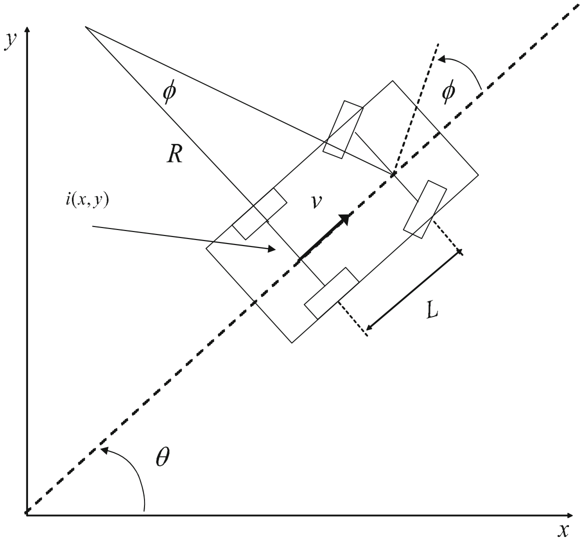


Fig. 6.2 The model of the 4-wheel autonomous vehicle (car-like vehicle)



6.2.3 Controller Design for UGVs

Flatness-based control can be used for steering the vehicle along a desirable trajectory. In the case of the autonomous vehicle of Eq. (6.1), the flat output is the cartesian position of the center of the wheel axis, denoted as $\eta = (x, y)$, while the other model parameters can be written as:

$$v = \pm||\dot{\eta}|| \begin{pmatrix} cos(\theta) \\ sin(\theta) \end{pmatrix} = \frac{\dot{\eta}}{v} \; tan(\phi) = l det(\dot{\eta}\ddot{\eta})/v^3 \tag{6.6}$$

These formulas show simply that θ is the tangent angle of the curve and $tan(\phi)$ is the associated curvature. With reference to a generic driftless nonlinear system $\dot{q},\ q \in R^n, w \in R^m$, dynamic feedback linearization consists in finding a feedback compensator of the form

$$\begin{aligned} \dot{\xi} &= \alpha(q,\xi) + b(q,\xi)u \\ w &= c(q,\xi) + d(q,\xi)u \end{aligned} \tag{6.7}$$

with state $\xi \in R^v$ and input $u \in R^m$, such that the closed-loop system of Eqs. (6.1) and (6.7) is equivalent under a state transformation $z = T(q,\xi)$ to a linear system. The starting point is the selection of a m-dimensional output $\eta = h(q)$ to which a desired behavior can be assigned (this is the previously defined *flat output*). One then proceeds by successively differentiating the output until the input appears in a nonsingular way. If the sum of the output differentiation orders equals the dimension $n + v$ of the extended state space, full input-state-output linearization is obtained. The closed-loop system is then equivalent to a set of decoupled input-output chains of integrators from u_i to η_i. The exact linearization procedure is illustrated for the vehicle's model of Eq. (6.2). As flat output $\eta = (x, y)$ the coordinates of the center of the wheel axis is considered. Differentiation with respect to time then yields [382, 422]

$$\dot{\eta} = \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \end{pmatrix} \cdot \begin{pmatrix} v \\ \omega \end{pmatrix} \tag{6.8}$$

showing that only v affects $\dot{\eta}$, while the angular velocity ω cannot be recovered from this first-order differential information. To proceed, one needs to add an integrator (whose state is denoted by ξ) on the linear velocity input

$$v = \xi,\ \dot{\xi} = \alpha \Rightarrow \dot{\eta} = \xi \begin{pmatrix} cos(\theta) \\ sin(\theta) \end{pmatrix} \tag{6.9}$$

where α denotes the linear acceleration of the vehicle. Differentiating further one obtains

$$\begin{aligned} \ddot{\eta} &= \dot{\xi} \begin{pmatrix} cos(\theta) \\ sin(\theta) \end{pmatrix} + \xi\dot{\theta} \begin{pmatrix} sin(\theta) \\ cos(\theta) \end{pmatrix} = \\ &= \begin{pmatrix} cos(\theta) & -\xi sin(\theta) \\ sin(\theta) & \xi cos(\theta) \end{pmatrix} \begin{pmatrix} \alpha \\ \omega \end{pmatrix} \end{aligned} \tag{6.10}$$

and the matrix multiplying the modified input (α, ω) is nonsingular if $\xi \neq 0$. Under this assumption one defines

$$\begin{pmatrix} \alpha \\ \omega \end{pmatrix} = \begin{pmatrix} cos(\theta) & -\xi sin(\theta) \\ sin(\theta) & \xi cos(\theta) \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{6.11}$$

and $\ddot{\eta}$ is denoted as

$$\ddot{\eta} = \begin{pmatrix} \ddot{\eta}_1 \\ \ddot{\eta}_2 \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = u \tag{6.12}$$

which means that the desirable linear acceleration and the desirable angular velocity can be expressed using the transformed control inputs u_1 and u_2. Then, the resulting dynamic compensator is (return to the initial control inputs v and ω)

$$\begin{gathered}\dot{\xi}=u_1cos(\theta)+u_2sin(\theta)\\ v=\xi\\ \omega=\tfrac{u_2cos(\theta)-u_1sin(\theta)}{\xi}\end{gathered}\tag{6.13}$$

Being $\xi\in R$, it is $n+v=3+1=4$, equal to the output differentiation order in Eq. (6.12). In the new coordinates

$$\begin{gathered}z_1=x\\ z_2=y\\ z_3=\dot{x}=\xi cos(\theta)\\ z_4=\dot{y}=\xi sin(\theta)\end{gathered}\tag{6.14}$$

The extended system is thus fully linearized and described by the chains of integrators, in Eq. (6.12), and can be rewritten as

$$\ddot{z}_1=u_1,\ \ \ddot{z}_2=u_2\tag{6.15}$$

The dynamic compensator of Eq. (6.13) has a potential singularity at $\xi=v=0$, i.e., when the vehicle is not moving, which is a case not met while executing the trajectory tracking. It is noted, however, that the occurrence of such a singularity is structural for nonholonomic systems. In general, this difficulty must be obviously taken into account when designing control laws on the equivalent linear model. A nonlinear controller for output trajectory tracking, based on dynamic feedback linearization, is easily derived. Assume that the autonomous vehicle must follow a smooth trajectory $(x_d(t),y_d(t))$ which is persistent, i.e., for which the nominal velocity $v_d=(\dot{x}_d^2+\dot{y}_d^2)^{\frac{1}{2}}$ along the trajectory never goes to zeros (and thus singularities are avoided). On the equivalent and decoupled system of Eq. (6.15), one can easily design an exponentially stabilizing feedback for the desired trajectory, which has the form

$$\begin{gathered}u_1=\ddot{x}_d+k_{p_1}(x_d-x)+k_{d_1}(\dot{x}_d-\dot{x})\\ u_2=\ddot{y}_d+k_{p_1}(y_d-y)+k_{d_1}(\dot{y}_d-\dot{y})\end{gathered}\tag{6.16}$$

and which results in the following error dynamics for the closed-loop system

$$\begin{gathered}\ddot{e}_x+k_{d_1}\dot{e}_x+k_{p_1}e_x=0\\ \ddot{e}_y+k_{d_2}\dot{e}_y+k_{p_2}e_y=0\end{gathered}\tag{6.17}$$

where $e_x=x-x_d$ and $e_y=y-y_d$. The proportional-derivative gains are chosen as $k_{p_1}>0$ and $k_{d_1}>0$ for $i=1,2$. Knowing the control inputs u_1,u_2, for the linearized system one can calculate the control inputs v and ω applied to the

vehicle, using Eq. (6.7). The above result is valid, provided that the dynamic feedback compensator does not meet the singularity. In the general case of design of flatness-based controllers, the following theorem assures the avoidance of singularities in the proposed control law [382]:
Theorem: Let $\lambda_{11}, \lambda_{12}$ and $\lambda_{21}, \lambda_{22}$, be respectively the eigenvalues of two equations of the error dynamics, given in Eq. (6.7). Assume that, for $i = 1, 2$ it is $\lambda_{11} < \lambda_{12} < 0$ (negative real eigenvalues), and that λ_{i2} is sufficiently small. If

$$\min_{t\geq 0}|| \begin{pmatrix} \dot{x}_d(t) \\ \dot{y}_d(t) \end{pmatrix} || \geq \begin{pmatrix} \dot{\varepsilon}_x^0 \\ \dot{\varepsilon}_y^0 \end{pmatrix} \tag{6.18}$$

with $\dot{\varepsilon}_x^0 = \dot{\varepsilon}_x(0) \neq 0$ and $\dot{\varepsilon}_y^0 = \dot{\varepsilon}_y(0) \neq 0$, then the singularity $\xi = 0$ is never met.

6.2.4 Derivative-Free Kalman Filtering for UGVs

It is assumed now that the vehicle's velocity has to be estimated through the processing of the sequence of position measurements by a filtering algorithm. To this end, the derivative-free Kalman Filter for MIMO nonlinear dynamical systems can been used. From the application of the differential flatness theory presented in Sect. 6.4.3 for transforming the initial nonlinear vehicle's model into a linearized equivalent that is finally written in the Brunovsky form, one has Eq. (6.12) which means $\ddot{x} = u_1$ and $\ddot{y} = u_2$. Next, the state variables $x_1 = x$, $x_2 = \dot{x}$, $x_3 = y$, and $x_4 = \dot{y}$ are defined. Considering the state vector $x \in R^{4\times 1}$, the following matrices are also defined

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \\ C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{6.19}$$

Using the matrices of Eq. (6.26), one obtains the Brunovsky form of the MIMO robotic unicycle

$$\begin{aligned} \dot{x} &= Ax + Bv \\ y &= Cx \end{aligned} \tag{6.20}$$

where the new input v is given by $v = [u_1(x,t), u_2(x,t)]^T$. For the robotic model of Eq. (6.20), state estimation can be performed using the standard Kalman Filter recursion, as described in Eqs. (4.5) and (4.6).

6.2.5 Simulation Tests

6.2.5.1 Extended Kalman Filter-Based Navigation of the Autonomous Vehicle

The vehicle's kinematic model of Eq. (6.1) is considered. A GPS sensor or encoders placed at the vehicle's wheels can be used to provide measurements of the cartesian coordinates of the vehicle $x(t)$ and $y(t)$ (displacement of the vehicle), over a sampling period T. The values of the vehicle's velocity components along the x and y axes are not directly available and are estimated through the processing of the sequence of position measurements with the use of a filtering algorithm. Computing the vehicle's speed through the differentiation of the position measurements would introduce cumulative errors in the value of the vehicle's velocity, which in turn can affect the performance of the control loop. To avoid such errors, an estimation of the vehicle's velocity is obtained through the processing of the sequence of position measurements with the use of a filtering algorithm.

Assuming a constant sampling period $\Delta t_k = T$, the measurement equation is $z(k+1) = \gamma(x(k)) + v(k)$, where $z(k)$ is the vector containing the sequence of measurements of the cartesian coordinates of the vehicle and $v(k)$ is the measurement noise vector.

$$z(k) = [x(k) + v_1(k), y(k) + v_2(k)], \ k = 1, 2, 3\ldots \tag{6.21}$$

To obtain the Extended Kalman Filter (EKF), the kinematic model of the vehicle is linearized about the estimates $\hat{x}(k)$ and $\hat{x}^-(k)$, and the control input $U(k)$ is applied. The EKF recursion consists of the measurement update part and of the time update part as described in Eqs. (4.13) and (4.14), respectively. One has to use the input gain matrix $L(k)$

$$L(k) = \begin{pmatrix} Tcos(\theta(k)) & 0 \\ Tsin(\theta(k)) & 0 \\ 0 & T \end{pmatrix} \tag{6.22}$$

and to compute the Jacobian $J_\phi(\hat{x}(k))$ given by the following

$$J_\phi(\hat{x}(k)) = \begin{pmatrix} 1 & 0 & -v(k)sin(\theta)T \\ 0 & 1 & v(k)cos(\theta)T \\ 0 & 0 & 1 \end{pmatrix} \tag{6.23}$$

while for the elements of the process noise covariance matrix which is defined as $Q(k) = diag[\sigma^2(k), \sigma^2(k), \sigma^2(k)]$ indicative values are $\sigma^2(k) = 10^{-3}$.

Using the estimated state vector, function $\phi(x)$ appearing in the state equations part and $\gamma(x)$ appearing in the measurements equations part of the vehicle's kinematic model become $\phi(\hat{x}(k)) = [\hat{x}(k), \hat{y}(k)]^T$, and $\gamma(\hat{x}(k)) = [\hat{x}(k), \hat{y}(k)]$, respectively. The associated Jacobian matrix $J_\gamma^T(\hat{x}^-(k))$ is given by

$$J_\gamma(\hat{x}^-(k)) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \tag{6.24}$$

The vehicle is steered by the flatness-based controller analyzed in Sect. 6.4.3

$$\begin{array}{c} u_1 = \ddot{x}_d + K_{p_1}(x_d - x) + K_{d_1}(\dot{x}_d - \dot{x}) \\ u_2 = \ddot{y}_d + K_{p_2}(y_d - y) + K_{d_2}(\dot{y}_d - \dot{y}) \\ \dot{\xi} = u_1 cos(\theta) + u_2 sin(\theta) \\ v = \xi, \ \ \omega = \frac{u_2 cos(\theta) - u_1 sin(\theta)}{\xi} \end{array} \tag{6.25}$$

The use of EKF for estimating the vehicle's velocity along the x-axis (denoted as $\dot{x}$) and the vehicle's velocity along the y-axis (denoted as $\dot{y}$) enables the successful application of nonlinear steering control of Eq. (6.25).

Indicative results about tracking of the circular reference trajectory with the use of the Extended Kalman Filters are shown in Figs. 6.3, 6.4, 6.5, and 6.6. In Fig. 6.3, one can notice the accuracy of tracking of the reference trajectory, succeeded by the proposed state estimation-based control scheme. In Fig. 6.4, the accuracy of tracking of the x and y axis position setpoints is depicted. In Fig. 6.5, the associated x and y axis tracking errors are shown. Finally, in Fig. 6.6, the x and y axis velocity estimation errors are given.

Indicative results about tracking of the eight-shaped reference trajectory with use of the Extended Kalman Filter are shown in Fig. 6.7, 6.8, 6.9, and 6.10. In Fig. 6.7

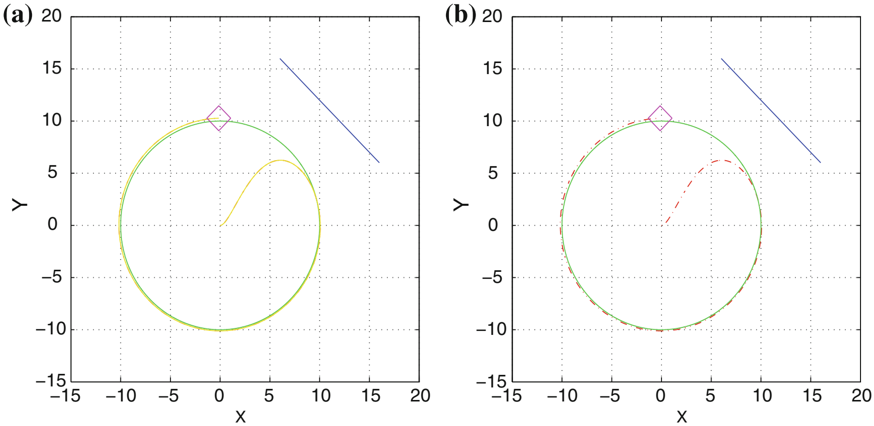


Fig. 6.3 **a** Tracking of a circular reference trajectory (*green line*) by the autonomous vehicle and associated estimation of the vehicle's position provided by the Extended Kalman Filter (*continuous yellow line*). **b** Tracking of a circular reference trajectory (*green line*) by the autonomous vehicle and real position of the vehicle (*dashed red line*)

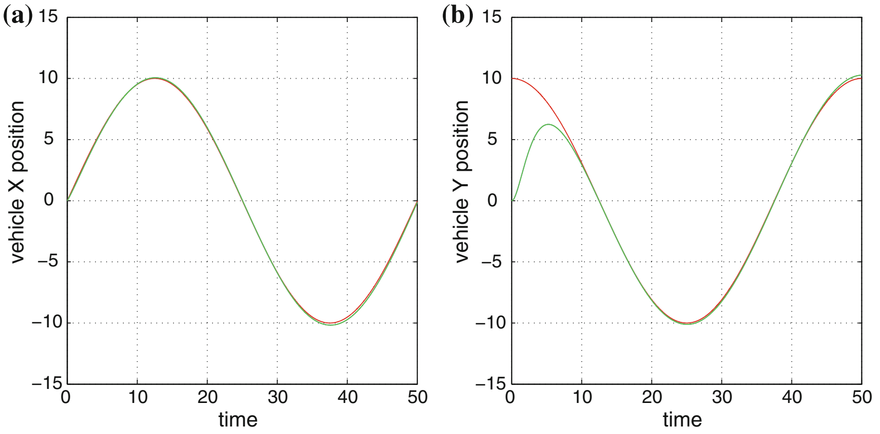


Fig. 6.4 Tracking of a circular reference trajectory with the use of the EKF: **a** tracking of the x-axis reference setpoint, **b** tracking of the y-axis reference setpoint

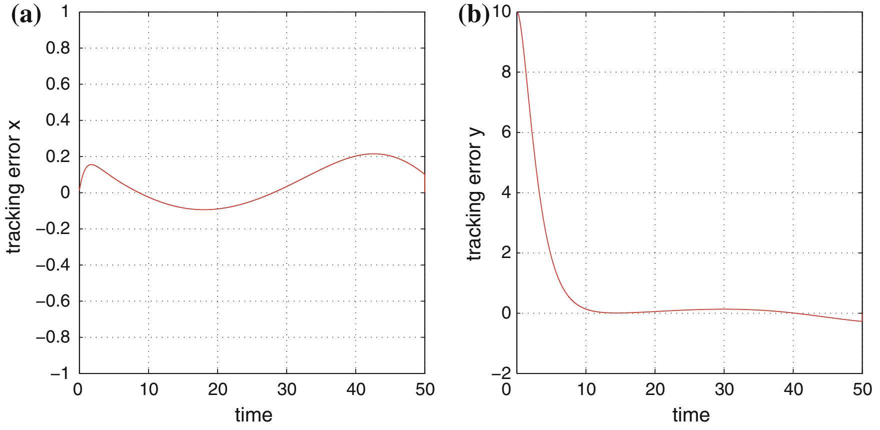


Fig. 6.5 Tracking of a circular reference trajectory with use of the EKF: **a** tracking error along the x-axis, **b** tracking error along the y-axis

one can notice the accuracy of tracking of the reference trajectory, succeed by the proposed state estimation-based control scheme. In Fig. 6.8 the accuracy of tracking of the x and y axis position setpoints is depicted. In Fig. 6.9, the associated x and y axis tracking errors are shown. Finally, in Fig. 6.10 the x and y axis velocity estimation errors are given.

Indicative results about tracking of the complex-curved reference trajectory with use of the Extended Kalman Filter are shown in Fig. 6.11, 6.12, 6.13, and 6.14. In Fig. 6.11, one can notice the accuracy of tracking of the reference trajectory, succeeded by the proposed state estimation-based control scheme. In Fig. 6.12, the

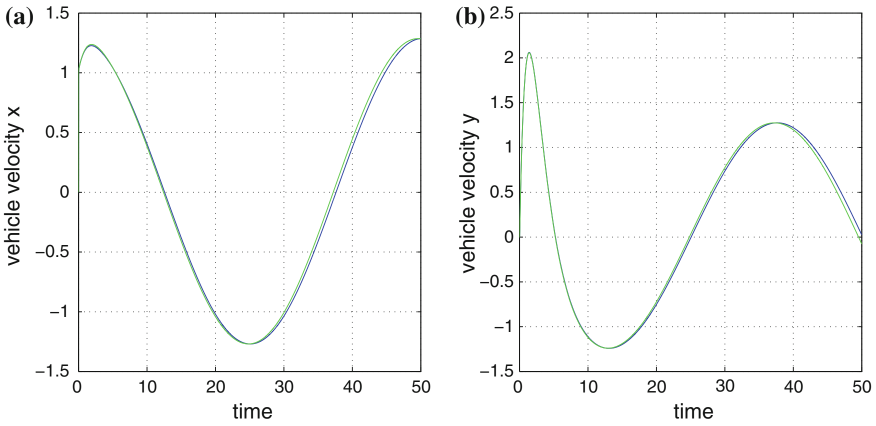


Fig. 6.6 Tracking of a circular reference trajectory by the autonomous vehicle with use of the EKF: **a** convergence of the estimated x-axis velocity (*green line*) to the associated real velocity (*blue line*), **b** convergence of the estimated y-axis velocity (*green line*) to the associated real velocity (*green line*)

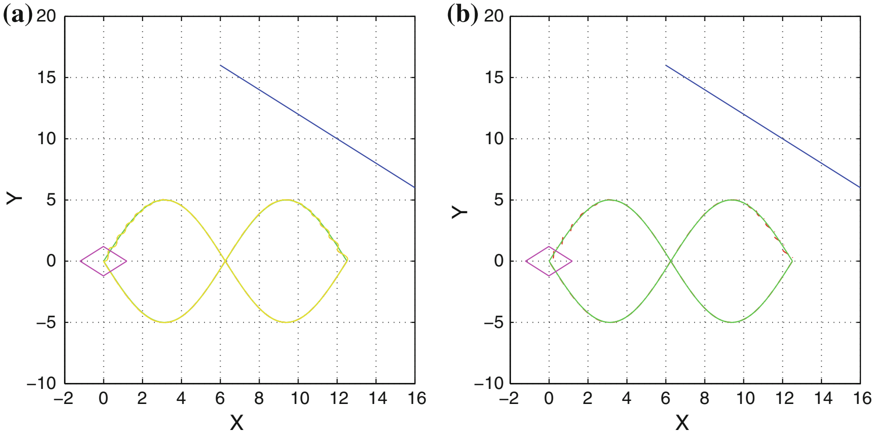


Fig. 6.7 **a** Tracking of an eight-shaped reference trajectory (*green line*) by the autonomous vehicle and associated estimation of the vehicle's position provided by the Extended Kalman Filter (*yellow line*). **b** Tracking of a circular reference trajectory (*green line*) by the autonomous vehicle and real position of the vehicle (*dashed red line*)

accuracy of tracking the x and y axis position setpoints is depicted. In Fig. 6.13, the associated x and y axis tracking errors are shown. Finally, in Fig. 6.14, the x and y axis velocity estimation errors are given.

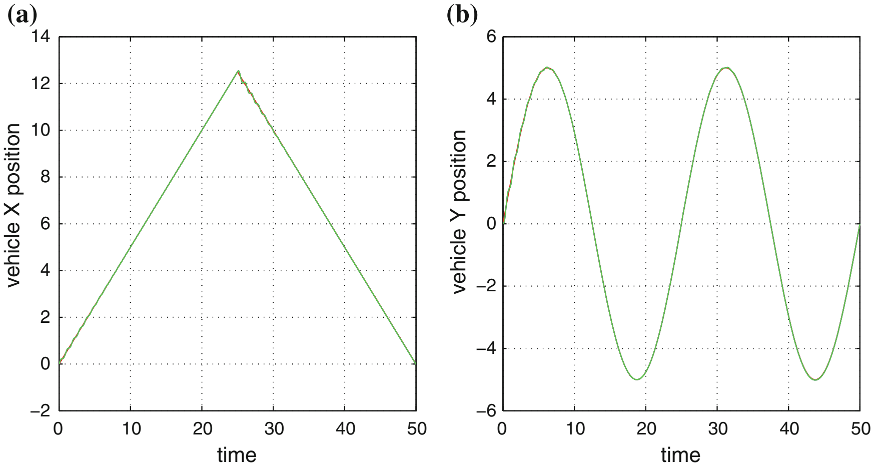


Fig. 6.8 Tracking of an eight-shaped reference trajectory with use of the EKF: **a** tracking of the x-axis reference setpoint, **b** tracking of the y-axis reference setpoint

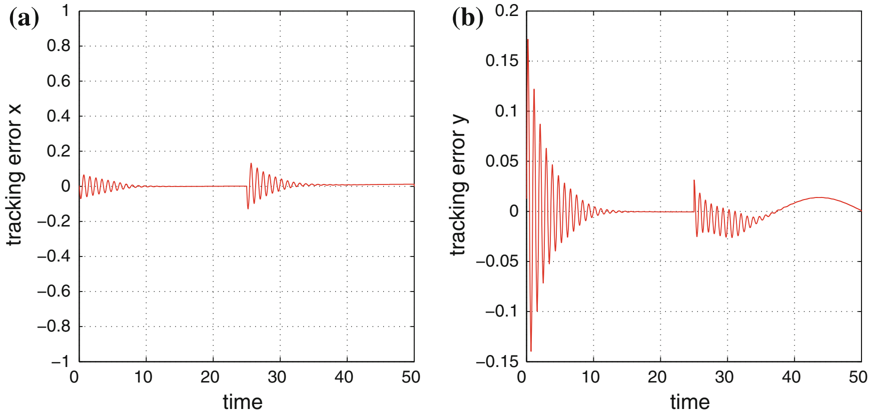


Fig. 6.9 Tracking of an eight-shaped reference trajectory with use of the EKF: **a** tracking error along the x-axis, **b** tracking error along the y-axis

6.2.6 Derivative-Free Kalman Filter-Based Navigation of the Autonomous Vehicle

The performance of the proposed derivative-free nonlinear Kalman Filter was tested in the problem of state estimation-based control for autonomous navigation of the previously described robotized vehicle (car-like robot) (Fig. 6.1). The differentially flat model of the autonomous vehicle and its transformation to the Brunovksy form has been analyzed in Sect. 6.2.2. It was assumed that only measurements of the

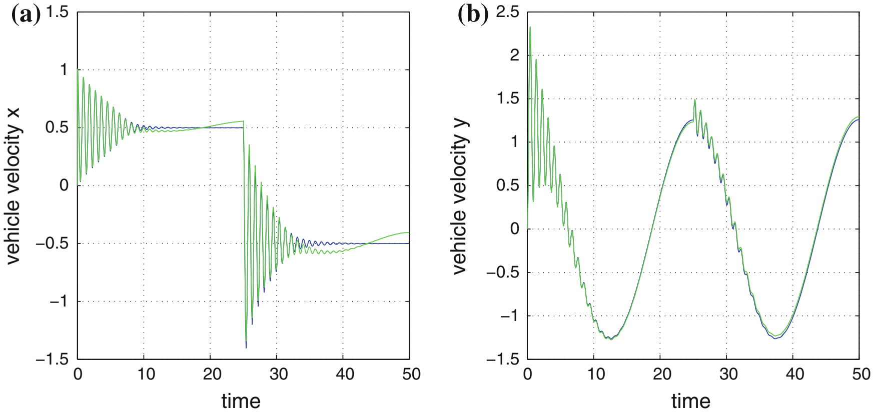


Fig. 6.10 Tracking of an eight-shaped reference trajectory by the autonomous vehicle with use of the EKF: **a** convergence of the estimated x-axis velocity (*green line*) to the associated real velocity (*blue line*), **b** convergence of the estimated y-axis velocity (*green line*) to the associated real velocity (*green line*)

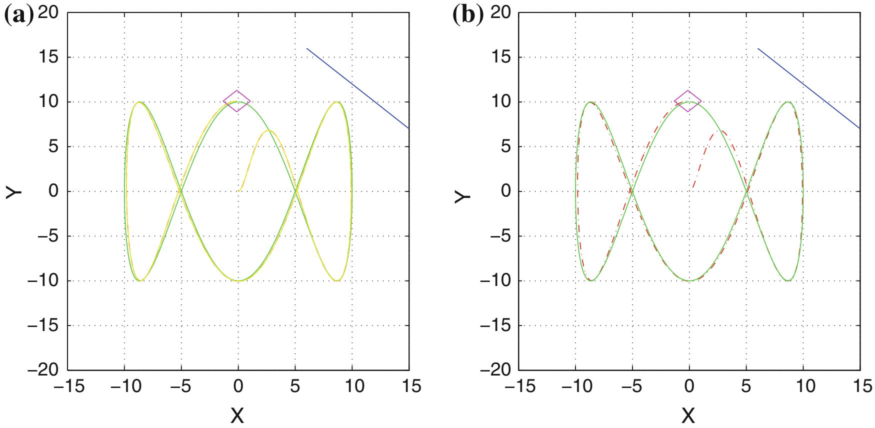


Fig. 6.11 **a** Tracking of a complex-curved reference trajectory (*green line*) by the autonomous vehicle and associated estimation of the vehicle's position provided by the Extended Kalman Filter *(yellow line*). **b** Tracking of a circular reference trajectory (*green line*) by the autonomous vehicle (*red dashed line*) and real position of the vehicle (*dashed red line*)

cartesian coordinates of the vehicle (displacement on the xy-plane) could be obtained through a GPS unit (localization of moderate accuracy), RTK-GPS (localization of higher accuracy), or through laser, visual, and sonar sensors with reference to specific landmarks (the latter measuring approaches require transformation from a local to a global coordinates system).

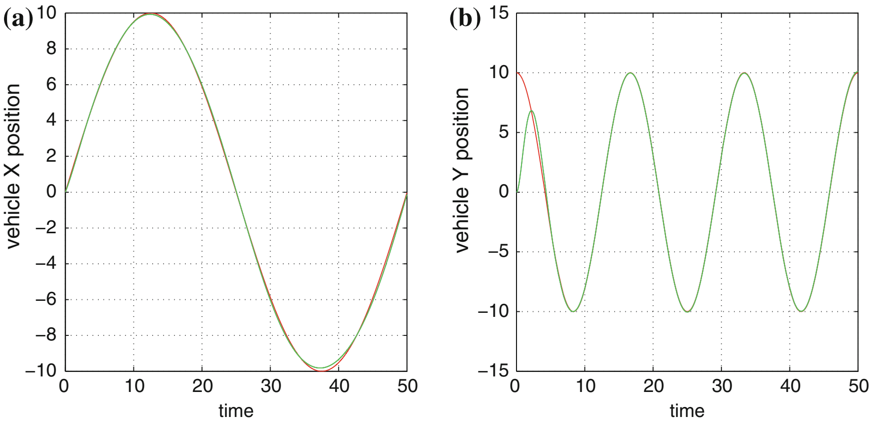


Fig. 6.12 Tracking of a complex-curved reference trajectory with use of the EKF: **a** tracking of the x-axis reference setpoint, **b** tracking of the y-axis reference setpoint

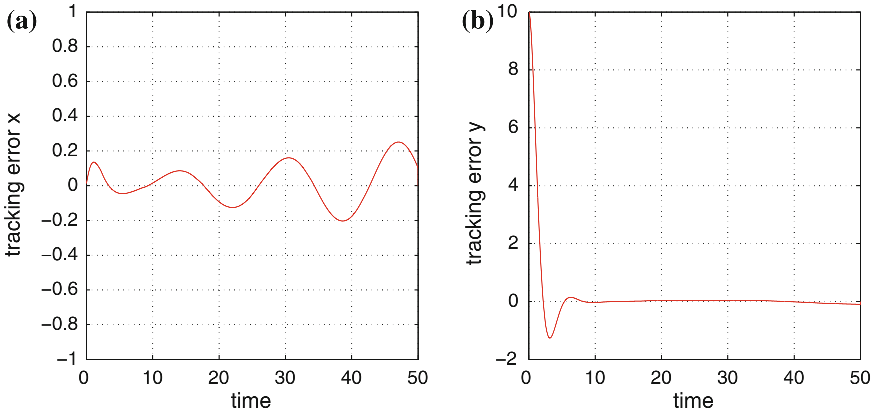


Fig. 6.13 Tracking of a complex-curved reference trajectory with use of the EKF: **a** tracking error along the x-axis, **b** tracking error along the y-axis

Indicative results about tracking of the circular reference trajectory with use of the derivative-free Kalman Filter are shown in Figs. 6.15, 6.16, and 6.17. Comparing the estimation performed by the derivative-free MIMO nonlinear Kalman Filter to the one performed by the Extended Kalman Filter it can be noticed that the derivative-free filtering approach results in more accurate state estimates. Finally, it is noted that following a similar methodology the chapter's approach can be applied also to 4-wheel autonomous vehicles.

Indicative results about tracking of the circular reference trajectory with use of the derivative-free Kalman Filter are shown in Figs. 6.15, 6.16, and 6.17. In Fig. 6.15, one can notice the accuracy of tracking of the reference trajectory, succeeded by

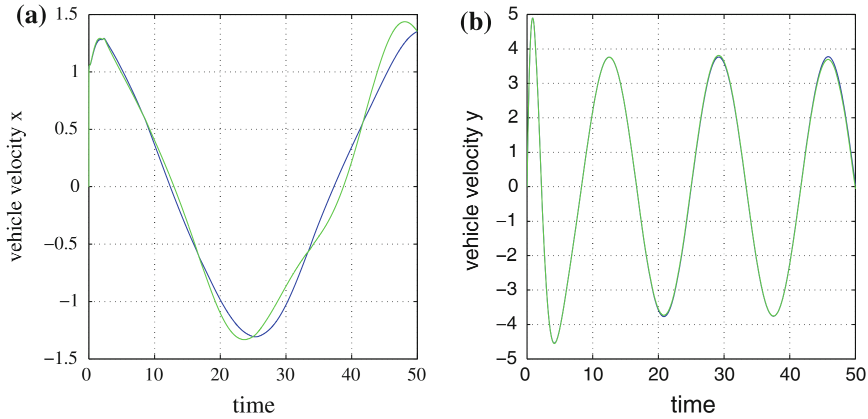


Fig. 6.14 Tracking of a complex-curved reference trajectory by the autonomous vehicle with use of the EKF: **a** convergence of the estimated x-axis velocity (*green line*) to the associated real velocity (*blue line*), **b** convergence of the estimated y-axis velocity (*green line*) to the associated real velocity (*green line*)

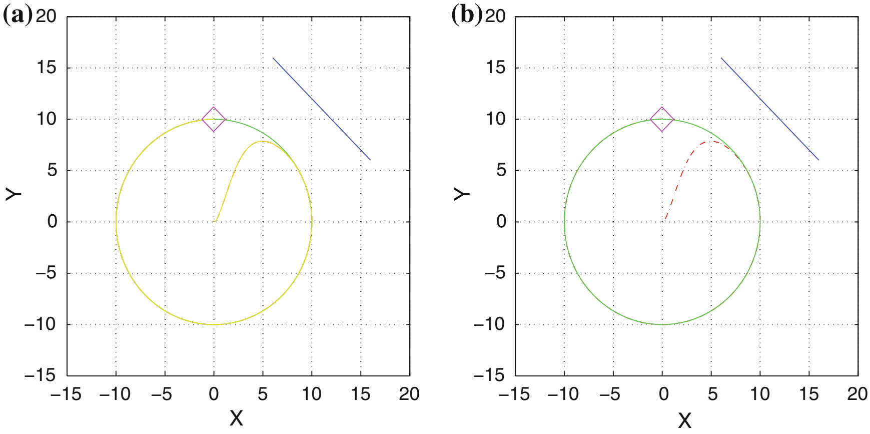


Fig. 6.15 **a** Tracking of a circular reference trajectory (*green line*) by the autonomous vehicle and associated estimation of the vehicle's position provided by the derivative-free Kalman Filter (*yellow line*). **b** Tracking of a circular reference trajectory (*green line*) by the autonomous vehicle and real position of the vehicle (*dashed red line*)

the proposed state estimation-based control scheme. In Fig. 6.16, the associated x and y axis tracking errors are shown. Finally, in Fig. 6.17, the x and y axis velocity estimation errors are given.

Indicative results about tracking of the eight-shaped reference trajectory with the use of Extended Kalman Filter are shown in Figs. 6.18, 6.19, 6.20, and 6.21. In Fig. 6.18, one can notice the accuracy of tracking of the reference trajectory,

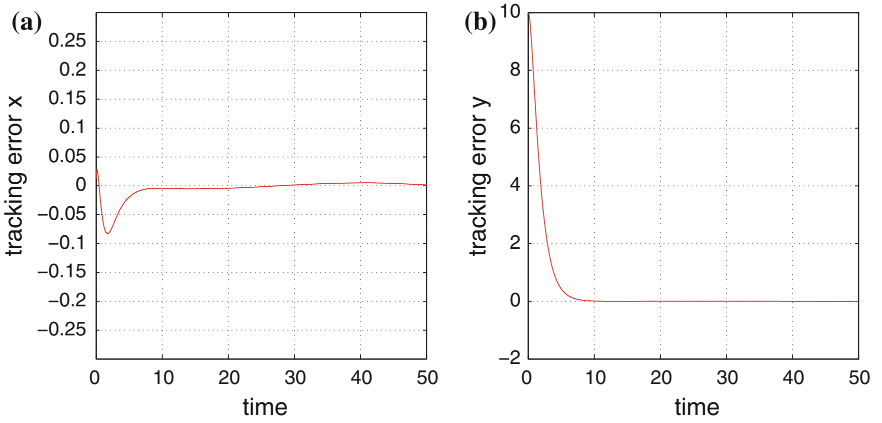


Fig. 6.16 Tracking of a circular reference trajectory with the use of DKF: **a** tracking error along the x-axis, **b** tracking error along the y-axis

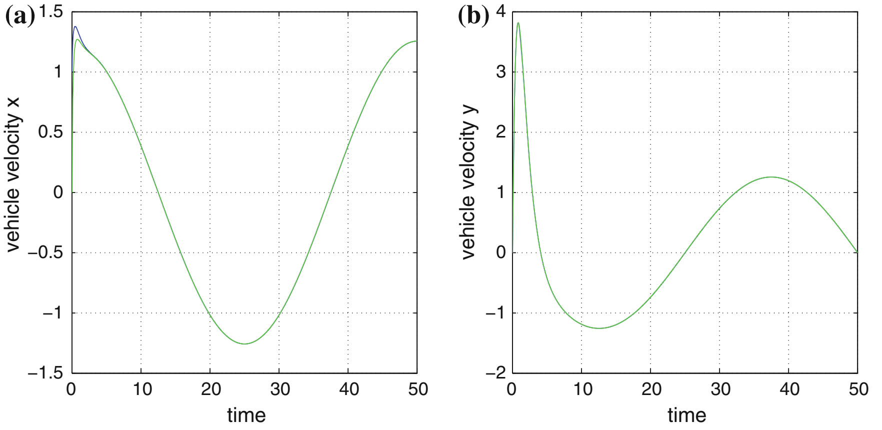


Fig. 6.17 Tracking of a circular reference trajectory by the autonomous vehicle with the use of DKF: **a** convergence of the estimated x-axis velocity (*green line*) to the associated real velocity (*blue line*), **b** convergence of the estimated y-axis velocity (*green line*) to the associated real velocity (*green line*)

succeeded by the proposed state estimation-based control scheme. In Fig. 6.19, the accuracy of tracking of the x and y axis position setpoints is depicted. In Fig. 6.20, the associated x and y axis tracking errors are shown. Finally, in Fig. 6.21, the x and y axis velocity estimation errors are given.

Indicative results about tracking of the complex-curved reference trajectory with use of the Extended Kalman Filter are shown in Figs. 6.22, 6.23, 6.24, and 6.25. In Fig. 6.22, one can notice the accuracy of tracking the reference trajectory, succeeded by the proposed state estimation-based control scheme. In Fig. 6.23, the accuracy of

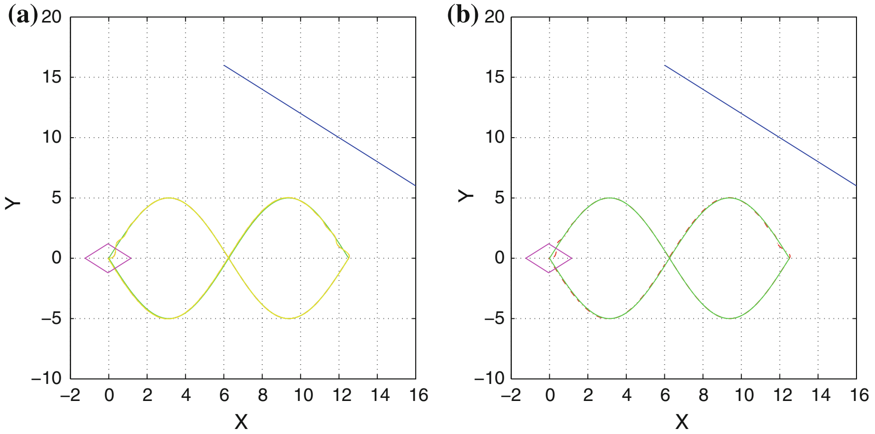


Fig. 6.18 **a** Tracking of an eight-shaped reference trajectory (*green line*) by the autonomous vehicle and associated estimation of the vehicle's position provided by the derivative-free Kalman Filter (*yellow line*). **b** Tracking of a circular reference trajectory (*green line*) by the autonomous vehicle and real position of the vehicle (*dashed red line*)

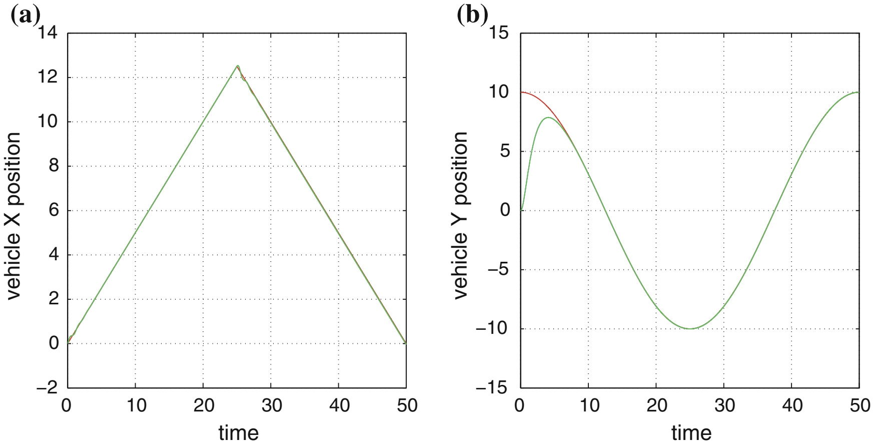


Fig. 6.19 Tracking of an eight-shaped reference trajectory with use of the DKF: **a** tracking of the x-axis reference setpoint, **b** tracking of the y-axis reference setpoint

tracking the x and y axis position setpoints is depicted. In Fig. 6.24, the associated x and y axis tracking errors are shown. Finally, in Fig. 6.25, the x and y axis velocity estimation errors are given.

Comparing the estimation performed by the derivative-free MIMO nonlinear Kalman Filter with the one performed by the Extended Kalman Filter it can be noticed that the derivative-free filtering approach results in more accurate state estimates. Moreover, comparing the associated state estimation-based control loop of the

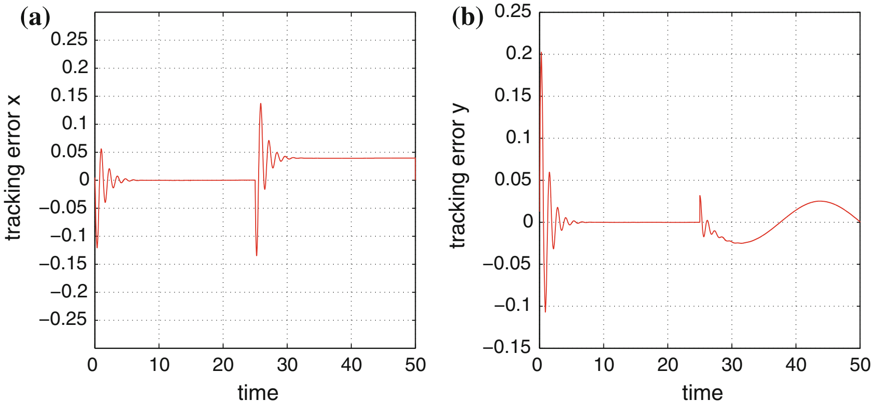


Fig. 6.20 Tracking of an eight-shaped reference trajectory with use of the DKF: **a** tracking error along the x-axis, **b** tracking error along the y-axis

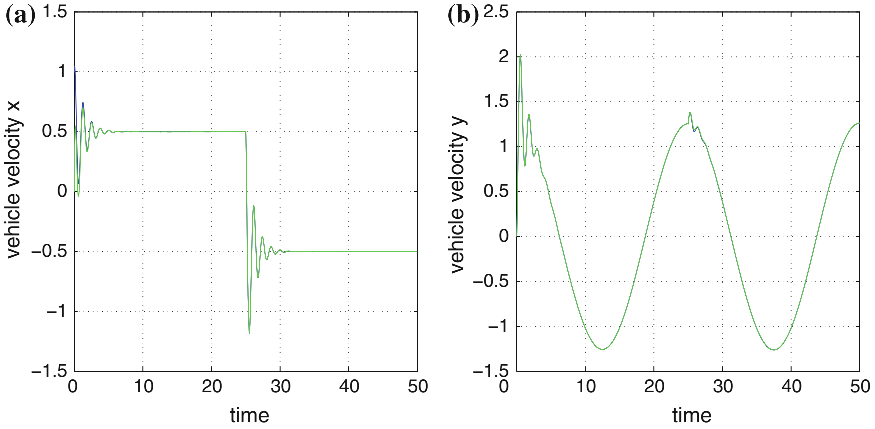


Fig. 6.21 Tracking of an eight-shaped reference trajectory by the autonomous vehicle with use of the DKF: **a** convergence of the estimated x-axis velocity (*green line*) to the associated real velocity (*blue line*), **b** convergence of the estimated y-axis velocity (*green line*) to the associated real velocity (*green line*)

autonomous vehicle that was based on the derivative-free MIMO nonlinear Kalman Filter to the control that was based on the Extended Kalman Filter it was observed that the first control scheme was significantly more robust and capable of tracking with better accuracy the desirable trajectories. These findings show the suitability of the considered derivative-free MIMO nonlinear Kalman Filter for localization, control, and autonomous navigation of autonomous vehicles. Finally, it is noted that the chapter's approach can also be applied to various types of 4-wheel robotic vehicles.

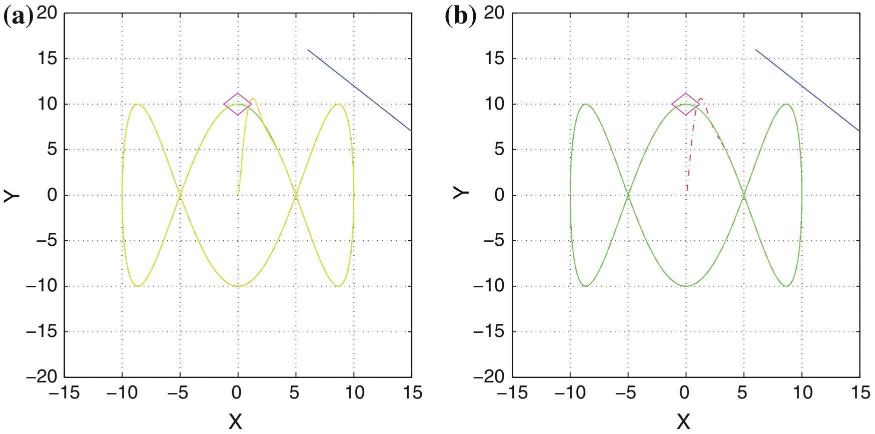


Fig. 6.22 **a** Tracking of a complex-curved reference trajectory (*green line*) by the autonomous vehicle and associated estimation of the vehicle's position provided by the derivative-free Kalman Filter (*yellow line*). **b** Tracking of a circular reference trajectory (*green line*) by the autonomous vehicle and real position of the vehicle (*dashed red line*)

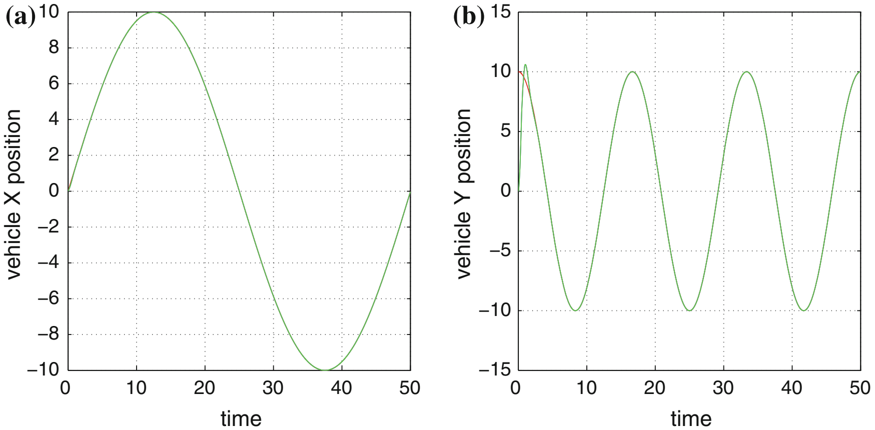


Fig. 6.23 Tracking of a complex-curved reference trajectory with use of the DKF: **a** tracking of the x-axis reference setpoint, **b** tracking of the y-axis reference setpoint

Results about the accuracy of estimation provided by the considered nonlinear filtering algorithms, as well as about the accuracy of tracking succeeded by the associated state estimation-based control loop are given in Table 6.1 [437]. It can be noticed that the Derivative-free nonlinear Kalman Filter is significantly more accurate and robust than the Extended Kalman Filter. Its accuracy is comparable to the one of the Unscented Kalman Filter. Moreover, its accuracy is also comparable to the one succeeded by the Particle Filter for a moderate number of particles (e.g.,

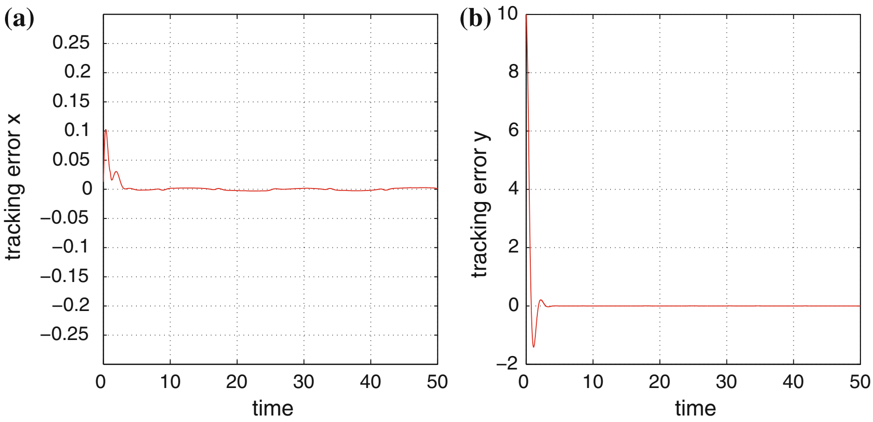


Fig. 6.24 Tracking of a complex-curved reference trajectory with use of the DKF: **a** tracking error along the x-axis, **b** tracking error along the y-axis

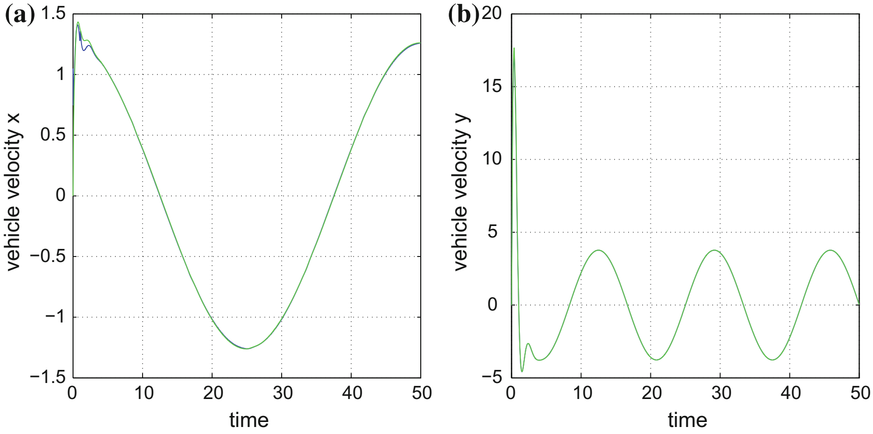


Fig. 6.25 Tracking of a complex-curved reference trajectory by the autonomous vehicle with use of the DKF: **a** convergence of the estimated x-axis velocity (*green line*) to the associated real velocity (*blue line*), **b** convergence of the estimated y-axis velocity (*green line*) to the associated real velocity (*green line*)

$N = 1000$). The Particle Filter can succeed to diminish further the variance of estimation; however, this requires a large number of particles (e.g., $N > 1500$) and induces additional computational cost (see Table 6.2).

The aggregate runtime and the cycle time (duration of each iteration of the filter's algorithm) for the Derivative-free nonlinear Kalman Filter, for the Extended Kalman Filter, for the Unscented Kalman Filter, and for the Particle Filter (in case of $N = 1000$ particles), using the Matlab platform on a PC with a 1.6 GHz Intel i7 processor, is depicted in Table 6.2. It can be noticed that the Derivative-free

Table 6.1 RMSE of tracking (Gaussian noise)

	RMSE$_x$	RMSE$_y$	RMSE$_\theta$
SPKF	0.0146	0.0162	0.0012
PF	0.0060	0.0031	0.0230
EKF	0.0371	0.0294	0.1835
DKF	0.0086	0.0103	0.0004

Table 6.2 RMSE of tracking (Rayleigh noise)

	RMSE$_x$	RMSE$_y$	RMSE$_\theta$
SPKF	0.0218	0.0238	0.0026
PF	0.0092	0.0049	0.0717
EKF	0.0475	0.0338	0.2445
DKF	0.0091	0.0102	0.0021

Table 6.3 Run time of nonlinear estimation algorithms

	SPKF	PF	EKF	DKF
Total runtime (s)	111.14	623.92	104.39	89.69
Cycle time (s)	0.0222	0.1248	0.0209	0.0179

nonlinear Kalman Filter is faster than the other nonlinear estimation algorithms. Actually, the Derivative-free nonlinear Kalman Filter (DKF) appears to be 20 % faster than the Unscented Kalman Filter (UKF). The improved speed of the DKF can be more apparent in higher dimensional, multidegree of freedom systems, where the computation of cross-covariance matrices used by the UKF will impose more numerical operations (Table 6.3).

6.3 State Estimation-Based Control and Synchronization of Cooperating Vehicles

6.3.1 Overview

In the previous section, differential flatness theory was used for filtering and control in unmanned ground vehicles (UGVs). Next, the results are generalized toward filtering and control of cooperating vehicles. There are many types of field operations that are performed by cooperating vehicles [56]. In this section, a method for autonomous navigation of agricultural robots under a master-slave scheme is developed. The method comprises the following elements: (i) a path planner for generating automatically the trajectory that has to be followed by the cooperating robotized vehicles, (ii) a nonlinear controller that makes the robots track with precision the desirable trajectories, (iii) a distributed filtering scheme for estimating the motion characteristics of the vehicles through the fusion of measurements about the vehicles' coordinates coming from multiple position sensors (e.g., multiple GPS devices). The autonomous navigation of the cooperating agricultural robots is finally implemented through state estimation-based control. The nonlinear controller uses the estimated state vector of the robots, as provided by distributed filtering, so as to generate the control signal that defines the robots' speed and heading angle.

A solution to decentralized information fusion over sensor networks, such as the network that collects measurements for the system of the robotic harvesters, can be obtained with the use of distributed Kalman Filtering [23, 163, 215, 264, 265, 326, 369, 380, 381, 485, 534, 552]. In this section, the previously analyzed derivative-free approach to Extended Information filtering will be used. In the proposed derivative-free Kalman Filtering method, the system is first subject to a linearization transformation and next state estimation is performed by applying the standard Kalman Filter recursion to the linearized model. Another issue that has to be taken into account for the autonomous functioning of the robotized vehicles is nonlinear control for precise tracking of desirable trajectories [69]. The chapter proposes flatness-based control for steering the cooperating UGVs along their reference paths [427, 535]. The performance of the proposed distributed filtering-based control scheme is evaluated through simulation experiments.

Such a multirobot system performs distributed information processing for estimating the position and motion characteristics of the vehicles. At a first stage, measurements from onboard sensors are combined with measurements from multiple position sensors (e.g., GPS devices) and are initially processed by local filters to provide local state vector estimates. At a second stage, the local state estimates for the robotic vehicles are fused using a distributed filtering algorithm. Thus an aggregate state vector of the robotic harvesters is obtained (see Fig. 6.26). Such a filtering approach has several advantages: (i) it is fault tolerant: if a local information processing unit is subject to a fault then state estimation is still possible, (ii) the information processing scheme is scalable and can be expanded with the inclusion of more local information processing units (local filters), (iii) the bandwidth for the exchange of information between the local units and the aggregate filter remains limited since there is no transmission of raw measurements but only transmission of local state estimates and of the associated covariance matrices.

Under the assumption of a Gaussian measurement model, a solution to distributed information fusion for the robotized vehicles can be obtained with the use of distributed Kalman Filtering [369, 520]. Distributed state estimation in the case of non-Gaussian models has been also studied in several other research works [329, 462]. In this section, the Derivative-free distributed nonlinear Kalman Filter will be compared against the Extended Information Filter, which is actually an approach for fusing state estimates provided by local Extended Kalman Filters [264, 427].

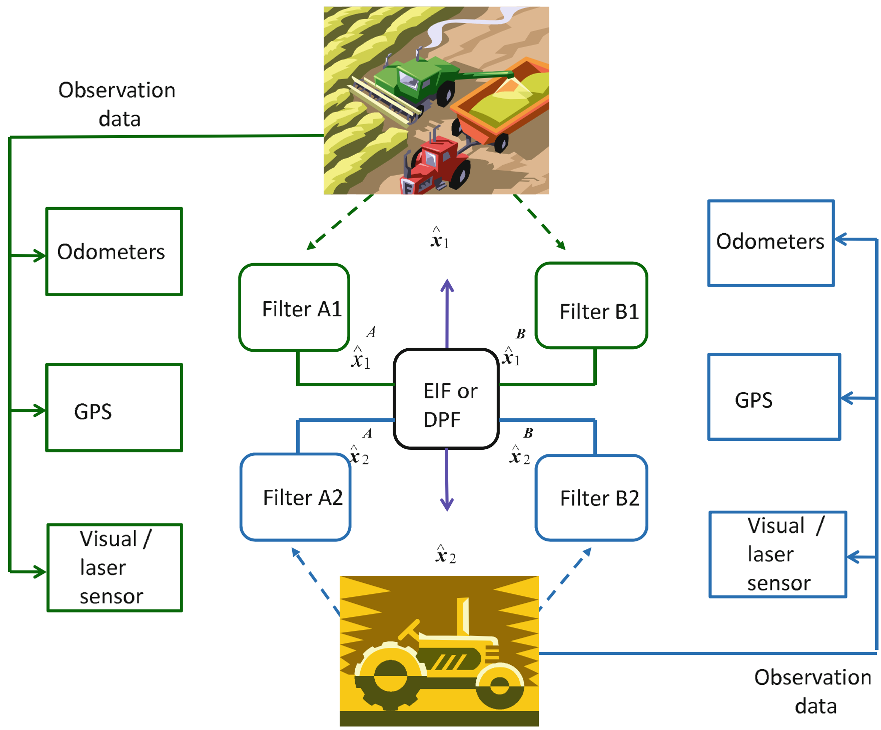


Fig. 6.26 Sensor fusion at the local filters for obtaining local state estimates

6.3.2 Distributed Kalman Filtering for Unmanned Ground Vehicles

Multiple cooperating mobile robots are considered. The coordinates and motion characteristics are monitored by several sensors, as shown in Fig. 6.27. Each mobile robot is taken to be described by the kinematic model of a unicycle vehicle. Estimation for the individual vehicles with the use of the Extended Kalman Filter was described in Sect. 6.2. Moreover, distributed estimation for multiple cooperating vehicles can be based on the Extended Information Filter which was analyzed in Chap. 4.

On the other side, the concept of Derivative-free distributed nonlinear Kalman Filter has been described in Chaps. 4 and 5. It is assumed that distributed filtering is used to estimate with accuracy the position and motion characteristics of each agricultural vehicle. Each local filtering within this distributed state estimation scheme provides an estimate of the vehicle's state vector by fusing measurements from multiple RTK-GPS units. If the derivative-free Kalman Filter is used in place of the Extended Kalman Filter, then in the EIF formulation, given in Eqs. (4.53) and (4.54) the following matrix substitutions should be performed: $J_{\phi}(k) \rightarrow A_d$, $J_{\gamma}(k) \rightarrow C_d$, where matrices A_d and C_d are the discrete-time equivalents of matrices A and C

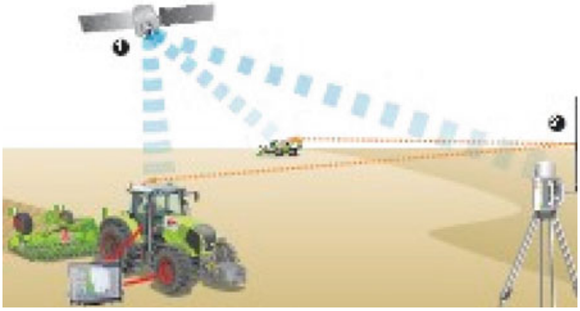


Fig. 6.27 Precise localization of the cooperating vehicles through multisensor fusion and distributed filtering

which have been defined in Eq. (6.26). Both the stages of the Kalman Filter and of the Extended Kalman Filter have been explained in Chap. 4. Matrices A_d and C_d can be computed using established discretization methods. Moreover, the covariance matrices $P(k)$ and $P^{-}(k)$ are the ones obtained from the linear Kalman Filter update equations given also in Chap. 4.

It is assumed now that the vehicle's velocity has to be estimated through the processing of the sequence of position measurements by a filtering algorithm. To this end, the derivative-free Kalman Filter for MIMO nonlinear dynamical systems can been used. From the application of differential flatness theory presented in Sect. 6.2.2 for transforming the initial nonlinear vehicle's model into a linearized equivalent that is finally written in the Brunovsky form, one has Eq. (6.12) which means $\ddot{x} = u_1$ and $\ddot{y} = u_2$. Next, the state variables $x_1 = x$, $x_2 = \dot{x}$, $x_3 = y$, and $x_4 = \dot{y}$ are defined. Considering the state vector $x \in R^{4\times 1}$, the following matrices are also defined

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \tag{6.26}$$

Using the matrices of Eq. (6.26), one obtains the Brunovsky form of the MIMO robot model $\dot{x} = Ax + Bv$ and $y = Cx$, where the new input v is given by $v = [u_1(x,t), u_2(x,t)]^T$. This is a robotic model, for which state estimation can be performed using the standard Kalman Filter recursion.

6.3.3 Simulation Tests

Master-slave cooperation of two agricultural robots was considered (see Fig. 6.26). The master tractor generates a reference path and the motion characteristics (velocity, acceleration, orientation) that the slave tractor has to follow. It was assumed that measurements of the xy coordinates of the vehicles could be obtained through multiple GPS units (localization of moderate accuracy), or multiple local RTK-GPS stations

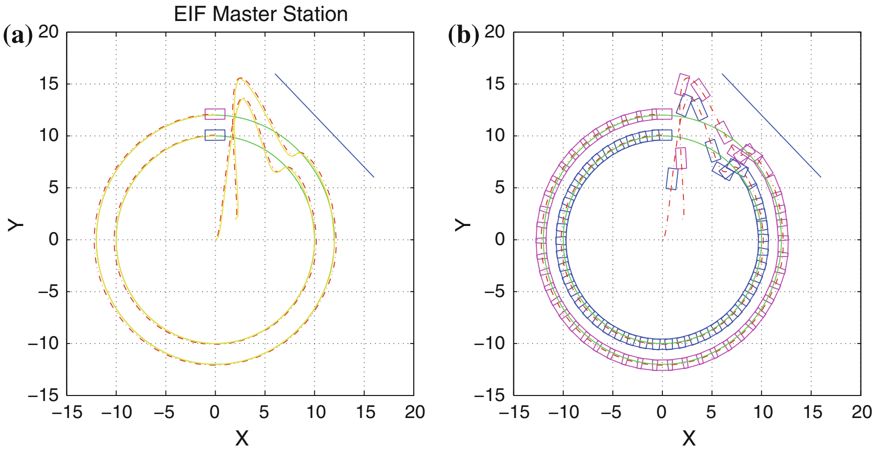


Fig. 6.28 Extended Information Filtering and flatness-based control for cooperating robot harvesters: **a** synchronized tracking of reference path 1, **b** position of the synchronized vehicles every 100 sampling periods

(localization of higher accuracy). Moreover, localization of the vehicles could be performed using measurements of their distance from a reference surface. This distance can be measured with the use of different onboard sensors, e.g., laser, sonar, or vision sensors. The measurements from the GPS were combined with the distance sensor measurements and were initially processed by local filters to provide local state vector estimates. At a second stage, the local state estimates for the robotic vehicles were fused using the Extended Information Filter. Using the outcome of the Extended Information Filter state, estimation-based control was implemented.

Indicative results about tracking of various trajectories (e.g., reference paths followed by the vehicles to perform harvesting) with use of the Extended Information Filter are shown in Figs. 6.28, 6.29, 6.30, and 6.31. It can be noticed that the Extended Information Filter provides accurate estimates of the vehicle's state vector thus also resulting in efficient tracking of the reference trajectories. Finally, it is noted that the paper's approach can be applied also to various types of 4WD agricultural vehicles.

6.4 Distributed Fault Diagnosis for Autonomous Vehicles

6.4.1 Integrity Testing in Navigation Sensors of AGVs

Integrity testing of navigation sensors in Automatic Ground Vehicles (AGVs) can be succeeded through the processing of measurements from distributed sources [378, 427, 515]. On the one side, one can have estimation of the AGV's cartesian coordinates through the fusion of measurements coming from various onboard sensors

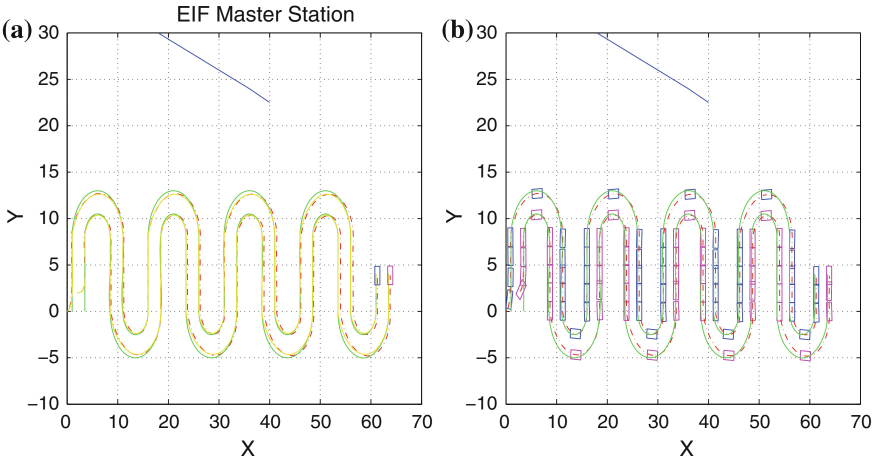


Fig. 6.29 Extended Information Filtering and flatness-based control for cooperating robot harvesters: **a** synchronized tracking of reference path 2, **b** position of the synchronized vehicles every 100 sampling periods

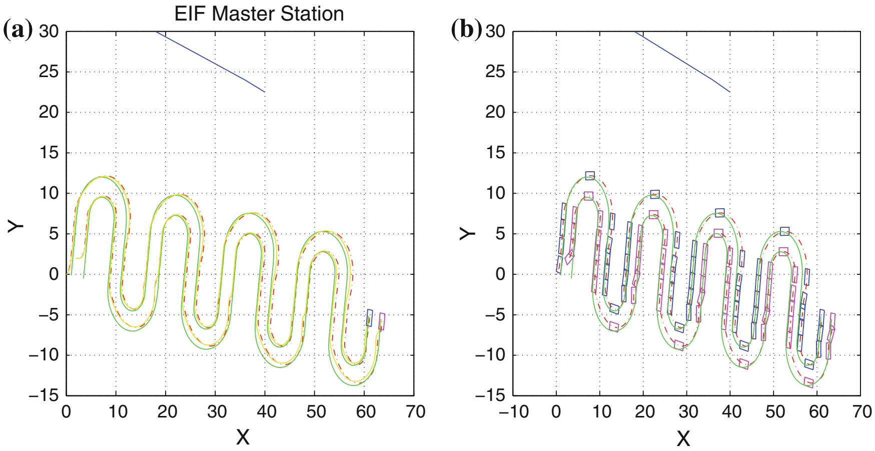


Fig. 6.30 Extended Information Filtering and flatness-based control for cooperating robot harvesters: **a** synchronized tracking of reference path 3, **b** position of the synchronized vehicles every 100 sampling periods

(considered as reference sensors, e.g., GPS devices and distance measuring devices such as laser and vision sensors). Based on these measurements, distributed filtering algorithms produce estimates of the vehicle's state vector through a multistage fusion procedure. On the other side, one can have an estimate of the AGV's cartesian coordinates using the measurements provided by failure-prone sensors such as wheel encoders or IMU. Residuals generation takes place through the comparison of the

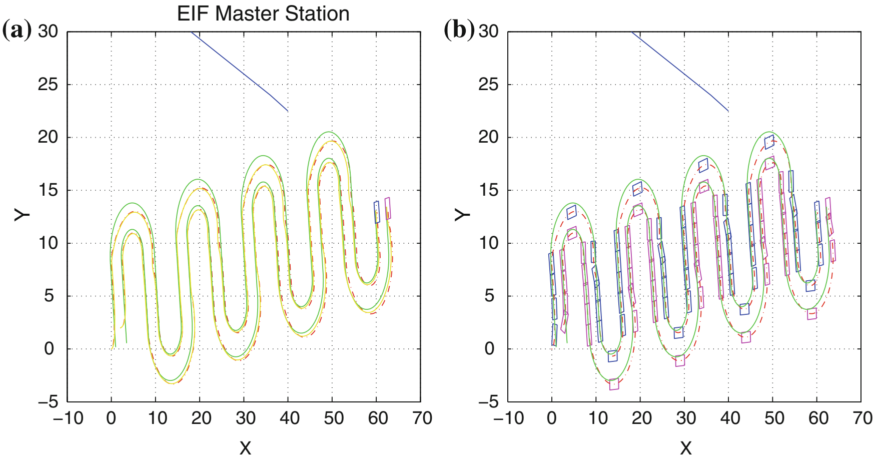


Fig. 6.31 Extended Information Filtering and flatness-based control for cooperating robot harvesters: **a** synchronized tracking of reference path 4, **b** detailed motion of the synchronized vehicles

coordinates of the vehicle computed through the measurements of the monitored sensors against the coordinates which are estimated by processing measurements from the reference sensors. Finally, the statistical processing of residuals by a suitable fault detection and isolation algorithm can provide an indication about the condition of the navigation sensors and failures that may have appeared [31, 157, 290, 414].

This section implements distributed filtering with the use of the Derivative-free Extended Information Filter (DEIF), for generating such residuals. As explained, to apply the Derivative-free Extended Information Filter it is necessary first to perform transformation of the system's dynamic or kinematic model into a canonical form, and this is achieved by using a change of coordinates (diffeomorphism) that is based on differential flatness theory. Unlike the Extended Information Filter (EIF), the proposed filter avoids approximation errors caused by the linearization of the AGV kinematic model and does not require the computation of Jacobians [112, 264, 369]. The Derivative-free Extended Information Filter appears to be faster than the EIF, while also providing very accurate (in terms of variance) state estimates. The application of the Derivative-free Extended Information Filter to AGVs confirms and extends the initial results about the performance of derivative-free nonlinear Kalman Filtering given in [427].

6.4.2 Sensor Fusion for AGV Navigation

To improve the accuracy of the estimation of the AGV's state vector, fusion of the measurements provided by onboard sensors can be performed (IMU, GPS, gyrocompasses, laser, vision sensors). The inertial coordinates system OXY is defined.

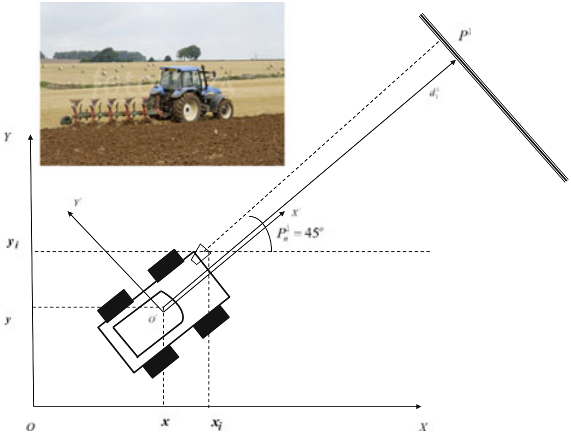


Fig. 6.32 Reference frames for the AGV

Furthermore, the coordinates system $O'X'Y'$ is considered (Fig. 6.32). $O'X'Y'$ results from OXY if it is rotated by an angle θ. The coordinates of the center of gravity of the AGV with respect to OXY are (x, y). It is assumed that the coordinates of the distance measuring sensor (e.g., vision sensor or radar) with respect to $O'X'Y'$ are $x_i^{'}, y_i^{'}$. The orientation of the AGV with respect to OXY is $\theta_i^{'}$. Thus the coordinates of the distance measuring sensor with respect to OXY are (x_i, y_i) and its orientation is θ_i, and are given by:

$$\begin{array}{c} x_i(k) = x(k) + x_i^{'}sin(\theta(k)) + y_i^{'}cos(\theta(k)) \\ y_i(k) = y(k) - x_i^{'}cos(\theta(k)) + y_i^{'}sin(\theta(k)) \\ \theta_i(k) = \theta(k) + \theta_i^{'} \end{array} \tag{6.27}$$

For the localization of the AGV, a first set of measurements comprises the AGV's cartesian coordinates and comes from an IMU or a GPS. A second set of measurements consists of the readings of the AGV's distance $d(k)$ from a reference surface P^j, and can be provided from a vision sensor or laser. A reference surface P^j in the AGV's 2D motion area can be represented by P_r^j and P_n^j (see Fig. 6.32), where (i) P_r^j is the normal distance of the AGV from the origin O, (ii) P_n^j is the angle between the normal line to the plane and the x-direction. The sensor providing this second set of measurements is at position $[x_i(k), y_i(k)]$ with respect to the inertial coordinates system OXY and its orientation is $\theta_i(k)$. Using the above notation, the distance of the GPS sensor, from the plane P^j is represented by P_r^j, P_n^j (see Fig. 6.32) [427]:

$$d_i^j(k) = P_r^j - x_i(k)cos(P_n^j) - y_i(k)sin(P_n^j) \tag{6.28}$$

In this section, it will be assumed that the measurements vector is given by $\gamma(x(k)) = [x(k), y(k), d(k)]$, where $[x(k), y(k)]$ are the AGV's cartesian coordinates, and $d(k)$ is the distance measurement. Moreover, it is assumed that at every time instant more than one GPS measurements are available and that multiple estimates of the vehicle's coordinates can be obtained by fusing the individual GPS measurements with the measurement of the distance from the reference surface.

In the case of multisensor fusion with the use of the EKF a constant sampling period $\Delta t_k = T$ is assumed and the measurement equation for the vehicle is $z(k) = \gamma(x(k)) + v(k)$, where $z(k)$ is the vector containing the sequence of measurements of the cartesian coordinates of the vehicle and $v(k)$ is the measurement noise vector. Apart from the coordinates of the vehicle which are provided by the IMU or GPS one can also consider a distance measurement $d(k)$ from a landmark surface, as shown in Fig. 6.32. Thus, the measurements vector becomes

$$z(k) = [x(k) + v_1(k), y(k) + v_2(k), d(k) + v_3(k)],\ k = 1, 2, 3 \ldots \tag{6.29}$$

To obtain the Extended Kalman Filter (EKF), the kinematic model of the vehicle is linearized about the estimates $\hat{x}(k)$ and $\hat{x}^{-}(k)$, and the control input $U(k)$ is applied, as already explained in Sect. 6.2.5.1. The EKF recursion consists of the measurement update part and of the time update part as described in Eqs. (4.13) and (4.14), respectively. The input gain matrix $L(k)$ for the vehicle model can be written in the form

$$L(k) = \begin{pmatrix} Tcos(\theta(k)) & 0 \\ Tsin(\theta(k)) & 0 \\ 0 & T \end{pmatrix} \tag{6.30}$$

while the Jacobian matrices $J_{\phi}(\hat{x}(k))$ and $J_{\gamma}^{T}(\hat{x}^{-}(k))$ are given by

$$J_{\phi}(\hat{x}(k)) = \begin{pmatrix} 1 & 0 & -v(k)sin(\hat{\theta}(k))T \\ 0 & 1 & v(k)cos(\hat{\theta}(k))T \\ 0 & 0 & 1 \end{pmatrix} \tag{6.31}$$

$$J_{\gamma}(\hat{x}^{-}(k)) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -cos(P_n) & -sin(P_n) & -x_i cos(\hat{\theta}(k) - P_n) - y_i sin(\hat{\theta}(k) - P_n) \end{pmatrix} \tag{6.32}$$

For the elements of the noise covariance matrix $Q(k) = diag[\sigma^2(k), \sigma^2(k), \sigma^2(k)]$ indicative values can be $\sigma^2(k) = 10^{-3}$. Using the estimated state vector, function $\gamma(x)$ appearing in the measurements equations part of the vehicle's kinematic model becomes $\gamma(\hat{x}(k)) = [\hat{x}(k), \hat{y}(k), \hat{d}(k)]$.

6.4.3 Canonical Form for the AGV Model

From the application of the differential flatness theory presented in Sect. 6.2, the initial nonlinear vehicle's model is transformed into a linearized equivalent. This is finally written in the Brunovsky form, and one has Eq. (6.15) which means $\ddot{x} = u_1$ and $\ddot{y} = u_2$. Next, the state variables $x_1 = x$, $x_2 = \dot{x}$, $x_3 = y$ and $x_4 = \dot{y}$ are used. Considering the state vector $x \in R^{4 \times 1}$, the following matrices are also defined

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \\ C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{6.33}$$

Using the matrices of Eq. (6.33), one obtains the Brunovsky (canonical) form of the MIMO model of the AGV.

$$\begin{aligned} \dot{x} &= Ax + Bv \\ y &= Cx \end{aligned} \tag{6.34}$$

where the new input v is given by $v = [u_1(x,t), u_2(x,t)]^T$. For the system of Eqs. (6.33) and (6.34), state estimation is possible by applying the standard Kalman Filter. The system is first turned into discrete-time form using common discretization methods and then the recursion of the linear Kalman Filter described in Eqs. (4.5) and (4.6) is applied.

6.4.4 Derivative-Free Extended Information Filtering for UGVs

As mentioned above, for the system of Eq. (6.33), state estimation is possible by applying the standard Kalman Filter. The system is first turned into discrete-time form using common discretization methods and then the recursion of the linear Kalman Filter described in Eqs. (4.5) and (4.6) is applied.

If the derivative-free Kalman Filter is used in place of the Extended Kalman Filter then in the EIF formulation, given in Eqs. (4.53) and (4.54), the following matrix substitutions should be performed: $J_\phi(k) \rightarrow A_d$, $J_\gamma(k) \rightarrow C_d$, where matrices A_d and C_d are the discrete-time equivalents of matrices A and C which have been defined Eq. (6.33) and which appear also in the measurement and time update of the standard Kalman Filter recursion. Matrices A_d and C_d can be computed using established discretization methods. Moreover, the covariance matrices $P(k)$ and $P^-(k)$ are the ones obtained from the linear Kalman Filter update equations given in Sect. 4.2.2.

Finally, state estimates for the AGV which are provided by multiple local Kalman Filters are fused into one single state estimate, either with the use of the Extended Information Filter or with the use of the Derivative-free distributed nonlinear Kalman Filter.

6.4.5 Simulation Tests

Distributed state estimation takes place in two levels (i) at the lower level local state estimates are generated by local nonlinear filters, such as Extended Kalman Filters or derivative-free nonlinear Kalman Filters. The latter, provide local state estimates through the fusion of the cartesian coordinates of the AGV with the distance of the AGV from a reference surface, (ii) at the higher level, fusion of the local state estimates is performed with the use of distributed state estimation approaches, such as the Extended Information Filter, and the previously described Derivative-Free Extended Information Filter.

It is considered that the previously analyzed AGV model is monitored by $n = 2$ local filters that fuse different GPS measurements of the vehicle's coordinates with a measurement of the distance from the reference surface (provided by a vision or laser sensor). The distributed filtering architecture and the residuals generation procedure is shown in Fig. 6.33. The reference frames defining the AGV's motion and the AGV's distance from the reference surface are according to Fig. 6.32 and have been analyzed in [408, 427]. To decide on the existence of a navigation sensor failure or intermittence, the produced residuals can be tested against a fault threshold that is chosen according to the Generalized Likelihood Ratio criterion or the χ^2 test [31, 290, 414].

In the simulation experiments, various AGV trajectories were considered and an indicative one (circular path of radius equal to 10 m) is depicted in Fig. 6.34 and 6.37. Regarding computation time it was observed that the Derivative-free Extended Information Filter was significantly faster than the Extended Information Filter, due to no need to compute online Jacobian matrices and partial derivatives. Results on the EIF and the Derivative-free EIF performance, in estimating the state vector the AGV when fusing measurements from multiple onboard sensors, are given in Figs. 6.35, 6.36, 6.38, and 6.39, respectively. In case of an IMU or encoder fault (additive fault to the sensor's output), one can observe deviation between the distributed filtering-based coordinates and the coordinates which are computed using the measurements of the monitored sensor. Further processing of the residuals by a FDI algorithm with optimally selected fault threshold enables detection of incipient navigation sensor faults as well as fault isolation [31, 157, 290, 414].

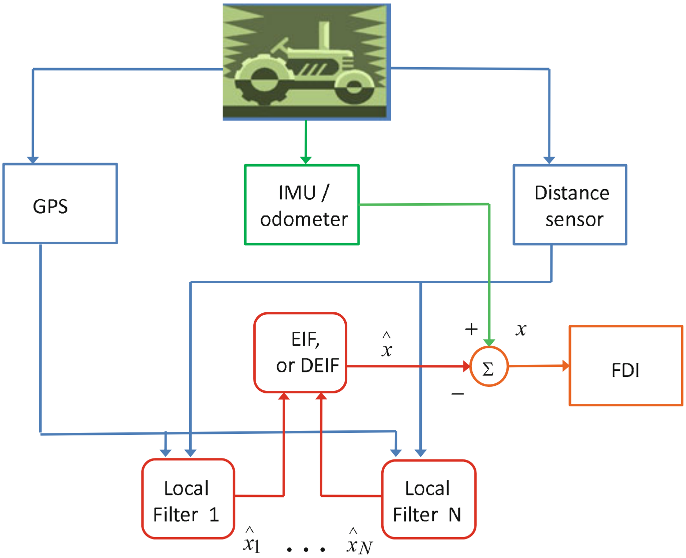


Fig. 6.33 Integrity monitoring for the AGV navigation sensors using a distributed filtering scheme

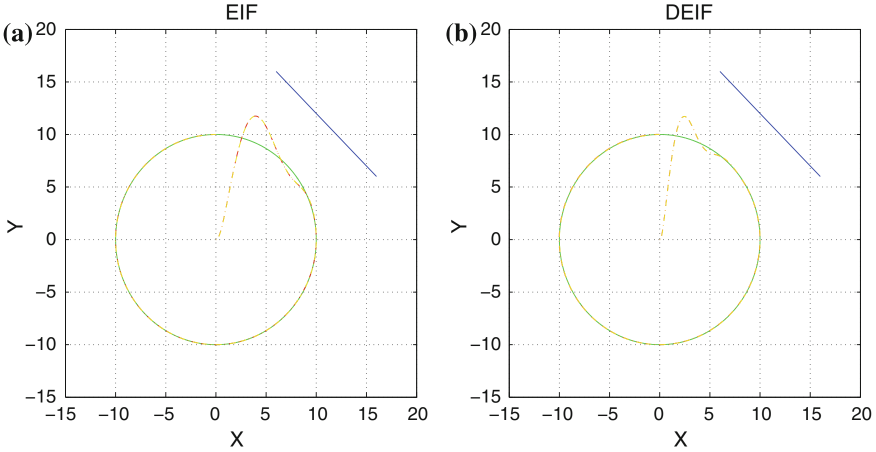


Fig. 6.34 Fault-free sensor case. Trajectory generated by state estimation-based control of the AGV with the use of: **a** standard Extended Information Filter, **b** Derivative-free nonlinear Extended Information Filter

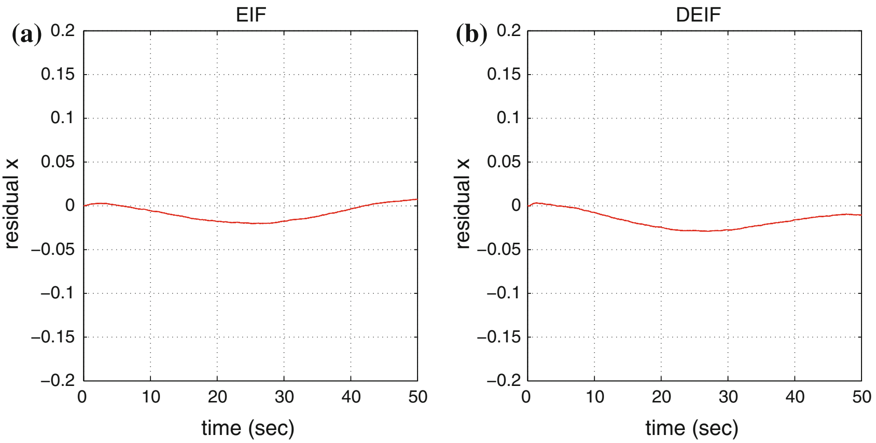


Fig. 6.35 Fault-free sensor case. Residual of the x-axis position of the AGV generated with **a** standard Extended Information Filter, **b** Derivative-free nonlinear Extended Information Filter

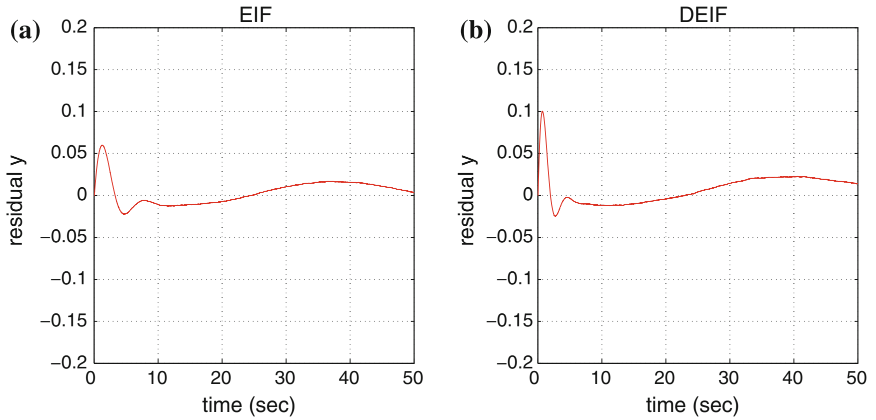


Fig. 6.36 Fault-free sensor case. Residual of the y-axis position of the AGV generated with **a** standard Extended Information Filter, **b** Derivative-free nonlinear Extended Information Filter

6.5 Velocity Control of 4-Wheel Vehicles

6.5.1 Overview

Next, it will be shown that differential flatness theory-based filtering and control can be applied to more advanced models of autonomous vehicles, such as the 4-wheel ground vehicle. The precise modeling of the vehicles' dynamics improves the efficiency of vehicles controllers in adverse cases, for example in high velocity, when performing abrupt maneuvers, under mass and loads changes or when moving on

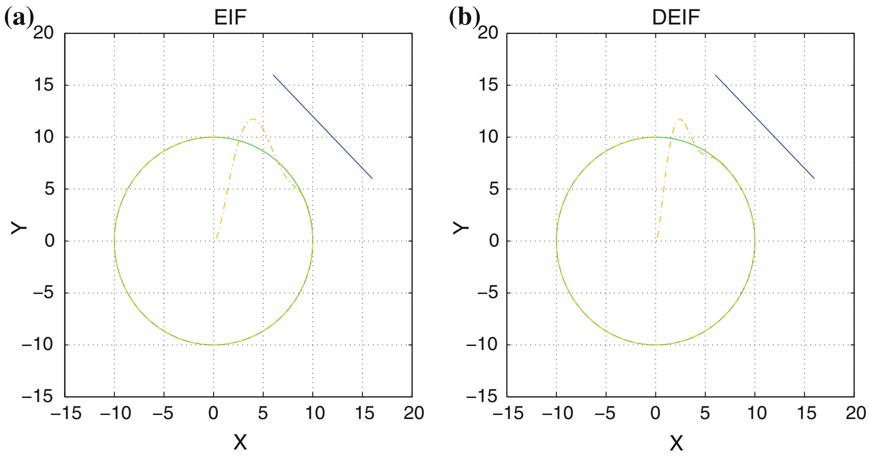


Fig. 6.37 Faulty sensor case. Trajectory generated by state estimation-based control of the AGV with the use of: **a** standard Extended Information Filter, **b** Derivative-free nonlinear Extended Information Filter

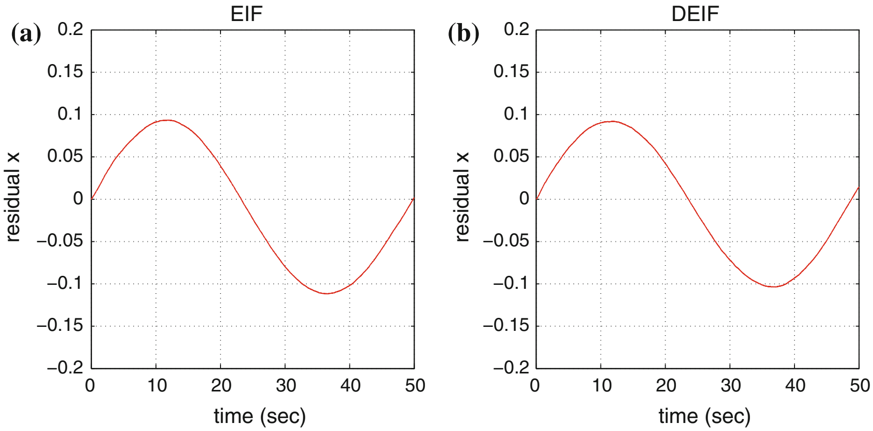


Fig. 6.38 Faulty sensor. Residual of the x-axis position of the AGV generated with **a** standard Extended Information Filter, **b** Derivative-free nonlinear Extended Information Filter

rough terrain. Using model-based control approaches, it is possible to design a nonlinear controller that maintains the vehicle's motion characteristics within desirable ranges [39, 331, 337, 349, 350, 360, 404, 406, 516, 589]. When the vehicle's dynamics is subject to modeling uncertainties or when there are unknown forces and torques exerted on the vehicle it is important to be in position to estimate in real-time disturbances and unknown dynamics so as to compensate for them through the control input and to maintain the satisfactory performance of the vehicle's automated steering system. To this direction, estimation for the unknown dynamics of the vehicle and state estimation-based control schemes have been developed [209, 331, 496, 545].

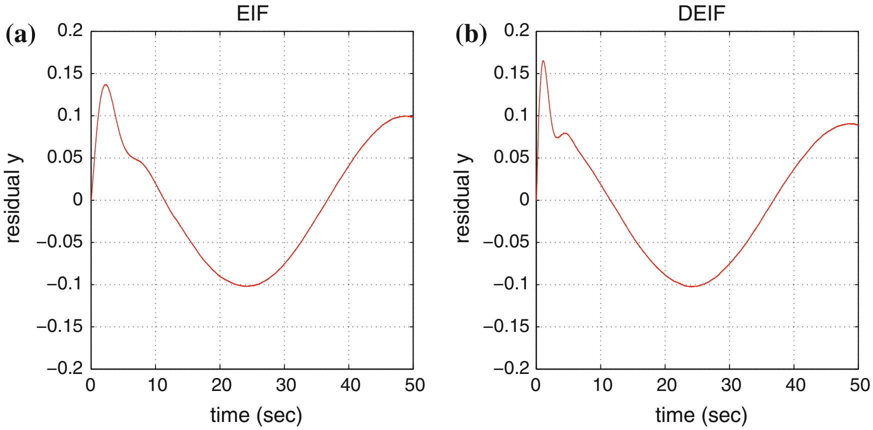


Fig. 6.39 Faulty sensor case. Residual of the y-axis position of the AGV generated with **a** standard Extended Information Filter, **b** Derivative-free nonlinear Extended Information Filter

The objective of this section is two-fold. On the one side it analyzes the design of a controller for autonomous navigation of automatic ground vehicles (AGVs). On the other side, it proposes a solution to the problem of four-wheel vehicle control under model uncertainties and external disturbances. Considering, that only under ideal conditions the dynamic model of the vehicle is precisely known (e.g., there may be variations in the transported mass, or in the cornering stiffness coefficients characterizing the interaction of the tires with the ground, or in the position of the vehicle's center of gravity) and that in several cases there is uncertainty about the forces and torques developed on the vehicle (e.g., traction and braking torques on the wheels, forces due to traction of implements, or lateral forces which generate torques affecting the yaw stability of the vehicle), the need for designing robust controllers of the autonomous vehicles becomes obvious [44, 488, 497, 554]. By compensating efficiently such disturbance forces and torques, safety features of the vehicle are improved and its autonomous functioning remains reliable even under adverse road conditions.

Dynamic analysis for the 4-wheel vehicle is provided. A 3-DOF model is introduced having as elements the vehicle's velocity along the horizontal and vertical axis of an inertial reference frame as well as the rate of change of its orientation angle (this is the angle defined by the vehicle's longitudinal axis and the horizontal axis of the frame). Lateral forces are shown to affect the vehicle's motion and to be dependent on the longitudinal and lateral velocity of the vehicle, on the yaw rate and on the cornering stiffness coefficients for the front and rear tires. The control inputs to the vehicles' dynamic model are the traction/bracking wheel torque and the turn angle of the steering wheel. Since the parameters of the dynamic model of the vehicle cannot always be known with precision or may be time-varying (e.g., cornering stiffness coefficients, transported mass) and since there may be unmodeled external forces and torques exerted on the vehicle (e.g., due to road condition, disturbances in traction forces) it is important to design a control loop with robustness to the aforementioned sources of uncertainty and disturbances, as well as to be in position to estimate in real-time such disturbances through the processing of measurements from a small number of onboard sensors.

Next, it is shown how a nonlinear controller for the aforementioned vehicle's model can be obtained through the application of differential flatness theory [152, 465, 516, 535]. The flat output for the vehicle's model is a vector comprising the x-axis velocity and a second variable based on a linear relation between the y-axis velocity and the rate of change of the orientation angle [349, 350]. By expressing all state variables and the control input of the four-wheel vehicle model as functions of the flat output and its derivatives, the system's dynamic model is transformed into the linear Brunovksy (canonical) form [317, 479]. For the latter model, it is possible to design a state feedback controller that succeeds accurate tracking of the vehicle's velocity setpoints.

By exploiting the vehicle's exactly linearized model and its transformation into a canonical form it is possible to design a linear state estimator for approximating the system's state vector through the processing of measurements coming from a small number of onboard sensors. To this end, *Derivative-free nonlinear Kalman Filtering* is used. Unlike the Extended Kalman Filter, the proposed filtering method provides estimates of the state vector of the nonlinear system without the need for derivatives and Jacobians calculation [422, 427]. By avoiding linearization approximations, the proposed filtering method improves the accuracy of estimation of the system's state variables. Moreover, it is shown that it is possible to redesign the Kalman Filter in the form of a disturbance observer, and using the estimation of the disturbance to develop an auxiliary control input that compensates for their effects. In this way, the vehicle's control and autonomous navigation system can become robust with respect to uncertainties in the model's parameters or uncertainties about external forces and torques. It is also noted that in terms of computation speed, the proposed Kalman Filter-based disturbance estimator for the vehicle is faster than disturbance estimators that may be based on other nonlinear filtering approaches (e.g., Extended Kalman Filter, Unscented Kalman Filter or Particle Filter) thus becoming advantageous for the real-time estimation of the unknown vehicle dynamics [436, 437].

The efficiency of the proposed nonlinear control and Kalman Filter-based disturbances estimation scheme is evaluated through numerical simulation tests. It will be shown that by accurately estimating disturbance forces and torques, the control loop succeeds elimination of the tracking error for all state variables of the vehicle.

6.5.2 Dynamic Model of the Vehicle

6.5.2.1 Definition of Parameters in 4-Wheel Vehicle Dynamic Model

With reference to Fig. 6.40 (where the lateral forces applied on the wheels are considered to define the vehicle's motion) one has the following parameters: β is the angle

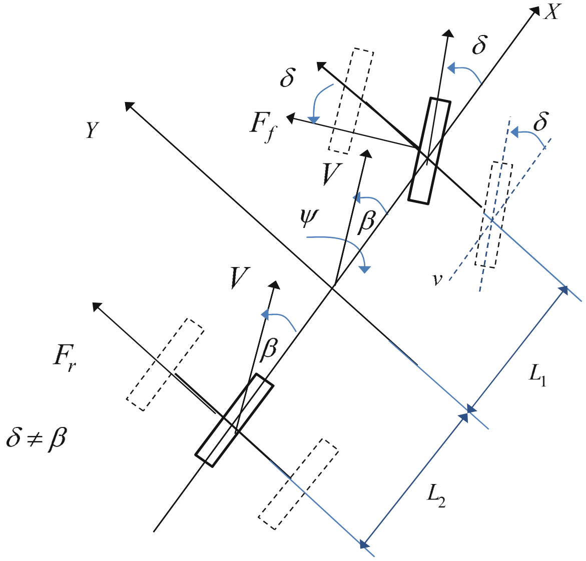


Fig. 6.40 Nonlinear 4-wheeled vehicle model

between the velocity and the vehicle's transversal angle, V is the velocity vector of the vehicle, ψ is the yaw angle (rotation round the z axis), f_x: is the aggregate force along the x axis, f_y is the aggregate force along the y axis, T_z is the aggregate torque round the z axis, and δ is the steering angle of the front wheels [349, 360, 535].

The motion of the vehicle is described by the following set of equations:

1. Longitudinal motion

$$-mV(\dot{\beta}+\dot{\psi})sin(\beta)+m\dot{V}cos(\beta)=f_x \tag{6.35}$$

2. Lateral motion

$$mV(\dot{\beta}+\dot{\psi})cos(\beta)+m\dot{V}sin(\beta)=f_y \tag{6.36}$$

3. yaw turn

$$I\ddot{\psi}=T_z \tag{6.37}$$

The above-described vehicle dynamics can be also written in matrix form

$$\begin{pmatrix} -sin(\beta) & cos(\beta) & 0 \\ cos(\beta) & sin(\beta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} mV(\dot{\beta}+\dot{\psi}) \\ m\dot{V} \\ I\ddot{\psi} \end{pmatrix} = \begin{pmatrix} f_x \\ f_y \\ T_z \end{pmatrix} \tag{6.38}$$

Finally, a matrix relation is provided about the transformation of forces on a tire into forces and torques along the vehicle's axes:

$$\begin{pmatrix} f_x \\ f_y \\ T_z \end{pmatrix} = \begin{pmatrix} -sin(\delta) & 0 \\ cos(\delta) & 1 \\ L_1 cos(\delta) & -L_2 \end{pmatrix} \begin{pmatrix} F_f \\ F_r \end{pmatrix} \tag{6.39}$$

6.5.2.2 Vehicle Dynamical Model with Longitudinal and Lateral Forces

The previous model of Fig. 6.40 is reexamined considering that $\dot{\beta} = 0$ and that ψ is the yaw angle formed between the vehicle's longitudinal axis and the horizontal axis of an inertial reference frame. Moreover, it is assumed that apart from the lateral forces, there are traction torques transferred from the engine to the front wheels as well as braking torques on the rear and front wheels. Due to the distance between the wheels axes and the vehicle's center of gravity, torques are also generated along the vehicle's z-axis. With reference to Fig. 6.41, the model of the vehicle's dynamics is formulated as follows [349, 360, 535]:

$$\begin{aligned} m\alpha_x &= m(\dot{V}_x - \dot{\psi}\dot{V}_y) = F_{x_1} + F_{x_2} \\ m\alpha_y &= m(\dot{V}_y + \dot{\psi}\dot{V}_x) = F_{y_1} + F_{y_2} \\ I_z\ddot{\psi} &= T_{z_1} + T_{z_2} \end{aligned} \tag{6.40}$$

where a_x and a_y are accelerations along the axes of the inertial reference frame and $\dot{V}_x$, $\dot{V}_y$ in a reference frame that rotates with the yaw rate $\dot{\psi}$. The forces F_{x_i}, $i = 1, 2$

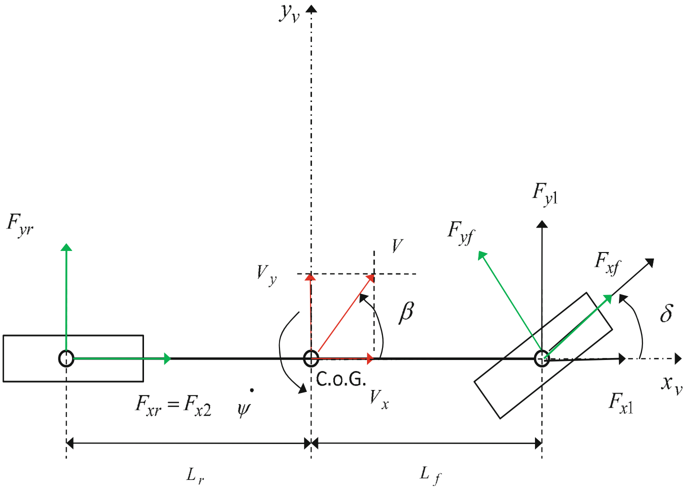


Fig. 6.41 Vehicle model with longitudinal and lateral forces

on the vehicle's longitudinal axis and $F_{y_i},\ i=1,2$ on the vehicle's transversal axis are computed from the horizontal and vertical forces applied on the vehicle's wheels as follows:

$$\begin{aligned} F_{x_1} &= F_{x_f}cos(\delta) - F_{y_f}sin(\delta) \\ F_{x_2} &= F_{x_r} \\ F_{y_1} &= F_{y_f}sin(\delta) + F_{y_f}cos(\delta) \\ F_{y_2} &= F_{y_r} \\ T_{z_1} &= L_f(F_{y_f}cos(\delta) + F_{x_f}sin(\delta)) \\ T_{z_2} &= -L_rF_{y_r} \end{aligned} \tag{6.41}$$

About the longitudinal and the lateral forces applied to the vehicle one has:

1. Longitudinal force on the front wheel

$$F_{x_f} = \left(\frac{1}{R}\right)(I_r\dot{\omega}_f + T_m - T_{b_f}) \tag{6.42}$$

2. Longitudinal force on the rear wheel

$$F_{x_r} = -\left(\frac{1}{R}\right)(T_{b_r} + I_r\dot{\omega}_r) \tag{6.43}$$

3. Lateral force on the front wheel (taking that the angle β between the vehicle's longitudinal axis and the wheel's velocity vector is approximated by $\beta = \frac{V_y+\dot{\psi}L_f}{V_x}$)

$$F_{y_f} = C_f\left(\delta - \frac{V_y+\dot{\psi}L_f}{V_x}\right) \tag{6.44}$$

4. Lateral force on the rear wheel (taking that for the rear wheel the steering angle is $\delta=0$ and that the angle β between the vehicle's longitudinal axis and the wheel's velocity vector is approximated by $\beta = \frac{V_y-\dot{\psi}L_r}{V_x}$).

$$F_{y_r} = -C_r\frac{V_y-\dot{\psi}L_r}{V_x} \tag{6.45}$$

where C_f and C_r are the cornering stiffness coefficients for the front and rear tires, respectively. Nominal values of these cornering stiffness coefficients can be estimated through identification procedures. The substitution of Eq. (6.42) to Eq. (6.45) into Eq. (6.40) results into

$$\begin{aligned} m\dot{V}_x &= m\dot{\psi}V_y - \tfrac{I_r}{R}(\dot{\omega}_r + \dot{\omega}_f) + \tfrac{1}{R}(T_m - T_{b_f} - T_{b_r}) + C_f(\tfrac{V_y+\dot{\psi}L_f}{V_x})\delta - C_f\delta^2 \\ m\dot{V}_y &= -m\dot{\psi}V_x - C_f(\tfrac{V_y+\dot{\psi}L_f}{V_x}) - C_r(\tfrac{V_y-\dot{\psi}L_f}{V_x}) + (\tfrac{1}{R})(T_m - T_{b_f})\delta + (C_f - \tfrac{I_r}{R}\dot{\omega}_f)\delta \\ I_z\ddot{\psi} &= -L_fC_f(\tfrac{V_y+\dot{\psi}L_f}{V_x}) + L_rC_r(\tfrac{V_y-\dot{\psi}L_f}{V_x}) + \tfrac{L_f}{R}(T_m - T_{b_f})\delta + L_f(T_m - \tfrac{I_r}{R})\delta \end{aligned} \tag{6.46}$$

The motion of the vehicle along its longitudinal axis is controlled by the traction or braking wheel torque $T_\omega = T_m - T_b$ with $T_b = T_{b_f} + T_{b_r}$ and the lateral movement via the steering angle δ. The two control inputs of the four-wheel vehicle model are

$$\begin{aligned} u_1 &= T_\omega \\ u_2 &= \delta \end{aligned} \tag{6.47}$$

A first form of the vehicle's dynamic model is

$$\dot{x} = f(x,t) + g(x,t)u + g_1 u_1 u_2 + g_2 u_2^2 \tag{6.48}$$

where

$$f(x,t) = \begin{pmatrix} \frac{I_r}{mR}(\dot{\omega}_r + \dot{\omega}_f) \\ \dot{\psi} V_x + \frac{1}{m}\left(-C_f \frac{(V_y + L_f \dot{\psi})}{V_x} - C_r \frac{(V_y - L_f \dot{\psi})}{V_x}\right) \\ \frac{1}{I_z}\left(-L_f C_f \frac{(V_y + L_f \dot{\psi})}{V_x} + L_r C_r \frac{(V_y - L_f \dot{\psi})}{V_x}\right) \end{pmatrix} \tag{6.49}$$

$$g(x,t) = \begin{pmatrix} \frac{1}{mR} & \frac{C_f}{m}\left(\frac{V_y + L_f \dot{\psi}}{V_x}\right) \\ 0 & \left(\frac{C_f R - I_r \dot{\omega}_f}{mR}\right) \\ 0 & \frac{(L_f C_f R - L_f I_r \dot{\omega}_f)}{I_z R} \end{pmatrix} \tag{6.50}$$

$$g_1 = \begin{pmatrix} 0 \\ \frac{1}{mR} \\ \frac{L_f}{I_z R} \end{pmatrix} \quad g_2 = \begin{pmatrix} -\frac{C_f}{m} \\ 0 \\ 0 \end{pmatrix} \quad x = \begin{pmatrix} V_x \\ V_y \\ \dot{\psi} \end{pmatrix} \quad u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{6.51}$$

The previously analyzed nonlinear model of the vehicle's dynamics can be simplified if the control inputs $u_1 u_2$ and u_2^2 are not taken into account. In the latter case, the dynamics of the vehicle takes the form

$$\dot{x} = f(x,t) + g(x,t)u \tag{6.52}$$

6.5.3 Flatness-Based Controller for the 3-DOF Vehicle Model

6.5.3.1 Flatness-Based Controller for the 4-Wheel Vehicle

To show that the four-wheel vehicle is differentially flat, the following flat outputs are defined [349, 350]:

$$\begin{aligned} y_1 &= V_x \\ y_2 &= L_f m V_y - I_z \dot{\psi} \end{aligned} \tag{6.53}$$

Then it holds that all elements of the system's state vector can be written as functions of the flat outputs and their derivatives. Indeed, for $x = [V_x,\ V_y,\ \dot{\psi}]^T$ it holds

$$V_x = y_1 \tag{6.54}$$

$$V_y = \frac{y_2}{L_f m} - \left(\frac{I_z}{L_f m}\right)\left(\frac{L_f m y_1 \dot{y}_2 + C_r(L_f + L_r)y_2}{C_r(L_f + L_r)(I_z - L_f L_r m) + (L_f m y_1)^2}\right) \tag{6.55}$$

$$\dot{\psi} = \frac{L_f m y_1 \dot{y}_2 + C_r(L_f + L_r)y_2}{C_r(L_f + L_r)(I_z - L_f L_r m) + (L_f m y_1)^2} \tag{6.56}$$

Expressing the system's state variables as functions of the flat outputs, one has the following state-space description for the system

$$\begin{pmatrix} \dot{y}_1 \\ \ddot{y}_2 \end{pmatrix} = \Delta(y_1, y_2, \dot{y}_2)\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \Phi(y_1, y_2, \dot{y}_2) \tag{6.57}$$

where

$$\Delta(y_1, y_2, \dot{y}_2) = \begin{pmatrix} \Delta_{11}(y_1, y_2, \dot{y}_2) & \Delta_{12}(y_1, y_2, \dot{y}_2) \\ \Delta_{21}(y_1, y_2, \dot{y}_2) & \Delta_{22}(y_1, y_2, \dot{y}_2) \end{pmatrix} \tag{6.58}$$

with

$$\Delta_{11}(y_1, y_2, \dot{y}_2) = \frac{1}{mR} \tag{6.59}$$

$$\Delta_{12}(y_1, y_2, \dot{y}_2) = \tfrac{C_f}{m}(\tfrac{V_y + L_f \dot{\psi}}{y_1}) \tag{6.60}$$

$$\Delta_{21}(y_1, y_2, \dot{y}_2) = \tfrac{C_r(L_f + L_r)(V_y - L_r \dot{\psi}) - L_f m \dot{\psi} y_1^2}{m R y_1^2} \tag{6.61}$$

$$\begin{aligned} \Delta_{22}(y_1, y_2, \dot{y}_2) = \left(-L_f m y_1 + \tfrac{L_r C_r (L_f + L_r)}{y_1}\right) \tfrac{(L_f C_f R - L_f I_r \dot{\omega}_f)}{I_z R} + \\ + \tfrac{((C_r(L_f + L_r))(V_y - L_r \dot{\psi}) - L_f m \dot{\psi} y_1^2)}{y_1^2} \cdot \tfrac{C_f (V_y + L_f \dot{\psi})}{m y_1} - \tfrac{C_r(L_f + L_r)}{y_1} \tfrac{R C_f - I_r \dot{\omega}_f}{mR} \end{aligned} \tag{6.62}$$

Moreover, about matrix $\Phi(y_1, y_2, \dot{y}_2)$ it holds

$$\Phi(y_1, y_2, \dot{y}_2) = \begin{pmatrix} \Phi_1(y_1, y_2, \dot{y}_2) \\ \Phi_2(y_1, y_2, \dot{y}_2) \end{pmatrix} \tag{6.63}$$

with elements

$$\Phi_1(y_1, y_2, \dot{y}_2) = \dot{\psi} V_y - \tfrac{I_r}{mR}(\dot{\omega}_r + \dot{\omega}_f) \tag{6.64}$$

$$\begin{aligned} \Phi_2(y_1, y_2, \dot{y}_2) = -L_f m y_1 f_3(x,t) - \tfrac{C_r(L_f + L_r)}{y_1} f_2(x,t) + \\ + \tfrac{C_f(L_f + L_r)(V_y - L_r \dot{\psi}) - L_f m \dot{\psi} y_1^2}{y_1^2} f_1(x,t) + \tfrac{L_r C_r (L_f + L_r)}{y_1} f_3(x,t) \end{aligned} \tag{6.65}$$

According to the above, the system's control input can be also written as a function of the flat output and its derivatives. Thus one has

$$\begin{pmatrix}\dot{y}_1\\ \ddot{y}_2\end{pmatrix} = \Delta(y_1, y_2, \dot{y}_2)\begin{pmatrix}u_1\\ u_2\end{pmatrix} + \Phi(y_1, y_2, \dot{y}_2) \tag{6.66}$$

i.e.,

$$\begin{pmatrix}u_1\\ u_2\end{pmatrix} = \Delta^{-1}(y_1, y_2, \dot{y}_2)^{-1}(\begin{pmatrix}\dot{y}_1\\ \ddot{y}_2\end{pmatrix} - \Phi(y_1, y_2, \dot{y}_2)) \tag{6.67}$$

which means that provided that matrix $\Delta(y_1, y_2, \dot{y}_2)$ is invertible, the control input $u = [u_1, u_2]^T$ can be written as a function of the flat output and its derivatives. The nonsingularity of matrix $\Delta(y_1, y_2, \dot{y}_2)$ depends on the determinant

$$det(\Delta(y_1, y_2, \dot{y}_2)) = \frac{(I_r\dot{\omega}_f - C_f R)(L_f^2 y_1^2 m^2 - C_r(L_f + L_r)L_r L_f m + C_r I_z L_r)}{I_z R^2 y_1 m^2} \tag{6.68}$$

This determinant has nonzero values because it holds:

(i) $(I_r\dot{\omega}_f - C_f R) \neq 0$ since for the wheels rotational acceleration one has $\dot{\omega}_f < \frac{C_f R}{I_r}$, and also
(ii) $(L_f^2 y_1^2 m^2 - C_r(L_f + L_r)L_r L_f m + C_r I_z L_r) \neq 0$ when $I_z > L_f m$.

The differentially flat model of the vehicle can be also written in a canonical form after defining the new control input vector

$$\begin{pmatrix}v_1\\ v_2\end{pmatrix} = \Delta(y_1, y_2, \dot{y}_2)\begin{pmatrix}u_1\\ u_2\end{pmatrix} + \Phi(y_1, y_2, \dot{y}_2) \tag{6.69}$$

thus one obtains a MIMO system description into canonical form, i.e.,

$$\begin{pmatrix}\dot{y}_1\\ \dot{y}_2\\ \ddot{y}_2\end{pmatrix} = \begin{pmatrix}0 & 0 & 0\\ 0 & 0 & 1\\ 0 & 0 & 0\end{pmatrix}\begin{pmatrix}y_1\\ y_2\\ \dot{y}_2\end{pmatrix} + \begin{pmatrix}1 & 0\\ 0 & 0\\ 0 & 1\end{pmatrix}\begin{pmatrix}v_1\\ v_2\end{pmatrix} \tag{6.70}$$

Once the vehicle's model is written in the differentially flat form, the controller that enables tracking of a desirable trajectory defined by y_1^{ref}, y_2^{ref}, $\dot{y}_2^{ref}$ is given by

$$\begin{aligned} v_1 &= \dot{y}_1^{ref} - k_{p_1}(y_1 - y_1^{ref}) \\ v_2 &= \ddot{y}_2^{ref} - k_{d_2}(\dot{y}_2 - \dot{y}_2^{ref}) - k_{p_2}(y_2 - y_2^{ref}) \end{aligned} \tag{6.71}$$

and defining the error variables $e_1 = y_1 - y_1^{ref}$ and $e_2 = y_2 - y_2^{ref}$ one has the following tracking error dynamics for the closed-loop system

$$\begin{aligned} \dot{e}_1 + k_{p_1}e_1 &= 0 \\ \ddot{e}_2 + k_{d_2}\dot{e}_2 + k_{p_2}e_2 &= 0 \end{aligned} \tag{6.72}$$

Therefore, the suitable selection of gains $k_{p_1>0}$ and $k_{p_2} > 0, k_{d_2} > 0$ assures the asymptotic elimination of the tracking errors, i.e., $lim_{t\rightarrow\infty}e_1(t) = 0$ and $lim_{t\rightarrow\infty}e_2(t) = 0$.

The control input that is finally applied for the vehicle's steering is given by

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \Delta^{-1}(y_1, y_2, \dot{y}_2)^{-1}(\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} - \Phi(y_1, y_2, \dot{y}_2)) \tag{6.73}$$

or equivalently

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \Delta^{-1}(y_1, y_2, \dot{y}_2)^{-1}[\begin{pmatrix} \dot{y}_1^{ref} - k_{p_1}(y_1 - y_1^{ref}) \\ \ddot{y}_2^{ref} - k_{d_2}(\dot{y}_2 - \dot{y}_2^{ref}) - k_{p_2}(y_2 - y_2^{ref}) \end{pmatrix} - \Phi(y_1, y_2, \dot{y}_2)] \tag{6.74}$$

The transformation of the vehicle's model into a canonical form, through the application of differential flatness theory, facilitates not only the design of a feedback controller for trajectory tracking but also the design of filters for the estimation of the state vector of the vehicle out of a limited number of sensor measurements.

6.5.4 Estimation of Vehicle Disturbance Forces with Kalman Filtering

6.5.4.1 State Estimation with the Derivative-Free Nonlinear Kalman Filter

To account for model uncertainties and external disturbances, observer-based estimation has been proposed in various forms (disturbance observers, extended state observers, etc.) [85, 107, 108, 354].

It was shown that the initial nonlinear model of the vehicle can be written in the MIMO canonical form

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \ddot{y}_2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \dot{y}_2 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \tag{6.75}$$

Thus one has a MIMO linear model of the form

$$\begin{aligned} \dot{y}_f &= A_f y_f + B_f v \\ z_f &= C_f y_f \end{aligned} \tag{6.76}$$

where $y_f = [y_1, y_2, \dot{y}_2]^T$ and matrices A_f,B_f,C_f are in the MIMO canonical form

$$A_f = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \quad B_f = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \quad C_f^T = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \tag{6.77}$$

where the measurable variables $y_1 = V_x$, $y_2 = L_f m V_y - I_z \dot{\psi}$ are associated with the linear velocity of the vehicle V_x, V_y and with its angular velocity $\dot{\psi}$. For the aforementioned model, and after carrying out discretization of matrices A_f, B_f, and C_f with common discretization methods, one can perform linear Kalman filtering using Eqs. (4.5) and (4.6). This is *Derivative-free nonlinear Kalman filtering* for the model of the vehicle which, unlike EKF, is performed without the need to compute Jacobian matrices and does not introduce numerical errors due to approximative linearization with Taylor series expansion.

6.5.4.2 Kalman Filter-Based Estimation of Disturbances

It is assumed that disturbance forces affect the nonlinear vehicle model along its longitudinal and transversal axis and that disturbance torques affect the nonlinear vehicle model on its z axis. For example, disturbance forces be due to a force vector that coincides with the vehicle's longitudinal axis (e.g., traction disturbance) or disturbance torques can be due to unmodeled lateral forces. These disturbance forces and torques change dynamically in time and their dynamics is given by

$$\begin{aligned} \tilde{d}_x &= f_{d_x}(V_x, V_y, \dot{\psi}) \\ \tilde{d}_y &= f_{d_y}(V_x, V_y, \dot{\psi}) \\ \tilde{d}_\psi &= T_{d_\psi}(V_x, V_y, \dot{\psi}) \end{aligned} \tag{6.78}$$

Since the state variables of the vehicle's dynamic model can be written as functions of the flat outputs y_1 and y_2 and of their derivatives it also holds

$$\begin{aligned} \tilde{d}_x^{(i)} &= f_{d_x}^{(i)}(y_1, y_2, \dot{y}_2) \\ \tilde{d}_y^{(i)} &= f_{d_y}^{(i)}(y_1, y_2, \dot{y}_2) \\ \tilde{d}_\psi^{(i)} &= T_{d_\psi}^{(i)}(y_1, y_2, \dot{y}_2) \end{aligned} \tag{6.79}$$

where $i = 1, 2, \ldots$ stands for the ith order derivative of the disturbance variable.

Considering the effect of disturbance functions on the initial nonlinear state equation of the vehicle and the linear relation between the initial state variables $[V_x, V_y]$ and the state variables of the flat system description $[y_1, y_2]$ one has the appearance of the disturbance terms in the canonical form model of Eq. (6.70)

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \ddot{y}_2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \dot{y}_2 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} + \begin{pmatrix} \frac{1}{m}\tilde{d}_x \\ 0 \\ L_f \dot{\tilde{d}}_y - \dot{\tilde{d}}_\psi \end{pmatrix} \tag{6.80}$$

Next, the state vector of the model of Eq. (7.66) is extended to include as additional state variables the disturbance forces $\tilde{d}_x$, $\tilde{d}_y$, and $\tilde{d}_{psi}$. Then, in the new state-space description one has $z_1 = y_1$, $z_2 = y_2$, $z_3 = \dot{y}_2$, $z_4 = \tilde{f}_a = \frac{1}{m}\tilde{d}_x$, $z_5 = \dot{\tilde{f}}_a$, $z_6 = \dot{\tilde{f}}_b = L_f\dot{\tilde{d}}_y - \dot{\tilde{d}}_\psi$, $z_7 = \ddot{\tilde{f}}_b$, which takes the form of matrix equations

$$\dot{z} = \tilde{A}\cdot z + \tilde{B}\cdot\tilde{v} \tag{6.81}$$

where the control input is

$$\begin{gathered} \tilde{v} = \left(v_1 \; v_2 \; \frac{1}{m}\ddot{\tilde{d}}_x \; L_f\tilde{d}_y^{(3)} - \tilde{d}_\psi^{(3)}\right)^T \quad \text{or} \\ \tilde{v} = \left(v_1 \; v_2 \; \ddot{\tilde{f}}_a \; \tilde{f}_b^{(3)}\right)^T \end{gathered} \tag{6.82}$$

with

$$\tilde{A} = \begin{pmatrix} 0&0&0&1&0&0&0 \\ 0&0&1&0&0&0&0 \\ 0&0&0&0&0&1&0 \\ 0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0 \end{pmatrix} \; \tilde{B} = \begin{pmatrix} 1&0&0&0 \\ 0&0&0&0 \\ 0&1&0&0 \\ 0&0&0&0 \\ 0&0&1&0 \\ 0&0&0&0 \\ 0&0&0&1 \end{pmatrix} \; \tilde{C}^T = \begin{pmatrix} 1&0 \\ 0&1 \\ 0&0 \\ 0&0 \\ 0&0 \\ 0&0 \\ 0&0 \end{pmatrix} \tag{6.83}$$

where the measurable state variables are z_1 and z_2. Since the dynamics of the disturbance terms $\tilde{f}_a$ and $\tilde{f}_b$ are taken to be unknown in the design of the associated disturbances' estimator one has the following dynamics:

$$\dot{z}_o = \tilde{A}_o\cdot z + \tilde{B}_o\cdot\tilde{v} + K(C_o z - C_o\hat{z}) \tag{6.84}$$

where $K \in R^{7\times 2}$ is the state estimator's gain and

$$\tilde{A}_o = \begin{pmatrix} 0&0&0&1&0&0&0 \\ 0&0&1&0&0&0&0 \\ 0&0&0&0&0&1&0 \\ 0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0 \end{pmatrix} \; \tilde{B}_o = \begin{pmatrix} 1&0&0&0 \\ 0&0&0&0 \\ 0&1&0&0 \\ 0&0&0&0 \\ 0&0&0&0 \\ 0&0&0&0 \\ 0&0&0&0 \end{pmatrix} \; \tilde{C}_o^T = \begin{pmatrix} 1&0 \\ 0&1 \\ 0&0 \\ 0&0 \\ 0&0 \\ 0&0 \\ 0&0 \end{pmatrix} \tag{6.85}$$

Defining as $\tilde{A}_d$, $\tilde{B}_d$, and $\tilde{C}_d$, the discrete-time equivalents of matrices $\tilde{A}_o$, $\tilde{B}_o$, and $\tilde{C}_o$, respectively, a Derivative-free nonlinear Kalman Filter can be designed for the aforementioned representation of the system dynamics [436, 437]. The associated Kalman Filter-based disturbance estimator is given by *measurement update*:

$$\begin{aligned} K(k) &= P^{-}(k)\tilde{C}_d^T[\tilde{C}_d \cdot P^{-}(k)\tilde{C}_d^T + R]^{-1} \\ \hat{x}(k) &= \hat{x}^{-}(k) + K(k)[z(k) - \tilde{C}_d\hat{x}^{-}(k)] \\ P(k) &= P^{-}(k) - K(k)\tilde{C}_d P^{-}(k) \end{aligned} \tag{6.86}$$

time update:

$$\begin{aligned} P^{-}(k+1) &= \tilde{A}_d(k)P(k)\tilde{A}_d^T(k) + Q(k) \\ \hat{x}^{-}(k+1) &= \tilde{A}_d(k)\hat{x}(k) + \tilde{B}_d(k)\tilde{v}(k) \end{aligned} \tag{6.87}$$

To compensate for the effects of the disturbance forces it suffices to use in the control loop the modified control input vector

$$v = \begin{pmatrix} v_1 - \hat{\tilde{f}}_a \\ v_2 - \hat{\tilde{f}}_b \end{pmatrix} \text{ or } v = \begin{pmatrix} v_1 - \hat{z}_4 \\ v_2 - \hat{z}_6 \end{pmatrix} \tag{6.88}$$

6.5.5 Simulation Tests

To evaluate for the performance of the proposed nonlinear control scheme, as well as for the performance of the Kalman Filter-based disturbances, estimator simulation experiments have been carried. Different velocity setpoints have been assumed (for velocity along the horizontal and vertical axis of the inertial reference frame, as well as for angular velocity round the vehicle's z axis). Moreover, different disturbance forces and torques have been assumed to affect the vehicles' dynamic model. Using the representation of the vehicle's dynamics given in Eqs. (6.48)–(6.51), two generalized disturbance forces/torques have been considered: the first denoted as $\tilde{f}_a$ was associated with state variable y_1, while the second one denoted as $\tilde{f}_b$ was associated with the state variable y_2. It was also assumed that the change in time of the generalized forces and torques was defined by the second derivative of the associated variable, i.e., $\ddot{\tilde{f}}_a$ and $\ddot{\tilde{f}}_b$. The disturbances dynamics was completely unknown to the controller and their identification was performed in real time by the disturbance estimator. The control loop used in the vehicle's autonomous navigation is given in Fig. 6.42.

The measurable variables used by the control and disturbances' estimation scheme were the vehicle's velocity V_x along the longitudinal axis, the vehicle's velocity V_y along the lateral axis and the vehicle's yaw rate $\dot{\psi}$. The first two variables can be obtained with the use of onboard accelerometers, while the third variable can be obtained with the use of a gyrocompass. The longitudinal axis of the vehicle is

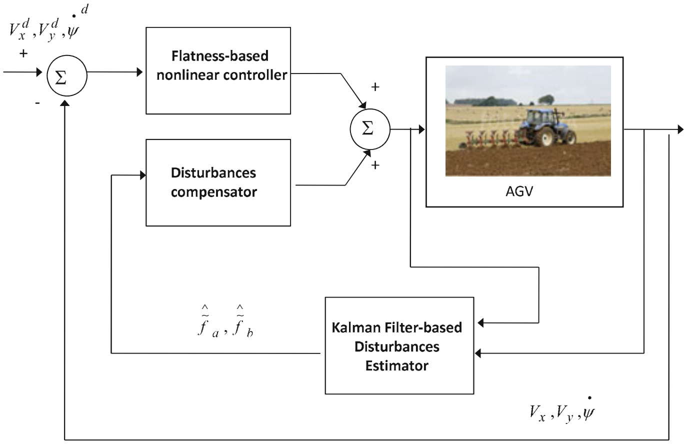

$V_x^d, V_y^d, \dot{\psi}^d$
+
-
Σ
Flatness-based nonlinear controller
Disturbances compensator
+
Σ
+
AGV
$\hat{\tilde{f}}_a, \hat{\tilde{f}}_b$
Kalman Filter-based Disturbances Estimator
$V_x, V_y, \dot{\psi}$

Fig. 6.42 Control loop for the autonomous vehicle comprising a flatness-based nonlinear controller and a Kalman Filter-based disturbances estimator

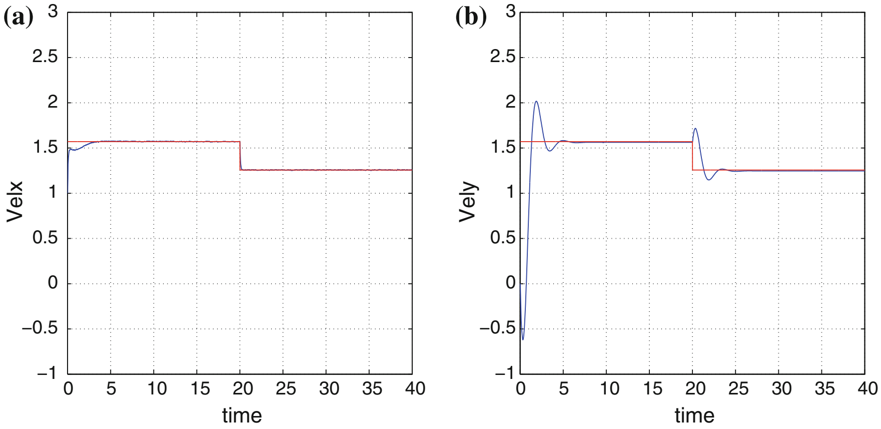


Fig. 6.43 Vehicle control under disturbances profile 1: **a** convergence of x-axis velocity V_x to the desirable setpoint, **b** convergence of the y-axis velocity V_y to the desirable setpoint

denoted as x-axis, while the lateral axis of the vehicle is denoted as y-axis. As it can be seen in Figs. 6.43, 6.44, 6.45, 6.46, 6.47, 6.48, 6.49, and 6.50, the proposed nonlinear controller succeeded accurate tracking of velocity setpoints. Moreover, the efficient estimation of disturbance forces and torques that was succeeded by the Kalman Filter-based disturbance estimator enabled their compensation through the inclusion of an additional control term in the loop.

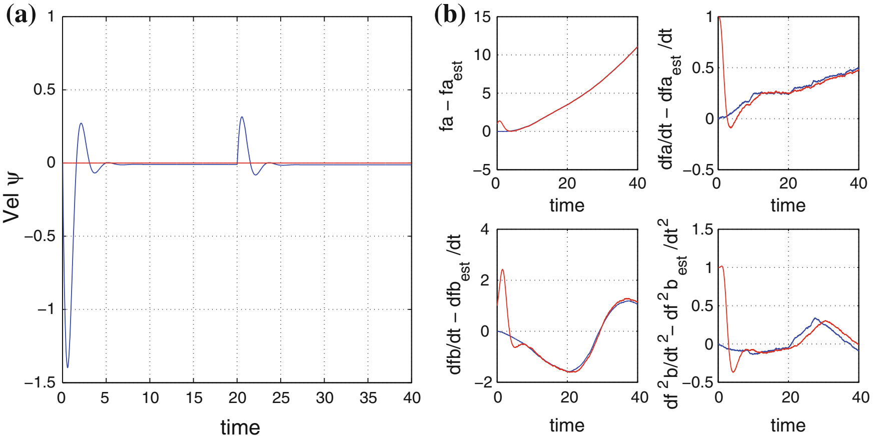


Fig. 6.44 Vehicle control under disturbances profile 1: **a** convergence of yaw rate $\dot{\psi}$ to the desirable setpoint, **b** estimation of the disturbance terms and of their rate of change

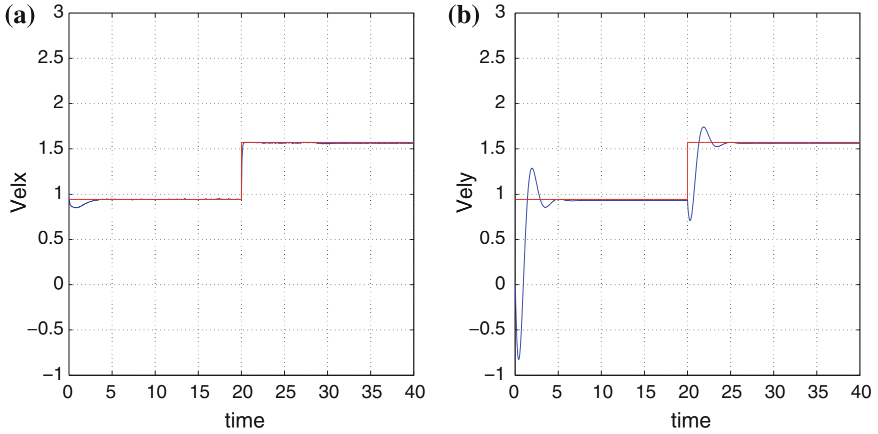


Fig. 6.45 Vehicle control under disturbances profile 2: **a** convergence of x-axis velocity V_x to the desirable setpoint, **b** convergence of the y-axis velocity V_y to the desirable setpoint

6.6 Active Vehicle Suspension Control

6.6.1 Overview

Another application area of differential flatness theory-based control and filtering for intelligent vehicles design is active suspension control. In the recent years there has been systematic effort towards designing vehicles of improved safety and comfort and to this end the development of active suspension control systems has been an

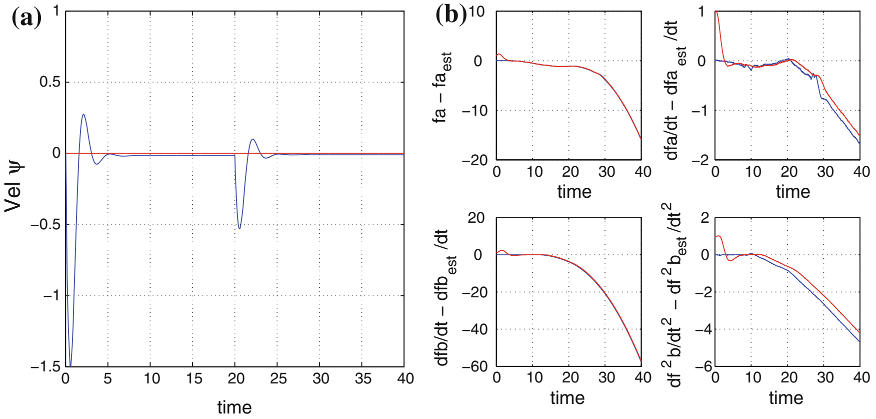


Fig. 6.46 Vehicle control under disturbances profile 2: **a** convergence of yaw rate $\dot{\psi}$ to the desirable setpoint, **b** estimation of the disturbance terms and of their rate of change

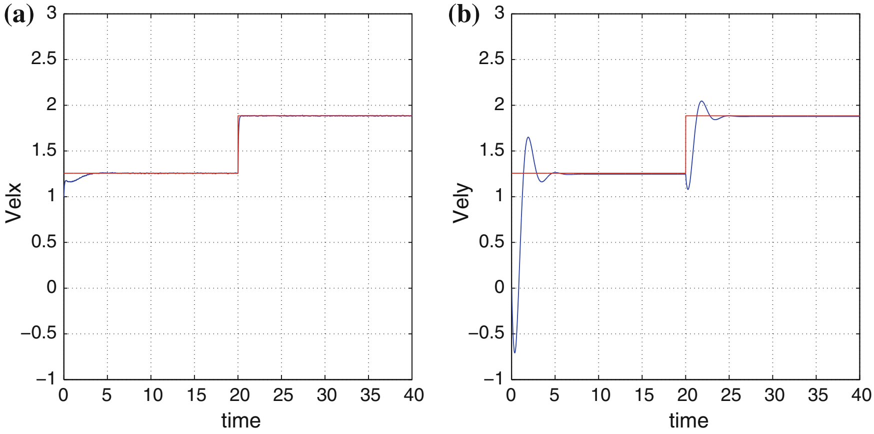


Fig. 6.47 Vehicle control under disturbances profile 3: **a** convergence of x-axis velocity V_x to the desirable setpoint, **b** convergence of the y-axis velocity V_y to the desirable setpoint

important research topic. One can note several results about active suspension control systems exhibiting robustness to external disturbance forces and being capable of efficiently suppressing the vibrations induced to the vehicle by these disturbances. H_∞ controllers have been developed taking into account worst case disturbances on the suspension models [134, 165, 579]. Moreover, there have been results on operating the suspension's control loop under limited information provided by a small number of onboard sensors. This can be seen in the case of developing some type of state estimator or statistical filter to approximate the nonmeasurable elements of the suspension's state vector and the unknown disturbance forces. Particularly, one can note the use of Linear Quadratic Gaussian (LQG) control where Kalman Filtering is combined with an optimal controller [193, 196, 206, 269, 341].

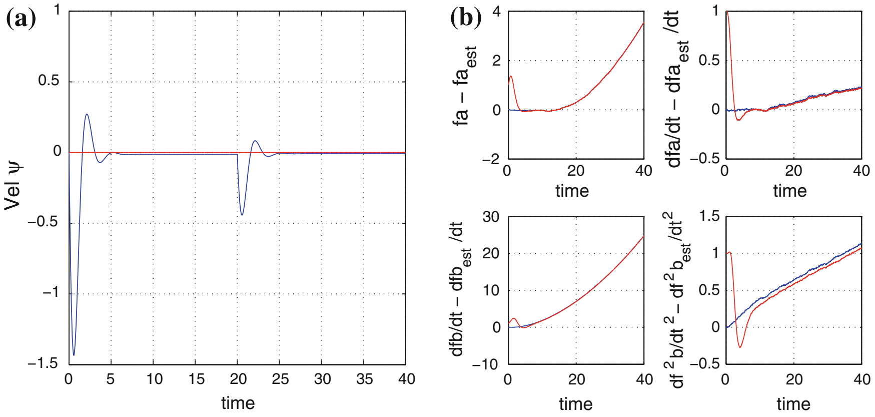


Fig. 6.48 Vehicle control under disturbances profile 3: **a** convergence of yaw rate $\dot{\psi}$ to the desirable setpoint, **b** estimation of the disturbance terms and of their rate of change

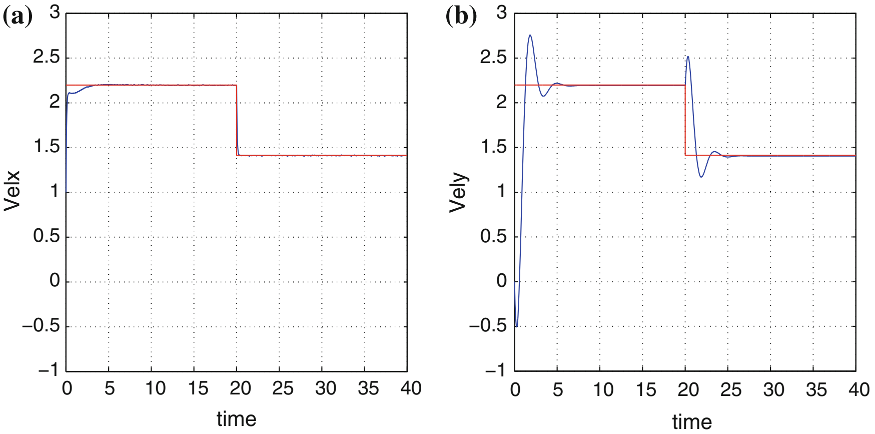


Fig. 6.49 Vehicle control under disturbances profile 4: **a** convergence of x-axis velocity V_x to the desirable setpoint, **b** convergence of the y-axis velocity V_y to the desirable setpoint

Moreover in [207] the application of a sliding-mode controller together with Kalman Filtering has been proposed for implementing state estimation-based control of the suspension's model. Additionally, disturbance observers have been used for simultaneous estimation of the suspension's state vector and of the unknown external disturbances. The suitability of disturbance observers for vibration control problems has been shown in [26], while the efficiency of disturbance estimators in vehicle control loops and especially in the suspension control problem has been demonstrated in [40, 121, 123, 248, 249, 491]. Finally, a scheme of distributed Kalman Filtering has been applied to disturbances and state vector estimation for the suspension's mechanism in [270].

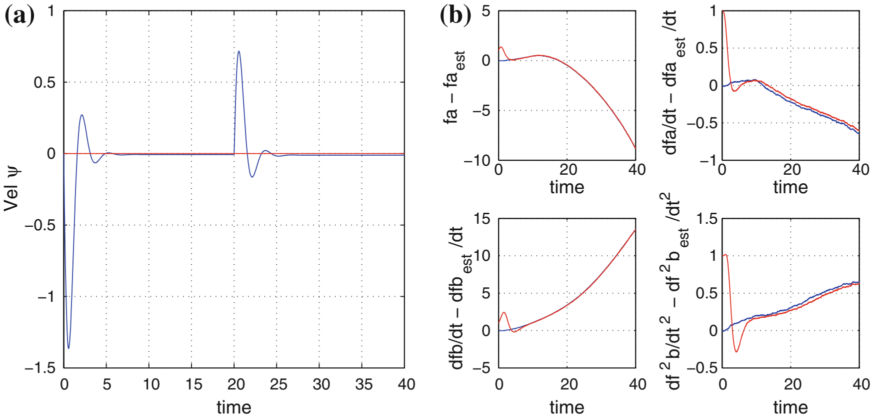


Fig. 6.50 Vehicle control under disturbances profile 4: **a** convergence of yaw rate $\dot{\psi}$ to the desirable setpoint, **b** estimation of the disturbance terms and of their rate of change

In this section an approach to solve the problem of active control of vehicle suspensions is developed with the use of flatness-based controller and a Kalman Filter-based disturbances estimator. First, dynamic analysis for the vehicle's suspension model is provided. Active vehicle suspension control systems are underactuated and the efficient suppression of disturbance inputs (e.g., due to rough road surface) is important for attaining the performance objectives of the control loop. The elements of the state vector are variables denoting the displacement of the sprung and unsprung masses from their zero position and variables denoting the linear velocities of these masses. The control inputs to the model are the force generated by the actuator placed between the two masses (which aims at the suppression of vibrations) and the unknown disturbance force that is generated due to contact of the tire with the road surface. The model can take the form of a linear state-space equation. Moreover, by assuming nonlinearities in the spring and damper terms of the suspension a nonlinear dynamical model is obtained.

Next, it is shown how a controller for the aforementioned suspension model can be obtained through the application of differential flatness theory [152, 465, 516, 535]. The flat output for the suspension's model is a scalar variable which is equal to the weighted sum of the elements of the suspension's state vector. By expressing all state variables and the control input of the suspension model as functions of the flat output and its derivatives the system's dynamic model is transformed into the linear Brunovksy (canonical) form [317, 479]. For the latter model it is possible to design a state feedback controller that enables accurate tracking of the vehicle's velocity setpoints. However, since measurements are available only for certain elements of the transformed state vector, to implement a state feedback control loop the rest of the elements of the suspension's transformed state vector have to be estimated with the use of an observer or filter. To this end the concept of *Derivative-free* *nonlinear Kalman Filtering* is employed. By avoiding linearization approximations, the proposed filtering method succeeds better estimation of the system's state variables [422, 427].

A particular difficulty, in the case of state estimation of the suspension's model is the existence of the unmodeled disturbance forces. It is shown that it is possible to redesign the Kalman Filter in the form of a disturbance observer and using the estimation of the disturbance to develop an auxiliary control input that compensates for the disturbances effects [85, 107, 108, 183, 354]. In this way the suspension's control system can become robust with respect to uncertainties in the model's parameters or uncertainties about external forces. It is also noted that in terms of computation speed the proposed Kalman Filter-based disturbance estimator is faster than disturbance estimators that may be based on other nonlinear filtering approaches (e.g., Extended Kalman Filter, Unscented Kalman Filter or Particle Filter) thus becoming advantageous for the real-time estimation of the unknown suspension dynamics.

The efficiency of the proposed control and Kalman Filter-based disturbances estimation scheme is evaluated through numerical simulation tests. It will be shown that the accurate estimation of the disturbance forces which are exerted on the suspension enables their efficient compensation. This is achieved by introducing an additional element in the controller that produces a counterdisturbance input based on the estimated value for the disturbance variable. This control scheme finally results in minimizing the effects of the disturbances on the vehicle's parts.

6.6.2 Dynamic Model of Vehicle Suspension

6.6.2.1 Dynamics of the 2-DOF Suspension

The suspension system is depicted in Fig. 6.51 and its dynamics is written as

$$\begin{array}{c} m_1\ddot{x}_1 + c_1(\dot{x}_1 - \dot{x}_2) + k_1(x_1 - x_2) = f \\ m_2\ddot{x}_2 + c_1(\dot{x}_2) - \dot{x}_1 + k_1(x_2 - x_1) + \\ +c_2(\dot{x}_2 - \dot{\zeta}) + k_2(x_2 - \zeta) = -f \end{array} \tag{6.89}$$

Variable x_1 denotes the sprung mass displacement while variable x_2 denotes the unsprung mass displacement. Tyre's deflection ζ and its time derivatives $\zeta^{(i)}$. $i = 1, 2, \ldots$ represent unknown external disturbance inputs due to road surface roughness and are assumed to be bounded. The mass that needs regulation is the sprung mass m_1 which is also considered to be larger than m_2. The control force f is generated by an actuator placed between the two masses (see Fig. 6.51).

A normalization is performed to the model through the following procedure: (i) the normalized time is defined as $\tau = t\sqrt{\frac{k_1}{m_1}}$, (ii) the normalized input force is $u = \frac{f}{k_1}$. The system constant coefficients are redefined as $\varepsilon = \frac{m_2}{m_1}$, $\gamma_1 = \frac{c_1}{m_1}\sqrt{\frac{m_1}{k_1}}$,

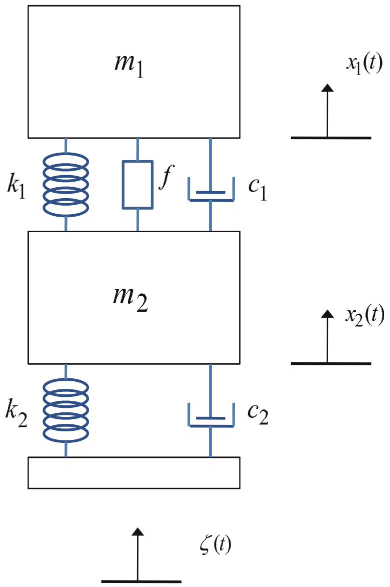


Fig. 6.51 An active vehicle suspension system where both the sprung and the unsprung mass are connected to a spring and a damper

$\gamma_2 = \frac{c_2}{m_1}\sqrt{\frac{m_1}{k_1}}$, $\kappa = \frac{k_2}{k_1}$. Thus, the dynamics model of the vehicle's suspension can be rewritten as

$$\begin{aligned} \ddot{x}_1 + \gamma_1(\dot{x}_1 - \dot{x}_2) + (x_1 - x_2) = u \\ \varepsilon\ddot{x}_2 + \gamma_1(\dot{x}_2 - \dot{x}_1) + (x_2 - x_1) + \\ + \gamma_2(\dot{x}_2 - \dot{\zeta}) + \kappa(x_2 - \zeta) = -u \\ y = x_1 \end{aligned} \tag{6.90}$$

The model of Eq. (6.90) can be also written is state-space form after defining the state variables $z_1 = x_1$, $z_2 = \dot{x}_1$, $z_3 = x_2$, $z_4 = \dot{x}_2$. Thus one has

$$\begin{aligned} \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & -\gamma_1 & 1 & \gamma_1 \\ 0 & 0 & 0 & 1 \\ \frac{1}{\varepsilon} & \frac{\gamma_1}{\varepsilon} & -\frac{1+\kappa}{\varepsilon} & -\frac{\gamma_1+\gamma_2}{\varepsilon} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} + \\ + \begin{pmatrix} 0 \\ 1 \\ 0 \\ -\frac{1}{\varepsilon} \end{pmatrix} u + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{\kappa}{\varepsilon} & \frac{\gamma_2}{\varepsilon} \end{pmatrix} \begin{pmatrix} \zeta \\ \dot{\zeta} \end{pmatrix} \end{aligned} \tag{6.91}$$

where all terms associated with disturbance ζ can be represented by the new variable Δ. Thus one obtains

$$\frac{d}{d\tau}x_s = Ax_s + Bu + \Delta \tag{6.92}$$

6.6.2.2 A Nonlinear Model of Vehicle Suspension Dynamics

The dynamical model of the two-degrees of freedom vehicle suspension (see Fig. 6.52) is given as follows [491]

$$\begin{aligned} m_s\ddot{z}_s + F_{s_d} + F_{s_k} &= F_A \\ m_u\ddot{z}_u - F_{s_k} - F_{s_d} + K_t(z_u - z_r) &= -F_A \end{aligned} \tag{6.93}$$

where F_A is the force generated by the actuator, F_{s_k} is the force associated with the suspension's spring term, F_{s_c} is the force associated with the suspension's damper term and $F_t = k_t(z_u - z_r)$ is a spring force associated with elasticity coefficient k_t and denoting the spring-type behavior of the wheel when in contact with the road's surface.

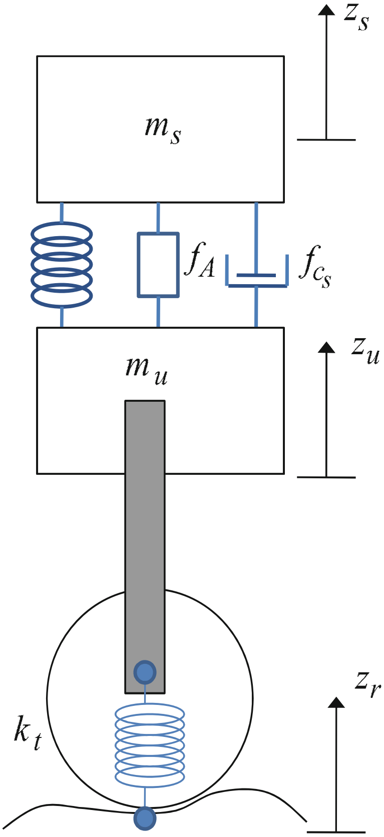


Fig. 6.52 An active suspension system

It holds that

$$F_{s_k}(z_s, z_u) = k_s(z_s - z_u) + k_{n_s}(z_s - z_u)^3 \tag{6.94}$$

$$\begin{aligned} F_{s_c}(z_s, z_u) = b_s(\dot{z}_s - \dot{z}_u) + \\ +b_{ns}(\dot{z}_s - \dot{z}_u)^2 sgn(\dot{z}_s - \dot{z}_u) \end{aligned} \tag{6.95}$$

Denoting the state variables $x_1 = z_s$, $x_2 = \dot{z}_s$, $x_3 = z_u$, $x_4 = \dot{z}_u$ one has

$$\begin{aligned} &\dot{x}_1 = x_2 \\ &\dot{x}_2 = -\tfrac{1}{m_s}[k_s(x_1 - x_3) + k_{n_s}(x_1 - x_3)^3 + \\ &+b_s(x_2 - x_4) + b_{ns}(x_2 - x_4)^2 sgn(x_2 - x_4)] + \tfrac{1}{m_s}u \\ &\dot{x}_3 = x_4 \\ &\dot{x}_4 = -\tfrac{k_t}{m_u}x_3 + \tfrac{1}{m_u}[k_s(x_1 - x_3) + \\ &+k_{n_s}(x_1 - x_3)^3 + b_s(x_2 - x_4) + \\ &+b_{ns}(x_2 - x_4)^2 sgn(x_2 - x_4)] - \tfrac{1}{m_u}u + \tfrac{k_t}{m_u}z_r \end{aligned} \tag{6.96}$$

where the term $\frac{1}{m_u}z_r$ can be considered as a disturbance term. Denoting the nonlinear functions

$$\begin{aligned} f_1(x,t) = -\tfrac{1}{m_s}[k_s(x_1 - x_3) + k_{n_s}(x_1 - x_3)^3 + \\ +b_s(x_2 - x_4) + b_{ns}(x_2 - x_4)^2 sgn(x_2 - x_4)] \end{aligned} \tag{6.97}$$

$$g_1(x,t) = \tfrac{1}{m_s} \tag{6.98}$$

$$\begin{aligned} f_2(x,t) = -\tfrac{k_t}{m_u}x_3 + \tfrac{1}{m_u}[k_s(x_1 - x_3) + k_{n_s}(x_1 - x_3)^3 + \\ +b_s(x_2 - x_4) + b_{ns}(x_2 - x_4)^2 sgn(x_2 - x_4)] \end{aligned} \tag{6.99}$$

$$g_2(x,t) = -\tfrac{1}{m_u} \tag{6.100}$$

one has the following state-space description

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f_1(x,t) + g_1(x,t)u \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= f_2(x,t) + g_2(x,t)u \end{aligned} \tag{6.101}$$

6.6.3 *Flatness-Based Control for a Suspension Model*

6.6.3.1 A Flatness-Based Controller for the Vehicle Suspension Model

It can be proven that the suspension model is differentially flat by defining the following flat output [491]

$$F = \begin{pmatrix}0 & 0 & 0 & 1\end{pmatrix} C_o^{-1} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} \\ \text{i.e., } F = \frac{\varepsilon}{\kappa}x_1 - \frac{\varepsilon\gamma_2}{\kappa^2}\dot{x}_1 + \frac{\varepsilon\kappa - \gamma_2^2\varepsilon}{\kappa^2}x_2 - \frac{\varepsilon^2\gamma_2}{\kappa^2}\dot{x}_2 \tag{6.102}$$

where C_o stands for the system's controllability matrix.

$$C_o = [B, AB, A^2B, A^3B] \tag{6.103}$$

It can be shown that all state variable of the system and the control input can be written as functions of the flat output and its derivatives. Indeed for the unperturbed system, i.e., if from the model of Eq. (6.102) the disturbance input ζ and its derivatives $\dot{\zeta}$ are ignored it holds

$$\begin{aligned} F &= \frac{\varepsilon}{\kappa}x_1 - \frac{\varepsilon\gamma_2}{\kappa^2}\dot{x}_1 + \frac{(\varepsilon\kappa - \gamma_2^2)\varepsilon}{\kappa^2}x_2 - \frac{\varepsilon^2\gamma_2}{\kappa^2}\dot{x}_2 \\ \dot{F} &= \frac{\varepsilon}{\kappa}\dot{x}_1 + \frac{\varepsilon\gamma_2}{\kappa}x_2 + \frac{\varepsilon^2}{\kappa}\dot{x}_2 \\ \ddot{F} &= -\varepsilon x_2 \\ F^{(3)} &= \varepsilon\dot{x}_2 \end{aligned} \tag{6.104}$$

while one also has

$$\begin{aligned} x_1 &= \frac{\kappa}{\varepsilon}F + \frac{\gamma_2}{\varepsilon}\dot{F} + \ddot{F} \\ x_2 &= \frac{\kappa}{\varepsilon}\dot{F} + \frac{\gamma_2}{\varepsilon}\ddot{F} + F^{(3)} \\ x_3 &= \frac{-1}{\varepsilon}\ddot{F} \\ x_4 &= \frac{-1}{\varepsilon}F^{(3)} \end{aligned} \tag{6.105}$$

while with the use of $u = \ddot{x}_1 + \gamma_1(\dot{x}_1 - \dot{x}_2) + (x_1 - x_2)$ it can be also concluded that the control input is a function of the flat output and its derivatives.

Taking also into account the effects of the disturbance input ζ and of its derivative $\dot{\zeta}$ the flat output and its derivative are formulated as follows

$$\begin{aligned} F &= \frac{\varepsilon}{\kappa}x_1 - \frac{\varepsilon\gamma_2}{\kappa^2}\dot{x}_1 + \frac{\varepsilon\kappa - \gamma_2^2\varepsilon}{\kappa^2}x_2 - \frac{\varepsilon^2\gamma_2}{\kappa^2}\dot{x}_2 \\ \dot{F} &= \frac{\varepsilon}{\kappa}x_1 - \frac{\varepsilon\gamma_2}{\kappa^2}x_2 + \frac{\varepsilon^2}{\kappa}\dot{x}_2 - \frac{\varepsilon\gamma_2}{\kappa}\zeta(\tau) - \frac{\varepsilon\gamma_2^2}{\kappa^2}\dot{\zeta}(\tau) \\ \ddot{F} &= -\varepsilon x_2 + \varepsilon\zeta(\tau) + \varepsilon\gamma_2(1 - \frac{1}{\kappa})\dot{\zeta}(\tau) - \frac{\varepsilon\gamma_2^2}{\kappa^2}\ddot{\zeta}(\tau) \\ F^{(3)} &= -\varepsilon\dot{x}_2 + \varepsilon\dot{\zeta}(\tau) + \varepsilon\gamma_2(1 - \frac{1}{\kappa})\ddot{\zeta}(\tau) - \frac{\varepsilon\gamma_2^2}{\kappa^2}\zeta^{(3)}(\tau) \end{aligned} \tag{6.106}$$

Differentiating one more time the flat output one obtains

$$\begin{aligned}&F^{(4)}=-x_1-\gamma_1\dot{x}_1+(1+\kappa)x_2+\\&+(\gamma_1+\gamma_2)\dot{x}_2+u-\kappa\zeta(\tau)-\gamma_2\dot{\zeta}(\tau)+\\&+\varepsilon\ddot{\zeta}(\tau)+\varepsilon\gamma_2(1-\tfrac{1}{\kappa})\zeta^{(3)(\tau)}-\\&-\varepsilon\tfrac{\gamma_2^2}{\kappa^2}\zeta^{(4)}(\tau)\end{aligned}\tag{6.107}$$

Aggregating all terms other than u into one variable

$$\begin{aligned}&\phi(\tau)=x_1-\gamma_1\dot{x}_1+(1+\kappa)x_2+(\gamma_1+\gamma_2)\dot{x}_2-\\&-\kappa\zeta(\tau)-\gamma_2\dot{\zeta}(\tau)+\varepsilon\ddot{\zeta}(\tau)+\\&+\varepsilon\gamma_2(1-\tfrac{1}{\kappa})\zeta^{(3)}(\tau)-\tfrac{\varepsilon\gamma_2^2}{\kappa^2}\zeta^{(4)}(\tau)\end{aligned}\tag{6.108}$$

one has the system dynamics

$$F^{(4)}=u+\phi(\tau)\tag{6.109}$$

or equivalently, in state-space form

$$\begin{pmatrix}\dot{F}_1\\\dot{F}_2\\\dot{F}_3\\\dot{F}_4\end{pmatrix}=\begin{pmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\\0&0&0&0\end{pmatrix}\begin{pmatrix}F_1\\F_2\\F_3\\F_4\end{pmatrix}+\begin{pmatrix}0\\0\\0\\1\end{pmatrix}\big(u+\phi(\tau)\big)\tag{6.110}$$

with the state variable $F_i(\tau),\ i=1,\dots,4$ to stand for the $(i-1)$th order derivative $F^{(i-1)}(t)$ of the flat output. The estimation of the term $\phi(\tau)$ by a disturbance observer enables to design a controller for the vehicle's suspension model as follows

$$\begin{aligned}&u(\tau)=F_d^{(4)}(\tau)-k_1(F^{(3)}(\tau)-F_d^{(3)}(\tau))-\\&-k_2(\ddot{F}(\tau)-\ddot{F}_d(\tau)-k_3(\dot{F}(\tau)-\\&-\dot{F}_d(\tau))-k_4(F(\tau)-F_d(\tau))-\hat{\phi}(\tau)\end{aligned}\tag{6.111}$$

6.6.4 Compensating for Model Uncertainty with the Use of the H_∞ Kalman Filter

The Kalman Filter for the canonical form of the suspension model given in Eq. (6.110) can be redesigned to cope with the case of maximum errors of some linear combination of states for worst case assumptions of process noise, measurement noise and disturbances. This can be useful in state estimation for the vehicle suspension model, as a method for model uncertainty compensation. Filters designed to minimize a weighted norm of state errors are called H_∞ or minimax filters [177, 490].

The discrete-time H_∞ filter uses the same state-space model as the Kalman Filter, which has the form

$$\begin{aligned} x(k+1) &= A(k)x(k) + B(k)u(k) + w(k) \\ z(k) &= C(k)x(k) + v(k) \end{aligned} \tag{6.112}$$

$E[w(k)] = 0$, $E[w(k)w(k)^T] = Q(k)\delta ij$, $E[v(k)] = 0$, $E[v(k)v(k)^T] = R(k)\delta_{ij}$ and $E(w(k)v(k)^T) = 0$. The update of the state estimate is again given by

$$\hat{x}(k) = \hat{x}^-(k) + K(k)(z(k) - C(k)\hat{x}^-(k)) \tag{6.113}$$

that minimizes the trace of the covariance matrix of the state vector estimation error

$$J = \frac{1}{2}E\{\tilde{x}(k)^T \cdot \tilde{x}(k)\} = \frac{1}{2}tr(P^-(k)) \tag{6.114}$$

where $\tilde{x}^-(k) = x(k) - \hat{x}^-(k)$ and $P^-(k) = E[\tilde{x}^-(k)^T \cdot \tilde{x}^-(k)]$. The H_∞ filtering approach defines first a transformation

$$d(k) = L(k)x(k) \tag{6.115}$$

where $L(k) \in R^{n \times n}$ is a full rank matrix. The use of the transformation given in Eq. (6.115) allows certain combinations of states to be given more weight than others. Next, defining the estimation error variable $\tilde{d}_1(i) = d(i) - \hat{d}(i)$, the cost function of the H_∞ filter is initially formulated as

$$\begin{aligned} J(k) &= \textstyle\sum_{i=0}^{k-1} \tilde{d}(i+1)^T S(i)\tilde{d}(i+1)/b \\ b &= \tilde{x}^-(0)^T P^-(0)^{-1}\tilde{x}^-(0) + \textstyle\sum_{i=0}^{k-1} w^T(i+1)Q(i+1)^{-1}w(i+1) + \\ &\quad + \textstyle\sum_{i=0}^{k-1} v^T(i)R(i)^{-1}v(i) \end{aligned} \tag{6.116}$$

where S_i is a positive-definite symmetric weighting matrix. It can be observed that both matrices $S(k)$ and $L(k)$ appear in the cost function and thus affect the solution $\hat{x}^-(k+1)$ of the optimization problem. The objective is to find state vector estimates $\hat{x}^-(k)$ and $\hat{x}(k)$ that keep the cost function below a given value $1\ \theta$ for worst case conditions, i.e.,

$$J(k) < \frac{1}{\theta} \tag{6.117}$$

By rewriting Eq. (6.116) and substituting Eq. (6.112) a modified cost functional is obtained

$$\begin{gathered} J_a(k) = -\tfrac{1}{\theta}\tilde{x}^-(0)^T P^-(0)\tilde{x}^-(0) + \textstyle\sum_{i=0}^{k-1} \Gamma(i) \\ \Gamma(i) = (x(i+1) - \hat{x}^-(i+1))^T W_i (x(i+1) - \hat{x}^-(i+1)) - \\ -\tfrac{1}{\theta}(w^T(i+1)Q(i+1)^{-1}w(i+1) + (y(i) - C(i)x^-(i))^T R(i)^{-1}(y(i) - C(i)x^-(i))) \end{gathered} \tag{6.118}$$

and

$$W(i) = L(i)^T S(i) L(i) \tag{6.119}$$

This cost function does not include the dynamic model of the system given in Eq. (6.112) and this is added by using a vector of Lagrange multipliers $\lambda(i+1)$. This gives

$$\begin{aligned} &J(k) = -\tfrac{1}{\theta}\tilde{x}^-(0)^T P^-(0)\tilde{x}^-(0)+ \\ &+\textstyle\sum_{i=0}^{k-1}(\Gamma_i + 2\frac{\lambda(i+1)^T}{\theta})(A(i)\hat{x}(i) + B(i)u(i)+ \\ &+w(i) - x(i+1)) + \tfrac{2\lambda(0)^T}{\theta}x(0) - \tfrac{2\lambda(0)^T}{\theta}x(0) \end{aligned} \tag{6.120}$$

The cost function of the filter given in Eq. (6.120) can be used as the basis for the solution. It is aimed to find equations defining $\hat{x}^-(k+1)$, or equivalently a measurement weighting matrix (similar to the Kalman gain matrix), that minimize the cost for worst case assumptions about $x(0)$, $w(i)$ and $y(i)$. Thus, the optimization objective is formulated as

$$J^*(k) = \min_{x_i} \max_{x(0),\, w(i),\, y(i)} J(k) \tag{6.121}$$

It is noted that the estimation algorithm uses knowledge of the output measurement $y(i)$ but receives no information about the initial conditions of the system $x(0)$ and the process noise $w(i)$. Under this assumption, the estimation should be able to compensate for worst case values for the unknown parameters. This is a game theoretic problem that is solved in two steps.

In the first step of optimization, partial derivatives of $J(k)$ with respect to $x(0)$, $w(i)$ and $\lambda(i)$ are set to zero so as to maximize the cost function of Eq. (6.120), now being dependant only on the terms $\hat{x}^-(k+1)$ and $y(k)$ which are included in Γ_i. In the second step of optimization, the partial derivatives of $J(k)$ with respect to $\hat{x}^-(k+1)$ and $y(k)$ are set to zero, to obtain a condition for the filter's gain matrix that minimizes this cost functional. From the optimization conditions $\partial J(k)/\partial x_0 = 0^T$, $\partial J(k)/\partial w(i) = 0^T$, $\partial J(k)/\partial \lambda(i) = 0^T$ ones obtains an expression of $J(k)$ as function of $\hat{x}^-(k+1)$ and $y(k)$. Next, from the optimization conditions $\partial J(k)/\partial \hat{x}^-(i+1) = 0^T$, and $\partial J(k)/\partial y(i) = 0^T$ one obtains the filter's equations.

The recursion of the H_∞ Kalman Filter can be formulated again in terms of a *measurement update* and a *time update* part

measurement update:

$$\begin{aligned} &D(k) = [I - \theta W(k)P^-(k) + C^T(k)R(k)^{-1}C(k)P^-(k)]^{-1} \\ &K(k) = P^-(k)D(k)C^T(k)R(k)^{-1} \\ &\hat{x}(k) = \hat{x}^-(k) + K(k)[y(k) - C\hat{x}^-(k)] \end{aligned} \tag{6.122}$$

time update:

$$\begin{aligned}\hat{x}^{-}(k+1) &= A(k)x(k)+B(k)u(k)\\ P^{-}(k+1) &= A(k)P^{-}(k)D(k)A^{T}(k)+Q(k)\end{aligned} \tag{6.123}$$

where it is assumed that parameter θ is sufficiently small to maintain

$$P^{-}(k)-\theta W(k)+C^{T}(k)R(k)^{-1}C(k) \tag{6.124}$$

positive definite. When $\theta = 0$ the H_∞ Kalman Filter becomes equivalent to the standard Kalman Filter. It is noted that apart from the process noise covariance matrix $Q(k)$ and the measurement noise covariance matrix $R(k)$ the H_∞ Kalman filter requires tuning of the weight matrices L and S, as well as of parameter θ.

6.6.5 *Robust State Estimation with the Use of Disturbance Observers*

6.6.5.1 State Estimation with the Derivative-Free Nonlinear Kalman Filter

Previous results about state estimation through transformation to linear canonical forms can be found in [334, 335, 436, 437]. It was shown that the dynamical model of the suspension can be written in the MIMO canonical form of Eq. (6.110). Thus one has a MIMO linear model of the form

$$\begin{aligned}\dot{y}_f &= A_f y_f + B_f v\\ z_f &= C_f y_f\end{aligned} \tag{6.125}$$

where $y_f = [F_1, F_2, F_3, F_4]^T$ and matrices A_f,B_f,C_f are in the form

$$A_f = \begin{pmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\\0&0&0&0\end{pmatrix}\; B_f = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}\; C_f^T = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \tag{6.126}$$

where the measurable variables $y_1 = F$ is associated with the displacement of the sprung and unsprung mass in the suspension model. For the aforementioned model, and after carrying out discretization of matrices A_f, B_f and C_f with common discretization methods one can apply linear Kalman filtering using Eqs. (4.5) and (4.6). This is Derivative-free nonlinear Kalman filtering for the model of the suspension which is performed without the need to compute Jacobian matrices and does not introduce numerical errors due to approximative linearization with Taylor series expansion.

6.6.5.2 Kalman Filter-Based Estimation of Suspension Disturbance Forces

Considering the effects of disturbances on the suspension's model and after applying a transform on the system's state variables according to the differential flatness theory it has been shown that the suspension model is described by

$$F^{(4)} = u + \phi(\tau) \tag{6.127}$$

The suspension's state-space model of Eq. (6.110) will be extended to take into account also the dynamics and the effects of the disturbance input $\phi(t)$. The extended state vector of the suspension model is defined as $z \in R^{8\times1}$ with $z_1 = F, z_2 = \dot{F}, z_3 = \ddot{F}, z_4 = F^{(3)}, z_5 = \phi, z_6 = \dot{\phi}, z_7 = \ddot{\phi}, z_8 = \phi^{(3)}$. The dynamics of the disturbance is assumed to be defined by its fourth-order derivative, i.e., $\phi^{(4)} = f_d(F, \dot{F}, \ddot{F}, F^{(3)})$. Thus one has the extended state-space model

$$\begin{matrix} \dot{z} = \tilde{A}\cdot z + \tilde{B}\cdot\tilde{v} \\ q = \tilde{C}z \end{matrix} \tag{6.128}$$

with

$$\tilde{A} = \begin{pmatrix} 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0 \end{pmatrix}, \tilde{B} = \begin{pmatrix} 0&0 \\ 0&0 \\ 0&0 \\ 1&0 \\ 0&0 \\ 0&0 \\ 0&0 \\ 0&1 \end{pmatrix}, \tilde{C}^T = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{6.129}$$

where the measurable variable is z_1 and the control input is

$$\tilde{v} = \left(u, \ \phi^{(4)}\right)^T \tag{6.130}$$

The disturbance estimator has the following structure

$$\begin{matrix} \dot{\hat{z}} = \tilde{A}_o\cdot\hat{z} + \tilde{B}_o\cdot\tilde{v} + K(z_1 - \hat{z}_1) \\ \hat{q} = \tilde{C}_o\hat{y} \end{matrix} \tag{6.131}$$

where the estimator's gain $K \in R^{8\times1}$ and matrices $\tilde{A}_o$, $\tilde{B}_o$ and $\tilde{C}_o$ are defined as

$$\tilde{A}_o = \begin{pmatrix} 0&1&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&1&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&0 \end{pmatrix}, \tilde{B}_o = \begin{pmatrix} 0&0\\ 0&0\\ 0&0\\ 1&0\\ 0&0\\ 0&0\\ 0&0\\ 0&0 \end{pmatrix}, \tilde{C}_o^T = \begin{pmatrix} 1\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0 \end{pmatrix} \tag{6.132}$$

The disturbance estimator's gain $K \in R^{8\times 1}$ will be computed through the Kalman Filter recursion.

Defining as $\tilde{A}_d$, $\tilde{B}_d$, and $\tilde{C}_d$, the discrete-time equivalents of matrices $\tilde{A}_o$, $\tilde{B}_o$ and $\tilde{C}_o$ respectively, a Derivative-free nonlinear Kalman Filter can be designed for the aforementioned representation of the system dynamics [436, 437]. The associated Kalman Filter-based disturbance estimator is given by
measurement update:

$$\begin{aligned} K(k) &= P^-(k)\tilde{C}_d^T[\tilde{C}_d \cdot P^-(k)\tilde{C}_d^T + R]^{-1} \\ \hat{x}(k) &= \hat{x}^-(k) + K(k)[z(k) - \tilde{C}_d\hat{x}^-(k)] \\ P(k) &= P^-(k) - K(k)\tilde{C}_d P^-(k) \end{aligned} \tag{6.133}$$

time update:

$$\begin{aligned} P^-(k+1) &= \tilde{A}_d(k)P(k)\tilde{A}_d^T(k) + Q(k) \\ \hat{x}^-(k+1) &= \tilde{A}_d(k)\hat{x}(k) + \tilde{B}_d(k)\tilde{v}(k) \end{aligned} \tag{6.134}$$

To compensate for the effects of the disturbance forces it suffices to use in the control loop the modified control input vector $v_1 = u - \phi(t)$.

6.6.6 Simulation Tests

To evaluate for the performance of the proposed Kalman Filter-based and disturbances estimation scheme for the vehicle's suspension model simulation tests were carried out. Different disturbance forces were assumed to be exerted on the vehicle's wheel due to its contact with the rough road surface. The disturbances dynamics was completely unknown to the controller and their identification was performed in real time by the disturbance estimator. The parameters of the suspension mechanism were as follows: $m_1 = 2000\,\mathrm{Kg}$, $m_2 = 40\,\mathrm{Kg}$, $K_1 = 1.0 \cdot 10e4\,\mathrm{N/m}$, $K_2 = 1.0 \cdot 10e4\,\mathrm{N/m}$, $c_1 = 1790\,\mathrm{N \cdot s/m}$ and $c_2 = 20\,\mathrm{N \cdot s/m}$. The control loop used for the vehicle's suspension is given in Fig. 6.53.

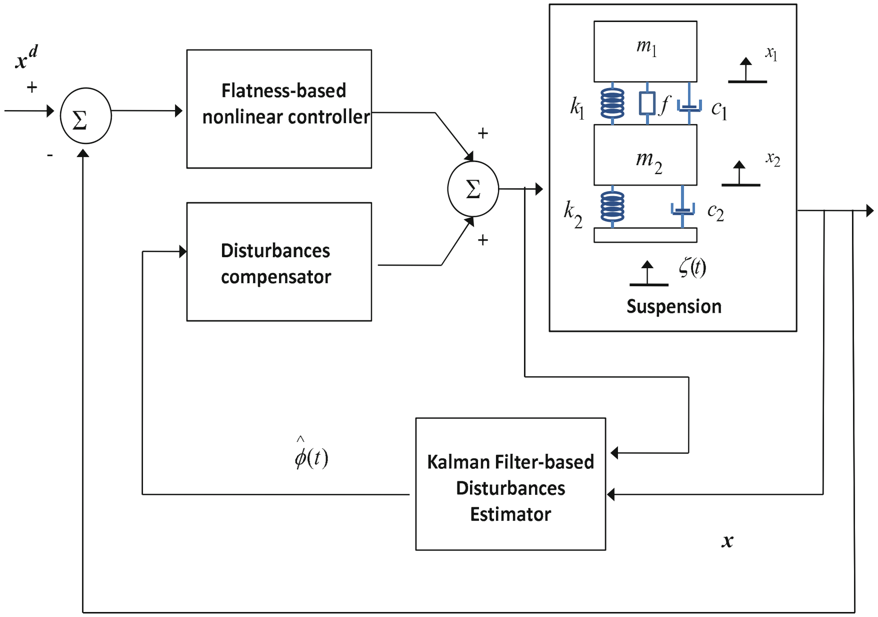


Fig. 6.53 Control loop for the vehicle's suspension comprising a flatness-based nonlinear controller and a Kalman Filter-based disturbances estimator

The monitored parameters were the state variables of the suspension. The control input was the force generated by the actuator. The measured parameters were the position and velocity of the sprung and unsprung mass. The dynamics of the disturbance force was assumed to be defined by its fourth-order derivative. The extended state vector used by the disturbance observer was of dimension $x \in R^8$, where the first four state variables were describing the suspension's model whereas the rest four state variables were associated with the dynamics of the disturbance force. The real-time estimation of the external disturbance that was provided by the Kalman Filter was used by an additional control term in the control loop to generate a counter disturbance control input. Thus, the disturbance's effects on the vehicle's parts were eliminated and vibrations were efficiently suppressed. As shown in Figs. 6.54, 6.55, 6.56, 6.57, 6.58, 6.59, 6.60, 6.61, 6.62, 6.63, 6.64 and 6.65 fast stabilization of the suspension's sprung and unsprung masses to the desirable setpoints was succeeded and accurate estimation of the unknown disturbances forces was performed. Moreover, is should be taken into account that for common nominal values of k_1 and m_1 one obtains $t < \tau$, i.e., $t = \sqrt{\frac{m_1}{k_1}}\tau$ which finally gives the real time scale of suspension's active control system.

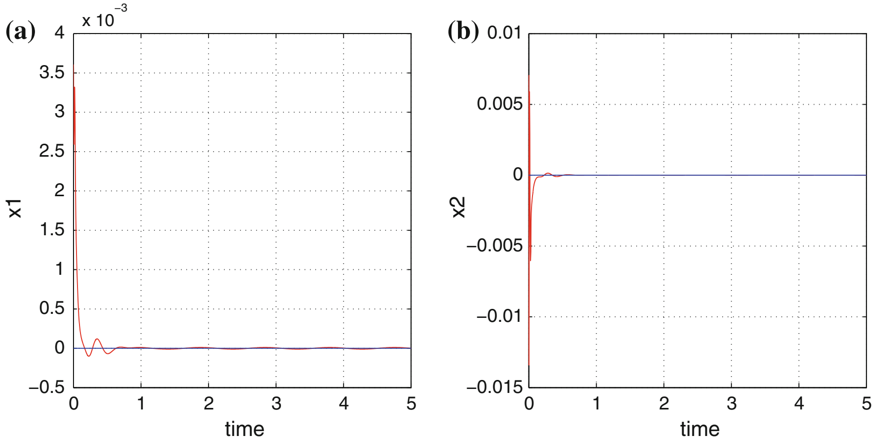


Fig. 6.54 Suspension control under disturbances profile 1: **a** convergence of sprung mass position x_1 to the desirable setpoint, **b** convergence of sprung mass velocity x_2 to the desirable setpoint

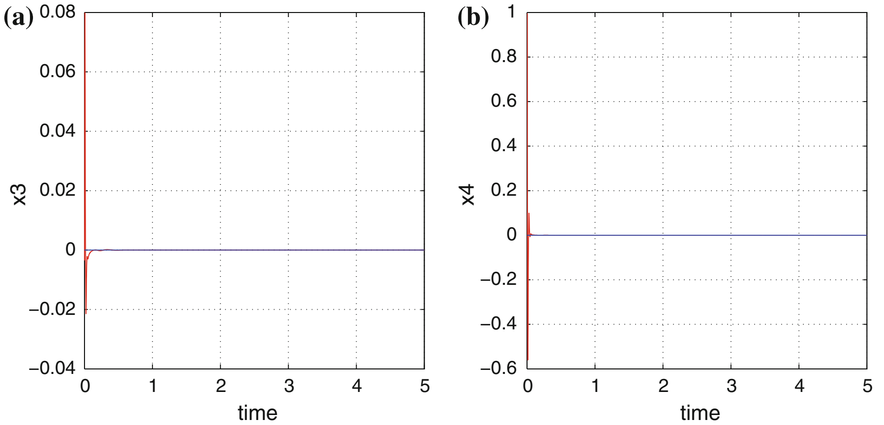


Fig. 6.55 Suspension control under disturbances profile 1: **a** convergence of unsprung mass position x_2 to the desirable setpoint, **b** convergence of unsprung mass velocity x_4 to the desirable setpoint

6.7 State Estimation-Based Control of Quadrotors

6.7.1 Overview

Another type of autonomous vehicles in which one can apply the differential flatness theory-based methods of filtering and control is Unmanned Aerial Vehicles (UAVs). Quadrotors are four-rotor helicopters characterized by a nonlinear 6-DOF unstable dynamical model. To succeed autonomous navigation of the quadrotors,

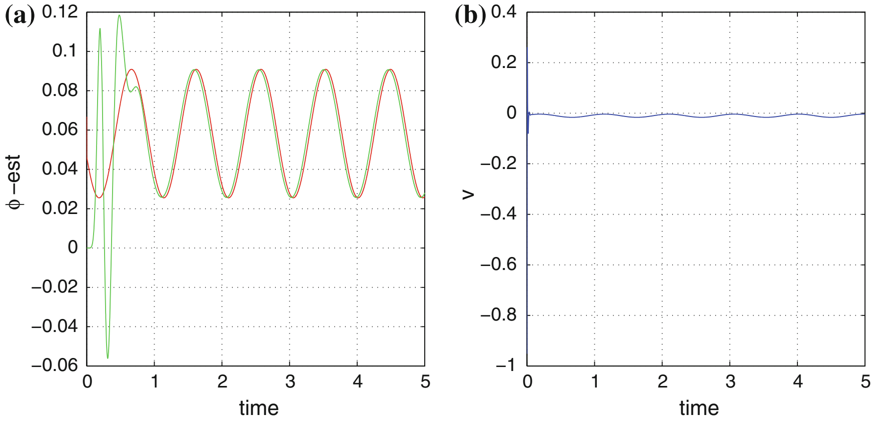


Fig. 6.56 Suspension control under disturbances profile 1: **a** estimation of the disturbance terms, **b** control input generated by the actuator

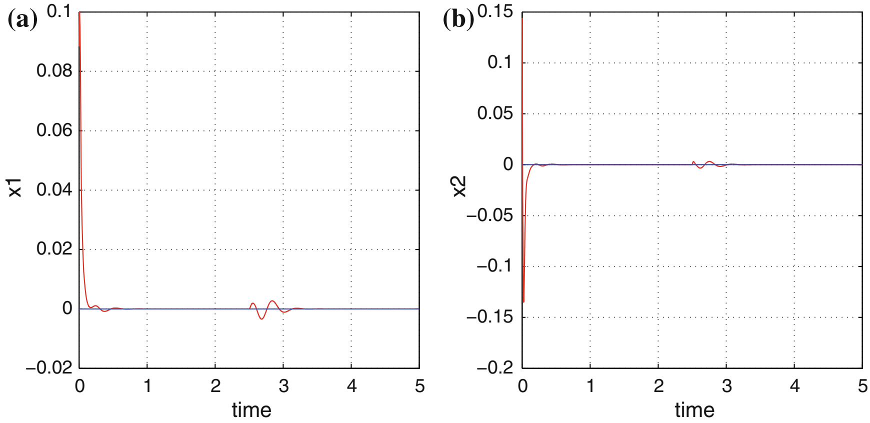


Fig. 6.57 Suspension control under disturbances profile 2: **a** convergence of sprung mass position x_1 to the desirable setpoint, **b** convergence of sprung mass velocity x_2 to the desirable setpoint

it is necessary to design efficient control algorithms that will exhibit robustness to parametric uncertainties and to external disturbances. One can cite several results on quadrotors' control. An approach for quadrotors' control that is based on the transformation of their dynamical model in the linear canonical form and which is associated with differential flatness theory has been given in [541]. Moreover, in [3] a flatness-based control approach is applied to quadrotors' motion control. A predictive controller complemented by an H_∞ term for additional robustness has been analyzed and tested in the quadrotor's flight control problem in [397, 398]. In [52] motion control of the quadrotor was implemented using controllers of the LQR-type and of the PID-type, while Kalman Filtering has been used to provide position

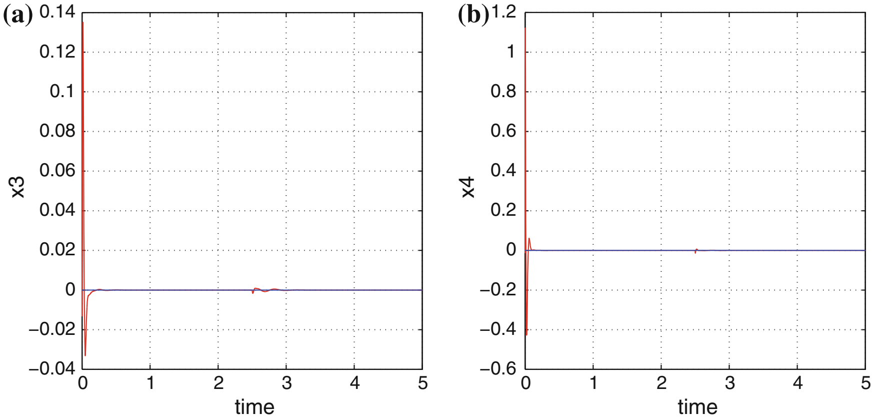


Fig. 6.58 Suspension control under disturbances profile 2: **a** convergence of unsprung mass position x_2 to the desirable setpoint, **b** convergence of unsprung mass velocity x_4 to the desirable setpoint

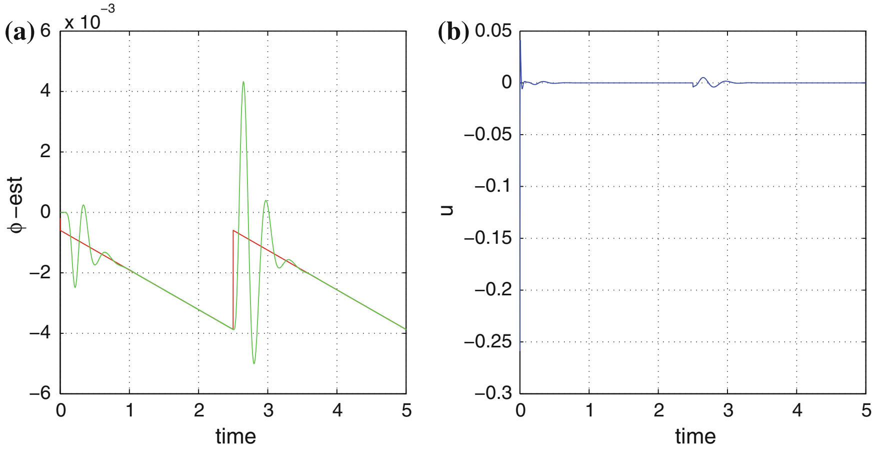


Fig. 6.59 Suspension control under disturbances profile 2: **a** estimation of the disturbance terms, **b** control input generated by the actuator

estimates out of a visual measurements system. In [77] two control strategies are employed as baseline controllers for the quadrotor's model: a LQR controller which is based on a linearized model of the quadrotor and a Sliding-Mode Controller which is based on a nonlinear model of the quadrotor. Moreover, differential flatness theory has been used for trajectory planning. In [271] and in [136], adaptive control schemes have been proposed for the quadrotor's model. The stability of the control loop is confirmed through the Lyapunov approach. In [43] quadrotor's control with the use of a sliding-mode controller and a sliding-mode disturbance observer has been proposed.

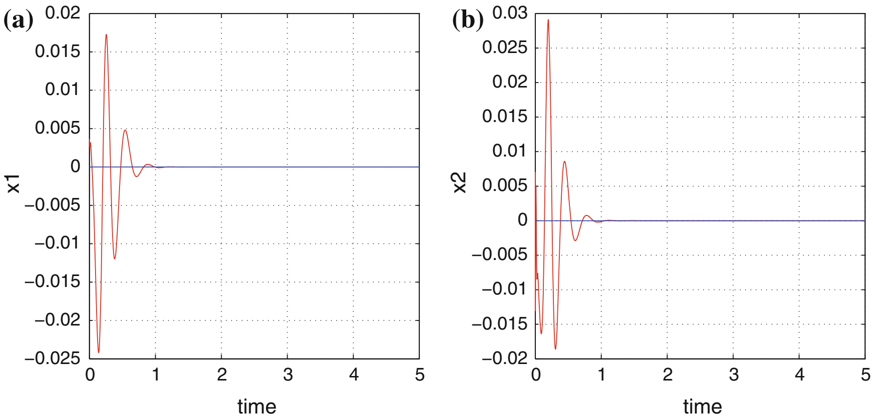


Fig. 6.60 Suspension control under disturbances profile 3: **a** convergence of sprung mass position x_1 to the desirable setpoint, **b** convergence of sprung mass velocity x_2 to the desirable setpoint

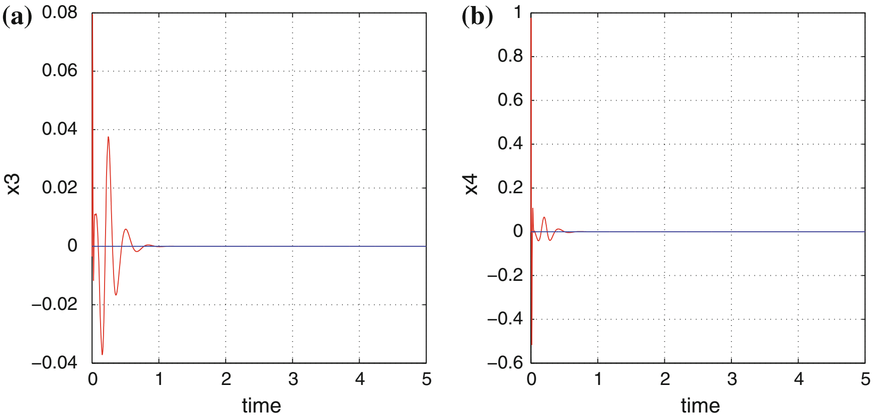


Fig. 6.61 Suspension control under disturbances profile 3: **a** convergence of unsprung mass position x_2 to the desirable setpoint, **b** convergence of unsprung mass velocity x_4 to the desirable setpoint

In this section, a new flatness-based control method is developed for quadrotors that is based on differential flatness theory together with the use of a disturbance observer that is also in accordance to differential flatness theory (the previously analyzed Derivative-free nonlinear Kalman Filter). The differential flatness theory-based design of the controller uses a change of coordinates (diffeomorhism) that transforms the state-space equation of the quadrotor's model into the linear canonical (Brunovsky) form [55, 152, 340, 427, 465, 535]. For the linearized equivalent of the quadrotor it is easier to design a state feedback controller using techniques for linear feedback controllers' synthesis. To supply the quadrotor's control loop with additional robustness, a disturbance observer is used. The disturbance observer

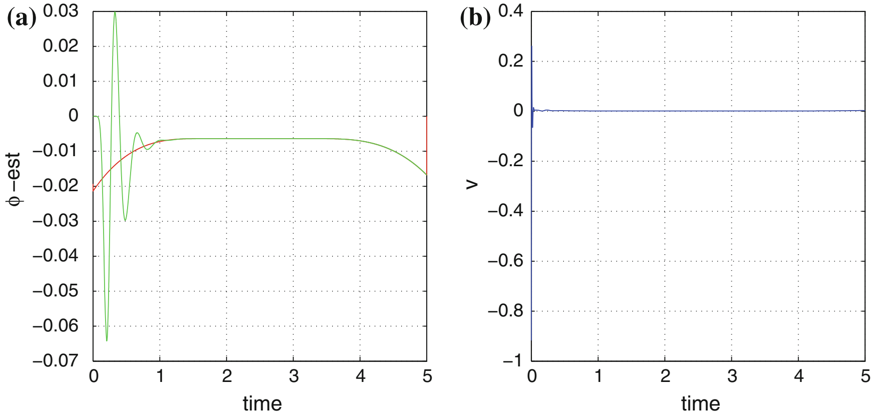


Fig. 6.62 Suspension control under disturbances profile 3: **a** estimation of the disturbance terms, **b** control input generated by the actuator

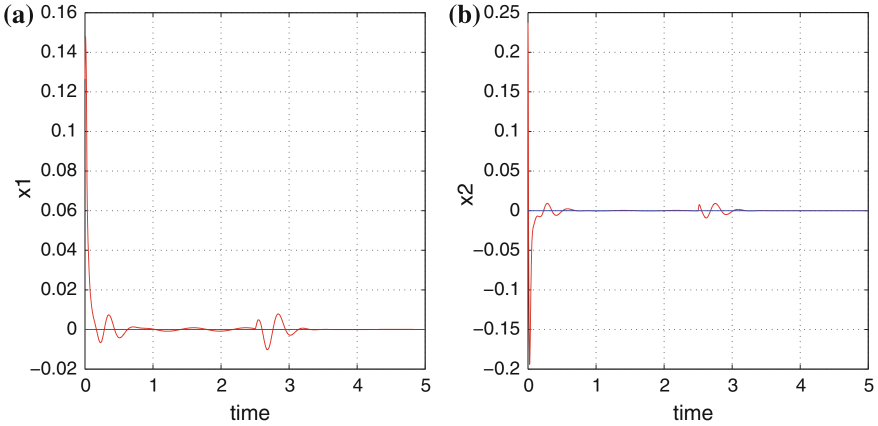


Fig. 6.63 Suspension control under disturbances profile 4: **a** convergence of sprung mass position x_1 to the desirable setpoint, **b** convergence of sprung mass velocity x_2 to the desirable setpoint

makes use of the standard Kalman Filter recursion on the linearized model of the quadrotor. It is capable of estimating simultaneously the quadrotor's linear and rotational velocities, as well as the vector of disturbances that affect the quadrotor's model without the need to compute Jacobian matrices. The accurate estimation of the disturbance inputs enables to introduce an additional control term that compensates for the disturbances' effects. The accurate tracking of reference trajectories that is performed by the quadrotor despite the existence of external disturbances is shown in simulation experiments.

Differential flatness theory has specific advantages when used in nonlinear control systems [55, 152, 340, 427, 465, 535]. By enabling an exact linearization of the

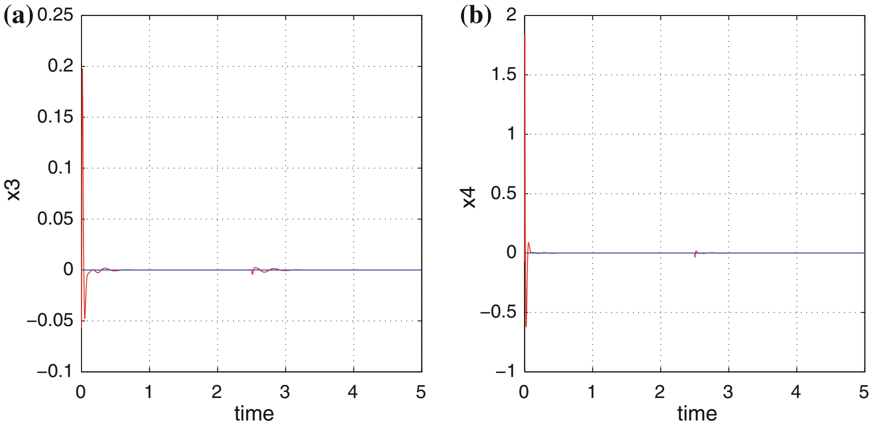


Fig. 6.64 Suspension control under disturbances profile 4: **a** convergence of unsprung mass position x_2 to the desirable setpoint, **b** convergence of unsprung mass velocity x_4 to the desirable setpoint

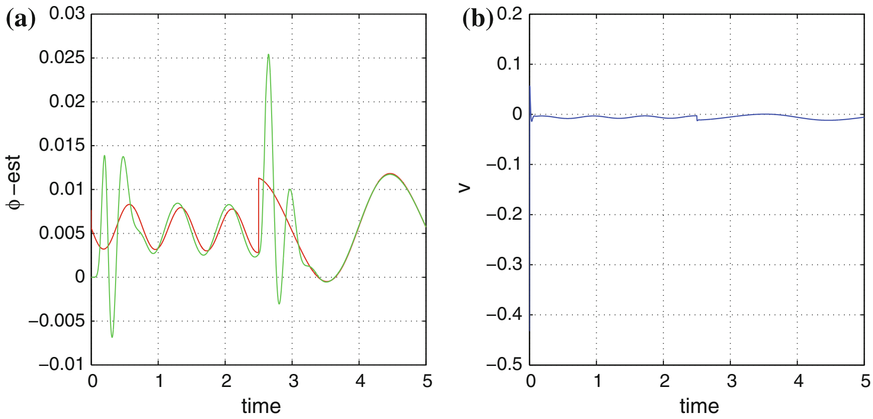


Fig. 6.65 Suspension control under disturbances profile 4: **a** estimation of the disturbance terms, **b** control input generated by the actuator

system's dynamical model it makes possible to avoid the use of linear models of local validity in the controller's design. The controller performs efficiently despite the change of operating points. After the design of such a state feedback controller, one can consider the inclusion in the control loop of supplementary control terms that will provide additional robustness. Thus one can design flatness-based adaptive fuzzy controllers or flatness-based sliding-mode controllers. As mentioned above, it is also possible to use a disturbance estimator-based auxiliary control input for compensating for the effects of disturbances in the feedback control loop. Moreover, the use of differential flatness theory in the design of state estimators and filters has also several strong points. One can perform estimation of the complete state vector of the system without the need to compute partial derivatives and Jacobian matrices. Additionally, by avoiding numerical errors which are due to approximate linearization of the system's dynamic model (e.g., with the use of expansion in Taylor series), linear estimation algorithms can be implemented. In the case of Kalman Filtering, this means that one can perform state estimation with the use of the standard Kalman Filter recursion, thus preserving the method's optimality features and providing state estimates of improved precision (e.g., comparing to Extended Kalman Filtering).

6.7.2 Kinematic Model of the Quadropter

As shown in Figs. 6.66 and 6.67, two reference frames are defined [397, 398]. The first one $B = [B_1, B_2, B_3]$ is attached to the quadropter's body, whereas the second $E = [E_x, E_y, E_z]$ is considered to be an inertial coordinates system. The Euler angles defining rotation round the axes of the inertial reference frame E_1, E_2, E_3 are defined as θ, ϕ, and ψ, respectively. The two reference frames are connected to each other through a rotation matrix

$$R = \begin{pmatrix} C\psi C\theta & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ S\psi C\theta & S\psi S\theta S\phi + C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & C\theta S\phi & C\theta C\phi \end{pmatrix} \tag{6.135}$$

where $C = cos(\cdot)$ and $S = sin(\cdot)$.

The connection between velocities in the two reference frames is as follows:

$$V_E = R \cdot V_B \tag{6.136}$$

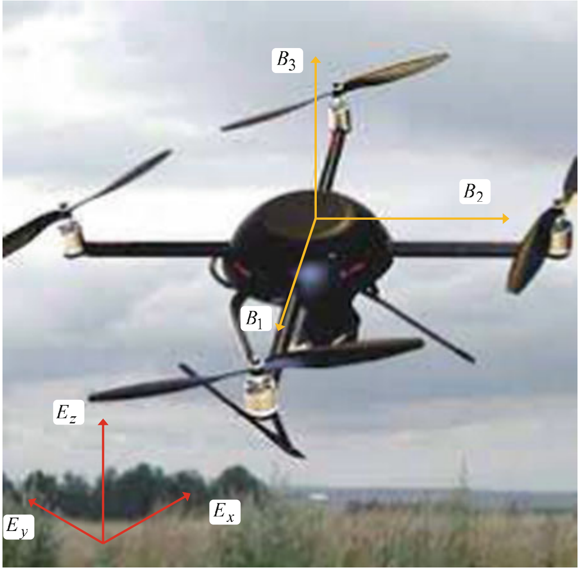


Fig. 6.66 Quadrotor performing a surveillance task and the associated reference frames

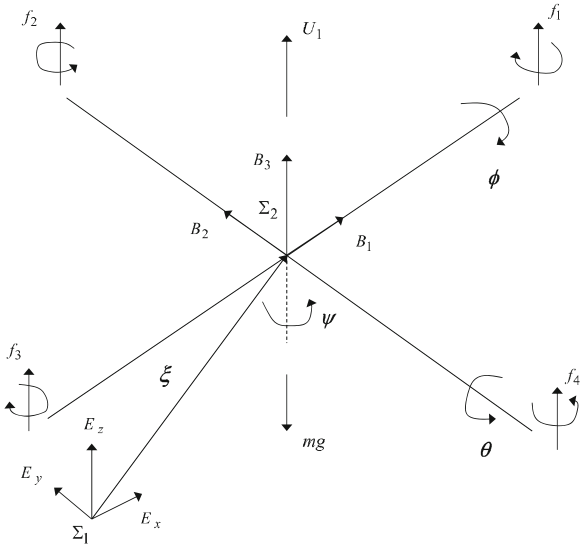


Fig. 6.67 Reference axes for the quadropter

where $V_E = [u_E, v_E, w_E]$ and $V_B = [u_B, v_B, w_B]$ are the linear velocity vectors expressed in the two reference frames. About the angular velocities in the two reference frames the following relation holds

$$\dot{\eta} = W^{-1}\omega \tag{6.137}$$

that is

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & sin(\phi)tan(\theta) & cos(\phi)tan(\theta) \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi)sec(\theta) & cos(\phi)sec(\theta) \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \tag{6.138}$$

where $\eta = [\phi, \theta, \psi]^T$ is the angular velocities vector in the inertial reference frame and $\omega = [p, q, r]^T$ is the angular velocities vector in the body-fixed reference frame.

6.7.3 Euler-Lagrange Equations for the Quadropter

The Euler-Lagrange equation for the quadropter is formulated as follows:

$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{q}_i}) - \frac{\partial L}{\partial q_i} = \begin{pmatrix} f_\xi \\ \tau_\eta \end{pmatrix} \tag{6.139}$$

where the Lagrangian is defined as $L(q, \dot{q}) = E_{C_{trans}} + E_{C_{rot}} - E_p$, where $E_{C_{trans}}$ is the kinetic energy of the quadrotor due to translational motion, $E_{C_{rot}}$ is the kinetic energy of the quadrotor due to rotational motion, and E_p is the total potential energy of the quadrotor due to lift. The generalized state vector is $q = [\xi^T, \eta^T]^T \in R^6$, $\tau_\eta \in R^3$

is the torques vector that causes rotation round the axes of the body-fixed reference frame, and $f_\xi = R\hat{f} + \alpha_T$ is the translational force applied to the quadrotor due to the main control input U_1 along the z-axis direction, while $\alpha_T = [A_x, A_y, A_z]^T$ are the aerodynamic forces vector, defined along the axes of the inertial reference frame. Since the Lagrangian does not contain cross-coupling between the $\dot{\xi}$ and the $\dot{\eta}$ terms, the Lagrange-Euler equations can be divided into translational and rotational dynamics.

The translational dynamics of the quadropter is given by

$$m\ddot{\xi} + mge_3 = f_\xi \tag{6.140}$$

where $e_3 = [0, 0, 1]^T$ is the unit vector along the z axis of the inertial reference frame. Equation (6.140) can be written using the following three equations

$$\begin{array}{c} \ddot{x} = \frac{1}{m}(cos(\psi)sin(\theta)cos(\phi) + sin(\psi)sin(\phi))U_1 + \frac{A_x}{m} \\ \ddot{y} = \frac{1}{m}(sin(\psi)sin(\theta)cos(\phi) - cos(\psi)sin(\phi))U_1 + \frac{A_y}{m} \\ \ddot{z} = -g + \frac{1}{m}(cos(\theta)cos(\phi))U_1 + \frac{A_z}{m} \end{array} \tag{6.141}$$

where m is the quadropter's mass and g is the gravitational acceleration.

The rotational dynamics of the quadropter is given by

$$M(\eta)\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} = \tau_\eta \tag{6.142}$$

where the inertia matrix $M(\eta)$ is defined as

$$M(\eta) = \begin{pmatrix} I_{xx} & 0 & -I_{xx}S\theta \\ 0 & I_{yy}C^2\phi + I_{zz}S^2\phi & (I_{yy} - I_{zz})C\phi S\phi C\theta \\ -I_{xx}S\theta & (I_{yy} - I_{zz})C\phi S\phi C\theta & I_{xx}S^2\theta + I_{yy}S^2\phi C^2\theta + I_{zz}C^2\phi C^2\theta \end{pmatrix} \tag{6.143}$$

and the Coriolis matrix is

$$C(\eta, \dot{\eta}) = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{pmatrix} \tag{6.144}$$

where the elements of the matrix are

$$\begin{aligned} &c_{11} = 0 \\ &c_{12} = (I_{yy} - I_{zz})(\dot{\theta}C\phi S\phi + \dot{\psi}S^2\phi C\theta) + (I_{zz} - I_{yy})\dot{\psi}C^2\phi C\theta \\ &c_{13} = (I_{zz} - I_{yy})\dot{\psi}C\phi S\phi C^2\theta \\ &c_{21} = (I_{zz} - I_{yy})(\dot{\theta}C\phi S\phi + \dot{\psi}S^2\phi C\theta) + (I_{yy} - I_{zz})\dot{\psi}C^2\phi C\theta + I_{xx}\dot{\psi}C\theta \\ &c_{22} = (I_{zz} - I_{yy})\dot{\phi}C\phi S\phi \end{aligned}$$

$$
\begin{aligned}
& c_{23} = -I_{xx}\dot{\psi}S\theta C\theta + I_{yy}\dot{\psi}S^2\phi C\theta S\theta + I_{zz}\dot{\psi}C^2\phi S\theta C\theta \\
& c_{31} = (I_{yy} - I_{zz})\dot{\psi}C^2\theta S\phi C\phi - I_{xx}\dot{\theta}C\theta \\
& c_{32} = (I_{zz} - I_{yy})(\dot{\theta}C\phi S\phi S\theta + \dot{\phi}S^2\phi C\theta) + (I_{yy} - I_{zz})\dot{\phi}C^2\phi C\theta + I_{xx}\dot{\psi}S\theta C\theta - \\
& -I_{yy}\dot{\psi}S^2\phi S\theta C\theta - I_{zz}\dot{\psi}C^2\phi S\theta C\theta \\
& c_{33} = (I_{yy} - I_{zz})\dot{\phi}C\phi S\phi C^2\theta - I_{yy}\dot{\theta}S^2\phi C\theta S\theta - \\
& -I_{zz}\dot{\theta}C^2\phi C\theta S\theta + I_{xx}\dot{\theta}C\theta S\theta
\end{aligned} \tag{6.145}
$$

Thus, the mathematical model that describes the quadrotor's rotational motion is given by

$$\ddot{\eta} = M(\eta)^{-1}(\tau_\eta - C(\eta, \dot{\eta})\dot{\eta}) \tag{6.146}$$

Denoting $w = M(\eta)^{-1}(\tau_\eta - C(\eta, \dot{\eta})\dot{\eta})$, one has the following notation for the rotational dynamics

$$\begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = \begin{pmatrix} w_a \\ w_b \\ w_c \end{pmatrix} \tag{6.147}$$

Considering small variations of the heading angle of the quadrotor round $\psi = \frac{\pi}{2}$, denoting $w_1 = U_1/m$ and taking also that the aerodynamic coefficients $A_x, A_y, A_z \ll m$, a simplified quadropter is formulated as follows [541]

$$
\begin{array}{ccc}
\ddot{x} = w_1 sin(\phi) & \ddot{y} = w_1 cos(\phi)sin(\theta) & \ddot{z} = w_1 cos(\phi)cos(\theta) - g \\
\ddot{\phi} = w_a & \ddot{\theta} = w_b & \ddot{\psi} = w_c
\end{array} \tag{6.148}
$$

6.7.4 *Design of Flatness-Based Controller for the Quadrotor's Model*

It will be shown, that the quadrotor's model given in Eq. (6.148) is a differentially flat one, i.e., that all its state variables and the associated control inputs can be written as functions of a new variable called flat output and of its derivatives.

The following state variables are introduced $x_1 = x$, $x_2 = \dot{x}$, $x_3 = y$, $x_4 = \dot{y}$, $x_5 = z$, $x_6 = \dot{z}$, $x_7 = \phi$, $x_8 = \dot{\phi}$, $x_9 = \theta$, $x_{10} = \dot{\theta}$, $x_{11} = \psi$, $x_{12} = \dot{\psi}$. Thus, one has the following state-space description for the quadrotor's dynamic model

$$
\begin{array}{cccc}
\dot{x}_1 = x_2 & \dot{x}_2 = w_1 sin(x_7) & \dot{x}_3 = x_4 & \dot{x}_4 = w_1 cos(x_7)sin(x_9) \\
\dot{x}_5 = x_6 & \dot{x}_6 = w_1 cos(x_7)cos(x_9) & \dot{x}_7 = x_8 & \dot{x}_8 = w_a \\
\dot{x}_9 = x_{10} & \dot{x}_{10} = w_b & \dot{x}_{11} = x_{12} & \dot{x}_{12} = w_c
\end{array} \tag{6.149}
$$

The flat output of the system is taken to be the vector $y_f = [x_1, x_3, x_5, x_7, x_9, x_{11}]^T$. It holds that

$$\begin{array}{llll}
x_1=[1\,0\,0\,0\,0\,0]y_f & x_2=[1\,0\,0\,0\,0\,0]\dot{y}_f & x_3=[0\,1\,0\,0\,0\,0]y_f & x_4=[0\,1\,0\,0\,0\,0]\dot{y}_f \\
x_5=[0\,0\,1\,0\,0\,0]y_f & x_6=[0\,0\,1\,0\,0\,0]\dot{y}_f & x_7=[0\,0\,0\,1\,0\,0]y_f & x_8=[0\,0\,0\,1\,0\,0]\dot{y}_f \\
x_9=[0\,0\,0\,0\,1\,0]y_f & x_{10}=[0\,0\,0\,0\,1\,0]\dot{y}_f & x_{11}=[0\,0\,0\,0\,0\,1]y_f & x_{12}=[0\,0\,0\,0\,0\,1]\dot{y}_f
\end{array} \tag{6.150}$$

According to Eq. (6.150) all state variables of the quadropter can be written as functions of the flat output and its derivatives. Using this and Eq. (6.149), one also has that the control inputs of the quadropter's model, w_1, w_a, w_b, and w_c can be written as functions of the flat output and its derivatives. Therefore, it is confirmed that the system is a differentially flat one.

Defining now the new control inputs

$$\begin{array}{lll}
v_1=w_1 sin(x_7) & v_2=w_1 cos(x_7) sin(x_9) & v_3=w_1 cos(x_7) cos(x_9) \\
v_4=w_a & v_5=w_b & v_6=w_c
\end{array} \tag{6.151}$$

one has the following state-space description for the system

$$\begin{pmatrix} \dot{y}_{f_1} \\ \ddot{y}_{f_1} \\ \dot{y}_{f_2} \\ \ddot{y}_{f_2} \\ \dot{y}_{f_3} \\ \ddot{y}_{f_3} \\ \dot{y}_{f_4} \\ \ddot{y}_{f_4} \\ \dot{y}_{f_5} \\ \ddot{y}_{f_5} \\ \dot{y}_{f_6} \\ \ddot{y}_{f_6} \end{pmatrix} = \begin{pmatrix}
0&1&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&1&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&1&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&1&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&1&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&0&1 \\
0&0&0&0&0&0&0&0&0&0&0&0
\end{pmatrix} \begin{pmatrix} y_{f_1} \\ \dot{y}_{f_1} \\ y_{f_2} \\ \dot{y}_{f_2} \\ y_{f_3} \\ \dot{y}_{f_3} \\ y_{f_4} \\ \dot{y}_{f_4} \\ y_{f_5} \\ \dot{y}_{f_5} \\ y_{f_6} \\ \dot{y}_{f_6} \end{pmatrix} + \begin{pmatrix}
0&0&0&0&0&0 \\
1&0&0&0&0&0 \\
0&0&0&0&0&0 \\
0&1&0&0&0&0 \\
0&0&0&0&0&0 \\
0&0&1&0&0&0 \\
0&0&0&0&0&0 \\
0&0&0&1&0&0 \\
0&0&0&0&0&0 \\
0&0&0&0&1&0 \\
0&0&0&0&0&0 \\
0&0&0&0&0&1
\end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{pmatrix} \tag{6.152}$$

and the measurement equation for this system becomes

$$\begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \end{pmatrix} = \begin{pmatrix}
1&0&0&0&0&0&0&0&0&0&0&0 \\
0&0&1&0&0&0&0&0&0&0&0&0 \\
0&0&0&0&1&0&0&0&0&0&0&0 \\
0&0&0&0&0&0&1&0&0&0&0&0 \\
0&0&0&0&0&0&0&0&1&0&0&0 \\
0&0&0&0&0&0&0&0&0&0&1&0
\end{pmatrix} \begin{pmatrix} y_{f_1} \\ \dot{y}_{f_1} \\ y_{f_2} \\ \dot{y}_{f_2} \\ y_{f_3} \\ \dot{y}_{f_3} \\ y_{f_4} \\ \dot{y}_{f_4} \\ y_{f_5} \\ \dot{y}_{f_5} \\ y_{f_6} \\ \dot{y}_{f_6} \end{pmatrix} \tag{6.153}$$

Thus, using differential flatness theory, the quadrotor's model has been written in a MIMO linear canonical (Brunovsky) form, which is both controllable and observable.

After being written in the linear canonical form, the quadrotor's state-space equation comprises six subsystems of the form

$$\ddot{y}_{f_i} = v_i, \ i = 1, \ldots, 6 \tag{6.154}$$

For each one of these subsystems, a controller can be defined as follows:

$$v_i = \ddot{y}_{f_i}^d - k_{d_i}(\dot{y}_{f_i} - \dot{y}_{f_i}^d) - k_{p_i}(y_{f_i} - y_{f_i}^d), \ i = 1, \ldots, 6 \tag{6.155}$$

The control scheme is implemented in the form of two cascading loops. The inner control loop controls rotation angles, while the outer control loop sets the desired values of the rotation angles, so as to control position in the xyz-reference system. The computation of the reference setpoints for the rotation angles $\phi_d(t)$, $\theta_d(t)$, and $\psi_d(t)$ and for the cartesian coordinates $x_d(t)$,$y_d(t)$, and $z_d(t)$ takes into account the constraints imposed by the system dynamics.

6.7.5 Estimation of the Quadrotor's Disturbance Forces and Torques with Kalman Filtering

It was shown that the initial nonlinear model of the quadrotor can be written in the MIMO canonical form of Eqs. (6.152) and (6.153). Next, it is assumed that the quadrotor's model is affected by additive input disturbances, thus one has

$$\begin{gathered} \ddot{x}_1 = (w_1 + d_1)sin(x_7) \\ \ddot{x}_3 = (w_1 + d_1)cos(x_7)sin(x_9) \\ \ddot{x}_5 = (w_1 + d_1)cos(x_7)cos(x_9) \\ \ddot{x}_7 = w_a + d_a \\ \ddot{x}_9 = w_b + d_b \\ \ddot{x}_{11} = w_c + d_c \end{gathered} \tag{6.156}$$

or using the new state variables y_{f_i} $i = 1, \ldots, 12$ of the differential flatness theory-based model and the transformed inputs $v_i, i = 1, \ldots, 6$ one has

$$\begin{gathered} \ddot{y}_{f_1} = v_1 + d_1 sin(y_{f_7}) \\ \ddot{y}_{f_3} = v_2 + d_1 cos(y_{f_7})sin(y_{f_9}) \\ \ddot{y}_{f_5} = v_3 + d_1 cos(y_{f_7})cos(y_{f_9}) \\ \ddot{y}_{f_7} = v_4 + d_a \\ \ddot{y}_{f_9} = v_5 + d_b \\ \ddot{y}_{f_{11}} = v_6 + d_c \end{gathered} \tag{6.157}$$

while by redefining the disturbance terms as $\tilde{d}_1 = d_1 sin(y_{f_7})$, $\tilde{d}_2 = d_1 cos(y_{f_7}) sin(y_{f_9})$, $\tilde{d}_3 = d_1 cos(y_{f_7}) cos(y_{f_9})$, $\tilde{d}_4 = d_a$, $\tilde{d}_5 = d_b$ and $\tilde{d}_6 = d_c$, the dynamics of the disturbed system can be written as

$$\begin{aligned} \ddot{y}_{f_1} &= v_1 + \tilde{d}_1 \\ \ddot{y}_{f_3} &= v_2 + \tilde{d}_2 \\ \ddot{y}_{f_5} &= v_3 + \tilde{d}_3 \\ \ddot{y}_{f_7} &= v_4 + \tilde{d}_4 \\ \ddot{y}_{f_9} &= v_5 + \tilde{d}_5 \\ \ddot{y}_{f_{11}} &= v_6 + \tilde{d}_6 \end{aligned} \tag{6.158}$$

The system's dynamics can be also written as $\dot{y}_{f_1} = y_{f_2}$, $\dot{y}_{f_2} = v_1 + \tilde{d}_1$, $\dot{y}_{f_3} = y_{f_4}$, $\dot{y}_{f_4} = v_2 + \tilde{d}_2$, $\dot{y}_{f_5} = y_{f_6}$, $\dot{y}_{f_6} = v_3 + \tilde{d}_3$, $\dot{y}_{f_7} = y_{f_8}$, $\dot{y}_{f_8} = v_4 + \tilde{d}_4$, $\dot{y}_{f_9} = y_{f_{10}}$, $\dot{y}_{f_{10}} = v_5 + \tilde{d}_5$, $\dot{y}_{f_{11}} = y_{f_6}$, $\dot{y}_{f_6} = v_6 + \tilde{d}_6$.

Without loss of generality, it is assumed that the dynamics of the disturbances terms are described by their second-order derivative, i.e., $\ddot{\tilde{d}}_i = f_{d_i}$, $i = 1, \ldots, 6$. Next, the extended state vector of the system is defined so as to include disturbance terms as well. Thus one has the following state variables

$$\begin{array}{llllll} z_{f_1} = y_{f_1} & z_{f_2} = y_{f_2} & z_{f_3} = y_{f_3} & z_{f_4} = y_{f_4} & z_{f_5} = y_{f_5} & z_{f_6} = y_{f_6} \\ z_{f_7} = y_{f_7} & z_{f_8} = y_{f_8} & z_{f_9} = y_{f_9} & z_{f_{10}} = y_{f_{10}} & z_{f_{11}} = y_{f_{11}} & z_{f_{12}} = y_{f_{12}} \\ z_{f_{13}} = \tilde{d}_1 & z_{f_{14}} = \dot{\tilde{d}}_1 & z_{f_{15}} = \ddot{\tilde{d}}_1 & z_{f_{16}} = \tilde{d}_2 & z_{f_{17}} = \dot{\tilde{d}}_2 & z_{f_{18}} = \ddot{\tilde{d}}_2 \\ z_{f_{19}} = \tilde{d}_3 & z_{f_{20}} = \dot{\tilde{d}}_3 & z_{f_{21}} = \ddot{\tilde{d}}_3 & z_{f_{22}} = \tilde{d}_4 & z_{f_{23}} = \dot{\tilde{d}}_4 & z_{f_{24}} = \ddot{\tilde{d}}_4 \\ z_{f_{25}} = \tilde{d}_5 & z_{f_{26}} = \dot{\tilde{d}}_6 & z_{f_{27}} = \ddot{\tilde{d}}_5 & z_{f_{28}} = \tilde{d}_6 & z_{f_{29}} = \dot{\tilde{d}}_6 & z_{f_{30}} = \ddot{\tilde{d}}_6 \end{array} \tag{6.159}$$

Thus, the disturbed system can be described by a state-space equation of the form

$$\begin{aligned} \dot{z}_f &= A_f z_f + B_f v \\ z_f^{meas} &= C_f z_f \end{aligned} \tag{6.160}$$

where matrices $A_f \in R^{30 \times 30}$, $B_f \in R^{30 \times 6}$, and $C_f \in R^{6 \times 30}$, are described in Eq. (6.162).

For the model of Eqs. (6.160) and (6.162), and after carrying out discretization of matrices A_f, B_f, and C_f with common discretization methods, one can implement the standard Kalman Filter algorithm using Eqs. (4.5) and (4.6). This is Derivative-free nonlinear Kalman filtering for the model of the quadropter which, unlike EKF, is performed without the need to compute Jacobian matrices and does not introduce numerical errors due to approximative linearization with Taylor series expansion.

The dynamics of the disturbance terms $\tilde{d}_i$, $i = 1, \ldots, 6$ are taken to be unknown in the design of the associated disturbances' estimator. Defining as $\tilde{A}_d$, $\tilde{B}_d$, and $\tilde{C}_d$, the discrete-time equivalents of matrices $\tilde{A}_f$, $\tilde{B}_f$, and $\tilde{C}_f$, respectively, one has the following dynamics:

$$\dot{\hat{z}}_f = \tilde{A}_f \cdot \hat{z}_f + \tilde{B}_f \cdot \tilde{v} + K(z_f^{meas} - \tilde{C}_f \hat{z}_f) \tag{6.161}$$

where $K \in R^{30\times 6}$ is the state estimator's gain. The associated Kalman Filter-based disturbance estimator is given by [422, 453]

$$A_f = \begin{pmatrix} 0_{1\times 1} & 1 & 0_{1\times 28} \\ 0_{1\times 12} & 1 & 0_{1\times 17} \\ 0_{1\times 3} & 1 & 0_{1\times 26} \\ 0_{1\times 15} & 1 & 0_{1\times 14} \\ 0_{1\times 5} & 1 & 0_{1\times 24} \\ 0_{1\times 18} & 1 & 0_{1\times 11} \\ 0_{1\times 7} & 1 & 0_{1\times 22} \\ 0_{1\times 21} & 1 & 0_{1\times 8} \\ 0_{1\times 9} & 1 & 0_{1\times 20} \\ 0_{1\times 24} & 1 & 0_{1\times 5} \\ 0_{1\times 11} & 1 & 0_{1\times 18} \\ 0_{1\times 27} & 1 & 0_{1\times 2} \\ 0_{1\times 13} & 1 & 0_{1\times 16} \\ 0_{1\times 14} & 1 & 0_{1\times 15} \\ 0_{1\times 30} & & \\ 0_{1\times 16} & 1 & 0_{1\times 13} \\ 0_{1\times 17} & 1 & 0_{1\times 12} \\ 0_{1\times 30} & & \\ 0_{1\times 19} & 1 & 0_{1\times 10} \\ 0_{1\times 20} & 1 & 0_{1\times 9} \\ 0_{1\times 30} & & \\ 0_{1\times 22} & 1 & 0_{1\times 7} \\ 0_{1\times 23} & 1 & 0_{1\times 6} \\ 0_{1\times 30} & & \\ 0_{1\times 25} & 1 & 0_{1\times 4} \\ 0_{1\times 26} & 1 & 0_{1\times 3} \\ 0_{1\times 30} & & \\ 0_{1\times 28} & 1 & 0_{1\times 1} \\ 0_{1\times 29} & 1 & \\ 0_{1\times 30} & & \end{pmatrix} B_f = \begin{pmatrix} 0_{1\times 6} & & \\ 1 & 0_{1\times 5} & \\ 0_{1\times 1} & 1 & 0_{1\times 4} \\ 0_{1\times 6} & & \\ 0_{1\times 2} & 1 & 0_{1\times 3} \\ 0_{1\times 6} & & \\ 0_{1\times 3} & 1 & 0_{1\times 2} \\ 0_{1\times 6} & & \\ 0_{1\times 4} & 1 & 0_{1\times 1} \\ 0_{1\times 6} & & \\ 0_{1\times 5} & 1 & \\ 0_{18\times 6} & & \end{pmatrix} C_f = \begin{pmatrix} 1 & 0_{1\times 29} & \\ 0_{1\times 2} & 1 & 0_{1\times 27} \\ 0_{1\times 4} & 1 & 0_{1\times 25} \\ 0_{1\times 6} & 1 & 0_{1\times 23} \\ 0_{1\times 8} & 1 & 0_{1\times 21} \\ 0_{1\times 10} & 1 & 0_{1\times 19} \end{pmatrix} \tag{6.162}$$

measurement update:

$$\begin{aligned} &K(k) = P^{-}(k)\tilde{C}_d^T[\tilde{C}_d \cdot P^{-}(k)\tilde{C}_d^T + R]^{-1} \\ &\hat{z}_f(k) = \hat{z}_f^{-}(k) + K(k)[z_f^{meas}(k) - \tilde{C}_d\hat{z}_f^{-}(k)] \\ &P(k) = P^{-}(k) - K(k)\tilde{C}_d P^{-}(k) \end{aligned} \tag{6.163}$$

time update:

$$\begin{aligned} &P^{-}(k+1) = \tilde{A}_d(k)P(k)\tilde{A}_d^T(k) + Q(k) \\ &\hat{z}_f^{-}(k+1) = \tilde{A}_d(k)\hat{z}_f(k) + \tilde{B}_d(k)\tilde{v}(k) \end{aligned} \tag{6.164}$$

To compensate for the effects of the disturbance forces it suffices to use in the control loop the modified control input vector

$$v = \begin{pmatrix} v_1 - \hat{\tilde{d}}_1 \\ v_2 - \hat{\tilde{d}}_2 \\ v_3 - \hat{\tilde{d}}_3 \\ v_4 - \hat{\tilde{d}}_4 \\ v_5 - \hat{\tilde{d}}_5 \\ v_6 - \hat{\tilde{d}}_6 \end{pmatrix} \text{ or } v = \begin{pmatrix} v_1 - \hat{z}_{13} \\ v_2 - \hat{z}_{16} \\ v_3 - \hat{z}_{19} \\ v_4 - \hat{z}_{22} \\ v_5 - \hat{z}_{25} \\ v_6 - \hat{z}_{28} \end{pmatrix} \tag{6.165}$$

6.7.6 Simulation Tests

Initial simulation experiments were concerned with flight control of the quadropter in the disturbance-free case. The considered reference trajectories are shown in Fig. 6.68. The implementation of the flatness-based control enabled accurate tracking of the reference trajectories. Convergence has been succeeded for the linear position and velocity variables to the associated setpoints as it can be seen in Fig. 6.69a, b and in Fig. 6.70a. Moreover, there has been convergence of the angular position and velocity variables to the associated setpoints as it can be seen in Fig. 6.70a and in Fig. 6.71a, b.

Additional simulation experiments were concerned with control of the quadropter in flight under disturbance forces and torques. The estimation of the disturbance forces and torques is shown in Fig. 6.72. The implementation of the flatness-based

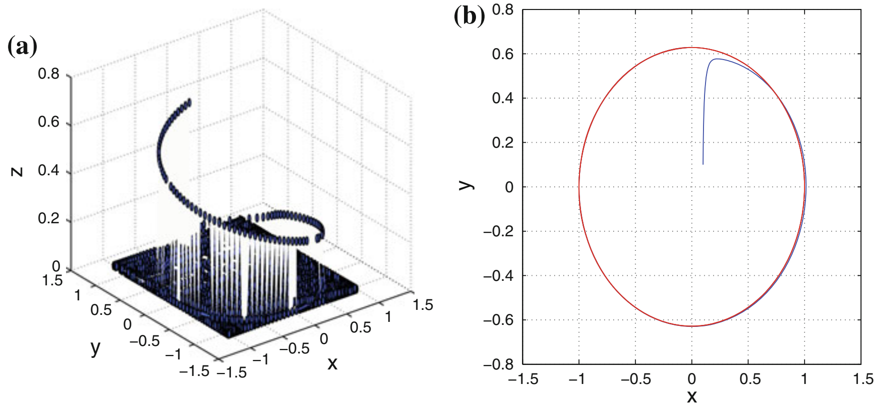


Fig. 6.68 Control of the quadrotor in the disturbance-free case: **a** trajectory of the quadrotor in the cartesian space, **b** projection of the quadrotor's trajectory in the xy plane

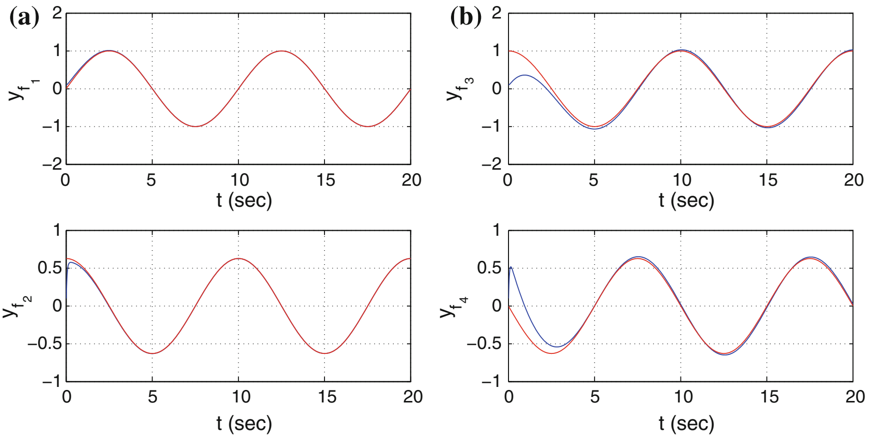


Fig. 6.69 Control of the quadrotor in the disturbance-free case: **a** position and velocity along the x axis, **b** position and velocity along the y axis

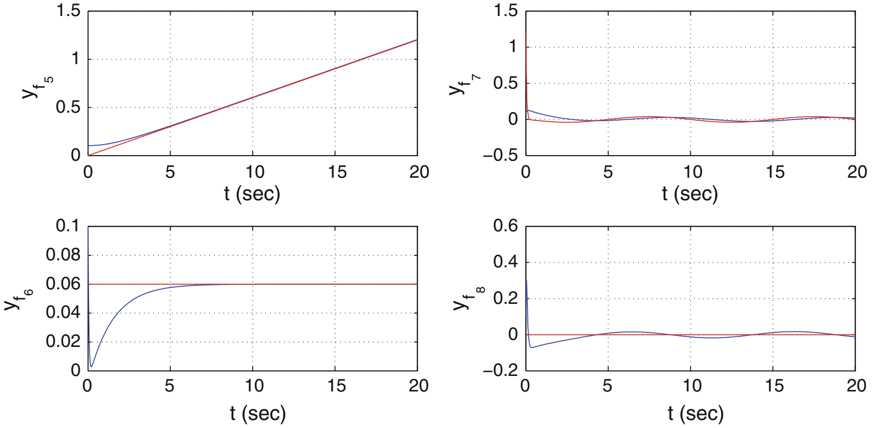


Fig. 6.70 Control of the quadrotor in the disturbance-free case: **a** position and velocity along the z axis, **b** rotation angle ϕ and associated angular speed

control enabled accurate tracking of the reference trajectories. There has been convergence of the linear position and velocity variables to the associated setpoints as it can be seen in Fig. 6.73a, b and in Fig. 6.74a. Moreover, there has been convergence of the angular position and velocity variables to the associated setpoints as it can be seen in Fig. 6.74b and in Fig. 6.75a, b.

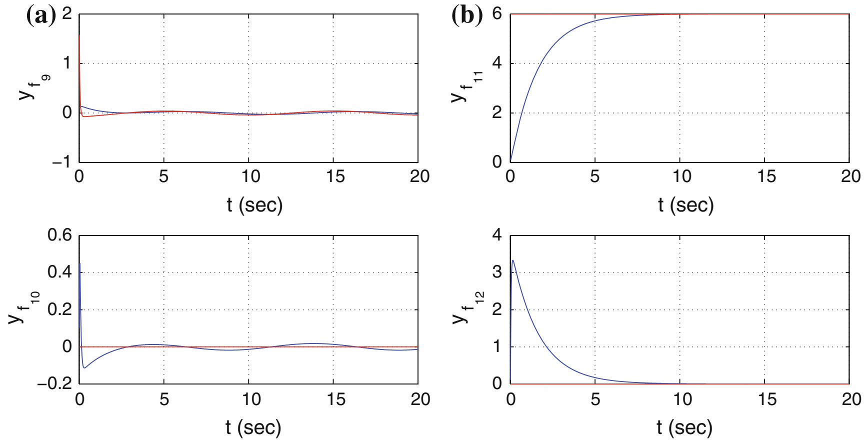


Fig. 6.71 Control of the quadrotor in the disturbance-free case: **a** rotation angle θ and associated angular speed, **b** rotation angle ψ and associated angular speed

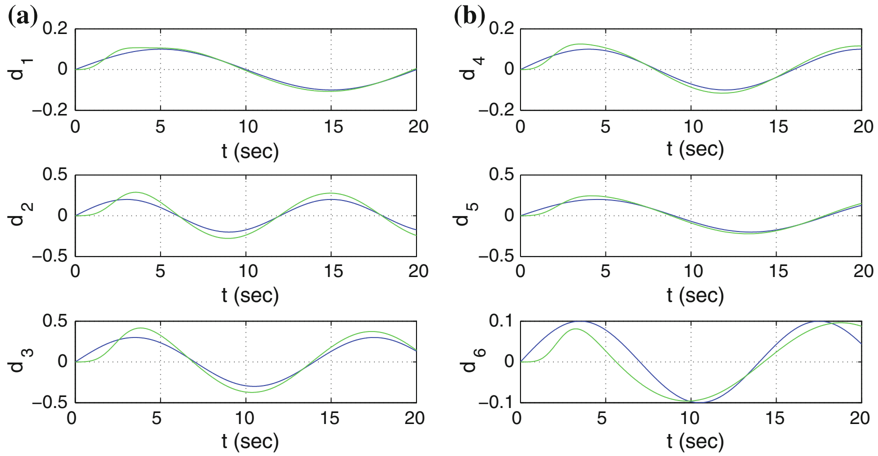


Fig. 6.72 Use of the Derivative-free nonlinear Kalman Filter in estimation of disturbances: **a** associated with linear motion, **b** associated with the rotational motion of the vehicle (*blue line* real value, *green line* estimated value)

6.8 State Estimation-Based Control of the Underactuated Hovercraft

6.8.1 Overview

Another significant application field for differential flatness theory-based methods for filtering and control is localization and autonomous navigation of unmanned

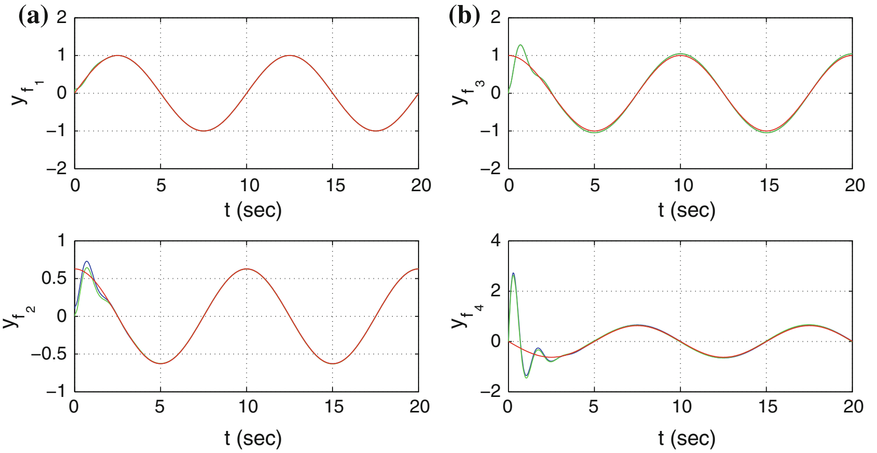


Fig. 6.73 Control of the quadrotor in the presence of external disturbances **a** position and velocity along the x axis, **b** position and velocity along the y axis (*blue line* real value, *green line* estimated value, *red line* setpoint)

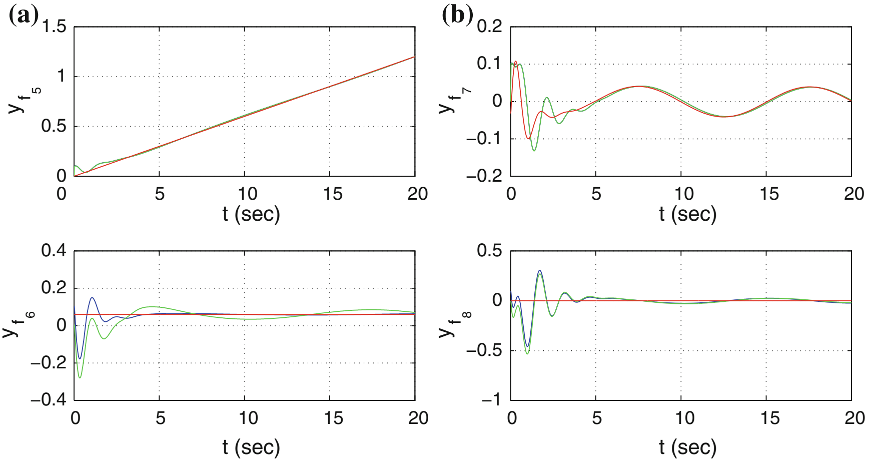


Fig. 6.74 Control of the quadrotor in the presence of external disturbances: **a** position and velocity along the z axis, **b** rotation angle ϕ and associated angular speed (*blue line* real value, *green line* estimated value, *red line* setpoint)

surface vessels (USVs) and autonomous underwater vessels (AUVs). In particular, the problem of autonomous navigation of surface vessels such as hovercrafts has received particular attention, since it can find use in both security purposes and passenger transportation [114, 401, 451, 492, 509]. The problem of control and trajectory tracking for unmanned surface vessels of the hovercraft type is nontrivial because the associated kinematic model is a high-order nonlinear one [17, 128, 184, 499, 531]. Another problem that has to be dealt with is that the hovercraft's model

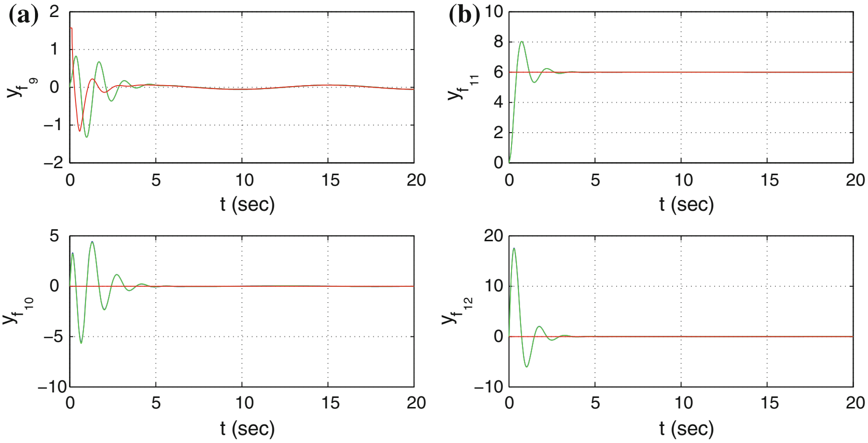


Fig. 6.75 Control of the quadrotor in the presence of disturbances: **a** rotation angle θ and associated angular speed, **b** rotation angle ψ and associated angular speed (*blue line* real value, *green line* estimated value, *red line* setpoint)

is underactuated [2, 34, 41, 191, 384, 486, 502]. Indicative results on control of underactuated dynamical systems can be found in [399, 400, 445]. Moreover, as explained in Chap. 2, the hovercraft's model cannot be subjected to static feedback linearization, but admits only dynamic feedback linearization. This means that to succeed 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 chapter proposes a solution to the control problem of hovercrafts control 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 functions of one single algebraic variable which is the flat output and of the flat output's derivatives [55, 152, 263, 286, 340, 427, 464, 465, 495]. 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 succeed 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 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 vessel and to identify additive disturbance terms that affect the system, the Derivative-free nonlinear Kalman Filter is redesigned as a disturbance observer [31, 408, 414, 435–437, 447]. This estimation algorithm consists of the standard

Kalman Filter recursion applied on the linearized equivalent of the hovercraft and of an inverse transformation that is based on differential flatness theory which permits to compute estimates of the state variables of the initial nonlinear system.

6.8.2 Lie Algebra-Based Control of the Underactuated Hovercraft

In Chap. 2, it was shown that the extended model of the vessel is a differentially flat one. Using its differential flatness properties, the vessel's model was transformed into the linear canonical (Brunovsky) form. It will be shown that one can arrive at an equivalent description of the hovercraft's dynamics using Lie algebra [461]. The sixth-order state-space equation of the underactuated hovercraft model (Figs. 2.3 and 6.76) is given by

$$\begin{aligned} \dot{x} &= ucos(\psi) - vsin(\psi) \\ \dot{y} &= usin(\psi) + vcos(\psi) \\ \dot{\psi} &= r \\ \dot{u} &= v{\cdot}r + \tau_u \\ \dot{v} &= -u{\cdot}r - \beta v \\ \dot{r} &= \tau_r \end{aligned} \tag{6.166}$$

where x and y are the cartesian coordinates of the vessel, ψ is the orientation angle, u is the surge velocity, v is the sway velocity, and r is the yaw rate. The hovercraft's model is also written in the matrix form:

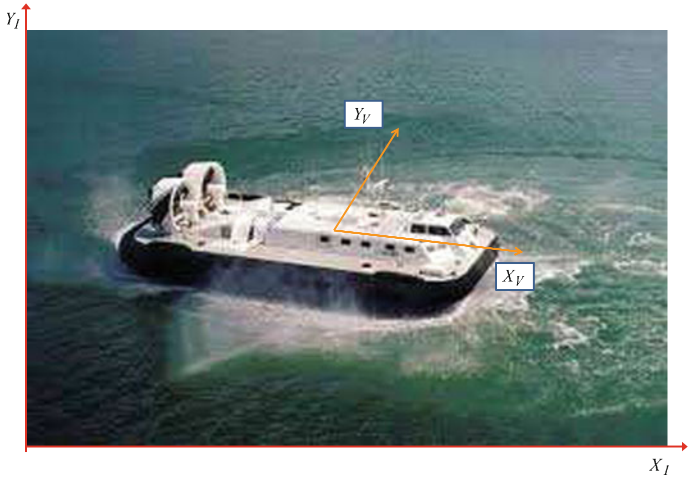


Fig. 6.76 Underactuated hovercraft performing maneuvers and the associated reference frames

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} ucos(\psi) - vsin(\psi) \\ usin(\psi) + vcos(\psi) \\ r \\ vr \\ -ur + \beta v \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \tau_u \\ \tau_r \end{pmatrix} \tag{6.167}$$

or equivalently, one has the description

$$\dot{\tilde{x}} = f(\tilde{x}) + g\tilde{v} \tag{6.168}$$

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 $g(\tilde{x}) = [g_a, g_b] \in R^{6\times 2}$, while the control input is the vector $\tilde{v} = [\tau_u, \tau_r]^T$.

The system's state vector is extended by including as additional state variables the control input τ_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 system becomes

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{r} \\ \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} = \begin{pmatrix} ucos(\psi) - vsin(\psi) \\ usin(\psi) + vcos(\psi) \\ r \\ vr + z_1 \\ -ur + \beta v \\ 0 \\ z_2 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \ddot{\tau}_u \\ \tau_r \end{pmatrix} \tag{6.169}$$

or equivalently, one has the description

$$\dot{z} = f(z) + g(z)\tilde{v} \tag{6.170}$$

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

$$z_{1.1} = x \;\; z_{2,1} = y \tag{6.171}$$

Moreover, the new state variables are defined

$$\begin{aligned} z_{1,2} &= L_f z_{1,1} \;\; z_{2,2} = L_f z_{2,1} \\ z_{1,3} &= L_f^2 z_{1,1} \;\; z_{2,3} = L_f^2 z_{2,1} \\ z_{1,4} &= L_f^3 z_{1,1} \;\; z_{2,4} = L_f^3 z_{2,1} \end{aligned} \tag{6.172}$$

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

$$\begin{array}{c}\dot{z}_{1,4}=L_f^4z_{1,1}+L_{g_a}L_f^3z_{1,1}\ddot{\tau}_u+L_{g_b}L_f^3z_{1,1}\tau_r\\ \dot{z}_{2,4}=L_f^4z_{2,1}+L_{g_a}L_f^3z_{2,1}\ddot{\tau}_u+L_{g_b}L_f^3z_{2,1}\tau_r\end{array} \tag{6.173}$$

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

$$\begin{array}{c}z_{1,2}=L_fz_{1,1}\Rightarrow\\ z_{1,2}=\frac{\partial z_{1,1}}{\partial x}f_1+\frac{\partial z_{1,1}}{\partial y}f_2+\frac{\partial z_{1,1}}{\partial \psi}f_3+\frac{\partial z_{1,1}}{\partial u}f_4+\frac{\partial z_{1,1}}{\partial v}f_5+\frac{\partial z_{1,1}}{\partial r}f_6+\frac{\partial z_{1,1}}{\partial z_1}f_7+\frac{\partial z_{1,1}}{\partial z_2}f_8\Rightarrow\\ z_{1,2}=1{\cdot}f_1\Rightarrow z_{1,2}=ucos(\psi)-vsin(\psi)\end{array} \tag{6.174}$$

Similarly, one obtains

$$\begin{array}{c}z_{1,3}=L_f^2z_{1,1}\Rightarrow\\ z_{1,3}=\frac{\partial z_{1,2}}{\partial x}f_1+\frac{\partial z_{1,2}}{\partial y}f_2+\frac{\partial z_{1,2}}{\partial \psi}f_3+\frac{\partial z_{1,2}}{\partial u}f_4+\frac{\partial z_{1,2}}{\partial v}f_5+\frac{\partial z_{1,2}}{\partial r}f_6+\frac{\partial z_{1,2}}{\partial z_1}f_7+\frac{\partial z_{1,2}}{\partial z_2}f_8\Rightarrow\\ z_{1,3}=(-usin(\psi)-vcos(\psi))f_3+cos(\psi)f_4-sin(\psi)f_5\Rightarrow\\ z_{1,3}=(-usin(\psi)-vcos(\psi))r+cos(\psi)(vr+z_1)-sin(\psi)(-ur+\beta v)\Rightarrow\\ z_{1,3}=\tau_ucos(\psi)+\beta vsin(\psi)\end{array} \tag{6.175}$$

Equivalently, it holds that

$$\begin{array}{c}z_{1,4}=L_f^3z_{1,1}\Rightarrow z_{1,4}=L_fz_{1,3}\Rightarrow\\ z_{1,4}=\frac{\partial z_{1,3}}{\partial x}f_1+\frac{\partial z_{1,3}}{\partial y}f_2+\frac{\partial z_{1,3}}{\partial \psi}f_3+\frac{\partial z_{1,3}}{\partial u}f_4+\frac{\partial z_{1,3}}{\partial v}f_5+\frac{\partial z_{1,3}}{\partial r}f_6+\frac{\partial z_{1,3}}{\partial z_1}f_7+\frac{\partial z_{1,3}}{\partial z_2}f_8\end{array} \tag{6.176}$$

while after intermediate operations one obtains

$$\begin{array}{c}z_{1,4}=(-\tau_usin(\psi)+\beta vcos(\psi))f_3+\beta sin(\psi)f_5+cos(\psi)f_7\Rightarrow\\ z_{1,4}=(-\tau_usin(\psi)+\beta cos(\psi))r+\beta sin(\psi)(-ur+\beta v)+cos(\psi)z_2\Rightarrow\\ z_{1,4}=(\dot{\tau}_usin(\psi)+\beta vcos(\psi))r+\beta sin(\psi)(-ur+\beta v)+cos(\psi)z_2\end{array} \tag{6.177}$$

or, using the extended state vector variables notation one has

$$z_{1,4}=z_2cos(\psi)-z_1sin(\psi)r-\beta ursin(\psi)-\beta^2vsin(\psi)+\beta vcos(\psi)r \tag{6.178}$$

It also holds that

$$\begin{array}{c}\dot{z}_{1,4}=L_f^4z_{1,1}+L_{g_a}L_f^3z_{1,1}\ddot{\tau}_u+L_{g_b}L_f^3z_{1,1}\tau_r\Rightarrow\\ \dot{z}_{1,4}=L_fz_{1,4}+L_{g_a}z_{1,4}\ddot{\tau}_u+L_{g_b}z_{1,4}\tau_r\end{array} \tag{6.179}$$

where

$$L_fz_{1,4}=\frac{\partial z_{1,4}}{\partial x}f_1+\frac{\partial z_{1,4}}{\partial y}f_2+\frac{\partial z_{1,4}}{\partial \psi}f_3+\frac{\partial z_{1,4}}{\partial u}f_4+\frac{\partial z_{1,4}}{\partial v}f_5+\frac{\partial z_{1,4}}{\partial r}f_6+\frac{\partial z_{1,4}}{\partial z_1}f_7+\frac{\partial z_{1,4}}{\partial z_2}f_8 \tag{6.180}$$

which gives

$$\begin{gathered}L_f z_{1,4} = (-z_2 sin(\psi) - z_1 cos(\psi)r - \beta urcos(\psi) - \beta^2 vcos(\psi) - \beta vsin(\psi)r)r+ \\ +(-\beta rsin(\psi))(vr + z_1) + (-\beta^2 sin(\psi) + \beta cos(\psi)r)(-ur + \beta v)+ \\ +(-z_1 sin(\psi) - \beta vsin(\psi) + \beta vcos(\psi))0 + (-sin(\psi)r)z_2\end{gathered} \tag{6.181}$$

while after some intermediate computations one obtains

$$\begin{gathered}L_f z_{1,4} = -2z_2 sin(\psi)r - z_1 cos(\psi)r^2 - \\ -\beta vr^2 sin(\psi) - \beta z_1 rsin(\psi)- \\ -\beta ur^2 cos(\psi) + \beta^2 ursin(\psi)- \\ -\beta^3 vsin(\psi) - \beta^2 vrcos(\psi) - \beta ur^2 cos^2(\psi) + \beta^2 vrcos(\psi) \\ -\beta vr^2 sin(\psi)\end{gathered} \tag{6.182}$$

In a similar manner, one computes

$$\begin{gathered}L_{g_a} z_{1,4} = \frac{\partial z_{1,4}}{\partial x} g_{a_1} + \frac{\partial z_{1,4}}{\partial y} g_{a_2} + \frac{\partial z_{1,4}}{\partial \psi} g_{a_3} + \frac{\partial z_{1,4}}{\partial u} g_{a_4}+ \\ +\frac{\partial z_{1,4}}{\partial v} g_{a_5} + \frac{\partial z_{1,4}}{\partial r} g_{a_6} + \frac{\partial z_{1,4}}{\partial z_1} g_{a_7} + \frac{\partial z_{1,4}}{\partial z_2} g_{a_8} \\ L_{g_a} z_{1,4} = \frac{\partial z_{1,4}}{\partial z_2} \Rightarrow L_{g_a} z_{1,4} = cos(\psi)\end{gathered} \tag{6.183}$$

and also

$$\begin{gathered}L_{g_b} z_{1,4} = \frac{\partial z_{1,4}}{\partial x} g_{b_1} + \frac{\partial z_{1,4}}{\partial y} g_{b_2} + \frac{\partial z_{1,4}}{\partial \psi} g_{b_3} + \frac{\partial z_{1,4}}{\partial u} g_{b_4}+ \\ +\frac{\partial z_{1,4}}{\partial v} g_{b_5} + \frac{\partial z_{1,4}}{\partial r} g_{b_6} + \frac{\partial z_{1,4}}{\partial z_1} g_{b_7} + \frac{\partial z_{1,4}}{\partial z_2} g_{b_8} \\ L_{g_b} z_{1,4} = \frac{\partial z_{1,4}}{\partial r} \Rightarrow L_{g_b} z_{1,4} = -z_1 sin(\psi) - \beta usin(\psi) + \beta vcos(\psi)\end{gathered} \tag{6.184}$$

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

$$\begin{gathered}z_{2,2} = L_f z_{2,1} \Rightarrow z_{2,2} = \frac{\partial z_{2,1}}{\partial x} f_1 + \frac{\partial z_{2,1}}{\partial y} f_2 + \frac{\partial z_{2,1}}{\partial \psi} f_3 + \frac{\partial z_{2,1}}{\partial u} f_4+ \\ +\frac{\partial z_{2,1}}{\partial v} f_5 + \frac{\partial z_{2,1}}{\partial r} f_6 + \frac{\partial z_{2,1}}{\partial z_1} f_7 + \frac{\partial z_{2,1}}{\partial z_2} f_8 \Rightarrow \\ z_{2,2} = 1 \cdot f_2 \Rightarrow z_{2,2} = usin(\psi) + vcos(\psi)\end{gathered} \tag{6.185}$$

Equivalently, one has

$$\begin{gathered}z_{2,3} = L_f^2 z_{2,1} \Rightarrow z_{2,3} = L_f z_{2,2} \Rightarrow \\ z_{2,3} = \frac{\partial z_{2,2}}{\partial x} f_1 + \frac{\partial z_{2,2}}{\partial y} f_2 + \frac{\partial z_{2,2}}{\partial \psi} f_3 + \frac{\partial z_{2,2}}{\partial u} f_4 + \frac{\partial z_{2,2}}{\partial v} f_5 + \frac{\partial z_{2,2}}{\partial r} f_6 + \frac{\partial z_{2,2}}{\partial z_1} f_7 + \frac{\partial z_{2,2}}{\partial z_8} f_8 \Rightarrow \\ z_{2,3} = (ucos(\psi) - vsin(\psi))r + sin(\psi)(vr + z_1) + cos(\psi)(-ur + \beta v) \Rightarrow \\ z_{2,3} = z_1 sin(\psi) + \beta vcos(\psi)\end{gathered} \tag{6.186}$$

In an equivalent manner, one obtains

$$\begin{gathered}
z_{2,4} = L_f^3 z_{2,1} \Rightarrow z_{3,3} = L_f z_{2,3} \Rightarrow \\
z_{2,4} = \tfrac{\partial z_{2,3}}{\partial x} f_1 + \tfrac{\partial z_{2,3}}{\partial y} f_2 + \tfrac{\partial z_{2,3}}{\partial \psi} f_3 + \tfrac{\partial z_{2,3}}{\partial u} f_4 + \tfrac{\partial z_{2,3}}{\partial v} f_5 + \tfrac{\partial z_{2,3}}{\partial r} f_6 + \tfrac{\partial z_{2,3}}{\partial z_1} f_7 + \tfrac{\partial z_{2,3}}{\partial z_2} f_8 \Rightarrow \\
z_{2,4} = (cos(\psi) - \beta v sin(\psi)) f_3 + \beta cos(\psi) f_5 + \sin(\psi) f_7 \Rightarrow \\
z_{2,4} = (z_1 cos(\psi) - \beta v sin(\psi)) r + (\beta cos(\psi)(-ur + \beta v) + \sin(\psi) z_2 \Rightarrow \\
z_{2,4} = z_1 cos(\psi) r - \beta v r sin(\psi) - \beta u r cos(\psi) + \beta^2 v cos(\psi) + z_2 sin(\psi)
\end{gathered} \tag{6.187}$$

Moreover, it holds that

$$\dot{z}_{2,4} = L_f^4 z_{2,1} + L_{g_a} L_f^3 z_{2,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{2,1} \tau_r \tag{6.188}$$

where

$$\begin{gathered}
L_f^4 z_{2,1} = L_f z_{2,4} \Rightarrow \\
L_f^4 z_{2,1} = \tfrac{\partial z_{2,4}}{\partial x} f_1 + \tfrac{\partial z_{2,4}}{\partial y} f_2 + \tfrac{\partial z_{2,4}}{\partial \psi} f_3 + \tfrac{\partial z_{2,4}}{\partial u} f_4 + \tfrac{\partial z_{2,4}}{\partial v} f_5 + \tfrac{\partial z_{2,4}}{\partial r} f_6 + \tfrac{\partial z_{2,4}}{\partial z_1} f_7 + \tfrac{\partial z_{2,4}}{\partial z_2} f_8
\end{gathered} \tag{6.189}$$

which gives

$$\begin{gathered}
L_f^4 z_{2,1} = [-z_1 sin(\psi) r - \beta v r cos(\psi) + \beta u r sin(\psi) - \beta^2 v sin(\psi) + z_2 cos(\psi)] r + \\
[-\beta r cos(\psi)](vr + z_1) + [-\beta r sin(\psi) + \beta^2 cos(\psi)](-ur + \beta v) + \\
[z_1 cos(\psi) - \beta v sin(\psi) - \beta u cos(\psi)]0 + [cos(\psi) r] z_2 + [sin(\psi)]0
\end{gathered} \tag{6.190}$$

and after additional computations one arrives at the form

$$\begin{gathered}
L_f^4 z_{2,1} = -z_1 r^2 sin(\psi) - \beta v r^2 cos(\psi) + \beta u r^2 sin(\psi) - \beta^2 v r sin(\psi) + z_2 r cos(\psi) - \\
-\beta v r^2 cos(\psi) - \beta r z_1 cos(\psi) + \beta u r^2 sin(\psi) - \beta^2 r v sin(\psi) - \\
-\beta^2 u r cos(\psi) + \beta^2 v cos(\psi) + z_2 r cos(\psi)
\end{gathered} \tag{6.191}$$

Proceeding as before, one computes

$$\begin{gathered}
L_{g_a} L_f^3 z_{2,1} = L_{g_a} z_{2,4} \Rightarrow \\
L_{g_a} L_f^3 z_{2,1} = \tfrac{\partial z_{2,4}}{\partial x} g_{a_1} + \tfrac{\partial z_{2,4}}{\partial y} g_{a_2} + \tfrac{\partial z_{2,4}}{\partial \psi} g_{a_3} + \tfrac{\partial z_{2,4}}{\partial u} g_{a_4} + \\
+ \tfrac{\partial z_{2,4}}{\partial v} g_{a_5} + \tfrac{\partial z_{2,4}}{\partial v} g_{a_6} + \tfrac{\partial z_{2,4}}{\partial z_1} g_{a_7} + \tfrac{\partial z_{2,4}}{\partial z_2} g_{a_8} \Rightarrow \\
L_{g_a} L_f^3 z_{2,1} = \tfrac{\partial z_{2,4}}{\partial z_2} \Rightarrow L_{g_a} L_f^3 z_{2,1} = sin(\psi)
\end{gathered} \tag{6.192}$$

Equivalently, one computes

$$\begin{gathered}
L_{g_b} L_f^3 z_{2,1} = L_{g_b} z_{2,4} \Rightarrow \\
L_{g_b} L_f^3 z_{2,1} = \tfrac{\partial z_{2,4}}{\partial x} g_{b_1} + \tfrac{\partial z_{2,4}}{\partial y} g_{b_2} + \tfrac{\partial z_{2,4}}{\partial \psi} g_{b_3} + \tfrac{\partial z_{2,4}}{\partial u} g_{b_4} + \\
+ \tfrac{\partial z_{2,4}}{\partial v} g_{b_5} + \tfrac{\partial z_{2,4}}{\partial r} g_{b_6} + \tfrac{\partial z_{2,4}}{\partial z_1} g_{b_7} + \tfrac{\partial z_{2,4}}{\partial z_2} g_{b_8} \Rightarrow \\
L_{g_b} L_f^3 z_{2,1} = \tfrac{\partial z_{2,4}}{\partial v} \Rightarrow \\
L_{g_b} L_f^3 z_{2,1} = z_1 cos(\psi) = \beta v sin(\psi) - \beta u cos(\psi)
\end{gathered} \tag{6.193}$$

The aggregate dynamics of the input-output linearized system is

$$\begin{array}{c} x^{(4)} = L_f^4 z_{1,1} + L_{g_a} L_f^3 z_{1,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{1,1} \tau_r \\ y^{(4)} = L_f^4 z_{2,1} + L_{g_a} L_f^3 z_{2,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{2,1} \tau_r \end{array} \tag{6.194}$$

By defining the new control inputs

$$\begin{array}{c} v_1 = L_f^4 z_{1,1} + L_{g_a} L_f^3 z_{1,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{1,1} \tau_r \\ v_2 = L_f^4 z_{2,1} + L_{g_a} L_f^3 z_{2,1} \ddot{\tau}_u + L_{g_b} L_f^3 z_{2,1} \tau_r \end{array} \tag{6.195}$$

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

$$\begin{array}{c} x^{(4)} = v_1 \\ y^{(4)} = v_2 \end{array} \tag{6.196}$$

which can be also written in the state-space form

$$\begin{array}{c} \dot{\tilde{z}} = A\tilde{z} + B\tilde{v} \\ \tilde{z}^m = C\tilde{z} \end{array} \tag{6.197}$$

or using that $\tilde{z} = [z_{1,1}, z_{1,2}, z_{1,3}, z_{1,4}, z_{2,1}, z_{2,2}, z_{2,3}, z_{2,4}]^T$, equivalently one has

$$\begin{pmatrix} \dot{z}_{1,1} \\ \dot{z}_{1,2} \\ \dot{z}_{1,3} \\ \dot{z}_{1,4} \\ \dot{z}_{2,1} \\ \dot{z}_{2,2} \\ \dot{z}_{2,3} \\ \dot{z}_{2,4} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_{1,1} \\ z_{1,2} \\ z_{1,3} \\ z_{1,4} \\ z_{2,1} \\ z_{2,2} \\ z_{2,3} \\ z_{2,4} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \tag{6.198}$$

while the associated measurement equation is

$$\begin{pmatrix} z_1^m \\ z_2^m \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_{1,1} \\ z_{1,2} \\ z_{1,3} \\ z_{1,4} \\ z_{2,1} \\ z_{2,2} \\ z_{2,3} \\ z_{2,4} \end{pmatrix} \tag{6.199}$$

A suitable feedback control law for the linearized system is

$$\begin{aligned} v_1 &= x_d^{(4)} - k_1^1(x^{(3)} - x_d^{(3)}) - k_2^1(\ddot{x} - \ddot{x}_d) - k_3^1(\dot{x} - \dot{x}_d) - k_4^1(x - x_d) \\ v_2 &= y_d^{(4)} - k_1^2(y^{(3)} - y_d^{(3)}) - k_2^2(\ddot{y} - \ddot{y}_d) - k_3^2(\dot{y} - \dot{y}_d) - k_4^2(y - y_d) \end{aligned} \tag{6.200}$$

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

$$\tilde{v} = \tilde{f} + \tilde{M}\tilde{u} \tag{6.201}$$

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

$$\tilde{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \; \tilde{f} = \begin{pmatrix} L_f^4 z_{1,1} \\ L_f^4 z_{2,1} \end{pmatrix} \; \tilde{M} = \begin{pmatrix} L_{g,a}L_f^3 z_{1,1} & L_{g_b}L_f^3 z_{1,1} \\ L_{g,a}L_f^3 z_{2,1} & L_{g,b}L_f^3 z_{2,1} \end{pmatrix} \; \tilde{u} = \begin{pmatrix} \ddot{\tau}_u \\ \tau_r \end{pmatrix} \tag{6.202}$$

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

$$\tilde{u} = \tilde{f} + \tilde{M}^{-1}(\tilde{v} - \tilde{f}) \tag{6.203}$$

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.

Since in the case of transformation of the nonlinear vessel dynamics to an equivalent linear form, with the use of differential flatness theory, one arrives at the state-space model of Eqs. (6.198) and (6.199). The design of the flatness-based feedback controller for the hovercraft is also given by Eqs. (6.198)–(6.203).

6.8.3 Flatness-Based Control of the Underactuated Vessel

In Chap. 2 it has been proven that the model of the underactuated vessel given in Eq. (6.166) is a differentially flat one. By exploiting differential flatness properties, the hovercraft's dynamic model was transformed into the canonical form and a state feedback controller was designed for it.

For the linearized equivalent of the system it is also possible to perform state estimation using the Derivative-free nonlinear Kalman Filter. Before computing the Kalman Filter stages, the matrices A, B, and C previously defined in Eq. (6.197) 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:

$$\begin{aligned} K(k) &= P^- C_d^T [P^- C_d^T P + R]^{-1} \\ \hat{z}(k) &= \hat{z}^-(k) - K(k)[C_d z(k) - C_d \hat{z}^-(k)] \\ P(k) &= P^-(k) - K(k) C_d P^-(k) \end{aligned} \tag{6.204}$$

time update:

$$\begin{aligned} P^-(k+1) &= A_d^T P(k) A_d + Q(k) \\ \hat{z}^-(k+1) &= A_d \hat{z}(k) + B_d u(k) \end{aligned} \tag{6.205}$$

Moreover, by using the nonlinear transformations which are provided by differential flatness theory according to Eqs. (2.88), (2.91), (2.93), and (2.98) one can obtain estimates of the state variables of the initial nonlinear hovercraft model.

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

Next, it is assumed that the input-output linearized equivalent of the system, is subjected to disturbance terms which express the effects of both modeling uncertainty and of external perturbations. Thus one has

$$\begin{aligned} x^{(4)} &= v_1 + \tilde{d}_1 \\ y^{(4)} &= v_2 + \tilde{d}_2 \end{aligned} \tag{6.206}$$

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 to be estimated with the use of the Kalman Filter, finally the dependence on knowledge of initial conditions becomes obsolete). It holds that

$$\tilde{d}_1^{(n)} = f_{d_1} \ \tilde{d}_2^{(n)} = f_{d_2} \tag{6.207}$$

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

$$z_{d,1} = \tilde{d}_1 \ z_{d,2} = \dot{\tilde{d}}_1 \ z_{d,3} = \tilde{d}_2 \ z_{d,4} = \dot{\tilde{d}}_2 \tag{6.208}$$

and the extended state-space description of the system becomes

$$\begin{pmatrix} \dot{z}_{1,1} \\ \dot{z}_{1,2} \\ \dot{z}_{1,3} \\ \dot{z}_{1,4} \\ \dot{z}_{2,1} \\ \dot{z}_{2,2} \\ \dot{z}_{2,3} \\ \dot{z}_{2,4} \\ \dot{z}_{d,1} \\ \dot{z}_{d,2} \\ \dot{z}_{d,3} \\ \dot{z}_{d,4} \end{pmatrix} = \begin{pmatrix} 0&1&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&0&0&0&0&0&0 \end{pmatrix} \begin{pmatrix} z_{1,1} \\ z_{1,2} \\ z_{1,3} \\ z_{1,4} \\ z_{2,1} \\ z_{2,2} \\ z_{2,3} \\ z_{2,4} \\ z_{d,1} \\ z_{d,2} \\ z_{d,3} \\ z_{d,4} \end{pmatrix} + \begin{pmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 0&0&0&0 \\ 1&0&0&0 \\ 0&0&0&0 \\ 0&0&0&0 \\ 0&0&0&0 \\ 0&1&0&0 \\ 0&0&0&0 \\ 0&0&1&0 \\ 0&0&0&0 \\ 0&0&0&1 \end{pmatrix} \begin{pmatrix} \ddot{\tau}_u \\ \tau_r \\ f_{d_1} \\ f_{d_2} \end{pmatrix} \tag{6.209}$$

while the associated measurement equation is

$$\begin{pmatrix} z_{1,1} \\ z_{2,1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} z_e \tag{6.210}$$

where $z_e = [z_{1,1}, z_{1,2}, z_{1,3}, z_{1,4}, z_{2,1}, z_{2,2}, z_{2,3}, z_{2,4}, z_{d,1}, z_{d,2}, z_{d,3}, z_{d,4}]^T$ is the extended state vector. Thus, the extended state-space description of the hovercraft model takes the form

$$\begin{aligned} \dot{z}_e &= A_e z_e + B_e v_e \\ z_e^{meas} &= C_e z_e \end{aligned} \tag{6.211}$$

For the extended state-space description of the system, one can design a state estimator of the form

$$\begin{aligned} \dot{\hat{z}}_e &= A_o z_e + B_o v_e + K(z_e^{meas} - C_o \hat{z}_e) \\ \hat{z}_e^{meas} &= C_o \hat{z}_e \end{aligned} \tag{6.212}$$

where for the matrices A_o and C_o it holds $A_o = A_e$ and $C_o = C_e$ while for matrix B_o one has

$$B_o^T = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.213}$$

Again the Kalman Filter recursion provides joint estimation of the nonmeasurable 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:

$$\begin{aligned} K(k) &= P_e^- C_{e_d}^T [P_e^- {C_{e_d}}^T P_e + R_e]^{-1} \\ \hat{z}_e(k) &= \hat{z}_e^-(k) - K(k)[C_{e_d} z_e(k) - C_{e_d} \hat{z}e^-(k)] \\ P_e(k) &= P_e^-(k) - K(k) C_{e_d} P_e^-(k) \end{aligned} \tag{6.214}$$

time update:

$$\begin{aligned} P_e^-(k+1) &= {A_{e_d}}^T P_e(k) A_{e_d} + Q_e(k) \\ \hat{z}_e^-(k+1) &= A_{e_d} \hat{z}_e(k) + B_{e_d} v_e(k) \end{aligned} \tag{6.215}$$

To compensate for the disturbances effects, the modified control input that is applied to the system is

$$\begin{aligned} v_1 &= x_d^{(4)} - k_1^1(x^{(3)} - x_d^{(3)}) - k_2^1(\ddot{x} - \ddot{x}_d) - k_3^1(\dot{x} - \dot{x}_d) - k_4^1(x - x_d) - \hat{z}_{d,1} \\ v_2 &= y_d^{(4)} - k_1^2(y^{(3)} - y_d^{(3)}) - k_2^2(\ddot{x} - \ddot{y}_d) - k_3^2(\dot{y} - \dot{y}_d) - k_4^2(y - y_d) - \hat{z}_{d,2} \end{aligned} \tag{6.216}$$

6.8.5 Simulation Tests

The performance of the proposed nonlinear control scheme was evaluated in the case of several reference setpoints. The associated results are presented in Figs. 6.77, 6.78, 6.79, 6.80, and 6.81. It can be observed that in all cases the nonlinear feedback controller succeeded 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. 6.77a, 6.78a, 6.79a, 6.80a, and 6.81a. The estimation of the disturbance terms that were applied to the hovercraft model are depicted in Figs. 6.77b, 6.78b, 6.79b, 6.80b and 6.81b, respectively. It can be noticed again that the proposed feedback nonlinear control scheme succeeded 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 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. 6.77, 6.78, 6.79, 6.80, and 6.81). It is noteworthy that the dynamic feedback linearization procedure which has been implemented on the hovercraft's model, results in the canonical

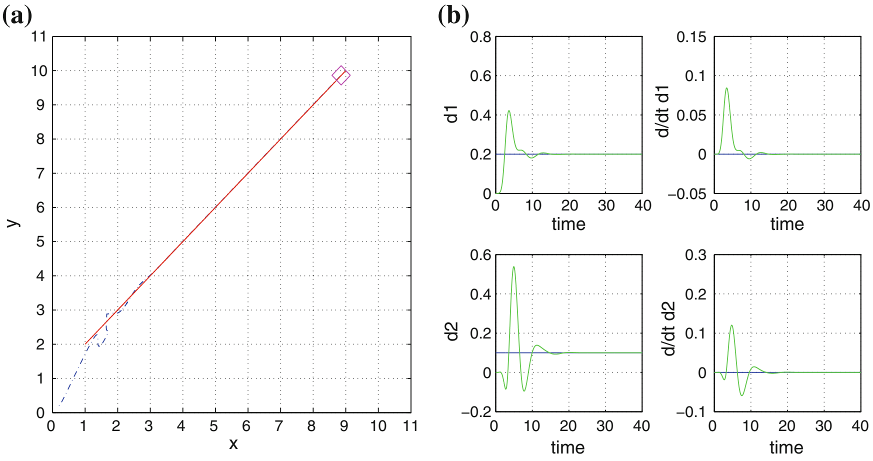


Fig. 6.77 Reference path 1. **a** Trajectory tracking for states x, y of the underactuated hovercraft. **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

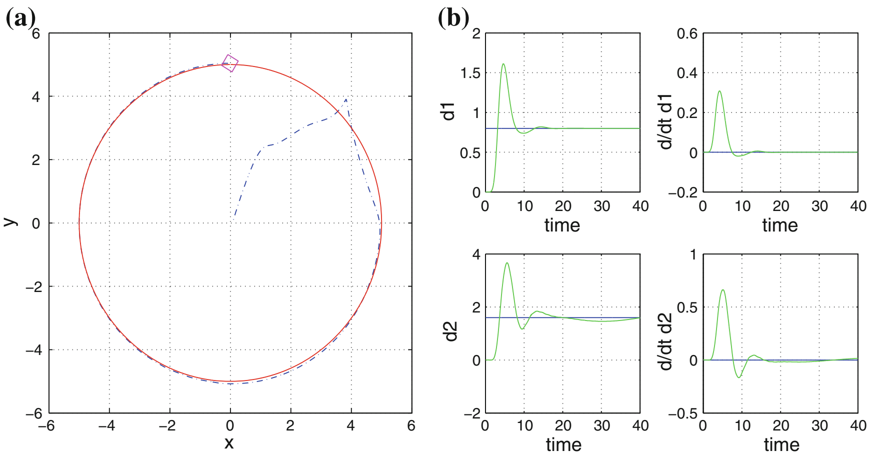


Fig. 6.78 Reference path 2. **a** Trajectory tracking for states x, y of the underactuated hovercraft. **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

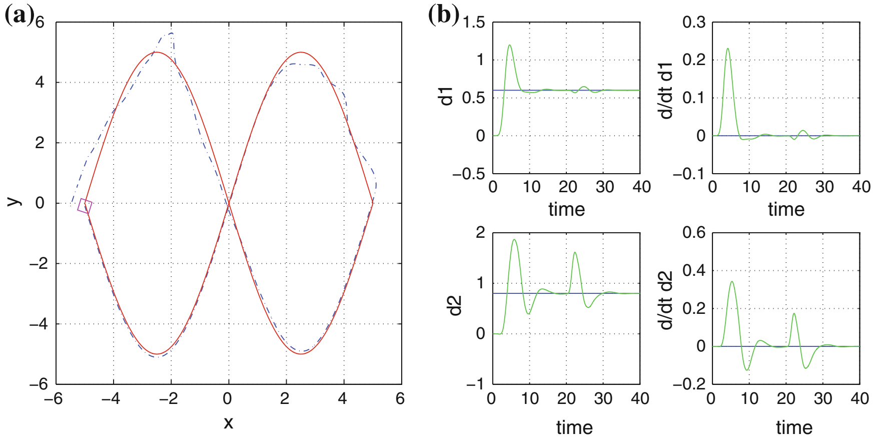


Fig. 6.79 Reference path 3. **a** Trajectory tracking for states x, y of the underactuated hovercraft. **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

form description of Eqs. (2.123) and (2.124) which is confirmed to 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 of the hovercraft is also 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

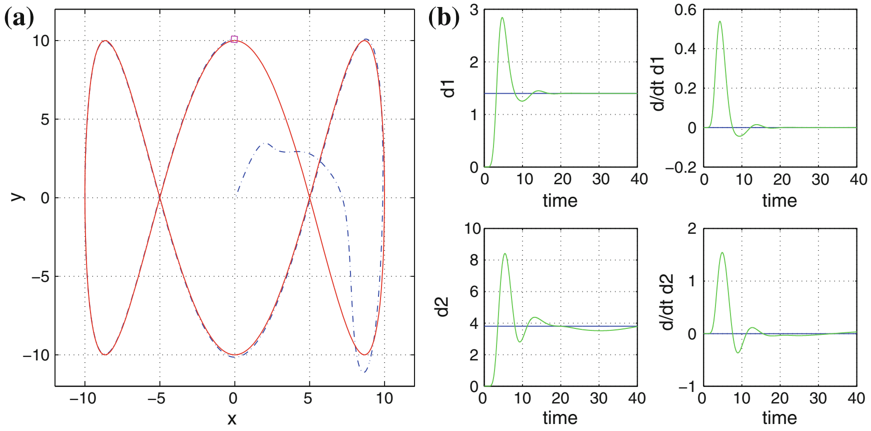


Fig. 6.80 Reference path 4. **a** Trajectory tracking for states x, y of the underactuated hovercraft. **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

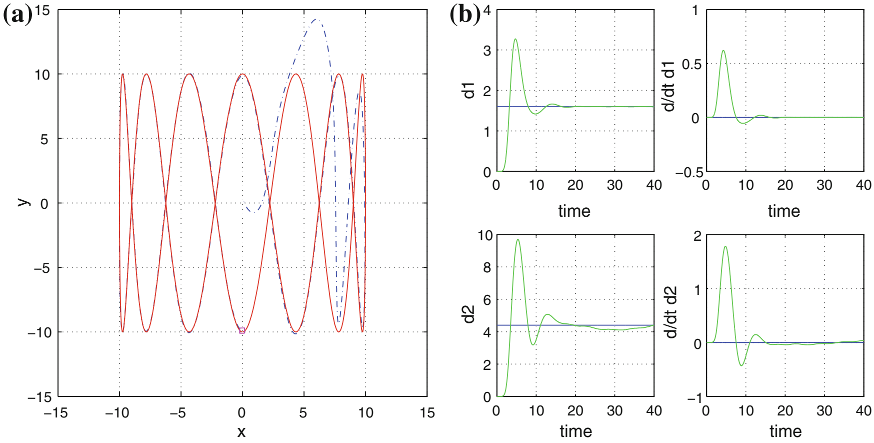


Fig. 6.81 Reference path 5. **a** Trajectory tracking for states x, y of the underactuated hovercraft. **b** Estimation of disturbance inputs affecting the hovercraft, with the use of the Derivative-free nonlinear Kalman Filter

inherent model of $g(x,t)$ its inverse never becomes 0. In such a case, the singularity problem is avoided, (ii) for certain areas of the state vector space $x \in R^n$ the zeroing of $g(x,t)^{-1}$ becomes possible. For the latter case, the avoidance of singularities can be succeeded by a state variable transformation into a new state-space representation which does not include any points of singularity.

Finally, it is noted that the presented simulation experiments have been performed under the assumption that the hovercraft was subjected 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 [129, 176, 312, 359, 482, 602]. First, it has been proven that the feedback control applied on the input-output linearized model of the hovercraft succeeded 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). Thus, stabilization can be succeeded using pole placement methods. 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 additional robustness features. 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.