ABSTRACT

The importance of scheduling has increased in recent years due to the growing consumer demand for variety of reduced product life cycles, changing markets with global competition and rapid development of new processes and technologies. Two cases of the task scheduling exist in literature namely independent and dependent task scheduling. The independent or dependent tasks can be static or dynamic. Scheduling static tasks require a priori knowledge of its execution time on the processing nodes. Dynamic task scheduling scheme does not require a prior knowledge of the tasks to be scheduled. The tasks considered for scheduling can be preemptive or non-preemptive in nature. The minimization of the make span is the intent of the task scheduling problem.

The thesis work comprises a snapshot of particle swarming techniques including variations in the algorithm, and its applications in the area of multiprocessor scheduling. Algorithm inspired from Particle Swarm Optimization (PSO) method and its variants are successfully implemented for multiprocessor scheduling optimization problems. Its efficiency for solving the problem is to minimize the make span criterion. Genetic Algorithm, PSO and its variants are used to solve the task scheduling problem. Static task scheduling is implemented for both dependent and independent tasks. Dynamic task scheduling is analyzed under two conditions namely with load

balancing and without load balancing. Load balancing is also considered for the cost optimization.

PSO has undergone many changes since its introduction in 1995. As researchers have learned about the technique, they have derived new versions, developed new applications, and published theoretical studies of the effects of the various parameters and aspects of the algorithm.The basic PSO developed by Kennedy and Eberhart (1995) simulates the movement of flocks of birds. PSO is a population based search algorithm which works on a set of feasible solutions.

The thesis aims at solving the multiprocessor scheduling problem using PSO and its variant techniques namely PSO with dynamically varying inertia, Elite PSO with mutation, Hybrid PSO, Parallel PSO, Orthogonal PSO and Parallel Orthogonal PSO methods.

The introduction of inertia factor in the basic equation of the PSO algorithm has proven a significant improvement in the results when applied to the multiprocessor scheduling problem. The value of the inertia factor plays a major role in the achievement of the optimal solution. The fixed inertia and dynamically varying inertia are applied to solve the task assignment problem. PSO with variable inertia yields an improved performance than fixed inertia when applied to multiprocessor scheduling problem.

The performance of the PSO Algorithm is enhanced by modifying its basic working methodology. The PSO with varying inertia method is also combined with another proposed technique known as elitism with mutation to

achieve an improvement in the result when applied to the task assignment problem. Elitism is combined with mutation to prevent the algorithm from being stuck at local optimum.

Hybridization of the PSO algorithm and the Simulated Annealing (SA) algorithm is done to enhance the performance of the algorithm. Simulated Annealing is chosen because it is good at finding the local optimum. Performance improvement is achieved when hybridization technique of PSO and SA is applied to multiprocessor scheduling. This hybrid technique is also compared with another version of hybridized PSO namely the combination of the PSO and the Hill Climbing concept.

Parallelization of the PSO algorithm is proposed to speed up the execution. Two versions of parallelization are done namely the Synchronous Parallel PSO and the Asynchronous Parallel PSO. The result infers that the Asynchronous version performs better than the Synchronous Parallel version.

Further, the Orthogonal PSO (OPSO) is proposed to refine the initial population. The parallelization of the OPSO algorithm (POPSO) is also proposed to enhance the results. The results infer that the POPSO outperforms all other methods tested. The scope for the future studies is that the proposed approaches can be tested for pre-emptive tasks.