

Exponential fields formulation for WMR navigation

Edgar A. Martínez-García^{a,b,*} and Rafael Torres-Cordoba^a

^aLaboratorio de Robótica, Institute of Engineering and Technology, Universidad Autónoma de Ciudad Juárez, Ciudad Juárez, Chihuahua, México

^bRobótica y Tecnología S. de R.L. Juárez, Chihuahua, Mexico

Abstract. In this manuscript, an autonomous navigation algorithm for wheeled mobile robots (WMR) operating in dynamic environments (indoors or structured outdoors) is formulated. The planning scheme is of critical importance for autonomous navigational tasks in complex dynamic environments. In fast dynamic environments, path planning needs algorithms able to sense simultaneously a diversity of obstacles, and use such sensory information to improve real-time navigation control, while moving towards a desired goal destination. The framework tackles 4 issues: 1) Reformulation of the Social Force Model (SFM) adapted to WMR; 2) the cohesion of a general inertial scheme to represents motion in any coordinate system; 3) control of actuators rotational speed as a general model regardless kinematic restrictions; 4) assuming detection of features (obstacles/goals), adaptive numeric weights are formulated to affect navigational exponential components. Simulation and experimental outdoors results are presented to show the feasibility of the proposed framework.

Keywords: Navigation, exponential components, dynamic model, directional fields

1. Introduction

Nowadays, service robotics is an exponential growing area where human-robot interaction and cooperation play a critical roll in many task applications. Most service robotics core-capabilities rely on navigation and mapping in dynamic environments. Regardless the sensing fashion, the system must feedback to account for recognition capabilities. Service robots form a wide variety of applications, tasks, and missions, requiring a wide range of capability issues: computational organization; smart algorithms; sensors coordination and data fusion schemes; physical devices

according to the tasks accomplishment; human-robot interaction; interfaces control; and real time autonomous navigation capabilities. This manuscript is centred on a general scheme to accomplish WMR autonomous navigation. During high speed navigational tasks, robots face unexpected hazards that must quickly be avoided, where the navigation control algorithm has strong relationship with the stability and manoeuvrability for the whole system. To avoid potential crashes, reliable planning algorithms must be computationally efficient while considering important WMR and motion dynamic effects. In order for a WMR to have the required abilities to navigate autonomously, a wide variety of aspects must be integrated, among which include merging intelligent planners, distributed-based architecture systems, collective sensing capabilities, and all essentials that

*Corresponding author: Edgar A. Martínez-García, Laboratorio de Robótica, Institute of Engineering and Technology, Universidad Autónoma de Ciudad Juárez, Chihuahua, México. E-mail: edmartin@uacj.mx.

depend on the kinematic models as means to create sophisticated control algorithms. Enhanced systems providing navigation safety will form a significant part of the market for intelligent automatic devices [24]; because autonomous navigational developments such as perception enhancement, route guidance, intelligent tracking control, and collision warning have the potential to provide navigation much easier and specially safe [25]. The focus of the present work concerns a model system for autonomous navigation, with capabilities that helps to avoid possible collisions directly. Direct avoidance includes features such as speed and steer control that automatically slows the robot/vehicle if getting too close to any obstacle.

Social directional fields are algorithms that incrementally explore free space, while searching for a suitable path, acting as navigation planners applicable to a rich class of intelligent WMR to develop a great variety of paths. The Social Force Model (SFM) was first reported by Helbing and Molnar [16] to simulate pedestrians motion in crowd environments. They proposed a SFM analytically solved in terms of acceleration vectors that generalizes the influences of a particle motion. However, the acceleration vector cannot mathematically be easily integrated because a descriptive motion model is not given. Some works dealing with potential fields treat the motion as a particle, making this approach an applicable model to most rolling-based locomotion systems, and do not include geometry of motion constrained by the robot kinematics. The concept of the SFM to simulate pedestrian dynamics of forces interacting in crowd environments has some amazing similarities with gases and fluid-dynamic [38], which attracted interest for modelling vehicular traffic systems.

In the work of Gayle et al. [34], a similar approach to the present context in using social potential fields are built on physics-based reactive motion planning presented with polyhedron-like soft bots (assuming point, point-like, circular or polygonal robot primitives); physics-based solve motion planning problems, and social potential fields exhibit various social behaviours, exhibiting relative easily and effectiveness at coordinating motion among multiple robots. However in its context it did not consider the variety of multi-body kinematic descriptions.

Shimoda et al. [17] proposed a potential field-based method for high speed navigation of unmanned ground vehicles on uneven terrain, where local minima and maxima problems were addressed with a simple

randomization technique. Kareem et al. [18] presented a fuzzy potential field approach for mobile robot motion planning using two fuzzy models for each field, repulsive and attractive respectively. Baronov [19] presented a reactive control law to navigate a single sensor-enabled vehicle to ascend or descend a scalar potential field. Ren et al. [20] proposed a potential function based on generalized sigmoid functions constructed from combinations of implicit primitives or from sampled surface data. Chunyu et al. [21] addressed a reactive control design for point-mass vehicles with limited sensor range to track targets while avoiding static and moving obstacles in a dynamically evolving environment. Enxiu Shi et al. [22] presented an analysis of the common problems found in potential fields for mobile robot such as obstacle avoidance, local trap and vibrating, an evolutionary method for improving potential fields was proposed. Charifa and Bikdash [23] compared the behaviour of several variants of artificial potential function methods with emphasis on the quality of path geometry, and velocity and acceleration profiles. Loizou and Kyriakopoulos [30] proposed a kinematic framework for modelling mobility of multiple robots that mathematically combines heterogeneous locomotion constraints for control of the system.

Today's vehicle-like robots have more complex functions with features of object detection and positioning capabilities Stanek et al. [9], specially long-term navigational tasks Broggi et al. [8]. Numerous kinematic planners that compute the shortest manoeuvring feasible path for vehicles explicitly considering vehicle dynamics is found in Moriwaki and Tanaka [12]; and similarly Werling et al. [6] presented a search for an optimal path using dynamic simulations to determine the traversable or cost of specific terrain segments. Tychonevich et al. [7] presented a work based on selecting a path that is ensured to be statically safe. Such approaches do not account for vehicle dynamics and speed. Other works on multi-layered planners to achieve optimal navigation of high speed mobile robots on outdoors using layered control has been proposed Iagnemma et al. [13], and Plaku et al. [11]. High-level planning layers generate optimal desired trajectories as series of way-points, and low-level layer to guide the vehicle using potential fields. But optimal global trajectory generation must be off-line to compute a topographic model, and then dynamic obstacles are not considered. Fuzzy genetic-based planners has been develop to account for obstacles in terrain regions

to determine vehicle speed, or similar approaches to represent traversable, or very cluttered environments Selekwa et al. [2]. Global trajectory planners that determine traversable directly, use a search where first select a series of “best” traversable paths, which are then further optimized to minimize travel time, but no consideration of detailed vehicle models exist Mann and Shiller [14]. Artificial potential fields have long been successfully employed for robot navigation to solve several navigation problems Masound [4]. To develop real-time obstacle avoidance Huang et al. [1]; trajectory generation Mora and Tornero [3]; dynamic control of mobile robots with moving obstacles and goals Geand and Cui [15] and Selekwa et al. [2]; general robot path planning applied to (non) holonomic platforms Goncalves et al. [5]; simulations on multiple robotic agents moving guided by potential fields along urban intersections lanes was presented in Teja et al. [10].

Our analysis regarding cited references summarizes that directional fields are physics-based models generally solvable equations of first and second order, with implicit control in the model itself for high speeds. Many practical robotic navigation problems have been successfully solved. Local minima and maxima has been largely treated by the robotics community, and currently there exist a variety of available solutions. Directional fields are relatively easy to adapt in multiple robotic systems, regardless, either functional or locomotive robots heterogeneity. Directional fields are proved to have the path geometry quality if adjustments and algebraic restrictions are included (depending on the task). They can be stated as reactive control laws, having great flexibility to be adjusted to any kinematics structure. Some contributions are of practical implementation, while some others are real-time intractable, and definitely are off-line solutions. Our contribution research is another approach of directional fields applied to mobile robotics; the SFM was stated for other scientific applications (gas kinetics and crowd scenarios simulations). We are reformulating this approach to be adapted to WMR for on-line planning with exponential-based distributions. This manuscript presents a framework that includes the kinematics and motion dynamics model in continuous-time merged with a general model that reformulates the SFM to solve the motion-planning problem. The contributions of the present work has the following issues, the SFM is reformulated as a general velocity-based motion framework that expresses internal and external causes and effects of motion constrained to an

ideal and a maximal speeds control. The speed control is based on a functional form of motors rotational speed rate, and the robot’s size to determine the vehicle yaw speed, and with such basis the actual and posterior position vectors are formulated. The combined scheme allows any forward kinematics, since it depends on the locomotion robot’s design, and a weighting factor yielded from multiple sensing features. Using the motion dynamic equations, motion is no longer purely geometric because directions are computed on acceleration components and unit vectors.

This manuscript is organized as follows. In section 2, a description of the problem to be solved is presented. The section 3 introduces the robot main control law model. Section 4 details the general social accelerations model, and the mechanism to yield adaptive numeric weights to affect the robot steering navigation function. Section 5 presents the derivation of a model to describe the robot motion in either local or global framework. Further, section 6 concerns the derivation of a motion dynamics model that describes Newton’s law of motion. We describe an equilibrium condition between causes and effects of motion in any inertial frame. Section 7 describes in detail the formulation to infer the actual and next robot position vector. Section 8 presents the directional fields navigation scheme that is comprised of exponential accelerative vector components. Section 9 discusses the simulation and experimental results. Finally, section 10 summarizes the work conclusions.

2. Problem definition

We state a traditional navigation problem as illustrated by Fig. 1. A WMR μ with fixed inertial frame $\mathbf{x}_t^\mu = (x, y)^T$, heading to θ_t , and passing through a sequence of local goals at \mathbf{x}_t , each with direction \mathbf{m}_t ; while leading the robot toward a goal destination γ . The Cartesian distance between two points is generally defined by the norm of their geometric difference $\|\vec{\delta}_{\mu\alpha}\| = \|\mathbf{x}_t^\mu - \mathbf{x}_t^\alpha\|$ (distance between robot μ and obstacle α_t). Goals are established to exert attractive accelerative fields F_t^γ which easily conduct the robot. Likewise, detected obstacles α exert repulsive accelerative fields F_t^α . Both types of fields, in combination form an enriched directional map. One of the problems in this work is how to reformulate the SFM as a reliable navigation solution for WMR. The SFM was originally presented to solve other scientific problems

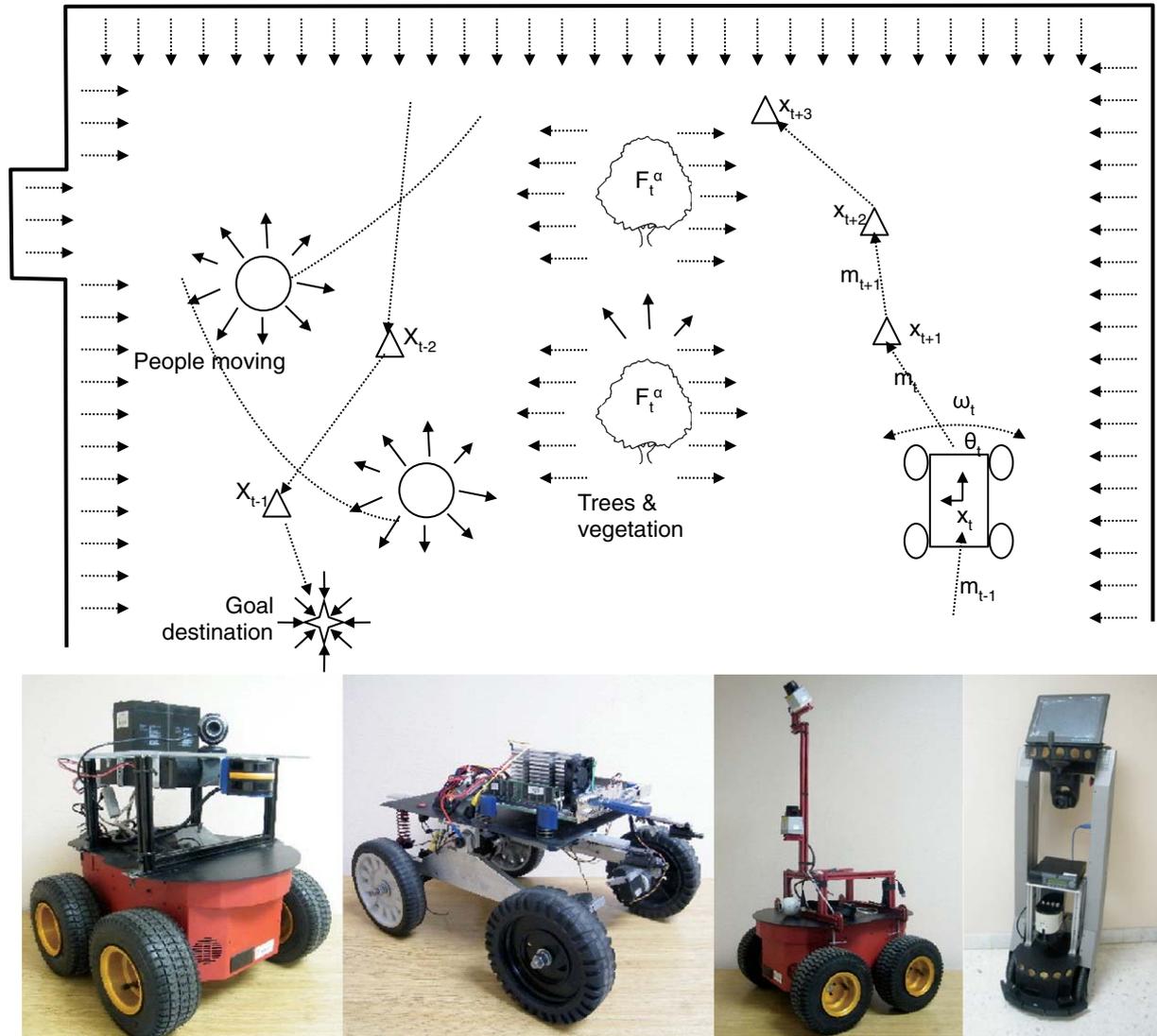


Fig. 1. Above: An outdoor environment with multiple static/dynamic obstacles and goals. Below: Different kinematic structures deployed with the approach.

in different scientific communities, and was presented in a non-integrable form by Helbing and Molnar [16]. Because of the SFM's properties of no-collision of fast motion simulating molecules gas-kinetics, a suitable mathematical form is proposed for practical use in WMR.

3. General navigation model

The reformulation of the social directional field presented in this manuscript begins by introducing a

general control equation to govern the robot's speed by (1). This relationship warrants the vehicle to keep a safe referenced velocity v^o while navigating along the direction θ_t . The factor η is the gain value that if adjusted, defines the control convergence of the speed (see Fig. 2-top).

$$v_t = (\hat{v}_{t-1} + \eta(v_t^o - \hat{v}_t))h \left(\frac{v^{max}}{\|\hat{v}_t\|} \right) \quad (1)$$

The velocity vector v_t is proportionally adjusted using the feedback velocity error w.r.t. a reference

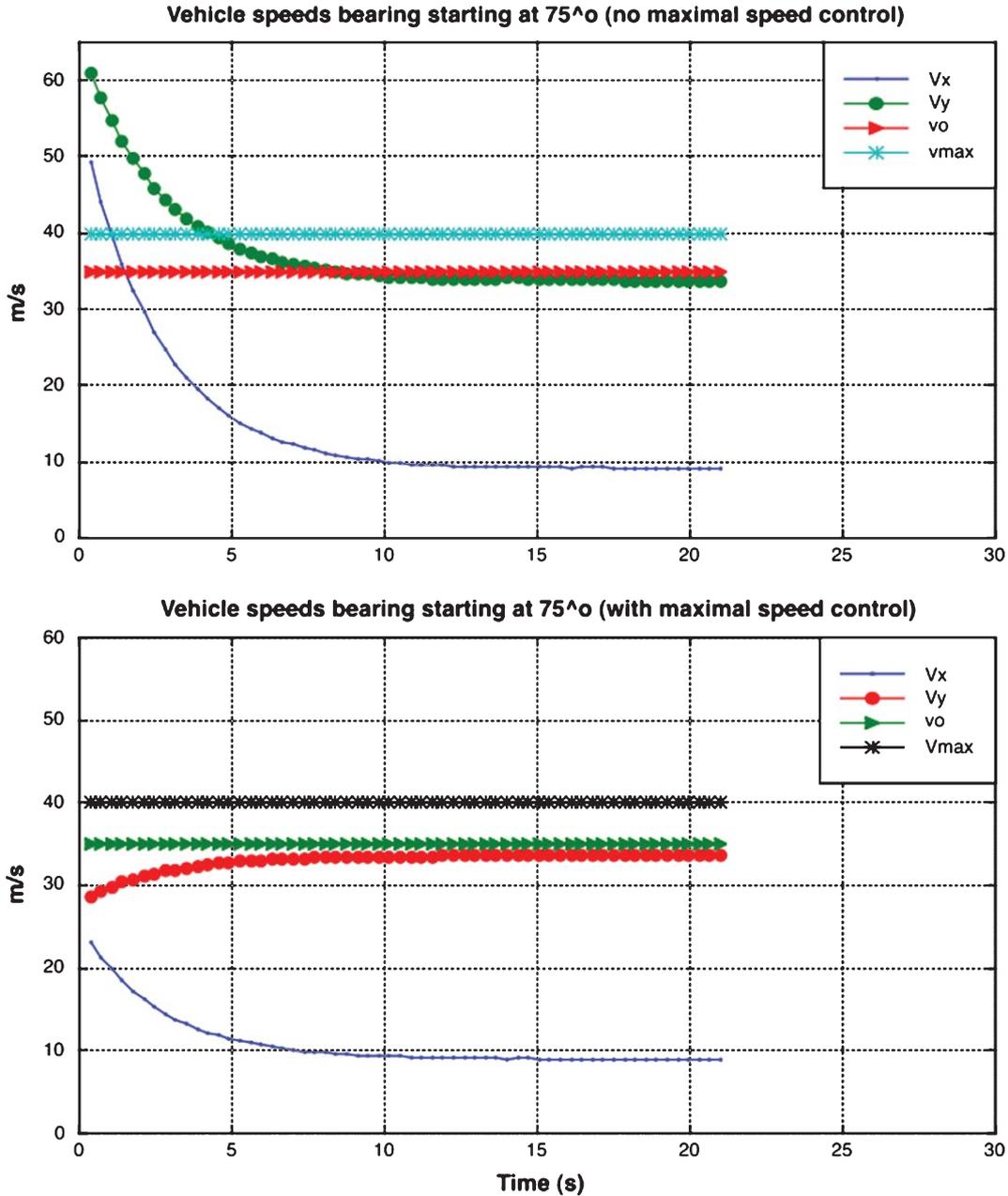


Fig. 2. Above: reference velocity control; Below: maximal allowable velocity control. The parameters for this instance are $\hat{v}_x = 55m/s$ and $\hat{v}_y = 65m/s$ with an angle of motion oriented at 75° with an $\eta = 0.125$, $v^o = 35m/s$ and $v^{max} = 40m/s$.

speed, which in particular happens when the magnitude reduction function $h(\cdot) = 1$. The non-stationary ideal velocity is denoted by $\mathbf{v}^o = v^o(\cos(\theta_t), \sin(\theta_t))^T$. When an unexpected collision suddenly occurs, very short periods of time are generally taken. As a result,

the model (1) controls the velocity peaks exceeding maximal allowable velocities when the function value is in the range $0 < h(\cdot) < 1$, as modelled in equation (2), the \mathbf{v}_t with $h(\cdot) \neq 1$ has behaviour as depicted in 2-bottom.

$$h(v^{max}, \|\hat{\mathbf{v}}_t\|) = \begin{cases} 0, & \|\hat{\mathbf{v}}_t\| = 0 \\ 1, & 0 < \|\hat{\mathbf{v}}_t\| \leq v^{max} \\ \frac{v^{max}}{\|\hat{\mathbf{v}}_t\|}, & v^{max} < \|\hat{\mathbf{v}}_t\| \end{cases} \quad (2)$$

The velocity model (1) recursively controls two aspects; first, the real velocity fluctuating around the ideal velocity v^o magnitude; and second, removing divergent magnitudes overpassing a maximal allowable velocity value. The real velocity vector $\hat{\mathbf{v}}_t$ at actual time t is defined in (3), the real velocity model in this context will involve the motion causes, motion effects, and random fluctuations perturbing acceleration components. The real velocity vector $\hat{\mathbf{v}}_t$ is expressed in terms of two global accelerative components that yield $\frac{d\hat{\mathbf{v}}}{dt}$ as in equation (3). One term is the social directional field vector is $\mathbf{F}_t = (f_x, f_y)^T$ expresses the internal and external causes of motion by Newton's 2nd law of motion \mathbf{F}/m with $m = 1$, affecting the WMR motion. The second term $\mathbf{a}_t = (a_x, a_y)^T$ is the general acceleration representing accelerative behaviour for any inertial system (global \mathbf{a}^I , or local \mathbf{a}^R).

$$\frac{d\hat{\mathbf{v}}}{dt} = \mathbf{F}_t - \mathbf{a}_t^I \quad (3)$$

4. Social field functions

Social field functions \mathbf{F}_t stated in (3) as the general real motion model $\hat{\mathbf{v}}_t$, expresses robot's global behaviour involving dynamic causes of motion. We encompass three causes of motion dynamics (sensors are deployed in the process for detection), *next desired goals* (an approach is presented by visual detection in Fig. 3), *obstacles position* (LIDARs), and *final goal destinations*. Eq. (4) models the social directional fields described in terms of global accelerations \mathbf{F}_t . Cases of motion are internal (\mathbf{F}_t^α), and external (\mathbf{F}_t^γ and \mathbf{F}_t^β).

$$\mathbf{F}_t = \mathbf{F}_t^o + \sum_{\alpha} \mathbf{F}_t^\alpha + \sum_{\gamma} \mathbf{F}_t^\gamma \quad (4)$$

Since the robot's navigation depends on sensor observations, only sensor data feature are used as regions of interest to exert weighted navigation functions. Each accelerative force is defined with an adaptive numeric weight $w(\mathbf{m}_t, \mathbf{f}_t)$ yielded by the bearing location of the targets (local goal destination, or obstacles) within the sensors field of view as defined in equation (5).

$$\mathbf{F}_t = \mathbf{F}_t^o + \sum_{\alpha} w(\mathbf{m}_t, -\mathbf{f}_t^\alpha) \mathbf{f}_t^\alpha + \sum_{\gamma} w(\mathbf{m}_t, \mathbf{f}_t^\gamma) \mathbf{f}_t^\gamma \quad (5)$$

The repulsive and attractive behaviour, which affect the vehicle's behaviour accentuate the magnitudes of the motion functions given in expressions (6) and (7), where $\vec{\delta}_\mu = \mathbf{x}_t^\alpha - \mathbf{x}_t^\mu$ is a distance vector between the positions of a goal/obstacle and the actual vehicle μ , and the $\mathbf{m}_t = (\mathbf{x}_{t+1}^\mu - \mathbf{x}_t^\mu) / \|\mathbf{x}_{t+1}^\mu - \mathbf{x}_t^\mu\|$ is a unit vector expressing the direction towards a next desired location \mathbf{x}_{t+1} . Thus, the weighting factor w_t will affect the repulsive acceleration behaviour according to,

$$\mathbf{F}_t^\alpha(\mathbf{m}, \vec{\delta}_{\mu\alpha}) = w(\mathbf{m}_t, -\mathbf{f}_t^\alpha) \mathbf{f}(\vec{\delta}_{\mu\alpha}) \quad (6)$$

similarly the weighting factor will affect the attractive acceleration by,

$$\mathbf{F}_t^\gamma(\mathbf{m}, \vec{\delta}_{\mu\gamma}) = w(\mathbf{m}_t, \mathbf{f}_t^\gamma) \mathbf{f}(\vec{\delta}_{\mu\gamma}) \quad (7)$$

The influence of the weight w_t depends on how parallel the sensing direction ϕ_t of a goal/obstacle and the actual acceleration \mathbf{f} are, as described in expression (8). If the actual orientation of the vector acceleration \mathbf{f} is about the same as the actual vehicle desired orientation \mathbf{m}_t , then no change of direction is required for the vehicle. It is expected that the orientation of the goal/obstacle sensed at bearing ϕ_t is approximately along the direction of the next desired position. But, if the orientations of vectors \mathbf{m}_t and ϕ_t are different, then it means that the horizontal component $\mathbf{f}_t \cos \phi$ must be decreased by the vehicle yaw changes.

$$w_t = \begin{cases} 1, & \mathbf{m}_t \cdot \mathbf{f}_t \geq \|\mathbf{f}_t\| \cos(\phi_t^i) \\ \lambda_t, & \text{otherwise} \end{cases} \quad (8)$$

The influence of rotations that the vehicle must carry out is given by an influence term λ_t which is an average of the fusion of all multi-sensory observations. As we established that the vehicle is instrumented with s_n different sensor devices i . Thus, λ_t is valued within the range $0 \leq \lambda \leq 1$ based on an effective angle of view ϕ_t ,

$$\lambda_t = \sin \left(\frac{1}{s_n} \sum_i \frac{\phi_t^i}{\pi} \right) \quad (9)$$

Where s_n is the total number of sensors involved in the perception of the objective (goal/obstacle), and $(\phi_t^i)^{s_n}$ are the angles at which each sensor i detected the same objective. In fact, expression (9) defines a greater

Table 1
Algorithm ROBOT'S SENSOR DATA

- 1: **read-from-lidar**(θ_t, \mathbf{x}_t)
- 2: $\mathbf{Z}_t = \text{read-lidar}()$; $\mathbf{Z}_t = \{\mathbf{z}_t^1, \dots, \mathbf{z}_t^m\}$
- 3: $\mathbf{z} = (\phi_i, d_i)^T$;
- 4: $\mathbf{x}_t^{s_i} = (d_i \cos(\phi_i), d_i \sin(\phi_i))^T$
- 5: $\mathbf{X}_t = \{\mathbf{x}_t^{s_1}, \dots, \mathbf{x}_t^{s_m}\}$
- 6: $\mathbf{X}_t^I = \mathbf{R}(\theta_t) \cdot \mathbf{x}_t^{s_i} + \mathbf{x}_t$
- 7: **return** \mathbf{X}_t^I

numeric weight to objectives located nearly along the longitudinal vehicle's axis (fixed-frame, defined at 90°). Sensing modality for environment mapping is by deploying a laser range finder. The important features, which the robot is able to perceive are very critical because on this issue, the robot defines the numeric weighting factors to impact significantly the navigation functions. The Table 1 defines \mathbf{z} as the vector of a polar measurement (one point among the 681 of the laser scan). Likewise, $\mathbf{x}_t^{s_i}$ defines the Cartesian representation of a point \mathbf{z}^i . The Table 1 directly returns a sensor observation in a global Cartesian representation \mathbf{X}_t^I .

In outdoor experiments under natural light conditions [35–37] the attractive areas \mathbf{x}_t^γ (middle-road triangles) and repulsive features \mathbf{x}_t^α (line lanes circles) were detected on-line using vision as depicted by the sequence of images of Fig. 3. The attractive local goals (γ) are featuring the next desired positions $\{\mathbf{x}_t^\gamma, \mathbf{x}_{t+1}^\gamma \dots\}$. Line lanes regions represent repulsive objectives (α). Other feature points over

the horizon and/or dynamic and static obstacles are used for complementing localisation by monocular visual odometry, which is out of the scope of this manuscript.

5. Inertial frames motion

Definition of robot's motion must be described in local/global Cartesian frames to represent acceleration maps (Fig. 11). We define a generalised particle motion scheme, in which neither causes of motion, nor vehicle kinematic restrictions are regarded. Such scheme is useful to model accelerative motion behaviour denoted by \mathbf{a}_t , already described by eq. (3) to describe part of the real acceleration $\frac{d\mathbf{v}}{dt}$. Let us consider the linear velocity components of any vehicle-like robot with averaged velocity v_t . By defining the velocity vector $\mathbf{v}_t^R = (v_x, v_y)^T$ in the vehicle fixed-frame, the components XY represent the $2D$ plane of motion and is given by the expression (10),

$$\mathbf{v}_t^R = v_t \begin{pmatrix} \cos(\theta_t) \\ \sin(\theta_t) \end{pmatrix} \quad (10)$$

Where θ_t is the vehicle's angle of motion w.r.t. robot's initial posture. By transforming the original vehicle fixed-frame using a transformation matrix \mathbf{R} with rotation angle ψ between the robot's frame, and the global system. The new expression for the global frame becomes as expressed by (11), which is the velocity

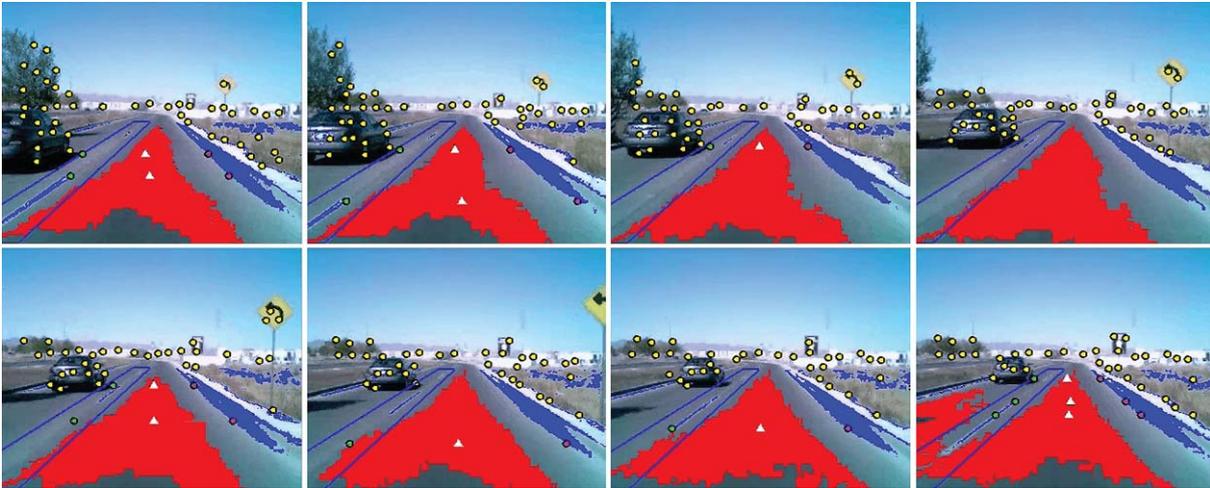


Fig. 3. Outdoor experiments using vision were realised to detect ϕ_i corresponding to γ and α . The triangles define attraction areas for the weighting function $w(\mathbf{u}, \mathbf{f}_t^\gamma)$ to impact $\mathbf{F}_t^\gamma(\delta)$.

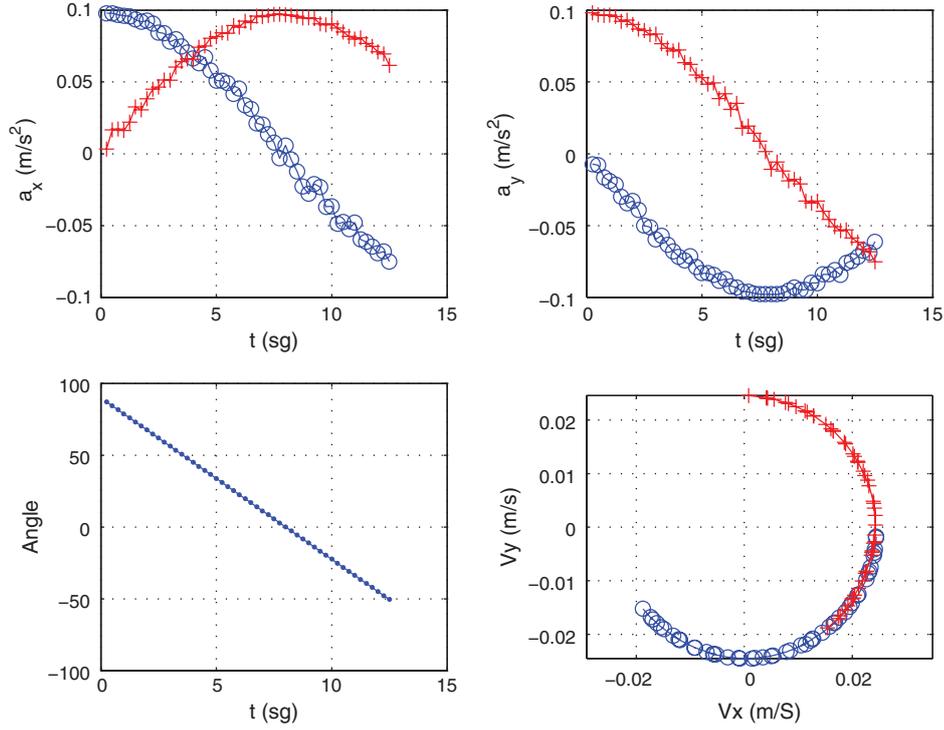


Fig. 4. Inertial frames descriptive model (12) yielded results of accelerations represented in two inertial frames, local frame (blue circles), and global frame (red crosses), with $\psi = \frac{\pi}{2}$ and θ_t .

behaviour without wheels kinematic constraints (see Appendix A for algebraic details on (12)).

$$\mathbf{v}_t^I = v_t \begin{pmatrix} \cos(\theta_t + \psi_t) \\ \sin(\theta_t + \psi_t) \end{pmatrix} \quad (11)$$

Now, the acceleration vector in global frame is obtained by eq. (12),

$$\mathbf{a}_t^I = v_t(\dot{\theta}_t + \dot{\psi}_t) \begin{pmatrix} -\sin(\theta_t + \psi_t) \\ \cos(\theta_t + \psi_t) \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos(\theta_t + \psi_t) \\ \sin(\theta_t + \psi_t) \end{pmatrix} \quad (12)$$

Inversely transforming the acceleration into the robot's coordinate framework, and without loss of generality, the resulting simplified mathematical expression is now written in eq. (13) as the robot's local frame, hence $\dot{\psi} = 0$.

$$\mathbf{a}_t^R = v_t \dot{\theta} \begin{pmatrix} -\sin(\theta_t) \\ \cos(\theta_t) \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos(\theta_t) \\ \sin(\theta_t) \end{pmatrix} \quad (13)$$

Eq. (12) is about the same as eq. (13). In the former, the rotation frame angle ψ is being considered for transformation into the global frame. Latter expression has no rotated inertial frames, hence $\psi = 0$. Hereafter, eq. (12) may be used as the general frame solution. Being $\psi_t \neq 0$ when global inertial Cartesian frame is required.

Simulation results using eq. (12) yields Fig. 4. The robot moved while decreasing its angle at constant rate from initially 90° up to -50° , drawing a curved trajectory. Blue circle curves depict local frame data, and red crosses plot robot's motion in global coordinates for $\psi_t = \pi/2$.

6. Dynamics model derivation

According to eq. (3), we derive the motion dynamics where robot's internal and external causes are explicitly defined. According to Newton's law of motion, the next equation establishes that the sum of all accelerations in the system is equal to the global acceleration,

$$a_t = \frac{1}{m} \sum_i \sqrt{fx_i^2 + fy_i^2} \equiv \sum_i \sqrt{fx_i^2 + fy_i^2} \quad (14)$$

For the sake of analysis, we define an unitary mass $m = 1$. By simplifying, eq. (14) becomes (15). The acceleration \mathbf{a}_t is known as we provide an explicit functional form denoting robot's global behaviour in any inertial frame. In addition, \mathbf{F}_t is defined although involving descriptive equations of dynamic effects. We state that the approximated real acceleration model is denoted by the equilibrium condition,

$$\mathbf{a}_t^R = \mathbf{F}_t^\mu \quad (15)$$

We state that eq. (15) describes the boundary case of equilibrium for $\mathbf{a}_t = 0$ by rising the following statements,

- (1) condition $d\hat{\mathbf{v}}_t/dt = 0$ for uniform velocity, no acceleration is performed and $\hat{\mathbf{v}}_t$ is constant.
 - a) therefore for eq. (1), $\hat{\mathbf{v}}_{t-1} \equiv \hat{\mathbf{v}}_t$, and $\eta = 1$, then $\mathbf{v}_t = 0$
- (2) the condition for eq. (1), $\hat{\mathbf{v}}_t = 0$, when the robot is initially stopped, or when it reached its final goal destination.
 - a) therefore for eq. (1), $\hat{\mathbf{v}}_t = 0$ and $\eta = 1$, then $\mathbf{v}_t = 0$

when such limit case condition occurs, sum of all accelerations will meet the condition for the equilibrium case when,

$$\frac{d\hat{\mathbf{v}}}{dt} = \mathbf{F}_t - \mathbf{a}_t^R = 0$$

The equilibrium condition of global accelerative model is then demonstrated by algebraic development to validate the expression (15). By substituting (12) and (4) in (3) with $\psi_t = 0$ (both described in common inertial frame) as follows,

$$v_t \dot{\theta}_t \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos \theta_t \\ \sin \theta_t \end{pmatrix} = \mathbf{F}_t^o + \sum_{\alpha} \mathbf{F}_t^{\alpha} + \sum_{\gamma} \mathbf{F}_t^{\gamma} \quad (16)$$

The term \mathbf{F}_t^o is the robot's internal motivation with functional form in (17), which makes the robot move along multiple local goals. Where $\delta = \|\mathbf{x}_{t+1} - \mathbf{x}_t\|$ is the distance between the actual robot location \mathbf{x}_t and the next desired goal \mathbf{x}_{t+1} .

Table 2

Algorithm DIRECTIONAL FIELD FOR INTERNAL MOTIVATION

- 1: **internal-motivation**(v^o, τ)
- 2: $\mathbf{v}_t = v_t(\cos(\theta_t + \psi_t), \sin(\theta_t + \psi_t))^T$
- 3: $\mathbf{X}_t^I = \mathbf{read-from-lidar}(\theta_t, \mathbf{x}_t)$
- 4: $\mathbf{x}_t^s = \mathbf{feature-extraction}(\mathbf{X}_t^I)$
- 5: $\|\delta\| = (\mathbf{x}_t^s - \mathbf{x}_t)^{1/2}$
- 6: $\mathbf{x}_t = \mathbf{actual-position}$
($\hat{\phi}_t^1, \dots, \hat{\phi}_t^n, W, L, r, x_0, y_0$)
- 7: $\mathbf{x}_{t+1} = \mathbf{next-desired-position}(\mathbf{x}_t, \phi_t, \theta_t, w_t)$
- 8: $\mathbf{F}_t^o = \frac{1}{\tau} \left(\frac{v^o}{\|\delta\|} (\mathbf{x}_{t+1} - \mathbf{x}_t) - \mathbf{v}_t \right)$
- 9: **return** \mathbf{F}_t^o

$$\mathbf{F}_t^o = \frac{1}{\tau} \left(\frac{v^o}{\|\delta_t\|} (\mathbf{x}_{t+1} - \mathbf{x}_t) - \mathbf{v}_t \right) \quad (17)$$

Likewise, the ideal linear speed v^o sets a desired speed in xy components. There is a vector of actual measured velocity \mathbf{v}_t , and a relaxation time τ that defines the time taken for speeds change.

The internal motivation is implemented through Table 2.

Arranging (16) by dropping off the velocity measurement, the next algebraic steps are developed

$$v_t \dot{\theta}_t \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos \theta_t \\ \sin \theta_t \end{pmatrix} = \frac{1}{\tau} \left(\frac{v^o}{\|\delta_t\|} (\mathbf{x}_t - \mathbf{x}_{t+1}) - \mathbf{v}_t \right) + \sum_{\alpha} \mathbf{F}_t^{\alpha} + \sum_{\gamma} \mathbf{F}_t^{\gamma} \quad (18)$$

therefore,

$$\tau \left(v_t \dot{\theta}_t \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos \theta_t \\ \sin \theta_t \end{pmatrix} - \sum_{\alpha} \mathbf{F}_t^{\alpha} - \sum_{\gamma} \mathbf{F}_t^{\gamma} \right) = \frac{v^o}{\|\delta_t\|} (\mathbf{x}_t - \mathbf{x}_{t+1}) - \mathbf{v}_t \quad (19)$$

We are treating the usual condition where both v_t and $\dot{\theta}_t$ are approximately uniforms in small periods of time τ (with very small variations for τ), where in our context we define that $\int_t v_t \dot{\theta}_t \sin \theta_t dt \approx v_t \theta_t \sin \theta_t$, since $\sin \theta_{t-1} \approx \sin \theta_t$.

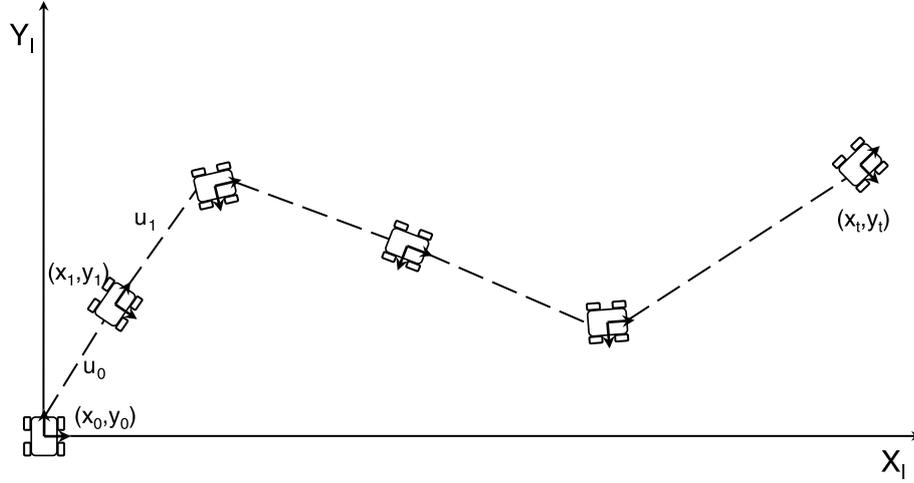


Fig. 5. Vehicle displacement overtime.

$$v_t \theta_t \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} + v_t \begin{pmatrix} \cos \theta_t \\ \sin \theta_t \end{pmatrix} - \sum_{\alpha} \mathbf{v}_t^{\alpha} - \sum_{\gamma} \mathbf{v}_t^{\gamma} \\ = \frac{v^o}{\|\delta_t\|} (\mathbf{x}_t - \mathbf{x}_{t+1}) - \mathbf{v}_t \quad (20)$$

Hence, due to units of time of τ , left-sided equation terms changed from m/s^2 into m/s , for which there is an integrable functional form for $\hat{\mathbf{v}}_t$. Thus, by dropping off the real approximated velocity vector,

$$\hat{\mathbf{v}}_t = \frac{v^o}{\|\delta_t\|} (\mathbf{x}_t - \mathbf{x}_{t+1}) + \sum_{\alpha} \mathbf{v}_t^{\alpha} + \sum_{\gamma} \mathbf{v}_t^{\gamma} \\ - v_t \left(\theta_t \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} - \begin{pmatrix} \cos \theta_t \\ \sin \theta_t \end{pmatrix} \right) \quad (21)$$

Our equation has now been solved in (22) as to have a model of motion that combines the robot's fixed-frame, with external dynamic constraints,

$$\hat{\mathbf{v}}_t = \frac{v^o}{\|\delta_t\|} (\mathbf{x}_t - \mathbf{x}_{t+1}) + \sum_{\alpha} \mathbf{v}_t^{\alpha} + \sum_{\gamma} \mathbf{v}_t^{\gamma} \\ - v_t \begin{pmatrix} \sin \theta_t & \cos \theta_t \\ -\cos \theta_t & \sin \theta_t \end{pmatrix} \cdot \begin{pmatrix} -\theta_t \\ 1 \end{pmatrix} \quad (22)$$

7. Position model

The robot position vector $\mathbf{x}_t = (x, y)^T$ is a summation of all estimated positions overtime with respect to a common inertial frame from its starting position up to

actual time. As in (22), obtaining the vector \mathbf{x}_t (Fig. 5) calculating the position vector $\hat{\mathbf{v}}_t$. The position vector is calculated by involving the actuators rotational kinematic model to quantify displacements. Most commonly used techniques, Wang and Low [31] are used to compute the angular velocity based on discrete odometry strategies (though prone to incrementally fail when skids and/or slips occur). We propose a solution to deduce \mathbf{x}_t based on controlling the wheels actuator. We combine our approach with an inertial-based localization method, using an in-house made unit system (for now out of the scope of this work), as such algorithm has been previously reported in Martínez-García et al. [26]. Although, the kinematic parameters are fundamentals for any planning algorithm as described by Campion et al. [28], we provide a general solution that only consider any robot's angular velocity ω_t equation.

$$\omega_t = Kg(\dot{\phi}_t^1, \dots, \dot{\phi}_t^k) \quad (23)$$

Where K is a constant, and the function $g(\cdot)$ represents the yaw rate model with wheels rotational velocities $\dot{\phi}_t^j$ as input parameters (its number will depend on the type of kinematic structure). In eq. (23), the angular velocity is directly controlled by the wheels rotation. It is worth highlighting that the framework allows integration of other kinds of kinematic constraints by changing the ω_t model accordingly, such as the Ackerman type, Campion et al. [28], and Macek et al. [33]; synchronised type Fox et al. [29]; differential drive Song and Chang [32]; or the platforms studied by Loizou and Kyriakopolous [30].

Table 3

Algorithm ACTUAL ROBOT POSITION

- 1: **actual-position** ($\dot{\varphi}_t^1, \dots, \dot{\varphi}_t^n, x_0, y_0$)
- 2: $\Phi_t =$ **robot-actual-orientation**
($\theta_0, K, \dot{\varphi}_t^1, \dots, \dot{\varphi}_t^n$)
- 3: $\theta_t = \theta_0 + \Phi_t$
- 4: $\mathbf{x}_t = (x_0, y_0)^T + \int_t v_0$
 $+ \int_t \dot{v}_t dt (\cos \theta_t, \sin \theta_t)^T dt$
- 5: **return** \mathbf{x}_t

The approach to infer \mathbf{x}_t and \mathbf{x}_{t+1} is by quantifying the wheels angular displacement directly by the speed drivers. We take advantage of the control hardware (motor drivers) which works under asymptotic functions (although a non-linear motor speed curve will vary from product to product). A general relationship between actuator's angular speed $\dot{\varphi}_t$ and a digital control variable Ω_t is kinematically given by

$$\dot{\varphi}_t(\Omega) = \left(\frac{a}{1 + e^{-\Lambda\Omega - \mu}} \right) - b \quad (24)$$

Where a and b are constants that adjust the non-linear angular velocity behaviour curve, Λ is the constant of fast asymptotic fall, Ω is a control digital word which is associated with an angular speed given directly by a user program, and μ is the central value of the velocity curve. By solving (24), we integrate the equation to obtain the next expression (25) in terms of wheels instantaneous angle of rotation,

$$\int_a^b \dot{\varphi}(\Omega) d\Omega = \varphi(\Omega) = (a - b)\Omega + \frac{a}{\Lambda} \ln(1 + e^{\Lambda(\mu - \Omega)}) \quad (25)$$

We synthesize the robot's direction and deduce a formal position model equation as expressed in the vector form by (26) with the k rotation velocities $\dot{\varphi}_t^k$.

$$\mathbf{x}_t = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \int_{t_0}^{t_n} (v_0 + \int_{t_1}^{t_n} \dot{v}_t dt \begin{pmatrix} \cos(\theta_0 + K \int_{t_1}^t g(\dot{\varphi}_t^1, \dots, \dot{\varphi}_t^k) dt \\ \sin(\theta_0 + K \int_{t_1}^t g(\dot{\varphi}_t^1, \dots, \dot{\varphi}_t^k) dt \end{pmatrix} dt) dt \quad (26)$$

Inferring the robot's actual position is synthesised in pseudo-code form by the algorithm 3. Nevertheless, the problem of robot skid/slip is overcome by combining with the method reported in Martínez-García et al. [26] that deploys an in-house made inertial unit. It works reasonable because yaw rates can directly be controlled by using low level commands. Thus, by simplifying previous expression,

$$\Phi_t = K \int_{t_1}^{t_n} g(\dot{\varphi}_t^1, \dots, \dot{\varphi}_t^n) dt \quad (27)$$

Substituting (25) in (27) and algebraically solving,

$$\Phi_t = K \left((a - b)\Omega^r + \frac{a}{\Lambda} \ln(1 + e^{\Lambda(\mu - \Omega^r)}) - (a - b)\Omega^l + \frac{a}{\Lambda} \ln(1 + e^{\Lambda(\mu - \Omega^l)}) \right) \quad (28)$$

then,

$$\Phi_t = K \left((a - b)(\Omega^r - \Omega^l) + \frac{a}{\Lambda} \ln \left(\frac{1 + e^{\Lambda(\mu - \Omega^r)}}{1 + e^{\Lambda(\mu - \Omega^l)}} \right) \right) \quad (29)$$

We assume similar approach as Fox et al. [29], where a dynamic motion algorithm derived from the robot's general dynamics was presented. Hence, the actual position vector is written as,

$$\mathbf{x}_{t_n} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \int_{t_0}^{t_n} (v_0 + \int_{t_1}^{t_n} \dot{v}_t dt \begin{pmatrix} \cos(\theta_0 + \Phi_t) \\ \sin(\theta_0 + \Phi_t) \end{pmatrix} dt) dt \quad (30)$$

The orientation θ_t is solved by integration w.r.t. the time interval $[t_0, t_n]$, in which wheels rotations are controlled rather than collecting absolute odometry measurements (as commonly proposed by other approaches). Thus, reformulating the robot's angle by

$$\theta_t = \theta_0 + K \int_{t_1}^t g(\dot{\varphi}_t^1, \dots, \dot{\varphi}_t^n) dt \quad (31)$$

and solving for the instantaneous angle,

$$\theta_t = \theta_0 + \Phi_t \quad (32)$$

We assumed that the magnitude of the vehicle's angular acceleration $d\omega/dt$ at every control loop is

much smaller than the magnitude of the angular velocity. The robot's orientation algorithm is presented in Table 4, which obtains the actual robot's angle. The skidding and slipping aspects are treated in Martínez-García et al. [27].

Arranging the actual position vector to be algorithmically implemented in terms of the robot kinematic

Table 4

Algorithm ROBOT'S ACTUAL ORIENTATION

- 1: **robot-actual-orientation** ($\theta_0, K, \dot{\phi}_t^1, \dots, \dot{\phi}_t^n$)
- 2: Set constants $\Omega, a, b, \Lambda, \mu$
- 3: $\Phi_t = K \left((a-b)(\Omega^r - \Omega^l) + \frac{a}{\Lambda} \ln \frac{1+e^{\Lambda(\mu-\Omega^r)}}{1+e^{\Lambda(\mu-\Omega^l)}} \right)$
- 4: $\theta_t = \theta_0 + \Phi_t$
- 5: **return** θ_t

structure,

$$\mathbf{x}_t = \begin{pmatrix} \dot{x}_{t+1} \\ \dot{y}_{t+1} \end{pmatrix} \Delta t + \frac{r}{2} g(\dot{\phi}_t^r, \dots, \dot{\phi}_t^l) \Delta t \begin{pmatrix} \cos(\theta_0 + \Phi_t) \\ \sin(\theta_0 + \Phi_t) \end{pmatrix} \quad (33)$$

According to Fig. 6 to alter the robot's orientation towards a next desired destination \mathbf{x}_{t+1} , the control is based on the collection of consecutive sensor data. The next desired position \mathbf{x}_{t+1} determines a Cartesian objective, either attractive or repulsive. Rather than a Cartesian point, this objective is defined as an area. Furthermore, the next position vector model arises from a function \mathbf{f}'_t that uses the actual orientation θ_t .

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \int_t w_t(\mathbf{m}_t, \mathbf{f}'_t) \mathbf{f}'_t(\delta_\mu) dt \quad (34)$$

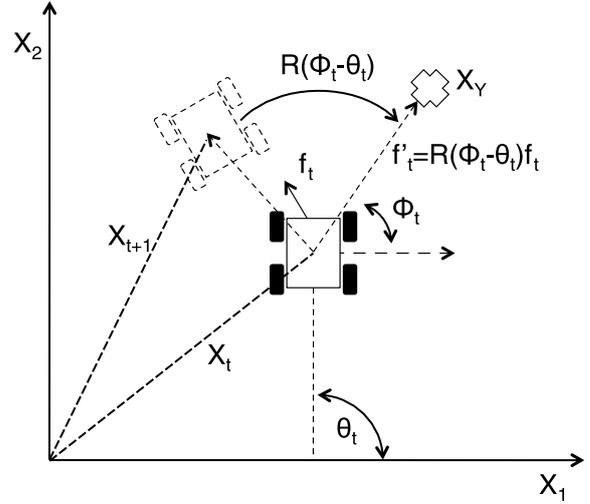
In this approach we alter the actual orientation θ_t by weighting the accelerative navigation function $w_t(\cdot) \mathbf{f}_t^{\gamma, \alpha}$ previously given in eq. (6) and eq. (7). The algorithm 5 basically computes the next desired position by calculating a weighting factor w_t w.r.t. ϕ_i arisen from a sensor feature (obstacle or goal) to impact the navigation functions.

The fundamentals of this algorithm is focused on eq. (8) describing $\mathbf{m}_t \cdot \mathbf{f}_t$. This expression quantifies the alignment of perpendicularity between yaw

Table 5

Algorithm NEXT ROBOT'S DESIRED POSITION

- 1: **next-desired-position** ($\mathbf{x}_t, \phi_i, \theta_t, w_t$)
- 2: $\mathbf{m}_t = (\mathbf{x}_{t+1}^\mu - \mathbf{x}_t^\mu) / \|\mathbf{x}_{t+1}^\mu - \mathbf{x}_t^\mu\|$
- 3: **if** $\mathbf{m}_t \cdot \mathbf{f}_t \geq \|\mathbf{f}_t\| \cos(\phi_t^i)$
- 4: **then** $w_t = 1$
- 5: **else** $w_t = \sin\left(\frac{1}{s_n} \sum_i \frac{\phi_i}{\pi}\right)$
- 6: **endif**
- 7: $\mathbf{f}'_t = \text{new-robot-bearing}(\phi_i, \theta_t, \mathbf{f}_t)$
- 8: $\mathbf{x}_{t+1} = \mathbf{x}_t + \int_t w_t \mathbf{f}'_t dt$
- 9: **return** \mathbf{x}_{t+1}

Fig. 6. Geometric definition of \mathbf{f}'_t

rate and a desired orientation \mathbf{m}_t . If \mathbf{m}_t and \mathbf{f}_t are approximately aligned, then it means that the velocity orientation is projected along the actual desired goal and altering direction is not required. However, $\|\mathbf{f}_t\| \cos(\phi)$ is the acceleration magnitude along the horizontal axis (common frame) respect to objective angle ϕ . A very small value of $\|\mathbf{f}_t\| \cos(\phi)$, signifies that practically no change in direction is required. If such magnitude is too large, an important correction in orientation must be established through the weighting factor λ . The objective (attractive or repulsive) is represented by sensor data features, the more sensors detect the same feature, the more certainty about the direction objective will improve the weighting factor λ . If ϕ is very near or along the vehicle heading axis (about 90°), then $\lambda = 1$ approximately (see eq. (9)). The actual accelerative force \mathbf{f}_t is altered and defined as \mathbf{f}'_t , there is an objective angle correction $(\phi - \theta_t)$, thus, the direction of $\|\mathbf{f}_t\|$ is rotated by eq. (35),

$$\mathbf{f}'_t = \mathbf{R}(\phi_i - \theta_t) \mathbf{f}_t \quad (35)$$

where $\mathbf{R}(\phi - \theta_t)$ an Euler rotation matrix that corrects the yaw. Thus, extending the expression, we now have,

$$\mathbf{f}'_t = \begin{pmatrix} f_x \cos(\phi_i - \theta) - f_y \sin(\phi_i - \theta) \\ f_x \sin(\phi_i - \theta) - f_y \cos(\phi_i - \theta) \end{pmatrix} \quad (36)$$

by developing in the vector form $\mathbf{f}'_t = (f_x, f_y)^T$, the new next-position vector,

$$\mathbf{x}_{t+1} = \begin{pmatrix} x_t \\ y_t \end{pmatrix} + \iint_t \mathbf{f}'_t \, d^2t = \begin{pmatrix} x_t \\ y_t \end{pmatrix} + \iint_t \begin{pmatrix} f_x \cos(\phi - \theta_t) - f_y \sin(\phi - \theta_t) \\ f_x \sin(\phi - \theta_t) - f_y \cos(\phi - \theta_t) \end{pmatrix} d^2t \quad (37)$$

The new desired direction w.r.t. the actual orientation is given by the vector \mathbf{f}'_t , which is a transformation into the global coordinate frame, since observations are locals. See Table 6,

Table 6

Algorithm NEW ROBOT'S DESIRED BEARING

- 1: **new-robot-bearing**($\phi_i, \theta_t, \mathbf{f}_t$)
- 2: $\mathbf{f}'_t = \mathbf{R}(\phi_i - \theta_t)\mathbf{f}_t$
- 3: **return** \mathbf{f}'_t

8. Navigation using directional fields

A gradient vector field assigns the gradient of some function to each Cartesian point. The potential function approach directs a robot as if it were moving in a gradient vector field. Gradients can be viewed as accelerations acting on a positive sense, attracted to the negative goal. Obstacles also have a positive sense which forms a repulsive acceleration directing the robot away from obstacles. The combination of repulsive and attractive accelerations directs the robot from the start location to the goal location while avoiding obstacles. Thus, a potential function is a differentiable real-valued function, which can be viewed as energy, and hence the gradient of the potential is acceleration. The gradient is a vector which points in the direction that locally maximises the function.

8.1. Repulsive function

Repulsive potential fields are suitable navigation functions that define the path course of a robot to safely avoid collisions. There exist an important number of potential functions that describe numerous behaviours Choset et al. [39]. The actual research work follows the approach of the social force model, which is based on exponential distributions. For the case of obstacles avoidance, eq. (38) is a general exponential-based potential field function. f_t^α is a scalar that exhibits a

Table 7

Algorithm REPULSIVE ACCELERATIVE FORCE

- 1: **repulsive-acceleration**($u_\alpha^o, R, \mathbf{x}_t$)
- 2: $\mathbf{X}_t^I = \text{read-from-lidar}(\theta_t, \mathbf{x}_t)$
- 3: **for** $j=1$ **to** $\text{num-obstacles}(\mathbf{X}_t^I)$
- 4: $\vec{\delta}_t = \mathbf{x}^\alpha - \mathbf{x}_t$
- 5: $\frac{d\mathbf{v}_t^\alpha}{dt} = \frac{u_\alpha^o}{\|\vec{\delta}_t\|} e^{\|\vec{\delta}_t\|/R} \left(\frac{1}{\|\vec{\delta}_t\|} - \frac{R}{\vec{\delta}_t} \right) (x_\alpha - x_\mu, y_\alpha - y_\mu)^T$
- 6: $\mathbf{F}_t^\alpha = \mathbf{F}_t^\alpha + \frac{d\mathbf{v}_t^\alpha}{dt}$
- 7: **endfor**
- 8: **return** \mathbf{F}_t^α

general 1D repulsive potential behaviour (Fig. 7 right). u_α^o is a constant that adjusts the acceleration amplitude. R is a stationary value defining the territorial object region, or defined as the asymptotic potential falling value.

$$f_t^\alpha = -\nabla_{\mu\alpha} u_\alpha^o R \frac{e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R}}{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|} \quad (38)$$

The denominator is determined by the factor ($R^{-1}\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|$) and defines the function to respond fast against situations in too close interaction with obstacles. Solving for its gradient operator, we obtain that $\mathbf{f}_t^\alpha \equiv \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$ (see Appendix B).

$$\mathbf{f}_t^\alpha = u_\alpha^o \frac{e^{\|\vec{\delta}_{\mu\alpha}\|/R}}{\|\vec{\delta}_{\mu\alpha}\|} \left(\frac{1}{\|\vec{\delta}_{\mu\alpha}\|} - \frac{R}{\vec{\delta}_{\mu\alpha}} \right) \begin{pmatrix} x_\alpha - x_\mu \\ y_\alpha - y_\mu \end{pmatrix} \quad (39)$$

Previous expression is defined in terms of velocities by (40), where such term will satisfy the real velocity of equation (22),

$$\mathbf{v}_t^\alpha = \int_t \mathbf{f}_t^\alpha \, dt \quad (40)$$

The repulsive directional fields feedback by sensor observations are computed by the algorithm 7. In this manuscript we calculate artificial repulsive fields using only LIDAR data, because of the ranged nature of data, which makes easier and accurate the social fields map.

8.2. Attractive function

Similarly, equations controlling the robot course to a global goal destination yield motion behaviour as depicted by Fig. 7-left. A goal destination γ is not

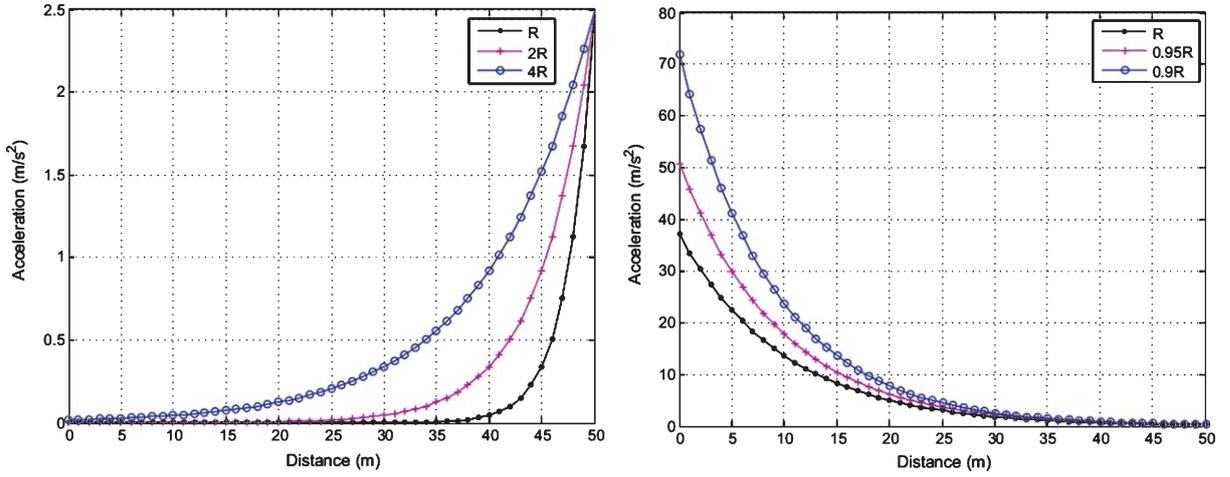


Fig. 7. Exponential acceleration behaviour in 1D. Left: Attractive potential; Right: Repulsive potential.

locally visible to any robot's sensor on-board. Its relevance arises from the fact that all γ s are defined a priori becoming intersection points along the global path. The general 1D artificial potential equation is defined by,

$$f_t^\gamma = -\nabla_{\mu\gamma} u_\gamma^o e^{-\|\bar{\delta}_{\mu\gamma}\|/R} \quad (41)$$

Where R represents the radius of a goal's territorial scope. The distance vector between the vehicle μ and the goal γ is defined by $\bar{\delta} = \mathbf{x}_t^\gamma - \mathbf{x}_t^\mu$. The constant factor u_γ^o scales the attractive accelerative forces amplitude. Let us notice that unlike the repulsive function, the attractive potential model do not have any denominator. Its motion effect of attraction is less stronger than the repulsive effects. Solving for its gradient operator next equation is defined (see Appendix B.1),

$$\mathbf{f}_t^\gamma = u_\gamma^o \frac{e^{-\|\bar{\delta}_{\mu\gamma}\|/R}}{R\|\bar{\delta}\|} \begin{pmatrix} x_\mu - x_\gamma \\ y_\mu - y_\gamma \end{pmatrix} \quad (42)$$

Finding a general solution for eq. (22), previous expression is rather defined in terms of velocities, than accelerations (43), as a term that partially satisfies (22),

$$\mathbf{v}_t^\gamma = \int_t \mathbf{f}_t^\gamma dt \quad (43)$$

The attractive accelerative forces are computed according to algorithm 8. Only one global destination is computed at time instead of the whole set of γ s, which is a different procedure from the repulsive algorithm 7. Only the main algorithm sets each γ , and until the

Table 8

Algorithm ATTRACTIVE ACCELERATIVE FORCES

- 1: **attractive-acceleration**($u_\gamma^o, R, \mathbf{x}_t, \mathbf{x}_t^\gamma$)
- 2: $\bar{\delta}_\gamma = \mathbf{x}_t^\gamma - \mathbf{x}_t$
- 3: $\mathbf{F}_t^\gamma = \frac{u_\gamma^o}{R\|\bar{\delta}\|} e^{\|\bar{\delta}_\gamma\|/R} (x_\alpha - x_\mu, y_\gamma - y_\mu)^T$
- 4: **return** \mathbf{F}_t^γ

actual γ has been reached by the robot, then the next global goal is considered.

By combining both directional social fields $\mathbf{F}_t^\alpha + \mathbf{F}_t^\gamma$, figure 8 illustrates an obstacle and a goal gradients interaction of robot accelerations.

9. Experimental results

We carried out on-line outdoor experiments in several modalities. First, although for a diversity of tasks we instrumented four different mobile robotic platforms with URG Hokuyo LIDARs, and similar vision systems on-board each robot, the interest of this manuscript was to demonstrate how the robot autonomously navigated on campus in the presence of multiple walking people, and a diversity of obstacles in the same environment (Fig. 10).

We established a set of known goal destinations γ^i , and the robot was able to reach them. While navigating, different groups of people moved toward several destinations and appeared and disappeared randomly from the robot field of view (Fig. 10). The robot built

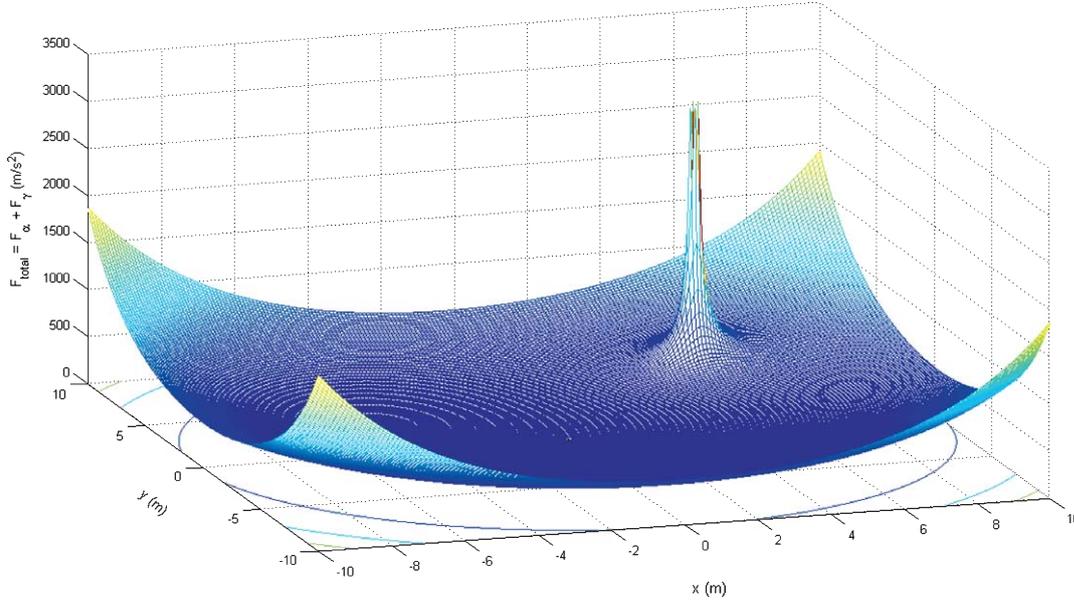


Fig. 8. 3D potential fields combining $F_{\gamma} + F_{\alpha}$, at $< -5, 0 > \gamma$ is located, and at coordinates (4, 5) an obstacle is located.

up a map of accelerative interactions between attractive and repulsive directional fields (Figs. 11 and 12). We found robot's stable motion behaviours using weighing factors during experiments, even with occurrence of too narrowed approaches with dynamic obstacles.

Second, in another experiment we instrumented a vehicle with on-board computers with Linux and vision sensors for on-line visual features extraction under natural light conditions along 5 km route. However, no control was implemented in the vehicle, so that, in these kind of experiments the purpose was to yield reliable weighting navigation parameters δ_i off-line. We obtained features of critical interest (see Fig. 3), directions to local goals (\mathbf{m}_i) leading to global destinations (γ^i) within navigable areas of urban roadways.

In further experimental simulations, Figure 9 shows how the reformulated social functions evolve to safely avoid two obstacles. There exist two obstacles at $\mathbf{x}_{\alpha} = (15, 50)^T$ and $\mathbf{x}_{\beta} = (-2, 20)^T$. The vehicle parameters for this experiment are $v^o = 0.5\text{m/s}$ from 0m to 100m along the vertical Y -axis. In Fig. 9(top-left), the vehicle navigates vertically from $\mathbf{x}_{t_0}^{\mu} = (0, 0)^T$ up to $\mathbf{x}_{t_n}^{\mu} = (0, 100)^T$, where location of γ is denoted by a triangle. Figure 9 (top-right) depicts the vehicle's attractive accelerations towards the goal. As the distance $\|\gamma - \mu\|$ is getting shorter gradually, the rate of motion behaviour is decreasing until the vehicle

reaches γ . In Fig. 9 (middle left-right), the acceleration components yielded by the presence of both obstacles are depicted versus the distance respect the actual vehicle positions. It is worth noting at Fig. 9 (down left and right) how the acceleration components evolve (dots x -component, and crosses y -component) along the y -axis. In particular at y -coordinate 20m and 50m to avoid obstacles β , and subsequently α , and then return to the desired trajectory.

The navigation algorithm yields autonomous control over the robot, listed in Table 9. Initial variables are set, and this algorithm is basically a continuous loop control program.

In Fig. 10, real experimental data are depicted using \mathbf{x}_t^{μ} to register LIDAR-based sensor data within a common Cartesian frame. The purpose of this work is not to accurately match laser scans, but to safely avoid collisions. The outdoor experiment shows feasibility of the algorithms (Tables 3 and 5). The mobile robot collected more than 50,000 laser scans within a coursed path nearly 100m long. According to Fig. 10, the continuous line depicts the vehicle path yielded autonomously by following a set of goal destinations established a priori by the human user at $\gamma_1 = (10, 1)^T$, $\gamma_2 = (32, 1)^T$, $\gamma_3 = (32, 6)^T$, $\gamma_4 = (5, 7)^T$, $\gamma_5 = (-3, 3)^T$. The potential fields repulsive influence was yielded by each of

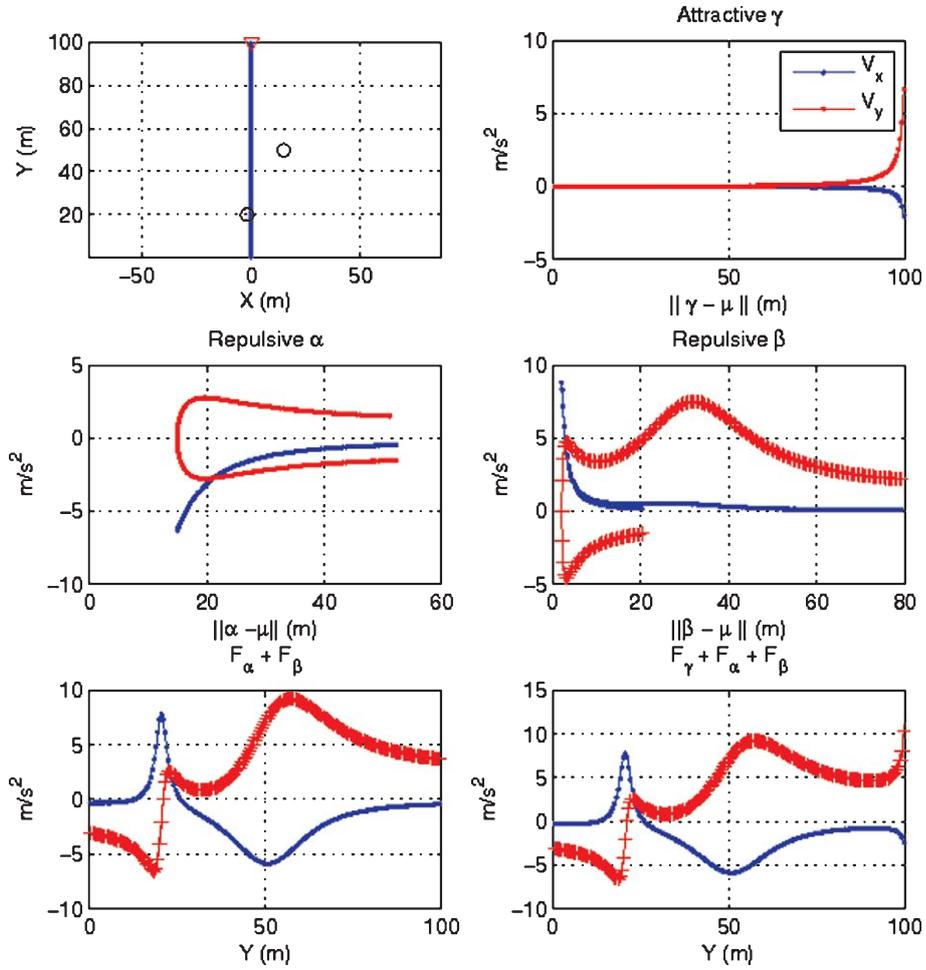


Fig. 9. The robot initial position at $\mathbf{x}_\mu = (0, 0)^T$ and its goal destination is at $\mathbf{x}_\gamma = (0, 100)^T$, an obstacle at $\mathbf{x}_\alpha = (15, 50)^T$, and the obstacle $\mathbf{x}_\beta = (-2, 20)^T$.

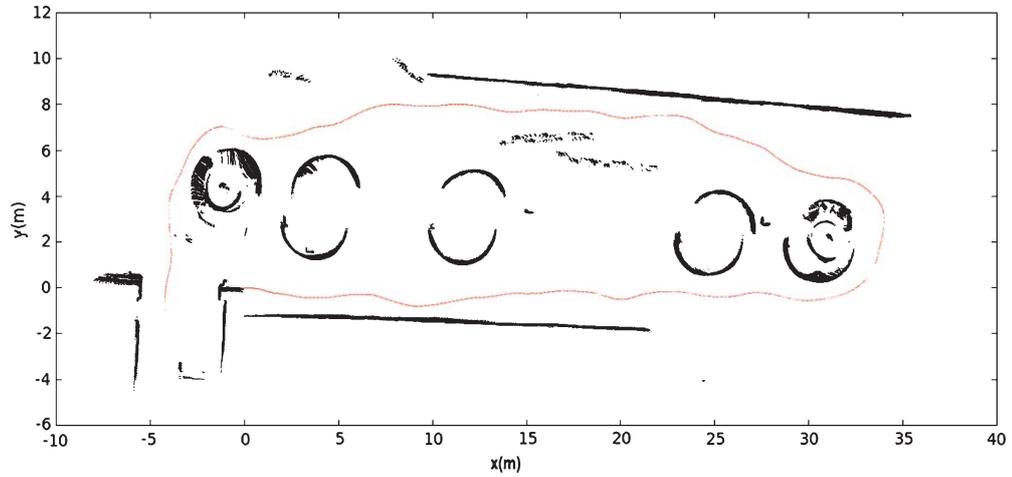


Fig. 10. Path generation from autonomous navigation (thinner line), and LIDAR based mapping (dotted lines) results from inferring \mathbf{x}_t and \mathbf{x}_{t+1} .

Table 9

Algorithm SOCIAL DIRECTIONAL FIELD NAVIGATION ALGORITHM

- 1: **Autonomous-motion**()
- 2: Initial sets $\gamma_1, \dots, \gamma_m, \vec{\xi}_0 = (0, 0, 0)^T, \dot{\phi}_t^1, \dots, \dot{\phi}_t^n, K, \tau, u_\alpha^o, u_\gamma^o, R$
- 3: **do-while** $\|\hat{v}\| \neq 0$
- 4: **for i=1: m**, will visit $\gamma_1, \dots, \gamma_m$
- 5: $\theta_t = \mathbf{robot-actual-orientation}(\theta_0, K)$
- 6: $\mathbf{x}_t = \mathbf{actual-position}(\dot{\phi}_t^1, \dots, \dot{\phi}_t^n, K)$
- 7: $(\mathbf{f}', \phi_t) = \mathbf{new-robot-bearing}(\theta_t)$
- 8: $\mathbf{x}_{t+1} = \mathbf{next-desired-position}(\mathbf{x}_t, \phi_t, \theta_t, w_t)$
Internal accelerative force
- 9: $\mathbf{F}_t^o = \mathbf{internal-motivation}(v^o, \tau)$
Repulsive accelerative forces
- 10: $\mathbf{F}_t^\alpha = \mathbf{repulsive-acceleration}(u_\alpha^o, R, \mathbf{x}_t)$
Attractive accelerative forces
- 11: $\mathbf{F}_t^\gamma = \mathbf{attractive-acceleration}(u_\gamma^o, R, \mathbf{x}_t, \mathbf{x}_{t+1})$
The real velocity is computed
- 12: $\mathbf{R}_1 = (\sin \theta_t, -\cos \theta_t)^T, \mathbf{R}_2 = (\cos \theta_t, \sin \theta_t)^T$
- 13: $\hat{v} = \frac{v^o}{\|\delta_t\|}(\mathbf{x}_t - \mathbf{x}_{t+1}) + \sum_\alpha \mathbf{v}_t^\alpha + \sum_\gamma \mathbf{v}_t^\gamma - v_t(\mathbf{R}_1 \quad \mathbf{R}_2)^T \cdot (-\theta_t, 1)^T$
Yielded velocity is controlled
- 14: $\mathbf{v}_t = (\hat{v}_{t-1} + \eta(\mathbf{v}_t^o - \hat{v}_t)) \cdot h(v^{max}, \|\hat{v}_t\|)$
- 15: **end-for**
- 16: **end do-while**

the 681 measurements compounding an observation (one single laser scan). The exerted accelerative force amplitude was adjusted accordingly.

Each sensor observation is comprised of a high density repulsive local map, but concurrently combined with a priori attractive directional fields. The resulting experimental directional fields map is depicted in Fig. 11 and Fig. 12 also corresponding to experiment of Fig. 10. The nearer the obstacle, the larger becomes the accelerative potential force magnitude exerted by the proposed model.

10. Conclusions

We treated a reformulation of the Social Force Model to provide a general navigation scheme to robotic tasks that could be of practical applications for generalized dynamic environments. We summarize that directional fields are physics-based models that are generally solvable equations in terms of velocities and accelerations. From experiments we found that high speeds are implicitly controlled in the model itself. In the proposed framework we found that directional

fields are relatively easy to adapt to any kinematic structure, proved to have path geometry quality with adjustments, and can be stated as reactive control laws.

A formulation of the SFM was developed to WMR, and demonstrated to achieve autonomous robot navigation in dynamic environments. We summarise our work by synthesising an extensive formulae in continuous-time on an effective navigation control that proportionally corrects the velocity error, and poses a maximal speed limit factor. It has been experimentally demonstrated to be suitable for navigation in dynamic environments. From experimental results the control model removed speed disturbances (commonly in artificial directional field functions), not allowing excessive changes of speed and kept the linear velocity very near to a reference velocity providing safety and real-time. The velocity control is a recursive model using the approximated real velocity \hat{v}_t .

The analytical model for \hat{v}_t was deduced from merging motion dynamic forces with accelerative inertial frames into a physics based model described by the 2nd Newton's law. We assumed a boundary equilibrium condition denoted by $\mathbf{a}_t^R = \mathbf{F}_t$. We deduced an equation for \hat{v}_t that involved external causes of motion,

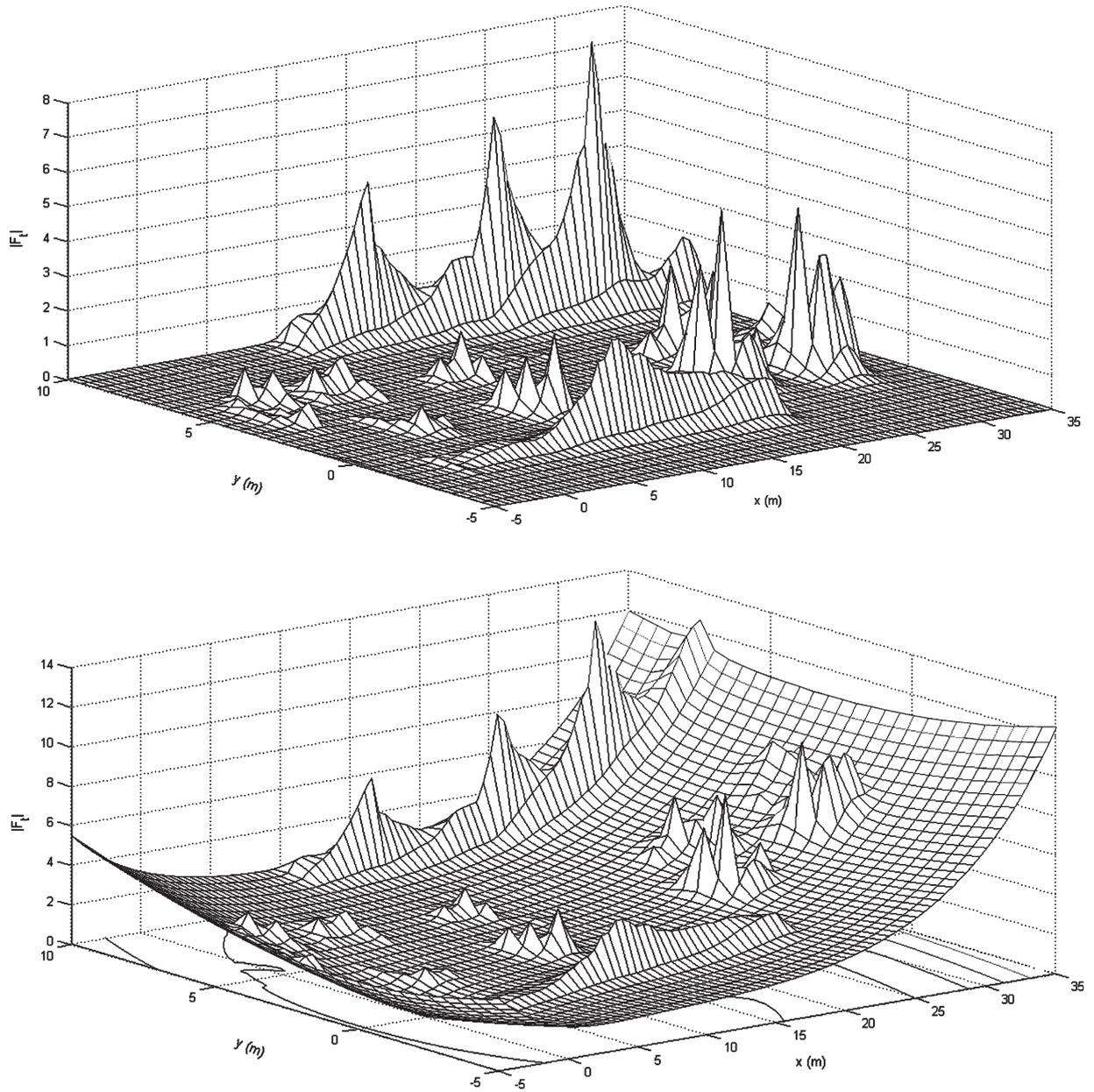


Fig. 11. Social potential fields dynamic interaction showed as 3D outdoor global mapping yielded through an autonomous navigation task. Above: robot's interaction in the repulsive map; Below: robot's interaction with numerous potential fields $F_\gamma + F_\alpha$.

as well as a generalised model for the yaw rate ω_r . So that, any robot with any kinematic description can be involved.

In such equation, we established two boundary statements, and we found that the model resulted suitable for robots navigating along roadways approximately

straight, where the approximated real velocity is usually not affected by speed perturbations. Nevertheless, the reaction speed of the velocity and steering control against fast obstacle detection situations was enhanced. It was obtained by automatic adjusting of numeric weighting factors directly fed by sensing

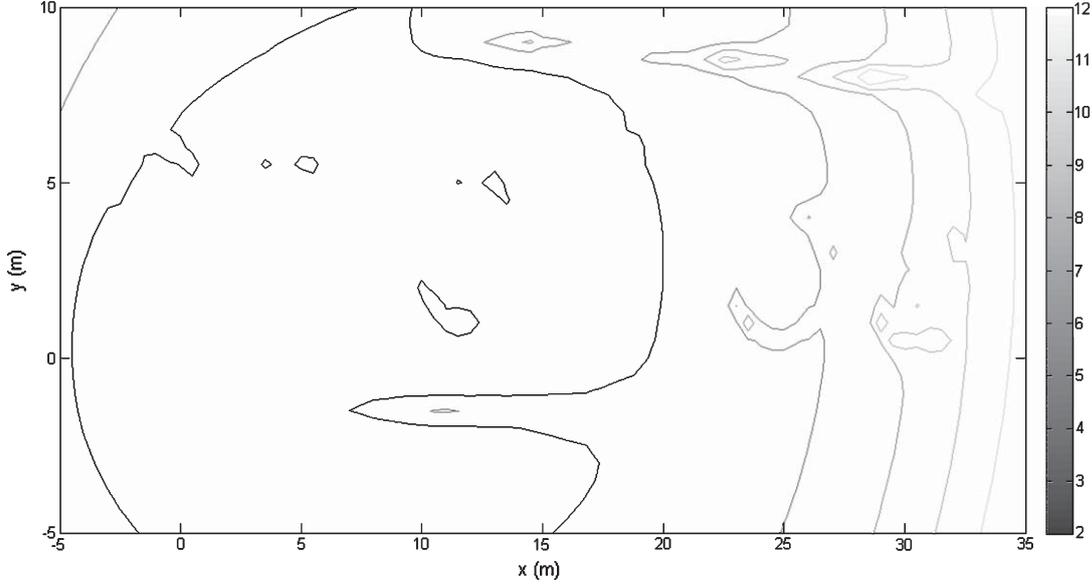


Fig. 12. 2D potential fields mapped by the mobile robot (top view), using the last γ .

features (feedback). We presented vision-based experimental results of relevant features extraction from urban roadways to realistically illustrate the parameters ϕ_t^i of the weighting function w_t and its mechanism λ_t to impact the bearing.

Furthermore, we presented a reformulation of the social force model, which is integrated as a set of terms to complete the general solution of the Newton's Law physics-based equation. This reformulation was adapted to mobile robots instrumented with both, vision sensors and URG Hokuyo LIDARs. The sensing parameters worked as variables for directional field equations featuring normal distributions equations as navigational functions. These, in practice resulted suitable to yield directional fields in real-time from sensor observations of high density measurement points. We provided outdoor experimental results by combining and mapping the directional fields along a trajectory of nearly 100m in the presence of dynamic obstacles, and from such results the reader can easily analyse the efficacy and feasibility of the exponential equations. Finally, we experimentally demonstrated in complex dynamic outdoor environments the proposed approach on autonomous navigation by combining the set of algorithms proposed in real vehicle-like robotic platforms as well.

Acknowledgments

This work was partially supported by the projects CONACyT No. 141701 as well as PROMEP No. DGPDI/SPROMEP/2-9/075. In addition, the authors would like to thank the members and students of the Laboratorio de Robótica at UACJ whom collaborated in the development of this research work.

Appendix A. Inertial frames formulation

The Euler rotation matrix \mathbf{R} and its inverse are critical in most mathematical definitions of this manuscript. Because of \mathbf{R} is a non-singular matrix according to $\mathbf{R}^{-1}\mathbf{R} = \mathbf{I}$ or $\mathbf{R}\mathbf{R}^{-1} = \mathbf{I}$, and since \mathbf{R} is an orthogonal matrix, hence $\mathbf{R}^{-1} = \mathbf{R}^T$. Thus, let us demonstrate it,

$$\mathbf{R}(\psi_t) = \begin{pmatrix} \cos \psi_t & -\sin \psi_t \\ \sin \psi_t & \cos \psi_t \end{pmatrix};$$

$$\mathbf{R}^{-1}(\psi_t) = \begin{pmatrix} \cos \psi_t & \sin \psi_t \\ -\sin \psi_t & \cos \psi_t \end{pmatrix} \quad (\text{A1})$$

if $\cos \psi_t \cos \psi_t - (-\sin \psi_t) \sin \psi_t \neq 0$, then \mathbf{R} is invertible, if only if $\det \mathbf{R} \neq 0$, where $\det \mathbf{R} = \cos \psi_t \cos \psi_t - (-\sin \psi_t \sin \psi_t)$.

$$\mathbf{R}^{-1} = \frac{1}{\det \mathbf{R}} \begin{pmatrix} \cos \psi_t & -(-\sin \psi_t) \\ -(\sin \psi_t) & \cos \psi_t \end{pmatrix} \quad (\text{A2})$$

and,

$$\mathbf{R}^{-1} = \frac{1}{\cos \psi_t \cos \psi_t - (-\sin \psi_t) \sin \psi_t} \times \begin{pmatrix} \cos \psi_t & -(-\sin \psi_t) \\ -(\sin \psi_t) & \cos \psi_t \end{pmatrix} \quad (\text{A3})$$

Thus, from previous definitions, \mathbf{R} as well as \mathbf{R}^{-1} will be used to describe motion in both inertial frames accordingly. In the present manuscript we present two ways to formulate an equation for \mathbf{a}_t^R . Firstly, we inversely transform the acceleration into the vehicle coordinate framework by,

$$\mathbf{a}_t^R = \mathbf{R}_Z^{-1}(\psi_t) \mathbf{a}_t^I \quad (\text{A4})$$

Thus, substituting terms in previous equations,

$$\mathbf{a}_t^R = \begin{pmatrix} \cos(\psi_t) & \sin(\psi_t) \\ -\sin(\psi_t) & \cos(\psi_t) \end{pmatrix} \left(v_t(\dot{\theta}_t + \dot{\psi}_t) \begin{pmatrix} -\sin(\theta_t + \psi_t) \\ \cos(\theta_t + \psi_t) \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos(\theta_t + \psi_t) \\ \sin(\theta_t + \psi_t) \end{pmatrix} \right) \quad (\text{A5})$$

Algebraically developing,

$$\mathbf{a}_t^R = v_t(\dot{\theta}_t + \dot{\psi}_t) \begin{pmatrix} -\cos(\psi_t) \sin(\psi_t + \theta_t) + \sin(\psi_t) \cos(\psi_t + \theta_t) \\ \sin(\psi_t) \sin(\psi_t + \theta_t) + \cos(\psi_t) \cos(\psi_t + \theta_t) \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos(\psi_t) \cos(\theta_t + \psi_t) + \sin(\psi_t) \sin(\theta_t + \psi_t) \\ -\sin(\psi_t) \cos(\psi_t + \theta_t) + \cos(\psi_t) \sin(\psi_t + \theta_t) \end{pmatrix} \quad (\text{A6})$$

By substituting trigonometric identities, our expression is simplified,

$$\mathbf{a}_t^R = v_t(\dot{\psi}_t + \dot{\theta}_t) \begin{pmatrix} \sin(\psi_t - (\psi_t + \theta_t)) \\ \cos(\psi_t - (\psi_t + \theta_t)) \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos(\psi_t - (-\psi_t + \theta_t)) \\ \sin(\psi_t + (-\psi_t + \theta_t)) \end{pmatrix} \quad (\text{A7})$$

Thus,

$$\mathbf{a}_t^R = v_t(\dot{\psi}_t + \dot{\theta}_t) \begin{pmatrix} \sin(-\theta_t) \\ \cos(-\theta_t) \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos(-\theta_t) \\ \sin(-\theta_t) \end{pmatrix} \quad (\text{A8})$$

Using the identities $\sin(-\theta_t) = -\sin(\theta_t)$ and $\cos(-\theta_t) = \cos(\theta_t)$. Thus, without lost of generality,

the resulting simplified mathematical expression is now written as in equation (A9), where in the first term, $\dot{\psi}_t$ is still existing, nevertheless it does not yield any impact because within the robot's motion frame ψ_t is always zero.

$$\mathbf{a}_t^R = v_t \dot{\theta} \begin{pmatrix} -\sin(\theta_t) \\ \cos(\theta_t) \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos(\theta_t) \\ \sin(\theta_t) \end{pmatrix} \quad (\text{A9})$$

Secondly, another way to find a functional form for \mathbf{a}_t^R is from equation $\mathbf{v}_t^R = \mathbf{R}^{-1} \mathbf{v}_t^I$ and its derivative is as it follows,

$$\mathbf{a}_t^R = \dot{\mathbf{v}}_t^R = \mathbf{R}^{-1} \dot{\mathbf{v}}_t^I + \dot{\mathbf{R}}^{-1} \mathbf{v}_t^I \quad (\text{A10})$$

By algebraically developing the second term in the right side of previous equation,

$$\mathbf{a}_t^R = \mathbf{R}^{-1} \mathbf{a}_t^I + \begin{pmatrix} -\sin \psi_t & \cos \psi_t \\ -\cos \psi_t & -\sin \psi_t \end{pmatrix} \dot{\psi}_t v_t \begin{pmatrix} \cos(\psi_t + \theta_t) \\ \sin(\psi_t + \theta_t) \end{pmatrix} \quad (\text{A11})$$

arranging terms and signs

$$\mathbf{a}_t^R = \mathbf{R}^{-1} \mathbf{a}_t^I - \dot{\psi}_t v_t \begin{pmatrix} \sin \psi_t & -\cos \psi_t \\ \cos \psi_t & \sin \psi_t \end{pmatrix} \begin{pmatrix} \cos(\theta_t + \psi_t) \\ \sin(\theta_t + \psi_t) \end{pmatrix} \quad (\text{A12})$$

thus,

$$\mathbf{a}_t^R = \mathbf{R}^{-1} \mathbf{a}_t^I - \dot{\psi}_t v_t \begin{pmatrix} \sin \psi_t \cos(\theta_t + \psi_t) - \cos \psi_t \sin(\theta_t + \psi_t) \\ \cos \psi_t \cos(\theta_t + \psi_t) + \sin \psi_t \sin(\theta_t + \psi_t) \end{pmatrix} \quad (\text{A13})$$

$$\mathbf{a}_t^R = \mathbf{R}^{-1} \mathbf{a}_t^I - \dot{\psi}_t v_t \begin{pmatrix} \sin(\psi_t - \theta_t - \psi_t) \\ \cos(\psi_t - \theta_t - \psi_t) \end{pmatrix} \quad (\text{A14})$$

and

$$\mathbf{a}_t^R = \mathbf{R}^{-1} \mathbf{a}_t^I - \dot{\psi}_t v_t \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} \quad (\text{A15})$$

$$f_t^\alpha = -\nabla_{\mu\alpha} u_\alpha^o R \frac{e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R}}{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|} \quad (\text{B1})$$

Deriving w.r.t. $\partial f^\alpha / \partial x$ and $\partial f^\alpha / \partial y$,

$$\begin{aligned} \left\langle \frac{\partial \mathbf{f}_t}{\partial x}, \frac{\partial \mathbf{f}_t}{\partial y} \right\rangle &= \left(\frac{R u_\alpha^o}{\|(\mathbf{x}_\alpha - \mathbf{x}_\mu)\|} e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R} \left(\frac{1}{2R} \right) \|\mathbf{x}_\alpha - \mathbf{x}_\mu\|^{-1} 2(x_\alpha - x_\mu) \right) + \\ &\quad \left(\frac{R u_\alpha^o}{\|(\mathbf{x}_\alpha - \mathbf{x}_\mu)\|} e^{(\mathbf{x}_\alpha - \mathbf{x}_\mu)^{1/2}/R} \left(\frac{1}{2R} \right) \|\mathbf{x}_\alpha - \mathbf{x}_\mu\|^{-1} 2(y_\alpha - y_\mu) \right) + \\ &\quad \left(e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R} u_\alpha^o R \left(\frac{-1}{2} \right) \|\mathbf{x}_\alpha - \mathbf{x}_\mu\|^{-3} 2(x_\alpha - x_\mu) \right) \\ &\quad \left(e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R} u_\alpha^o R \left(\frac{-1}{2} \right) \|\mathbf{x}_\alpha - \mathbf{x}_\mu\|^{-3} 2(y_\alpha - y_\mu) \right) \end{aligned} \quad (\text{B2})$$

Arranging and ordering the terms,

$$\left\langle \frac{\partial \mathbf{f}^\alpha}{\partial x}, \frac{\partial \mathbf{f}^\alpha}{\partial y} \right\rangle = \left(\frac{u_\alpha^o R (x - x_\mu) e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R}}{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|^3} - \frac{u_\alpha^o (x - x_\mu) e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R}}{(\mathbf{x}_\alpha - \mathbf{x}_\mu)} \right) \quad (\text{B3})$$

Thus, algebraically arranging and factorising common terms,

$$\left\langle \frac{\partial \mathbf{f}^\alpha}{\partial x}, \frac{\partial \mathbf{f}^\alpha}{\partial y} \right\rangle = \left(u_\alpha^o (x - x_\mu) e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R} \left(\frac{1}{(\mathbf{x}_\alpha - \mathbf{x}_\mu)} - \frac{R}{(\mathbf{x}_\alpha - \mathbf{x}_\mu)^{3/2}} \right) \right) \quad (\text{B4})$$

Now, developing the first term of right-side of equation,

$$\begin{aligned} \mathbf{a}_t^I &= v_t (\dot{\theta}_t) \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos \theta_t \\ \sin \theta_t \end{pmatrix} \\ &\quad - \dot{\psi}_t v_t \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} \end{aligned} \quad (\text{A16})$$

Finally,

$$\mathbf{a}_t^I = v_t \dot{\theta}_t \begin{pmatrix} -\sin \theta_t \\ \cos \theta_t \end{pmatrix} + \dot{v}_t \begin{pmatrix} \cos \theta_t \\ -\sin \theta_t \end{pmatrix} \quad (\text{A17})$$

Appendix B. Repulsive directional field formulation

The equation (B1) is a general function with the gradient operator, where u_α^o is a constant defining the acceleration amplitude, and R is a stationary value defining the asymptotic potential falling value.

in order to facilitate let us use notation \mathbf{f}_t^α instead, and let us define too $\|\delta_{\mu\alpha}\| = \|\mathbf{x}_\alpha - \mathbf{x}_\mu\|$,

$$\mathbf{f}_t^\alpha = \frac{u_\alpha^o e^{\|\mathbf{x}_\alpha - \mathbf{x}_\mu\|/R}}{\|\delta_{\mu\alpha}\|} \begin{pmatrix} x_\alpha - x_\mu \\ y_\alpha - y_\mu \end{pmatrix} \left(\frac{1}{\|\delta_{\mu\alpha}\|} - \frac{R}{(\mathbf{x}_\alpha - \mathbf{x}_\mu)} \right) \quad (\text{B5})$$

Finally, some terms of the derived equation may be substituted and simply expressed as,

$$\mathbf{f}_t^\alpha = u_\alpha^o \frac{e^{\|\delta_{\mu\alpha}\|/R}}{\|\delta_{\mu\alpha}\|} \left(\frac{1}{\|\delta_{\mu\alpha}\|} - \frac{R}{\delta_{\mu\alpha}} \right) \begin{pmatrix} x_\alpha - x_\mu \\ y_\alpha - y_\mu \end{pmatrix} \quad (\text{B6})$$

Previous expression if defined in terms of a velocity vector,

$$\mathbf{v}_t^\alpha = \int_t \mathbf{f}_t^\alpha dt \quad (\text{B7})$$

B.1. Attractive directional field formulation

The attractive potential function general equation is defined by,

$$f_t^\gamma = -\nabla_{\mu\gamma} u^o e^{-\|\bar{\delta}_{\mu\gamma}\|/R} \quad (\text{B8})$$

Thus, by deriving the function w.r.t. x and y , it yields,

$$\left\langle \frac{\partial f_t^\gamma}{\partial x}, \frac{\partial f_t^\gamma}{\partial y} \right\rangle = \begin{pmatrix} -u^o e^{-\|\mathbf{x}_\gamma - \mathbf{x}_\mu\|/R} \left(\frac{-1}{2}\right) \frac{\|\mathbf{x}_\gamma - \mathbf{x}_\mu\|}{R} 2(x_\gamma - x_\mu) \\ -u^o e^{-\|\mathbf{x}_\gamma - \mathbf{x}_\mu\|/R} \left(\frac{-1}{2}\right) \frac{\|\mathbf{x}_\gamma - \mathbf{x}_\mu\|}{R} 2(y_\gamma - y_\mu) \end{pmatrix} \quad (\text{B9})$$

simplifying the expression it now becomes

$$\mathbf{f}_t^\gamma = u^o e^{-\|\mathbf{x}_\gamma - \mathbf{x}_\mu\|/R} \begin{pmatrix} \frac{x_\gamma - x_\mu}{R\|\mathbf{x}_\gamma - \mathbf{x}_\mu\|} \\ \frac{y_\gamma - y_\mu}{R\|\mathbf{x}_\gamma - \mathbf{x}_\mu\|} \end{pmatrix} \quad (\text{B10})$$

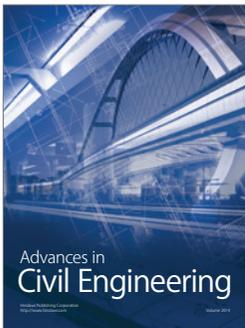
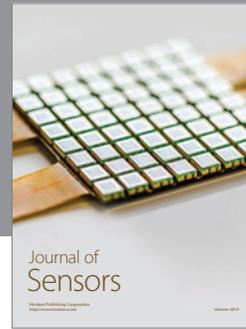
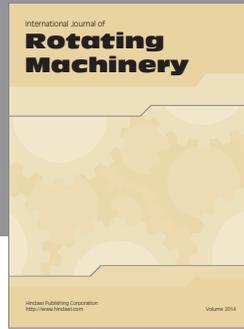
Algebraically arranging, the 2D potential function becomes as follows,

$$\mathbf{f}_t^\gamma = u^o \frac{e^{-\|\bar{\delta}_{\mu\gamma}\|/R}}{R\|\bar{\delta}\|} \begin{pmatrix} x_\mu - x_\gamma \\ y_\mu - y_\gamma \end{pmatrix} \quad (\text{B11})$$

References

- [1] W.H. Huang, B.R. Fajen, J.R. Fink and W.H. Warren, Visual navigation and obstacle avoidance using a steering potential function, *Robotics and Autonomous Systems* **54** (2005), pp. 288–299, ISSN 0921-8890.
- [2] M.F. Selekwa, D.D. Dunlap, D. Shi and E.G. Collins Jr., Robot navigation in very cluttered environments by preference-based fuzzy behaviors, *Robotics and Autonomous Systems* **56** (2007), pp. 231–246.
- [3] M.C. Mora and J. Tornero, Path planning and trajectory generation using multi-rate predictive artificial potential fields, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Acropolis Convention Center, Nice France, pp. 2990–2995, 2008, ISBN 978-1-4244-2058-2.
- [4] A.A. Masound, Managing the dynamics of a harmonic potential field-guided robot in a cluttered environment, *IEEE Transaction on Industrial Electronics* **56**(2), February 2010, pp. 488–496.
- [5] V.M. Goncalves, L.C.A. Pimenta, C.A. Maia, B.C.O. Dutra and G.A.A. Pereira, Vector fields for robot navigation along time-varying curves in n -dimensions, *IEEE Transactions on Robotics* **26**(4) August 2010, pp. 647–659, ISSN 1552-3098.
- [6] M. Werling, L. Groll and G. Bretthauer, Invariant trajectory tracking with a full-size autonomous road vehicle, *IEEE Transactions on Robotics* **26**(4), August 2010, pp. 758–765, ISSN 1552-3098.
- [7] L.A. Tychonievich, R.P. Burton and L.P. Tychonievich, Versatile Reactive Navigation, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 11–15, USA, pp. 2966–2972, 2009, ISBN 978-1-4244-3804-4.
- [8] A. Broggi, L. Bombini, S. Cattani, P. Cerri and R.I. Fedriga, Sensing requirements for a 13,000 km intercontinental Autonomous drive, *IEEE Intelligent Vehicles Symposium*, San Diego California, USA, June 21–24, pp. 500–505, 2010, ISBN 978-1-4244-7868-2.
- [9] G. Stanek, D. Langer and M. Muller-Bessler, JUNIOR 3: Test platform for advanced driver assistance systems, *IEEE Intelligent Vehicles Symposium*, University of California San Diego, June 21–24, pp. 143–149, 2010, ISBN 978-1-4244-7868-2.
- [10] A. Teja, d. Karthikeya and K. Madhava A mixed autonomy coordination methodology for multi-robotic traffic control, *IEEE International Conference on Robotics and Biomimetics*, Bangkok, Thailand, pp. 1273–1278, 2009, ISBN 978-1-4244-2679-9.
- [11] E. Plaku, L.E. Kavraki and M. Vardi, Motion planning with dynamics by synergistic combination of layers of planning, *IEEE Transaction on Robotics* **26**(3) (2010), pp. 469–482.
- [12] K. Moriwaki and K. Tanaka, Navigation control for electric vehicles using nonlinear state feedback H_∞ control, *Nonlinear Analysis* **71** (2009), pp. e2020–e2933.
- [13] K. Iagnemma, S. Shimoda and Z. Shiller, Near-optimal navigation of high speed mobile robots on uneven terrain, *IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, Acropolis Convention Center, Nice France, Sept. 22–26, 2008, pp. 4098–4013, ISBN 978-1-42442058-2/08.
- [14] M.P. Mann and Z. Shiller, “Dynamic stability of off-road vehicles: A geometric approach,” *Proceedings of IEEE Int. Conf. on Robotics and Automation* 2006, pp. 3705–3710.
- [15] S. Geand and Y. Cui, Dynamic motion planning for mobile robots using potential field method, *Autonomous Robots* **13** (2002).
- [16] D. Helbing and P. Molnar, “Social force model for pedestrians dynamics”, *Physical Review E* **51** (1995), p. 4282.
- [17] S. Shimoda, Y. Kuroda and K. Iagnemma, Potential Field Navigation of High Speed unmanned ground vehicles on uneven terrain, *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005.
- [18] Kareem Mohammad, Garibeh Mohammad and Feilat Eyad, Dynamic motion planning for autonomous mobile robot using fuzzy potential field, *Proceeding of the 6th International Symposium on Mechatronics and its applications*, Sharjah, UAE, March 24–26, 2009.
- [19] Dimitar Baronov and John Baillieul, Autonomous vehicle control for ascending/descending along a potential field with two applications, 2008 American Control Conference, Westin Seattle Hotel, Seattle, Washington, USA, June 11–13, 2008, p. 678–683.
- [20] Jing Ren, Kenneth A. McIsaac, Rajni V. Patel and Terry M. Peters, A Potential Field Model Using Generalized Sigmoid Functions, *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics* **37**(2), April 2007, pp. 447–484.
- [21] Jiangmin Chunyu, Zhihua Qu, Eytan Pollak and Mark Falash, A new reactive target-tracking control with obstacle avoidance in a dynamic environment, 2009 American Control

- Conference, Hyatt Regency Riverfront, St. Louis, MO, USA, June 10–12, 2009, pp. 3872–3877.
- [22] S.H.I. Enxiu, C.A.I. Tao, H.E. Changlin, S.H.I. Enxiu and G.U.O. Junjie, Study of the New Method for Improving Artificial Potential Field in Mobile Robot Obstacle Avoidance, *Proceedings of the IEEE International Conference on Automation and Logistics*, August 18–21, Jinan, China, 2007, p. 282–286.
- [23] Samer Charifa and Marwan Bikdash, Comparison of geometrical, kinematic, and dynamic performance of several potential field methods, *IEEE Southeastcon 2009*, March 5–8, 2009, pp. 18–23.
- [24] Richard Bishop, A survey of intelligent vehicle applications worldwide, *Proceedings of the IEEE intelligent Vehicles Symposium 2000*, Dearborn (MI), USA, October 3–5, 2000, pp. 25–30.
- [25] Yang Chen, Hangen He, and Xiangjing An, Motion planning of intelligent vehicles: A survey, *IEEE International Conference on Vehicular Electronics and Safety*, 2006. ICVES 2006. Dec. 13–15 p. 333–336.
- [26] Edgar Martínez-García, Alfredo Mar, and Rafael Torres-Córdoba, Dead-reckoning inverse and direct kinematic solution of a 4W independent driven rover, *IEEE ANDESCON 2010*, Bogotá Col., Sep 15th–17th, 2010, paper 57, ISBN: 978-1-4244-6741-9.
- [27] Edgar Martínez-García and Rafael Torres-Córdoba, 4WD Skid-steer trajectory control of a rover with spring-based suspension analysis: Direct and inverse kinematic parameters solution, *Intelligent Robotics and Applications*, Springer Lecture Notes in Computer Science, **6425**, Nov. 2010, ISBN 978-3-642-16586-2.
- [28] G. Campion, G. Baston and B.D. Andrea-Novel, Structural properties and classification of kinematic and dynamic models of wheeled mobile robots, *IEEE Transaction on Robotics and Automation* **12**(1) February 1996.
- [29] D. Fox, W. Burgard and S. Thrun, The dynamic window approach to collision avoidance, *IEEE Robotics & Automation Magazine* **4** (1997), pp. 23–33.
- [30] S.G. Loizou and K.J. Kyriakopoulos, Navigation of multiple kinematically constrained robots, *IEEE Transactions on Robotics* **24**(1), February 2008, pp. 221–231.
- [31] D. Wang and C.B. Low, Modeling and analysis of skidding and slipping wheeled mobile robots: Control design perspective, *IEEE Transactions on Robotics* **24**(3) June 2008, pp. 676–687.
- [32] K. Song and C.C. Chang, Reactive navigation in dynamic environment using multisensory predictor, *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics* **29**(6), December 1999, pp. 870–880.
- [33] K. Macek, K. Thoma, R. Glatzel and R. Siewart, Dynamics modeling and parameter identification for autonomous vehicle navigation, *In Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 3321–3326.
- [34] R. Gayle, W. Moss, M.C. Lin and D. Manocha, Multi-robot coordination using generalized potential fields, *IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 1–8.
- [35] R. Benenson, S. Petti, T. Fraichard and M. Parent, Integrating perception and planning for autonomous navigation of urban vehicles, *Proc. of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 9–15, 2006, Beijing, China, pp. 98–104.
- [36] W. Rong-ben, Z. Rong-hui, J. Li-sheng, G. Xiu-hong and G. Lie, Research on System design and control technology of vision-based cybercar, *Proc. 2006 IEEE/RSJ International Conference on Intelligent Robots and systems*, October 9–15, Beijing, China.
- [37] J.M. Armingol, A. de la Escalera, C. Hilario, J.M. Collado, J.P. Carrasco, M.J. Flores, J.M. Pastor and F.J. Rodriguez, IVVI: Intelligent vehicle based on visual information, *Robotics and Autonomous Systems*, *Science Direct* **55**, 904–916.
- [38] G.E. Brown and A.D. Jackson, *The Nucleon-Nucleon Interaction*, North-Holland Publishing, Amsterdam, 1976.
- [39] H. Choset, K.M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L.E. Kavraki and S. Thrun, *Principles of Robot Motion, Theory, Algorithms and Implementations*, MIT Press, 2005.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

