International Journal of Systems Science
Vol. 37, No. 6, 15 May 2005, 399–414

Taylor & Francis
Taylor & Francis Group

Coordinated path following control of multiple wheeled robots using linearization techniques

R. GHABCHELOO*†, A. PASCOAL†, C. SILVESTRE† and I. KAMINER‡

†Institute for Systems and Robotics/Instituto Superior Técnico (IST),
Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
‡Department of Mechanical and Astronautical Engineering,
Naval Postgraduate School, Monterey, CA 93943, USA

(*Received 4 May 2005; in final form 9 December 2005*)

The paper addresses the problem of steering a fleet of wheeled robots along a set of given spatial paths, while keeping a desired inter-vehicle formation pattern. This problem arises for example when multiple vehicles are required to scan a given area in cooperation. In a possible mission scenario, one of the vehicles acts a leader and follows a path accurately, while the other vehicles follow paths that are naturally determined by the formation pattern imposed. The paper solves this and other related problems using a simple algorithm that builds on linearization techniques and gain scheduling control theory. Using this set-up, path following (in space) and inter-vehicle coordination (in time) are almost decoupled. Path following for each vehicle amounts to reducing a conveniently defined generalized error vector to zero. Vehicle coordination is achieved by adjusting the speed of each of the vehicles along its path, according to information on the position of all or some of the other vehicles. No other information is exchanged among the robots. The set-up adopted allows for a simple analysis of the resulting coordinated path following control system. The paper describes the structure of the coordination system proposed and addresses challenging problems of robustness with respect to certain types of vehicle failures.

Keywords: Coordinated motion control; Path following; Trimming trajectories; Linearization; Gain scheduling; Wheeled robots

1. Introduction

In recent years, there has been widespread interest in the problem of coordinated motion control of fleet of autonomous vehicles. Applications include aircraft and spacecraft formation flying control (Giuletti *et al*. 2000, Queiroz *et al*. 2000, Beard *et al*. 2001, Pratcher *et al*. 2001), coordinated control of land robots (Desai *et al*. 1998, Ögren *et al*. 2002), and control of multiple surface and underwater vehicles (Stilwell and Bishop 2000, Encarnação and Pascoal 2001, Skjetne *et al*. 2002, Skjetne *et al*. 2003, Lapierre *et al*. 2003a). The work reported in the literature addresses a large class of topics that include, among others, leader/follower formation flying, control of the ''center of mass'' and radius of dispersion of swarms of vehicles, and uniform coverage of an area by a group of surveying robots.

At first inspection, the problem of coordinated motion control seems to fall within the domain of decentralized control. However, as clearly pointed out in Fax and Murray (2002a, 2002b), it possesses several unique aspects that are at the root of new challenges to system designers. Among these, the following are worth stressing:

(i) except for some cases in the area of aircraft control, the motion of one vehicle does not directly affect the motion of the other vehicles, that is, the vehicles

*Corresponding author. Email: reza@isr.ist.utl.pt

International Journal of Systems Science
ISSN 0020–7721 print/ISSN 1464–5319 online © 2006 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/00207720500438324

are dynamically decoupled; the only coupling arises naturally out of the specification of the tasks that they are required to accomplish together.

(ii) there are strong practical limitations to the flow of information among vehicles, which is severely restricted by the nature of the supporting communications network. In marine robotics, for example, underwater communications rely on the propagation of acoustic waves. This sets tight limits on the communication bandwidths that are achievable. Thus, as a rule, no vehicle will be able to communicate with the entire formation Fax and Murray (2002b). Furthermore, a reliable vehicle coordination scheme should exhibit some form of robustness against certain kinds of vehicle failures or loss of inter-vehicle communications.

A rigorous methodology to deal with some of the above issues has emerged from the work reported in Fax and Murray (2002a, 2002b), which addresses explicitly the topics of information flow and cooperation control of vehicle formations simultaneously. The methodology proposed builds on an elegant framework that involves the concept of Graph Laplacian (a matrix representation of the graph associated with a given communication network). In particular, the results in Fax and Murray (2002b) show clearly how the Graph Laplacian associated with a given inter-vehicle communication network plays a key role in assessing stability of the behaviour of the vehicles in a formation. It is however important to point out in that work that: (i) the dynamics of the vehicles are assumed to be linear and time-invariant and identical for all the vehicles, and (ii) the information exchanged among vehicles is restricted to linear combinations of the vehicle state variables.

Inspired by the progress in the field, this paper tackles a problem in coordinated vehicle control that departs slightly from mainstream work reported in the literature. Specifically, we consider the problem of *coordinated path following* where *multiple vehicles are required to follow pre-specified spatial paths while keeping a desired inter-vehicle formation pattern in time*. This mission scenario occurs naturally in underwater robotics (Pascoal *et al*. 2000): namely, in the operation of multiple autonomous underwater vehicles for fast acoustic coverage of the seabed. In this important case, two or more vehicles are required to fly above the seabed at the same or different depths, along geometrically similar spatial paths, and map the seabed using copies of the same suite of acoustic sensors. By requesting that the vehicles traverse identical paths so as to make the acoustic beam coverage overlap along the seabed, large areas can be covered in a short time. This imposes constraints on the inter-vehicle formation pattern. Similar scenarios can of course be envisioned for land and air vehicles.

To the best of our knowledge, previous work on coordinated path following control has been restricted to the area of marine robotics. See for example Lapierre *et al*. (2003a, 2003b) and the references therein. However, the solutions developed so far for underactuated vehicles are restricted to two vehicles in a leader-follower type of formation and lead to complex control laws. There is therefore a need to re-examine this problem, to try and arrive at efficient and practical solutions. In an attempt to do this, the present paper considers a similar problem for wheeled robots in the hope that the solution derived for this simpler case will shed some light into the problem of coordinated path following for the more complex case of air and marine robots. Different cooperation strategies are considered that go beyond the commonly adopted leader-follower structure.

The methodology for coordinated path following of wheeled robots proposed in this paper builds on previous work on path following and trajectory tracking for air and marine vehicles using linearization techniques and gain scheduling control theory. See (Kaminer *et al*. 1998, Silvestre 2000, Silvestre *et al*. 2002) and the references therein. The circle of ideas explored in these references can be readily transposed to wheeled robots to yield the following results: (i) the trimming (equilibrium) paths of wheeled robots are circumferences parameterized by the vehicle's linear speed and yaw rate, the circles degenerating into straight lines when the yaw rate is zero, (ii) tracking of a trimming path by a vehicle is equivalent to driving a conveniently defined generalized tracking error to zero, and (iii) the linearization of the so-called generalized error dynamics about any trimming trajectory is time invariant. Based on these results, the problem of integrated design of guidance and control systems for accurate tracking of paths that consist of the union of trimming paths can be cast in the framework of gain scheduled control theory. In this context, the vehicle's linear speed and yaw rate play the role of scheduling variables that interpolate the parameters of linear controllers designed for a finite number of representative trimming paths. Guidelines for path following control system design and implementation follow from the work of (Kaminer *et al*. 1995), where it is shown how a simple implementation strategy, referred to as the $\mathcal{D}$-implementation, avoids feedforwarding the values of state and control inputs at trimming.

Using this set-up, *path following and inter-vehicle coordination (in time) are essentially decoupled*. Path following for each vehicle amounts to reducing a conveniently defined error vector to zero. Vehicle coordination is achieved by adjusting the speed of each of

the vehicles along its path, according to information on the position of some of the other vehicles. No other information is exchanged among the robots. The set-up adopted allows for a simple analysis of the resulting coordinated path following control system. The paper describes the structure of the coordination system proposed and addresses challenging problems of robustness with respect to vehicle failures. The coordination system is simple and holds great potential to be extended and applied to the case of air and marine robots.

The paper is organized as follows. Section 2 introduces some basic notation and the dynamic model of a wheeled robot, defines the notion of trimming paths, and presents two possible solutions to the problem of path following for a single vehicle. Section 3 introduces a simple strategy for multiple vehicle cooperation. The stability of the combined path following and coordination algorithms is analyzed in section 4. Section 5 contains the results of simulations that illustrate the performance of the coordinated path following system proposed. Section 6 is an introduction to the problem of robustness of the coordinated control scheme proposed with respect to single vehicle failures. Finally, section 7 contains the main conclusions and describes problems that warrant further research.

2. Basic notation. Path following

This section introduces some basic notation and offers two solutions to the problem of path following for a single vehicle.

Consider the wheeled robot of the unicycle type shown in figure 1, together with a spatial path to be followed. The vehicle has two identical parallel, nondeformable rear wheels. It is assumed that the plane of each wheel is perpendicular to the ground and that the contact between the wheels and the ground is pure rolling and nonslipping, i.e., the velocity of the center of mass of the robot is orthogonal to the rear wheels axis. Each rear wheel is powered by a motor which generates a control torque. This will in turn generate a control force and torque applied to the vehicle.

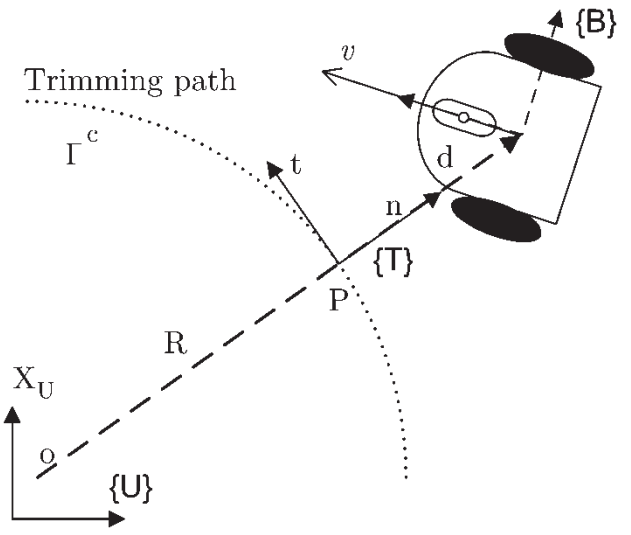


Figure 1. Frames and error variables.

The following notation will be used in the sequel. The symbol $\{A\} := \{x_A, y_A, z_A\}$ denotes a reference frame with origin at O_A and unit vectors x_A, y_A, z_A. Often, for simplicity of presentation, we omit writing explicitly the third component of the reference frames, because the wheeled robot is restricted to move in the horizontal plane. Let {U} and {B} be inertial and body-fixed reference frames, respectively and assume that the origin O_B of {B} is coincident with the center of mass of the vehicle. Further let $[x, y]^T$ denote the position of O_B in {U} and ψ the parameter that describes the orientation of {B} with respect to {U} (i.e., the robot's orientation with respect to the inertial x-axis). Define v and r as the linear and rotational velocities of {B} with respect to {U}, respectively expressed in {B}. With the above notation, the simplified kinematic and dynamic equations of the wheeled robot can be written as

$$Dynamics \begin{cases} \dot{v} = F/m \\ \dot{r} = N/I \end{cases} \tag{1}$$

$$Kinematics \begin{cases} \dot{\psi} = r \\ \dot{x} = v\cos(\psi) \\ \dot{y} = v\sin(\psi) \end{cases} \tag{2}$$

where m denotes the mass of the robot, I is the moment of inertia about its vertical body-axis, and F and N denote the total force and torque respectively, applied to the vehicle. Define

$$x_1 = [v, r]^T; \quad x_2 = [\psi, x, y]^T; \quad u = [F, N]^T \tag{3}$$

and assume that $m = I = 1$ in the appropriate units. Equations (1) and (2) can obviously be cast in the general state space form

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, u) \\ \dot{x}_2 &= f_2(x_1, x_2), \end{aligned} \tag{4}$$

where f_1, f_2 are nonlinear functions of their arguments and x_1 and x_2 represent the dynamic and kinematic states, respectively. Even though the simple dynamical model of the wheeled robot does not require that f_1 be a function of x_1, it is convenient to adopt

the general state space form because it allows for the inclusion of dissipative (velocity dependent) terms if needed. Notice that the evolution of x_1 does not depend on x_2.

Following Silvestre (2000) and Silvestre *et al.* (2002) a trimming trajectory of the wheeled robot is a set

$$\Upsilon^c := \{(x_1^c, x_2^c(\cdot), u^c) : f_1(x_1^c, u^c) = 0\} \tag{5}$$

parameterized by the 2-tuple of vectors x_1^c, u^c that make $f_1(x_1, u) = 0$. The notation $x_2^c(\cdot)$ represents the corresponding time-history of state variable x_2 at trimming. Stated in simple terms, along a trimming (also called equilibrium) trajectory, the input u is held fixed, and the dynamic variables remain constant ($\dot{x}_1 = 0$). Notice, however that the kinematic variables x_2 are allowed to be functions of time.

In the case of the wheeled robot, it is trivial to show that the only possible trimming trajectories correspond to circumferences and straight lines. In other words, with u set to a constant value u^c, v and r assume constant values v^c and r^c respectively, and the $x_2^c(\cdot)$ component of Υ^c is such that $\psi^c(t) = r^c t + \psi_0$, where ψ_0 denotes the heading angle at time $t = 0$. Straightforward computations show that in this case the origin O_B of the wheeled robot is driven along a circumference with radius $R = |v^c/r^c|$, where $|\cdot|$ stands for the absolute value. The circle degenerates into a straight line when $r^c = 0$.

Notice in (5) that a trimming trajectory is specified in terms of all state and input variables at trimming. However, given fixed values v^c and r^c, the corresponding input u^c and the state $x_2^c(\cdot)$ are, apart from the initial conditions, uniquely determined. In this sense, v^c and r^c determine uniquely the values of $[x^c, y^c]^T$, and thus of the corresponding path (curve in space) Γ^c traversed by the vehicle. Formally,

$$\Gamma^c := \{\Pi_p x_2^c(\cdot) : (x_1^c, x_2^c(\cdot), u^c) \in \Upsilon^c\} \tag{6}$$

where $\Pi_p : \mathbb{R}^3 \to \mathbb{R}^2$ denotes the operator that extracts the last two components of $x_2^c(\cdot)$. Clearly, a trimming path is simply obtained from a trimming trajectory by keeping the 2-D vector corresponding to the position of the wheeled robot. From the above discussion, and with a slight abuse of language, it can be stated that $\Gamma^c = \Gamma^c(v^c, r^c)$, that is, a trimming path is uniquely determined by the trimming values v^c and r^c or, equivalently, by v^c and c_c, where c_c denotes the path curvature. In what follows it is assumed that Γ^c can be parameterized in some convenient geometric manner, for example in terms of its curvilinear abcissa s (length along the path).

In the sequel, we consider the case where the wheeled robot is required to follow a general path that consists of the union of trimming paths. The emphasis is therefore placed on the development of controllers for accurate following of trimming paths. Consider now figure 1, and suppose it is required for the wheeled robot to follow the trimming path Γ^c, that is, for O_B to converge to and follow the 2-D curve Γ^c at constant linear and rotational speeds v^c and r^c, respectively. A solution to this problem can be easily obtained by recalling the work of Micaelli and Samson (1992, 1993) on path following, from which the following intuitive explanation is obtained: a path following controller should look at i) the distance from the vehicle to the path and ii) the angle between the vehicle velocity vector and the tangent to the path, and reduce both to zero. This motivates the development of the kinematic model of the vehicle in terms of a Serret-Frenet frame {T} that moves along the path; {T} plays the role of the body axis of a "virtual target vehicle" that should be tracked by the "real vehicle". Using this set-up, the abovementioned distance and angle become part of the coordinates of the error space where the path following control problem is formulated and solved.

Formally, given O_B assume that the closest point P on the path is well defined and consider the Serret-Frenet frame $\{T\} := \{t, n\}$ with its origin at P. As is well known, t and n are the tangent and normal to the curve at P, respectively, where the positive direction of t is defined by traversing the path along increasing values of its length s. Let d_e be the y-component of vector d from P to O_B, expressed in {T} (the x-component is zero). Clearly, d is colinear with unit vector n. Further let $\psi_T = \psi^c$ parameterize the rotation matrix from {T} to {U}, satisfying the relation

$$\dot{\psi}_T = c_c \dot{s} \tag{7}$$

where c_c is the curvature of the path and $\dot{s}$ denotes the time derivative of the curvilinear abcissa s of P. Since there is no slippage, $\{B\} = \{T\}$ at trimming.

Equipped with this notation, we now derive two algorithms for path following by resorting to linearization techniques. See Kaminer *et al.* (1998), Silvestre (2000), Silvestre *et al.* (2002) for an introduction to these techniques and for their application to path following control of air and marine robots. The two algorithms build on two different error coordinates and will henceforth be referred to as the *Decoupling* and the *State Transformation* algorithm.

2.1. *The decoupling algorithm*

Given a trimming path, consider the path-following error coordinates

$$\begin{aligned} v_e &= v - v^c \\ r_e &= r - r^c \\ d_e &= \Pi\, d \\ \psi_e &= \psi - \psi^c = \psi - \psi_T, \end{aligned} \tag{8}$$

where $\Pi = [\,0 \quad 1\,]$. Convergence of a vehicle to the path is equivalent to driving the above error variables to zero. Notice that with the simplified wheeled robot model, $F^c = N^c = 0$. Following the methodology exposed in Micaelli and Samson (1993), straightforward computations show that the error dynamics can be written as

$$\begin{aligned} \dot{v}_e &= F \\ \dot{r}_e &= N \\ \dot{d}_e &= v \sin(\psi_e) \\ \dot{\psi}_e &= r - \frac{c_c v \cos(\psi_e)}{1 - d_e c_c}. \end{aligned} \tag{9}$$

Furthermore, the evolution of the closest point on the path is easily seen to be given by

$$\dot{s} = \frac{v \cos(\psi_e)}{1 - d_e c_c}. \tag{10}$$

It is important to point out that equations (9) and (10) are only valid when $d_e c_c < 1$, that is, when the vehicle is "sufficiently close" to the path.

At this point, it is important to examine the equations above. Notice that the first equation in (9) is completely independent of the remaining ones. This means that the forward speed v of the vehicle can be manipulated at will by manipulating F, no matter what the evolution of the variables r_e, d_e, and ψ_e is. In particular, when following a desired trimming path *without any inter-vehicle coordination requirements*, the forward speed is simply set to the trimming value v_c. It is then up to the path following controller to manipulate the torque N so as to drive r_e, d_e, and ψ_e to zero.

This simple circle of ideas is at the core of the technique of coordinated path following proposed in this paper: for each vehicle in the formation, the torque N is computed so as to achieve path following for a given set of possible forward speeds and trimming paths, while F controls the forward speed of the vehicle in order to meet the required inter-vehicle formation requirements.

The first step in the decoupling approach to path following is to linearize the error dynamics about trimming conditions to obtain

$$\begin{aligned} \delta\dot{v}_e &= \delta F \\ \delta\dot{r}_e &= \delta N \\ \delta\dot{d}_e &= v^c \delta\psi_e \\ \delta\dot{\psi}_e &= \delta r_e - c_c^2 v^c \delta d_e - c_c \delta v_e. \end{aligned} \tag{11}$$

The resulting system is time-invariant, as proven in Silvestre (2000) for a more general class of systems. An important assumption is made at this point: since the variable v_e is controlled independently (to meet the formation requirements), δv_e is simply viewed as a vanishing perturbation and thus ignored in the design of a path following controller that will drive r_e, d_e, and ψ_e to zero. This assumption will be re-visited and proved rigorously later in the paper. In the case of pure path following about a trimming path (that is, without any formation requirements), δv_e is naturally set to zero. In what follows, it is assumed that the curvature c_c is upper bounded. Ignoring the first independent equation in (11) yields the sub-system

$$\begin{aligned} \delta\dot{r}_e &= \delta N \\ \delta\dot{d}_e &= v^c \delta\psi_e \\ \delta\dot{\psi}_e &= \delta r_e - c_c^2 v^c \delta d_e, \end{aligned} \tag{12}$$

for which a stabilizing controller is sought. Notice in the equations the explicit dependence of the dynamics on the path curvature c_c and trimming forward speed $v^c = r^c/c_c$. It is thus natural that the resulting controllers show dependence, that is, be scheduled on the same variables. As is customary in gain scheduling control, the scheduling is done on the *actual values of the variables*, that is, on c_c and r/c_c. In what follows the $\mathcal{D}$-methodology introduced in Kaminer *et al.* (1995) for the design and implementation of gain scheduled controllers is adopted. See also Khalil (2000), Chapter 10. The $\mathcal{D}$-methodology addresses explicitly the problem of controller implementation on the original nonlinear plant and avoids feed-forwarding the values of the relevant variables at trimming. As in Kaminer *et al.* (1995), append to the original system (12) an extra state z, defined by

$$\dot{z} = \delta d_e, \tag{13}$$

aimed at driving the steady state of δd_e to zero (in general, one should include as many integrators as the number of control signals). Since the linearized system

with input δN and state $[\delta r_e, \delta d_e, \delta \psi_e, z]^T$ is controllable, arbitrary closed loop eigenvalue placement can be achieved with the state feedback control law

$$\delta N = -k_1 \delta r_e - k_2 \delta d_e - k_3 \delta \psi_e - k_4 z, \tag{14}$$

yielding the closed-loop characteristic polynomial

$$\lambda^4 + k_1\lambda^3 + (k_2 + (c_c v^c)^2)\lambda^2 + v^c(k_2 + k_1 c_c^2 v^c)\lambda + k_4 v^c. \tag{15}$$

Without loss of generality, and for simplicity of exposition, select the desired values of the closed loop eigenvalues to be coincident and equal to $-\lambda_p \,\mathrm{rad\,s^{-1}}; \lambda_p > 0$. This can be done with the state feedback gains

$$\begin{aligned} k_1 &= 4\lambda_p \\ k_2 &= \frac{4\lambda_p^3}{v^c} - 4\lambda_p c_c^2 v^c \\ k_3 &= 6\lambda_p^2 - (c_c v^c)^2 \\ k_4 &= \frac{\lambda_p^4}{v^c} \end{aligned} \tag{16}$$

that show clearly the dependence on the trimming values of c_c and v^c. Obviously, the gains can also be defined in terms of c_c and r^c/c_c. For implementation purposes, the actual values of c_c and r/c_c are used. Figure 2 shows the final implementation of the gain scheduled controller on the nonlinear plant, using the $\mathcal{D}$-methodology (Kaminer *et al.* 1995).

Notice how the integrator was moved in front of the plant and derivative operators were introduced at the appropriate variables. As explained in Kaminer *et al.* (1995), this procedure does not introduce any unstable pole-zero cancelations. In practice, the derivative operator is approximated by $s/(s\tau + 1)$, with τ sufficiently small. With the implementation proposed, there is no need to introduce trimming values for any of the dynamics variables, namely for the angular velocity r. The importance of this property can hardly be overemphasized. In fact, given a desired path with a known curvature c_c and given an arbitrary translational velocity v of the vehicle, the current scheme will make the vehicle converge to the path and acquire the correct rotational speed so as to follow the path with the desired radius. This is done without knowing the translational velocity explicitly. As explained later, this property is extremely important for coordinated path following because we do not require that all vehicles know the required values of their trimming speeds. The scheme also avoids feedforwarding the trimming value for the input F and can thus cope with velocity dependent friction forces not taken into account in the simplified design model introduced above.

In the scheme proposed, d_e can be easily computed for circumferences and straight lines as follows. Let $(\bar{x}_0, \bar{y}_0, R)$ denote a circumference with center $[\bar{x}_0, \bar{y}_0]^T$ and radius R, and let (x_0, y_0, m) be a straight line with slope m, passing through a point with coordinates $[x_0, y_0]$. Further let $[x, y]^T$ denote the position of the center of mass of the robot. Then,

$$\text{Line: } d_e = \begin{cases} \dfrac{y - mx - (y_0 - mx_0)}{\sqrt{1+m^2}}; & m < \infty \\ -x + x_0; & \text{otherwise} \end{cases}$$

$$\text{Circ.: } d_e = R - \sqrt{(x - \bar{x}_0)^2 + (y - \bar{y}_0)^2}$$

where the correct sign must be chosen with respect to the normal axis of the tangent frame.

A solution to the path following problem that avoids feeding back all state variables can also be obtained using output feedback. Since the system with input δN and output $[\delta r_e, \delta d_e, z]^T$ is controllable and observable, a controller $K(s)$ can be designed (using any of the methods available in the literature)

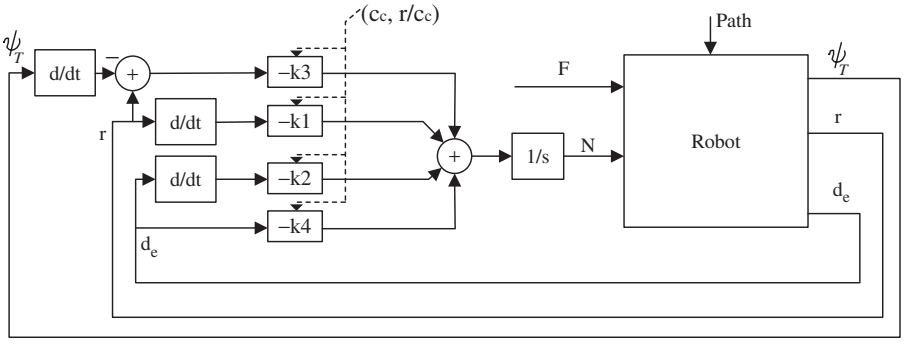


Figure 2. Gain Scheduled Controller Implementation using the $\mathcal{D}$-methodology – state feedback.

and scheduled on c_c and r/c_c. As an example, an output feedback controller was designed that requires measurements of d_e and r_e only, that is, ψ_e is not measured.

Figure 3 shows the final implementation of the gain scheduled controller on the nonlinear plant, using the $\mathcal{D}$-methodology. The figure shows clearly how easy the implementation of the gain scheduled controller is. Figure 4 illustrates the behaviour of a wheeled robot following a circular path at velocities 0.3, 0.4 and $0.5\,\mathrm{m\,s^{-1}}$, with an output feedback controller designed for $v^c = 0.4\,\mathrm{m\,s^{-1}}$. As expected, the robot "learns" the required rotational speeds automatically.

2.2. *State transformation*

The decoupling methodology for path following is naturally suited to deal with the case where the forward speed of the robot is held constant at a given trimming speed. An alternative scheme that can deal easily with speed variations about a given trimming value requires the introduction of a new variable $\eta = r - c_c v$ that equals zero at trimming, that is, $\eta_c = 0$.

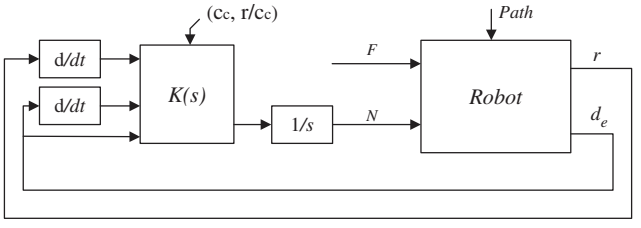


Figure 3. Gain Scheduled Controller Implementation using the $\mathcal{D}$-methodology – output feedback case.

To this effect, define the error variables

$$\begin{aligned} v_e &= v - v^c \\ \eta_e &= \eta - \eta_c = r - c_c v \\ d_e &= \Pi\, d \\ \psi_e &= \psi - \psi^c = \psi - \psi_T. \end{aligned} \tag{17}$$

The new error space differs from the previous one in the equation for η_e only. By defining a new control variable

$$u_e = N - c_c F,$$

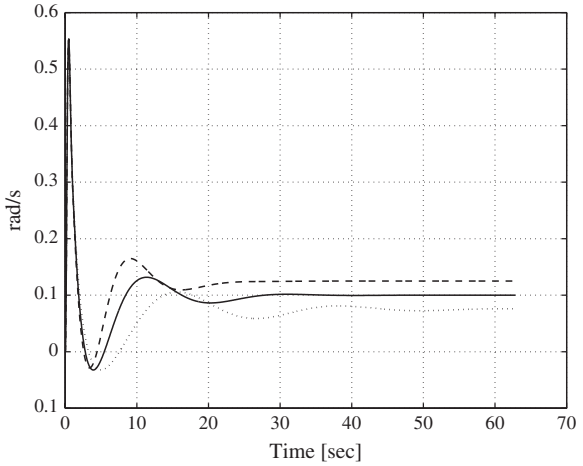
the corresponding error dynamics become

$$\begin{aligned} \dot{v}_e &= F \\ \dot{\eta}_e &= u_e \\ \dot{d}_e &= v \sin \psi_e \\ \dot{\psi}_e &= \eta_e + c_c v - \frac{c_c v \cos \psi_e}{1 - d_e c_c}. \end{aligned}$$

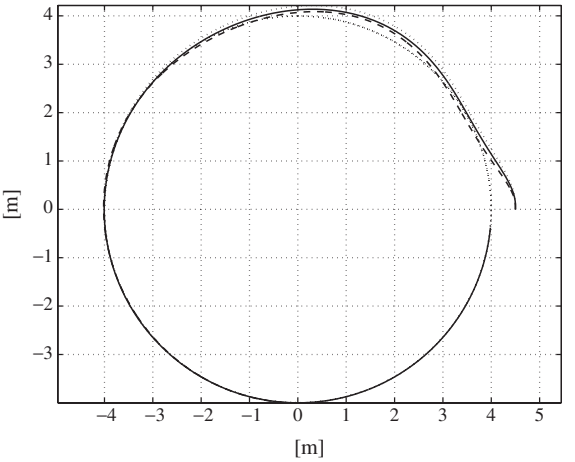
Linearizing the above equations yields

$$\begin{aligned} \delta\dot{v}_e &= \delta F \\ \delta\dot{\eta}_e &= \delta u_e \\ \delta\dot{d}_e &= v^c \delta\psi_e \\ \delta\dot{\psi}_e &= \delta r_e - c_c^2 v^c \delta d_e. \end{aligned} \tag{18}$$

Equation (18) shows that the dynamics of δv_e have become uncoupled from those of $\delta\eta_e$, δd_e, and $\delta\psi_e$. This is in contrast with the previous methodology, where δv_e was a coupling variable that was not taken into consideration during the design phase. As will be



(a) Angular speeds



(b) Spatial paths

Figure 4. Path following of the robot for $v^c = 0.3$, 0.4 and $0.5\,\mathrm{ms^{-1}}$.

seen, this uncoupling will render the proof of stability of the overall coordinated path following system simple. The price paid is the added complexity of the path following control law adopted.

In what follows, consider equation (18) with the dynamics of δv_e deleted. The relevant equations resemble those in (12) with δN replaced by δu_e. Notice the important fact that *all relevant state variables and input equal zero at trimming*. At this point, a simple state-feedback or an output-feedback path following controller can be designed to drive η_e, d_e and ψ_e to zero. The design procedure follows closely that adopted for the decoupling method. For example, appending an integrator to δd_e and choosing the state feedback control law

$$\delta u_e = -k_1 \delta\eta_e - k_2 \delta d_e - k_3 \delta\psi_e - k_4 z \qquad (19)$$

with $k_i;\ i = 1, \ldots, 4$ scheduled on c_c and r/c_c as in (16) places all the closed loop eigenvalues at $-\lambda_p \,\mathrm{rad\,s^{-1}}$. An output gain-scheduled feedback control law can also be designed and implemented as shown in figure 5. Notice that the implementation does not require the use of the $\mathcal{D}$-methodology because the trimming values of all relevant state and input variables are zero.

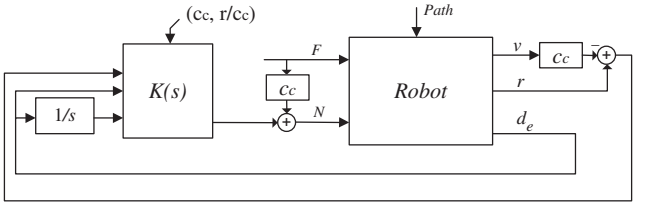


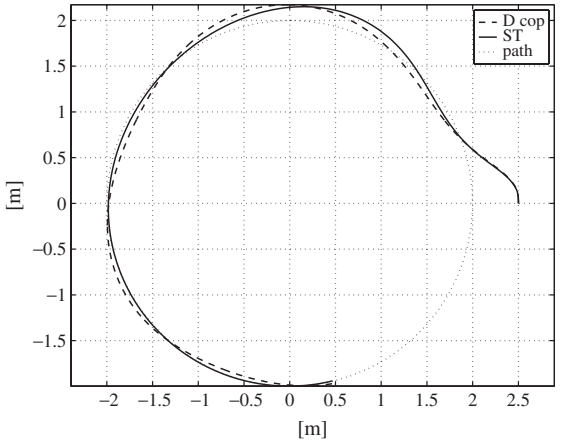
Figure 5. Controller implementation, state transformation method.

A quick comparison of the two methods shows that the state transformation strategy has the advantage of decoupling the velocity equation from the other variables.

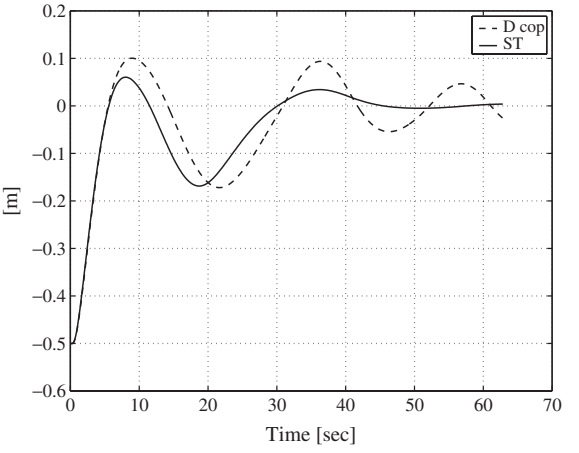
Figure 6 shows the results of simulations aimed at illustrating the performance of the two methodologies for path following. In the simulations, it was required that a wheeled robot follow a circumference with a given radius while the forward speed undergoes variation imposed by a periodic square signal in F. Notice that the state transformation methodology eliminates completely the variations in forward speed. As will be seen later, this allows for a very simple proof of stability of the complete coordinated path following system. The downside of the method is that it requires (at a path following level) knowledge of the forward speed v and force F. This is totally avoided in the decoupling method. The latter seems at first inspection to be far worse than the state transformation method because it cannot, at the path following level, eliminate the variations in forward speed about a trimming value. However, it will be later proved that the deviation in speed goes to zero at steady state, and this is the reason why the decoupling method works. The advantage of the decoupling method is the fact that no variables related to synchronization must be used. In summary, the state transformation method yields better performance at the price of increased controller complexity. It is up to the designer to decide what method to use depending on practical considerations.

3. Coordination

We now consider the problem of coordinated path following control that is the main focus of the



(a) Spatial paths



(b) Path following error, d_e

Figure 6. Performance comparison between the decoupling and state transformation methods.

present paper. In the most general set-up, one is given a set of $n \geq 2$ wheeled robots and a set of n trimming paths Γ_k; $k = 1, 2, \ldots, n$ and require that robot i follow path Γ_i. We further require that the vehicles traverse the paths in such a way as to maintain a desired formation pattern compatible with those paths. The speeds at which the robots are required to travel can be imposed in a number of ways; for example, by nominating one of the robots as a formation coordinator, assigning it a desired speed, and having the other robots adjust their speeds accordingly. Figure 7 shows the particular cases where 3 vehicles are required to follow trimming paths Γ_i; $i = 1, 2, 3$ while keeping a desired "triangle" or "in-line" formation pattern.

We assume each path is parameterized in terms of a single parameter s_i (e.g. its curvilinear abscissa, as measured from some adequately chosen point on the path). In the simplest case, the paths Γ_i may be obtained as simple translations of a "template" path Γ^t (figure 7(a)). We also consider the case of scaled circumferences with a common center and different radii R_i (figure 7(b)). In this paper, for simplicity of presentation, we restrict ourselves to "in-line" formation patterns.

Assuming that path following controllers have been implemented separately for each robot, it now remains to synchronize these in time so as to achieve the desired formation pattern. As will become clear, this will be achieved by adjusting the speeds v_i of the robots as a function of the "along-path" distances between them. Formally, we define the distances between vehicles i and j as

$$s_{i,j} = s_i - s_j; \quad i, j = 1, \ldots, n; \quad i \neq j \tag{20}$$

in the case of shifted straight lines, and as

$$\bar{s}_{i,j} = \bar{s}_i - \bar{s}_j; \quad i, j = 1, \ldots, n; \quad i \neq j, \tag{21}$$

with $\bar{s}_i = s_i / R_i$, in the case of scaled circumferences. See figure 8. Notice that the definition of $\bar{s}_{i,j}$ relies on a normalization of the lengths of the circumferences involved and is equivalent to computing the angle

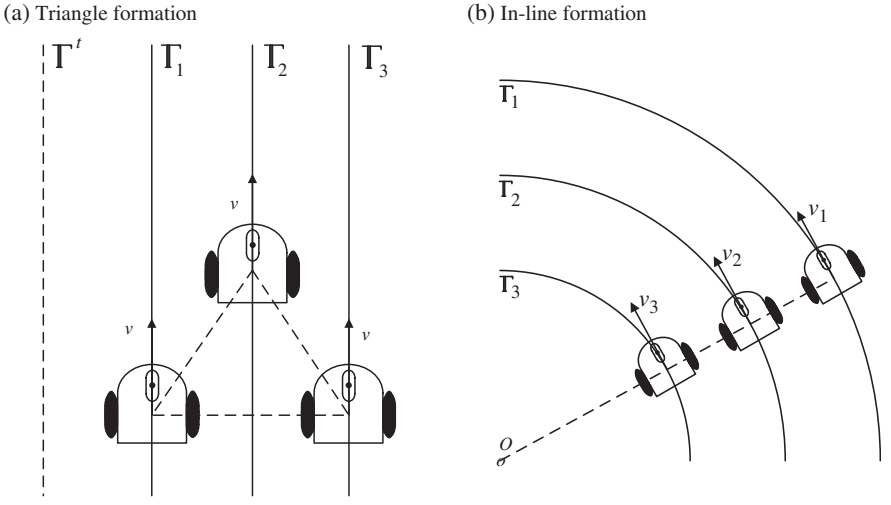


Figure 7. Coordination.

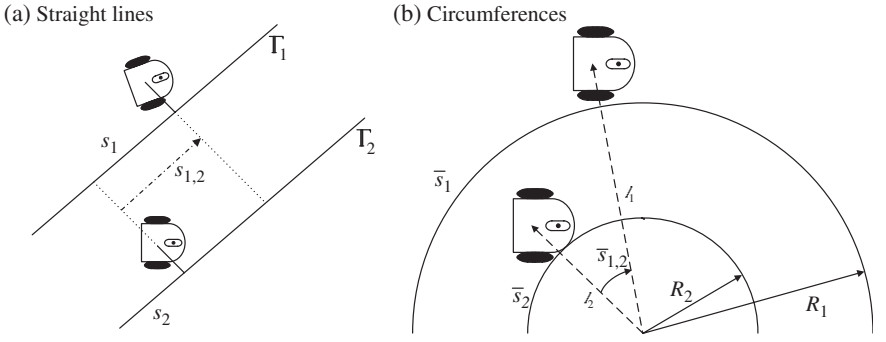


Figure 8. Along-path distances.

between vectors l_i and l_j directed from the center of the circumferences to vehicles i and j, respectively.

Given the positions $[x_i, y_i]^T$ and $[x_j, y_j]^T$ of robots i and j, respectively it is trivial to compute that

$$\text{Line: } s_{i,j} = \begin{cases} \dfrac{(x_i - x_j) + m(y_i - y_j)}{\sqrt{1+m^2}}; & m < \infty \\ y_i - y_j; & \text{otherwise} \end{cases}$$

$$\text{Circ.: } \bar{s}_{i,j} = \text{atan2}(y_i - \bar{y}_0, x_i - \bar{x}_0) - \text{atan2}(y_j - \bar{y}_0, x_j - \bar{x}_0).$$

To compute the evolution of the along-path distance, use (10), (21) to obtain

$$\dot{\bar{s}}_{i,j} = \frac{v_i \cos(\psi_{e_i})}{R_i(1 - d_{e_i} c_{c_i})} - \frac{v_j \cos(\psi_{e_j})}{R_j(1 - d_{e_j} c_{c_j})}, \tag{22}$$

which degenerates into

$$\dot{s}_{i,j} = v_i \cos(\psi_{e_i}) - v_j \cos(\psi_{e_j}) \tag{23}$$

for straight lines. In the case of circumferences $c_{c_k} = \pm 1/R_k$, the sign depending on the direction of motion. Clearly, the objective is to drive $\bar{s}_{ij}$ (or $s_{i,j}$) to zero by manipulating v_i; $i = 1, 2, \ldots, n$ about their trimming values v_i^c. Notice that in the case of "in-line" coordinated path following the along path distances are zero at trimming, that is, $s_{i,j}^c = \bar{s}_{i,j}^c = 0$.

Adopting a set-up similar to the one used for path following control, one is naturally led to consider the coordination system that is obtained by linearizing the relevant dynamics equations about trimming, that is,

$$\begin{aligned} \delta\dot{v}_{e_i} &= \delta F_i \\ \delta\dot{s}_{i,j} &= \delta v_{e_i} - \delta v_{e_j} \end{aligned}$$

for straight lines and

$$\begin{aligned} \delta\dot{v}_{e_i} &= \delta F_i \\ \delta\dot{\bar{s}}_{i,j} &= \frac{\delta v_{e_i}}{R_i} - \frac{\delta v_{e_j}}{R_j} + f_i(\delta d_{e_i}) - f_j(\delta d_{e_j}) \end{aligned} \tag{24}$$

for circumferences, where

$$f_k(\delta d_{e_k}) = \frac{v_k^c \,\text{sign}(c_{c_k})}{R_k^2} \delta d_{e_k}. \tag{25}$$

In the equations above, there are disturbance-like terms $f_k(\delta d_{e_k})$ that come from the path-following system. At this point, it is assumed that these disturbances tend asymptotically to zero. As will be seen later, this fact is trivial to prove in the case of the state-transformation methodology for path following. A similar conclusion will also be derived for the decoupling strategy. As a consequence of this assumption, the relevant linearized equations for circumferences reduce to

$$\begin{aligned} \delta\dot{v}_{e_i} &= \delta F_i \\ \delta\dot{\bar{s}}_{i,j} &= \frac{\delta v_{e_i}}{R_i} - \frac{\delta v_{e_j}}{R_j}. \end{aligned} \tag{26}$$

In the sequel, we analyze the case of *coordinated path following for circumferences, the results carrying over in an obvious manner to straight lines*.

The final step in the design of a coordination controller for (26) is to seek a general control law of the form

$$\delta F_i = g_i(\delta v_{e_i}, \bar{s}_{i,j}); \quad j \neq i; \quad j \in J_i$$

such that $\bar{s}_i$ is driven to zero asymptotically. In the above equation, J_i denotes the set of vehicles that vehicle i communicates with. It is then straightforward to implement the controller in a nonlinear setting by once again exploiting the results available in (Kaminer *et al*. 1995). For practical reasons, we require that the $g_i(\cdot,\cdot)$ depend on δv_{e_i} only, that is, vehicle i does not have access to the speeds of the other vehicles.

At this point, different strategies can be adopted for control, depending on the flow of information among the vehicles. Figure 9 shows three representative configurations, illustrated for the case of 3 robots.

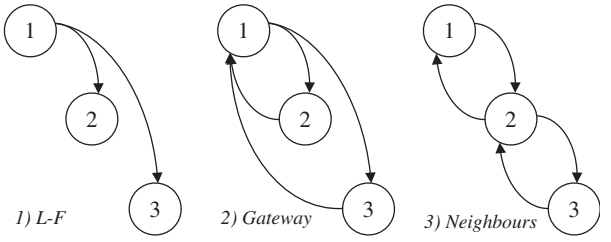


Figure 9. Information flow diagram: (1) Leader-Follower, (2) Gateway, (3) Neighbours.

Leader-Follower. This configuration captures the case where a vehicle, elected as the "Leader", executes a path following algorithm at a required forward speed and relays its position to the remaining vehicles. It is up to the "Followers" to keep the formation, based on info received from the Leader.

Gateway. In this configuration, a vehicle serves as a Gateway (vehicle 1 in figure 9, gateway configuration). Each of the remaining vehicles sends its position to Gateway and receives the Gateway's position. Coordination is thus achieved through the Gateway vehicle.

Neighbors. In this case, each vehicle communicates with its immediate neighbours. The n vehicles are indexed according to the spatial pattern they are required to achieve, and vehicle i; $i = 2, \ldots, n-1$ communicates with vehicles $i-1$ and $i+1$. Vehicles 1 and n communicate with vehicles 2 and $n-1$, respectively.

Different control laws can now be proposed for the configurations above. In what follows we analyze the Leader-Follower and the Gateway configurations.

3.1. *Leader-Follower*

Suppose vehicle 1 is the Leader. A possible coordination control law is

$$\begin{aligned} \delta F_1 &= -a_1 \delta v_{e_1} \\ \delta F_i &= -a_i \delta v_{e_i} + R_i b_i \delta \bar{s}_{1,i}; \quad i = 2, \ldots, n \end{aligned} \tag{27}$$

which corresponds to having the leader adjust its own speed independently, while the remaining vehicles control their speeds in response to the along-path distances between them and the leader.

Equation (26), together with (27), define the coordination system dynamics. Simple computations show that this linear system exhibits one eigenvalue at $-a_1 \,\mathrm{rad}\, s^{-1}$, and $n-1$ pairs of eigenvalues at the roots of $\lambda^2 + a_i\lambda + b_i = 0$, i.e. the eigenvalues are independent of the radii of the circumferences that the vehicles must follow. Clearly, the closed-loop eigenvalues are stable if a_i and b_i are positive. Furthermore, the eigenvalues can be assigned arbitrarily in the left half complex plane by proper choice of parameters a_i and b_i.

Notice, however that the implementation of the coordination system proposed requires that we have access to δv_{e_i}; $i = 1, 2, \ldots, n$. This is easy to do for vehicle 1, because it acts as a leader and sets the "pace" for the formation by traveling at the desired speed v_1^c, which is set in advance. However, when it comes to the remaining vehicles, it is best not to feedforward the desired speeds v_i^c; $i = 2, \ldots, n$, lest the radii be different from their expected values. In this case, one should require that the vehicles "learn" their speeds v_i^c; $i = 2, \ldots, n$ automatically. This can be done by changing the above control law to include integrators on the states $\delta \bar{s}_{1,i}$ and doing a $\mathcal{D}$-implementation of the resulting control scheme, following the circle of ideas introduced in Kaminer *et al.* (1995). Formally, define the new states $\delta e_{1,i}$ through

$$\delta \dot{e}_{1,i} = \delta \bar{s}_{1,i}. \tag{28}$$

and modify the control law (27) to incorporate extra feedback terms from the new states, yielding

$$\begin{aligned} \delta F_1 &= -a_1 \delta v_{e_1} \\ \delta F_i &= -a_i \delta v_{e_i} + R_i b_i \delta \bar{s}_{1,i} + R_i c_i \delta e_{1,i}. \end{aligned} \tag{29}$$

Straightforward computations show that the resulting (coordination) closed loop system exhibits one eigenvalue at $-a_1 \,\mathrm{rad}\, s^{-1}$, and $n-1$ pairs of complex eigenvalues at the roots of polynomial $\lambda^3 + a_i\lambda^2 + b_i\lambda + c_i$. Again, the eigenvalues are independent of the radii R_i and can be placed arbitrarily in the left half complex plane. Using the methodology exposed in Kaminer *et al.* (1995) it is simple to go from perturbed to global variables and to arrive at the final coordination control law

$$\begin{aligned} F_1 &= -a_1(v_1 - v_1^c) \\ \dot{z}_i &= -R_i c_i \bar{s}_{1,i} + a_i \dot{v}_i \\ F_i &= -z_i + R_i b_i \bar{s}_{1,i}, \end{aligned} \tag{30}$$

where the derivative $\dot{v}_i$ can be computed numerically using an approximate differentiation operator.

As in Kaminer *et al.* (1995), it can be shown that the linearization of the full nonlinear system (about the trimming conditions corresponding to the situation where the vehicles execute perfect coordinated path following) has the same set of eigenvalues as those for the linear designs, that is, they are the roots of $\lambda^3 + a_i\lambda^2 + b_i\lambda + c_i$. Thus, from a local point of view, the coordination error converges to zero and the velocities of the robots are synchronized. Notice in the above control law that only the desired velocity v_1^c of the leader must be provided. As for the other velocities, and since they appear only through their derivatives and their steady state values are constant, it is not required to feedforward their trimming values.

3.2. *Gateway*

Inspired by the previous coordination control law, and taking into consideration the communications structure for the *Gateway* configuration (with vehicle 1 as the Gateway vehicle) suggests the control law

$$\begin{aligned} \delta F_1 &= -a_1 \delta v_{e_1} - b_0 \delta e_v - R_1(b_1 \Sigma \delta \bar{s}_{1,i} + c_1 \Sigma \delta e_{1,i}) \\ \delta F_i &= -a_i \delta v_{e_i} + R_i(b_i \delta \bar{s}_{1,i} + c_i \delta e_{1,i}) \\ \delta \dot{e}_{1,i} &= \delta \bar{s}_{1,i} \\ \delta \dot{e}_v &= \delta v_{e_1}, \end{aligned}$$

where Σ denotes the summation operator over all $i \geq 2$. Define $z_1 = a_1 \delta v_{e_1} + b_0 \delta e_v + R_1 c_1 \Sigma \delta e_{1,i}$ and

$z_i = a_i \delta v_{e_i} - R_i c_i \delta e_{1,i}$; $i \geq 2$. A straightforward application of the $\mathcal{D}$-methodology yields the final nonlinear coordination control law

$$F_1 = -z_1 - R_1 b_1 \Sigma \bar{s}_{1,i}$$
$$\dot{z}_1 = a_1 \dot{v}_1 + b_0 (v_1 - v_1^c) + R_1 c_1 \Sigma \bar{s}_{1,i}$$

for the Gateway vehicle and

$$F_i = -z_i + R_i b_i \bar{s}_{1,i}$$
$$\dot{z}_i = a_i \dot{v}_i - R_i c_i \bar{s}_{1,i}.$$

for the remaining vehicles. The problem of finding a set of gains to obtain a stable system with adequate transient performance is not tackled in this paper. Instead, we simply prove that the above system can always be stabilized in a trivial manner. To simplify the presentation, the analysis is done for the case of three vehicles. Let $a_i = a$, $b_i = b$ and $c_i = c$; $i = 1, \ldots, 3$. The closed loop eigenvalues are easily seen to be the roots of

$$(\lambda^3 + a\lambda^2 + b\lambda + c)\big(\lambda(\lambda + a)(\lambda^3 + a\lambda^2 + 3b\lambda + 3c) + b_0(\lambda^3 + a\lambda^2 + b\lambda + c)\big) = 0. \quad (31)$$

The result follows from the fact that if a, b, and c are chosen so as to make $\lambda^3 + a\lambda^2 + b\lambda + c$ a stable polynomial, then so is (31) for any positive b_0. This can be easily seen by applying the Routh-Hurwitz criterion.

4. Stability of the coordinated path following control system

In the previous sections, the general problem of coordinated path following was broken down into two problems: path following and vehicle coordination (that is, synchronization in time), according to a desired formation pattern. This procedure, even though not fully justified from a theoretical point of view, allowed for the derivation of simple control laws for the two problems taken separately. In fact, the design of a path following controller using the decoupling methodology relied on the assumption that the coupling term $-c_c \delta v_e$ in the dynamics (11) could indeed be viewed as a vanishing perturbation coming from the coordination level. Conversely, the design of the coordination controller assumed that the perturbation terms δd_{ek} could also be viewed as perturbations being reduced to zero at the path following level. In view of these yet unjustified assumptions, the control laws derived should, at this stage, be simply viewed as candidates to be brought together to yield a combined coordinated path following controller, the stability of which must be proven rigorously. This is the objective of this section. A proof of stability is done for the Leader-Follower configuration and for a state feedback control law at the path following level. Once again, two path following strategies are considered: state transformation and decoupling.

4.1. *State transformation*

The state transformation case is trivial to analyze because there is true decoupling of the path following and coordination schemes. In fact, close examination of the error dynamics in (18) shows that the evolution of the coordination-related variable δv_e is totally independent of the path following related variables. Furthermore, the latter are driven asymptotically to zero with the control law (19) proposed. As a consequence, the variables δde_k that appear at the coordination level in (24)–(25) vanish asymptotically. The coordination scheme with the control law (25) proposed is therefore an asymptotically stable system driven by vanishing external perturbations. As a consequence, all relevant state variables are also driven to zero asymptotically. The same conclusion can be reached by considering the linearized dynamics of the system that arise when a state-feedback/state-transformation control strategy is used for path following and a Leader-Follower configuration is adopted at the coordination level. From (18), (19), (24), and (29) the complete dynamics can be written as

$$\begin{aligned}
\delta \dot{v}_{e_1} &= -a_1 \delta v_{e_1} \\
\delta \dot{\eta}_{e_1} &= -k_1 \delta \eta_{e_1} - k_2 \delta d_{e_1} - k_3 \delta \psi_{e_1} - k_4 z_1 \\
\delta \dot{d}_{e_1} &= v_1^c \delta \psi_{e_1} \\
\delta \dot{\psi}_{e_1} &= \delta \eta_{e_1} - c_{c_1}^2 v_1^c \delta d_{e_1} \\
\dot{z}_1 &= \delta d_{e_1} \\
\delta \dot{v}_{e_i} &= -a_i \delta v_{e_i} + R_i b_i \delta \bar{s}_{1,i} + R_i c_i \delta e_{1,i} \\
\delta \dot{\eta}_{e_i} &= -k_1 \delta \eta_{e_i} - k_2 \delta d_{e_i} - k_3 \delta \psi_{e_i} - k_4 z_i \\
\delta \dot{d}_{e_i} &= v_i^c \delta \psi_{e_i} \\
\delta \dot{\psi}_{e_i} &= \delta \eta_{e_i} - c_{c_i}^2 v_i^c \delta d_{e_i} \\
\dot{z}_i &= \delta d_{e_i} \\
\delta \dot{\bar{s}}_{1,i} &= \frac{\delta v_{e_1}}{R_1} - \frac{\delta v_{e_i}}{R_i} + \frac{v_1^c \operatorname{sign}(c_{c_1})}{R_1^2} \delta d_{e_1} \\
&\quad - \frac{v_i^c \operatorname{sign}(c_{c_i})}{R_i^2} \delta d_{e_i} \\
\delta \dot{e}_{1,i} &= \delta \bar{s}_{1,i}; \quad i = 2, \ldots, n
\end{aligned} \quad (32)$$

Using the controller gains in (16) yields the closed loop characteristic polynomial $(\lambda + a_1) \times (\lambda + \lambda_p)^{4n} \Pi_{i=2\ldots n}(\lambda^3 + a_i\lambda^2 + b_i\lambda + c_i)$. Clearly, the set of eigenvalues of the total linearized path following control system consists of the union of the two sets of eigenvalues determined for the two systems taken separately. Since the two can be designed to be stable, the stability of the complete coordinated path following system follows.

4.2. *Decoupling*

The decoupling strategy is harder to analyze. Consider the Leader-Follower configuration where vehicle 1 is the leader. Once again, we only indicate how a set of gains can be found so as to make the linearized coordinated path following system stable. From (11), (14), (24), and (29), the closed-loop dynamics can be written as

$$\begin{aligned}
\delta\dot{v}_{e_1} &= -a_1\delta v_{e_1}\\
\delta\dot{r}_{e_1} &= -k_1\delta r_{e_1} - k_2\delta d_{e_1} - k_3\delta\psi_{e_1} - k_4 z_1\\
\delta\dot{d}_{e_1} &= v_1^c\delta\psi_{e_1}\\
\delta\dot{\psi}_{e_1} &= \delta r_{e_1} - c_{c_1}^2 v_1^c\delta d_{e_1} - c_{c_1}\delta v_{e_1}\\
\dot{z}_1 &= \delta d_{e_1}\\
\delta\dot{v}_{e_i} &= -a_i\delta v_{e_i} + R_i b_i\delta\bar{s}_{1,i} + R_i c_i\delta\bar{e}_{1,i}\\
\delta\dot{r}_{e_i} &= -k_1\delta r_{e_i} - k_2\delta d_{e_i} - k_3\delta\psi_{e_i} - k_4 z_i\\
\delta\dot{d}_{e_i} &= v_i^c\delta\psi_{e_i}\\
\delta\dot{\psi}_{e_i} &= \delta r_{e_i} - c_{c_i}^2 v_i^c\delta d_{e_i} - c_{c_i}\delta v_{e_i}\\
\dot{z}_i &= \delta d_{e_i}\\
\delta\dot{\bar{s}}_{1,i} &= \frac{\delta v_{e_1}}{R_1} - \frac{\delta v_{e_i}}{R_i}\\
&\quad + \frac{v_1^c\,\mathrm{sign}(c_{c_1})}{R_1^2}\delta d_{e_1} - \frac{v_i^c\,\mathrm{sign}(c_{c_i})}{R_i^2}\delta d_{e_i}\\
\delta\dot{e}_{1,i} &= \delta\bar{s}_{1,i};\quad i = 2,\ldots,n.
\end{aligned} \tag{33}$$

Without loss of generality, consider the case of two vehicles only. The corresponding closed-loop characteristic polynomial is

$$(\lambda + a_1)(\lambda + \lambda_p)^4\Bigg((\lambda^3 + a_2\lambda^2 + b_2\lambda + c_2)(\lambda + \lambda_p)^4 - \left(\frac{v_2^c}{R_2}\right)^2\lambda(\lambda + 4\lambda_p)(b_2\lambda + c_2)\Bigg). \tag{34}$$

Notice that $(\lambda + a_1)(\lambda + \lambda_p)^4$ can be chosen to be stable by proper design of the path following controller and thus plays no role in the stability analysis that follows. Choose $a_2 = 3\xi$, $b_2 = 3\xi^2$, and $c_2 = \xi^3$ for some $\xi > 0$, and let $k = \max_{v_2^c, R_2}\{(v_2^c/R_2)^2\}$, that is, let k be the largest expected trimming value of the rotational speed of the Follower. With the above choices, the relevant part of the characteristic polynomial that depends on k becomes

$$(\lambda + \xi)^3(\lambda + \lambda_p)^4 - 3\xi^2 k\lambda(\lambda + 4\lambda_p)(s + \xi/3) \tag{35}$$

Notice that the poles at $-\lambda_p$ come from the path following, and those at ξ come from coordination level. Let $\xi = m\lambda_p$, and for a given k choose $\lambda_p = 2m\sqrt{k}$. A straightforward but cumbersome application of the Routh-Hurwitz stability criteria reveals that the above system is stable for $m > m_0 = 0.63$. Thus, given any k, λ_p can be chosen large enough (note that $\lambda_p = 2mk^{1/2}$) to guarantee stability. To better illustrate this result, a numerical example was run with $k = 0.25\,\mathrm{rad}^2\,\mathrm{s}^{-2}$ and $\xi = 2m_0^2 k^{1/2} = 0.4\,\mathrm{rad\,s}^{-1}$. The critical λ_p was computed as $\lambda_{p_0} = 2m_0 k^{1/2} = 0.63\,\mathrm{rad\,s}^{-1}$. Figure 10 shows the root loci for varying $\lambda_{p_0} - 0.5 < \lambda_p < \lambda_{p_0} + 2$. Clearly, the system is stable for $\lambda_p > \lambda_{p_0}$.

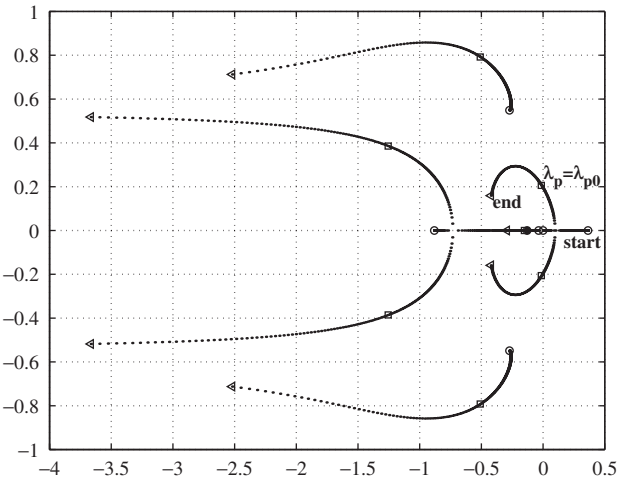


Figure 10. Root locus for varying λ_p.

5. Simulations

This section contains the results of simulations that illustrate the performance of the coordinated path following control system developed. Figures 11 and 12 correspond to simulations where 4 wheeled robots were required to follow paths that consists of portions of straight lines and nested arcs of circumferences, while holding an in-line formation pattern. In the simulation, the controller gains were scheduled on the path's

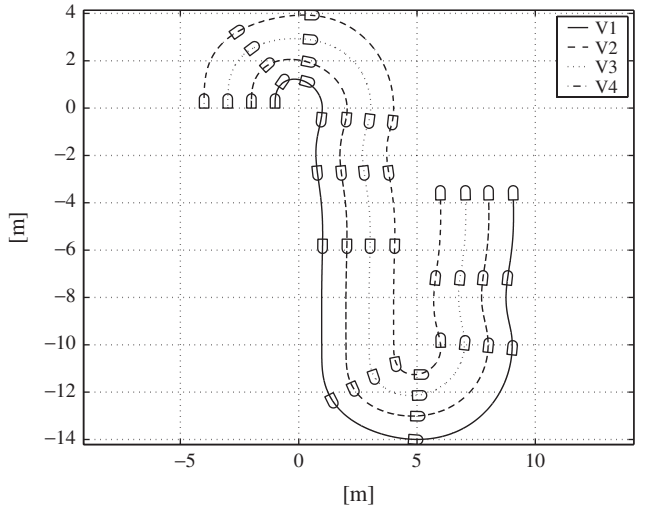


Figure 11. Coordinated path following: Gateway configuration.

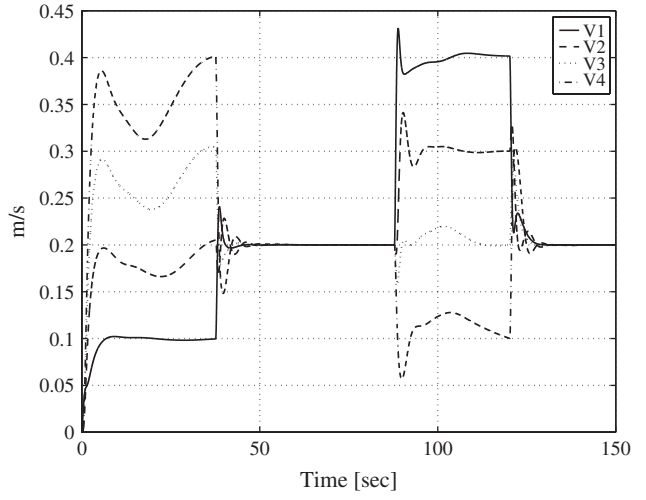


Figure 12. Coordinated path following: vehicle speeds.

curvature as well as on the rotation speed of the vehicles, as explained before. The simulation assumed a Gateway configuration, where the Gateway vehicle (vehicle number 1, denoted $V1$) was assigned a piecewise constant speed profile that acted as a reference for the actual speed of that vehicle, shown in figure 12. During the first phase of the maneuver, along an arc of a circumference, the desired speed of $V1$ was set to $0.1\,\mathrm{m\,s^{-1}}$. All vehicles started at zero speed, on top of their respective paths. During the successive legs of the mission, the reference speed for vehicle $V1$ was set to 0.2, 0.4, and $0.2\,\mathrm{m\,s^{-1}}$. Notice how the remaining vehicles adjust their speeds to meet the formation requirements. Figure 13 shows the coordination errors, as captured by the along-path distances between vehicle 1

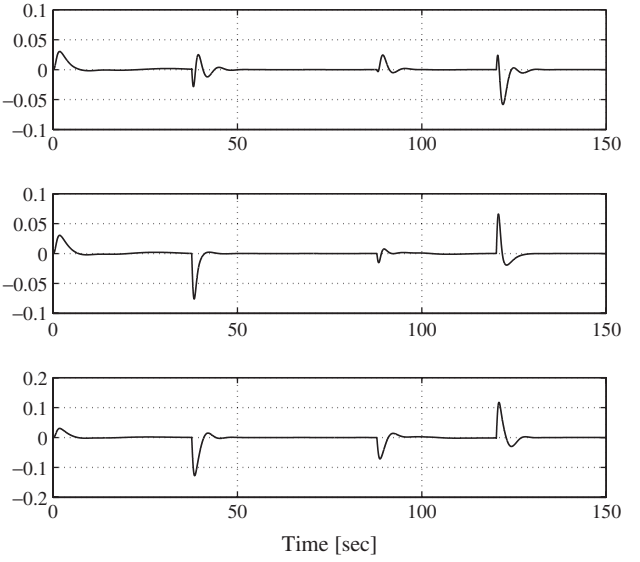


Figure 13. Coordinated path following, the coordination error between vehicle 1 and the remaining vehicles, 2, 3 and 4.

and the remaining vehicles. The figures include information on both $s_{i,j}$ and $\bar{s}_{i,j}$. Because the paths to be followed consist of segments of straight lines and semi-circumferences, normalization factors were used to make the above variables non-dimensional.

6. Robustness against vehicle failures: a brief discussion

The previous sections described a set of solutions to the problem of coordinated path following of multiple wheeled robots. However, the important issue of robustness against vehicle failures or loss of inter-vehicle communications was not addressed explicitly. Two possible failure situations are described below as illustrative examples.

Vehicle failures. This situation occurs when one or more of the vehicles cannot achieve their desired formation speeds. For example, in a Leader-Follower configuration one of the vehicles may get ''stuck'' at some fixed velocity that is different from the assigned one. The need then arises to assess the stability of the resulting formation and, in particular, to find out if the along-path ''coordination errors'' $\bar{s}_{i,j}$ remain bounded.

Communication failures. In this case, one or more of the inter-vehicle communication links fail temporarily or permanently. For example, the communication link between two vehicles may fail briefly at random instants of time, or one vehicle may only be able to broadcast its position to a subset of the remaining vehicles and receive information from another subset. In both cases, it is crucial to find out if the formation remains stable.

The study of these problems is far from complete and warrants further research efforts. Here, for the sake of completeness, we simply touch upon a number of simple problems in the area of robustness against vehicle failures by examining the coordination dynamics only. Further details are given in Ghabcheloo *et al.* (2004).

Leader-Follower configuration. In the absence of vehicle failures, the coordination dynamics can be written as

$$\begin{aligned} \dot{v}_1 &= -a_1 v_1 + a_1 v_1^c \\ \dot{v}_i &= -z_i + R_i b_i \bar{s}_{1,i} \\ \dot{\bar{s}}_{1,i} &= \frac{1}{R_1} v_1 - \frac{1}{R_i} v_i \\ \dot{z}_i &= -a_i z_i + R_i (a_i b_i - c_i) \bar{s}_{1,i} \end{aligned} \tag{36}$$

Suppose now that due to a failure the speed of the leader is fixed at some positive speed v_0. The dynamics of the remaining vehicles become

$$\begin{aligned} \dot{v}_i &= -z_i + R_i b_i \bar{s}_{1,i} \\ \dot{\bar{s}}_{1,i} &= \frac{1}{R_1} v_0 - \frac{1}{R_i} v_i \\ \dot{z}_i &= -a_i z_i + R_i (a_i b_i - c_i) \bar{s}_{1,i}. \end{aligned} \tag{37}$$

Clearly, the eigenvalues of (37) are those of (36) except for the eigenvalue at $-a_1$. Thus, if the dynamics without failures are asymptotically stable, so are the dynamics in the case where the leader fails. As a consequence, $1/R_1 v_0 - 1/R_i v_i, \bar{s}_{1,j};\ i = 2, \ldots, n$ are driven to zero. Stated intuitively, all vehicles slow down or speed up in order to adopt the speeds that are required to maintain formation. Should the failure occur in one the followers, say vehicle $k \geq 1$, it is possible to show that all the remaining vehicles will still synchronize with the Leader.

Gateway configuration. In the case of the Gateway configuration, a similar analysis shows that if for some reason the velocity of the Gateway vehicle is fixed at v_0, then (as in the case of the Leader-Follower configuration) the other vehicles will adapt their velocities so as to keep the formation. It is interesting to examine the case where a vehicle other than the Gateway vehicle has a failure and its speed gets fixed at v_0.

Without any loss of generality, assume that the vehicle with a failure is vehicle 2. In this case, straightforward computations show that the coordination dynamics become

$$\begin{aligned} \dot{v}_1 &= -z_1 - R_1 b_1 \Sigma \bar{s}_{1,i} \\ \dot{v}_i &= -z_i + R_i b_i \bar{s}_{1,i} \\ \dot{\bar{s}}_{1,2} &= \frac{1}{R_1} v_1 - \frac{1}{R_2} v_0 \\ \dot{\bar{s}}_{1,i} &= \frac{1}{R_1} v_1 - \frac{1}{R_i} v_i \\ \dot{z}_1 &= -a_1 z_1 + b_0 (v_1 - v_1^c) - R_1 (a_1 b_1 - c_1) \Sigma \bar{s}_{1,i} \\ \dot{z}_i &= -a_i z_i + R_i (a_i b_i - c_i) \bar{s}_{1,i} \end{aligned}$$

for $i = 3, \ldots, n$. If properly designed, the coordination law will ensure that the above system is stable. It can then be concluded that all vehicles except vehicle 2 will learn their correct speed and keep the formation. Vehicle 2 will however exhibit a finite steady state error

$$\bar{s}_{1,2} \to \frac{b_0}{c_1} \left(\frac{v_1^c}{R_1} - \frac{v_0}{R_2} \right). \tag{38}$$

This error can be reduced to zero by adding an extra integral state on the coordination error (Ghabcheloo *et al.* 2004).

7. Conclusions and suggestions for further research

The paper offered a solution to the problem of steering a fleet of wheeled robots along a set of given spatial paths, while keeping a desired inter-vehicle formation pattern. The methodology proposed builds on linearization techniques and draws heavily on previous work on the implementation of gain-scheduled controllers. Using this set-up, path following and inter-vehicle coordination are essentially decoupled. Path following for each vehicle amounts to reducing a conveniently defined error vector to zero. Vehicle coordination is achieved by adjusting the speed of each of the vehicles along its path, according to information on the position of the remaining vehicles only. This allowed for a simple analysis of the resulting coordinated path following control system. The resulting control system is simple to implement and avoids feedforwarding the desired speed of all the vehicles. In fact, only the velocity of one of the vehicles is required, the other vehicles recruiting their velocities automatically to keep the formation. The paper considered different vehicle configurations and addressed some of the problems that arise when the vehicles or the communications network fail. In particular, it was shown in the case of vehicle failure how the remaining vehicles adjust their speeds to try and maintain formation. The work reported was but a first attempt to derive simple, easy to implement control systems for coordinated path following. Further work is required to fully address the problems that arise when communications fail temporarily. Extending the results to a full nonlinear setting and applying them to

the control of more complex vehicles (air and marine robots) is also a subject for future research.

Acknowledgement

This work was supported by the Portuguese FCT POSI programme under framework QCA III. The first author benefited from a PhD grant from the Foundation for Science and Technology (FCT), Portugal.

References

R. Beard, J. Lawton and F. Hadaegh, ''A coordination architecture for spacecraft formation control'', *IEEE Trans. Control Syst. Technol.*, 9, pp. 777–790, 2001.

J. Desai, J. Otrowski and V. Kumar, ''Controlling formations of multiple robots'', in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA98)*, 1998, pp. 2864–2869.

P. Encarnação and A. Pascoal, ''Combined trajectory tracking and path following: an application to the coordinated control of marine craft'', in *Proceedings of the IEEE Conference on Decision and Control (CDC'2001)*, Orlando, FL, 2001.

A. Fax and R. Murray, ''Information flow and cooperative control of vehicle formations'', in *Proceedings of the 2002 IFAC World Congress*, Barcelona, 2002a.

A. Fax and R. Murray, ''Graph Laplacians and stabilization of vehicle formations'', in *Proceedings of the 2002 IFAC World Congress*, Barcelona, 2002b.

T. Ghabcheloo, A. Pascoal and C. Silvestre, Coordinated path following control using linearization techniques. Internal Report CPF01, Institute for Systems and Robotics, January 2004.

F. Giuletti, L. Pollini and M. Innocenti, ''Autonomous formation flight'', *IEEE Control Syst. Mag.*, 20, pp. 34–44, 2000.

I. Kaminer, A. Pascoal, P. Khargonekar and E. Coleman, ''A velocity algorithm for the implementation of gain-scheduled controllers'', *Automatica*, 31, pp. 1185–1191, 1995.

I. Kaminer, A. Pascoal, E. Hallberg and C. Silvestre, ''Trajectory tracking for autonomous vehicles: an integrated approach to guidance and control'', *J. Guid. Control Dynam.*, 21, pp. 29–38, 1998.

H.K. Khalil, in *Nonlinear Systems*, 3rd ed., Upper Saddle River, NJ: Pearson Education, 2000.

L. Lapierre, D. Soetanto and A. Pascoal, ''Coordinated motion control of marine robots'', in *Proceedings of the 6th IFAC Conference on Manoeuvering and Control of Marine Craft (MCMC2003)*, Girona, Spain, 2003a.

R. Lapierre, D. Soetanto and A. Pascoal, ''Nonlinear path following control of autonomous underwater vehicles'', in *Proceedings of the 1st IFAC Conference on Guidance and Control of Underwater Vehicles (GCUV'03)*, Newport, UK, 2003b.

A. Micaelli and C. Samson, ''Path following and time-varying feedback stabilization of a wheeled robot'', in *Proceedings of the International Conference ICARCV'92*, Singapore, 1992.

A. Micaelli and C. Samson, ''Trajectory-tracking for unicycle-type and two-steering-wheels mobile robots'', Technical Report No. 2097, INRIA, Sophia-Antipolis France, 1993.

P. Ögren, M. Egerstedt and X. Hu, ''A control lyapunov function approach to multiagent coordination'', *IEEE Trans. Robot. Autom.*, 18, pp. 847–851, 2002.

A. Pascoal *et al.*, ''Robotic ocean vehicles for marine science applications: the european ASIMOV Project'', in *Proceedings of the OCEANS'2000 MTS/IEEE*, Rhode Island, Providence, USA, 2000.

M. Pratcher, J. D''Azzo and A. Proud, ''Tight formation control'', *J. Guid. Control Dynam.*, 24, pp. 246–254, 2001.

M. Queiroz, V. Kapila and Q. Yan, ''Adaptive nonlinear control of multiple spacecraft formation flying'', *J. Guid. Control Dynam.*, 23, pp. 385–390, 2000.

C. Silvestre, ''Multi-objective optimization theory with applications to the integrated design of controllers/plants for autonomous vehicles''. PhD thesis (in English, Instituto Superior Técnico, Lisbon, Portugal (2000).

C. Silvestre, A. Pascoal and I. Kaminer, ''On the design of gain scheduled trajectory tracking controllers'', *Int. J. Robust Nonlinear Control*, 12, pp. 797–839, 2002.

R. Skjetne, S. Moi and T. Fossen, ''Nonlinear formation control of marine craft'', in *Proceedings of the IEEE Conference on Decision and Control (CDC2002)*, Las Vegas, NV, 2002.

R. Skjetne, I. Flakstad and T. Fossen, ''Formation control by synchronizing multiple maneuvering systems'', in *Proceedings of the 6th IFAC Conference on Manoeuvering and Control of Marine Craft (MCMC2003)*, Girona, Spain, 2003.

D. Stilwell and B. Bishop, ''Platoons of underwater vehicles'', *IEEE Control Syst. Mag.*, December, pp. 45–52, 2000.