

Optimization of Machining Parameters Using Estimation of Distribution Algorithms

Shu-tong Xie, Yin-biao Guo*, Hai-bin Huang, Jing Lin
Department of Mechanical and Electrical Engineering
Xiamen University
Xiamen 361005, P.R.China
E-mail: stxie@126.com

Abstract—In computer numerical control (CNC) machining problems, it is important to reduce the production cost. To deal with the nonlinear optimization problem of machining parameters which aims to minimize the unit production cost (UC) in multi-pass turning operations, two estimation of distribution algorithms (EDAs) incorporated with gene repair method are proposed to search the optimal solution for machining parameters. Computer simulation results show that the proposed algorithms are efficient in searching the optimal machining parameters, which significantly reduce the unit production cost.

Keywords- Machining parameters; Optimization; Estimation of distribution algorithm; Turning

I. INTRODUCTION

In manufacturing world, the primary objective in machining operations is to produce high-quality products with low cost. In order to minimize the machining cost for machining economics problem, the optimization of machining parameters is one of the most important issues since these parameters strongly affect the cost and quality [1]. The machining optimization problem in multipass turnings becomes very complicated when plenty of practical constraints have to be considered [2]. Conventional optimization techniques, such as geometric programming [3] and dynamic programming [4, 5], may be useful for some specific problems. However they are inclined to give local optimal results. Therefore, meta-heuristic algorithms have been introduced in solving machining economics problems because of their power in global searching and robustness as well. Chen and Tsai [6] have employed the combination of simulated annealing (SA) and pattern search (PS) for solving these problems. Genetic algorithm (GA) was used by many researchers [7-9] to optimize the machining parameters in multi-pass turnings. To deal with the same optimization problem in multipass turnings, Ants colony optimization (ACO) [9-11] was also developed. Recently, particle swarm optimization (PSO) [12] was applied to optimize the multi-pass turning operations. We attempted to develop estimation of distribution algorithms (EDAs) [13-16] with gene repair method to tackle the complicated optimization problems.

The rest of this paper is organized as follows. Section 2 introduces a machining model of multipass turning operations. Section 3 proposes two EDAs to search the optimal results of machining parameters. And the handling methods of constraints are described in detailed. Simulation results and discussions are given in section 4. In the last section, we conclude this work.

II. MATHEMATICAL MODEL OF MULTI-PASS TURNING OPERATIONS

In this paper, the multi-pass turning operations include multiple rough cuts and a single finish cut. The mathematical model proposed by Chen and Tsai [6, 9] was adopted for optimizing machining parameters. The objective of the optimization was to determine the optimal machining parameters, including cutting speed, feed rate and depth of cut for both rough and finish machining (namely $V_r, V_s, f_r, f_s, d_r, d_s$), in order to minimize UC. The UC for multi-pass turning operations can be divided into four basic cost elements.

- (1) Cutting cost by actual time in cut (C_M).
- (2) Machine idle cost for loading and unloading operations and idling tool motion (C_I).
- (3) Tool replacement cost (C_R).
- (4) Tool cost (C_T).

Hence UC can be represented as

$$\begin{aligned}
 UC &= C_M + C_I + C_R + C_T \\
 &= \left[\frac{\pi DL}{1000V_r f_r} \left(\frac{d_t - d_s}{d_r} \right) + \frac{\pi DL}{1000V_s f_s} \right] k_0 \\
 &\quad + \left[t_c + (h_1 L + h_2) \left(\frac{d_t - d_s}{d_r} + 1 \right) \right] k_0 \\
 &\quad + \left[\frac{\pi DL}{1000V_r f_r} \left(\frac{d_t - d_s}{d_r} \right) + \frac{\pi DL}{1000V_s f_s} \right] \frac{t_e}{T_p} k_0 \\
 &\quad + \left[\frac{\pi DL}{1000V_r f_r} \left(\frac{d_t - d_s}{d_r} \right) + \frac{\pi DL}{1000V_s f_s} \right] \frac{k_t}{T_p}
 \end{aligned} \tag{1}$$

Where

UC : Unit production cost excluding material cost (\$/piece)

k_0 : Direct labor cost, including overhead (\$/min)

k_t : Cutting edge cost (\$/edge)

D, L : Diameter and length of work-piece (mm)

This work is supported by the Important Sci-Tech Special Projects of Fujian Province(No.2006HZ0002-4) and Science Foundation for the Youth of Fujian Province, China (2009J05136), * Corresponding author

- d_t : Depth of material to be removed (mm)
 h_1, h_2 : Constants relating to cutting tool travel and approach/departure time (min)
 t_c : Preparation time for loading and unloading time (min)
 t_e : Time required to exchange a tool (min)
 T_p : Tool life of weighted combination of T_r and T_s (min)
 V_r, V_s : Cutting speeds in rough and finish machining (m/min)
 f_r, f_s : Feed rates in rough and finish machining (mm/rev)
 d_r, d_s : Depths of cut for each pass of rough and finish machining (mm)
 n : Number of rough cuts, an integer

$$n = \frac{d_t - d_s}{d_r}, \text{ and } n \in Z_+ \quad (2)$$

For the minimization of UC, practical constraints that present the states of machining process are to be considered, which are listed as following. And Detailed constraints equations can be seen in [6, 9]

- (1) Parameter bounds
- (2) Tool-life constraint
- (3) Operating constraints consisting of surface finish constraint, force and power constraint
- (4) Stable cutting region constraint
- (5) Chip-tool interface temperature constraint
- (6) Roughing and finishing parameter relations

III. ESTIMATION OF DISTRIBUTION ALGORITHMS

EDA is a class of population-based evolutionary algorithm [13-16]. While evolutionary algorithms, like GA, use a population of configurations to search for a solution to an optimization problem, EDA uses a probability function to represent an individual in the population. This probability function will be replaced in each generation. For this reason, EDA is also often called “probability-model building evolutionary algorithm”. According to different probability model used in EDA, EDA can be classified into three classes: without dependencies, bivariate dependencies and multivariate dependencies. In this paper, two kinds of EDA with the probability without dependencies, population based incremental learning (PBIL) and univariate marginal distribution algorithm (UMDA) [13-16], are developed to search the optimal machining parameters set.

The joint probability distribution of random vector $P(x)$ in PBIL is represented by

$$P_{l+1}(x) = (1 - \alpha)P_l(x) + \alpha \frac{1}{N} \sum_{k=1}^N x_l^k \quad (3)$$

Where l stands for the number of generations in the population, α is the learning rate, N is the number of selected best individuals.

On the other hand, the probability vector in UMDA can be calculated by (3) where α equals one, which is only different from the PBIL algorithm.

A. Probability Vector and the Framework of EDAs

The probability vector $P(x)$ consists of five variables to be optimized, i.e., cut speed, feed rates of both rough and finish machining, and the depths of cut of finish machining. Each variable is represented by a 20-bit binary string, as shown in Table I. The depth of cut of rough d_r can be obtained by (2) $n = (d_t - d_s) / d_r$ when n is fixed.

TABLE I. REPRESENTATION OF PROBABILITY VECTOR

1-20	21-40	41-60	61-80	81-100
V_r	f_r	V_s	f_s	d_s

The probability vector encoding, named as constraints-encoding method, is different from the chromosome encoding method in GA or PSO algorithms proposed previously [7-12]. The difference is that encoding bounds of the variables V_r, V_s, f_r, f_s, d_s are adjusted reasonably according to the specific constraints, while the previous algorithms directly use the given bound of these variants.

There are three advantages in using this encoding method, which are described as follows:

- (1) It can greatly reduce the number of infeasible individuals in the population;
- (2) It eliminates some constraints, such as the surface finish constraint, to avoid the corresponding penalty functions and to reduce the complexity of the problems;
- (3) It reduces the running time of the algorithm as the length of variables is reduced because the available bounds of them get small.

The framework of PBIL (UMDA) is shown in Fig. 1, in which MaxGen is the maximum number of iterations of the algorithm. The handling of constraints methods, penalty functions and gene repair, will be described in following sub-sections. To realize the extensive exploration in the first period of evolution, and speedy convergence in the last period of it, the gene repair method is employed after the half of number of iterations.

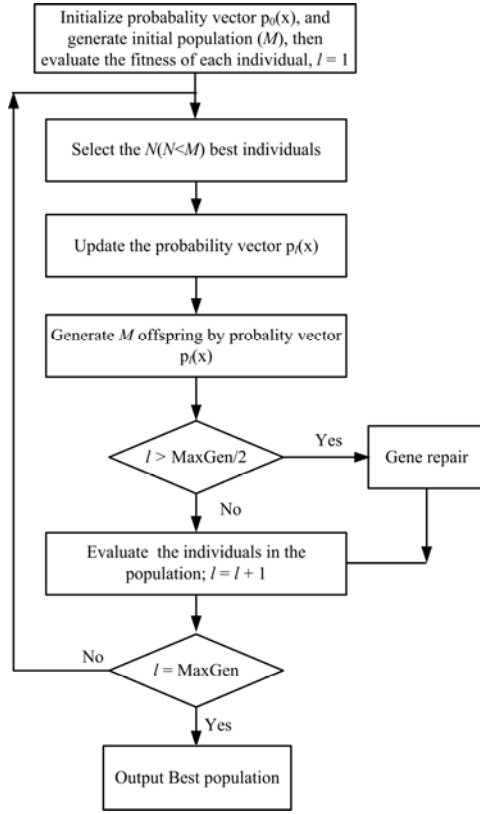


Figure 1. Flow chart of the proposed PBIL (UMDA) algorithm

B. Handling of Constraints

1) Penalty function

The penalty function is used to penalize the individuals who violate the constraints. The more the constraints are violated, the heavier the penalty will be needed. As a result, the fitness of them will be small, which leads to the probability in the probability vector to generate them being smaller. In this way, the penalty function is expressed as

$$Penalty(X) = \sum_{i=1}^k a_i \times h_i, \text{ where}$$

$$a_i = \begin{cases} 0 & \text{Satisfy constraints} \\ 1 & \text{Violate constraint}(s) \end{cases}$$

where k is the number of constraints, h_i is a penalty constant.

The individual that obtains lower UC gets higher fitness value. Thus the fitness function can be formulized as

$$fitness(X) = \begin{cases} -UC(X) & \text{Satisfy constraints} \\ -(UC(X) + Penalty(X)) & \text{Violate constraint}(s) \end{cases}$$

2) Gene repair

During the reproduction of the offspring some invalid individuals which violate some constraints will be generated. In fact, a number of them can be turned into valid individuals

by modifying the wrong gene of them, which is a gene repair method. However it is very difficult to repair all wrong genes in a vector (chromosome) generally. The depths of cut of finish machining d_s which is a gene in vector must satisfy the maximum of number of constraints at the same time. So in order to decrease the complexity of gene repair method, we only repair the gene d_s by adjust the value of it to satisfy all the relative constraints.

IV. SIMULATION AND DISCUSSION

In all the simulations, algorithms were run in a computer with a Core Duo T2080 CPU and 1GB memory. Each problem was tested 100 times with different random initial populations. As for PBIL, the population and offspring sizes are both set to 2000 and the algorithm stops after 50 generations. The truncation selection rate and learning rate is set to 0.9 and 0.9, respectively. As for UMDA, the parameters are the same as those of the PBIL. But it need not set the learning rate. We used the machining experiment example in Chen and Tsai [6, 9].

Fig. 2 compares the results obtained by the proposed algorithms and other previous meta-heuristic algorithms when the total depth of cuts $d_t = 6$. It can be seen from this figure that the proposed algorithms can achieve much better results than GA [9], PSO [12], SA [6] and FEGA [8]. From Table II, the average optimal UC obtained by PBIL and UMDA are still smaller than those given by previous algorithms. The number of points searched by our algorithms is not so big. At last, the performance of PBIL is similar to the UMDA. The main reason should be that the probability model used by two algorithms is the same, namely variants without dependencies.

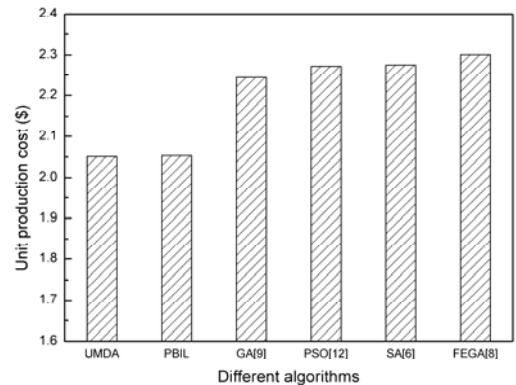


Figure 2. Comparison of optimal results between our algorithms and others

TABLE II. PERFORMANCE OF DIFFERENT ALGORITHMS

Algorithm	Avg. UC (\$)	Std. Error	Points
FE-GA[8]	2.3057	N/A	60,000
SA/PS[6]	2.2959	0.01624	18,571
PSO[12]	>2.2721	N/A	2,000
ACO[9]	2.2705	N/A	100,000
GA[9]	2.2538	N/A	100,000
PBIL	2.1101	0.03028	100,000
UMDA	2.1070	0.03628	100,000

V. CONCLUSION

Two EDA-based algorithms, PBIL and UMDA, were proposed to optimize machining parameters in multi-pass turning operations. By incorporating vector constraints-encoding and gene repair method into EDAs, the number of infeasible individuals in the evolutionary population was greatly reduced. Computer simulations showed that the proposed algorithms were efficient and effective. The optimal results achieved by our algorithms were much better than those obtained by other algorithms proposed previously, and the unit production cost was significantly reduced.

REFERENCES

- [1] I. Mukherjee, and P. K. Ray, "A review of optimization techniques in metal cutting processes," *Comput Ind Eng*, vol. 50, no. 1-2, pp. 15-34, May, 2006.
- [2] Y. C. Shin, and Y. S. Joo, "Optimization of machining conditions with practical constraints," *Int J Prod Res*, vol. 30, pp. 2907-2919, 1992.
- [3] D. S. Ermer, "Optimization of the constrained machining economics problem by geometric programming," *J Eng Ind Trans ASME*, vol. 93 Ser B, no. 4, pp. 1067-1072, 1971.
- [4] J. S. Agapiou, "The optimization of machining operations based on a combined criterion, part 1. The use of combined objectives in single-pass operations," *J Eng Ind Trans ASME*, vol. 114, no. 4, pp. 500-507, 1992.
- [5] J. S. Agapiou, "The optimization of machining operations based on a combined criterion, part 2. Multipass operations," *J Eng Ind Trans ASME*, vol. 114, no. 4, pp. 508-513, 1992.
- [6] M. C. Chen, and D. M. Tsai, "A simulated annealing approach for optimization of multi-pass turning operations," *Int J Prod Res*, vol. 34, no. 10, pp. 2803-2825, 1996.
- [7] G. C. Onwubolu, and T. Kumalo, "Optimization of multipass turning operations with genetic algorithms," *Int J Prod Res*, vol. 39, no. 16, pp. 3727-3745, Nov, 2001.
- [8] M. C. Chen, and K. Y. Chen, "Optimization of multipass turning operations with genetic algorithms: a note," *Int J Prod Res*, vol. 41, no. 14, pp. 3385-3388, Sep, 2003.
- [9] R. S. Sankar, P. Asokan, R. Saravanan, S. Kumanan, and G. Prabhakaran, "Selection of machining parameters for constrained machining problem using evolutionary computation," *Int J Adv Manuf Technol*, vol. 32, no. 9-10, pp. 892-901, Apr, 2007.
- [10] K. Vijayakumar, G. Prabhakaran, P. Asokan, and R. Saravanan, "Optimization of multi-pass turning operations using ant colony system," *Int J Mach Tools Manuf*, vol. 43, no. 15, pp. 1633-1639, Dec, 2003.
- [11] Y. C. Wang, "A note on 'optimization of multi-pass turning operations using ant colony system'," *Int J Mach Tools Manuf*, vol. 47, no. 12-13, pp. 2057-2059, Oct, 2007.
- [12] J. Srinivas, R. Giri, and S. H. Yang, "Optimization of multi-pass turning using particle swarm intelligence," *Int J Adv Manuf Technol*, vol. 40, no. 1-2, pp. 56-66, Jan, 2009.
- [13] H. Muhlenbein, and G. Paass, "From recombination of genes to the estimation of distribution part 1, binary parameter," in 4th International Conference on Parallel Problem Solving From Nature-PPSN IV, 1996, pp. 178-187.
- [14] H. Muhlenbein, "The equation for response to selection and its use for prediction," *IEEE T Evolut Comput*, vol. 5, no. 3, pp. 303-346, 1997.
- [15] P. Larranaga, and J. A. Lozano, *Estimation of distribution algorithms: A new tool for evolutionary computation*, Boston/Dordrecht/London: Kluwer Academic Publisher, 2002.
- [16] J. A. Lozano, P. Larranaga, I. Inza, and E. E. Bengoetxea, *Towards a New Evolutionary Computation: Advances in the Estimation of Distribution Algorithms*, Berlin: Springer-Verlag, 2006.