4 Comparison metrics

The comparison metrics used in this paper are makespan, Scheduling Length Ratio (SLR), speedup, efficiency, slack and frequency of best schedules.

4.1 Makespan

Makespan represents the point of time where computation of the last task in the graph completes. This denotes the total processing time of the input graph. It is defined as follows:

$$makespan = max\{AFT(t_{last})\}, \tag{17}$$

where $AFT(t_{last})$ denotes the Actual Finish Time of the last task. Actual Finish Time of a task is the point of time where the task finishes its execution on some allocated processor. Makespan and schedule length are used interchangeably in this paper.

4.2 Schedule length ratio

The Schedule Length Ratio (SLR) is a commonly used metric to compare scheduling algorithms, and it is defined as the fraction of the makespan to the sum of the smallest execution cost of all tasks on the critical path of the DAG [36].

$$SLR = \frac{makespan}{\sum_{t_i \in CP_{MIN}} \min_{p_j \in P}(w_{(i,j)})}, \tag{18}$$

where $w_{(i,j)}$ denotes the execution cost of task t_i on processor p_j. In Eq. (18), the denominator gives the lower bound on the schedule length. Thus, the algorithm having the lowest value of SLR than the other algorithms is the best algorithm.

4.3 Speedup

The speedup is defined as the fraction of the sequential computation time to the parallel computation time or the makespan of the schedule.

$$Speedup = \frac{Sequential\ computation\ time}{makespan}. \tag{19}$$

In Eq. (19), the numerator can be calculated by allocating all tasks to one processor which gives the minimum sum of the computation cost of the DAG.

4.4 Efficiency

Efficiency is defined as the speedup divided by the quantity of processors used in each run.

$$Efficiency = \frac{Speedup}{Number\ of\ processors\ used}. \tag{20}$$

4.5 Frequency of best schedules

This metric is used to show that how many schedules produced by an algorithm are better, worse and equal when compared to other algorithms used in the experimentation. The results of algorithms for this metric are represented by a comparison table.

4.6 Slack

This metric is used to determine the robustness of the schedules generated by an algorithm when compared with the uncertainty in the tasks processing time [6,31]. It can be expressed as follows:

$$Slack = \frac{\left[\sum_{t_i \in V} makespan - b_{level}(t_i) - t_{level}(t_i)\right]}{n}, \tag{21}$$

where n is the task quantity, $b_{level}(t_i)$ is the length of the largest path from task t_i to the end node, and $t_{level}(t_i)$ is the length of the largest path from the starting task to the task t_i. It signifies the ability of the schedule to consume delays in task computation. The slack of a task denotes the time slot where the task can be deferred without enhancing the schedule length. Schedule length and slack are contradictory metrics such that higher schedule lengths give large slack.

5 Experimental results and discussion

Here, we provide the evaluation results of the HEFT, PETS, LDCP, Lookahead, CEFT and PEFT algorithms and discuss the results using various performance metrics on various graphs. For this purpose, three sets of task graphs are taken into consideration: peer set task graphs, randomly generated task graphs and the task graphs derived from real-world applications. The experiments are performed on a Dell PowerEdge R420 server with CentOS (version 7.3-1611), Intel(R) Xeon(R) CPU E5-2420 v2 @ 2.20 GHz processor, 2 processor sockets having 6 execution cores per processor and 192 GB of memory.

5.1 Randomly generated application graphs

The random application graphs are produced with a Random DAG generator [1] to benchmark and estimate the performance of the well-known list scheduling algorithms on various parameters. The DAGs generated here possess different characteristics which rely on some input parameters as mentioned below.

– n: It represents the DAG size.