Simultaneous Optimization of Production Planning and Inspection Planning for Flexible Manufacturing Systems

Jian-Hung Chen
Department of Computer Science and Information Engineering
Chung-Hua University
707, Sec. 2, Wu-Fu Road, Hsin-Chu 300, Taiwan
jh.chen@ieee.org

ABSTRACT

Quality assurance in flexible manufacturing systems (FMSs) has become a matter of great importance in recent years. The possibility for offering high-quality products at lower costs has become an essential for a manufacturer to keep in a competitive edge. In this paper, an approach to the multi-objective optimization of production planning and inspection planning in flexible manufacturing systems is presented. A multi-objective memetic algorithm MOMA is proposed to solve the problems having six objectives: minimizing total machining time, machine workload unbalance, greatest machine workload, total tool cost, total inspection time and number of inspections. A schemata-guided local search strategy is proposed for enhancing performances of MOMA. High efficiency of MOMA arises from that multiple objectives can be optimized simultaneously without using heuristics and a set of good non-dominated solutions can be obtained providing additional degrees of freedom for the exploitation of resources of FMSs. Experimental results demonstrate effectiveness of the proposed approach using MOMA for production planning and inspection planning of FMSs.

Categories and Subject Descriptors

H.4.m [**Information Systems Applications**]: Miscellaneous

General Terms

Algorithms, Management, Performance

Keywords

Inspection planning, production planning, flexible manufacturing systems, multi-objective optimization

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'07, July 7–11, 2007, London, England, United Kingdom.
Copyright 2007 ACM 978-1-59593-697-4/07/0007 ...$5.00.

1. INTRODUCTION

A flexible manufacturing system (FMS) is a production system consisting of a set of numerically controlled machines which are connected through an automated guided vehicle (AGV) system [19]. Since FMS is capable of producing a variety of part types and handling flexible routing of parts instead of running parts in a straight line through machines, FMS gives great advantages through its flexibility, such as dealing with machine and tool breakdowns, changes in schedule, product mix, and alternative routes.

Recently, quality assurance has become an important issue in the integration of FMSs, a machining cell can be further integrated with sophisticated inspection equipments (e.g., coordinate measuring machines, CMMs) [1, 8]. Incorporating coordinate measuring machines for automatic inspection of parts produced in FMSs have become an important manufacturing issue. In consequence, the possibility for offering high-quality products at lower costs has become an advantage for a manufacturer to keep in a competitive edge in nowadays. However, it imposes the necessity of decision-making at which process should part inspections be performed in order to minimize the costs associated with quality assurance and the production costs.

While FMS offers many strategic and operational benefits over conventional manufacturing systems, its efficient management requires solutions to complex product planning and inspection planning problems with multiple objectives and constraints. Therefore, decisions regarding multi-objective production and inspection planning problems (MOPIPs) have to be made before the start of actual production, and consist of organizing the limited resource constraints efficiently.

During the past decades, a number of production planning approaches [2, 3, 10, 11, 13, 14, 17, 18, 19] have been developed for automated planning and increased efficiency of production planning approaches. Meanwhile, only a few approaches [12, 16] proposed rule-based scheduling policies to solve inspection planning problems. However, from a system designer's point of view, it is very desirable to obtain a set of deterministic solutions that considering both production planning and inspection planning problems simultaneously.

Multi-objective evolutionary algorithms (MOEAs) have been recognized to be well-suited for solving MOOPs because their abilities to exploit and explore multiple solutions in parallel and to find a widespread set of non-dominated solutions in a single run [5]. Several MOEAs based on Pareto dominance relationship [15] are proposed to solve MOOPs directly, and present more promising results than single-objective optimization techniques theoretically and empirically [5, 20]. By making use of Pareto dominance relationship, MOEAs are capable of performing the fitness assignment of multiple objectives without using relative preferences of multiple objectives. Thus, all the objective functions can be optimized simultaneously. As a result, MOEA seems to be an alternative approach to solving production planning and inspection planning problems on the assumption that no prior domain knowledge is available.

One of the recent growing areas in evolutionary algorithms (EAs) research is memetic agorithms (MAs). MAs are population-based meta-heuristic search methods inspired by Darwinian principles of natural evolution and Dawkins notion of a meme defined as a unit of cultural evolution that is capable of local refinements [4]. From an optimization point of view, MAs are hybrid EAs that combine global and local search by using an EA to perform exploration while the local search method performs exploitation. Combining global and local search is known as an efficient strategy in many successful optimization approaches [9].

In this paper, a novel approach using a multi-objective memetic algorithm MOMA is proposed to solve multi-objective production and inspection planning problems (MOPIPs) having six objectives: minimizing total machining time, machine workload unbalance, greatest machine workload, total tool cost, total inspection time and number of inspections. A schemata-guided local search strategy is also proposed for enhancing performances of MOMA. The proposed approach can obtain a set of non-dominated solutions for decision makers in a single run. Decision makers can easily distinguish between the costs of different production plans and choose more than one satisfactory production plans which may offers high-quality products at lower costs at a time. Six benchmark problems with different complexities are derived to evaluate the performance of the proposed approach. It is shown empirically that MOMA can converge to better solutions than traditional MOGA in solving MOPIPs.

This paper is organized as follows: Section 2 describes the investigated MOPIPs. Section 3 presents the proposed memetic multi-objective genetic algorithm MOMA for solving investigated problems. Section 4 gives the experimental results and analysis of the proposed algorithm. Section 5 summarizes our conclusions.

2. PROBLEM STATEMENT

The aim of production planning and inspection planning is to develop a cost-effective and operative inspection and production plan over planning phases. In this section, a multi-objective production and inspection planning problem (MOPIPs) is established. With the assignments of inspection and assignments of operations to machines, six optimization objectives: minimizing total machining time, machine workload unbalance, greatest machine workload, total tool cost, total inspection time and number of inspections, are considered in our problems.

2.1 The FMS environment

An FMS consists of a set of identical and/or complementary numerically controlled machines, coordinate measuring machines and tool systems. All components are connected through an automatic guided vehicle (AGV) system. Fig. 1 shows the layout of a simple FMS with several machines, AGVs and a tool system.

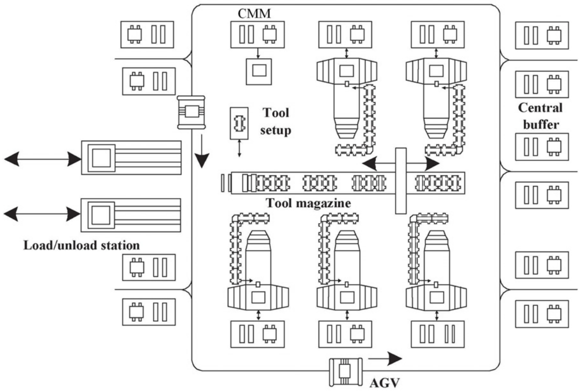

Figure 1: FMS with several machines, a coordinate measuring machine (CMM), AGVs and a central tool magazine.

In order to design production plans and inspection plans of FMSs, the environment within which the FMS under consideration operates can be described below.

1. The investigated FMS consists of several identical and/or complementary manufacturing machines and a coordinate measuring machine. The types and number of manufacturing machines are known. There is a sufficient input/output buffer space at each machine.

2. A *part* type requires a number of *operations.* A number of part types will be manufactured simultaneously in batches. Parts can choose one or more machines at each of their operation stages, and the transportation of the parts within different machines is handled by an AGV system.

3. A machine can perform several types of operations, and an operation can be performed on alternative machines.

4. A machine can only process an operation at one time. Operations to be performed in the machine are non-preemptive. Operation lot splitting is ignored in this paper.

5. A *production plan* consists of part indices, operation indices, and a series of machine indices corresponding to operations of all parts. Based on a production plan, each operation is operated on its corresponding machine. An illustrative production plan of 3 parts and 10 operations is presented in Fig. 2, and the operations are operated on 3 different machines. An example of the series of machine indices to be optimized is Y=[1 1 1 3 1 2 2 2 3 3].

6. A *inspection plan* consists of part indices, operation indices, and a series of binary indices corresponding operations of all parts. Based on a inspection plan, if an index is true, a part will be transport to coordinate measuring machine for inspection after the corresponding operation is finished. An example of a series of binary indices to be optimized is Z=[0 1 0 0 1 0 0 1 0 1].

7. *P*rocessing times of operations in machines are available and deterministic. The tool costs of operations in machines are known.

8. *I*nspection time is a ratio of operation processing time. A ratio of 0.05 means that the mean inspection operation time is 5% of the sum of all previously uninspected machining operation times. In this paper, a ratio of 0.05 is assumed.

9. Workload on each machine is contributed by those operations assigned to a machine.

10. A load/unload (L/U) station serves as a distribution center for parts not yet processed and as a collection center for parts finished. All vehicles start from the L/U station initially and return to there after accomplishing all their assignments. There are sufficient input/output buffer spaces at the L/U station.

11. The number of AGVs is given and the transportation time among different machines are known. Some machines may not be linked.

12. AGVs carry a limited number of products at a time. They move along predetermined paths, with the assumption of no delay because of congestion. Preemption of trips is not allowed.

13. It is assumed that all the design, layout and set-up issues within FMS have already been resolved.

14. Real-time issues, such as traffic control, congestion, machine failure or downtime, scraps, rework, and vehicle dispatches for battery changer are ignored here and left as issues to be considered during real-time control.

Part index	1	2	3
Operation index	1 2 3 4	1 2 3	1 2 3
Machine index	1 1 1 3	1 2 2	2 3 3
Inspection index	0 1 0 0	1 0 0	1 0 1

Figure 2: A inspection plan and production plan of 3 parts and 10 operations, operated on 3 different machines. For example, the operation 2 of the part 1 is assigned to the machine 1, and the part 1 will be transport for inspection after the operation 2 of the part 1.

2.2 Mathematical formulation of MOPIPs

2.2.1 Notations

In order to formulate MOPIPs, the following notations are introduced:

- i : part index, $i = 1, 2, 3, ..., I$.
- j : operation index for part i, $j = 1, 2, 3, ..., J_i$.
- k, l : machine index $k, l = 1, 2, 3, ..., K$.
- pv_i : production volume (unit) for part i.
- c_{ijk} : tool costs to perform operation j of part i using machine k.
- pt_{ijk} : processing time per unit to perform operation j of part i using machine k.
- tm_{kl} : transportation time from machine k to l. If machines k and l are not linked, it is set to be a negative value for constraint handling.
- abl : available capacity of AGV per trip, abl is set to 10 in this paper.
- $MaxWL_k$: maximum workload of machine k.
- $MaxIP$: maximum number of inspection.
- wl_k : workload in machine k, $WL_k = pt_{ijk} \times pv_i$.
- RWL_k : workload ratio in machine k, $RWL_k = \frac{wl_k}{MaxWL_k}$.
- $AvgRWL$: average workload ratio of machines.
- y_{ij} : a machine index, if machine k is selected to perform operation j of part i, then $y_{ij} = k$. $y_{ij} \in [1, k]$.
- z_{ij} : $\begin{cases} 1, & \text{if operation } j \text{ of part } i \text{ is selected for inspection;} \\ 0, & \text{otherwise.} \end{cases}$
- b_{ij} : $\begin{cases} 1, & \text{if there exists an inspection in or after the operation } j \text{ of part } i; \\ 0, & \text{otherwise.} \end{cases}$
- s_{ikl} : $\begin{cases} 1, & \text{if part } i \text{ is to transfer from machine } k \text{ to } l; \\ 0, & \text{otherwise.} \end{cases}$
- x_{ijk} : $\begin{cases} 1, & \text{if machine } k \text{ is selected to perform operation } j \text{ of part } i; \\ 0, & \text{otherwise.} \end{cases}$
- n_{ikl} : the number of trips between machine k and l for part i,
$$n_{ikl} = s_{ikl} \times \lceil \frac{pv_i}{abl} \rceil,$$
where the bracket represents a ceiling operation.
- t_{ikl} : total transportation time between machines k and l for part i,
$$t_{ikl} = n_{ikl} \times tm_{kl}.$$
- Y : a series of machine indices y_{ij} corresponding to operations of all parts in a production plan.
- Z : a series of binary indices z_{ij} corresponding to operations of all parts in a inspection plan.

2.2.2 *Objectives*

There are six objectives to be optimized in FMSs, described below.

1. Minimization of total machining time. This objective is to minimize the processing time and transportation time for producing the parts. The total machine processing time (f_1) is defined as Equation 1, the transportation time (f_2) is defined as Equation 2, and the total flow time (F_1) is defined as Equation 3. Transportation between unlinked machines are penalized in f_2.

$$f_1 = \sum_{i=1}^{I} \sum_{j=1}^{J_i} \sum_{k=1}^{K} pv_i \times pt_{ijk} \times x_{ijk}, \quad (1)$$

$$f_2 = \sum_{i=1}^{I} \sum_{j=1}^{J_i-1} \sum_{k=1}^{K} \sum_{l=1}^{K} t_{ikl} \times x_{ijk} \times x_{i(j+1)l}, \quad (2)$$

$$F_1 = f_1 + f_2. \quad (3)$$

2. Minimization of machine workload unbalance. Balancing the machine workload can avoid creating bottleneck machines. The objective function (F_2) is defined as Equation 4.

$$F_2 = \sum_{k=1}^{K} (RWL_k - AvgRWL)^2. \quad (4)$$

3. Minimization of greatest machine workload. Pursuing this objective also implies attempting to minimize the total flow time. The objective function (F_3) is defined as Equation 5.

$$F_3 = max(RWL_k). \quad (5)$$

4. Minimization of total tool cost. Tool costs consider the consumptions of tools, tool life issues, tool expenses and the number of tool copies. The objective function (F_4) is defined as Equation 6.

$$F_4 = \sum_{i=1}^{I} \sum_{j=1}^{J_i} \sum_{k=1}^{K} c_{ijk} \times x_{ijk}. \quad (6)$$

5. Minimization of total inspection time. This objective is to minimize the inspection time of parts. The inspection time is $p\%$ of the sum of all previously uninspected machining operation times. Although this objective is proportion of f_1 in Equation 2, it should be considered independently. Otherwise, if the inspection time is integrated in F_1, it would be dominated by f_1 and f_2. In consequence, the results will be biased in optimizing F_1. The objective function (F_5) is defined as Equation 7.

$$F_5 = \sum_{i=1}^{I} \sum_{j=1}^{J_i} \sum_{k=1}^{K} \frac{p}{100} \times pv_i \times pt_{ijk} \times x_{ijk} \times b_{ij}, \quad (7)$$

6. Minimization of number of inspections. The number of inspections consider the consumptions of inspection machines and their related expenses. Decision makers usually expect that the number of inspections is smaller than a maximum number of inspection $MaxIP$. The objective function (F_6) is defined as Equation 8.

$$F_6 = \sum_{i=1}^{I} \sum_{j=1}^{J_i} z_{ij}. \quad (8)$$

2.2.3 *Multi-objective Mathematical Model*

The overall multi-objective mathematical model of MOPIPs can be formulated as follows. Given the production volume pv_i, the processing time pt_{ijk}, the maximum workload $MaxWL_k$, the maximum number of inspection $MaxIP$, the available capacity of AGV per trip abl, the transportation time tm_{kl} and the tool costs c_{ijk}, find a series of machine indices Y and a series of binary indices Z for operations and inspection of all parts, such that

$$minimize \quad F_1, F_2, F_3, F_4, F_5, F_6 \quad (9)$$

subject to

$$\sum_{k=1}^{K} x_{ijk} = 1, \ \ \forall(i,j), \quad (10)$$

$$tm_{kl} \geq 0, \ \ \forall(k,l), \quad (11)$$

$$RWL_k \leq 1, \ \ \forall i. \quad (12)$$

$$F_6 \leq MaxIP. \quad (13)$$

The constraint, Equation 10, ensures that only one machine is selected for each operation of a part. Equation 11 ensures an AGV path exists between machines k and l. Equation 12 is to ensure the machine workload wl_k is smaller or equal to its maximum machine workload $MaxWL_k$. Equation 13 is to ensure the number of inspections F_6 is smaller or equal to the maximum number of inspections $MaxIP$.

If the total number of machines is m and the total number of operations is n, then the complexity of the investigated problem is $O(m^n \cdot 2^n)$.

3. MULTI-OBJECTIVE MEMETIC ALGORITHM MOMA

MOMA differs from MOGA mainly in its local search strategy. A summary of Pareto dominance relationship and the fitness assignment strategy for handling multiple objective functions is described in Section 3.1. The proposed schemata-guided local search strategy is presented in Section 3.2. MOMA for solving MOPIPs is presented in Section 3.3, including the representation of chromosomes, genetic operators, constraint handling, and the procedure of MOMA.

3.1 Fitness assignment strategy

Many MOEAs differ mainly in the fitness assignment strategy, which is known as an important issue in solving MOOPs [5]. MOMA uses a generalized Pareto-based scale-independent fitness function GPSIFF considering the quantitative fitness values in Pareto space for both dominated and nondominated individuals. GPSIFF makes the best use of Pareto dominance relationship to evaluate individuals using a single measure of performance.

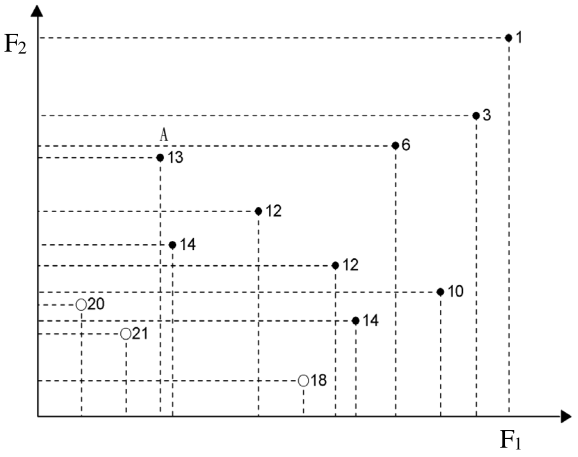

Figure 3: The circles represent non-dominated solutions and the black dots are dominated solutions. The fitness values are calculated by GPSIFF.

3.1.1 Pareto dominance relationship

Assume all the objective functions F_m are to be minimized. Mathematically, MOOPs can be represented as the following vector mathematical programming problems:

$$Minimize \quad F(X) = \{F_1(X), F_2(X), ..., F_m(X)\}, \qquad (14)$$

where X denotes a solution and $F_m(X)$ is generally a nonlinear objective function. When the following inequalities hold between two solutions X_1 and X_2, X_2 is a *non-dominated solution* and is said to *dominate* $X_1 (X_2 \succ X_1)$:

$$\forall m : F_m(X_1) \geq F_m(X_2) \ and \ \exists n : F_n(X_1) > F_n(X_2). \ (15)$$

When the following inequality hold between two solutions X_1 and X_2, X_2 is said to *weakly dominate* $X_1 (X_2 \succeq X_1)$:

$$\forall m : F_m(X_1) \geq F_m(X_2). \qquad (16)$$

A feasible solution X^* is said to be a *Pareto-optimal solution* if and only if there does not exist a feasible solution X where X dominates X^*. The corresponding vector of Pareto-optimal solutions is called *Pareto-optimal front.* An example in a bicriteria space is shown in Fig. 3, where the circles represent non-dominated solutions and the black dots are dominated solutions.

3.1.2 GPSIFF

The used GPSIFF is described below. Let the fitness value of an individual X be a tournament-like score obtained from all participant individuals by the following function:

$$F(X) = p - q + c, \qquad (17)$$

where p is the number of individuals which can be dominated by the individual X, and q is the number of individuals which can dominate the individual X in the objective space. Generally, a constant c can be optionally added in the fitness function to make fitness values positive. In this paper, c is the number of all participant individuals. GPSIFF can assign discriminative fitness values to not only non-dominated individuals but also dominated ones. Fig. 3 illustrate an example of fitness values of 12 participant individuals for a bicriteria optimization problem (c=12). For example, considering the individual A with a fitness value 13, in the rectangle formed by A, two individuals dominates A (q=2) and three individuals is dominated by A (p=3). Therefore, the fitness value of A is 3-2+12=13.

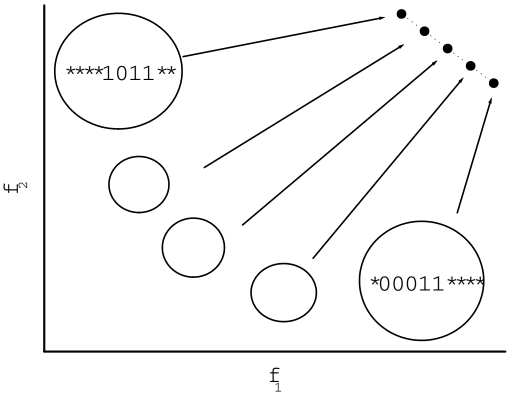

Figure 4: The population were divided into several species, and each species optimizes its own niche (Pareto-optimal solution).

3.2 Schemata-Guided Local Search Strategy

Based on schema theorem and the niche hypothesis [6, 7], a schemata-guided local search strategy is proposed to be combined with MOGA for improving the convergence speed to the Pareto-front. Extended from the niche hypothesis, it is assumed that, given a MOOP with Q Pareto-optimal solutions, Q Pareto-optimal solutions can be regarded as Q niches of the MOOP. In the worst case, to ensure MOEAs is capable of searching Q Pareto-optimal solutions, it is assumed that the population were divided into Q species (subpopulations). Thus, each species is expect to optimize its own niche (Pareto-optimal solution), as shown in Figure 4. Therefore, the optimal schemata of a species is its Pareto-optimal solution.

Let the schema of species be H_q, where the fixed positions are the maximum common string of all individuals in its species and the others are "don't care"(*). Since species are in the same population, a schemata of a species may be disrupted by schemata of the other species due to genetic operators. The disruption between species can be further classified into the following two types:

1. **Species disrupt noise**: The fixed schemata of H_{origin} are altered to "don't care" schemata by the corresponding positions of the schemata H_{other}. Thus, a species requires more time for fixing it's "don't care" schemata.

2. **Species hitchhiking noise**: The "don't care" schemata of H_{origin} are altered to fixed schema by the corresponding positions of the schema H_{other}. If the altered schemata are located in the similarity regions of their optimal schemata, the change is good for the schemata H_{origin}. On the contrary, the change is bad for the schemata H_{origin}.

Based on the foregoing inference, it is desired that a species should keep its good schemata (building blocks) while making good efforts to alter its "don't care" schemata to its ideal optimal schemata. As results, a schemata-guided local search strategy is proposed based on this guideline. Information of fixed and "don't care" schemata in species are utilized to guide local search. However, the key question of this local search strategy is that how do we classify population to different species when true Pareto-optimal solutions of MOOPs are unknown. To deal with this question, it is assumed that the best individuals in each objective functions are the *pioneers* of each species. These pioneers will be used to classify all individuals in population to different species.

Given a maximum local search times $MaxLS$ and a temporary elite set E', the procedure of the proposed schemata-guided local search strategy is written as follows:

Step 1: Identification Identify the best individuals $B_q, q = 1, 2, ..., Q$, in each objective from the current population. For MOPIPs, Q=6.

Step 2: Classification Classify the current population into Q species by the best solutions in each objective.

Step 3: Compute Schemata For each species, compute its schemata H_q. Both fixed and "don't care" schemata are identified.

Step 4: Parameter setting Let $q = 1, counter = 0$.

Step 5: Perturbation Perturb B_q into a new solution B'_q. According to H_q, apply the mutation operator only on "don't care" locations of B_q with a mutation probability p_m.

Step 6: Evaluation Evaluate the objective functions of B'_q. Let $counter = counter + 1$.

Step 7: Comparison There is 3 cases in comparisons of B_q and B'_q. **Case 1**: If B_q dominates B'_q and $counter < MaxLS$, go to Step 5. **Case 2**: If B_q is dominated by B'_q, replace B_q by B'_q. **Case 3**: If B_q and B'_q doesn't dominated each other. Stored B'_q in a temporary elite set E'.

Step 8: Termination test Let $q = q + 1$ and counter=0, if q¿Q, stop the local search strategy. Otherwise, go to Step 5.

3.3 MOMA for solving MOPPP

3.3.1 Chromosome representation

A series of machine indices Y and a series of binary indices Z for operations of all parts is directly encoded as a integer chromosome. For the first series of indices Y, the range of each gene of Y is [1, K]. Each gene of Y stands for a machine index. For the second series of indices Z, each gene of Z is either 1 or 0, which represent an inspection of this manufacturing part is to be conducted or not.

3.3.2 Genetic Operators

The genetic operators used in the proposed approach are widely used in literature. The selection operator of MOMA uses a binary tournament selection which works as follows. Choose two individuals randomly from the population and copy the better individual into the intermediate population.

Crossover is a recombination process in which genes from two selected parents are recombined to generate offspring chromosomes. The two-points crossover is used in MOMA. In a two-points crossover operation, the first cutting point is selected randomly among Y, and the second cutting point is selected randomly among Z. The genes on the sides of the cutting point are exchanged between the parent chromosomes. A crossover operation is illustrated in Fig. 5.

individual	genes
P_1	1 1 1 3 1 2 3 2 3 3 0 1 0 0 1 0 0 1 0 1
P_2	2 2 3 1 1 2 2 1 2 2 0 0 0 1 1 0 1 1 0 1
C_1	$\underline{2\ 2\ 3\ 1\ 1}$ 2 3 2 3 3 $\underline{0\ 0\ 0\ 1\ 1\ 0}$ 0 1 0 1
C_2	$\underline{1\ 1\ 1\ 3\ 1}$ 2 2 1 2 2 $\underline{0\ 1\ 0\ 0\ 1\ 0}$ 1 1 0 1

Figure 5: An illustrated crossover operation. Assuming the generated cutting points are 5 and 16, P_1 and P_2 are parents, then C_1 and C_2 are the generated offsprings.

A simple mutation operator is used to alter genes. For each gene, randomly generate a real value from the range [0, 1]. If the value is smaller than the mutation probability p_m, replace its index with a randomly generated integer among its possible values.

3.3.3 Constraint Handling

Based on the proposed chromosome representation, Equation 10 is always satisfied. If Equation 11 is violated, the transportation time between machines k and l, tm_{kl}, is set to be a large value, 10^7. In this way, f_2 will be penalized. For each machine k, if Equation 12 is not satisfied, one is added to RWL_k, as follows:

$$RWL_k = \begin{cases} \frac{wl_k}{MaxWL_k}, & \text{if } wl_k \leq MaxWL_k; \\ \frac{wl_k}{MaxWL_k} + 1, & \text{otherwise.} \end{cases} \quad (18)$$

If Equation 13 is not satisfied, F_6 is penalized with a value $MaxIP$.

3.3.4 Procedure of MOMA

Since it has been recognized that the incorporation of elitism may be useful in maintaining diversity and improving the performance of multi-objective EAs [5], MOMA selects a number of elitists from an elite set E in the selection step. The elite set E with capacity E_{max} maintains the best non-dominated solutions generated so far. In addition, an external set $\overline{E}$ with no capacity is used to store all the non-dominated solutions ever generated so far. The procedure of MOMA is written as follows:

Step 1: Initialization Randomly generate an initial population of N_{pop} individuals and create two empty elite sets E, $\overline{E}$ and an empty temporary elite set E'.

Step 2: Evaluation For each individual Y in the population, compute $F_1(Y)$, $F_2(Y)$, $F_3(Y)$, $F_4(Y)$, $F_5(Y)$, and $F_6(Y)$.

Step 3: Fitness assignment Assign each individual a fitness value by using GPSIFF.

Step 4: Local Search Apply the proposed schemata-guided local search strategy. Non-dominated solutions obtained by the local search strategy will be stored in temporary elite set E'.

Step 5: Update elite sets Add the non-dominated individuals in both the population and E' to E, and empty

Table 1: The parameter settings of MOMA and MOGA.

Parameters	Value
N_{pop}	100
E_{max}	100
p_s	0.25
p_c	0.6
p_m	0.05
$MaxLS$	3

E'. Considering all individuals in E, remove the dominated ones in E. Add E to $\overline{E}$, remove the dominated ones in $\overline{E}$. If the number of non-dominated individuals in E is larger than E_{max}, randomly discard excess individuals.

Step 6: Selection Select $N_{pop} - N_{ps}$ individuals from the population using the binary tournament selection and randomly select N_{ps} individuals from E to form a new population, where $N_{ps} = N_{pop} \times p_s$ and p_s is a selection proportion. If N_{ps} is greater than the number N_E of individuals in E, let $N_{ps} = N_E$.

Step 7: Recombination Perform the two-points crossover operation with a recombination probability p_c.

Step 8: Mutation Apply the mutation operator to each gene in the individuals with a mutation probability p_m.

Step 9: Termination test If a stopping condition is satisfied, stop the algorithm and output $\overline{E}$. Otherwise, go to Step 2.

4. RESULTS AND DISCUSSION

Considering the real manufacturing environment, we derived the AGV transportation time matrix and six benchmark problems: *m3o10*, *m4o20*, *m5o100*, *m5o200*, *m10o100* and *m10o200*, where *mxoy* stands for the x machine and y operation problem. In order to further investigate the performance of MOMA, a MOGA (MOMA without the local search strategy) is also implemented to solve MOPIPs. The solutions obtained by MOGA are used as the baseline performance for comparisons. The parameter settings of MOMA and MOGA are given in Table 1. All the parameters of MOMA and MOGA in each experiment are the same. Thirty independent runs were performed per test problems, compared with the same number of function evaluations $100xy$.

The coverage metric $C(A, B)$ of two solution sets A and B [20] used to compare the performance of two corresponding algorithms considering the six objectives:

$$C(A, B) = \frac{|\{a \in A, b \in B, a \succeq b\}|}{|B|}, \tag{19}$$

where $\succeq$ stands for weakly dominate in Pareto dominance relationship. The value $C(A, B) = 1$ means that all individuals in B are weakly dominated by A. On the contrary, $C(A, B) = 0$ denotes that none of individuals in B is weakly dominated by A. Because the C measure considers the weakly dominance relationship between two sets A and B, $C(A, B)$ is not necessarily equal to $1 - C(B, A)$. The comparison results of two solution sets using the coverage metric are depicted using box plots. A box plot provides an excellent visual result of a distribution. The box stretches from the lower hinge (defined as the 25th percentile) to the upper hinge (the 75th percentile) and therefore contains the middle half of the scores in the distribution. The median is shown as a line across the box.

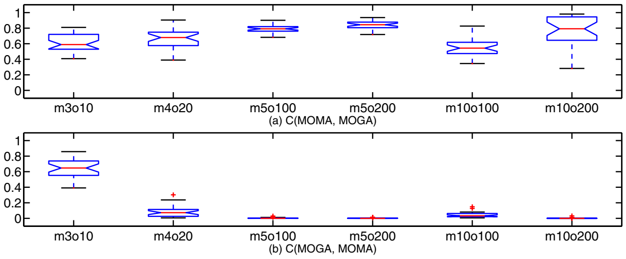

Figure 6: Box plots based on the cover metric. (a) C(MOMA, MOGA), (b) C(MOGA, MOMA).

For each run, the solutions set of two algorithms are compared using the coverage metric. Fig. 6 depicts the coverage metrics of $C(MOMA, MOGA)$ and $C(MOGA, MOMA)$ from 30 runs. In solving the small problem *m3o10*, Fig. 6 shows that the performance of MOMA is slightly better than MOGA. However, as the complexity of problems increases, Fig. 6 shows that 50%-90% of the non-dominated solutions obtained by MOGA are weakly dominated by the non-dominated solutions obtained by MOMA in solving the problems *m4o20*, *m5o100*, *m5o200*, *m10o100* and *m10o200*. On the contrast, the non-dominated solutions of MOGA dominate nearly 3-10% of the non-dominated solutions obtained by MOMA. The results indicate that MOMA can converge to better solutions more quickly than MOGA. It reveals that the proposed schemata-guided local search strategy plays an important role in MOMA.

For practitioners of FMSs, by taking advantage of this multi-objective approach, a satisfactory production plan can be fast obtained by given relative preferences from practitioners, and practitioners can also choose several alternative production plans at a time. These additional degrees of freedom could provide a better exploitation of the resources of FMSs. On the contrary, single-objective approaches have to perform multiple runs in order to obtain a set of non-dominated solutions.

The complexity of the investigated problem is determined by the numbers of operations and machines. If the complexity increases, a large computation time may be necessary to solve a large-scale problem. In practical, MOMA can utilize some specific heuristic rules (smallest processing time, smallest tool cost, smallest inspection time etc.) such that the results may be better.

5. CONCLUSIONS

In this paper, a novel approach to production planning of flexible manufacturing systems (FMSs) using an multi-objective memetic algorithm MOMA is proposed. A schemata-guided local search strategy is integrated with MOMA for enhancing the performance of MOMA. The investigated multi-objective production and inspection planning problem (MOPIP) has six objectives: minimizing total machining time, machine workload unbalance, greatest machine workload, total tool cost, total inspection time and number of inspections.

The advantages of the proposed approach are that MOMA can optimized multiple objectives without decomposing problems into sub-problems, and MOMA makes use of Pareto dominance relationship to solve problems without using relative preferences of multiple objectives. While prior domain knowledge for the decomposition of problems or relative preferences of multiple objectives are not available, the proposed approach is an expedient method to solve production planning and inspection planning of FMSs, compared with the decomposition and preference-based approaches in literature. In addition, the proposed approach can obtain a set of non-dominated solutions for decision makers in a single run. Decision makers can easily distinguish between the costs of different production plans and choose more than one satisfactory production plans at a time. These additional degrees of freedom could provide a better exploitation of the resources of FMSs.

Experimental results demonstrated that the quality of non-dominated solutions obtained by MOMA is better than that of MOGA in terms of convergence speed and accuracy using the same number of function evaluations. The results indicate that the proposed approach is a generalized and efficient approach to solving MOPIPs. Moreover, the results also show that the proposed schemata-guided local search could be good bonus for MOMA. Possible future work of this paper can be

1. the number of pioneers for classifying species;
2. selection of pioneers for classifying species;
3. determine potential "don't care" locations for local search;
4. use information of "fixed" schemata of a species to alter "don't care" location of another species;
5. performance of the proposed schemata-guided; local search in different evolution phases.

6. ACKNOWLEDGMENTS

We thank the anonymous reviewers for their comments. This work was supported by the National Science Council of Taiwan, R.O.C., under Contract NSC-95-2221-E-216-036.

7. REFERENCES

[1] J. A. Bosch, editor. *Coordinate Measuring Machines and Systems*. Marcel Dekker Inc., New York, 1995.

[2] J.-H. Chen and S.-Y. Ho. A novel approach to production planning of flexible manufacturing systems using an efficient multi-objective genetic algorithm. *International Journal of Machine Tools and Manufacture*, 45(7-8):949–957, June 2005.

[3] Y.-J. Chen and R. G. Askin. A multiobjective evaluation of flexible manufacturing system loading heuristics. *International Journal of Production Research*, 28(5):895–911, 1990.

[4] R. Dawkins. *The selfish Gene*. Oxford University Press, 1976.

[5] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley-Interscience series in systems and optimization. John Wiley & Sons, 2001.

[6] D. E. Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley Pub. Co., 1989.

[7] J. H. Holland. *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor, MI, 1975. ISBN: 0-262-58111-6.

[8] M. Innala and S. Torvinen. Integrating computer aided quality assurance in flexible engineering and production processes. *International Journal of Production Economics*, 41(1):179–184, 1995.

[9] H. Ishibuchi, T. Yoshida, and T. Murata. Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling. *IEEE Trans. on Evolutionary Computation*, 7(2):204–223, 2003.

[10] P. Kumar, N. K. Tewari, and N. Singh. Joint consideration of grouping and loading problems in a flexible manufacturing system. *International Journal of Production Research*, 28(7):1345–1356, 1990.

[11] C. S. Lee, S. S. Kim, and J. S. Choi. Operation sequence and tool selection in flexible manufacturing systems under dynamic tool allocation. *Computers & Industrial Engineering*, 45(1):61–73, 2003.

[12] J. Lee and C. F. F. Inspection sequencing and part scheduling for flexible manufacturing systems. *European Journal of Operational Research*, (65):344–355, 1996.

[13] M. Liang. Integrating machine speed, part selection and machine loading decision in FMSs. *Computer & Industrial Engineering*, 26(3):599–608, 1994.

[14] M. Liang and P. S. Dutta. A mixed-integer programming approach to the machine loading and process planning problem in a process layout environment. *International Journal of Production Research*, 28(8):1471–1484, 1990.

[15] V. Pareto. *Cours d'economie politique professe a l'universite de Lausanne*, volume 1. F. Rouge, Lausanne, 1896.

[16] M. Siemiatkowski and W. Przybylski. Simulation studies of process flow with in-line part inspection in machining cells. *Journal of Materials Processing Technology*, 171(1):27–34, 2006.

[17] M. S. Sodhi, R. G. Askin, and S. Sen. Multiperiod tool and production assignment in flexible manufacturing systems. *International Journal of Production Research*, 32(6):1281–1294, 1994.

[18] R. Swarnkar and M. K. Tiwari. Modeling machine loading problem of FMSs and its solution methodology using a hybrid tabu search and simulated annealing-based heuristic approach. *Robotics and Computer-Integrated Manufacturing*, 20(3):199–209, 2004.

[19] H. Tempelmeier and H. Kuhn. *Flexible manufacturing systems: decision support for design and operation*. Wiley series in systems engineering. Wiley, New York, 1993.

[20] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strengthen Pareto approach. *IEEE Transaction on Evolutionary Computation*, 4(3):257–271, 1999.