

Assignment 1 - OpenMP

- **Use OpenMP to parallelize the deqn code**
 - The overall objective is to achieve good speedup by inserting OpenMP directives in the deqn code
 - You also need to
 - benchmark the runtime of each relevant loop and the runtime of the whole parallel program against the number of threads
 - Analyze the overhead of OpenMP

change the num of threads

Assignment 1 - OpenMP

- **Write a report**
 - Explain in detail what you did with the sequential code
 - benchmark the runtime of each relevant loop and the runtime of the whole parallel program against the number of threads; present the runtimes in graph or table; analyze the results
 - Discuss the iteration scheduling in your program *dynamic / static*
 - Analyze the overhead of OpenMP
 - Presentation skills, spelling, punctuation and grammar
 - Up to four A4 pages

Submission

- Put all the codes and the report (pdf file) in a package and submit the package through Tabula
- Deadline: 12 noon, Feb 5th, Monday, 2019

deqn

- Model the transfer of heat through a material
- Expressed as a partial differential equation

u: temperature.

$$\frac{\partial u}{\partial t} - \alpha \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = 0$$

x, y, space.

*Solve u at any
time t,
and (x, y)*

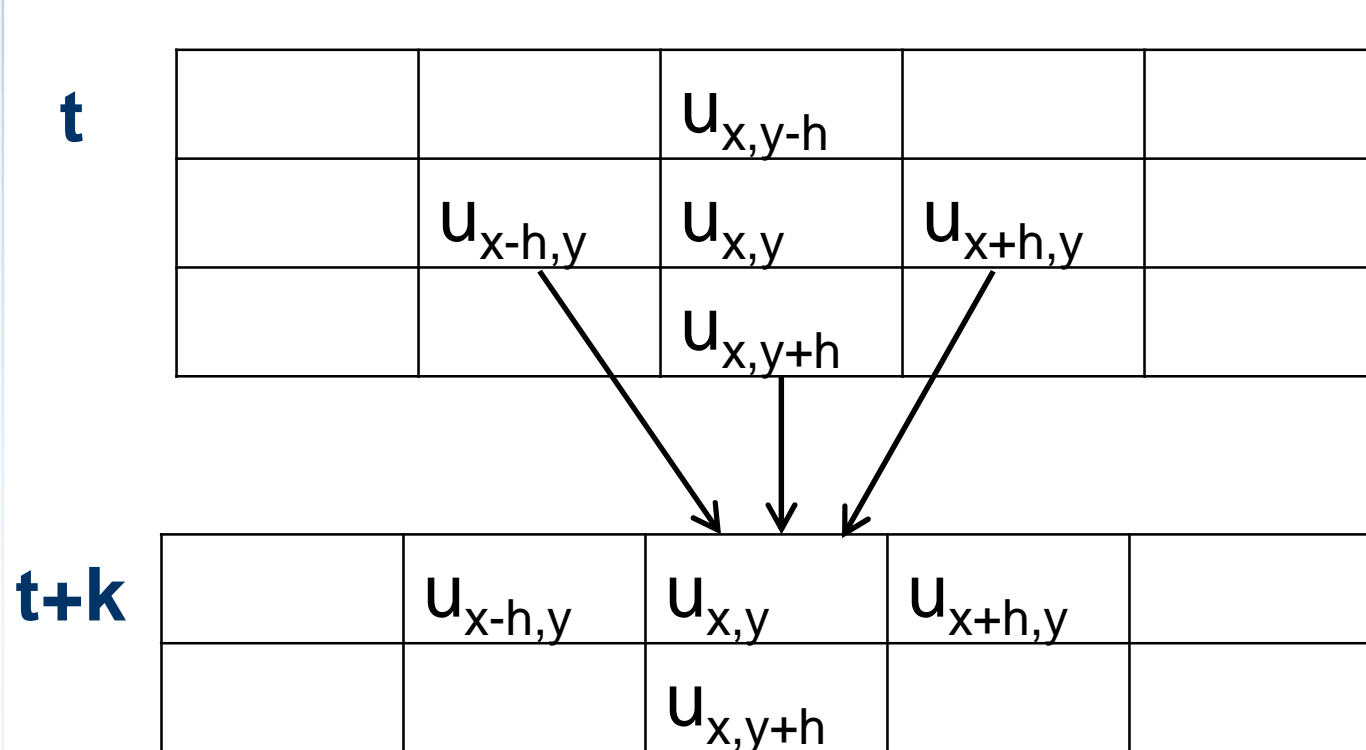
- Discretise the equation over time and space using finite difference method

$$u_{x,y}^{t+k} = r u_{x+h_x,y}^t + r u_{x-h_x,y}^t + r' u_{x,y+h_y}^t + r' u_{x,y-h_y}^t + (1 - 2r - 2r') u_{x,y}^t$$

- Given the initial temperature of the material, we can calculate the temperature of the material at any time point and space location

Data Dependency

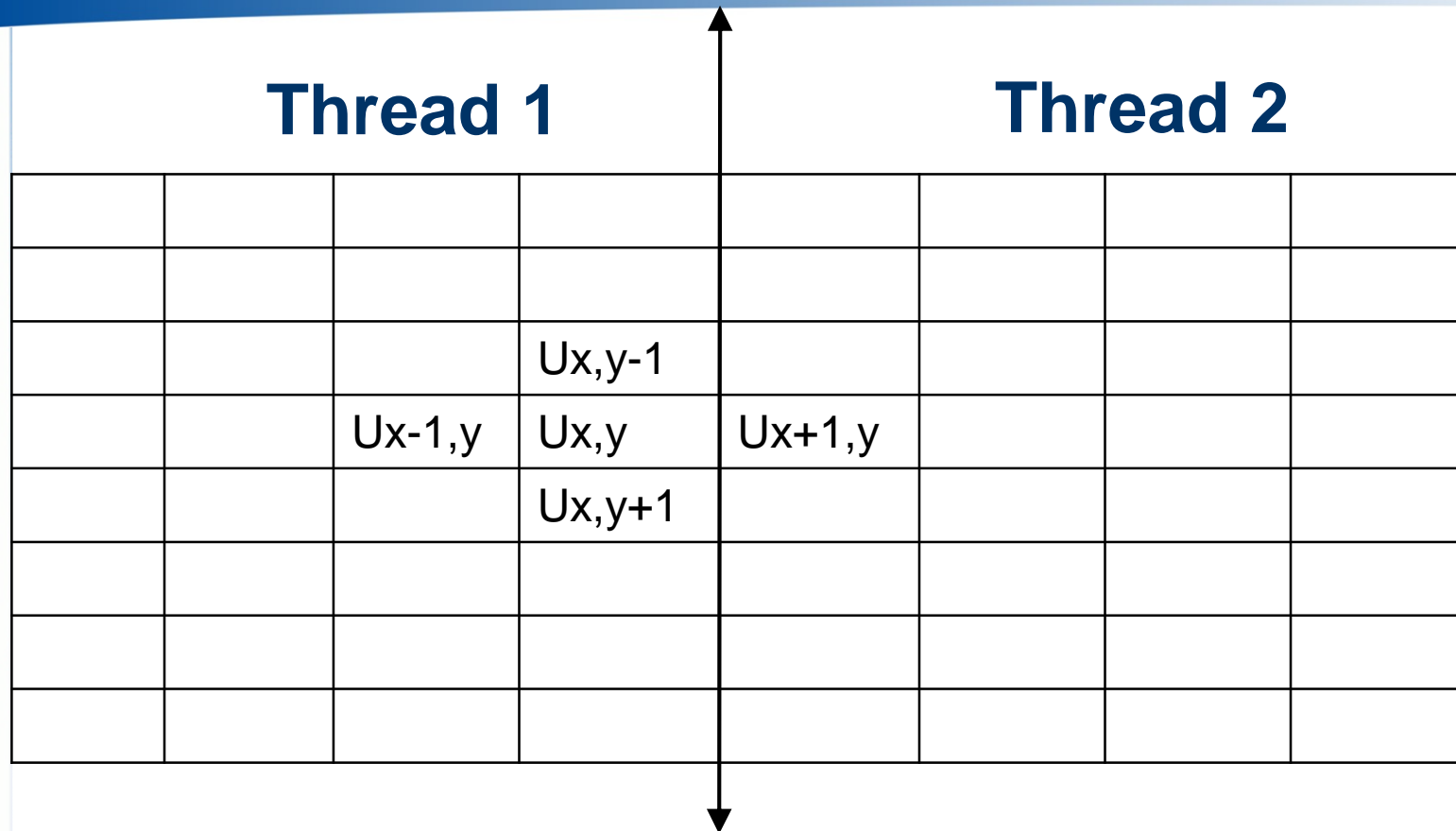
$$u_{x,y}^{t+k} = ru_{x+h_x,y}^t + ru_{x-h_x,y}^t + r'u_{x,y+h_y}^t + r'u_{x,y-h_y}^t + (1 - 2r - 2r')u_{x,y}^t$$



at $t+k$,
 $u_{x,y}$ needed
 time t .
 and these 5
 values

Question: How to parallelize the computation?

Parallelism



- Each thread calculates the temperature at any time and at its local space in parallel
- Thread communication