



# Attractor dynamics approach to joint transportation by autonomous robots: theory, implementation and validation on the factory floor

Toni Machado<sup>1</sup> · Tiago Malheiro<sup>1</sup> · Sérgio Monteiro<sup>1</sup> · Wolfram Erlhagen<sup>2</sup> · Estela Bicho<sup>1</sup> 

Received: 1 November 2016 / Accepted: 2 April 2018 / Published online: 12 April 2018  
© Springer Science+Business Media, LLC, part of Springer Nature 2018

## Abstract

This paper shows how *non-linear attractor dynamics* can be used to control teams of two autonomous mobile robots that coordinate their motion in order to transport large payloads in unknown environments, which might change over time and may include narrow passages, corners and sharp U-turns. Each robot generates its collision-free motion online as the sensed information changes. The control architecture for each robot is formalized as a non-linear dynamical system, where by design attractor states, i.e. asymptotically stable states, dominate and evolve over time. Implementation details are provided, and it is further shown that odometry or calibration errors are of no significance. Results demonstrate flexible and stable behavior in different circumstances: when the payload is of different sizes; when the layout of the environment changes from one run to another; when the environment is dynamic—e.g. following moving targets and avoiding moving obstacles; and when abrupt disturbances challenge team behavior during the execution of the joint transportation task.

**Keywords** Joint transportation · Autonomous robots · Mobile robots · Obstacle avoidance · Unknown environments · Attractor dynamics

## 1 Introduction

A large number of scenarios—e.g. warehouses, depots, ports, construction sites, and industrial processes—require a specific object, cargo or payload to be transported from a point A (initial or loading location) to a point B (destination or unloading location). The practicality of using robots to provide assistance in such scenarios has already been demonstrated, and these are often used in process automation. See for instance Widyotriatmo and Hong (2011), where autonomous forklifts aid in the handling of materials in industry, Durrant-Whyte (1996) for a scenario depicting autonomous guided vehicles (AGV) which transport ISO-standard cargo containers in port environments, or Sprunk

et al. (2017) for a focus on the navigation of omnidirectional transport vehicles in industrial environments. However, when the objects to be transported are larger in size, it is preferable to use teams of (smaller and cheaper) robots that jointly carry the payload to its destination, instead of resorting to single large robots. One example in space exploration is a Mars Rover pair, which has cooperatively transported a long payload (Trobi-Ollennu et al. 2002). Another more recent application is that of the car transportation system developed at Tohoku University, which uses teams of two or four mobile robots (Endo et al. 2008; Kashiwazaki et al. 2011). Nevertheless, neither of these two examples has dealt with the problem of obstacle avoidance.

In a joint transportation task, the team of robots should be able to navigate in their environments while still maintaining the relative distance between robots, which is equal to a predefined distance associated to payload sizes. Despite related to the problem of multi-robot formation control (Hess et al. 2009; Monteiro and Bicho 2010; Sabattini et al. 2011; Bayram and Bozma 2016), there is one fundamental difference: the robots are physically connected by the payload. This means that the distance between robots is to be kept within tight boundaries and that, when avoiding obstacles, the complete team should be able to navigate around them.

---

**Electronic supplementary material** The online version of this article (<https://doi.org/10.1007/s10514-018-9729-2>) contains supplementary material, which is available to authorized users.

---

✉ Estela Bicho  
estela.bicho@dei.uminho.pt

<sup>1</sup> Department of Industrial Electronics, University of Minho, Guimaraes, Portugal

<sup>2</sup> Department of Mathematics and Applications, University of Minho, Guimaraes, Portugal

The purpose of this paper is to propose a dynamic control architecture for teams of two autonomous mobile robots engaged in joint transportation tasks, and which are challenged by all of the following functional constraints: (1) transportation by the team of a long object from point *A* to a destination point *B*; (2) avoidance of obstacles (either static or dynamic); (3) navigation in environments unknown *a priori*, and of which the robots have only limited and local sensor information; (4) ability to maneuver in environments that may be cluttered, with narrow corridors and/or tight corners; (5) flexible behavior to enable coping with changes in payload size, as well as in environment layout; and (6) minimal explicit communication.

These constraints are of great importance, since they open up a range of possibilities of cargo displacement in application scenarios where autonomous robots must transport objects of different dimensions, in environments whose layout may change over time and which are co-inhabited by other vehicles or human operators. The projects reported in literature—including our prior work (Soares et al. 2007; Machado et al. 2013)—only focus on some of the abovementioned constraints (c.f. Sect. 2 for further discussion).

In order to deal with the aforementioned functional constraints, we are of the view that a guiding vision for robot controller design can be inspired by the strategy of Human action in joint transportation tasks. In Streuber and Chatziastros (2007), a behavioral experiment was presented, in which several teams of two human subjects were studied while transporting a stretcher. For each experiment, there was a *leader* (in front) and a *helper* (at the rear). The subjects were provided with different types of sensory information feedback: *visual*, which provided local information about stretcher location and orientation; *attitude* (in the case of the *helper*); and *haptic feedback*, i.e. information provided by physical contact with the transported object, allowing for the inference of relative orientation, distance and velocity. After analyzing the human subjects' trajectories, one important conclusion reached was that the rear human subject (i.e. *helper*) tended to align the stretcher with the heading direction of the human subject in front. This behavior benefits maneuverability and collision avoidance in the event of corners. Another interesting result is that the information acquired visually was not essential to the success of the mission. In fact, by not possessing all the visual information available, the team resorted to guiding the stretcher along more conservative and safer trajectories, with the ensuing drawback of sometimes longer trajectories.

Based on these findings of how a Human–Human team performs joint action in transportation tasks, we have proposed a *leader–helper* distributed control architecture in Machado et al. (2013), which as design principle makes the *helper* robot steer in such a way that it tries to make the transported payload align with the *leader's* heading direction.

Regarding sensorial information, one used the relative heading direction and distance between robots (somehow equivalent to haptic feedback), as well as information about the *leader's* heading direction. In the abovementioned paradigm, this can be viewed as reduced visual information. Explicit communication was minimal, since the *helper* did not communicate with the *leader*, and the latter only communicated the heading direction to the former, in the form of a bearing which corresponded to the difference between the *leader's* heading direction and the direction at which the payload was sensed. This means that, in practice, there was no need to keep an estimate of the *leader's* heading direction. Moreover, the responsibility of avoiding collisions of the payload with obstacles was only assigned to the *helper*. In cluttered environments this caused delays in the termination of the transportation task.

Thus, in the work reported here, the *Leader* also assumes an active role by sharing the obligation with the *Helper* of keeping the payload away from obstructions. Besides safer navigation, another important benefit resides in the fact that there is an increase in the team's average speed, which allows for a speedier termination of the task. The extended control architecture developed is structured in terms of elementary behaviors. A particular aspect of our approach lies in the use of the theory of non-linear dynamical systems—that offers essential concepts and tools such as attractor, repeller, stability, bifurcation, trajectory in state space and the like—as a mathematical framework to design, integrate and implement the behaviors required. Specifically, the time course of the control variables are obtained from (fixed point) solutions of dynamical systems. By design, the attractor solutions (asymptotically stable states) dominate these solutions. The advantage is that the overt behavior of each robot is generated as a time series of attractor states, which therefore contribute to making the system robust in the face of perturbations (for basic ideas, see e.g. Schöner et al. 1995; Bicho and Schöner 1997; Bicho et al. 2000; Bicho 2000; Monteiro and Bicho 2010, and related work c.f. Sect. 2.2).

The dynamic control architecture was implemented and tested on the team of mobile robots seen in Fig 1. The desired functionalities mentioned above are observed and documented, with results from simulations and real robot implementations, which also include validation on a factory floor.

The remainder of the paper is structured as follows: Sect. 2 presents related work on the subject of object transportation by teams of autonomous mobile robots, as well as on the dynamical systems approach to robotics. Section 3 describes the robot team. Section 4 presents the dynamical systems that govern the robots' behavior, as well as their implementation on the robots and a stability analysis. Section 5 presents a selection of results, demonstrating that the robot team can cope with the functional requirements imposed for the joint



**Fig. 1** The team of autonomous mobile robots transporting a long box as cargo/payload. Here, the validation scenario is a factory floor

transportation task. The paper ends with a conclusion and an outlook for future work in Sect. 6. Three appendixes further complement the work reported here.

## 2 Related work

### 2.1 Object transportation by teams of mobile robots

In literature on the subject of object transportation by teams of mobile robots, one typically finds four different approaches to physically transport, or move, an object from a point A to point B: (1) pushing the object (e.g. Sudsang 2002; Gross and Dorigo 2009); (2) pulling the object (e.g. Yamashita et al. 1998; Donald et al. 2000; Cheng et al. 2009); (3) placing the object on top of the robots (e.g. Tang et al. 2004; Stouten and Graaf 2004; Loh and Traechtler 2012); (4) using robots with manipulators, or some sort of gripping devices (e.g. Ahmabadadi and Nakano 2001; Tanner et al. 2003).

In the context of this paper, we are especially interested in solutions which might be useful in industrial scenarios, where the objects to be transported might be large and heavy. As such, solutions based on the pushing and pulling of the object are inadequate, while solutions relying on mobile platforms with industrial manipulators, though quite flexible, are not robust enough for heavy loads.

There are two general approaches to control a team of robots transporting a payload: Centralized (e.g. Hashimoto et al. 1993; Tanner et al. 2003; Yamashita et al. 2003; Wada and Torii 2013) and Decentralized control schemes (e.g. Tang et al. 2004; Trebi-Ollennu et al. 2002; Fujii et al. 2007). The centralized approach relies on a unique agent (which may be an outside entity) to compute the teams overall path. This is undertaken either by providing a reference trajectory—which can be virtual—for the transported object, or one for each robot, or each robot’s action. For a recent study, see e.g. Yamaguchi et al. (2015), which proposes a path-following feedback control law for a cooperative transportation system

with two car-like vehicles, in order to follow parametric curve paths at variable velocities.

The limited success of the centralized approach is mainly due to computational and communication costs, particularly in environments that may be dynamic.

In the decentralized approach, each robot is completely autonomous, in that it senses the world, communicates (or not) with the team and computes its own behavior based on limited, local and sensorial information. It has been claimed that the decentralized approach has several advantages over centralized approaches (see e.g. Cao et al. 1997; Parker 2000; Jones and Mataric 2005). However, a major difficulty resides in achieving purposive team behavior in that precise control and coordination of the robots can be extremely difficult. From the perspective of a cooperating robot, the (sensed) environment—consisting of the manipulated object, the other robot(s) and the world scenario (be it static or dynamic)—presents complex behavior.

The decentralized approaches mainly follow *leader–follower*, *master–slave* or *leader–helper* strategies. These can be defined as synonymous of a team, where one robot (the *leader*) heads the way for the team; the remaining robots keep up with the *leader*’s path by taking object sizes into account. Several authors have minimized the coordination effort in decentralized approaches by relying on precomputed trajectories, either for the leader or for the transported object, or for some reference point. In Yang et al. (2004), for instance, a *leader–follower* control architecture, which relies on a planned trajectory of the object in transportation, is presented. Similarly, simulations of two robots transporting a ladder along a corridor with a 90° corner were presented in Asahiro et al. (2001). The approach is distributed in the sense that each robot computes its own trajectory. This is computed at the start of the mission, from the initial to the final position, based on knowledge of the environment. At run-time the robots try to follow these *a priori* defined set-points, with local adjustments to compensate for robot inequalities and obstacle avoidance. In Abou-Samah et al. (2006), the authors developed *leader–follower* and decentralized architectures for two mobile robots equipped with 2 DOF revolute joint manipulators transporting payloads in cooperation. In both approaches, a reference trajectory is required as input. In Yufka et al. (2010), a virtual *leader–follower* formation control approach is used. The transported object is assumed to be the virtual leader, while the carrying robots are the followers, in a formation that is dependent on object shape. A reference trajectory is required for the object acting as virtual leader, and this is assumed to be provided. The consequence is that knowledge of the environment is required and that moving obstacles are not considered. The work in Kim and Minor (2010) also uses the reference trajectory of the transported object, thus presenting the same disadvantages as the previous study. The existence of such preplanned trajectories does

not cope well with the requirements of flexibility, in terms of possible adaptation to online changes in the environments layout, and which might require triggering a simple local obstacle avoidance maneuver or a different path selection.

The expression of awareness of the other robots' behavior can be achieved by explicitly communicated information, or through information obtained by other means (e.g. object linkage). In Pereira et al. (2002), the authors focus on implicit vs explicit communication in a team of two robots carrying an object, and show that the former can be sufficient to accomplish the transportation task. In Trebi-Ollennu et al. (2002), Bouloubasis and McKee (2005) and Tsiamis et al. (2015), one finds different leader–follower control schemes to coordinate the motion of two robots transporting a long payload, and which only rely on implicit communication achieved through the common payload object. However, obstacle avoidance was not addressed in none of these studies.

The *leader–follower* control architecture presented in Takeda et al. (2003) is suitable for dynamical environments from the perspective of the follower robot, and does not make use of explicit communication either. However, it is assumed that the desired trajectory of the transported object needs to be known a priori and must be provided to the leader robot. A similar approach was followed in Yang et al. (2004), where the *follower(s)* (there can be more than one) have to maintain relative position(s) with the object. Obstacle avoidance was introduced by using fuzzy reasoning, yet this was demonstrated only in simple environments (i.e. an environment with two single obstacles of a circular shape). Another *leader–follower* strategy was presented in Fujii et al. (2007). The leader trajectory is estimated from sensed forces in the linkage between robot and payload. The difference here is that, instead of a tight coupling between object and robot, the object is suspended from a hook on the robot, thus configuring a type of loose handling. Nothing in this study is said about obstacle avoidance and maneuvers in tight spaces.

In sum, most of the previous works do not address obstacle avoidance, and none refer/demonstrate the ability to deal with cluttered and unknown environments. This means that the follower not only has to avoid collisions and follow the leader, but it also has to assist in steering the transported object so that maneuvers in tight spaces, which can change over time (e.g. a corridor may become obstructed), are easier and possible. The difficulty of the joint transportation task is aggravated if there is no knowledge of the environment and the robots have to generate their trajectories online. With this purpose in mind, in this paper we present a dynamic control architecture for teams of two robots, which implements a decentralized *leader–helper* strategy enabling joint transportation in narrow spaces and on-line collision avoidance. Contrary to some of the previous related works, no reference trajectories are computed or communicated to the robots. Each robot plans its motion online as sensed informa-

tion changes. Furthermore, the ability to avoid obstacles is provided for either static or moving obstacles. Each robot's control architecture is formalized as a non-linear dynamical system. Important, validation in real environments, which also include a factory floor, are presented.

## 2.2 Dynamical systems approach

The control strategy reported here is based on the so-called *Attractor Dynamics Approach to Behavior Generation*. The basic ideas of this framework—also known as the *Dynamical Systems Approach to Robotics*—were first presented in Schöner et al. (1995) (see also Monteiro and Bicho 2010). Since then, it has been demonstrated in several different environments and different tasks. One of these studies was undertaken by Steinhage (1997), who simulated complex navigation behavior. Initially the approach was applied to control the motion of an autonomous vehicle and was based on 2D representations of the navigational space around the vehicle. Bicho and colleagues extended the approach and have shown that rather limited sensory input is sufficient, and that the computational cost of the dynamical system approach can be kept low (Bicho and Schöner 1997; Bicho 2000; Bicho et al. 2000). In Bicho (2000), which deals with navigation in indoor environments, additional behaviors such as “wall detection” and “wall following” were integrated with obstacle avoidance and target acquisition, thus demonstrating how the behavioral structure and the dynamical system approach can be scaled to fulfill different tasks (see also Althaus et al. 2001). Ellekilde and Perram (2005) have used the approach to plan tool trajectories for industrial robot manipulators, while Iossifidis and Schoener (2006) and Reimann et al. (2010) have applied it to the autonomous avoidance of obstacles and joint-limits in a redundant robot arm. Subsequently, in Monteiro and Bicho (2010), we applied this approach to the (loose) formation control of simulated and real robots. To the best of our knowledge, the first approach to the problem of joint transportation by means of teams of robots using the dynamical systems framework was carried out by our group (Soares et al. 2007; Machado et al. 2013, 2016). This poses new challenges: the coordination and control of teams of autonomous mobile robots transporting a rigid large object without collisions, in real unstructured and dynamic environments, constitutes an extremely difficult endeavor, since the robots are physically tightly coupled.

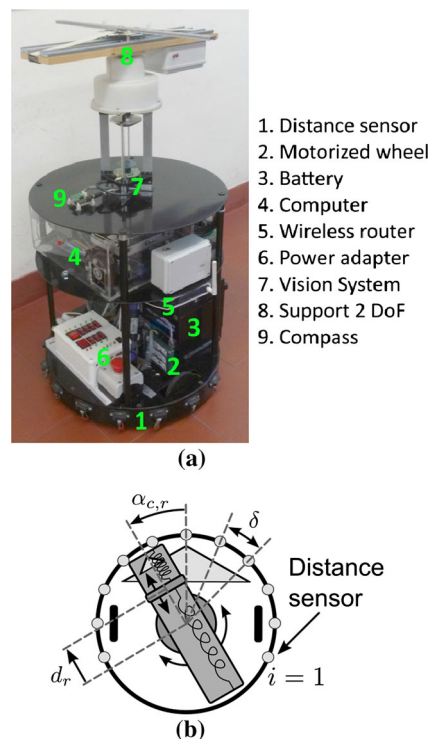
This paper results as a corollary and extension of our research reported in Machado et al. (2016). The following is a new contribution: (1) Lyapunov's stability analysis proves the asymptotic stability of the control laws that govern the robots' behavior. This is a positive feature compared to typical behavior based approaches, since usually, the control law is based only on the superposition of heuristic functions for each behavior, and mathematically proving the asymptotic

stability of the control law is difficult and sometimes impossible. (2) We present and analyze in detail the dynamical systems that generate the robots' behavior. Importantly, we demonstrate relevant issues concerning the implementation on the robots, highlighting that odometry and/or calibration errors are of no significance. (3) We present, analyze and discuss a set of new results, namely: (a) we show that there is nothing in our approach that forces us to work with static targets; the robotic team can also follow a moving target, which can be another robot or a human operator/co-worker; (b) it is shown that the team is able to handle unexpected events during task execution, even though these may cause abrupt perturbations (e.g. a human subject throwing obstacles onto to robots' path); (c) for the very first time, validation on a real factory floor, in a scenario aggregating several challenging situations, e.g. narrow passages with U-shaped turns, and the appearance of a human operator driving a pallet stacker which disturbs the team.

### 3 Robot team

The mission of the team, as depicted in Fig. 1, is to transport long objects from a starting location (the payload loading location) to a destination location (the payload unloading location). Both the loading and unloading actions are not subject of study in this paper. The *Leader* robot holds an extremity of the payload and leads the team from an initial position to a detected target destination while avoiding sensed obstacles. The *Helper* robot holds the other extremity of the object and helps the *Leader* to carry the payload. This implies that the *Helper* has to steer in such a way that it always maintains an appropriate orientation and distance from the *Leader*, which simultaneously subsumes two task constraints: assisting in the transportation of the object and avoiding collisions with obstacles sensed in the meantime. It is important to highlight that the *Leader* is also responsible for keeping the payload away from obstacles (c.f. Sect. 4).

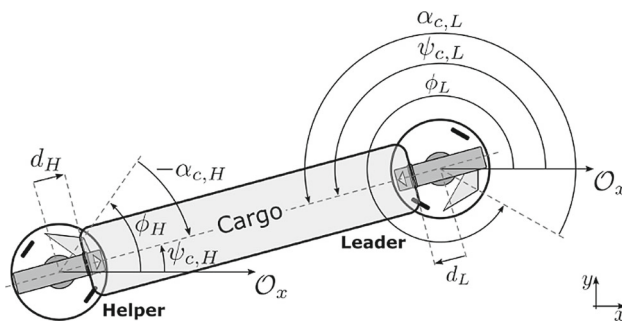
The two robots possess similar characteristics (depicted in Fig. 2a). Each consists of a cylindrical platform with two differential motorized wheels, driven by electronic circuitry that guarantees an accurate control of rotation speed, and two caster wheels for balance. Each robot is equipped with a ring of ( $N$ ) distance sensors—centered on the rotation axis—which are used to measure distance to obstructions at the directions they are pointing towards in space. These sensors are arranged such that their sensitive cones just touch. The distance range is a parameter that can be set. An omnidirectional vision system is used by the *Leader* robot to detect targets identified by specific colors. Image processing is based on color blob extraction. The size and location of the blob, followed by ad-hoc calibration, allows one to compute an estimate of the distance and relative direction of



**Fig. 2** The *payload support* is a 2 DOF system, which is based on a prismatic joint coupled to a rotational joint. The prismatic joint is equipped with returning springs. The *payload support base* is instrumented to output the relative displacement,  $d_r$ , and bearing,  $\alpha_{c,r}$ , of the transported payload, in relation to the robot's center and the heading direction, respectively. The angular displacement between distance sensors is  $\delta$  rad. **a** Photo of one of the robots used in the experiments. **b** Scheme (top view) showing the 2 DOF *payload support base* and the  $N$  ( $= 11$ ) distance sensors

the target. All this is carried out in real time (up to 15 fps) and with a maximum error of 5% in direction and 10% in distance. A compass is used to keep a record of the robot's heading direction over time. This information is only used to monitor and document results; it is not used to control the robot's motion. All programming, control and computation are undertaken onboard. A wireless router enables access in order to facilitate configuration and communication.

Each robot is also equipped with a 2 DOF dedicated support—*payload support base*—mounted on its top and centered to hold the transported object. Each support base is composed of a prismatic joint (to which the object is attached), which is coupled to a free rotational joint centered on the robot (see Fig. 2b). During acceleration and maneuvers, the object is displaced along the prismatic joint, which rotates to accommodate these changes. In order to guarantee that the displacement on each robot's payload support is approximately the same, the prismatic joint is equipped with springs that also try to return the *payload support base* to the center after a transient. A similar solution has been proposed in Hashimoto et al. (1993). The maximum displacement of



**Fig. 3** The leader and helper's heading direction are  $\phi_L$  and  $\phi_H$ , respectively. These are measured in relation to an arbitrary, but fixed, external reference frame (to obtain these angles, one should consider that the reference frame moves with each robot's center but that its rotation is locked, as in Bicho et al. 2000). The support base on each robot ensures the displacement of the payload in relation to the robot's center,  $d_L$  or  $d_H$ , as well as the payload bearing,  $\alpha_{c,L}$  or  $\alpha_{c,H}$ , i.e. the payload angle in relation to the current heading direction of the Leader and Helper robots, respectively

the payload allowed is 20 cm to each side of the robot's center and along the prismatic joint. When above this value, the payload falls down.

A description follows of the control system which governs each robot's behavior.

## 4 The dynamical systems for joint transportation

For the following, please refer to Fig. 3.

In order to model each robot's behavior, we use as control variables the heading direction,  $\phi_r$ , in relation to an arbitrary but fixed reference frame, and path velocity,  $v_r$  ( $r = \{L, H\}$ , where  $L \equiv \text{Leader}$  and  $H \equiv \text{Helper}$ ). Behavior is generated by providing values to these variables, which control the robot's wheels. The time course for each of these variables is obtained from the (fixed point) solutions of dynamical systems. The fixed point attractor solutions (i.e. the asymptotically stable states) dominate these solutions by design. In the present system, the behavioral dynamics of heading direction,  $\phi_r(t)$ , and path velocity,  $v_r(t)$ , are defined by differential equations

$$\frac{d\phi_r}{dt} = f_{des,r}(\phi_r) + F_{obs,r}(\phi_r) + f_{vir,r}(\phi_r) \quad (1)$$

$$\frac{dv_r}{dt} = g_{des,r}(v_r) \quad (2)$$

where the vector fields consist of a number of contributions that express independent task constraints or elementary behaviors. In isolation, each contribution creates an attractor (an asymptotically stable state) or a repeller (an unstable state) of the dynamics of the behavioral/control variable,

with a specified strength and range of attraction or repulsion, respectively.

In the dynamical system defined by (1):  $f_{des,r}(\phi_r)$  models target acquisition behavior by dynamically orienting the robot to a desired target direction, which is achieved by erecting an attractor state at this direction;  $F_{obs,r}(\phi_r)$  models robot obstacle avoidance behavior by erecting repellers, which make the heading direction avoid the undesired directions (e.g. directions at which obstructions are sensed by the distance sensors);  $f_{vir,r}(\phi_r)$  models payload collision avoidance behavior, which keeps the payload safe from collisions. The resulting dynamical system is non-linear and may present multiple stable states (attractors) that change over time, as the robots move and/or the environment changes.

Equation (2) defines a dynamical system which attracts path velocity to a desired value, as is later explained in this section (c.f. Sect. 4.2).

By design, parameters are tuned so that the control variables are mostly very close to one attractor of the resulting dynamics, i.e. the variables follow one of the moving attractors extremely closely. This implies that each robot's behavior is generated as a time series of asymptotically stable states. The fact that only attractor solutions matter can be used to design the layout of attractors and repellers by using the qualitative theory of dynamical systems. Qualitative changes in behavior emerge through bifurcations in the vector fields. The local bifurcation theory helps to design the dynamics, so that these qualitative changes are automatically carried out under the appropriate environmental conditions (e.g. sensory information or shared information within the team of robots).

The next subsections build the individual contributions to the vector fields in (1) and (2) for the Leader and the Helper. One also simultaneously discuss relevant issues concerning the implementation on the robots, highlighting that odometry and/or calibration errors are not a relevant issue here.

### 4.1 Heading direction dynamics

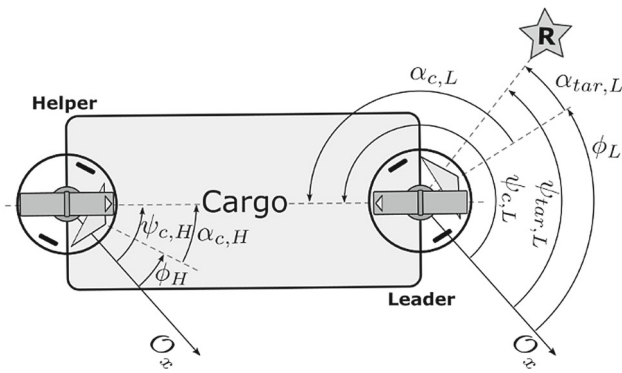
The dynamical system governing the heading direction of each robot  $r \in \{L, H\}$  is given by (1). It is the outcome of the integration of several components, which are specified below.

#### 4.1.1 Target acquisition behavior

Orientation towards a desired heading direction  $\psi_{des,r} \in [0, 2\pi]$  is modeled by

$$f_{des,r}(\phi_r) = -\lambda_{des,r} \sin(\phi_r - \psi_{des,r}) \quad (3)$$

which erects an attractor for  $\phi_r$  at  $\psi_{des,r}$  with an attraction strength defined by  $\lambda_{des,r} (> 0)$ , corresponding to the relax-



**Fig. 4** Desired heading directions for both robots, *Leader* and *Helper*, respectively  $\psi_{tar,L}$  and  $\psi_{c,H}$ . The star represents the mission target. Note that:  $\alpha_{c,r} = \psi_{c,r} - \phi_r$ ; and  $\psi_{c,L} = \psi_{c,H} - \pi$

ation rate to that attractor (i.e. inverse of local relaxation time).

Next, we explain how to compute the attractor value  $\psi_{des,r}$  for the *Leader* and *Helper* (refer to Fig. 4).

**For the Leader (r=L)** The desired heading direction is the direction at which it sees the mission target, i.e.  $\psi_{des,L} = \psi_{tar,L}$ .

**For the Helper (r=H)** The desired heading direction is the direction that aligns itself with the transported object, i.e.  $\psi_{des,H} = \psi_{c,H}$ .

It is important to highlight that, for the implementation of (3) on the robots, there is no need to maintain an estimate of the robots' heading direction. This is because  $(\phi_L - \psi_{tar,L}) = \alpha_{tar,L}$  is the bearing angle at which the *Leader* sees the mission target, which is provided by its vision system, while  $(\phi_H - \psi_{c,H}) = -\alpha_{c,H}$ , is directly given by the payload support base (see Fig. 2b or 3). This implies that calibration and/or odometry errors are of no significance.

### 4.1.2 Robot obstacle avoidance behavior

$F_{obs,r}(\phi_r)$  is given by:

$$F_{obs,r}(\phi_r) = \sum_{i=1}^{N_r} f_{obs,i,r}(\phi_r) \tag{4}$$

where  $N_r$  represents the number of distance sensors, and  $f_{obs,i,r}(\phi_r)$  models a repulsive forcelet, which ensures the collision avoidance of robot  $r$  and an obstruction sensed by its distance sensor  $i$ . Each of these contributions is defined by:

$$f_{obs,i,r}(\phi_r) = \lambda_{obs,i,r}(\phi_r - \psi_{obs,i,r}) \exp\left(-\frac{(\phi_r - \psi_{obs,i,r})^2}{2\sigma_{i,r}^2}\right) \tag{5}$$

It erects a repeller at a direction specified by  $\psi_{obs,i,r}$ , with a repulsion strength defined by  $\lambda_{obs,i,r} (\geq 0)$ , and with  $\sigma_{i,r}$  setting the angular range over which the repeller exerts its repulsive force (see Bicho et al. 2000 for details).

The computation of  $\psi_{obs,i,r}$  is performed as follows: for a single free-moving robot, the directions at which repellers are erected,  $\psi_{obs,i,r}$ , are directly the directions at which the obstructions are sensed (Bicho et al. 2000). This approach is not valid here for the robots in the team because they are linked by the payload that they jointly transport. Hence, the presence of that payload must also be accounted for during each robot's obstacle avoidance behavior.

Consider the situation depicted in Fig. 5a. In this situation, the *leader* is moving away from the obstacle, while the *Helper* has the obstacle on its left. More specifically, its sensors  $i = 7$  and  $i = 8$  are detecting obstructions (see Fig. 5b). If the directions of the repellers were to be the directions at which these sensors are pointing in space, then the *Helper* would turn clockwise and move around the obstacle, keeping it to its left. The problem is that the payload could collide with the obstacle if the obstacle was high enough. To avoid this problem, the *Helper* must remain on the same side of the obstacle as the *Leader* and payload. In this particular scenario, the *Helper* has to turn counterclockwise. This is accomplished by shifting the repellers from the directions relating to sensor  $i = 7$  and  $i = 8$  to sensor sector  $i = 5$ .

With this in mind, for the general case, the direction at which each repeller  $i$  is erected is made:

$$\psi_{obs,i,r} = \phi_r + \Psi_{obs,i,r} \tag{6}$$

where  $\Psi_{obs,i,r}$ , accounts for the fact that a payload is being carried with a bearing angle  $\alpha_{c,r}$  (see Fig. 2b), and is defined by:

$$\Psi_{obs,i,r} = \begin{cases} -\delta, & (\alpha_{c,r} \geq 0) \wedge (0 \leq \varrho_i \delta \leq \alpha_{c,r}) \\ +\delta, & (\alpha_{c,r} < 0) \wedge (\alpha_{c,r} \leq \varrho_i \delta \leq 0) \\ \varrho_i \delta, & \text{otherwise} \end{cases} \tag{7}$$

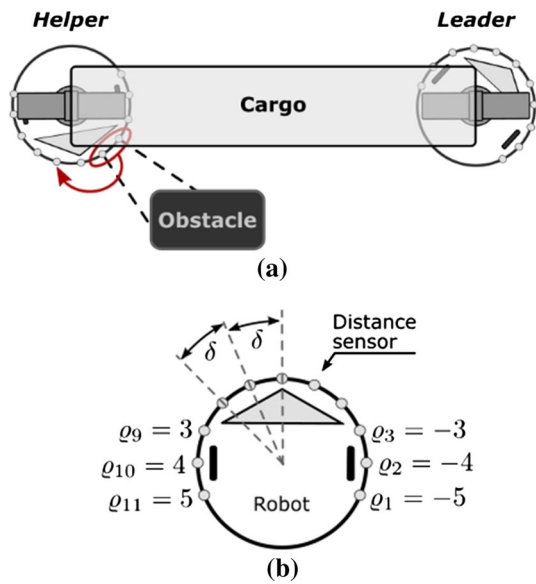
$, \forall \alpha_{c,r} \in [-\pi, \pi].$

Here,  $\delta$  represents the angular distance between the center of two consecutive distance sensors and  $\varrho_i = \{-(N - 1)/2, \dots, 0, \dots, (N - 1)/2\}$  constitutes the sensor's position in relation to the robot's heading direction (see Fig. 5b).

The repulsion strength,  $\lambda_{obs,i,r}$ , of each contribution  $f_{obs,i,r}$  is a decreasing function of the distance sensed,  $d_{i,r}$ :

$$\lambda_{obs,i,r} = \beta_{1,r} \exp\left(-\frac{d_{i,r}}{\beta_{2,r}}\right) \tag{8}$$

The parameter  $\beta_{1,r} (> 0)$  is the maximal strength of repulsion, while  $\beta_{2,r} (> 0)$  fixes the distance over which the repulsion contribution decays. The larger the sensed distance



**Fig. 5** Obstacles detected between the *helper*'s heading direction and payload movement direction are moved to a *strategic* position, in order to avoid payload collision with obstacles. **a** Obstacle contribution shifted, **b** positions of the distance sensors (when  $N = 11$ )

$d_{i,r}$  to any obstruction detected by the distance sensor  $i$ , the weaker the repulsion from the direction  $\psi_{obs,i,r}$ . For additional details and how to compute  $\sigma_{i,r}$  in (5) please see Bicho et al. (2000).

Finally, and very importantly, with regard to the implementation on the robots, note that if one replaces (6) into (5) one obtains:

$$f_{obs,i,r}(\phi_r) = -\lambda_{obs,i,r} \Psi_{obs,i} \exp\left(-\frac{\Psi_{obs,i,r}^2}{2\sigma_{i,r}^2}\right) \quad (9)$$

which means, once more, that there is no need whatsoever to maintain an estimate of the robots' heading directions,  $\phi_r$ , in the implementation. Again, this implies that calibration and/or odometry errors do not matter.

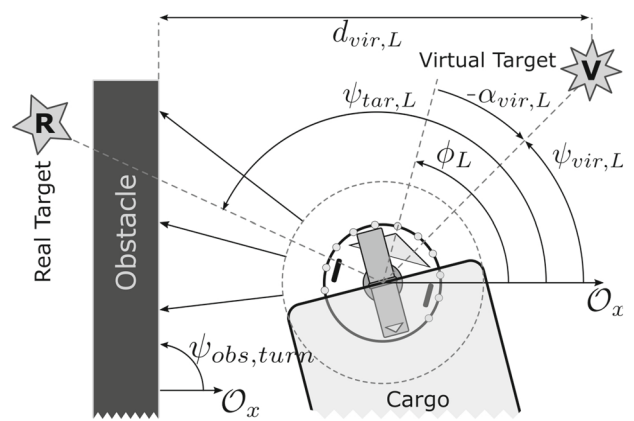
### 4.1.3 Payload collision avoidance behavior

The behavior for keeping the payload safe from collisions is modeled by an attractive force:

$$f_{vir,r}(\phi_r) = -\lambda_{vir,r} \sin(\phi_r - \psi_{vir,r}) \quad (10)$$

where  $\lambda_{vir,r} (> 0)$  is the relaxation rate (strength of attraction) to the attractor erected at the heading direction given by  $\psi_{vir,r} \in [0, 2\pi]$ . This specifies a virtual target, which ensures that the payload is moved away from sensed obstructions.

The parameters for this behavior are set differently for the *Leader* and the *Helper*.



**Fig. 6** Setting the virtual target for the *Leader*, for the *payload collision avoidance* behavior

**For the Leader ( $r = L$ )** In this case, the behavior specified by  $f_{vir,L}(\phi_L)$  is responsible for moving the *Leader* (and hence the payload) away from obstructions that appear on the side to which the *Leader* is turning (at a distance specified by  $d_{vir,L}$ ), with the purpose of moving the payload away from obstacles, thus facilitating its partner's role (see Fig. 6).

$f_{des,L}(\phi_L)$  and  $f_{obs,i,L}(\phi_L)$  (given by (3) and (5) respectively) can be used to indicate if the target and sensed obstructions are to the right or to the left of the *Leader* robot (regarding its heading direction  $\phi_L$ ). Negative values of  $f_{des,L}(\phi_L)$  indicate that the desired real target position lies to the right, while positive values mean that the target lies to the left. Conversely, negative values of  $f_{obs,i,L}(\phi_L)$  indicate that an obstruction sensed by sensor  $i$  rests on the left side of the robot, while positive values indicate the the obstruction is on the right side.

With this in mind, the heading direction to the virtual target is given by:

$$\psi_{vir,L} = \psi_{obs,turn} + \Upsilon_{vir,L} \quad (11)$$

with

$$\Upsilon_{vir,L} = \begin{cases} -\Psi_{vir,L}, & \alpha_{tar,L} > \Psi_{thres} \wedge \\ & \wedge f_{obs,Sleft,L} \neq 0 \\ +\Psi_{vir,L}, & \alpha_{tar,L} < -\Psi_{thres} \wedge \\ & \wedge f_{obs,Sright,L} \neq 0 \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

where  $\alpha_{tar,L} = -(\phi_L - \psi_{tar,L}) \in [-\pi, \pi]$ . Note that  $\alpha_{tar,L}$  can be given directly by the *Leader*'s vision system, and hence there is no need to maintain an estimate of the *Leader*'s heading direction,  $\phi_L$ , in the implementation.  $\psi_{obs,turn}$  is the orientation of the sensed obstructions on the side to which the robot is turning (relating to reference  $O_x$ ).



This value is computed from obstacle sensor measurements:

$$\psi_{obs,turn} = \phi_L + \alpha_{obs,turn} \tag{13}$$

where  $\alpha_{obs,turn}$  is the angle created between the obstruction and the heading direction.

$\Psi_{vir,L}$ , in 12, is a parameter that sets the amount of deviation in relation to the sensed obstruction.  $\Psi_{thres}$  defines the threshold from which one considers the current heading direction to be different from the desired real heading direction (real target).  $f_{obs,S_{left},L}$  and  $f_{obs,S_{right},L}$  are the contributions resulting from sensor sectors  $i = S_{left}$  and  $i = S_{right}$ , respectively, given by:

$$f_{obs,S_{side},L} = \sum_{i=S_{side}} f_{obs,i,L} \tag{14}$$

where  $f_{obs,i,L}$  is the contribution of the *Leader* robot’s distance sensor  $i$ .

For the *Helper* ( $r=H$ ) The behavior corresponding to  $f_{vir,H}(\phi_H)$  is responsible for aligning the direction of the transported object,  $\psi_{c,H}$ , with the *Leader*’s heading direction,  $\phi_L$ , causing  $(\phi_L - \psi_{c,H}) \rightarrow 0$ . It does so by erecting an attractor in the direction of a virtual target,  $\psi_{vir,H}$  (see Fig. 7), given by:

$$\psi_{vir,H} = \psi_{c,H} + \Psi_{vir,H} \tag{15}$$

with  $\psi_{vir,H} \in [0, 2\pi]$  and  $\Psi_{vir,H} = \pm\pi/2$  depending on the *Leader* turning left or right:

$$\Psi_{vir,H} = \begin{cases} -\pi/2, & \phi_L - \psi_{c,H} < 0 \\ \pi/2, & \phi_L - \psi_{c,H} \geq 0 \end{cases} \tag{16}$$

where  $(\phi_L - \psi_{c,H}) = (\pi - \alpha_{c,L})$ , as illustrated in Fig. 4, signals whether the *Leader* is turning left ( $\phi_L - \psi_{c,H} < 0$ ) or right ( $\phi_L - \psi_{c,H} \geq 0$ ). Note that the  $\alpha_{c,L}$  value constitutes the unique information which is explicitly communicated between the robots, and it is communicated from the *Leader* to the *Helper*.

Finally and very importantly, with regard to implementation on the robot, note that if (15) is replaced in (10) one gets:

$$f_{vir,H}(\phi_H) = \lambda_{vir,H} \sin(\alpha_{c,H} - \Psi_{vir,H}) \tag{17}$$

because  $\phi_H - \psi_{vir,H} = \alpha_{c,H} - \Psi_{vir,H}$  (see Figs. 2b or 3).

This implies that, in the implementation of this behavior there is no need whatsoever to maintain an estimate of the *Helper*’s or *Leader*’s heading direction. This again implies that calibration and/or odometry errors are of no significance.

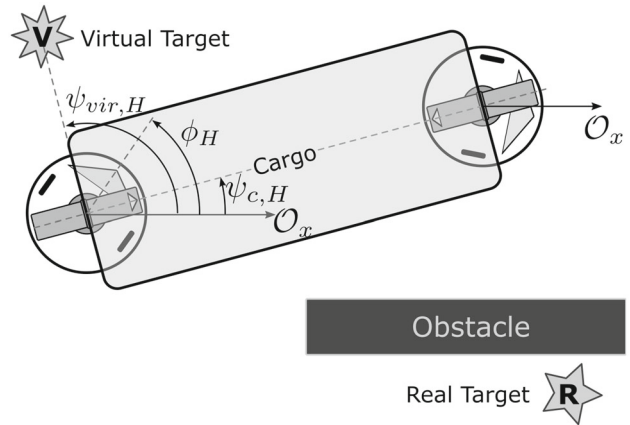


Fig. 7 Setting the virtual target for the *Helper*, for the *payload collision avoidance behavior*

#### 4.1.4 Behavioral integration of $f_{des,r}$ and $f_{vir,r}$

Since  $f_{des,r}(\phi_r)$  and  $f_{vir,r}$ , with  $r \in \{L, H\}$ , are defined using the sinus function, and that the addition of two sinuses of the same frequency constitutes another sinus, then, in order to ease the proper balancing between these two contributions with regard to obstacle avoidance behavior, they are merged together in a single contribution:

$$f_{desvir,r}(\phi_r) = f_{des,r}(\phi_r) + f_{vir,r}(\phi_r) = -\lambda_{desvir,r} \cdot \sin(\phi_r - \psi_{desvir,r}) \tag{18}$$

where  $\psi_{desvir,r}$  is the resultant attractor, which should be located between the directions represented by  $\psi_{des,r}$  and  $\psi_{vir,r}$  ( $\psi_{desvir,r} \in [\psi_{des,r}, \psi_{vir,r}]$ ), and  $\lambda_{desvir,r}$  is the strength of attraction (relaxation rate) to it.

The resultant attractor  $\psi_{desvir,r}$  is computed differently for the *Leader* and for the *Helper*.

For the *Leader* ( $r=L$ ) The direction of the resultant attractor is defined by the sigmoid function:

$$\psi_{desvir,L} = \psi_{vir,L} + \frac{\psi_{tar,L} - \psi_{vir,L}}{1 + \exp\left[-\mu_1 (d_{obs,turn} - d_{vir,L}) + \ln\left(\frac{\psi_{tar,L} - \psi_{obs,turn}}{\psi_{obs,turn} - \psi_{vir,L}}\right)\right]} \tag{19}$$

with  $(\psi_{tar,L} - \psi_{vir,L}) \in [0, 2\pi]$ .  $\mu_1$  being the slope of the sigmoid, i.e. tells us how fast the robot moves to a distance  $d_{vir,L}$  from the sensed obstructions,  $\psi_{obs,turn}$  is the average orientation of the sensed obstructions, and  $d_{obs,turn}$  is the minimum distance to those obstructions. The parameter  $d_{vir,L}$  is used to keep the cargo away from the obstructions with potential collision when the *Leader* curves.

Regarding the implementation on the *Leader*: in (19), if we replace  $\psi_{tar,L}$  with  $\psi_{tar,L} = \phi_L + \alpha_{tar,L}$ ,  $\psi_{obs,turn}$  with

(13),  $\psi_{vir,L}$  with (11), and then finally replace  $\psi_{desvir,L}$  in (18), the term  $\phi_L$  is canceled out, and one gets

$$f_{desvir,L} = \lambda_{desvir,L} \sin \left( \alpha_{vir,L} + \frac{\alpha_{tar,L} - \alpha_{vir,L}}{1 + \exp \left[ -\mu_1 (d_{obs,turn} - d_{vir,L}) + \ln \left( \frac{\alpha_{tar,L} - \alpha_{obs,turn}}{-\gamma_{vir,L}} \right) \right]} \right) \quad (20)$$

This implies that, for the implementation of the integrated behaviors, there is once again no need to keep an estimate of the robot’s heading direction, and thus calibration and/or odometry errors do not matter.

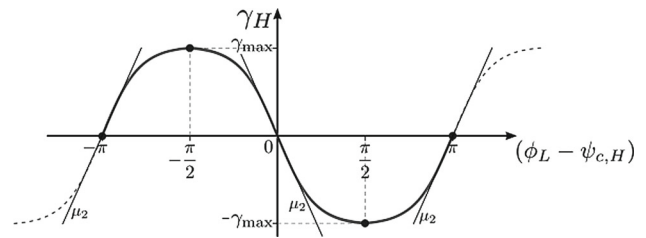
**For the Helper (r=H)** The resultant attractor is given by

$$\psi_{desvir,H} = \psi_{c,H} + \gamma_H \quad (21)$$

where  $\gamma_H$  represents intentional non-alignment in relation to payload direction,  $\psi_{c,H}$ , and its value depends on the Leader’s heading direction,  $\phi_L$ , as well as on the direction of the payload,  $\alpha_{c,L}$ , as follows:

$$\gamma_H = \begin{cases} \gamma_{max} \cdot \frac{\frac{2}{1 + \exp(-\mu_2 \cdot (\epsilon + \pi))} - 1}{\frac{2}{1 + \exp(0.5\pi\mu_2)} - 1}, & \epsilon < -\frac{\pi}{2} \\ -\gamma_{max} \cdot \frac{\frac{2}{1 + \exp(-\mu_2 \cdot \epsilon)} - 1}{\frac{2}{1 + \exp(0.5\pi\mu_2)} - 1}, & |\epsilon| \leq \frac{\pi}{2} \\ \gamma_{max} \cdot \frac{\frac{2}{1 + \exp(-\mu_2 \cdot (\epsilon - \pi))} - 1}{\frac{2}{1 + \exp(0.5\pi\mu_2)} - 1}, & \epsilon > \frac{\pi}{2} \end{cases} \quad (22)$$

with  $\epsilon = (\phi_L - \psi_{c,H}) = \pi - \alpha_{c,L}$  (see Fig. 4), and  $\alpha_{c,L}$  is directly measured by the payload support base on the Leader and communicated to the Helper (i.e. there is no need to know  $\phi_L$  in the implementation). The parameters  $\mu_2 > 0$  and  $\gamma_{max} \in [0, \pi/2]$  allow for a control of the curvature radius that the helper will draw in order to align the payload with the leader’s heading direction. When the Leader navigates in a heading direction so that that it pulls the payload, then  $-\pi/2 \leq \epsilon \leq \pi/2$  and the Helper tries to align the payload accordingly. When the Leader is pushing the payload, i.e.  $-\pi/2 > \epsilon > \pi/2$ , then the Helper moves away in order



**Fig. 8** Graphic representation of the  $\gamma_H$  for the Helper as a function of  $(\phi_L - \psi_{c,H})$ , where  $\mu_2$  is used to control how fast the Helper aligns the payload with the Leader’s heading direction. Note that:  $\psi_{c,H} = \psi_{c,L} - \pi$  thus  $\phi_L - \psi_{c,H} = \pi - \alpha_{c,L}$

to position itself behind the Leader first, and then align the payload. This function is depicted in Fig. 8.

Finally, and very importantly, with regard to implementation on the robot, note that if (21) is replaced in (18) one gets

$$f_{desvir,H}(\phi_H) = \lambda_{desvir,r} \sin(\alpha_{c,H} + \gamma_H). \quad (23)$$

Hence, there is no need to maintain an estimate of the Helper or Leader’s heading directions in the implementation.

#### 4.1.5 Integration of $F_{obs,r}$ and $f_{desvir,r}$ and the importance of noise

The integration of all behaviors is obtained by summing their respective contributions to the vector field of the heading direction dynamics, which finally is given by:

$$d\phi_r/dt = f_r(\phi_r) = f_{desvir,r}(\phi_r) + F_{obs,r}(\phi_r) \quad (24)$$

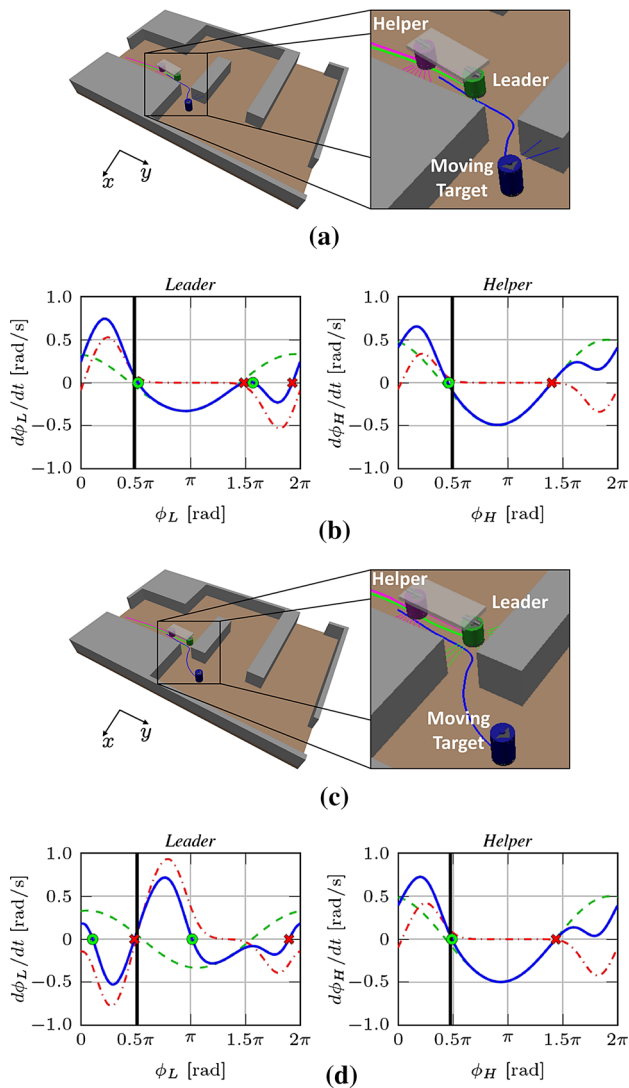
Qualitative changes of behavior arise when the number and/or stability of fixed points change. These changes correspond to bifurcations in the vector field of (24), brought about by changing sensory information as the robots move or because the environment is dynamic (see Fig. 9). For instance, an attractor pointing along a path heading between two obstacles may become unstable and turn into a repeller (i.e. unstable state) as the team approaches the obstacles. At such bifurcations, the heading direction may come to lie exactly on a repeller (a former attractor that turned unstable).

To ensure escape from repellers within a limited time, the vector field (24) is augmented by means of a stochastic contribution:

$$d\phi_r/dt = f_r(\phi_r) + f_{stoch,r} \quad (25)$$

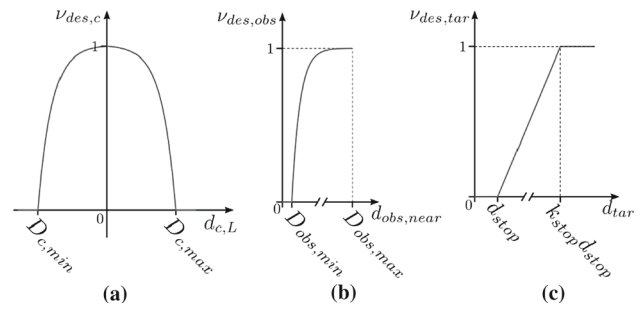
where

$$f_{stoch,r} = \sqrt{Q_r} \cdot \xi_n \quad (26)$$



**Fig. 9** Heading direction dynamics, showing the individual contributions  $f_{des,vir}$  (dashed green line) and  $F_{obs,r}$  (dashed-dot red line) and their integration (solid blue line). The current value of  $\phi_r$  is indicated by a solid vertical line. green circles 'o' and red crosses 'x' mark the attractors and repellers, respectively, of the resulting dynamical systems. As can be seen, a bifurcation in the *Leader's* heading direction has taken place. The stability of the fixed points has changed. In particular, the attractor  $\phi_L$  was in become a repeller. Noise in the system will guarantee that this variable moves away from the repeller and converges to an attractor of the resultant dynamics. Here it will converge to the attractor near  $\pi$ . **a** Before bifurcation: snapshot, **b** before bifurcation: vector fields, **c** after bifurcation: snapshot, **d** after bifurcation: vector fields (Color figure online)

is chosen as Gaussian white noise,  $\xi_n$ , so that  $Q_r$  is the effective variance of the noise. This stochastic contribution exists in addition to sensory and motor noise, which may vary as a function of environmental conditions. Since behavior is generated by asymptotically stable states (attractors), the system is robust in the face of noise.



**Fig. 10** The *Leader's* path velocity attenuation term profiles. **a** Attenuation term as a function of payload displacement, **b** attenuation term as a function of distance to the nearest obstacle, **c** attenuation term as a function of distance to the final destination

### 4.2 Path velocity control

The robots' path velocity,  $v_r$ , with  $r = \{L, H\}$ , is defined by a linear dynamical system:

$$dv_r/dt = g_{des,r}(v_r) = -\lambda_{v,r} \cdot (v_r - v_{des,r}) \tag{27}$$

where  $\lambda_{v,r} > 0$ , is the relaxation rate to the desired path velocity  $v_{des,r}$ .

The definition of the desired path velocity,  $v_{des,r}$ , is different for the *Leader* and *Helper*.

**For the Leader (r=L)** The desired velocity is:

$$v_{des,L} = V_{des,L} \cdot v_{des,c} \cdot v_{des,obs} \cdot v_{des,tar} \tag{28}$$

in which  $V_{des,L}$  constitutes a parameter that allows one to set the *Leader's* maximum path velocity, and the factors  $\{v_{des,c}, v_{des,obs}, v_{des,tar}\} \in [0, 1]$  are attenuations of that velocity.

The factor  $v_{des,c}$  (see Fig. 10a) is given by

$$v_{des,c} = 1 - \frac{1 - \exp\left(\mu_s \cdot \frac{|d_{c,L}|}{D_{c,max}}\right)}{1 - \exp(\mu_s)} \tag{29}$$

where  $d_{c,L}$  is the displacement of the transported object as measured by the *Leader's* payload support base. The displacement value is a measure of the relative path velocity to the *Helper*, since the displacement is symmetric and approximately of the same magnitude in both robots. The parameter  $\mu_s > 0$  controls the exponential decay, and  $D_{c,max}$  sets the maximum value allowed for the displacement of the payload (which is intrinsically dependent on the length of the support base).

$v_{des,obs}$  will ensure a decrease in the *Leader*'s path velocity in the presence of obstructions (see Fig. 10b):

$$v_{des,obs} = \frac{1 - \exp[-\mu_{obs}(d_{obs,near} - D_{obs,min})]}{1 - \exp[-\mu_{obs}(D_{obs,max} - D_{obs,min})]} \quad (30)$$

So that it depends on the closest distance to the nearest obstacle,  $d_{obs,near}$ . The parameter  $\mu_{obs} > 0$  sets the exponential decay. The parameters  $D_{obs,min}$  and  $D_{obs,max}$  are the minimum and maximum distances, respectively, which are measured by the distance sensors.

Finally, the attenuation factor  $v_{des,tar}$  will decrease the *Leader*'s velocity in the vicinity of the final destination (see Fig. 10c):

$$v_{des,tar} = \begin{cases} 0, & d_{tar} < d_{stop} \\ \frac{d_{tar}-d_{stop}}{(k_{stop}-1)\cdot d_{stop}}, & d_{stop} \leq d_{tar} \leq k_{stop} \cdot d_{stop} \\ 1, & d_{tar} > d_{stop} \end{cases} \quad (31)$$

where  $d_{tar}$  is the distance to the target destination. The parameters  $d_{stop}$  and  $k_{stop} (\geq 1)$  allow one to define the stop distance and the distance below which the velocity should start to decrease.

**For the Helper (r=H)** In the same way as the *Leader*, the *Helper* should also adjust its desired path velocity by reducing it, if necessary, to its partner's velocity. This is accomplished by means of a PID controller that sets the *Helper*'s path velocity,  $v_{des,H}$ , as a function of the displacement of the payload measured by the *Helper*'s support base. This approach was already presented in Machado et al. (2013).

### 4.3 Hierarchy of relaxation rates

When sensory information changes, the attractors shift. In order to ensure the stability of the control system, i.e., that the control/state variables always relax to an attractor of the resulting dynamics before it moves (as induced by the changing sensorial information), as well as to ensure that obstacle avoidance behavior has precedence over the other components, the following hierarchy of relaxation rates must be observed (see Bicho et al. 2000 for details):

$$\lambda_{obs,i,r} \ll \lambda_{v,r}, \quad \lambda_{desvir,r} \ll \lambda_{v,r}, \quad \lambda_{desvir,r} \ll \lambda_{obs,i,r}. \quad (32)$$

### 4.4 Stability analysis

The analysis of stability is relatively simple. For the nonlinear dynamical system of (24),<sup>1</sup> which governs each robot's head-

<sup>1</sup> Note that, although the vector field  $f_r(\phi_r)$  ( $r=Leader, Helper$ ) changes, as the robot moves or the sensorial information changes, these

ing direction,  $\phi_r$ , one uses a Lyapunov's method. In order to do so, one can consider the following Lyapunov function

$$V_r(\phi_r) = V_{desvir,r}(\phi_r) + V_{obs,r}(\phi_r) + K_{pot,r} \quad (33)$$

where  $K_{pot,r} \in \mathbb{R}$  and

$$V_{desvir,r}(\phi_r) = -\lambda_{desvir,r} \cos(\phi_r - \psi_{desvir,r}) \quad (34)$$

$$V_{obs,r}(\phi_r) = \sum_{i=1}^N \lambda_{obs,i,r} \sigma_{i,r}^2 \exp\left(-\frac{(\phi_r - \psi_{obs,i,r})^2}{2\sigma_{i,r}^2}\right) \quad (35)$$

By choosing sufficiently large values for  $K_{pot,r}$  one can always guarantee that the following condition holds:

$$V_r(\phi_r) > 0, \quad \forall \phi_r \in [0, 2\pi[.$$

Next, since the following also holds

$$\dot{V}_r(\phi_r) = \frac{dV_r(\phi_r)}{dt} = -[f_r(\phi_r)]^2 < 0, \quad \forall \phi_r \in [0, 2\pi[ \setminus \{\tilde{\phi}_r\} \quad (36)$$

where  $f_r(\phi_r)$  is given by (24) and  $\tilde{\phi}_r$  are the fixed points (i.e. zeros) of  $f_r(\phi_r)$ , then the stability criteria of the Lyapunov's direct method are satisfied (e.g. La Salle and Lefschetz 2012), and one can claim that the dynamical system which governs each robot's heading direction is asymptotically stable, i.e.  $\phi_r$  is always converging to an attractor. That this is true in practice can be seen in all the results presented in Sect. 5, where one can see that each robot's heading direction,  $\phi_r$ , is always relaxing and staying very close to an attractor of the resultant dynamics.

Path velocity is controlled by the linear dynamical system (27). It can be easily proven, by means of the linear stability theory, that it is also asymptotically stable because  $dg_r(v_r)/dv_r = -\lambda_{r,v} < 0$ .

A similar stability analysis—but for an application related to motion control of multi-robot formations—may be found in Monteiro and Bicho (2010).

## 5 Results

In this section, by resorting to computer simulations and experiments with real robots, a selection of results is presented. These highlight the flexible nature of the dynamic control architecture in coping with several adverse and changing situations. A set of five **videos** provided as supplementary material show the robots' behavior (see "Appendix C"). The parameters of the dynamics were the

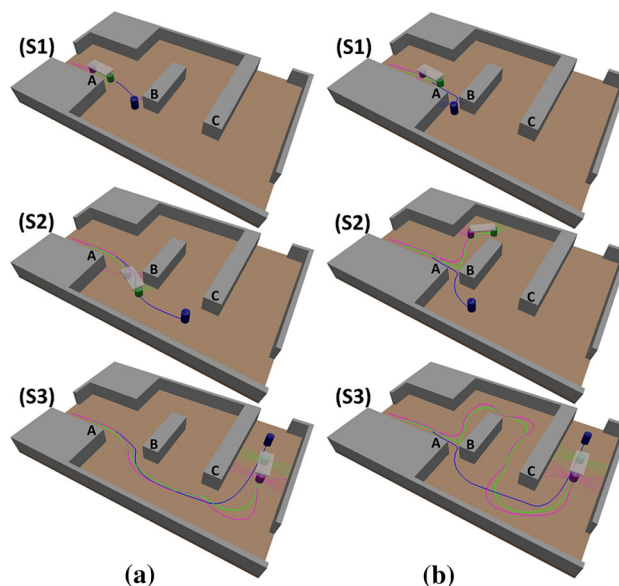
dynamical systems are autonomous, since  $f_r(\phi_r)$  does not explicitly depend on time.

same for all the (simulated and real) experiments and their values can be found in “Appendix B”. Each experiment is documented in the text with sample snapshots of the videos. For each robot, one also presents plots showing the temporal evolution of the fixed points (i.e. attractors and repellers) of the heading direction dynamics, in (24), together with the temporal evolution of the heading directions,  $\phi_r$ . The goal is to show that bifurcations occur when qualitative changes in behavior need to occur, and that each control variable  $\phi_r$  follows one of the resultant attractors very closely. This demonstrates that the control systems are asymptotically stable, which therefore make them robust in the face of disturbances. For the experiments with real robots, the temporal evolution of the robots’ path velocity  $v_r$  and displacements  $d_r$  of the object/cargo, relating to the robots’ centre, are also presented and analyzed.

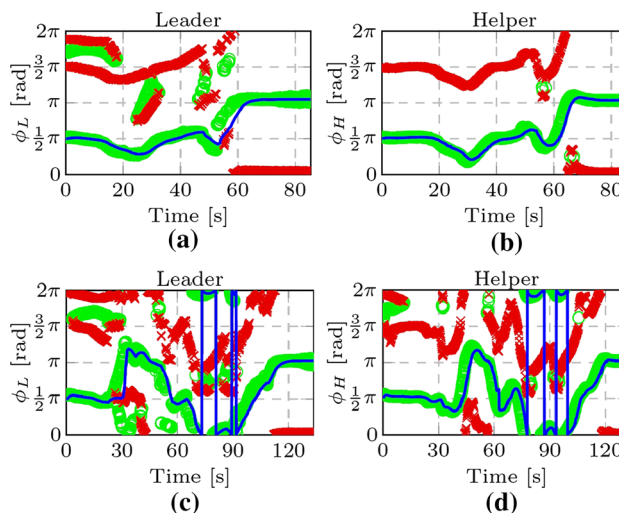
### 5.1 Following moving targets and flexibility in environments with layout changes

Here, one first demonstrates that there is nothing in the proposed approach that forces us to work with static targets. Importantly, one also shows the ability to generate qualitatively different trajectories if the environment changes. Fig. 11 shows two runs: in each of these the two-robots which jointly carry the cargo have to follow a moving agent (here a mobile robot), and this drives the team to the final target destination. However, the environment layout changes from one run to another. These two runs can be seen in **Video #1**. Note that there is no communication between this moving agent and the robotic team; thus, the team could also be following a human operator/co-worker. As shown in the first run (panel a), the conductor agent navigates in a cluttered space, heading towards the destination location. The robotic team is able to carry the load while simultaneously closely following the moving agent and avoiding collisions with the walls and crossing corners (i.e. L-turns). In the run depicted in panel (b), the conductor agent executes a similar path. It moves around obstacle B on its right has before (snapshots S1–S2). However, the passage (between A and B) is now not large enough to allow the robotic team to move through it. Therefore, instead of placing an attractor in the direction of the passage, the behavioral dynamics positions a repeller at this direction and an attractor on the left side. This bifurcation in the dynamics corresponds to the decision to turn left. Thus, a new and qualitative different trajectory is generated (online) for each robot in the team. Note that the robotic team in this second scenario makes two U-turns, one to avoid obstacle B and another to move around wall C.

For these two runs, Fig. 12 shows the complete temporal evolution of the resulting fixed points of the heading direction dynamics, in (1) for both robots, together with the temporal evolution of the robots’ heading direction. From the analysis

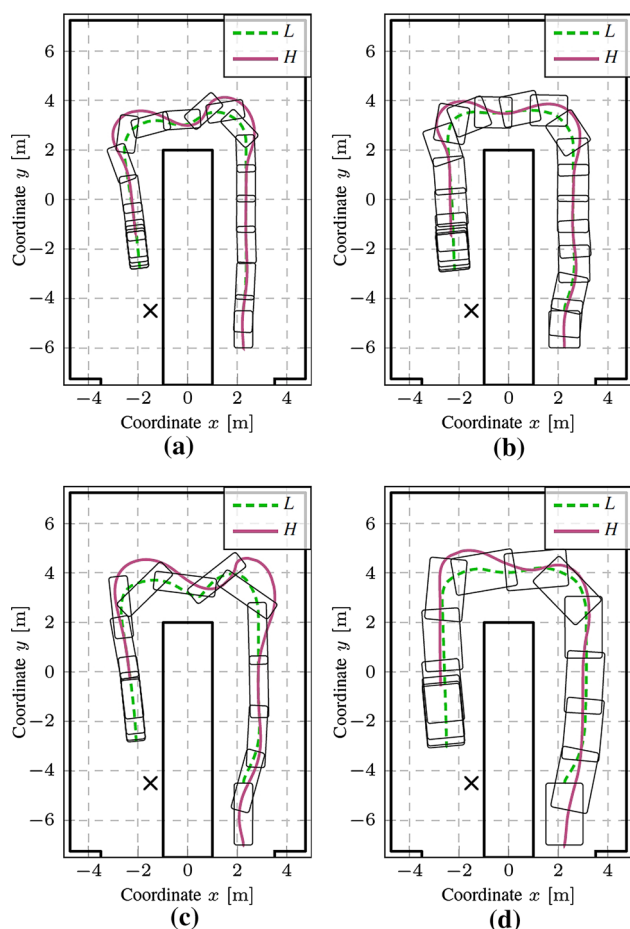


**Fig. 11** Snapshots of videos illustrating the capacity to follow moving targets and to cope with environments that change in layout (see **Video #1**). Legend: blue robot—moving target, green robot—Leader, magenta robot—Helper. **a** Run 1: large passage between A and B,  $t = \{14, 31, 75\}$  s. **b** Run 2: narrow passage between A and B,  $t = \{21, 57, 126\}$  s (Color figure online)



**Fig. 12** Time course of the resulting fixed points—attractors as green circles and repellers as red crosses—of the heading directions dynamics (left for Leader and right for Helper) and the heading direction—blue line—for the runs depicted in Fig. 11. **a** Run 1. **b** Run 2 (Color figure online)

of these plots, it is clear that the systems are asymptotically stable. This is due to the fact that the system’s states (i.e. the heading direction of each robot) are able to track the moving attractors, being always in, or very near, one of the stable fixed points of the dynamics.

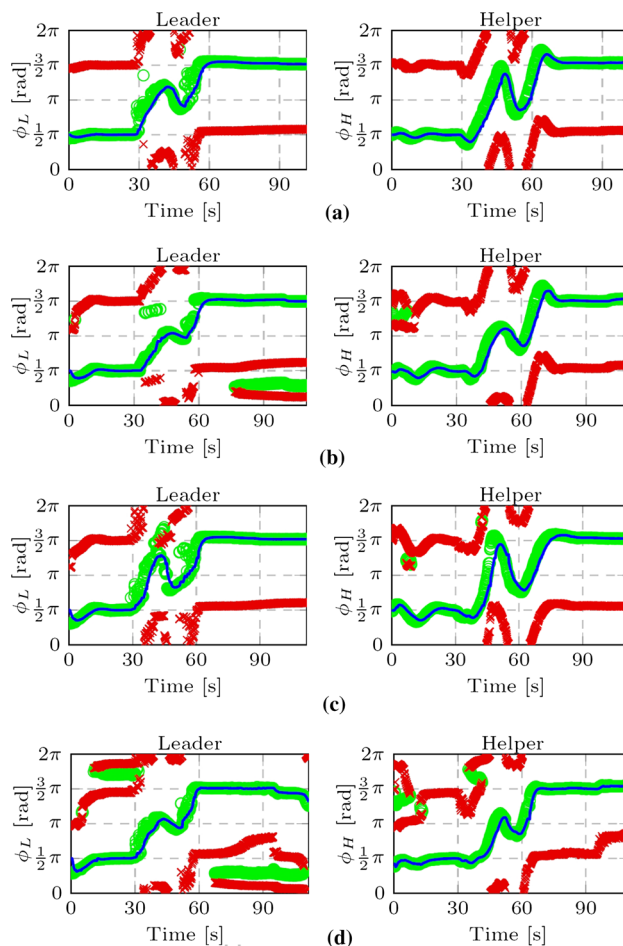


**Fig. 13** Robots' paths—dashed green line for *Leader* and solid magenta line for *Helper*—when transporting cargos—rectangle with rounded corners—with different sizes in a U-turn scenario (see **Video #2**). **a** Cargo [m]:  $1.5 \times 0.75$ . **b** Cargo [m]:  $1.5 \times 1.25$ . **c** Cargo [m]:  $2.5 \times 0.75$ . **d** Cargo [m]:  $2.5 \times 1.5$  (Color figure online)

## 5.2 Carrying cargos of different sizes

The ability to transport objects of different sizes is of the utmost importance in the scenarios considered, where from run to run the team may be carrying cargos of different lengths and widths. This implies adaptable team behavior so as to adjust to these changes. Such adaptability is demonstrated in several simulation runs depicted in Fig. 13. Each panel in this figure shows both the robots' and the cargo's path in a scenario with a sharp U-turn. The payload dimensions are different in each panel, which result in distinct trajectories; however, in all the team completes the sharp U-turn successfully (see also **Video #2**).

For the four experiments depicted in Figs. 13, 14 shows the complete temporal evolution of the resulting fixed points of the heading direction dynamics, in (1) for both robots, together with the temporal evolution of the robots' heading direction. Once again, it is clear that the systems are always

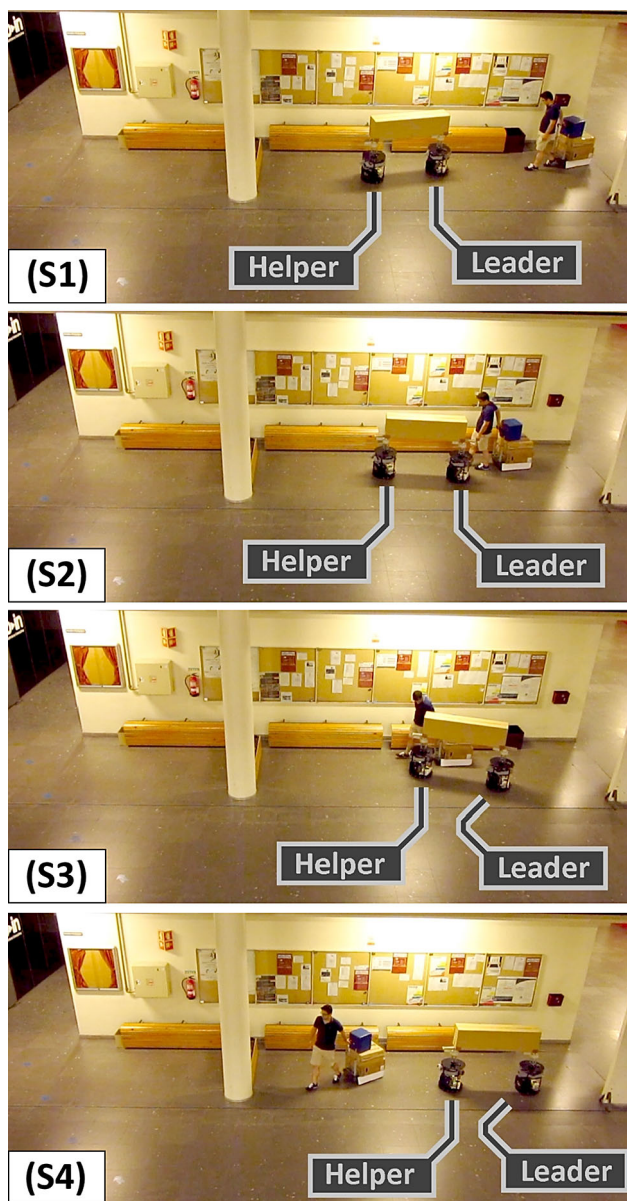


**Fig. 14** Time course of the resulting fixed points—attractors as green circles and repellers as red crosses—of the heading directions dynamics (left for *Leader* and right for *Helper*) and the heading direction—blue line—for the runs depicted in Fig. 13. **a** Cargo [m]:  $1.5 \times 0.75$ . **b** Cargo [m]:  $1.5 \times 1.25$ . **c** Cargo [m]:  $2.5 \times 0.75$ . **d** Cargo [m]:  $2.5 \times 1.5$  (Color figure online)

asymptotically stable.  $\phi_L$  and  $\phi_H$  are always in, or very near, a fixed point attractor.

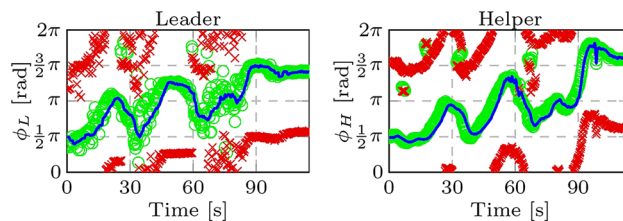
## 5.3 The challenge of moving obstacles

**Video #3** and Fig. 15 show the team's behavior in a scenario which challenges the ability to cope with dynamic obstacles. The experiment reported here also serves to demonstrate the robust nature of the control system governing the robots' state variables, which is able to withstand noisy information (either sensorial or communicated). The robot team moves towards the target destination and, as it does so, a human operator pulling a trolley drives in the opposite direction (snapshot S1). The *Leader* robot detects this and steers in a direction in order to avoid collision (snapshots S2); the human operator and trolley are then sensed by the *Helper* robot, which also avoids them (snapshots S3–S4). The complete tempo-

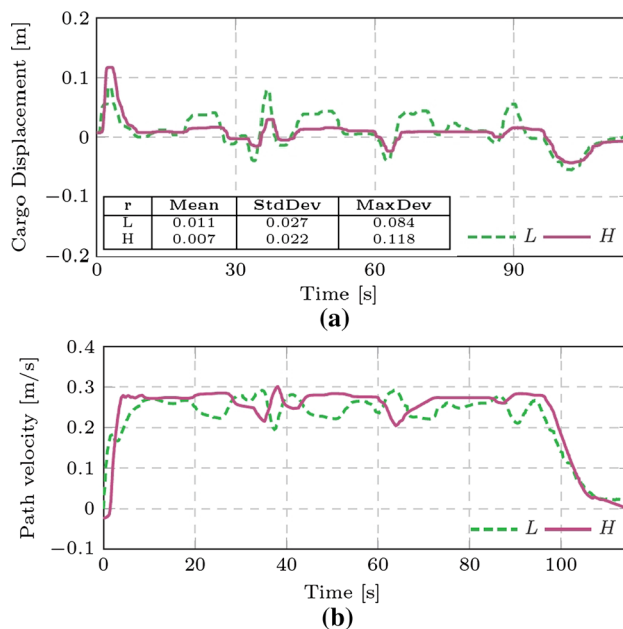


**Fig. 15** Snapshots of video ( $t = \{62, 65, 67, 71\}$  s) illustrating the capacity to avoid moving obstacles (see **Video #3**)

ral evolution of the robots’ heading direction, as well as the temporal evolution of the fixed points, are shown in Fig. 16. As intended the state variables heading direction,  $\phi_L$  and  $\phi_H$ , are always very close to one of the (resulting) attractors, which means that the robots’ behavior is governed by a sequence of asymptotically stable states (attractors) that contribute to making the behavior robust to perturbations. This is clearly shown in the time course of the *Leader*’s heading direction (left plot in Fig. 16). Noise in the *Leader*’s location system has caused the fixed points to jump up and down. However, the behavioral variable did not accompany these instant changes. This is so because, since it is governed by a dynamical system, there is always exponential relaxation



**Fig. 16** Time course of the resulting fixed points—*attractors* as green circles and *repellers* as red crosses—of the heading directions dynamics (left for *Leader* and right for *Helper*) and the heading direction—blue line—for the runs depicted in Fig. 15



**Fig. 17** Overview of cargo displacements and robots’ path velocity for the scenario shown in Fig. 15 ( $L=Leader$ ,  $H=Helper$ ). **a** Cargo displacements, **b** path velocities

from one state to another. This noise in the *Leader*’s location system is not observable in its overt behavior, and thus the team successfully accomplishes the joint transportation task.

Figure 17 shows the temporal evolution of the cargo displacements on the top of each robot ( $d_L$  and  $d_H$ ) and the robots’ path velocity ( $v_L$  and  $v_H$ ), respectively.

As can be seen, the maximum value of the displacements measured on the *Leader* and *Helper* are 8.4 and 11.7 cm, respectively, which are rather below the maximum value allowed for displacements (i.e.  $d_{H,max} = d_{L,max} = 20$  cm) (see Fig. 17a). This is so because the robots’ path velocity (Fig. 17b) is able to adequately compensate for the relative displacements of the cargo and maintains the distance between the robots within boundaries, even when they are both challenged by moving obstacles.

## 5.4 Abrupt perturbations

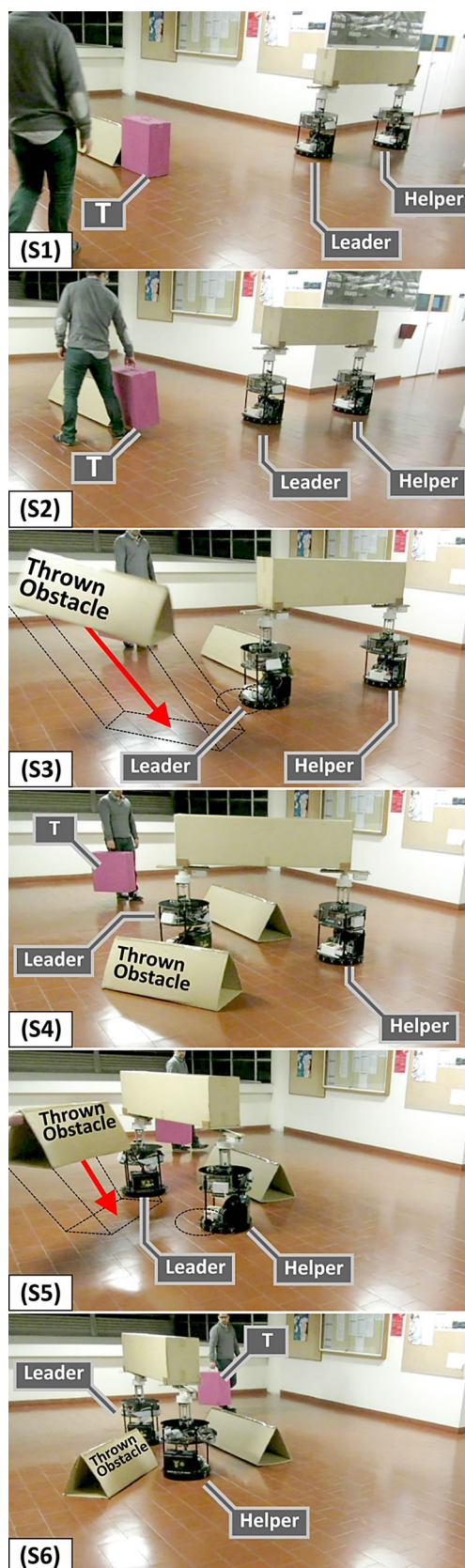
Besides being able to deal with moving obstacles and targets, the team should also successfully handle unexpected events during task execution, even though these may cause abrupt perturbations. One of these events can be the sudden appearance of an obstacle in the robot's path (e.g. a box that fell from a nearby shelf). With the purpose of testing this ability, a synthetic scenario (see **video #4** and Fig. 18) was constructed, where a human subject continuously moved the target location,  $T_1$ —magenta box, to force the team to navigate close and around the obstacles (snapshots  $S1$ – $S2$ ). In order to create a more demanding situation, a second human subject threw obstacles onto to the robots' path. First, the obstacle landed close to the *Leader* (see snapshots  $S3$ – $S4$ ); it reacted immediately to avoid the collision by performing a fast counter-clockwise turn (which can also be seen in Fig. 19). The obstacle was subsequently removed and thrown again, but now landed close to the *Helper* which responded by adequately avoiding collision (snapshots  $S5$ – $S6$ ).

The remainder of the experiment continued with the team contouring the centered obstacle and avoiding more abrupt perturbations caused by a human throwing obstacles onto their path.

Figure 20 shows the temporal evolution of the cargo displacements, as measured on the top of the robots, and the robots' path velocity, respectively. An analysis of these plots show that, even when abrupt perturbations occurred, the robots were able to quickly adjust their path velocity and to maintain the displacements of the cargo within boundaries. The maximum observed displacements are 14.2 and 11.3 cm for the *Leader* and *Helper*, respectively.

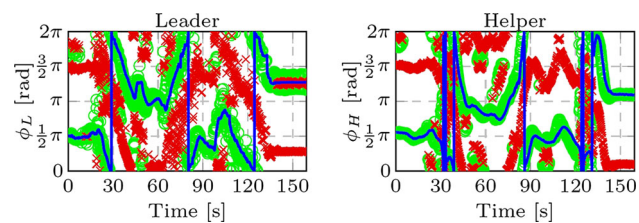
## 5.5 Validation on a factory floor

Finally, validation on a real factory floor was performed (see **video #5**). Figure 21 depicts the factory floor layout and the paths followed by the robots and cargo. In addition to the challenge presented by a real factory environment, this scenario tried to aggregate most of the situations described separately in previous subsections. Namely, the appearance of a human operator driving a pallet stacker (moving obstacle A in Fig. 21), which disturbed the *Helper* (see snapshot  $S1$  in Fig. 22). Subsequently, the team was confronted with a narrow passage, first near obstacles B and C (see snapshots  $S3$  and  $S4$  in Fig. 22) and later in a U-shaped turn, with a narrow passage, near obstacles D and E (see snapshots  $S5$  and  $S6$  in Fig. 22). The team was able to bypass all the difficulties while expectedly presenting system behaviors that are asymptotically stable. This can be seen in the temporal evolution of the fixed points of the heading direction dynamics, shown in Fig. 23.

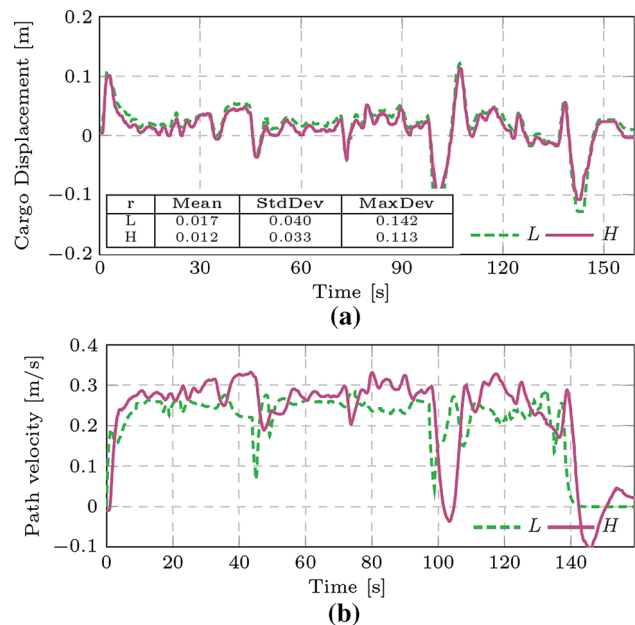


**Fig. 18** Snapshots of video ( $t = \{11, 14, 22, 25, 29, 32\}$  s) illustrating the ability to avoid abrupt perturbations (see **video #4**)

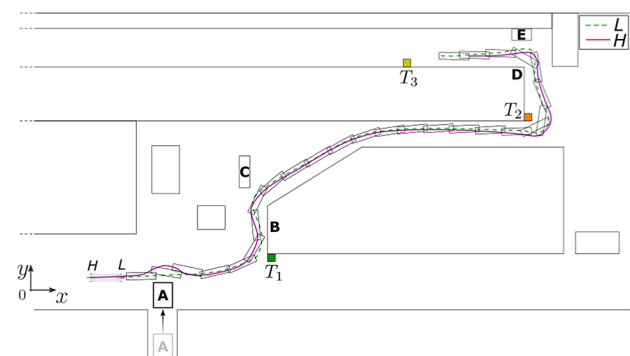




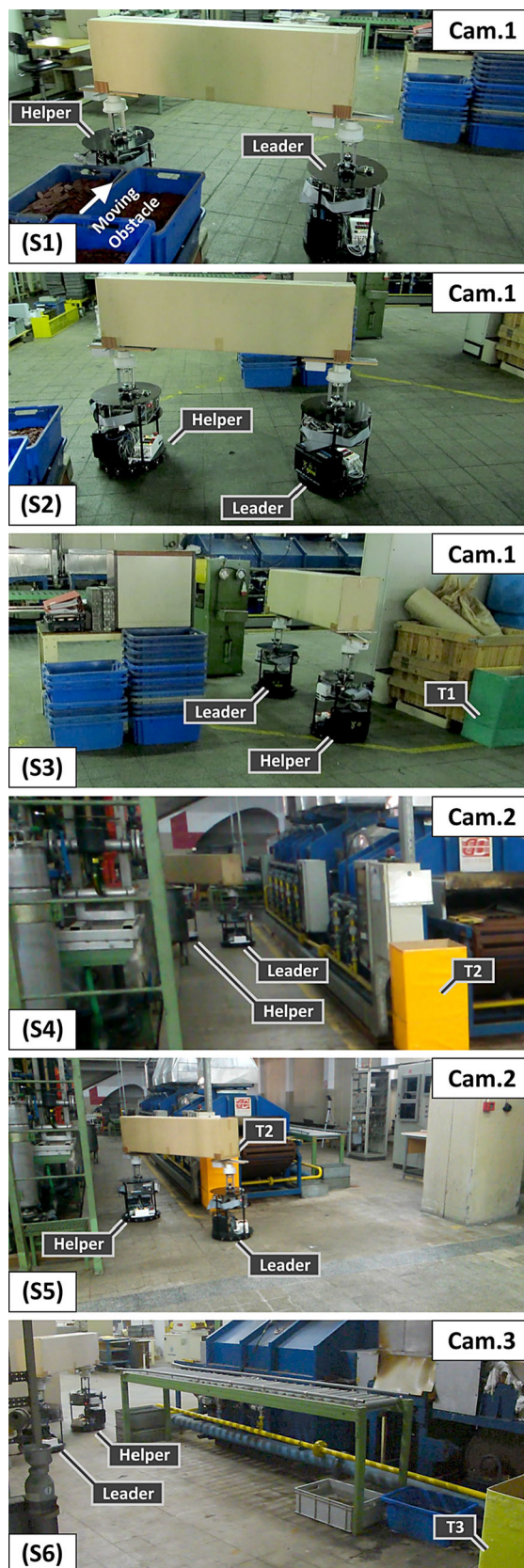
**Fig. 19** Time course of the resultant fixed points— attractors as green circles and repellers as red crosses—of the heading directions dynamics (left for *Leader* and right for *Helper*) and the heading direction—blue line—for the runs depicted in Fig. 18 (Color figure online)



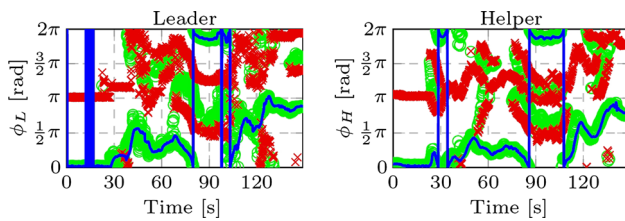
**Fig. 20** Overview of cargo displacements and robots' path velocity for the scenario shown in Fig. 18 ( $L=Leader$ ,  $H=Helper$ ). **a** Cargo displacements, **b** path velocities



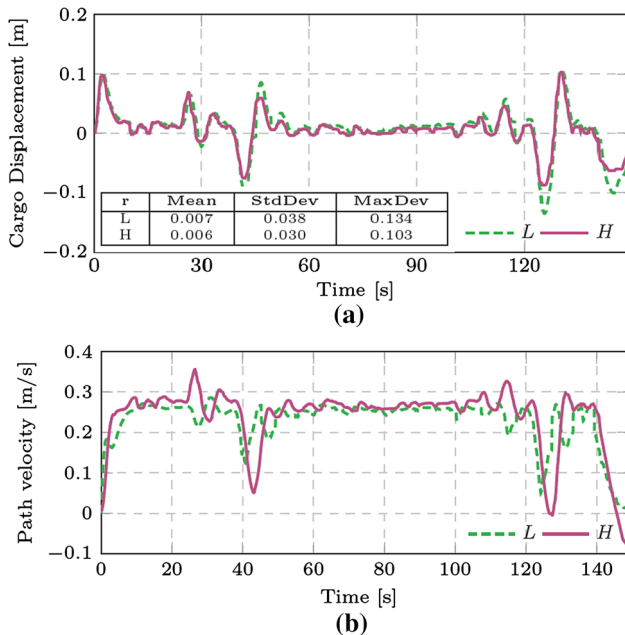
**Fig. 21** Layout of the factory floor with a plot of the robots' trajectories



**Fig. 22** Snapshots of video ( $t = \{25, 28, 48, 65, 100, 118\}$  s) illustrating the team navigating on a factory floor (see **Video** #5)



**Fig. 23** Time course of the resulting fixed points—attractors as green circles and repellers as red crosses—of the heading directions dynamics (left for *Leader* and right for *Helper*) and the heading direction—blue line—for the runs depicted in Fig. 22 (Color figure online)



**Fig. 24** Overview of cargo displacements and robots' path velocity for the scenario shown in Fig. 22 ( $L=Leader$ ,  $H=Helper$ ). **a** Cargo displacements, **b** Path velocities

The temporal evolution of the cargo displacements, as measured on the top of the robots, as well as the robots path velocity are both presented in Fig. 24. Once again, the analysis of these plots shows that the robots were able to adjust their path velocity and maintain the displacements of cargo within boundaries.

## 6 Conclusions

This paper has presented a distributed and dynamic control architecture, which allows teams of two autonomous mobile robots to perform joint transportation tasks in unknown environments that can change over time. The control architecture was inspired by how human teams perform transportation tasks, and is especially suitable for dealing with cluttered environments. The controllers are designed by following the *attractor dynamics approach to behavior-based robotics*,

where each robot's behavior is structured in terms of elementary behaviors; these set attractors at desired values for state variables, and repellers at values to be avoided. These attractors and repellers are integrated into a vector field which governs each of the state variables, heading direction and path velocity. Since these values depend on sensorial and communicated information, this means that attractors and repellers move over time. By design, the system is tuned to be always in, or very close to, an attractor of the overall dynamics, and to track it very closely as it moves over time. This ensures that the system is impervious to disturbances since the behavior is generated by an evolution of attractor (asymptotically stable) states.

In order to validate the controllers proposed for the team of robots, experimentation in different scenarios was conducted. These experiments included: computer simulations, real robotic experiments in large halls, as well as validation on a real factory floor. The robots were unaware of the environment map and the only information provided, which was limited to the *Leader*, was related to the destination location. All the experiments were performed by using the same parameter values, both with simulated and real robots. The only explicit communication exchanged, from the *Leader* to the *Helper*, was the bearing of the *Leader* to the transported payload. For implementation on the robots, there was no need to maintain an estimate of the robots' heading direction. This is important because it implied that odometry or calibration errors were of no significance. In all of the tested scenarios, the team not only performed successfully but also smoothly. It presented stable behavior and was able to cope with situations where: the environment included tight corridors with corners or sharp U-turns; the environment was dynamic, in the sense that obstacles or targets could move; a sudden obstacle appeared close to one of the robots or the load, also called an abrupt disturbance; as well as changes in load sizes.

The robots have no prior knowledge of the environment and generate online their collision free trajectory (also collision free for the payload). This is a positive feature since as we have shown and discussed allows flexible behavior to cope with very dynamic environments and situations. However, it is possible to find special configurations of obstacles that can trap the robotic team in sub regions of the workspace. To solve this, the robotic team can be endowed with the capacity to evaluate if it is progressing, in average, toward the target destination as times evolves. In case it is not one can deactivate the target acquisition behavior temporally. On real factory floors the probability that the robotic team is trapped is very low. This is so because the logistics may provide a sequence of intermediate (reachable) targets, which will drive the robotic team from the initial/loading location to the final destination or unloading location.

The focus of future work lies in extending the study to larger teams of robots on real factory floors, and the development of methods that would allow the *Leader* to be substituted by a human subject (implying that explicit communication should be removed).

As a final note, it is important to stress that the Non-linear Attractor Dynamics approach reported here is quite different from the well-known Potential Field approach (Khatib 1986) used to generate collision-free paths for vehicles. Both approaches represent obstacles and targets as repellers and attractors, respectively. In the potential field approach, the gradient of a scalar potential field is used to generate the robot's trajectory. Thus, the path is generated by the transient solutions of a dynamical system. On the other hand, in the non-linear attractor dynamics approach, the path is generated by a sequence of attractor solutions. Thus, the transient solutions of the potential field approach are replaced by a sequence of attractor solutions (i.e. asymptotical stable states) of a dynamical system, which therefore contribute to the asymptotical stability of the overall control system (for a comparison of these two approaches see e.g. Fajen et al. 2003; Costa e Silva et al. 2006; Hernandez et al. 2014).

**Acknowledgements** This work was supported by FCT—*Fundação para a Ciência e Tecnologia* within the scope of the Project PEst-UID/CEC/00319/2013 and by the Ph.D. Grants SFRH/BD/38885/2007 and SFRH/BPD/71874/2010, as well as funding from FP6-IST2 EU-IP Project JAST (Proj. Nr. 003747). We would like to thank the anonymous reviewers, whose comments have contributed to improve the paper.

## Appendix A: Robot kinematics

The path velocity,  $v$ , and angular velocity,  $\omega$ , of our robotic platforms were controlled by setting the linear speeds of the two driving wheels as follows:

$$v_{right} = v + \frac{D_{wheels}}{2} \omega \quad (37)$$

$$v_{left} = v - \frac{D_{wheels}}{2} \omega \quad (38)$$

where  $D_{wheels}$  is the distance between the two driving wheels.  $\omega = d\phi/dt$  is obtained directly from the behavioral dynamics for the heading direction, Eq. 25, and  $v$  results from integrating Eq. 27, by following a Euler method, i.e.,  $v = v + dt.g(v)$ , with  $dt$  being the time step.

## Appendix B: Values of parameters used in the experiments

$N_L = 11$ ;  $N_H = 21$ ;  $\delta_L = \pi/8$ ;  $\delta_H = \pi/16$ ;  $\beta_{1,r} = 2$ ;  $\beta_{2,r} = 0.5 \times C_l$ , where  $C_l$  is the cargo's length;  $\Psi_{vir,L} =$

$\pi/4$ ;  $\Psi_{thres} = \pi/6$ ;  $\lambda_{desvir,L} = 0.4$ ;  $\lambda_{desvir,H} = 0.5$ ;  $\mu_1 = 2$ ;  $\mu_2 = 2$ ;  $\sqrt{Q_r} = 0.01$ ;  $\Psi_v = \pi/4$ ;  $\gamma_{max} = 5\pi/12$ ;  $\lambda_{v,L} = 10/3$ ;  $\lambda_{v,H} = 2$ ;  $V_{des,L} = 0.3$ ;  $\mu_s = 1$ ;  $D_{c,max} = 0.2$ ;  $\mu_{obs} = 7$ ;  $D_{obs,min} = 0.1$ ;  $D_{obs,max} = 1.5$ ;  $k_{stop} = 2$ ;  $d_{stop} = 1.25$ .  $D_{wheels} = 45$  cm.  $dt \approx 50$  ms.

## Appendix C: Videos

In the supplementary material one can find the following videos, which further complement the results:

**Video #1** : illustrates the capacity of the robot team to transport the cargo while following moving targets and the ability to cope with environments that change in layout.

**Video #2**: demonstrates the capacity of the robot team to transport cargos of different sizes in U-turn scenarios.

**Video #3**: demonstrates the capacity of the robot team to avoid moving obstacles.

**Video #4**: shows the capability of the robot team to avoid abrupt perturbations that may appear during the joint transportation task.

**Video #5**: shows the robot team navigating on a factory floor, where their joint transportation behavior is challenged by a cluttered environment, with narrow passages and a moving obstacle (human operator driving a pallet stacker).

## References

- Abou-Samah, M., Tang, C., Bhatt, R., & Krovi, V. (2006). A kinematically compatible framework for cooperative payload transport by nonholonomic mobile manipulators. *Autonomous Robots*, 21, 227–242.
- Ahmadabadi, M., & Nakano, E. (2001). A "constrain and move" approach to distributed object manipulation. *IEEE Transactions on Robotics and Automation*, 17(2), 157–172.
- Althaus, P., Christensen, H. I., & Hoffmann, F. (2001). Using the dynamical system approach to navigate in realistic real-world environments. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, Grenoble, France*.
- Asahiro, Y., Chang, E., Mali, A., Suzuki, I., & Yamashita, M. (2001). A distributed ladder transportation algorithm for two robots in a corridor. In *Proceedings of the international conference on robotics and automation, ICRA 2001* (pp. 3016–3021).
- Bayram, H., & Bozma, I. (2016). Coalition formation games for dynamic multirobot tasks. *The International Journal of Robotics Research*, 35(5), 514–527.
- Bicho, E. (2000). *Dynamic approach to behavior-based robotics: Design, specification, analysis, simulation and implementation*. Aachen: Shaker Verlag.
- Bicho, E., & Schöner, G. (1997). The dynamic approach to autonomous robotics demonstrated on a low-level vehicle platform. *Robotics and Autonomous Systems*, 21(1), 23–35.
- Bicho, E., Mallet, P., & Schöner, G. (2000). Target representation on an autonomous vehicle with low-level sensors. *The International Journal of Robotics Research*, 19(5), 424–447.
- Bouloubasis, A., & McKee, G. (2005). Cooperative transport of extended payloads. In *Proceedings of the 12th international*

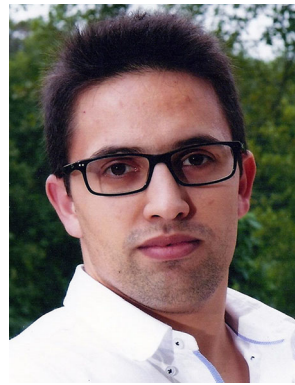
- conference on advanced robotics (ICAR'05), Seattle, USA (pp. 882–887).
- Cao, Y., Kusunaga, A., & Kahng, A. (1997). Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, 4, 7–27.
- Cheng, P., Fink, J., Kumar, V., & Pang, J. S. (2009). Cooperative towing with multiple robots. *Journal of Mechanisms and Robotics*, 1(1), 011008.
- Costa e Silva, E., Bicho, E., Erlhagen, W. (2006). The potential field method and the nonlinear attractor dynamics approach: What are the differences? In *Control 2006 7th Portuguese conference on automatic control, Lisboa, Portugal* (pp. 816–822).
- Donald, B., Garipey, L., & Rus, D. (2000). Distributed manipulation of multiple objects using ropes. In *IEEE international conference on robotics and automation, 2000. Proceedings. ICRA'00, IEEE* (Vol. 1, pp. 450–457).
- Durrant-Whyte, H. F. (1996). An autonomous guided vehicle for cargo handling applications. *The International Journal of Robotics Research*, 15(5), 407–440.
- Ellekilde, L. P., & Perram, J. W. (2005). Tool center trajectory planning for industrial robot manipulators using dynamical systems. *The International Journal of Robotics Research*, 24(5), 385–396.
- Endo, M., Hirose, K., Hirata, Y., Kosuge, K., Kanbayashi, T., Oomoto, M., et al. (2008). A car transportation system by multiple mobile robots-icart. In *2008 IEEE/RSJ international conference on intelligent robots and systems, IEEE* (pp. 2795–2801).
- Fajen, B. R., Warren, W. H., Temizer, S., & Kaelbling, L. P. (2003). A dynamical model of visually-guided steering, obstacle avoidance, and route selection. *International Journal of Computer Vision*, 54(1/2/3), 13–24.
- Fujii, M., Inamura, W., Murakami, H., Tanaka, K., & Kosuge, K. (2007). Cooperative control of multiple mobile robots transporting a single object with loose handling. In *IEEE international conference on robotics and biomimetics, Sanya* (pp. 816–822).
- Gross, R., & Dorigo, M. (2009). Towards group transport by swarms of robots. *International Journal of Bio-Inspired Computation*, 1(1–2), 1–13.
- Hashimoto, M., Oba, F., & Zenitani, S. (1993). Coordinative object-transportation by multiple industrial mobile robots using coupler with mechanical compliance. In *Proceedings of the international conference on industrial electronics, control and instrumentation, Maui, USA* (pp. 1577–1582).
- Hernandes, A. C., Guerrero, H. B., Becker, M., Jokeit, J. S., & Schöner, G. (2014). A comparison between reactive potential fields and attractor dynamics. In *2014 IEEE 5th Colombian workshop on circuits and systems (CWCAS), IEEE* (pp. 1–5).
- Hess, M., Saska, M., & Schilling, K. (2009). Application of coordinated multi-vehicle formations for snow shoveling on airports. *Intelligent Service Robotics*, 2(4), 205–217.
- Iossifidis, I., & Schoener, G. (2006). Dynamical systems approach for the autonomous avoidance of obstacles and joint-limits for a redundant robot arm. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems* (pp. 580–585).
- Jones, C., & Mataric, M. (2005). Behavior-based coordination in multi-robot system. In S. S. Ge & F. Lewis (Eds.), *Autonomous mobile robots: Sensing, control, decision-making, and applications*. New York: Marcel Dekker, Inc.
- Kashiwazaki, K., Yonezawa, N., Endo, M., Kosuge, K., Sugahara, Y., Hirata, Y., et al. (2011). A car transportation system using multiple mobile robots: ICART II. In *2011 IEEE/RSJ international conference on intelligent robots and systems, IEEE* (pp. 4593–4600).
- Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5(1), 90–98.
- Kim, Y., & Minor, M. (2010). Coordinated kinematic control of compliantly coupled multirobot systems in an array format. *IEEE Transactions on Robotics*, 26(1), 173–180.
- La Salle, J., & Lefschetz, S. (2012). *Stability by Liapunov's direct method with applications by Joseph L. Salle and Solomon Lefschetz* (Vol. 4). New York: Elsevier.
- Loh, C. C., & Traechtler, A. (2012). Cooperative transportation of a load using nonholonomic mobile robots. *International symposium on robotics and intelligent sensors 2012, Procedia Engineering* (Vol. 41, pp. 860–866) (IRIS 2012).
- Machado, T., Malheiro, T., Erlhagen, W., & Bicho, E. (2016). Multi-constrained joint transportation tasks by teams of autonomous mobile robots using a dynamical systems approach. In *2016 IEEE international conference on robotics and automation (ICRA), IEEE* (pp. 3111–3117).
- Machado, T., Malheiro, T., Monteiro, S., Bicho, E., & Erlhagen, W. (2013). Transportation of long objects in unknown cluttered environments by a team of robots: A dynamical systems approach. In *2013 IEEE international symposium on industrial electronics (ISIE), IEEE* (pp. 1–6).
- Monteiro, S., & Bicho, E. (2010). Attractor dynamics approach to formation control: Theory and application. *Autonomous Robots*, 29, 331–355.
- Parker, L. E. (2000). Current state of the art in distributed autonomous mobile robotics. In L. Parker, G. Bekey, & J. Barhen (Eds.), *Distributed autonomous robotic Systems 4* (pp. 3–12). Tokyo: Springer.
- Pereira, G., Pimentel, B., Chaimowicz, L., & Campos, M. (2002). Coordination of multiple mobile robots in an object carrying task using implicit communication. In *Proceedings of the international conference on robotics and automation, Washington, DC* (pp. 281–286).
- Reimann, H., Iossifidis, I., Schöner, G. (2010). Generating collision free reaching movements for redundant manipulators using dynamical systems. In *2010 IEEE/RSJ international conference on intelligent robots and systems (IROS), IEEE* (pp. 5372–5379).
- Sabatini, L., Secchi, C., & Fantuzzi, C. (2011). Arbitrarily shaped formations of mobile robots: Artificial potential fields and coordinate transformation. *Autonomous Robots*, 30(4), 385–397.
- Schöner, G., Dose, M., & Engels, C. (1995). Dynamics of behavior: Theory and applications for autonomous robot architectures. *Robotics and Autonomous Systems*, 16, 213–245.
- Soares, R., Bicho, E., Machado, T., & Erlhagen, W. (2007). Object transportation by multiple mobile robots controlled by attractor dynamics: theory and implementation. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, San Diego, CA* (pp. 937–944).
- Sprunk, C., Lau, B., Pfaff, P., & Burgard, W. (2017). An accurate and efficient navigation system for omnidirectional robots in industrial environments. *Autonomous Robots*, 41(2), 473–493.
- Steinhage, A. (1997). Dynamical systems for the generation of navigation behavior. Ph.D. thesis, Ruhr-Universität Bochum, Germany
- Stouten, B., & Graaf, A. (2004). Cooperative transportation of a large object: Development of an industrial application. In *Proceedings of the international conference on robotics and automation* (pp. 2450–2455).
- Streuber, S., & Chatziastros, A. (2007). Human interaction in multi-user virtual reality. In *Proceedings of the 10th international conference on humans and computers (HC 2007)*.
- Sudsang, A. (2002). Sweeping the floor: Moving multiple objects with multiple disc-shaped robots. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, Lausanne, Switzerland* (pp. 2825–2830).
- Takeda, H., Wang, Z. D., & Kosuge, K. (2003). Collision avoidance algorithm for two tracked mobile robots transporting a single object in coordination based on function allocation concept-

utilization of environmental information by visual sensor. In *Proceedings of the 11th international conference on advanced robotics, ICAR 2003, Coimbra, Portugal* (pp. 488–493).

- Tang, C., Bhatt, R., & Kroví, V. (2004). Decentralized kinematic control of payload transport by a system of mobile manipulators. In *Proceedings of the IEEE international conference on robotics and automation, New Orleans, LA* (pp. 2462–2467).
- Tanner, H., Loizou, S., & Kyriakopoulos, K. (2003). Nonholonomic navigation and control of cooperating mobile manipulators. *IEEE Transactions on Robotics and Automation*, 19(1), 53–64.
- Trebi-Ollennu, A., Nayar, H., Aghazarian, H., Ganino, A., Pirjanian, P., Kennedy, B., et al. (2002). Mars rover pair cooperatively transporting a long payload. In *Proceedings of the IEEE international conference on robotics and automation* (Vol. 3, pp. 3136–3141).
- Tsiamis, A., Bechlioulis, C. P., Karras, G. C., & Kyriakopoulos, K. J. (2015). Decentralized object transportation by two nonholonomic mobile robots exploiting only implicit communication. In *2015 IEEE international conference on robotics and automation (ICRA)*, IEEE (pp. 171–176).
- Wada, M., & Torii, R. (2013). Cooperative transportation of a single object by omnidirectional robots using potential method. In *2013 16th international conference on advanced robotics (ICAR)*, IEEE (pp. 1–6).
- Widyotriatmo, A., & Hong, K. S. (2011). Navigation function-based control of multiple wheeled vehicles. *IEEE Transactions on Industrial Electronics*, 58(5), 1896–1906.
- Yamaguchi, H., Nishijima, A., & Kawakami, A. (2015). Control of two manipulation points of a cooperative transportation system with two car-like vehicles following parametric curve paths. *Robotics and Autonomous Systems*, 63, 165–178.
- Yamashita, A., Arai, T., Ota, J., & Asama, H. (2003). Motion planning of multiple mobile robots for cooperative manipulation and transportation. *IEEE Transactions on Robotics and Automation*, 19(2), 223–237.
- Yamashita, A., Sasaki, J., Ota, J., & Arai, T. (1998). Cooperative manipulation of objects by multiple mobile robots with tools. In *Proceedings of the 4th Japan-France/2nd Asia-Europe congress on mechatronics, Citeseer* (Vol. 310, p. 315).
- Yang, X., Watanabe, K., Izumi, K., & Kiguchi, K. (2004). A decentralized control system for cooperative transportation by multiple non-holonomic mobile robots. *International Journal of Control*, 77(10), 949–963.
- Yufka, A., Parlaktuna, O., & Ozkan, M. (2010). Formation-based cooperative transportation by a group of non-holonomic mobile robots. In *2010 IEEE international conference on systems man and cybernetics (SMC)*, Istanbul, Turkey (pp. 3300–3307).



**Toni Machado** received the B.Eng., M.Sc. and Ph.D. degrees in Industrial Electronics and Computers from the University of Minho, Portugal, in 2005, 2008 and 2016, respectively. The subject of his M.Sc. and Ph.D. degrees were about object transportation by a team of autonomous mobile robots. He is currently with the Bosch Car Multimedia Portugal S.A. as a specialist in robotics, control, and automation and his research interests are focused on autonomous mobile robotic systems.



**Tiago Malheiro** received his M.Sc. in Industrial Electronics and Computers Engineering, with specialization on “Automation, Control and Robotics” and “Embedded Systems” from University of Minho, Portugal, in 2011. He is currently working toward the Ph.D. in robotics focusing on the development of pro-active robots to assist dependent persons. His research interests are focused on Robotics, Non-linear Dynamical Systems, and Embedded Systems.



**Sergio Monteiro** received his Ph.D. in Automation and Control from the University of Minho in 2007, with a thesis on the formation control of robot teams using the attractor dynamic approach. Currently he serves as an Assistant Professor at the Department of Industrial Electronics of the same University, where he teaches courses on signal processing and on robotics. He also integrates the “Mobile and Anthropomorphic Robotics Laboratory” of the “Centro Algoritmi” research center. His research interests are mostly focused on industrial and service robotics with an emphasis on cooperative multirobot systems.



**Wolfram Erhagen** is Associate Professor at the Department of Mathematics and Applications at the University of Minho, Portugal. He has been PI in several European and national projects in the ICT topic, including robotics. His multidisciplinary research covers the multi-scale analysis of neuronal activity, the functional modelling of brain circuits, and the implementation of neuro-based models, formalized as dynamical systems, in autonomous robots. In close cooperation with experimental groups he applies his theoretical investigations to problems of motor planning, visual perception and reasoning with the ultimate goal to bridge Cognitive Sciences to Robotics.



**Estela Bicho** is Associate Professor at the Department of Industrial Electronics at University of Minho, Portugal, where she is responsible for courses in Non-linear Dynamical Systems, Control and Robotics and heads the research lab on Autonomous (mobile and anthropomorphic) Robotics & Dynamical systems. She obtained the Ph.D. degree in Robotics, Automation and Control, in 1999, from the University of Minho. Her Ph.D. work entitled “Dynamic Approach to Behavior

Based-Robotics” received the honour price from Portuguese-IBM (1999). Her research concentrates on the use of dynamical systems for the design and implementation of control architectures for flexible control of high-DOF robotics systems, including human–robot interaction and multi-robot systems. She has been PI in several national and EU funded research projects in robotics.