Int J Adv Manuf Technol (2007) 32: 902–910
DOI 10.1007/s00170-006-0425-6

ORIGINAL ARTICLE

N. R. Abburi · U. S. Dixit

Multi-objective optimization of multipass turning processes

Received: 18 May 2005 / Accepted: 9 December 2005 / Published online: 18 March 2006

Abstract In this paper, a methodology is proposed for the multi-objective optimization of a multipass turning process. A real-parameter genetic algorithm (RGA) is used for minimizing the production time, which provides a nearly optimum solution. This solution is taken as the initial guess for a sequential quadratic programming (SQP) code, which further improves the solution. Thereafter, the Pareto-optimal solutions are generated without using the cost data. For any Pareto-optimal solution, the cost of production can be calculated at a higher level for known cost data. An objective method based on the linear programming model is proposed for choosing the best among the Pareto-optimal solutions. The entire methodology is demonstrated with the help of an example. The optimization is carried out with equal depths of cut for roughing passes. A simple numerical method has been suggested for estimating the expected improvement in the optimum solution if an unequal depth of cut strategy would have been employed.

Keywords Turning process · Multi-objective optimization · Real-parameter genetic algorithm · Sequential quadratic programming · Linear programming · Lagrange multiplier

Nomenclature

C	Constant in extended Taylor's tool life equation
C_o	Operating cost (\$/min)
C_t	Tool cost (\$)
d	Depth of cut (mm)
d_F	Depth of cut for the finishing pass (mm)
d_R	Depth of cut for the roughing pass (mm)
D_f	Final diameter of the work piece (mm)
D_0	Initial work piece diameter (mm)
f_F	Feed for the finishing pass (mm/rev)
f_R	Feed for the roughing pass (mm/rev)
F_c	Total production cost per piece (\$)
F_{max}	Cutting force (kgf)
F_t	Fraction of tool consumed per piece
k, α, β	Constants used in the empirical relation for cutting force
L	Length of machining (mm)
m	Number of roughing passes
n	Exponent of extended Taylor's tool life equation
p, q, r	Exponents of speed, feed, and depth of cut in tool life equation
p_c, p_m	Crossover and mutation probabilities
P_{max}	Maximum power (kW)
r_i, u_i	Random numbers between 0 and 1
R	Nose radius of cutting tool (mm)
$R_{t_{max}}$	Peak-to-valley height of surface roughness for finishing pass
t_c	Tool change time (min)
t_s	Tool setting time per pass (min)
t_{ts}	Total tool setting time (min)
T_f	Tool life for the finishing pass (min)
T_L	Loading/unloading time per component (min)
T_{max}, T_{min}	Maximum and minimum allowed values of tool life
T_P	Total production time per component (min)
T_r	Tool life for the roughing pass (min)
T_{tF}	Total cutting time for finishing pass (min)
T_{tR}	Total cutting time for roughing passes (min)
v_F	Cutting speed for the finishing pass (m/min)
v_R	Cutting speed for the roughing pass (m/min)
η	Machine efficiency
η_c	Crossover index
η_m	Mutation index

N. R. Abburi · U. S. Dixit (✉)
Department of Mechanical Engineering,
Indian Institute of Technology,
Guwahati, 781 039, India
e-mail: uday@iitg.ernet.in
Tel.: +91-361-2582657
Fax: +91-361-2690762

1 Introduction

One of the most investigated problems in the area of machining is process optimization. Starting from the early work of Gilbert [1], a number of researchers have proposed various algorithms for optimizing a single-objective function reflecting machining performance [2–10]. Commonly used objective functions are the minimization of production cost, the maximization of production rate, and the maximization of profit rate. Abuelnaga and El-Dardiry [11] discussed a number of traditional optimization methods and highlighted the relative advantages and disadvantages of the methods for solving the problems of machining economics. They also pointed out that a weighted combination of several objective functions might be a suitable strategy for the optimization of machining processes.

In the recent years, there has been a trend for using non-traditional optimization techniques in lieu of traditional methods [12–16]. Some of these have been used for the optimization of multipass turning processes. These methods avoid the problem of getting trapped in local optima and enable to obtain a global (or nearly global) optimum solution. They are also ideally suited for multiple optimal solutions and for solving multi-objective problems. However, except in a few papers, only single-objective machining optimization problems have been solved. Davim and Antonio [17] have optimized the drilling of particulate metal-matrix composites by forming a utility function, as a weighted addition of various objectives. This function is minimized by a genetic algorithm. Sarvanan and Sachithanandam [18] have used a genetic algorithm for the multi-objective optimization of a surface grinding process. Production rate and production cost have been combined to make a single objective function. Baskar et al. [19] solved the same problem using the ant colony algorithm. Tosun and Ozler [20] optimized a hot turning process by using Taguchi's method, taking tool life and surface roughness of the machined surface as the twin objectives, which are combined to make a single objective by associating suitable weights with each objectives. Selection of the weights is a highly subjective decision and none of these authors have thrown light on this aspect.

In the present work, a combination of a real-parameter genetic algorithm (RGA) and sequential quadratic programming (SQP) is used for obtaining Pareto-optimal solutions in a multipass turning process. The multi-objective optimization problem has been formulated in a manner which does not require the cost data during the optimization phase. The cost data is needed only during the higher-level decision stage, in which the best among the various Pareto-optimal solutions is chosen. This is a major advantage of the procedure, as the cost data may not be easily available to the engineer running the optimization code. Technology remaining the same, the cost data may keep on fluctuating with the market conditions. Moreover, an accurate cost determination procedure usually provides a significantly different cost from a roughly estimated one [21]. In the present methodology, the higher-level decision maker can quickly select the best among the various Pareto-optimal solutions with the available knowledge of cost components. A linear programming model has been proposed to help in the higher-level decision process.

Apart from the methods described, simple heuristic techniques have been adopted to improve the efficiency of RGA. An equal depth of cut strategy has been adopted; however, a procedure for estimating the deviation of the optimum value from an unequal depth strategy has also been provided. Although a multipass turning process has been chosen as an example in this work, most of the techniques discussed in this paper are of a general nature and can be easily applied to other machining processes.

2 Problem formulation

In this section, the optimization problems have been formulated. Section 2.1 discusses the objective function for the maximum production rate. The constraints are discussed in Section 2.2. Finally, Section 2.3 formulates minimization of the cost as a multi-objective optimization problem.

2.1 Objective function for maximum production rate

Consider the multipass turning of a cylindrical work piece of length L and initial diameter of D_0, the final diameter being D_f. Maximizing the production rate is equivalent to minimizing the total production time per component T_P, which is expressed as:

$$T_p = T_{tR} + \frac{t_c T_{tR}}{T_r} + T_{tF} + \frac{t_c T_{tF}}{T_f} + T_L + t_{ts} \tag{1}$$

where T_{tR} is the total cutting time of rough machining, t_c the time required for changing a tool, T_r the tool life for rough machining, T_{tF} the total cutting time of finish machining, T_f the tool life for finish machining, T_L the loading and unloading time, and t_{ts} is the tool setting time. The total cutting time for rough machining is obtained as the summation of the cutting times for m roughing passes, i.e.:

$$T_{tR} = \sum_{i=1}^{m} t_{r_i} = \sum_{i=1}^{m} \frac{\pi L D_{i-1}}{v_{R_i} f_{R_i}}$$

where v_{R_i} and f_{R_i} are the cutting speed and feed, respectively, at the ith roughing pass and D_{i-1} is the work piece diameter at the beginning of that pass.

An interesting observation is that some researchers have employed an unequal depth of cut strategy for multipass turning process. However, in most of the cases, it seems reasonable to take equal depths of cut, which is a usual shop floor practice. This is because, compared to the cutting speed and feed, the depth of cut has a much lesser influence on tool life. Attainment of the optimum solution with unequal depths of cut in roughing passes may be due to the presence of multiple optimal solutions, which the traditional methods fail to capture. For example, a close look at Table 1 of Gupta et al. [6] and some calculations reveal the presence of multiple solutions. It is clearly seen that equal depths of cut would have given the same cost. Thus, to simplify the computations, an equal depth of cut strategy is adopted here. In Section 6, a simple method is proposed for an estimation of the deviation of an equal depth optimum solution with an unequal depth solution.

Let d_R be the depth of cut in each pass. Then:

$$\sum_{i=1}^{m} D_{i-1} = D_0 + (D_0 - 2d_R) + \ldots + D_0 - 2\sum_{i=1}^{m-1} d_R$$
$$= mD_0 - m(m-1)d_R$$

Thus, the total time in roughing is:

$$T_{tR} = \frac{\pi L}{v_R f_R}(mD_0 - m(m-1)d_R) \tag{2}$$

The total cutting time for a finishing pass is given by:

$$T_{tF} = \frac{\pi L}{v_F f_F}\left(D_f + 2d_F\right) \tag{3}$$

The total tool setting time is given as:

$$t_{ts} = (m+1)t_s \tag{4}$$

where t_s is the setting time for each pass.

The tool life for roughing and finishing are obtained by the famous Taylor's tool life equation, i.e.:

$$T_r = \frac{C}{v_R^p f_R^q d_R^r}, T_f = \frac{C}{v_F^p f_F^q d_F^r} \tag{5}$$

2.2 Machining constraints

Minimization of the total production time per component is carried out by imposing the following constraints:

– Tool life constraints:

$$T_{\min} \leq T_r,\ T_f \leq T_{\max} \tag{6}$$

– Surface finish constraint:

$$\frac{f_F^2}{8R} \leq R_{t_{\max}} \tag{7}$$

– Cutting force constraints:

$$\begin{aligned} &k f_R^{\alpha} d_R^{\beta} \leq F_{\max} \\ &k f_F^{\alpha} d_F^{\beta} \leq F_{\max} \end{aligned} \tag{8}$$

– Machine power constraints:

$$\begin{aligned} &\frac{k f_R^{\alpha} d_R^{\beta} v}{6120\eta} \leq P_{\max} \\ &\frac{k f_F^{\alpha} d_F^{\beta} v}{6120\eta} \leq P_{\max} \end{aligned} \tag{9}$$

– Geometric relation:

$$D_0 - 2md_R - D_f - 2d_F = 0 \tag{10}$$

– Variable bounds:

$$\begin{aligned} &v_{\min} \leq v_R,\ v_F \leq v_{\max} \\ &f_{\min} \leq f_R, f_F \leq f_{\max} \\ &d_{\min} \leq d_R,\ d_F \leq d_{\max} \end{aligned} \tag{11}$$

– Variable type:

m is an integer (12)

The meaning of the variables are as outlined in the Nomenclature section at the start of this paper.

Table 1 Data used for the example

D_0=50 mm	L=300 mm	$v_{F_{\max}} = 500$ m/min	$v_{F_{\min}} = 5$ m/min
$v_{R_{\max}} = 500$ m/min	$v_{R_{\min}} = 5$ m/min	$f_{F_{\max}} = 0.9$ mm/rev	$f_{F_{\min}} = 0.1$ mm/rev
$f_{R_{\max}} = 0.9$ mm/rev	$f_{R_{\min}} = 0.1$ mm/rev	$d_{F_{\max}} = 3.0$ mm	$d_{F_{\min}} = 1.0$ mm
$d_{R_{\max}} = 3.0$ mm	$d_{R_{\min}} = 1.0$ mm	$F_{\max}$=200 kgf	$P_{\max}$=5 kW
$T_{r_{\max}} = 45$ min	$T_{r_{\min}} = 25$ min	R=1.2 mm	t_s=0.51 min/pass
$T_{f_{\max}} = 45$ min	$T_{f_{\min}} = 25$ min	C_o=\$ 0.5/min	η=0.85
$R_{t_{\max}} = 10$ μm	C=6×10^{11}	t_c=1.5 min/edge	
k=108	p=5	T_L=0.75 min/piece	
α=0.75	q=1.75		
β=0.95	r=0.75		

2.3 Minimization of the production cost

The production cost of a component is given by:

$$F_c = C_o T_P + C_t \left(\frac{T_{tR}}{T_r} + \frac{T_{tF}}{T_f} \right) \tag{13}$$

where C_o is the operating cost per minute and C_t is the tool cost. The operating cost consists of overheads, labor, coolant, and electricity costs. The term associated with C_t is basically the fraction of the tool being consumed in the production of one piece. Denoting it by F_t, the production cost can be expressed as:

$$F_c = C_o T_P + C_t F_t \tag{14}$$

From the above expression, it is observed that, for minimizing F_c, both T_p and F_t have to be minimized. Thus, the minimization of cost problem can be converted into the following multi-objective problem:

$$\begin{aligned} &\text{minimize } T_p \text{ and } F_t \text{ subject} \\ &\text{to the constraints of Section 2.2} \end{aligned} \tag{15}$$

The above problem does not require the values of C_o and C_t. As the minimizations of T_p and F_t are two conflicting objectives, one can obtain various Pareto-optimal solutions. The best amongst the Pareto-optimal solutions can be chosen at a later stage with known cost data.

The theoretical minimum possible value of F_t is zero, corresponding to an infinite tool life. Thus, Pareto-optimal solutions will cover a range of F_t from 0 to that corresponding to the minimum possible T_p. However, depending on the value of C_o and C_t, some of these solutions will increase T_p and F_c simultaneously, which is undesirable. A condition when the increase in T_p would reduce the cost can be derived as follows.

Writing Eq. 14 in differential form:

$$\mathrm{d}F_c = C_o \mathrm{d}T_p + C_t \mathrm{d}F_t \tag{16}$$

The requirement $\mathrm{d}F_c<0$ leads to:

$$\frac{\mathrm{d}T_p}{\mathrm{d}F_t} > -\left(\frac{C_t}{C_o} \right) \tag{17}$$

The left-hand side of the above inequality is the Lagrange multiplier λ associated with an equality constraint (F_t=a prescribed value) corresponding to the optimization problem of minimizing T_p with the constraints of Section 2.2 along with this equality constraint. The prescribed value in the equality constraint is the value of F_t at which the condition is being checked. Note that the Lagrange multiplier λ is a negative quantity. Therefore, the F_t should be lowered only until the magnitude of λ is less than C_t/C_o. With the rough estimate of the costs known at a lower level, only those Pareto-optimal solutions that do not violate this condition need to be generated.

Equation 16 can also be written as:

$$\mathrm{d}F_c = C_o \left(1 + \frac{C_t}{C_o} \frac{\mathrm{d}F_t}{\mathrm{d}T_p} \right) \mathrm{d}T_p = C_o \left(1 + \frac{C_t}{C_o \lambda} \right) \mathrm{d}T_p \tag{18}$$

The expression in the parentheses should be negative in order to reduce the production cost at the expense of some increase in the production time. With this expression, one can assess the difference between the minimum cost and the cost corresponding to the minimum production time. If the ratio of the tool cost to the operating cost is very small, the difference between the two costs will also be small.

3 Optimization methodology

In this work, a real-parameter genetic algorithm (RGA) and sequential quadratic programming (SQP) have been applied in succession for minimizing the production time in the multipass turning process. An RGA is very efficient in reaching up to nearly global optima. With an initial population size of 50, the RGA is run four times up to 100 generations (or until convergence based on the average fitness value of the population) for obtaining an initial estimate, treating the number of roughing passes as real. Taking the best solution of the RGA, SQP carries out the local search in order to find out the global optimum. If the number of roughing passes *m* turns out to be a non-integer value, two runs of SQP are executed, one with the nearest lower integer value and the other with the nearest higher integer value. The best solution between these two runs is selected. SQP also provides the Lagrange multipliers associated with the constraints, which cannot be obtained by genetic algorithms in an obvious manner. SQP is a well-established traditional optimization technique and has been described in many textbooks [22, 23]. RGA is of recent origin [24] and, here, some techniques have been used to make it more effective.

RGA uses real numbers instead of binary and, hence, is suitable for a continuous search space. It does not suffer from Hamming cliffs and achieves sufficient precision with reasonable population sizes [24]. It operates on the population of potential solutions by the principle of the “survival of the fittest.” The initial population is random and the population keeps on evolving towards betterment in successive generations. In each generation, the population is operated on by three main operators, *reproduction*, *crossover*, and *mutation* to create a new population. If no significant improvement in the average fitness value of the population is observed for five successive generations, convergence is assumed. The three operators are described next.

1. *Reproduction*
   In reproduction, good solutions in a population are probabilistically assigned a larger number of copies

and mating pools are formed. There are a number of ways of carrying out reproduction. Here, the proportionate reproduction operator is used, in which copies of a member is proportional to its fitness value. The fitness value is calculated from a fitness function $F(x)$, which is related to the objective function. The higher the value of the fitness function, the closer the objective function value to the desired objective. The optimization problems used in the present work are of minimization type. Hence, the following fitness function has been used:

$$F(x) = \frac{1}{1 + f(x)} \tag{19}$$

where $f(x)$ is the objective function. The members of the reproduced population are called parents and are used for the next genetic operation, namely, crossover.

2. *Crossover*
In the crossover operation, new members are created by exchanging the information between two parent members. In a binary-coded GA, generally, a single-point crossover is used. Two strings are selected at random and crossed at a random site to generate the new offspring (children). In RGA, the term "crossover" is really a misnomer. In the present work, a simulated binary crossover operator (SBX) [24] is used, which has a similar search power to that in a single-point crossover in a binary-coded GA. This works as follows.
Choose a random number $u_i \in [0, 1]$. Calculate:

$$\beta = \begin{cases} (2u_i)^{\frac{1}{\eta_c + 1}} & \text{if } u_i \leq 0.5 \\ \left(\frac{1}{2(1 - u_i)}\right)^{\frac{1}{\eta_c + 1}} & \text{otherwise} \end{cases} . \tag{20}$$

where η_c is a crossover index, which is a non-negative real number. A large value of η_c gives a higher probability for creating a "near-parent" solution and a small value of η_c allows distant solutions to be selected as offspring. Offspring are given by:

$$\begin{aligned} x_i^{c1} &= 0.5\left[(1 + \beta)x_i^{p1} + (1 - \beta)x_i^{p2}\right] \\ x_i^{c2} &= 0.5\left[(1 - \beta)x_i^{p1} + (1 + \beta)x_i^{p2}\right] \end{aligned} \tag{21}$$

where x_i^{c1} and x_i^{c2} denote the ith variables of the child members' chromosomes and x_i^{p1} and x_i^{p2} are the ith variables of the parent members. The crossover operation is performed with a crossover probability of p_c. In the present work, based on numerical experiments and heuristics, the crossover index η_c is varied in the range 40–65, and the crossover probability is varied in the range 1–0.5 uniformly up to 100 generations. This is done in order to explore the entire domain and to generate significantly different members in the early generations. In the later generations, only near-parent solutions are generated so as not to destroy the good solutions.

3. *Mutation*
Mutation provides a local perturbation in order to provide diversity to the population and reduce the possibility of being trapped in a local optimum. In the binary-coded GA, mutation is carried out by altering one or more bits in the chromosome string. Here, a mutation operator based on polynomial mutation [24] is used. Accordingly, the mutated value y_i of x_i is given by:

$$y_i = x_i + \left(x_i^u - x_i^l\right)\bar{\delta}_i \tag{22}$$

Here, x_i^{u} and x_i^{l} are the upper and lower bound values of the ith variable, respectively. The parameter $\bar{\delta}_i$ is given by:

$$\bar{\delta}_i = \begin{cases} (2r_i)^{\frac{1}{\eta_m + 1}} - 1, & \text{if } r_i < 0.5 \\ 1 - [2(1 - r_i)]^{\frac{1}{\eta_m + 1}}, & \text{if } r_i \geq 0.5 \end{cases} \tag{23}$$

where η_{m} is the mutation operator and r_i is a random number in [0, 1].
In the present work, mutation operations are performed with the mutation index η_{m} which is varied from 200–150, and the mutation probability p_{m} is varied from 0–0.5 up to 100 generations. This is to make the effect of mutation negligible in early generations. In the later generations, the mutation starts producing diversity in the population and can perform the local search.

The optimization procedure will be demonstrated by means of an example in Section 5. After obtaining the solution for the minimum production time, SQP is used to obtain Pareto-optimal solutions by solving the minimization of the production time problem with the additional equality constraint of $F_{\mathrm{t}}=F_{\mathrm{t}}^*$, a prescribed value). Different Pareto-optimal solutions are obtained by gradually reducing F_{t}^* and noting the Lagrange multiplier corresponding to the equality constraint. The search for Pareto-optimal solutions is stopped as soon as the magnitude of the Lagrange multiplier crosses $C_{\mathrm{t}}/C_{\mathrm{o}}$. If even a rough estimate is not available for $C_{\mathrm{t}}/C_{\mathrm{o}}$, then F_{t}^* is reduced up to 0. Usually, with 7–8 Pareto-optimal solutions, a continuous curve representing the entire set of non-dominated solutions can be constructed.

4 A method for higher-level decision

In this section, a linear programming model is proposed for choosing the best among the various Pareto-optimal solutions. Corresponding to each Pareto-optimal solution, the production cost F_c can be calculated from Eq. 14, since at a higher level, C_t and C_o will be known. The proposed method minimizes the overall cost of the production subject to a time constraint. This is illustrated with an example of machining two different types of components on a single machine. The procedure can be easily extended to the case when several types of components are machined on the same machine.

Suppose an industry requires n_1 number of components of type 1 and n_2 number of components of type 2 in a total time T_{tot}. The minimum value of T_{tot} corresponds to the time when both components are machined with cutting parameters providing the maximum production rate. The maximum value of T_{tot} corresponds to the time when both components are machined with cutting parameters providing the minimum production cost. The optimization problem is:

$$\begin{array}{ll} \text{minimize} & n_1(F_c)_1 + n_2(F_c)_2 \\ \text{subject to} & n_1(T_p)_1 + n_2(T_p)_2 \leq T_{tot} \end{array} \tag{24}$$

where $(T_p)_1$ and $(T_p)_2$ are the individual times taken to produce components of type 1 and type 2, respectively. The cost of the production of individual components can be approximated as:

$$(F_c)_1 = A_1 - B_1(T_p)_1, (F_c)_2 = A_2 - B_2(T_p)_2 \tag{25}$$

This is a reasonably close approximation in the vicinity of the minimum production time. Now, the problem in Eq. 24 can be modeled as the following linear programming problem:

$$\begin{array}{l} \text{minimize } n_1B_1(T_p)_1 + n_2B_2(T_p)_2 \\ \text{subject to } n_1(T_p)_1 + n_2(T_p)_2 \leq T_{tot,} \\ \{(T_p)_1\}_{\text{max production rate}} \leq (T_p)_1 \leq \{(T_p)_1\}_{\text{min production cost,}} \\ \{(T_p)_2\}_{\text{max production rate}} \leq (T_p)_2 \leq \{(T_p)_2\}_{\text{min production cost,}} \end{array} \tag{26}$$

After solving the above optimization problem, the updated values of B_1 and B_2 can be obtained at the solution and the procedure can be repeated if significantly different values are obtained. Usually, the solution converges in one or two iterations. Having obtained the production times, the process parameters are found by solving the inverse problem using SQP.

5 Examples

In this section, examples have been used to illustrate the proposed methods. The data for the examples is provided in Table 1, which is the same as that used by Gupta et al. [6]. Section 5.1 compares the performance of the proposed optimization scheme (a combination of an RGA and SQP) with RGAs alone and the method of Gupta et al. [6]. Gupta et al. used a method known as "optimal sub-division" of the depth of cut. In this method, an "unequal depth of cut" strategy is adopted. The problem is solved in two phases. In the first phase, solutions corresponding to separate minimum costs for the individual rough passes and finish passes are determined and tabulated for various fixed values of the depth of cut. In the second phase, the optimal number of passes, the optimal subdivisions of depths of cut for different passes, and the minimum total production costs are determined using an integer programming model. With finer sub-divisions, this method is expected to give a nearly optimal solution, albeit, with increased computational time. Gupta et al. have reported minimum production costs lower than those observed by Shin and Joo [5]. In view of these observations, it was considered appropriate to compare the results of the proposed methodology with those of Gupta et al. [6]. Pareto-optimal solutions are shown in Section 5.2. Section 5.3 illustrates a procedure for a higher-level decision.

5.1 Comparison of the proposed optimization scheme with other schemes

A comparison of the proposed scheme with the results of Gupta et al. is made in Table 2. For this purpose, the cost of production is minimized by taking the objective function as Eq. 13. Tool cost C_t is taken as \$2.5. Columns 4 and 5 of the table show that the minimum production costs obtained from the proposed scheme are always lower than those reported by Gupta et al. [6]. It is to be noted that, while calculating the cutting speeds at each pass, Gupta et al. used the same job diameter, although the job diameter keeps reducing after each machining pass. If only RGA is used with fixed η_c, η_m, p_c, and p_m, then the results are inferior to the proposed scheme and the computational time (≈10 seconds on a Pentium IV computer) was about three times the time taken in the proposed scheme. These results were obtained by running the code four times and it took up to 200 generations to converge. The fixed values of η_c, η_m, p_c, and p_m were decided based on the recommendation available in the literature [24] and a few numerical experiments. The implementation of the variable η_c, η_m, p_c, and p_m strategy improved the performance of the RGA; however, its performance is still inferior to the proposed scheme. Moreover, SQP in the proposed scheme provides the values of the Lagrange multipliers associated with the constraints, enabling us to perform a sensitivity analysis.

5.2 Pareto-optimal solutions

Consider the case of turning for a total depth of cut of 8 mm, for which the minimum total production time per piece is obtained as 4.266 min. For this production time, the fraction of the tool consumed per piece (F_t) is 0.0750. There is a possibility of reducing the production cost, albeit, with an increase in the total production time. Figure 1 depicts the Pareto-optimal solutions corresponding to the objective function of Eq. 15. The variation of the magnitude of the Lagrange multiplier, which is basically the rate of increase in the minimum total production time with respect to the reduction in F_t, is shown in Fig. 2. It is observed that the magnitude of this rate keeps on increasing with reducing F_t.

A Pareto-optimal solution for which the magnitude of the Lagrange multiplier is more than (C_t/C_o) is not of interest, as discussed in Section 2.3. Thus, for a tool cost of $2.5, only the Pareto-optimal solutions towards the right of the line $|\lambda|=5$ are of interest in Fig. 1. It is seen that there is only a slight variation in the production times among these solutions. Hence, one can conclude that, for this cost data, the minimum production cost and minimum production time models will yield almost the same results. This is also clear from Table 3, where it is seen that, for C_t/C_o of 5, the minimum cost of production is 2.320, same (up to 3 decimal places) as that corresponding to the minimum production time. The table clearly shows that the Pareto-optimal solutions corresponding to $|\lambda|>5$ increase the production cost and production time simultaneously. The same happens for $|\lambda|>20$ if $C_t/C_o=20$.

For the very high tool cost (say, $C_t/C_o=200$), all of the Pareto-optimal solutions towards the right of line $|\lambda|=200$ are dominant either in terms of the production rate or the production cost. Unlike in the case of low tool cost, there is a large amount of variation in the production times. Table 3 shows that, corresponding to the minimum production time, the cost of production is $9.629, much more than the minimum production cost of around $3.490. However, the production time corresponding to the minimum cost is about 44% higher than the minimum production time. Therefore, in this case, choosing the best among the Pareto-optimal solutions becomes very important.

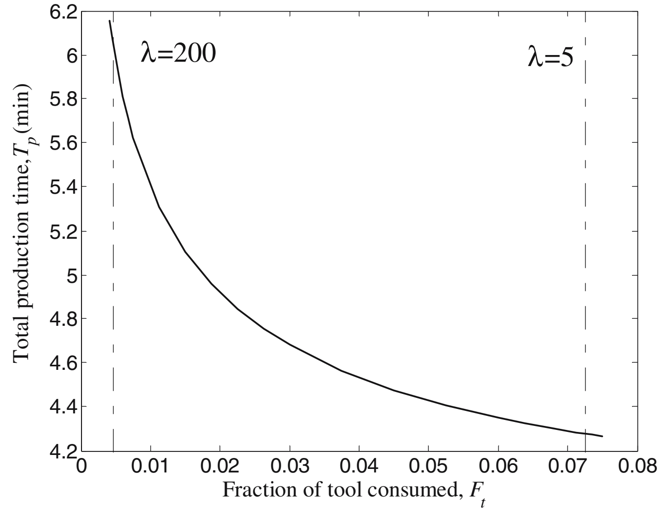

Fig. 1 Pareto-optimal solutions

5.3 A linear-programming-based higher-level decision

Let us consider the case of producing a shaft with two steps. Each step is 300-mm long. The depths of cut to be turned for each step are 6 mm and 8 mm, respectively. The methodology of Section 4 can be easily employed in this case, as the machining of each step can be treated as the machining of one component. When both steps are machined to provide maximum production rates, the time for machining the job is 7.684 min; however, the production cost per piece is $17.556. Depending on the production requirement, the production time can be increased to reduce the production cost. Table 4 shows some linear programming solutions for various available times on the machine. If the available time for a piece is 9 min, step 2 is machined with parameters providing the maximum production rate, but the time for machining step 1 is increased. This provides an overall cost of $13. For the available time of 11.5 min, step 1 is machined to provide

Table 2 Comparison of the results obtained by the two real-parameter genetic algorithms (RGAs) (with and without variable η_c, η_m, p_c, and p_m) and the proposed scheme with the results of Gupta et al. [6]

Total depth of cut (mm)	Production cost per piece ($)			
	RGA (η_c=50, η_m=200, p_c=0.7, p_m=0.01)	RGA (variable η_c, η_m, p_c, and p_m)	Proposed scheme (RGA+SQP)	Gupta et al. [6]
6.0	2.094	1.982	1.864	1.940
8.0	2.419	2.368	2.320	2.481
8.5	2.467	2.464	2.369	2.551
9.0	2.631	2.471	2.418	2.611
9.5	2.909	2.861	2.711	3.005
10.0	2.874	2.821	2.754	3.022

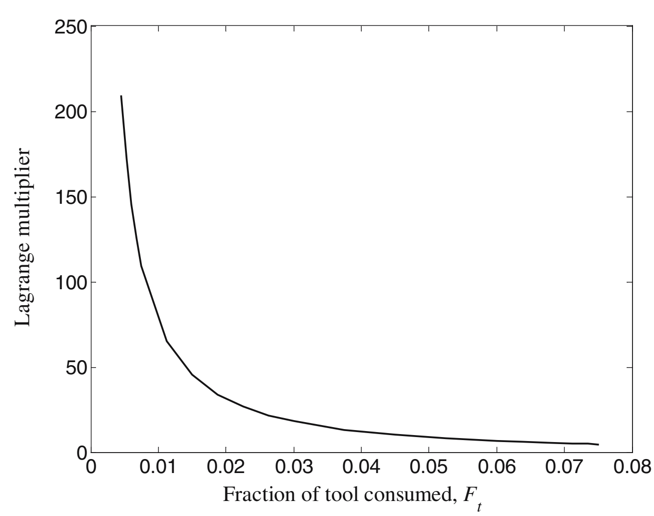

Fig. 2 Variation of Lagrange multiplier with F_t

Table 3 Different Pareto-optimal minimum production times and corresponding costs at different C_t/C_o ratios

F_t	T_p (min)	$\lvert\lambda\rvert$	Production cost per piece ($)		
			C_t/C_o=5	C_t/C_o=20	C_t/C_o=200
0.0750	4.266	4.75	2.320	2.882	9.629
0.0735	4.274	4.90	2.320	2.872	9.487
0.0713	4.285	5.15	**2.320**	2.855	9.267
0.0675	4.305	5.62	2.321	2.828	8.902
0.0600	4.351	6.75	2.326	2.775	8.176
0.0450	4.477	10.32	2.351	2.688	6.738
0.0300	4.681	18.13	2.415	2.641	5.341
0.0263	4.755	21.70	2.443	**2.640**	5.002
0.0225	4.846	26.63	2.479	2.648	4.673
0.0075	5.623	109.56	2.830	2.886	3.562
0.0060	5.812	145.30	2.921	2.966	3.506
0.0041	6.155	232.97	3.088	3.119	**3.490**
0.0040	6.191	243.85	3.105	3.135	3.497

the minimum cost. The overall cost in this case is $7.917. For the available time of 13 min, both of the steps are machined at their minimum possible costs. The total production time is 12 min, indicating the idle time of 1 min/piece.

6 Unequal depth of cut versus equal depth of cut

In the present work, an equal depth of cut strategy is adopted. For estimating the difference with an unequal depth of cut strategy, a simple numerical procedure may be adopted. In this procedure, after the solution with an equal depth of cut d_r has been obtained, the SQP code is run three times for obtaining the minimum production time in a single-pass rough turning process with a fixed depth of cut. The depths of cut in the runs are kept equal at $0.9d_r$, $1.1d_r$, and d_r, respectively. Let the obtained minimum production time in these runs be x_1, x_2, and x_3, then the quantity:

$$\varepsilon = \frac{|x_1 + x_2 - 2x_3|}{0.2d_r x_3}(d_{\max} - d_r) \tag{27}$$

provides a measure of the expected relative improvement in the solution if an equal depth of cut strategy is adopted. If ε is very small, say 0.01, there is no need to go for an unequal depth of cut strategy.

As an example, consider the solution for the turning of a total depth of cut of 8 mm. The solution for the maximum production rate is: ν_r=117 m/min, f_r=0.71 mm/rev, d_r=2.5 mm, ν_f =152 m/min, f_f =0.31 mm/rev, d_f =3 mm, and m=2. In order to assess the improvement in the solution if an unequal depth of cut strategy is employed, the minimum times for a single-pass rough turning process with depths of cut of 2.25 mm, 2.75 mm, and 2.5 mm are calculated. These values, called x_1, x_2, and x_3, respectively, are 0.5412 min, 0.6579 min, and 0.5996 min. The value (x_1+x_2−2x_3) is -1×10^{-4} min, the negative sign indicating the tendency of improvement if an unequal depth of cut strategy is adopted. However, Eq. 27 provides ε=0.00017, which is much smaller than 0.01. Thus, for the example problem studied in this paper, an equal depth of cut strategy is appropriate.

Table 4 Linear programming solutions

T_{tot} (minute)	Production time per step (min)		Production cost per step ($)	
	Step 1	Step 2	Step 1	Step 2
7.684	3.418	4.266	7.927	9.629
9	4.734	4.266	3.371	9.629
11.5	4.889	6.61	2.831	5.086
13	4.889	7.12	2.831	4.096

7 Conclusions

In this paper, an optimization methodology is proposed for the optimization of a multi-pass turning process. A combination of a real-parameter genetic algorithm (RGA) and sequential quadratic programming (SQP) minimizes the time of production, along with the satisfaction of several constraints. The performance of the RGA has been improved by the continuous variation of certain parameters of the RGA through generations (iterations). A major advantage of the proposed methodology is that various Pareto-optimal solutions can be generated without the knowledge of the costs involved. With the availability of cost data, the best solution can be chosen at a higher level. For that purpose, a linear programming model may be used. The entire methodology is demonstrated with the help of examples. The minimum production costs obtained from the present optimization algorithm are lower than that reported by Gupta et al. [6]. In this work, an equal depth of cut strategy is employed. A procedure has been suggested to assess the change in the solution if an unequal depth of cut strategy is employed.

The main focus of the present paper is on showing the effectiveness of the proposed optimization methodology. Therefore, simple relations have been used for the estimation of the tool life and surface roughness. In real-life practice, these may be computed with the help of neural networks [25, 26], the optimization procedure remaining the same. Some of the techniques employed in this paper can be extended to the optimization of other machining processes.

References

1. Gilbert WW (1950) Economics of machining. Machining theory and practice. American Society of Metals, Cleveland, Ohio, pp 465–485
2. Ermer DS (1971) Optimization of the constrained machining economics problem by geometric programming. J Eng Ind 93:1067–1072
3. Petropoulos PG (1973) Optimal selection of machining rate variables by geometric programming. Int J Prod Res 11(4): 305–314
4. Lambert BK, Walvekar AG (1978). Optimization of multipass machining operations. Int J Prod Res 16(4):259–265
5. Shin YC, Joo YS (1992) Optimization of machining conditions with practical constraints. Int J Prod Res 30(12):2907–2919
6. Gupta R, Batra JL, Lal GK (1995) Determination of optimal subdivision of depth of cut in multipass turning with constraints. Int J Prod Res 33(9):2555–2565
7. Yeo SH (1995) A multipass optimization strategy for CNC lathe operations. Int J Prod Econ 40(2–3):209–218
8. Lee BY, Tarng YS (2000) Cutting parameter selection for maximizing production rate or minimizing production cost in multistage turning operations. J Mater Process Technol 105: 61–66
9. Wang J (1998) Computer-aided economic optimization of end-milling operations. Int J Prod Econ 54(3):307–320
10. Shabtay D, Kaspi M (2002) Optimization of the machining economics problem under the failure replacement strategy. Int J Prod Econ 80(3):213–230
11. Abuelnaga AM, El-Dardiry EA (1984) Optimization methods for metal cutting. Int J Mach Tool Des Res 24(1):11–18
12. Chen MC, Tsai DM (1996) A simulated annealing approach for optimization of multipass turning operations. Int J Prod Res 34 (10):2803–2825
13. Onwubolu GC, Kumalo T (2001) Optimization of multipass turning operations with genetic algorithms. Int J Prod Res 39 (16):3727–3745
14. Amiolemhen PE, Ibhadode AOA (2004) Application of genetic algorithms—determination of the optimal machining parameters in the conversion of a cylindrical bar stock into a continuous finished profile. Int J Mach Tools Manuf 44:1403–1412
15. Shunmugam MS, Reddy SVB, Narendran TT (2000) Selection of optimal conditions in multi-pass face-milling using genetic algorithm. Int J Mach Tools Manuf 40:401–414
16. Saravanan R, Ashokan P, Vijaykumar K (2003) Machining parameters optimisation for turning cylindrical stock into a continuous finished profile using genetic algorithm (GA) and simulated annealing (SA). Int J Adv Manuf Technol 21(1):1–9
17. Davim JP, Antonio CAC (2001) Optimization of cutting conditions in machining of aluminium matrix composites using a numerical and experimental model. J Mater Process Technol 112:78–82
18. Saravanan R, Sachithanandam M (2001) Genetic algorithm (GA) for multivariable surface grinding process optimization using a multi-objective function model. Int J Adv Manuf Technol 17(5):330–338
19. Baskar N, Saravanan R, Ashokan P, Prabhaharan G (2004) Ants colony algorithm approach for multi-objective optimisation of surface grinding operations. Int J Adv Manuf Technol 23:311–317
20. Tosun N, Ozler L (2004) Optimisation of hot turning operations with multiple performance characteristics. Int J Adv Manuf Technol 23(11–12):777–782
21. Cauchick-Miguel PA, Coppini NL (1996) Cost per piece determination in machining process: an alternative approach. Int J Mach Tools Manuf 36(8):939–946
22. Arora JS (1989) Introduction to optimum design. McGraw-Hill, New York
23. Rao SS (1984) Optimization: theory and applications, 2nd edn. Wiley, New York
24. Deb K (2001) Multiobjective optimization using evolutionary algorithms. Wiley, Singapore
25. Kohli A, Dixit US (2004) A neural network based methodology for prediction of surface roughness in turning process. Int J Adv Manuf Technol 25(1–2):118–129
26. Ojha DK, Dixit US (2005) An economic and reliable tool life estimation procedure for turning. Int J Adv Manuf Technol 26 (7–8):726–732