Hamid Tabatabaee, Mohammad-R. Akbarzadeh-T, Naser Pariz, 2014. Dynamic task scheduling modeling in unstructured heterogeneous multiprocessor systems. *Journal of Zhejiang University-SCIENCE C (Computers & Electronics),* **15**(6):423-434. [doi:10.1631/jzus.C1300204]

JZUS C (Comput. & Electron)

Dynamic task scheduling modeling in unstructured heterogeneous multiprocessor systems

Key words: Dynamic task scheduling, Fuzzy logic, Genetic algorithms, Unstructured environment, Linear switching state space

Corresponding author: Hamid Tabatabaee
E-mail: hamid.tabatabaee@iauq.ac.ir

Introduction

- Many real world applications, such as distributed computing systems and management of projects, human resources, or financial resources, are examples of task scheduling problems.
- Task scheduling can be simply defined as the process of allocating dependent tasks to resources so as to minimize the total execution time. This problem is generally NP-complete.
- Task scheduling is nonlinear due to the many constraints in the environment, tasks, and resources that must be considered.
- A novel method is proposed to schedule tasks to a set of heterogeneous processors in a time-varying environment based on a linear switching state space (LS^3) system.

IJCS C (Comput & Electron)

System model

- **Task graph**: A task graph is a directed acyclic graph (DAG) $G=(V, E, \lambda, c)$ representing a program P according to the graph model, where V is the set of tasks in DAG, E is the set of edges in DAG, and λ is the instruction
- **Task**: A task is defined as the most basic element where further parallelism is neither necessary nor possible. Hence, all of the instructions for a task execute sequentially

JZUS-C (Comput & Electron)

$$\boldsymbol{X}[k+1]=\max\left[\boldsymbol{0}, \boldsymbol{X}[k]-\Delta T\cdot\operatorname{diag}\left\{\left(\sum_i \mathrm{CL}(\lambda_j^i)\frac{1}{\xi}\right)^{-1}\right\}\boldsymbol{U}\right],$$

Scheduling method

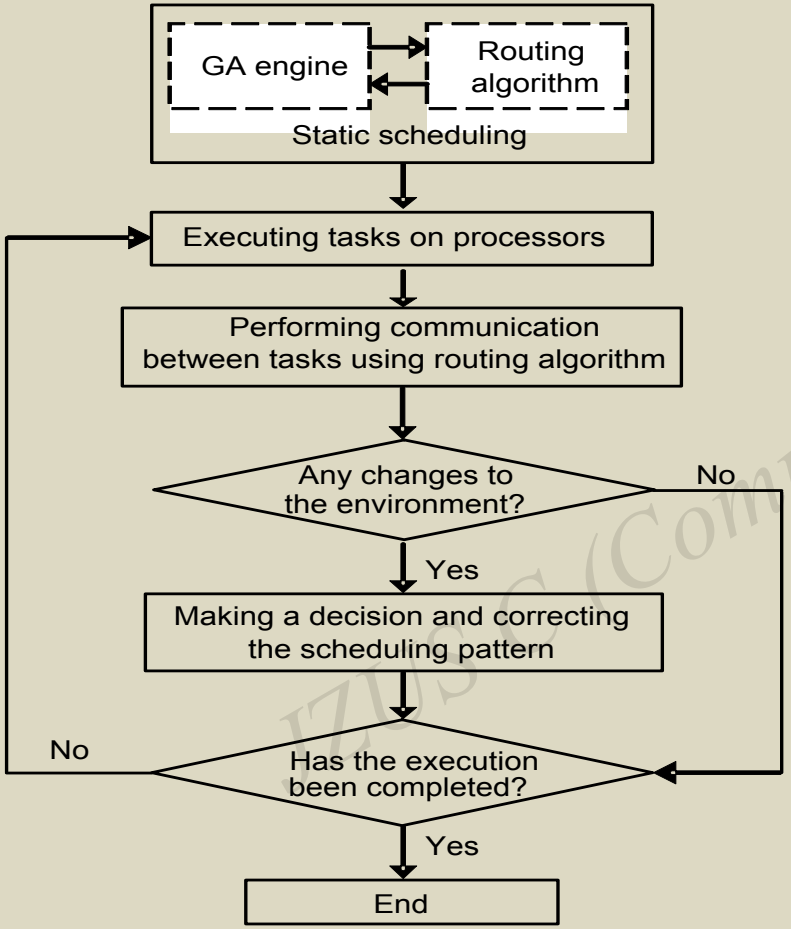


Fig. 1 Schematic of the linear switching state space (LS^3) scheduling mechanism

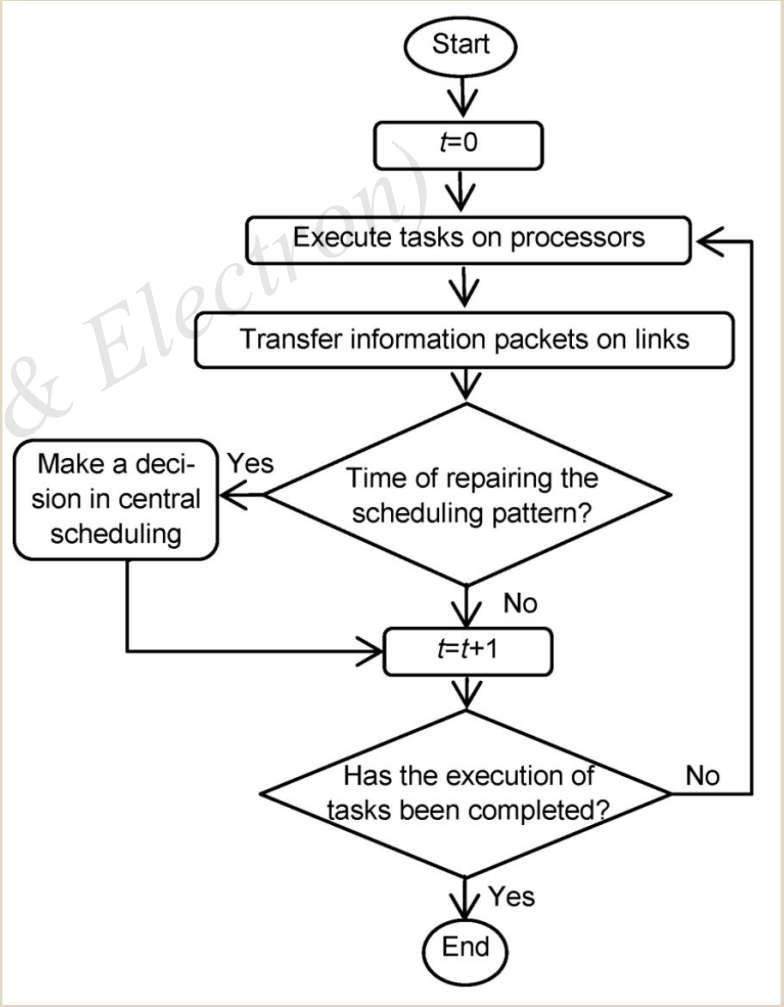
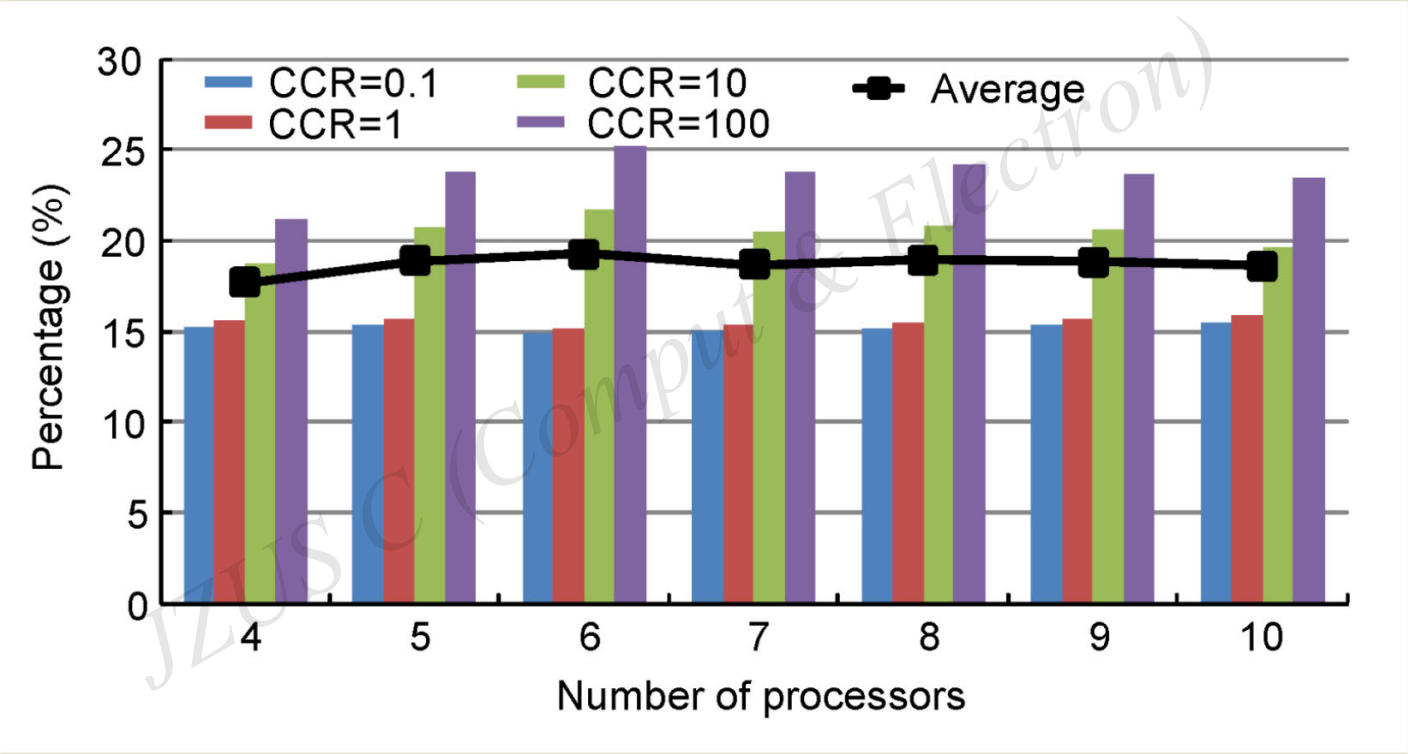


Fig. 3 Flowchart of time-varying scheduling

IZJS C (Comput & Electron)

Experimental results



JZUS C (Comput & Electron)

Fig. 5 Percentage improvement of LS^3 to GA (Crăciun *et al.*, 2010) for different CCRs and processor numbers

Conclusions

- A time-varying heterogeneous multiprocessor system is assumed in which the computational power of each processor may change over time.
- To schedule a DAG on such a multiprocessor system, we propose an algorithm based on a linear switching space model using a fuzzy heuristic approach.
- The fuzzy decision-making procedure migrates a scheduled task on one processor to another to achieve a better makespan.

JZUS-C (Comput & Electron)