Journal of
MECHANICAL and ENERGY
ENGINEERING

ISSN: 2544-0780 | e-ISSN: 2544-1671
Vol. 3(43) | No. 1 | April 2019 | pp. 13-24
DOI: 10.30464/jmee.2019.3.1.13

GENETIC ALGORITHMS SOLUTION TO THE SINGLE-OBJECTIVE MACHINING PROCESS OPTIMIZATION TIME MODEL

Patrick AMIOLEMHEN[1*], Joshua ESEIGBE[1]

[1] Department of Production, Faculty of Engineering, University of Benin, P.M.B 1154, Benin City, Nigeria,
e-mail – patrick.amiolemhen@uniben.edu

(Received 21 February 2019, Accepted 29 March 2019)

Abstract: Minimum Production Time model of the machining process optimization problem comprising seven lathe machining operations were developed using Genetic Algorithms solution method. The various cost and time components involved in the minimum production cost and minimum production time criteria respectively, as well as all relevant technological/practical constraints were determined. An interactive, user-friendly computer package was then developed in Microsoft Visual Basic.Net environment to implement the developed models. The package was used to determine optimal machining parameters of cutting speed, feed rate and depth of cut for the seven machining operations with twenty-three technological constraints in the conversion of a cylindrical metal bar stock into a finished machined profile. The result of the single-objective machining process optimization models shows that the minimum production time is 21.84 min.

Keywords: production time, optimization, machining model, genetic algorithms, development

1. INTRODUCTION

Taylor first realized the importance of machining optimization [1] in his pioneering work "On the Art of Cutting Metals". Since then, optimization of machining processes remains an ongoing activity, as evidenced by the optimization studies that were carried out over the last century [2]. In machining process optimization criteria are usually based on three objectives of: the minimum total cost per component; the maximum production rate; and the maximum profit-rate criterion [3-5].

Selection of cutting parameters is usually a difficult task, where the following aspects are required: knowledge of machining; empirical equations relating the tool life, forces, power, surface finish, etc., to develop realistic constraints; specification of machine tool capabilities; development of an effective optimization criterion; and knowledge of mathematical and numerical optimization techniques [6, 7].

Several optimization techniques have been employed for machining process optimization since the introduction of computers to machining systems.

Linear programming was used for machining process optimization [8-10] developed a Nelder-Mead simplex method to determine the optimum machining conditions. In most of the works above, the problems were simplified by considering only one or two variables such as the cutting speed and feed rate, in order to optimize the economical machining performance. They assumed that a single cut can achieve the required maximum metal removal rate (MRR).

Geometric Programming (GP), one of the non-linear optimization techniques, has been extensively adopted [11], in which the constrained models are converted into a dual geometric programming formulation and then into an unconstrained nonlinear programming formulation.

Traditional non-linear optimization techniques have also been extensively used. Wen et al [12] adopted the successive quadratic programming method to solve the non-linear off-line optimization scheme for a surface grinding process. Xiao et al [13] applied an iterative Newton's method for a non-linear internal cylindrical plunge grinding process. Jha and Hornik [14] used the generalized reduced gradient method to optimize the tool geometry and cutting conditions in plain milling process.

Sonmez and Baykasoglu [6] outlined the development of an optimization strategy to determine the optimum cutting parameters for multi-pass milling operations such as plain milling. The developed strategy was based on the maximum production rate and incorporated eight technological constraints. The optimum number of passes was determined via dynamic programming and the optimum values of cutting conditions were found based on the objective function by using the geometric programming technique. Jang [15] developed a unified optimization approach for the selection of the machining parameters (cutting speed, feed, and depth of cut) to provide the maximum metal removal rate. Ermer [16] analyzed a nonlinear objective function with inequality constraints to determine the optimal machining conditions by geometric programming. Lambert and Walvekar [17] developed a dynamic

programming model for the multi-pass turning operation under constraints of force, cutting power and surface finish to determine values of machining variables and minimum production cost. They considered two-pass turning operations. Shin and Joo [18] presented a model for the multi-pass turning operation using a fixed machining interval. They used dynamic programming for the selection of depth of cut for individual passes. The final finish pass was fixed based on the minimum allowable depth of cut and the remaining depth of cut was divided into a number of rough passes of equal sizes to obtain the minimum total cost. Lee et al [19] developed a fuzzy non-linear programming model to optimize machining operations. The model was used to select the tool-holder, insert and cutting conditions (feed, speed and depth of cut). They used dynamic programming to select optimal cutting conditions.

The traditional non-linear optimization techniques are mostly gradient-based and possess many limitations in application to today's complex machining models. Secondly, they cannot deal with integer/discrete design variables directly; integer design variables have to be approximated from continuous values. Therefore, one must resort to non-systematic optimization techniques, such as Evolutionary Algorithm.

Groover [20] used Monte Carlo simulation to study the machining economic problem considering tool wear and surface roughness. Dereli et al [21] explained the application of Genetic Algorithms (GAs) for determination of optimal sequence of machining operations based on either minimum tool change or minimum tool traveling distance or safety. Srikanth and Kamala [22] applied a Real Coded Genetic Algorithm (RCGA) to determine minimum surface roughness values, and their corresponding optimum cutting parameters, for turning process. But they only considered four constraints. Saravanan et al [23] showed an optimization method for cutting conditions in continuous profile machining in order to minimize the production cost. For the optimization method, they used Genetic Algorithms (GAs) and Simulated Annealing (SA) and compared the results. Amiolemhen and Ibhadode [24] applied Genetic Algorithms (GAs) to determine the optimal machining parameters in the conversion of a cylindrical bar stock into a continuous finished profile using the minimum production cost criterion. They developed single and multi-pass models for seven machining processes involved in continuous prolife machining.

Genetic Algorithms solution which has been used extensively in non-linear machining optimization problems [25-29], is the choice for this research work.

Hence, this paper employs Genetic algorithms to determine the minimum production time in the conversion of cylindrical bar stock into a continuous finished part. A user friendly and iterative computer package developed in the Microsoft Visual Basic.Net environment is employed to determine the optimal machining parameters for machining a continuous finished profile from bar stock.

2. METHODOLOGY

2.1. Machining process optimization models

Mathematical models have been developed for the following machining processes involved in the conversion of a cylindrical bar stock into a continuous finished part. These machining processes are: facing; turning; centreing; drilling; boring; chamfering; and parting. The models for the above machining operations are summarized in Table 1. The time model for each cutting operation is minimized subject to the constraints specified by the given equations.

These equations are given below:

– bounds on cuttings peed:

$$\text{roughing: } v_{rL} \le v_r = \frac{\pi D N_r}{1000} \le v_{rU}, \quad (1)$$

$$\text{finishing: } v_{sL} \le v_s = \frac{\pi D N_s}{1000} \le v_{sU}, \quad (2)$$

– bounds on feed rate:

$$\text{roughing: } f_{rL} \le f_r \le f_{rU}, \quad (3)$$

$$\text{finishing: } f_{sL} \le f_s \le f_{sU}, \quad (4)$$

– bounds on depth of cut:

$$\text{roughing: } d_{rL} \le d_r \le d_{rU}, \quad (5)$$

$$\text{finishing: } d_{sL} \le d_s \le d_{sU}, \quad (6)$$

– tool Life constraint:

roughing: $T_L \le T_r = \dfrac{C_o}{v_r^{\alpha} f_r^{\beta} d_r^{\gamma}} \le T_U$, (7)

finishing: $T_L \le T_s = \dfrac{c_o}{v_s^{\alpha} f_s^{\beta} d_s^{\gamma}} \le T_U$, (8)

– cutting force constraint:

roughing: $F_r = k_f f_r^{\mu} d_r^{\nu} \le F_U$, (9)

finishing: $F_s = k_f f_s^{\mu} d_s^{\nu} \le F_U$, (10)

– cutting power constraint:

roughing: $P_r = \dfrac{k_f f_r^{\mu} d_r^{\nu} v_r}{60,000\,\eta} \le P_U$, (11)

finishing: $P_s = \dfrac{k_f f_s^{\mu} d_s^{\nu} v_s}{60,000\,\eta} \le P_U$, (12)

– chip-tool interface temperature constraint:

roughing: $Q_r = k_q v_r^{\tau} f_r^{\varphi} d_r^{\delta} \le Q_U$, (13)

finishing: $Q_s = k_q v_s^{\tau} f_s^{\varphi} d_s^{\delta} \le Q_U$, (14)

– dimensional accuracy constraint:

roughing: $DA_r = k_g v_r^{\chi} f_r^{\varsigma} d_r^{\psi} \le DA_U$, (15)

finishing: $DA_s = k_g v_s^{\chi} f_s^{\varsigma} d_s^{\psi} \le DA_U$, (16)

– stable cutting region constraint:

roughing: $SC_r = v_r^{\lambda} f_r d_r^{\upsilon} \ge SC$, (17)

finishing: $SC_s = v_s^{\lambda} f_s d_s^{\upsilon} \ge SC$, (18)

– surface finish constraint:

finishing: $SR_s = \dfrac{f_s^2}{8R} \le SR_U$, (19)

– miscellaneous constraints:

Finishing cutting speed: $v_s \ge 1.2 v_r$, (20)

Finishing Feed rate: $f_s \le 0.6 f_r$, (21)

Finishing depth of cut: $d_s \le 0.5 d_r$, (22)

Total depth of cut constraint: $d_s = d_t - n$, (23)

– bounds on number of rough cuts:

$$N_L = \frac{d_t - d_{sU}}{d_{rU}} \le n \le N_U = \frac{d_t - d_{sL}}{d_{rL}}. \quad (24)$$

Tab. 1. Multi-pass machining operations models

S/n	Machining operation	Time functions	Constraints
1	Facing	$T_{uf} = \left[\dfrac{\pi D^2}{2000 v_r f_r}\left(\dfrac{d_t}{d_r}\right)\right] + \left[t_c + \left(h_1\dfrac{D}{2} + h_2\right)\left(\dfrac{d_t}{d_r}\right)\right] + \dfrac{t_e}{T}\left[\dfrac{\pi D^2}{2000 v_r f_r}\left(\dfrac{d_t}{d_r}\right)\right]$	1, 3, 5, 7, 9, 11, 13
2	Turning	$T_{ut} = \left[\dfrac{\pi DL}{1000 v_r f_r} N_{rp} + \dfrac{\pi DL}{1000 v_s f_s}\right] + \left[t_c + (h_1 L + h_2)(N_{rp}+1)\right] + t_e\left\{\dfrac{1}{T_r}\left[\dfrac{\pi DL}{1000 v_r f_r} N_{rp}\right] + \dfrac{1}{T_s}\left[\dfrac{\pi DL}{1000 v_s f_s}\right]\right\}$	1 - 22
3	Centreing	$T_{uc} = \left[\dfrac{\pi DL}{1000 v_r f_r}\right] + \left[t_c + (h_1 L + h_2)\right] + \dfrac{t_e}{T}\left[\dfrac{\pi DL}{1000 v_r f_r}\right]$	1, 3, 5,7, 9, 11, 13
4	Drilling	$T_{ud} = \left[\dfrac{\pi DL}{1000 v_r f_r}\left(\dfrac{d_t}{d_r}\right)\right] + \left[t_c + (h_1 L + h_2)\left(\dfrac{d_t}{d_r}\right)\right] + \dfrac{t_e}{T}\left[\dfrac{\pi DL}{1000 v_r f_r}\left(\dfrac{d_t}{d_r}\right)^2\right]$	1, 3, 5, 7, 9, 11, 13
5	Boring	$T_{ub} = \left[\dfrac{\pi DL}{1000 v_r f_r} N_{rp} + \dfrac{\pi DL}{1000 v_s f_s}\right] + \left[t_c + (h_1 L + h_2)(N_{rp}+1)\right] + t_e\left\{\dfrac{1}{T_r}\left[\dfrac{\pi DL}{1000 v_r f_r} N_{rp}\right] + \dfrac{1}{T_s}\left[\dfrac{\pi DL}{1000 v_s f_s}\right]\right\}$	1 - 22
6	Parting	$T_{up} = \left[\dfrac{\pi D^2}{2000 v_r f_r}\right] + \left[t_c + (h_1 L + h_2)\right] + \dfrac{t_e}{T}\left[\dfrac{\pi D^2}{2000 v_r f_r}\right]$	1, 3, 5, 7, 9, 11, 13
7	Chamfering	$T_{uch} = 4\left[\dfrac{\pi DL}{1000 v_r f_r}\right] + \left[t_c + 4(h_1 L + h_2)\right] + 4\dfrac{t_e}{T}\left[\dfrac{\pi DL}{1000 v_r f_r}\right]$	1, 3, 5, 7, 9, 11, 13

2.2. Optimization by Genetic Algorithms

Outline of the Basic Genetic Algorithm

1. [**Start**] Generate random population of n chromosomes (suitable solutions for the problem)
2. [**Fitness**] Evaluate the fitness f(x) of each chromosome x in the population
3. [**New population**] Create a new population by repeating the following steps until the new population is complete:

– [**Selection**] Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected);
– [**Crossover**] With a crossover probability, P_c crossover the two parents to form two new offsprings (children). If no crossover was performed, offspring is the exact copy of parents;
– [**Mutation**] With a mutation probability, P_m mutate new offsprings at each locus (position in chromosome); and
– [**Accepting**] Place new offsprings in the new population.
– [**Replace**] Use new generated population for a further run of the algorithm.
– [**Test**] If the end condition is satisfied, **stop**, and return the best solution in current population.
– [**Loop**] Go to step **2**.

2.3. The Genetic Algorithms procedure

Generate initial population

a. Determine population size

The population size used in this work is, np = 20, in accordance with the recommendation of Schaffer [30].

b. Initialisation

The solution space of the population size, np = 20 is generated randomly between the bounds of each decision variable. In this work the decision variables are cutting speed, feed rate and depth of cut.

I. Choosing solution representation

The string of bits or genes in the chromosome could be binary, real integer number, etc [31]. In this work, binary string format of finite length was adopted.

II. Determination of chromosome lengths

The total length of each design variable represented in a binary string is determined as follows:

– Choosing level of precision

The level of precision or the number of decimal places of each decision variable, $p = 4$ was adopted.

– Evaluate integer parameter of each decision variable

The integer parameter is given as [31]:

$$c = \left(b_j - a_j\right) \times 10^p$$

where: p = level of precision or number of decimal places of the variable and ($b_j - a_j$) = range of domain of each of the variable.

– Estimate of chromosome (binary string) length

According to Gen and Cheng [31], if binary coding is used, the integer parameters of each variable always lie between:

$$2^{N_j - 1} < c \le 2^{N_j}$$

where: N_j = length of chromosome (binary string) of each design variable

III. Determination of the integer value of each chromosome

The required integer value of each chromosome is determined as follows [31]:

$$x' = \left(\frac{x_j - a_j}{b_j - a_j}\right)\left(2^{N_j} - 1\right)$$

where: x_j = the actual value of the decision variable, x' = integer value of the binary number, a_j = lower value of the range of the decision variable, and b_j = upper value of the range of the decision variable.

IV. Transformation of the integer values into binary strings

The transformation of the integer values of the decision variables into binary strings is done as transformation of real numbers from base 10 to base 2 as follows:

{bN,bN-1, ..., b1}

Evaluation of the initial population

a. Determination of values of objective functions

The values of the objective functions are determine by substituting feasible values of the decision variables into the various optimization models developed.

Objective function value , i = 1, 2, 3... np

b. Evaluation of fitness of each chromosome

Since the objective function is a minimization problem, the fitness function of the i^th^ solution is thus evaluated by:

$$f_i(x) = g_{max}(x) - g_i(x), \; i = 1, 2, 3 \ldots n_p$$

where: $g_{max}(x)$ is the maximum objective function value and $g_i(x)$ is the objective function value of the i^{th} solution.

Creation of a new population

After the transformation of the integer values into binary strings, Genetic Algorithms operators are applied. Here the three operators (reproduction, crossover, and mutation) are used.

a. Reproduction

The two chromosomes (strings) with best fitness and the second best fitness are allowed to live and produce offspring in the next generation, after evaluation. These chromosomes are the "elites chromosomes".

b. Selection and crossover

The cumulative probability is used to decide which chromosomes will be selected to crossover. The cumulative probability is calculated in the following steps:

I. Selection of pairs of chromosomes for mating

The Roulette wheel selection process was used selection and the cumulative probability, Cf_i, is used to decide which parents will be selected for mating. And, the Roulette wheel is constructed as follows:

Calculate fractional fitness (selection probability), Pf_i, for each chromosome:

$$Pf_i = \frac{f_i(x)}{\text{Fitness\_total}} = \frac{f_i(x)}{\sum_{i=1}^{n_p} f_i(x)}$$

Calculate the cumulative fitness (probability), Cf_i, for each chromosome:

$$Cf_i = \sum_{k=1}^{i} Pf_i, i = 1, 2, 3 \ldots np$$

The selection process was done by spinning the Roulette wheel n_p times and each time, a single chromosome is selected for a new population, such that $r \in [0, 1]$, and if $r \le Cf_i$, then select first chromosome; otherwise select the i^{th} chromosome ($2 \le i \le n_p$) such that $Cf_{i-1} < r \le Cf_i$.

II. Application of crossover operator to the selected pairs of chromosomes

The crossover probability used is, $P_c = 0.80$. Then, a random number was generated such that, $r \in (0, 1)$; and if $r < P_c$, then crossover is carried out otherwise it is left unchanged.

c. Application of mutation operator to the reproduced chromosomes

Mutation alters one or more genes with a probability equal to the mutation rate (of the order of 0.005 to 0.01). A random number is generated such that, r □ (0, 1); and if $r < Pm$, then that bit is complemented otherwise it is left unchanged.

d. Formation of a new population

After the mutation exercise, new strings are created which are then added to the two elite chromosomes from the initial population to form a new population.

Evaluation of final population

a. Decoding the newly formed population

The newly formed chromosomes after the mutation operation are usually decoded as follows:

$$x_i = a_j + \left(\sum_{i=1}^{N_j} b_i \times 2^i\right)\left(\frac{b_j - a_j}{2^{N_j} - 1}\right)$$

b. Evaluation of objective function values

The objective function values of the model being applied are determined using the newly formed population and then the results are checked for optimality.

Termination method

A new population is created as a result of completing one-iteration of the Genetic Algorithms. The iteration is terminated if optimum results are obtained; otherwise it is repeated until the maximum number of GA generation is reached [31].

In this work, the Genetic Algorithms procedure was terminated after 50 generations.

2.4. Implementation

The elements of the proposed models developed using Genetic Algorithm have been implemented in the software developed in Microsoft Visual Basic.Net environment and run on a Pentium 4 PC with 3.0 GHz Intel Processor and 2 GB of RAM. The values set for different parameters of the genetic algorithm are shown in Table 2.

Tab. 2. Genetic Algorithms parameters

Population size	20
No of population generation	50
Length chromosomes	49
Selection operator	Roulette Wheel
Crossover operator	One-point operator
Crossover probability	0.80
Mutation probability	0.01
Fitness measure	Single-obj. min

2.5. Illustrative example

An illustrative example has been adopted from [24], [29] to demonstrate the performance of the proposed models. Table 3 shows the data of the illustrative example.

Tab. 3. Data of Chen and Tseng [32] and Onwubolu and Kumalu [29]

v_{rL} = 90 m/min	v_{rU} = 500 m/min	v_{sL} = 90 m/min
v_{sU} = 500 m/min	f_{rL} = 0.1 mm/rev	$\alpha = 5$
$\tau = 0.40$	f_{sL} = 0.1mm/rev	f_{sU} = 1.0 mm/rev
d_{rL} = 1.0 mm	d_{rU} = 3.0 mm	d_{sL} = 1.0 mm
$\nu = 0.95$	$\mu = 0.75$	K_o = 0.5\$/min
K_t = 2.5 \$/min	T_L = 25 min	T_U = 45min
SR_U = 10μm	$\varsigma = 0.9709$	Q_U = 1000°C
h_2 = 0.3 min	F_U = 5.0 kgf	P_U = 200 kW
R = 1.2mm	$\eta = 0.85$	$C = 140$
$K_f = 108$	$K_q = 132$	d_{sU} = 3.0 mm
$\Phi = 0.2$	$h_1 = 7x10^{-4}$ min/mm	T_e = 1.5 min/edge
$\delta = 0.105$	f_{rU} = 1.0 mm/rev	$\beta = 1.75$
$C_o = 6x10^{11}$	$k_r = 100.66$	$X = -0.2848$
Tc = 0.75 min/piece	$\psi = 0.4905$	$\upsilon = -1$
$\gamma = 0.75$	$\lambda = 2$	

3. RESULTS AND DISCUSSION

Fig.1 contains the optimum results of the seven machining processes considered using the minimum production time model for the 50 population generations. This table also shows the optimum

cutting parameters of the seven machining processes considered and the overall production time per workpiece.

Fig. 2 shows the plots of maximum selection probability (fractional fitness) and corresponding minimum costs with respect to the number of generations. The fitness plot shows that the selection probability varies within the range of 0.063 – 0.119. The time plot shows that time of turning is about 9.9 min/piece from the 1st to the 5th generations. It then drops to about 8.7 min/piece from the 6th generation and remains constant at this value to the 47th generation. Thereafter the cost drops to about 8.32 min/piece from the 48th to the 50th generations. These plots show that in the neighbourhood of a drop in fractional fitness as the number of generations increase, there is a corresponding drop in the turning time. Between the 5th and 6th generations the fractional fitness drops from 0.095 to 0.071 when the time correspondingly drops from 9.9 min/piece to 8.32 min/piece. Similarly between the 47th and 48th generations, the fractional fitness drops from 0.095 to 0.070 while the production time correspondingly drops from 8.68 min/piece to 8.32 min/piece. These points of reduction in cost with respect to corresponding drops in fractional fitness relate to when the Genetic Algorithms solution is being reset by the crossover and mutation operators.

Fig. 3 shows the plot of minimum production time for the seven machining operations carried out on the workpiece as given by Table 11. The Figure shows that the production time drops rapidly from 38.02 min/piece from the 1st generation to 34.3 min/piece at the 2nd generation and 34.1 min/piece from the 3rd generation to 24.7 min/piece at the 4th generation, and then to 22.8 min/piece at the 7th generation. The time drops between the 1st and the 7th generations represent a time slope of 1.586 min/generation. Whereas that between the 7th generation (22.8 min/piece) to the 48th generation (28.1 min/piece) is 0.024 min/generation representing 66 times the time slope between the 1st and 7th generations. This goes to show how effective the GAs solution technique is, in quickly converging to the optimum value.

Table 4 shows the optimum cutting parameters and optimum machining time of the seven machining processes considered using the minimum production time model as well as the overall production time per workpiece.

Using the data supplied by Ibhadode [33], the developed models gave production time for the monoplex die shown in Fig. 5 as 415.13 min. The details of these results are shown in Table 5.

But, when the original Taylor's tool life was replaced with the modified Taylor's tool life, the production time became 360.25 min. These results are also shown in Table 5. Examination of the optimum solutions given in Table 5 has shown that for the two cases, the optimum production times for the monoplex die container are superior to the conventional recommended solutions (Ibhadode; 2009). While this was anticipated from the optimization analysis, it further confirms that the optimization models are reliable tools for application on the shop floor.

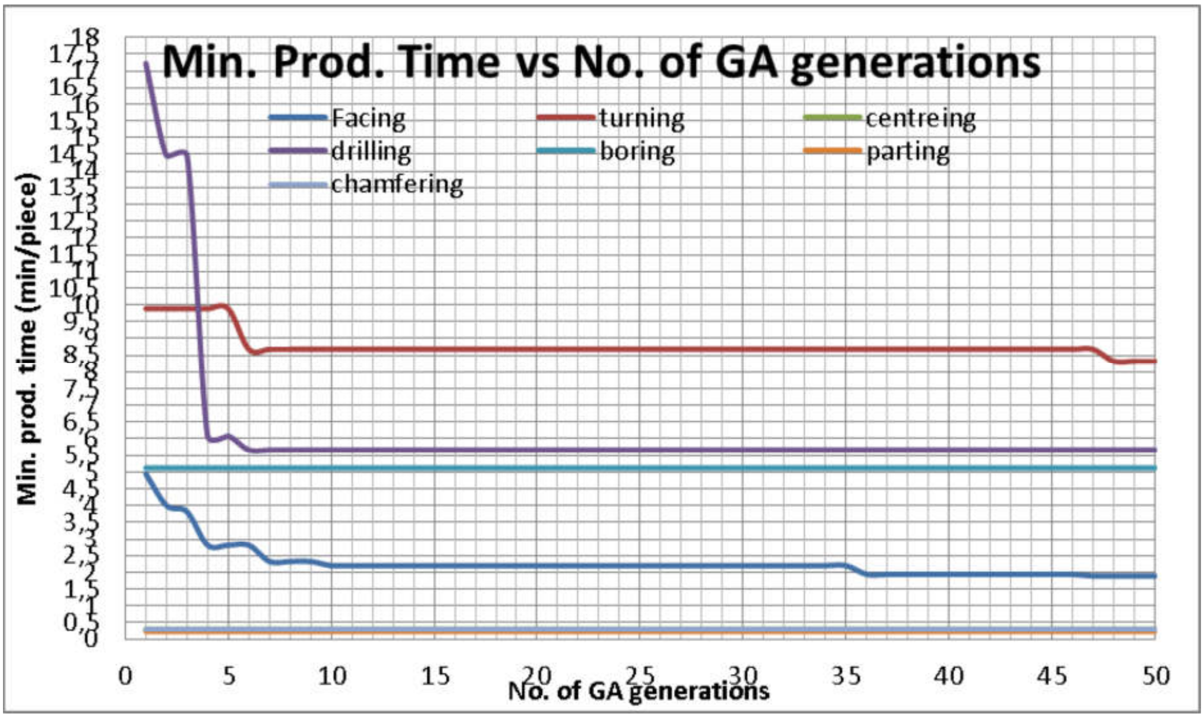

Fig. 2. Minimum unit production time for 50 generations

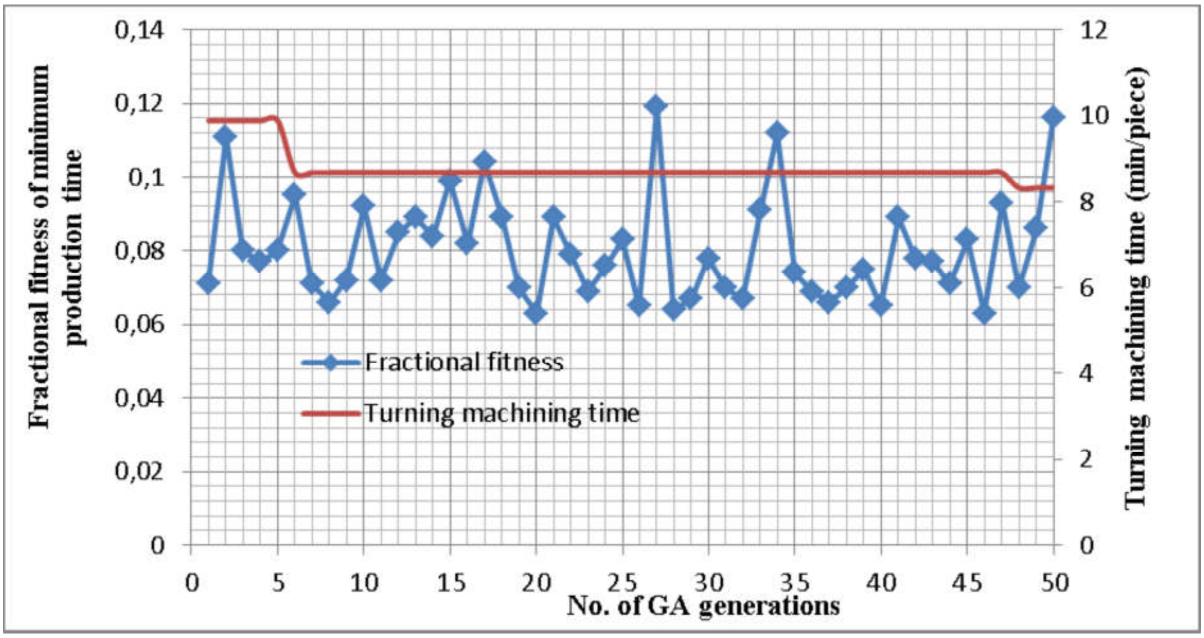

Fig. 3. Plots of optimum fractional fitness and turning time against number of generations

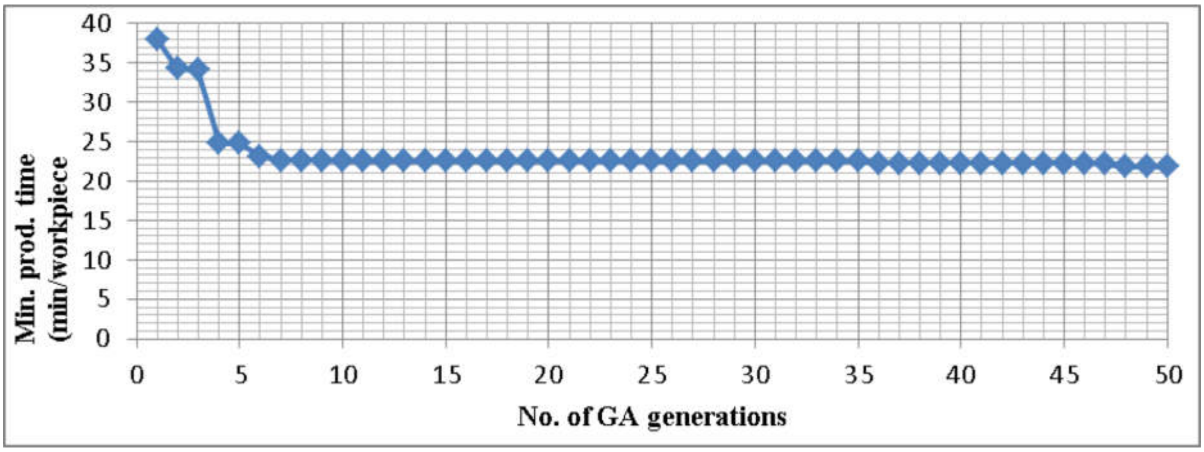

Fig. 4. Time variation with generations for multi-pass turning

Tab. 4. Optimum results obtained for the machining operations using the proposed models

S/N	Machining operation	Cutting parameters						Min. prod. Time (min/piece)
		v_r (m/min)	v_s (m/min)	f_r (mm/rev)	f_s (mm/rev)	d_r (mm)	d_s (mm)	
1	facing	126.981	-	0.820	-	2.994	-	1.893
2	turning	135.621	162.745	1.000	0.600	2.945	1.473	8.320
3	centreing	141.252	-	1.000	-	2.000	-	0.304
4	drilling	166.871	-	0.859	-	3.000	-	5.668
5	boring	141.213	169.456	0.993	0.596	2.563	1.282	5.126
6	Parting	166.702	-	1.000	-	2.654	-	0.228
7	chamfering	128.44	-	1.000	-	2.750	-	0.302
							Total	21.841

Tab. 5. Comparison of conventional method and the developed models

S/N	Machining process	Production time using data from Ibhadode [33] (min)	Production time using data from Ibhadode [33] and modified Taylor's tool life used in the Production time model (min)
1.	Facing	64.48	10.55
2.	Centreing	1.52	0.19
3.	Drilling	4.88	0.62
4.	Boring	215.71	47.41
5.	Parting	64.26	10.16
6.	Chamfering	9.40	1.02
	Total	415.13	360.25

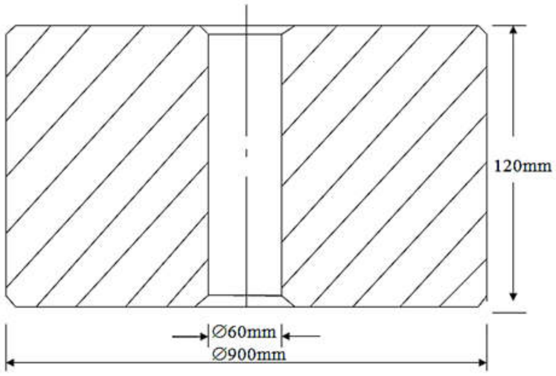

Fig. 5. Monoplex die [33]

4. CONCLUSIONS

Single-objective machining process optimization models were developed for seven machining processes involved in the production of a monoplex die container using the minimum production time model and subject to 22 technological constraints. The proposed model when implemented in Genetic Algorithms methodology gave an optimum production time of 21.84min/workpiece.

The results show that the minimum production time models predict that turning, drilling and boring have the first, second and third highest production time components respectively, for the workpiece considered. Thus, the models suggest that turning, drilling and boring operations are very important operations which demand the most production resources for the workpiece under consideration. It is therefore very important to ensure that the optimum cutting parameters of cutting speed, feed rate and depth of cut are used as derived.

The models also show that the operations of centering, parting and chamfering require the least production resources for the workpiece considered.

A comparison of the models developed (in which the optimum cutting parameters are determined by applying GAs to the models) with the conventional method of using static cutting parameters showed that the models predict better production times. Thus, the models perform better than the conventional method.

A robust Genetic Algorithms solution that is fast and efficient was developed and used to implement the optimization models.

Acknowledgements

The Authors wish to thank Prof. A. O. A.

Ibhadode, Vice chancellor, Federal University of Petroleum Resources, Effurun, Delta State, Nigeria; for allowing us to use his materials to validate the developed models.

Nomenclature

Symbols

Cf_i	cumulative fitness of a population
C_o	tool-life constant, dependent on cutting tool material/work-piece combination
D	diameter of work-piece (mm)
DA_r	dimensional accuracy in roughing machining operation (mm)
DA_U	limit of dimensional accuracy (mm)
F	$\{F_r, F_s\}$, cutting forces during rough and finishing machining (kgf)
F_U	maximum allowable cutting force (kgf)
L	length of work-piece (mm)
N	$\{N_r, N_s\}$, spindle speeds for roughing and finishing machining (rpm)
$\{N_L, N_U\}$	lower and upper bounds of the number of rough cuts
N_j	length of chromosome (binary string) of each design variable
NP	nondeterministic polynomial
N_{rp}	number of rough passes
P	$\{P_r, P_s\}$, cutting powers during roughing and finishing machining (kW)
Pf_i	% fitness of each chromosome
P_U	maximum allowable cutting power (kW)
Q	$\{Q_r, Q_s\}$, chip-tool interface temperature constraints for roughing and finishing machining (°C)
Q_U	maximum allowable chip-tool interface temperature (°C)
R	nose radius of cutting tool (mm)
SC_r	stable cutting region for roughing machining
SC_s	stable cutting region for finishing machining
SC_U	limit of stable cutting region
SR_U	maximum allowable surface roughness (μm)
T	$\{T_r, T_s\}$, expected tool-lives for roughing and finishing machining (min)
T_L, T_U	lower and upper bounds for tool life for roughing and finishing machining (min)
T_i	machine idling time (min)
T_m	actual machining time (min)
T_p	tool life of weighted combination of T_r and T_s (min)
b_i	$\{ b_{i-1}, b_{i-2}, \ldots\ldots b_0 \}$ binary string comprising genes
d	$\{d_r, d_s\}$, depth of cut in rough and finish machining operations (mm)
d_r	$\{d_{rL}, d_{rU}\}$, lower and upper bound of depth of cut in roughing machining (mm)
d_{rb}	depth of cut in roughing for boring (mm)
d_{rt}	depth of cut in roughing for straight turning (mm)
d_s	$\{d_{sL}, d_{sU}\}$, lower and upper bound of depth of cut in finish machining (mm)
d_t	depth of material to be removed (mm)
f	$\{f_r, f_s\}$, feed rates in roughing and finishing machining operations (rev/mm)
f_{ij}	the i^{th} objective function value in the j^{th} position of the current population
f_r	$\{f_{rL}, f_{rU}\}$, lower and upper bound of feed rate in roughing machining (rev/mm)
f_s	$\{f_{sL}, f_{sU}\}$, lower and upper bound of feed rate in finishing machining (rev/mm)
g_i	{i = 1, 2, …., J}, J numbers of inequality constraints
h_1	constant relating to tool travel and approach/departure time (min/mm)
h_2	constant relating to tool travel and approach/departure time (min)

h_k {k = 1, 2, …., K}, K numbers of equality constraints
k_f constant pertaining to a specific tool-workpiece combination for cutting force and cutting power
k_q constant pertaining to the constraint of chip-tool interface temperature
k_r constant pertaining to the constraint of dimensional accuracy
l { l_v , l_d, l_f }lengths of range of the variables of cutting speed, depth of cut and feed rate
l_r run back length (mm)
m number of objective functions
n number of rough cuts (an integer)
n_t an exponent that depends on cutting conditions
n_p population size
n_{pb} number of passes in roughing boring
n_{pt} number of passes in roughing turning
q {q_v, q_d, q_f } levels of precision of the variables of cutting speed, depth of cut and feed rate
r $r \in (0,1)$ random number
t_c constant term(due to loading and unloading operations) (min)
t_e tool exchange time (min)
v {v_r, v_s}, cutting speeds in rough and finish machining operations (m/min)
v_r {v_{rL}, v_{rU}}, lower and upper bound of cutting speed in rough machining (m/min)
v_s {v_{sL}, v_{sU}}, lower and upper bound of cutting speed in finish machining (rev/mm)
x {x_1 , x_2} lower and upper values of the variables
x' integer value of the corresponding random binary string
z {z_v , z_d, z_f}binary string lengths of the variables

Greek letters

α, β, δ constants in the modified Taylor's tool life equation relating to cutting speed, feed rate and depth of cut
μ, v constants relating to expression of cutting force and cutting power constraints
η machine efficiency
θ a weight for T_p [0,1]
λ, υ constants relating to expression of stable cutting region constraint
τ, φ, δ constants relating to expression of chip-tool interface temperature constraint
χ, ς, ψ constants relating to the dimensional accuracy constraint

Acronyms

CNC Computer Numerical Control
GAs Genetic Algorithms

References

1. Taylor, F. W., 1907. On the art of cutting metals. Transactions of ASME. 28: 31-35.
2. Ermer, D. S., 1997. A Century of Optimizing Machining Operations. Journal of Manufacturing Science and Engineering- Transactions of the ASME. 119: 817-822.
3. Koenigsburger, F., 1964. Design Principals of Metal Cutting Machine Tools. Peragamon, Oxford.
4. Armarego, E. J. A. and Brown R. H., 1980. The machining of metals. Prentice Hall, Inc. Englewood Cliffs, New Jersey.
5. Merchant, E., 1998. Proceedings of the CIRP International Workshop on Modeling of Machining Operations, Atlanta, GA, USA.
6. Sonmez, A. I. and Baykasoglu A., 1999. Dynamic optimization of multi-pass milling operations via geometric programming. International Journal Machine Tools and Manufacture. 39: 297-320.
7. Kumar, R. and Kumar V., 2000. Optimum selection of machining conditions in abrasive flow machining using neural networks. Journal of Material Processing Technology. 108:62- 67.
8. Ermer, D. S. and Patel D. C., 1974. Maximization of the Production Rate with Constraints by Linear Programming and Sensitivity Analysis. Proc. NAMRC. 2: 436- 449.
9. Milner, D. A., 1976. Use of Linear Programming for Machinability Data Optimization. ASME J. Mech. Design. 100: 286-291.
10. Agapiou, J. S., 1992. The Optimization of machining operations based on a combined criterion; Parts 1 &2: The use of combined objectives in single pass operations. Computers Ind. Trans. ASME. 114: 500-507.
11. Ermer, D. S., 1997. A Century of Optimizing Machining Operations. Journal of Manufacturing Science and Engineering- Transactions of the ASME. 119: 817-822.
12. Wen, X. M., Tay A.O.O. and Nee A.Y.C., 1992. Micro-Computer-Based Optimization of the Surface Grinding Process. Journal of Materials Processing Technology. 29: 75-90.
13. Xiao, G., Malkin S. and Sanai K., 1992. Intelligent Control of Cylindrical Plunge Grinding. Proceedings of the ACC, Chicago, IL. 391-398. 30.
14. Jha, N. K. and Hornik K., 1995. Integrated Computer-aided Optimal Design and Finite Element Analysis of a Plain Milling Cutter. Appl. Math. Modeling. 19: 343-352.
15. Jang, Y. D., 1992. A unified Optimization model of a machining process for specified conditions of machined surface and process performance. International Journal of Production Research. 30(3): 647-663.
16. Ermer, D. S., 1971. Optimization of the Constrained Machining Economics Problem by Geometric Programming. Journal of Engineering for Industry. Transactions of ASME. 93: 1067.
17. Lambert, B. and Walvekar A., 1978. Optimization of multi-pass machining operations. International Journal of Production Research. 16: 259-265.
18. Shin, Y. C. and Joo Y. S., 1992. Optimization of machining conditions with practical constraints. International Journal of Production Research. 30: 2907-2919.
19. Lee, Y. H., Yang B. H. and Moon K. S., 1999. An economic Machining Process model using fuzzy non-linear programme and neural network. International Journal of production Research. 37(4): 835-847.
20. Groover, M. P., 1975. Monte Carlo Simulation of the Machining Economics Problem. Transactions of the ASME. 97: 931-938.
21. Dereli, T., Filiz I. H and Baykasogln A., 2001. Optimizing Cutting Parameters in Process Planning of prismatic Parts by using Genetic Algorithms. International Journal of Production Research. 39(15): 3303-3328.
22. Srikanth, T. and kamala V., 2008. A Real Coded Genetic Algorithm for optimization of Cutting Parameters in Turning. International Journal of Computer Science and Network Security. 8(6): 189-193.
23. Saravanan, R., Asokan P. and Vijaya-Kumar K. 2003. Machining Parameters Optimization for turning Cylindrical Stock into a Continuous Finished Profile using Genetic Algorithms and Simulated Annealing. International Journal of Advance Manufacturing technology. 21: 1-9.

24. Amiolemhen, P. E. and Ibhadode A.O.A., 2004. Application of Genetic Algorithms- Determination of the Optimal Machining Parameters in the Conversion of a Cylindrical bar stock into a continuous finished profile. International Journal of Machine tools and Manufacture. 140 (1-3): 340 -345.
25. Chen, J. and Gao D. M., 2006. Multi-objective genetic algorithm for the optimization of road surface cleaning process. Journal Zhejiang University. Science A, 7(8):1416-1421.
26. Dereli, T. and Filiz I. H., 1999. Optimization of Process Planning Functions by Genetic Algorithms. Computers and Industrial Engineering. 36: 281-308.
27. Dereli, T. and Filiz I. H., 2000. Allocating optimal Index Positions on Tool Magazines using Genetic Algorithms. Robotics and Autonomous Systems. 33: 155-167.
28. Ahmad, N., Tanaka T. and Saito Y., 2006. Cutting parameters optimization and constraints investment for turning process by GA with self- organizing Adaptive penalty strategy. JMSE International Journal, series C. 49(2): 293-300.
29. Onwubolu, G. C. and Kumalo T., 2002. Multi- pass turning optimization based on Genetic Algorithms. International Journal of Production Research. 39(16): 3727-3745.
30. Schaffer, J. D., 1989. A study of control parameters affecting on line performance of Genetic algorithms for function optimization. Proceedings of the third International Conference on Genetic Algorithms, Los Altos. 51-60.
31. Gen, M. and Cheng R., 1997. Genetic Algorithms and Engineering Design. Wiley Publishers, U.S.A.
32. Chen, M. C. and Tseng H. Y., 1998. Machining parameters selection for stock removal turning in process planning using a float encoding genetic algorithm. Journal of the Chinese Institute of Engineers. 16(4): 493-506.
33. Ibhadode, A. O. A., 2009. Precision die design by the die expansion method. Trans. Tech Publications Ltd., Zurich.

Biographical notes

Patrick Amiolemhen is an Associate Professor in the Department of Production Engineering, University of Benin, Benin City. He holds a Bachelor of Engineering (B. Eng.) in Mechanical Engineering from the University of Benin, Benin City in 1991 and M. Eng. and PhD degrees in Production Engineering (Manufacturing Option from the University of Benin, Benin City in 203 and 2009 respectively. He is a certified mechanical engineer; a member of the Nigerian Society of Engineers as well as a member of the Nigerian Institution of Production Engineers. He has published both national and international journal articles. Since 2006 he has been a researcher in the Department of Production Engineering, University of Benin, where he teaches machines design; engineering mathematics; machine tool technology; operations & production management at the undergraduate levels; while at the postgraduate level he teaches machines design; machine tool design; metal machining process optimization and human resources management. His present research areas focus on design and manufacture of engineering machineries and application of computational algorithms to multi-objectives optimization in engineering.

Joshua Eseigbe received his Bachelor's degree in Production Engineering from the University of Benin, Benin City Nigeria in 2012, and master's degree in industrial engineering from the same university in 2017. He is currently undergoing his doctorate degree in the Department of Production, University of Benin, Benin City. His research interests include; engineering design, optimization and industrial engineering.