Heuristic-based Task-to-Thread Mapping in Multi-Core Processors

Presenter:

Mohammad Samadi Gharajeh

Co-authors:

Sara Royuela, Luis Miguel Pinho, Tiago Carvalho, Eduardo Quiñones

School of Engineering, Polytechnic Institute of Porto, Portugal

Universitat Politècnica de Catalunya, Barcelona, Spain

Barcelona Supercomputing Center, Barcelona, Spain

September 2022

Introduction

- Enhance the performance of real-time applications using OpenMP
- Emerge a challenge with the predictability of OpenMP applications
- Study the applicability of heuristic algorithms in mapping
- Achieve a work-conserving and load-balancing mapping
- Minimize scheduling time (end-to-end response time) and idle time

Motivations

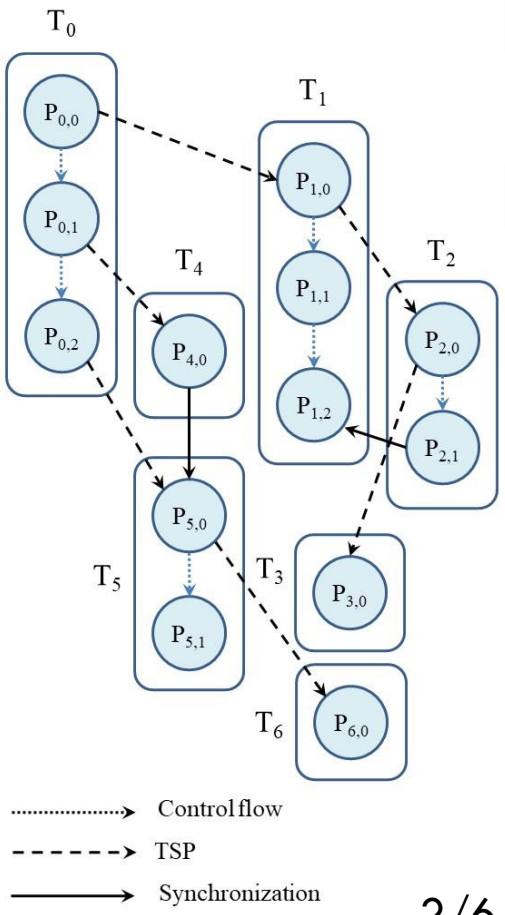
The main problems with the existing mappings:

- They do not schedule tasks on threads based on global information of the system.
- They do not use temporal conditions of the system (e.g., execution time and deadline) in each phase of the mapping process.
- Most of them do not apply various heuristic techniques.

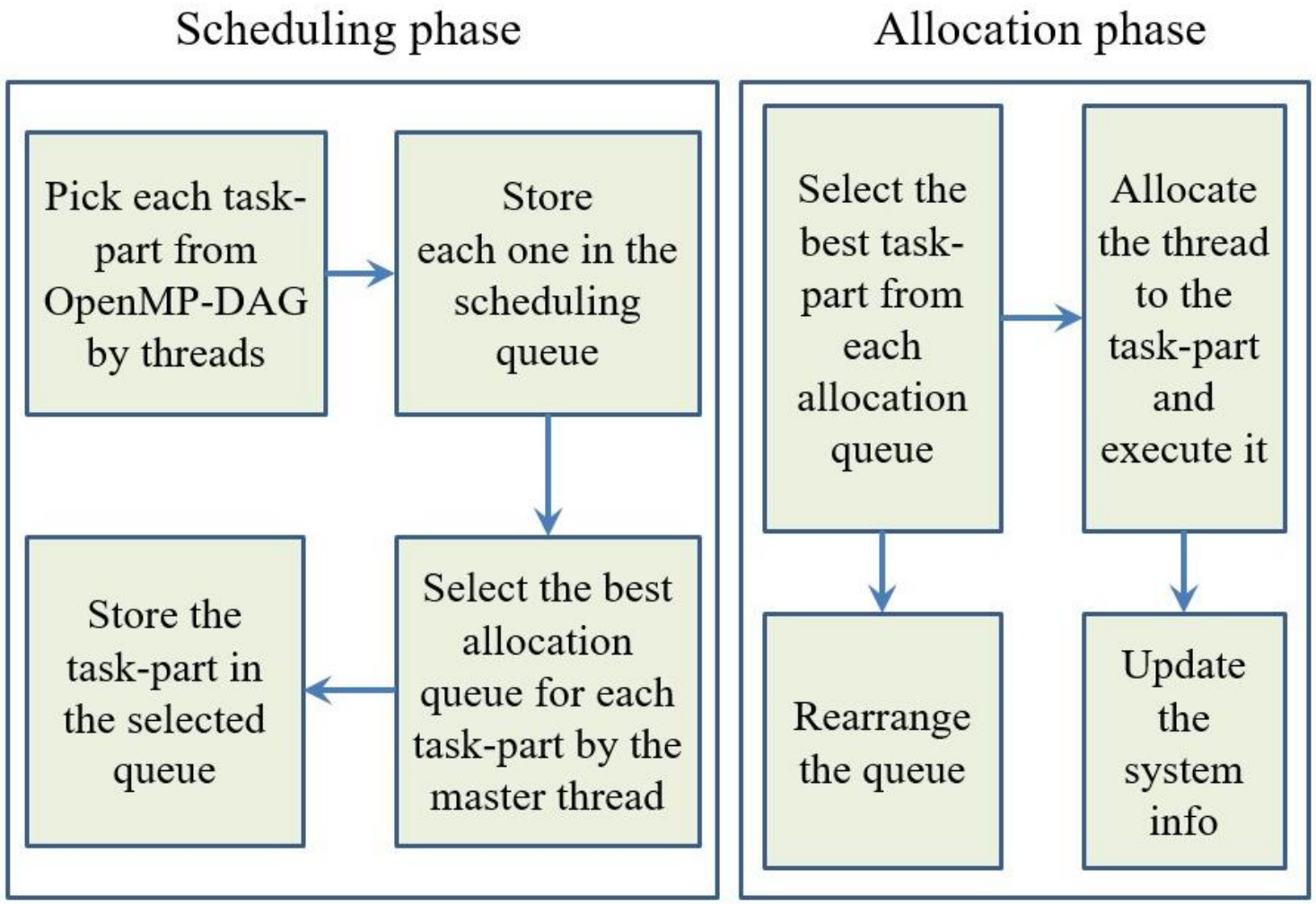
Source code

```
#pragma omp parallel num_threads(4)
#pragma omp single                          // T0
{
   code00;
   #pragma omp task                         // T1
   {
      code10;
      #pragma omp task                      // T2
      {
         code20;
         #pragma omp task                   // T3
            code30;
         code21;
      }
      code11;
      #pragma omp taskwait
      code12;
   }
   code01;
   #pragma omp task depend(out:x)           // T4
      code40;
   code02;
   #pragma omp task depend(in:x)            // T5
   {
      code50;
      #pragma omp task                      // T6
         code60;
      code51;
   }
}
```

Direct Acyclic Graph (DAG)



Proposed Mapping Algorithm



Proposed Mapping Algorithm

The suggested scheduling heuristics:

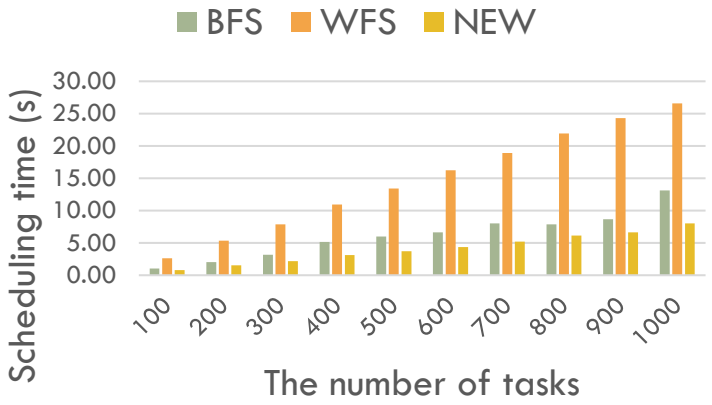
- *First-fit scheduling*: the thread (i.e., allocation queue) containing the minimum number of task-parts
- *Best-fit scheduling*: the queue with the minimum execution time
- *Optimum-fit scheduling*: the queue including the maximum response time (of the task-parts)
- *Multi-criteria scheduling*: the queue having the characteristics of the previous heuristics

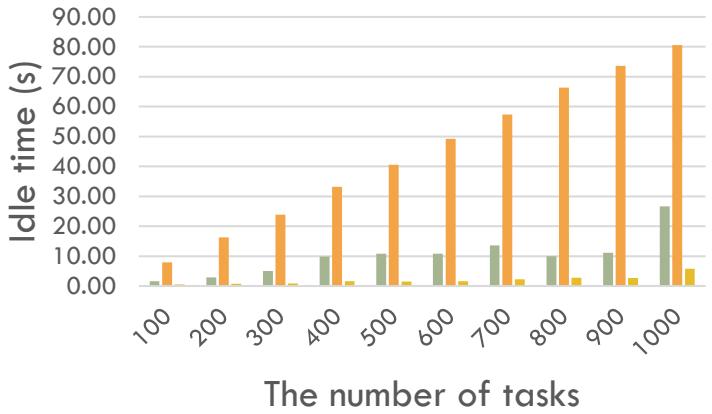
The suggested allocation heuristics:

- *Best-fit allocation*: the task-part having the minimum execution time
- *Optimum-fit allocation*: the task-part with the maximum response time
- *Multi-criteria allocation*: the task-part including the shortest execution time and the longest response time

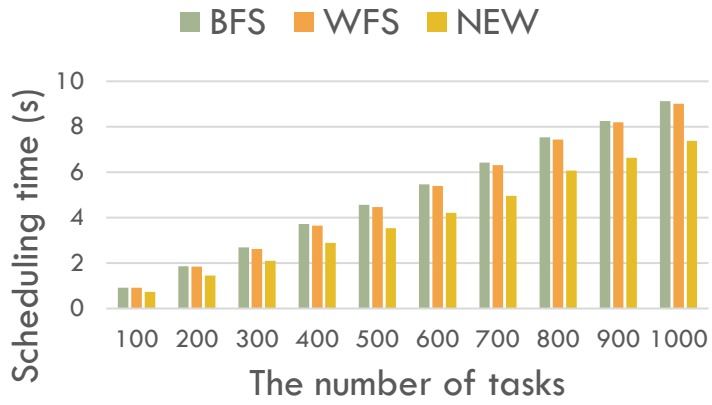
Evaluation Results

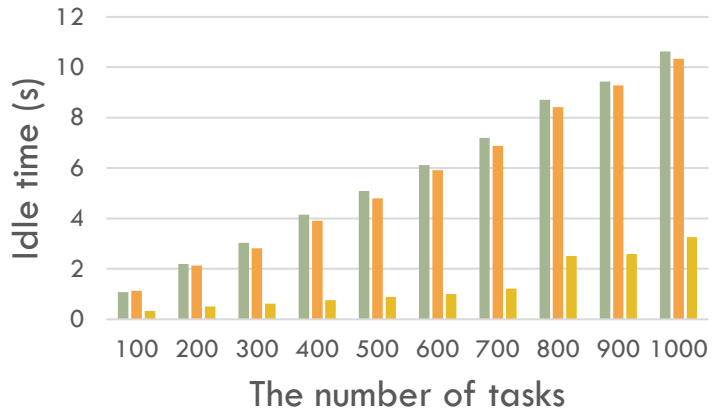
Tied tasks





Untied tasks





The proposed mapping:

- Decreases the scheduling time about 25% less than that of BFS and about 45% less than that of WFS
- Reduces the idle time about 75% less than that of BFS and about 85% less than that of WFS

The best heuristic pair: *best-fit*, *multi-criteria*

Future Works

The evaluation of the mapping under more complex system models, supporting:

- The creation of multiple child tasks by a task-part at TSP
- Different graphs (in size and proportion)
- Different numbers of threads

THANKS FOR YOUR ATTENTION!

For more information:
mmasa@isep.ipp.pt

isep Instituto Superior de Engenharia do Porto
UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH
BSC Barcelona Supercomputing Center
Centro Nacional de Supercomputación