Reaction-Diffusion Navigation Robot Control: From Chemical to VLSI Analogic Processors

Andrew Adamatzky, Paolo Arena, *Senior Member, IEEE*, Adriano Basile, *Student Member, IEEE*, Ricardo Carmona-Galán, *Member, IEEE*, Benjamin De Lacy Costello, Luigi Fortuna, *Fellow, IEEE*, Mattia Frasca, *Member, IEEE*, and Angel Rodríguez-Vázquez, *Fellow, IEEE*

***Abstract*—We introduce a new methodology and experimental implementations for real-time wave-based robot navigation in a complex, dynamically changing environment. The main idea behind the approach is to consider the robot arena as an excitable medium, in which moving objects—obstacles and the target—are represented by sites of autowave generation: the target generates attractive waves, while the obstacles repulsive ones. The moving robot detects traveling and colliding wave fronts and uses the information about dynamics of the autowaves to adapt its direction of collision-free motion toward the target. This approach allows us to achieve a highly adaptive robot behavior and thus an optimal path along which the robot reaches the target while avoiding obstacles. At the computational and experimental levels, we adopt principles of computation in reaction–diffusion (RD) nonlinear active media. Nonlinear media where autowaves are used for information processing purposes can therefore be considered as RD computing devices. In this paper, we design and experiment with three types of RD processors: experimental and computational Belousov–Zhabotinsky chemical processor, computational CNN processor, and experimental RD-CNN very large-scale integration chip—the complex analog and logic computing engine (CACE1k). We demonstrate how to experimentally implement robot navigation using space-time snapshots of active chemical medium and how to overcome low-speed limitation of this "wetware" implementation in CNN-based silicon processors.**

***Index Terms*—Belousov–Zhabotinsky (BZ) reaction, excitation waves, nonlinear dynamics, reaction–diffusion (RD) media, robot navigation control.**

I. Introduction

ROBOT navigation in complex, and often a dynamically changing environment has remained one of the actual problems of robotics for several decades. Dealing with the issue, we must answer the questions: how to calculate an ideally optimal and shortest path between two sites of an experimental arena along which a robot may travel without colliding with obstacles and with the aim to reach a given target (see, e.g., [1]–[3]). If the whole experimental arena can be mapped into the computational architecture of the robot navigator, then, the problem could be solved using the paradigm of artificial potential fields: obstacles generate repulsive fields and targets generate attractive fields. Thus, the robot tries to align its trajectory with attractive gradients and repulsive anti-gradients [4]–[6]. The potential field-based robotic controllers were implemented in silicon in [7], [8]. In particular, in [7], only very large-scale integration (VLSI) prototypes of resistive, preprogrammed grids containing six nodes were built. In direct realization [8] of a chip implementing the resistive grid, the drawbacks of the approach were revealed: Local voltage difference becomes negligible for large grids which leads to errors in the determination of the local path and shows high noise sensitivity. More sophisticated navigation techniques are based on heat transfer [9], diffusion [10], and fluid dynamics [11].

There are experimental implementations of chemical controllers that calculate an optimal path in an obstacle-filled environment, most of them are based on the excitation-wave dynamics in the Belousov–Zhabotinsky (BZ) medium. The obstacles are represented by physical heterogeneity of substrate [12], local disturbances of concentrations [13], [14], or singularities in illumination [15]. The reaction–diffusion (RD) approaches were experimentally verified with real robots in [16], using on-board BZ reactors.

The combination of the attraction–repulsion approach was previously employed in [17]: The repulsive field in this case was produced by an experimental RD processor, the attractive field was then computed using a cellular-automaton model of an excitable chemical medium.

In this paper, the methodology is extended by using two parallel chemical reactors with BZ media for the trajectory evaluation. On the basis of the results obtained, it is possible to evaluate the suitability of the approach, but the main drawback, in view of a real-time implementation, is due to the limited speed of diffusion of the wavefronts in the chemical processors.

Manuscript received August 1, 2003; revised December 16, 2003. The work of P. Arena, A. Basile, L. Fortuna, and M. Frasca was supported in part by the Italian "Ministero dell'Istruzione, dell'Universit e della Ricerca" (MIUR) under Project Firb RBNE01CW3M and Project PRIN "Innovative Bio-Inspired Strategies for the Control of Motion Systems." The work of R. Carmona-Galan and A. Rodriguez-Vazquez has been supported in part by LOCUST under Project IST-2001–38097. This work was supported in part by EPSRC inder Grant GR/R31225/01. This paper was recommended by Guest Editor B. Shi.

A. Adamatzky is with the Faculty of Computing, Engineering and Mathematical Sciences, University of the West of England, Bristol BS16 1QY, U.K. (e-mail: andrew.adamatzky@uwe.ac.uk).

P. Arena, L. Fortuna, and M. Frasca are with the Dipartimento di Ingegneria Elettrica Elettronica e dei Sistemi, Universitá degli Studi di Catania, 95100 Catania, Italy (e-mail: parena@diees.unict.it; lfortuna@diees.unict.it; mfrasca@diees.unict.it).

A. Basile is with the Dipartimento di Ingegneria Elettrica Elettronica e dei Sistemi, Universitá degli Studi di Catania, 95100 Catania, Italy, and also with the Robotics and Automation Group, ST Microelectronics, 95121 Catania, Italy (e-mail: abasile@diees.unict.it).

R. Carmona-Galan and A. Rodriguez-Vazquez are with the Institute of Microelectronics of Seville, Centro Nacional de Microelectrónica (IMSE-CNM), Universidad de Sevilla, 41012 Seville, Spain.

B. De Lacy Costello is with the Faculty of Applied Sciences, University of the West of England, Bristol BS16 1QY, U.K. (e-mail: ben.delacy-costello@uwe.ac.uk).

Digital Object Identifier 10.1109/TCSI.2004.827654

On the other hand, autowaves have been known about for many years by the cellular nonlinear network (CNN) scientific community [18]–[20]. Autowaves were proven to arise in very different conditions and from very different mathematical models [22], but all belonging to the class of RD systems, and as such, they can be mapped into particular CNN structures, the so-called RD-CNNs. [23].

In this paper, the same approach used to generate attractive and repulsive fronts in a chemical medium is integrated into an approach uniquely based on the parallel and real-time processing capabilities of CNNs. In fact, the robot arena is mapped on a CNN: its role is to select obstacles and identify them as sources of repulsive autowaves. Subsequently, a RD-CNN generates autowave fronts. The robot is allowed to freely move within the arena. A simple AND operation reveals the wavefront collisions with the robot, and a steering command is sent to the robot for the generation of highly adaptive trajectories. In particular, two CNN algorithms are introduced in the paper: the former based on a motion detection algorithm, while the latter on a particular assumption about the robot visual representation. The algorithm introduced in this paper is targeted for the new family of RD-CNN chip, the complex analog and logic computing engine (CACE) chip. The use of a VLSI chip is also shown as the final step that will implement the whole strategy for real-time control of an autonomous robot trajectory. The availability of programmable chips for the hardware generation of complex phenomena is really a breakthrough in the field of mobile robotics, since the implementation of classical algorithms, like the potential field in real-time within a changing environment is not feasible. In our case, the chip response to the visual input and the corresponding robot command can stay within the classical inter frame rate. Under these conditions, the overall control efficiency could even depend on the robot capability to follow the trajectory reference. Another interesting issue regards the fact that RD-CNN were found able to generate self-organized dynamics for the locomotion control in legged robots [24]. In this paper, the same spatial temporal dynamics, and in particular the same CNN cell structure and parameters are used for the high level, trajectory control.

From these considerations, it derives that RD-CNN can represent an emerging paradigm for the real-time control in bio-inspired robotics. This methodology allows the selection of the optimal robot trajectory in dynamically changing environment in extremely reduced times: under this perspective such an algorithm, that solves a complex task, succeeds in being even faster than, in several cases, the robot typical reaction times.

II. Analog Processors for Robot Navigation Control

RD phenomena occur in many systems [22], but they obey the same laws that can be represented under the following general RD equations:

$$\frac{d\mathbf{u}}{dt} = f(\mathbf{u}) + \nabla^2\mathbf{u} \tag{1}$$

in which $\mathbf{u}$ is a vector of at least two elements. For instance, $\mathbf{u}$ can represent the dynamics of an activator/inhibitor system diffusing in a two dimensional medium and showing pattern formation. In this paper, we are interested in nonlinear media, represented by (1) showing wave propagation. These waves are very important for the navigation control, since the trajectory computation can be easily realized through their generation, propagation and interaction. Under this perspective, nonlinear media represent a class of parallel computers with unique features: parallel input of data, parallel information processing, and parallel outputs [25].

In our approach, the RD medium is devoted to control the trajectory of a robot in an environment with obstacles, where a target point for the robot to reach can also be fixed. Both obstacles and target are mapped onto the nonlinear medium as autowave sources, stimulating different regions or cells. A key characteristic of autowaves is that they annihilate when they collide [26], in such a way they intrinsically determine the path of points equidistant from the obstacles.

This paper deals with two realizations of the RD paradigm, based on chemical and silicon implementations. The first is a chemical processor employing a generation and spreading of autowaves in BZ reaction. Information about positions of obstacles and targets is transmitted by autowaves and used to compute the robot trajectory. The same methodology is implemented by using another, silicon-based, active nonlinear medium, an RD-CNN structure where autowaves have been shown to take place [27]. The mathematical model of the single RD-CNN generic cell C_{ij} is reported here

$$\begin{cases} \dot{x}_{i,j;1} = & -x_{i,j;1} + (1+\mu)y_{i,j;1} - s_1 y_{i,j;2} \\ & +i_1 + D_1(y_{i+1,j;1} + y_{i-1,j;1} \\ & +y_{i,j+1;1} + y_{i,j-1;1} - 4y_{i,j;1}) \\ \dot{x}_{i,j;2} = & -x_{i,j;2} + (1+\mu)y_{i,j;2} + s_2 y_{i,j;1} \\ & +i_2 + D_2(y_{i+1,j;2} + y_{i-1,j;2} \\ & +y_{i,j+1;2} + y_{i,j-1;2} - 4y_{i,j;2}) \end{cases} \tag{2}$$

considering also, as the output nonlinearity, the classical saturation one.

The BZ reaction is often modeled by using lattices of third-order nonlinear cells, implementing the Oregonator third-order nonlinear system. Although it is well known that the Oregonator equations can be mapped in a RD-CNN [28], as well as each nonlinearity can be approximated by a piecewise linearity (PWL) [29]; in this paper, the CNN algorithms generating autowaves will consider the model in (2). The advantages are twofold: each single cell is a second-order model, instead of the third one, required for the Oregonator model; moreover, in our case, the nonlinearity is PWL, so directly compatible with the chip requirements.

The two implementations (chemical and CNN) of the RD processor described in the next sections obviously show some differences not only in the analog processor used, but also in how input is treated, how stimulation takes place and how output is further elaborated, but clearly the general paradigm is unique.

The main limitation of chemical processors is that waves travel very slowly: the VLSI CNN implementation allows us to overcome this limitation by keeping all the peculiarities of the paradigm. Under this point of view, the CNN approach allows a real-time computation based on wave computing of the trajectory for a moving robot in an environment that can be critical and extremely time varying and unstructurated, and, therefore permits us to solve a primary problem in mobile robotics.

Moreover, taking into account that the same model for the RD-CNN was also successfully used as the basic neuron model for bio-inspired locomotion control [24], [30], it derives that the class of RD-CNN discussed in this paper represents a unifying paradigm for the solution of locomotion control, from the low level of the motion pattern generation, to the very high level of the trajectory planning.

III. Navigating Robot in Chemical Media

A. Experimental

RD chemical processors were prepared using a thin-layer BZ reaction [21] mixture of three solutions: A, an acidic bromate stock solution incorporating potassium bromate and sulphuric acid; B, solution of malonic acid; C, sodium bromide.

To prepare a thin layer of the BZ medium, we mixed solutions A (7 ml), B (3.5 ml), and C (1.2 ml). When the solution had become colorless, ferroin (1 ml), used as catalyst, was added and the mixture was transferred to a Petri dish (layer thickness 1 mm). The catalyst was also a visual indicator of the excitation activity in the BZ medium. Excitation waves in the BZ reaction were initiated using a silver colloid solution. Targets and obstacles were represented by local wave excitation in separate instances of the BZ medium.

Space–time dynamics of the excitation in the reaction medium were recorded using a FujiFilm 2600z digital camera (zoom lens $f = $ 6-18 mm equivalent to $f = 38 - 114$ on a 35-mm camera). The images were recorded using the built-in macro function with a resolution of 320×240 pixels and using a transmissive lighting method. No filters were used. As regards the typical computation time, it is related to the speed of the wavefront in the BZ medium. The excitation wavefront moves with the speed around 0.2–2 mm/min (dependent on many factors). Essentially, about tens of minutes are needed for the experiment if a Petri dish of 9 cm is used.

B. Principles of Navigation

We navigate the robot in RD media using a combination of attractive and repulsive patterns generated by excitation waves in chemical media. In our previous papers [14], [17], we demonstrated feasibility of this approach with attractive wavefronts implemented in cellular-automaton models and repulsive fronts in real chemical systems.

Ideally, it would be reasonable to have two excitable chemical systems (whose wavefronts can be detected using different visual indicators) physically coexisting in one reactor. In such an ideal case, the robot will be attracted by wavefronts of one system—which represents a target—and repelled by wavefronts of the other system—which represents obstacles. This physical coexistence does not make an easy experimental option [31].

Therefore, in this paper, we navigate robots using two separate and isolated (from each other) chemical reactors containing the BZ medium. Obstacles are mapped onto one reactor and targets onto another. We assume that the robot detects concentration of the chemical species using optical sensors from spatial snapshots of the BZ medium activity. Analysis of wave profiles [16] shows that RGB color scheme gradations of blue color very efficiently characterize the specific of excitation wave dynamics in the experimental BZ medium. The cross-section profile of the BZ wavefront shows a steep rise at the wavefront's head and gradual descent at the wavefront's tail. Moving on the medium the robot avoids sites of "repulsion," i.e., points of the medium with blue color gradations exceeding a certain threshold. This guarantees that the robot will, in all likelihood, not cross a repelling wavefront. Thus, a collision with any obstacles is avoided. Relatively gradual descent of the wavefront's tail allows the robot to move along a gradient of color gradation to reach the target. The excitation wave patterns are discrete, i.e., comparing to the situation of an attractive field, the robot can not move "smoothly" along the color gradients. Initially, the robot must somehow "jump" on top of a wavefront and only then freely roll down the tail. The "jump" will be implemented using a kind of "energy accumulation" when modeling robot behavior. The wavefront's head is much shorter than the wavefront's tail; therefore, once it has crossed the wavefront, the robot will be trapped inside the target wave because it will not be able to accumulate enough energy to jump over the tail.

C. Pixbot and its Behavior

To verify the robot navigation ideas described, we constructed a software model of "pixbot"—a pixel-size robot which moves in discrete steps on images (attractive medium $\mathbf{A}$ and repelling medium $\mathbf{R}$) of the spatial excitation dynamics of the BZ medium. At each step of discrete time, the pixbot can move to one of 8 pixels closest to $(x,y)^t$ (the pixbot's coordinates at time step t).

If the pixbot spends too much time wandering in the same local domain, it is forced to jump randomly; this will allow the pixbot to mount wavefronts.

Because of significant influence of the stochastic component on the pixbot behavior the pixbot reaches the target by various routes [Fig. 1(a) and (b)] and it may wander for a while between two wavefronts, generated by the same target (Fig. 1). There is no guarantee the best (i.e., oldest or strongest target) will be chosen because as soon as first wavefront is crossed there is no way back [Fig. 1(a) and (c)].

An example of the pixbot's collision-free movement toward a target is shown in Fig. 2. The trajectory is relatively straightforward when the obstacles are absent [Fig. 2(a)]. The trajectory's disorder is more visible in domains where pixbot's path deviates from the straight line between start point and the target [Fig. 2(b)].

To test stability of the algorithm in conditions of the onboard chemical reactor, we put Petri with BZ medium onto a specially constructed table rotating 1° s. The centrifugal forces disturbs wavefronts and make their structure more irregular, this does not prevent the pixbot from reaching the target. This demonstrate that the RD approach will work when implemented in real life conditions.

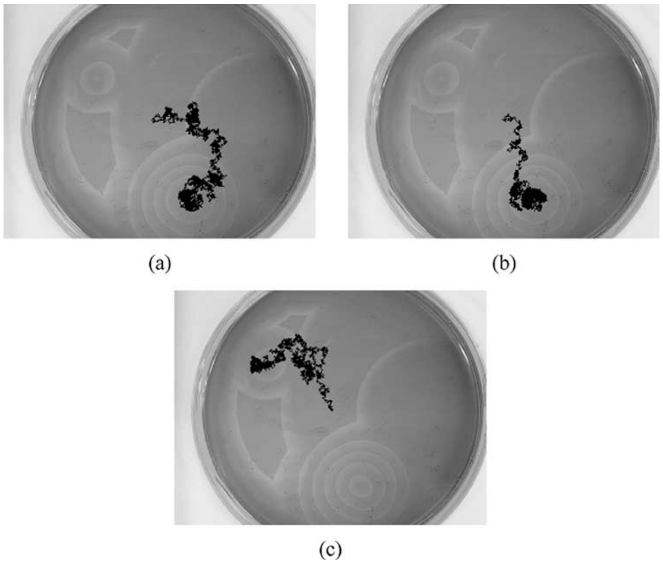

Fig. 1. Stochastic component of pixbot behavior determine various routes used to reach the target (a) and (b), and two targets can be randomly selected (a), (b), and (c). No obstacles are present in this particular experiment. Trajectories of pixbot are shown on images of BZ medium.

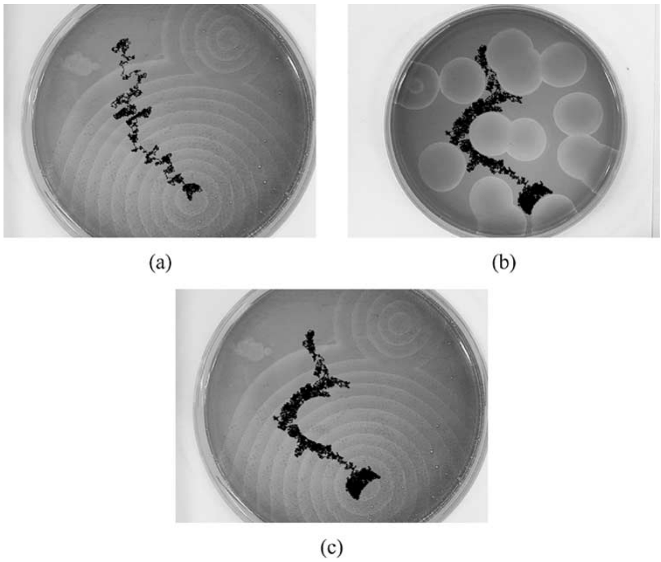

Fig. 2. Trajectories of pixbot moving toward the target (a) without obstacles and with obstacles (b) and (c).

IV. CNN Implementation of Robot Navigation Control

A. General Description of the CNN Implementation

The CNN framework constitutes a real-time implementation of the basic methodology just applied to a chemical wave processor. Moreover, thanks to the universal computing properties of a CNN chip, a huge number of additional algorithms can be used, making them able to perform additional complex image-filtering routines, that are needed to have a complete and really working algorithm for robot trajectory planning. In fact, a first, essential step that has to be performed *a priori*, lies in the target/obstacle recognition. In our case, the input is provided by a video camera making a motion picture of the environment where the robot moves. Let us suppose that the obstacles can be detected if some *a priori* information is given. For example, let us assume that all the obstacles have a circular shape. In this case an on-chip CNN algorithm, already implemented and tested on the ACE4k chip [32], can be applied. Of course, CNNs allow us to implement a huge number of different image filtering and detail extraction routines that can be adapted to the environment characteristics. As a result of this filtering procedure, an image is provided, containing only the obstacles surrounding the robot, whose position is assumed to be known at any given time. This image, containing only the obstacles is processed by a RD-CNN in which autowaves propagate: this constitutes the *complex* analog processor. In particular, the CNN cells corresponding to the position of obstacles are stimulated by setting initial conditions for the first layer of the CNN to $x_{ij} = +1$. Waves, generated at the obstacle positions, propagate until they collide with the robot. In our CNN autowave model, we have an even sharper wavefront with respect to the chemical case just discussed: the wavefronts do not involve more than 3 pixels in the propagation direction, (one forward and one behind the maximum intensity pixel). So, also in this case, the field information is discrete: the robot "senses" the field only when it collides with a wavefront. Depending on the kind of source generating the wave (obstacle or target) and on the particular compass direction the wavefront comes from, the new direction for the robot can be directly determined. Thus, the trajectory planning is not performed before the robot starts moving, but occurs step by step while the robot moves, soon after each wave-robot collision. In particular, after the completion of a robot movement, one could establish to process another frame or not, i.e., one can decide either to reset the RD-CNN and start with a new configuration of obstacles and target, or to continue with the previously acquired configuration. This allows to account also for moving obstacles or to speed-up the algorithm.

The details of the algorithm used and experimental results are presented in the following section.

B. The CNN Algorithm

Two different CNN algorithms are both discussed here for the sake of completeness and clarity. In fact, the second one is simpler, and therefore more easily implementable on CNN chips. In general two complementary RD-CNNs, as in the case of chemical reactors, are used: one is the medium for propagation of waves generated by obstacles, while the second deals with the target.

A flow diagram of the first algorithm is shown in Fig. 3, where the case of sources generated by obstacles is shown (the algorithm for the target is complementary to this). The algorithm consists of the following steps:

Step 1) Autowaves propagation on the RD-CNN. This is the core of the algorithm, it consists of a RD-CNN of equations (2) in which waves propagate and interact. The RD-CNN maps the space in which the robot moves. Sources of the autowaves are generated in correspondence of obstacles or target. In practice this can be accomplished by a suitable preprocessing algorithm of the image captured by the camera. Of course, the preprocessing algorithm based on quite traditional steps

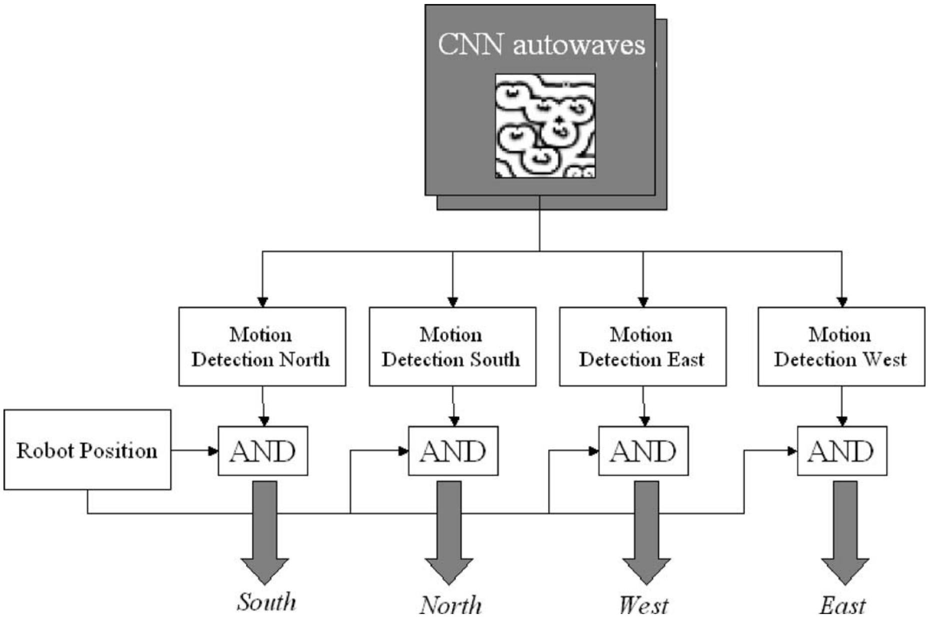

Fig. 3. Flow diagram of the CNN algorithm based on motion-detection templates (the case of autowaves generated by obstacles is shown).

to identify obstacles and target can be implemented by using CNN templates.

Step 2) Motion detection. The output of the RD-CNN is then processed by a one layer CNN (1L-CNN) able to detect motion in one of the four compass directions. This step is implemented by using motion-detection templates [33]. These templates are time delay templates. However, other CNN-based algorithms for motion detection that are not based on time delay CNN and therefore that are suitable for an implementation on VLSI CNN chip have been developed [34], **[36]**.

Step 3) Final step. At this point, an AND operation between the result of motion detection and an image with black pixels only at the robot position gives the final result (simple instructions for the robot as for instance "go south").

This algorithm is very effective: if wave generated by the obstacles are a few times faster than those generated by the target, very smooth trajectories for the robot are obtained. However, this can be done efficiently only in simulation, while the motion detection algorithm could be quite time consuming for on chip implementation.

For these reasons, another simpler but more efficient algorithm for hardware implementation has been taken into account. The focus of this algorithm is to make simpler the CNN processing phase while adding more interaction capabilities to the robot. In fact while in the previous algorithm the motion detection phase was needed to determine the direction of colliding wavefronts, in this algorithm the robot has been endowed with "sensing pixels" in different positions. So, the corresponding direction can be easily detected according to which of the pixels is first reached by the colliding wavefront.

The flow diagram of this algorithm is shown in Fig. 4. The robot is now viewed as a 4-pixel object containing three active pixels (one in the front and two in the back), as shown in Fig. 4. This allows to simplify Step 2) of the previous algorithm, leaving the other steps unchanged. Several other robot geometries can be taken into consideration, according to the degree of accuracy in detecting the colliding wavefront direction.

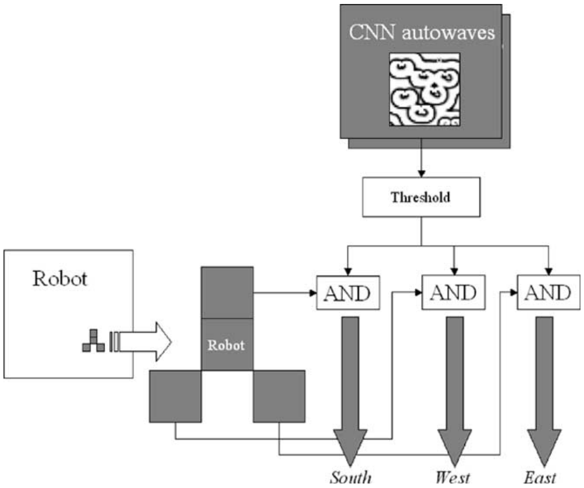

Fig. 4. Flow diagram of the simplified CNN algorithm (the case of autowaves generated by obstacles is shown).

Step 2) now consists of a threshold operation on the RD-CNN result. The purpose of this threshold operation is to clearly identify the wavefront's head.

After the threshold operation, the AND operation in Step 3) is now between the snapshot of wave propagation, represented by the state of the first layer of the RD-CNN, and the image containing the robot. Depending on which of the robot pixels is first reached by the wavefront, a particular motion instruction is given to the robot.

The simplified algorithm can be therefore summarized in the following three steps.

Step 1) Autowaves propagation on the RD-CNN.

Step 2) Threshold operation on the output of the RD-CNN operated by the 1-layer CNN.

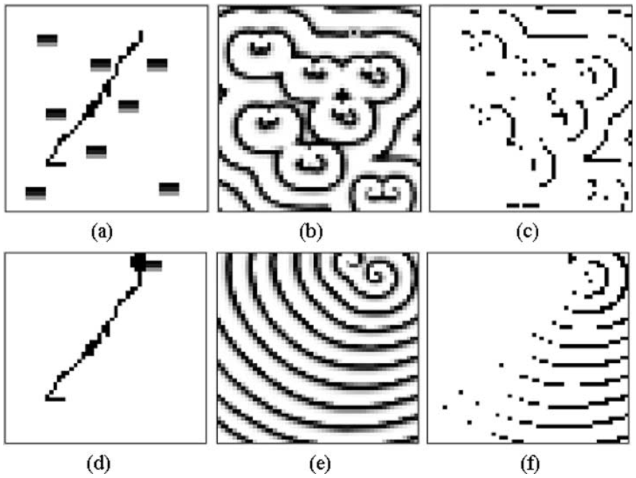

Fig. 5. Simulation results by using the algorithm based on motion-detection. (a) Obstacle position and robot trajectory. (b) Obstacle-generated waves. (c) Result of motion-detection for the east direction on the target-generated waves. (d) Target position and robot trajectory. (e) Target-generated waves. (f) Result of motion-detection for the east direction on the target-generated waves.

Step 3) Final step: AND operation between the result of threshold operation and the robot image.

The parameters of the RD-CNN are chosen according to $\mu = 0.7$, $s = 1$, $D_1 = 0.1$, and $D_2 = 0.1$, thus leading to the following templates:

$$A_{11} = \begin{pmatrix} 0 & 0.1 & 0 \\ 0.1 & 1.3 & 0.1 \\ 0 & 0.1 & 0 \end{pmatrix}$$

$$A_{12} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$A_{21} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$A_{22} = \begin{pmatrix} 0 & 0.1 & 0 \\ 0.1 & 1.3 & 0.1 \\ 0 & 0.1 & 0 \end{pmatrix}. \qquad (3)$$

C. Simulation Results

Simulation results have been obtained by using a dedicated framework, written in C++, for the simulation of the robot and of the CNN algorithm. A 50 × 50 CNN has been simulated. As regards the results obtained by implementing the first algorithm, Fig. 5 shows the first example in which eight randomly placed obstacles and a target near the north-east corner are taken into account. Their positions are shown in Fig. 5(a) and (d) with the robot trajectory. The obstacles generate repulsive autowave fronts [Fig. 5(b)], while the target attractive ones [Fig. 5(e)]. The robot, inside this attractive/repulsive field, is able to reach the target while avoiding obstacles. Fig. 5(c) and (f) show the result of motion-detection (for the east direction) on obstacle-generated and target-generated wavefronts.

Finally, Fig. 6 reports a case in which only obstacles are considered. Obstacles are placed all around the robot. They generate repulsive waves (Fig. 6(b)). Therefore, the action imposed to the robot is to escape by this generated field. After several trials, the robot is indeed able to find out the way to escape from the obstacles.

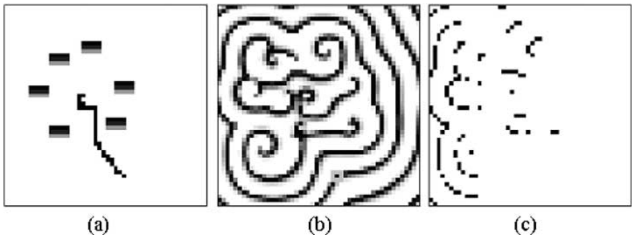

Fig. 6. Simulation results by using the algorithm based on motion-detection: case with only obstacles.

Fig. 7 shows several examples of simulation results obtained by using the simplified algorithm. In this case, instead of a target to reach, the robot focuses to proceed in a prespecified direction (in this case north). So, in the simulation, only obstacles are taken into account.

As can be seen from the simulation results, both of the algorithms presented are really effective. However, there can be cases in which the robot could be trapped into a local minimum, for example, when target and obstacles are in particular symmetrical points with respect to the robot. In such cases, the robot moves alternatively between the same two positions. To overcome such a condition, a random perturbation on the robot trajectory, is added in analogy with the algorithm presented for the chemical case. Moreover, in view of a real implementation, it is not necessary that the video feedback is done on the whole robot arena, but it can be assumed that the processing takes place only on a certain region surrounding the robot. Due to the adaptive characteristics of the methodology, the environment focused by the camera can vary as the robot moves. So, for example, at the beginning, when the target could be out of the captured area, the robot is assigned to move only in the direction of the target. So, its first task could be to avoid obstacles while moving in a particular direction. As the robot position approaches the target, the final focus could be taken into consideration. Moreover, taking limited, next-to-robot frames, improves the computational efficiency, since distant obstacles do not generate wavefronts; so, the local trajectory is influenced only by obstacles in the immediate vicinity of the robot. These aspects are really important, in view of a real implementation on the RD-CNN chip.

D. Results on a Roving Robot

The algorithms discussed previously are quite general and, also in view of the considerations just reported, they can be implemented with two different installations of the camera: on the ceiling of the laboratory (as it has been assumed in the simulations) or on board, taking a picture of a small planar space in front of the robot. Thus, a *world-centered perception*, or a *robot-centered perception* can be implemented by using the general paradigm of RD navigation control. The experimental setup is provided by a real environment, where a roving robot is required to move in a prespecified direction, in this case south: so, no particular target positions are given. The camera is connected to a personal computer simulating the RD-CNN simplified algorithm.

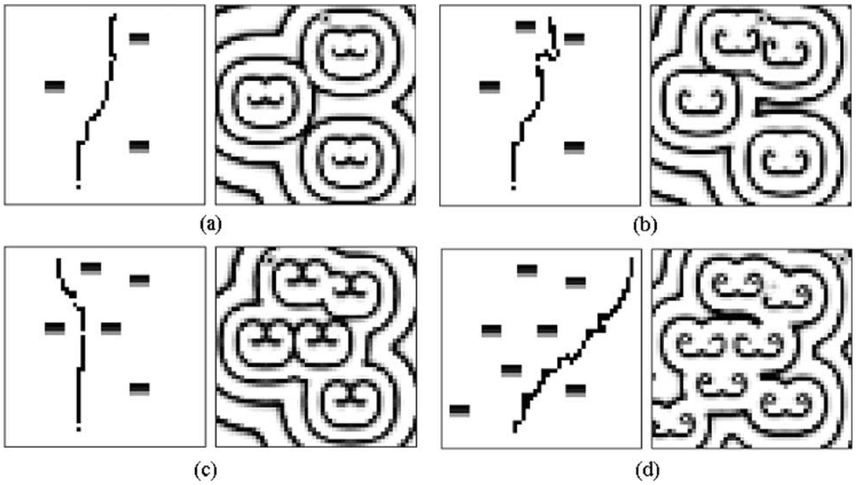

Fig. 7. Simulation results by using the simplified CNN algorithm: case with only obstacles.

Unlike the simulation case, here, obstacles are smaller than the robot. This could represent a disadvantage and a careful investigation on the most appropriate dimensions should be performed in order to optimize the strategy. However, it has been experimentally observed that in the main potentially dangerous situation represented by two very close obstacles, the two obstacle wavefronts are fused before reaching the robot so that the robot perceives just one obstacle.

In the first experiment, the *world-centered perception* case is reproduced. At each step, the robot camera takes a picture of the environment: obstacles are used as initial conditions for the RD-CNN. When the wavefronts reach the robot, depending on the direction the wavefront comes from, the robot executes the proper command. The robot succeeds in going in the prespecified direction avoiding two obstacles, as can be appreciated in Fig. 8. Here, the positions successively occupied by the robot are reported through black points. From the analysis of the figures it emerges that the robot is indeed able to avoid both the obstacles, but it passes much closer to the one than the other. This potential problem can be explained as follows. Since autowave fronts annihilate while colliding, if the robot moves in a direction where a wavefront has just been annihilated, it will “sense” the subsequent front when the robot will be next to its source, i.e., the obstacle. A suitable synchronization among the robot motion and the wave generation should avoid the problem.

The second experiment refers to the *robot-centered perception* strategy: the camera is placed on the robot via a vertical pole whose height is about 1 m. The camera is positioned in such a way that the focal plane is almost parallel to the ground and the robot is situated in the central bottom position within the frames. In this condition, the frames refer to the environment in the forward motion direction of the robot. A picture of the robot is shown in Fig. 9. The whole experiment is reported in Fig. 10. Here, some snapshots are illustrated: as can be noticed, the robot is able to avoid both the obstacles, while proceeding in the prespecified direction. The corresponding video can be seen at the web page: *http://www.dees.unict.it/users/parena/biorobots.html*

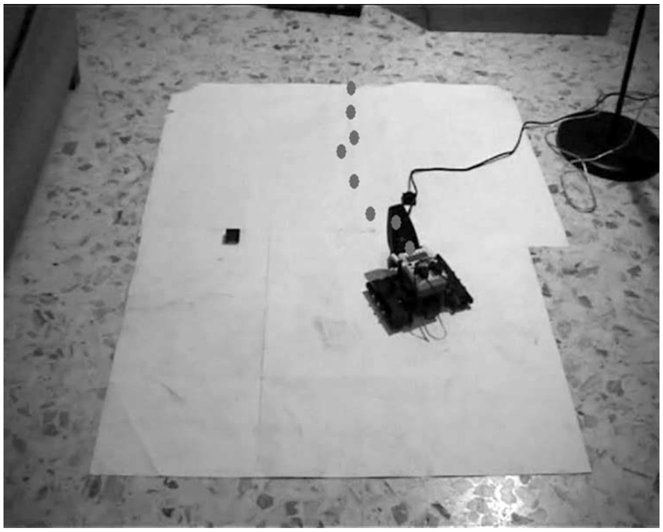

Fig. 8. Example of a typical trajectory of the robot while avoiding an obstacle. The control is based on the RD-CNN simplified algorithm, the camera is placed on the ceiling of the laboratory.

V. Implementation on RD-CNN Chip Prototype

A. Architecture of the Chip and Basic Processor Structure

The CACE1k prototype chip is a programmable mixed-signal array processor designed, fabricated and tested within a Spanish–Hungarian collaboration [35] under the framework of the DICTAM EU Project [34]. The current paper only discusses one of the possible applications of the chip. It consists of 32×32 identical cells (Fig. 11), with boundary conditions appropriate for CNN dynamics. There is also an input/output (I/O) interface, a timing and control unit, and a program memory. The interface is based on a serializing–deserializing analog multiplexor. The program memory is composed of 24 blocks of SRAM of 64 B of capacity, 1 kB dedicated to the analog program, and 0.5 kB to the logic program. In addition, the analog instructions and reference signals need to be transmitted to every cell in the network in the form of analog

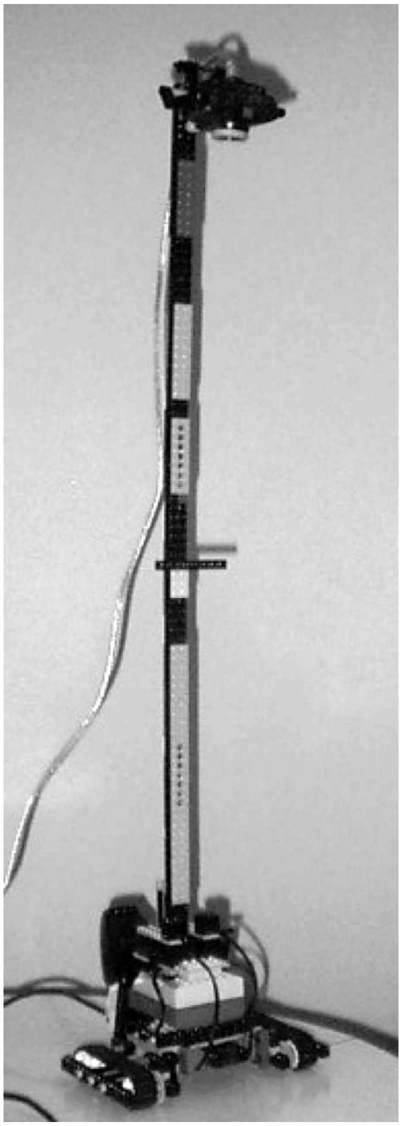

Fig. 9. Photo of the roving robot (with on-board camera).

voltages. Thus, a bank of D/A converters interfaces the analog program memory with the processing array. Distributing analog references across large distances within a chip is not a trivial task. Apart from the problems derived from electromagnetic interference, voltage drops in long metal lines carrying currents can be quite noticeable. Thus, signal buffering and low-resistance paths must be provided to avoid this. Finally, the timing unit is composed by an internal clock/counter and a set of finite state machines (FSMs) that generate the internal signals that enable the processes of images uploading/downloading and program memory accesses.

Each processing element of the mixed-signal programmable array incorporates two coupled continuous-time CNN cores [Fig. 12(a)]. Each core belongs to one of the two layers of the network. The synaptic connections between processing elements in both, the same or the other, layers are represented by arrows in the diagram. The basic processor also contains a programmable local logic unit (LLU), a local analog memory (LAM), and local logic memory (LLM) to store intermediate results. All the blocks in the cell communicate via an intracell data bus, which is multiplexed to the array I/O interface. Control bits and switch configuration are passed to the cell directly from the global programming unit. The internal structure of each CNN core is depicted in Fig. 12(b). They receive contributions from the rest of the processing nodes in the neighborhood which are summed and integrated in the state capacitor. The two layers differ in that the first layer has a scalable time constant, controlled by the appropriate binary code, while the second layer has a fixed time constant. The evolution of the state variable is also driven by self-feedback and by the feedforward action of the stored input and bias patterns. There is a voltage limiter for implementing the full signal range (FSR) CNN model [37]. The state variable is transmitted in voltage form to the synaptic blocks in the periphery of the cell, where weighted contributions to the neighbors are generated. There is also a current memory that will be employed for cancellation of the offset of the synaptic blocks. Initialization of the state, input and/or bias voltages is done through a mesh of multiplexing analog switches that connect to the cell's internal data bus. The CACE1K chip can be connected to the CNN Chip Prototyping and Development System (CCPS) platform designed in Budapest [39]–[41] for the ACE4K and ACE16K CNN chips. In this way, all the facilities already developed for the ACE family chip, can be used for the new one.

B. Prototype Chip Data

The prototype chip has been designed and fabricated in a 0.5 $\mu\mathrm{m}$ single-poly triple-metal CMOS technology [38]. Its dimensions are 9.27×8.45 sq. mm. (Fig. 13). The cell density achieved is 29.24 $\mathrm{cells/mm}^2$. The time constant of the layers is around 100 ns (unscaled). The programmable dynamics of the chip permit the observation of different phenomena of the type of propagation of waves, pattern generation, etc. More details on the CNN chip such as operating waveforms can be found in [42]. Table I summarizes the most relevant data of the prototype chip.

C. Results of Autowave Propagation and Robot Control Through the CACE1k Chip

In this section, some preliminary experimental results for the robot control obtained by using the CACE1k chip are shown. These results refer to the *robot centered perception* case; they are obtained by taking into account as initial condition the frame of Fig. 10. In particular, the results of the processing for the first frame of Fig. 10 are reported here. The picture taken from the camera corresponding to this frame is shown in Fig. 14. The image contains two obstacles. Each obstacle is represented a square composed of 4 pixels. However, the size of the CNN used in the two cases is slightly different: in fact, the image size (32×32 pixels) for the experiments on chip is slightly smaller. This is not a limitation of the approach for various reasons: first of all, the size is enough for experiments with the on-board camera, since the dynamically changing environment is not a limitation for the proposed approach; on the other hand, if wider frames are required, well-known consolidated tiling techniques allow higher resolution pictures; finally, wider CACE chips, possibly available in the future, will allow direct processing

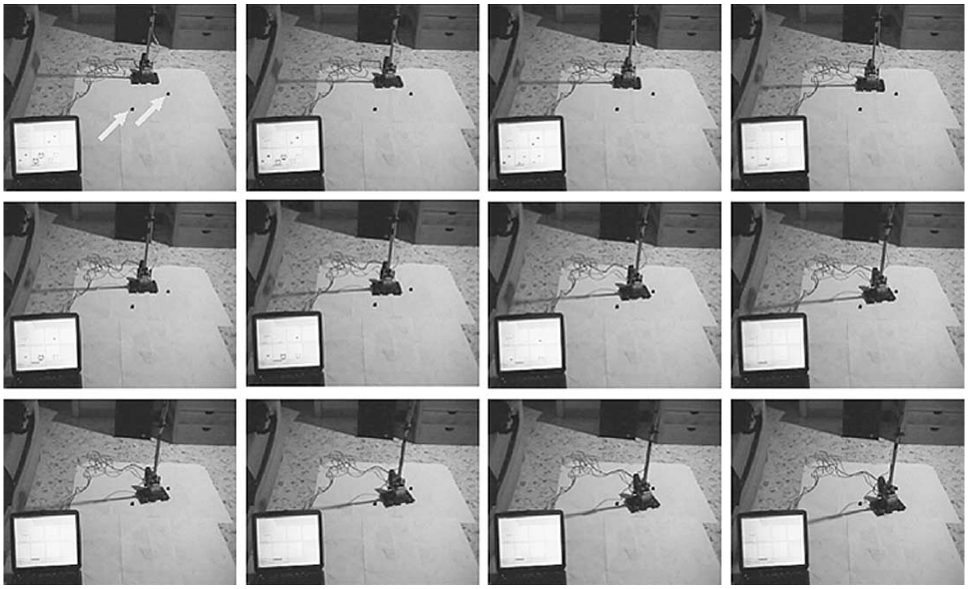

Fig. 10. Example of robot-centered navigation control based on RD-CNN. Several snapshot of a video are shown. The obstacle position is indicated in the first frame by two arrows.

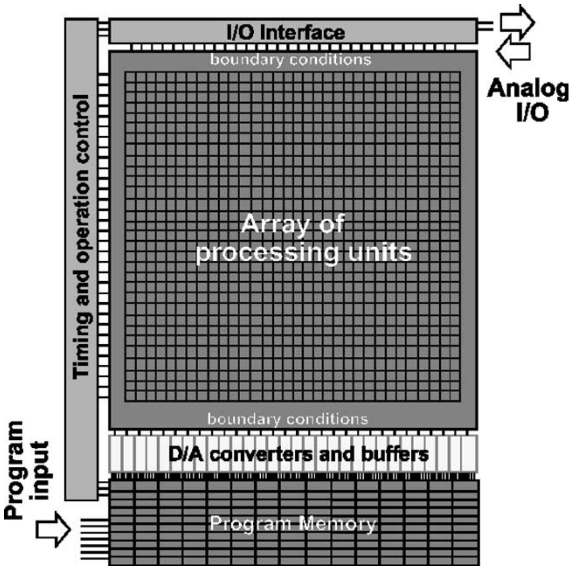

Fig. 11. Architecture of the CACE1k chip.

of larger images. The algorithm implemented involves the following steps.

- Input frames at each step are acquired by a web cam; they are 320×240 color images (see Fig. 14).
- Images are resized to 64×64, a dimension compatible with the 1L-CNN chip (ACE4k), taking the robot position fixed at the bottom center.
- Since obstacles are assumed to be 1-pixel wide black objects in the resized image, after a threshold operation an "isolated pixel extraction" template is applied, in order to filter out all the other noisy information in the environment, except the obstacles. The robot is also deleted, but its position is *a priori* known. Of course, other more sophisticated extraction routines could be efficiently applied, according to the hypotheses on the environment, exploiting the template algorithms already developed for the 1L-CNN chips.
- Some diffusion steps are also applied, to obtain dimensions useful to onset autowaves.
- A canvas resize, from 64×64 to 32×32, leaving the central lower part of the image, is then applied; this image represents the initial condition for the CACE1k processing.
- Autowave are generated and propagate on the CACE1k chip. The templates used to reproduce autowaves on chip, modified with respect to equation (3), enriched with inclusion of the diagonal diffusion, and scaled for on chip working, are the following:

$$A_{11} = \begin{pmatrix} 0.6 & 0.8 & 0.6 \\ 0.6 & -0.2 & 0.8 \\ 0.6 & 0.8 & 0.6 \end{pmatrix}$$

$$A_{12} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$A_{21} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -1.5 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$A_{22} = \begin{pmatrix} 0.6 & 0.8 & 0.6 \\ 0.8 & 0.1 & 0.8 \\ 0.6 & 0.8 & 0.6 \end{pmatrix} \quad (4)$$

  while bias and control templates are $i_1 = -0.7, i_1 = -2.9$, $\mathbf{B_1} = \mathbf{B_2} = \mathbf{0}$. This step is illustrated in Fig. 15(a), where, each frame is sampled at $T = 400$ ns, i.e., the elaboration time required by the CACE1k chip.
- Autowave generation from obstacles is processed by an AND operation with the robot pixels: this step is schematized in Fig. 15(c). As the first wavefront collides with the robot pixels, the corresponding steering command is sent to the robot. It makes a movement which continues until the next frame is processed and a new command is generated.

As can be noticed, the core of the algorithm lies in the on-chip autowave generation and propagation, used to drive the robot. The other parts of the algorithm are implemented on the

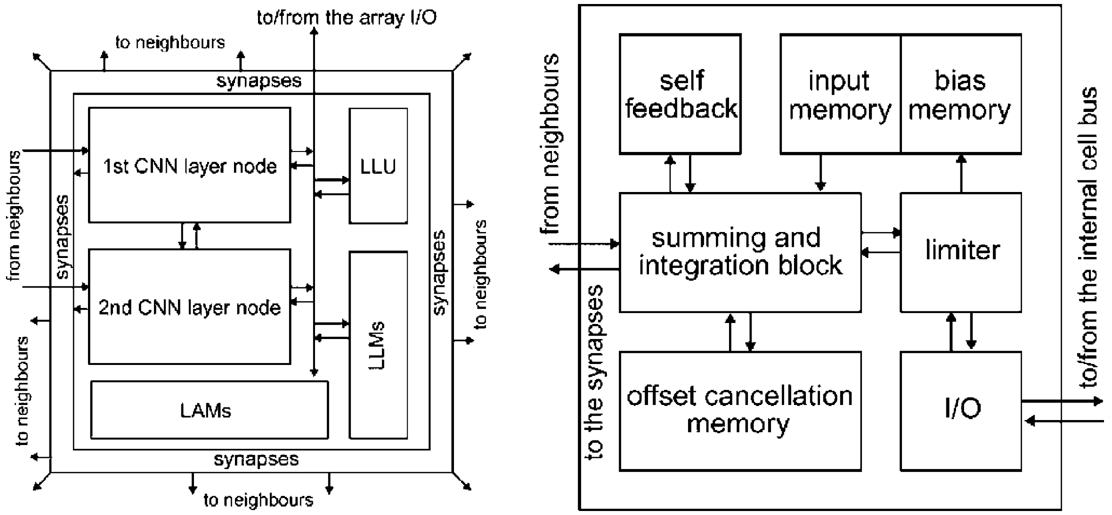

Fig. 12. Block scheme of a cell of the CACE1k chip.

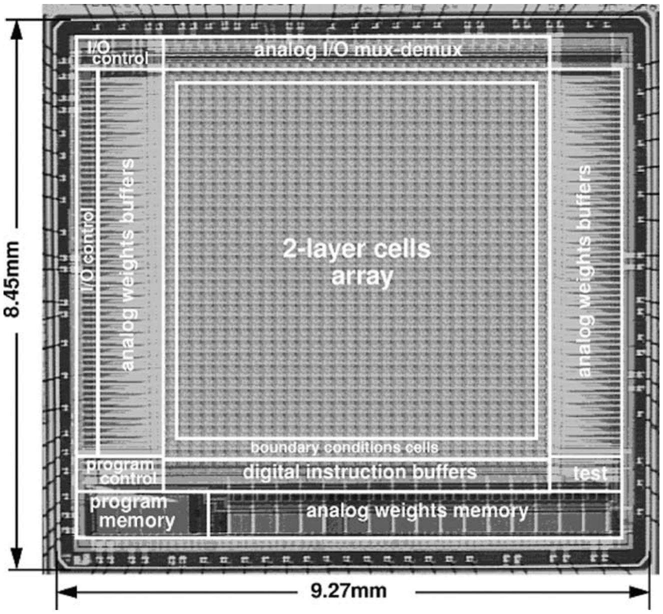

Fig. 13. Microphotograph of the prototype CACE1k chip.

TABLE I
CACE1K PROTOTYPE CHIP

Technology	$0.5\mu m$ CMOS 1-P 3-M
Number of cells	32×32
Die area	$9.27 \times 8.45 sq.mm.$
Die area (w/o pads)	$8.77 \times 7.94 sq.mm.$
Array area	$5.98 \times 5.93 sq.mm.$
Package	ceramic PGA - 100
Power supply voltage	$3.3V$
Logic ”0” / Logic ”1”	$0V/3.3V$
Accuracy on the weights	$8b$
Image samples resolution	$7-8b$
I/O rates	$10Ms/s$
CNN time constant	below $100ns$

ACE chip, with the help of a host PC. Of course this latter part can be optimized with respect to the particular information to draw from the environment. So, it is important to evaluate the time spent by the algorithm as regards the part running on the CACE1K chip. This involves, as just discussed, an autowave propagation plus some AND operations. Given the chip dimensions and considering the robot-centered case, the wavefront has to propagate, in the worst case, for about 30 pixels before colliding with the robot. According to the measures on the CACE1K, the autowaves propagation, for the template set used, involves 2 pixels each 400 ns. So, the whole autowave propagation in the worst case should spend about 6 μs before collision. Concurrently an AND operation between the wavefronts and the robot pixels has to be periodically performed. Given the robot dimensions, a suitable time interval has been assumed to be just 400 ns, each 2-pixel wave propagation. In the worst case, 15 AND operations are required. The time spent for an AND operation should by itself spend a few nanoseconds, but, since a transfer to digital signal processing on the CCPS is required for the image to be processed, the whole time grows up to 75 μs for each AND operation. The whole time spent by the logical operation is therefore about 1.1 ms, which practically represents the whole time spent by the algorithm. Incidentally, the computational burden of the whole algorithm is equivalent to a sequence of logical AND logical operations.

D. Remarks

As mentioned in the introduction, the main difference between the approach proposed and the potential field approach is that the whole environment information is perceived by the robot through a wave collision, that it uses to move within the arena. This methodology fully exploits the massive parallelism of analog processing. The core of the traditional potential field

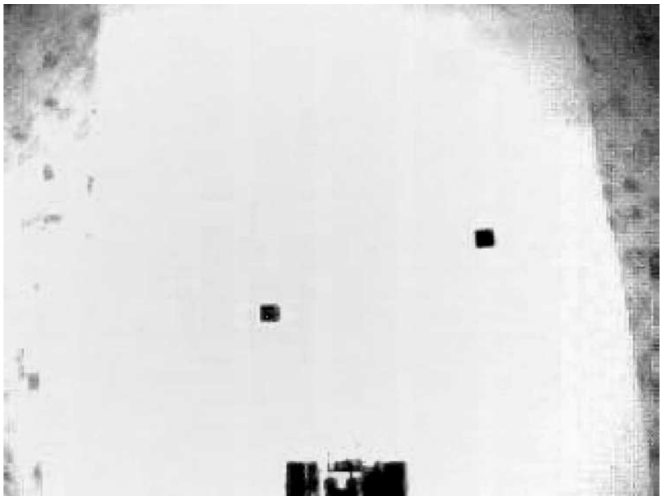

Fig. 14. Typical view from on the robot on-board camera, the high contrast allows to simplify the detection of obstacles.

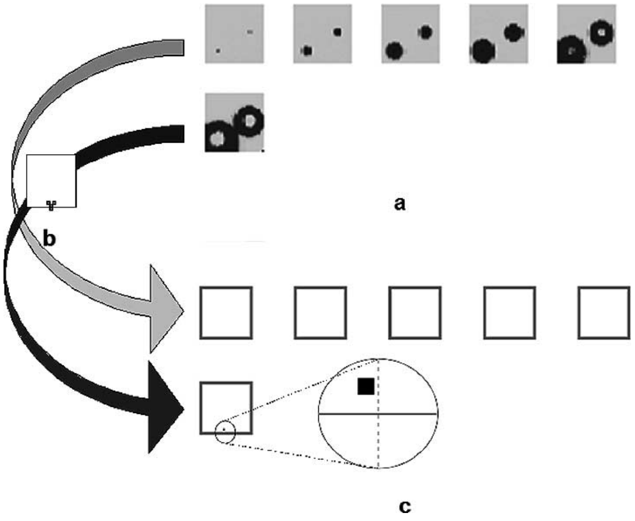

Fig. 15. CACE1K chip experiment: (a) Autowave propagation in an environment with two obstacles; frames are sampled at $T = 400$ ns. (b) Image containing the robot scheme. (c) Schematic representation of the results of the AND operation between the snapshots in (a) and image (b). The last image reveals a collision and a command is sent to the robot.

approach is the calculation, for each environment state, of the potential function for each point of the robot arena. Starting from these considerations, in case of moving obstacles on dynamically changing situations, the advantages of the approach proposed are apparent: in our case each obstacle/target is a wave source independently on its position. On the other hand, when handling with moving obstacles, the visual approach is traditionally seldom used for the drawbacks just outlined: other algorithms, using different, but simpler sensors (infrared, ultrasonic, etc), are implemented. However, with these approaches the characteristics of the objects detected cannot be gained, while the visual filtering capabilities of CNNs, joined to this new methodology, allow to embed very efficient object classification routines, which further enhance the performance of the approach. The algorithm presented is just a first version, which could be optimized to overcome some minor drawbacks. For example, an important issue to address is the ratio between the robot and the obstacles dimension. If this is not taken into account, the robot could collide with two obstacles when trying to pass in between, attracted by the target. In this case, the algorithm should be enriched by adding a simple routine which uses the wave generated by the first obstacle and colliding with the other, to calculate the distance between them. If this results below the robot dimension, a virtual obstacle can be positioned between the other two in order to eliminate that candidate trajectory. The presence of local minima traps the robot: These could arise in situations characterized by a high symmetry among the robot and the target-obstacles positions. Also, this condition could be overcome by adding a disturbance on these "static" trajectories, in analogy with the algorithm described in Section III.

VI. Conclusion

We have applied a novel paradigm of computing based on RD equations to real-time robot navigation in a complex, dynamically changing environment. The environment in which the robot moves is mapped onto an excitable medium where obstacles and the target generate autowaves. The waves generated by obstacles are interpreted by the robot as repulsive stimuli while wave generated by the target as attractive stimuli. The robot optimizes its trajectory employing spatio-temporal dynamics of "repulsive" and "attractive" events. Each element of the excitable medium acts as a very primitive computing device. The medium's sites take on ideally continuous values and the site interacts locally; therefore, the path planning problem is solved through an analog and massively parallel computation, allowing also for a real-time control of the robot's trajectory, also when high motion speeds are required to the robotic structure.

This universal paradigm of wave-based computation in nonlinear media can be implemented in various types of substrates. In this paper, we investigated how the path planning is implemented in liquid-phase chemical and CNN-based silicon experimental processors. We found that both chemical and silicon processors work equally well. Of course, other approaches could be used, such as for example, hybrid techniques, where RD dynamics in microarrays can be electrically controlled via arrays of integrated microelectrodes [43], [44]. The chemical processors offer a high degree of parallelism, fault tolerance and ease of implementation of parallel I/O, however they are limited by low speed of wavefront propagations. This large time constant of the process is overcome in the CNN-based fully silicon implementation.

With respect to traditional schemes for the computation of the path planning, the approach presented here is totally innovative since experimental massively parallel devices are implemented in "wetware" and hardware. Another advantage of the approach is the adaptability of the parallel computation to a vast class of optimization problems. Moreover, it is worth remarking that the paper presents one of the first applications of the novel CACE1k chip, that opens the way to new interesting real-time applications of the paradigm of emergent parallel computation.

Acknowledgment

The authors would like to thank Dr. L. O. Chua for stimulating the collaboration between the three research groups at the

University of Catania, Italy, University of the West of England, U.K., and University of Sevilla, Spain.

References

[1] O. Takahashi and R. J. Schilling, "Motion planning in a plane using generalized Voronoi diagram," *IEEE Trans. Robot. Automat.*, vol. 5, pp. 143–150, 1989.

[2] C. S. Tzafestas and S. G. Tzafestas, "Recent algorithms for fuzzy and neurofuzzy path planning and navigation of autonomous mobile robots," *Syst. Sci.*, vol. 25, pp. 25–39, 1999.

[3] R. Murphy, *An Introduction to AI Robotics*. Cambridge, MA: MIT Press, 2000.

[4] C. J. Fourie, "Intelligent path planning for a mobile robot using a potential field algorithm," in *Proc. 29th Int. Symp. Robotics. Advanced Robotics: Beyond 2000, DMG Business Media*, Redhill, U.K., 1998, pp. 221–224.

[5] E. S. H. Hou and D. Zheng, "Mobile robot path planning based on hierarchical hexagonal decomposition and artificial potential fields," *J. Robot. Syst.*, vol. 11, pp. 605–614, 1994.

[6] B. Hussien and R. W. McLaren, "Real-time robot path planning using the potential function method," *Autom. Constr.*, vol. 2, pp. 241–250, 1993.

[7] G. F. Marshall and L. Tarassenko, "Robot path planning using VLSI resistive grids," *Proc. IEE, Vision, Image, Signal Processing*, vol. 141, pp. 267–272, 1994.

[8] M. R. Stan, W. P. Burleson, C. I. Connolly, and R. A. Grupen, "Analog VLSI for robot path planning," *J. VLSI Signal Processing*, vol. 8, pp. 61–73, 1994.

[9] Y. F. Wang and G. S. Chirikjian, "A new potential field method for robot path planning," in *Proc. IEEE Int. Robot. Automat. Conf.*, San Francisco, CA, 2000, pp. 977–982.

[10] G. K. Schmidt and K. Azarm, "Mobile robot path planning and execution based on a diffusion equation strategy," *Adv. Robot.*, vol. 7, pp. 479–490, 1993.

[11] C. Louste and A. Liegeois, "Near optimal robust path planning for mobile robots: The viscous fluid method with friction," *J. Intell. Robot. Syst.*, vol. 27, pp. 99–112, 2000.

[12] O. Steinbock, A. Tóth, and K. Showalter, "Navigating complex labyrinths: Optimal paths from chemical waves," *Sci.*, vol. 267, pp. 868–871, 1995.

[13] K. Agladze, N. Magome, R. Aliev, T. Yamaguchi, and K. Yoshikawa, "Finding the optimal path with the aid of chemical wave," *Phys. D*, vol. 106, pp. 247–254, 1997.

[14] A. Adamatzky and B. De Lacy Costello, "Collision-free path planning in the Belousov–Zhabotinsky medium assisted by a cellular automaton," in *Naturwissenschaften 89*, vol. 10, 2002, pp. 474–478.

[15] N. G. Rambidi and D. Yakovenchuk, "Chemical reaction-diffusion implementation of finding the shortest paths in a labyrinth," *Phys. Rev. E*, vol. 63, pp. 3–7, 2001.

[16] A. Adamatzky, B. De Lacy Costello, C. Melhuish, and N. Ratcliffe, "Experimental reaction–diffusion chemical processors for robot path planning," *J. Intell. Robt. Syst.*, vol. 37, pp. 233–249, 2003.

[17] A. Adamatzky and B. De Lacy Costello, "Reaction-diffusion path planning in a hybrid chemical and cellular-automaton processor," *Chaos, Solitons, Fractals*, vol. 16, pp. 727–736, 2003.

[18] L. O. Chua and L. Yang, "Cellular neural networks: Theory," *IEEE Trans. Circuits Syst. I*, vol. 35, pp. 1257–1272, Oct. 1988.

[19] ——, "Cellular neural networks: Applications," *IEEE Trans. Circuits Syst. I*, vol. 35, pp. 1273–1290, Oct. 1988.

[20] L. O. Chua and T. Roska, "The CNN paradigm," *IEEE Trans. Circuits Syst. I*, vol. 40, pp. 147–156, Mar. 1993.

[21] R. Field and A. T. Winfree, "Travelling waves of chemical activity in the Zaikin–Zhabotinsky–Winfree reagent," *J. Chem. Education*, vol. 56, p. 754, 1979.

[22] J. D. Murray, *Mathematical Biology*. Berlin, Germany: Springer-Verlag, 1989.

[23] *Special Issue on Nonlinear Waves, Patterns, and Spatio-Temporal Chaos, IEEE Trans. Circuits Syst. I*, Oct. 1995, vol. 42.

[24] P. Arena, L. Fortuna, and M. Branciforte, "Reaction-diffusion CNN algorithms to generate and control artificial locomotion," *IEEE Trans. Circuits Syst. I*, vol. 46, pp. 253–260, Feb. 1999.

[25] A. Adamatzky, *Computing in Nonlinear Media and Automata Collectives*. Bristol, U.K.: IoP Publishing, 2001.

[26] *Self-Organization: Autowaves and Structures Far from Equilibrium*, V. I. Krinsky, Ed., Springer-Verlag, Berlin, Germany, 1984.

[27] P. Arena, S. Baglio, L. Fortuna, and G. Manganaro, "Self-organization in a two-layer CNN," *IEEE Trans. Circuits Syst. I*, vol. 45, pp. 157–162, Feb. 1998.

[28] L. O. Chua, "CNN: A paradigm for complexity," in *Visions of Nonlinear Science in the 21st Century*. ser. A, J. L. Huertas, W. K. Chen, and R. N. Madan, Eds. Singapore: World Scientific, 1999, vol. 26.

[29] P. Julian, A. Desages, and B. D'Amico, "Orthonormal high-level canonical PWL functions with applications to model reduction," *IEEE Trans Circuits Syst. I*, vol. 47, pp. 702–712, May 2000.

[30] P. Arena, L. Fortuna, and M. Frasca, "Multi-template approach to realize central pattern generators for artificial locomotion control," *Int. J. Circuit Theory Applicat.*, vol. 30, pp. 441–458, 2002.

[31] B. P. J. De Lacy Costello and A. I. Adamatzky, "On multi-tasking in parallel chemical processors: Experimental findings," *Int. J. Bifurc. Chaos*, vol. 13, pp. 521–533, 2003.

[32] P. Arena, L. Fortuna, and L. Occhipinti, "A CNN algorithm for real time analysis of DNA microarrays," *IEEE Trans Circuits Syst. I*, vol. 49, pp. 335–340, Mar. 2002.

[33] T. Roska, L. Kèk, L. Nemes, A. Zaràndy, M. Brendel, and P. Szolgay, Eds., "CNN software library (Template and Algorithms)," Hungarian Academy of Sciences, Computer and Automation Institute, Budapest, Hungary, DNS-1-1998, (CADET-15), 7.2 ed., 1998.

[34] IMSE-CNM. Dynamic image computing using tera-speed analogic visual microprocessors. EU founded Project. [Online]. Available: http://www.imse.cnm.es/Proyectos/dictam/

[35] T. Roska and A. Rodriguez-Vasquez, "Toward visual microprocessors," *Proc. IEEE*, vol. 90, pp. 1244–1257, July 2002.

[36] P. Arena, A. Basile, M. Bucolo, L. Fortuna, and A. Virzì, "A bio-inspired visual feedback locomotion control based on CNN universal machine," *J. Circuits, Syst., Comput.*, to be published.

[37] S. Espejo, R. Carmona, R. Domínguez-Castro, and A. Rodríguez-Vázquez, "A VLSI oriented continuous-time CNN model," *Int. J. Circuit Theory Applicat.*, vol. 24, no. 3, pp. 341–356, 1996.

[38] R. Carmona, F. Jiménez-Garrido, R. Domínguez-Castro, S. Espejo, and A. Rodríguez-Vázquez, "A CMOS analog parallel array processor chip with programmable dynamics for early vision tasks," in *Proc. 28th Eur. Solid-State Circuits Conf. (ESSCIRC'02)*, Florence, Italy, Sept. 2002, pp. 371–374.

[39] A. Zaràndy, T. Roska, P. Szolgay, S. Zöld, P. Földesy, and I. Petràs, "CNN chip prototyping and development systems," in *Proc. Eur. Conf. Circuit Theory and Design s*, Stresa, Italy, 1999, pp. 69–81.

[40] I. Szatmàri, A. Zaràndy, P. Földesy, and L. Kèk, "An analogic CNN engine board with the 64 × 64 analog CNN-UM chip," in *Proc. Int. Symp. Circuits Syst.*, Geneva, Switzerland, 2000.

[41] A. Zaràndy, "ACE box: High-performance visual compute based on the ACE4k analogic array processor chip," in *Proc. Eur. Conf. Circuit Theory Design*, Espoo, Finland, Aug. 28–31, 2001, pp. 361–364.

[42] R. Carmona, F. Jiménez-Garrido, R. Domínguez-Castro, S. Espejo, and A. Rodríguez-Vázquez, "Bio-inspired analog VLSI design realizes programmable complex spatio-temporal dynamics on a single chip," in *Proc. 2002 Conf. Design Automation and Test in Europe*, Paris, France, Mar. 2002, ISBN 0-7695-1471-5, pp. 362–366.

[43] M. Hiratsuka, T. Aoki, H. Morimitsu, and T. Higuchi, "Implementation of reaction-diffusion cellular automata," *IEEE Trans. Circuits Syst. I*, vol. 49, pp. 10–16, Jan. 2002.

[44] ——, "Implementation of a Redox microarray: An experimental model for future nanoscale biomolecular computing using integrated circuits," *Proc. IEE Nanobiotechnol.*, vol. 150, no. 1, pp. 9–14, 2003.

Andrew Adamatzky is a Reader in Computing, Faculty of Computing, Engineering and Mathematical Sciences, University of the West of England, Bristol, U.K.

He performs research in unconventional computing, cellular automata, massive parallel computing, applied mathematics, theoretical computer science, collective intelligence, and robotics. He has published two books, over 100 papers, and put together few monographs and journal issues.

Paolo Arena (S'93–M'97–SM'01) received the degree in electronic engineering and the Ph.D. degree in electrical engineering from The University of Catania, Catania, Italy, in 1990 and 1994, respectively.

He is currently an Associate Professor of System Theory, University of Catania. He has published more than 130 technical papers and several international patents. His research interests include adaptive and learning systems, nonlinear systems, neural networks, cellular neural networks, collective behavior in living and artificial neural structures, and bio-image analysis.

Dr. Arena serves as an Associate Editor of the IEEE Transactions on Circuits and Systems—I: Regular Papers.

Adriano Basile (S'02) received the degree in computer science engineering working on cellular neural networks applied to biomedical images, from the University of Catania, Catania, Italy, in 2000, and is currently working toward the Ph.D. degree in electronic and control engineering at the same university.

He works for the Dynamic Image Computing using Tera-speed Analogic Visual Microprocessors (DICTAM) Project, University of Catania. He is with the Robotics and Automation Group, ST Microelectronics, Catania, Italy. His scientific interests include cellular neural networks and their applications to image, bio-image, and video processing and complex dynamics

Ricardo Carmona-Galán (M'99) received the degrees of Licenciado and Doctor (Ph.D.) in physics, in the speciality of electronics, from the University of Seville, Seville, Spain, in 1993 and 2002, respectively.

From 1994 to 1996, he worked at the National Center for Microelectronics, Seville, Spain, funded by an IBERDROLA S. A Grant. From July 1996 to June 1998, he was a Research Assistant in the Electronics Research Laboratory, Department of Electrical Engineering and Computer Sciences, University of California at Berkeley. He is currently with the Department of Analog Design, Institute of Microelectronics of Seville (IMSE), Centro Nacional de Microelectrónica (CNM-CSIC), Seville, Spain. Since October 1999, he is an Assistant Professor in the Department of Electronics and Electromagnetism, School of Engineering, University of Seville, where he teaches "Circuit Analysis and Synthesis" and "Circuit Synthesis Laboratory" for the degree of Telecommunication Engineer. His main areas of interest are linear and nonlinear analog and mixed-signal integrated circuits, in particular, the design and VLSI implementation of cellular neural networks and analog memory devices for real-time image processing and vision chips.

Dr. Benjamin De Lacy Costello is a Research Fellow, Faculty of Applied Sciences, University of the West of England, Bristol, U.K.

He has a highly interdisciplinary research background having worked on a number of industrial, European Commission and Research Council funded projects. His research interests are in the materials science area and he has experience of the synthesis and analysis of polymer, ceramic, and composite thin films. The major focus of this work was the production of advanced materials for biological/environmental sensing applications. More recently, his research interests have been focused in the area of nonlinear chemical reactions including the Belousov–Zhabotinsky reaction and particularly the coupling and control of these reactions toward tangible outcomes in computation and engineering.

Luigi Fortuna (M'90–SM'99–F'00) is a Full Professor of System Theory at the University of Catania, Catania, Italy, since 1994.

He has published more than 280 technical papers ad is co-author of six books including *Cellular Neural Networks* (New York, Springer, 1999). He holds several U.S. patents. His scientific interests include nonlinear science and complexity, chaos, cellular neural networks with applications in bioengineering.

Mattia Frasca (S'03–M'04) was born in Siracusa, Italy, in 1976. He received the graduate degree in electronics engineering and the Ph.D. degree in electronics and automation engineering from the University of Catania, Catania, Italy, in 2000 and 2003, respectively.

Currently he is a Lecturer, University of Catania. His scientific interests include nonlinear systems and chaos, cellular neural networks, and bio-inspired robotics.

Angel Rodríguez-Vázquez (M'80–SM'95–F'96) received the Liceniado en físcia electrónica and the Ph.D. degrees from the University of Seville, Seville, Spain, in 1977 and 1983, respectively.

He is a Professor of Electronics in the Department of Electronics and Electromagnetism, University of Seville. He is also a Member of the Research Staff at the Institute of Microelectronics of Seville, Centro Nacional de Microelectrónica (IMSE-CNM), Seville, Spain, where he heads a research group on Analog and Mixed-Signal Integrated Circuits. His research interests are in the design of analog front-ends for mixed-signal circuits and systems-on-chip, telecom circuits, CMOS imagers and vision chips, sensory-processing-actuating systems-on-chip, and bio-inspired integrated circuits. On these topics, he has published seven books, 36 book chapters in other books, approximately 100 journal papers, and about 300 conference papers. He is also a member of the editorial staff of the *International Journal on Circuit Theory and Applications*, *Analog Integrated Circuits*, and *Signal Processing Journal*.

Dr. Rodríguez-Vázquez was co-recipient of the 1995 Guillemin–Cauer Award of the IEEE Circuits and Systems Society. In 1992, he received the Young Scientist Award of the Seville Academy of Science. In 1996, he was elected Fellow of the IEEE for contributions to the design and applications of analog/digital nonlinear ICs. He served as an Associate Editor of the IEEE Transactions on Circuits and Systems—I: Fundamental Theory and Applications from 1993 to 1995. Currently, he is an Associate Editor for IEEE Transactions on Circuits and Systems—II: Express Briefs. He was a Guest Editor of the IEEE Transactions on Circuits and Systems—I: Regular Papers Special Issue on "Advances on Analog-to-Digital and Digital-to-Analog converters."