ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

# ΕΜ 361: Παράλληλοι Υπολογισμοί

**Ενότητα:** Projects

Διδάσκων: Χαρμανδάρης Ευάγγελος
ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

# Άδειες Χρήσης

# Χρηματοδότηση


- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

**Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης**

**Χαρμανδάρης Βαγγέλης, Τμήμα Εφαρμοσμένων Μαθηματικών**

**Πανεπιστήμιο Κρήτης, Χειμερινό Εξάμηνο 2010/11**


## Projects

- **Πρόβλημα Ν-σωματιδίων.**
- **Image Processing (Fast Fourier Transform).**
- **Mandelbrot Set.**
- **Κυτταρικά Αυτόματα - "Game of Life".**
- **Multi-colored Algorithms.**

# Πρόβλημα N-Σωματιδίων

-- Finding positions and velocities of bodies in space subject to forces from other bodies, using Newtonian laws of physics.

-- Simulate movement of bodies in time.

**Gravitational *N*-Body Problem Equations**

Gravitational force between two bodies of masses $m_a$ and $m_b$ is:

$$F = \frac{Gm_a m_b}{r^2}$$

$G$ is the gravitational constant and $r$ the distance between the bodies. Subject to forces, body accelerates according to Newton's 2nd law:

$$F = ma$$

$m$ is mass of the body, $F$ is force it experiences, and $a$ the resultant acceleration.

# Πρόβλημα Ν-Σωματιδίων

-- Assume a system with N bodies. Procedure:

a) Initialization of the system (initial coordinates, velocities)

b) Calculate forces.

c) Numerical solution of PDEs.

d) Statistics (store trajectories, calculate properties, ..etc.)

-- Various methods for the numerical solution of PDEs. Example:

$$F = \frac{m(v^{t+1} - v^{t})}{\Delta t} \qquad v^{t+1} = v^{t} + \frac{F \Delta t}{m}$$

$$x^{t+1} - x^{t} = v \Delta t$$

# Πρόβλημα Ν-Σωματιδίων

**-- Parallel implementation**

➢ Different Algorithms:

A) **Direct algorithm:** divide number of particles with number of processors, i.e. N/P bodies per processor.

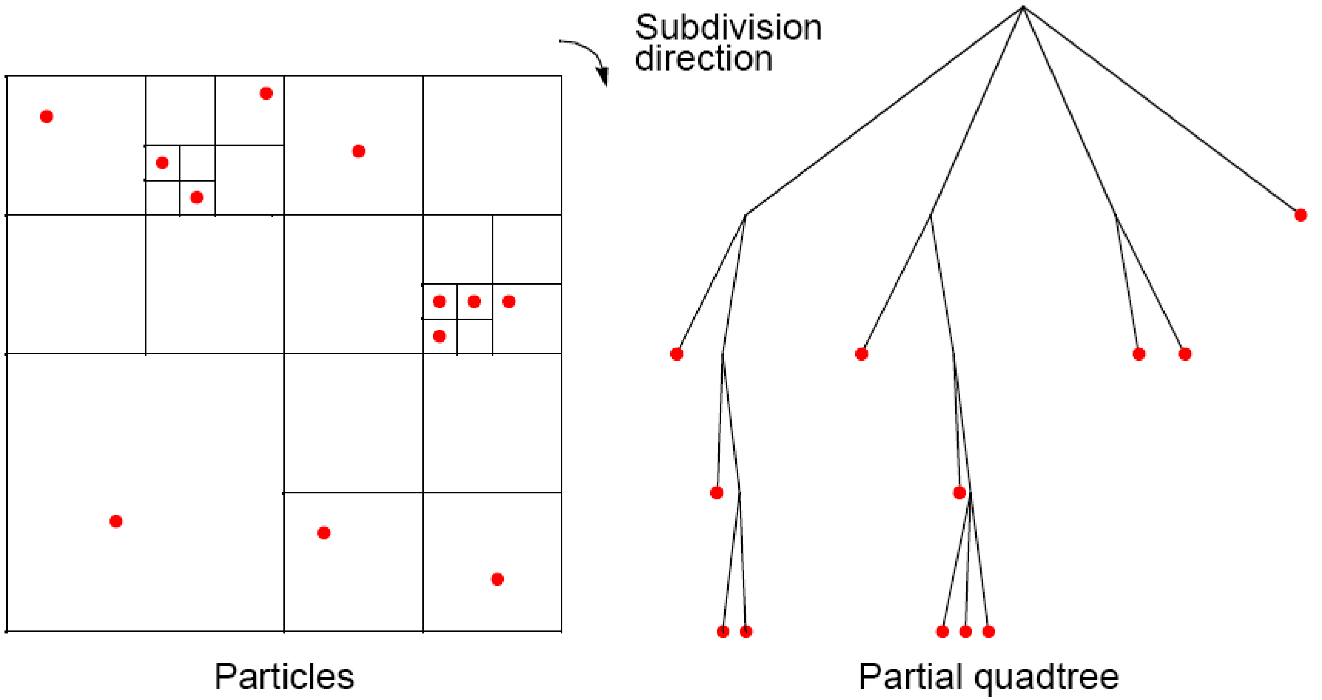
The sequential algorithm is an O(N2) algorithm (for one iteration) as each of the N bodies is influenced by each of the other N - 1 bodies.

Not feasible to use this direct algorithm for most interesting *N*-body problems where *N* is very large.

# Πρόβλημα Ν-Σωματιδίων

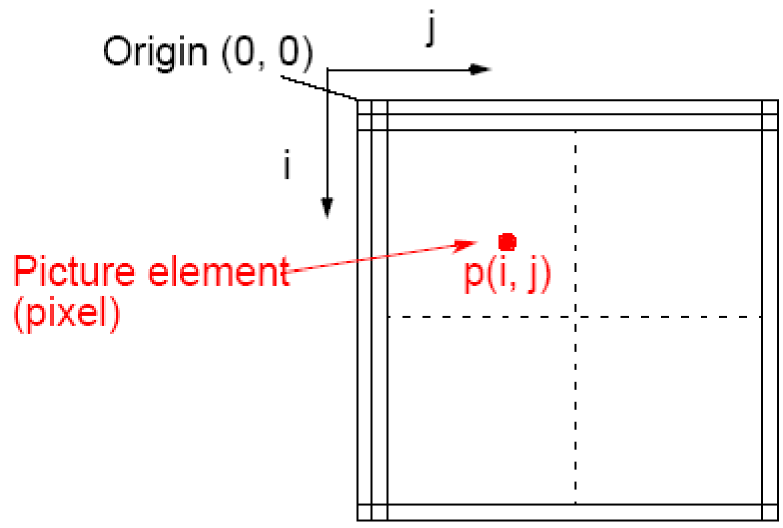
**-- Parallel implementation**

B) **Barnes-Hut Algorithm:** Recursive division of 2-dimensional space.



# Image Processing

Stored images: 2D array of pixels (picture elements)




**Perform various operations on images:**

| | | |
|---|---|---|
| $x_0$ | $x_1$ | $x_2$ |
| $x_3$ | $x_4$ | $x_5$ |
| $x_6$ | $x_7$ | $x_8$ |

- Smoothing (mean value):

$$x_4' = \frac{x_0 + x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8}{9}$$

# Image Processing

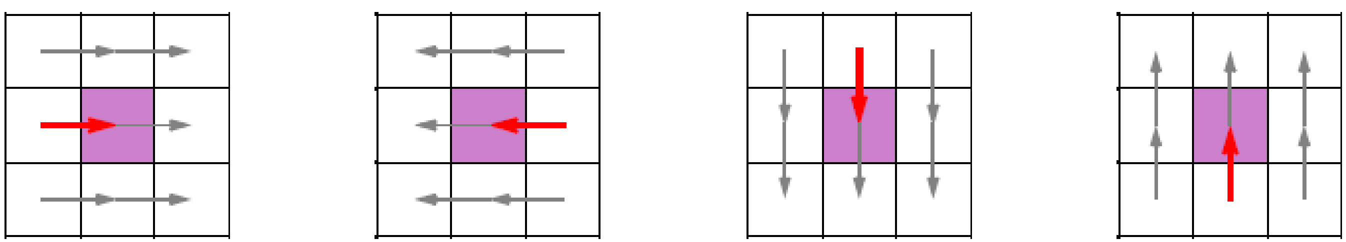
**Serial – Sequential code:**

Nine steps to compute the average for each pixel, or 9*n* for *n* pixels.

A sequential time complexity of $O(n)$.

**Parallel Code:**

Number of steps can be reduced by separating the computation into four data transfer steps in lock-step data-parallel fashion.



Step 1 Each pixel adds pixel from left | Step 2 Each pixel adds pixel from right | Step 3 Each pixel adds pixel from above | Step 4 Each pixel adds pixel from below

# Image Processing

**Perform various operations on images:**

## Using a 3 × 3 Weighted Mask

$$\underset{\text{Mask}}{\begin{array}{|c|c|c|}\hline w_0 & w_1 & w_2 \\ \hline w_3 & w_4 & w_5 \\ \hline w_6 & w_7 & w_8 \\ \hline\end{array}} \otimes \underset{\text{Pixels}}{\begin{array}{|c|c|c|}\hline x_0 & x_1 & x_2 \\ \hline x_3 & x_4 & x_5 \\ \hline x_6 & x_7 & x_8 \\ \hline\end{array}} = \underset{\text{Result}}{\begin{array}{|c|c|c|}\hline \phantom{x} & & \\ \hline & x_4' & \\ \hline & & \\ \hline\end{array}}$$

The summation of products, $w_i x_i$, from two functions $w$ and $x$ is the (discrete) *cross-correlation* of $f$ with $w$ (written as $f \otimes w$).

# Image Processing

**Perform various operations on images:**

- Edge detection: Highlighting edges of object where an edge is a significant change in gray level intensity.

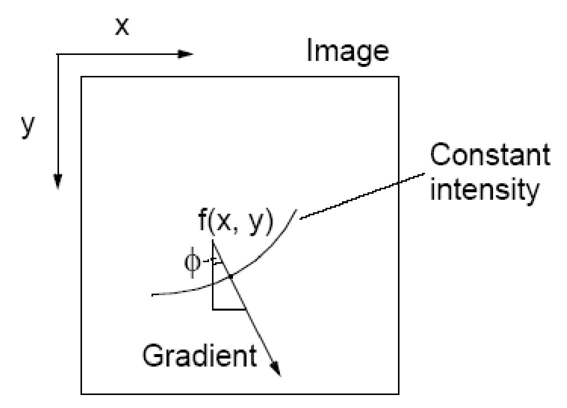
A two-dimensional discretized gray level function, *f*(*x*,*y*).

## Sobel Operator

Derivatives are approximated to

$$\frac{\partial f}{\partial y} \approx (x_6 + 2x_7 + x_8) - (x_0 + 2x_1 + x_2)$$

$$\frac{\partial f}{\partial x} \approx (x_2 + 2x_5 + x_8) - (x_0 + 2x_3 + x_6)$$




Operators implementing first derivatives will tend to enhance noise.

# Image Processing

**Perform various operations on images:**

Edge Detection with Sobel Operator

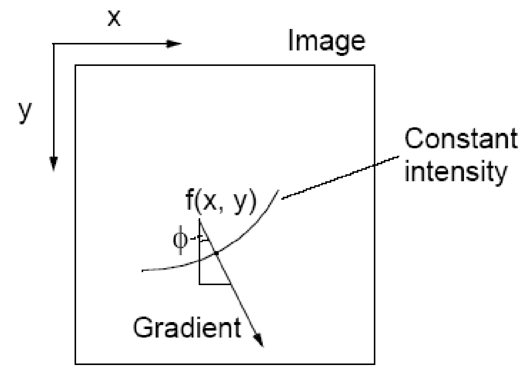
(a) Original image (Annabel)

(b) Effect of Sobel operator

# Image Processing

**Perform various operations on images**

➢ 2nd Order Derivatives: Laplace Operator




## Laplace Operator

The Laplace second-order derivative is defined as

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

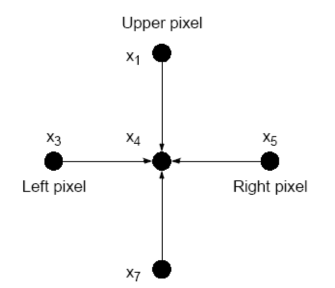
approximated to

$$\nabla^2 f = 4x_4 - (x_1 + x_3 + x_5 + x_7)$$

which can be obtained with the single mask:

| | | |
|---|---|---|
| 0 | −1 | 0 |
| −1 | 4 | −1 |
| 0 | −1 | 0 |

**Pixels used in Laplace operator**



# Effect of Laplace operator

# Mandelbrot Set

Set of points in a complex plane that are quasi-stable (will increase and decrease, but not exceed some limit) when computed by iterating the function

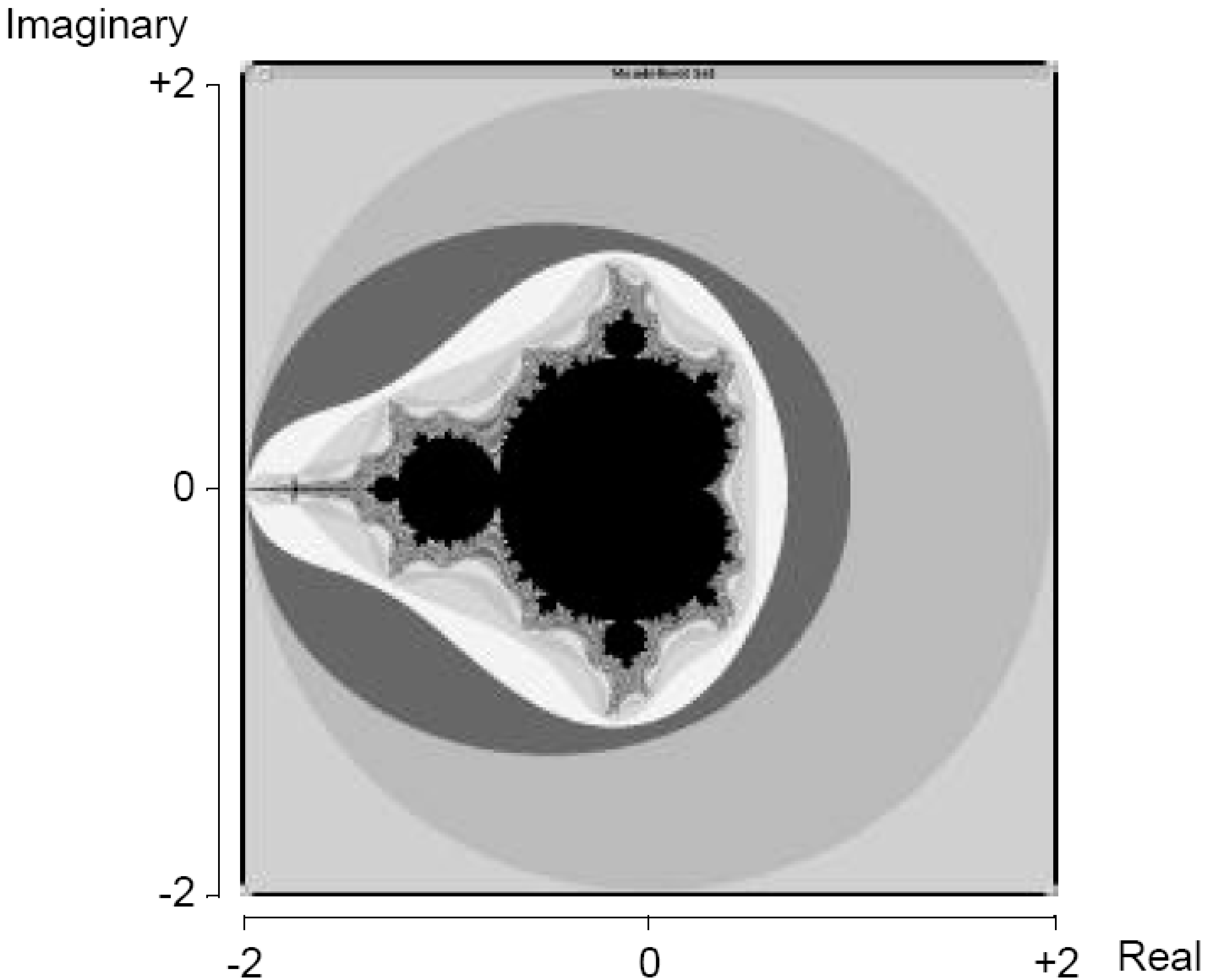
$$z_{k+1} = z_k^2 + c$$

where $z_{k+1}$ is the $(k + 1)th$ iteration of the complex number $z = a + bi$ and $c$ is a complex number giving position of point in the complex plane. The initial value for $z$ is zero.

Iterations continued until magnitude of $z$ is greater than 2 or number of iterations reaches arbitrary limit. Magnitude of $z$ is the length of the vector given by

$$z_{\text{length}} = \sqrt{a^2 + b^2}$$

# Mandelbrot Set



# Mandelbrot Set

**-- Parallel implementation**

➢ Different Algorithms:

A) **Static Task Assignment:**

Simply divide the region in to fixed number of parts, each computed by a separate processor.

Not very successful because different regions require different numbers of iterations and time.

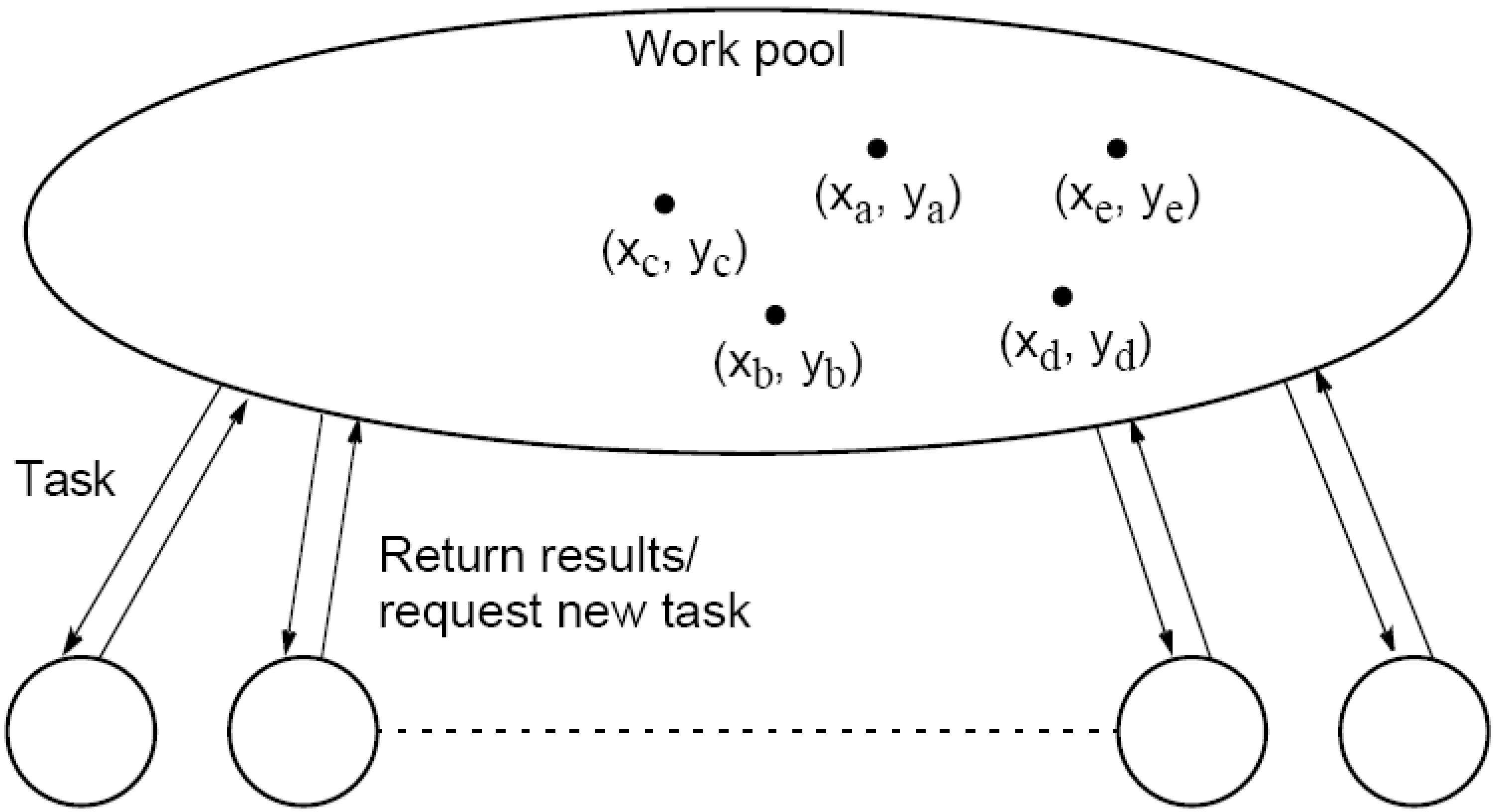
B) **Dynamic Task Assignment:**

Have processor request regions after computing previous regions

# Mandelbrot Set

## Dynamic Task Assignment

**Work Pool/Processor Farms**



# Cellular Automata ("Game of Life")

The problem space is divided into cells.

Each cell can be in one of a finite number of states.

Cells affected by their neighbors according to certain rules, and all cells are affected simultaneously in a "generation."

Rules re-applied in subsequent generations so that cells evolve, or change state, from generation to generation.

Most famous cellular automata is the "Game of Life" devised by John Horton Conway, a Cambridge mathematician.

# The Game of Life
**[by Gardner 1967]**

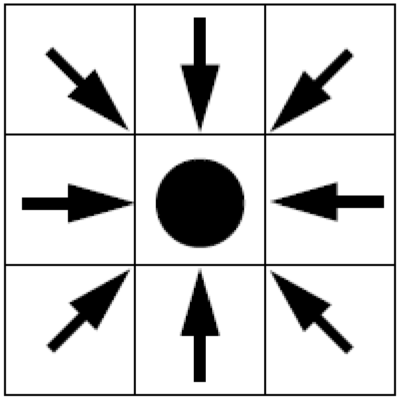
Board game - theoretically infinite two-dimensional array of cells. Each cell can hold one "organism" and has eight neighboring cells, including those diagonally adjacent. Initially, some cells occupied. The following rules apply:

1. Every organism with two or three neighboring organisms survives for the next generation.

2. Every organism with four or more neighbors dies from overpopulation.

3. Every organism with one neighbor or none dies from isolation.

4. Each empty cell adjacent to exactly three occupied neighbors will give birth to an organism.



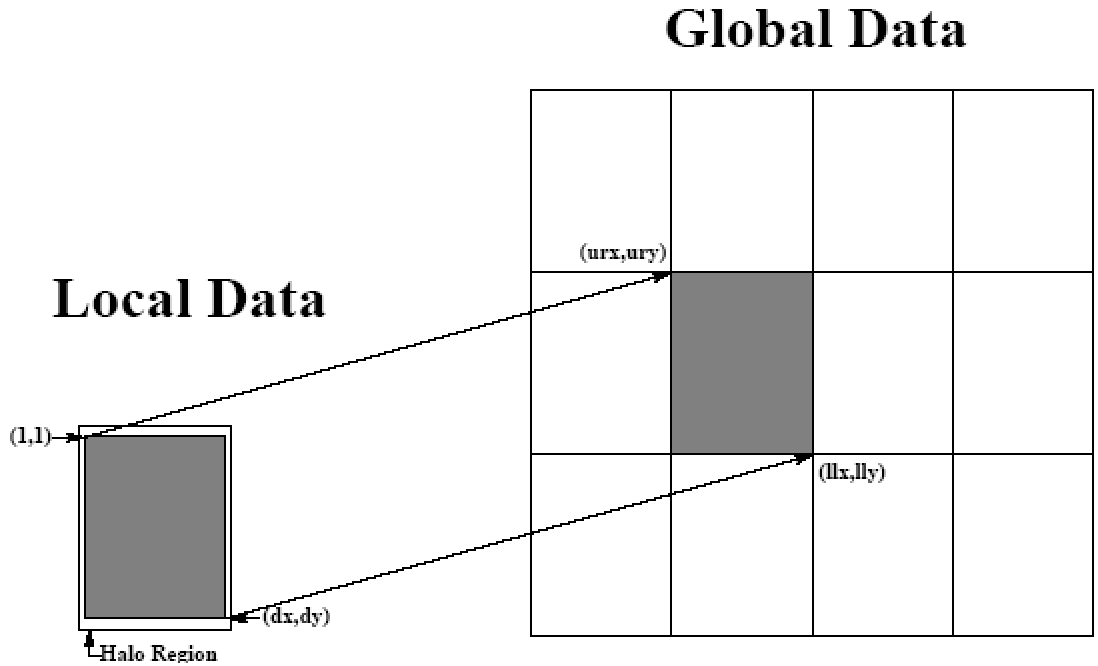
These rules were derived by Conway "after a long period of experimentation."

# The Game of Life

- Initial configuration:

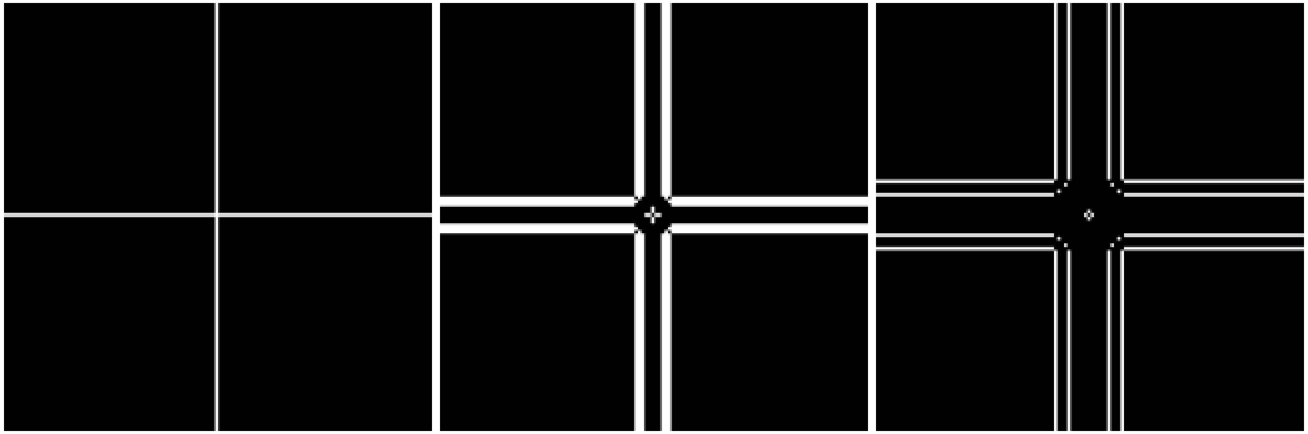
$$Cell_{ij} = \begin{cases} Alive & \text{if (i,j)=(XSIZE/2,YSIZE/2)} \\ Dead & \text{otherwise} \end{cases}$$

- Mapping – distribution of board:



# The Game of Life

- Evolution of "Life": exchange info with neighboring procs.



## Serious Applications for Cellular Automata

**Examples**

- fluid/gas dynamics
- the movement of fluids and gases around objects
- diffusion of gases
- biological growth
- airflow across an airplane wing
- erosion/movement of sand at a beach or riverbank.

# Multi-Coloured, Multi-Grid Algorithms

**Πρόβλημα – Μοντέλο (συνοριακών τιμών) : Laplace + επιπλέον όρος:**

$$\frac{\partial^2 U(x,y)}{\partial x^2} + \frac{\partial^2 U(x,y)}{\partial y^2} + a\frac{\partial^2 U(x,y)}{\partial x \partial y} = f(x,y)$$

**Διαμέριση:**

$$h = \frac{1}{N+1}, \qquad x_i = i*h, \;\; y_i = j*h, \qquad i,j = 0,1,\ldots,N+1$$

**Προσέγγιση παραγώγων:**

$$\frac{\partial^2 U(x,y)}{\partial x^2} \approx \frac{1}{h^2}\left(U_{i+1,j} + U_{i-1,j} - 2U_{i,j}\right)$$

$$\frac{\partial^2 U(x,y)}{\partial y^2} \approx \frac{1}{h^2}\left(U_{i,j+1} + U_{i,j-1} - 2U_{i,j}\right)$$

$$\frac{\partial^2 U(x,y)}{\partial x \partial y} \approx \frac{1}{2h}\left(\left(\frac{\partial U}{\partial y}\right)_{i+1,j} - \left(\frac{\partial U}{\partial y}\right)_{i-1,j}\right)$$

$$\frac{\partial^2 U(x,y)}{\partial x \partial y} \approx \frac{1}{4h^2}\left(U_{i+1,j+1} - U_{i-1,j+1} - U_{i+1,j-1} + U_{i-1,j-1}\right)$$

# Multi-Coloured, Multi-Grid Algorithms

**Πρόβλημα Συνοριακών Τιμών - Laplace + επιπλέον όρος:**

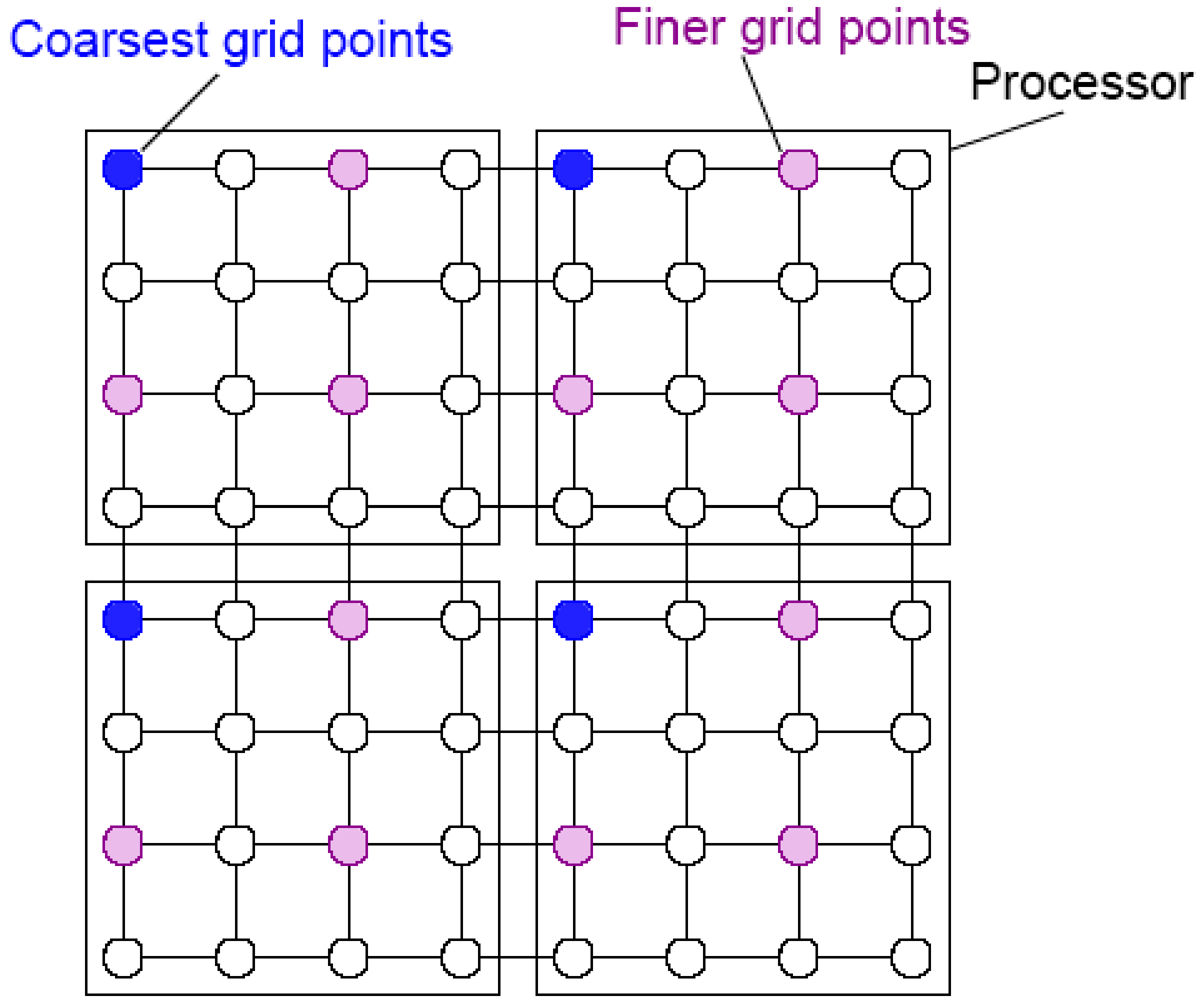
$$\left(U_{i+1,j} + U_{i-1,j} + U_{i,j+1} + U_{i,j-1} - 4U_{i,j}\right) + \frac{a}{4}\left(U_{i+1,j+1} - U_{i-1,j+1} - U_{i+1,j-1} + U_{i-1,j-1}\right) - h^2 f_{i,j} = 0$$

## Παράλληλη Εφαρμογή της SOR: Multi-colored Αλγόριθμος

Four-color ordering: το κεντρικό σημείο έχει διαφορετικό «χρώμα» από όλα τα γειτονικά σημεία με τα οποία συνδέεται , δηλαδή από τα οποία παίρνει πληροφορία.

# Multi-Coloured, Multi-Grid Algorithms

**Multi-Grid Αλγόριθμοι:**



# Genetic Algorithms

**Searching and Optimization Algorithms:**

➢ A search technique used in computing to find exact or approximate solutions to optimization and search problems.

➢ Genetic algorithms are a particular class of evolutionary algorithms (EA) that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover.

**Outline of the Basic Genetic Algorithm**

**[Start]** Generate random population of *n* chromosomes (suitable solutions for the problem)
**[Fitness]** Evaluate the fitness *f(x)* of each chromosome *x* in the population
**[New population]** Create a new population by repeating following steps until the new population is complete

- **[Selection]** Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected)
- **[Crossover]** With a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.
- **[Mutation]** With a mutation probability mutate new offspring at each locus (position in chromosome).
- **[Accepting]** Place new offspring in a new population

**[Replace]** Use new generated population for a further run of algorithm
**[Test]** If the end condition is satisfied, **stop**, and return the best solution in current population
**[Loop]** Go to step **2**

# Βιβλιογραφία


- *Designing and Building Parallel Programs*, Ian Foster, Addison-Wesley 1994.
- *Parallel Computing: Theory and Practice*, M. J. Quinn, McGraw-Hill, 1994.
- *Parallel Programming: Techniques amd applications Using Networked Workstations and Parallel Computers*, B. Wilkinson, M. Allen, Prentice-Hall, 1999.
- http://www.epcc.ed.ac.uk/library/documentation/training/: On-line courses include MPI, HPF, Mesh Generation, Introduction to Computational Science, HPC in Business.

# Τέλος Ενότητας

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης