# Homework - 4

Muhammad Talha Khan (46281174)
CS-230 Distributed Computer Systems

## Problem Statement

We are provided with 5 physical processors which will be referred to in this assignment as $\pi_1,\ \pi_2, \ldots \pi_5$. In addition to this we have 6 SPMD Jobs to run. The resources each job has are stated below:

$$
\begin{array}{l}
J_1 : \text{has 4 VPs} \\
J_2 : \text{has 3 VPs} \\
J_3 : \text{has 4 VPs} \\
J_4 : \text{has 1 VPs} \\
J_5 : \text{has 7 VPs} \\
J_6 : \text{has 2 VPs}
\end{array}
$$

For the sake of simplicity we will assume that no VP migration is allowed. Also we are provided with an initial spatial schedule/allocation which is also stated below.

$$
\begin{array}{ccccc}
 & & & & 6 \\
6 & & & & 5 \\
5 & 5 & & & 5 \\
3 & 4 & 5 & 5 & 3 \\
1 & 3 & 5 & 3 & 2 \\
1 & 2 & 1 & 1 & 2 \\
\pi_1 & \pi_2 & \pi_3 & \pi_4 & \pi_5
\end{array}
$$

A spatial schedule/allocation for a job $J_j$ is a $n$-tuple produced by a fucntion $A$, more precisely it is $A(J_j) = (a_1, a_2 \ldots, a_n)$ where $a_i$ defines the number of VPs of job $J_j$ assigned to physical processor $\pi_i$. The allocation functions for the above allocation are:

$$
\begin{array}{l}
A(J_1) = (2, 0, 1, 1, 0) \\
A(J_2) = (0, 1, 0, 0, 2) \\
A(J_3) = (1, 1, 0, 1, 1) \\
A(J_4) = (0, 1, 0, 0, 0) \\
A(J_5) = (1, 1, 2, 1, 2) \\
A(J_6) = (1, 0, 0, 0, 1)
\end{array}
$$

However, it can also be represented visually (Graphically) as it is shown above and for this assignment this is how I will show allocation. Also, I will be using a system schedule matrices to answer the problems.

# Problem 1: Finding a Legal Periodic Temporal Schedule

A temporal schedule for a time slice $t$ is defined as a $n$ tuple function , $S(t)$, such that $S(t)[i]$ is the job run by $\pi_i$ in the time slice $t$. A legal periodic schedule for an allocation $A$ is, $S_p = S(t), S(t+1), \ldots, S(t+1-p)$ if and only if there exists some $p$ such that $S(t) = S(t+p)$ for all $t > t_s$. Here, $t_s$ is the startup time of the schedule and $p$ is the period of the schedule. Also, each job $J$ can execute $k$ number of times in a period where $k$ is a positive integer and does not need to be the same for all jobs.

For the above allocation a legal periodic temporal schedule is given below in the form of schedule matrix.

| Time/PE | $\pi_1$ | $\pi_2$ | $\pi_3$ | $\pi_4$ | $\pi_5$ |
|---|---|---|---|---|---|
| 1 | 1 | 2 | 1 | 1 | 2 |
| 2 | 1 | 3 | 5 | 3 | 2 |
| 3 | 3 | 5 | 5 | 5 | 3 |
| 4 | 5 | 4 | | | 5 |
| 5 | 6 | 4 | | | 5 |
| 6 | | 4 | | | 6 |
| 7 | 1 | 2 | 1 | 1 | 2 |
| 8 | 1 | 3 | 5 | 3 | 2 |
| 9 | 3 | 5 | 5 | 5 | 3 |
| 10 | 5 | 4 | | | 5 |
| 11 | 6 | 4 | | | 5 |
| 12 | | 4 | | | 6 |

The above temporal schedule is shown for **2** periods. The above schedule is legal because for any time slice there is no VP of a job which is more than one global communication ahead of other VPs of the same job. The number of cycles in its period are **6** which satisfies the existence of $p$ condition. Also each job executes $k = 1$ number of times except for $J_4$ which executes $k = 3$ times in a period. The idling ratio for this periodic temporal schedule is given below.

$$Idling\ Ratio = \frac{7}{30} = 0.233$$

# Problem 2: Finding a Schedule with better Idling Ratio

For this part I will also change the provided allocation/spatial schedule. By looking at the spatial schedule above we can see that $\pi_3$ and $\pi_4$ are idle for most of the time. So, they can be assigned more work while they are sitting idle and this can lead to a better idling ratio. The new spatial scheduling that I choose is given below:

$$A(J_1) = (2, 0, 1, 1, 0)$$
$$A(J_2) = (0, 1, 0, 0, 2)$$
$$A(J_3) = (0, 1, 1, 1, 1)$$

$$A(J_4) = (1, 0, 0, 0, 0)$$
$$A(J_5) = (1, 2, 2, 2, 0)$$
$$A(J_6) = (1, 0, 0, 0, 1)$$

The system scheduling matrix for this allocation for two periods is shown below.

| Time/PE | $\pi_1$ | $\pi_2$ | $\pi_3$ | $\pi_4$ | $\pi_5$ |
|---|---|---|---|---|---|
| 1 | 1 | 2 | 1 | 1 | 2 |
| 2 | 1 | 5 | 5 | 5 | 2 |
| 3 | 5 | 5 | 5 | 5 | 6 |
| 4 | 6 | 3 | 3 | 3 | 3 |
| 5 | 4 | 3 | 3 | 3 | 3 |
| 6 | 1 | 2 | 1 | 1 | 2 |
| 7 | 1 | 5 | 5 | 5 | 2 |
| 8 | 5 | 5 | 5 | 5 | 6 |
| 9 | 6 | 3 | 3 | 3 | 3 |
| 10 | 4 | 3 | 3 | 3 | 3 |

The period of the above schedule is **5** and it is also legal because for any time slice no VP of a job is more than one global communication ahead of the same job. Since we have to run a total of **21** VPs and the number of physical processor our system has are **5**. We can only make the idling ratio zero if we run VPs of a job more than once on a processor that it was assigned to, in our case we just need to run a job with which requires 4 VPs so that we can fill the unused time slots (Can be other ways too). In this case the job that runs more than once in a period is $J_3$. Each job runs exactly once except for $J_3$ which runs 2 steps in a period. The Idling ratio for this periodic temporal schedule is given below:

$$Idling\ Ratio = \frac{0}{25} = 0$$

# Problem 3: Best Periodic Temporal Schedule

There can be more than one best periodic temporal schedules in this case, where they all have an idling ratio of zero. The one I have given in part 2 is one of those examples as it has an idling ratio of zero.