Efficient parallel implementation of Potts model

Carmen B. Navarrete[1,2], Carlos Castañeda[1], Manuel Alfonseca[1], Alfonso Ortega[1] and Eloy Anguiano[1,2]

[1] Universidad Autónoma de Madrid, Dpto Ingeniería Informática, Madrid, SPAIN,
[2] Centro de Referencia Linux (CRL, UAM-IBM), Universidad Autónoma de Madrid, Madrid, SPAIN

Abstract. In this work we present a different approach to simulate the Ferromagnetic Potts model [1] described by Renfrey B. Potts in 1952 and it is a generalization of the Ising [2] model. It is well-known that the Potts model has a disadvantage when it is implemented using a Monte Carlo simulation with a standard Metropolis algorithm. Many parallel algorithms have recently been developed to reduce this "drawback" [3,4]. The problem of these algorithms is that they are much more difficult to parallelize efficiently. Our work describes a parallel Monte Carlo solution, using a C and an MPI library, which is more efficient than the traditional methods.

The Potts model consists of spins that are placed on a lattice; the lattice is usually taken to be a two-dimensional rectangular Euclidean lattice, but is often generalized to other dimensions or other lattices. The Hamiltonian function is defined as follows:

$$H = J \sum_{(i,j),(i',j')} 1 - \delta_{\sigma_{(i,j)}\sigma_{(i',j')}} \tag{1}$$

where J is a positive constant, δ is the Kronecker delta, $1 < \delta_{(i,j)} < N_d$ denotes the orientation of the spin at (i, j), N_d is the number of possible domains in the system at the begining of the simulation and $(i, j), (i', j')$ denotes the first neighbourhood area.

The probability function is defined as:

$$P = e^{-\frac{\Delta H}{K_B T}} \tag{2}$$

where K_B is the Boltzman constant, T is a certain temperature value. The new state of the spin is accepted when $\Delta H > 0$ with probability P or 1 if $\Delta H \leqslant 0$

Evolution of the model proceeds using the Metropolis Montecarlo [5] simulation as follows:

1. Select randomly a lattice spin
2. Choose randomly a neighboring spin of the current site
3. If the neighboring spin is equal to the spin of the current site, go to step 1
4. Change the current site spin for the spin of its neighboring site, according to the probability P as in expresion 2

5. Return to step 1

The lattice is represented as a torus and it is continually updated: for each lattice point, a different spin state is proposed, and the new overall energy calculated. It depends on the neighbor's interactions and the overall temperature. If the new energy is smaller than the old one, the new state is accepted. If not, there is still a certain chance that will be accepted, leading to random spin flips representing the overall temperature.

The use of clustering computing to solve computational problems has been the focus of high-performance computing community for more than two decades.

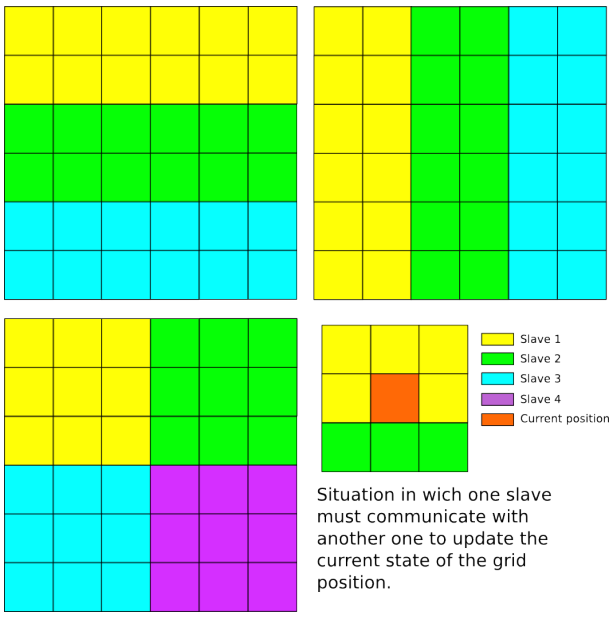

Fig. 1. The possible domain descomposition and explanation of the situation in which one proccessor must send a signal to another to be able to updating the current grid position.

Our goal is to parallelize the Potts model to be executed in a parallel environment, created by an heterogeneous cluster, using the MPI library [6]. The aim is to increase the speed-up and the eficiency compared to the tradiciontal Potts simulation. There are several publications about parallel implementations of this model such us the one described in [3]. The main problem designing a parallel or distributed algorithm resides in the communication and synchronization of processes for its concurrent execution in different processors. This is a very hard non-deterministic optimization problem that not allways have the best solution. In many cases, only an approximation can be calculated. These pro-

cesses are the result of applying a division method to the domain of the problem. Four simulations have been performed: one simulation for the non-parallel applications and three more to study the best domain descomposition, in order to optimize the communications between pairs of processes, to increase the speed up of the execution of the problem. In these simulations we have decided to use a domain descomposition based on horizontal or vertical stripes or based on a square descomposition, also taking into account the processor speed of each node of the cluster. The goal of our algorithm is the way we solve the possible situation in which one node needs some information in order to update the current grid position that is in another node of the cluster (see figure 1). The results obtained with this simulation are equivalent to the ones obtained with the standard Metropolis Montecarlo algorithm. The ongoing work will consist on applying these concepts to simulate the cell growth using a Cellular Potts Model (CPM [7]) and using a dynamic domain descomposition based on the cell dimensions.

References

1. Renfrey B. Potts. Some generalized order-disorder transformations. volume 48, pages 106–109. Proceedings of the Cambridge Philosophical Society, 1952.
2. E.Ising. Beitrag zur theorie des ferromagnetismus. *J Physics*, 31, 1925.
3. E. Gusatto, J. C. M. Mombach, F. Piccini Cercato, and G. H. Cavalheiro. An efficient parallel algorithm to evolve simulations of the cellular potts model. *Parallel Processing Letters*, 15(1-2):199–208, 2005.
4. C. , G. , I. , and A. . A parallel implementation of the cellular potts model for simulation of cell-based morphogenesis. *Comput Phys Commun*, 176(11-12):670–681, 2007.
5. M. Rosenbluth A. Teller E. Teller N. Metropolis, A. Rosenbluth. Equation of state calculations by fast computing machines. *J. Chem. Phys*, 21(13):1087–1092, 1953.
6. Message Passing Interface Forum. MPI: A message-passing interface standard. Technical Report UT-CS-94-230, 1994.
7. J.A. Glazier F. Graner. Simulation of biological cell sorting using a 2-dimensional extended potts model. *Phys. Rev Lett*, 69(13):2013–2016, 1992.