Autonomous Navigation of a Lego Robot with Neural Networks

Salvador Ibarra M., José A. Castán R., Julio Laria M., Javier Guzmán O., Mayra G. Terviño B., Julissa Pérez C. and Emilio Castán R.

***Abstract*— The paper presents an autonomous navigation model to an unmanned vehicle using neural networks. In particular the Lego NXT robot is used to create a vehicle capable of navigating in an autonomous way based on information providing for an ultrasonic sensor. Such information is getting into a training neural network to decide the maneuver that robot must do to achieve the required task. For the scope of this research work, two neural network approaches have been developed. In order to perform the training phase, a specific simulator was generated in MatLab language. Some experiments were executed to corroborate the performance of the two neural networks and the extrapolation of their models to the Lego robot development language (Bricx Command Center). Finally, from de obtained results, some conclusions are discussed.**

***Index Terms*— Lego NXT robot, neural network, autonomous navigation strategy.**

I. Introduction

THE Lego NXT robot is a technological device widely used by professors, to begin students in the paradigms of the robotics world. Although the Lego robot might be "confused" with a simple toy, this robot is a very advanced device because not only is based on the Lego philosophy *"constructs whatever you imagination tell you";* but it allows users to develop an empathy with the device. In this sense, the Lego robot has been used effectively in the Institute of Computer Images and Computer at the RWTH Aachen University, where students begin to work with this devices from his first semester, developing programs where they put to test their MatLab language skills for digital image processing. After this, students are able to develop solutions for more specific and advanced problems, which should solve from the Lego platform.

Reference [1] describes the educational benefits gained through conducting practice with the Lego robot on issues such as Artificial Intelligence, Reactive Control, Heuristic Search, Predicate Logic and Reinforcement Learning among others. The authors highlight the enthusiasm of the students to accept the development of programming projects using the Lego robot.

For example in [2], author suggests that nowadays, teachers are using a wide range of models supporting students with different learning skills, can understand the subject of their classes. To keep the interest of the students, the computer courses consider the development of programs and applications using Lego robots. The implementation over these devices has been a key aspect to promote the programming, artificial intelligence and automation research areas.

According to this, the use of a Lego robot in navigation methodologies, is an interesting proposal to put in practice the programming skills as well as the adaptation of artificial intelligence models to increase the cognitive skills and decision-making structures in a totally autonomous and intelligence entity. To do this, neural networks arise as a proper solution to endow autonomous entities with the capability to take decisions as the human beings.

Methods for collaborative decision-making in multi-agent systems are, in most cases, intense applications with sophisticated software algorithms using advanced computer technologies. In spite of this, important theoretical aspects on cooperation have been untreatable [3]. In this sense, the advances of the artificial intelligence community in path planning, adaptation, learning, fuzzy logic and knowledge representation together with other technics like control theory [4-5], social metaphors [6-7] or bidding models [8], represents a fresh path to a nearest progress.

In this way, some researches tend to perform complex control and cooperation using agents. Agents are defined as intelligent systems capable of flexible and autonomous actions in a dynamic, unpredictable and typically cooperative scenario [9]. In this sense, some results have been obtained for coordinated actions using agent technology [3]. Recently, an analysis of the literature [3], [10-12] shows that for more complex tasks (p. Eg., Robotic soccer, rescue missions, etc), is required a reliable performance as well as high level of robustness in dynamics, unpredictable and hazardous environments.

At the end, it is possible to argue that a robot able to reason about what information uses to make a decision, such robot will improve its individual decisions, which impact in a positive way to the correct development of a global goal.

II. Definition of The Neural Networks Models

The main goal of neural networks is endow to any entity with the capability of processing information in order to act in a reliable way, considering the requirement and constrains involving in the decisions itself. In this sense, two neural networks model have been used: Backpropagation BPNN

Manuscript received March 23, 2015; revised April 08, 2015.

S. Ibarra, J. A. Castán, J. Laria, J. Guzmán, M. G. Treviño and J. Pérez are members of the Computer Technology Group at the Engineering School at the Autonomous University of Tamaulipas, Mexico. (sibarram@uat.edu.mx).
E. Castán is professor at the Electric and Electronic Department at the Madero Institute of Technology.

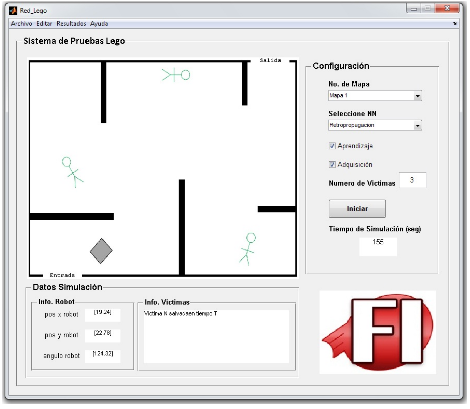
.

and Counterpropagation CPNN [13-14]. Backpropagation is an algorithm of supervised learning used to training artificial neural networks. This algorithm of minimizing the squared error using gradient descent, so that the essential part of the algorithm is calculating the partial derivatives of the error with respect to the parameters of the neural network.

In BPNN is necessary to clarify the concept of propagation: this network interconnects multiple layered perceptrons, but not each layer perceptrons are interconnected. However, each layer perceptron provides an input to each of the next layer perceptrons. This means that each neuron transmit its output signal to each neuron in the next layer, which is the fundamental architecture of a network backpropagation.

Otherwise, in CPNN several layered perceptrons are interconnected so that five layers are formed: two input layers, two output layers and one hidden layer. In the input layer, it is performed a pre-processing to prepare the input data for the calculus. This procedure solves problem like the saturation of the neural activities in front of greater quantity of input signals. In the hidden layer of the neural network, this layer is composed of a se of elements known as instars process. This process allows a neuron to learn by a new vector by adjusting the equivalent weights of the inputs at the moment, with a proportional rate to the output.

Due to the poor features of the Lego NXT developer suite, which it is not capable of supporting the implementation of neural networks, a simple but reliable simulator was developed in MatLab. In this sense, with MatLab and their libraries GUIDE and SIMULINK was possible to develop the testbed for the Lego robot (see Fig. 1). This simulator allows representing the physical characteristics of the robot's body and their capabilities (linear and angular velocities).



No. de Mapa: Mapa 1
Seleccione NN: Retropropagacion
☑ Aprendizaje
☑ Adquisición
Numero de Victimas: 3
Tiempo de Simulación (seg): 155
pos x robot: [19.24]
pos y robot: [22.78]
angulo robot: [124.32]
Victima N salvadaen tiempo T

Fig. 1. Lego simulator on MatLab

The simulator facilitates different configuration according to:

- Type of map.
- Selection of a neural network model.
- # of victims.
- Position of the robot.
- Information about the victims.
- Learning.
- Data acquisition.
- Time of simulation.

The general navigation strategy of the robot is very simply and, is a response directly obtained from the ultrasonic sensor. In fact, the robot starts all the simulation with a frontal movement until it detects information grounded on the environment (e.i., victims or object). In particular, the robot reacts to the sensor information according to the selected model of neural network.

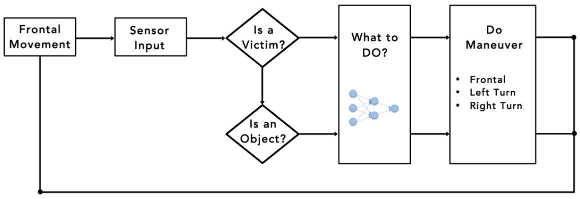


Fig. 2. . General scheme of the robot's movement strategy.

To validate the performance of the simulator and the neural network models, a set of 5 scenarios with different configuration were designed. In particular, for each scenario 10,000 trials were executed. The preliminary results generated from such proves are presented in Table 1. The robot is aware about the number of victims defined at each experiment; however, the robot does not know where are the victims. Besides, the robots cannot leave the map without having "rescued" all the victims. A victim is rescued when the robot is positioned at their coordinates. According with the presented results, the BPNN appears to be the best option for the scope of this research. The improvement rate of each scenario is calculated by using the distance between the execution time of each model.

Table 1. Preliminary results of the validation phase.

Scenario	NN	Victims	Execution Time (sec)	Improvement rate (%)	Selected NN
1	BPNN	1	57	29.62%	BPNN
	CPNN		81		
2	BPNN	2	115	19.58%	BPNN
	CPNN		143		
3	BPNN	3	187	18.34%	BPNN
	CPNN		229		
4	BPNN	4	261	16.34%	BPNN
	CPNN		312		
5	BPNN	5	337	33.13%	BPNN
	CPNN		504		

III. Implementation on the Lego NXT Robot

The implementation of the BPNN in Bricx Command Center lets put into practice the abilities of the Lego robot. In this sense, two controlled scenarios are proposed (see Figs. 3 and 4) where the Lego robot must avoid crashing with the obstacles (represented by some boxes and other things) and victims are represented by green figures. Both scenario are totally unknown to the robot, and can be reconfigured by the robot itself, because the robot can move some of the objects involuntarily. The robot begins each experiment from a determined point. Another interesting aspect is related with environmental lights. The robot is equipped with a color sensor. With this device the robot can look for the victim. The use of this sensor is a new challenge because the degree of the color recognition suffered a loss of quality under some kind of light or when the robot moves so fast. To solve this, it was needed to put a patch to avoid the direct exposition of the sensor with external light.

The ultrasonic sensor was set at 10 cm in relation with the basis of the scenario. This distance is to facilitate the identification of the objects over the environment.

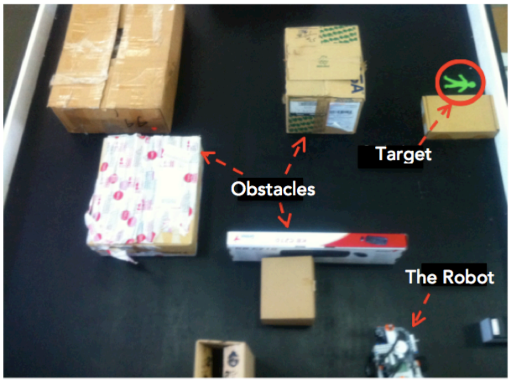


Fig. 3. Real scenario 1.

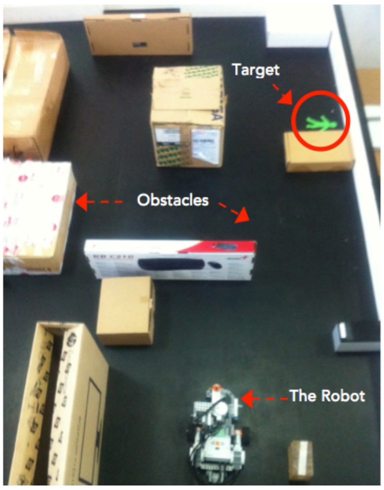


Fig. 4. Real scenario 2.

IV. Experimental Results

The experiments were developed over the scenarios aforementioned. Some of the aspects to evaluate the results are the following:

ci_{r_i,v_j} certainty index of the robot i to rescue the victim j.

$time_{r_i,v_j}$ time needed for the robot i to rescue the victim j.

$error_{r_i,v_j}$ errors of the robot i when it tray to rescue the victim j.

In this sense, an error is computed when the robot states that has made a rescue but it is not over the victim. These errors can occur for different situation such as: the low level of the batteries of the robot or the color sensor cannot do a well lecture.

In addition, the performance of the robot i to solve a particular victim j is calculated by the eq. (1).

$$ci_{r_i,v_j}=\sum_{V_j=1}^{n}\left(\frac{time_{r_i,v_j}+error_{r_i,v_j}}{time_{r_i,v_j}*error_{r_i,v_j}}\right)*V_j \qquad (1)$$

The experiments are devoted to illustrate the effectiveness of the proposed BPNN against a base model generated by a random algorithm in Lego Mindstorm [15]. For each scenario was performed a set of 10 experiments with different number of victims. In each case, the experiment computes the time that the robot needs to rescue the stated victims, the number of errors and the improvement rate IR between the BPNN and the base model using the simple eq. (1).

$$IR=100-\left(\frac{\text{worst case}}{\text{best case}}*100\right) \qquad (2)$$

In this sense, the Table 2 presents the results of the first scenario. The results clearly show how the performance of the robot improves when it is able to make decision based on a reasoning process like neural networks. In short, the BPNN demonstrates to be a useful approach for the scope of this research. In addition, Table 3 introduces the results of the second scenario. These results confirm the proper performance of the BPNN approach proposed in this paper.

Table 2. Experimental results of the scenario 1.

Case	# Victims	Time		# errors		CI_{BASE}	CI_{BPNN}	CI_{BASE} vs CI_{BPNN}
		Base	BPNN	Base	BPNN			
1	1	5	4	5	3	0.4000	0.5833	**31.42**
2	1	7	3	7	2	0.2857	0.8333	**65.71**
3	1	6	4	9	2	0.2777	0.7500	**62.97**
4	2	12	10	12	3	0.3333	0.8666	**61.53**
5	2	14	9	13	4	0.2967	0.7222	**58.91**
6	2	15	8	11	3	0.3151	0.9166	**65.62**
7	3	20	18	18	6	0.3166	0.6667	**52.51**
8	3	19	15	20	5	0.3078	0.8000	**61.52**
9	3	22	15	20	6	0.2863	0.7000	**59.10**
10	5	54	28	29	11	0.2650	0.6331	**58.14**

Table 3. Experimental results of the scenario 2.

Case	# Victims	Time		# errors		CI_{BASE}	CI_{BPNN}	CI_{BASE} vs CI_{BPNN}
		Base	BPNN	Base	BPNN			
1	1	8	6	2	2	0.6250	0.6667	**6.25**
2	1	9	6	2	1	0.6111	0.6667	**8.33**
3	1	8	5	3	1	0.4583	0.7000	**34.52**
4	2	21	15	7	4	0.3810	0.6333	**39.85**
5	2	19	17	6	4	0.4386	0.6176	**28.99**
6	2	22	14	7	2	0.3766	0.8095	**53.48**
7	3	29	28	18	5	0.2701	0.7071	**61.80**
8	3	34	29	15	4	0.2882	0.8534	**66.23**
9	3	31	28	12	4	0.3468	0.8571	**59.54**
10	5	47	39	24	6	0.3147	0.9615	**67.27**

V. Conclusions

A decision process is a routine that human beings must do in daily life. People must decide what food to eat and what clothes to wear at every day and in most cases, these decisions are reached unconsciously. It means, that some decisions are reached without using a high degree of mental complexity. In this sense, the demands of these days require new technologies that support human beings to make sure and trustworthy decisions in different scenarios. Over time, different efforts have been performed in the artificial intelligence research field, to develop theories, techniques and systems to study and understand the behavior and reasoning skills of a virtual intelligent entity (i.e., an intelligent robot). Such works have showed significant advances, aiming to attack complex, dynamical and typically open environments. However, the quality and quantity of the information about the problem and its perspective of the whole scenario will limit the robot skills, specially when this robot must work in a real environment. In this sense, a cognitive model must be capable of providing robots with the appropriate structure to make effective and trustworthy decisions.

In this light, the paper promotes the use of neural networks to endow autonomous entities with reasoning skills, able to facilitate the complex decision-making process. In particular two neural network models were studied: backpropagation neural network and counterpropagation neural network. To validate the operatively of the NN model a simulator on MatLab was developed. Some proofs were executed using the developed system. The preliminary results show how the BPNN is a model more adequate for the scope of this work. Moreover, using the Lego NXT robot, another set of experiments was developed. In this case, the BPNN model was compared against a base model generated in Lego Mindstorm in 10 cases for two different scenarios. To the end, it is possible to argue that the Lego NXT robot is a technological device that can be used not only as a toy, but can be a powerful tool for research corroborating important practical aspects.

ACKNOWLEDGMENT

This work was developed by professors of the Computer Technology Group of the Engineering School at the Autonomous University of Tamaulipas, Mexico.

REFERENCES

[1] Parson S. y Sklar E. " Teaching AI using LEGO Mindstorm", Deparment of Computer and Information Science, Brooklyn Collage, City University of New York, Ed. American Association for Artificial Intelligence (www.aaai.org), 2008.

[2] Schafer J. B., "Hands-on Artificial Inteligence Education Using LEGO Mindstorm^{TM}: Lesson Learned", Department of Computer Science, University of Northern Iowa, 2009.

[3] L.E. Parker, "Distributed Intelligence: Overview of the Field and its Applications in Multi-Robot Systems", In Journal of Physical Agents, vol. 2, issue 1, pp. 5-14, March, 2008.

[4] R. Murray, K.J. Astrom, S. Boyd, R.W. Brockett and G. Stein. "Future Directions in Control in an Information-Rich World", In *IEEE Control Systems Magazine*, vol. 23, no. 2, pp. 20-33, April, 2003.

[5] R. Sanz, O. Holland, A. Sloman, A. Kirilyuk, W. Edmondson and S. Torrance. "Self-aware Control Systems", In *Research Whitepaper for the Bio-inspired Intelligent Information Systems Call*, 2004.

[6] M. Esteva, D. de la Cruz and C. Sierra. "ISLANDER: an electronic institutions editor", In Proc. of the *Autonomous Agents and Multiagent Systems (AAMAS'02),* ACM 1-58113-480-0/02/2007, Bologna, Italy, 2002.

[7] J. A. Rodríguez-Aguilar. "On the design and construction of Agent-mediated Electronic-Institutions", PhD Thesis, Artificial Intelligent Research Institute, Universitat Autònoma de Barcelona, 2001.

[8] D. Busquets, R. López de Màntaras, C. Sierra, T. G. Dietterich. "A multi-agent architecture integrating learning and fuzzy techniques for landmark-based robot navigation", In *Lecture Notes in Computer Science*, ISBN: 3-540-00011-9, vol. 2504, pp. 269-281, 2002.

[9] M. Luck, P. McBurne, O. Shehory and S. Willmott. "Agent Technology: Computing as Interaction", In A Roadmap for Agent Based Computing, Compiled, written and edited by M. Luck, P. McBurney, O. Shehory, S. Willmott and the AgentLink Community, pp. 11-12, 2005.

[10] T. Fang, L.E. Parker, "A Complete Methodology for Generating Multi-Robot Task Solutions using ASyMTRe-D and Market-Based Task Allocation", In Proc. of the IEEE International Conference on Robotics and Automation, pp. 3351-3358, April 2007.

[11] Jeong-Ki Yoo, Yong-Duk Kim, Bum-Joo Lee, In-Won Park, N.S. Kuppuswamy, Jong-Hwan Kim, **"**Hybrid Architecture for Kick Motion of Small-sized Humanoid Robot, HanSaRam-VI", In Proc. of the International Joint Conference SICE-ICASE'06, pp. 1174-1179, October, 2006.

[12] B. Duffy. "Robots Social Embodiment in Autonomous Mobile Robotics", In *Journal of Advanced Robotic Systems*, vol. 1, no. 3, pp. 155-170, ISSN 1729-8806.

[13] Padern, Xavier , Introducción a las Redes Neuronales, 2000.

[14] Edgar Nelson Sánchez Camperos, Alma Yolanda Alanís García, Redes Neuronales conceptos fundamentales y aplicaciones a control automático, 2006.

[15] Lego Mindstorm. http://www.lego.com/en-us/mindstorms/?domainredir=mindstorms.lego.com.