

# Compact Internal Representation as a Protocognitive Scheme for Robots in Dynamic Environments

Jose A. Villacorta-Atienza<sup>1</sup>, Luis Salas<sup>2</sup>, Luis Alba<sup>2</sup>, Manuel G. Velarde<sup>3</sup>, Valeri A. Makarov<sup>1</sup>

1. School of Mathematics. Universidad Complutense de Madrid. Madrid, Spain.
2. AnaFocus. Seville, Spain.
3. Instituto Pluridisciplinar. Universidad Complutense de Madrid. Madrid, Spain.

## ABSTRACT

Animals for surviving have developed cognitive abilities allowing them an abstract representation of the environment. This Internal Representation (IR) could contain a huge amount of information concerning the evolution and interactions of the elements in their surroundings. The complexity of this information should be enough to ensure the maximum fidelity in the representation of those aspects of the environment critical for the agent, but not so high to prevent the management of the IR in terms of neural processes, i.e. storing, retrieving, etc. One of the most subtle points is the inclusion of temporal information, necessary in IRs of dynamic environments. This temporal information basically introduces the environmental information for each moment, so the information required to generate the IR would eventually be increased dramatically. The inclusion of this temporal information in biological neural processes remains an open question. In this work we propose a new IR, the Compact Internal Representation (CIR), based on the compaction of spatiotemporal information into only space, leading to a stable structure (with no temporal dimension) suitable to be the base for complex cognitive processes, as memory or learning. The Compact Internal Representation is especially appropriate for be implemented in autonomous robots because it provides global strategies for the interaction with real environments (roving robots, manipulators, etc.). This paper presents the mathematical basis of CIR hardware implementation in the context of navigation in dynamic environments. The aim of such implementation is the obtaining of free-collision trajectories under the requirements of an optimal performance by means of a fast and accurate process.

*Keywords:* internal representation, decision making, autonomous robots, cognition, hardware implementation.

## Introduction

The evolution has provided to animals a complete set of cognitive abilities which allow them for dealing with their environments to ensure their survival. These mental capabilities were developed according the necessities of each animal, from less evolved nervous systems to the most complex and abstract processes in the human brain. Nonetheless, despite the differences, all cases share a common point: they should deal with time-evolving environments, i.e. with situations that change in time. Regardless the processes supporting the perception and processing of the surrounding spatial information, the inclusion of the temporal features, i.e. when the *changes* in the environment are considered, introduces a dramatic

increasing in the amount of information which should be processed; in short, it introduces an extra dimension (time) in the whole cognitive process.

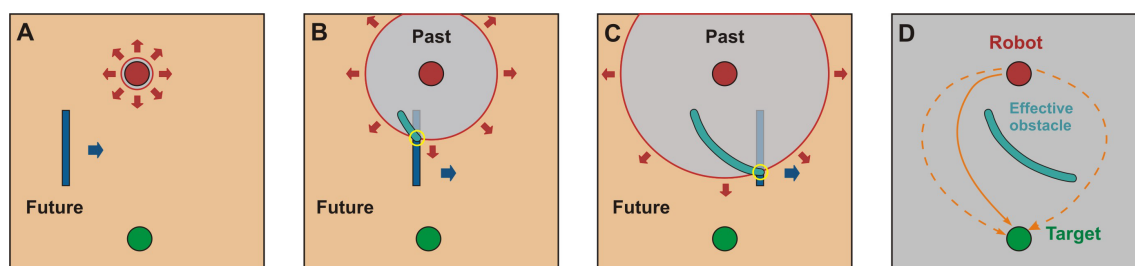
Real nervous systems have developed different strategies to deal with this situation, as spatiotemporal integration<sup>1,2</sup>, time-frequency domain<sup>3</sup>, etc. Nonetheless these processes take place at small scale (neurons or microcircuits) so they are just a piece of the puzzle. More intriguing are the mechanisms underlying the processing of the whole sensory information for creating the *cognition* of the situation in which the animal is embedded. The concept of cognition is controversial<sup>4</sup> and a deep discussion on this topic is beyond the scope of this work, however in order to proceed rigorously we consider the concept of *protocognition*. As introduced previously by the authors<sup>5</sup> protocognition refers to the set of processing abilities allowing to a natural or artificial agent for understanding its environment, creating motor strategies to deal with it. In the context of this work the agent will understand its situation when it is able to predict both the evolution of its environment and the *consequences* of this evolution over itself. This knowledge would provide to the agent the necessary background for generating proper motor actions to fulfill the required objective (reach a safe location, avoid collisions, capture a specific object, etc.).

### **The Compact Internal Representation**

Recently the authors have proposed a novel protocognitive paradigm for dynamic environments that solves the problem of the massive influx of sensory information by selecting only that spatiotemporal information critical for the agent and arranging it properly<sup>6</sup>. This paradigm is based on the concept of *internal representation*, an abstraction of the world constructed by the agent from its sensory perceptions. In a first approach the internal representation of a static environment could be considered as a matching between the external objects and their abstract representations in the agent's 'mind', i.e. a simple copy. The problem arises when this environment changes in time; then the objects making up the agent's world could change their status (position, orientation, physical features, etc.) so an environment abstract reproduction for each time would be neither feasible nor reliable in terms of information processing, storing and retrieving. To circumvent these constraints we have proposed a new type of internal representation, the Compact Internal Representation or CIR. The CIR reflects the spatiotemporal features of a dynamic environment in a static (stable) state of a neural network. Conceptually spiking and according with the protocognition requirements, this internal representation is generated by: 1) predicting the evolution of the environment, 2) considering all possible interactions between the agent and its environment, 3) extracting from this time stream those events critical for the agent and 4) projecting them univocally onto the space<sup>6</sup>. As previously remarked, the final aim of the CIR as a protocognitive process is to provide motor strategies to deal with the represented situation; in this sense the CIR will contain an infinite set of possible strategies to be followed by the agent. Therefore in presence of a complex time-evolving situation CIR is a global decision making process allowing infinite possible options or solutions to this situation from a single internal representation with minimum resources.

One of the natural contexts for protocognitive Compact Internal Representation is the navigation in realistic environments, where targets must be reached avoiding collisions against moving obstacles. In these circumstances the realization of the CIR is aimed to the object movement, so the prediction of the agent's external world evolution is referred to the forecast of the trajectories of such objects. Then the critical interactions between agent and environment will be the possible collisions. The CIR is then generated by obtaining the collision points in the space-time and projecting them into their spatial position by means of the causality principle. Figure 1 illustrates the creation process of the CIR for an

agent (red circle) in a square arena which should reach a target (green circle) avoiding a vertical bar moving from left to right (blue rectangle). CIR is based on mentally considering all possible agent positions for each time, represented by the expanding red circumference. This wavefront corresponds to the virtual present of the agent, whereas the virtual past (visited locations) and future (positions to be reached) match with the enclosed grey and orange areas respectively. The critical events for the agent will be the potential collisions with the moving vertical bar, which will correspond to those locations where agent and obstacle could coincide. Therefore these positions will arise from the matching between the expanding wavefront (possible agent positions) and the obstacle, represented by the yellow circle. According to the causality principle none of future events could affect to the past, so once the critical event is located in the space (yellow circle) and time (when that happened) it is projected only into the space by ‘freezing’ this location, because during the posterior evolution it will be enclosed by the wavefront and thus it will belong to the past, i.e. it will not change. This process leads to a static set of points denoting those positions where the agent could collide with the obstacles (vertical bar in this case). This set will be named *effective obstacle* and its avoidance by the agent will ensure a free-collision navigation, so an infinite number of possible strategies exist for safely reaching the target from the initial position (orange trajectories, where continuous one represents the shortest path). This static representation is the Compact Internal Representation (Fig. 1.D).



**Figure 1.** Compact Internal Representation process for an agent (red circle) which must reach the target (green circle) avoiding the moving obstacle (vertical blue bar). The agent mentally evaluates the evolution of the obstacle (by predicting its trajectory) and its own possible actions, represented by the propagating wavefront (red expanding circumference). This wavefront represents the virtual present of the agent, the gray region enclosed is the virtual past and the external area corresponds to the agent virtual future. Where moving obstacle and wavefront match a static effective obstacle is created, contained in the past. Once the whole arena has been covered the set of critical events for the agent will be represented by the effective obstacles (dangerous positions) and the CIR has been generated. The CIR contains infinite possible pathways from the agent's initial position to the target just by avoiding the effective obstacles; by following anyone of these trajectories the agent will prevent the collision against the moving obstacle.

### Mathematical model of Compact Internal Representation

The protocognitive Compact Internal Representation is modeled by two coupled neural networks<sup>6</sup>. A recurrent neural network (RNN) processes the sensory information about the environment for predicting the trajectories of the moving objects (obstacles and targets), whereas the Causal Neural Network (CNN) deals with the generation of the CIR according to the process detailed in the previous section. The process of the CIR generation is made up two concurrent phases: an active diffusion, the Wave Regime, that creates the propagating wavefront representing the possible agent's positions in each moment, and a passive diffusion, the Diffusion Regime, responsible for defining the final state of those neurons

progressively enclosed into the ‘past’ region. Mathematically both processes could be described by the reaction diffusion non linear system implemented in a neural network:

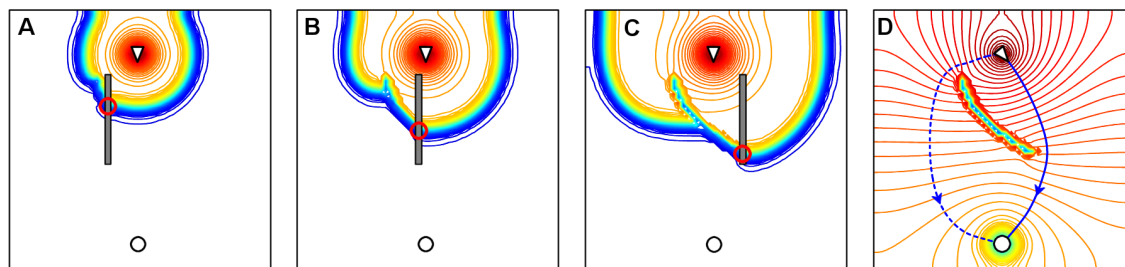
$$\begin{aligned}\dot{r}_{ij} &= q_{ij} \left( H(r_{ij}) [f(r_{ij}) - v_{ij}] + d \Delta r_{ij} - r_{ij} p_{ij} \right) \\ \dot{v}_{ij} &= (r_{ij} - 7v_{ij} - 2) / 25\end{aligned}\quad (1)$$

where the indices  $i$  and  $j$  stand for the neurons in the lattice, and  $r_{ij}(t)$  and  $v_{ij}(t)$  are the variables for the temporal evolution of the  $ij$  neuron. In particular the final values of variable  $r(t)$  for the whole network will describe the CIR. The function  $f(r_{ij})$  is a cubic non linearity  $(f(r) = (-r^3 + 4r^2 - 2r - 2) / 7)$  controlling the evolution of the wavefront. The function  $H(r)$  controls the transition between Wave and Diffusion Regimes: when the state of the  $ij$  neuron is under a threshold value,  $r_{ij}(t) \leq r_h$ , then  $H(r_{ij}) = 1$  and that neuron is subject to the active diffusion, but when  $r_{ij}(t) > r_h$  then  $H(r_{ij}) = 0$  and the neuron state evolves according passive diffusion.  $\Delta r_{ij} = r_{i+1,j} + r_{i-1,j} + r_{i,j+1} + r_{i,j-1} - 4r_{i,j}$  denotes the Laplace discrete operator describing the diffusion in the network. The target is represented by the reaction term  $r_{ij} p_{ij}$ , where the binary variable  $p_{ij}(t)$  is equal to 1 if the  $ij$  neuron is occupied by a target and 0 if not. Finally the influence of effective obstacles is represented by the binary variable  $q_{ij}(t)$ , equal to 1 for  $t < t_*$  and equal to 0 for  $t \geq t_*$ , where at time  $t_*$  some of the moving obstacles and the wavefront have match in the  $ij$  cell<sup>6</sup>.

The Compact Internal Representation for the situation illustrated in figure 1 has been obtained by means of equation (1) in figure 2, where the value of the variable  $r(t)$  for each element of the lattice is represented (60 x 60 network). Fig. 2.A shows the initial situation, where the agent, triangle located at the top of the square arena should reach the immobile target, circle at the bottom, avoiding the collision against the moving vertical bar. Figures 2.B and C depict the evolution of the propagating wavefront and how at the points where the wavefront touches the obstacle an effective obstacle is created by freezing the state of the corresponding cell (red circle). It can be observed the concurrence of Wave and Diffusion Regimes: the Wave Regime is represented by the wavefront and in the region enclosed by that the Diffusion Regime takes place. Once the effective obstacle is created and the Wave Regime ceases only passive diffusion occurs and a gradient-type profile appears, from the agent’s position to the target. The flow lines of this profile will be the trajectories to be followed by the agent to navigate without collisions towards the target (where the agent’s velocity is related with the propagation velocity of the wavefront). Figure 2.D shows the CIR solving the required situation: the arrangement of lattice final values  $\{r_{ij}\}$  contains the effective obstacles as well as the diffusive profile with the infinite possible strategies (trajectories) for the agent (the solid one denotes the shortest path).

It should be observed that Compact Internal Representation is a static structure, i.e. it does not change with time, although it contains the critical information about temporal evolution of the agent’s environment. Therefore the processing of a realistic dynamic situation in which the agent is involved is reduced by compacting the massive information perceived from the external world into a well-structured final stable state of a neural network. This fact makes the CIR a suitable solution for supporting protocognition in animals as well as in artificial agents<sup>5</sup>. After the CIR has been created the agent just have to choose one strategy among the infinite choices provided by the CIR to solve the problem. This selection will be based on the agent’s *motivation*, a crucial point when realistic autonomous agents are considered. Basically motivation arises from the competition among diverse concurrent processes involved in the survival of the agent. For instance, in terms of an autonomous robot navigating in a real

terrain, its motivation could be the result of the time to accomplish the mission, the type of terrain, the state of engines and the battery level. According to the weight of each one of these circumstances the motivation could change and thus the strategy or trajectory to be adopted, e.g. a low battery level could lead to select the shortest pathway even though it could compromise the mission, or a poor state of engines could advise a longer, but easy-terrain, trajectory. Nonetheless, for the sake of simplicity, in the simulations of the case illustrated in Fig. 2 and diverse realistic cases (videos can be watched at: <http://www.mat.ucm.es/~vmakarov/IRNN.html>) the movement of the agent has been described according to the shortest trajectory.



**Figure 2.** Compact Internal Representation of the situation illustrated in figure 1 and generated by solving the equation 1 (see text for details).

### Numerical solution of the CIR model and its hardware implementation

The Compact Internal Representation model described by equation 1 is structured according to the properties of a neural network thus it is particularly suitable for being implemented in autonomous robots as a reliable and robust decision making process, as well as basis for a more complex cognitive behaviors as memory or learning<sup>5</sup>. From a functional point of view the critical point for implementing the CIR in robots is the optimization of the numerical algorithms to solve the proposed mathematical model which can be properly realized in a hardware platform.

The CIR illustrated in figure 2 has been obtained by solving the equation 1 and subsequent by means of the standard Runge-Kutta method for solving non linear differential equation systems<sup>7</sup>. This algorithm was programmed in the mathematical software Matlab (v. R2010a, The MathWorks, Inc.) and solved in a PC under Windows Vista OS. The first approach to the CIR hardware implementation could be to directly transcribe the Runge-Kutta method to the hardware language; however the success of these options strongly depends on the hardware platform because its features should optimally fit the method requirements. Runge-Kutta algorithm is a fast and reliable method for numerically solving differential equations but it has a significant cost in terms of processing blocks and memory resources<sup>7</sup>.

The hardware platform selected for implementing the Compact Internal Representation is a chipset based on Field Programmable Gate Array or FPGA<sup>8</sup>. FPGA allows for flexible and fast processing from the proper arrangement of simple logic blocks reproducing the dynamics of the system described by eq. 1. The suitability of FPGAs lays on their programmability, which allows for constructing nearly ‘on route’ the hardware design by connecting the chipset block by block. Even though this hardware is slower and more power-consuming than other systems as Application-Specific Integrated Circuit or ASIC, it is the

proper platform for developing a ‘prototype’; once tested the reliability and performance of the design it could be implemented in a more suitable and definitive chipset.

The simplicity of the FPGA processing blocks and the limitation of the required resources, particularly in the available memory, make difficult the hardware implementation of the Runge-Kutta method as the mathematical procedure to generate the Compact Internal Representation. For that reason the numerical solution of Eq. 1 must be attained by a more suitable algorithm based on a simpler structure. In this work we propose a method for solving the non linear reaction-diffusion system in Eq. 1 based on the Euler method by combining the forward and backward versions according to the different parts of the systems. In order to properly present this algorithm, let us consider the general differential equation

$$\frac{dx(t)}{dt} = g(x(t)) \quad (2)$$

then it could be solved by approximating the differential operator by its discrete simplest approach:

$$\left. \frac{dx(t)}{dt} \right|_{t=t_n} \approx \frac{x(t_{n+1}) - x(t_n)}{h} \quad (3)$$

where variable  $t$  has been discretized as the set  $\{t_n\}_{n=1}^T$  and  $h = t_{n+1} - t_n \quad \forall n \in \{1, \dots, N\}$ . From there the differential equation (2) could be approximated by:

$$\frac{x(t_{n+1}) - x(t_n)}{h} = g(x(t_n)) \Rightarrow x(t_{n+1}) = x(t_n) + h g(x(t_n)). \quad (4)$$

This first approach is named *Euler forward method* and requires a small  $h$  value, and therefore many steps, to solve the given equation with a good accuracy. Nevertheless each step is reduced to additions, products and evaluation of the function  $g(x(t_n))$ , reducing the complexity of the numerical processing.

This simple case serves to introduce the Euler method and its limitations. Equation 1 describes a system discrete in space (neurons of the network) but with a continuous dependence on time. The application of Euler approximation requires a time discretization thus by transforming the Eq. 1 according to the analysis leading from Eq. 2 to Eq. 4 an equation system discrete in space and time is obtained. The problem comes from the presence of a diffusive term in the CIR system; under this circumstance the numerical solving by a discretization in time and space should take into account the Friedrich-Levy-Courant condition. The FLC conditions points the inconsistency of a discrete numeric scheme for solving diffusive-like equations when the discrete evolutions of time and space variables are not properly coupled, in such a way that the advance in the spatial dimension is not faster than those in the temporal dimension. Under the approximation provided by the Euler method it can be proven that a slight change in the method premises allows for applying the Euler method in diffusion systems but obeying the FLC condition<sup>7</sup>. According to the previous reasoning the differential operator could be approximating now by the discrete approach:

$$\left. \frac{dx(t)}{dt} \right|_{t=t_n} \approx \frac{x(t_n) - x(t_{n-1})}{h} \quad (5)$$

so the differential equation (2) could be approximated now by:

$$\frac{x(t_n) - x(t_{n-1})}{h} = g(x(t_n)) \Rightarrow x(t_n) = x(t_{n-1}) + h g(x(t_n)). \quad (6)$$

This second approach is named *Euler backward method*. As can be observed the new approach does not introduce a minor change: now the value of the solution at time  $t_n$  should be obtained by solving for  $x(t_n)$  but considering the function  $g(x)$ , i.e. this is an *implicit* equation in  $x(t_n)$ , whose numerical solution could be, in general, difficult to attain, particularly in our case, where this function is a polynomial of degree three.

In order to deal with this difficulty let us suppose first the problem in its most simplified version, i.e. assuming that no active diffusion and target exist. In these conditions only the Eq. 1 will be reduced to

$$\dot{r}_{ij} = d\Delta r_{ij} = d(r_{i+1,j}(t) + r_{i-1,j}(t) + r_{ij+1}(t) + r_{ij-1}(t) - 4r_{ij}(t)). \quad (7)$$

Discretizing in time and applying the Euler backward scheme:

$$\frac{r_{ij}(t_n) - r_{ij}(t_{n-1})}{h} = dL(r_{ij}(t_n)) = d(r_{i+1,j}(t_n) + r_{i-1,j}(t_n) + r_{ij+1}(t_n) + r_{ij-1}(t_n) - 4r_{ij}(t_n)). \quad (8)$$

where  $L$  denotes the discrete laplacian operator. Isolating the terms in  $t_n$  and  $t_{n-1}$ :

$$(1 + 4hd)r_{ij}(t_n) - hd(r_{i+1,j}(t_n) + r_{i-1,j}(t_n) + r_{ij+1}(t_n) + r_{ij-1}(t_n)) = r_{ij}(t_{n-1}). \quad (9)$$

Matrices are the best way for representing this linear system, by writing the matrix  $(r_{ij})_{i,j=1}^N$  as a column vector  $R = [r_{11}; \dots; r_{N1}; r_{12}; \dots; r_{N2}; \dots; r_{1N}; \dots; r_{NN}]$ . According to this notation the system (9) could be written as:

$$\tilde{S}R(t_n) = R(t_{n-1}), \quad (10)$$

with the matrix  $\tilde{S}$  will be given by the expression:

$$\begin{aligned} \tilde{S} = & \text{diag}((1 + 4 * h * D) * \text{ones}(1, M), 0) + \text{diag}_0((-h * D) * \text{ones}(1, M - 1), 1) + \\ & + \text{diag}_0((-h * D) * \text{ones}(1, M - 1), -1) + \dots \\ & \dots + \text{diag}((-h * D) * \text{ones}(1, M - N), N) + \text{diag}((-h * D) * \text{ones}(1, M - N), -N), \end{aligned} \quad (11)$$

where in order to reduce notation Matlab code has been used in the writing. The subindex 0 stands for upper and lower diagonals with zeros regularly located. For instance, for a network of 3x3 elements that matrix will be:

$$\tilde{S} = \begin{pmatrix} 1+4hd & -hd & 0 & -hd & 0 & 0 & 0 & 0 & 0 \\ -hd & 1+4hd & -hd & 0 & -hd & 0 & 0 & 0 & 0 \\ 0 & -hd & 1+4hd & 0 & 0 & -hd & 0 & 0 & 0 \\ -hd & 0 & 0 & 1+4hd & -hd & 0 & -hd & 0 & 0 \\ 0 & -hd & 0 & -hd & 1+4hd & -hd & 0 & -hd & 0 \\ 0 & 0 & -hd & 0 & -hd & 1+4hd & 0 & 0 & -hd \\ 0 & 0 & 0 & -hd & 0 & 0 & 1+4hd & -hd & 0 \\ 0 & 0 & 0 & 0 & -hd & 0 & -hd & 1+4hd & -hd \\ 0 & 0 & 0 & 0 & 0 & -hd & 0 & -hd & 1+4hd \end{pmatrix}. \quad (12)$$

Then defining  $S = \tilde{S}^{-1}$  the system (10) will be given by:

$$R(t_n) = S R(t_{n-1}). \quad (13)$$

This simplified system does not contain any non linear term, so if we include the active diffusion and we apply the same process we should solve for  $x(t_n)$  from an implicit non linear equation so the solution could not be expressed in matrix terms and its numerical solution would be quite complex. In order to tackle the problem let us assume that Euler backward method is only applied to the diffusive part of the CIR system, keeping the Euler forward method for the remaining terms. In such a way it is straightforward (but tedious) to prove that the CIR system

$$\begin{aligned} \dot{r}_{ij} &= q_{ij} \{ H(r_{ij}) (f(r_{ij}) - v_{ij}) + d \Delta r_{ij} - r_{ij} p_{ij} \} \\ \dot{v}_{ij} &= (r_{ij} - 7v_{ij} - 2) / 25 \end{aligned} \quad (14)$$

with  $f(r) = (-r^3 + 4r^2 - 2r - 2) / 7$  and from its discretization:

$$\begin{aligned} \frac{r_{ij}(t_n) - r_{ij}(t_{n-1})}{h} &= q_{ij}(t_{n-1}) \{ H(r_{ij}(t_{n-1})) (f(r_{ij}(t_{n-1})) - v_{ij}(t_{n-1})) - r_{ij}(t_{n-1}) p_{ij}(t_{n-1}) + d L(r_{ij}(t_n)) \} \\ \frac{v_{ij}(t_n) - v_{ij}(t_{n-1})}{h} &= \frac{r_{ij}(t_{n-1}) - 7v_{ij}(t_{n-1}) - 2}{25} \end{aligned}$$

can be written as

$$\begin{aligned} R(t_n) &= S Q(t_{n-1}) \{ R(t_{n-1}) + h H(R(t_{n-1})) (f(R(t_{n-1})) - V(t_{n-1})) - h R(t_{n-1}) P(t_{n-1}) \} \\ V(t_n) &= V(t_{n-1}) + h (R(t_{n-1}) - 7V(t_{n-1}) - 2) / 25 \end{aligned} \quad (15)$$

where matrices  $(v_{ij})_{i,j=1}^N$ ,  $(q_{ij})_{i,j=1}^N$  and  $(p_{ij})_{i,j=1}^N$  have been transformed into the column vectors:

$$\begin{aligned} V &= [v_{11}; \dots; v_{N1}; v_{12}; \dots; v_{N2}; \dots; v_{1N}; \dots; v_{NN}] \\ Q &= [q_{11}; \dots; q_{N1}; q_{12}; \dots; q_{N2}; \dots; q_{1N}; \dots; q_{NN}] \\ P &= [p_{11}; \dots; p_{N1}; p_{12}; \dots; p_{N2}; \dots; p_{1N}; \dots; p_{NN}]. \end{aligned} \quad (16)$$

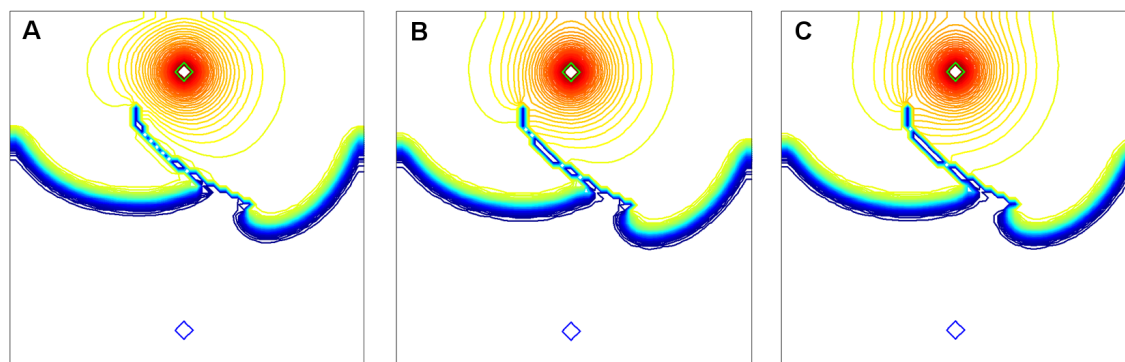
Finally the Von Neumann boundary conditions are imposed, thus each cell in the obstacle boundary has the same value than that neighbour cell perpendicular to the boundary.

In summary the combined application of Euler forward and backward methods allows for solving the CIR non linear system as a series of matrix calculations and evaluations (Eq. 15), significantly reducing the computational costs and requirements of the hardware system.

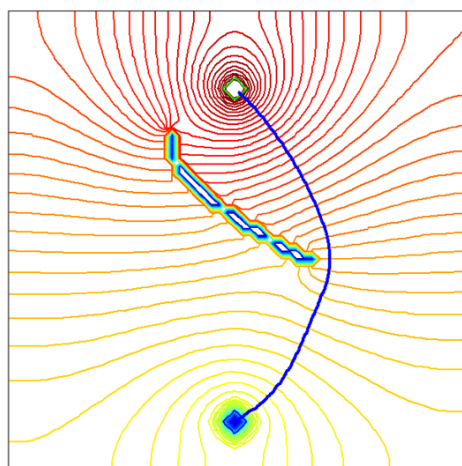
### Performance of the numerical scheme for the hardware CIR model

The performance of the proposed numeric scheme in equation 15, as a method based on the Euler approximation, depends on the  $h$  value, i.e. on the precision of the temporal evolution. The smaller  $h$  the more accurate is the obtained Compact Internal Representation. Nevertheless it should be noted that reducing  $h$  increases the number of steps needed to complete the simulation until the final time. Figure 3 shows one screenshot of the CIR process for the moving vertical bar example. The simulations were generated by means of the numerical scheme given in Eq. 15 by taking three different  $h$  values. Figure 3.A corresponds to  $h = 0.1$  and took 57 seconds of CPU, whereas for Fig. 3.B and 3.C  $h$  was equal to 0.01 and 0.001, with a CPU time of 565 and 7013 seconds respectively. As expected from the features of the Euler approximation, the required computation time increases linearly with the decreasing of  $h$ , but Fig. 3 reveals that the value of  $h$  affects differently to the dynamics of the CIR generation.

The CIR system evolves according to two concurrent different diffusion types, the Wave and the Diffusion Regimens. The Wave Regimen concerns the wavefront propagation and thus the creation of the effective obstacles, whereas the Diffusion Regimen concerns the creation of the gradient profile, containing the set of trajectories for the robot navigation. Figure 3 shows how, regardless the  $h$  value the effective obstacles are nearly equal in all cases, however the passive diffusion profile changes dramatically with  $h$ . This effect can be observed in the isopotential lines and in their behavior in the vicinity of the boundaries (the border of the arena is considered as an obstacle). The assumed Von Neumann boundary conditions models a repulsive effect of obstacles in the profile, i.e. the ‘virtual agents’ rebounds against the obstacles<sup>6</sup>. Nonetheless in the case of the smallest  $h$  those isopotential lines behave as if the boundaries were absorbent, transiting to the repulsive case when the  $h$  value increases. This phenomenon clearly affects to the final CIR because, even though the effective obstacles are not influenced by the precision of the approximation, the navigation trajectories are obtained from the final diffusive profile so a non reliable final profile could compromise the final action strategies of the robot. Considering the more accurate value of  $h$  ( $h = 0.001$ ) the performance of the system 15 is indistinguishable from the CIR obtained by the Runge-Kutta method (Fig. 2.D), as shown in figure 4, where the shortest trajectory from the robot initial position to the target is drawn.



**Figure 3.** Snapshots for the CIR generation from the equation 15. A, B and C correspond to three different  $h$  values,  $h = 0.1, 0.01$  and  $0.001$  respectively. This value affects the Diffusive Regimen but not the creation of the effective obstacles.



**Figure 4.** Compact Internal Representation obtained from equation 15 with  $h = 0.001$ , with the shortest trajectory to be followed by the robot from its initial position (upper) to the target (bottom).

### Numerical scheme implementation in FPGA chipset

The numerical method proposed in equation 15 offers significant advantages for its hardware implementation in a FPGA chipset: it operates in single steps by using simple mathematical operations and by evaluating matrices, so the computational requirements can be easily fulfilled by this platform. Nonetheless a balance between accuracy and computation time should be acquired. The initial estimations suggest that with a temporal step  $h = 0.1$  the Compact Internal Representation generated by a FPGA based on the equation 15 and shown in figure 4 would be obtained in 0.3 seconds, whereas for  $h = 0.01$  it will need 3 seconds to create the CIR. As discussed previously a reliable CIR (in terms of the robot trajectories) would require a value of  $h = 0.001$ , but the needed computational time could be a problem for attaining the real-time generation of the CIR.

There exist two complementary strategies to achieve a real-time CIR chipset based on the properties of this internal representation. On the one side the effective obstacles are not sensitive to the value of  $h$  so the CIR creation could be divided in two stages: in the first one  $h$  could be low, e.g. equal to 0.1, thus the propagation of the wave and the effective obstacles could be correctly generated in a short time. Once the wavefront has covered the whole arena the system is only passively diffusive and it would be just governed by the equation 13. Then the  $h$  value could be increased, e.g. to 0.001, ensuring the correct diffusive profile, and thus the right trajectories, but with a significantly reduction of the CPU time because the hardware implementation of this equation is significantly cheaper in terms of computation resources. On the other hand a biological-inspired processing scheme could be adopted for the robot decision making<sup>5</sup>. This processing has two different pathways, the subconscious and the conscious pathways. The conscious one deals with the creation of the CIR corresponding to a novel situation, generated by the chipset, whereas the unconscious one retrieves the CIR previously associated to a familiar situation when such situation is found again. The proper combination of these strategies could significantly improve the final performance of the CIR hardware implementation in autonomous robots.

## Conclusions

The Compact Internal Representation proposes a novel global decision making suitable for autonomous robots in which the prediction about the future evolution of its environment is properly combined with its possible actions to obtain a representation of the robot's environment. This representation, the CIR, will contain a set of infinite strategies suitable to be followed by the robot to deal with this situation on base of its motivation. The final aim of this approach is its implementation in a hardware platform to attain a real-time CIR generation. The ideal system for a prototype hardware implementation is the FPGA but its requirements demand a reformulation of the numerical scheme needed to solve the CIR non linear equations. This reformulation is based on a mixture of the Euler forward and backward method and lead to numerically simple equations easily solvable by matrix algebra. The critical parameter of this new numeric scheme is the temporal step  $h$ , whose value determines the accuracy of the created CIR and the time demanded by the CPU to obtain the result. High  $h$  values provide accurate Compact Internal Representations but consuming significant resources and time. Despite of these limitations the CIR real-time hardware implementation could be achieved by combining diverse  $h$  values in different stages of the CIR generation, taking advantages from the features of the proposed numeric scheme.

## Acknowledgments

This study has been sponsored by the EU grant SPARK II (FP7-ICT-216227) and by the Spanish Ministry of Education and Science under the grant FIS2010-20054 and a Juan de la Cierva fellowship (J.A.V.).

## Bibliography

- [1] Kandel, E.R.; Schwartz, J.H. & Jessell, T.M., [Principles of neural science]. McGraw-Hill, New York, 4th ed. (2000).
- [2] Koch, C. [Biophysics of Computation: Information Processing in Single Neurons] Oxford University Press: New York, New York, (1999).
- [3] Sosnik, R., Haidarliu, S., Ahissar, E. J., "Temporal frequency of whisker movement. I. Representations in brain stem and thalamus". *Neurophysiol.* 86(1), 339-53 (2001).
- [4] Godfrey-Smith, P. "Environmental Complexity and the Evolution of Cognition" [The evolution of intelligence]. Sternberg R. and Kaufman J. eds. Lawrence Erlbaum Associates (2001).
- [5] Makarov V., Villacorta-Atienza J.A. "Compact internal representation as a functional basis for protocognitive exploration of dynamic environments". [Recurrent Neural Networks]. InTech. (2011).
- [6] Villacorta-Atienza J.A., Velarde M.G., Makarov V. "Compact internal representation of dynamic situations: Neural network implementing the causality principle". *Biol. Cybern.* 03, 285–297 (2010).
- [7] Burden, R L. and Faires J. D. [Numerical Analysis]. Brooks Cole. (2004).

[8] Salas L., Alba L., Villacorta-Atienza J.A. , Velarde M.G., Makarov V. "FPGA Implementation of a modified FitzHugh-Nagumo neuron based Causal Neural Network for Compact Internal Representation of dynamic environments". Proceedings SPIE Microtechnologies 2011. Prague, Czech Republic. (2011).