

# Modelling lava flows by Cellular Nonlinear Networks (CNN): preliminary results

C. Del Negro<sup>1</sup>, L. Fortuna<sup>2</sup>, and A. Vicari<sup>1, 2</sup>

<sup>1</sup>Istituto Nazionale di Geofisica e Vulcanologia – Sezione di Catania, Italy

<sup>2</sup>Università di Catania – Dipartimento di Ingegneria Elettrica Elettronica e dei Sistemi, Italy

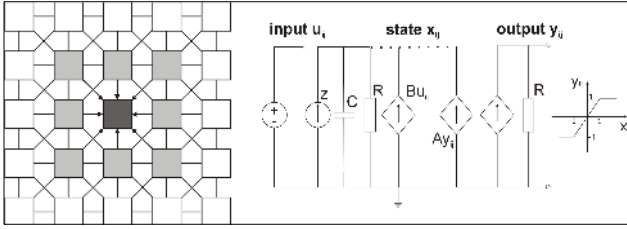
Received: 23 June 2004 – Revised: 28 September 2004 – Accepted: 7 October 2004 – Published: 19 May 2005

**Abstract.** The forecasting of lava flow paths is a complex problem in which temperature, rheology and flux-rate all vary with space and time. The problem is more difficult to solve when lava runs down a real topography, considering that the relations between characteristic parameters of flow are typically nonlinear. An alternative approach to this problem that does not use standard differential equation methods is Cellular Nonlinear Networks (CNNs). The CNN paradigm is a natural and flexible framework for describing locally interconnected, simple, dynamic systems that have a lattice-like structure. They consist of arrays of essentially simple, nonlinearly coupled dynamic circuits containing linear and non-linear elements able to process large amounts of information in real time. Two different approaches have been implemented in simulating some lava flows. Firstly, a typical technique of the CNNs to analyze spatio-temporal phenomena (as Autowaves) in 2-D and in 3-D has been utilized. Secondly, the CNNs have been used as solvers of partial differential equations of the Navier-Stokes treatment of Newtonian flow.

## 1 Introduction

An important goal of volcanology is the development of models to simulate the evolution of high hazard phenomena. In particular, knowledge of the likely path of a lava flow is useful for long-term land use planning (Wadge et al., 1994). The advance of lava flows produced by volcanic eruptions has been studied through field observations as well as through analytical and numerical modeling. Simulations of lava flow emplacement attempt to understand how the complex interaction between a flow's physical properties and emplacement characteristics lead to the final flow dimensions and morphology observed in the field. Unfortunately, it is very difficult to establish straightforward relationships

between physical properties of a lava flow and its morphologic parameters such as length or surface structure (Pinkerton and Wilson, 1994; Myamoto and Sasaki, 1998; Gregg and Fink, 1996). The forecasting of lava flow paths is a typically complex problem, especially when it consists of an unconfined multiphase stream whose temperature, rheologic parameters, and local flow rate all vary with space and time (Crown and Baloga, 1999). The complexity is associated with several aspects including, among the others, the presence of numerous interacting elements, the nonlinearity of the interconnections, the appearance of global-level properties absent at local level, and the ability of self-organization. Moreover, a common characteristic of the complex systems is the possibility to amplify a small local phenomenon carrying all the system in a new state. The velocity of lava flowing in a channel depends on the physical properties of the lava (density, viscosity, yield strength), environmental factors (gravity, underlying topography), and the channel dimensions (Harris and Rowland, 2001). Lava viscosity and yield strength can, in turn, be related to temperature and crystallization. The problem is more difficult to solve when lava runs down a real topography, considering that the relations between characteristic parameters of flow are typically nonlinear. Because it is hard to deal with so many parameters, simplified models have been proposed using some approximations (Wadge, 1978; Huppert, 1982; Pieri, 1986; Crisp and Baloga, 1990; Dragoni, 1993). Even if many of these models have been constructed taking in account various aspects of flow emplacement and cooling, most previous works seem to be too simplified. Generally, the methods are based on empirically obtained equations for very simple cases, and they are difficult to apply in general conditions. For example, the method of Ishiara (Ishiara et al., 1990), based on the Dragoni model (Dragoni et al., 1986), is valid only if the plane is inclined. These methods don't consider the case of flow driven by the pressure gradient (effect of self-gravity), which was solved by some works including Miyamoto and Sasaki (1998), Miyamoto et al. (2004), and Mei and Yuhi (2001). A number of numerical models have also been pro-



**Fig. 1.** In the left side, a typical CNN structure composed by a two-dimensional array of  $M \times N$  cells arranged in a rectangular grid. In the right side, the basic cell defined as a nonlinear first order circuit:  $u$ ,  $x$  and  $y$  are respectively the input, the state variable and the output of the cell. The output is related to the state by a nonlinear equation. Each cell interacts with its neighbours by means of the voltage controlled current sources.

duced (e.g. Harris and Rowland, 2001). Some of them are intended mainly as an aid to hazard assessment, rather than to analyze flow dynamics (Young and Wadge, 1990). Others numerical models focus on the effects produced by lava flows on composite channel shapes (Macedonio and Longo, 1999), neglecting thermo-rheology characteristics of magma.

An alternative approach to standard differential equation methods in modelling complex phenomena is represented by novel parallel computing paradigms. A model well suited for parallel computing, namely that of Cellular Automata (CA), was introduced by Crisci et al. (1986, 1999) in simulating some lava flows. They designed a three dimensional CA model that has been successively reduced in a two dimensional model (Barca et al., 1988, 1993, 1994) to shorten computation costs. Moreover, Miyamoto and Sasaki (1997) included other expedients to overcome some problems connected with mesh dependence of which CA suffers. An evolution of CA is represented by Cellular Nonlinear Networks (CNNs) that are solvers of dynamic complex systems (Manganaro et al., 1999), whose behavior can be exclusively drawn in terms of local interactions of their simple constituent parts. In particular, CNNs are peculiarly interesting either for reasons of absolute performance or for reasons of cost/performance ratio (Chua and Roska, 1993; Roska et al., 1995).

We present two different approaches of CNNs to simulate lava flows. Firstly, we describe a reaction-diffusion system (Danieli et al., 2003; Sorbello et al., 2003). These systems can be considered as an ensemble of a large number of identical subsystems, coupled to each other by diffusion. Traditionally, the local subsystems are defined through a set of nonlinear differential equations. CNNs are a powerful tool for the modeling of reaction-diffusion systems (Manganaro et al., 1999). Secondly, we used the CNNs as solvers of partial differential equations (Roska et al., 1995; Kozek et al., 1995; Vicari et al., 2003). The equations considered are the classic Navier-Stokes equations for the motion of a fluid.

We apply both these methods to simulate lava flows down on Mt. Etna. We describe the isothermal and rheological evolution of a non-channelized flow. A parametric study on the

major computational variables in order to guarantee an adequate resolution of the main large-scale processes and to optimize computer time has also been conducted.

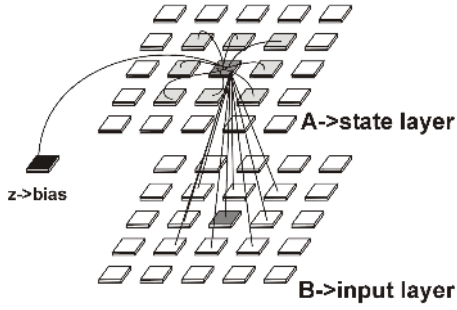
## 2 The CNN paradigm

CNNs were introduced by Chua and Yang (1988a, 1988b), who proposed using an array of essentially simple, nonlinearly coupled, dynamic circuits to process large amounts of information in real time. The fundamental building block of the CNN is the cell, a lumped circuit containing linear and nonlinear elements (Fig. 1). Therefore, a CNN (Fig. 1) is an array of cells characterized from an input, a state and an output. Every cell is coupled only to the near cells (direct interaction), but more distant cells can however be influenced indirectly through a propagation process. There are many application fields of CNNs, for example solving or simulating partial differential equations (PDE) and systems of locally interconnected ordinary differential equations (ODE). When solving a system of PDEs, four variables are represented in either continuous or discrete form: time, value of the state variable, interaction of parameters, and space. This procedure transforms a PDE into a set of ODE's, and the original spatially continuous system is transformed into an array of small, discrete, interacting systems. The CNN paradigm is a natural and flexible framework for describing locally interconnected, simple, dynamic systems, which have a lattice-like structure. The equation that regulates the behavior of a CNN structure, is:

$$\frac{dx(t)}{dt} = -x(t) + A * y(t) + B * u(t) + Bias \quad (1)$$

where  $u(t)$  is the input of the cell,  $x(t)$  represents the state variable,  $y(t)$  is the output of the cell, and  $A$  and  $B$  are, respectively, the feedback template coefficients and the control template coefficients. It is important to note that the equation of the CNN structure can be found as an analog circuit model (Fig. 1) in which the values of capacitor ( $C$ ) and resistance ( $R$ ) determine the rate of change of the dynamics of the circuit. When the circuit parameters are scaled or normalized for convenience, the dimensionless Eq. (1) is obtained.

The feedback and control template represent the coupling coefficients of the cells and they completely define the behavior of the network with a given input and initial condition. The template is usually space invariant (the coefficients of the matrix have constant values for entire structure of CNN), or, in some complex cases, the template has the property of space variance. With single-layer representation, only one state variable can be considered. This can be generalized if the system order (the number of the state variables) of the cell is increased, introducing a multilayer CNN (MCNN). A MCNN (Fig. 2) is composed of cells having several state variables, one for each layer. We can imagine a MCNN arranged in several single-layer arrays, stacked one above the other, in which a full layer-to-layer interaction is possible. Any layer can be used to perform a separate processing and, of course, any layer works in parallel to the others.



**Fig. 2.** Multilayer CNN (MCNN) composed by cells having several state variables, one for each layer. The interaction between the state variables that belong to different layers can be complete, while the cell-to-cell interaction is restricted to neighbours.

We have implemented two different approaches of MCNN, in which two or three layer CNNs have been considered. In particular, the CNNs have been used to simulate a typical process of reaction-diffusion in 2-D and in 3-D (to test the capability of CNNs to solve a 3-D problem in a relatively short time), and to solve the Navier-Stokes equations in which some simplified hypotheses have been made.

### 3 A CNN as a generator of spatio-temporal phenomena to simulate lava flows

In this approach a two layer CNN is used to generate spatio-temporal phenomena known as Autonomous waves or Autowaves. These are a particular class of nonlinear waves, which propagate without forcing input, in strongly non-linear active media. This phenomenon is often encountered in combustion waves or in chemical reactions, as well as in many biological processes, such as propagation in nerve fibers or heart excitation. Autowaves have two typical characteristics: their shape keeps constant during propagation and reflection and interference do not take place. An example of these properties can be observed in the simulation of a spiral wave (Fig. 3).

A reaction-diffusion type PDE can describe the excitable medium, which gives rise to autowave phenomena:

$$\frac{\partial u(\mathbf{x}, \mathbf{y}, t)}{\partial t} = \nabla^2 u(\mathbf{x}, \mathbf{y}, t) + f(u) \quad (2)$$

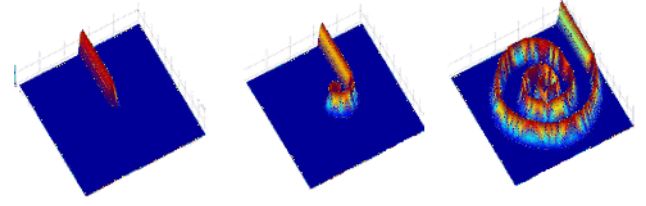
where  $\nabla^2 u(\mathbf{x}, \mathbf{y}, t)$  is the diffusion term;  $f(u)$  describes the active local dynamics of the medium.

The constant template used to generate these phenomena are:

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = 0, \quad (3)$$

where

$$A_{11} = A_{22} = \begin{pmatrix} 0.05 & 0.1 & 0.05 \\ 0.1 & 1.65 & 0.1 \\ 0.05 & 0.1 & 0.05 \end{pmatrix},$$



**Fig. 3.** Output of the first layer of a CNN processing a spiral wave with the same time constant for all cells: the wavefront shape remains unchanged during propagation and no reflection takes place.

$$A_{12} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad A_{21} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \quad (4)$$

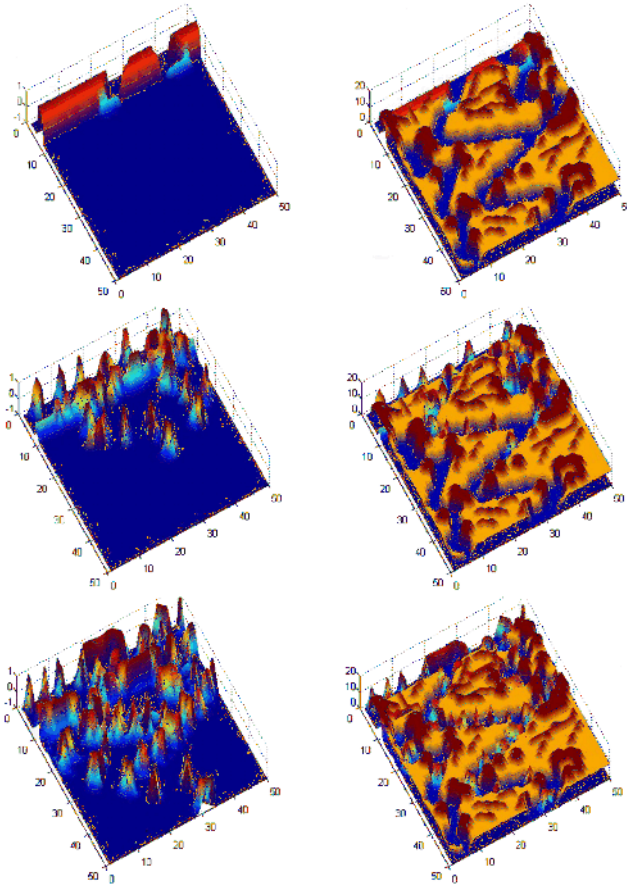
The choice of template coefficients is not obvious and is made with technique of “trial and error”. A particular simulator, implemented using C++ environment, was developed that permits to add:

- Noise on time constants, to modify the RC term (time constant of the system) in the equation of the cell
- Noise on templates, to test the sensitivity of the parameters
- Space variant templates
- Space variant time Constants
- Rewiring

The term Rewiring refers to the possibility to create a connection between two distant cells. In Fig. 3 the autowaves propagation is homogeneous because all cells have the same time constant. Alternatively, it is possible to assign a different time constant to each cell; in this way regular variations of time constant produce paths that autowaves follow with different speeds. Higher time constants correspond to lower propagation speeds through these routes and vice-versa. This could be a way to introduce surface topography and fluid viscosity in the simulations. The same thing can be obtained through space-variant template coefficients.

We have analyzed the autowave behavior in presence of a simulated topography, including falls, rises, bends and obstacles. In Fig. 4 the evolution of the autowave and the values of the capacitor used to generate the topography are shown. In this way, it is simple to control the speed of the autowave, increasing or decreasing the acceleration when a fall or rise is met, and maintaining a constant speed on a plane. These preliminary results highlight the potential of the method for determining lava flow paths. It is important to note that with this kind of approach it is difficult to find a physically meaningful link between the time constant of the system and the physical properties of the flow (for example, viscosity) or the real topography.

We have also tried another approach by extending the application of autowaves in 3-D. Generally, autowaves have



**Fig. 4.** In the left side, it is shown the output of the first layer of the CNN processing a spiral wave with different time constant for each cell. In the right side, capacitor values, used to represent the topography, are overlapped to the output of the first layer of CNN model, opportunely normalized.

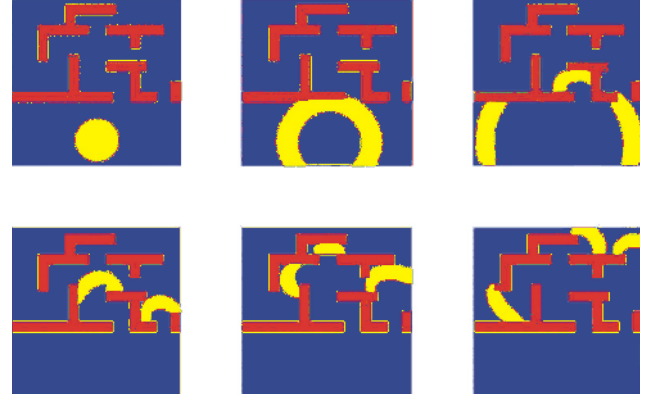
been used to simulate a traveling wave front in a 2-D labyrinth (Manganaro et al., 1999). It can be seen here that a wave front propagates throughout the medium with a constant speed, breaking into different arms at each fork (Fig. 5). These features could be used to find the shortest path (or the most probable) between two points in the labyrinth (Perez-Munozuri et al., 1993; Arena et al., 1997). If we represent a real topography as a 3-D labyrinth, then it is possible to simulate lava flow over a topography with an autowave that moves in a labyrinth, using a two layers CNN. The feedback template A is reported in the Eqs. (3) and (4). The control template B is not zero anymore (Perez-Munozuri et al., 1993):

$$B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}, \quad (5)$$

where:

$$B_{11} = B_{22} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad B_{21} = B_{12} = 0. \quad (6)$$

In order to check the ability of the CNN paradigm to solve a 3-D problem, using autowaves theory, a three dimensional



**Fig. 5.** Output of the first layer of a CNN in which a wavefront moves in a 2-D labyrinth. The inputs values of the two layers of the CNN represent the labyrinth (red pixel). The wavefronts of the first layer propagate throughout the medium with a constant speed, breaking into different arms at each fork (yellow pixel).

structure has been implemented. The simulations were carried out taking into account the following features:

- the initial condition of the cell to obtain an autowave in which the starting-point represents the emission point (source position);
- the 3-D labyrinth as a bias in order to draw the real topography; in particular, the topography has been transformed in a mask, in which only binary states are present. So, the input of the cells can take only the 1 and 0 values, in which the state “1” describes the volcano surface, and the state “0” the atmosphere. In this way, using the input mask, it is easy to update the configuration of obstacles deriving from the superimposition of previous lava flows;
- the parameters of the interconnections of the cells to simulate gravity force.

The 3-D labyrinth used for the test consists of a 3-D array having  $100 \times 100 \times 33$  cells. Figure 6 shows the result of a simulation of a lava flow on the southern flank of Etna volcano, during the eruption of 2001. The simulated path is in good agreement with the field observations. Every simulation takes only a few minutes and this permits displaying results quickly, enabling rapid assessment and multiple runs. The results have shown that autowave propagation in a 3-D labyrinth is a very fast approach for lava flow simulations.

#### 4 CNN as solvers of PDE for lava flow modelling

PDEs and CNNs share the property that their dynamic behavior depends only on their local interactions. We have therefore utilized the CNNs to solve the Navier-Stokes equation of fluid motion. The general model describing fluid motion



in a time-dependent domain for incompressible fluids has the form:

$$\rho \frac{\partial \mathbf{w}}{\partial t} + \rho (\mathbf{w} \text{grad}) \mathbf{w} - \rho \mathbf{F} + \text{grad}(p) - \eta \nabla^2 \mathbf{w}, \quad (7)$$

where  $\mathbf{w}$  is a vector of fluid speed (state variable),  $\rho$  is fluid density,  $\eta$  is fluid viscosity,  $p$  is shear stress factor, and  $\mathbf{F}$  is an external force.

In general, such an equation doesn't have an exact analytical solution, except in cases where some simplified hypotheses have been adopted. Therefore, some assumptions on magma rheology, that make our problem easier from the numerical point of view, have been made. Lava is treated as an incompressible, isothermal, homogeneous, and constant viscosity fluid with Newtonian rheology, even if it is well known that a lava flow has the typical Bingham characteristic. Moreover, temperature is not included specifically in any calculations. Although more realistic non-Newtonian models were adopted in the past (e.g. Dragoni et al., 1986), only simple geometries have been considered. On the contrary, we considered the effect of flow on a complex topography.

The lava flow is represented by a two-dimensional array of square cells, therefore vertical motion of lava is neglected in the analytical formulation. The Navier-Stokes equation has been considered in a general form, so a three layer CNN has been introduced, in which the first two layers represent the components of velocity respectively on  $x$  and  $y$  direction, and the third layer represents the term of pressure. In this case the pressure field must be computed by another equation.

Starting from Eq. (7), and considering only two components of velocity ( $u$  and  $v$ ), we obtain (Roska et al., 1995; Kozek et al., 1995):

$$\frac{\partial u}{\partial t} = \frac{\eta}{\rho} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - u \frac{\partial u}{\partial y} - \frac{1}{\rho} \frac{\partial p}{\partial x} + F_x \quad (8)$$

$$\frac{\partial v}{\partial t} = \frac{\eta}{\rho} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - v \frac{\partial v}{\partial y} - \frac{1}{\rho} \frac{\partial p}{\partial y} + F_y. \quad (9)$$

When incompressibility is assumed, the continuity equation in rectangular coordinates has the form:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \quad (10)$$

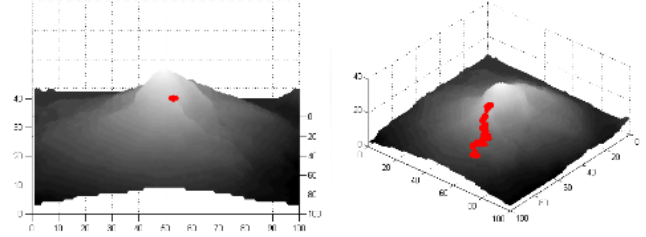
Utilizing Eq. (10), we can express the Eqs. (8) and (9) in the conservation form:

$$\frac{\partial u}{\partial t} = F_x - \frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\eta}{\rho} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \frac{\partial u^2}{\partial x} - \frac{\partial (uv)}{\partial y} \quad (11)$$

$$\frac{\partial v}{\partial t} = F_y - \frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{\eta}{\rho} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - \frac{\partial v^2}{\partial y} - \frac{\partial (uv)}{\partial x}. \quad (12)$$

The pressure field can be computed from the Poisson equation:

$$\nabla^2 p = - \left( \frac{\partial^2 u^2}{\partial x^2} + 2 \frac{\partial^2 (uv)}{\partial x \partial y} + \frac{\partial^2 v^2}{\partial y^2} \right). \quad (13)$$



**Fig. 6.** Application of the 3-D labyrinth experiment using the real topography of Etna volcano. The 3-D labyrinth is composed by a 3-D array of 100x100x33 cells.

The analogies between CNNs and a set of partial differential equations can be made evident using a discretization process. We introduced a 2-D spatial grid (Fig. 1) into the two-dimensional spatial variable, and we approximated the spatial derivatives by finite difference terms with the following formulas (only one variable state is reported):

$$\frac{\partial^2 u}{\partial x^2} \cong \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} \quad (14)$$

$$\frac{\partial^2 u}{\partial y^2} \cong \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} \quad (15)$$

$$\frac{\partial u}{\partial x} \cong \frac{u_{i,j} - u_{i-1,j}}{\Delta x} \quad (16)$$

$$\frac{\partial u}{\partial y} \cong \frac{u_{i,j} - u_{i,j-1}}{\Delta y} \quad (17)$$

$$\frac{\partial p}{\partial x} \cong \frac{p_{i,j} - p_{i-1,j}}{\Delta x}. \quad (18)$$

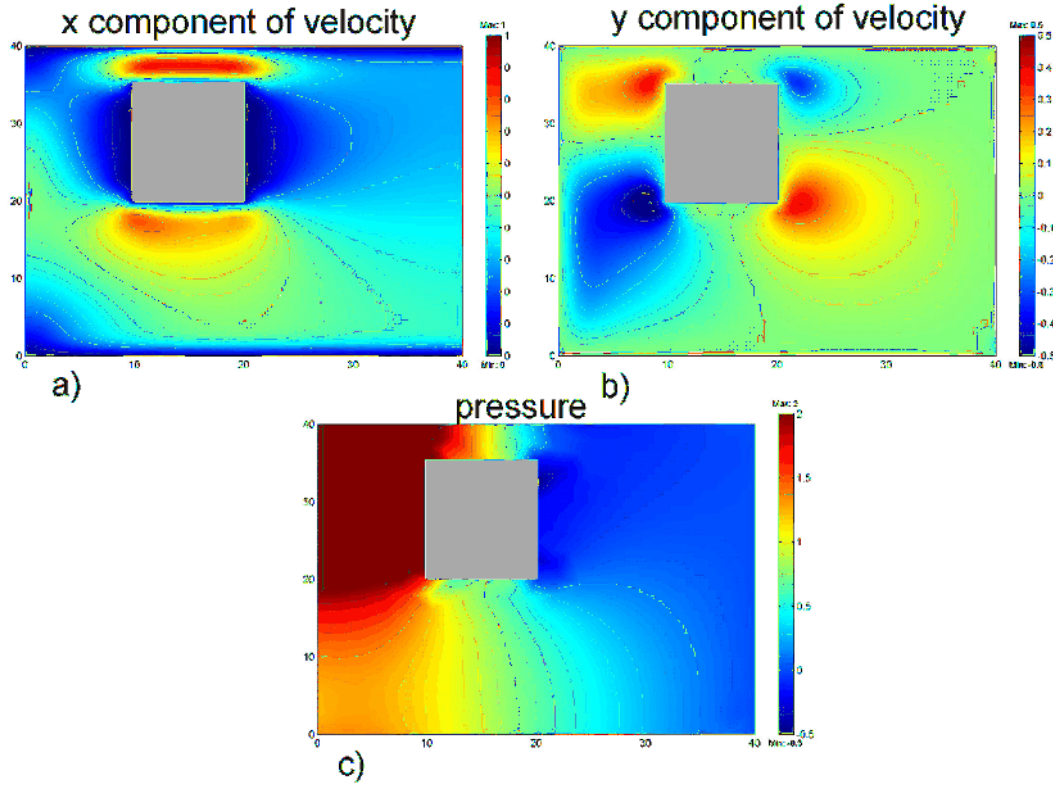
Therefore the model described by Eq. (11) is approximated by a time-continuous, spatio-discrete model:

$$\begin{aligned} \frac{du_i}{dt} = & -\frac{1}{\rho} \left( \frac{-p_{i-1,j} + p_{i+1,j}}{2h} \right) + \frac{\eta}{\rho} \left( \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h^2} \right) + \\ & \frac{\eta}{\rho} \left( \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h^2} \right) + \\ & - \left( \frac{-(u_{i-1,j})^2 + (u_{i+1,j})^2}{2h} \right) - \left( \frac{-(u_{i,j-1}v_{i,j-1}) + (u_{i,j+1}v_{i,j+1})}{2h} \right). \end{aligned} \quad (19)$$

A similar equation holds for the  $v$  component. The Poisson equation for pressure is approximated by:

$$\begin{aligned} \nabla^2 p = & - \left( \frac{-(u_{i-1,j})^2 - 2(u_{i,j})^2 + (u_{i+1,j})^2}{h^2} \right) - \\ & \left( \frac{-(v_{i-1,j})^2 - 2(v_{i,j})^2 + (v_{i+1,j})^2}{h^2} \right) + \\ & - 2 \left( \frac{-(u_{i-1,j}v_{i-1,j+1}) + (u_{i+1,j+1}v_{i+1,j+1}) + (u_{i-1,j-1}v_{i-1,j-1}) - (u_{i+1,j-1}v_{i+1,j-1})}{4h^2} \right). \end{aligned} \quad (20)$$

In this system the speed vector ( $w$ ) in the Eq. (7) has been



**Fig. 7.** Distribution of the state variables in a pipeline around an obstacle computed by the CNN model as solver of PDE: velocity distribution in the (a) x-direction and in the (b) y-direction, and (c) the pressure surface for an array of  $40 \times 40$  cells.

decomposed along the two directions  $u$  and  $v$ . This choice has been made to obtain a simpler CNN structure and, consequently, to improve performance of the simulator. In such a way, the structure of the CNN will be composed of three layers, each of which is associated to one of the variable states  $u$ ,  $v$  and  $p$ .

The feedback and the control template of the model are (only for the  $u$  layer):

$$\begin{aligned}
 A^{u,u} &= \frac{1}{h} \begin{pmatrix} 0 & \frac{v}{h} & 0 \\ \frac{v}{h} - \frac{1}{2}u_{ij} & \frac{1-4v}{h} & \frac{v}{h} \\ 0 & \frac{v}{h} & 0 \end{pmatrix} u_{ij}, \\
 A^{v,u} &= \frac{1}{2h} \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} u_{ij} v_{ij}, \\
 A^{p,u} &= \frac{1}{2h} \begin{pmatrix} 0 & 0 & 0 \\ -\frac{1}{\rho} & 0 & \frac{1}{\rho} \\ 0 & 0 & 0 \end{pmatrix} p_{ij}, \quad B^u = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} F_{ij}^u. \quad (21)
 \end{aligned}$$

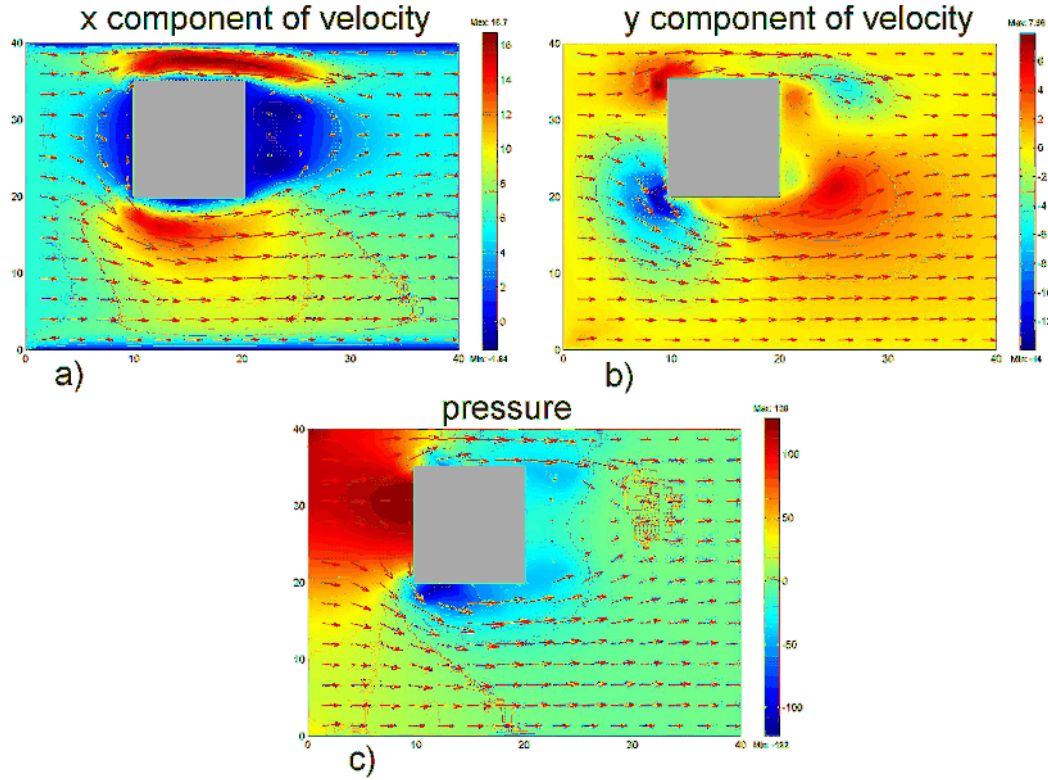
where  $h$  is the discrete step of the space (equidistant in both direction  $x$  and  $y$ ) and  $v = \eta/\rho$ .

The template coefficients of the model are space-variant and with variable coefficients. Also, the elements of the matrices are functions of the state variables (or of the layers); this means that a nonlinear CNN model has been obtained.

With respect to the previous approach with CNNs (as generators of spatio-temporal phenomena to represent reaction-diffusion equations), the computational complexity has been increased. However, the CNN approach has some advantages with respect to traditional finite element methods: among these the time-continuity of the CNN model and natural computing parallelism of this analog architecture.

#### 4.1 Modelling results

To test the performance of the simulator, we computed the flow pattern in a pipeline around an obstacle by the CNN model. In particular, the pressure surface and the velocity distributions in the  $x$  and  $y$  direction for an array of  $40 \times 40$  cells, obtained with the CNN algorithm, are reported in Fig. 7. The parameters utilized for this simulation are viscosity = 850 Pa/s, density = 2500 kg/m<sup>3</sup>, and  $h = 0.1$ . In this condition, a laminar flow corresponding to a Reynolds Number of 25 has been reproduced (no turbulent flow). To check the goodness of the method, a comparison with finite element method (FEM) solution has been made. In Figs. 8a and 8b the surface plot of components of velocity in the  $x$  and  $y$  directions are reported. Red arrows indicate the resultant flux. In Fig. 8c the value of the pressure is shown. The space distribution of the state variables is consistent with expected effects. A flow, uniformly distributed, starting from an initial condition of velocity and pressure, decreases its  $x$ -



**Fig. 8.** Distribution of the state variable in a pipeline around an obstacle computed by FEM: surface plot of component of velocity in the (a) x-direction and (b) y-direction, and (c) value of the pressure for an array of  $40 \times 40$  cells. The red arrows are the resultant flux.

component of velocity when the front side of the obstacle occurs (null value in the wall of the pipe), while it increments in correspondence to the lateral side of the obstacle. It is possible to do similar considerations for the y-component of velocity and pressure. The results of the simulations with the CNN model are comparable to those of the FEM one. A different kind of normalization between the two methods has been applied, and for this reason, different numerical values of the output have been obtained.

In the second example, a simple geometry representing a channel is shown. In Fig. 9 the geometry of the channel (a white and black picture) and, superimposed on it, the value of the state of the system in two different instants of time (blue scale images) are schematically represented. The flux starts from an initial condition of velocity in a specific point of the space (Fig. 9a). Successively, the flux goes down in the channel and follows its shape (Fig. 9b).

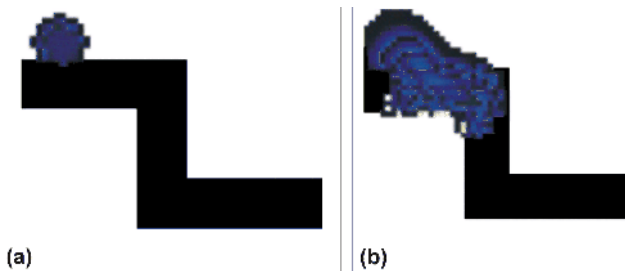
In the third example, we have simulated the lava flow over the 2-D topography of the north-east flank of Etna volcano during the eruptive crisis of 2001 (Fig. 10). In this case the input mask of the simulator represents the topography (as a gray scale image), which is taken into account in the CNN model by the bias term. Indeed, the matrix of the heights is transformed in a matrix of accelerations that represent a forcing input to the system. It is important to note that where the topography is flat the velocity of the flux tends to a low value. In this case, there isn't any condition to stop the flux, so it is

possible that a branch of the flow begins to start again. This is an improbable condition, but some adjustments about the structure of the equations (for example removing the hypothesis of Newtonian fluid and introducing the Bingham fluid ones), or considering the cooling effect, could bring more satisfying and realistic results.

## 5 Conclusions

Two different approaches to the use of CNN to simulate lava flow have been tried. The first approach deals with the CNN in its more classic way, without modifying the condition on the output of the cell and choosing constant template coefficients. The evolution of an autowave, before in 2-D and after in 3-D, has been adapted to lava flow simulation. In the 2-D case, the parameters of CNN basic equation (time constant of the system) have been changed in order to simulate flows with different viscosity values or real topography. In the 3-D case, a labyrinth has been replaced with the topography of the Etna volcano and an input mask has been introduced to update the configuration of new obstacles.

In the second approach, the CNNs have been used as solvers of differential equations, changing the classic structure of the cell, and using space-varying templates. We used the Navier-Stokes equations in their more general form, though some restrictions on the rheology of the fluid are necessary. Obviously, the problem has considerably increased



**Fig. 9.** Geometry of the channel used in the simulation using a CNN model as solver of PDE. The value of the state of the system is showed in two different instants of time (a) and (b) of the simulation.

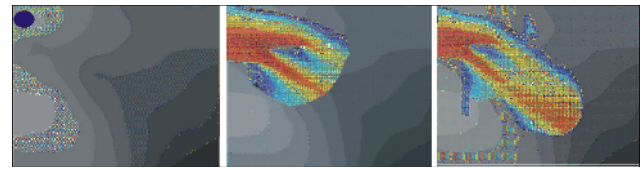
the computational complexity, but in this case a direct link with the rheological parameters of the system has been maintained. Simulations on different real topographies have also been made, obtaining encouraging results. Clearly, many of the assumptions that render the system unreal, from the physical point of view, must be overcome. However, in this context, the CNN paradigm has proved to be a flexible framework for describing a wide variety of phenomena and a suitable method for the simulation of lava flows.

Finally, it is worth to note that CNNs have some features that make them tailor made for hardware implementations. One of these, of course, is the local connectivity, while another one is the fact that the cells are mainly identical. These advantages have been leading to the development of a complex CNN architecture (Arena et al., 1997): the so-called CNN Universal Machine (CNUM). Basically, it consists of an electronic architecture in which the analog CNN has been completed by digital logic sections. In this architecture the feedback and control template play the role of the instructions of the CPU, i.e. the template determine the task that the CNUM processor must accomplish. In this configuration the performance of CNNs is very high, i.e. about 8000 frame/s for image processing (Manganaro et al., 1999). The two layer CNN chip is actually in advanced construction stage. As soon as a CNUM dedicated to simulate lava flows will be operable, we could perform multiple run in a few seconds.

**Acknowledgements.** The authors are grateful to G. Wadge, H. Miyamoto, and L. Chua for their constructive criticism of an earlier draft of this paper. The simulations of lava flow paths were developed in the frame of the TecnoLab, the laboratory for the technological advance in geophysics organized by DIEES-UNICT and INGV-CT. This research was supported by project EPOT of the Gruppo Nazionale per la Vulcanologia of the INGV.

Edited by: B. Malamud

Reviewed by: L. Chua, H. Miyamoto, and G. Wadge



**Fig. 10.** 2-D topography of Etna volcano (bias term of the CNN model) used in the simulation of the lava flow outpoured on the north-east flank during the eruptive crisis of 2001: (on the left) the blue dot indicates the initial condition of the velocity; (in the center) and (on the right) show two different moments of the simulation. The values of the velocity are superimposed on the topography.

## References

- Arena, P., Caponetto, R., Fortuna, L., and Manganaro, G.: Cellular neural networks to explore complexity, *Soft Computing*, 1, 120–136, 1997.
- Barca, D., Crisci, G. M., Di Gregorio, S., Marabini, S., and Nicoletta, F. P.: Nuovo modello cellulare per flussi lavici: colate di Pantelleria, *Bollettino GNV*, 4, 41–51, 1988.
- Barca, D., Crisci, G. M., Di Gregorio, S., and Nicoletta, F. P.: Cellular Automata methods for modeling lava flow: simulation of the 1986–1987 Etnean eruption, in: *Active Lavas: Monitoring and Modelling*, edited by: Kilburn, C. R. J. and Luongo, G. (UCL Press, London), 283–301, 1993.
- Barca, D., Crisci, G. M., Di Gregorio, S., and Nicoletta, F. P.: Cellular Automata for simulating lava flows: a method and examples of the Etnean eruptions, *Transport Theory and Statistical Physics*, 23, 1–3, 195–232, 1994.
- Chua, L. O. and Yang, L.: Cellular Neural Networks: Theory, *IEEE Transactions on Circuit and Systems*, 35, 10, 1257–1272, 1988a.
- Chua, L. O. and Yang, L.: Cellular Neural Networks: Applications, *IEEE Transactions on Circuit and Systems*, 35, 10, 1273–1290, 1988b.
- Chua, L. O. and Roska, T.: The CNN Paradigm, *IEEE Transactions on Circuit and Systems-I: Fundamental Theory and Applications*, 40, 3, 147–156, 1993.
- Crisci, G. M., Di Gregorio, S., Pindaro, O., and Ranieri, G.: Lava flow simulation by a discrete cellular model: first implementation, *Int. J. Modelling and Simulation*, 6, 137–140, 1986.
- Crisci, G. M., Di Francia, A., Di Gregorio, S., Nicoletta, F., Rongo, R., and Spataro, W.: Analysing lava risk for the Etnean area by cellular automata methods of simulation, *Nat. Hazards*, 20, 215–229, 1999.
- Crisp, J. and Baloga, S.: A model for lava flows with two thermal components, *J. Geophys. Res.*, 95, 1255–1270, 1990.
- Crown, D. A. and Baloga, S. M.: Pahoehoe dimensions, morphology, and branching relationships at Mauna Ulu, Kilauea Volcano, Hawaii, *Bull. Volcanol.*, 61, 288–305, 1999.
- Danieli, F., Ganci, G., Melita, D., Scalia, O., Del Negro, C., Fortuna, L., Frasca, M., and Vicari, A.: Lava Flow simulations using Cellular Neural Networks: Part I, in *Proceedings of Gruppo Nazionale per la Vulcanologia General Assembly 2003*, Roma, Italy, 9–11 June 2003.
- Dragoni, M.: Modelling the rheology and cooling of lava flows, in *Active Lavas: Monitoring and Modelling*, edited by: Kilburn, C. R. J. and Luongo, G., (UCL Press, London), 235–261, 1993.
- Dragoni, M., Bonafede, M., and Boschi, E.: Downslope flow models of a Bingham liquid: implications for lava flows, *J. Volc.*



- Geotherm. Res., 30, 305–325, 1986.
- Gregg, T. K. P. and Fink, J. H.: Quantification of extraterrestrial lava flow effusion rates through laboratory simulation, *J. Geophys. Res.*, 101, 16 891–16 900, 1996.
- Harris, J. L. and Rowland, S. K.: FLOWGO: a kinematic thermorheological model for lava flowing in a channel, *Bull. Volcanol.*, 63, 20–44, 2001.
- Huppert, H. E.: Flow and instability of a viscous current down a slope, *Nature*, 300, 427–429, 1982.
- Ishihara, K., Iguchi, M., and Kamo, K.: Numerical simulation of lava flows on some volcanoes in Japan, in: *Lava Flows and Domes: Emplacement Mechanisms and Hazard Implications*, edited by: Fink, J. K (Springer, Berlin), 174–207, 1990.
- Kozek, T., Chua, L.O., Roska, T., Wolf, D., Tetzlaff, R., Puffer, F., and Lotz, K.: Simulating nonlinear waves and partial differential equations via CNN – Part II: Typical Examples, *IEEE Transactions on Circuit and System – Part I: Fundamental Theory and Applications*, 42, 816–820, 1995.
- Macedonio, G. and Longo, A.: Lava Flow in a Channel with a Bifurcation, *Phys. Chem. Earth (A)*, 24, 11–12, 953–956, 1999.
- Manganaro, G., Arena, P., and Fortuna, L.: *Cellular Neural Networks. Chaos, Complexity and VLSI Processing*, Springer-Verlag, 1999.
- Mei, C. C. and Yuh, M.: Slow flow of a Bingham fluid in a Shallow Channel of Finite Width, *J. Fluid Mechanics*, 431, 135–160, 2001.
- Miyamoto, H. and Sasaki, S.: Simulating lava flows by an improved cellular automata method, *Computers & Geosciences*, 23, 3, 283–292, 1997.
- Miyamoto, H. and Sasaki, S.: Numerical simulations of flood basalt lava flows: roles of some parameters on lava flow morphologies, *J. Geophys. Res.*, 103, 27 489–27 502, 1998.
- Miyamoto, H. J., Dohm, M., Beyer, R. A., and Baker, V. R.: Fluid dynamical implications of anastomosing slope streaks on Mars., *J. Geophys. Res.*, 109, doi:10.1029/2003JE002234, 2004.
- Perez-Munuzuri, V., Perez-Villar, V., and Chua, L.O.: Autowaves for image processing on a two-dimensional CNN array of excitable nonlinear circuits: flat and wrinkled labyrinths, *IEEE Transactions on Circuit and System – Part I: Fundamental Theory and Applications*, 40, 3, 174–181, 1993.
- Pieri, D. C.: Eruption rate, area, and length relationship for some Hawaiian lava flows, *J. Volc. Geotherm. Res.*, 30, 29–45, 1986.
- Pinkerton, H. and Wilson, L.: Factors controlling the lengths of channel-fed lava flows, *Bull. Volcanol.*, 56, 108–120, 1994.
- Roska, T., Chua, L. O., Kozek, T., Tetzlaff, R., and Puffer, F.: Simulating nonlinear waves and partial differential equations via CNN – Part I: Basic techniques, *IEEE Transactions on Circuit and System – Part I: Fundamental Theory and Applications*, 42, 807–815, 1995.
- Sorbello, A., Sciuto, A., Trovato, R., Currenti, G., Del Negro, C., Fortuna, L., Graziani, S., and Napoli, R.: Lava Flow simulations using Cellular Neural Networks: Part II, in: *Proceedings of Gruppo Nazionale per la Vulcanologia General Assembly 2003*, Roma, Italy, 9–11 June 2003.
- Vicari, A., Currenti, G., Del Negro, C., Fortuna, L., and Napoli, R.: Applicazione delle Reti Neurali Cellulari per la simulazione di flussi lavici: Risultati Preliminari, in: *Proceedings of 22° Convegno Nazionale GNGTS*, Roma, Italy, 18–20 November 2003.
- Wadge, G.: Effusion rate and the shape of a lava flows fields on Mt. Etna, *Geology*, 6, 503–506, 1978.
- Wadge, G., Young, P., and McKendrick, I. J.: Mapping lava flow hazards using computer simulation, *J. Geophys. Res.*, 99, B1, 489–504, 1994.
- Young, P. and Wadge, G.: FLOWFRONT: simulation of a lava flow, *Computers & Geosciences*, 16, 8, 1171–1191, 1990.