

Tool path planning of hole-making operations in ejector plate of injection mould using modified shuffled frog leaping algorithm

Amol M. Dalavi^{a,*}, Padmakar J. Pawar^b, Tejinder Paul Singh^a

^aDepartment of Mechanical Engineering, Symbiosis Institute of Technology, Symbiosis International University, Gram Lavale, Mulshi, Pune 412115, India

^bDepartment of Production Engineering, K.K. Wagh Institute of Engineering Education and Research, Nashik, India

Received 22 December 2015; received in revised form 4 April 2016; accepted 6 April 2016

Available online 8 April 2016

Abstract

Optimization of hole-making operations in manufacturing industry plays a vital role. Tool travel and tool switch planning are the two major issues in hole-making operations. Many industrial applications such as moulds, dies, engine block, automotive parts etc. requires machining of large number of holes. Large number of machining operations like drilling, enlargement or tapping/reaming are required to achieve the final size of individual hole, which gives rise to number of possible sequences to complete hole-making operations on the part depending upon the location of hole and tool sequence to be followed. It is necessary to find the optimal sequence of operations which minimizes the total processing cost of hole-making operations. In this work, therefore an attempt is made to reduce the total processing cost of hole-making operations by applying relatively new optimization algorithms known as shuffled frog leaping algorithm and proposed modified shuffled frog leaping algorithm for the determination of optimal sequence of hole-making operations. An industrial application example of ejector plate of injection mould is considered in this work to demonstrate the proposed approach. The obtained results by the shuffled frog leaping algorithm and proposed modified shuffled frog leaping algorithm are compared with each other. It is seen from the obtained results that the results of proposed modified shuffled frog leaping algorithm are superior to those obtained using shuffled frog leaping algorithm.

© 2016 Society of CAD/CAM Engineers. Publishing Services by Elsevier. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Keywords: Hole-making operations; Shuffled frog leaping algorithm; Modified shuffled frog leaping algorithm; Injection mould; Ejector plate

1. Introduction

Mould carries large number of holes of various sizes. In hole-making operations of mould, to achieve the final size of each hole may require different machining operations like drilling with pilot tool, enlargement or tapping/reaming depending upon requirement of diameter, surface finish and depth of cut. Machining of hole or holes may require tool or combination of tools to achieve the final size diameter of hole. E.g. for hole H₃ shown in Fig. 1, may require one of {T₁, T₂, T₃}, {T₁, T₃}, {T₂, T₃}, and {T₃} tools to obtain the final size. Various combinations of tools for individual hole to achieve

the desired size of hole has impact on optimum cutting speeds, tool switch time and tool travel time [19].

In machining processes, it takes more machining time for tool switching and table movement from one position to another. To reduce the tool travel, the spindle is not moved till desired hole is completely machined by various tools which increases the tool switch time and cost. On the other side to reduce tool switch time, the same tool may be used for all drilling operations of same size which in turn increases the tool travel time and cost. Typically 70% of total time in manufacturing processes is spent on tool and part movements [27]. Luong and Spedding [25] presented the process planning in hole-making operations by developing a generic knowledge based methodology. Kolahan and Liang [19] report a tabu-search (TS) technique to reduce the total machining cost of hole-making operations of application example of plastic injection mould. Three components of total machining cost

*Corresponding author. Tel.: +91 9921517645.

E-mail address: amol.dalavi83@gmail.com (A.M. Dalavi).

Peer review under responsibility of Society of CAD/CAM Engineers.

Nomenclature

X_{i+1}	New position of frog
X_i	Previous position of frog
r	Random number values between 0 to 1
X_b	Position of best frog among the memeplexes
X_w	Position of worst frog among the memeplexes
X_g	Position of global best frog in search space which best among all frogs.
w	Inertia weight
C_1	Search acceleration factor with positive values
C_2	Search acceleration factor with positive values
D	The total holes to be machined in the part
(x_d, y_d)	are the co-ordinates of point d
(x_e, y_e)	are the co-ordinates of point e
l_{de}	Non-productive tool travel time required for moving tool from the point d to the point e in rectilinear direction
d	Tool type index in ascending order according to the tool diameters, $d=1, \dots, D$

e, f	Hole index, $e=1, \dots, E$ $f=1, \dots, E$
de, I	Index for the last tool to be used on hole e
C_{de}	Combined tool and machining costs when tool type d is used on hole e .
a	Cost per unit tool switch time
b	Cost per unit non-productive traveling time
m_{dT}	The total operations required for hole d , $d=1, 2, \dots, D$
M	$m_1 + m_2 + \dots + m_D$, the total of operations in the part
T_{deT}	The tool required for operation e of hole d .
$a_{dd'}$	The tool travel time for traveling from hole d to hole d'
$S_{de, d'e'}$	The time required for switching the tool $T_{d'e'}$ when tool T_{de} is in spindle
X_{def}	1 if operation e of hole d is machined in position f of operation order, otherwise 0, where
d	$1, 2, \dots, D$, $e=1, 2, \dots, m_i$, $f=1, 2, \dots, M$
$\delta(T_{de}, T_{d'e'})$	1 if $T_{de} \neq T_{d'e'}$, otherwise 0

namely tooling and machining cost, non-productive tool travel cost and tool switching cost were considered for the optimization of hole-making operations. Alam et al. [1] presented the case study of injection moulds with the aim of achieving minimum total processing time of machining using genetic algorithm (GA) and compared GA results with simulated annealing (SA). Qudeiri and Hidehiko [34] used genetic algorithm to obtain concise cutting tool path for machine operations. Liyun [22] presented the process planning optimization by using an genetic simulated annealing algorithm.

Guo et al. [10] modeled a complicated operation sequencing process and applied modified particle swarm optimization

(PSO) algorithm on case study of three prismatic parts and compared the results of PSO with GA. Guo et al. [11] presented a case study of five-axis prismatic parts for sequencing the operations using modified particle swarm optimization approach.

Ghaiebi and Solimanpur [9] presented a case study by application of the ant colony optimization (ACO) algorithm for achieving optimal path of machining holes in a typical industrial part. Six bench mark problems were attempted in order to validate the performance of their ACO algorithm and compared ACO results with dynamic programming (DP). Oscar et al. [32] presented a methodology to generate optimal sequences of G commands to minimize the manufacturing time of computer numerical control machine (CNC) using ACO. Liu et al. [21] used ACO algorithm for process planning optimization of hole-making operations of a case study with objective to minimize non-productive tool time and tool switching time. Kiani et al. [18] used ant colony algorithm to achieve the optimal sequence of operations that gives concise cutting trajectory in computer numerical control machine. Narooei et al. [28] used ACO algorithm for optimizing the tool path i.e. to minimize non-productive tool travel of case study involving multiple holes. Simulation of machining operation is considered similar to traveling salesmen problem (TSP). Jiang et al. [16] compared the performance of ant colony optimization, genetic algorithm and the common sequence method for replugging tour planning of seedling transplanter. Results obtained using ACO and GA were more suitable than common sequence method.

Hsieh et al. [12] investigated the optimal sequence of hole-making operations by minimizing the non-productive tool travel time and tool switch time, in which various tools were required to obtain the desired size of hole on part using immune based evolutionary approach (IA) and compared its

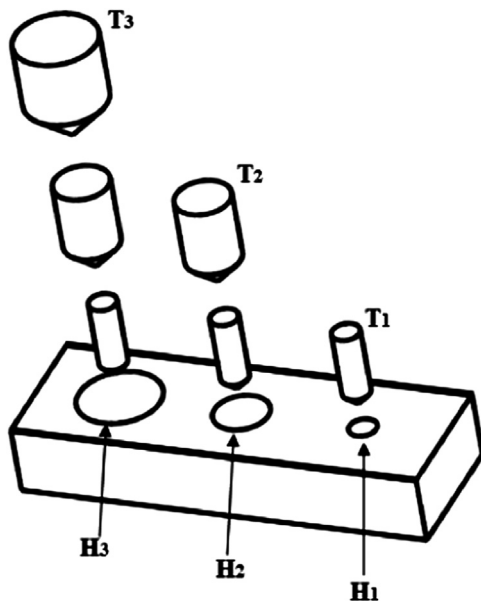


Fig. 1. Diagrammatic depiction of part which requires various tools to machine a hole to its final size [19].

results with ACO and PSO. Tamjidy and Shahla [38] presented an evolutionary algorithm to reduce the tool travel and tool switching time during hole-making operations based on geographic classification of biological organism. Performance of their proposed algorithm was validated based on test functions adopted from in literature. Nassehi et al. [29] used evolutionary algorithms for generation and sequence optimization of tool path for case study on milling. Ismail [15] used firefly algorithm (FA) to reduce non-productive tool travel time in PCB holes drilling process. Srivatsava [37] presented firefly algorithm for achieving optimal test sequence generation. Marinakis and Marinaki [26] used bumble bees mating optimization algorithm for the open vehicle routing problem i.e. to reduce vehicle travel distance. Two benchmark problems were considered in order to validate their proposed algorithm. Lim et al. [20] used cuckoo search (CS) algorithm for optimization of sequence in PCB holes drilling process. Apart from this work, Begonã et al. [3] evaluated a monitoring method, based on internal signals from spindle torque, to detect non-desired burr formation during drilling operations. David et al. [5] presented experimental analysis of hole-making using ball helical milling on titanium alloys.

It is understood from the literature discussed here that most of the researchers have worked in the area of minimization of non-productive tool travel time and tool switching time. Kolahan and Liang [19] has considered three elements of total processing costs, tooling and machining cost, non-productive tool travel cost and tool switching cost.

It is also found in the literature related to this area that the non-traditional optimization methods such as tabu search, genetic algorithm, particle swarm optimization, ant colony algorithm, immune algorithm, cuckoo search, firefly algorithm, bumble bees mating optimization algorithm and biogeography based optimization (BBO) algorithm etc. has been used to solve the problem of optimization of hole-making operations. Tabu search that uses only one solution can easily neglect some promising areas of the search space also they may not find optimal solution or exact solution. Most widely used advanced optimization technique is the genetic algorithm. Genetic algorithm gives near optimal solution for complex problems [35]. Also GA requires more parameters [6]. In ACO algorithm, convergence is slow due to pheromone evaporation and CPU time requirement is more [6]. Immune based evolutionary approach requires more parameters. PSO algorithm was usually found to perform better than other algorithms in terms of success rate and solution quality [6]. Problem solving success of the cuckoo search and differential evolution algorithms are relatively better than the PSO [4]. Basic cuckoo search algorithm may easily fall into local optimum solution [13].

Firefly algorithm (FA) has limitations like it gets trapped into several local optima. Also FA does not memorize or remember any history of better situation for each firefly [33]. Honey bees mating optimization algorithm may miss the optimum and provide a near optimum solution in a limited runtime period [30]. Biogeography-based optimization (BBO)

is poor in exploiting the solutions. Also there is no provision for selecting the best members from each generation [2].

It is necessary to use non-traditional optimization algorithm which is robust and gives correct solution for complex problems [35]. Hence in this work attempt is made to minimize the total processing cost of hole-making operations of ejector plate of injection mould of a completely new application example, which consists of three elements of costs namely tooling and machining cost, non-productive tool travel cost and tool switching cost using shuffled frog leaping algorithm [6,8] and proposed modified shuffled frog leaping algorithm.

Ejector plate is member in injection mould assembly which pushes the ejector pins. It is mounted on the ejector retainer plate to form the ejector unit. Main function of ejector plate is to prevent the shrinkage of element. Next section discuss about the SFLA algorithm.

2. Shuffled frog leaping algorithm

Shuffled frog leaping algorithm is a meta-heuristic optimization technique, originated by Eusuff and Lansey, which is similar to the conduct of a group of frogs while searching for the maximum amount of food site [8]. Shuffled frog leaping algorithm consists of random frogs called 'population' which are further divided into different parts called 'memeplexes'. Individual frog carries out two search mechanisms called local and global search mechanisms to get optimum solution. Through these two mechanisms behavior of individual frog is influenced by neighboring frog to obtain the best solution. Thereafter the frog population is shuffled. Local and global search mechanisms were carried out until convergence criteria are achieved [23,24].

Shuffled frog leaping algorithm can be used for discrete optimization problems [8]. It has been successfully applied to several engineering optimization problems such as economic load dispatch problem [36], multiobjective optimal power flow [31], project management [7], and traveling salesman problem [23].

The most well-known benefit of Shuffled frog leaping algorithm is its fast convergence speed [6]. The Shuffled frog leaping algorithm combines the advantages of the both the genetic-based memetic algorithm (MA) and the social behavior-based PSO algorithm [14,17].

Flowchart of shuffled frog leaping algorithm is as shown in Fig. 2.

The various steps in shuffled frog leaping algorithms are as follows [8]:

1. Generate virtual frog randomly called population ' p '.
2. Evaluate the fitness of population ' p '.
3. Sort the population ' p ' in descending order.
4. Partition the population ' p ' in ' m ' memeplexes.
5. Frogs i is expressed as $X_i = (X_{i1}, X_{i2}, \dots, X_{is})$ where ' S ' represents number of variables.
6. Identify the worst frog X_w and best frog X_b within each memeplexes.
7. Identify the global best frog X_g in entire population ' p '.

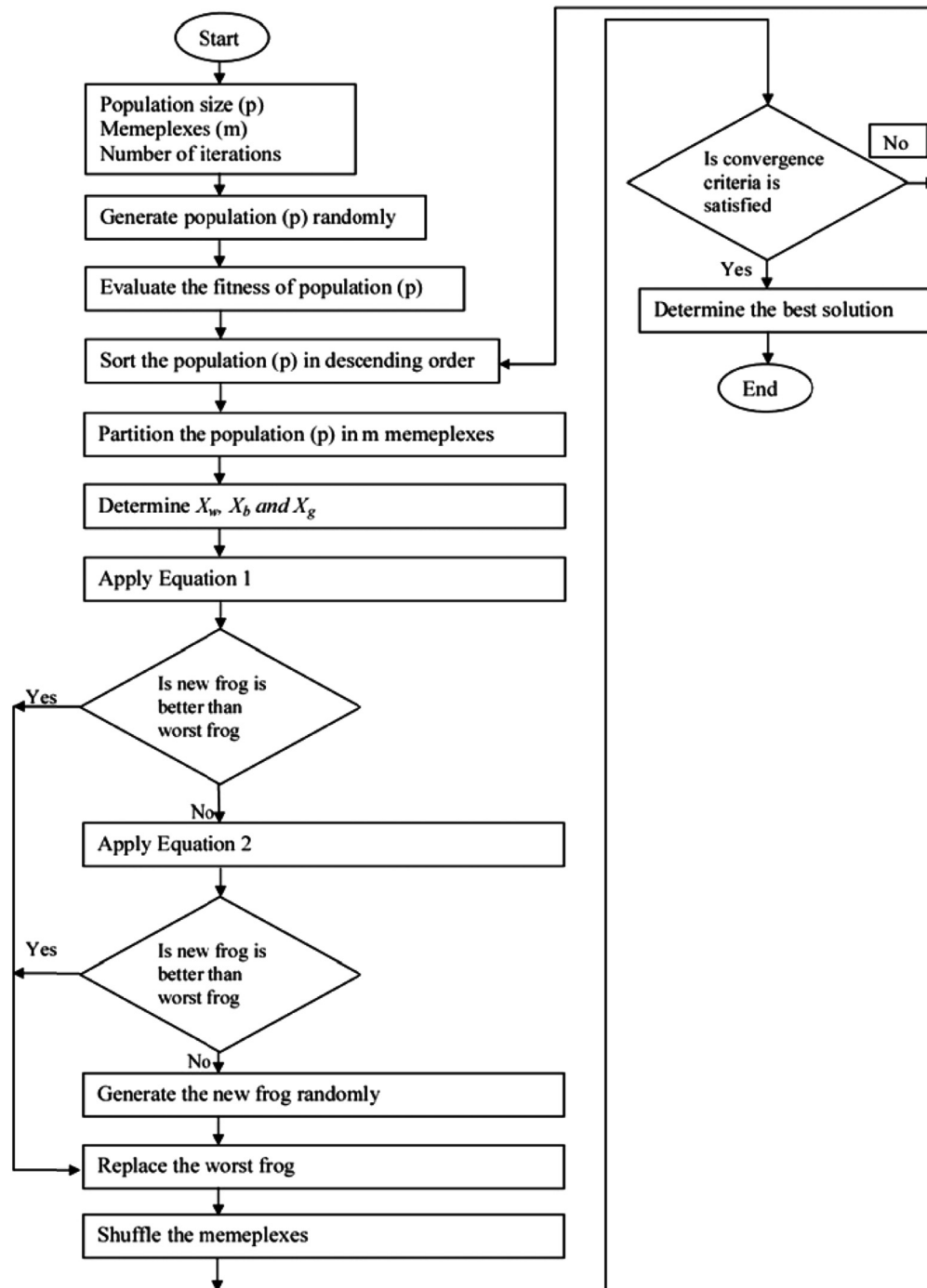


Fig. 2. Flowchart of shuffled frog leaping algorithm.

8. Apply the local search for new positions (X_{i+1}) by following (Eqs. (1) and 2).

$$X_{i+1} = X_i + r \times (X_b - X_w) \quad (1)$$

If fitness of new frog generated by above Eq. (1) better than previous frog then replace it with new frog.

9. If not, apply the Eq. (2) to obtain better position

$$X_{i+1} = X_i + r \times (X_g - X_w) \quad (2)$$

If fitness of new frog generated by above Eq. (2) better than previous frog then replace it with new frog, else replace the worst frog randomly.

The local search and the shuffling processes continue until convergence criteria are satisfied.

Modified shuffled frog leaping algorithm is discussed in next section.

3. Modified shuffled frog leaping algorithm (mSFLA)

In order to widen the search capability and overcome premature convergence, the local search mechanism is modified in existing shuffled frog leaping algorithm is discussed in this section. Flowchart for modified shuffled frog leaping algorithm is similar to the flowchart of shuffled frog leaping algorithm except local search Eq. (3) and global search Eq. (4), which is discussed in following steps of mSFLA.

The various steps in modified shuffled frog leaping algorithms are as follows:

1. Generate virtual frog randomly called population 'p'.
2. Estimate the fitness of population.
3. Group the population in descending manner.
4. Divide the population in 'm' memeplexes.
5. Frogs 'i' is expressed as $X_i = (X_{i1}, X_{i2}, \dots, X_{is})$ where 'S' stands for number of variables.
6. Select the worst frog X_w and best frog X_b within each memeplexes.
7. Select the global best frog X_g in entire population.
8. Apply the local search for new generations by following (Eqs. (3) and 4)

$$X_{i+1} = w \times X_i + C_1 \times r \times (X_b - X_w) \quad (3)$$

Weight factor 'w' is introduced on right hand side of Eq. (3) in order to widen the search capability and to avoid premature convergence. Similarly 'w' is introduced in right side of Eq. (4) below. If fitness of new frog generated by above Eq. (3) better than previous frog then replace it with new frog.

When the difference between worst frog X_w and best frog X_b becomes small, change in frog X_w 's position will be very small, hence it might stuck in local optimum and results into premature convergence. To avoid such event, in right hand side of Eq. (3) search acceleration factor with positive values C_1 is introduced [7]. Similarly C_2 is introduced in right hand side of Eq. (4).

9. If not, apply the Eq. (4)

$$X_{i+1} = w \times X_i + C_2 \times r \times (X_g - X_w) \quad (4)$$

These factors C_1 , C_2 and w are positive constant values.

10. If fitness of new frog generated by above (Eqs. (3) and 4) better than previous frog then replace it with new frog, else replace the worst frog randomly.

The local search and the shuffling processes continued until convergence criteria are satisfied. Next section discuss about the optimization model used for hole-making operations.

4. Formulation of an optimization model for hole-making of operations

In order to minimize the total processing cost of hole-making operation, the following optimization model is formulated based on analysis given by Kolahan and Liang [19] and Ghaiebi and Solimanpur [9] considering following components of total cost:

4.1. Tool travel cost

Tool travel cost is the cost of moving the tool from its previous location to the current drilling position. Tool travel cost is proportional to the non-productive traveling distance required for the spindle to move between two drilling locations.

It is assumed that, at a time, two axis drill press can travel only in one direction and a rectilinear distance function is used in this paper [9]. The non-productive tool travel time required for moving tool from the point d to the point e in rectilinear direction is given by

$$l_{de} = |x_d - x_e| + |y_d - y_e| \quad (5)$$

4.2. Tool switch cost

It occurs whenever a different tool is used for the next operation. If tool type required for operation is not available on the spindle, then the required tool must be loaded on the spindle prior to performing operation.

4.3. Tool and machining costs

Tool cost includes the new tool cost and the cost of machine down time required to replace the tool. Machining cost comprises the operating cost and the machine overhead cost. Depth of cut, feed rate, and cutting speeds affect tool and machining costs. The actual combined tooling and machining costs when tool type d is used on hole e can be expressed as C_{de} [19]. Mathematical model is as below:

Minimize

$$\sum_{d=1}^D \sum_{e=1}^{m_d} \sum_{d'=1}^D \sum_{e'=1}^{m_{d'}} \sum_{f=1}^{M-1} b \quad (6)$$

$$* l_{de} x_{def} x_{d'e'f+1} + \sum_{d=1}^D \sum_{e=1}^{m_d} \sum_{d'=1}^D \sum_{e'=1}^{m_{d'}} \sum_{j=1}^{M-1} a$$

$$* S_{de,d'e'} \delta(T_{de}, T_{d'e'}) x_{def} x_{d'e'f+1} + C_{de}$$

Subject to

$$\sum_{f=1}^{M-1} x_{def} = 1, d = 1, 2, \dots, D, e = 1, 2, \dots, m_d \quad (7)$$

$$\sum_{d=1}^D \sum_{e=1}^{m_d} x_{def} = 1, f = 1, 2, \dots, M \quad (8)$$

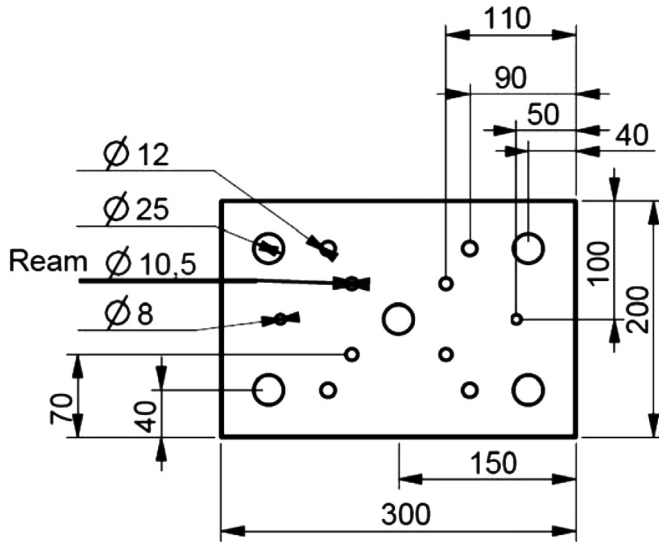


Fig. 3. Top view of example part.

$$x_{def} \leq \sum_{f'=f+1}^M x_{d,e+1,f'}, d=1,2,\dots,D, e=1,2,\dots,m_i-1 \quad (9)$$

$$x_{def} \in \{0,1\}, \forall d,e,f \quad (10)$$

5. Industrial application example

The shuffled frog leaping algorithm and modified shuffled frog leaping algorithm were coded as per mathematical model given in Section 4 in order to obtain the optimal path of drilling the holes for a part shown in Figs. 3 and 4 using code blocks C++ and run on a windows 8 PC with Intel core i3 CPU @ 1.90 GHz. Data required for calculating the tool travel time which is discussed in Section 4 is shown in Fig. 3. Fig. 4 shows the identification numbers for various holes on application example. Application example shown in Fig. 3 consists of total 15 holes which require carry out total 47 hole-making operations such as drilling, enlargement and reaming to finish the part. The details of tool diameters are given in Table 1. Whenever a different tool is required for machining of a particular hole, the tool switch time is taken as 2 s for CNC machine.

Table 2 presents combinations of tools required for machining of individual hole of application example presented in this section. For example, for machining of Ø25 hole shown in Fig. 3 requires tool 1, 3, 4, 5 and 6 tools as given in Table 1. Thickness of industrial application example is 80 mm.

6. Results and discussion

In this section obtained results of optimization of shuffled frog leaping algorithm and modified shuffled frog leaping algorithm for the application example discussed in Section 5 are given in Tables 3 and 4 as per the mathematical model given Section 4.

Considering tool information given in Table 1 and tool switch times for whenever different tool is required for

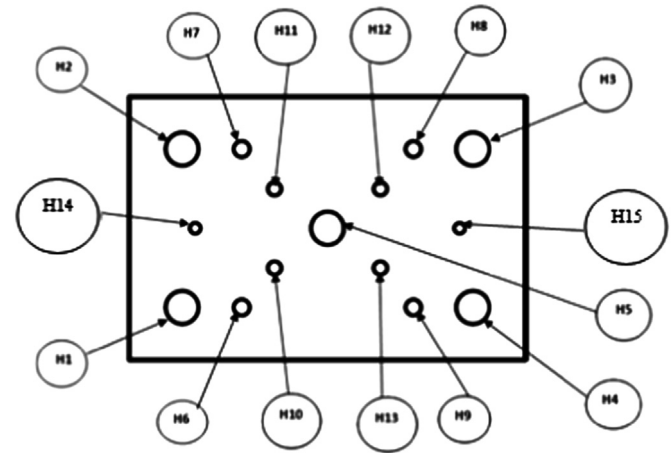


Fig. 4. Identification numbers for various holes.

Table 1

Details of tools required hole-making for application example.

Tool type <i>d</i>	Drill						Reamer
	1	2	3	4	5	6	7
Tool diameter (mm)	8	10	12	16	20	25	10.5

Table 2

Desired set of tools for machining individual hole in the application example.

Hole	Ø8	Ø10.5	Ø12	Ø25
Desired tools	1	1-2-7	1-3	1-3-4-5-6

machining is taken 2 s. Results obtained using shuffled frog leaping algorithm and modified shuffled frog leaping algorithm are discussed below:

6.1. Following algorithm specific parameters for shuffled frog leaping algorithm are obtained through various computational experiments

Frog population = 100
 Quantity of memplexes = 10
 Quantity of sub frogs = 10
 Number of iterations = 100

With above parameters of shuffled frog leaping algorithm (Section 6.1) an optimal sequence hole-making operation of industrial application mould given in Table 4.

Table 3 gives best possible sequence using shuffled frog leaping algorithm having optimum values of airtime of 2.083 min and tool switch time of 1.29 min.

6.2. Following algorithm specific parameters for modified shuffled frog leaping algorithm are obtained through various computational experiments

$C_1 = 1.0$,
 $C_2 = 0.95$,
 $w = 0.95$,

Table 3
Optimal sequence hole-making operations using SFLA.

→									
Tool-hole combination	8-H1	12-H1	16-H1	20-H1	25-H1	8-H2	12-H2	16-H2	20-H2
Tool-hole combination	25-H2	8-H3	12-H3	8-H4	16-H3	20-H3	25-H3	12-H4	16-H4
Tool-hole combination	20-H4	25-H4	8-H5	12-H5	16-H5	20-H5	25-H5	8-H6	12-H6
Tool-hole combination	8-H8	12-H8	8-H9	8-H10	10-H10	10.5-H10	8-H7	8-H11	10-H11
Tool-hole combination	8-H12	10-H12	10.5-H12	12-H9	8-H13	10-H13	8-H15	10.5-H13	10.5-H11
Tool-hole combination	12-H7	8-H14							

Table 4
Optimal sequence hole-making operations using modified SFLA.

→									
Tool-hole combination	8-H1	12-H1	16-H1	20-H1	25-H1	8-H2	12-H2	16-H2	20-H2
Tool-hole combination	25-H2	8-H3	12-H3	16-H3	20-H3	25-H3	8-H4	12-H4	16-H4
Tool-hole combination	20-H4	25-H4	8-H5	12-H5	16-H5	20-H5	25-H5	8-H6	12-H6
Tool-hole combination	8-H10	8-H11	8-H7	8-H8	12-H8	8-H9	12-H9	10-H10	10.5-H10
Tool-hole combination	10-H11	8-H12	10-H12	8-H15	8-H13	10-H13	10.5-H13	10.5-H12	10.5-H11
Tool-hole combination	12-H7	8-H14							

Table 5
Result comparison of application example.

Method	C_{de} in Rs.	$b*I_{de}$ in Rs.	$a*S_{de}$ in Rs.	Total Processing cost in Rs.
SFLA	338	52.075	32.25	422.325
Modified SFLA	338	43.75	30	411.75

Rs.=Indian Rupees.

Frog population = 100

Quantity of memplexes = 10

Quantity of sub frogs = 10

Number of iterations = 100

With above parameters of modified shuffled frog leaping algorithm (Section 6.2) an optimal sequence hole-making operation of industrial application mould given in Table 4.

Table 4 gives best possible sequence using modified shuffled frog leaping algorithm having optimum values of airtime of 1.75 min and tool switch time of 1.2 min.

For the above parameter setting and obtained results of optimal sequence using SFLA and modified SFLA, the results of optimization of total processing cost of hole-making operations for the application example mentioned in Section 5 are as given in Table 5. It is calculated as per the mathematical model given Section 4.

Process parameter assumed for this application example are, a =Rs.25/min and b =Rs.25/min.

Tooling and machining cost for all hole-making operations = Rs.338.

7. Conclusion

Optimization of hole-making operations involves a large number of possible sequences to complete hole-making operations on the part depending on location of hole and tool

sequence to be followed. To achieve this, proper determination operations sequence which minimizes the total non productive time and tool switching time of hole-making operations is essential. This paper presents recently developed shuffled frog leaping algorithm and proposed modified shuffled frog leaping algorithm and are applied on an application example to reduce the total machining cost hole-making operations. The obtained results using shuffled frog leaping algorithm and modified shuffled frog leaping algorithm are compared with each other. It is observed that the results of optimization for total processing cost using modified shuffled frog leaping algorithm in 100 generations are 3% superior to shuffled frog leaping algorithm. Tooling and machining cost of hole-making operations of this application example will remain same, only tool travel and tool switch time is crucial. Results of optimization of tool travel and tool switch cost only, achieved by modified shuffled frog leaping algorithm in 100 generations are 13% superior to those obtained using shuffled frog leaping algorithm for application example discussed in Section 5. The improvement of tool travel and tool switch cost obtained by using modified shuffled frog leaping algorithm is thus significant and clearly indicates the potential of this method to solve real life problems related to hole making for various industrial applications.

References

- [1] Alam MR, Lee KS, Rahman M. Process planning optimization for the manufacture of injection moulds using a genetic algorithm. *Int. J. Comput. Integr. Manuf.* 2003;16(3):181–91.
- [2] Ammu PK, Sivakumar KC, Rejimoan R. Biogeography-based optimization – a Survey. *Int. J. Electron. Comput. Sci. Eng.* 2012;2(1):154–60.
- [3] Begonã, P, et al. Monitoring of drilling for burr detection using spindle torque. *Int. J. Mach. Tools Manuf.* 2005;45(14):1614–21.
- [4] Civicioglu P, Besdok E. A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. *Artif. Intell. Rev.* 2013;39(4):315–46.
- [5] David, O, et al. Hole making using ball helical milling on titanium alloys. *Mach. Sci. Technol.* 2012;16:173–88.

- [6] Elbeltagi E, Tarek H, Donal G. Comparison among five evolutionary based optimization algorithms. *Adv. Eng. Inform.* 2005;**19**:43–53.
- [7] Elbeltagi E, Tarek H, Donald G. A modified shuffled frog-leaping optimization algorithm: applications to project management. *Struct. Infrastruct. Eng.* 2007;**3**(1):53–60.
- [8] Eusuff MM, Lansey KE, Pasha F. Shuffled frog-leaping algorithm: a memetic metaheuristic for discrete optimization. *Eng. Optim.* 2006;**38**(2):129–54.
- [9] Ghaiebi H, Solimanpur M. An ant algorithm for optimization of hole-making operations. *Comput. Ind. Eng.* 2007;**52**(2):308–19.
- [10] Guo, et al. Operation sequencing optimization using a particle swarm optimization approach. *Proc. Inst. Mech Eng. B: J. Eng. Manuf.* 2006;**220**(12):1945–58.
- [11] Guo, et al. Operation sequencing optimization for five-axis prismatic parts using a particle swarm optimization approach. *Proc. Inst. Mech Eng. B: J. Eng. Manuf.* 2009;**223**(5):485–97.
- [12] Hsieh YC, Lee YC, You PS. Using an effective immune based evolutionary approach for the optimal operation sequence of hole-making with multiple tools. *J. Comput. Inf. Syst.* 2011;**7**(2):411–8.
- [13] Huang L, Ding S, Yu S, Wang J, Lu K. Chaos-enhanced Cuckoo search optimization algorithms for global optimization. *Appl. Math Model* 2016;**40**:3860–75.
- [14] Huynh TH. A modified shuffled frog leaping algorithm for optimal tuning of multivariable PID controllers. *Proc ICIT* 2008:1–6.
- [15] Ismail, M.M., 2012. Firefly algorithm for path optimization in PCB holes drilling process. In: Proceedings of the Green and Ubiquitous Technology (GUT) International Conference Jakarta IEEE. pp.110–113.
- [16] Jiang Z, Zhou M, Tong M, Jiang H, Yang Y, Wang A, You Z. Comparing an ant colony algorithm with a genetic algorithm for replugging tour planning of seedling transplanter. *Comput. Electron. Agric.* 2015;**113**:225–33.
- [17] Kennedy, J, Eberhart R.C.,1995. Particle swarm optimization. In: Proceedings of the IEEE Conference on Neural Network, 4. pp. 1942–1948.
- [18] Kiani K, Sharifi M, Shakeri M. Optimization of cutting trajectory to improve manufacturing time in computer numerical control machine using ant colony algorithm. *Proc. Inst. Mech Eng. B: J. Eng. Manuf.* 2014;**228**(7):811–6.
- [19] Kolahan F, Liang M. Optimization of hole-making operations: a tabu-search approach. *Int. J. Mach Tools Manuf.* 2000;**40**:1735–53.
- [20] Lim WCE, Kanagaraj G, Ponnambalam SG. Cuckoo search algorithm for optimization of sequence in PCB holes drilling process. *Emerg. Trends Sci. Eng. Technol. Lect. Notes Mech. Eng.* 2012:207–16.
- [21] Liu X, Hong Y, Ni Z, Qi J, Qiu Z. Process planning optimization of hole-making operations using ant colony algorithm. *Int. J. Adv. Manuf. Technol.* 2013;**69**(1–4):753–69.
- [22] Liyun, X.U., 2014. Optimization of process planning for cylinder block based on feature machining elements. In: IEEE International Conference Conference on Systems, Man and Cybernetics (SMC), San Diego, CA.
- [23] Luo XH, Yang Y, Li X. Solving TSP with shuffled frog-leaping algorithm. *Proc. ISDA* 2008;**3**:228–32.
- [24] Luo, Ping LU, Qianqiang, WU, Chenxi, 2011. Modified shuffled frog leaping algorithm based on new searching strategy. In: Proceedings of the 7th International Conference on Natural computation.
- [25] Luong LHS, Spedding T. An integrated system for process planning and cost estimation in hole-making. *Int. J. Manuf. Technol.* 1995;**10**:411–5.
- [26] Marinakis Y, Marinaki A. Bumble Bees mating optimization algorithm for the open vehicle routing problem. *Swarm Evolut. Comput.* 2014:1580–94.
- [27] Merchant RL. World trends and prospects in manufacturing technology. *Int. J. Veh. Des.* 1985;**6**:121–38.
- [28] Naroei KN, Ramli R, Rahman MZ, Ibrahimi F, Qudeiri JA. Tool routing path optimization for multi-hole drilling based on ant colony optimization. *World Appl. Sci. J.* 2014;**32**(9):1894–8.
- [29] Nassehi A, Essink W, Barclay J. Evolutionary algorithms for generation and optimization of tool paths. *CIRP Ann. – Manuf. Technol.* 2015;**64**(1):455–8.
- [30] Niknam T, Mojarad HD, Meymand HZ, Firouzi BB. A new honey bee mating optimization algorithm for non-smooth economic dispatch. *Energy* 2011;**36**(2):896–908.
- [31] Niknam T, Narimani MR, Jabbari M, Malekpour AR. A modified shuffle frog leaping algorithm for multi-objective optimal power flow. *Energy* 2011;**36**:6420–32.
- [32] Oscar MR, Rodríguez N, Sepúlveda R, Melin P. Methodology to Optimize manufacturing time for a CNC using a high performance implementation of ACO. *Int. J. Adv. Robot Syst.* 2012;**9**:121.
- [33] Pal S, Rai C. Comparative study of firefly algorithm and particle swarm optimization for noisy non-linear optimization problems. *J. Intell. Syst. Appl.* 2012;**10**:50–7.
- [34] Qudeiri JA, Hidehiko Y. Optimization of operation sequence in CNC machine tools using genetic algorithm. *J. Adv. Mech. Des. Syst. Manuf.* 2007;**1**(2).
- [35] Rao, R.V., 2011. Modeling and optimization of Modern Machining processes. Springer series in advanced manufacturing.
- [36] Roy P, Pritam Roy, Chakrabarti A. Modified shuffled frog leaping algorithm with genetic algorithm crossover for solving economic load dispatch problem with valve-point effect. *Appl. Soft Comput* 2013;**13**:4244–52.
- [37] Srivatsava PR. Optimal test sequence generation using firefly algorithm. *Swarm Evolut. Comput.* 2013;**8**:44–53.
- [38] Tamjidy M, Shahla P. Biogeography based optimization (BBO) algorithm to minimize non-productive time during hole-making process. *Int. J. Prod. Res.* 2015;**53**(6):880–1894.