We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

18

Action Selection and Obstacle Avoidance using Ultrasonic and Infrared Sensors

Fernando Montes-González, Daniel Flandes-Eusebio and Luis Pellegrin-Zazueta
Departamento de Inteligencia Artificial, Universidad Veracruzana
México

1. Introduction

The use of genetic algorithms in behavior based robotics is employed to adjust the parameters of the behavioral patterns that the robot executes. In order to model these patterns, Khepera robots can be used as one of many options. Therefore, in this paper we are proposing to expand the basic sensing capabilities of the robot by the development of a rapid ultrasonic detection turret for the Khepera II robot. The use of this turret favors the detection of different kinds of objects. Our purpose is to simulate the kind of behavior that a robot encounters in the navigation of human environments. For example, objects like short tables or desks standing on their legs above the floor. The standard Khepera II robot is equipped with eight infrared sensors, thus we designed an ultrasonic turret with three more sensors in order to grant the robot with detection capabilities for objects with short legs. Later on, we developed an obstacle avoidance behavior with the use of genetic algorithms. Therefore, in section 2 a brief background on genetic algorithms is provided. In section 3, we discuss some topics related to ultrasonic sensing, and we introduce the integration of these sensors into the Khepera. Section 4 describes the employed parameters for the use of the genetic algorithm and the robot equipped with ultrasonic and infrared sensing capabilities. The description of some experiments for testing the setup of the robot is explained in Section 5. Then, in section 6 we explore the integration of the extended capabilities of the Khepera within a foraging task with Action Selection. Finally, we provide a general conclusion in section 7.

2. Evolutionary Robotics and Genetics Algorithms

In writing several examples of solutions have been provided for the development of robot behavior. Commonly, the implementation of a particular behavior is carried out once the experimental setup is established. For example, robots can be set in a semi-structured environment where they solve particular tasks. Take the work of Bajaj and Ang Jr. for instance (Bajaj & Ang, 2000), where the *standard* Khepera has to solve a maze by avoiding obstacles and following walls. In the mentioned work and similar works, the use of Genetic Algorithms (Holland, 1975) is preferred over existent evolutionary methods like: Evolutionary Strategies, Genetic and Evolutionary Programming and Co-evolution. The

Open Access Database www.intehweb.com

Source: Frontiers in Evolutionary Robotics, Book edited by: Hitoshi Iba, ISBN 978-3-902613-19-6, pp. 596, April 2008, I-Tech Education and Publishing, Vienna, Austria

development of a basic genetic algorithm is a solid approach for starting to work on evolutionary robotics. Therefore, in our paper we chose the use of this method for tuning an obstacle avoidance behavior. It is important to notice that the resultant behavior, which is shaped and nearly optimized by the use of the genetic algorithm, ultimately depends on the right choice of the fitness function. Next we provide a brief background on genetic algorithms to support the development of our work.

The use of genetic algorithms and neural networks (Nolfi & Floreano, 2000) offers a good solution to the problem of modeling an obstacle-avoidance behavior in maze-like environments. Neural controllers require the setup of a chosen topology, and this can be done by the use of some rules of thumb. Once the topology is decided the weights of the neural controller have to be configured. A common approach relies on the use of backpropagation training that is a form of supervised learning where the neural net has to learn a known response to a particular configuration of the sensors in the robot. The general misclassification error is calculated and decreased over time when the neural network is trained. However, this kind of learning requires the design of training and validation data. On the other hand, the use of genetic algorithms is a form of gradient ascent approach that refines at each step of the optimization the quality of initial random solutions.

The optimization of neural controllers with genetic algorithms requires the representation, as a vector, of the weights of the neural controller. Then, a common practice consists of a direct encoding of the neural weights as an array that represents the genetic material to be manipulated by artificial evolution. A single neural controller represents one of the many individuals that form a population, which in turn are candidates for providing a good solution to the task that is to be solved. On the other hand, the fittest individuals of one population are used to breed the children that will be evaluated in the next generation. Therefore, the quality of a solution ('fitness') is measured to acknowledge whether a candidate solution is or not a good solution to the behavior we are trying to model. If the fitness of all candidate solutions is plotted, we will end up with a convoluted space where all possible fitness solutions can be represented. Therefore, this fitness landscape is formed by mountains and valleys, where landmarks in the mountains represent good quality solutions and landmarks near valleys are poor solutions.

The search of the best solution within a fitness landscape requires the guidance of the genetic algorithm to move uphill to find improved solutions. Nevertheless, a few downhill steps may be necessary in order to climb to the highest mountain. Therefore, exploration is guided by the use of a fitness formula that defines the behavior to be shaped, and three genetic operators are employed to create new solutions from existent ones. Thus, the current evaluated population spawns a new generation by the *selection* of a subset of the best individuals, the reproduction of the best individuals in pairs by the *crossover* of their genetic material, and the *mutation* of some of the material genetic of the individuals in the new population. The application of these operators to an initial random population of weights will produce refined solutions over time, and then the fitness evaluation will shape the final behavior through the breeding of the fittest individuals. However, few iterations are needed before this occurs.

3. Ultrasonic Sensing and its Integration with the Khepera Robot

The standard capabilities of the Khepera have already been extended in other works. However few are related to the use of ultrasonic sensors. Take for instance the work of

Chapman, et al. (Chapman, et al., 2000) where a wind-sensor is built for solving a maze. The work of Webb, et al. (Webb, et al., 2005) provided ears to a Khepera in order to simulate a female cricket. Furthermore, Odenbach, et al. (Odenbach, et al., 1999) fitted a wireless communication system on the Khepera to allow the identification of another Khepera in contrast to surrounding obstacles. Böndel, et al. (Böndel, et al., 1999) extends the Khepera to pick up small holed cubes. Another extension of the Khepera by Goerke, et al. (Goerke, et al., 1999) allows the robot to play golf. In contrast, the work of Winge (Winge, 2004) is the closest work to the one presented here. However, he makes use of the SRF04 sonar[1] and a major inconvenience is the programming of the Khepera microcontroller to estimate the measured distance to the objects.

In our work we are extending the sensing capabilities of the Khepera robot (Mondana, et al., 1993) with the use of a rapid ultrasonic detection turret; thus, we introduce how these sensors work. They use a non-audible pulse of 40 KHz, which travels through the air. When an object is close to the transmitter-receptor of the ultrasonic sensor, sound waves bounce back from that object and this bounce-back of the sound is then detected by the ultrasonic sensor. An elegant solution for the measurement of distance using ultrasonic sensors consists on the calculation of the time-estimation of the bounce-backed sound to the receptor. In order to calculate distance, we assume that the speed of the sound in the air is already known, and can be calculated using the rule of three from the next formula

$$D = v_s \cdot \frac{t_{propagation}}{2} \tag{1}$$

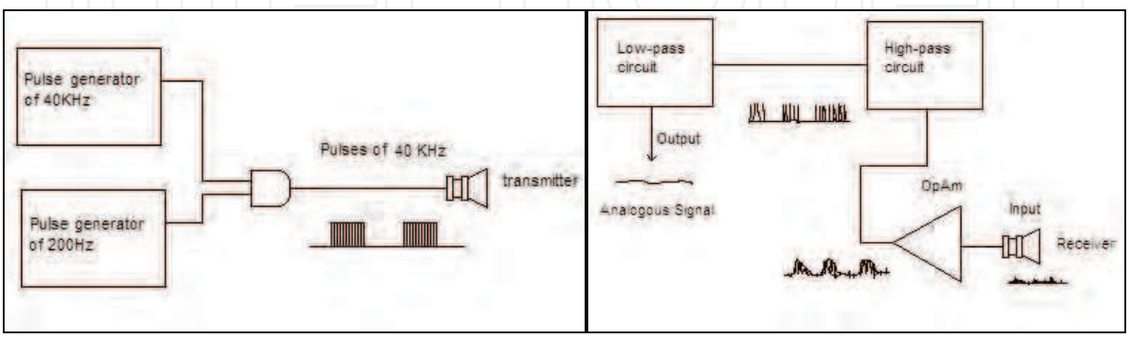
where: v_s = 340[m/s] is the propagation speed of the acoustic waves in the air; and $t_{propagation}$[s] is the total propagation delay of the acoustic waves.

However, by following an approach such as this requires the use of an analogue-digital converter to transform the resultant data into a digital format that can be processed by the Khepera's microcontroller. An alternative method calculates the distance from the measured intensity of the bounced-back sound into the receptor. As a result, we obtain an analogue signal that can be passed to the microcontroller of the Khepera. The latter method is less accurate, though is cheap and requires a reduced amount of electronic components. The measurement of the bounced-back sound is a rapid solution that makes a popular choice for prototype development. Thus, for the design of a rapid ultrasonic detection system we measure the intensity of the sound that bounces-back from objects in the near range.

The General I/O (*Gen I/O*) extension turret of the Khepera is widely used to expand the standard capabilities of this robot. The Gen I/O for these purposes offers 8 digital inputs; 2 analog inputs with adjustable gain; 1 analog differential input; 4 digital low power outputs; 1 digital high-power output; and 1 motor control (full H Bridge). Therefore, we will fit three ultrasonic sensors on the top of a Khepera robot in order to grant the robot with the capability of detecting farther short-legged objects than the infrared sensors can detect. The use of the Gen I/O permits the connection of two of the three ultrasonic sensors to the analog inputs with adjustable gain, and the third one to the analog differential input. The three sonar inputs are transformed by the analog convert of the robot into data that can be

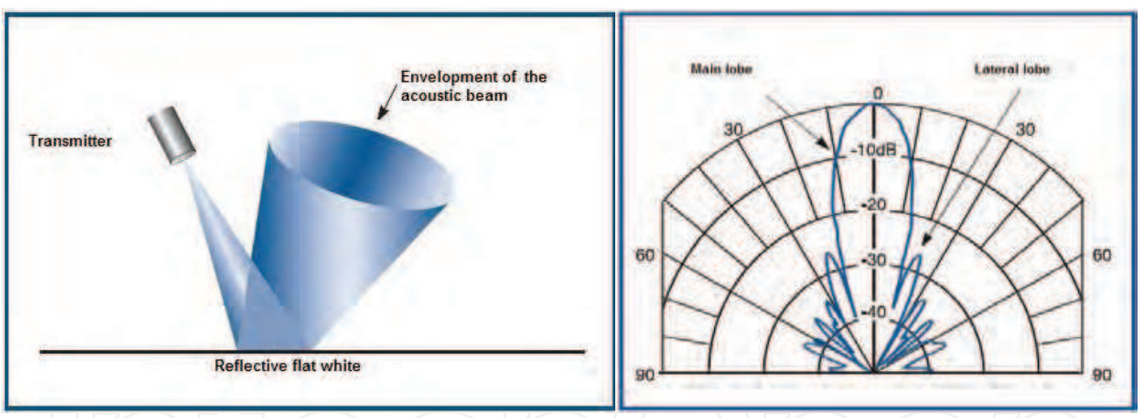
[1] A modular sonar fitted with a transmitter and a receiver that employs input signal-conditioners, and the output of a digital pulse signal, to measure distance using external logic devices.

interpreted by the microprocessor of the Khepera. Then, for the construction of the ultrasonic turret, we should take into account that the distance calculated from detected obstacles should be converted into an analog value ranging from 0 to 5 Volts. The interpretation of this digital value is dependent of the application to be implemented. In our case, we scale this range to 0-1024 values for the readings of detected obstacles. Next, the ultrasonic turret is powered from the robot with a source of 5 volts with a maximum electrical current of 250 mA. This value of the electrical current is enough for the analog circuit on the extension board to work. The design of the transmitter, shown in two blocks, is presented in Figure 1(a).



a) Transmitter b) Receptor

Figure 1. Block diagram of the ultrasonic transmitter/receptor



a) Envelopment [Airmar, 2006] b) Degradation [Airmar, 2006]

Figure 2. Behavior of the acoustic beam

The transmitter works in the follow manner, a pair of pulse generators are employed, one generator outputs pulses of 40 KHz and the second produces a 200 Hz signal. Then, both generators produce a train of pulses. The pulse train uses a carrier signal of 40 KHz. The implementation of the ultrasonic transmitter makes use of regular oscillator circuits like the LM555, and a couple of standard digital gates to generate the pulse train. Next, standard inverse-gates reduce the width of the pulses and the signal is then feed into the ultrasonic transmitter. The receptor circuit is presented in Figure 1(b), and for its implementation we use an OpAm (Operational Amplifier) such as the LM324 that facilitates the use of non-symmetrical sources (*GND and +5V*).

The transmitter is able to reach objects within a scope defined by a 10 degree angle scope (Fig. 2(a)); thus allowing a degradation of 3dB when the object is out of range (Fig. 2(b)).

This degradation provides a good estimation in accuracy of the measured distance of objects within sight and out of range.

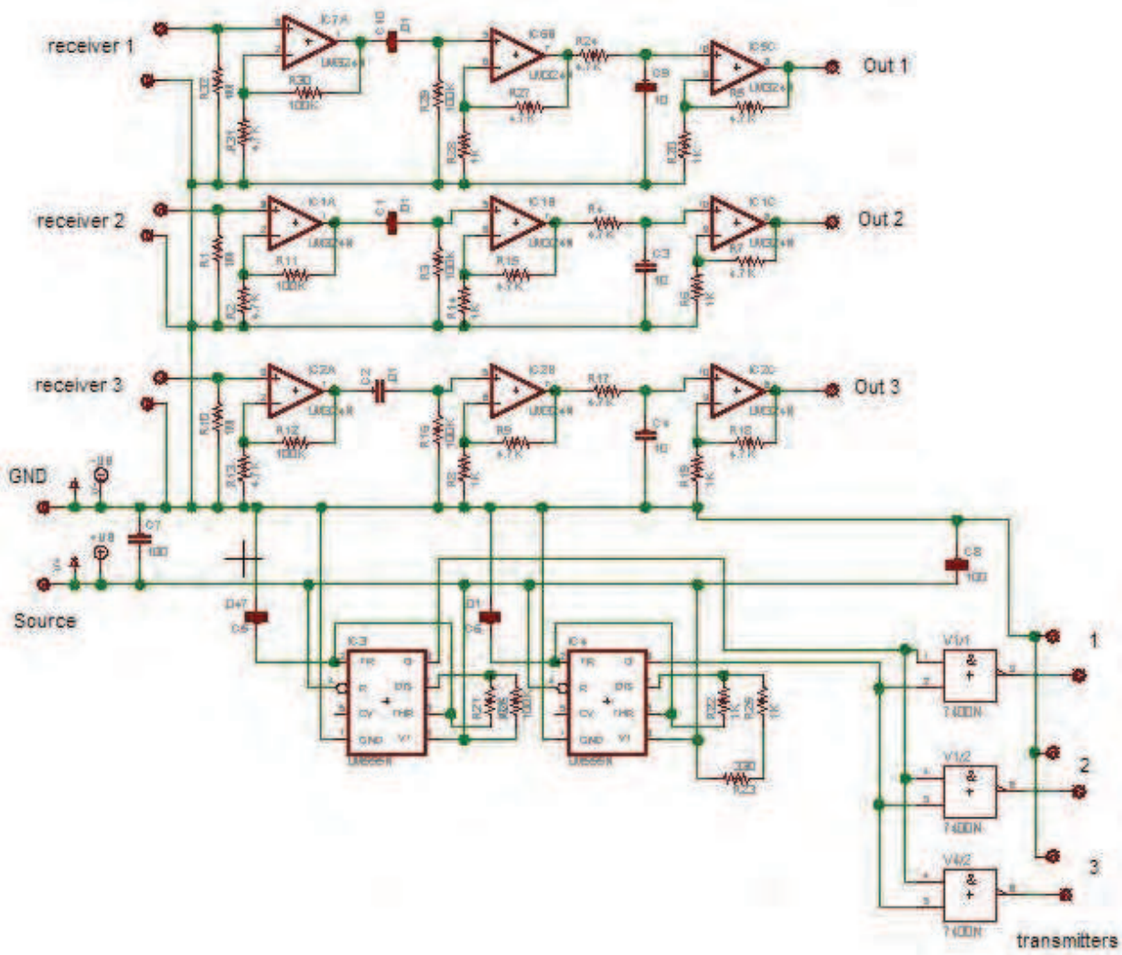
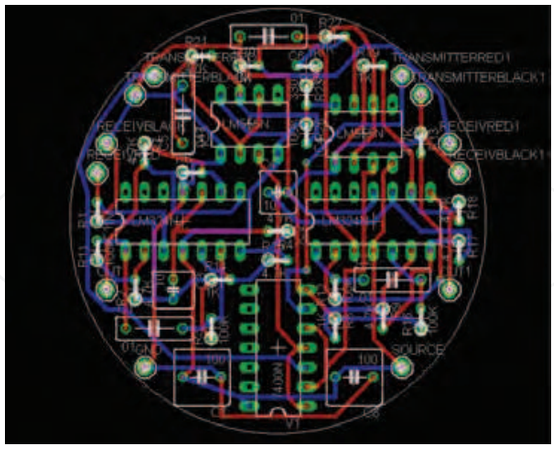


Figure 3. Schematics of the extension circuit of the ultrasonic sensors

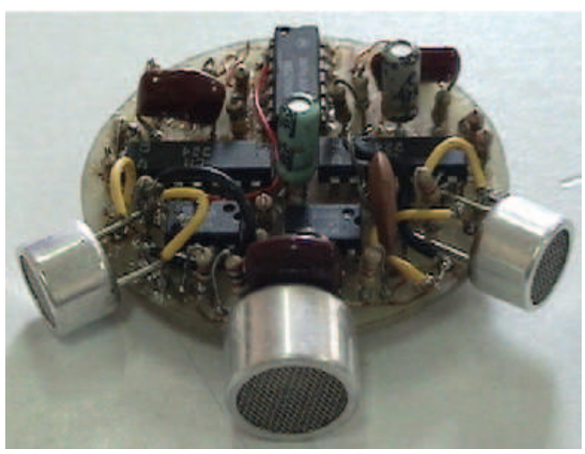
The receptor circuit is made of three amplified phases: the first one amplifies the signal, within the frequency response range of the OpAm, at a maximum gain of 21 Volts/Volts; the second phase is a high-pass filter that cuts low frequencies down to 40 KHz and boosts the carrier signal by a factor of 4.7, which is enough to detect an increase in the proximity of detected objects; finally, the third phase is a low-pass filter that eliminates the carrier signal of 40 KHz to obtain a variation of the object-proximity signal within a range of 0 to 5 Volts that provides a good accuracy in the detection of obstacles within a maximum range of 20 centimeters. The ultrasonic sensor has a response-frequency that facilitates the detection of the 40 KHz carrier signal; therefore, noise from the environment is ignored. The construction of the extension board required that the design was implemented onto a single physical board (Fig. 3).

Due to the small size of the components of the circuit board; all the circuits were fitted into a small circular area similar to that occupied by the Gen I/O turret (Fig. 4(a)). Therefore, the three ultrasonic sensors were fitted onto a single board (Fig. 4(b)) on top of the Khepera. The intensity in the readings of the ultrasonic sensors can be adjusted by the gains in the

amplifiers of the Gen I/O. Therefore, a calibration of these gains is made to fine-tune the ultrasonic perception of the robot.



a) Printed Circuit Board (PCB)



b) Ultrasonic Turret Board

Figure 4. Implementation of the ultrasonic turret board

4. The Use of Genetic Algorithms with Ultrasonic Sensing

The development of the obstacle-avoidance behavior required the selection of a neural network topology. Thus, a fully connected feedforward multilayer perceptron neural network with no recurrent connections was employed for the development of the avoidance behavior. The input layer in this network consists of 7 neurons that are connected to a hidden layer of 6 neurons. In turn, these hidden neurons send projections to the 2 neurons in the output layer. The six frontal infrared sensor-readings of the Khepera range from nothing detected (≈ 0) to something very close (≈ 1023), and in a similar way the three sonar sensor readings go from 0 to 1024. Then, a vector of 7 sensors is formed, **sensors** = [IR_1, IR_2, $max(IR_3, SNR_1)$, SNR_2, $max(IR_4, SNR_3)$, IR_5, IR_6] with *IR* for the infrared sensors, and *SNR* for the sonars. As can be noticed from the equation for some elements of the vector we employ the most relevant information either from the infrared or the sonar sensors. Therefore, we are integrating the information from both the IR and the SNR sensors in one single vector.

Then, this vector is passed to the neural network, which is optimized by the genetic algorithm. Finally, the output of the network is scaled to ±20 values for the DC motors.
The weights of the neural network are transcribed using a direct encoding to the vector **c** of 54 elements. Next, random initial values are generated for this vector $\mathbf{c}_i$, $-1 < \mathbf{c}_i < 1$. The initial population G_0 is composed of $n = 100$ neural controllers. The two best individuals of a generation are copied as a form of *elitism*. *Tournament selection*, for each of the $(n/2)$-1 local competitions, produces two parents for breeding a new individual using a single random *crossover* point with a probability of 0.5. The new offspring is affected with a *mutation* probability of 0.01. Individuals of the new offspring are evaluated for about 7 seconds in the robot simulator. The fitness plot for this behavior is shown in Figure 5.

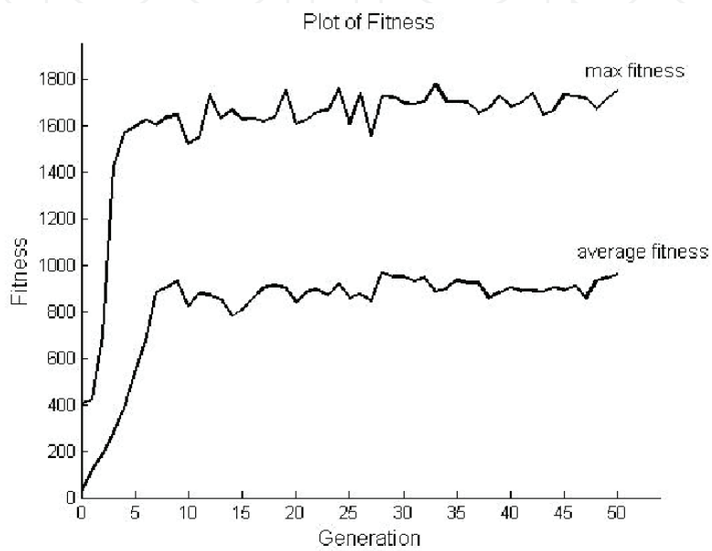


Figure 5. The fitness for the avoidance behavior is plotted over 50 generations

The fitness formula for the obstacle behavior was

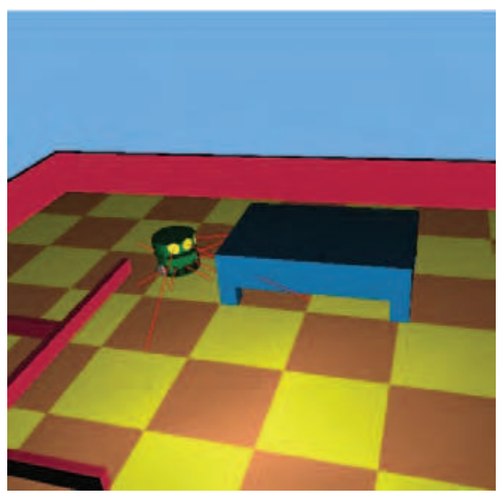
$$f_q = \sum_{i=0}^{3500} abs(ls_i)(1-\sqrt{ds_i})(1-\max_sensor_i) \tag{2}$$

where for iteration i: *ls* is the linear speed in both wheels (the absolute value of the sum of the left and right speeds), *ds* is the differential speed on both wheels (a measurement of the angular speed), and *max_sensor* is the maximum sensor value in the input vector. The use of a fitness formula like this rewards those fastest individuals who travel on a straight line while avoiding obstacles.
In our work we use a fitness formula similar to those of Floreano and Mondana (Floreano & Mondana, 1994) and Montes-Gonzalez (Montes, 2007). In a formula such as this the fitness is measured based on motor speeds and the obstruction of object-detection sensors. Therefore, an increase of the sensory capabilities of the robot may not be reflected in the fitness calculation. Since the addition of extra parameters, for including the information from infrared and sonar sensors may be redundant, it could affect the final selection of the fittest individuals. Therefore, we decided not to use any extra parameters.

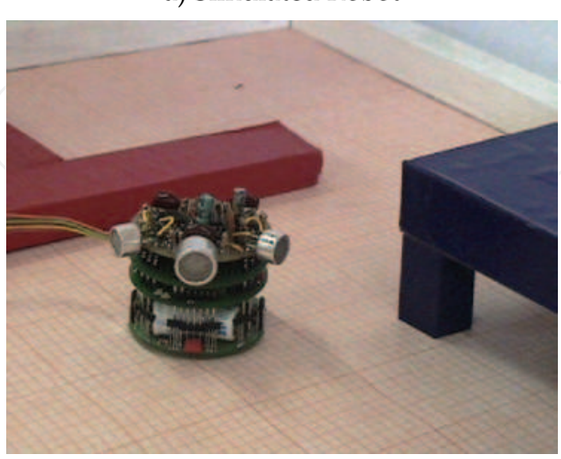
5. Experiments with the Khepera using Ultrasonic and Sonar Sensors

The transference of the neural controller to the Khepera required a hybrid approach that combines the use of a robot simulator (Fig. 6(a)) and the real robot (Fig. 6(b)). Usually the optimization of the neural network in the robot simulator Webots (Webots, 2006) lasted about a day. Then, the network is transferred to the robot, and the input scaled to appropriate values to compensate the artificial sensor readings of the simulator. It is important to notice that a hybrid approach is faster than a full-time optimization in the real robot.

The obstacle avoidance behavior is guided by the adjustment of the neural weights after optimization, and for evaluating the neural controllers we set the robot in a trial arena. Four walls define the borders of the arena, and short walls in the arena define a maze-like environment. A short-legged table is set inside the arena. Despite the space between the legs of the table and the floor not being detected by the infrared sensors; the Khepera has to avoid all kinds of obstacles. In Figure 6(b), we show the final setup for the real robot: an arena surrounded with white tall-walls, a blue table inside of the arena, and obstacles scattered around the arena as red short barriers.



a) Simulated Robot



b) Khepera Robot

Figure 6. The arena where the avoidance behavior occurs

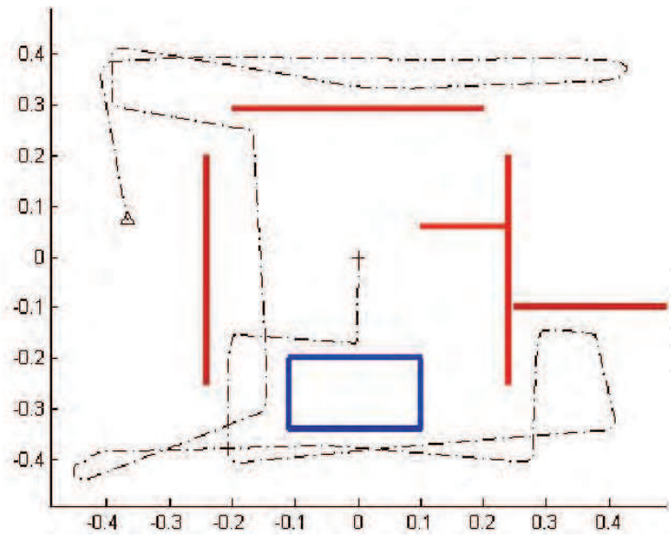


Figure 7. The plot of the trail of the robot in the arena with short obstacles and the table set inside the arena. Red Lines represent short walls that are mostly detected by the infrared sensors. On the other hand, the blue lines form a rectangular shape, which is the table with short legs that can only be detected with the use of the sonars. The dashed line shows the trail of the robot when running in the arena. The cross indicates the start of the avoidance behavior, and the triangle the stop of this behavior

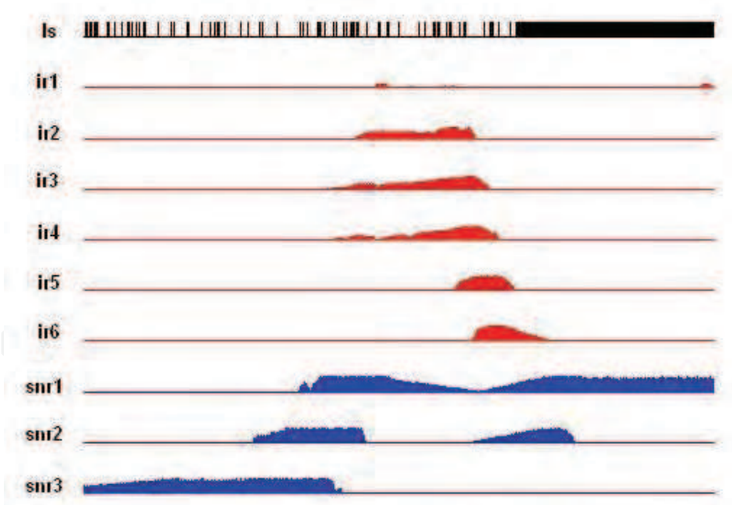


Figure 8. Plot of the infrared and the sonar sensor readings with the output of the linear motor speed. The black output represents the linear speed labeled as *ls*, this linear speed is calculated as in the fitness formula 2 and shows the robot running forward (a suppression of the signal represents a turning maneuver). The six frontal infrared sensors (*ir*1 to *ir*6) are shown in red and indicate the presence of a close obstacle in front of the sensors. The sonars (*snr*1 to *snr*3) are indicated in blue and represent activity when either a tall or legged object is near

The use of the simulator required that the sonars were modeled as if they drew a single ray proximity rather than a beam cone projection. However, as can be noticed in Figure 2(b) the frequency response of the sonar produces a peak where appropriate detection occurs. Therefore, the simplification of the sonar as a simple ray projection is consistent with the measured distance as the intensity of the bounced-back sound. The avoidance behavior is shown in Figure 7, where the plot of the trail of the robot is shown. Additionally, the plot of the Khepera motor output and its sensor-readings resume this avoidance behavior (Fig. 8)

The linear speed output [as in 2], the infrared sensors output, and the sonar-readings are plotted for about 500 iterations of the trial of the robot in the arena. It is importance to notice that the forward traveling of the robot is canceled when a turning maneuver is done, then the detection of objects to be avoided is accomplished by the ring of infrared and sonar sensors. In Figure 8 we observe that the second half of the graph presents the activation of one of the side sonars, and even though this occurs, the robot keeps moving forward because it is running aside of one of the tall walls.

6. The Development of a Foraging Task with Action Selection

In the previous section, we presented how the capabilities of the Khepera can be extended to avoid a maze-like environment with obstacles having short-legs. The use of evolution facilitated the development of an avoidance behavior similar to that shown for a robot, set in an open arena, finding a safe position next to walls. However, we need to explain first how this avoidance behavioral-pattern can be set within an action selection system. The presence of an arbitration scheme within a functional system, where sub-systems exist that compete for gaining control of some associate shared resources, can be identified as an action selection mechanism that solves the action selection problem. Different models have been proposed to design systems, which are able to exhibit a variety of behavior and to arbitrate between them (e.g. behavior based architecture Brooks (1986), Arkin (1998)). Nevertheless, these models based on explicit design does not seem to be scalable enough for being applied to the development of systems capable of displaying a large variety of behavioral patterns that cope with task/environmental variations. One model that is modular and able to cope with the variations of a dynamic environment is CASSF (Central Action Model with Sensor Fusion). A complete description of the implementation of a foraging behavior using this model can be found in the work of Montes-Gonzalez & Marín-Hernández (2004). In this study we are focusing on the development of the avoidance behavior in non-homogeneous maze-like environment. Once the avoidance behavioral pattern has been developed, this can be added to the behavioral repertoire of CASSF (Figure 9).

Additionally, CASSF can be set within a foraging framework with evolved behavior (Montes-Gonzalez, et al., 2006a). On this implementation CASSF employs five basic behaviors that are: *cylinder-seek*, *cylinder-pickup*, *wall-seek*, *cylinder-deposit* and *look-around*. These behaviors can be described as follows: *cylinder-seek* explores the arena searching for food while avoiding obstacles; *cylinder-pickup* clears the space for collecting cylinders; *wall-seek* locates walls while avoiding obstacles; *cylinder-deposit* lowers and opens an occupied gripper; then *look-around* makes a full spin of the robot trying to locate the nearest perceptible cylinder. The development of a foraging framework similar to this one should be sufficient to complete the same task with the substitution of the avoidance

behavior in *wall-seek* by the avoidance behavioral pattern described in this work. As a consequence, collection of cylinders in an arena with short-walls can be completed using an improved avoidance-behavioral pattern.

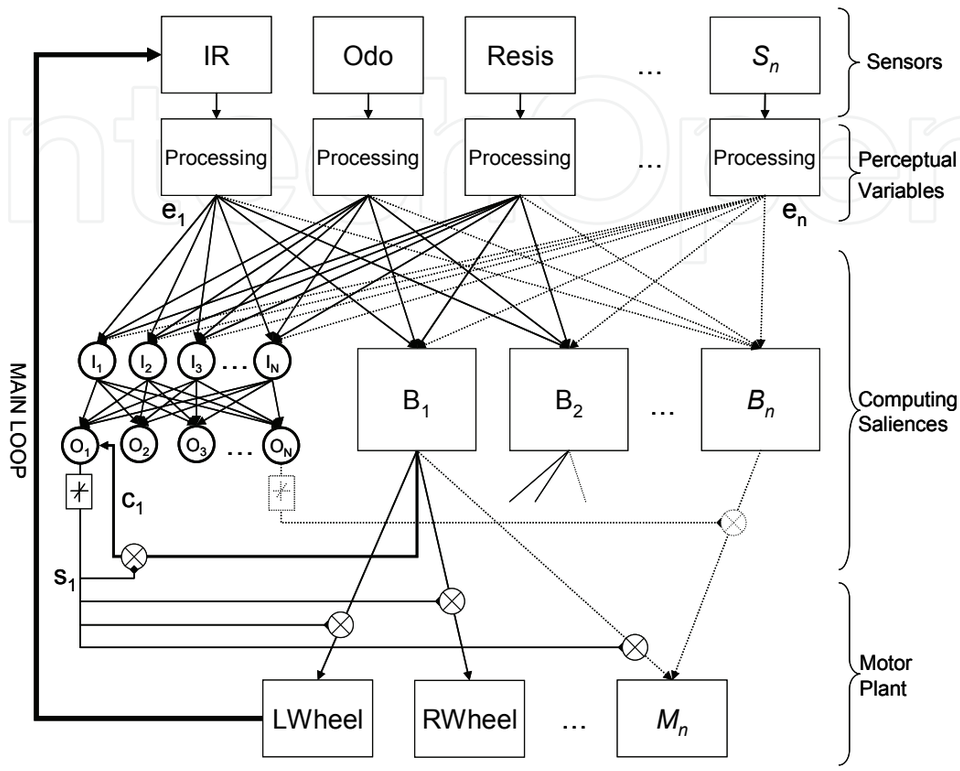


Figure 9. In the CASFF model, perceptual variables (e_i) form the input to the decision neural network. The output of the selected behavior with the highest salience (s_i) is gated to the motors of the Khepera. The busy-status signal (c_1) from behavior B1 to the output neuron O1 should be noticed. The repertoire of behaviors B_1 to B_n can be extended in CASSF by preserving similar connections for each of the additional behaviors that are added. Furthermore, behaviors can be improved off-line and then reinstalled back into the behavioral sub-system

A model such as CASSF is an effective Action Selection Mechanism (Montes-Gonzalez, et al., 2006b) that is centralized and presents sufficient persistence to complete a task. Furthermore, selection parameters of this model have been optimized by the use of evolution. The adjustment of selection parameters and behavior has been optimized by co-evolution in CASSF as described in (Montes-Gonzalez, 2007). The implementation of a foraging framework can be carried out in CASSF by determining some behavioral patterns that can be integrated in time to complete such a task. However, on this work we intend to focus on how the standard Khepera architecture can be improved by the use of evolution and the addition of extra sensors. Therefore, the foraging task has been explained in a concise manner. Next, we draw a general discussion on the further development of these experiments.

7. General Discussion

The use of sonars facilitates the detection of objects not in the direct vicinity of the robot. The sonars send out and ultrasound signal that is reflected back when an object is reached. An elegant solution to estimate the distance of an object employs the Khepera's microcontroller to measure the return time on an ultrasound signal. However, the estimation of distance following this approach is slow, expensive and still requires the use of the General I/O extension turret. In this study a better solution is provided by fitting three ultrasonic sensors on top of the Gen I/O turret, then distance is estimated from the intensity of the reflected sound. The information of the sonars is combined with that of the infrared sensors; then the input signals of both types of sensors are processed using a neural controller with optimized weights by means of a genetic algorithm. In order to speed up the optimization time a robot simulator is employed and the behavior is finally transferred with minor adjustments to the real robot. For the detection of objects like a short-legged table, the robot presents difficulties for the perception of this type of objects relying only on the infrared sensors. Therefore, the improved perception of the Khepera is able to locate non-homogenous collision objects that need to be avoided. The avoidance behavior with an improved perception can be included within a foraging behavioral framework to complete this task by maneuvering in an environment where non-homogenous-collision objects have been scattered around the arena. Additionally, the work presented here can be extended by the optimization of the neural network topology with the use of the genetic algorithm. Finally, it is our intention to expand these experiments by including a foraging prey that is able to collect food-items and run below obstacles like the table also a predator that has to follow the prey avoiding all kinds of obstacles.

8. Acknowledgments

This work has been sponsored by CONACyT-MEXICO grant SEP-2004-C01-45726.

9. References

Airmar-Technology-Corporation (2006). *Specifications Data-sheet for the A30 Ultrasonic Sensor*, http://airmar.com/.

Arkin, R. C. (1998). Behavior-Based Robotics. MIT Press, ISBN 9780262011655, USA.

Bajaj, D. & Ang Jr., M. H. (2000). An Incremental Approach in Evolving Robot Behavior, *Presented at the Sixth International Conference on Control, Automation (ICARCV'2000), Robotics and Vision*, CD-ROM with ISBN 9810434456, Singapore, DEC 2000, Nanyang Technological University, Singapore.

Böndel, D., Grünewald, M., Hanna, T. & Sandmeier, F. (1999). Collective Cube Clustering using a specially designed hardware add-on, In: *Proceedings of the 1st International Khepera Workshop*, Löffler Axel, Mondana Francesco and Rückert Ulrich (Eds.), pp. 253-254, Heinz Nixdorf Institut Paderborn Universität, ISBN 3931466639, Paderborn, Germany.

Brooks, R. A. (1986). A Robust Layered Control System For A Mobile Robot, IEEE Journal Of Robotics And Automation, Vol. 2, Issues 1, MAR 1986, pp. 14-23, ISSN 08824967.

Chapman, T., Hayes, A. & Tilden, M. (2000). Reactive Maze Solving with Biologically-Inspired Wind Sensor. In: *From Animals to Animats 6: Proceedings of the 6th International Conference on the Simulation of Adaptive Behavior*, Berthoz A., Floreano D., Meyer J., Roitblat H. and Wilson S. (Eds.), pp. 81-87, ISBN 0262632004, France, AGO 2000, MIT Press, Paris.

Floreano, D. & Mondana, F. (1994). Automatic Creation of an Autonomous Agent: Genetic Evolution of a Neural-Network Driven Robot, In: *From Animals to Animats 3: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, D. Cliff, P. Husbands, J.-A. Meyer, and S. Wilson (Eds.), pp. 421-430, MIT Press, ISBN 0262531224, Brighton, England.

Goerke, N., Kintzler, F., Raabe, A. & Roggisch, D. (1999). Golf Playing Khepera, In: *Proceedings of the 1st International Khepera Workshop*, Löffler Axel, Mondana Francesco and Rückert Ulrich (Eds.), pp. 245-246, Heinz Nixdorf Institut Paderborn Universität, ISBN 3931466639, Paderborn, Germany.

Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, ISBN 0262581116, Ann Arbor.

Mondana, F., Franzi, E. & Paolo, I. (1993). Mobile robot miniaturisation: A tool for investigating in control algorithms, *Presented at the Proceedings of the 3rd International Symposium on Experimental Robotics III LNCIS Vol. 200*, T.Y. Yoshikawa & F. Miyazaki (Eds.), pp. 501-513, ISBN 3540199055, Japan, OCT 1993, Springer-Verlag, Kyoto.

Montes-González, F. M., and A. Marín-Hernández (2004), Central Action Selection using Sensor Fusion. *Presented at the Proceedings of the Fifth Mexican International Conference in Computer Science* (ENC'04), Baeza-Yates R., Marroquín J.L. and Chávez E (Eds.), pp. 289-296, ISBN 0-7695-2160-6, Colima, SEP 2004, IEEE Press, Mexico.

Montes-González, F., J. Santos Reyes, and H. Ríos Figueroa (2006a), Integration of Evolution with a Robot Action Selection Model, *Presented at the Mexican International Conference on Artificial Intelligence (MICAI 2006), LNAI 4293,* A. Gelbukh and C. A. Reyes-García (Eds.), pp. 1160-1170, ISBN 9783540490265, Apizaco, NOV 2006, Springer Berlin / Heidelberg, Mexico.

Montes-González, F. M., A. Marín Hernández, and H. Ríos Figueroa (2006b), An Effective Robotic Model of Action Selection, In: *CAEPIA 2005, LNAI 4177,* R. Marín, et al. (Eds.), pp. 123-132, ISBN 9783540459149, Santiago de Compostela, NOV 2005, Springer Berlin / Heidelberg, Spain.

Montes-Gonzalez, F. (2007). The Coevolution of Robot Behavior and Central Action Selection, In: *LNCS 4528 Nature Inspired Problem-Solving Methods in Knowledge Engineering IWINAC 2007 Part II*, J. Mira and J.R. Álvarez (Eds.), pp. 439–448, Springer Berlin / Heidelberg, ISBN 9783540730545, La Manga del Mar Menor, Spain.

Nolfi, S. & Floreano, D. (2000) . *Evolutionary Robotics*. The MIT Press, ISBN 0262140705, USA.

Odenbach, C., Rüping, S., Löffler, A. & Rückert, U. (1999). An operating wireless CAN Communication System for Khepera Robots, In: *Proceedings of the 1st International Khepera Workshop*, Löffler Axel, Mondana Francesco and Rückert Ulrich (Eds.), pp. 181-187, Heinz Nixdorf Institut Paderborn Universität, ISBN 3931466639, Paderborn, Germany.

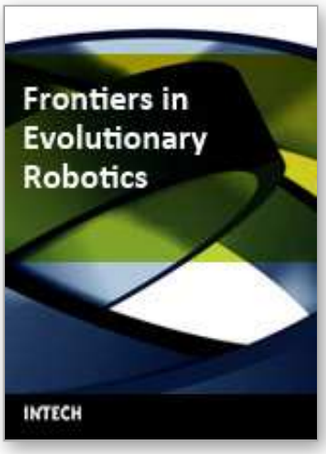
Torben-Nielsen, B., Webb, B. & Reeve, R. (2005). New ears for a robot cricket. Lecture notes in computer science, Vol., Num 3696, SEP 2005, pp. 297-304, ISSN 03029743.

Webots (2006). *Commercial Mobile Robot Simulation Software*. http://www.cyberbotics.com.

Winge, F. (2004). *Sonar for a Khepera*. Dept. of Computer Science, Master Thesis, Lund University.

IntechOpen

IntechOpen



Frontiers in Evolutionary Robotics

Edited by Hitoshi Iba

ISBN 978-3-902613-19-6

Hard cover, 596 pages

Publisher I-Tech Education and Publishing

Published online 01, April, 2008

Published in print edition April, 2008

This book presented techniques and experimental results which have been pursued for the purpose of evolutionary robotics. Evolutionary robotics is a new method for the automatic creation of autonomous robots. When executing tasks by autonomous robots, we can make the robot learn what to do so as to complete the task from interactions with its environment, but not manually pre-program for all situations. Many researchers have been studying the techniques for evolutionary robotics by using Evolutionary Computation (EC), such as Genetic Algorithms (GA) or Genetic Programming (GP). Their goal is to clarify the applicability of the evolutionary approach to the real-robot learning, especially, in view of the adaptive robot behavior as well as the robustness to noisy and dynamic environments. For this purpose, authors in this book explain a variety of real robots in different fields. For instance, in a multi-robot system, several robots simultaneously work to achieve a common goal via interaction; their behaviors can only emerge as a result of evolution and interaction. How to learn such behaviors is a central issue of Distributed Artificial Intelligence (DAI), which has recently attracted much attention. This book addresses the issue in the context of a multi-robot system, in which multiple robots are evolved using EC to solve a cooperative task. Since directly using EC to generate a program of complex behaviors is often very difficult, a number of extensions to basic EC are proposed in this book so as to solve these control problems of the robot.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Fernando Montes-Gonzalez, Daniel Flandes-Eusebio and Luis Pellegrin- Zazueta (2008). Action Selection and Obstacle Avoidance Using Ultrasonic and Infrared Sensors, Frontiers in Evolutionary Robotics, Hitoshi Iba (Ed.), ISBN: 978-3-902613-19-6, InTech, Available from: http://www.intechopen.com/books/frontiers_in_evolutionary_robotics/action_selection_and_obstacle_avoidance_using_ultrasonic_and_infrared_sensors

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820

Fax: +385 (51) 686 166
www.intechopen.com

Fax: +86-21-62489821

IntechOpen

IntechOpen

© 2008 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.