Journal of Intelligent & Robotic Systems (2018) 91:35–58
https://doi.org/10.1007/s10846-018-0805-9

Mobile Robot Control and Navigation: A Global Overview

Spyros G. Tzafestas[1]

Received: 7 February 2018 / Accepted: 26 February 2018 / Published online: 10 March 2018

Abstract

The aim of this paper is to provide a global overview of mobile robot control and navigation methodologies developed over the last decades. Mobile robots have been a substantial contributor to the welfare of modern society over the years, including the industrial, service, medical, and socialization sectors. The paper starts with a list of books on autonomous mobile robots and an overview of survey papers that cover a wide range of decision, control and navigation areas. The organization of the material follows the structure of the author's recent book on mobile robot control. Thus, the following aspects of wheeled mobile robots are considered: kinematic modeling, dynamic modeling, conventional control, affine model-based control, invariant manifold-based control, model reference adaptive control, sliding-mode control, fuzzy and neural control, vision-based control, path and motion planning, localization and mapping, and control and software architectures.

Keywords Mobile robot · Autonomous mobile robot · Control · Path planning · Motion planning · Navigation · Localization · Mapping · Control architecture · Software architecture

Mathematics Subject Classification (2010) 68T40 · 70E60 · 93C85 · 70Q05 · 70B15

1 Introduction

Terrestial (Ground) robots are distinguished in *fixed-place* robots and *mobile robots*. Fixed-place robots are robots of which the base is fixed at a specific place, and hence they have a workspace limited by their kinematic structure and the size of their links. Unlike fixed-place robots, mobile robots are robots that can move from one place to another autonomously, i.e., they have the special feature of moving around freely within a predefined workspace to perform given tasks and achieve desired goals. Today's *autonomous mobile robots* **(AMRs)** can move around safely in cluttered surroundings, understand natural speech, recognize real objects, locate themselves, plan paths, navigate themselves, and generally think by themselves. The design of AMRs employs the methodologies and technologies of intelligent, cognitive, and behavior-based control, and attempts to maximize flexibility of performance subject to minimal input dictionary and minimal computational complexity.

Therefore, AMRs belong to the broad class of *intelligent robots*. Ronald Arkin [7], defines an intelligent robot as 'a machine able to extract information from its environment and use knowledge about its work to move safety in a meaningful and purposive manner'. In general, a robot is referred to in the literature as a machine that performs an '*intelligent connection between perception and action*'.

The navigation and control field of AMRs has achieved over the years high maturity, both in theory and practice, and a large number of authored and edited books have been published in the international scene. Twenty authored books are listed in Table 1, in which the authors' names, publication years, and gross contents of them are provided [1–20]. Ten edited books written by invited or conference authors are [21–30]. Also, several special issues of international journals, and numerous state-of-art papers exist in the literature which present surveys of particular areas within the autonomous mobile robotics field [31–52]. A brief summary of some of these papers is given in Section 2.

✉ Spyros G. Tzafestas
tzafesta@cs.ntua.gr

1 School of Electrical and Computer Engineering, National Technical University of Athens, Zographou 15773, Athens, Greece

Table 1 Books on Autonomous Robot Navigation and Control (1991–2017)

Author	Year	Content
A.M.Meystel [1]	1991	Intelligent motion control. Evolution of autonomous mobile robots (AMR). Autonomous mobility. Cognitive control of AMR. Nested hierarchical control. Intelligent Modules (planner, navigator, pilot).Cartographer.
J.C. Latombe [2]	1991	Configuration space (CS) of rigid object. Obstacles in CS. Road map methods. Cell de composition (exact, approximate). Potential field methods. Dealing with uncertainty. Movable objects.
J.L. Leonard [3]	1992	The navigation problem. Sonar sensor model. Model-based localization. Map building. Simultaneous map building and localization. Directed sensing strategies. Why use sonar?
J.L. Jones [4]	1995	TuteBot. Computational Hardware. Designing and prototyping. Sensors. Mechanics. Motors. Robot programming. Robot applications. Robot design principles. Unsolved problems.
H.R. Everett [5]	1995	Design considerations. Dead reckoning. Odometer sensors. Doppler and inertial navigation. Typical mobility configurations. Tactile and proximity sensing. Triangulation ranging. Active triangulation. Range from focus. Return signal intensity. Guide path following. Position location sensors. Ultrasonic and optical position location systems.
J. Borenstein [6]	1996	Sensors for mobile robot positioning. Heading sensors. Active beacons. Sensors for map-based positioning. Landmark navigation. Active beacon navigation systems.
R.C. Arkin [7]	1998	Whence behavior? Animal behavior. Robot behavior. Behavior-based architectures. Representation issues. Hybrid deliberative/reactive architectures. Perceptual basis for behavior-based control. Adaptive behavior. Social behavior. Fringe robotics-Beyond behavior.
J. Canny [8]	1998	Robot motion planning problems. Motion constraints. Elimination theory. The roadmap algorithm. Performance improvements. Lower bounds for motion planning. Motion planning with uncertainty.
X.Zhu [9]	2001	Outdoor mobile robots. Motion Control. Cooperative motion control and architecture. Kinematic motion control.
U.Nehmzow [10]	2003	Robot hardware. Robot learning. Making sense of raw sensor. Navigation. Novelty detection. Simulation. Modeling robot-environment. Robot behavior analysis. Locomotion. Mobile robot kinematics. Perception. Mobile robot localization. Planning and Navigation.
R. Siegwart [11]	2005	Locomotion. Mobile robot kinematics. Perception. Mobile robot localization. Planning and Navigation.
F. Cuesta [12]	2005	Fuzzy systems. Stability analysis. Bifurcations in simple Takagi-Sugeno fuzzy systems. Intelligent control of mobile robots with fuzzy perception. Stability of mobile robots with fuzzy reactive navigation. Intelligent system for parallel parking of cars and tractor-trailers.
K. Berns [13]	2009	Historical overview of autonomous land vehicles. Vehicle kinematics. Sensor systems. Where am I? The localization problem. Map building. Navigation strategies. Control architectures. Software frameworks.

Table 1 (continued)

Author	Year	Content
G. Dudek [14]	2010	Fundamental problems. Mobile robot hardware. Non-visual sensors and algorithms. Visual sensors and algorithms. Representing and reasoning about space. System control. Pose maintenance and localization. Mapping and related tasks. Robot collectives. Robots in practice. The future of mobile robots.
G. Cook [15]	2011	Mobile robot control. Robot attitude. Robot navigation. Application of Kalman filtering. Remote sensing. Target tracking including multiple targets with multiple sensors. Obstacle mapping and its application to robot navigation. Operating a robotic manipulator. Remote sensing via UAVs. Robot parts. Kinematics.
C.A. Berry [16]	2012	Introduction. Hardware. Control. Feedback control. Representation. Control architectures. Software. Navigation. Localization. Mapping. Simultaneous localization and mapping.
R. Tiwari [17]	2012	Graph based path planning. Common planning techniques. Evolutionary robotics. Behavioral path planning. Hybrid graph-based methods. Hybrid behavioral methods. Multi-robot-systems.
A. Kelley [18]	2013	Introduction. Math fundamentals. Numerical methods. Dynamics. Optimal estimation. State estimation. Sensors for state estimation. Control. Perception. Localization and mapping. Motion planning.
S.G. Tzafestas [19]	2014	Mobile robots. Mobile robot kinematics. Mobile robot dynamics. Mobile robot sensors. Mobile robot control. Lyapunov-based method. Affine systems and invariant manifold methods. Adaptive and robust methods. Fuzzy and neural methods. Vision-based methods. Mobile manipulation modeling and control. Mobile robot path, motion, and task planning. Mobile robot localization and mapping. Generic systemic and software architectures. Experimental studies. Mobile robots at work.
L. Jaulin [20]	2017	Three-dimensional modeling. Feedback linearization. Model-free control. Guidance. Instantaneous localization. Identification. Kalman filter.

2 Overview of Twelve Survey Papers on Mobile Robots

The AMR areas covered by these surveys are:

- Deep learning in AMRs.
- Multiple cooperative AMRs.
- Self-organized pattern formation in AMRs.
- AMR learning paradigms and applications.
- Performance measures of mobile manipulators.
- Multi-robot coordination and decision problems.
- Gathering fat robots and patrolling by mobile robots.
- Control architectures of AMRs.
- Autonomous search and pursuit-evasion.
- Control of mobile robots with trailers.
- Simulation tools and their selection via user selection.
- Geometric registration and configuration selection.

A short outline of them is as follows:

- *Tai and Liu* [31], provide a survey of deep learning (DL) landscape in robotics which includes the development of deep learning in related fields, especially the essential distinctions between image processing and robotic tasks. DL methods are distinguished in two classes: perception, and control systems. DL may be an answer for the future of robotics and serve for solving robust and generic robotic tasks.
- *Cao, Fukunago, and Kalug* [32], review the research on systems composed of multiple autonomous mobile robots exhibiting cooperative behavior. Groups of robots are constructed aiming at studying group architecture, resource conflict of cooperation, learning, and geometric problems. This paper gives a critical survey of existing works along with a discussion of open problems in this field.

- *Yamauchi* [33], provides a survey of pattern formation of autonomous robots focusing on self-organization of mobile robots, especially the power of forming patterns. The existing results show that the robot system's formation power is determined by their asynchrony, obliviousness, and visibility. Besides the existing results, this paper presents a number of open problems related to pattern formation in mobile robotics.
- *Cunha* [34] provides an overview of the state-of-art of the different applications of machine learning methodologies in mobile robotics. He starts by presenting the credit assignment problem, namely: temporal credit assignment, structural credit assignment, and task credit assignment. Then, an overview is given of various learning paradigms characterized by solving each credit assignment problem in a different way.
- *Bostelman, Hong and Marvel* [35], give a survey of available research results concerning the performance measurement of mobile manipulators. The survey provides a literature review of mobile manipulation research with examples and experimental applications. Also the survey presents an extensive list of planning and control references with factors into performance measurement.
- *Yan, Jouandeau and Cherif* [36], deal with the analysis of multi-robot coordination, especially with the literature on problems related to communication mechanisms, planning strategies, and decision making. Having made great progress in basic problems of single-robot control, much of the current research is focused to the study of multi-robot coordination. The authors also review a series of related problems.
- *Bandettini, Luporini and Viglietta* [37], provide a survey of the open problems of (i) gathering fat (non-point) robots which may be *opaque* in the sense that other robots cannot 'see through it', and (ii) the problem of *boundary patrolling* by mobile robots with constraints only on speed and visibility. For at most four robots an algorithm exists in the literature, but the question is whether gathering is always possible for more than 4 robots. A set of mobile robots with constraints only on speed and visibility is working in a polygonal environment having boundary and possible obstacles. The robots must perform a perpetual movement, so that the maximum time span, in which a point of the boundary is not watched by any robot, is minimized.
- *Medeiros* [38], presents a survey of the flavor in existing robot control architectures and identifies attributes of intelligent robot control architectures. He discusses the NASREM architecture, the subsumption architecture, the LAS architecture, and the TCA architecture.
- *Chung, Hollinger and Isler* [39], present a survey of recent results in pursuit evasion and autonomous search relevant to mobile robotic applications. A taxonomy of search problems is given that highlights the differences resulting from several assumptions on the searches, targets, and the environment. Then, a list of a number of fundamental results in the areas of pursuit-evasion and probabilistic search is provided, including a discussion of the field of implementations of mobile robotic systems. Finally, several current open problems in the area are highlighted to explore avenues for future work.
- *David and Manivanan* [40], review a number of existing studies in the control of mobile robots with trailers, which accomplish their task in a faster and cheaper way than an individual robot. The main issue of their study is the complexity and stability of the complete system which is nonlinear and unstable. This paper provides a survey of various control strategies and algorithms for the backward motion of mobile robots with trailers, and identifies some unsolved problems in this area.
- *Ivaldi, Padvis and Nori* [41], overview the panorama of simulation tools that are presently used in robotics. They propose to evaluate user feedback as a way to make an objective and quantitative comparison, which is actually difficult to be done since many of the tools are not open source. To this end, they created an on-line survey about the use of dynamical simulation tools, and analyzed the participants' answers to get a descriptive information *fiche* for the most relevant tools. This can be helpful to robotics workers in choosing the best simulation tool for each particular application.
- *Pomerleau, Colas and Siegwart* [42], review the geometric registration algorithms in robotics, which associate sets of data into a common coordinate system and have been extensively used in object reconstruction, inspection, and localization of mobile robots. They, focus on mobile robotics applications in which point clouds are to be registered, presenting a formalization of geometric registration and casting algorithms proposed in the literature. They review some applications of this framework in mobile robotics that cover different kinds, of platforms, environments, and tasks. The ultimate goal of the authors was to provide guidelines for the selection of geometric registration configuration.

In the next Sections 3 through 9, an overview of the following mobile robot control and navigation topics will be provided:

- Mobile robot kinematics and dynamics.
- Mobile robot control (standard control, state feedback linearized control, invariant manifold-based control).
- Mobile robot adaptive and robust control (MRAC, sliding mode control).

- Mobile robot fuzzy and neural control.
- Mobile robot vision based control.
- Mobile robot path and motion planning.
- Mobile robot localization and mapping.

Section 10 will review the major intelligent control and software architectures for mobile robot systems.

3 Mobile Robot Kinematics and Dynamics

3.1 Mobile Robot Kinematics

Robot Kinematics deals with the configuration of robots in their workspace, the relations between their geometric parameters, and the constraints imposed in their trajectories. The Kinematic equations depend on the geometrical structure of the robot. For example, a fixed robot can have a Cartesian, cylindrical, spherical, or articulated structure, and a mobile robot may have one two, three, or more wheels with or without constraints in their motion. The study of kinematics is a fundamental prerequisite for the study of dynamics, the stability features, and the control of the robot. The development of new and specialized robotic kinematic structures is still a topic of ongoing research, toward the end of constructing robots that can perform more sophisticated and complex tasks in industrial and societal applications.

As in fixed-place robots, the fundamental mathematical tool is the concept of '*homogeneous transformations*', and the concept of '*nonholonomic constraints*'. The kinematics of robots (articulated, mobile, etc.) involves the *direct kinematics* (from the joint space $\mathbf{q}\varepsilon R^n$ to the task space $\mathbf{p}\varepsilon R^m$) and *inverse dynamics* (from $\mathbf{p}\varepsilon R^m$ to $\mathbf{q}\varepsilon R^n$):

$$\mathbf{p} = \mathbf{f}(\mathbf{q}), \;\; \mathbf{q} = \mathbf{f}^{-1}(\mathbf{p})$$

and, respectively, *direct differential kinematics and inverse differential kinematics:*

$$d\mathbf{p} = \mathbf{J}d\mathbf{q}, \quad d\mathbf{q} = \mathbf{J}^{-1}d\mathbf{p}, \quad \mathbf{J} = \left[\frac{\partial x_i}{\partial q_j}\right] \in R^{m\times n}$$

where $\mathbf{p} = [x_1, x_2, \ldots, x_m]^{\mathrm{T}}$.

The major *nonhlonomic* wheeled mobile robots **(WMR)** are: (i) the differential drive WMR, (ii) the tricycle WMR, and (iii) the car-like WMR, which have been extensively studied. The basic kinematic model which is used for modeling the above WMR is the unicycle (Fig. 1). This model is described by the equations:

$$\dot{x}_Q = v_Q \cos\phi, \;\; \dot{y}_Q = v_Q \sin\phi, \;\; \dot{\phi} = v_\phi \tag{1}$$

By eliminating v_Q from the first two equations, we obtain the nonholonomic constraint:

$$-\dot{x}_Q \sin\phi + \dot{y}_Q \cos\phi = 0 \tag{2}$$

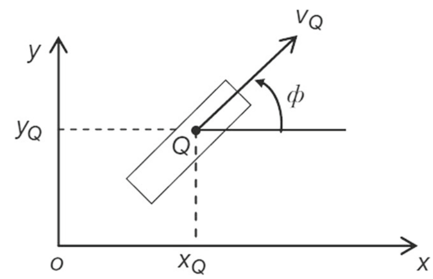

Fig. 1 Kinematic structure of a unicycle

An extension of the car-like WMR is the car-pulling trailer WMR, in which N single-axis trailers are attached to a car-like robot with rear-wheel drive.

The two classes of *holonomic* WMRs that have been mostly studied are: (i) the multi-wheel omnidirectional WMR with orthogonal (universal) wheels, (ii) the four-wheel omnidirectional WMR with mecanum wheels (roller angle ±45°).

Representative references on mobile kinematics, where several complexities (e.g., motion with slip, motion in uneven terrain, etc.) are handled, are [53–59].

3.2 Mobile Robot Dynamics

Like kinematics, dynamics is distinguished in:

- Direct dynamics.
- Inverse dynamics.

Direct dynamics provides the dynamic equations that describe the dynamic responses of the robot to given forces/torques $\tau_1, \tau_2, ..., \tau_m$ that are exerted by the motors.

Inverse dynamics provides the forces/torques that must be exerted to get desired trajectories of the robot links.

In the inverse dynamic model the inputs are the desired trajectories of the link variables, and outputs the motor torques.

The dynamic equations of a WMR are derived by the Newton-Euler and Lagrange equations which have been fully studied in mechanics.

The general Lagrange model of a multilink robot is:

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}, \quad \mathbf{q} = [q_1, q_2, ..., q_m]^{\mathrm{T}} \tag{3}$$

where, for any $\dot{\mathbf{q}} \neq \mathbf{0}, \mathbf{D}(\mathbf{q})$ is an $n \times n$ positive definite matrix. The corresponding model for a nonholonomic robot with m nonholonomic constraints:

$\mathbf{M}(\mathbf{q})\dot{\mathbf{q}} = 0$, $\mathbf{M}(\mathbf{q})$an $m \times n$ matrix, has he form:

$$\mathbf{D}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{M}^{\mathrm{T}}(\mathbf{q})\boldsymbol{\lambda} = \mathbf{E}\boldsymbol{\tau} \tag{4}$$

where $\boldsymbol{\lambda}$ is a vector Lagrange multiplier. Now, defining a matrix $\mathbf{B}(\mathbf{q})$ such as $\dot{\mathbf{q}}(t) = \mathbf{B}(\mathbf{q})\mathbf{v}(t)$, and working on Eq. 4, we get the unconstrained Lagrange model:

$$\bar{\mathbf{D}}(\mathbf{q})\dot{\mathbf{v}} + \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{v} + \bar{\mathbf{g}}(\mathbf{q}) = \bar{\mathbf{E}}\tau \tag{5}$$

where $\bar{\mathbf{D}} = \mathbf{B}^{\mathrm{T}}\mathbf{D}\mathbf{B},\ \bar{\mathbf{C}} = \mathbf{B}^{\mathrm{T}}\mathbf{D}\dot{\mathbf{B}} + \mathbf{B}^{\mathrm{T}}\mathbf{C}\mathbf{B},\ \bar{\mathbf{g}} = \mathbf{B}^{\mathrm{T}}\mathbf{g},\ \bar{\mathbf{E}} = \mathbf{B}^{\mathrm{T}}\mathbf{E}$.

The dynamic model (5) has found extensive use for finding the dynamic models of WMRs of any kind (with slip, uneven terrains, differential drive, omnidirectional, etc.), and applying control schemes [60–63].

The Newton-Euler dynamic model is simpler in the sense that does not need extensive derivations like the Lagrange model, and in many cases it is preferred [64–66].

4 Wheeled Mobile Robot Control

4.1 Standard Controllers

The standard general robot controllers that have been studied over the years are:

- Proportional plus integral controller.
- Lyapunov function-based controller.
- Computed torque controller.
- Resolved motion rate controller.
- Resolved motion acceleration controller.

All these controllers have been extensively applied to WMRs. The control procedure of a WMR involves two stages, namely [19, 67]:

- Kinematic tracking/stabilizing control.
- Dynamic tracking control.

The first uses the kinematic model of the robot, with a proper candidate Lyapunov function, and yields the control laws for the linear and angular velocities (v, ω) of the robot. The second uses the Lagrange model or the Newton dynamic model which may involve or not the motor dynamics and the gear box. This procedure belongs to the general class of back-stepping control [67–70]. A solution of the control of omnidirectional robots, which uses the resolved acceleration control scheme, was provided in [19, 71]. The case of parking control of a car-like WMR was studied in [72], and a formation (leader-follower) controller was derived in [73].

4.2 Advanced Controllers

Here, we will deal with two kinds of advanced controllers, namely:

- State feedback linearization-based controllers.
- Invariant manifolds-based controllers.

The first class of controllers uses the affine dynamic model [74, 75]:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{g}_o(\mathbf{x}) + \sum_{i=1}^{m} \mathbf{g}_i(\mathbf{x})u_i,\ \mathbf{x} \in R^n,\ \mathbf{g}(\mathbf{x}) \in R^n \\ &= \mathbf{g}_o(\mathbf{x}) + \mathbf{G}(\mathbf{x})u,\ \mathbf{G}(\mathbf{x}) = [g_1(x); ..., g_m(x)]\end{aligned}$$

where $\mathbf{x} = [x_1, x_2, ...x_n]^{\mathrm{T}},\ \mathbf{u} = [u_1, u_2, ..., u_m]^{\mathrm{T}}(m \le n)$.

The drift term $\mathbf{g}_o(\mathbf{x})$ represents the general kinematic constraints of the system. Fundamental concepts of affine theory are:

- *Diffeomorphism:* A function of the form:

$$\mathbf{z} = \varphi(\mathbf{x}), \mathbf{x} \in R^n, \mathbf{z} \in R^n$$

where $\boldsymbol{\varphi}(x)$ is a vector function (field) with the properties (i) $\boldsymbol{\varphi}^{-1}(\boldsymbol{\varphi}(\mathbf{x})) = \mathbf{x}$, $\boldsymbol{\varphi}(\boldsymbol{\varphi}^{-1}(\mathbf{z})) = \mathbf{z}$ for all $\mathbf{x} \in R^n$ and $\mathbf{z} \in R^n$, and (ii) both functions have continuous partial derivatives of any order (smooth functions).

- *Lie derivative:* The function

$$\frac{\partial s(\mathbf{x})}{\partial \mathbf{x}} = \nabla s(\mathbf{x}) = \left[\frac{\partial s(\mathbf{x})}{\partial x_1}, \frac{\partial s(\mathbf{x})}{\partial x_2}, ..., \frac{\partial s(\mathbf{x})}{\partial x_n}\right]$$

where $s(\mathbf{x}) = s(x_1, x_2, ..., x_n) \in R$ is a smooth scalar real-valued function and $\mathbf{f}(\mathbf{x}) \in R^n$ is a vector field, is said to be the Lie derivative of $s(\mathbf{x})$ along the field $\mathbf{f}(\mathbf{x})$.

- *Lie bracket:* It is symbolized by $[\mathbf{f}, \mathbf{g}](\mathbf{x})$ and defined as: $[\mathbf{f}, \mathbf{g}](\mathbf{x}) = (\partial\mathbf{g}/\partial\mathbf{x})\mathbf{f}(\mathbf{x}) - (\partial\mathbf{f}/\partial\mathbf{x})\mathbf{g}(\mathbf{x}), \mathbf{x} \in R^n$, where $\partial\mathbf{f}/\partial\mathbf{x}$ and $\partial\mathbf{g}/\partial\mathbf{x}$ are the Jacobian matrices of the fields $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$.

The second class of mobile robot stabilizing control uses the concept of *invariant manifolds*, and leads to nonlinear controllers directly. A manifold is a topological space which is locally Euclidean, i.e., around every point there is a neighborhood which is topologically the same as the *open unit ball* in R^n.

Invariant Manifold A manifold $M = \{\mathbf{x} \in R^n : \mathbf{s}(\mathbf{x}) = \mathbf{0}\}$, where $\mathbf{s} : R^n \rightarrow R^m$ is a smooth map, is *invariant* for the dynamic system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ if all system trajectories starting in M at $t = t_o$ remain in this manifold for all $t \ge t_0$. This implies that the Lie derivative of $\mathbf{s}$ along the vector field $\mathbf{f}$ is zero, i.e., $\mathbf{L}_{\mathbf{f}}\mathbf{s}(\mathbf{x}) = \mathbf{0}$, for all $\mathbf{x} \in M$.

The concepts of *invariant set* and *invariant manifold* extend the concept of equilibrium point (which is an *invariant monoset*), and enable the construction of Lyapunov functions for nonlinear systems. The invariant set-based Lyapunov stability is based on the *LaSalle* local and global invariant set theorems [75], the *Krasovskii* theorem, and the *Brockett* theorem [76].

4.2.1 Mobile Robot State Feedback Linearization

There are two classes of linearization via state feedback: (i) input-state linearization, and (ii) input-output linearization [75]. The nonholonomic mobile robots can be modeled by the affine system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}, \mathbf{x} \in R^n, \mathbf{u} \in R^m$$

For single-input systems ($u \in R$) there has been derived a generalized linearizing feedback law through the derivation of a generalized controllable canonical form [74, 75]. This control law was adapted to mobile (differential drive, car-like) robots in [77]. Then, the trajectory tracking control was solved for the resulting linearized systems, using conventional linear state-feedback control [19, 77]. Analogous results were obtained in [59, 78].

4.2.2 Mobile Robot Invariant Manifolds-Based Stabilizing Control

This approach leads directly to nonlinear controllers without prior feedback linearization. The two general models that have been used are: (i) the nonholonomic Brockett integrators (simple, double, extended), and (ii) the (2-n) chain models. This approach treats in an elegant way the nonholonomic constraints, and a rich literature exists with a large repertory of different controllers [76, 79–85]. The simplest problem is that of stabilizing control of a unicycle in chained model form (Fig. 1):

$$\dot{z}_1 = u_1, \dot{z}_2 = u_2, \dot{z}_3 = z_2 u_1$$

where $\mathrm{u}_1 = \mathrm{v}_\phi$ and $\mathrm{u}_2 = \mathrm{v}_\mathrm{Q} - \mathrm{z}_3\mathrm{u}_1$.

The problem is to find a static quasi-continuous state feedback control law $\mathbf{u} = \mathbf{u}(\mathbf{z}), \mathbf{u} = [u_1, u_2]^\mathrm{T}, \mathbf{z} = [z_1, z_2, z_3]^\mathrm{T}$. It can be easily verified that the control law:

$$\underline{\mathbf{u}} = [-k_1 z_1, -k_1 z_2], k_1 > 0 \tag{6a}$$

makes the origin $\mathbf{z} = [z_1, z_2, z_3]^\mathrm{T} = [\,0\;\;0\;\;0\,]^\mathrm{T}$ globally asymptotically stable. The resulting closed-loop system is:

$$\dot{\mathbf{z}} = \mathbf{f}(\mathbf{z}), \quad \mathbf{f}(\mathbf{z}) = [-k_1 z_1, -k_1 z_2, -k_1 z_1 z_2]^\mathrm{T} \tag{6b}$$

The manifold:

$$M = \{\mathbf{z} \in R^3 : s(\mathbf{z}) = z_1 z_2 - 2z_3\}$$

is an invariant manifold of the closed-loop system, since: $L_\mathbf{f} s(\mathbf{z}) = \sum_{i=1}^{3} (\partial s/\partial z_i) f_i(z_1, z_2, z_3) = z_2(-k_1 z_1) + z_1(-k_1 z_2) - 2(-k_1 z_1 z_2) = 0$. The time derivative of $s(\mathbf{z})$ along the trajectories of Eq. 6b is found to be:

$$\dot{s}(\mathbf{z}) = z_1 u_2 - z_2 u_1 = 0$$

This means that the trajectories, once on the surface (manifold) M, remain there. Now, since $z_1(t) \to 0$ and $z_2(t) \to 0$ as $t \to \infty$, for any trajectory on M we have $z_3(t) \to 0$ as $t \to \infty$, and so: $[z_1(t), z_2(t), z_3(t)]^\mathrm{T} \to [\,0\;\;0\;\;0\,]^\mathrm{T}$ as $t \to \infty$. We see that M does not depend on k_1. Now, to construct a stabilizing control law which makes M an attractive manifold the control law (6a) must be enhanced so as to satisfy the *attractivity condition*:

$$\begin{aligned} &\text{If } s(\mathbf{z}) < 0, \quad then \quad \dot{s}(\mathbf{z}) > 0 \; \mathbf{z} \in R^3 \\ &\text{If } s(\mathbf{z}) > 0, \quad then \quad \dot{s}(\mathbf{z}) < 0 \; \mathbf{z} \in R^3 \end{aligned} \tag{7a}$$

A possible enhancement is:

$$\mathbf{u} = \begin{bmatrix} -k_1 z_1 - z_2 H(s)/\left(z_1^2 + z_2^2\right) \\ -k_1 z_2 + z_1 H(s)/\left(z_1^2 + z_2^2\right) \end{bmatrix}, z_1^2 + z_2^2 \neq 0 \tag{7b}$$

where the scalar mapping $\mathrm{H}(s)$ is selected so as:

$$sH(s) < 0,$$

to assure the satisfaction of the attractivity condition (7a). A function $\mathrm{H}(s)$ with this property is $H(s) = -k_2 s$. It can be easily verified that the closed-loop system with the controller (7b) and $z_1^2(0) + z_2^2(0) \neq 0$ drives the unicycle to the origin, while avoiding the manifold

$$M^* = \left\{z \in R^3 : z_1^2 + z_2^2 = 0, z_1 z_2 - 2z_3 \neq 0\right\}$$

5 Mobile Robot Adaptive and Robust Control

Adaptive control is suitable for systems that involve slowly varying parameters or uncertainties/disturbances due to load variation, fuel consumption, etc. [86, 87]. *Robust control* is applied in cases where there are strong parameter variations or uncertainties, under the assumption that bounds of these uncertainties are a priori unknown [87]. The adaptive controllers (control laws or control algorithms) improve their performance as the adaptation evolves with time. Robust controllers can face fast disturbances, fast variations, and non-modeled characteristics, and are attempt to keep an acceptable performance right from the beginning. Almost always, the adaptive control techniques require some linear parameterization of the dynamics of the nonlinear system under control.

5.1 Adaptive Control

The two widely used adaptive control methods are:

- The model reference adaptive control (**MRAC**) method.
- The self-tuning control (**STC**) method.

The typical adaptive control method that has been applied to wheeled mobile robots is the **MRAC** method [88–91].

For example, in [90, 91] the problem solved is that of designing an adaptive feedback tracking controller for the differential drive WMR which is described by (τ_R = right-wheel torque, τ_L = left-wheel torque):

$$\dot{v} = (1/mr)(\tau_R + \tau_L) = (1/m)\tau_a$$
$$\dot{\omega} = (2a/Ir)(\tau_R - \tau_L) = (1/I)\tau_b,\ \tau_\alpha = (\tau_R + \tau_L)/\mathrm{r},$$
$$\tau_b = (\tau_R - \tau_L)/\mathrm{r}$$
$$\dot{x} = v\cos\phi,\ \ \dot{y} = v\sin\phi,$$

with state vector $\mathbf{p} = [x, y, \phi]^T$. The tracking control laws, for τ_a and τ_b, are found to be:

$$\tau_a = \hat{m}\dot{v}_d + K_a\tilde{v}_c,\ \ \tau_b = \hat{I}\dot{\omega}_d + K_b\tilde{\omega}_c$$

where $\tilde{v}_c = v_d - v_c$, $\tilde{\omega}_c = \omega_d - \omega_c$, v_d and ω_d are the desired linear and angular velocity of the WMR, and v_c, ω_c are the corresponding reference velocities. The reference models for v_c and ω_c are assumed to be linear, namely: $\dot{v}_r + \beta_{rv}v_r = 0, \dot{\omega}_r + \beta_{r\omega}\omega_r = 0, \beta_{rv} > 0, \beta_{r\omega} > 0$.

Then, selecting a proper candidate Lyapunov function and choosing the parameter dynamics such that $\dot{V} \leq 0$, the parameter adaptation laws are found.

In [92] the adaptive control problem is solved using the input-output linearization for an m-input m-output affine system. This approach is then followed to solve the same problem for the differential drive WMR. In [19] the MRAC problem is solved for an omnidirectional robot using the original Landau method [93].

5.2 Robust Control

A powerful robust control method for nonlinear systems is the *sliding mode control* **(SMC)** method [75], which was applied to mobile robots in [94, 95]. This method was originally applied to a single-input single-output canonical nonlinear model:

$$\dot{x}_1 = x_2,\ \dot{x}_2 = x_3 ... \dot{x}_{n-1} = x_n$$
$$\dot{x}_n = b(\mathbf{x}) + a(\mathbf{x})u + d(t)$$
$$y = x_1$$

where $u(t)$ is the scalar input, $y(t)$ the scalar output, and $d(t)$ a scalar disturbance input.

The nonlinear function $b(\mathbf{x})$ is not exactly known but with some imprecision (error)$|\Delta b(\mathbf{x})|$ which is bounded from above by a known function of x. The tracking error is $\tilde{\mathbf{x}}(t) = \mathbf{x}(t) - \mathbf{x}_d(t)$, where $\mathbf{x} = [y, \dot{y}, \ddot{y}, ..., y^{(n-1)}]^T$ and $\mathbf{x}_d(t)$ is the desired trajectory $\mathbf{x}_d = [y_d, \dot{y}_d, \ddot{y}_d, ..., y_d^{n-1}]^T$. The function $a(\mathbf{x})$ is also known with uncertainty. The solution is based on a time-varying sliding surface S within the state space R^n which is defined as:

$$s(\mathbf{x}, t) = 0,\ \ s(\mathbf{x}, t) = (d/dt + \Lambda)^{n-1}\tilde{\mathbf{x}}(t)$$

where Λ is a positive constant that represents the control signal bandwidth. Under the condition $\mathrm{x_d}(0) = \mathrm{x}_0$, the trajectory tracking problem $x(t) = x_d(t)$ is equivalent to the problem of remaining on the sliding surface $S(t)$ for all t.

Thus, to assure the trajectory tracking $\boldsymbol{x}(\mathrm{t}) \rightarrow \boldsymbol{x}_\mathrm{d}(\mathrm{t})$, the condition $s(\mathbf{x}, t) = 0$ should be maintained, which can be done if $u(t)$ is selected such that outside the surface $S(t)$ the following sliding condition holds:

$$(1/2)ds^2/dt \leq -\gamma|s|,$$

where γ is a positive constant. The solution is found by treating $s^2(\mathbf{x}, t)$ as a Lyapunov function and assuring that it remains a Lyapunov function despite the presence of the disturbance and the model uncertainty.

Let a second-order system,

$$d^2x/dt^2 = b(\mathbf{x}) + u$$

where x is the scalar output, u the scalar control input, and the function $b(\mathbf{x})$ (possibly nonlinear or time-varying) is approximately known with uncertainty bound $\rho_{\max}$, i.e.:

$$|\hat{b} - b| \leq \rho_{\max}$$

The resulting sliding-mode controller is:

$$u = \hat{u} - k\,\mathrm{sgn}\,(s),\quad \hat{u} = -\hat{b} + \ddot{x}_d - \Lambda\dot{\tilde{x}},\quad k = \rho_{\max} + \gamma$$

which results in the desired sliding condition:

$$(1/2)ds^2/dt \leq -\gamma|s|$$

In practice, to avoid chattering of the control u, the *signum* (sgn) function is replaced by the *saturation* (sat) function. The robust control problem was also solved using the Lyapunov stabilization method [75]. The above methods were applied to design robust control (sliding-mode, Lyapunov function-based control) of WMRs in [94, 95]. For example, in the case of the differential drive robot, we have a two dimensional sliding surface:

$$\underline{\mathbf{s}}^T = [s_1, s_2]$$

and so the Lyapunov function should be selected as:

$$V = (1/2)\underline{\mathbf{s}}^T\mathbf{s} = (1/2)s_1^2 + (1/2)s_2^2$$

where $\dot{\mathbf{s}} = -\mathbf{Hs} - \boldsymbol{\Lambda}\mathrm{sgn}\,(\mathbf{s})$, $\mathrm{sgn}(\mathbf{s})^T = [\mathrm{sgn}(s_1), \mathrm{sgn}(s_2)]$ and $\boldsymbol{\Lambda} = diag[\lambda_1, \lambda_2]$. The control law that satisfies the corresponding sliding condition is:

$$u_i = \hat{u}_i - k_i sat(s_i/U)(i = 1, 2)$$

where U is the thickness of the boundary layer, and $\hat{u}_i$ are proper functions determined analogously to the scalar input case. An alternative solution of the WMR sliding model control problem, using polar coordinates to represent the position and orientation of the WMR, was given in [96].

6 Mobile Robot Fuzzy and Neural Control

Fuzzy logic **(FL)** *and neural networks* **(NN)** have found wide application in the identification, planning, and control of mobile robots. Fuzzy logic offers a unified approximate (linguistic) way of drawing conclusions from uncertain data using uncertain rules. NNs offer the possibility of learning and training either autonomously (*unsupervised learning*) or non-autonomously (*supervised learning*), or via evaluation of their performance (*reinforcement learning*) [97–100]. In many practical cases (including mobile robots) use is made of combined (hybrid) *neurofuzzy systems* **(NFSs)** that provide better performance. *Fuzzy sets* were coined by Lofti Zadeh [97], and constitute an extension of the classical concept of (crisp) set which has broken the Aristotelian (true-non true, yes-no) dichotomy.

A fuzzy set A is defined as:

$$A = \{(x, \mu_A(x))|x \in X, \mu_A(x) : X \to [0, 1]\},$$

where X is the so-called *reference superset (or universe of discourse)*, and $\mu_A(x)$ is the *membership function* which takes values in the full closed interval between 0 and 1 (i.e., $0 \leq \mu_A(x) \leq 1$). In the special case where $\mu_A(x)$ takes only the values 0 and 1, then A reduces to a classical (crisp) set, e.g., $A = \{x_1, x_3, x_5\} \subset X = \{x_1, x_2, x_3, x_4, x_5\}$ or $A = \{(x_1, 1), (x_2, 0), (x_3, 1), (x_4, 0), (x_5, 1)\}$.

The three fundamental operations of fuzzy sets (*intersection, union, complement)* are defined as extensions of the respective operations of classical sets. Also, the standard properties of sets (*De Morgan, absorption, associativity, distributivity, idempotency)* hold here. The *fuzzy inference* (or *fuzzy reasoning*) is an extension of the classical inference based on the *modus ponens* and *modus tollens rules*. On the basis of these rules Zadeh has formulated the so-called *'max-min fuzzy composition'* inference rule:

$$B = A \circ R$$

where 'o' denotes the *max-min operation*, and:

$$A = \{(x, \mu_A(x))|x \in X\}, \mathrm{B} = \{(y, \mu_\mathrm{B}(y))\},$$
$$R = \{(x, y), \mu_R(x, y)|(x, y) \in X \times Y\}$$
$$\mu_R(x_i, y_j) = \min\{\mu_A(x_i), \mu_B(y_j)\}(Mamdani\ rule)$$
$$\mu_R(x_i, y_j) = \mu_A(x_i)\mu_\mathrm{B}(y_j)(Larsen\text{'s } rule)$$

The general structure of a *fuzzy logic controller* **(FLC)** involves four units: (i) a fuzzy **IF-THEN** rule base **(FRB)**, (ii) a fuzzy inference mechanism **(FIM)**, (iii) an input fuzzification unit **(IFU)**, and (iv) an output defuzzification unit **(ODU)** (Fig. 2).

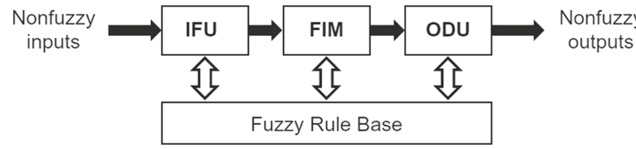

Fig. 2 General structure of a fuzzy logic controller

The two main types of fuzzifier are the *singleton fuzzifier* and the *bell fuzzifier*. The two most popular defuzzification methods are the *center of gravity* **(COG)** method and the *mean of maxima* **(MM)** method.

Neural networks **(NNs)** are large-scale systems that involve a large number of nonlinear processors called *artificial neurons*, described by a *state*, a list of weighted inputs from other neurons, and an *equation* governing their operation [99] (Fig. 3a,b).

In Fig. 3a, b, x_i, i = 1,…,n are the neuron inputs, b is a bias constant, w_i, i = 1,…,n are the synaptic weights, Σ is a summation element, and $f(x) = 1/[1 + \exp(-\beta x)]$ is the log-sigmoid activation function, with β being a constant.

The NN weights adjust their values through learning (minimization of a certain objective function via the gradient or the Newton-Raphson algorithm, and error back propagation). The optimal values of the weights are stored as the strengths of the neurons' interconnections.

The fuzzy systems and the NNs are suitable for systems or processes that cannot be modeled with concise and accurate

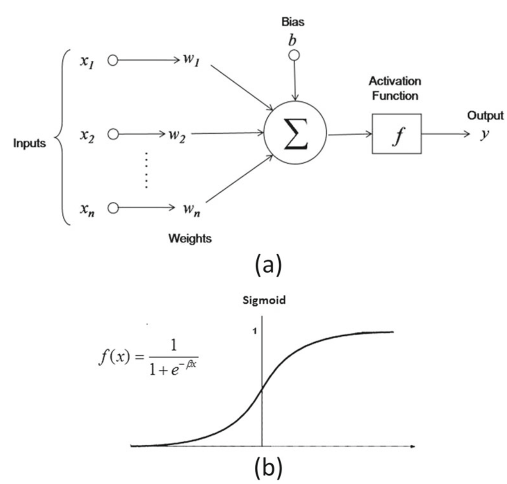

Fig. 3 **a** Model of an artificial neuron, **b** Typical activation function (log-sigmoid function). Other activation functions are: hard-limit (threshold), symmetric hard-limit, and tan-sigmoid (tanh). Source: www.inspirehep.net (/record/1300728/plots)

mathematical models (e.g., pattern recognition, machine vision, control systems, human-based operations, etc.). The three primary features of NNs are: (i) use of large amounts of sensory information, (ii) collective processing capability, and (iii) learning and adaptation capability [99]. The two NNs that were mostly used in decision and control systems are the *multilayer perceptron* **(MLP)** networks, and the *radial basis functions* **(RBF)** networks that have always one layer of hidden nodes (Fig. 4a, b). Other NN models include the *recurrent* (or dynamic) NNs, the *self-organizing maps* (Kohonen NNs), the *convolutional NNs*, the *Hopfield NNs*, and the *Boltzmann machine* [99].

The typical structure of a fuzzy robot control loop is shown in Fig. 5 [19].

The general structure of a neurocontrolled robot with supervised learning is shown in Fig. 6.

Other kinds of neural control involve unsupervised learning and reinforcement learning. The most general type of neurocontrol involves two NNs: the first is used as *feed forward controller* **(FFC)** and the second as *feedback controller* **(FBC)** [101]. This scheme is known as *feedback error learning neurocontroller* **(FELN)** (Fig. 7).

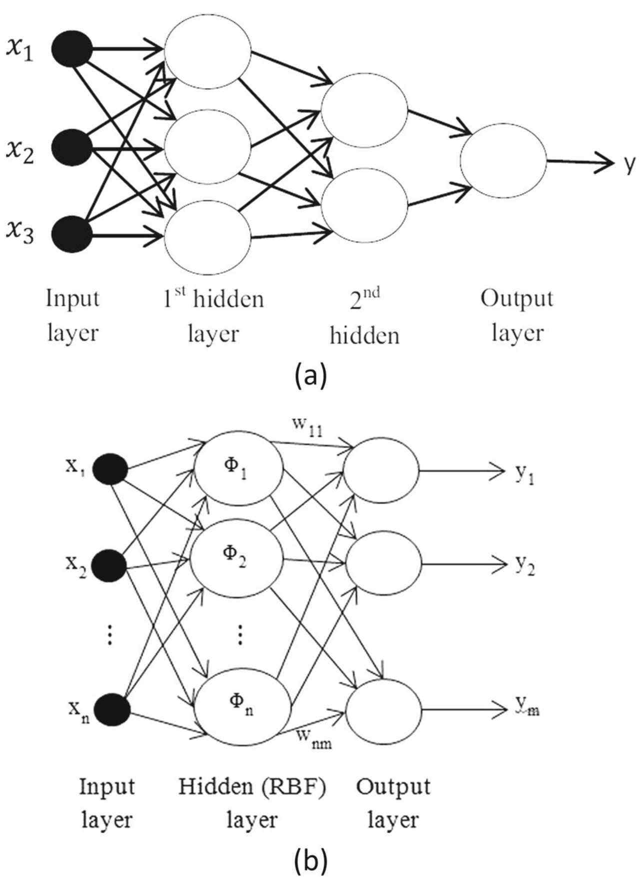

Fig. 4 **a** Multilayer perceptron with two layers of hidden nodes, **b** Radial basis function network (ϕ_i, i = 1, 2, . . ., mare the radial basis functions, typically Gaussian functions). Source: Neurosolutions: What is a neural network?

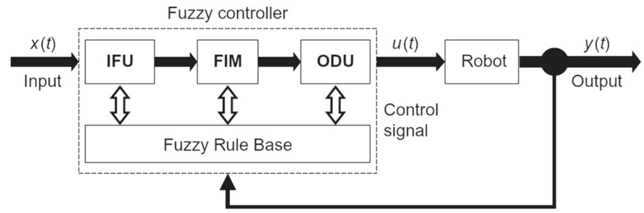

Fig. 5 Fuzzy robot control loop

Representative references where the fuzzy control method was applied are [102–109]. In [102, 103] a direct adaptive fuzzy tracking control scheme is presented, and in [104] a decentralized fuzzy logic control (FLC) scheme for multiple WMRs is described. The structure of the adaptive fuzzy tracking controller of [102, 103] is shown in Fig. 8.

The control system of Fig. 8 involves two loops, namely: (i) the kinematic tracking loop, and (ii) the dynamic control loop, where the dynamic controller is replaced by an FLC which receives crisp values that are fuzzified in the IFU unit, and gives crisp values for the robot inputs (torques) after deffuzification in the ODU unit (Fig. 2). The kinematic controller remains a crisp controller, since it does not involve any unknown (or potentially unknown) parameter. All of its variables are known or measured.

In [109] the fuzzy local path tracking controller for a Dubins car is described. The kinematic model of Dubins car is found from the standard car-like model by omitting the equation for the steering angle velocity. The resulting controller is a *multirate controller*.

In [110] a fuzzy sliding mode control scheme, applied to a WMR, is described. The WMR is assumed to move on a surface $g(x, y, z) = 0$ along a continuously differentiable path $\mathbf{p}(r) = [x(r), y(r), z(r)]$ of the center of gravity of the robot with respect to a world coordinate frame. The sliding mode controller is similar to that described in Section 4.2, and has a diagonal structure, namely:

$$u_{fuzzy} = -K_{fuzzy}(\tilde{x}, \dot{\tilde{x}}, \Lambda)\mathrm{sgn}(s)$$

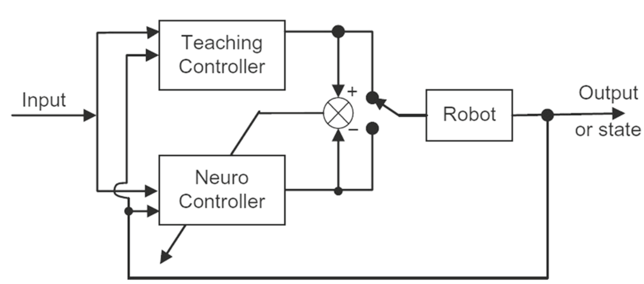

Fig. 6 Structure of a robot controlled by a NN with supervised learning

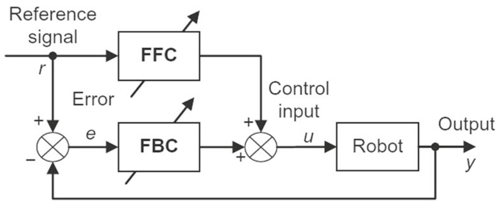

Fig. 7 General structure of FELN

with the condition:

$$K_{fuzzy}(\tilde{x}_1, \dot{\tilde{x}}_1, \Lambda) \leq K_{fuzzy}(\tilde{x}_2, \dot{\tilde{x}}_2, \Lambda)$$

for $|\Lambda\tilde{x}_1+\dot{\tilde{x}}_1| \leq |\Lambda\tilde{x}_2+\dot{\tilde{x}}_2|$. In [110] a *reduced complexity sliding mode fuzzy logic controller* **(RC-SMFLC)** is presented, which can be described as:

$$u_{fuzzy} = -K_{fuzzy}(|s|\text{sgn}(s))$$

where s is the distance from the diagonal. The **RC-SMFLC** is applied in parallel with a standard PD controller, as shown in Fig. 9.

This technique was applied to provide a powerful solution of the parallel car-parking control problem.

In [111], the fuzzy two-step (back-stepping) procedure of [102, 103] was applied using an **MLP** neural network controller in place of a fuzzy controller.

Finally, in [112] the control of a differential drive robot is described which uses *radial-basis networks* for the estimation task.

7 Mobile Robot Vision-Based Control

Vision is a powerful robotic sensor which can be used for environment measurement without physical contact [113]. *Visual robot control or visual servoing* is a feedback control methodology that uses one or more vision sensors (cameras) to control the motion of the robot. Specifically, the control inputs for the robot motors are produced by processing image data (typically, extraction of contours, features, corners, and other visual primitives). In robotic manipulators, the purpose of visual control is to control the pose of the robot's end-effector relative to a target object or a set of target features. In mobile robots, the visual controller's task is to control the vehicle's pose

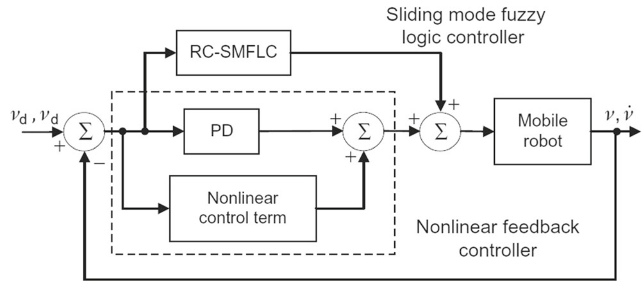

Fig. 9 Structure of hybrid PD-SMFLC control

with respect to some landmarks. Tracking stability can be assured only if the vision sensing delays are sufficiently small and/or the dynamic model of the robot has sufficient accuracy. Over the years many techniques were developed for compensating this delay of the visual system in robot control. A rich literature has been oriented to the control of nonholonomic systems in order to handle various challenging problems associated with vision-based control.

Vision-based robot controllers **(VRCs)** depend on whether the vision system provides set-points as input to the robot joint controllers or computes directly the joint level inputs, and whether the error signal is determined in task space coordinates or directly in terms of image features.

Therefore, VRCs are classified in the following three categories [113, 114, 116, 117].

- *Dynamic look-and-move system*: Here the robot joint controller is eliminated and replaced by a visual servo controller which directly computes the inputs of the joints, and stabilizes the robot using only vision signals. Actually, most implemented VRCs are of the look-and-move type because internal feedback with a high sampling rate provides the visual controller with an accurate axis dynamic model. Also, look-and-move control separates the kinematics singularities of the system from the visual controller, and bypasses the low sampling rates at which the direct visual control can work.
- *Position-based visual robot control* **(PBVRC)**: Here, use is made of features extracted from the image and used together with a geometric model of the target and the available camera model to determine the pose of the

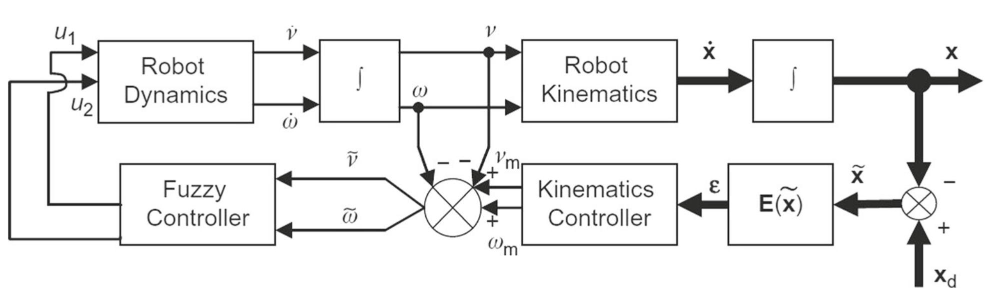

Fig. 8 Structure of overall adaptive fuzzy tracking controller for differential drive robot

target with respect to the camera. Thus, the feedback loop is closed using the error in the estimated pose space.

- *Image-based visual robot control* (**IBVRC**): Here, direct computation of the control signals is performed using the image features. IBVRC reduces the computational time, does not need image interpretation, and eliminates the errors of sensors' modeling and camera calibration. But its implementation is more difficult due to the complex nonlinear dynamics of robots.

The **PBVRC** and **IBVRC** control schemes have the structure shown in Fig. 10 [19].

Two fundamental concepts of VRC are (i) the kinematic transformations, and (ii) the camera visual transformations. Consider a fixed-base robot manipulator working in 3D space. The motion of its end-effector is described in world coordinates x, y, z by a translational velocity v and an angular velocity ω, where:

$$\mathbf{v}(t) = [v_x, v_y, v_z]^{\mathrm{T}}, \boldsymbol{\omega}(t) = [\omega_x, \omega_y, \omega_z]^{\mathrm{T}}$$

Let $\mathbf{p} = [x, y, z]^{\mathrm{T}}$ be a point rigidly attached to end-effector, then:

$$\dot{\mathbf{p}} = \boldsymbol{\omega} \times \mathbf{p} + \mathbf{v}, \tag{8}$$

where $\boldsymbol{\omega} \times \mathbf{p}$ is the cross product of $\boldsymbol{\omega}$ and $\mathbf{p}$, i.e.:

$$\boldsymbol{\omega} \times \mathbf{p} = [\omega_y z - y\omega_z, \omega_z x - z\omega_x, \omega_x y - x\omega_y]^{\mathrm{T}} \tag{9}$$

The combined velocity vector $\mathbf{r} = [\mathbf{v}^{\mathrm{T}}, \boldsymbol{\omega}^{\mathrm{T}}]^{\mathrm{T}}$ is known as *velocity screw* (or *velocity twist*) of the robotic and effector.

In compact form (8) and (9) can be written as:

$$\dot{\mathbf{p}} = \mathbf{J}_0(\mathbf{p})\dot{\mathbf{r}}, \quad \mathbf{J}_0(\mathbf{p}) = [\, \mathbf{I}_{3\times 3} \mid \mathbf{S}(\mathbf{p}) \,] \tag{10}$$

where $\mathbf{S}(p)$ is the skew symmetric matrix [114]:

$$\mathbf{S}(\mathbf{p}) = \begin{bmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{bmatrix} \tag{11}$$

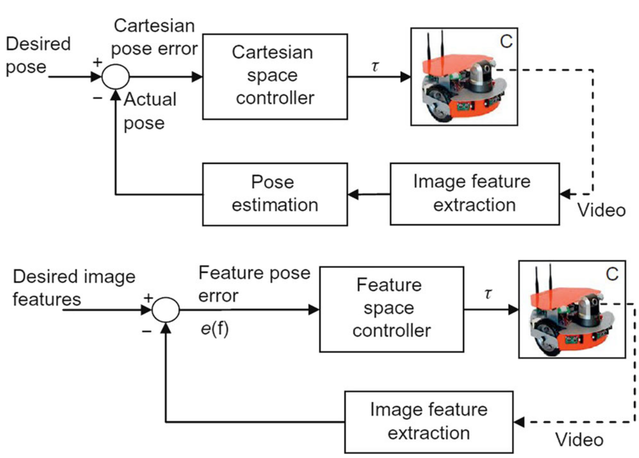

Fig. 10 **a** PBVRC loop (up), **b** IBVRC loop (down)

The platform of a mobile robot is moving with a linear velocity $\mathbf{v}(t) = [v_x, v_y, 0]^{\mathrm{T}}$ and an angular velocity $\boldsymbol{\omega}(t) = [0, 0, \omega]^{\mathrm{T}}$(where $\omega = \omega_z$). Thus, Eq. 9 becomes $\boldsymbol{\omega} \times \mathbf{p} = [-y\omega, x\omega, 0]^{\mathrm{T}}$, and Eq. 8 gives:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} v_x - y\omega \\ v_y + x\omega \end{bmatrix}, \dot{z} = 0 \quad (z = \text{constant} = 0) \tag{12}$$

Equation 12, combined with $\dot{\phi} = \omega$, gives the overall equation:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -y \\ 0 & 1 & x \\ 0 & 0 & 1 \end{bmatrix} \dot{\mathbf{r}}, \quad \dot{\mathbf{r}} = \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \tag{13}$$

If the WMR involves a steering angle ψ, then the corresponding equation $\dot{\psi} = \omega_\psi$ should be added. The camera visual transformations are typical derived using the perspective projection model, in which a point $\mathbf{p} = [x, y, z]^{\mathrm{T}}$ whose coordinates are expressed with respect to the camera coordinate frame $\mathbf{A}_c$, projects onto the image plane point $\mathbf{f} = [x_{im}, y_{im}]^{\mathrm{T}}$ given by:

$$\mathbf{f}(x, y, z) = \begin{bmatrix} x_{im} \\ y_{im} \end{bmatrix} = \frac{l_f}{z} \begin{bmatrix} x \\ y \end{bmatrix} \tag{14}$$

where l_f is the camera's focal length (Fig. 11).

Differentiating Eq. 14 we get:

$$\dot{\mathbf{f}} = \begin{bmatrix} \dot{x}_{im} \\ \dot{y}_{im} \end{bmatrix} = \mathbf{J}_c(x_{im}, y_{im}, l_f)\dot{\mathbf{p}}, \quad \mathbf{J}_c = \begin{bmatrix} l_f/z & 0 & -x_{im}/z \\ 0 & l_f/z & -y_{im}/z \end{bmatrix} \tag{15}$$

Now, combining Eqs. 10, 11, 14, and 15 we find:

$$\begin{aligned} \dot{\mathbf{f}} &= \mathbf{J}_c(x_{im}, y_{im}, z, l_f)\mathbf{J}_0\,(\mathbf{p})\dot{\mathbf{r}} \\ &= \mathbf{J}_{im}(x_{im}, y_{im}, z, l_f)\dot{\mathbf{r}} \end{aligned} \tag{16}$$

where $\mathbf{J}_{im}(x_{im}, y_{im}, z, l_f) = \mathbf{J}_c(x_{im}, y_{im}, z, l_f)\mathbf{J}_0\,(\mathbf{p})$ is called the *image Jacobian* which depends on the distance z of the end effector (or the target point being imaged, in general).

The image Jacobian matrix of a unicycle-type WMR with a pinhole on board camera and a target with three feature points in the camera field of view was derived in [115] and has the form (16). A position-based visual controller for path following by nonholonomic robots is described in [116]. Several image based controllers for mobile robots are presented in [117–121]. An image-based control scheme

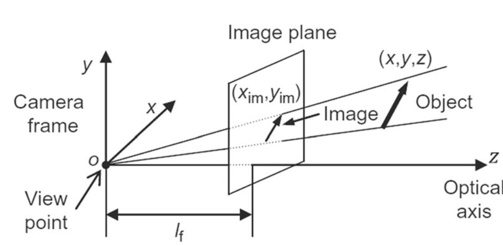

Fig. 11 Geometry of camera lens system

of mobile robots with catadioptric cameras is presented in [122]. An online estimation scheme of the image Jacobian matrix for uncalibrated stereo vision feedback is provided in [123]. A stable vision-based controller for nonholonomic WMRs to keep a landmark in the field of view is provided in [124]. An homography-based mobile robot control scheme with nonholonomic and field-of-view constraints is presented in [125]. The above references describe only a few of the published methods and schemes of mobile robot visual control. More methods can be found in the references cited therein, and in books on mobile robots (e.g., [14, 15, 19]). Two works concerning omnidirectional vision-based mobile robot control are described in [126, 127].

8 Mobile Robot Path and Motion Planning

Robot planning is concerned with the general problem of figuring out how to move to get from one place to another place (*path planning, motion planning)* and how to perform a desired task *(task planning)* [128, 129]. Here we will be concerned with *path planning and motion planning.*

Path planning of mobile robots is one of the basic operations needed to implement the navigation of the robot. These operations are:

- Self-localization.
- Path planning.
- Map-building and map interpretation.

Robot localization provides the answer to the question 'where am I?' The *path planning* operation provides the answer to the question 'how should I get to where I am going?.' Finally, the *map building/interpretation* operation provides the geometric representation of the robot environment in notations suitable for describing location in the robot's reference frame. So far, it seems that there is not a generic method for mobile robot positioning/localization. The specific techniques that exist are divided in two categories:

- Relative localization methods.
- Absolute localization methods.

Relative localization is performed by *odometry* or *inertial navigation*. *Absolute localization* uses active bacons, recognition of artificial landmarks, recognition of natural landmarks, and model matching.

Path planning may be either *local or global*. Local path planning is performed while the robot is moving, taking data from local sensors. In this case, the robot has the ability to generate a new path in response to the changes of the environment. Global path planning can be performed only if the environment (obstacles, etc.) is static and perfectly known to the robot. In this case, the path planning algorithm produces a complete path from the start point to the goal point before the robot starts its motion.

Motion planning is the process of selecting a motion and the corresponding inputs such that all constraints (obstacle avoidance, risk avoidance, etc.) are satisfied. Motion planning can be considered as a set of computations which provide subgoals or set points for the control of the robot. These computations and the resulting plans are based on a suitable model of the robot and the environment in which it is moved. The process by which the robot executes (follows) the planned motion is the control process studied in Sections 4–7.

We recall that the motion of the robot can be described in three different spaces:

- *Task or Cartesian space.*
- *Joints' (motors') space.*
- *Actuators' space.*

A very broad classification of free (obstacle-avoiding) path planning involves three categories, which include six distinct strategies. These are the following:

- *Reactive control* ('Wander' routine, circumnavigation, potential fields, motor schemas).
- *Representational world modeling* (certainty grids).
- *Combinations of both* (vector field histogram).

In many cases, the above techniques do not assure that a path is found that passed obstacles although it exits, and so they need a higher level algorithm to assure that the mobile robot does not end up in the same position over and over again. In practice, it may be sufficient that the robot detects that it is 'stuck' despite the fact that a feasible path way exists, and calls for help. In indoor applications, a maneuver for avoiding an obstacle is a good action. Outdoor situations are more complex, and more advanced perception techniques are needed (e.g., for distinguishing a small tree from an iron pole).

A research topic that received much attention over the years is the *piano-mover's problem*, which is well known to most people that tried a couch or big table through a narrow door. The object has to be tilted and moved around through the narrow door. One of the first research works on this problem is described in Latombe [2].

On the basis of the way the information about the robot's environment is obtained, most of the path planning methods can be classified into two categories:

1. *Model-based approach*
2. *Model-free approach*

In the first category, all the information about the robot's workspace is learned beforehand, and the user specifies the geometric models of objects, and a description of them in terms of these models. In the model-free approach, some of the information about the robot's environment is obtained via sensors (e.g., vision, range, touch sensors). The user has to specify all the robotic motions needed to accomplish a task.

The obstacles that may exist in a robotic work environment are distinguished into *stationary obstacles* and *moving obstacles*. Therefore, two types of path finding problems have to be solved, namely:

- *Path planning among stationary obstacles.*
- *Path planning among moving obstacles.*

The path planning methodology for stationary obstacles is based on the configuration space concept, and is implemented by the so-called road map planning methods. The path planning problem for the case of moving obstacles is decomposed into two subproblems:

- *Plan a path to avoid collision with static obstacles.*
- *Plan the velocity along the path to avoid collision with moving obstacles.*

This combination constitutes the robot motion planning [130–133].

Configuration **q** of a robot is an n-tuple of real numbers that specifies the n parameters required to determine the position of the robot in physical space. The *configuration space* (***CS***) of the robot is the set of values that its configuration **q** may take. The subset of *CS* of configurations that are not in collision with any obstacles that exist in the robot's environment is called the *free configuration space* CS_{free}. In terms of *CS*, the path planning problem of a robot is the problem of finding a path in the free configuration space CS_{free}. Examples of solution of the path planning problem of robotic manipulators via CS_{free} can be found in many references (e.g., [134]).

Typically, CS_{free} path planning methods involve two operations:

- *Collision checking* (i.e., check whether a configuration, or a path between two configurations, lies entirely in CS_{free}).
- *Kinematic steering* (i.e., find a path between two configurations $\mathbf{q}_0$ and $\mathbf{q}_f$ in *CS* that satisfies the kinematic constraints, without taking into account obstacles).

The robot navigation maps that are used to represent the environment can be a continuous geometric description or a decomposition-based geometric map or a topological map. These maps must be converted to discrete maps appropriate for the path algorithm under implementation. This conversion (or decomposition) can be done by four general methodologies, namely [2, 130]:

- *Road maps* (visibility graphs, Voronoi diagrams).
- *Cell decomposition.*
- *Potential fields.*
- *Vector field histograms.*

Representative works on mobile robot path planning and navigation using the above methodologies are the following:

- *Gasparetto* [135]: Solution of 2-D constrained WMR trajectory planning.
- *Hatzivasiliou and Tzafestas* [136]: WMR path planning in structured environment.
- *Garcia and Santos* [137]: WMR navigation with complete coverage in unstructured environment.
- *Safadi* [138]: Local path planning using virtual potential field.
- *Ding, Jiang, Bian and Wang* [139]: Local path planning based on virtual potential field.
- *Koren and Borenstein* [140]: Potential field methods for WMR navigation.
- *Borenstein* [141]: Vector field histogram for fast WMR obstacle avoidance.
- *Wang, Yong and Ang Jr.* [142]: Hybrid global path planning and local WMR navigation in indoor environment.
- *Garrido, Moreno, Blanco and Jurewicz* [143]: WMR path planning using Voronoi diagram and fast marching.
- *Garrido, Moreno and Blanco* [144]: Exploration of a cluttered environment using Voronoi diagrams.
- *Arney* [145]: Autonomous WMR path planning by approximate cell decomposition.
- *Katevas, Tzafestas and Pneumatikatos* [146]: Approximate cell decomposition with local node refinement WMR path planning.
- *Olunloyo and Ayomoh* [147]: WMR navigation using hybrid virtual force field.
- *Katevas, Tzafestas and Matia* [148]: Global and local path strategies for WMR navigation.
- *Katevas and Tzafestas* [149]: The active kinematic histogram method for path planning of non-point nonholonomically constrained mobile robots.
- *Zelinski, Jarvis, Byrne and Yuta* [150]: Planning paths of complete coverage of an unstructured environment by a mobile robot.
- *Zelinski and Yuta* [151]: Unified approach to WMR planning, sensing, and navigation.
- *Choi, Lee, Baek and Oh* [152]: Online complete coverage WMR path planning.

9 Mobile Robot Localization and Mapping

Localization and *mapping* are two of the five basic operations needed for the navigation of a robot. Very broadly, other fundamental capabilities and functions of an integrated robotic system from the task/mission specification to the motion control/task execution (besides path planning) are the following:

- Cognition of the task specification.
- Perception of the environment.
- Control of the robot motion.

The robot must have the ability to perceive the environment via its sensors in order to create the proper data for finding its location (*localization*) and determining how it should go to its destination in the produced map (*path planning*). The desired destination is found by the robot through processing of the desired task/mission command with the help of the cognition process. The path is then provided as input to the robot's motion controller which drives the actuators such that the robot follows the commanded path. The general structure of the interrelations between the above operations has the form of Fig. 12 [19].

The sensors have inaccuracies/uncertainties and so the localization based on data provided by them needs to employ stochastic estimation of the relevant parameters, variables and states. The three primary estimation methods used in mobile robot localization and mapping are [153–155, 168]:

- *Kalman estimation* (filtering, prediction).
- *Bayesian estimation.*
- *Particle filter* **(PF)**.

But other techniques such as fuzzy and neural estimators (approximators) have also been used [23, 170]. The basic concept of localization is the *dead reckoning* (*relative localization*) which can be performed by simple WMR kinematic analysis.

Absolute WMR localization can be performed by:

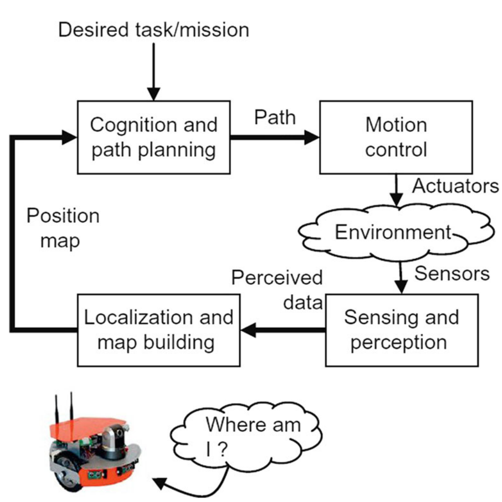

Fig. 12 Structure of an autonomous mobile robot (interconnection of cognition/path planning, perception and control)

- Trilateration.
- Triangulation.
- Map matching.

In general, the sensor imperfections can be grouped in: *sensor noise and sensor aliasing* categories [156–158]. The sensor noise is primarily caused by the environmental variations that cannot be captured by the robot. Examples of this in vision systems are the illumination conditions, the blooming, and the blurring. In sonar systems, if the surface accepting the emitted sound is relatively smooth and angled, much of the signal will be reflected away, failing to produce a return echo. Another source of noise in sonar systems is the use of multiple sonar emitters (16–40 emitters) that are subject to echo interference effects. The second imperfection of robotic sensors is the aliasing, that is, the fact that sensor readings are not unique. In other words, the mapping from the environmental states to the robot's perceptual inputs is many-to-one (not one-to-one). The sensor aliasing implies that (even if no noise exists) the available amount of information is in most cases not sufficient to identify the robot's position from a single sensor reading. The above issues suggest that in practice special sensor signal processing/fusion techniques should be employed to minimize the effect of noise and aliasing, and thus get an accurate estimate of the robot position over time.

Relative localization is performed by dead reckoning, i.e., by the measurement of the movement of a wheeled mobile robot (WMR) between two locations. This is done repeatedly as the robot moves and the movement measurements are added together to form an estimate of the distance traveled from the starting position. Since the individual estimates of the local positions are not exact, the errors are accumulated and the absolute error in the total movement estimate increases with traveled distance. The term dead reckoning comes from the sailing days term '*deduced reckoning*' [156].

For a WMR, the dead reckoning method is *called* 'odometry', and is based on data obtained from incremental wheel encoders [159].

The basic assumption of odometry is that wheel revolutions can be transformed into linear displacements relative to the floor. This assumption is rarely ideally valid because of wheel slippage and other causes. The errors in odometric measurement are distinguished in:

- *Systematic errors* (e.g., due to unequal wheel diameters, misalignment of wheels, actual wheel base is different than nominal wheel base, finite encoder resolution, encoder sampling rate).
- *Non-systematic errors* (e.g., uneven floors, slippery floors, over-acceleration, nonpoint contact with the

floor, skidding/fast turning, internal and external forces).

Systematic errors are cumulative and occur principally in indoor environment. Nonsystematic errors are dominating in outdoor environments.

The problem of placing a WMR at an unknown location in an unknown environment, while the WMR is incrementally building a consistent map of this environment, is known as **SLAM** (*simultaneous localization and mapping*). The SLAM problem was first studied in [171] by Durrant-Whyte who established a statistical basis for describing relationships between landmarks and manipulating geometric uncertainty.

To solve the SLAM problem one needs to use a total (joint) state which incorporates the WMR's pose (position/orientation) and every landmark position. This joint state should be estimated and updated following each landmark observation. For the environment observation/sensing there is available a whole gamma of sensors (sonars, cameras, laser range finders, etc.) [157, 158]. The estimation of the joint state can be performed by an *extended Kalman filter* **(EKF)**, or a *Bayesian estimator* **(BE)**, *or a particle filter/estimator* **(PF)**. EKF is an extension of the Kalman filter that covers nonlinear stochastic models. Bayesian estimators describe the WMR motion and feature observations directly using the underlying probability density functions and Bayes updating law. The PF (also called sequential Monte Carlo estimator) is based on simulation [171]. An outline of the above methods is provided in [19].

Representative references devoted to the mobile robot localization and mapping problem are:

- *Beke and Gurvis* [160]: Mobile robot localization using landmarks.
- *Hu and Gu* [161]: Robot landmark-based localization.
- *Andersen and Concalves* [162]: Vision-based mobile robot localization using triangulation.
- *Castellanos and Tardos* [163]: Multisensor fusion for mobile robot localization and map building.
- *Guivant and Nebot* [164]: Optimal simultaneous localization and mapping.
- *Rekleitis, Dudek and Milios* [165]: Probabilistic cooperative localization and mapping.
- *Bailey and Durrant-Whyte* [166]: Simultaneous localization and mapping (SLAM).
- *Rekleitis, Dudek and Millios* [167]: Multirobot collaboration for robust exploration.
- *Crisan and Doucet* [168]: Convergence of particle filtering methods.
- *Rituerto, Puig and Guerrero* [169]: Mobile robot SLAM with omnidirectional camera.
- *Rigatos and Tzafestas* [170]: Fuzzy modeling and multi-sensor fusion via extended Kalman filtering
- *Kim and Chung* [171]: SLAM with omnidirectional stereo vision sensor.

10 Intelligent Control and Software Architectures for Mobile Robots

Systemic (intelligent control) architectures are used to integrate controllers and high-level functional units for achieving overall intelligent performance of mobile robots. Autonomous mobile robots should be extremely self-reliant to operate in complex, partially unknown environments via control systems assuring in real time that the robot will perform correctly its tasks despite the above constraints. In addition, the software architectures have to face the high degree of heterogeneity among the subsystems involved, and deal with the strict operational requirements posed by the real-time interactions with the robot's environment.

The achievement of autonomous behavior is assured by using techniques of *intelligent control* **(IC)** which started with the development of generic *intelligent control architectures* **(ICAs)**. The principal ICAs are the following:

- *Hierarchical ICA* **(Saridis)** [172, 173]
- *Multiresolutional/nested ICA* **(Meystel)** [174, 175].
- *Reference model ICA* **(Albus)** [176, 177].
- *Behavior-based ICAs, namely: subsumption ICA* **(Brooks)** [178, 179], and *motor schemas ICA* **(Arkin)** [180, 181].
- *Task ICA* **(Simmons)** [182–184].

These architectures were expanded, enriched, or combined over the years in several ways [185]. Most of the software systems and integrated hardware-software systems developed for intelligent mobile robot control follow, in one or the other way, one of these generic architectures or suitable combinations of them.

- The *hierarchical intelligent control architecture has* three main levels, namely (Fig. 13):

  1. *Organization level* which implements the higher-level functions (e.g., learning, decision making).

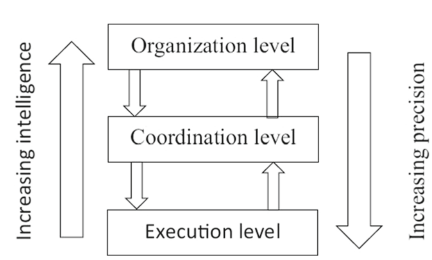

Fig. 13 Saridis' hierarchical ICA

2. *Coordination level* which consists of several coordinators.
3. *Execution level* which involves the actuators, the hardware controllers, and the sensing devices, and executes the action programs issued by the coordination level.

Saridis has developed a complete analytic theory for this architecture, formulating and exploiting the *Principle of Increasing Precision with Decreasing Intelligence* **(PIPDI)** using the information entropy concept [186].

- *The Multiresolutional Intelligent Control Architecture* **(MICA)** was developed by Meystel and first applied to intelligent mobile robots. It follows the commonsense model *Planner-Navigator-Pilot-Execution Controller. The Planner* delivers a rough plan. The Navigator computes a more precise trajectory of the motion to be executed. The Pilot develops online tracking open-loop control. Finally, the Execution Controller executes plans and compensations computed by the planner, the navigator, and the pilot. This scheme is implemented in the form of the so-called *multiresolution 6-box*. Each level contains perception (P), knowledge representation, interpretation and processing (K), and planning and control (P/C) operations.
- The *Reference model architecture* **(RMA)** was developed and expanded by Albus and colleagues, and is suitable for modular expansion. Its control structure involves the following:

  1. Task decomposition.
  2. World modeling.
  3. Sensory processing.
  4. Value judgment.

The various control elements are clustered into computational nodes arranged in hierarchical layers, each one of which has a particular function and a specific timing behavior. The main design issues addressed by RMA are: (i) real-time task and software execution, (ii) smart interface/communication, (iii) information/knowledge base management, and (iv) optimal allocation of resources.

- The *Behavior-Based Control Architectures* **(BBCA)** are based on the concept of *agent* and can be implemented using knowledge-based systems, neural, fuzzy or neuro-fuzzy structures. The two most common behavior-based architectures are the '*subsumption*' architecture developed by Brooks, and *the 'motor schema'* architecture developed by Arkin. The *subsumption architecture* follows the decomposition of the behavior paradigm and was first employed in the autonomous robot Shakey. Complex actions subsume simple behaviors (Fig. 14). The reactions are organized in a hierarchy of levels where each level corresponds to a set of possible behaviors.

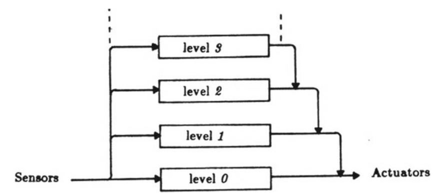

Fig. 14 The subsumption architecture

- The *motor schemas architecture* **(MSA)** was more strongly motivated by biological sciences and uses the theory of schemas originated by Kant. Schemas represent a means by which understanding is able to categorize sensory perception in the process of realizing knowledge of experience. The three representative definitions of the schema concept are [7]:

  1. A pattern of action or a pattern for action.
  2. An adaptive controller (based on an identification procedure).
  3. A perceptual entity that corresponds to a mental entity.

The capabilities of schema-based analysis and design of behavior-base systems are:

1. It can explain motor behaviors in terms of the concurrent control of several different activities.
2. It can store both how to react and how to realize this reaction.
3. It can be used as a distributed model of computation.

  - The *task control architecture* **(TCA)** is a high-level robot operating system with an integrated set of commonly needed mechanisms to support distributed communications, task decomposition, resource management, execution monitoring, and error recovery. A system based on TCA involves a set of specific modules and a general purpose reusable control module. The modules communicate with each other and with the central control by passing messages. The TCA possesses many features of the blackboard **(BB)** architectures [187], but differs from them because (although it maintains control information centrally), the actual data need to solve problems in distributed way among the system's processes.

The *control software architecture* of a semiautonomous/ autonomous mobile robot must meet the following desirable requirements (characteristics) [188, 189].

- *Robot hardware abstraction/portability.*
- *Extendibility/scalability* (capability to add new hardware modules and new software components to the system).

- *Reusability* (e.g., software reuse of components, structure, framework, and software patterns).
- *Repeatability* (which means that running the same program on the same input gives the same result).
- *Run-time overhead* (which is specified by memory and CPU requirements, frequency and end-to-end latency)

The control system of a semiautonomous/autonomous intelligent robot is required to possess the following general features [38].

- *Reactivity* to the environment.
- *Robustness* against imperfect inputs and unexpected events or sudden failures.
- *Multiple sensor integration* to compensate the limited accuracy, reliability and applicability of individual sensors.
- *Modularity* (i.e., the system modules must be able to be separately and incrementally designed, implemented, debugged, and maintained).
- *Expandability* (i.e., the ability to build the system incrementally).
- *Adaptability* (ability to adapt to rapid and unpredictable changes of the world state).

In practice, to keep the overall system complexity at a reasonable level, specific compromises are made such as:

- *Reduction of the autonomy level* and allocation of more difficult tasks or decisions to a human operator.
- *Reduction of the environment complexity* by changing the environment so as to make it more robot friendly (e.g., by introducing landmarks for navigation, etc.).

Very broadly, the majority of mobile robot architectures found in the literature can be classified according to the following three aspects [38].

- The way their modules are interconnected *(hierarchical vs. centralized architectures)* (Fig. 15a).
- The way their modules and the environment communicate *(reactive vs. deliberative control)* (Fig. 15b).
- The function of the modules *(functional vs. behavior systems).*

Three ways, of merging deliberative and reactive behavior were suggested by Arkin, and are shown in Fig. 16 [7]:

- Hierarchical integration of planning and reaction.
- Planning to guide reaction (i.e., allowing planning to choose and set parameters for the reactive control).
- Coupled planning and reacting (each of these two concurrent actions guides the other).

One of the first robotic control schemes that were designed using the hybrid deliberative (hierarchical) and reactive (schema based) architecture is the *autonomous robot architecture* **(AuRA)** developed by Arkin.

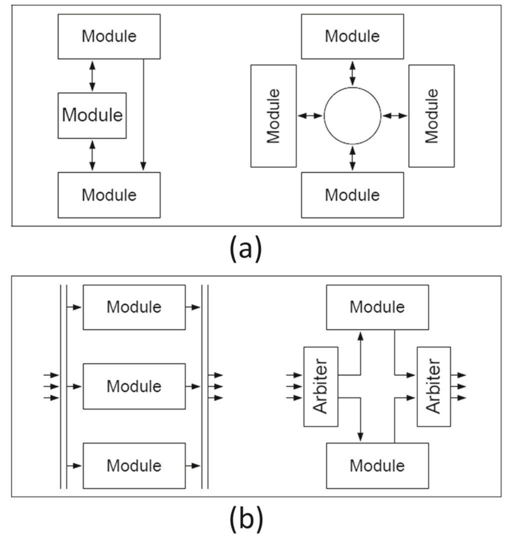

Fig. 15 **a** Hierarchical vs. centralized systems, **b** Reactive vs. deliberative (planning) systems

Two important software architectures are: (i) the **Jde** *(component-oriented)* architecture which uses schemas combined in dynamic hierarchies to unfold the global behavior [190], and (ii) the *layered mobile robot control architecture* which involves four hierarchical layers [191].

Three representative research prototypes of integrated mobile robots are: (i) the **SENARIO** robotic wheel chair [192], (ii) the **KAMRO** (Karlsruhe Autonomous Intelligent Mobile Robot) [193], and (iii) the Munich **ROMAN** intelligent mobile manipulator [194]. The SENARIO mobile robot uses a 'virtually' centralized hierarchical control architecture, and has two alternative operating modes, viz., (i) semi-autonomous mode, and (ii) fully autonomous mode. The KAMRO robot uses a natural language (NL) human-robot interface and performs the following functions: (i) task specification and representation, (ii) execution representation, (iii) explanation of error recovery, and (iv) description and updating of the environment representation.

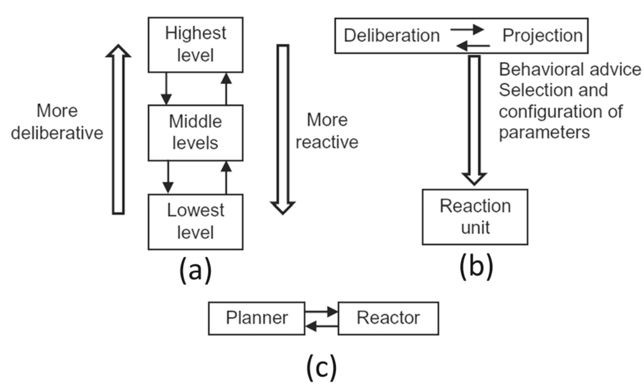

Fig. 16 **a** Hierarchical hybrid deliberative-reactive structure, **b** Planning to guide reaction systems, **c** Coupled planning and reacting scheme

The Munich mobile manipulator ROMAN involves a task planner and coordinator, and a multi-modal robot interface for natural voice/speech based dialog between the human user and the robot, and the vision-based system localization and navigation.

11 Conclusions

Autonomous mobile robots **(AMRs)** need to be able to move purposefully and without human help in real-world environments (i.e., environments that have not been specifically engineered for the robot). Currently, there is still a gap between the available methodology/technology and the new application and market demands which require fully autonomous robots. This gap continuously motivates researchers and practitioners in robotics to develop novel methods and techniques which overcome the large uncertainties that are inherent in the real world environments, by incorporating all necessary details and temporary features, and facing unknown changes of spatial relations between objects or sensory imprecisions and inaccuracies. The outcome of this continuous effort is exhibited by the large amount of published results, over the years, many of which have substantially contributed toward the development and construction of AMRs with higher-level autonomy and social interaction capabilities [195]. The present paper has attempted to provide a global spherical overview of a large number of publications in the field, focusing particularly in the area of control and navigation methodologies. Comprehensible descriptions of the methodologies can be found in the cited books, special issues, and papers. Some further references in the field that deal with mobile robot modeling, exponential stabilization, path planning, path following, trajectory tracking, and navigation problems are [196–212].

References

Authored Books

1. Meystel, A.M.: Autonomous Mobile Robots: Vehicles with Cognitive Control. World Scientific, Singapore (1991)
2. Latombe, J.C.: Robot Motion Planning. Kluwer, Boston (1991)
3. Leonard, J.L.: Directed Sonar Sensing for Mobile Robot Navigation. Springer, Berlin (1992)
4. Jones, J.L., Flynn, A.M., Seiger, B.A.: Mobile Robots: Inspiration to Implementation. Peters A.K., Ltd/CRC Press, New York (1995)
5. Everett, H.R.: Sensors for Mobile Robots: Theory and Applications. Peters A.K. Ltd/CRC Press, New York (1995)
6. Borenstein, J., Everett, H.R., Feng, L.: Navigating Mobile Robots: Systems and Techniques. Peters A.K. Ltd/CRC Press, New York (1996)
7. Arkin, R.C.: Behavior-Based Robotics. MIT Press, Cambridge (1998)
8. Canny, J.: The complexity of Robot Motion Planning. MIT Press, Cambridge (1998)
9. Zhu, X., Kim, Y., Minor, M.A., Qiu, C.: Autonomous Mobile Robots in Unknown Outdoor Environments. CRC Press /Taylor and Francis Group, Boca Raton (2017)
10. Nehmzow, U.: Mobile Robotics: A Practical Introduction. Springer, London (2003)
11. Siegwart, R., Nourbakhsh, I.: Autonomous Mobile Robots. MIT Press, Cambridge (2005)
12. Cuesta, F., Ullero, A.: Intelligent Mobile Robot Navigation. Springer, Berlin (2005)
13. Berns, K., Von Puttkamer, E.: Autonomous Land Vehicles: Steps Towards Service Robots. Springer, Berlin (2009)
14. Dudek, G., Jenkin, M.: Computational Principles of Mobile Robotics. Cambridge University Press, Cambridge (2010)
15. Cook, G.: Mobile Robots: Navigation, Control and Remote Sensing. Wiley, Hobolen (2011)
16. Berry, C.A.: Mobile Robotics for Multidisciplinary Study. Morgan & Claypool, San Rafael (2012)
17. Tiwari, R., Shukla, A., Kala, R.: Intelligent Planning for Mobile Robotics: Algorithmic Approaches. IGI Global, Hershey (2012)
18. Kelly, A.: Mobile Robots: Mathematics, Models, and Methods. Cambridge University Press, Cambridge (2013)
19. Tzafestas, S.G.: Introduction to Mobile Robot Control. Elsevier, Inc., New York (2004)
20. Jaulin, L.: Mobile Robotics. ISTE Press-Elsevier, New York (2017)

Edited Books

21. Patnaik, S., Jain, L.C., Tzafestas, S.G., Resconi, G., Konar, A. (eds.): Innovations in Robot Mobility and Control. Springer, Berlin (2005)
22. Tzafestas, S.G. (ed.): Advances in Intelligent Autonomous Systems. Kluwer, Dordrecht/Boston (1999)
23. Driankov, D., Saffioti, A. (eds.): Fuzzy Logic Techniques for Autonomous Vehicle Navigation. Physica-Verlag, Heidelberg (2011)
24. Azad, A.K.M., Cowan, N.J., Tokhi, M.O., Virk, G.S., Eastman, R.D. (eds.): Adaptive Mobile Robotics. World Scientific, Singapore (2012)
25. Katevas, N. (ed.): Mobile Robotics in Healthcare. IOS Press, Amsterdam (2001)
26. Rembold, U. (ed.): Intelligent Autonomous Systems (IAS-4). IOS Press, Amsterdam (1995)
27. Faruk, E., Ceccareli, M. (eds.): Mobile Robots for Dynamic Environments. ASME Press, New York (2015)
28. Fujimoto, H., Tokhi, M.O., Mochiyama, H., Virk, G.S. (eds.): Emerging trends in mobile robotics. World Scientific, Singapore (2010)
29. Topalov, A.V. (ed.): Recent Advances in Mobile Robotics. InTech, Open Access, Austria (2011)
30. Buchlo, J. (ed.): Mobile Robotics, Moving Intelligence. InTech, Austria (2000)

Survey Papers

31. Tai, L., Liu, M.: Deep learning in mobile robotics: from perception to control systems–a survey of why and why not. J. LATEX Class Files **14**(8), 1–16 (2015)
32. Cao, Y.U., Fukunago, A.S., Kalug, A.B.: Cooperative mobile robotics: antecedents and directions. Auton. Robots **4**, 1–23 (1997)
33. Yamauchi, Y.: A survey of pattern formation of autonomous robots: Asynchrony, obliviousness and visibility. J. Phys.-Conf. Ser. 473012016 (2013)

34. Cunha, J.: A survey on machine learning in mobile robots. Elect. Telecommun. **5**(3), 304–308 (2011)
35. Bostelman, R., Hong, T., Marvel, J.: Survey research for performance measurement of mobile manipulators. J. Stand. Technol. **121**, 342–366 (2016)
36. Yan, Z., Jouandeau, N., Cherif, A.A.: A survey and analysis of multi-robot coordination. Int. J. Adv. Robotic Syst **10**(12), 1–18 (2013)
37. Bandettini, A., Luporini, F., Viglietta, G.: A survey of open problems for mobile robots. Cornell University Library. arXiv:111.2259v1 [cs. RO] (2011)
38. Medeiros, A.A.D.: A survey of control architectures for autonomous mobile robots. J. Brazilian Comput. Soc. **4**(3). (Online) (1998)
39. Chung, T.H., Hollinger, G.A., Isler, V.: Search and pursuit-evasion in mobile robotics. Auton. Robots **31**, 299–316 (2011)
40. David, J., Manivanan, P.V.: Control of Truck-Trailer Mobile Robots: a Survey. Intelligent Service Robotics. Springer, Berlin, (Online) (2014)
41. Ivaldi, S., Padvis, V., Nori, F.: Tools for Dynamics Simulation of Robots: a Survey Based on User Feedback. arXiv:1402.750v1 [cs.RO] (2014)
42. Pomerleau, F., Colas, F., Siegwart, R.: A review of point cloud registration algorithms for mobile robotics. Foundations and Trends in Robotics. Now Publishers **4**(1), 1–104 (2015)

Journal Special Issues

43. Madhavan, R., Scrapper, C., Kleiner, A. (eds.): Special issue: characterizing mobile robot localization and mapping. Auton. Robot. **27**(4), 431–448 (2009)
44. Petrovic, I. (ed.): Special issue: advanced mobile robotics. J. Comput. Inform. Technol. **17**(1), 1–120 (2009)
45. Tzafestas, S.G. (ed.): Special issue: autonomous mobile robots in health care services. J. Intell. Robotic Syst. **22**(3–4), 177–374 (1998)
46. Tzafestas, S.G. (ed.): Special issue: intelligent mobile robots. Adv. Robot. **12**(4), 313–481 (1998)
47. Tzafestas, S.G. (ed.): Special issue: research on autonomous robotic wheelchairs in Europe. IEEE Robot. Autom. Mag. **8**(1), 4–65 (2001)
48. Cetto, J.A., Freze, U., Tenorth, M. (eds.): Special issue: selected papers from 6th European conference on mobile robots. Robot. Auton. Syst. **69**, 1–98 (2015)
49. Lopez-Nicolas, G., Mezouar, Y. (eds.): Special issue: visual control of mobile robots. Robot. Auton. Syst. **62**(11), 1611–1668 (2014)
50. Lee, S., Menegatti, E., Lee (eds.): Special issue: intelligent autonomous systems. Robot. Auton. Syst. **62**(11), 1669–1848 (2014)
51. Nguyen, C.C., Zheng, Y.F. (eds.): Special issue: mobile robots. J. Robot. Syst. (Currently: Journal of Field Robotics) **14**(4), 229–340 (1997)
52. Chrysostomou, D., Goher, K., Muscato, G., Tokhi, U., Virk, G.S. (eds.): Special issue: real-world mobile robot systems. Int. J. Indust. Robot. **44**(4), 393–563 (2017)

Mobile Robot Kinematics

53. Muir, P.F., Neuman, C.P.: Kinematic modeling of wheeled mobile robots. J. Robot. Syst. **4**(2), 281–329 (1987)
54. Killough, S.M., Pin, F.G.: Design of an omnidirectional and holonomic wheeled platform design. In: Proceedings of IEEE Conference on Robotics and Automation, pp. 84–90, Nice (1992)
55. Sidek, N., Sarkar, N.: Dynamic modeling and control of nonholonomic mobile robot with lateral slip. In: Proceedings of Seventh WSEAS International Conference on Signal Processing Robotics and Automation (ISPRA'08), pp. 66–74, Cambridge (2008)
56. Chakraborty, N., Ghosal, A.: Kinematics of wheeled mobile robots on uneven terrain. Mech. Mach. Theor. **39**, 1273–1287 (2004)
57. Ashmore, M., Barnes, N.: Omni-drive robot motion on curved paths: the fastest path between two points is not a straight line. In: Proceedings of 15th Australian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence (AI'02), pp. 225–236. Springer, London (2002)
58. Huang, L., Lim, Y.S., Li, D., Teoh, C.E.L.: Design and analysis of four-wheel omnidirectional mobile robot. In: Proceedings of Second International Conference on Autonomous Robots and Agents. Palmerston North, New Zealand (2004)
59. Phairoh, T., Williamson, K.: Autonomous mobile robots using real time kinematic signal correction and global positioning robot control. In: Proceedings of IAJC-IJME International Conference on Engineering and Technology. Sheraton, Nashville, TN, Paper 087/IT304 (2008)

Mobile Robot Dynamics

60. Sidek, S.N.: Dynamic Modeling and Control of Nonholonomic Wheeled Mobile Robot Subjected to Wheel Slip. PhD Thesis, Vanderbilt University, Nashville (2008)
61. Williams, R.L. ., Carter, B.E., Gallina, P., Rosati, G.: Dynamic model with slip for wheeled omni-directional robots. IEEE Trans. Robot. Autom. **18**(3), 285–293 (2002)
62. Moret, E.N.: Dynamic Modeling and Control of a Car-Like Robot. MSc Thesis, Virginia Polytechnic Institute and State University. Blacksburg (2003)
63. Handy, A., Badreddin, E.: Dynamic modeling of a wheeled mobile for identification, navigation and control. In: Proceedings of IMACS Conference on Modeling and Control of Technological Systems, pp. 119–128, Lille (1992)
64. Watanabe, K., Shiraishi, Y., Tzafestas, S.G., Tang, J., Fukuda, T.: Feedback control of an omnidirectional autonomous platform for mobile service robots. J. Intell. Robot. Syst. **22**, 315–330 (1998)
65. Pin, F.G., Killough, S.M.: A new family of omnidirectional and holonomic wheeled platforms for mobile robots. IEEE Trans. Robot. Autom. **10**(4), 480–489 (1994)
66. Moore, K.L., Flann, N.S.: A six-wheeled omnidirectional autonomous mobile robot. IEEE Control Syst. Mag. **20**(6), 53–66 (2000)

Mobile Robot Control: Standard Controllers

67. Kanayama, Y., Kimura, Y., Noguchi, T.: A stable tracking control method for a nonholonomic mobile robot. IEEE Trans. Robot. Autom. **7**, 1236–1241 (1991)
68. Tian, Y., Sidek, N., Sarkar, N.: Modeling and control of a nonholonomic wheeled mobile robot with wheel slip dynamics. In: Proceedings of IEEE Symposium on Computational Intelligence in Control and Automation, pp. 7–14, Nashville (2009)
69. Chang, C.F., Huang, C.I., Fu, L.C.: Nonlinear control of a wheeled mobile robot with nonholonomic constraints. In: Proceedings of 2004 IEEE International Conference on Systems, Man and Cybernetics, pp. 5404–5409, The Hague (2004)
70. Ashoorizad, M., Barzamini, R., Afshar, A., Zouzdani, J.: Model reference adaptive path following for wheeled mobile robots. In: Proceedings of International Conference on Information and Automation (IEEE/ICIA'06), pp. 289–294, Colombo (2006)
71. Watanabe, K., Shiraishi, Y., Tang, J., Fukuda, T., Tzafestas, S.G.: Autonomous control for an omnidirectional mobile robot with

feedback control system [Chapter 13]. In: Tzafestas, S.G. (ed.) Advances in Intelligent Autonomous Systems, pp. 289–308. Kluwer, Boston/Dordrecht (1999)
72. Lee, S., Youm, Y., Chung, Y.: Control of car-like mobile robots for posture stabilization. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'99), pp. 1745–1750, Kyongju (1999)
73. Panimadai Ramaswamy, S.A., Balakrishnan, S.N.: Formation control of car-like mobile robots: a Lyapunov function based approach. In: Proceedings of 2008 American Control Conference, Seattle (2008)

Mobile Robot Control: Advanced Controllers

74. Isidori, A.: Nonlinear Control Systems: an Introduction. Springer, Berlin/New York (1985)
75. Stotine, J.J., Li, W.: Applied Nonlinear Control. Prentice Hall, Engewood Cliffs (1991)
76. Brockett, R.W.: Asymptotic Stability and Feedback Stabilization: Differential Geometric Control Theory. Birkhauser, Boston (1983)
77. Yun, X., Yamamoto, Y.: On Feedback Linearization of Mobile Robots Technical Report (CIS). University of Pennsylvania, Department of Computer and Information Science (1992)
78. Yang, E., Gu, D., Mita, T., Hu, H.: Nonlinear tracking control of a car-like mobile robot via dynamic feedback linearization. In: Proceedings of Control 2004. University of Bath, UK, [paper 1D-218] (2004)
79. DeVon, D., Brett, T.: Kinematic and dynamic control of a wheeled mobile robot. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2125–2126, San Diego (2007)
80. Astolfi, A.: Exponential stabilization of a wheeled mobile robot via discontinuous control. J. Dyn. Syst. Meas. Control **121**, 121–126 (1999)
81. Reyhanoglu, M.: On the stabilization of a class of nonholonomic systems using invariant manifold technique. In: Proceedings of the 34th IEEE Conference on Decision and Control, pp. 2125–2126, New Orlean (1995)
82. Tayebi, A., Tadijne, M., Rachid, A.: Invariant manifold approach for the stabilization of nonholonomic chained systems: application to a mobile robot. Nonlinear Dyn. **24**, 167–181 (2001)
83. Watanabe, K., Yamamoto, K., Izumi, K., Maeyama, S.: Underactuated control for nonholonomic mobile robots by using double integrator model and invariant manifold theory. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2862–2867, Taipei (2010)
84. Peng, Y., Liu, M., Tang, Z., Xie, S., Luo, J.: Geometry stabilizing control of the extended nonholonomic double integrator. In: Proceedings of IEEE International Conference on Robotics and Biometrics, pp. 926–931. Tianijn (2010)
85. Izumi, K., Watanabe, K.: Switching manifold control for an extended nonholonomic double integrator. In: Proceedings of International Conference on Control and Automation Systems, pp. 896–899. Kintex, Gyeonggi-do (2010)

Mobile Robot Adaptive and Robust Control

86. Marino, R., Tomei, P.: Nonlinear Control Design: Geometric, Adaptive and Robust. Prentice Hall, Upper River (1995)
87. Marino, R.: Adaptive control of nonlinear systems: basic results and applications. IFAC-Rev. Control **21**, 55–66 (1997)
88. Ashoorizad, M., Barzarnimi, R., Afshar, A., Jouzdani, J.: Model reference adaptive path following for wheeled mobile robots. In: Proceedings of International Conference on Information and Automation, pp. 289–294, Colombo, (IEEE/ICIA2006) (2006)
89. Alicja, M.: A new universal adaptive tracking control law for nonholonomic wheeled mobile robots moving in R^3 space. In: Proceedings of IEEE International Conference Robotics and Automation. Detroit (1999)
90. Pourboghrat, F., Karlsson, M.P.: Adaptive control of dynamic mobile robots with nonholonomic constraints. Comput. Elect. Eng. **28**, 241–253 (2002)
91. Gholipour, A., Dehghan, S.M., Ahmadabadi, M.N.: Lyapunov-based tracking control of non-holonomic mobile robot. In: Proceedings of 10th Iranian Conference on Electrical Engineering, vol. 3, pp. 262–269, Tabriz (2002)
92. Fetter Lages, W., Hemerly, E.M.: Adaptive liberalizing control of mobile robots. In: Kopacek, P., Pereira, C.E. (eds.) Intelligent Manufacturing Systems-a Volume from IFAC Workshop, pp. 23–29. Pergamon Press, Gramado (1998)
93. Landau, Y.D.: Adaptive Control: The Model Reference Approach. Marcel Dekker, New York (1979)
94. Zhang, Y., Hong, D., Chung, J.H., Velinsky, S.: Dynamic model-based robust tracking control of differentially steered wheeled mobile robot. In: Proceedings of the American Control Conference, pp. 850–855, Philadelphia (1988)
95. Yang, J.M., Kim, J.K.: Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots. IEEE Trans. Robot. Autom. **15**(3), 578–587 (1999)
96. Chwa, D.: Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates. IEEE Trans. Control Syst. Technol. **12**(4), 637–644 (2004)

Mobile Robot Fuzzy and Neural Control

97. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
98. Kosko, B.: Neural Networks and Fuzzy Systems: a Dynamical System Approach to Machine Intelligence. Prentice Hall, Englewood Cliffs (1992)
99. Haykin, S.: Neural Network: a Comprehensive Foundation. Macmillan College Publishing. Upper Saddle River, New Jersey (1994)
100. Tzafestas, S.G. (ed.): Soft Computing and Control Technology. World Scientific Publishers, Singapore/London (1997)
101. Omatu, S., Khalid, M., Yusof, R.: Neuro-Control and Its Applications. Springer, London/Berlin (1996)
102. Das, T., Narayan, K.I.: Design and implementation of a adaptive fuzzy logic-based controller for wheeled mobile robots. IEEE Trans. Control Syst. Technol. **14**(3), 501–510 (2006)
103. Castillo, O., Aguilar, L.T., Cardenas, S.: Fuzzy logic tracking control for unicycle mobile robots. Eng. Lett. **13**(2), EL.13-2-4:73-77 (2006)
104. Driesen, B.J., Feddema, J.T., Kwok, K.S.: Decentralized fuzzy control of multiple nonholonomic vehicles. J. Intell. Robot. Syst. **26**, 65–78 (1999)
105. Moustris, G., Tzafestas, S.G.: A robust fuzzy logic path tracker for non-holonomic mobile robots. J. Artif. Intell. Tools **14**(6), 935–965 (2005)
106. Tzafestas, S.G., Deliparashos, K.M., Moustris, G.P.: Fuzzy logic path tracking control for autonomous non-holonomic robots: design of system on a chip. Robot. Auton. Syst. **58**, 1017–1027 (2010)
107. Moustris, G.P., Deliparashos, K.M., Tzafestas, S.G.: Feedback equivalence and control of mobile robots through a scalable FPGA architecture. In: Veleninov Topalov, A. (ed.) Recent Advances in Mobile Robotics. InTech (2011). www.interchopen.com/books
108. Rigatos, G.G., Tzafestas, C.S., Tzafestas, S.G.: Mobile robot motion control in partially unknown environments using a sliding-mode fuzzy-logic controller. Robot. Auton. Syst. **33**, 1–11 (2000)

109. Moustris, G., Tzafestas, S.G.: Switching fuzzy tracking control for the Dubins car. Control Eng. Prac. **19**(1), 45–53 (2011)
110. Rigatos, G.G., Tzafestas, S.G., Evangelidis, G.J.: Reactive parking control of nonholonomic vehicles via a fuzzy learning automation. IEE Proc. Control Theory **148**(2), 169–179 (2001)
111. Lewis, F.L., Campos, J., Selmic, R.: Neuro-Fuzzy Control of Industrial Systems with Actuator Nonlinearities. SIAM, Philadelphia (2002)
112. Fierro, R., Lewis, F.L.: Control of a nonholonomic mobile robot: Backstepping kinematics into dynamics. J. Robot Syst. **14**(3), 149–163 (1997)

Mobile Robot Vision-Based Control

113. Haralik, R.M., Shapiro, L.G.: Computer and Robot Vision. Addison Wesley, Reading (1993)
114. Corke, P.: Visual control of robot manipulators: a review. In: Hashimoto, K. (ed.) Visual Serving, pp. 1–31. Word Scientific, Singapore (1993)
115. Tsakiris, D., Samson, C., Rives, P.: Vision-based time–varying stabilization of a mobile manipulator. In: Proceedings of Fourth International Conference on Control, Automation, Robotics and Vision (ICARV'96). Westin Stamford, Singapore (1996)
116. Cherubini, A., Chaumette, F., Oriolo, G.: A position-based visual servoing scheme for following paths with nonholonomic robots. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008), pp. 1648–1654, Nice (2008)
117. Burshka, D., Hager, G.: Vision-based control of mobile robots. In: Proceedings of 2001 IEEE International Conference on Robotics and Automation, pp. 1707–1713, Seoul (2001)
118. Cherubini, A., Chaumette, F., Oriolo, G.: An image-based visual servoing scheme for following paths with nonholonomic mobile robots. In: Proceedings of International Conference on Control Automation, Robotics and Vision, pp. 108–113, Hanoi (2008)
119. Das, A.K., Fierro, R., Kumar, V., Southall, B., Spletzer, J., Taylor, C.J.: Real-time vision-based control of a nonholonomic mobile robot. In: Proceedings of 2001 IEEE International Conference on Robotics and Automation, pp. 1714–1719, Seoul (2001)
120. Carelli, R., Soria, C.M., Morrales, B.: Vision-based tracking control for mobile robots. In: Proceedings of Twelfth International Conference on Advanced Robotics (ICAR'05), pp. 148–152, Seattle (2005)
121. Maya-Mendez, M., Morin, P., Samson, C.: Control of a nonholonomic mobile robot via sensor-based target tracking and pose estimation. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5612–5618, Beijing (2009)
122. Abdelkader, H.H., Mezouar, Y., Andreff, N., Martinet, P.: Image-based control of mobile robot with central catadioptric cameras. In: Proceedings of 2005 IEEE International Conference on Robotics and Automation, pp. 3533–3538, Barcelona (2005)
123. Qian, J., Su, J.: Online estimation of image Jacobian matrix by Kalman-Bucy filter for uncalibrated stereo vision feedback. In: Proceedings of International Conference on Robotics and Automation (ICRA 2002), pp. 562–567 (2002)
124. Gans, N.R., Hutchinson, S.A.: A stable vision-based control scheme for nonholonomic vehicles to keep a landmark in the field of view. In: Proceedings of 2007 International Conference on Robotics and Automation, pp. 2196–2200, Rome (2007)
125. Lopez-Nicolas, G., Gans, N.R., Bhattacharya, S., Sagues, C., Guerrero, J.J., Hutchinson, S.: Homography-based control scheme for mobile robots with nonholonomic and field-of-view constraints. IEEE Trans. Syst. Man Cybern. Cybern **40**(4), 1115–1127 (2010)
126. Okamoto, J. Jr., Grassi, V. Jr.: Visual servo control of a mobile robot using omnidirectional vision. In: Proceedings of Mechatronics Conference 2002, pp. 413–422. University of Twente (2002)
127. Gong, X.: Omnidirectional Vision for an Autonomous Surface Vehicle [Ph.D. dissertation]. Virginia Polytechnic Institute and State University (2008)
128. Russel, S., Norwig, P.: Artificial Intelligence. Prentice Hall, Upper Saddle River (2003)

Mobile Robot Path and Motion Planning

129. Kumar, V., Zefran, M., Ostrowski, J.: Motion planning and control of robots. In: Nof, S. (ed.) Handbook of Industrial Robotics, pp. 295–315. New York, Wiley (1999)
130. Lozano-Perez, T.: Spatial planning: a configuration space approach. IEEE Trans. Comput. **32**(2), 108–120 (1983)
131. Erdmann, M., Lozano-Perez, T.: On multiple moving obstacles. Algorithmica **2**(4), 477–521 (1987)
132. Fugimura, K.: Motion Planning in Dynamic Environments. Springer, Berlin/Tokyo (1991)
133. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. Int. J. Robot. Res. **5**(1), 90–98 (1986)
134. Sheu, P.C.Y., Xue, Q.: Intelligent Robotic Systems. World Scientific Publishers, Singapore/London (1993)
135. Gallina, P., Gasparetto, A.: A technique to analytically formulate and solve the 2-dimensional constrained trajectory planning for a mobile robot. J. Intell. Robot. Syst. **27**(3), 237–262 (2000)
136. Hatzivasiliou, F.V., Tzafestas, S.G.: A path planning method for mobile robots in a structured environment. In: Tzafestas, S.G. (ed.) Robotic Systems: Advanced Techniques and Applications. Kluwer, Dordrecht/Boston (1992)
137. Garcia, E., De Santos, P.G.: Mobile-robot navigation with complete coverage of unstructured environment. Robot. Auton. Syst. **46**, 195–204 (2004)
138. Safadi, H.: Local path planning using virtual potential field. Report COMP 765: Spatial Representation and Mobile Robotics-Project. School of Computer Science, McGill University, Canada (2007)
139. Ding, F.G., Jiang, P., Bian, X.Q., Wang, H.J.: AUV local path planning based on virtual potential field. In: Proceedings of IEEE International Conference on Mechatronics and Automation, vol. 4, pp. 1711–1716, Niagara Falls (2005)
140. Koren, Y., Borenstein, J.: Potential field methods and their inherent liminations for mobile robot navigation. In: Proceedings of IEEE Conference on Robotics and Automation, pp. 1398–1404, Sacramento (2005)
141. Borenstein, J., Koren, Y.: The vector field histogram: fast obstacle avoidance for mobile robots. IEEE J. Robot. Autom. **7**(3), 278–288 (1991)
142. Wang, L.C., Yong, L.S., Ang Jr. M.R.: Hybrid of global path planning and local navigation implemented on a mobile robot in indoor environment. In: Proceedings of IEEE International Symposium on Intelligent Control, pp. 821–826 (2002)
143. Garrido, S., Moreno, L., Blanco, D., Jurewicz, P.: Path planning for mobile robot navigation using Voronoi diagram and fast marching. Int. J. Robot. Autom. **2**(1), 42–64 (2011)
144. Garrido, S., Moreno, L., Blanco, D.: Exploration of a cluttered environment using Voronoi transform and fast marching. Robot. Auton. Syst. **56**(12), 1069–1081 (2008)
145. Arney, T.: An efficient solution to autonomous path planning by approximate cell decomposition. In: Proceedings of International Conference on Information and Automation for Sustainability (ICIAFS 07), pp. 88–93, Colombo (2007)
146. Katevas, N.I., Tzafestas, S.G., Pnevmatikatos, C.G.: The approximate cell decomposition with local node refinement

global path planning method: path nodes refinement and curve parametric interpolation. J. Intell. Robot. Syst. **22**, 289–314 (1998)
147. Olunloyo, V.O.S., Ayomoh, M.K.O.: Autonomous mobile robot navigation using hybrid virtual force field concept. Eur. J. Sci. Res. **31**(2), 204–228 (2009)
148. Katevas, N.I., Tzafestas, S.G., Matia, F.: Global and local strategies for mobile robot navigation. In: Katevas, N. (ed.) Mobile Robotics in Healthcare. IOS Press, Amsterdam (2001)
149. Katevas, N.I., Tzafestas, S.G.: The active kinematic histogram method for path planning of non-point non-holonomically constrained mobile robots. Adv. Robot. **12**(4), 375–395 (1998)
150. Zelinsky, A., Jarvis, R.A., Byrne, J.C., Yuta, S.: Planning paths of complete coverage of an unstructured environment by a mobile robot. In: Proceedings of International Symposium on Advanced Robotics, Tokyo (1993)
151. Zelinsky, A., Yuta, S.: A unified approach to planning, sensing and navigation for mobile robots. In: Proceedings of International Symposium on Experimental Robotics, Tokyo (1993)
152. Choi, Y.H., Lee, T.K., Baek, S.-H., Oh, S.-Y.: Online complete coverage path planning for mobile robots based on linked spiral paths using constrained inverse distance transform. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5688–5712, St. Louis (2009)

Mobile Robot Localization and Mapping

153. Papoulis, A.: Probability, Random Variables and Stochastic Processes. McGraw-Hill, New York (1965)
154. Meditch, J.S.: Stochastic Optimal Linear Estimation and Control. McGraw-Hill, New York (1969)
155. Anderson, B.D.O., Moore, J.B.: Optimal Filtering. Englewood Cliffs, Prentice Hall (1979)
156. Borenstein, J., Everett, H.R., Feng, L.: Navigating Mobile Robots: Sensors and Techniques. A.K. Peters Ltd, Wellesley (1999)
157. Adams, M.D.: Sensor Modeling Design and Data Processing for Automation Navigation. World Scientific, Singapore (1999)
158. Davis, E.R.: Machine Vision: Theory, Algorithms, Practicalities. Morgan Kaufmann, San Francisco (2005)
159. Kleeman, L.: Advanced sonar and odometry error modeling for simultaneous localization and map building. In: Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1866–1871, Sendai (2004)
160. Betke, M., Gurvis, L.: Mobile robot localization using landmarks. IEEE Trans. Robot. Autom. **13**(2), 251–263 (1997)
161. Hu, H., Gu, D.: Landmark-based navigation of industrial mobile robots. Int. J. Indust. Robot. **27**(6), 458–467 (2000)
162. Andersen, C.S., Concalves, J.G.M.: Determining the Pose of a Mobile Robot Using Triangulation: a Vision Based Approach. Technical Report No. 1. European Union Joint Research Center (1995)
163. Castellanos, J.A., Tardos, J.D.: Mobile Robot Localization and Map Building: a Multisensory Fusion Approach. Springer, Berlin (1999)
164. Guivant, J.E., Nebot, E.M.: Optimization of the simultaneous localization and map-building algorithm for real-time implementation. IEEE Trans. Robot. Autom. **17**(3), 242–257 (2001)
165. Rekleitis, I., Dudek, G., Milios, E.: Probabilistic cooperative localization and mapping in practice. Proc. IEEE Robot. Autom. Conf. **2**, 1907–1912 (2003)
166. Bailey, T., Durrant-Whyte, H.: Simultaneous localization and mapping (SLAM). Part I. IEE Robot. Autom. Mag. **13**(2), 99–110 (2006). Part II, ibid, (3) 108–117
167. Rekleitis, I., Dudek, G., Milios, E.: Multirobot collaboration for robust exploration. Ann. Math. Artif. Intell. **31**(1–4), 7–40 (2001)
168. Crisan, D., Doucet, A.: A survey of convergence results on particle filtering methods for practitioners. IEEE Trans. Signal Process. **50**(3), 736–746 (2002)
169. Rituerto, A., Puig, L., Guerrero, J.J.: Visual SLAM with an omnidirectional camera. In: Proceedings of Twentieth International Conference on Pattern Recognition (ICPR'10), pp. 348–351, Istanbul (2010)
170. Rigatos, G.G., Tzafestas, S.G.: Extended Kalman filtering for fuzzy modeling and multisensory fusion. Math. Comput. Model. Dyn. Syst. **13**(3), 251–266 (2007)
171. Kim, J.M., Chung, M.J.: SLAM with omnidirectional stereo vision sensor. In: Proceedings of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 442–447, Las Vegas (2003)

Mobile Robot Intelligent Control and Software Architectures

172. Saridis, G.N.: Toward the realization of intelligent controls. Proc. IEEE **8**, 1115–1133 (1979)
173. Saridis, G.N.: Foundations of intelligent controls. In: Proceedings of IEEE Workshop on Intelligent Control, pp. 23–28, Troy (1985)
174. Meystel, A.M.: Architectures of intelligent control: The science of autonomous intelligence. In: Proceedings of IEEE International Symposium on Intelligent Control, pp. 42–48, Chicago (1993)
175. Meystel, A.M.: Multidirectional hierarchical decision support systems. IEEE Trans. Syst. Man Cybern.-Part C SMR-AR **33**, 86–101 (2003)
176. Albus, J.S., Quintero, R.: Towards a reference model architecture for real-time intelligent control systems. In: Robotics and Manufacturing, vol. 3. ASME, New York (1990)
177. Albus, J.M.: Outline for a theory of intelligence. IEEE Trans. Syst. Man Cybern. **SMC-21**(3), 473–509 (1991)
178. Brooks, R.A.: A robust layered control system for a mobile robot. IEEE J. Robot. Autom. **RA-2**, 14–23 (1986)
179. Brooks, R.A.: Intelligence Without Reason. AI Memo, No. 1293, AI Laboratory, MIT (1991)
180. Arkin, R.C.: Motor schema-based mobile robot navigation. Int. J. Robot Res. **8**(4), 92–112 (1989)
181. Arkin, R.C.: Cooperation without communication: multi-agent schema based robot navigation. J. Robot. Syst. **9**(2), 351–364 (1992)
182. Simmons, R.G.: Concurrent planning and execution for autonomous robots. IEEE Control Syst. Mag. **12**(1), 46–50 (1992)
183. Simmons, R.G.: Monitoring and error recovery for autonomous walking. In: Proceedings of IEEE International Conference on Intelligent Robots and Systems (IROS'92). Raleigh (1992)
184. Coste-Maniere, E., Simmons, R.: Architecture: The backbone of robotic systems. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA'2000), pp. 67–72, San Francisco (2000)
185. Xie, W., Ma, J., Yang, M.: Research on classification of intelligent robotic architecture. J. Comput. **7**(2), 450–457 (2012)
186. Saridis, G.N.: Analytical formulation of the principle of increasing precision and decreasing intelligence for intelligent machines. Automatica **25**(3), 461–467 (1989)
187. Hayes-Roth, B.: A blackboard architecture for control. Artif. Intell. **26**(3), 251–321 (1985)
188. Oreback, A., Christensen, H.L.: Evaluation of architectures for mobile robots. Auton. Robot. **14**, 33–49 (2003)
189. Jawari, D., Deris, S., Mamat, R.: Software reuse for mobile robot applications through analysis patterns. Int. Arab J. Inform. Technol. **4**(3), 220–228 (2007)
190. Canas, J.M., Ruiz-Ayucar, J., Aguero, C., Martin, F.: Jde-neoc: component oriented software architecture for robotics. J. Phys. Agents **1**(1), 1–6 (2007)

191. Kerry, M.: Simplifying Robot Software Design Layer by Layer. National Instruments RTC Magazine. http://rtcmagazine.com/articles/view/102283 (2013)
192. Katevas, N.I., Tzafestas, S.G., Koutsiouris, D.G., Pneumatikatos, C.G.: The SENARIO autonomous navigation system. In: Tzafestas, S.G. (ed.) Proceedings 1st MobiNet Symposium on Mobile Robotics Technology for Health Care Services, pp. 87–89, Athens (1997)
193. Laengle, T., Lueth, T.C., Remboldt, U., Woern, H.: A distributed control architecture for autonomous mobile robots: implementation of the Karlsruhe multi-agent robot architecture. Adv. Robot. **12**(4), 411–431 (1998)
194. Fischer, C., Schmidt, G.: Multi-modal human-robot interface for interaction with a remotely operating mobile service robot (ROMAN). Adv. Robot. **12**(4), 375–395 (1998)
195. Tzafestas, S.G.: Sociorobot World: A guided Tour for All. Springer, Berlin (2016)

Point Stabilization, Path Planning/Following and Trajectory Tracking Control

196. De Wit, C.C., Sordalen, O.: Exponential stabilization of mobile robots with nonholonomic constraints. IEEE Trans. Autom. Control **37**(11), 1791–1797 (1992)
197. Do, K., Jiang, Z., Pan, J.: Simultaneous tracking and stabilization of mobile robots: An adaptive approach. IEEE Trans. Autom. Control **49**(7), 1147–1151 (2004)
198. Jiang, Z.-P., Nijmeijer, H.: Tracking control of mobile robots: a case study in backstepping. Automatica **33**, 1393–1399 (1997)
199. Kostic, D., Adinandra, S., Caarls, J., Nijmeijer, H.: Collision-free tracking control of unicycle mobile robots. In: Proceedings of the IEEE Conference on Decision and Control, pp. 5667–5672 (2009)
200. Kant, K., Zuckler, S.: Toward efficient trajectory planning: the path-velocity decomposition. Int. J. Robot. Res. **5**, 72–89 (1986)
201. Tzafestas, S.G., Stamou, G.: A fuzzy path planning algorithm for autonomous robots moving in an unknown and uncertain environment. In: Proceedings of European Robotics and Intelligent Systems Conference (EURISCON'94), pp. 140–149, Malaga (1994)
202. Katevas, N., Sgouros, N.-M., Tzafestas, S.G., Papakonstantinou, G., Beatie, G., Bishop, G., Tsanakas, P., Koutsouris, D.G.: The autonomous mobile robot SENARIO: a sensor-aided intelligent navigation system for powered wheel chairs. IEEE Robot. Autom. Mag. **4**(4), 60–70 (1997)
203. Samson, C.: Control of chained systems application to path following time-varying point-stabilization of mobile robots. IEEE Trans. Autom. Control **40**(1), 64–77 (1995)
204. Aguilar, A., Hespanha, J.: Trajectory tracking and path following of underactuated autonomous vehicles with parametric modeling uncertainty. IEEE Trans. Autom. Control **52**(8), 1362–1379 (2007)
205. Coelho, P., Nunes, U.: Path following control of mobile robots in the presence of uncertainties. IEEE Trans. Robot. **21**(2), 252–261 (2005)
206. Deliparaschos, K.M., Moustris, G.P., Tzafestas, S.G.: Autonomous SoC for fuzzy robot path tracking. In: Proceedings of EUCA European Control Conference (ECC'2007), pp. 5471–5478. Kos (2007)
207. Skoundrianos, E.N., Tzafestas, S.G.: Mobile robot modeling using local model networks. In: Proceedings of EUCA European Control Conference (ECC'2003), Cambridge (2003)
208. Zavlangas, P.G., Tzafestas, S.G.: Hierarchical motion control system for mobile robot path planning and navigation. In: Proceedings of 2002 Japan-USA Symposium on Flexible Automation (JUSFA'2002). Tokyo (2002)
209. Akkizidis, I.S., Tzafestas, S.G.: Navigation of a mobile robot using fuzzy clustering techniques. In: Proceedings of 11th International Workshop on Robotics in Alpe-Adria-Danube Region. Balatonfared (2002)
210. Tzafestas, C.S., Tzafestas, S.G.: Full-state modeling, motion planning and control of mobile manipulators. Stud. Inf. Control **10**(2), 109–127 (2001)
211. Tzafestas, S.G., Melfi, A., Krikochoritis, A.: Omnidirectional mobile manipulator modeling and control: analysis and simulation. Syst. Anal. Model. Control **40**, 329–364 (2001)
212. Moustris, G.P., Deliparaschos, K.M., Tzafestas, S.G.: Tracking control using the strip-wise affine transformation: an experimental SoC design. In: Proceedings of EUCA European Control Conference. Budapest, Hungary, [paper MoC3.5] (2009)

Spyros G. Tzafestas was born in Corfu, Greece on December 3, 1939 He earned a B.Sc. in Physics (1962) and a Postgraduate Diploma in Electronics and Communications (1963) from Athens University, D.I.C in Electrical Engineering-Automatic Control Systems from Imperial College of London, M. Sc. (Eng.) in Control from London University in 1967, and Ph.D. in Systems and Control from Southampton University of the United Kingdom in March 1969.

He worked at 'Demokritos' Nuclear Research Center as a research leader of the Computer Control and Automation Group of the Computer Science Division from April 1969 to September 1973. From October 1973 to March 1985 he served as a Professor of Control Systems, Director of the Automatic Control Systems Laboratory, at the Electrical Engineering Department of the University of Patras, Patras, Greece. He then gained a position at the School of Electrical and Computer Engineering of the National Technical University of Athens (NTUA) where he served as Professor of Control and Robotics. He founded the Intelligent Robotics and Automation Laboratory of which was the Director up to his retirement on August 31, 2006. From 1999 to 2009 he was the Director of the NTUA Institute of Communication and Computer Systems (ICCS). He holds a D.Sc. (Southampton University, 1978) and two honorary doctorates in engineering (Dr.-Ing. Eh. ,TU Munich, Germany, 1997, and Docteur (Honoris Causa), EC Lille France, 2003).

Over the years he worked in the following areas: 1. Distributed parameter control systems, 2. Computing and micro-processor systems, 3. Multivariable and multidimensional systems, 4. Walsh series-based modeling and control, 5. Robotic systems, 6. AI, KB, and computational intelligence control techniques, 7. Applications to nuclear reactors and manufacturing systems, and 8. Roboethics/robophilosophy. He is known for important contributions in the above areas. He is a LIFE FELLOW of IEEE and a FELLOW of IET (IEE).

He served as the founding editor of the Journal of Intelligent and Robotic Systems (1988-2006), and he is the Chief Editor of the Springer book series on Intelligent Systems, Control and Automation. He has edited 30 research books, 20 Conference Proceedings, and 26 journal special issues in his fields of expertise. He has organized and/or chaired many international conferences (IEEE CDC, EUCA, IMACS, IASTED, etc.). He has served as President of EUCA and Vice President of IMACS, and has been the scientific coordinator of many national and European projects in IT, CIM, Robotics, Intelligent Systems, and Control. He is the author of seven international books and seven Greek books on automation, control, robotics and Artificial/Computational Intelligence. He has received many world-wide scientific awards and his biography is included in more than 20 international biographical volumes. Currently, Dr. Tzafestas continues his scientific work at NTUA as a Professor Emeritus-Senior Research Associate of the School of Electrical and Computer Engineering.