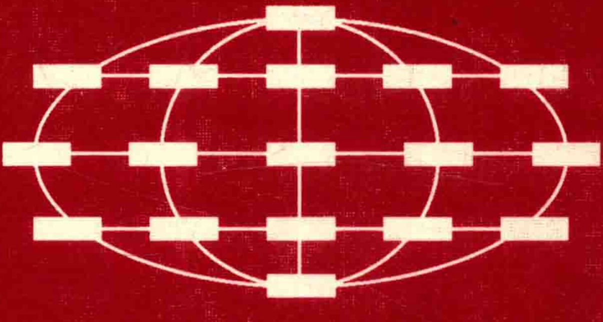
Victor Malyshkin (Ed.)

LNCS 2127

# Parallel Computing Technologies

6th International Conference, PaCT 2001
Novosibirsk, Russia, September 2001
Proceedings



Springer

Victor Malyshkin (Ed.)

# Parallel Computing Technologies

6th International Conference, PaCT 2001
Novosibirsk, Russia, September 3-7, 2001
Proceedings

Springer

Volume Editor

Victor Malyshkin
Supercomputer Software Department
ICM and MG SB RAS
pr. Lavrentiev 6, 630090 Novosibirsk, Russia
E-mail: malysh@ssd.sscc.ru

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Parallel computing technologies : 6th international conference ; proceedings
/ PaCT 2001, Novosibirsk, Russia, September 3 - 7, 2001. Victor Malyshkin
(ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Hong Kong ; London ;
Milan ; Paris ; Tokyo : Springer, 2001
(Lecture notes in computer science ; Vol. 2127)
ISBN 3-540-42522-5

CR Subject Classification (1998): D, F.1-2, C, I.6

ISSN 0302-9743
ISBN 3-540-42522-5 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

http://www.springer.de

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN 10839972 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 2127
Edited by G. Goos, J. Hartmanis and J. van Leeuwen

Springer
*Berlin*
*Heidelberg*
*New York*
*Barcelona*
*Hong Kong*
*London*
*Milan*
*Paris*
*Tokyo*

# Preface

The PaCT-2001 (Parallel Computing Technologies) conference was a four-day conference held in Akademgorodok (Novosibirsk), September 3–7, 2001. This was the sixth international conference in the PaCT series, organized in Russia every odd year.

The first conference, PaCT-91, was held in Novosibirsk (Academgorodok), September 7–11, 1991. The next PaCT conferences were held in Obninsk (near Moscow), August 30 – September 4, 1993; in St.Petersburg, September 12–15, 1995; in Yaroslavl September 9–12, 1997; and in Pushkin (near St.Petersburg) from September 6–10, 1999. The PaCT proceedings are published by Springer-Verlag in the LNCS series.

PaCT-2001 was jointly organized by the Institute of Computational Mathematics and Mathematical Geophysics of the Russian Academy of Sciences (Novosibirsk), the State University, and the State Technical University of Novosibirsk.

The purpose of the conference was to bring together scientists working with theory, architecture, software, hardware, and solution of large-scale problems in order to provide integrated discussions on parallel computing technologies. The conference attracted about 100 participants from around the world. Authors from 17 countries submitted 81 papers. Of those submitted, 36 papers were selected for the conference as regular ones; there were also 4 invited papers. In addition there were a number of posters presented. All the papers were internationally reviewed by at least three referees. As usual a demo session was organized for the participants.

Many thanks to our sponsors: the Russian Academy of Sciences, the Russian Fund for Basic Research, the Russian State Committee of Higher Education, the European Commission (Future and Emerging Technologies, Directorate General–Information Society) for their financial support. Organizers highly appreciated the help of the Association Antenne-Provence (France).

June 2001

Victor Malyshkin
Novosibirsk, Academgorodok

# Organization

PaCT 2001 was organized by the Supercomputer Software Department of the Institute of Computational Mathematics and Mathematical Geophysics SB RAS in cooperation with the State University of Novosibirsk and the State Technical University of Novosibirsk.

## Program Committee

| | |
|---|---|
| V. Malyshkin | Program Chair (Russian Academy of Sciences) |
| F. Arbab | (Centre for MCS, The Netherlands) |
| O. Bandman | (Russian Academy of Sciences) |
| A. Bode | (Technical University of Munich, Germany) |
| T. Casavant | (University of Iowa, USA) |
| P. Ciancarini | (University of Bologna, Italy) |
| P. Degano | (State University of Pisa, Italy) |
| A. Doroshenko | (Academy of Sciences, Ukraine) |
| D. Etiemble | (University of Toronto, Canada) |
| B. Goossens | (University Paris 7 Denis Diderot, France) |
| S. Gorlatch | (Technical University of Berlin, Germany) |
| A. Hurson | (Pennsylvania State University, USA) |
| V. Ivannikov | (Russian Academy of Sciences) |
| Y. Karpov | (State Technical University, St.Petersburg) |
| B. Lecussan | (State University of Toulouse, France) |
| J. Li | (University of Tsukuba, Japan) |
| T. Ludwig | (Technical University of Munich, Germany) |
| G. Mauri | (University of Milan, Italy) |
| N. Mirenkov | (The University of Aizu, Japan) |
| I. Pottosin | (Russian Academy of Sciences) |
| M. Raynal | (IRISA, Rennes, France) |
| B. Roux | (Institut de Mecanique des Fluides de Marseilles, France) |
| G. Silberman | (IBM TJ Watson Research Center, USA) |
| P. Sloot | (University of Amsterdam. The Netherlands) |
| V. Vshivkov | (Russian Academy of Sciences) |

## Organizing Committee

| | |
|---|---|
| N. Dikansky | |
| V. Malyshkin | |
| B. Mikhailenko | |
| G. Shvedenkov | |
| A. Vostrikov | |
| S. Achasova | Financial Director |
| O. Bandman | Publication Chair |
| A. Selikhov | Secretariat |
| N. Kuchin | |
| S. Piskunov | |
| S. Pudov | |
| A. Usov | |
| I. Virbitskaite | |

## Referees

S. Achasova
F. Arbab
P. Baldan
O. Bandman
D. Beletkov
A. Bernasconi
O. Bessonov
A. Bode
H. Cai
T. Casavant
J. Cazin
P. Ciancarini
M. Danelutto
P. Degano
A. Doroshenko
D. Etiemble
K. Everaars
E. Fleury
P. Fraigniaud
F. Gartner
C. Germain
B. Goossens
R. Gori
S. Gorlatch
J. Guillen-Scholten
A. Hameurlain
J.-M. Helary
T. Herault
R. Hoffmann
A. Hurson
V. Il'in
H.A. Jacobsen
N. Kalinina
M. Kandemir
Yu. Karpov
V. Korneev
N. Kuchin
M. Kutrib
Sh. Kutten
R. Lechtchinsky
B. Lecussan
J. Li
T. Ludwig
T. Lyubimova
V. Markova
G. Mauri
N. Mirenkov
V. Morozov
A. Mostefaoui
V. Narayanan
A. Nepomniaschaya
V. Nepomniaschy
M. Ostapkevich
S. Pallotino
S. Pelagatti
B. Philippe
S. Piskunov
T. Plaks
I. Pottosin
P. Raghavan
M. Raynal
S. Rogazinsky
B. Roux
S. Sharyi
G. Silberman
P. Sloot
A. Vasilache
A. Vazhenin
I. Virbitskaite
V. Vshivkov
A. Zavanella
E. Zehendner
V. Zissimopoulos

# Table of Contents

## Theory

## Software and Architecture

## Applications

# A Hybrid Approach to Reaction-Diffusion Processes Simulation

Olga Bandman

Supercomputer Software Department
ICMMG, Siberian Branch
Russian Academy of Science
Pr. Lavrentieva, 6, Novosibirsk, 630090, Russia
bandman@ssd.sscc.ru

**Abstract.** A hybrid approach for simulating reaction-diffusion processes is proposed. It combines into a single iterative procedure Boolean operations of Cellular Automata Diffusion with real number computation of nonlinear reaction function. The kernel of the proposed approach is in constructing methods for transforming reals into spatial distribution of Boolean values. Two algorithms are proposed and illustrated by the simulation of some well studied typical reaction-diffusion phenomena. Computational features of the methods are discussed and problems for future research are outlined.

## 1 Introduction

There is a number of well known Cellular Automata diffusion and Gas-Lattice models [1,2,3], as well as some trials to find cellular automata simulating kinetic and chemical processes. Following [4], all these models should be considered as "alternatives rather than approximations of Partial Differential Equations (PDE) solutions". These discrete models have a number of computational advantages, the most important being the absolute stability of computation and the absence of rounding off errors. These properties attract the mathematicians, while the specialists in chemistry, biology and physics are interested in creating models of phenomena, which have no mathematical description at all. Such Cellular Automata (CA) are constructed on the basis of kinetic or chemical microscopic dynamics. Boolean cell states simulate the existence or the absence of an abstract particle (molecule, velocity component, concentration, etc.) at certain points of time and space. Cell operations are represented as Boolean functions of states in the cell neighborhood. To obtain physical interpretation of Boolean results, a sum of state values over an area around each cell is calculated. Two prominent examples are a deterministic chemical CA, proposed in [5], and a "Stochastic Cellular Automaton" from [6], which are intended for simulation chemical processes in active media. In [7] a reaction-diffusion CA is presented, based on a neurolike model, whose elementary automaton executes a threshold function and has a refractory period after the active state. In [8,9] many very interesting industrial application of Cellular-Automata models are presented.

V. Malyshkin (Ed.): PaCT 2001, LNCS 2127, pp. 1–16, 2001.

An important problem not yet completely solved in the above approaches is to prove the correspondence of the cellular array evolution to the modeled phenomenon, as well as the way of accounting physical parameters (density, viscosity, diffusion coefficient, pressure, etc) in the array function parameters. The most correct approach to solve these problems might be a natural experiment which, however, is impractical. But such experiments are sometimes impractical. Certain particular results have been obtained theoretically for the CA-diffusion with Margolus neighborhood [2] and for Gas-Lattice FHP-model [10]. In both cases the proofs of the CA evolution correspondence to the modeled phenomenon are done by reducing the CA to the PDE of the modeled phenomenon.

There are many problems also in studying reaction-diffusion processes by PDE analysis. They are investigated literally by the piece (equations of Gordon, Fitz-Nagumo, Belousov-Zhabotinsky, etc.), and with much difficulty, because analytical solutions are impossible due to the nonlinearity, and numerical methods are limited by stability and accuracy problems [12,13].

Unfortunately up to now no method is known for determining a CA-model of process when its PDE description is known. The latter is a system of first order PDEs, having in their right sides two additive terms: 1) a Laplacian to represent the diffusion, and 2) a nonlinear function to represent the reaction (in chemistry) or the advective process (in hydrodynamics), phase conversion (in crystallization), population evolution (in ecology). The first is perfectly modeled by CA, and the second is easy to count without the danger to make the computation unstable.

From the above it follows, that it makes sense to find methods which combine CA-diffusion with calculation of reaction function in reals. We propose to state the problem as follows: given a reaction-diffusion PDE, a discrete cellular algorithm is to be constructed whose evolution approximate that of finite-difference PDE. Obviously, it should be an iterative algorithm, at each step performing the operation of transforming spatially distributed Boolean values into the averaged and reals and the inverse operation referred to as allocation procedure. The latter is precisely the most crucial point of the algorithm. Thus, we propose to exploit well studied CA-models of a diffusion [3] combining it with the integer approximation of reaction function.

The motivation for such an approach contains two arguments.The first is based on the wish to use the great experience of nonlinear phenomena study by PDE solving. The second reason is to obtain rather simple discrete models to replace PDEs, the solution of which is sometimes impractical. We do not know attempts to use such an approach, so we shall try to fill the gap.

To give a mathematical background of the proposed methods the formalism of *Parallel Substitution Algorithm* (PSA) [14] is used, which allows to combine real number and Boolean computation in a unique iterative process.

Apart from Introduction and Conclusion the paper contains four sections. In the second section main concepts and formalisms used in the paper are presented. The general scheme and two algorithms of transforming PDE into a discrete cellular automaton are presented in the third section. In the forth section the computer simulation results are given. In the short fifth section the properties of proposed methods are discussed and problems for future investigation are outlined.

## 2 Continuous and Discrete Forms of Spatial Dynamics Representation

### 2.1 Reaction-Diffusion Partial-Differential Equations

Let us consider reaction-diffusion process as a function of concentration of a certain substance of time and space. The traditional representation of the most simple one-dimensional reaction-diffusion process has the form of the following PDE;

$$\frac{du}{dt} = d(\frac{\partial^2 u}{\partial x^2}) + F(u) \tag{1}$$

where $u$ is a variable with the normalized domain from 0 to 1, $t, x$ are continuous time and space, $d$ is a diffusion coefficient, $F(u)$ a differentiable nonlinear function, satisfying certain conditions, which in [11] are given as follows.

$$\begin{array}{llll} F(0) = F(1) = 0; & & F(u) > 0 & \text{if } 0 < u < 1; \\ F'(0) = \alpha; & \alpha > 0; & F'(u) < \alpha; & \text{if } 0 < u < 1; \end{array} \tag{2}$$

The conditions (2) are met by a second order polinome (Fig. 1a) of the form

$$F(u) = \alpha u(1 - u); \tag{3}$$

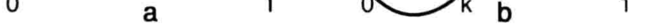


**Fig. 1.** The nonlinear functions used in typical reaction-diffusion equation

Equation (3) describes also the propagating front of the autocatalitic reaction (Field-Noyes model [15]). The equation (1) with $F(u)$ like (2) is studied in details [11,14]. It is known, that with the initial conditions

$$u(x,0) = \begin{cases} 1 & \text{if } x < 0, \\ 0 & \text{if } x \geq 0. \end{cases} \tag{4}$$

it generates an autowave of the type *propagating front*, which moves (at $t \to \infty$) with the velocity

$$V = 2\sqrt{d\alpha}, \tag{5}$$

In ecological research functions satisfying (2) are classified as *logistic* ones and considered to be basic, although some others are also studied, for example, those represented by third order polinomes (Fig. 1b), such as

$$F(u) = \alpha u(1-u)(u-k), \quad 0 < k < 1, \tag{6}$$

which meet the following conditions:

$$\begin{array}{l} F(0) = F(k) = F(1) = 0, \quad 0 < k < 1; \\ F(u) < 0 \ \text{ if } \ 0 < u < k; \\ F(u) > 0 \ \text{ if } \ k < u < 1; \\ F'(0) < 0, \ \ F'(k) > 0, \ \ F'(1) < 0, \end{array} \tag{7}$$

With $F(u)$ of the form (6) the propagating front velocity is

$$V = \sqrt{\alpha/2}(1-2k) \tag{8}$$

Moreover, when the initial condition have the form

$$u(x,0) = \begin{cases} u_0 & \text{if} \quad |x| \le l, \ k < u_0 \le 1, \\ 0 & \text{if} \quad |x| > l, \end{cases} \tag{9}$$

referred to as a “flash”, then the wave may attenuate, if $F(u)_{max}$ is not sufficiently large.

The above analytical characteristics of some simple and well studied reaction-diffusion phenomena are further used for comparing them with the similar ones obtained by simulation of CAs. Obviously, their correspondence would confirm the correctness of the proposed method.

### 2.2 Parallel Substitution Algorithm for Discrete Cellular Simulation

Parallel Substitution Algorithm (PSA) [14], is a convenient formalism for representing spatially distributed processes. It suits well to be used for our purpose, due to the fact that it allows to deal both with Boolean and real data. The following properties of PSA make it powerful for this purpose.

- PSA processes *cellular arrays*, which are sets of cells given as pairs $\mathbf{C}(A, M) = \{(a, m)\}$, where $a \in A$ is a cell *state*, and $m \in M$ is a cell *name*. $A$ - is an alphabet (in our case it is Boolean or real). $M$ is a *naming set* (in general case a countable one). On the set $M$ naming functions $\phi_l : M \to M$ are defined. The naming set is the set of discrete Cartesian coordinates, given as $m = \langle i, j, k \rangle$. In our case only shift naming functions are used. A set of namimg functions form determines the names of any cell *neighborhood.*
- Operations over a cellular array are specified by a set $\Phi = \{\Theta_i\}, i = 1, \ldots, n$, of parallel substitutions of the form

$$\Theta_i : C_i(m) * S_i(m) \to S'_i(m). \tag{10}$$

where

$$\begin{aligned} C_i(m) &= \{(y_{ik}, \phi_{ik}(m)) : k = 0, \dots, q_y\}, \\ S_i(m) &= \{(x_{ij}, \phi_{ij}(m)) : j = 0, \dots, q_x\}, \\ S'_i(m) &= \{(f_{ij}(X,Y), \phi_{ij}(m) : j = 0, \dots, q_x\}, \end{aligned} \tag{11}$$

In (10,11) $C_i(m), S_i(m)$ and $S'_i(m)$ are *local configurations*, $*$ meaning their union for any $m \in M$. Further only stationary parallel substitutions are used, in which the neighborhoods of $S_i(m)$ and $S'_i(m)$ are formed by identical sets of naming functions, which contain an identical naming function $\phi(m) = m$ referred to as a *central* cell of the substitution. A parallel substitution should meet the following conditions:

1) no pair of naming functions values in (11) are equal,

2) $x_{ij} \in X, y_{ik} \in Y$ are state variables or constants and $f_{il}(X,Y)$ are cellular functions with the domain from $A$.

- A substitution *is applicable* to $\mathbf{C}(A,M)$, if there is at least one cell named $m \in M$ such that $C_i(m) \cup S_i(m) \subseteq \mathbf{C}(A,M)$. Application of a substitution at a cell $(a,m) \in \mathbf{C}(A,M)$ yields changing cell states in $S_i(m)$ called *the base* by the corresponding ones from $S'_i(m)$, the set of cells $C_i(m)$ (called *a context*) remaining unchanged.

- There are three modes of parallel substitutions application.

1) *Synchronous mode*, when at each step all substitutions are applied at all cells at once. At this case in order to provide determinism of the computation, one should be careful not to allow the substitutions be contradictory when $|S'_i(m)| > 1$ [14].

2) *Asynchronous mode*, when any substitution is applied at any cell, one application being allowed at a time. There is no danger of contradictoriness in this case, but a generator of random numbers should be used to determine a next cell to which the substitutions are to be applied each time .

3) *2-step synchronous mode*, when cellular array under processing is to be partitioned into two parts, and at each time-step the substitutions act at one of them only.

- A Parallel Substitution Algorithm (PSA) is a set of substitutions together with indication of the mode of application. Implementation of a PSA over a cellular array $\mathbf{C}$ is an iterative procedure, where at each step the substitution set is executed at a set of cells, according to the given mode. The algorithm stops when no substitution is applicable to the array.

- A PSA may process not only one but a number of interacting arrays $\mathbf{C} = \{\mathbf{C}_1, \dots, \mathbf{C}_n\}$ as well. In the latter case each substitution $\Theta_i$ is allowed to be applied to only one array. It means that its base $S_i(m)$ is located in only one $C_l \in \mathbf{C}$, i.e. $m \in M_l$. As for the context $C_i(m)$, it may be located at any array, moreover, it may be composed of a number of local configurations, located in different arrays, i.e.

$$C_i(m) = C_1(m_1) * \dots * C_k(m_k), \quad k \le n; m_j \in M_j. \tag{12}$$

PSA is further used to represent reaction-diffusion processes by discrete fine-grained parallel algorithms.