IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE)
e-ISSN: 2278-1684,p-ISSN: 2320-334X, Volume 12, Issue 2 Ver. II (Mar - Apr. 2015), PP 50-53
www.iosrjournals.org

A Review on Non-Traditional Optimization Algorithm for Simultaneous Scheduling Problems

P. Sreenivas[1], S. Vijaya Kumar [2]
[1,2,] (Assistant Professor, Dept of Mechanical Engineering, K.S.R.M College of Engg, Kadapa, Andhra Pradesh, India)

Abstract- *This paper focuses on the applications of non-traditional optimization method. Here several un conventional optimization were available in literature are critically reviewed to solve this combinatorial optimization problem. In this paper In this paper, authors seek to assess the work done in the simultaneous scheduling domain by providing a review of many of the techniques used for the industrial and production environment. It is established that Non- conventional optimization methods should be considered complementary rather than competitive. In addition, this work suggests guide-lines on features that should incorporated to create a good scheduling system. Finally, the possible direction for future work is highlighted so that current barriers within applications of non traditional optimization method may be surmounted as researchers approach in the 21st century.*

Keywords *- non conventional optimizations, simultaneous scheduling, Industrial production environment, review*

I. Introduction

The application of optimization algorithms to real world problems has gained momentum in the last decade. Dating back to the early 1940s, diverse traditional mathematical methods such as linear programming (LP), nonlinear programming (NLP) or dynamic programming (DP) were first employed for solving complex optimization problems by resorting to different relaxation methods of the underlying formulation.

These techniques are capable of cost efficiently obtaining a global optimal solution in problem models subject to certain particularities (e.g. optimal sub structure ability and sub problem overlap for dynamic programming), but unfortunately their application range does not cover the whole class of NP complete problems, where an exact solution cannot be found in polynomial time. In fact, the solution space (and hence, the solving time) of the problem increases exponentially with the number of inputs, which makes them unfeasible for practical applications.

II. Literature Survey

In the course of the most recent three epochs much exploration has been done in this part. Lots of Metaheuristic algorithms have been advanced to create ideal schedule and part-releasing policies. Maximum of these algorithms comprise enumerative procedures, mathematical programming and approximation techniques, i.e., linear programming, integer programming, goal programming, dynamic programming, transportation and network analysis, branch and bound, priority-rule-based heuristics, local search algorithms (ITS, TS, SA), evolutionary algorithm (GA), etc. Of these techniques, little is specific to particular objectives, and few are precise to particular problem illustrations with respect to computational time required. Shankar and Tzen [1] measured scheduling problems in a random FMS as composite independent tasks. Lee [2] revealed a goal-programming model for multiple conflicting objectives in manufacturing. Toker et al. [3] proposed an approximation algorithm for 'n' job 'm' machine problem. Steeke and Soldverg [4] investigated various operating strategies on a caterpillar FMS by meansof deterministic simulation with the number of completed assemblies on a performance criterion manufacturing problem related with parallel identical machines throughout simulation. Chan and Pak [5] suggested two heuristic algorithms for solving the scheduling problem with the goal of minimizing total cost in a statictically loaded FMS. Shaw and Winston [6] spoke anartificial intelligence approach to the scheduling of FMS Schultz and Merkens [7] equated the performance of an ES, a GA and priority rules for production systems.

III. Objectives Of Simultaneous Scheduling

The scheduling is made to meet specific objectives. The objectives are decided upon the situation, market demands, company demands and the customer's satisfaction. There are two types for the scheduling

Objectives :
[1] Minimizing the make span
[2] Due date based cost minimization

The objectives considered under the minimizing the make span are,
(a) Minimize machine idle time

(b) Minimize the in process inventory costs
(c) Finish each job as soon as possible
(d) Finish the last job as soon as possible

The objectives considered under the due date based cost minimization are,
(a) Minimize the cost due to not meeting the due dates
(b) Minimize the maximum lateness of any job
(c) Minimize the total tardiness
(d) Minimize the number of late jobs

IV. Scheduling Techniques

These techniques are mainly divided into two categories i.e. Traditional and Non Traditional. A brief introduction of these techniques is given below.

(a) Traditional techniques:

These techniques are slow and guarantee of global convergence as long as problems are small. Traditional Techniques are also called as Optimization Techniques. They are

- Mathematical programming like Linear programming, Integer programming, Dynamic programming, Transportation etc.
- Enumerate Procedure Decomposition like Lagrangian Relaxation.

(b) Non traditional techniques:

These methods are very fast but they do not guarantee for optimal solutions. Non Traditional Techniques are also called as Approximation Methods. They involve

- Constructive Methods like Priority dispatch rules, composite dispatching rules.
- Insertion Algorithms like Bottleneck based heuristics, Shifting Bottleneck Procedure.
- Evolutionary Programs like Genetic Algorithm, Particle Swarm Optimization.
- Local Search Techniques like Ants Colony Optimization, Simulated Annealing, adaptive Search, Tabu Search, Problem Space methods
- Iterative Methods like Artificial Intelligence Techniques, Artificial Neural Network, Heuristics Procedure, Beam-Search, and Hybrid Techniques.

V. Some Non Traditional Techniques

5.1 Genetic Algorithm

In actual fact, a GA is a set of techniques which when common enable solutions to particular problems. To accomplish the objectives, the GA produces consecutive population alternate solutions until a solution is obtained which yields acceptable results. With in the generation of each successive population, improvements in the quality of the individual solutions are increased. In this manner, a GA can rapidly transfer to a fruitful result without having to inspect all likely solution to the problem. The procedure used is centered on the vital processes that regulate the growth of biological organisms, namely, natural selection and reproduction. These two processes together improve an organism's capacity to persist within its atmosphere in the following manner:

1. Usual selection governs which organisms will have the chance to reproduce and persist within a population.
2. Reproduction involves genes from two discrete individuals uniting to form offspring that take over the persistence features of their parents. These algorithms pursue to start the manner in which are useful genes reproduce themselves from end to end consecutive populations and in future subsidize to the steady ability of an organism to stay alive.

5.2 Simulated Annealing

The simulated annealing algorithm has its roots in statistical mechanics. The concern in simulated annealing initiated with the work ofKirkpatrick [9], and Cemy [10]. They proposed a simulated annealing algorithm, which is based on the comparison between the annealing process of solids and the problem of solving optimization problems. Launch in 1983, simulated annealing was advertised as a global optimization procedure that mimics the physical annealing process by which molten substances cool to crystalline lattices of minimal energy. This marketing scheme had a polarizing effect, attracting those who enchanted in metaphor and separating others who found metaphor inadequate at best and facile at worst. Actually, the emotional outbreaks that supplement many negotiations of simulated annealing are an unfortunate disturbance. Whatever its pros and cons, simulated annealing can be deal with in rigorous mathematics.

At the heart of the method of simulated annealing is an correlation with thermodynamics, exactly with the manner that liquids freeze and crystallize, or metals cool and anneal. At high temperatures, the molecules of a liquid move freely with respect to one another. If the liquid is cooled gently, thermal mobility is gone. In reality, if a liquid metal is cooled rapidly or "quenched," it does not reach this state but rather ends up in a polycrystalline or amorphous state having slightly higher energy. So the principle of the process is slow cooling, allowing sufficient time for restructuring of the atoms as they drop mobility. This is the mechanical definition of

annealing, and it is vital for confirming that a low energy state will be attained. Even if the analogy is not perfect, there is a logic in which all of the minimization algorithms correspond to rapid cooling or quenching. In all cases, we have gone greedily for the fast, nearby solution: From the starting point, go immediately downward as far as you can go. This leads to a local, but not necessarily a global, minimum. Nature's own minimization algorithm is based on quite a different procedure. The so-called Boltzmann probability distribution,
Prob (E) = exp(−E/kT)

5.3 swarm intelligence

PSO is a robust stochastic optimization technique based on the movement and intelligence swarms. It was developed in 1995 by James Kennedy (social psychologist) and Russell Eberhart (Electrical Engineer). PSO is a method for optimizing hard numerical functions on metaphor of fish. Suppose the following scenario, a flock of birds is randomly searching for food in an area, where there is only one piece of food available and none of them knows where it is, but they can estimate how for it would be at each iteration. The problem here is what is the best strategy to find and get the food. Obviously the simplest strategy is to follow the bird known as the nearest one to the food.

PSR inventors were inspired of such natural process based scenarios to solve the optimization problems. In PSO each single solution, called a particle, is considered as a bird, the group becomes a swarm (population) and the search space is the area to explore. Each particle has a fitness value calculated by a fitness function, and a velocity of flying towards the optimum, the food. All particles fly across the problem space following the particle nearest to the optimum. PSO starts with initial population of solutions, which is updated iteration by iteration. Therefore PSO can be counted as an evolutionary algorithm besides being a meta heuristics method, which allows exploiting the searching experience of a single particle as well as the best of the whole swarm.

In a PSO algorithm, [9] swarm is initiated randomly with finding the personal best (best value of each individual so far) and global best (best particle in the whole swarm). Initially, each individual with its dimensions and fitness value is assigned to its person best. The best individual among particle best swarm, with its dimension and fitness value is, on the other hand, assigned to the global best. Then a loop starts to converge to an optimum solution. In the loop, particle and global bests are determined to update the velocity first. Then the current position of each particle is updated with the current velocity. Evaluation is again performed to complete the fitness of particles in the swarm. This loop is terminated with a stopping criterion predetermined in advance. The application of the PSO requires that parameters are initialized and the population to be generated randomly.

5.4 Artificial Immune System

Artificial Immune Systems have emerged during the last decade. They are incited by many researchers to design and build immune-based models for a variety of application domains. Artificial immune systems can be defined as a computational paradigm that is inspired by theoretical immunology, observed immune functions, principles and mechanisms. The function of biological IS is to protect the body from the foreign matters, more known as antigens. Antigens stimulate the antibodies that reside in the body. The key roles of antibodies are to identify, bind and eliminate the antigens. Clonal selection explains the response of IS, when a non-self antigen pattern is recognized by the B cells. It is selected to proliferate and produce antibodies in high volume by cloning. The new clonal cells undergo hype rmutation for improving antibodies affinity that leads to antigenic neutralization and elimination. The overall procedure of clonal selection is schematically shown in Fig. 1

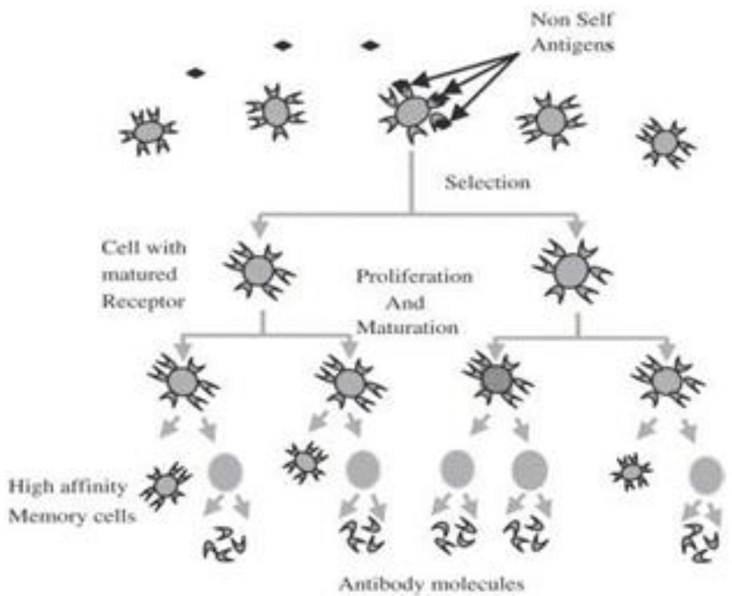

Fig. 1

5.5 Differential Evolution (DE) Algorithm

Differential Evolution (DE) is the Stochastic, population-based optimization algorithm. It is one of the Evolutionary Algorithms (EAs) for global optimization over continuous search space (Storn and Price [2], 1995). Its theoretical framework is simple and requires inexpensive computation in term of CPU time (Bin et al., 2008). Due to its advantage of relatively few control variables but performing well in convergence, DE has been widely applied and shown its strengths in many application areas.

Working principle of DE

In DE algorithm, solutions are represented as chromosomes based on floating-point numbers. In the mutation process of this algorithm, the weighted difference between two randomly selected population members is added to a third member to generate a mutated solution followed by a crossover operator to combine the mutated solution with the target solution so as to generate a trial solution. Then a selection operator is applied to compare the fitness function value of both competing solutions, namely, target and trial solutions to determine who can survive for the next generation. The basic DE algorithm consists of four steps, namely, initialization of population, mutation, crossover and selection.

1. Population initialization:
2. Mutation
3. Crossover
4. Selection operat

5.6 Ant Colony Algorithm

ACO algorithms are inspired by the foraging behaviour of natural ant colonies in which individual ants deposit a substance called pheromone on the path while moving from the nest to the food sources and vice versa. Other ants smell this pheromone to find the food sources. The more is the pheromone in a path, the higher would be the probability of selecting that path. Consequently, after some time all the ants would select the shortest path from the nest to the food source. For more information in this regard, the interested reader is referred to[**11**]

Any ant algorithm must specify the following elements:
(1) construction of solutions,
(2) heuristic information,
(3) pheromone updating rule,
(4) selection probability,
(5) local search algorithm, and
(6) stopping criterion.

VI. Conclusions

Since Simultaneous scheduling problems fall into the class of NP-complete problems, they are among the most difficult to formulate and solve. Some optimization problems (including various combinatorial optimization problems) are sufficiently complex that it may not be possible to solve for an optimal solution with the kinds of exact algorithms. In such cases, heuristic methods are commonly used to search for a good (but not necessarily optimal) feasible solution. Several metaheuristics are available that provide a general structure and strategy guidelines for designing a specific heuristic method to fit a particular problem.

A key feature of these metaheuristics procedures is their ability to escape from local optima and perform a robust search of a feasible region. This paper introduces the most prominent types of non-conventional type algorithms or meteheuristics. In addition, it may employ intensification and diversification strategies based on long-term memory to focus the search on promising continuous. The following are the advantages of non-traditional techniques over the traditional techniques:

- The non-traditional techniques yield a global optimal solution.
- The techniques use a population of points during search.
- Initial populations are generated randomly which enable to explore the search space.
- The techniques efficiently explore the new combinations with available knowledge to find a new generation.
- The objective functions are used rather than their derivatives

References

[1]. Shankar K,Tzen Y.J. A loading and dispatching problem in a random flexible manufacturing system. Int J Prod Res 23: 579- 595. 1985.
[2]. Lee, S.M., Jung H.J. A multi objective production planning model in flexible manufacturing environment. Int J Prod Res 27 (11): 1981-1992
[3]. Toker, A., Kondacki, S., Erkip, N., Job shop scheduling under a non-renewable resource constraint. J Oper Res Soc 45(8): 942-947, 1994.
[4]. Steeke, K.E., Soldberg, J.J., Loading and control policies for a flexible manufacturing system. Int J Prod Res 19(5):481 -490, 1982.
[5]. Chan, T.S., Pak, H.A., Heuristic job allocation in a flexible manufacturing system. Int J AdvManufTechnol 1(2):69-90, 1986.
[6]. Shaw, M.J., Whinston, A.B., An artificial intelligence approach to the scheduling of flexible manufacturing systems.IEEE Trans 21:170-182,1989.
[7]. Schulz, J., Mertens P., A comparison between an expert system, a GA and priority for production scheduling. In: Proceedings of the 1st international conference on operations and quantitative management, Jaipur, India, 2.506-513, 1997.
[8]. Richard W. Conway, William L. Maxwell, Louis W. Millep "Theory of Scheduling". Addison – Wesley Publishing Company (1967).
[9]. Kirkpatrick, S; Gelatt, D.D and Vecchi, M.P. (1983).Optimization by Simulated Annealing. Science, Vol.220,pp 671-680.
[10]. Cemy, V. (1985). Thermodynamic approaches to the travelling salesman problems. J Optim. TheoryAppl. Vol.45,pp 41-51
[11]. Solimanpur and prem virat, An ant algorithm for the single row layout problem in Fexible manufacturing systems, Computers & Operations Research 32 (2005) 583-598.