Management and Production Engineering Review
Volume 9 • Number 3 • September 2018 • pp. 71–78
DOI: 10.24425/119536

SEQUENCE OPTIMIZATION OF HOLE-MAKING OPERATIONS FOR INJECTION MOULD USING SHUFFLED FROG LEAPING ALGORITHM WITH MODIFICATION

Amol Macchindra Dalavi[1], Padmakar Jagannath Pawar[2], Tejinder Paul Singh[1]

[1] *Department of Mechanical Engineering, Symbiosis Institute of Technology, Symbiosis International University, India*
[2] *Department of Production Engineering, K. K. Wagh Institute of Engineering Education and Research, Nashik, India*

Corresponding author:
Amol Macchindra Dalavi
Department of Mechanical Engineering
Symbiosis Institute of Technology
Symbiosis International University
Gram Lavale, Mulshi 412115, Pune, India
phone: (+91) 9921517645
e-mail: amol.dalavi83@gmail.com

Received: 2 February 2017
Accepted: 1 September 2018

Abstract
Tool travel and tool switch planning are the two major issues in hole-making operations of industrial part which involves drilling, tapping etc. operations. It is necessary to find the sequence of operations, which minimizes the total non productive time and tool switch time of hole-making operations depending upon the hole location and the tool sequence to be followed. In this work, an attempt is made to reduce total non-productive time and tool switch time of hole-making operations by applying a relatively new algorithm known as shuffled frog leaping with modification for the determination of optimal sequence of operations. In order to validate the developed shuffled frog leaping algorithm with modification, it is applied on six different problems of holes and its obtained results are compared with dynamic programming (DP), ant colony algorithm (ACO), and immune based evolutionary approach (IA). In addition, an application example of injection mould is considered in this work to demonstrate the proposed approach. The result obtained by shuffled frog leaping algorithm with modification is compared with those obtained using ACO, particle swarm optimization (PSO) algorithm and IA. It is observed that the results obtained by shuffled frog leaping algorithm with modification are superior to those obtained using ACO, PSO and IA for the application example presented.

Keywords
hole-making operations, injection mould, shuffled frog leaping algorithm with modification.

Introduction

Injection mould involves a large number of holes with different diameters. In the process of machining of hole-making operations of mould require various tools with different diameters to achieve the final size diameter of the hole. Machining of hole or holes may require tool or combination of tools to achieve the final size diameter of the hole. For hole H_3 shown in Fig. 1, may require one of $\{T_1, T_2, T_3\}$, $\{T_1, T_3\}$, $\{T_2, T_3\}$, and $\{T_3\}$ tools to obtain the final size. Several combinations of tools for individual hole to achieve the desired size of the hole has impact on optimum cutting speeds, tool switch time and tool travel distance [1].

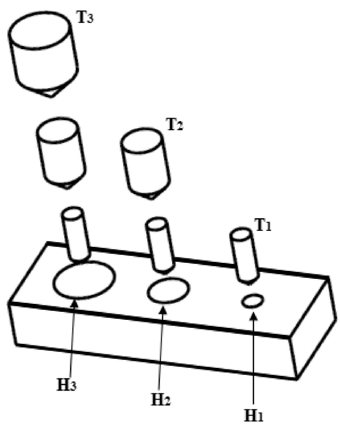

Fig. 1. Diagrammatic representation of part which requires various tools to machine a hole to its final size [1].

Tool switching and table movement from one position to another takes more amount of machining time in machining operations. To reduce the tool travel, the spindle is not moved till hole is completely machined by various tools which increase the tool switch time. On the other side to reduce tool switching time, the tool may be used for all operations which increases the tool travel cost. Tabu-Search technique used to reduce the overall machining cost of hole-making operations of application example of plastic injection mould [1]. Typically, 70% of overall time in manufacturing processes is spent on tool and part movements [2]. Process planning in hole-making operations carried out using generic knowledge based methodology [3]. Practical use of computer-aided process planning (CAPP) system presented to minimize the total processing time of machining of injection moulds [4]. Genetic algorithm (GA) used to obtain the least cutting tool path for machine operations [5]. Case study attempted using ant colony optimization (ACO) algorithm for achieving the optimal path of machining holes in a typical industrial part [6]. Optimal sequence of hole-making operations obtained using an immune based evolutionary approach (IA) [7]. Particle swarm optimization (PSO) evolutionary based method used for finding optimal sequence of hole-making operations with various tools in machining [8]. Algorithm based on geographic classification of biological organism used to reduce the tool travel and tool switching time during hole-making operations [9]. Optimal sequence of drilling path for rectangular matrix of holes carries out by using ACO [10]. Operation sequencing process is carried out using particle swarm optimization algorithm [11]. A case study of five-axis prismatic parts for sequencing the operations presented using particle swarm optimization approach [12]. Ant colony algorithm used to get the best sequence of operations that gives the least cutting trajectory in computer numerical control machine [13].

Opposition based learning ingrained shuffled frog-leaping algorithm proposed and tested on various benchmark functions and real life problem [14]. Detailed consideration is given to the meaning of “Biologicalisation” from the view of the design, function and operation of products, manufacturing processes, manufacturing systems, supply chains and industries [15]. Optimal foraging algorithm to minimize the total processing cost by identifying the optimal sequence of drilling operations [16]. Critical review carried out on multiple-hole drilling path optimization [17].

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 [1] has considered three elements of total processing costs, tooling & machining cost, non-productive tool travel cost and tool switching cost.

It is likewise 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 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 [18]. GA requires more parameters [19]. In ACO algorithm, convergence is slow due to pheromone evaporation and CPU time requirement is more [19]. 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 [19]. Biogeography-based optimization (BBO) is poor in exploiting the solutions. Also there is no provision for selecting the best members from each generation [20].

It is necessary to use non-traditional optimization algorithm which is robust and gives correct solution for complex problems [18]. In this work recently developed optimization algorithm known as shuffled frog leaping algorithm (SFLA) with modification [21, 22] is applied to reduce total non productive time of hole-making operations through determination optimal sequence for hole-making operations.

Modified shuffled frog leaping algorithm

The 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 [21]. 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 behaviour of individual frog is influenced by neighbouring frog to obtain the best solution. Thereafter, the frog population is shuffled and the local & global search mechanisms were carried out until convergence criteria are achieved [23].

Shuffled frog leaping algorithm can be used for discrete optimization problems [21]. It has been successfully applied to some engineering optimization problems such as economic load dispatch problem [24], multiobjective optimal power flow [25], project management [22] and traveling salesman problem [26].

The most well-known benefit of shuffled frog leaping algorithm is its fast convergence speed [19]. The shuffled frog leaping algorithm combines the advantages of the both the genetic-based memetic algorithm (MA) and the social behavior-based PSO algorithm [27, 28].

In order to widen the search capability and overcome premature convergence, the local search mechanism is modified in existing shuffled frog leaping algorithm as discussed in steps below [21]

1. Generate virtual frog randomly called population 'p''.
2. Estimate the fitness of the population.
3. Group the population in a descending manner.
4. Divide the population into 'm'' memeplexes,
5. Frogs i is expressed as $X_i = (X_{i1}, X_{i2}, \ldots, X_{is})$, where S stands for number of variables.
6. Select the worst frog X_w and the 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 Eq. (1)
$$X_{i+1} = w \cdot X_i + C_1 \cdot r \cdot (X_b - X_w), \tag{1}$$
where X_{i+1} – new position of frog, w – inertia weight, X_i – previous position of frog, C_1 – search acceleration factor with positive values, r –random number values between 0 to 1, X_b – position of best frog among the memeplexes, X_w – position of the worst frog among the memeplexes.
9. If fitness of new frog generated by above Eq. (1) better than previous frog then replace it with new frog. If not, apply the Eq. (2).
Weight factor 'w' is introduced on right hand side of Eq. (1) in order to widen the search capability & to avoid premature convergence. Similarly, 'w' is introduced in right-side of Eq. (2) below.
When the difference between the 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. (1), search acceleration factor with positive values of C_1 is introduced [22].
Similarly, C_2 is introduced in right hand side of Eq. (2).
10. Equation (2)
$$X_{i+1} = w \cdot X_i + C_2 \cdot r \cdot (X_g - X_w), \tag{2}$$
where C_2 – search acceleration factor with positive values, X_g – position of global best frog in search space which best among all frogs.
11. 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.
Search and the frog shuffling process continued until convergence criteria are satisfied.
In next section, computational experiments using proposed shuffled leaping algorithm with modification on bench marks problems of holes are discussed.

Computational experiments on bench mark problems

The performance of the proposed shuffled leaping algorithm with modification is evaluated through comparing its results of six bench mark problems consisting of 5, 10, 15, 20, 25 and 50 holes [6] with dynamic programming, ant colony optimization [6] and immune algorithm [7]. In this work it is assumed that tool will visit the each hole once and will return to initial position.

The positioning of holes in the benchmark problems, in order to reproducible is taken as follows [6]:

- The number of rows in the part is $\sqrt{J}$, which indicates the greatest integer value smaller than or equal to $\sqrt{J}$ and J is the total number of holes.
- It is assumed that there is 2 cm gap between individual hole in each direction.
- The location of the j-th hole is obtained as shown in Fig. 2 drawn using Pro/Engineer 5.0.

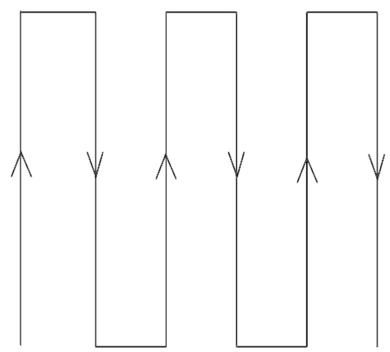

Fig. 2. The procedure to decide the position of the j-th hole in the part.

For example, when $J = 10$, the location of holes in the part are shown in Fig. 3 drawn using Pro/Engineer 5.0. For this case of 10 holes, the number of rows is $\lfloor\sqrt{10}\rfloor = 3$. Possible number of sequences for machining of 5, 10, 15, 20, 25, and 50 holes are 120, $3.6288e^{+6}$, $1.3076e^{+12}$, $2.4329e^{+18}$, $1.5511e^{+25}$, $3.0414e^{+64}$ respectively.

Table 1 gives the comparison of the results obtained by shuffled frog leaping algorithm with modification with dynamics programming (DP), ant colony optimization and Immune based evolutionary approach for objective function values.

The results of six bench mark problems using proposed shuffled frog leaping algorithm with modification in Sec. 3 motivates to apply it on application example discussed in Secs. 4 and 5.

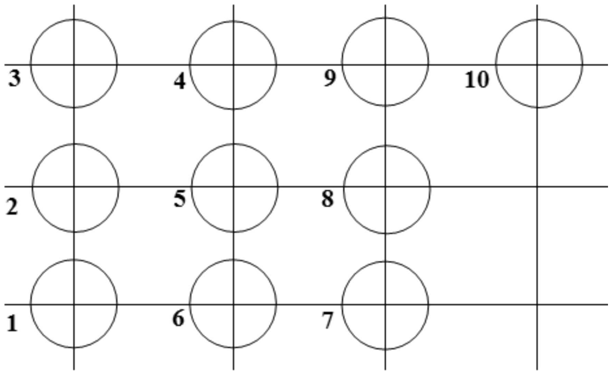

Fig. 3. Example of location of 10 holes on part [6].

Table 1
Comparison of results of bench mark functions.

Method	Number of holes [J]	Objective function value	Best sequence obtained
DP [6]	5	12	{1, 2, 3, 4, 5, 1}
	10	24	{1, 2, 3, 4, 5, 8, 9, 10, 7, 6, 1}
	15	32	{1, 2, 3, 4, 5, 8, 9, 10, 11, 14, 15, 13, 12, 7, 6, 1}
	20	40	{1, 2, 3, 4, 5, 6, 7, 10, 11, 12, 13, 20, 19, 14, 15, 18, 17, 16, 9, 8, 1}
	25	N/A	Not Applicable
	50	N/A	Not Applicable
ACO [6]	5	12	{1, 2, 3, 4, 5, 1}
	10	24	{1, 2, 3, 4, 5, 8, 9, 10, 7, 6, 1}
	15	32	{1, 2, 3, 4, 5, 8, 9, 10, 11, 14, 15, 13, 12, 7, 6, 1}
	20	40	{1, 2, 3, 4, 5, 6, 7, 10, 11, 12, 13, 20, 19, 14, 15, 18, 17, 16, 9, 8, 1}
	25	52	{1, 10, 11, 20, 21, 22, 19, 18, 17, 14, 15, 16, 25, 24, 23, 13, 12, 9, 8, 7, 6, 5, 4, 3, 2, 1}
	50	136	{1, 13, 10, 9, 20, 23, 34, 35, 22, 36, 50, 49, 48, 47, 45, 44, 41, 30, 29, 42, 43, 46, 39,38, 32, 31,40, 37, 33, 24, 26, 27, 25, 19, 21, 8, 7, 6, 5, 4, 11, 18, 17, 12, 3, 2, 16, 15, 28, 14, 1}
IA [7]	5	12	{1, 2, 3, 4, 5, 1}
	10	24	{1, 2, 3, 4, 9, 10, 8, 5, 7, 6, 1}
	15	32	{1, 2, 3, 4, 9, 10, 15, 14, 13, 12, 11, 7, 8, 5, 6, 1}
	20	40	{1, 2, 4, 5, 12, 13, 20, 19, 14, 15, 18, 17, 16, 9, 10, 11, 6, 7, 8, 1}
	25	52	{1, 2, 3, 4, 5, 6, 15, 16, 25, 24, 17, 7, 14, 13, 8, 9, 12, 19, 18, 23, 22, 21, 20, 11, 10, 1}
	50	104	{1, 2, 3, 4, 5, 6, 7, 8, 21, 22, 35, 36, 50, 49, 48, 47, 38, 33, 24, 25, 32, 31, 40, 45, 46, 39, 37, 34,23, 20, 9, 10, 19, 18, 11, 12, 17, 26, 27, 30, 41, 44, 43, 42, 29, 28, 15, 16, 13, 14, 1}
Modified shuffle frog leaping algorithm	5	10.82	{1, 2, 3, 5, 4, 1}
	10	21.64	{1, 2, 3, 4, 5, 9, 10, 8, 7, 6, 1}
	15	30.82	{1, 2, 3, 4, 5, 6, 7, 11, 12, 13, 14, 15, 10, 9, 8, 1}
	20	40	{1, 7, 2, 3, 4, 5, 6, 11, 12, 10, 15, 14, 13, 20, 19, 18, 17, 16, 9, 8, 1}
	25	52	{1, 2, 3, 4, 5, 6, 7, 8, 9, 12, 13, 14, 15, 16, 17, 24, 25, 23, 21, 22, 18, 19, 20, 11, 10, 1}
	50	116	{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 1}

Formulation of an optimization model for hole-making of operations

In order to reduce the total non-productive time of hole-making operation, the following optimization model is considered as given [6] considering following components of total time:

Tool travel time or airtime

Tool travel time is the time required for tool to move from location to another location of part, for machining operation [1].

Tool switch time

It occurs whenever a different tool is used for next operation. If tool type required for operation is not available on spindle, then the required tool must be loaded on the spindle prior to performing operation [1].

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 [7]. The airtime a_{de} needed from hole d to hole e is given by

$$a_{de} = \frac{|x_d - x_e|}{v_x} + \frac{|y_d - y_e|}{v_y}, \tag{3}$$

where $v_x = 2\pi r_x N_x / g_x(60)$ and $v_y = 2\pi r_y N_y / g_y(60)$ are speeds of x-axis direction and y-axis direction, respectively [28]. Note that if $v_x = v_y = 1$ in Eq. (3), then $a_{de} = l_{de}$. Where $l_{de} = |x_d - x_e| + |y_d - y_e|$, the rectilinear distance between holes d and e where hole d is located at (x_d, y_d) and hole e is located at (x_e, y_e).

Min

$$\sum_{d=1}^{D}\sum_{e=1}^{m_d}\sum_{\substack{d'=1\\ d'\neq 1}}^{D}\sum_{e'=1}^{m'_d}\sum_{f=1}^{M-1} a_{de}x_{def}x_{d'e'f+1}$$

$$\text{Min} \tag{4}$$

$$+\sum_{d=1}^{D}\sum_{e=1}^{m_d}\sum_{d'=1}^{D}\sum_{e'=1}^{m'_d}\sum_{j=1}^{M-1} S_{de,d'e'} \cdot \delta(T_{de}, T_{d'e'}) x_{def} x_{d'e'f+1}.$$

Subject to

$$\sum_{f=1}^{M-1} x_{def} = 1, \quad d = 1,2,...,D, \quad e = 1,2,...,m_i, \tag{5}$$

$$\sum_{d=1}^{D}\sum_{e=1}^{m_d} x_{def} = 1, \quad f = 1,2,...,M, \tag{6}$$

$$x_{def} \le \sum_{f'=f+1}^{M} x_{d,e+1,f'}, \quad d = 1,2,...,D, \quad e = 1,2,...,m_i - 1, \tag{7}$$

$$x_{def} \in \{0,1), \forall d,e,f, \tag{8}$$

where D the overall holes to be machined in the part, m_d the overall operations required for hole d, $d = 1,2,\ldots,D$, $M = m_1 + m_2 + \ldots + m_D$ the overall of operations in the part, T_{de} 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 $d = 1,2,\ldots,D$, $e = 1,2,\ldots,m_i$, $f = 1,2,\ldots,M$, $\delta(T_{de}, T_{d'e'}) = 1$ if $T_{de} \neq T_{d'e'}$, otherwise 0.

Application example

The proposed shuffled frog leaping algorithm with modification was coded as per mathematical modal given in Sec. 4, in order to obtain the optimal path of hole-making operations of holes for a part [6] shown in Fig. 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 Sec. 4 is shown in Fig. 4. Figure 5 shows the number sequence of holes on application example part. In the process of machining several industrial parts such as dies and moulds, operations like drilling, reaming, and taping account for a huge segment of processing. The details of tools diameter are given in Table 2 [6]. Details of tool switch times in minutes for this application example are given in Table 3 [6].

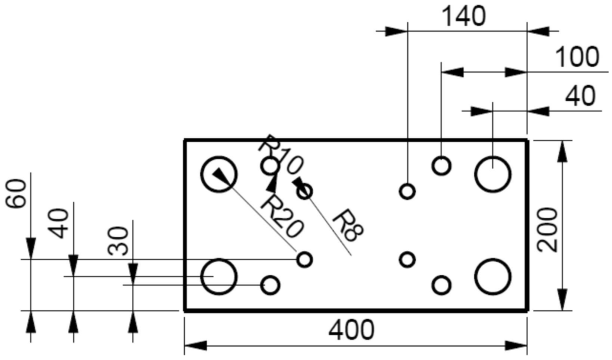

Fig. 4. Top view of example part [6].

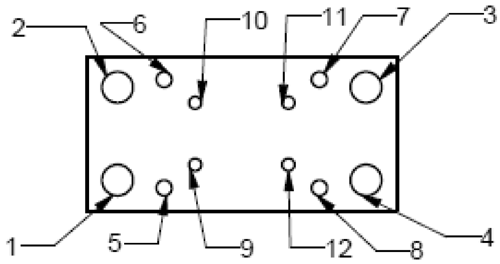

Fig. 5. Hole numbering of example part.

Table 4 [6] presents combinations of tools required for machining of individual hole of application example presented in this section. For example, for machining of R20 hole shown in Fig. 4 required 1, 3, 4 and 5 tools given in Table 4.

Table 2
Details of tools required hole-making for application example.

Tool type d	Drill					Reamer
	1	2	3	4	5	6
Tool diameter [mm]	10	15.8	20	30	40	16

Table 3
Details of tool switch times in minutes for application example.

Next in line tool ↓	Previous tool ⟶					
1	0.0	0.6	0.2	0.4	0.4	0.9
2	0.6	0.0	0.8	1.2	0.4	0.8
3	0.2	0.8	0.0	0.6	1.4	1.2
4	0.4	1.2	0.6	0.0	0.4	0.7
5	0.4	0.4	1.3	0.5	0.0	0.8
6	0.5	0.5	1.2	0.2	0.8	0.0

Table 4
Desired set of tools for machining individual hole in the application example.

Hole	R8	R10	R20
Desired tools	1-2-6	1-3	1-3-4-5

Results and discussion

In this section obtained results of optimization of proposed shuffled frog leaping algorithm with modification are compared with those obtained using ant colony algorithm [6], particle swarm optimization algorithm [8] and immune based evolutionary approach [7] for the application example mentioned in Sec. 5.

Considering tool information given in table 2 and tool switch times shown in Table 3. Results of shuffled frog leaping algorithm with modification are discussed below:

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

$C_1 = 1.0$,

$C_2 = 0.95$,

$w = 1.0$,

Frog population = 50,

Quantity of memeplexes = 10,

Quantity of sub frogs = 5,

Number of iterations = 100.

For the above parameter setting, the results of optimization using shuffled frog leaping algorithm with modification are compared as follows in Table 5.

Table 5
Result comparison of application example.

Method	Objective function values [min]	Air time [min]	Tool switch time [min]	Best possible sequence
ACO [6]	8.8	4.58	4.3	(5,1) (9,1) (12,1) (8,1) (7,1) (11,1) (10,1) (6,1) (2,1) (1,1) (5,3) (8,3) (11,2) (12,2) (9,2) (10,2) (6,3) (7,3) (3,1) (4,1) (4,3) (3,3) (2,3) (1,3) (1,4) (2,4) (3,4) (4,4) (12,6) (9,6) (10,6) (11,6) (3,5) (4,5) (1,5) (2,5)
PSO [8]	7.42	4.02	3.4	(9,1) (2,1) (6,1) (10,1) (3,1) (8,1) (4,1) (12,1) (11,1) (7,1) (7,3) (6,3) (5,1) (1,1) (2,3) (1,3) (5,3) (8,3) (4,3) (3,3) (3,4) (4,4) (4,5) (3,5) (11,2) (12,2) (9,2) (10,2) (10,6) (9,6) (12,6) (11,6) (2,4) (1,4) (1,5) (2,5)
IA [7]	6.26	3.86	2.4	(1,1) (5,1) (8,1) (4,1) (3,1) (7,1) (12,1) (9,1) (10,1) (2,1) (6,1) (6,3) (2,3) (1,3) (5,3) (8,3) (4,3) (3,3) (7,3) (11,1) (11,2) (12,2) (9,2) (10,2) (10,6) (11,6) (12,6) (9,6) (1,4) (2,4) (3,4) (4,4) (4,5) (3,5) (2,5) (1,5)
Modified shuffle frog leaping algorithm	6.12	3.72	2.4	(1,1) (5,1) (2,1) (4,1) (3,1) (7,1) (12,1) (8,1) (10,1) (9,1) (6,1) (6,3) (2,3) (1,3) (5,3) (8,3) (4,3) (3,3) (7,3) (11,1) (11,2) (12,2) (9,2) (10,2) (10,6) (11,6) (12,6) (9,6) (1,4) (2,4) (3,4) (4,4) (4,5) (3,5) (2,5) (1,5)

Table 5 gives the best possible sequence using shuffled frog leaping algorithm with modification having optimum values of airtime of 3.72 minutes and tool switch time of 2.4 minutes.

Conclusion

Optimization of hole-making operations involves a large number of possible sequences to complete hole-making operations on the part depending upon the hole location and tool sequence to be followed. To achieve this, proper determination of operation sequence which minimizes the total non productive time of hole-making operations is essential. This paper presents recently developed shuffled frog leaping algorithm with modification. The performance of the proposed shuffled leaping algorithm with modification is validated through checking its results of six bench mark problems consisting of 5, 10, 15, 20, 25 and 50 holes with dynamic programming, ant colony optimization and immune algorithm. The obtained results of benchmark functions using shuffled frog leaping algorithm with modification shows that for 5,10,15 holes problems respectively it is 10.0%, 10.0%, 4% superior to dynamic programming, ant colony optimization and immune algorithm. For 20 holes benchmark problem results obtained using proposed algorithm is same as of dynamic programming, ant colony optimization and Immune algorithm. For 25 holes benchmark problem results obtained using proposed algorithm are same as of ant colony optimization and immune algorithm. However for 50 holes benchmark problem proposed algorithm results are 12% inferior to Immune algorithm and 14% superior to ant colony optimization algorithm. Also, in this work, shuffled frog leaping algorithm with modification is applied on an application example to reduce the total non productive time and tool switch time in hole-making operations. The obtained results are compared with those obtained using ant colony algorithm, particle swarm optimization algorithm and immune based evolutionary approach. It is seen that the results of optimization achieved by shuffled frog leaping algorithm with modification for application example are 30%, 18% & 2.23 % superior to those obtained using ant colony algorithm, particle swarm optimization algorithm and Immune based evolutionary approach respectively in 100 generations. The improvement obtained by using shuffled frog leaping algorithm with modification is thus significant and clearly indicates the potential of this method to solve real life problems related to hole making for various industrial applications.

Shuffled Frog Leaping Algorithm with modification showed better results in various benchmarks problems of hole-making operations. However, in some cases the effectiveness has not been shown. This phenomenon needs to further investigated and the algorithm to be modified further by taking up research in future so that it works better on all kinds of problems and would be applied on industrial case studies of injection mould.

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

[1] Kolahan F, Liang M., *Optimisation of hole-making operations: a tabu-search approach*, International Journal of Machine Tools & Manufacture, 40, 1735–1753, 2000.

[2] Merchant R.L., *World trends and prospects in manufacturing technology*, International Journal for Vehicle Design, 6, 121–138, 1985.

[3] Luong L.H.S., Spedding T., *An integrated system for process planning and cost estimation in hole-making*, International Journal of Manufacturing Technology, 10, 411–415, 1995.

[4] Alam M.R., Lee K.S., Rahman M. et al., *Process planning optimisation for the manufacture of injection moulds using a genetic algorithm*, International Journal of Computer Integrated Manufacturing, 16, 3, 181–191, 2003.

[5] Qudeiri J.A., Hidehiko Y., *Optimisation of operation sequence in CNC machine tools using genetic algorithm*, Journal of Advanced Mechanical Design, Systems, and Manufacturing, 1, 2, 2007.

[6] Ghaiebi H., Solimanpur M., *An ant algorithm for optimisation of hole-making operations*, Computers and Industrial Engineering, 52, 2, 308–319, 2007.

[7] Hsieh Y.C., Lee Y.C., You P.S., *Using an effective immune based evolutionary approach for the optimal operation sequence of hole-making with multiple tools*, Journal of Computational Information Systems, 7, 2, 411–418, 2011.

[8] Hsieh Y.C., Lee Y.C., You P.S., Chen T.C., *Optimal operation sequence of hole-making with multiple tools in manufacturing: a PSO evolutionary based approach*, Key Engineering Materials, Accepted, in press, 2010.

[9] Tamjidy M., Shahla P., *Biogeography based optimization (BBO) algorithm to minimize non-productive time during hole-making process*, Int. J. Production Research, 53, 6, 880–1894, 2015.

[10] Adel T.A., Mohamed F.A., Karim H., *Optimum drilling path planning for a rectangular matrix of holes using ant colony optimisation*, International Journal of Production Research, 49, 19, 5877–5891, 2011.

[11] Guo et al., *Operation sequencing optimization using a particle swarm optimization approach*, Proc IMechE Part B: Journal of Engineering Manufacture, 220, 12, 1945–1958, 2006.

[12] Guo et al., *Operation sequencing optimization for five-axis prismatic parts using a particle swarm optimization approach*, Proc. IMechE Part B: Journal of Engineering Manufacture, 223, 5, 485–497, 2009.

[13] Kiani K., Sharifi M., Shakeri M., *Optimization of cutting trajectory to improve manufacturing time in computer numerical control machine using ant colony algorithm*, Proc. IMechE Part B: Journal of Engineering Manufacture, 228, 7, 811–816, 2014.

[14] Tarun Kumar Sharma, Millie Pant, *Opposition based learning ingrained shuffled frog-leaping algorithm*, Journal of Computational Science, 21, 307–315, 2017.

[15] Byrne G., Dimitrov D., Teti R., Houten F., Wertheim R., *Biologicalisation: biological transformation in manufacturing*, CIRP Journal of Manufacturing Sci. and Tech., 21, 1–32, 2018.

[16] Wei-Bo Zhang, Guang-Yu Zhu, *Drilling path optimization by optimal foraging algorithm*, IEEE Transactions on Industrial Informatics, 14, 7, 2018.

[17] Dewil R., Kü¸çükğlu İ., Luteyn C. et al., *A critical review of multi-hole drilling path optimization*, Arch. Computat. Methods Eng., 2018, https://doi.org/10.1007/s11831-018-9251-x.

[18] Venkata Rao R., *Modeling and optimization of Modern Machining processes*, Springer Series in Advanced Manufacturing, 2011.

[19] Elbeltagi E., Tarek H., Donal G., *Comparison among five evolutionary based optimisation algorithms*, Advanced Engineering Informatics, 19, 43–53, 2005.

[20] Ammu P.K., Sivakumar K.C., Rejimoan R., *Biogeography – based optimization – a survey*, International Journal of Electronics and Computer Science Engineering, 2, 1, 154–160, 2012.

[21] Eusuff M.M., Lansey K.E., Pasha F., *Shuffled frog-leaping algorithm: a memetic metaheuristic for discrete optimisation*, Eng. Optim., 38, 2, 129–154, 2006.

[22] Elbeltagi E., Tarek H., Donald G., *A modified shuffled frog-leaping optimization algorithm: applications to project management*, Structure and Infrastructure Engineering, 3, 1, 53–60, 2007.

[23] Luo Ping, Qinang Lu, Chenxi Wu, *Modified shuffled frog leaping algorithm based on new searching strategy*, Seventh International Conference on Natural Computation, 2011.

[24] Roy P., Roy Pritam, Chakrabarti A., *Modified shuffled frog leaping algorithm with genetic algorithm crossover for solving economic load dispatch problem with valve-point effect*, Applied Soft Computing, 13, 4244–4252, 2013.

[25] Niknam T., Narimani M.R., Jabbari M., Malekpour A.R., *A modified shuffle frog leaping algorithm for multi-objective optimal power flow*, Energy, 36, 6420–6432, 2011.

[26] Luo X.H., Yang Y., Li X., *Solving TSP with shuffled frog-leaping algorithm*, Proc. ISDA, 3, 228–232, 2008.

[27] Kennedy J., Eberhart R.C., *Particle swarm optimisation*, Proc. IEEE Conf. Neural Network, 4, 1942–1948, 1995.

[28] Onwubolu G.C., Clerc M., *Optimal path for automated drilling operations by a new heuristic approach using particle swarm optimisation*, International Journal of Production Research, 42, 3, 473–491, 2004.