International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:5, No:7, 2011

A Multi-Objective Optimization Model to the Integrating Flexible Process Planning And Scheduling Based on Modified Particle Swarm Optimization Algorithm (MPSO)

R. Sahraian, A. Karampour Haghighi, E. Ghasemi

***Abstract*—**Process planning and production scheduling play important roles in manufacturing systems. In this paper a multi-objective mixed integer linear programming model is presented for the integrated planning and scheduling of multi-product. The aim is to find a set of high-quality trade-off solutions. This is a combinatorial optimization problem with substantially large solution space, suggesting that it is highly difficult to find the best solutions with the exact search method. To account for it, a PSO-based algorithm is proposed by fully utilizing the capability of the exploration search and fast convergence. To fit the continuous PSO in the discrete modeled problem, a solution representation is used in the algorithm. The numerical experiments have been performed to demonstrate the effectiveness of the proposed algorithm.

***Keywords*—**Integrated process planning and scheduling, multi objective, MILP, Particle swarm optimization

I. Introduction

A job shop manufacturing environment is characterized by the make-to-order operation and customers' demand is more and more unpredictable. New manufacturing systems planning are required to deal with this environment, one which facilitates flexibility, reduces design cycle time, reduces time to market for new products and reduces order turnaround time for existing products [1].Process planning and scheduling play important roles in manufacturing systems. They are usually complementary activities. Process planning, an essential component for linking design and downstream manufacturing processes, provides a detailed operational guidance for scheduling [2]. One of the core activities in process planning is to decide which manufacturing resources to select and in which sequence to use, mainly based on the objective of achieving the correct quality, the minimal manufacturing cost and ensuring good manufacturability. Scheduling determines the most appropriate moment to execute each operation for the realized production orders, taking into account the due date of these orders, minimum makespan, balanced resource utilization, minimum transition time and etc., to obtain high productivity in the workshop [3]-[4].

A. K. Department of Industrial Engineering, Shahid Bahonar University of Kerman, Kerman, Iran. (phone: 0098 341-211-2861; fax: 0098 341-211-2861; e-mail: Ali.karampoor@ yahoo.com).

E. G. Department of Industrial Engineering, Shahid Bahonar University of Kerman, Kerman, Iran. (e-mail: Ghasemi.egh@gmail.com).

R. S. Department of Industrial Engineering, Shahed University, Tehran, Iran. (e-mail:).

A process plan is usually determined before the actual scheduling, regardless the scheduling objectives and with the assumption that all the resources are available. However, if a process plan is prepared offline without consideration of the actual shop floor status, due to changes or constraints in the manufacturing environment, it may become unfeasible and heavily unbalanced resource assignments.By integrating these functions, more flexible and effective schedules can be produced. Increasing production feasibility and optimality is the merit of integrated process planning and scheduling (IPPS) by combining both the process planning and scheduling problems [5].Due to industry's interest in improving the overall competitiveness in the global market place and reducing costs, the integration of process planning and scheduling has received increasing attention in recent years.Generally, the traditional job shop scheduling literature assumed that there is a single feasible process plan for each job [6]. This implies that no flexibility in the process plan is considered. Traditionally, these two functions are accomplished in two different stages. Production scheduling will get its input from the complete process planning. This results in conflicting objectives and the inability to communicate the dynamic changes in the shop floor [7]. Meanwhile, since the process planning is in advance of the scheduling, the schedule generated by these process plans may suffer from the lower resource utilization and poor on-time delivery performance. Therefore, to maintain the feasibility and optimum of the schedule, it is inevitable to revise the existing plans for some jobs.The most recent works related to the IPPS optimization can be generally classified into two categories: the enumerative approach and the simultaneous approach [3]. In the enumerative approach [2]-[8]-[9], multiple alternative process plans for each part are first generated. A schedule can be determined by iteratively selecting a suitable process plan from alternative plans of each part to replace the current plan until a satisfactory performance is achieved. The simultaneous approach [10]-[11] is based on the idea of finding a solution from the combined solution space of process planning and scheduling.Reference [12] addressed the simultaneous planning and scheduling of single-stage multi-product continuous plants with parallel lines.

Based on the process planning optimization, optimal scheduling is generated in the whole plan solution space in terms of available machines and precedence constraints [13].

Scheduling system has a large number of feasible process plans, so optimality of scheduling not only depends on the manufacturing resources but depends on the result of the process plans [14]. Therefore, process planning and scheduling are tightly interwoven with each other. Integrating the process planning and scheduling is necessary to achieve a global optimum in manufacturing, increase the flexibility and responsiveness of the systems.To address the above two optimization problems, some optimization approaches based on modern heuristics or evolutionary algorithms, have been developed in the last two decades and significant improvements have been achieved. Such as the genetic algorithm (GA) (for operation sequencing problem [15]; for IPPS problem [6]-[11], simulated annealing (SA) algorithm (for operation sequencing problem [16]; for IPPS problem [3], Tabu search algorithm (for operation sequencing problem [17], for IPPS problem [10]) and agent-based approach for IPPS problem [18]. However, these two optimization processes are well known as complicated decision problems for the multiple parts with complex structures and features. The major difficulties include: (1)both operation sequencing and IPPS problems are NP-hard combinatorial optimization problems .The search space of IPPS problem is usually very large because of involving multiple parts' scheduling while many previous developed methods could not find optimized solutions effectively and efficiently. (2) In sequencing operations and constraints of manufacturing resource utilization there are usually a number of precedence constraints, which make the search more difficult. (3) The objectives in the process planning and scheduling are in conflict with respect to each other, which are both important to production.Therefore, it is necessary to develop efficient models for the operation sequencing and the IPPS optimization problems therefore the optimization algorithms need to be more agile and efficient to solve practical cases.Motivated by this, we proposed a multi-objective optimization approach to perform the explorative search in such a large solution space. In this work we considered a MILP (Mix integer linear programming) scheduling model thus a slot-based Multi-Objective Multi-Product that readily accounts for sequence dependent transition times and set-up times, which set-up times and the machine change over times between operations is included in the transition time for each operation.The paper is organized as follows. Section 2 and 3 provide a summary of the process planning problems and job shop scheduling problem, respectively. In section 4, we present the problem statement. This is followed by the formulation of the multi objective MILP model proposed for the integrated process planning and scheduling. Section 5 introduces some basic concepts in the multi-objective optimization. Section 6 presents a PSO-based approach for resolving the integrated process planning and scheduling. Section 7 shows a numerical experiment to validate the performance of the proposed algorithm. Finally, the paper concludes in section 8.

II. Process planning problem

To conduct process planning, parts are represented by manufacturing features. Each feature can be manufactured by one or more machining operations (n operations in total for the part). Each operation can be executed by several alternative plans if different machines, cutting tools or set-up plans are chosen for this operation [19]-[20]. A set-up is usually defined as a group of operations that are machined on a specified machine with the same fixture.A process plan for a part consists of all the operations needed to machine the part and their relevant machines, cutting tools, TADs (Tools Approach Direction), and operation sequences. A good process plan of a part is built up based on two elements: (1) the optimized selection of the machine, cutting tool and TAD for each operation; and (2) the optimized sequence of the operations of the part.

III. Job Shop scheduling problem

A general scheduling problem can be stated as: n jobs {J1, J2...Jn}, job means a single part (or batch) or item, has to be processed through m machines {M1, M2... Mm}. The required Nomenclature has been defined (TABLE I). The processing of a job Jj on a machine Mi is called operation, Oij. For each operation, there is an associated processing time tij. In addition there may be a ready time (or release date) rj associated with each job and/or a due date by which time Jj should be completed. In general job shop scheduling, every job may have a different routing through machines. A problem of n jobs and m machines has an infinite number of feasible solutions since idle times between operations can vary. These number of feasible solutions increase exponentially along each parameter (such as number of machines and number of jobs). Due to the complex combinatorial problem, the theory and techniques of scheduling have received a lot of attention from OR practitioners, management scientists, production and operations research workers and mathematician [21].In the process planning and scheduling, different criteria are used to address specific practical cases. For instance, from the process planning perspective, the lowest manufacturing cost is usually a desired target, while the scheduling usually needs to look for the most balanced utilization of machines, the minimum number of tardy jobs, the shortest makespan and etc. To meet the various requirements in practical situations, further improvement is required on the optimization algorithm to make it more adaptive to accommodate diverse objectives for users to choose from them.

IV. Integrated process planning and scheduling

A slot-based MILP scheduling model is proposed that readily accounts for sequence dependent transition times and transition costs.

The problem is then to determine:

1. The products to be produced in each machine and in each time slot.

2. The sequence and detailed timing of production as well as the length of production times in each machine

The objective is to minimize transition times and total tardiness.

This section describes the proposed slot-based MILP model for the simultaneous planning and scheduling of multi-product.We postulate Nm asynchronous time slots for each machine m where Nm is the total number of products that can be processed by machine m (Fig. 1). by the model, Assignments of products to the slots are determined to define the sequence of product operations. Binary variable Wi,p,j,m,l means the p th process planning is applied to job i and the j th operation of this process planning needs to be processed on slot l of machine m . The length of each time slot is a variable to be determined by the model, and is equal to the summation of the assigned product's processing time and the corresponding transition times (Fig. 2). Sequence dependent transitions are activated depending on the assignments of products to slots.

A. Assignments and Model

Equation(1) represents the condition that exactly one operation of all products must be assigned to each slot of each machine.

According to (2) the processing time of operation j of product *i* in slot l of machine *m* is set to zero, if product i is not assigned to slot l of machine *m*. Constraint (2) also defines an upper bound, the length of each time period *H*, on the processing time.

$$\sum_{i}\sum_{j} wi,p,j,m,l = 1 \qquad (1)$$

$$\theta_{i,p,j,m,l} \leq H * w_{i,p,j,m} \qquad (2)$$

TABLE I
REQUIRED NOMENCLATURE

Nomenclature	description
i,k	Products
L	Slot
M	Machine
H	Duration of the time period
$\tau_{i,k}$	Transition time from machine *i* to machine *k*
$S_{i,p,j,k,f,z,m,l}$	Operation j of product *I of p th process plan* assigned to slot *l* of machine *m* is followed by product *k* assigned to slot *l* + 1 of machine
$W_{i,p,j,m,l}$	p th process planning is applied to job *i* and the j th operation of this process planning needs to be processed on slot *l* of machine
$\Theta_{i,p,j,m}$	the processing time of operation j of product *i* in slot *l* of machine *m*
$ET_{m,l}$	the end time of slot *l* of machine *m*
$ST_{m,l}$	the start time of slot *l* of machine *m*
$ST_{i,p,j}$	the starting time of operation *j of job i* of process plan *p*
C_i	the completion time of the job *i*.
d_i	the due date of the job *i*.
C^p	objective functions to minimize the transition times
Tar	objective functions to minimize the totol tardiness

1. *Transition Times*

In order to take into account sequence-dependent transitions, we introduce the transition time variable *Si,k,m,l:*

$$S_{i,p,j,k,f,z,m,l} \begin{cases} 1 & \text{if operation j of product } I \text{ of } p \text{ th process plan assigned to slot } l \text{ of machine } m \text{ is followed by product } k \\ 0 & \text{otherwise} \end{cases}$$

$$W_{i,p,j,m,l} \wedge W_{k,f,z,m,l+1} \Leftrightarrow S_{i,p,j,k,f,z,m,l} \qquad \forall i\neq k\in I(m),\ i\neq k \quad \forall l\in Lm, \forall m \qquad (3)$$

The proposition in (3) links these transition time variables ($S_{i,p,j,k,f,z,m,l}$) with the assignment variables ($W_{i,p,j,m,l}$). That is, $S_{i,p,j,k,f,z,m,l}$ should become 1 if and only if operation j of product *i of p th process plan* assigned to slot *l* of machine *m* is followed by product *k* assigned to slot *l* + 1 of machine *m.*

Mathematically, another way of enforcing the same condition in (3) is to use the following set of propositions:

$$\sum_{k\in I(m)} S_{i,p,j,k,f,z,m,l} = W_{i,p,j,m,l} \qquad (5)$$

2. *Timing Relation*

In (6), the end time of slot *l* of machine *m is* equal to the start time of slot *l* plus the corresponding transition times plus the processing time of the product assigned to slot.

According to (7), the end time of each slot must be equal to the start time of the consecutive slot and Eq. (8) represents the route constraint. Let $ST_{i,p,j}$ denote the starting time of operation j of job i of process plan p.

$$ET_{m,l} = ST_{m,l} + \sum_{i\in I(m)}\sum_{k\in I(m)} \tau_{i,k} * S_{i,p,j,k,f,z,m,l} + \sum_{i\in I(m)} \theta_{i,p,j,m,l} \qquad (6)$$

$$ET_{m,l} = ST_{m,l+1} \qquad (7)$$

$$ST_{i,p,(j+1)} \geq ST_{i,p,j} + \theta_{i,p,j,m} \qquad (8)$$

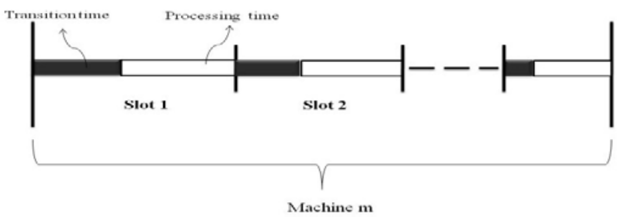

Fig. 1 All of the possible slots on the each mashine

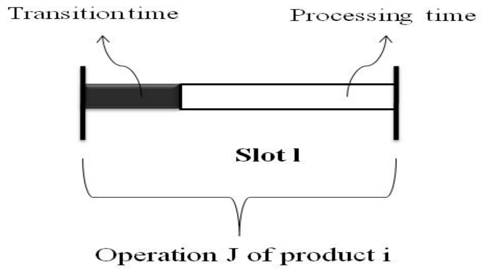

Fig. 2 Assign operation j of product i to slot l

3. Multi-Objective Functions

The objective is to determine a process plans/schedule solution, the process plans for all the involved jobs and a schedule, to achievement a good performance of the predefined criteria regarding process planning and scheduling functions.

From the economic point of view, the process planning takes the minimum manufacturing cost as the objective function.

One of the objective functions is to minimize the transition times (9)

$$C^{p}=\sum_{i}\sum_{k}\sum_{m}\sum_{l}\tau_{i,k}{}^{*}S_{i,p,j,k,f,z,m,l} \tag{9}$$

Another objective function define in terms of minimize the total tardiness that C_i be the completion time of the job i and d_i denote the due date of the job i.

$$\min Tar=\sum_{i}^{n}\max\left(c_i-d_i,0\right) \tag{10}$$

V. Multi-Objective optimization definition

A multi-objective optimization (MOO) problem has a number of conflicting objectives:

Minimize $(f_1(x), f_2(x)\ldots f_3(x))$ (11)

Subject to **X∈S**

The goal in an evolutionary multi-objective optimization is to find a finite number of Pareto-optimal solutions, instead of a single optimum, to the above problem. Some commonly used concepts in the MOO are also introduced.

- Solution comparison: For any two solutions *a* and *b*: *a* dominates *b, if*

$$f_i(a)\le f_i(b)\quad \forall i\in\{1,2,\ldots,k\} \tag{12}$$

$$f_i(b)\le f_i(a)\quad \exists j\in\{1,2,\ldots,k\} \tag{13}$$

And *a* is incomparable with *b, if*

$$f_i(a)\le f_i(b)\ \&\quad f_i(b)\le f_i(a)\quad \forall i\in\{1,2,\ldots,k\} \tag{14}$$

- Pareto optimal solution: a solution *a** is called the Pareto optimal solution if no solution in the decision space *X* can dominate *a**. Pareto optimal set, it is formed by all the Pareto optimal solutions.

VI. Modified-Particle swarm optimization algorithm

PSO simulates a social behavior such as bird flocking to a promising position for certain objectives in a multidimensional space. Like evolutionary algorithm, PSO conducts search using a population (called swarm) of individuals (called particles) that are updated from iteration to iteration. Each particle represents a candidate position (solution) to the problem at hand, resembling the chromosome of GA. the status of a particle is characterized by its position and velocity [22].

Every particle in the swarm moves according to its position and the best particle's position in the virtual search space, just like a bird flying in the sky. Assume that x_i is the position of particle *I,* P_{best} is the best position found by each particle so far. Each particle has its own best position. G_{best} is the best position found by the swarm so far. V_i is the velocity of particle *i* .A particle's movement is based on:

$$V_{I+1}=WV_i+C_1rand1()\left(P_{best}-X_i\right)+C_2rand2()\left(G_{best}-X_i\right) \tag{15}$$

$$X_{i+1}=X_i+V_{I+1} \tag{16}$$

rand1 () and rand2 () are random number between 0 and 1. C_1 is a positive constant, called as coefficient of self recognition component; C_2 is a positive constant, called as coefficient of the social component. The coefficient W is Inertia weight, which increases with inertia decreasing. Constant V_{max} means the largest value of each particle speed and V it is limited to [-V_{max} , V_{max}]. Constant X_{max} means the largest movement distance, and each particles position is limited to [-X_{max} , X_{max}].

A traditional PSO algorithm can be applied to optimize the IPPS in the following steps ():

1. Set the size of a swarm, the number of particles and the max number of iterations.
2. Initialize all the particles in the method introduced in Sections2and3. Decode every particle (solution) to get the schedule of the particle and then calculate the corresponding criteria of particle.
3. Set the local best particle and the global best particle with the best fitness.
4. Iterate the following steps until the max number of iterations is reached.

4.1. For each particle in the swarm, update Particle's velocity and position values.

4.2. Decode the particle into a solution in terms of new position values and calculate the fitness of the particle. Update the local best particle and the global best particle if a lower fitness is achieved.

5. Decode global best particle to get the optimized solution.

However, the traditional PSO algorithm introduced above is still not effective in resolving the operation sequencing problem. To enhance the ability of the traditional PSO algorithm to find the global optimum, new operations, including mutation and shift, have been developed and incorporated in a modified PSO algorithm.

Some modification details are depicted below:

Mutation: In this strategy, a process plan is first randomly selected in …. And an alternative process plan is then randomly chosen to replace the current process plan.

Shift: This operator is used to exchange the positions and velocities of two operations in a particle so as to change their relative positions in the particle.

During the optimization process, if the iteration number of obtaining the same best fitness is more than 10, then the mutation and shift operations are applied to the best particle to try to escape from the local optima.

To apply the PSO algorithm to the process planning optimization problem, two issues have to be handled first: the encoding and decoding scheme How to map a position vector to process planning's index or job' index is the key design in optimizing integrated model of process planning and scheduling with PSO. The encoding and decoding scheme can transform a position in the virtual space into the JSP solution space or reverse.

The process encoding uses a direct sequence [23]. Assume that n is the number of all jobs, a process particle is *n* dimension vector and every dimension weight corresponds to a job. $N_i\{1, \ldots, n_i\}$ is a set of feasible process planning for job *i.* Every dimension weight of process particle is selected form N_i randomly. The corresponding value of the *i* th dimensional weight denotes job *i* select the i th process planning. For example, a process particle is [2,1,1,1,2,2,3,3,3]` the corresponding value of the second dimensional weight is '1', that means job 2 will choose the first process plan from multiple process plans. The process particles' decoding is just like the descriptions in the section above.

Scheduling encoding uses an indirect style. A scheduling Q particle is dimension vector; Q is the total number of all operations for all jobs.

In the sequence of scheduling particle, the TABLE II represents operation index. Different figure indicates different jobs. The same figure will appear in different position, and the position in sequence denotes the operation.

For examples, a scheduling particle is [5,6,1,5,4,6,1,2,4,3,2,4,8,1,3,1,2 ,5,7,1,2,8,7,9,5] ` the 4^{th} figure means the second operation for job 5.

Scheduling particles decoding is relatively complex than process particle decoding. First, decode the weight of the particles to process sequence of jobs. Then insert the current operation according the process sequence's order to the operation permutation in turn. Each machine has its operation permutation. When a new operation will be inserted, we can determine the earliest completion time of the new operation and the machine on which the new operation could be realized. Calculate its starting and completion time. When the machine on which the new operation will be processed is idle during new operation's starting and completion time, insert the new operation into the operation permutation based on the same machine. Repeat the inserting procedure until all operations have been arranged. In the end, find the maximum completion time for each job. The maximum completion time for all jobs is defined as makespan.

VII. Numerical experiment

In our experiment, an imaginary data is used based on example that exhibited in [23], as a job shop that consists of 10 jobs and 10 machines. Every job has multiple feasible process plans. The feasible process plans and other parameter settings are shown (TABLE II) And transition time matrix between each machine added to mentioned example (TABLE III), for simplification we considered transition time is only consist of setup times between pairwise machine in the sequence of the assigned operation to each machine.

Note: in TABLE II the figures in the process plan column state the processing machine index. The figures in the Processing time column and the same row represent the processing time required for corresponsive operations.

For simplification, the parameters of the PSO algorithm recommended in [22] are used in the PSO algorithm for experiments in this paper (Swarm Size are set as 5000, Iter_Num as 200). Learning factor C_1 and C_2 were set to2.

In the final solution, the result of optimization as four non-dominated solutions showed (TABLE IV) that has been reached by the algorithm. As such, they are stored as non-dominated solutions in order to make a comparison. This result is of crucial importance, since it would provide more useful choices for user to make a further decision. If we don't integrate the process planning and scheduling to optimize, the optimization results will be not satisfied.

One of the non dominated solutions of process planning is [1,3,1,2,2,2,3,1,1,2], which means the first job will used the first process plan ,the second job will used the 3rd process plan and so on and the results of relative scheduling planning is[4,1,4,4,5,6,2,9,1,5,2,10,7,8,1,10,6,6,3,1,1,3,2,1,3,4,9,5,5,8,4 ,9,3,5,5,2,7,10,5,1,9,7,8,1], which means with regards to process planning solution and TABLE II, the first slot of machine 6 is operation 1 of job 4, the first slot of machine 9 is operation 1 of job 1, the first slot of machine 5 is operation 2 of job 4, the first slot of machine 2 is operation 3 of job 4, the second slot of machine 2 is operation 1 of job 5 and so on. The total transition time and total tardiness time of this solution is 57 and 18, respectively. These outcomes are very important, since it would give more applicable options for user to make an additional decision and considering the transition time and total tardiness as two objective of IPPS problem is expected to provide satisfactory results for manufacturing systems.

One of the non dominated solutions of process planning is [1,3,1,2,2,2,3,1,1,2], which means the first job will used the first process plan ,the second job will used the 3rd process plan and so on and the results of relative scheduling planning is[4,1,4,4,5,6,2,9,1,5,2,10,7,8,1,10,6,6,3,1,1,3,2,1,3,4,9,5,5,8,4 ,9,3,5,5,2,7,10,5,1,9,7,8,1], which means with regards to process planning solution and TABLE II, the first slot of machine 6 is operation 1 of job 4, the first slot of machine 9 is operation 1 of job 1, the first slot of machine 5 is operation 2 of job 4, the first slot of machine 2 is operation 3 of job 4, the second slot of machine 2 is operation 1 of job 5 and so on. The total transition time and total tardiness time of this solution is

57 and 18, respectively. These outcomes are very important, since it would give more applicable options for user to make an additional decision and considering the transition time and total tardiness as two objective of IPPS problem is expected to provide satisfactory results for manufacturing systems.

TABLE III
TRANSITION TIME MATRIX

	a	b	c	d	E	F	g	h	i	j
a	0	2	1	2	3	4	1	2	2	1
b	2	0	2	3	2	2	3	2	2	2
c	2	2	0	1	1	1	3	2	3	2
d	2	3	1	0	2	2	1	2	1	1
e	3	2	2	2	0	1	2	1	1	1
f	3	2	2	2	2	0	1	1	2	1
g	2	3	1	1	3	3	0	2	2	1
h	2	3	1	1	1	2	1	0	1	2
i	2	2	2	2	1	2	2	2	0	1
j	1	1	3	3	3	4	1	2	1	0

TABLE II
FEASIBLE PROCESS PLANS AND PROCESSING TIME

jobs index	Process id	process plan	processing time
1	**1**	**9-4-3-5-10-7-1-6**	**4-2-2-3-2-3-4-3**
	2	**10-1-2-4-8-7-3-9-5-6**	**3-2-2-2-3-2-3-2-2-2**
	3	**3-1-2-9-7-8-4-6-5**	**3-3-2-2-2-2-3-3-3**
2	**1**	**8-6-9-5-4**	**4-5-5-4-2**
	2	**2-1-4-3-7**	**5-5-3-5-2**
	3	**10-3-5-2**	**6-5-5-4**
3	**1**	**3-4-7-8**	**5-5-4-4**
	2	**9-6-5-4**	**6-4-4-4**
	3	**1-2-10-9**	**3-5-5-5**
4	**1**	**8-7-4-3-9**	**4-5-3-4-4**
	2	**6-5-2-8-7**	**3-5-4-4-4**
	3	**10-6-4-9**	**5-6-4-5**
5	**1**	**3-2-7-8-6-9-10**	**3-3-2-3-2-3-3**
	2	**2-10-9-7-5-4-6**	**4-2-3-3-3-2-2**
	3	**1-9-7-4-5-6**	**4-3-3-3-2-4**
6	**1**	**7-2-5**	**6-6-5**
	2	**6-9-10**	**6-5-6**
	3	**4-7-8**	**5-6-6**
7	**1**	**1-2-9**	**4-7-6**
	2	**3-4-5**	**6-6-5**
	3	**5-7-10**	**5-6-6**
8	**1**	**8-10-2**	**5-6-6**
	2	**5-7-9**	**4-7-6**
	3	**10-6-3**	**5-6-6**
9	**1**	**7-4-9-10**	**5-6-5-5**
	2	**3-5-8-2**	**6-6-5-4**
	3	**1-3-5-7**	**5-5-6-5**
10	**1**	**5-3-7**	**6-6-5**
	2	**4-1-9**	**6-5-6**
	3	**10-5-8**	**5-6-6**

TABLE IV
OPTIMAL PARETO SOLUTION

Pareto optimal solutions	process planning	Scheduling planning	F1	F2
1	3,1,3,2,3,3,2,3,1,1	2,8,5,1,1,3,5,9,9,4,10,3,1,3,1,7,2,4,2,5,9,3,5, 4,7,1,6,2,6,1,4,1,8,2,9,10,10,1,5,7,8,5,1,4,6	62	0
2	1,3,1,2,2,2,3,1,1,2	4,1,4,4,5,6,2,9,1,5,2,10,7,8,1,10,6,6,3,1,1,3, 2,1,3,4,9,5,5,8,4,9,3,5,5,2,7,10,5,1,9,7,8,1	57	18
3	3,3,2,1,1,2,3,1,1,2	5,8,2,10,2,10,4,4,3,5,5,4,9,1,5,7,7,5,5,6,6,1, 10,3,9,1,9,3,1,8,3,1,1,1,4,2,1,8,6,2,9,4,7,5,1	61	16
4	1,2,1,1,3,2,2,1,2,3	9,5,2,2,5,1,6,5,1,3,4,10,8,9,8,4,1,4,1,1,3,9,6, 8,5,10,4,7,10,5,2,7,2,5,4,7,6,1,1,1,3,2,3,9	55	24

VIII. CONCLUSION

In this research, the operation sequencing and the integrated process planning and scheduling problems have been modeled. We have presented a slot based multi objective MILP model. This model is effective to solve for large problems with long time horizons.

The crucial challenges include how to address different performance objectives to meet various practical requirements and how to develop a more effective and intelligent algorithm to identify good solutions in the vast search space of the integrated problem.

Solutions to the operation sequencing and the IPPS problems are encoded into PSO particles to intelligently search for the best sequence of the operations through leveraging the optimization strategies of the PSO algorithm. To explore the search space more effectively, new operators mutation and shift have been developed and incorporated to produce a modified PSO algorithm with improved performance. However, there is still potential for further improvement in computation efficiency and optimality if introducing new operators and characteristics of other algorithms.

REFERENCES

[1] Kim, K.H.; Egbelu, P.J. (1999). Scheduling in a production environment with multiple process plans per job. [J].International Journal of Production Research, Vol. 37, No. 12, pp. 2725–2753.
[2] Zhang, Y.F.; Saravanan, A.N. (2003). Integration of process planning and scheduling by exploring the flexibility of process planning. International Journal of Production Research, Vol. 41, No. 3, pp. 611–628.
[3] Li, WD.; Mcmahon, CA. (2007). A simulated annealing-based optimization approach for integrated process planning and scheduling. Int J Comput Integrated Manuf, Vol. 20, pp. 80–95.
[4] Aldaakhilallah, KA.; Ramesh, R. (1999). Computer-integrated process planning and scheduling(CIPPS): intelligent support for product design. Process planning and control, Int J Production Res, Vol. 37, No. 3, pp.481-500.
[5] Huang, SH.; Zhang, HC.; Smith, ML. (1995). A progressive approach for the integration of process planning and scheduling. IIE Trans, Vol. 1, pp. 456–64.
[6] Kim, YK.; Park, K.; Ko, J. (2003). A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling.Comput Operat Res, Vol. 30, pp. 1151–1171.
[7] Srikanth, K.; Iyera, Barkha Saxenab. (2004). Improved genetic algorithm for the permutation flowshop scheduling problem. Computers & Operations Research, vol. 31, no. 1, pp. 593–606.
[8] Tonshoff, HK.; Beckendorff, U.; Andres, N. (1989). FLEXPLAN: a concept for intelligent process planning and scheduling. In: Proceedings of the CIRP international workshop, Hannover, Germany, pp. 319–322.
[9] Sormaz, D.; Khoshnevis, B. (2003). Generation of alternative process plans in integrated manufacturing systems. J Intel Manuf Vol. 14, pp. 509–526.
[10] Yan, HS.; Xia, QF.; Zhu, MR.; Liu, XL.; Guo, ZM. (2003). Integrated production planning and scheduling on automobile assembly lines. IIE Trans., Vol. 35, pp. 711–25.
[11] Zhang, XD.; Yan, HS. (2005). Integrated optimization of production planning and scheduling for a kind of job-shop. Int J Adv Manuf Technol, Vol. 26, pp. 876–886.
[12] Erdirik-Dogan, Muge; Grossmann, Ignacio, E. (2008). Simultaneous planning and scheduling of single-stage multi-product continuous plants with parallel lines. Computers and Chemical Engineering, Vol. 32. pp. 2664–2683.
[13] Kumar, M.; Rajotia, S. (2003). Integration of scheduling with computer aided process planning. Journal of Materials Processing Technology, Vol. 138, No. 1/3, pp. 297–300.
[14] Moon, C.; Seo, Y. (2005). Evolutionary algorithm for advanced process planning and scheduling in a multiplant. [J]. Computers & Industrial Engineering, Vol. 48, No. 2, pp. 311–325.
[15] Ding, L.; Yue,Y.; Ahmet, K.;Jackson, M.; Parkin, R. (2005). Global optimization of a feature-based process sequence using GA and ANN techniques. Int J Prod Res, Vol. 43, No. 15, pp. 3247-72.
[16] Ma, GH.; Zhang, YF.; Nee, AY. (2000). A simulated annealing-based optimization for process planning. Int J Prod Res, Vol. 38, No. 12, pp. 2671–87.
[17] Lee, DH.; Kiritsis, D.; Xirouchakis, P. (2001). Search heuristics for operation sequencing in process planning. Int J Prod Res, Vol. 39, pp. 3771-88.
[18] Li, Xinyu; Zhang, Chaoyong; Gao, Liang; Li, Weidong; Shao, Xinyu. (2010). An agent-based approach for integrated process planning and scheduling. Expert system with application, Vol. 37, pp. 1256-1264.

[19] Case, K.; Harun, WA. (2000). Wan feature-based representation for manufacturing planning. Int J Prod Res, Vol. 38, No. 17, pp. 4285–300.
[20] Maropoulos, PG.; Baker, RP.; Integration of tool selection with design (part 1. Feature creation and selection of operations and tools). J Mater Process Technol, Vol. 107, pp. 127–134.
[21] French; Simon. (1982). Sequencing and Scheduling: An Introduction to the mathematics of the job-shop. Ellis Horwood Series in Mathematics and Its Applications, ed. G.M. Bell (Chichester: Ellis Hollwood Limited).
[22] Guoa, Y.W.; Lib, W.D.; Milehama, A.R.; Owena, G.W. (2009). Applications of particle swarm optimisation in integrated process planning and scheduling. Robotics and Computer-Integrated Manufacturing, Vol. 25, pp. 280–288.
[23] Zhu, Hengyun; Ye, Wenhua; Bei1, Guangxia. (2009). A Particle Swarm Optimization for Integrated Process Planning and Scheduling. IEEE 10th International Conference on Computer-Aided Industrial Design & Conceptual Design, pp. 1070-1074.