Pergamon

*Computers ind. Engng* Vol. 31, No. 1/2, pp. 371 – 374, 1996

0360-8352/96 $15.00 + 0.00

S0360-8352(96) 00154-4

# A TABU SEARCH APPROACH TO OPTIMIZATION OF DRILLING OPERATIONS

F. Kolahan and M. Liang
Department of Mechanical Engineering, University of Ottawa
Ottawa, Ontario, Canada K1N 6N5

**Abstract** -- This paper reports a tabu-search approach to minimize the cost in hole making processes. Four issues, namely, tool travel scheduling, tool switch scheduling, tool selection, and machining speed specification have been simultaneously addressed in this study. This problem has a structure similar to the Travelling Salesman Problem (TSP) and hence is NP-complete. To provide an efficient solution procedure, a tabu search approach has been proposed. The performance of the proposed approach is tested using an example problem.

## 1. INTRODUCTION

Hole-making accounts for a large portion of machining operations for many industrial parts such as dies and moulds. Due to the point-to-point machining feature in hole-making, a considerable amount of the processing time is spent in switching tools and moving from one drilling location to another. In addition, to drill a given hole to its final size, different sets of tools with different cutting speeds may be used which directly influences tooling and machining costs. The problem is to select a set of operations (tool-hole combinations) along with the best cutting speeds and sequence those operations in such a way that the total processing cost is minimized.

The above mentioned problem has not been directly addressed in the literature. Nevertheless, the similar problem for punching operations is studied by a number of researchers. In punching operations the main concern is to minimize those costs related to the non-cutting tool travel and the problems of tool selection and tool assignment do not exist [2]. However, the problem under consideration involves several issues. They are: a) tool travel routing, b) tool switch scheduling, c) tool-hole grouping, and d) selection of cutting speed for each operation. These issues are interlocked and concurrently contribute to the total processing cost. For this reason, the authors propose that the above four issues be solved simultaneously.

## 2. PROBLEM STATEMENT

### 2.1 Notation

The following notation will be used in this paper:
$i$=tool type index, $i=1,\ldots,I$; $j$=hole index, $j=1,\ldots,J$; $I_j$=set of tools used to drill hole $j$ to its final size; $N_j$ =number of tools in $I_j$; $ij$=operation index, $i=1,\ldots,N_j$, $j=1,\ldots J$; $s$=sequence index; $ij_{(s,l)}$ =an operation performed by tool type $i$ on hole $j$ in the $l$th position of sequence $s$; $L_j$ =depth of hole $j$, including the clearance; $t^*_{ij}$ =optimum processing time of operation $ij$; $T^*_{ij}$=optimum life of tool type $i$ performing cutting operation on hole $j$; $v^*_{ij}$=optimum cutting speed of tool type $i$ performing cutting operation on hole $j$; $f_i$=recommended feed rate for tool type $i$; $d_i$ =diameter of tool type $i$; $p_{ij(s,l)}$=tool travelling distance between the previous tool location and the current hole $j$ in the $l$th position of sequence $s$; $q_{ij(s,l)}$=tool switch time from the previous tool type to tool type $i$ required to perform current operation $ij$ in the $l$th position of sequence $s$; $a$=cost per unit non-productive travelling distance; $b$=cost per unit tool switch time; $c_m$ = cost per unit machining time; $c_i$=cost of tool type $i$; $W_s$ =a 0-1 integer variable, $W_s$ = 1 if sequence $s$ is selected; 0, otherwise; $x_{ij(s,l)}$=a 0-1 integer variable, $x_{ij(s,l)}=1$ if $\{\forall\, l'\leq l,\, j'=j \mid d_i > d_{i'}\ ;\ i,i'\in I_j\}$; 0, otherwise; $G(s)$=total processing cost for sequence $s$; $N(s)$=set of operations sequences in the neighbourhood $s$; $T\_size$=size of tabu list; $Mmax$ =maximum allowed number of moves;

$M\_ctr$=a counter used to record total number of moves made so far; $s^*$ =best sequence in the current neighbourhood; $s^{**}$best sequence found in the last neighbourhood;

## 2.2 *The problem*

Generally, a part, e.g., a plastic injection mould, may have many holes with different design specifications. For such a part, a particular tool may be required by several holes and also tools of different diameters may be used to drill a single hole to its final size. To reduce the tool traverse, it may be suggested that the spindle should not be moved until a hole is completed using several tools of different diameters. This however may lead to excessive tool switches. By the same token, though tool switches can be reduced by completing all operations on all the holes that require the current tool, the travel time may increase. Furthermore, the amount of tool movement and the number of tool switches will depend on which set of tools are used to drill each hole to its final size. The machining cost and tool cost are also affected by the tool combination used for each hole.

The structure of hole making problem is similar to that of Travelling Salesman Problem (TSP) in which each node (operation) in a tour (sequence of operations) must be visited only once. However this problem is more complicated than TSP as, unlike the TSP, the cost of visiting each node is determined by its position in the sequence of operations (constructed tour) and some nodes may be visited at no cost; i.e. some tools may not be used for some holes even though such operations are feasible. To efficiently solve the above mentioned problem, we report a tabu-search approach.

# 3. PROBLEM FORMULATION

The tooling and machining costs for a single operation $ij$ can be written as:

$$C_{ij} = \frac{t_{ij}}{T_{ij}} c_i + t_{ij} c_m \quad (1)$$

where the machining time, $t_{ij}$, is calculated by:

$$t_{ij} = \frac{\pi d_i L_j}{1000 V_{ij} f_i} \quad (2)$$

and the tool life expressions, $T_{ij}$, for different hole making processes are given by [3]:

$$T_{ij} = \left(\frac{8 d_i^{0.4}}{v_{ij} f_i^{0.7}}\right)^5 \quad (3) \qquad T_{ij} = \left(\frac{18.4 d_i^{0.4}}{v_{ij} \alpha_{ij}^{0.2} f_i^{0.5}}\right)^5 \quad (4) \qquad T_{ij} = \left(\frac{12.1 d_i^{0.3}}{v_{ij} \alpha_{ij}^{0.2} f_i^{0.65}}\right)^{2.5} \quad (5)$$

Equation (3) provides the tool life for drilling a new hole (without a pilot drill). Equations (4) gives the tool life for a drill bit when it is used to enlarge a hole. If a hole is enlarged by reaming or tapping Equation (5) is used. As the depth of cut is fixed in hole making, the optimum cutting speed, $V_{ij}^*$, for the constant feed rate can be found by solving the following differential equation:

$$\frac{dC_{ij}}{dv_{ij}} = 0 \quad (6)$$

where $V_{ij}$ is given by Equations (3) to (5) depending on the type of operation. Now the combined problem of tool selection, machining speed specification, tool travel scheduling, and tool switch scheduling can be formulated as the following non-linear mixed integer model:

$$\underset{s}{Min}\ G(s) = Min \sum_{s=1}^{k!} W_s \sum_{ij(s,l)=1}^{k} x_{ij(s,l)} \left( a p_{ij(s,l)} + b\, q_{ij(s,l)} + \frac{t^*_{ij(s,l)}}{T^*_{ij(s,l)}} c_i + t^*_{ij(s,l)} c_m \right) \quad (7)$$

subject to

$$\sum_{s=1}^{k!} W_s = 1 \quad (8)$$

where $k$ is the number of *possible* (feasible but not necessarily required) operations in sequence $s$. The objective is to minimize the total cost of processing a part with $J$ holes. There are two sets of 0-1 decision variables; $x_{ij(s,l)}$ related to the decision of selecting a set of *required* operations from the set of *possible* operations, and $W_s$ jointly used with the constraint set (8) to ensure that only one set of operations is selected. It is noted that for the selected sequence only the set of *required* operations,i.e., those operations with $x_{ij(s,l)} = 1$, are executed on the machine.

This model has a non-linear objective function with two sets of 0-1 decision variable. It can be solved optimally only for small size problems. To provide a quick and efficient solution procedure for larger problems, we propose a tabu search based heuristic approach.

## 4. THE SOLUTION PROCEDURE

### *4.1 Tabu search*

Tabu search, first introduced by Glover[2], is an optimization technique used to solve combinatorial optimization problems. Tabu search starts from an initial feasible solution and moves stepwise towards a good solution. To make a move, a set of neighbours solutions is generated from the current solution. A move is then made from the best solution found in the previous neighbourhood to the best permissible solution in the current neighbourhood. However, such a move may not be an improving one. To avoid the cycling problem, tabu list, *T_list*, is used. It is a list containing a certain number of previous moves which are not allowed for the current move. After each move the tabu list is updated by adding the current move and removing the oldest one from the list. During the search process the best solution found, *S_best*, is recorded and updated. The search process is terminated when a specified number of moves or a maximum allowed search time is reached. In this paper the former criterion is used.

### *4.1 Tabu search algorithm*

*Step 1. Initialize the search*

(a) Read the input data: $c_m$, $a$, $b$, *T_size*, *Mmax* and $p_{jj'}$, $c_i$, $q_{ii'}$, $l_j$, for all $i$'s and $j$'s. .

(b) Calculate total number of possible operations, $k$, and set *T_list*$=\{\varnothing\}$, *S_bes*$=\{\varnothing\}$, *M_ctr*$=0$, and *C_best*$=$*M_big* (a big positive number)

(c) Compute $G(s^{**})$ for $s^{**}$, a feasible starting sequence including $k$ "required" operations.

*Step 2 Search*

WHILE *M_ctr* < *Mmax* DO
set $G(s^*)=$ *M_big;*
  DO *mm* = 1 to *k*-1
    DO *nn* = *mm*+1 to *k*
      generate a new neighbour $s$ for $s^{**}$ by switching the operation in position *mm* and the operation in position *nn*.
      IF $s \in$ *T_list*, discard *s*, and continue; ELSE compute $G(s)$;
      IF $G(s) < G(s^*)$, set $G(s^*) \leftarrow G(s)$ and $s^* \leftarrow s$; ELSE discard s, and continue;
    ENDDO
  ENDDO
Set $G(s^{**}) \leftarrow G(s^*)$, $s^{**} \leftarrow s^*$ and update *T_list*;
  IF $G(s^{**}) <$ *Cbest*, set *Cbest* $\leftarrow G(s^{**})$, *Sbest* $\leftarrow s^{**}$, *M_ctr* $\leftarrow$ *M_ctr* $+1$;
  ELSE set *M_ctr* $\leftarrow$ *M_ctr* $+1$;
ENDWHILE

The algorithm is coded in C++ and run on a 486 DX 33 with 4 MB RAM.

## 5. CASE STUDY

The proposed procedure was used to determine the set of tools, sequence of operations, and cutting speeds for the part shown in Figure 1 which requires drilling, reaming, and tapping operations. The process parameters are: $c_m$=\$1/min, $a$=\$0.0008/mm, and $b$=\$1/min. The tool switch times are considered to be asymmetric in the range of 0.2 to 1.5 minutes depending on the tool location in the tool magazine. The computation was carried out for 10 different starting sequences. For each starting sequence, the search was terminated after 100 moves. To investigate the effect of tabu-list size on search performance, the search was repeated for tabu-list sizes 10, 15, 20, and 40. The initial set of possible tool-hole combinations used in this example is listed in Table 1. As listed in Table 1, there are a total of $80(=\Sigma_j N_j)$ possible operations. The neighbourhood size for this problem is $k(k-1)/2=3160$.

| $j$ | GP1-GP4 | GE1-GE4 | PR1-PR4 | C1-C4 | C'1-C'4 | P1-P4 | EB1-EB6 | ES1-ES2 |
|---|---|---|---|---|---|---|---|---|
| $I_j$ | 1-6-7-8-11 | 3-6-7 | 3-6-10 | 1-4-9 | 1 | 1-5-12 | 3 | 2 |

Table 1 Possible tool-hole combinations

The information about tools and recommended feed rates is given in Table 2.

It is found that a tabu-list with a size less than or equal to 15 leads to cycling while tabu-lists with the sizes 20 and 40 perform equally well in terms of final results. The best result is shown in Table 3.

The final solution consists of 56 required operations (reduced from 80 possible operations given in initial solution). The associated processing cost for the final solution is $64.8, including $45.2 machining and tool costs, $11.0 tool travel cost, and $8.6 tools switch cost.

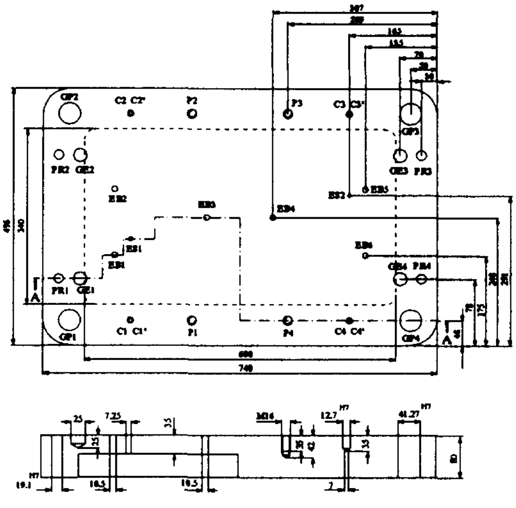

Figure 1 Upper holder of the plastic injection mould (courtesy of Komtech Plastics Corp.)

| | drills | | | | | | | | reams | | | tap |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tool type $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| $f_i$ (mm/rev) | 0.12 | 0.10 | 0.12 | 0.15 | 0.20 | 0.20 | 0.18 | 0.15 | 0.50 | 0.80 | 0.80 | 1.50 |
| $d_i$ (mm) | 7.0 | 7.25 | 10.5 | 12.5 | 13.0 | 19.0 | 25.0 | 41.0 | 12.7 | 19.1 | 41.2 | 16.0 |
| $c_i$ ($) | 10 | 12 | 15 | 15 | 14 | 20 | 26 | 50 | 55 | 70 | 85 | 45 |

Table 2 Tool data

## 6. COCLUSION

A tabu-search approach has been proposed to optimize drilling operations. This approach has been used to minimize drilling cost for a plastic injection mould. Based on the computaional result, decisions regarding tool travel, tool switch, tool selection, and machining speed specification can be simultaneously made. The computationl result shows more than 47% cost reduction can be achieved.

## REFERENCES


1. F. Glover. Tabu search: a tutorial. *Interfaces* **20**, 74-94 (1990).
2. B. Roychoudhury and J.F. Muth. Tool path optimization procedures for machine tools. *Int. J. of Computers and ind. Engng.* **28**, 731-743 (1995).
3. R. Zhao. Hand book for machinists. *Shanghai Science and Technology Press* (1992).


| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $i$ | 6- | 6- | 6- | 6- | 6- | 6- | 6- | 6- | 6- | 6- | 6- | 6- | 8- | 8- |
| $j$ | GP4 | PR4 | GE4 | GE3 | PR3 | GP3 | GP2 | PR2 | GE2 | GE1 | PR1 | GP1 | GP1 | GP4 |
| $v^*$ | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 45 | 45 |
| $i$ | 8- | 8- | 11- | 11- | 11- | 11- | 10- | 10- | 10- | 4- | 1- | 1- | 3- | 1- |
| $j$ | GP3 | GP2 | GP2 | GP1 | GP4 | GP3 | PR3 | PR2 | PR1 | C1 | C1' | C4' | EB6 | C3' |
| $v^*$ | 45 | 45 | 9 | 9 | 9 | 9 | 10 | 10 | 10 | 36 | 37 | 37 | 39 | 37 |
| $i$ | 1- | 3- | 3- | 3- | 3- | 3- | 2- | 2- | 7- | 7- | 7- | 7- | 10- | 4- |
| $j$ | C2' | EB2 | EB1 | EB3 | EB4 | EB5 | ES2 | ES1 | GE1 | GE2 | GE3 | GE4 | PR4 | C4 |
| $v^*$ | 37 | 39 | 39 | 39 | 39 | 39 | 40 | 40 | 50 | 50 | 50 | 50 | 10 | 36 |
| $i$ | 4- | 4- | 5- | 5- | 5- | 5- | 12- | 12- | 12- | 12- | 9- | 9- | 9- | 9- |
| $j$ | C3 | C2 | P2 | P3 | P4 | P1 | P1 | P2 | P3 | P4 | C4 | C3 | C2 | C1 |
| $v^*$ | 36 | 36 | 30 | 30 | 30 | 30 | 4 | 4 | 4 | 4 | 11 | 11 | 11 | 11 |

Table3 Operations sequence and the corresponding cutting speeds