Published in final edited form as:
J Intell Manuf. 2011 August 1; 22(4): 643–652. doi:10.1007/s10845-010-0382-7.

Optimization process planning using hybrid genetic algorithm and intelligent search for job shop machining

Mojtaba Salehi and **Ardeshir Bahreininejad**
Faculty of Engineering, Tarbiat Modares University, 14115 Tehran, Iran

Mojtaba Salehi: moj.salehi@modares.ac.ir; Ardeshir Bahreininejad: bahreininejad@modares.ac.ir

Abstract

Optimization of process planning is considered as the key technology for computer-aided process planning which is a rather complex and difficult procedure. A good process plan of a part is built up based on two elements: (1) the optimized sequence of the operations of the part; and (2) the optimized selection of the machine, cutting tool and Tool Access Direction (TAD) for each operation. In the present work, the process planning is divided into preliminary planning, and secondary/detailed planning. In the preliminary stage, based on the analysis of order and clustering constraints as a compulsive constraint aggregation in operation sequencing and using an intelligent searching strategy, the feasible sequences are generated. Then, in the detailed planning stage, using the genetic algorithm which prunes the initial feasible sequences, the optimized operation sequence and the optimized selection of the machine, cutting tool and TAD for each operation based on optimization constraints as an additive constraint aggregation are obtained. The main contribution of this work is the optimization of sequence of the operations of the part, and optimization of machine selection, cutting tool and TAD for each operation using the intelligent search and genetic algorithm simultaneously.

Keywords

Genetic algorithm; Intelligent search; Computer-aided process planning; Preliminary planning; Detailed planning; Operations sequencing

Introduction

Computer-aided process planning (CAPP) is an important interface between computer-aided design (CAD) and computer-aided manufacturing (CAM) in the computer integrated manufacturing (CIM) environment. The process planning activity includes interpretation of design data, selection and sequencing of operations to manufacture the part, selection of machines and cutting tools, determination of cutting parameters, choice of jigs and fixtures and calculation of machining times and costs. To clarify, the process planning, parts are represented by manufacturing features. Figure 1 shows a part composed of m features in which 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 (Case and Harun Wan 2000;Maropoulos and Baker 2000).

One of the most important tasks in the process planning is the operation sequencing that is concerned with the selection of machining operations in steps that can produce each form

Correspondence to: Ardeshir Bahreininejad, bahreininejad@modares.ac.ir.

feature of the part by satisfying relevant technological constraints specified in the part drawing, while minimizing the number of setups, maximizing the machines utilization, minimizing the number of tool changes, and so forth. In other words, the operation sequencing problem in the process planning is considered to produce a part with the objective of minimizing the sum of machines, setup and tool change costs. A good process plan for a part is built up based on two elements: (1) the optimized sequence of the operations of the part and (2) the optimized selection of the machine, cutting tool and TAD for each operation. Although many CAPP systems have been reported in the literature, only few have considered the optimization of the sequence of operations, and suggested alternative sequence of operations or process plans.

The basic input to any CAPP system is the part description. Usually the geometrical (shape, dimensions, etc.) and technological information (tolerances, surface finish, etc.) are inputted by a CAD modeler. Sometime this information is inputted by a user interface. Based on process capability, appropriate operations are selected for producing the features. Then, by considering constraints and relationship between operations, optimal/near optimal sequences are generated. Many search methods were developed for generating optimal sequence. After determining the optimal sequence, this sequence transmits for manufacturing by computer.

Operation sequencing is a complex task exhibiting combinatorial nature. As the operations sequencing problem involves various interdependent constraints, it is rather difficult to formulate and solve this problem using integer programming and dynamic programming methods alone. The operation sequencing problems may usually be modeled as large-scale and combinational optimization problems (Qiao et al. 2000). Integer programming, Genetic algorithms (GA), search heuristics, hybrid genetic algorithms, and simulated annealing approaches have been applied to operation sequencing.

Reddy et al. (1999) applied GA as a global search technique for a quick identification of optimal or near-optimal operation sequences in a dynamic planning environment. Since sequences may be obtained quickly, this algorithm can actually be used by the process planner to generate alternative feasible sequences for the prevailing operating environment. Qiao et al. (2000) used a GA-based approach for machining operation sequencing for prismatic parts. Four types of process planning rules including precedence rules, clustering rules, adjacent order rules and optimization rules were considered and encompassed quantitatively in the fitness calculations for alternative operation sequences. The proposed GA operates effectively by incorporating various production environment considerations into process planning. Lee et al. (2001) developed six local search heuristics based on Simulated Annealing (SA) and Tabu Search (TS) to obtain good operation sequences for practical-sized problems within a reasonable amount of computation time. The results on randomly generated problems showed that the TS-based algorithms perform better than the SA-based algorithms on overall average. In particular, one of the suggested TA algorithms gave optimal solutions for the most small-sized test problems within very short computation times. Li et al. (2002) modeled the process planning as a combinatorial optimization problem with constraints, and a hybrid GA and SA approach was developed to solve it. The evaluation criterion was the combined strengths of machine costs, cutting tool costs, machine change costs, tool change and setup cost. The GA was carried out in the first stage to generate some initially good process plans. Based on a few selective plans with Hamming distances between each other, the SA algorithm was employed to search for alternative optimal or near-optimal process plans. In the GA and SA some preliminarily defined precedence constraints between features and operations were manipulated. The case study showed that this hybrid approach could achieve highly satisfactory results. Automated processing planning based on GA and/or SA also have been reported by Ma et al. (2002) and Alam et al. (2003). Tang and Liu (2002) developed a modified GA for solving the flow shop sequencing problem with the objective of minimizing mean flow time. To improve the general GA procedure, two additional operations were introduced into the algorithm. One replaces the worst solutions in each generation with the best solutions found in previous generations. The other improves the most promising solution, through local search, whenever the best solution has not been updated for a certain number of generations. Ding et al. (2005) presented an optimization strategy for process sequencing based on multi-objective fitness including minimum manufacturing cost, shortest manufacturing time and best satisfaction of manufacturing sequence rules. A hybrid approach was proposed to incorporate a GA, neural network and Analytical Hierarchical Process (AHP) for process sequencing. A globally optimized fitness function was defined including the evaluation of manufacturing rules using AHP, calculation of cost and time and determination of relative weights using neural network techniques. Particle Swarm Optimization (PSO) was also used by Cagnina et al. (2004) and Guo et al. (2006). Li et al. (2004) implemented a constraint-based TS approach for optimization of process plans. Zhang et al. (2006) considered an integrated Resource Selection and Operation Sequences (iRS/OS). Several types of objectives, minimizing the make-span for orders, balancing workloads among machine tools and minimizing the total transition times between machines in a local plant were taken into account. A new two vectors-based coding approach was also proposed to improve the efficiency by designing a chromosome containing two types of information. Using such chromosome type, they adapted multistage operation-based Genetic Algorithm (moGA) to find the Pareto optimal solutions. Salehi and Tavakkoli-Moghaddam (2009) presented an approach in which the process planning was divided into preliminary planning, and secondary/detailed planning. Analyzing the constraints and generating the feasible sequences of operations were done in the preliminary planning using GA and optimizing the process plan was conducted in secondary/detailed planning using GA too.

Evolutionary algorithms, which mimic living organisms in achieving optimal survival solutions, may often outperform conventional optimization methods. In the past two decades, GA has been widely applied for solving complex manufacturing problems, e.g. job shop scheduling and process planning. A novel approach was proposed that incorporates two GAs for optimizing the process planning (Salehi and Tavakkoli-Moghaddam 2009). However, an intelligent search strategy was also introduced to incorporate with GA for solving this optimization problem. In this approach, the process planning is divided into preliminary planning, and secondary/detailed planning. In the preliminary stage, feasible sequences of operations are carried out considering compulsive constraints of operations using an intelligent search and during the secondary and detailed level of planning that is similar to proposed approach by Salehi and Tavakkoli-Moghaddam (2009), the optimized sequence of the operations of the part and the optimized selection of the machine, cutting tool and TAD for each operation is acquired using GA considering additive constraints as well. All of the proposed CAPP systems only regarded the one of these two planning, it means that they obtained the optimized sequence of operations by analyzing the compulsive constraints (preliminary planning) or the optimized selection of the machine, cutting tool and TAD for each operation using operations precedence relations and the additive constraints (secondary and detailed planning). With merging the preliminary and detailed planning, implementing of compulsive and additive constraints, optimizing the sequence of the operations of the part and optimizing the selection of the machine, cutting tool and TAD for each operation simultaneously, this approach has useful contributions.

This paper is organized into four sections. “Materials and methods” depicts our approach for determining the optimized operations sequence with determining a machine, cutting tool, and TAD for each operation. “Example study” presents an example and discuses about it. Finally, conclusions are summarized in “Conclusion”.

Materials and methods

Figure 2 shows the modular structure of the proposed CAPP system, with the planning activities divided into the preliminary planning and detailed planning. This modular structure is approximately similar to the presented modular structure in Salehi and Tavakkoli-Moghaddam (2009). The preliminary planning generates feasible sequences of operations that are independent of the resources, considering compulsive constraints. The detailed planning generates optimal/near-optimal sequences of operations and selects a machine, cutting tool and TADs for each operation of these sequences, considering additive constraints. We present Fig. 2 to illustrate an overall perspective about the CAPP system. Only the distinct boxes demonstrate the sections that are analyzed in this paper.

In the absence of a CAD modeler which will yield manufacturing features along with the geometrical and technological information, a user interface is developed in the present work for part representation in terms of form features and their attributes. The user interactively inputs the details from the engineering drawing of the part. Then, the system selects operations that possess the capability to produce the form features based on their geometrical and technological requirements that this paper doesn't consider this problem. The system also generates the compulsive constraints from the inter-relationships that exist between the operations. Using these constraints, an operations precedence graph is generated by the system. A novel searching strategy is incorporated in the system that searches the tree representing all the paths and generates initial sequences.

In the detailed planning, machines and cutting tools that can perform the operations are selected by the system from the machine database and tool database. Then, based on selected machines and cutting tools, TADs are determined. In the next step, optimization of the operation sequence and selection of the machine and cutting tool and TAD for each operation based on the additive constraints are carried out. Finally, the system generates a process sheet that tabulates the operation number, operation description and the machine tool, cutting tool and TAD for each operation.

Preliminary planning

If there are n operations for producing a part, then the aggregation $S = \{o_1, o_2, \ldots o_n\}$ of operations has n members. Here, o_i indicates the ith operation. Each combination sequence of all operations in S represents one solution. Operation sequencing can be regarded as a process in which a series of constraints r_i ($R = \{r_1, r_2, \ldots r_p\}$) is imposed on aggregation S one by one where r_i is a constraint. Then the final operation sequence which can fully or mostly satisfy R is considered to be the optimal operation sequence. The goal of operation sequencing is to find the optimal operation sequence. It is known through practice that in operation sequencing, the theoretic constraints cannot be satisfied fully, so if the final operation sequence can satisfy obligatory constraints and other additive constraints, the operation sequence is reasonable (Bo et al. 2006) According to this condition, the constraint aggregation of the operation sequence may be divided into two categories: a compulsive constraint aggregation and an additive constraint aggregation.

In this research, order and clustering constraints are considered as the compulsive constraint aggregation that is implemented for acquiring the feasible operation sequences in the preliminary planning and the optimization constraint is considered as the additive constraint aggregation that is implement for acquiring the optimized operation sequence and the optimized selection of the machine, cutting tool and TAD for each operation in the secondary and the detailed planning.

Analyzing constraints—In the preliminary planning, for producing the feasible sequences of operations the compulsive constraint aggregation is applied so that if a sequence of operations cannot satisfy the compulsive constraints, it is infeasible and must be eliminated. In this paper two compulsive constraints are considered:

1. Order constraint: An order constraint is represented in the form of $A \rightarrow B$ meaning that operation B cannot take place until operation A has finished. This constraint includes locating constraints, accessibility constraints, non-destruction constraints, first rough machining and last finish machining, first main machining and last subordinate machining, first basic reference and last others part, and so forth.

2. Clustering constraint: A clustering constraint is represented in the form of $A \leftrightarrow B$ which has two meanings. It may signify that two operations A and B must be done with one holding/setup or must be handled together, such as the constraints that refer to a datum: parallelism, perpendicularity, angularity, concentricity, circular run-out, total run-out, symmetry and the like. However, it may signify that two operations A and B must be performed together successively without attending the order, such as the constraint of rough machining and fine machining separation. This means all the operations in the same azimuth side must be done together successively and also means that the same kind of operations or the operation that can be applied with the same cutting tool must be carried out together successively. For clustering constraint, the order is unimportant.

In this paper, the analyses for the constraints are carried out in similar way as Salehi and Tavakkoli-Moghaddam (2009). However, in this research, an intelligent search is presented for producing the initial sequences of operations in the preliminary planning instead of using GA,

Intelligent search strategy—In this section, a graphical representation that can be also referred to as OPG (Operation Precedence Graph) is generated using the order constraints. Then, the intelligent search generates the sequences of operations based on the order constraints. These initial sequences are tested for validity using the clustering constraints and infeasible sequences are eliminated.

Some of features in a part will act as references to other features and their relevant operations can be carried out first. Thus, the operations precedence graphical will have some starting nodes assuming null nodes as their parent.

The system starts searching from the null node, selects any one of the starting nodes and stores its reference/setup number. While selecting the next node, any of the following cases may arise:

Case 1 A node may have one or more child nodes of the same setup. Then, Step 1 is followed.

Step 1 The system selects any of the children randomly and the search starts from that child node.

Case 2 A node does not have a child of same setup number or it may have child of same setup with either of its parents not selected earlier. Then, Step 2 is executed.

Step 2 The system traces back the parent hierarchy and looks for a matching child node of same setup number. If it finds a match, succeeding node is selected randomly among them. In case, no child node of same setup is available, all the left out nodes of the parent hierarchy are considered and any one of them is selected randomly and its setup number is stored.

Case 3 A node does not have a child. In such situation, Step 2 is repeated.

This process is repeated till all nodes are traversed.

After producing the sequences, validity of sequences must be checked by clustering constraints and the valid sequences must be selected for the detail and second planning.

Secondary and detailed planning

The optimized sequence of the operations of the part and the optimized selection of the machine, cutting tool and TAD for each operation are obtained in the secondary and detailed level of planning. For this purpose, a proposed GA is also applied to implement the optimization constraints as an additive constraint aggregation. The proposed GA in this section is similar to the GA presented in Salehi and Tavakkoli-Moghaddam (2009). In this paper, we describe the proposed GA in the fallowing stages.

Coding strategy—a gene in a string represents an operation ID and corresponding machine, tool and tool access direction (TAD), which is used to accomplish this operation. Sequence of operations is represented by the order of the genes in the string. Table 1 shows the representation of a six-operation process plan. 'Op3' represents operation 3; M04, T02 and −x in the other rows represent the machine, tool and TAD, respectively, that are used to perform operation 4, so are the other columns.

Population initialization—Once the number of the feasible sequences of the operations is prescribed, the procedures for population initialization are given as follows:

1. Randomly select one sequence from the available feasible sequences of the operations list.
2. Visit the first selected operation.
3. Randomly select machines and tools that can be used for manufacturing the operation.
4. Randomly select one amongst all possible TADs for the operation.
5. Repeat steps (3) and (4), until each operation has been assigned a machine, a tool and a TAD.
6. Repeat steps (1)–(5) until the feasible sequences of the operations are finished.

Fitness function—In this stage the additive constraint aggregation may be implemented. The optimization constraints are often considered as the additive constraint aggregation. These constraints mean that some target functions must be met in the technologic sequence decision, such as minimum processing times, minimum production cost and so on. In this research, the minimum processing time and minimum production cost can be employed to calculate the fitness of each operation sequence and to measure the efficiency of a manufacturing system.

1. Processing Time (PT), the commonly used criterion in practice, generally comprises Machining Time (MT), Machine Change Time (MCT), Tool Change Time (TCT) and Set-up Change Time (SCT). The fitness function is given as:

$$\begin{aligned}\text{Fitness}&=\text{PT}\\&=\text{MT}+\text{MCT}+\text{TCT}+\text{SCT}\end{aligned} \tag{1}$$

2. Total Production Cost (PC) consists of Machine Cost (MC), Tool Cost (TC), Machine Change Cost (MCC), Tool Change Cost (TCC) and Setup Change Cost (SCC). The fitness function is computed by:

$$\begin{aligned}\text{Fitness} &= \text{PC}\\ &= \text{MC}+\text{TC}+\text{MCC}+\text{TCC}+\text{SCC}\end{aligned} \quad (2)$$

We can use one of these criteria or a combination of them.

Selection operator—The most of selection methods, such as roulette wheel's extensions, scaling techniques, tournaments, elitist models, and ranking methods were presented for numerical optimization and their main objective was to reduce the sampling error and improve calculation precision. When using GAs for operations sequencing, the natural number format is used for coding. The fitness value of individual is only a relative concept (only used for value comparisons and the value itself is not concerned), so the problem of sampling error does not exist. Compared with other selection operators, the "tournament selection" is more suitable for the problem of operations sequencing (Bo et al. 2006). In order to guarantee the astringency of GA, the optimal (best) individual in one generation must be kept for the next generation. Other individuals in population are selected using the "tournament selection" operator. Suppose there have *W* individuals to be selected, selecting two individuals randomly from the population and keeping the better one for the next generation. Repeating this process *W-1* times and then at last all individuals in the next generation will be obtained.

Crossover operator—There are three crossover operators for path representation: Partially Mapped Crossover (PMX) (Gorges-Schleuter 1985) Order Crossover (OX) (Davis 1985) and Cycle Crossover (CX) (Oliver et al. 1987). The aim of PMX crossover is keeping the important similarities of parent and child generation. The OX crossover emphasizes that the sequence order is very important. The CX reserves the absolute position of the elements in the parent generation. The research of Oliver et al. (1987) indicated that OX is 11% better than PMX and 15% better than CX. In this work, an OX crossover operator is adopted to ensure the local precedence of operations is met and a feasible offspring is generated. The procedure of the crossover operations is described as follows (see Table 2):

1. Based on the chromosome length, a crossover point is randomly generated. Each string is then divided into two parts, the left side and the right side according to the cutting point.
2. Copy the left side of parent 1 to form the left side of child 1. According to the order of operations in parent 2, the operator constructs the right side of child 1 with operations of parent 2, whose IDs are the same as operations of the right side in parent 1.
3. The role of these parents will then be exchanged in order to generate another offspring child 2.

Mutation operator—The mutation operator proposed here randomly selects some individuals. Then the positions of two codes in each individual are swapped randomly. Example of the mutation operation is shown in Table 3.

A new operator is developed to check the feasibility of the string obtained. If any string violates the constraints, the string is considered infeasible and the total score is given a very high value so that it will not come in the next generations:

$$\mathrm{Fitness}_{[\text{ infeasible solution}]}=\alpha \tag{3}$$

where α is a positive large number.

The frequency of machine change increases time and cost for accomplishing a job. To decrease machine change, the machine change time or cost index can be increased. The chromosomes resulting from these three operators are often known as offspring or children and these form the next generation's population. This process is repeated for a desired number of generations, usually up to a point where the system converges to significant well-performing sequences.

Example study

A part with 19 design features reported by Zhang (1997) is used to evaluate the capability of the proposed algorithm for a CAPP system. Figure 3 shows the part with 23 operations and Table 4 gives the information of operations that reported by Zhang (1997). Table 5 gives the constraints that are obtained from a set of order constraints including fix-turability, tolerance factor, good manufacturing practice and operation stages requirement for machining one feature. This issue does not have the clustering constrains.

The operation precedence graph generated using the order constraints is shown in Fig. 4 along with the reference/setup number.

The system searches for all the precedence relations among the operations and generates alternate sequences as strings of operations. This search procedure is evident in Fig. 4. The search starts from the root node (null node '0') and selects any one the start nodes, '14' or '5'. Let us assume that node '5' is selected and reference/setup number 2 is stored. At this stage, case 1 in the search algorithm applies and node '5' will have one node '6' as the succeeding one. '6' is selected and its reference/setup number 2 is stored. The nodes '15' and '4' are possible succeeding nodes, and '4' gets selected randomly. At this stage, case 2 is applicable and Step 2 yields the succeeding node '15'. Node '15' is selected and this search procedure is repeated till all the nodes are traversed. The resulting sequence is shown in the first row in Table 6. Other initial sequences are also included in Table 6.

These initial sequences are tested for validity using the clustering constraints and infeasible sequences are eliminated. In this case, we have no clustering constraints thus all of the sequences in Table 6 are feasible.

For producing the part, available resources in the job shop and their cost indices are given in Table 7. The cost indices of machines, tools and setup changes are MCCI = 300, TCCI = 10, and SCCI = 90, respectively. For comparison, similar GA parameters used by Zhang (1997) are used. These parameters are: population size 50, crossover rate 0.7, mutation rates 0.6, and generations 8,000 (the stopping criterion). A final alternative optimal/near-optimal sequence is given in the Table 8.

It can be concluded that the precedence relationships shown in Table 5 are maintained, such as op9 after op8, op8 after op7 due to operation stages requirements, op5 before op21 due to fixturability, op1 and op2 will be finished before op3 due to good manufacturing practice, etc.

The number of iterations for producing the optimal process plans was about 50 generations for each run, and the computation time using a PC with an Intel Pentium 4.2 GHz processor, 1 GHz memory was about 10 s. In the final process plan produced by Zhang (1997), the result was produced after 8,000 generations, and the computation time on a Pentium PC at 133 MHz was about 7 min (Zhang 1997). Through comparison with the process plan reported in Zhang (1997), it can be seen that the proposed approach can generate the most optimal process plan, which has the same quality obtained by Zhang, within the much fewer number of iterations (50 compared to 8,000 iterations). In addition, the more significant improvement of the proposed approach is that it generates the feasible sequences based on the analysis of various constraints in the operation sequences using an intelligent search strategy.

Through comparison between the proposed approach and the method reported in Salehi and Tavakkoli-Moghaddam (2009) that implement a GA in preliminary planning instead of intelligent search, it can be seen that in this work, the preliminary planning gives fewer computation time. To verify this result we simulated 10 process planning problems considering different number of operations and diverse information for operations in order to solve problems using the two approaches. The results are shown in Fig. 5. It can be seen that the proposed approach in this paper solves the problem using the fewer computation time compared to the method reported in Salehi and Tavakkoli-Moghaddam (2009). In two GAs approach reported in Salehi and Tavakkoli-Moghaddam (2009), the total time computation comprises the time computation for each GA. While in our proposed approach this time comprises the time computation for one GA and the intelligent search.

Certain abstractions are necessary in the preliminary planning, as it is done without considering the specific resources such as machine tools, cutting tools, and so forth. Therefore, any reference to processes, setups and tools is only to be understood as abstractions. In the real word situation that there are more than one part for planning, their processes can be planned by the proposed approach. However, it is necessary that the processes of each part are considered as a cluster in the preliminary planning. By using this approach, the break down in the final stage of machining can be modeled for each part. In addition, the presented model can plan the clamping and fixturing. For this purpose, the coding scheme should be extended for the proposed chromosome. In other words, a row for the fixture and a row for the clam should be added in the chromosomes, and the fitness function should be extended so that it is included the cost of the fixturing or clamping change. By this approach, the fixturing or clamping change costs is also minimized.

Conclusion

In this paper, the process planning was divided into preliminary planning and secondary/ detailed planning. The preliminary planning is independent of resources, as it involves abstractions of processes, setups, and so forth. In this stage, after necessary operations for a part based on the form features selected, on the operations and their inter-relationships, the preliminary sequences are determined. During the preliminary planning, an efficient intelligent search is proposed to explore the large solution space of valid operation sequences under compulsive constraints. The intelligent search strategy presented in this paper generates the initial sequences from the operation precedence relation based on the order constraints, that the feasibility of them was checked by clustering constraints. The results of the preliminary planning are used in the next phase, the secondary/detailed planning in which the exact operations, in which machine tools, cutting tools and TADs are considered, and sequences of operations are optimized at the process level. A proposed genetic algorithm (GA) in this paper optimizes the sequences using a criterion involving the minimum processing time or minimum production cost that represents machine changes, setup changes and tool changes.

The process planning is a combinatorial problem with interacting constraints, and the preliminary planning reduces the number of combinations to be examined. The GA technique handles this combinatorial problem very well, and the reduction in the size of the problem at each stage makes the algorithm a very fast and efficient.

The proposed algorithm can obtain optimal/near-optimal solutions, requiring about 10–40 s, depending on the number of operations and the number of generations. Several case studies were considered and the related results were obtained in less than 40 s. For a typical case of an oil pathway broad with 29 operations, the results were obtained in 13.3 s for a population size of 15 and 100 generations. The case study involving 38 operations took 15.4 s. Since the computation time taken to generate the optimal/near-optimal sequences is low, the software can be run several times to facilitate the process planner in obtaining alternative sequences.

References

Alam MR, Lee KS, Rahman M, Zhang YF. Process planning optimization for the manufacture of injection moulds using a generic algorithm. International Journal of Computer Integration Manufacturing. 2003; 16(3):181–191.

Bo ZW, Hua LZ, Yu ZG. Optimization of process route by genetic algorithms. Robotics and Computer-Integrated Manufacturing. 2006; 22:180–188.

Cagnina, L.; Esquivel, S.; Gallard, R. Congress on Evolution Compution. Portland, USA: 2004. Particle swarm optimization for sequencing problems: a case study; p. 536-541.

Case K, Harun Wan WA. Feature-based representation for manufacturing planning. International Journal of Production Research. 2000; 38(17):4285–4300.

Davis, L. Applying adaptive algorithms to epistatic domains. Proceedings of the international joint conference on artificial intelligence; 1985. p. 162-164.

Ding L, Yue Y, Ahmet K, Jackson M, Parkin R. Global optimization of a feature-based process sequence using GA and ANN techniques. International Journal of Production Research. 2005; 43(15):3247–3272.

Gorges-Schleuter, M. ASPARAGOS an asynchronous parallel genetic optimization strategy. Proceedings of the first international conference on genetic algorithms; Hillsdale, NJ: Lawrence Erlbaum Associates; 1985. p. 422-727.

Guo YW, Mileham AR, Owen GW, Li WD. Operation sequencing optimization using a particle swarm optimization approach. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture. 2006; 220(B12):1945–1958.

Lee D-H, Kiritsis D, Xirouchakis P. Branch and fathoming algorithms for operation sequencing in process planning. International Journal of Production Research. 2001; 39:1649–1669.

Li WD, Ong SK, Nee AYC. Hybrid genetic algorithm and simulated annealing approach for the optimization of process plans for prismatic parts. International Journal of Production Research. 2002; 40:1899–1922.

Li WD, Ong SK, Nee AYC. Optimization of process plans using a constraint-based tabu search approach. International Journal of Production Research. 2004; 42(10):1955–1985.

Ma, GH.; Zhang, F.; Zhang, YF.; Nee, AYC. An automated process planning system based on genetic algorithm and simulated annealing. Proceedings of the ASME design engineering technical conference; 2002. p. 57-63.

Maropoulos PG, Baker RP. Integration of tool selection with design (part 1. Feature creation and selection of operations and tools). Journal of Mater Process Technology. 2000; 107:127–134.

Oliver, IM.; Smith, DJ.; Holland, JRC. A study of permutation crossover operators on the traveling salesman problem. Proceedings of the second international conference on genetic algorithms; Hillsdale, NJ: Lawrence Erlbaum Associate; 1987. p. 224-230.

Qiao L, Wang XY, Wang SC. A GA-based approach to machining operation sequencing for prismatic parts. International Journal of Production Research. 2000; 38:3283–3303.

NIH-PA Author Manuscript
NIH-PA Author Manuscript

Reddy SVB, Shunmugam MS, Narendran TT. Operation sequencing in CAPP using genetic algorithm. International Journal of Production Research. 1999; 37:1063–1074.

Salehi M, Tavakkoli-Moghaddam R. Application of genetic algorithm to computer-aided process planning in preliminary and detailed planning. Engineering Applications of Artificial Intelligence. 2009; 22:1179–1187.

Tang L, Liu J. A modified genetic algorithm for the flow shop sequencing problem to minimize mean flow time. Journal of Intelligent Manufacturing. 2002; 13(1):61–67.

Zhang H, Gen M, Seo Y. An effective coding approach for multiobjective integrated resource selection and operation sequences problem. Journal of Intelligent Manufacturing. 2006; 17(4):385–397.

Zhang, F. ME thesis. National University of Singapore; 1997. Genetic algorithm in computer-aided process planning.

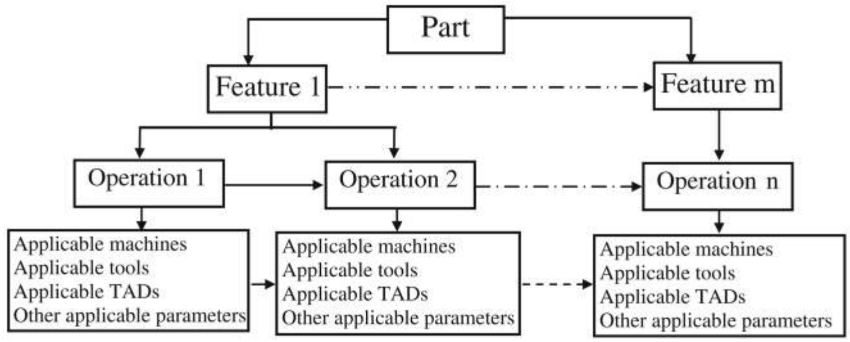

Fig. 1.
Representation of a process plan

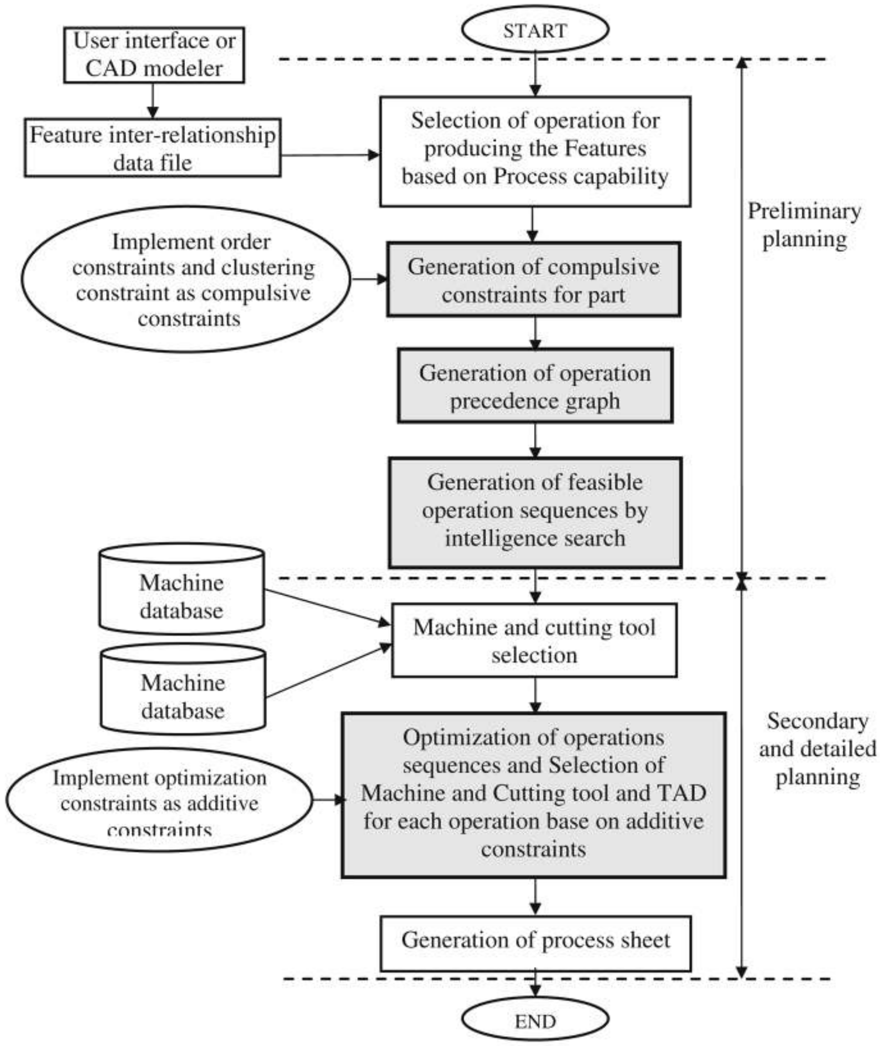

Fig. 2.

Schematic diagram of the proposed CAPP system

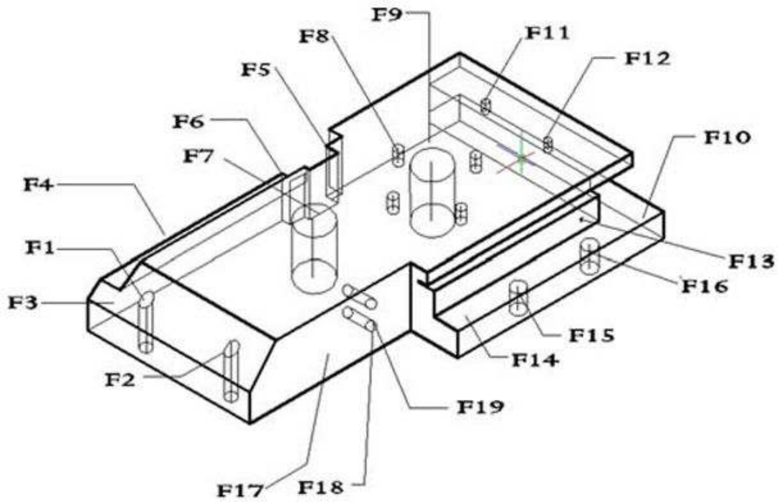

Fig. 3.
A prismatic part used by Zhang (1997)

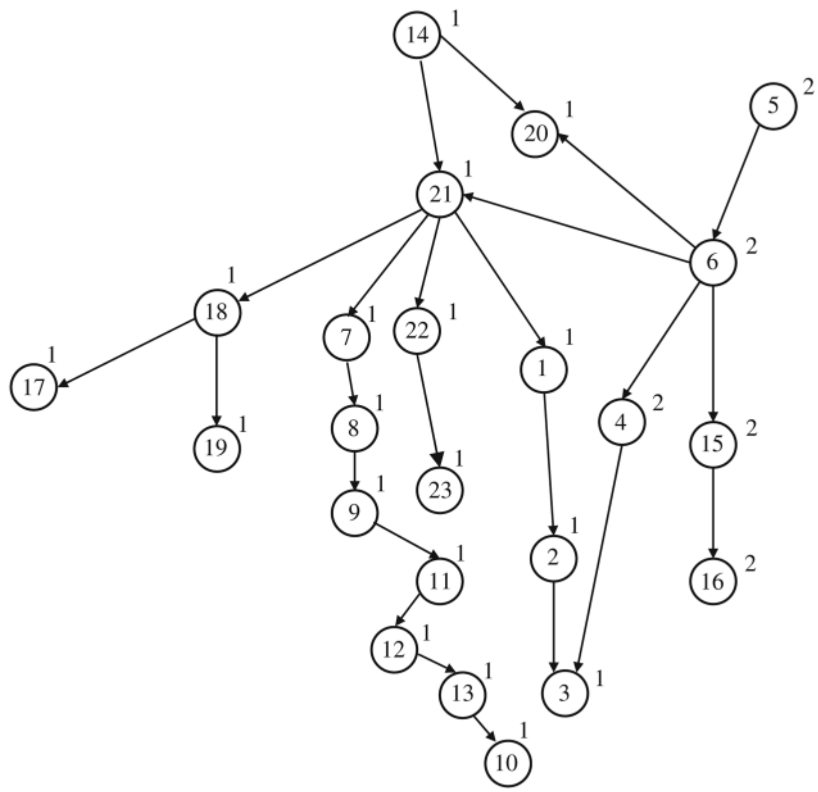

Fig. 4.
Operation precedence graph

NIH-PA Author Manuscript

NIH-PA Author Manuscript

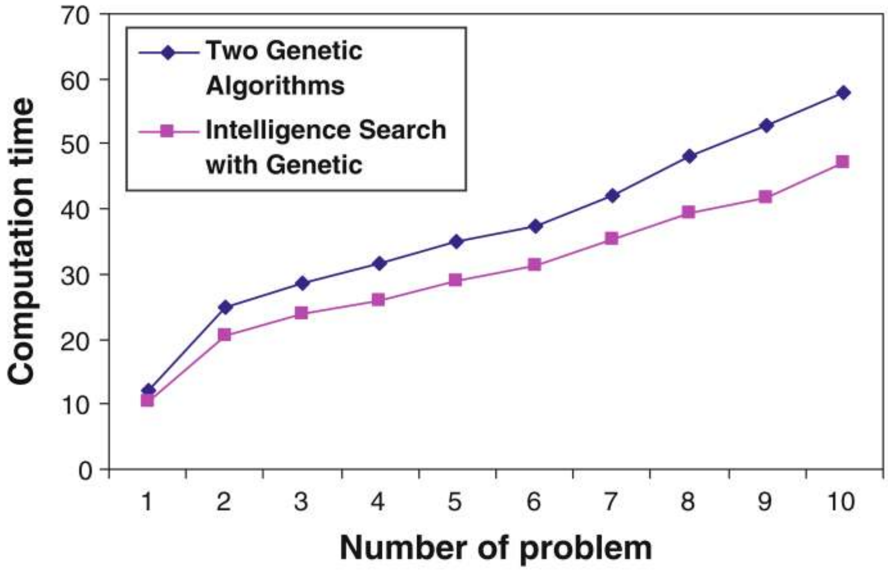

Fig. 5.
Comparison of the computation time between two approaches

NIH-PA Author Manuscript

NIH-PA Author Manuscript

Table 1

Representation of a process plan

Op3	Op2	Op4	Op6	Op1	Op5
M04	M01	M03	M02	M01	M02
T02	T03	T04	T02	T01	T03
$-x$	$+y$	$-z$	$-y$	$+z$	$-y$

Table 2

Crossover example

Parent 1	Op3	Op4	Op2	Op1	Op6	Op5
	M01	M04	M03	M02	M04	M02
	T04	T03	T04	T02	T03	T01
	−*x*	+*y*	−*z*	−*z*	+*z*	−*y*
Parent 2	Op2	Op5	Op3	Op6	Op1	Op4
	M04	M01	M02	M04	M04	M02
	T04	T03	T02	T03	T03	T01
	−*z*	−*x*	−*z*	+*z*	+*y*	−*y*
Child 1	Op3	Op4	Op2	Op5	Op6	Op1
	M01	M04	M04	M01	M04	M04
	T04	T03	T04	T03	T03	T03
	−*x*	+*y*	−*z*	−*x*	+*z*	+*y*
Child 2	Op2	Op5	Op3	Op4	Op1	Op6
	M04	M01	M01	M04	M02	M04
	T04	T03	T04	T03	T02	T03
	−*z*	−*x*	−*x*	+*y*	−*z*	+*z*

Table 3

Mutation example

Chromosome (before mutation)	Op3	Op4	Op2	Op1	Op6	Op5
	M01	M04	M03	M02	M04	M02
	T04	T03	T04	T02	T03	T01
	−*x*	+*y*	−*z*	−*z*	+*z*	−*y*
Chromosome (after mutation)	Op3	Op6	Op2	Op1	Op4	Op5
	M01	M04	M03	M02	M04	M02
	T04	T03	T04	T02	T03	T01
	−*x*	+*z*	−*z*	−*z*	+*y*	−*y*

Table 4

Operation information table in Zhang (1997)

Feature-ID	Operations	TAD candidates	Machine candidates	Tool candidates
F1	Drilling (op1)	$+z$, $-z$	M-01, M-02, M-03	T-01
F2	Drilling (op2)	$+z$, $-z$	M-01, M-02, M-03	T-01
F3	Milling (op3)	$+z$, $-z$	M-02, M-03	T-08
F4	Milling (op4)	$+y$, $-z$	M-02, M-03	T-05, T-06
F5	Milling (op5)	$+y$	M-02, M-03	T-05, T-06
F6	Milling (op6)	$+y$	M-02, M-03	T-05, T-06
F7	Drilling (op7)	$+z$, $-z$	M-01, M-02, M-03	T-02
F7	Reaming (op8)	$+z$, $-z$	M-01, M-02, M-03	T-03
F7	Boring (op9)	$+z$, $-z$	M-03, M-04	T-04
F8	Drilling (op10)	$+z$, $-z$	M-01, M-02, M-03	T-01
F9	Drilling (op11)	$+z$, $-z$	M-01, M-02, M-03	T-02
F9	Reaming (op12)	$+z$, $-z$	M-01, M-02, M-03	T-03
F9	Boring (op13)	$+z$, $-z$	M-03, M-04	T-04
F10	Milling (op14)	$+x$	M-02, M-03	T-05, T-06
F11	Drilling (op15)	$-z$	M-01, M-02, M-03	T-01
F12	Drilling (op16)	$-z$	M-01, M-02, M-03	T-01
F13	Milling (op17)	$-y$, $-z$	M-02, M-03	T-05, T-07
F14	Milling (op18)	$-y$, $-z$	M-02, M-03	T-05, T-06
F15	Drilling (op19)	$+z$, $-z$	M-01, M-02, M-03	T-01
F16	Drilling (op20)	$+z$, $-z$	M-01, M-02, M-03	T-01
F17	Milling (op21)	$-y$	M-02, M-03	T-05, T-06
F18	Drilling (op22)	$-y$	M-01, M-02, M-03	T-01
F19	Drilling (op23)	$-y$	M-01, M-02, M-03	T-01

Table 5

Constraint generated for the part

Order constraints	1 → 2, 1 → 3, 21 → 1, 21 → 2, 4 → 3, 14 → 3, 21 → 3, 5 → 4, 6 → 4, 8 → 10, 11 → 13, 5 → 15, 5 → 16, 5 → 17, 5 → 18, 5 → 19, 5 → 20, 5 → 21, 5 → 22, 5 → 23, 6 → 15, 6 → 16, 6 → 17, 6 → 18, 6 → 19, 6 → 22, 21 → 23, 22 → 23, 6 → 23, 7 → 8, 12 → 13, 7 → 10, 14 → 20, 7 → 12, 7 → 13, 21 → 7, 8 → 9, 8 → 11, 8 → 12, 8 → 13, 21 → 8, 21 → 9, 11 → 10, 11 → 12, 21 → 10, 5 → 6, 21 → 11, 12 → 13, 9 → 10, 9 → 11, 9 → 12, 9 → 13, 12 → 10, 21 → 12, 7 → 9, 21 → 13, 14 → 17, 14 → 18, 14 → 19, 14 → 21, 14 → 22, 14 → 23, 7 → 11, 15 → 16, 18 → 17, 18 → 19, 21 → 17, 21 → 18, 21 → 19, 21 → 20, 21 → 22
Clustering constraints	–

Table 6

Typical sequences generated using operation precedence relation

	Sequences	Validity
1	5 → 6 → 4 → 15 → 16 → 14 → 20 → 21 → 18 → 17 → 19 → 1 → 2 → 3 → 7 → 8 → 9 → 11 → 12 → 13 → 10 → 22 → 23	Valid
2	5 → 6 → 15 → 16 → 4 → 14 → 21 → 18 → 17 → 19 → 22 → 23 → 20 → 1 → 2 → 3 → 7 → 8 → 9 → 11 → 12 → 13 → 10	Valid
3	14 → 5 → 6 → 21 → 22 → 23 → 18 → 17 → 19 → 1 → 2 → 7 → 8 → 9 → 11 → 12 → 13 → 10 → 15 → 16 → 4 → 3 → 20	Valid
4	14 → 5 → 6 → 20 → 4 → 15 → 16 → 21 → 22 → 23 → 1 → 2 → 3 → 18 → 17 → 19 → 7 → 8 → 9 → 11 → 12 → 13 → 10	Valid
5	14 → 5 → 6 → 4 → 21 → 22 → 23 → 7 → 8 → 9 → 11 → 12 → 13 → 10 → 18 → 19 → 17 → 1 → 2 → 3 → 15 → 16 → 20	Valid
6	14 → 5 → 6 → 21 → 18 → 17 → 19 → 7 → 8 → 9 → 11 → 12 → 13 → 10 → 22 → 23 → 1 → 2 → 20 → 15 → 16 → 4 → 3	Valid
7	14 → 5 → 6 → 4 → 15 → 16 → 20 → 21 → 18 → 19 → 17 → 22 → 23 → 7 → 8 → 9 → 11 → 12 → 13 → 10 → 1 → 2 → 3	Valid
8	5 → 6 → 4 → 15 → 16 → 14 → 21 → 22 → 23 → 18 → 17 → 19 → 7 → 8 → 9 → 11 → 12 → 13 → 10 → 1 → 2 → 3 → 20	Valid
9	14 → 5 → 6 → 15 → 16 → 4 → 20 → 21 → 7 → 8 → 9 → 11 → 12 → 13 → 10 → 18 → 17 → 19 → 22 → 23 → 1 → 2 → 3	Valid
10	14 → 5 → 6 → 4 → 21 → 22 → 23 → 7 → 8 → 9 → 11 → 12 → 13 → 10 → 18 → 19 → 17 → 1 → 2 → 3 → 15 → 16 → 20	Valid

Table 7

Available resources in the job shop in Zhang (1997)

ID	Type	Cost indices
M-01	Press drill	10
M-02	Vertical milling	35
M-03	Vertical CNC milling	60
M-04	Boring machine	50
T-01	Drill φ 0.2	3
T-02	Drill φ 1.2	3
T-03	Reamer	8
T-04	Boring tool	15
T-05	Milling cutter	10
T-06	Milling cutter	15
T-07	Slot cutter	10
T-08	Chamfer tool	10
MCCI: 300 SCCI: 90 TCI: 10		

NIH-PA Author Manuscript
NIH-PA Author Manuscript

Table 8

An optimal/near-optimal sequence

order	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	10	21	22	23
Op-ID	14	5	6	4	21	22	23	20	1	18	15	16	19	2	17	7	3	8	9	11	12	13	10
M-ID	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	03	03	03	03	03
T-ID	05	05	05	05	05	01	01	01	01	01	01	01	01	01	05	02	08	03	04	02	03	04	01
TAD	+x	+y	+y	+y	−y	−y	−y	−z	−z	−z	−z	−z	−z	−z	−z	−z	−z	−z	−z	−z	−z	−z	−z