IL NUOVO CIMENTO Vol. 28 C, N. 2 Marzo-Aprile 2005
DOI 10.1393/ncc/i2005-10178-1

Parallel Genetic Algorithms for calibrating Cellular Automata models: Application to lava flows(*)

D. D'Ambrosio(1)(3), W. Spataro(1)(3), S. Di Gregorio(1)(3)
G. M. Crisci(2)(3) and R. Rongo(2)(3)

(1) *Dipartimento di Matematica, Università della Calabria - Arcavacata di Rende (CS), Italy*
(2) *Dipartimento di Scienze della Terra, Università della Calabria Arcavacata di Rende (CS), Italy*
(3) *HPCC, Università della Calabria - Arcavacata di Rende (CS), Italy*

(ricevuto il 31 Gennaio 2005; approvato l'1 Giugno 2005; pubblicato online il 23 Settembre 2005)

Summary. — Cellular Automata are highly nonlinear dynamical systems which are suitable for simulating natural phenomena whose behaviour may be specified in terms of local interactions. The Cellular Automata model SCIARA, developed for the simulation of lava flows, demonstrated to be able to reproduce the behaviour of Etnean events. However, in order to apply the model for the prediction of future scenarios, a thorough calibrating phase is required. This work presents the application of Genetic Algorithms, general-purpose search algorithms inspired to natural selection and genetics, for the parameters optimisation of the model SCIARA. Difficulties due to the elevated computational time suggested the adoption a Master-Slave Parallel Genetic Algorithm for the calibration of the model with respect to the 2001 Mt. Etna eruption. Results demonstrated the usefulness of the approach, both in terms of computing time and quality of performed simulations.

PACS 07.05.Tp – Computer modeling and simulation.
PACS 47.11.+j – Computational methods in fluid dynamics.
PACS 02.60.Pn – Numerical optimization.
PACS 91.40.Hw – Lava.
PACS 01.30.Cc – Conference proceedings.

1. – Introduction

Lava flows may be physically described by fluid-dynamics equations, *e.g.*, Navier-Stokes equations for viscous fluids [1]. However, forecasting the development of real events through approximate numerical methods may involve serious difficulties, as lava

(*) Paper presented at CAPI 2004, 8° Workshop sul calcolo ad alte prestazioni in Italia, Milan, November 24-25, 2004.

rheology can range (by cooling) from approximately Newtonian liquids to brittle solids [2]. Technological advances in computing science allowed to considerably extend the range of application for those classes of problems which may be expressed in terms of differential equations systems. Meanwhile, innovative numerical methods emerged from alternative computational paradigms. Among these, Cellular Automata (CA) are particularly suitable for capturing the peculiar characteristics of acentric systems, *i.e.* systems whose evolution can be described by considering the local interactions among their constituent *elementary* parts, that is typical of many fluid-dynamics phenomena [3, 4].

Classical homogeneous CA [5] are space and time discrete dynamical systems; they are based on a regular division of the space in regular cells, identified by integer coordinates (an example of regular hexagonal tessellation for a two-dimensional CA is reported in fig. 1, left). Time is subdivided in steps, which may be considered as constant time intervals. Each cell embeds an identical finite automaton (fa), whose state specifies the cell condition throughout a computational step. By definition, the set Q of fa states must be finite. In case a correspondence is defined between the physical and cellular space (*i.e.* the cell represents a portion of the physical space), examples of cell states may be the values of temperature and altitude a.s.l. or the presence/absence of a particle and its momentum.

The fa input of a cell c is given by the states of m neighbouring cells, usually including c. The neighbourhood conditions are defined by a geometrical relation, which is the same for each cell and invariant in time. It is specified by a m-ple of coordinates, so that the coordinates of the i-th neighbouring cell are simply obtained by adding the coordinates of c to the i-th element of the m-ple (cf. subsect. **2**˙2). The fa have an identical state transition function $\sigma : Q^m \to Q$, which is simultaneously applied to each cell. The state transition function expresses the property of locality for the CA: changes in the fa state are exclusively determined by the states of neighbouring cells. At step 0, fa are in arbitrary states (initial conditions) and the CA evolves changing the state of all fa simultaneously at each step, according to σ.

In the context of the growing CA application fields, an empirical method was developed by our research group *Empedocles* for modelling and simulating macroscopic complex phenomena involving surface flows [6]. The method was successfully applied to: lava flows simulations, concerning Etnean (Italy) events in 1986-1987, 1991, 2001 and 2002 [7-10]; debris/mud flows simulations, concerning the cases of the Tessina area (1992, Italy) [11], Mount Ontake (1984, Japan) [12], the Sarno area (1998, Italy) [13] and the Valle Caudina area (1999, Italy) [14]; pyroclastic flows simulations, concerning the 1991 Mount Pinatubo (Philippines) eruption [15, 16]. Alternative CA models were also proposed: Young and Wadge [17] presented a clever and fast CA simulation code, applicable only to simple lava flow fronts; Miyamoto and Sasaki [18] developed a very interesting CA method for modelling lava flows reducing the problem of the spurious symmetries in CA; successful attempts of simulating flow-type landslides through CA models were eventually devised by Segre and Deangeli [19] and Clerici and Perego [20].

The *Empedocles* empirical approach introduces a set of parameters which may strongly affect the behaviour of the system. As a consequence, besides input data (which define the initial state of the system), even CA parameters have to be provided with the greatest possible accuracy. The quality of input data is imposed by maps defining the region over which the phenomenon evolves, while parameters can undergo an appropriate calibration phase. Until recently [8, 21], calibration was performed *manually*, by visually comparing the outcomes of the simulations with the real event. An alternative solution is provided by automated search techniques, which are able to find satisfactory solutions in a reasonable amount of time. Genetic Algorithms (GAs) [22,23] are general-purpose search algorithms, inspired from genetics and natural selection. Even if they are able to find good solutions in a fair time for certain problems, they might take a long time for others. Nonetheless, Parallel Computing can be adopted with success to strongly increase the search in a GA execution.

As in a previous work concerning debris flow simulation [24], the application of a Master-Slave GA to the calibration of the CA model SCIARA [7] for lava flow simulation is presented. The paper is organized as follows: sect. **2** shows the empirical method for modelling complex macroscopic surface flows together with its application to lava flows, by defining the model SCIARA (in particular, the release SCIARA-hex1 [8]). Genetic Algorithms are illustrated in sect. **3**, while sect. **4** describes shortly the 2001 Mt. Etna event, used as a case-study for parameter optimization, and shows results and comments of the carried out experiments. Conclusions are reported at the end.

2. – Extended CA notion for surface flows and application to lava flows

Modelling spatially extended macroscopic surface flows needs more specifications for permitting a correspondence between the system with its evolution in the physical space/time, on the one hand, and the model with the simulations in the cellular space/time, on the other.

$$A = \langle R, X, Q, P, \sigma, G, \gamma \rangle$$

is the extended CA definition for modelling complex macroscopic surface flows: R, the cellular space, X, the neighbourhood, Q, the set of states, P, the set of global parameters, σ, the transition function, G and γ, respectively, cells with input external to cellular space and their supplementary transition function. In the next subsection, each element of the septuple is defined, commented and applied to the specification of the CA model for lava flows SCIARA (S*imulation by* C*ellular* I*nteractive* A*utomata of the* R*heology of* A*etnean lava flows*; *sciara* means the solidified lava path in Sicilian).

2˙1. *The cellular space* **R**. – $R = \{(x, y) \in \mathbb{Z}, -l_x \leq x \leq l_x, -l_y \leq y \leq l_y\}$ is the set of points, with integer coordinates, of the finite region where the phenomenon evolves. Each point typically identifies a hexagonal cell, instead of a square cell in order to reduce the effects of the spurious symmetries (fig. 1). $\mathbb{Z}$ is the set of integer numbers.

The precise correspondence between the real and cellular space implies that the cell dimension must be fixed. It is specified by the cell apothem $a \in P$.

2˙2. *The neighbourhood* **X**. – X=$\{(0, 0), (1, 0), (0, 1), (-1, 1), (-1, 0), (0, -1), (-1, -1)\}$ is the hexagonal neighbourhood; the coordinates of the neighbour cells of a cell c are obtained adding its coordinates to the X couples; neighbourhood includes the cell itself (called central cell) with index 0 and the six adjacent cell with indexes 1, . . . ,6 (fig. 1).

2˙3. *The set of states* **Q**. – The state of the cell must account for all the characteristics, relative to the space portion corresponding to the cell, which are assumed to be relevant to the evolution of the system. Each characteristic corresponds to a substate; permitted values for a substate must form a finite set: $Q = Q_1 \times Q_2 \times \ldots \times Q_n$ is the set of the possible values of a cell state, given by the Cartesian product of the values of the substates $Q_1, Q_2, \ldots, Q_n$. The substate value is considered always constant inside the cell.

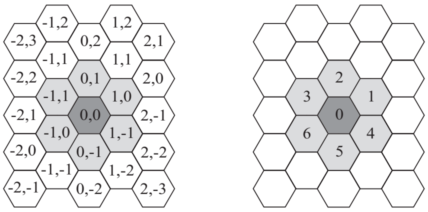


Fig. 1. – The SCIARA hexagonal neighbourhood. Left: the coordinates of the neighbouring cells with respect to the central one. Right: the relative indexes.

The cellular space should be three dimensional, but a reduction to two dimensions is allowed when quantities concerning the third dimension (the height) may be included among the substates of the cell in a phenomenon concerning the earth surface.

The SCIARA substates are: Q_a, the cell altitude; Q_{th}, the thickness of lava inside the cell; Q_T is the lava temperature inside the cell; Q^6_{of}, the six possible outflows from the central cell toward the adjacent cells; Q^6_{if}, the six possible inflows from the adjacent cells toward the central cell. Outflows and inflows are expressed in terms of thickness.

2˙4. *The set of global parameters* **P**. – The set of the global parameters P in SCIARA is given by: a, the apothem of the cell; p_t, the time correspondence of a step; *cool*, the cooling parameter; TV, TS and TI, the lava temperature at the vent, at the solidification and intermediate value, respectively; *adhV*, *adhS* and *adhI* the adherence (*i.e.* unmovable thickness of lava) at the emission temperature, at the solidification temperature and at intermediate temperature, respectively [8].

2˙5. *The transition function* $\boldsymbol{\sigma}$. – The state transition function σ must account for all the processes, which are assumed to be relevant to the system evolution, which is specified in terms of changes in the states values of the CA space. As well as the state of the cell can be decomposed in substates, the transition function σ may be split into elementary processes, defined by the functions $\sigma_1, \sigma_2, \ldots, \sigma_p$ with p being the number of the elementary processes.

The elementary processes are applied sequentially according a defined order. Each elementary process involves the update of the states of the cells. The application of all the elementary processes constitutes a CA step.

A precise time correspondence $p_t \in P$ for a CA step must be fixed in order to compare the system evolution to the simulation steps.

$\sigma : Q^7 \to Q$ is the deterministic transition function for SCIARA; it is composed by the following elementary processes in order:

– σ_{lf} computes the lava outflows;

– σ_{mix} determines the mixing of the remaining lava inside the cell with inflows;

– σ_{cool} computes the lava cooling due to radiation and solidification effects.

Determination of the lava flows. Outflows must minimise the differences in height (altitude plus lava thickness) in the neighbourhood after the lava distribution from the central cell to the adjacent ones. The differences in height (h with indexes related to the cells) are computed according to the following formula [25]:

$$\sum_{i<j} \mid h_i - h_j \mid, \qquad 0 \le i < 6, \quad 0 < j \le 6\,.$$

The rheological resistance of lava increases as temperature decreases. Because of complexities which are inherent in specifying lava rheology and its variation with temperature [8], resistance was modelled in terms of an adherence effect, measured by adh, which represents the amount of lava (expressed as a thickness) that cannot flow out of a cell because of rheological resistance. adh is assumed to vary with temperature T according to a simple inverse exponential relation $adh = ce^{-kT}$, where c and k are positive constants such that $adhV = ce^{-kTV}$ and $adhS = ce^{-kTS}$.

The quantity of lava, that can be distributed from the central cell towards the adjacent ones, is given by the lava thickness minus the adherence.

Lava mixing inside the cell. Lava mixing inside the cell determines:

- the new lava thickness inside the cell, by just adding inflows and subtracting outflows to the previous lava thickness;
- the new temperature of the mixed lava, by computing the weighted average of the temperatures of inflows and remaining lava in the cells over their thicknesses.

Lava cooling and solidification. The following approximated physical formula for cooling by radiation is applied [7]:

$$nT = T/\sqrt[3]{1 + (T^3 \cdot cool \cdot p_t/th)}\,,$$

where th is lava thickness inside the cell, T and nT are the old and the new temperature, respectively. The solidification process depends on lava temperature; it is trivially modelled by adding solidified lava thickness to the cell altitude, when the solidification temperature is reached.

2˙6. *Cells in* **G** *with external input to the cellular space.* – Sometimes, a kind of input from the *external world* to the cells of the CA must be considered; it accounts for describing an external influence which cannot straightforwardly be described in terms of local rules (*e.g.*, the lava alimentation at the vents) or for some kind of probabilistic approach to the phenomenon.

$G = G_1 \cup G_2 \cup \ldots \cup G_s$ is the set of cells, which undergo to the influences of the external world; s external influences are here considered, each one defines a subregion G_i $(1 \le i \le s)$ of the cellular space, where the influence is active. The situation is simple for SCIARA: the lava emission at the vents is the unique external influence; $G_1 \subseteq R$ is the set of cells corresponding to the vents.

2˙7. *The supplementary function* γ *for cells in* **G**. – $\gamma : \mathbb{N} \times G \times Q \to G \times Q$ expresses the external influences to cells of G in the cellular space; it determines the variation of the state Q for the cells in G. $\mathbb{N}$, the set of natural numbers, is here referred to the steps of CA. γ is specified by the sequential applications of the s functions $\gamma_1 : \mathbb{N} \times G_1 \times Q \to G_1 \times S_1$, $\gamma_2 : \mathbb{N} \times G_2 \times Q \to G_2 \times S_2$, $\ldots$, $\gamma_s : \mathbb{N} \times G_s \times Q \to G_s \times S_s$, where S_1, S_2, $\ldots$, S_s

are Cartesian products of elements of $S = \{Q_1, Q_2, \ldots, Q_n\}$. The function γ is applied before σ.

The only external influence defined in SCIARA, γ_1, accounts for the lava emissions according to data of real events or data of hypothetical events for scenarios creation: $\gamma : \mathbb{N} \times G_1 \times Q_{\mathrm{th}} \times Q_T \to G_1 \times Q_{\mathrm{th}} \times Q_T$ specifies the emitted lava at temperature TV from each source (vents) cell $c \in G_1$ at the CA step $t \in \mathbb{N}$. γ_1 determines a new lava thickness inside the cell by just adding the emitted lava thickness to the previous lava thickness; the new temperature is computed by the weighted average of the lava temperature inside the cell and the emitted lava at TV temperature over their thicknesses.

Let us remark, after the model exhibition, that some parameters cannot be measured (or carefully measured) due to their own nature, either for fundamental or for practical reasons: their values must be determined by comparing the model outcome with a set of experimental data. The apothem of the cell is the only prefixed parameter, as it value is imposed by input maps detail. The time correspondence of a CA step, p_t, is tightly linked with the cell apothem and the rheology of the specific phenomenon to be simulated. On this basis, it is possible to hypothesize an adequate variation range. The lava temperature at the vent, TV, and at the solidification, TS, cannot be measured with adequate precision: for instance, in general, different values may be provided by different observers. The intermediate temperature, TI, is an empirical parameter, introduced in order to better approximate the adherence computation with respect to previous versions of the model [7]. It can experimentally be determined by evaluating the response of the model with respect to different possible parameter values. As temperature parameters, the three adherence ones, $adhV$, $adhS$ and $adhI$, cannot be obviously determined by laboratory experiments; however, a range of possible values may be deduced by field observations. The cooling parameter, *cool*, has a precise physical meaning in the radiation equation. However, the context where the radiation equation is applied is complex: in general, lava determines a crust that lowers the cooling parameter. Moreover, crust forming and fissuring is a chaotic process, so that the cooling parameter we used may be considered as a kind of average value. Thus, for these considerations, the model parameters need to be tailored by comparing model predictions with experimental data and by modifying them in order to achieve a satisfactory agreement.

3. – Genetic algorithms

Diverse scientific fields, such as Physics, Biology and Economy, often have to deal with the classical problem of optimization. Genetic Algorithms are a part of evolutionary computing, which is a rapidly growing area of artificial intelligence. They may represent a valid methodology for solving search problems for which standardised optimisation techniques do not exist or are difficult to apply. GAs are general-purpose search algorithms inspired by Darwin's Theory of Evolution [22]. Problems are solved by an evolutionary process resulting in a best (*i.e.* fittest) solution (*i.e.* survivor); in other words, the solution is said to be evolved. GAs were formally introduced in the United States in the 1970s by John Holland at the University of Michigan. To use a genetic algorithm, one must represent a solution to a problem as a genome (or chromosome). The genetic algorithm then creates a population of solutions and applies simple random operators to *evolve* a good solution. Accordingly, the GA must investigate the so-called *search space*, defined as the set of all possible values that the genotype can assume. Evaluation of individuals is executed by choosing a suitable *fitness function*, which determines the goodness of the individual. The *selection* operator, which represents a metaphor of Darwinian Natural Selection, chooses individuals that undergo reproduction, by means of genetic operators, to form a new population of offspring. Classic genetic operators are *crossover* and *mutation*: they represent a metaphor of sexual reproduction and of genetic mutation, respectively. Eventually, as stated by the fundamental theorem of genetic algorithms [22], any change that actually increases the individual's fitness will be more likely to be preserved over the selection process, thus obtaining better generations. The iterative process continues until one of the possible termination criteria is met: in general, these include the attainment of a known optimal or acceptable solution level, if a maximum number of generations have been performed (usually empirically determined) or if a given number of generations without fitness improvement occurs. A GA is usually characterised by diverse parameters. The choice of the population size (*i.e.* the constant number of individuals forming initial and evolved populations) is one of the most important parameter, reflecting the size and complexity of the problem. Other parameters include the maximum number of generations to be performed, a crossover and a mutation probability. Furthermore, a selection method and a replacement strategy (*e.g.*, which and how many individuals of the old population must be replaced with the new offspring) must be specified. One of the most utilised selection methods is the tournament selection; one of the most adopted replacement strategies is the elitist one. Such aspects, as they were adopted in the present work, will be explained in the following section.

More in-depth knowledge of GAs can be found in [23].

3˙1. *Parameter optimization of Cellular Automata through Genetic Algorithms.* – Calibration is an essential phase of the development of a model which can allow its application for prediction purposes. Until recently, this phase was performed *manually*, by simply assigning initial reasonable values to parameters and analyzing the results provided by the model from a qualitative point of view. Subsequently, modifications to one or more parameters were undertaken until a satisfying simulation was accomplished. However, this method presented some problems as the lack of a quantitative evaluation of the results or the subjectiveness of the simulation assessment. In order to overcome these difficulties, an automated optimization method was devised, by adopting GAs. As previously stated, an appropriate fitness function must be devised in order to evaluate the goodness of a given simulation. In the present study, the fitness function e_1 was taken into account; it gives a measure of the overlapping (in terms of areal extent) between the real and simulated event.

Definition 3.1 (Definition of fitness function e_1). Let us denote with R and S the sets of CA cells affected by the real and simulated event, respectively. Let $m(R \cap S)$ and $m(R \cup S)$ be the measure of their intersection and union, respectively. We define the fitness function e_1 as follows:

$$e_1 = \sqrt{\frac{m(R \cap S)}{m(R \cup S)}}\,.$$

Note that the function e_1 gives values belonging to the interval $[0, 1]$. Its value is 0 if the actual and simulated events are completely disjoint, being $m(R \cap S) = 0$; it is 1 in case of a perfect overlap, being $m(R \cap S) = m(R \cup S)$. As a consequence, the goal for the GA is to find a set of CA parameters that maximise e_1.

4. – Implementation and calibration

In this work, GAs were adopted to optimize parameters of the CA model SCIARA, by considering the lava event which occurred on Mt. Etna (Sicily) in 2001.

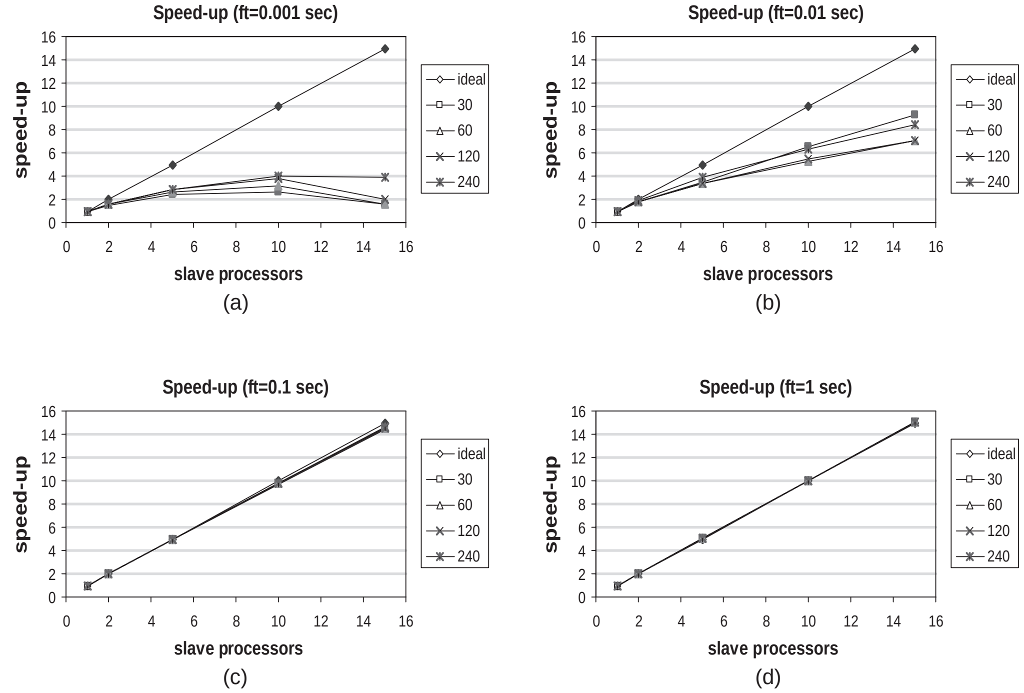
4˙1. *Optimization of the July 2001 Etnean eruption.* – At 3.00 AM on July 18th, 2001, an eruption started from the fracture of Mount Calcarazzi, on the southern flank of Mt. Etna (Sicily), 2100 m a.s.l. The event was fed by a medium lava flow rate (ca. 7 m^3/s) and, due to the steep descent of the terrain in that area, pointed southwards creating the main danger for the towns of Nicolosi and Belpasso: it was, in its maximum extension, only 4 km away from Nicolosi.

Such event was considered as case of study for the model calibration. The CA parameters to be optimized were encoded into the GA genotype as bit strings. A total of 100 individuals were considered to form the GA population and 8 bits were used to encode each parameter. On the basis of previous empirical attempts on assigning values to SCIARA parameters [6], initial ranges $[a_i, b_i]$ $(i = 1, 2, \ldots, 8)$ within which the values of the CA parameters p_i are allowed to vary were individuated. Thus, the GA search space became

$$\begin{aligned} S = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_8, b_8] = [30, 120] \times [1350, 1400] \times [1050, 1200] \times \\ \times [1250, 1340] \times [0.1, 1] \times [7, 15] \times [1.1, 3] \times [10^{-18}, 10^{-13}] \end{aligned}$$

At each step only the worst individuals (30) of the old population are replaced (*i.e.* the model adopts a steady-state replacement scheme), while the remaining (70) individuals, required to form the new population, are copied from the old one, choosing the best (*i.e.* the model is elitist as, step after step, it preserves the best individuals). While classical Holland's crossover and mutation operators have been employed, with probability 0.8 and 1/64 respectively, a tournament operator was adopted as selection scheme instead of the original proportional selection one [22]. The tournament selection consists of a series of tournaments in which two individuals are randomly selected, and the winner is chosen according to a prefixed probability, which must be greater for the fittest individual. In this work, the tournament selection probability has been set to 0.6, on the basis of previous experience on analogous experiments.

4˙2. *Parallel implementation.* – It is quite straightforward to make a GA implementation able to efficiently exploit several CPUs simultaneously. Accordingly, Parallel Computing represents a useful tool to speed up GAs executions [26]. Several examples of Parallel Genetic Algorithms (PGAs) have been proposed in literature, such as Master-Slave (Synchronous and Asynchronous), Static Subpopulation (with or without Migration), Dynamic Demes, and others. A PGAs' Taxonomy can be found in [27]. One of the simplest Parallel Genetic Algorithm is represented by the Synchronous Master-Slave model (Master-Slave in the following), in which a processor (the master) executes the GA steps (selection, population replacement, crossover and mutation), while several others (the slaves) evaluate the individual's fitness. The algorithm is synchronous as the master waits to receive the fitness values of each individual of the population before generating the new one; it represents a merely parallelization of the classical Genetic Algorithm, thus preserving the same dynamical behavior. In this work, calibration experiments were performed on a Nec TX7 machine composed by 4 quadri-processors Itanium class nodes, with an overall RAM memory of 32 GB and a performance of 64 GFLOPS.



(a)

(b)

(c)

(d)

Fig. 2. – Experiments carried out in order to determine speed-up. Linear speed-up is obtained starting from $f_t = 0.1$ seconds.

In order to assess the GA performance, several experiments have been carried out with the aim of supplying a possible linear *speed-up* trend. Linear speed-up permits to reduce execution time of a factor equal to the number of processing nodes. In general, speed-up is defined as the ratio of the sequential and parallel execution times. Thus if N_p processors are utilised, the ideal parallel time is $T_\mathrm{p} = T_\mathrm{s}/N_\mathrm{p}$, where T_s represents the sequential execution time. As a consequence, the ideal speed-up is $S = T_\mathrm{s}/T_\mathrm{p} = N_\mathrm{p}$: in such a case, the parallel program is linearly scalable. The tests were performed by executing 100 generations of four classic Holland's GA, respectively characterised by population sizes $n = 30$, 60, 120 and 240. For each GA, a *fictitious* fitness function was considered: it is simply idle for time intervals $f_t = 0.001$, 0.01, 0.1 and 1 seconds. For each n and f_t combination, a GA run was executed considering $N_p = 1$, 2, 5, 10, 15 slave processors, and the sequential and parallel execution times were estimated in order to determine the speed-up. Results demonstrated the linear scalability of the GA on the considered parallel architecture practically for $f_t \geqslant 0.1$ seconds (fig. 2).

Since typical times needed to evaluate a SCIARA candidate solution are generally longer (of the order of tens of minutes), the adopted Master-Slave GA may guarantee a quasi-linear speed-up. Achieving linear speed-up requires in fact (as in the previous tests) a constant execution time for the fitness function for each individual within each GA step. At the contrary, when the evaluation of the fitness function needs different times for different candidate solutions, a typical bottle-neck effect may be present, as the master processor has to wait for the slowest processor to finish its fitness evaluation. An exact estimation of the bottle-neck effect for our optimization task would require too

TABLE I. – *Parameters of the model SCIARA. For each parameter, the table reports the number of bits adopted for the encoding of the genotype, its GA variation range and the best evolved value.*

Parameter	Number of bits	Variation range	Best value
$p_1 = p_t$	$k_1 = 8$	$[a_1, b_1] = [30, 120]$	117.5
$p_2 = TV$	$k_2 = 8$	$[a_2, b_2] = [1350, 1400]$	1350.5
$p_3 = TS$	$k_3 = 8$	$[a_3, b_3] = [1050, 1200]$	1088.8
$p_4 = TI$	$k_4 = 8$	$[a_4, b_4] = [1250, 1340]$	1338.5
$p_5 = adhV$	$k_5 = 8$	$[a_5, b_5] = [0.1, 1]$	0.9
$p_6 = adhS$	$k_6 = 8$	$[a_6, b_6] = [7, 15]$	12.0
$p_7 = adhI$	$k_7 = 8$	$[a_7, b_7] = [1.1, 3]$	2.8
$p_8 = cool$	$k_8 = 8$	$[a_8, b_8] = [10^{-18}, 10^{-13}]$	10^{-17}

much time since a typical SCIARA simulation, as stated above, requires tens of minutes. However, an approximated worst-case speed-up evaluation can be derived by considering that, during the optimization experiments (in which 2 individuals were evaluated for each of the 15 slave processors), the execution times of the fastest individual, t_{f}, was about 14 minutes and the slowest, t_{s}, was about 16 minutes, with a difference of about $t_{\mathrm{d}} = t_{\mathrm{s}} - t_{\mathrm{f}} = 2$ minutes. By supposing that at each GA step only one individual requires a time t_{s} for its evaluation, while all the others are evaluated in a time t_{f}, and neglecting communication times among processors, the speed-up can be written as

$$S = \frac{T_{\mathrm{s}}}{T_{\mathrm{p}}} = \frac{G[(2N_{\mathrm{p}} - 1)t_{\mathrm{f}} + t_{\mathrm{s}}]}{G(t_{\mathrm{f}} + t_{\mathrm{s}})} = \frac{2N_{\mathrm{p}}t_{\mathrm{f}} + t_{\mathrm{d}}}{2t_{\mathrm{f}} + t_{\mathrm{d}}},$$

where G is the number of GA steps and N_{p} the number of slave processors (so that $2N_{\mathrm{p}}$ represents the number of individuals to be evaluated at each generation). In our case, since communication can be considered negligible for fitness execution times greater than 0.1 seconds (cf. fig. 2), $S \simeq 14$ (in spite of 15 when $t_{\mathrm{d}} = 0$), representing a good result for our particular optimization task.

4˙3. *Experimental results.* – By considering the 2001 Nicolosi case-study, four runs of 100 steps were carried out, each one with a different randomly generated initial population. The overall GA optimisation experiment lasted less than 3 days. It is worth to note that on a sequential machine, the same experiment would have lasted more than 2 months.

Table I illustrates the SCIARA parameters which were optimized using GAs, together with their variation range and values obtained from the best evolved individual. The corresponding CA simulation is shown in fig. 3. As the figure shows, the parameter-optimised simulation did not differ significantly from the real case. As confirmed by the value of the fitness function, $e_1 = 0.74$, a significant improvement is achieved with respect to the previously manually-obtained results, for which the value of e_1 was at most 0.65. The goodness of the simulation is also confirmed in terms of run-out, as the travel distance from the source of the simulated event is practically the same as the real one.

Even if only areal information is taken into account in this first study, further information regarding other physical data like lava thickness and temperature, could improve the overall GA search, by avoiding the GA to be stuck in possible local optima. This

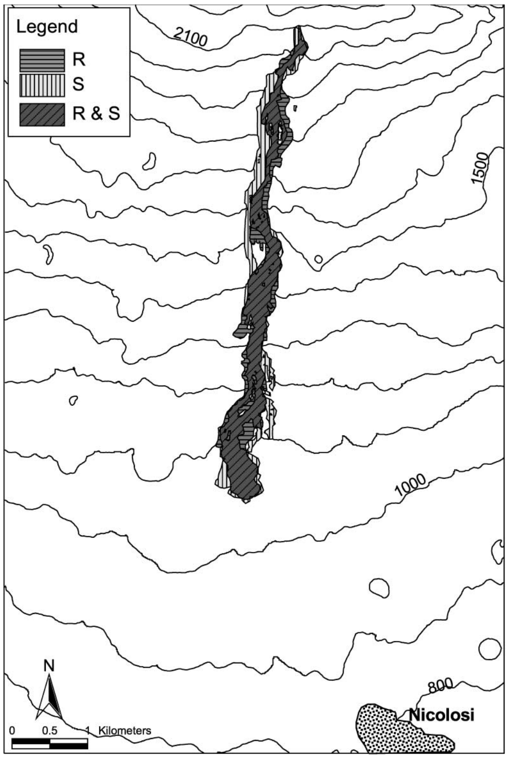


Fig. 3. – Comparison between the 2001 Mt. Etna event and the best SCIARA simulation, as obtained by adopting the parameters listed in table I. Legend key: R specifies the area affected by the real event; S specifies the area affected by the simulation; R&S specifies the area affected by the intersection between real and simulated events.

occurs typically when, for instance, in relation with the particular adopted fitness function, different sets of CA parameters, whose values can be significantly diverse, produce equivalent simulated phenomena. In the present study, such equivalence concerns the areal extents of both simulated and real events. However, by introducing further information on the real event in each cell, such as lava thickness or temperature throughout its path, local optima could tend to diminish, favouring the GA convergence.

5. – Conclusions

We presented an application of a Master Slave Genetic Algorithm to the calibration of the Cellular Automata model SCIARA for lava flows simulation, by considering the 2001 Nicolosi (Sicily, Italy) eruption case study. In general, calibration is needed in order to reliably apply computational methods which depend on sets of parameters. The adoption of a parallel model for parameter optimisation dramatically speeds up computational times: nonetheless, the methodology can be applied to other natural phenomena (*e.g.*, landslides, forest fires, etc) and models (*e.g.*, SCIDDICA for landslides [24], by the same research group), when systematic parameter evaluation is needed. Results demonstrated GAs reliability and, consequently, the SCIARA efficacy in the simulation of Etnean lava flows. However, further improvements could be achieved. For instance, the adoption of a more reliable topographic map could sensibly improve the model calibration. Besides this, other fitness functions which take into account also other physical properties (*e.g.*, lava thickness, temperature, etc.) might be useful for a better parameter optimization. Among future developments, a thorough investigation of these problems will be addressed.

* * *

This research has been partially funded by the Italian Ministry of Instruction, University and Research, project FIRB no. RBAU01RMZ4, *Simulazione dei flussi lavici con gli automi cellulari* (Lava flow simulation by cellular automata).

REFERENCES

[1] McBirney A. R. and Murase T., *Annu. Rev. of Earth and Planetary Sciences*, **12** (1984) 337.

[2] Johnson A. M., *Physical Processes in Geology* (Freeman, Cooper, San Francisco) 1973.

[3] Chopard B. and Droz M., *Cellular Automata Modelling of Physical Systems* (Cambridge University Press) 1998.

[4] Succi S., *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond* (Oxford University Press) 2004.

[5] Worsch T., *Future Generation Computer Systems*, **16** (1999) 157.

[6] Di Gregorio S. and Serra R., *Future Generation Computer Systems*, **16** (1999) 259.

[7] Barca D., Crisci G. M., Di Gregorio S. and Nicoletta F., *Transport Theory and Statistical Physics*, **23** (1994) 195.

[8] Crisci G. M., Rongo R., Di Gregorio S. and Spataro W., *J. Volcanology and Geothermal Research*, **132** (2004) 253.

[9] Crisci G. M., Di Gregorio S., Rongo R. and Spataro W., *Volcano Laboratory 2004 Geophysical Monograph Series*, **143** (2004) 343.

[10] Avolio M. V. , Crisci G. M., Di Gregorio S., Rongo R., Spataro W. and Trunfio G. A., to be published in *Computers* and *Geosciences*.

[11] Avolio M. V., Di Gregorio S., Mantovani F., Pasuto A., Rongo R., Silvano S. and Spataro W., *Int.J. Appl. Earth Observ. Geoinform.*, **2** (2000) 41.

[12] Di Gregorio S., Rongo R., Siciliano C., Sorriso-Valvo M. and Spataro W., *Physics and Chemistry of the Earth (A)*, **24** (1999) 97.

[13] D'Ambrosio D., Di Gregorio S., Iovine G., Lupiano V., Rongo R. and Spataro W., *Geomorphology*, **54** (2003) 91.

[14] Iovine G., Di Gregorio S. and Lupiano V., *Natural Hazards and Earth System Sciences*, **3** (2003) 457.

[15] Crisci G. M., Di Gregorio S., Rongo R. and Spataro W., *Future Generation Computer System*, **21** (2005) 1019.

[16] Avolio M. V., Crisci G. M., Di Gregorio S., Rongo R. and Spataro W., to be published in *Computers* and *Geosciences*.

[17] Young P. and Wadge G., *Computer* and *Geosciences*, **16** (1990) 1171.

[18] Miyamoto H. and Sasaki S., *Computer* and *Geosciences*, **23** (1997) 283.

[19] Segre E. and Deangeli C., *Nonlinear Processes in Geophysics*, **2** (1995) 1.

[20] Clerici A. and Perego S., *Geomorphology*, **33** (2000) 1.

[21] D'Ambrosio D., Di Gregorio S. and Iovine G., *Natural Hazards and Earth System Sciences*, **3** (2003) 545-559.

[22] Holland J.H., *Adaption in Natural and Artificial Systems* (University of Michigan Press, Ann Harbor) 1975.

[23] Mitchell M., *An Introduction to Genetic Algorithms* (MIT Press, Cambridge, Massachusetts) 1996.

[24] D'Ambrosio D., Spataro W. and Iovine G., to be published in *Computer & Geosciences.*

[25] Avolio M. V., *Esplicitazione della velocit per la Modellizzazione e Simulazione di Flussi di Superficie Macroscopici con Automi Cellulari ed Applicazioni alle Colate di Lava* (PhD Thesis, University of Calabria, Italy) 2005.

[26] Cant-Paz E., *Efficient and accurate Parallel Genetic Algoritms* (Kluwer Academic Publishers, Dordrecht, The Netherlands) 2000.

[27] Nowostawski M. and Poli R., in *Proceedings of the Third International conference on knowledge-based intelligent information engineering systems*, edited by L. C. Jain (IEEE Press, Adelaide) 1999, pp. 88-92.