Neurocomputing 175 (2016) 1019–1032

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Synthesis of odor tracking algorithms with genetic programming

B. Lorena Villarreal [a], Gustavo Olague [b,*], J.L. Gordillo [a]

[a] Center for Robotics and Intelligent Systems, Tecnológico de Monterrey, Monterrey, N.L., Mexico
[b] EvoVisión Group, Applied Physics Division, CICESE, Ensenada, B.C., Mexico

ARTICLE INFO

Article history:
Received 29 November 2014
Received in revised form
28 September 2015
Accepted 28 September 2015
Available online 6 November 2015

Keywords:
Odor tracking algorithm
Bio-inspired nose
Chemical sensors
Sniffing robot

ABSTRACT

At the moment, smell sensors for odor source localization in mobile robotics represent a topic of interest for researchers around the world. In particular, we introduce in this paper the idea of developing biologically inspired sniffing robots in combination with bioinspired techniques such as evolutionary computing. The aim is to approach the problem of creating an artificial nose that can be incorporated into a real working system, while considering the environmental model and odor behavior, the perception system, and algorithm for tracking the odor plume. Current algorithms try to emulate animal behavior in an attempt to replicate their capability to follow odors. Nevertheless, odor perception systems are still in their infancy and far from their biological counterpart. This paper presents a proposal in which a real-working artificial nose is tested as a perception system within a mobile robot. Genetic programming is used as the learning technique platform to develop odor source localization algorithms. Experiments in simulation and with an actual working robot are presented and the results compared with two algorithms. The quality of results demonstrates that genetic programming is able to recreate chemotaxis behavior by considering mathematical models for odor propagation and perception system.

1. Introduction

Around the world, different environmental conditions and sometimes negligence produce hazard zones that endanger population. Such disaster areas affected by hurricanes, earthquakes, fires and nuclear catastrophes need to be restored as soon as possible without risking more lives. Rescue teams work intensively to diminish the effects, but occasionally they cannot reach the complete area due to toxic environments, the potential presence of explosive materials, collapses, or a simple circumstance like inadequate space. Rescue robots have received considerable attention in recent years, thus providing solutions for those scenarios where human rescue teams are unable to work. Today, a major trend in robotics research is to incorporate different sensor capabilities inspired from solutions of the natural world. The idea is that as soon as robots are able to see, hear and touch, technology will be able to emulate the human capabilities for searching, mapping, exploration and localization of different targets, such as lost or injured people, safe trajectories, gas leaks, explosives, to mention but a few. Nevertheless, these capabilities may not be robust enough for real disaster scenarios due, for example, to poor visibility generated by the presence of obstacles that increase the difficulty of reaching specific target zones. A promising research area that could tackle such limits is based on the inclusion of olfaction capabilities inspired by solutions found in the animal kingdom. Animals use the sense of smell for diverse tasks like inspection, recognition, mating and hunting, despite not always being its principal perception mechanism. For example, dogs are trained to accomplish search and rescue operations within disaster areas, airports and borders. In this way, they use primarily the sense of smell to localize drugs, explosives, chemicals, hazardous substances and even persons [1]. Moreover, perception of the environment composition (odor molecules and concentrations) through olfaction might develop into a set of strategies that could be implemented within a sniffing robot so that it could find the direction of odor trails and follow them until it reaches a saturated zone [2] to finally locate and detect toxic gas leaks, the origin of a fire, and so on.

Nevertheless, an optimal perception system like those encountered in simple organisms is not currently available since artificial sensors differ significantly from their biological counterparts. Moreover, the algorithms based on natural processes attempt to emulate the behavior of some animals, such as casting, and sweeping spiral [3] without reaching the same level of performance; i.e., the odor source is not located with high accuracy or it requires a lot of time to be reached. In our work, we believe that the difference between both systems may be due to the fact that the natural smell sense, unlike an artificial perception system, is evolved over many years until it acquires a way of locating the

* Corresponding author.
E-mail addresses: b.lorenavillarreal@gmail.com (B.L. Villarreal), gustavo.olague@gmail.com (G. Olague), JLGordillo@itesm.mx (J.L. Gordillo).

http://dx.doi.org/10.1016/j.neucom.2015.09.108
0925-2312/

odor. Thus, the idea of synthesizing artificial odor tracking strategies will be developed through artificial evolution; in particular, the application of genetic programming in combination with our artificial nose. Next, we review the main works devoted to odor source localization.

1.1. Related work

In the literature, there are many algorithms that aim to increase the efficiency of odor source localization from the viewpoint of sensor usefulness. The techniques are generally classified through the following functions: *chemotaxis*, *anemotaxis* and *fluxotaxis*, depending on the environment and capabilities of odor sensors. This research centers on the development of an artificial nose and in particular this paper deals with the development of chemotaxis algorithms. Traditionally, the chemical gradient derived from certain chemicals in the environment is the basis for orientation and movement of an agent – mobile robot – and it forms the base of chemotaxis algorithms. In general, the approach mimics the perception of odor using single or multiple sensors placed at different positions while calculating gradient responses over time [2]. The onboard or remote computer is responsible for analyzing the signals and their variations with respect to time and space. The first robot charged with odor source localization was presented by Rozas et al. in 1991 [4]. The design consists of following odor gradients by taking two or more measurements by one sensor from different positions at different times. In this algorithm, the robot had to measure odor concentration at four different positions. Additionally, if the new measurement was smaller than the previous one, the robot returns to the last position. Through this routine, a robot takes a lot of time to get closer to the source. Later, in the mid 1990s the first sensor design used to obtain a measurement from two positions at the same time was presented by Ishida et al. [5,6]. This odor compass requires rotating the probe 360°, a process that took 20 s to obtain a direction and about one minute to recover from its initial state. Results showed that the system points to the trail direction but not always to the source position. Afterwards, a new stereo architecture implemented on a Koala mobile robot used measurements at different times and positions to obtain a gradient [7]. Nevertheless, the robot needed to be very close to the source for detection. Then, a mechanical implementation was presented [8] in which motor speed on each tire was proportional to their averaged concentration issued from an array of sensors. Hence, the robot was forced to turn when it reached some virtual walls, thus staying near to the odor source. Again, the robot needed to pass close to the source to detect the odor. Recently, work was presented using an unmanned aerial vehicle and a pseudo-gradient algorithm [9]. The Airrobot AR100-B micro-drone was used with an autonomous routine based on wind information and chemical gradient, sensed around the environment. Due to turbulence generated by its propellers, the drone should stay in the same place for a long time between measurements.

The principal drawback of previous algorithms were related to the sensors processing time since it took a lot of time to be ready for a second measurement (more than one minute). Some strategies even required more time since they need to cover the whole area several times (more than 20 min). Also, the odor source was not always reached due to multiple local maxima placed near the odor source. This happens because vapors are volatile and tend to homogenize the whole area, but in the case of constant gas leaks, maximum concentration is always at the exit of odor source. Sometimes the difference between both nostrils is provided by an airflow that helps us to circulate the odor around the system, creating a trail at the robot's rear part, and, as a consequence, sensors are constantly saturated by the same odor. Moreover, some systems, besides using chemotaxis, also collect wind information (anemotaxis), which is the most popular technique for outdoor environments. Thus, while considering that perception of wind speed is imperceptible for humans and common anemometers beyond 0.1 m/s [10], and that anemotaxis techniques are not appropriate for indoor environments that have small air currents. In this work, we provide evidence that considering only the chemical gradient is enough to reach an odor source indoors. With this as a foundation, all techniques could be improved based on chemical gradient, wind speed, mass flux, alone or in combination. This includes the use of robot teams or swarms where each one of them can have different behaviors or cooperate to reach bigger zones that will decrease the tracking time.

Robots could be designed to learn to use their odor sensors while considering the limitations at the moment of perceiving the environment. Comparatively, some research about perception systems that learn to discriminate odors was presented in 1999 [11]. In this case, an artificial neural network simulates the olfactory sensory neurons; thus, enabling discrimination of organic vapors. A similar approach in 2001 imitates the olfactory bulb including rank-order filtering over artificial neural networks [12]. Later, in 2004 the mathematical model for all biological olfactive layers using artificial neural networks was achieved [13]. On the other hand, Continuous Time Recurrent Neural Networks (CTRNNs) presented on 2013 [14], evolved odor source localization with a simulated robot equipped with a single chemical sensor and wind direction sensor. Schaffernicht and coworkers in 2014 [15] modeled and mapped the distribution of gas events, as well as detection and non-detection of a target gas using Bayesian Spatial Event Distribution. Recent work by Zhang and colleagues discusses localizing several odor sources [16], implementing a method based on niching Particle Swarm Optimization (PSO).

Moreover, to accomplish odor source localization, it is fundamental to consider the characteristics of the sensor with respect to desaturation time, concentration difference between sources, and reaction time. In our work, these features and their mathematical models are the basis of a learning perception system, which is used to derive a new technique that offers better results. The goal of this research is to obtain an algorithm that validates the use of chemical sensors to track and locate odor sources, especially when other sensors are limited or unavailable. Thus, the method relies on the perception of chemical odors while avoiding other sensors such as anemometers, cameras, sonars, and so on. In particular, the aim is to design algorithms, based only on olfaction (chemotaxis), that are able to follow straighter paths, thus reaching the source faster in comparison with current techniques. The algorithm for indoors obtained by genetic programming considers the mathematical models for odor propagation, and the perception system implemented into an unmanned ground vehicle (UGV) that looks for an optimal way of achieving the task in environments with imperceptible air currents for humans, while considering the limitations and advantages of the implementation.

Our paper is organized as follows: the problem statement is presented in Section 1.2 followed by a summary of our contributions in Section 1.3. In Section 2 the perception systems for chemical gradient are detailed. In particular, we describe the odor propagation model in 2.1, the chemotaxis technique in Section 2.2, and the artificial nose in Section 2.3. Section 3 details the chemotaxis algorithms. In Section 4 the description of our proposed methodology is comprehensively addressed. The simulation framework in Section 4.1, and the genetic programming method in Section 4.2. Then, in Section 5 the experimental setup and results are discussed. Finally, Section 6 provides the conclusions together with possible future work.

1.2. Problem statement

In the scenarios mentioned above, and others, a difficult problem is measuring and locating the odor source position caused by airflow patterns related to the phenomenon of advection and diffusion [17]. A solution of such real-world problems should consider a quick detection of the leaking gas origin since the faster the identification of an odor's trail direction the better. While anemotaxis is commonly used outdoors, since the vector component of advection is much bigger than that of diffusion; when indoors we present a solution based only on chemotaxis since airflow is mostly imperceptible. Hence, perception of the environment using an optimal signal processing model becomes crucial to understand and develop good control behaviors in the form of odor source localization algorithms that are simple to implement on a mobile platform [18–20]. In fact, the algorithm responsible to achieve the planning task depends not only on the environmental conditions, but also relies on the perception system and the physical characteristics of the robot; for example, desaturation time, measurement delays, sensors' position, robot's mechanics, and so on. Thus, the analysis should focus on the definition of the environment with the aim of characterizing the odors dynamic behavior, since different environments could require different kinds of sensors and architectures to adequately perceive the signals and provide useful information for future analysis.

In this research, Genetic Programming (GP) is used as the optimization tool for training an UGV in such a way that it could be seen as an odor source localization platform, while taking into account the capabilities and limitations of the real-working robot, whose physical implementation of the perception system is capable of obtaining the source direction from where an odor is coming. A simulation environment was created to test different algorithms; it considers the mathematical models for both the spread of odor and the properties of the perception system. In this way, the modeling and simulation of its physical properties are used to obtain an algorithm that is evolved with the GP-approach to produce an optimal localization strategy. Thus, in order to validate the proposal against different techniques of chemotaxis at the state-of-the-art, the algorithms are compared in simulation and in practice with a real-working robot by deploying the artificial nose in a specially conceived environment in such a way to measure the chemicals in the room and testing all studied algorithms under the same circumstances.

1.3. Research contributions

The work presented in this paper extends the results of the perception system previously proposed [21–23] in such a way as to diminish the common drawbacks outlined before. It implements a bio-inspired nose system with the capability of determining the direction of a source placed at a given distance by using a pair of nostrils divided by a septum. In the inhalation process, the artificial nose concentrates the odor molecules near the sensorial system, and at exhalation the nose desaturates the sensors. The proposed design complements the sensor model detailed before [7,24] by including the cyclic behavior of an artificial nose placed into a chamber. Thus, genetic programming for odor source localization is a novel technique that produces a precise algorithm while considering the capabilities and mathematical models of a given perception system and mobile robot. The secondary contributions of this research are listed below:

- The simulation was developed on Netlogo which is general enough to run any kind of experiments. It proved to be useful for visualizing and characterizing the mathematical models of the environment and the physical implementation.
- Odor source localization is achieved using only olfaction by chemotaxis for an unstructured indoor environment without using other sensors as anemometers and cameras.
- The algorithm obtained by GP is better in terms of task achievement in comparison with common techniques for odor source localization that are based on chemotaxis.
- This algorithm was implemented on a real working system at an indoor environment with small airflows and the odor source was reached faster than common techniques due to the tracking of straighter paths.

2. Perception of chemical gradient

Different techniques for odor source localization have been applied to approach several tasks while considering the environmental conditions and the kind of sensors that are used to design the perception system. Today, there are several proposals based on the design of sensor configurations and their deployment within a mobile robot. In particular, we will refer to them as openly exposed or cased. The first proposal is illustrated by sensors that are directly exposed to the environment without any isolation. In other words, bare sensors do not have a protective cover. In the work of Monroy et al. a gas sensor technology is deployed within a mobile robot following an open sampling system configuration [24]. The second proposal has two variants. On the one hand, the sensors are placed within a case while being continually exposed to the environment, in a way that the airflow is induced directly to the sensors placed within the isolated chamber. In Martinez et al. [25], a pair of chambers are used to perform two basic tasks of an olfactory robot to successfully track the odor source: navigate in a turbulent odor plume and recognize an aroma regardless of its concentration. Similar approaches [7,9,26] produce and direct airflow into an inlet through the sensors. When a constant odor source is present, as a gas leak, the sensors are being continually exposed to the scent not having time to recover from its original state. On the other hand, the second variant refers to the situation where the sensors are also placed within a case while being cyclically exposed to the environment [27,28]. In this new situation the idea is to use a chamber with the capability of isolating the sensors from the environment for a certain time so the sensors could have the opportunity to desaturate the sensors and prepare them for reading a new measurement.

2.1. Odor plume model based on chemotaxis

The propagation of odor molecules within the environment occurs in two different ways. Firstly, when no airflows are presented the chemical propagation is achieved by diffusion in a radial manner. Secondly, when airflows are presented the propagation is performed by advection in a laminar way. In [17], diffusion is described as the process by which matter is transported from one part of a system to another due to molecular motions. Each molecule presents a random motion and the set of random movements of all molecules results in the mix of the solute. The microscopic behavior, however, is not what determines the odor trail. Instead of this situation, the random walk of molecules takes place from a high concentration to a low concentration region, as a function of the concentration gradient, while is trying to homogenize the environment. The general form of the diffusion equation for a three-dimensional system is represented by

$$\frac{\partial C}{\partial t} = D\left(\frac{\partial^2 C}{\partial x^2} + \frac{\partial^2 C}{\partial y^2} + \frac{\partial^2 C}{\partial z^2}\right),$$

and is considered as a first-order approximation.

Odor sources propagate the molecules in different ways comprised in a pair of examples: a pulse which represents a drop of a fluid or a container, where the odor molecules tend to homogenize the environment by diffusion and the local maxima decreases with time; and a constant leak which is continuous and at the position of the outflow the concentration is increasing all the time. In this case, depending on the outflow pressure and the airflow speed generated by the leak, the propagation can be considered as primarily based on advection or diffusion.

In this research, a constant diffusion source will be used as the environment condition for both the simulation and the real-environment experiments, as the intention is not to find the presence of some chemical in a room but rather the gas leak source.

2.2. Chemotaxis techniques for odor source localization

As said before, the algorithms for odor source localization are primarily classified by the terms of chemotaxis, anemotaxis and fluxotaxis. The term "Chemotaxis" was taken from biology. It is used to describe the movement of an organism in response to chemical stimulus, for example, by chemotaxis a bacterium is able to find food looking for the highest concentration of molecules, or in the opposite case run from poisons. So, it is used when the orientation and movement of an agent (mobile robot) is based only on chemical gradient and it does not involve wind measurements or any other factors. The perception of the odor is done by measuring a single spot at different times or by measuring different positions at the same time. Multiple or single sensors can be used to reach different positions. On the other hand, anemotaxis, instead of following the gradient, considers the direction or current of a fluid [2] and the agent moves through it. While fluxotaxis [29] uses information of mass flux besides chemical concentration, fluid velocity and direction.

2.3. Perceptual system designed for mimicking chemotaxis

The goal of this research is to create an algorithm based on an artificial nose designed for chemotaxis. Such a perception system consists of a septum and a pair of nostrils, each one using an array of three sensors.

In most instances, odor molecules are persistent and saturate sensors. In that case, detecting differential changes of concentration to navigate would be even more difficult without a proper sensors' desaturation. The slow recovery occurs because chemical reactions produce inherent inertia on the sensor which is an intrinsic behavior of chemical sensors that cannot be prevented. Furthermore, the recovery is slow because desaturation in the environment is not instantaneous. Therefore, the designed nostril manages ventilation stages based on the olfactive cycle (inhalation/exhalation), for realistic operations [30,27]. Moreover, before the air exits the nostril, it passes through an organic filter serving as a cleaning device, decreasing expulsed concentration and saturation of the environment at the system's rear part. In mobile robotics applications, air cleaning is very important, because the mechanism should not create a trail of the robot's movement; otherwise odor accumulation at the rear could be detected as a leak source.

The proposed system estimates direction of a diffusion source, within a distance, based on the difference between two arrays of sensors. Nevertheless, both nostril responses, during one ventilation cycle, can have a small dissimilar behavior under the same conditions due to mechanical and electrical conditions. For example, a small discrepancy on the airflows generated by the ventilators can produce a proportional variation in the response, however this is calibrated by proportional gain adjustment for each sensor (kp_{left} and kp_{right}). A typical response at different angles is shown in Fig. 1. Note that the difference between both nostrils related to the absolute magnitude provides the direction of the diffusion source.

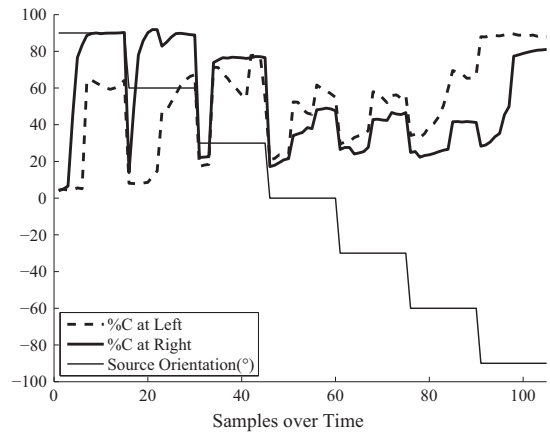

Fig. 1. Results of the response using the biologically inspired system. Note that the nostrils react to different levels of magnitude and at different times in relation to the direction (left/right) of the odor source. The uncertainty at 0 is solved through the application of a threshold. In the example, the robot follows a straight line.

The proposed model emulates a complete ventilation process for which different variables can be adjusted: the saturation level (R_{max}), the time constant of rise (τ_r), the retaining of air (τ_a), time to decay (τ_d), time before sensor started to respond (t_s), time of the rising period (Δt_r), and time of the sampling period (Δt_a).

In real-world applications, the smell process is cyclic, which means that the actual reading of the sensors depends on the last measurement. The model that represents this design using a continuous odorous environment is divided in three stages: (1) for inhalation

$$r_i(t) = r_e(t-1) + (R_{max} - r_e(t-1))\left(1 - \exp\left(-\frac{(t-t_s)}{\tau_r}\right)\right), \tag{1}$$

(2) for sampling

$$r_s(t) = r_i(t) + \tau_a(t - t_s - \Delta t_r)\left(1 - \frac{t - t_s - \Delta t_r}{2}\right), \tag{2}$$

and (3) for exhalation

$$r_e(t) = \frac{r_e(t-1) - r_s(t)\exp\left(-\frac{(t-t_s-\Delta t_r-\Delta t_a)}{\tau_d}\right)}{1 - \exp\left(-\frac{(t-t_s-\Delta t_r-\Delta t_a)}{\tau_d}\right)}. \tag{3}$$

where $r_i(t)$, $r_s(t)$ and $r_e(t)$ are the respective concentration values during inhalation, sampling and exhalation at the actual ventilation cycle. Consequently, $r_e(t-1)$ is the concentration value of the previous cycle, so after each cycle the initial reference is updated with the following formula

$$r_e(t-1) = r_e(t). \tag{4}$$

Therefore, according to the proposed design, it is possible to predict the behavior, based on the modeling and simulation of the physical properties. This will be used to derive an algorithm that works specifically by taking advantage of such features.

3. Basic algorithms for mimicking chemotaxis

When we talk about odor source localization techniques three stages are identified. The first one is based on the exploration of the area to find an odor trail while the robot is actually doing other

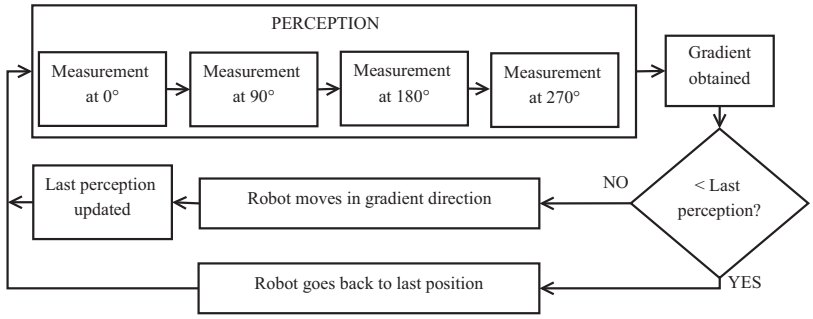

Fig. 2. Odor source localization algorithm proposed by Rozas to replicate chemotaxis [4].

tasks. The second one relies on the act of following the trail until the robot strikes a specific target while considering time limit, physical frontier and visual information. The last one is the validation of the source that could be corroborated through the combination of other sensor readings or with some concentration thresholds. The algorithms that were selected for comparison in our research are those that use local information to cover a certain region and apply exclusively odor information, hence imitating the capability of olfaction.

3.1. Rozas algorithm

The first technique to be implemented on a robot with the aim of emulating odor source localization was the algorithm of Rozas et al. [4]. It was designed to follow odor gradients, mimicking chemotaxis, by taking spatial measurements at different times. The algorithm depicted in Fig. 2 proposes a circular movement of the robot during the first part of the cycle. The odor is measured at four positions so the robot turns in response to the maximum concentration. Finally, the robot turns back when the new measurement is smaller than the previous one or moves forward otherwise.

3.2. Basic algorithm

In this work, the "basic algorithm" was created to consider characteristics of an artificial nose system using the difference between the pair of nostrils, to measure at a single time the chemicals from the environment by detecting gradient direction and stepping out towards the odor source location. The routine consists of two operational time cycles: aspiration process and robot displacement. In the aspiration process, at each time that the system inhales, the robot acquires odor concentration data through its sensors and saves it into the memory (*Mem*) of the acquisition system. Thus, the robot's algorithm starts the process by waiting a certain time (t_m) between each measurement. Then, for each inhalation step the average of the accumulated data is updated following the time sequence (t_i). As a result, the algorithm calculates a new direction by aligning the robot towards the odor source. The turning direction is limited by maximum angle (θ_{max}). Finally, the robot moves s steps forward. A threshold (thr) is considered to deal with the uncertainty between nostrils for the case where the source is located near to the front part of the robot. In that last case, the robot moves s/k steps, with k being an experimental constant. The algorithm is shown in Fig. 3.

In summary, the development of algorithms for odor source localization must take into account the characteristics of both environment and sensor model so as to achieve the improved desired behavior. In this paper, we propose an automated system based on genetic programming to create odor source localization

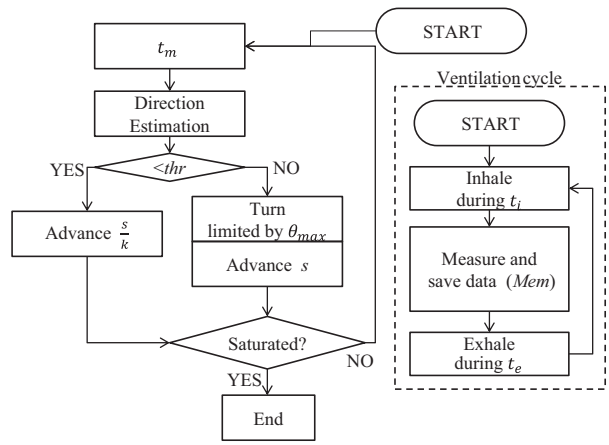

Fig. 3. Algorithm based on the reactive gradient ascend paradigm where the robot turns according to the input measurements while considering a ventilation cycle.

strategies that are able to cope with the limitations of the physical implementation – the mathematical model – and the operational cost and constraints.

4. GP-trained odor source localization technique

It is well known that evolutionary algorithms are powerful tools to tackle difficult optimization problems. Such techniques are inspired from the Darwinian theory of natural selection, which is mainly based on concepts such as inheritance, mutation, crossover, selection and so on. In our work, genetic programming uses a population of computer algorithms for odor source localization that is evolved along a number of generations. It means that, generation by generation, GP stochastically transforms the population of algorithms – candidates solutions – according to the evaluation associated with each solution of a general fitness function – for a given environment and sensor configuration – until an optimal algorithm is discovered. In this way, the candidate perceptual systems have different capabilities and limitations depending on their structure, architecture and response. The main reason to apply genetic programming as the method that synthesizes odor source localization techniques is the idea that a general solution may not apply to all perception systems because they may have different desaturation times, physical and mechanical differences, and measurement delays, among other characteristics previously mentioned which are crucial for this purpose. The robot behavior can be improved if it learns how to use its specific odor sense capabilities with a mathematical model

according to a target or objective. As a first proposal in this research, GP is used to produce an odor tracking algorithm that integrates in simulation the artificial nose with the capability of directionality and mathematical modeling. The main objective is to reach the odor source faster than the two chemotaxis algorithms described earlier by following a linear path from the current robot position to the odor source location.

One difficulty for estimating odor source localization, specifically during the tracking stage, is that air currents do not follow a simple behavior in the case of real-world scenarios; therefore, is not easy to derive a good simulation model that could be useful during the experiments. In this way, to successfully test the robot within a real working scenario, the performance of the algorithm should be tested on simulation to design new algorithms under an optimization strategy. With this purpose in mind, a simulation in NetLogo [31] based on the characteristics, properties of our robot, and the setup was designed, including both the environmental model and the bio-inspired sensor model. This simulation platform allows us to run any number of experiments, so the best algorithms can be tested in the real-world. In fact, the simulation is coupled with genetic programming to successfully test and validate the populations generated by our optimization strategy. This platform was also selected because the experiments can be visualized (Fig. 4) since it has an interface that communicates with MATLAB for data exchanging between the two applications. In our implementation, Matlab is responsible for the creation of the new generations based on fitness by means of inheritance, mutation, selection and crossover, while NetLogo runs the experiments and assigns the fitness to each candidate. In particular, for this application the terminals are the actions of the robot [32,33], which are programmed in NetLogo. At each generation, the simulation system reads the set of candidate programs and runs the experiments for each one of them. Then, it writes the fitness information in a second file that MatLab uses later to evolve a whole new generation. The results of such experiments for each generation are sent to MATLAB to be stored, processed and analyzed.

In this way, in order to develop a successful GP implementation, it is necessary to understand the robot initial conditions, environment and scenarios that will be tested in simulation and practice. This section further develops the environment and perceptual system models, the architecture, and the overall conditions that are used by genetic programming within the simulation.

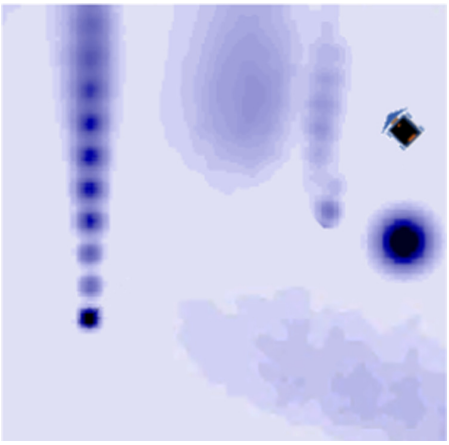

Fig. 4. An example of the different environments that can be simulated with the simulation platform.

4.1. Design of the simulation system

The aim is to test and compare different algorithms that require the use of a controlled environment useful at reporting the advantages and disadvantages of each proposal through an exhaustive statistical analysis. While considering a real environment, the odor saturates the room at each experiment, thus making it a real test for the same initial conditions, which are intractable and unreachable for short periods of time. Hence, the application of simulation becomes of major relevance and it depends on the design of suitable models that improve the overall results computed with the mathematical models. Moreover, simulation can be used hundreds of times to run experiments for testing later, since the best results obtained with genetic programming can be directly tested on the real robot.

NetLogo is a mesh-based multi-agent system that provides a programmable modeling environment for simulating natural and social phenomena. Modelers can give instructions to hundreds or thousands of agents since all are operating independently following a distributed model. The simulation consists of the odor propagation model that emulates the behavior of the odor within the environment. The sensor model includes the mathematical descriptors that characterize the inhalation, sampling and exhalation of the algorithm attempting to solve the odor source localization problem. In the case of odor propagation we provide the diffusion rate and wind speed. Also, for the sensor model, the specific characteristics of the artificial nose design are taken into account.

The simulated environment consists of a fixed odor source that is constantly diffusing the chemicals through the air, as well as a mobile robot capable of measuring the concentration difference using its two emulated nostrils, which are positioned at 45° and −45° respectively. Also, the odor source is placed for simplicity within one unit distance from the center of the robot. The proposed control strategy is depicted in Fig. 5. It contains both the environmental model and the sensor model. The algorithms are represented as syntax trees called here: “task trees”, and are composed of nodes and leaves. For this application the nodes are considered as functions and the leaves as the actions of the robot.

The output at each execution time produces a trace that is a relative direction pointed from the robot towards the source. The algorithm tries to equate this variable to zero. In this way, the robot will move forward to be aligned against the source while attempting to reach it. The final goal is to arrive at the odor source location. In this work, the same control representation will be used in the experiments with the real robot. The GP algorithm will be introduced in the next section.

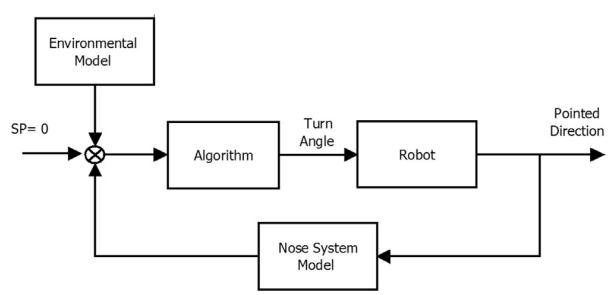

Fig. 5. Control implemented in NetLogo for odor source localization.

Table 1
Terminal set for genetic programming.

Name	Description
Move	Robot moves forward one step
Jump	Robot moves forward two steps
MeasureAverage	Robot averages the nostrils' differences using all accumulate measurements during the active olfactive cycle
MeasureIntake	Robot takes a single measurement from last nostrils' difference at sampling stage
MeasureTurn	Robot takes a measurement at sampling, then it turns following the nostril's difference
Turnmeasured	Robot turns in the direction calculated by MeasureAverage and MeasureIntake
Turn90	Robot turns 90° to left or right depending on last measurement
Turn45	Robot turns 45° to left or right depending on last measurement
Turnrandom	Robot turns random in a range of −90° to 90° without considering last measurement
Turnrandom45	Robot turns 45° in a random direction
HoldOn	Robot waits 1 time step
Goback	Robot turns 180° and moves forward one step

4.2. Integrating all models with the genetic programming methodology

The application of genetic programming as the methodology to synthesize odor sensing strategies starts with the definition of the terminal set shown in Table 1. To fully characterize the perceptions and actions of the robot, the function set is completed with the functions: PROGN2, PROGN3 and IF(a,b). In this way, PROGN* refers to the simplest nodes used for connecting parts of programs together while following the order from left to right. PROGN2 returns two subtrees in sequence while PROGN3 returns three subtrees. On the other hand, function IF is used primarily to determine the odor direction according to a concentration difference threshold. Thus, the function IF returns a when the threshold is reached or b otherwise. The combination of terminals and functions creates programs with many different behaviors (algorithms). In this work, we propose to validate the GP designs through a fitness function that evaluates each candidate solution using five parameters:

- *Distance reached* (ΔD): At the end of the experiment, this parameter tells us how close or far the robot travels from its initial position to the final location; the range varies from −0.75 to 1 with 1 being the best value.
- *Time used* (t_u): This parameter is the time that the robot takes to reach the source (t_{exp}) and is normalized by t_{max}, which is the total time of the experiment. Its range varies from 0 to 1, and 0 is the best value.
- *Facing to source* (f_s): This is a parameter that provides the percentage of times that the robot is facing the odor source and it considers a robot's detection area delimited by 45° with respect to the heading orientation. Its range varies from −1 to 0, with 0 representing the best value.
- *Getting closer* (N_c): It evaluates the percentage of movements associated with the times when the robot was actually moving closer to the source. Its range varies from −1 to 0, with 0 as the best value.
- *Arrived* (ϵ_a): This value serves as an additional 0.05 of the resulting evaluation in case the source has been reached by the robot.

Considering D_i as the initial distance from the robot to the source, D_f the final distance and D_{max} the maximum initial distance; hence, the parameters and the fitness function can be obtained as follows:

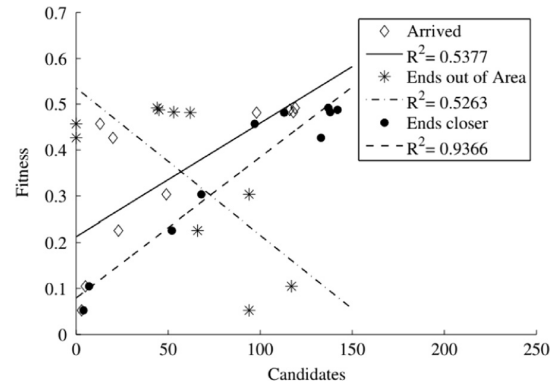

Fig. 6. Correlation between fitness and parameters of the algorithm. In particular, it outlines the number of experiments that ended closer to the source, reach the source (arrived), or get lost (ends out of area).

$$\Delta D = \begin{cases} -0.75 & \text{if } \dfrac{(D_f - D_i)}{D_{max}} < -0.75 \\ \dfrac{(D_f - D_i)}{D_{max}} & \text{otherwise,} \end{cases} \tag{5}$$

$$t_u = \frac{t_{exp}}{t_{max}}, \tag{6}$$

$$f_s = \frac{\text{headings}}{t_{max}}, \tag{7}$$

$$N_c = \frac{\text{times robot is moving closer}}{t_{max}}, \tag{8}$$

$$\epsilon_a = \begin{cases} 0.05 & \text{if robot reaches the source} \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

$$f_n = \Delta D \times 0.5 + (1 - t_u) \times 0.2 + f_s \times 0.25 + N_c \times 0.05 + \epsilon_a. \tag{10}$$

As seen in Eq. (10), the fitness for each candidate solution (f_n) is equal to the weighted sum of the individual parameters that were evaluated according to their importance in achieving the main goal assigned to the algorithm. First, 10 different algorithms were obtained with Genetic Programming using this fitness function but without weighted average. Then, weights of each parameter were adjusted after comparing 200 runs of each algorithm by using the sum of their fitness against the total amount of experiments that reached the source (arrived), get closer to the source (ends closer), or get lost (ends out of the experimental area). The aim was to obtain the trend lines for each comparison trying to reach an R^2 bigger than 0.5, hence indicating a linear tendency. Fig. 6 shows the results of using the weights mentioned above.

Finally, the probabilities for crossover and mutation were 0.5 and 0.05 respectively. They were defined with these values because even when the objective is to look for new and different algorithms, it is trying not to lose important information at the same time. In this case, by experimentation we notice that if mutation was bigger than 0.05, the candidates of the new generation result to be completely different due to the structure of the functions set. For example, if the robot had a behavior of "moving forward 3 steps", a mutation could change it in such a way that, instead of moving forward, it "turns to the left" in one of those steps. This is a good strategy to create new individuals but in this specific solution, it should not be that common because it diversifies the candidates in such a way that the number of generations

needed to reach the expected overall fitness increases exponentially. On the other hand, crossover helps us to diversify by 50% the behavior of the candidates but in turn they retain most of its structure. The roulette technique was used as the selection method. The number of candidate solutions was 100, and the GP algorithm evolved during 40 generations because the overall fitness stabilizes around this number of generations. The best final solution was a variation of gradient ascend with the particularity that instead of a constant $k=2$ the proposed solution defined $k=1$. Moreover, instead of saving the measurements during a time t_m, the algorithm waits until the inhalation cycle has finished.

5. Experiments and results

Evolutionary computation has proved to be a successful tool for solving problems in robotics. In this work, we show that genetic programming can be seen as a viable methodology for synthesizing odor source localization techniques and the results provide evidence about its usefulness within such robotics problem. Thus, to validate our claim, a comparison against three different algorithms including two of the state-of-the-art and one designed with the proposed technique were tested in simulation and with a real working system using an indoor environment. The artificial nose designed was implemented as the perception system in all cases, so the mathematical model of the nostril was considered for each algorithm also in simulation. All algorithms can be written as syntax trees as follows:

- Rozas: “progn2(EvalCircle,if(Goback,HoldOn))”.
- Basic: “progn3(MeasureAverage,HoldOn,if(progn3(Turn-measured,Jump,Move),Move))”.
- GP1: “progn2(HoldOn,if(progn2(MeasureTurn,Jump),Move))”.

Where “EvalCircle” is a function that turns the robot each 90° while obtaining the gradient direction.

It is important to keep in mind that the function “Move” turns on the motors of the robot for one step (1 s) at a constant speed of 0.2 m/s, while “Jump” does it for two steps. Each olfactive cycle lasts 30 s. Algorithms wait at least one olfactive cycle to decide the next direction to turn or move; thus, the speed of the robot is not crucial. The GP algorithm is capable of using both functions as many times as needed to reach more distance between measurements.

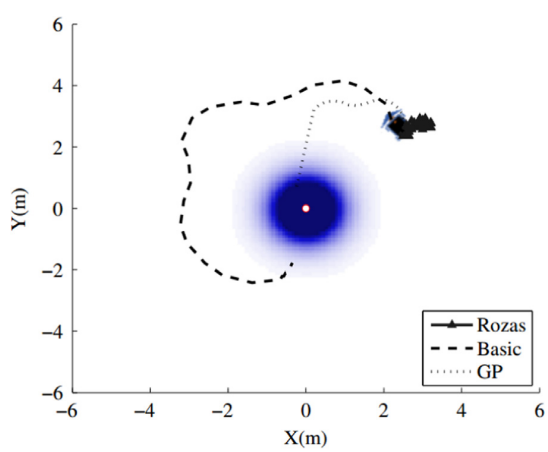

Fig. 7. Typical paths for comparison of the three algorithms.

5.1. Simulation results

The problem of odor source localization requires costly experiments that demand a controlled environment working with the same initial conditions. Simulation is seen as the solution for running and comparing multiple algorithms whose results were tested after 200 computing experiments for each solution. In this section, we discussed the algorithms behaviors that were obtained after using the mathematical models for odor propagation and perception, and how data were obtained during the execution of the algorithms. Fig. 7 shows the trajectories that were achieved through simulation, for the case where the robot travels a distance of 3.5 m from its source. In the experiments, the heading of the robot was perpendicular to the odor propagation direction.

A typical behavior attained by Rozas algorithm is that the robot's movement falls into a loop while moving backward and forward. This is probably due to the odor propagation and the relative angle of the robot with the source. In this way, regardless of whether the wind speed is high or low, when the robot moves near the source just a little out of center's odor-trail, the sensor will perceive a lower concentration and it will move back again with higher probability. As a result, it will find the same pattern over and over again even when the odor propagates by an ideal diffusion during the simulation. Other interesting results were zig-zag and spiral movements since both represent common behaviors of the robot performed under the basic algorithm. Sometimes the robot runs near the source but with some deviation due to the selected threshold because the difference between the two nostrils might be insufficient to create a turn, which causes the robot to pass by and take longer to relocate the source.

Each experiment had a timeout of 350 steps or around six minutes. The time taken for each GP iteration was variable because the experiments that reached the source ended before the timeout limit. The principal improvement between iterations was the averaged time used and the successful attempts of achieving the task. Table 2 shows the statistical results of the elapsed time (steps) and behavior of 10 different experiments of the basic algorithm. The minimum steps needed to complete the task were 94, achieved on the ninth experiment. It was the only candidate that achieve the task. Only ten experiments are shown at this point because it is necessary to notice that most of the behaviors were turns instead of straight movements.

The experiments of the GP1 algorithm were completed based on the same initial conditions and variables. It is clearly noticed that the trajectories of the robot follow a straighter path. In the first 10 experiments presented in Table 3, the odor zone was reached seven times. The minimum steps needed to complete the task were 110. The average steps to reach the odor zone were 217, while using the basic algorithm were 325, which means that the GP1 algorithm is 33% faster than the basic. On the other hand, the relation of turns compared with straight movements are 3:4 with the GP1 (3 turns for each 4 straight movements), while using the basic algorithm it is 3:2 (3 turns for each pair of straight movements).

A further analysis takes into account the results of 200 experiments for each algorithm and is shown in Table 4. The

Table 2
Results of 10 example experiments with the simulation environment using the basic algorithm. Results are classified as successful (S), close to target (C) or failure (F). “C” refers to runs that end near the source without reaching it.

Experiment	1	2	3	4	5	6	7	8	9	10
Elapsed time	350	350	350	350	350	350	350	350	94	350
Straight movements	91	90	100	103	90	88	93	90	29	92
Turns	143	142	134	131	145	146	141	144	34	143
Outcome	F	C	F	F	C	F	C	F	S	F

analysis considers the number of experiments that at the end of the run were closer or arrived to the source, the ones that did not arrive to the source, and how many ended farther away or even out of the experimental area.

Finally, the fitness was calculated as in the GP system by providing a quantitative comparison. In the results, we can observe that in Rozas algorithm, around 66% of the experiments (candidate solutions) finished closer to the source by measuring the relative distance to its initial position. Nevertheless, only 10% of the experiments reached the source. The impact of this algorithm is that none of the candidates ended out of the experimental area unlike Basic and GP1. Thus, more than 33% ended farther away from the source location. On the other hand, GP1 shows an important increment in fitness, basically because the candidates that reached the source represent almost 60% of the total amount and, similar to Rozas algorithm, 137 or 68% finished closer and only 63 or 31% ended farther away from the source, of which 44 ended out of the area. Fig. 8 shows examples of the trajectories that were achieved by the basic and GP1 algorithms.

By simply observing these results it is understood that both algorithms behave different even when they are topologically very similar. Nonetheless, this dissimilarity should be expressed statistically based on its behavioral characteristics. In this case, the objective function is not the best measure to directly conclude that both methods are different, since different algorithms can obtain a similar fitness or similar algorithms can obtain a different fitness depending on their behavior. In order to determine if both methods are different in behavioral space, we represent behaviors using behavior signatures (as in [34]), where authors compare each path followed by the robot using the Hamming distance over each algorithm and determine similarity between both of them using non-parametric tests. The comparison between 200 experiments of both algorithms is shown in Table 5.

The null hypothesis H_0 represents that both methods belong to the same distribution. Under different tests it was obtained $p < 0.01$, thus rejecting it and leading us to conclude that both algorithms behave statistically different.

Table 3
Results of 10 example experiments with the simulation environment using the GP1 algorithm. Results are classified as successful (S), close to target (C) or failure (F). "C" refers to runs that end near the source without reaching it.

Experiment	1	2	3	4	5	6	7	8	9	10
Elapsed time	220	100	253	111	350	350	220	193	230	136
Straight movements	85	39	100	41	150	148	80	72	90	56
Turns	62	28	69	31	84	86	67	57	64	35
Outcome	S	S	S	F	F	C	S	S	S	S

5.2. Real-world experimental setup

The aim of this research is to improve chemotaxis-based algorithms for real indoor environments through the application of GP and a biologically inspired nose system. The experimental setup also provides evidence about the independence between odor source localization techniques and wind measurements for indoor conditions. The robot uses only measurements of chemical substances in the environment to find, track and locate, a chemical plume based on the ability to determine the direction of the source placed at a given distance. The biologically inspired perception system computes the direction of the source by measuring the concentration difference in a pair of positions during a single time with the capability of emulating the olfactive cycle.

The physical experiments were realized in the "Laboratorio de Robótica del Área Noreste y Centro de México" at the Center for Robotics and Intelligent Systems of Tecnológico de Monterrey. The experiments consisted of determining the direction to the source in less than 15 min, as well as travelling and reaching the odor area, which is defined by a radius of 0.5 m, around the source, due to the robot's size and mechanical limitations. The experimental space was an indoor environment with a semi-closed area of $5 \times 5\ m^2$. The complete room is approximately four times bigger than the experimental area and it has a height of 10 m. Fig. 9 shows the layout of the room.

Table 4
Comparison results between three different algorithms.

Algorithms	Ended closer	Arrived	Did not arrive	Ended farther	Ended out of area	Fitness (−135 to 150)	Normalized fitness
Rozas	133	20	113	67	0	−13.51	0.426
Basic	56	25	31	144	46	−6.95	0.449
GP1	137	119	18	63	44	5.08	0.492

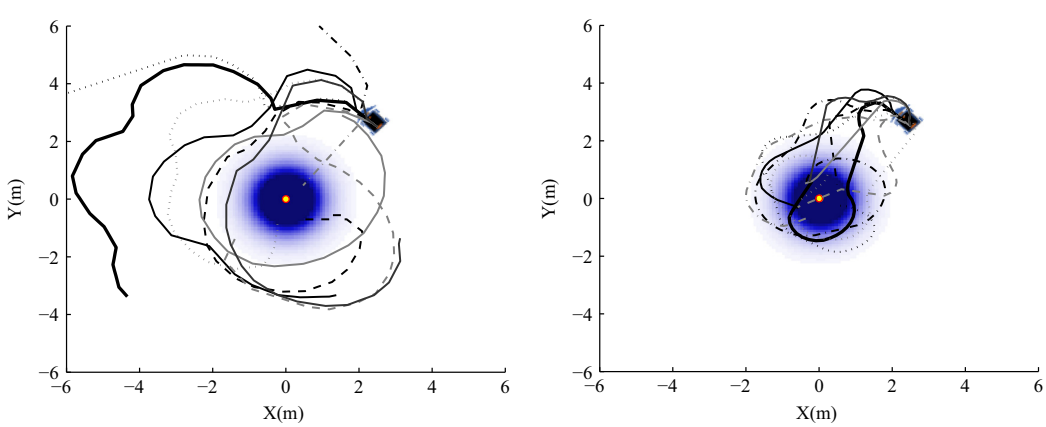

Fig. 8. (Left) Example paths of basic algorithm. (Right) Example paths of GP1 algorithm.

Table 5
Results of non-parametric tests between Basic and GP1 algorithms over behavioral space.

Algorithm	Test	p-value	H_0 vs H_A	Details	Significance (%)
Basic	Lilliefors	0.001	False	No normality	5
GP1	Lilliefors	0.001	False	No normality	5
Basic vs GP1	Kolmogorov–Smirnov	$1.3531e^{-25}$	False	Different distribution	5
Basic vs GP1	Wilcoxon	$3.2523e^{-16}$	False	Different distribution	5
Basic vs GP1	Kruskal–Wallis	$3.3007e^{-16}$	F	There is median difference	1

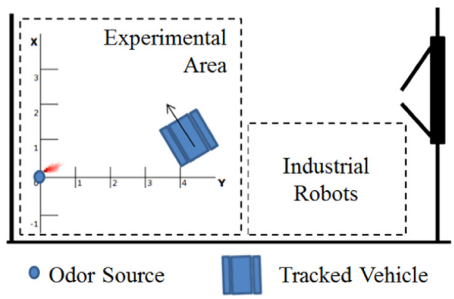

Fig. 9. Layout of the experimental area that was used to test the tracking algorithms.

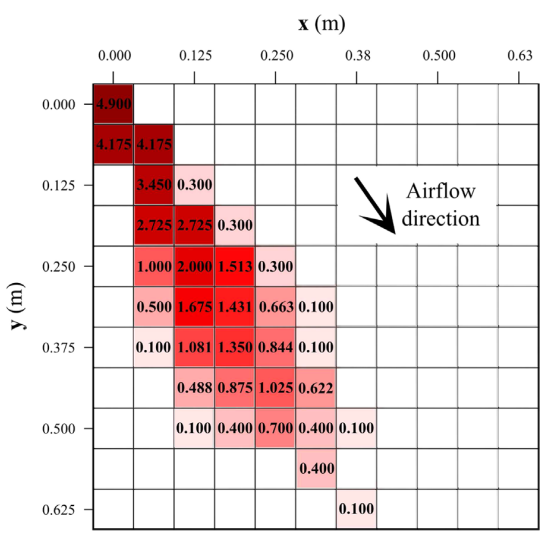

Fig. 10. Experimental grid map obtained with a vane anemometer using readings in a scale of m/s. The active range of the sensor is from 0.4 to 30 m/s and has a resolution of 0.1 m/s. The results consider an average of 10 measurements per position. Darker colors represent higher speeds. At distances bigger than 0.650 m the airflow is imperceptible and for this purpose is considerate that the odor propagates only by diffusion. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

The experimental setup considers the room airflows together with the odor source acting to simulate a gas leak. In the experimental area, the wind speed was reported manually every 6.25 cm using an anemometer. The sensibility of the sensor was 0.1 m/s, hence airflows below that range were not perceived by the anemometer. We designed a grid map of 80 × 80 measurement positions. Fig. 10 shows the results. The pressure at the source output was around 0.75 bar. The corresponding airflow generated at this position was 4.9 m/s. Note that a strong perception of a laminar flow takes place at the surroundings of the source. As a result, the odor source produces a small airflow negligible after 0.5 m distance. Some other airflows may occur but cannot be measured due to sensor resolution, which are inclusively below the capabilities of human perception. The airflow is scaled up to fit proportionally the experimental area shown in Fig. 9.

A tracked ground vehicle serves as transport for the implementation of the nose system and CAN network subsystem. Fig. 11 illustrates that the system can be mounted and thus be replicated on other types of vehicles.

The subsystem consists of three modules or nodes:

- A microcontroller that reads the voltages issued from the array of sensors for each nostril and sends the information to the bus. It also coordinates the inhalation and exhalation processes.
- A second microcontroller reads the information of an inertial measurement unit and puts it on the bus.
- A RF link takes the information and broadcasts it to a near computer for further processing.

5.3. Real-world results

A comparison between three algorithms under a real-world environment is reported in this section. The algorithms can behave differently for each experiment since the environmental conditions are not exactly the same and the sensor may respond with small variations at each time. Moreover, airflows below 0.1 m/s can be present during the experiments, hence increasing the variations. Thus, the robot must be capable of reaching the source under such indoor conditions by following the odor trail through chemotaxis. In the studied scenario the robot moves in one quadrant since the movement over the rest of the quadrants is restricted by the room walls. In this way, further work should consider the implementation of obstacle avoidance algorithms aided by other sensors such as cameras, sonars or lasers.

As stated before, this algorithm was evolved to find a base solution for odor source localization at indoor environments with a constant gas leak using only the smell sense, which has not been achieved properly in the state-of-the-art and is useful in many applications previously described. Further work and retraining of a general solution will be needed to involve different airflow models and configurations, turbulence, obstacle behavior, among other characteristics.

In these experiments, the source outlet was placed near the origin at (0, 0, 0.15) and the robot was placed at (−0.25, 3.25, 0) all coordinates are given in meters with the sensor placed at 0.15 m from the floor with its heading pointing towards the perpendicular of the odor source. The center of mass of the septum is considered by the position of the robot. As discussed earlier, the airflow is insufficient to be perceived by the robot at that distance hence it needs to look for the odor source. The surroundings of the source below 0.5 m are considered together with the area to be reached. In this case, the total time for the robot to reach the source area was 15 min, in order for the result to be considered as a positive run. Fig. 12 shows the initial stage of the experiments.

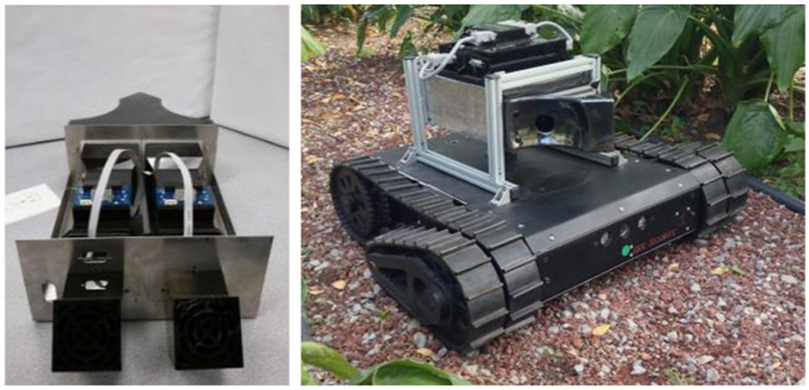

Fig. 11. Implementation of complete system in the vehicle.

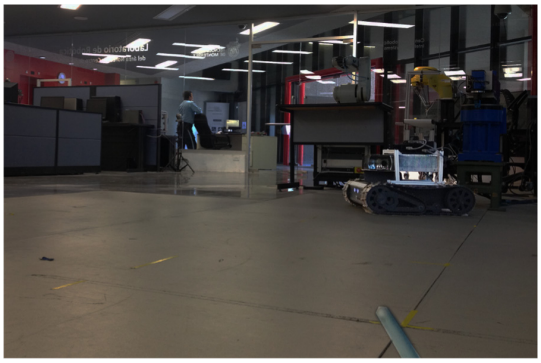

Fig. 12. Initial experimental setup.

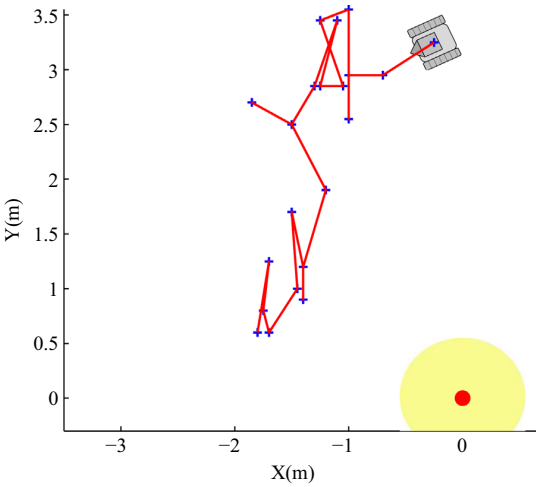

Fig. 13. Robot's path using the Rozas algorithm.

In Rozas algorithm the same simulated pattern was obtained after reaching the allowed time and the robot ended at (−1.8, 0.6, 0) pointing towards the opposite direction of the source. Indeed, it means that the robot ended closer but does not reach the source within the period of time. We noted that the robot seems to move only in one direction. It was also noticed in the experiments that the robot again falls into loops going forward and backward while passing through the same visited positions over and over again. Fig. 13 shows the path of the robot in a typical experiment.

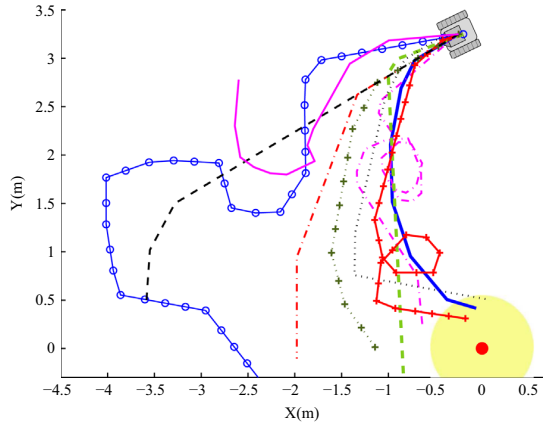

Fig. 14. Trajectories of the robot while using the basic algorithm. The circle with radius 0.5 m at position [0,0,0] delimits the zone to be reached.

Table 6
Results with a real-world environment using the basic algorithm. Results are classified as successful (S), close to target (C) or failure (F). "C" refers to runs that end near the source without reaching it.

Experiment	1	2	3	4	5	6	7	8	9	10
Elapsed time	840	604	159	333	900	195	248	290	285	457
Straight movements	69	44	14	28	60+	16	19	24	17	38
Turns	23	15	6	2	10+	4	2	5	2	13
Outcome	C	C	S	C	F	C	F	S	C	S

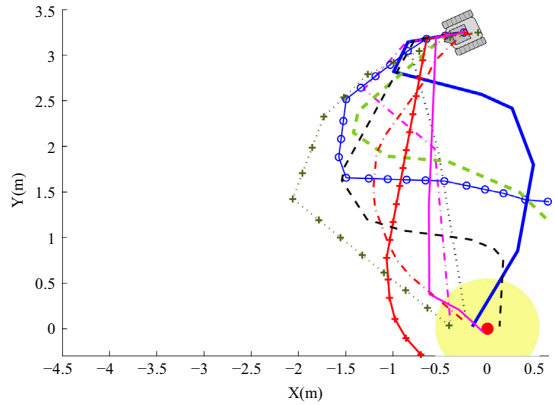

Fig. 15. Trajectories of the robot while using the GP algorithm. The circle with radius 0.5 m at position [0,0,0] delimits the zone to be reached.

A second issue of this algorithm is odometry and knowledge of its exact orientation's reference. As the robot rotates in different directions, it reaches its initial angle while pointing to the direction of maximum concentration. Thus, the absolute orientation needs to be calculated all the time with high accuracy. In a real experiment, it is a difficult task to perform since the robot can slid or skid during maneuvering. Hence, a good control of the rotational yaw must be used to achieve better results; otherwise the robot can get lost easily.

The performance of the basic algorithm and our GP1 were evaluated by running a set of ten experiments for each of them. Fig. 14 reports the paths achieved by the basic algorithm. The odor zone was reached in only three attempts. Note that it represents the same level of percentage reached within simulation. Also, it ended farther away two times while being five times closer to the target. According to the results we noticed that some of the trajectories are non-linear and even fall into spiral-shape behaviors. This is probably one major drawback of the basic algorithm since it affects the measurement time adjustments and computer's memory.

Table 6 shows the results of the elapsed time for each experiment. The minimum time needed to complete the task was 159 s and it was achieved on the third experiment. Also, the longest successful experiment took 457 s, which corresponds to experiment ten. Hence, on average, over all successful runs, the time to complete the task was 302 s.

In this way, the GP1 algorithm considered the same initial conditions, prototype and adjustment variables. Fig. 15 shows the results of ten experiments. The solutions show that the robot trajectories follow a straighter path – while being rapidly directed towards the odor source – when compared with previous algorithms, hence achieving a superior rate of successful results. In the experiments, the odor zone was reached seven times; it represents the same percentage level of the simulation. The rest of the experiments ended closer, which means that it had a failure rate of 0%. The experiments that ended close to the target were forced to stop because the robot got stuck at the walls. This problem could be improved by using obstacle avoidance techniques in future efforts. Nevertheless, these experiments cannot be considered as reporting a failure. As a result, we could also say that the elapsed time to reach the target was decreased with the GP algorithm.

Table 7 shows the results of ten different experiments. In the first experiment, the minimum time needed to complete the task was 84 s while experiment nine took 304 s and the average time to complete the task was 148 s. Those numbers were calculated by considering only the successful experiments.

Table 7
Results with a real-world environment using the GP1 algorithm. The solutions are classified as successful (S), close to target (C) or failure (F). "C" refers to runs that end closer to the source but do not reach it.

No experiment	1	2	3	4	5	6	7	8	9	10
Elapsed time	84	110	190	145	132	175	168	120	304	122
Straight movements	12	14	23	19	18	20	20	15	41	16
Turns	3	3	10	6	2	5	10	4	8	8
Outcome	S	S	C	C	S	C	S	S	S	S

In the basic algorithm, each time that the robot turns, the program moves the robot one step farther in comparison with the GP1 algorithm. This aspect is considered for the count of straight movements. As a result of this analysis, we can say that both straight movements and turns are more effective in GP1 than in the basic algorithm. In the case of the faster runs of both algorithms, the basic algorithm took twice as many turns and close to 15% more straight movements than GP1 for reaching the source, but required almost twice as long. It means that the robot takes more time to turn than advance. In this way, the evolved perception system was designed to properly help in calculating the direction, while the main difference relies in the capability of the algorithm's reaction.

The same statistical analysis was done in order to determine if both methods are different in behavioral space. The comparison between ten experiments of both algorithms is shown in Table 8. It was determined that both algorithms are different because $p < 0.05$ and also rejects the null hypothesis.

6. Conclusions

This paper presented simulation and experimental results of three different algorithms mimicking chemotaxis as performed by a robot. The algorithms for odor source localization were implemented in a simulation and successfully deployed in a real-world environment. The results in simulation match those achieved during the real-world test and therefore we can say that it is possible to design solutions to the OSL problem by the proposed technique. In fact, a better solution was obtained using GP. The quality achieved by the GP algorithm was significantly improved in comparison with two state-of-the-art chemotaxis algorithms.

In the case of the real-world experiments, three different algorithms were tested using a classical robot's trajectory as a way to outline the advantages and drawbacks of each proposal. Finally, a comparison of the hand-made algorithm against the computer synthesized algorithm was statistically analyzed through the results achieved in the real-world test that matches the simulation.

In the case of the basic algorithm, the robot attempts to cover the whole room in order to determine the position of the odor source. Nevertheless, according to the GP algorithm presented in this work, the time needed for reaching the odor source was of 84 s while the robot achieved this task by following the odor trail using a straighter path with a minimum number of movements. As a result of applying the synthesized algorithm using our artificial nose system, the overall performance to be expected is about 70% of successful attempts at reaching the odor source, while the remainder will end closer to the source. We note that during the experiments, the quality of the results was hampered by the mechanical characteristics of the robot and not by the perception system. Further work should devise new ways of improving the level of successful reaching attempts through the combination of vision and chemotaxis algorithms, obstacle avoidance techniques, and other proposals.

Table 8
Non-parametric tests over ten runs of Basic and GP1 algorithms over behavioral space.

Algorithm	Test	p-value	H_0 vs H_A	Details	Significance (%)
Basic	Lilliefors	0.001	False	No normality	5
GP1	Lilliefors	0.001	False	No normality	5
Basic vs GP1	Kolmogorov–Smirnov	0.0187	False	Different distribution	5
Basic vs GP1	Wilcoxon	0.0275	False	Different distribution	5
Basic vs GP1	Kruskal–Wallis	0.0282	False	There is median difference	1

Acknowledgements

This work has been supported by the National Council of Science and Technology of Mexico (Consejo Nacional de Ciencia y Tecnología - CONACyT), the "Laboratorio de Robótica del Área Noreste y Centro de México" founded by CONACyT, the Focus Group on Robotics at Tecnológico de Monterrey, as well as by the EvoVisión Team of CICESE and the project 155045 - "Evolución de Cerebros Artificiales en Visión por Computadora". First author supported by scholarship 32081 from CONACyT. In particular we want to thank Guy Albert Cardinau, Profesor at Tecnológico de Monterrey, for its advice and support as specialist in scientific writing.

References

[1] C. Browne, K. Stafford, R. Fordham, The use of scent-detection dogs, Ir. Vet. J. 59 (2) (2006) 97.
[2] G. Kowadlo, R.A. Russell, Robot odor localization: a taxonomy and survey, Int. J. Robot. Res. 27 (8) (2008) 869–894.
[3] N.J. Vickers, Mechanisms of animal navigation in odor plumes, Biol. Bull. 198 (2) (2000) 203–212.
[4] J. Rozas, R. Morales, D. Vega, Artificial smell detection for robotic navigation, in: 5th International Conference on Advanced Robotics (ICAR), 'Robots in Unstructured Environments', vol. 2, 1991, pp. 1730–1733.
[5] H. Ishida, T. Nakamoto, T. Moriizumi, Study of odor compass, in: IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems, 1996, 1996, pp. 222–226. http://dx.doi.org/10.1109/MFI.1996.572181.
[6] T. Nakamoto, H. Ishida, T. Moriizumi, Active odor sensing system, in: ISIE '97 – Proceedings of the IEEE International Symposium on Industrial Electronics, vols. 1–3, 1997, pp. 128–133.
[7] A. Lilienthal, T. Duckett, A stereo electronic nose for a mobile inspection robot, in: Proceedings of the IEEE International Workshop on Robotic Sensing (ROSE 2003), 2003.
[8] A. Lilienthal, T. Duckett, U. Nunes, A. deAalmeida, A. Bejczy, K. Kosuge, J. Macgado, Experimental analysis of smelling braitenberg vehicles, in: 11th International Conference on Advanced Robotics, 2003, pp. 375–380.
[9] P.P. Neumann, V. Hernandez Bennetts, A.J. Lilienthal, M. Bartholmai, J. H. Schiller, Gas source localization with a micro-drone using bio-inspired and particle filter-based algorithms, Adv. Robot. 27 (9) (2013) 725–738, http://dx.doi.org/10.1080/01691864.2013.779052 arXiv:http://www.tandfonline.com/doi/pdf/10.1080/01691864.2013.779052, URL 〈http://www.tandfonline.com/doi/abs/10.1080/01691864.2013.779052〉.
[10] D. Agdas, G.D. Webster, F.J. Masters, Wind speed perception and risk, PLoS One 7 (11) (2012) e49944, http://dx.doi.org/10.1371/journal.pone.0049944, pONE-D-12-08991[PII] 23226230[pmid] PLoS One, URL 〈http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3511475/〉.
[11] J. White, J.S. Kauer, Odor recognition in an artificial nose by spatio-temporal processing using an olfactory neuronal network, Neurocomputing 2627 (1999) 919–924, http://dx.doi.org/10.1016/S0925-2312(98)00137-4, URL 〈http://www.sciencedirect.com/science/article/pii/S0925231298001374〉.
[12] T. Roppel, D.M. Wilson, Biologically-inspired pattern recognition for odor detection, Pattern Recognit. Lett. 21 (3) (2000) 213–219, http://dx.doi.org/10.1016/S0167-8655(99)00150-6, URL 〈http://www.sciencedirect.com/science/article/pii/S0167865599001506〉.
[13] I. Valova, N. Gueorguieva, Y. Kosugi, An oscillation-driven neural network for the simulation of an olfactory system, Neural Comput. Appl. 13 (1) (2004) 65–79.
[14] G. de Croon, L. O'Connor, C. Nicol, D. Izzo, Evolutionary robotics approach to odor source localization, Neurocomputing 121 (2013) 481–497, Advances in Artificial Neural Networks and Machine Learning Selected papers from the 2011 International Work Conference on Artificial Neural Networks (IWANN 2011). http://dx.doi.org/10.1016/j.neucom.2013.05.028 URL 〈http://www.sciencedirect.com/science/article/pii/S0925231213005869〉.
[15] E. Schaffernicht, M. Trincavelli, A.J. Lilienthal, Bayesian spatial event distribution grid maps for modeling the spatial distribution of gas detection events, Sens. Lett. 12 (6-7) (2014) 1142–1146.
[16] J. Zhang, D. Gong, Y. Zhang, A niching pso-based multi-robot cooperation method for localizing odor sources, Neurocomputing 123 (2014) 308–317, contains Special issue articles: Advances in Pattern Recognition Applications and Methods. http://dx.doi.org/http://dx.doi.org/10.1016/j.neucom.2013.07.025 URL 〈http://www.sciencedirect.com/science/article/pii/S0925231213007698〉.
[17] J. Crank, The Mathematics of Diffusion, 2nd Ed., Clarendon Press, Bristol, England, 1975.
[18] J. Li, Q. Meng, Y. Wang, M. Zeng, Odor source localization using a mobile robot in outdoor airflow environments with a particle filter algorithm, Auton. Robots 30 (3) (2011) 281–292.
[19] A. Loutfi, S. Coradeschi, L. Karlsson, M. Broxvall, Putting olfaction into action: anchoring symbols to sensor data using olfaction and planning, in: Planning and Learning in A Priori Unknown or Dynamic Domains, 2005, p. 35.
[20] A. Loutfi, S. Coradeschi, A. Lilienthal, J. Gonzalez, Gas distribution mapping of multiple odour sources using a mobile robot, Robotica 27 (2009) 311–319.
[21] B.L. Villarreal, J.L. Gordillo, Perception model for the aspiration process of a biologically inspired sniffing robot, in: 2013 18th International Conference on Methods and Models in Automation and Robotics (MMAR), 2013, pp. 334–339.
[22] B.L. Villarreal, C. Hassard, J.L. Gordillo, Finding the direction of an odor source by using biologically inspired smell system, in: Advances in Artificial Intelligence (IBERAMIA), Lecture Notes in Computer Science, vol. 7637, Springer, Berlin, Heidelberg, 2012, pp. 551–560.
[23] B. Villarreal, G. Olague, J. Gordillo, Odor plume tracking algorithm inspired on evolution, In: J. Martnez-Trinidad, J. Carrasco-Ochoa, J. Olvera-Lopez, J. Salas-Rodrguez, C. Suen (Eds.), Pattern Recognition, Lecture Notes in Computer Science, Cancun, Mexico, vol. 8495, Springer International Publishing, 2014, pp. 321–330.
[24] J.G. Monroy, J. González-Jiménez, J.L. Blanco, Overcoming the slow recovery of mox gas sensors through a system modeling approach, Sensors 12 (10) (2012) 13664–13680.
[25] D. Martinez, O. Rochel, E. Hugues, A biomimetic robot for tracking specific odors in turbulent plumes, Auton. Robots 20 (3) (2006) 185–195, http://dx.doi.org/10.1007/s10514-006-7157-1.
[26] T. Lochmatter, X. Raemy, A. Martinoli, Odor source localization with mobile robots, Bull. Swiss Soc. Autom. Control 46 (2007) 11–14.
[27] B.L. Villarreal, J.L. Gordillo, Artificial Olfactory Method and System. Instituto Mexicano de la Propiedad Intelectual (IMPI), Mexico. Patent pending PCT/MX2013/000151, 2012.
[28] J. Gonzalez-Jimenez, J. Monroy, J. Blanco, The multi-chamber electronic nose-an improved olfaction sensor for mobile robotics, Sensors 11 (6) (2011) 6145–6164.
[29] D. Zarzhitsky, D. Spears, W. Spears, Distributed robotics approach to chemical plume tracing, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS), 2005, pp. 4034–4039. http://dx.doi.org/10.1109/IROS.2005.1545428.
[30] B.L. Villarreal, J.L. Gordillo, Perception aptitude improvement of an odor sensor: model for a biologically inspired nose, In: 5th Mexican Conference on Pattern Recognition, Lecture Notes in Computer Science, vol. 7914, Springer, Berlin, Heidelberg, 2013, pp. 126–135.
[31] U. Wilensky, Netlogo. 〈http://ccl.northwestern.edu/netlogo/〉, Center for connected learning and computer-based modeling, Northwestern University, Evanston, IL, 1999.
[32] L. Paic-Antunovic, D. Jakobovic, Evolution of automatic robot control with genetic programming, in: Proceedings of the 35th International Convention MIPRO, Opatija, Croatia, IEEE, 2012, pp. 817–822.
[33] C. Lazarus, H. Hu, Using genetic programming to evolve robot behaviours, in: Proceedings of the 3rd British Conference on Autonomous Mobile Robotics and Autonomous Systems, Manchester, 2001.
[34] L. Trujillo, G. Olague, E. Lutton, F. Fernndez de Vega, L. Dozal, E. Clemente, Speciation in behavioral space for evolutionary robotics, J. Intell. Robot. Syst. 64 (3–4) (2011) 323–351, http://dx.doi.org/10.1007/s10846-011-9542-z.

B. Lorena Villarreal graduated with Honorific Mention the Bachelor degree as a Mechatronics Engineer from the Tecnológico de Monterrey, Campus Monterrey in 2008. She also took courses on automotive engineering and design at the Fachhochschule Braunschweig/Wolfenbutel, in Wolfsburg, Germany and on Lean Manufacturing endorsed by the Institute on Industrial Engineers. She obtained her Ph.D. degree in 2014 from Tecnológico de Monterrey and was invited as a visiting researcher to collaborate with the EVOVision Group at the computer department of CICESE in Baja California. She received the MITs Technology Review award called Innovators under 35 México Edition in 2014. Recently, the Royal Academy of Engineering in collaboration with the University of Oxford gave her the opportunity to participate in a training course on technology commercialization as part of the Leaders in Innovation Fellowship. Her research interests are autonomous robotics and artificial intelligent systems.

Gustavo Olague received the Ph.D. degree in Computer Vision, Graphics and Robotics from INPG and INRIA. He is currently a Professor in the Computer Science Department at CICESE in Ensenada. Professor Olague has written over hundred conference and journal papers and co-edited two special issues in Pattern Recognition Letters and Evolutionary Computation, as well as served as co-chair of the Real-World Application track at the Genetic and Evolutionary Computation Conference. Dr. Olague has received numerous distinctions such as the Talbert Abrams award offered by the ASPRS; best paper awards at major conferences like GECCO, EvoIASP, and EvoHOT; and received two times the Bronze Medal at the Human-Competitive awards at GECCO. He is the author of the book Evolutionary Computer Vision published by Springer.

J.L. Gordillo graduated in Industrial Engineering from the Technological Institute of Aguascalientes, Mexico. He obtained both the D.E.A. degree and the Ph.D. in Computer Science from the National Polytechnic Institute of Grenoble, France, in 1983 and 1988, respectively. From 1989 to 1990 he was an Assistant Professor at the Department of Automatic Control of the Center for Advanced Studies and Research of the National Polytechnic Institute of Mexico (CINVESTAV-IPN).

Currently he is Director and Professor at the Center for Robotics and Intelligent Systems and (CRIS) at the Tecnológico de Monterrey (ITESM). He has been a Visitor Professor in the Computer Science Robotics Laboratory at Stanford University (1993), at the Project Sharp of INRIA Rhone-Alpes in France (2002 and 2004), at LAAS-CNRS in Toulouse, France (2007–2008), and at some other universities and research institutes. His research interests are in computer vision for robotics applications, in particular autonomous vehicles, and the development of virtual laboratories for education and manufacturing. He participated and leaded R&D projects with industry like Honewell Bull in France, Sun Microsystems, Peñoles and TV Azteca; and government entities like the Mexican Army, the French-Mexican Laboratory for Computer Science (LaFMI), the Institute of the Water of the Nuevo Leon state (IANL), and the National Council for Science and Technology in Mexico (CONACyT). In particular, he promoted and leaded the National Net on Robotics and Mechatronics (RobMec). Actually he leads the Robotics National Laboratory, founded by CONACyT and ITESM, and the Robotics Research Focus Group at the ITESM.