Odor Plume Tracking Algorithm Inspired on Evolution*

B. Lorena Villarreal[1], Gustavo Olague[2], and J.L. Gordillo[1]

[1] Center for Robotics and Intelligent Systems, Tecnológico de Monterrey, Monterrey, Nuevo León, México
[2] EvoVision Project, Computer Science Department, Applied Physics Division CICESE, Ensenada, Baja California, México

Abstract. Smell sensors in mobile robotics for odor source localization are getting the attention for researches around the world. To solve the problem, it must be considered the environmental model and odor behavior, the perception system and the algorithm for tracking the odors plume. Current algorithms try to emulate the behavior of the animals known by its capability to follow odors. Nevertheless, the odor perception systems are still in its infancy and far to be compared with the biological smell sense. This is why, an algorithm that considers the perception system capabilities and drawbacks, the environmental model and the odor behavior is presented on this work. Besides, an artificial intelligent technique (Genetic Programming) is used as a platform to develop odor source localization algorithms. It is prepared for different environment conditions and perception systems. A comparison between this improved algorithm and a pair of basic techiques for odor source localization is presented in terms of repeatability.

Keywords: odor source localization, smell, genetic programming, sniffing robot.

1 Introduction

Smell sensors are being developed to distinguish all types of odors, intensities and concentrations. Odor source localization algorithms should be useful with any kind of odor sensors arrays and a good signal analysis that improve the measurement. The applications could be the detection of toxic gas leaks, the fire origin of a disaster, search and rescue operations, etc.

Smell sensors implemented on mobile robots started in 1984 with the use of chemical sensitive robots in the nuclear industry [1]. There are many algorithms used to support and increase the efficiency of odor source localization.These are most commonly classified by the terms of *chemotaxis* and *anemotaxis* depending on the environment and the capabilities of the odor sensors. Chemotaxis is used when the orientation and movement of the agent (mobile robot) is based on the chemical gradient of the environment [2]. On the other hand, anemotaxis, instead of following the gradient, considers the direction or current of a fluid [3,4] and the agent moves through it.

Some other algorithms for odor source localization that include predefined airflow models, different environment conditions, different types of odor sources and obstacle maps are described in [5,6,7,8].

* This research has been supported by CONACYT and Laboratorio de Robótica del área Noreste y Centro de México.

J.F. Martínez-Trinidad et al. (Eds.): MCPR 2014, LNCS 8495, pp. 321–330, 2014.

The principal disadvantage of this algorithms is the sensorial system itself. The chemical reactions change the sensor in a way that the recovery to its original state is slow [9]. In this research, genetic programming is used to evolve a solution considering the capabilities of the perception system, as well as its limitations. The work is based on the assumption that the direction from where an odor is coming can be obtained using the difference between a pair of nostrils as it was implemented on [10]. The results show that the algorithm obtained by this technique improves the achievement rate of common algorithms based on chemotaxis. It decreases the time to complete the task and increases repeatability.

This document is organized as follows. First, the complete definition of the problem is presented in section 2. In section 3 the theoretical analysis is shown. In section 4 is discussed the implementation and experimental set up. The preliminary results are shown in section 5. Finally, in section 6 the conclusions and future work are presented.

2 Problem Definition

There are three problems that need to be analyzed when solving the task of tracing odor sources with a robot: characterize the dynamic behavior of the atmosphere and odors; adequately perceive this environment so that the information can be useful for future analysis and the algorithm or technique to locate the odor source using this information.

Currently, techniques and nature-based algorithms emulate the behavior of some animals, such as casting, and sweeping spiral [11]. However, currently available sensors differ from the characteristics of the biological sensors that these animals have. This occurs basically because their brain does not only use this ability, but gathers information from all other sensors. This is the way in which the animals learned how to locate the odor. When a try of simulate this localization technique with a robot is done, the results are not the most optimal because the odor source is not located with high accuracy or requires a lot of task time to be reached.

However, if the 3 sub-problems are seen as one, considering the limitations of the sensors (desaturation time, concentration difference between sources, reaction time), features based on nature that must have at least, the mathematical model of it, a new technique with better results can be obtained by means of genetic programming. This task can be achieved regardless the information from other sensors such as anemometers.

In this research is presented the development of a genetic program (GP) that produces an odor tracking algorithm that integrates the simulation of a system device with the capability of directionality and the odor propagation model as the environmental conditions.

3 Theoretical Analysis

Different techniques for odor source localization can be used depending on the environment conditions and the perception system. Moreover, there are different configurations for the implementation of chemical sensors. The most commonly used into mobile robots are: directly exposed, continually exposed and cyclic exposed. The first one refers to the placement of the sensor without isolation. Monroy et al. are using a complete inverted sensor model [12] to obtain an estimation of the odor. The sensor is placed into the robot without isolation.

The second configuration is considered when airflow is induced directly to the sensors placed into an isolated chamber. In [13] are using a pair of chambers. An inlet pipe samples the surroundings by an airflow generated by a micro-pump emulating the inhalation stage of the ventilation process. Similar approaches [6,2,14] 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 odor no having time to recover its original state. In the other hand, cyclic exposed [15] refers to the use of a chamber with the capability of isolate the sensors from the environment for a certain time and prepare it for a new measurement.

Then, the development of algorithms for odor source localization must take in care the characterisitcs of both, environment and sensor model to achieve a better behavior. This way, the drawbacks of the physical implementation can de reduced. By genetic programming, these can be taken into consideration to produce an algorithm specific for the environment and perception system used. In this section, the environmental and sensor models in which the GP is based, will be explained.

3.1 Environmental Model

The propagation of odor molecules in the environment occurs in two different ways. When no airflows are present, the propagation is done by diffusion in a radial manner. On the other hand, when airflow is present, the propagation is done by advection in a laminar way.

In [16], 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, the random walk of molecules take place from a high concentration region to a low concentration region, depending on the concentration gradient, trying to homogenize the environment.

The general form for the diffusion equation is for a three-dimensional system is represented by

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

As a first approximation, in this research a constant diffusion source will be used as the environment condition.

3.2 Nose Model

The perception system considered for this work is based on [9,10], which implements a bio-inspired nose system with the capability of determine the direction from where an odor is coming. It is achieved by the use of a pair of notrils divided by a septum. In the inhalation process, the nose is able to concentrate the odor molecules near the sensorial system and at exhalation, the nose desaturates the sensors. This design complements the sensor model from [6,12] by including the cyclic behavior of a sensor placed into a chamber.

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

In real 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 into a continuous odorous environment is divided in two stages. For inhalation

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

and for exhalation

$$r_e(t) = \frac{r_e(t-1) - r_i(t)(exp(-\frac{(t-t_s-\Delta t_r-\Delta t_a)}{\tau_d})}{1-exp(-\frac{(t-t_s-\Delta t_r-\Delta t_a)}{\tau_d})},$$

where $r_i(t)$ and $r_e(t)$ are the concentration values during inhalation and exhalation at the actual ventilation cycle. Consequently, $r_e(t-1)$ is the concentration value of the last cycle. So, after each cycle the initial reference is updated by

$$r_e(t-1) = r_e(t).$$

Based on this design with the presented behavior, the modeling and simulation of its physical properties can be used to obtain an algorithm developed to work specifically taking advantage of its features.

4 Implementation and Experimental Set-Up

Three algorithms of odor source localization are compared. The first is the one used by Rozas [17]. The second is going to be called for this purpose as " the basic algorithm". It was designed using ascend gradient method. The third is the one obtained by evolution. Them are going to be described in this section.

As the physical implementation requires a controlled environment for multiple characterization experiments and because the odor is extracted hardly from this environment to have exactly equal initial conditions, the experiments take a lot of time. Due this reason, a simulation environment was developed using NetLogo [18] to run any kind and quantity of experiments.

In the environmental model designed, the diffusion rate and the wind can be controlled varying the direction and the intensity or speed. The initial position, concentration and quantity of the sources can be also controlled. The sources can be spraying the odor at constant time intervals or can be always spreading it.

This environment simulates the diffusion through the air from a source as well as the interaction of the robot with the source considering the mathematical model described in section 3.2. The selected environment was a fixed odor source that diffuses through the air and a mobile robot capable of measuring the concentration difference at two emulated nostrils positioned 45° and −45° respectively and 1 unit distance from the center of the robot. The three algorithms are represented as syntax trees, just as in genetic programming, using the same platform for all simulations.

4.1 Rozas Algorithm

The first robot implemented for odor source localization was presented by Rozas et al. [17] in 1991. The algorithm was design to follow odor gradients by taking spatial measurements at different times, thus, by chemotaxis. The algorithm used is shown in Fig. 1.

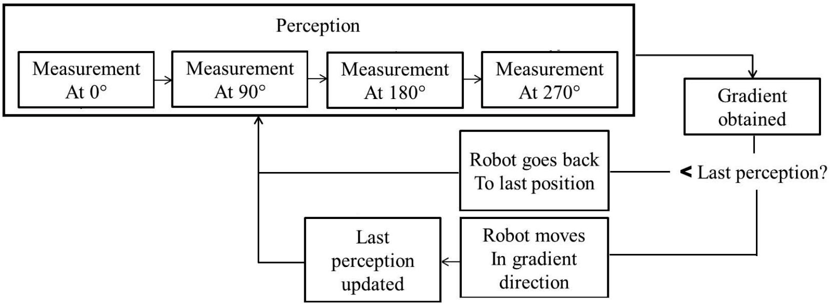

Fig. 1. Odor source localization algorithm implemented in [17] using chemotaxis

4.2 Basic Algorithm

It is a variation of the gradient descend method. The difference between two nostrils is used to detect the direction of the odor source and the step size of the robot, looking for the maximum concentration. The routine consists in two operating timed cycles: the aspiration process and the robot movement. In the aspiration process, each time the system inhales, the robot acquires odor concentration data through its sensors and saves it into the memory (Mem) of the adquisition system. In the other hand, the robot movement cylce starts by waiting certain time for measurement (t_m). Then, the average of the data accumulated by inhalation during this time lapse (t_i) is obtained. After that, it calculates its new direction and aligns with it. The direction of turn is limited by a maximum angle (θ_{max}). Finally, it moves s steps forward. A threshold (thr) is implemented to consider the uncertnainty between nostrils when the source is near to the front face of the robot. In that case the robot moves $\frac{s}{k}$, where k is an experimental constant. The algorithm is shown in Fig. 2. This routine continues until the robot reaches a saturation value and is considered to have arrived, or exceeding a preset time limit. It is important to consider that the measurements used to obtain the direction are only those when the system is between inhaling and exhaling. The measurements at exhalation are ignored.

4.3 Genetic Programming

Based on the understanding of that the bio-inspired nose has limitations, it can be noticed that a basic algorithm as presented before may not be the best solution. However, knowing the mathematical model and operation of the system, by means of genetic programming an adequated localization algorithm was evolved taking into account these constraints.

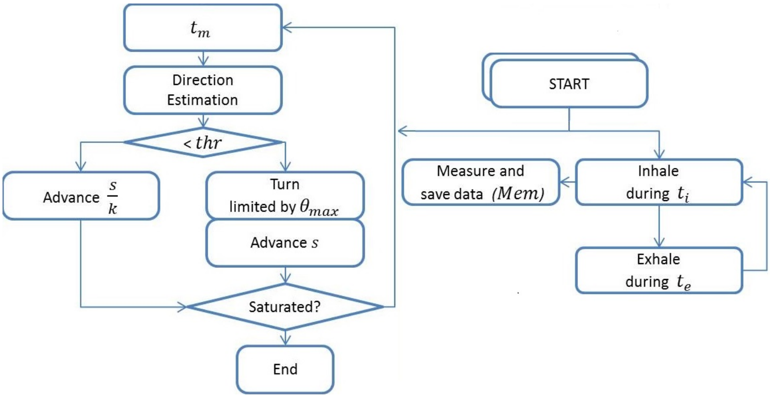

Fig. 2. Algorithm for reactive gradient ascend. The robot turns according to its measurement.

4.4 Parameters

NetLogo has an interface that communicates with Matlab and allows to exchange data between the two applications. The purpose of this integration is the use of genetic programming managed by this software whose experiments can be simulated by means of NetLogo. Matlab is responsible for the creation of the new generations based on the fitness while NetLogo runs the experiments an asigns the fitness on each candidate.

As in robotics, for this application the terminals are the actions of the robot [19,20]. The set of terminals for the GP was composed by:

– Move, Jump. Robot moves forward one or two steps respectively
– Measureminus. Robot average all measurements during last t_m
– Measurediff. Robot considers last measurement during last t_m
– MeasureTurn. Robot waits for sampling time and obtaines a measurement, then it turns depending on the nostril's difference.
– Turnmeasured. Robot turns in the direction calculated.
– 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 $90°$.
– Turnrandom45. Robot turns $45°$ in a random direction.
– HoldOn. Robots waits 1 time step.
– Goback. Robot turns $180°$ and moves forward one step.

The function set is composed by PROGN2, PROGN3 and IF(a,b). PROGN are the simplest nodes which are used for connecting parts of program together. It returns two or tree subtrees respectively in sequence. In the other hand, IF returns a when the treshold is reached and b otherwise.

The fitness function that evaluates each candidate is divided in 5 parameters:

– Distance reached (ΔD). At the end of the experiment, it evaluates how close or far the robot ends of the source relative to its initial position. Its range varies from -0.754 to 1, where 1 is better.

- Time Used (t_u). It is the time the robot takes to reach the source (t_{exp}) normalized by t_{max}, which is the time out of the experiment. Its range varies from 0 to 1, where 0 is better.
- Facing to Source (f_s). Considering that the robot's field of view (FoV) is at front, it refers to the percentage of times the robot's FoV is facing to the source. The FoV is considered $45°$. Its range varies from -1 to 0, where 0 is better.
- Getting closer (N_c). It evaluates the percentage of movements when the robot was actually moving closer to the source. Its range varies from -1 to 0, where 0 is better.
- Arrived (ϵ_a). It is an additional 0.05 evaluation if the source has been reached.

Considering D_i as the inital distance from the robot to the source, D_f the final distance and D_{max} the maximum initial distance, these parameters can be obtained as:

$$\Delta D = \begin{cases} -0.75 & \text{if } \frac{(D_f - D_i)}{D_{max}} < -0.75 \\ \frac{(D_f - D_i)}{D_{max}} & \text{otherwise} \end{cases}, \qquad \epsilon_a = \begin{cases} 0.05 & \text{if robot reaches the source} \\ 0 & \text{otherwise} \end{cases},$$

$$t_u = \frac{t_{exp}}{t_{max}}, \qquad f_s = \frac{\text{headings}}{\text{time steps of experiment}}, \qquad N_c = \frac{\text{times robot is moving closer}}{\text{time steps of experiment}}.$$

Finally, the fitness of each candidate (f_n) is the weighted sum of these parameters:

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

The weights of each parameter were calculated running 200 experiments of 10 different algorithms obtained by Genetic Programming. The results are presented in Table 1. The weights for each parameter were adjusted comparing the fitness versus: the quantity of experiments that at the end of the experiment reached the source, got closer to the source, or got lost (out of the experimental area). The trend lines for each comparison were adjusted trying to reach a r-squared bigger than 0.5 indicating a lineal tendency. Fig. 3 show the results using the weights mentioned before.

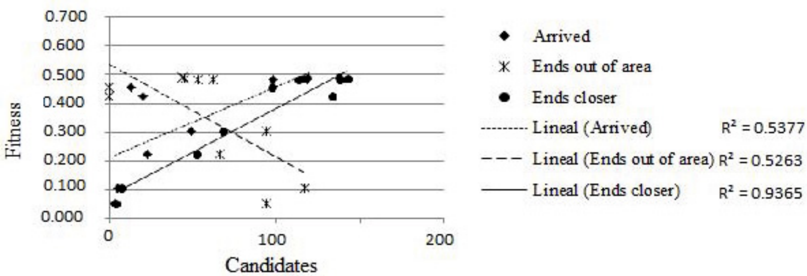

Fig. 3. Comparison between fitness and principal objectives of algorithm

Table 1. Comparison results of between 10 algorithms. 8 of them obtained by GP at different generations

	Ended Closer	Arrived	Didn't Arrived	Ended Farther	Ended Out of area	Fitness (-135 to 150)	Normalized Fitness
a1	133	20	113	67	0	-13.51	0.426
a2	52	23	29	148	66	-70.62	0.226
a3	142	116	26	58	45	3.87	0.487
a4	113	98	15	87	62	2.12	0.481
a5	68	49	19	132	94	-48.5	0.304
a6	4	3	1	196	94	-120.02	0.053
a7	138	118	20	62	53	2.42	0.482
a8	97	13	84	84	0	-4.82	0.457
a9	137	119	18	63	44	5.08	0.492
a10	7	5	2	193	117	-105.12	0.105

5 Experimental Results

Using genetic programming, a better algorithm was obtained. The probabilities for crossover and mutation were 0.5 and 0.05 respectively. These were defined with this 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. The roulette technique was used as the selection method. The number of candidates was 100 and the routine was evolved during 40 generations. The best algorithm obtained was a variation of gradient ascend, were instead of constant $k = 2$ was $k = 1$ and instead of save the measurements during t_m the algorithm waits until the inhalation cycle have finished. Fig. 4 represents the algorithm.

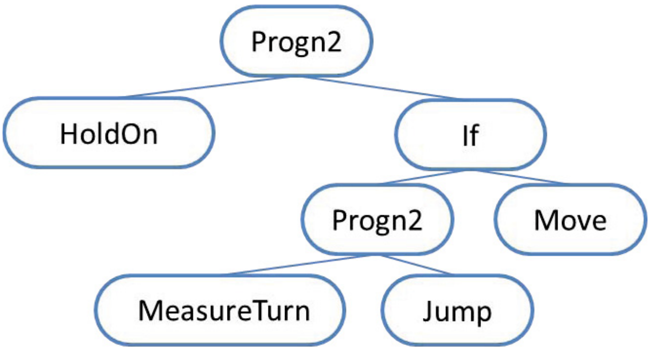

Fig. 4. Best so far algorithm at last generation. The robot waits for inhalation and then samples the odor. It turns and moves depending on this measurement.

Finally, the sintax trees compared were:

- Rozas. “progn2(evalcircle,if(goback,antmove))”
- Basic. “progn3(measureminus,HoldOn,if(progn3(turnmeasured,jump,move),move))”
- GP1. “progn2(HoldOn,if(progn2(MeasureTurn,jump),move))”

Table 2 show the results. It can be seen that in Rozas algorithm, around 60% of the experiments (candidates) finished closer to the source relative to its initial position. Nevertheless just 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. However, around 30% ended farther. In 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 just as in Rozas, only 30% ended farther.

Table 2. Comparison results between three different algorithms

	Ended Closer	**Arrived**	**Didn't Arrived**	**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

6 Conclusion and Future Work

A GP development was presented in this work for an odor plume tracking algorithm. Thanks to the obtained results, it can be said that, genetic programming is a powerful tool to develop odor source localization algorithms. A better solution was presented showing that the uncertainty of achievement was decreased.

Considering the capabilities of the perception system and the odor propagation model, the platform is prepared to run for several environments with different characteristics and perception systems. The next step is to find an algorithm in an environment where airflow is present, regardless the use of another kind of sensors. It must be compared with the common algorithms used in the literature. Then, the inclusion of obstacles and dynamic sources would be an interesting approach.

Acknowledgments. Specially thanks to CONACYT that has been supporting the research in conjunction with the Laboratorio de Robótica del área Noreste y Centro de México and to EvoVisión Team from CICESE that are providing the tools and expertise for this research.

References

1. Larcombe, M., Halsall, J.: Robotics in nuclear engineering: Computer-assisted teleoperation in hazardous environments with particular reference to radiation fields, vol. 9312. Graham & Trotman (1984)

2. Neumann, P.P., Hernandez Bennetts, V., Lilienthal, A.J., Bartholmai, M., Schiller, J.H.: Gas source localization with a micro-drone using bio-inspired and particle filter-based algorithms. Advanced Robotics 27(9), 725–738 (2013)
3. Kowadlo, G., Russell, R.A.: Robot odor localization: A taxonomy and survey. The International Journal of Robotics Research 27(8), 869–894 (2008)
4. Ishida, H., Tanaka, H., Taniguchi, H., Moriizumi, T.: Mobile robot navigation using vision and olfaction to search for a gas/odor source. Autonomous Robots 20(3), 231–238 (2006)
5. Cabrita, G., Sousa, P., Marques, L.: Odor guided exploration and plume tracking: Particle plume explorer. In: Proceedings of the 5th European Conference on Mobile Robots (ECMR 2011), pp. 183–188 (2011)
6. Lilienthal, A., Duckett, T.: A stereo electronic nose for a mobile inspection robot. In: Proceedings of the IEEE International Workshop on Robotic Sensing (ROSE 2003) (2003)
7. Loutfi, A., Coradeschi, S., Lilienthal, A., Gonzalez, J.: Gas distribution mapping of multiple odour sources using a mobile robot. Robotica 27, 311–319 (2009)
8. Ramirez, A., Lopez, A., Rodriguez, A., de Albornoz, A., De Pieri, E.: An infotaxis based odor navigation approach. In: Biosignals and Biorobotics Conference (BRC). ISSNIP, pp. 1–6 (2011)
9. Villarreal, B.L., Gordillo, J.L.: 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), pp. 334–339 (August 2013)
10. Villarreal, B.L., Hassard, C., Gordillo, J.L.: Finding the direction of an odor source by using biologically inspired smell system. In: Pavón, J., Duque-Méndez, N.D., Fuentes-Fernández, R. (eds.) IBERAMIA 2012. LNCS, vol. 7637, pp. 551–560. Springer, Heidelberg (2012)
11. Vickers, N.J.: Mechanisms of animal navigation in odor plumes. The Biological Bulletin 198(2), 203–212 (2000)
12. Monroy, J.G., González-Jiménez, J., Blanco, J.L.: Overcoming the slow recovery of mox gas sensors through a system modeling approach. Sensors 12(10), 13664–13680 (2012)
13. Martinez, D., Rochel, O., Hugues, E.: A biomimetic robot for tracking specific odors in turbulent plumes. Autonomous Robots 20(3), 185–195 (2006)
14. Lochmatter, T., Raemy, X., Martinoli, A.: Odor source localization with mobile robots. Bulletin of the Swiss Society for Automatic Control 46, 11–14 (2007)
15. Villarreal, B.L., Gordillo, J.L.: Method and artificial olfactive system (metodo y sistema olfativo artificial). Mexico. Instituto Mexicano de la Propiedad Intelectual (IMPI). Patent pending Mx/a/2012/014 508 (2012)
16. Crank, J.: The mathematics of diffusion. Oxford Univerity Press (1976)
17. Rozas, R., Morales, J., Vega, D.: Artificial smell detection for robotic navigation. In: 5th International Conference on Advanced Robotics (ICAR). Robots in Unstructured Environments, vol. 2, pp. 1730–1733 (1991)
18. Wilensky, U.: Netlogo. Center for connected learning and computer-based modeling. Northwestern University, Evanston (1999), `http://ccl.northwestern.edu/netlogo/`
19. Paic-Antunovic, L., Jakobovic, D.: Evolution of automatic robot control with genetic programming. In: 2012 Proceedings of the 35th International Convention, MIPRO, pp. 817–822. IEEE (2012)
20. Lazarus, C., Hu, H.: Using genetic programming to evolve robot behaviours. In: Proceedings of the 3rd British Conference on Autonomous Mobile Robotics and Autonomous Systems, Manchester (2001)