IIE Transactions (1995) **27**, 417–426

Tool selection for optimal part production: a Lagrangian relaxation approach

VERNON NING HSU[1], MARK DASKIN[2], PHILIP C. JONES[3] and TIMOTHY J. LOWE[3]

[1] *School of Business Administration, George Mason University, Fairfax, VA 22030-4444, USA*
[2] *Departments of Civil Engineering and Industrial Engineering, Northwestern University, Evanston, IL 60208, USA*
[3] *College of Business Administration, The University of Iowa, Iowa City, IA 52242, USA*

Received November 1992 and accepted October 1993

This paper extends previous work on implementation problems associated with a flexible system that produces flat sheet-metal parts with interior holes. The paper makes four main contributions. First, we formulate the problem of selecting tooling and design standards to minimize the cost of producing parts as an optimization model. Second, we develop a projected subgradient algorithm for the Lagrangian relaxation of the problem by using the model's special structure to develop relationships between the Lagrangian multipliers. Third, we demonstrate that the algorithm produces close to optimal solutions (duality gap less than 2%) very quickly on a number of problems derived using a substantial data set obtained from a Chicago area firm. Fourth, an important variant of the traditional repair kit problem is shown to be a special case of the tool selection problem.

1. Introduction

The traditional method for producing flat sheet-metal parts involves punch and die technology, which requires that a separate die be designed and built for each part. After the die is mounted and adjusted, cycle time is small because a single operation forms the perimeter of the part and punches all interior holes. A significant disadvantage of punch and die technology is that set-ups, e.g., die changeovers, can be time-consuming (often more than one shift), thereby inducing large production lot sizes with correspondingly high work-in-process inventory costs. Another disadvantage is that part *redesign* often requires designing and producing a new die, an expensive and lengthy process.

Laser punch press technology, a new manufacturing method, has the potential to overcome some of the disadvantages of punch and die technology, thereby significantly increasing the firm's flexibility in production and redesign. The laser cuts the perimeter of the part as well as some irregularly shaped holes in the interior of the part. A flexible arm with a gripper withdraws tools from a rack to punch the remaining interior holes one at a time. Although cycle times are often larger than their punch and die technology counterparts, set-ups can be virtually eliminated.

One of the authors worked on a project with a medium-sized Chicago-area firm whose management was very concerned about how to introduce an expensive new laser punch press into their existing operations. The firm produced several different parts. Viewed as a group, the parts required many different sizes and shapes of hole. A major issue in implementing the new technology was selecting appropriate tooling. To see why tool selection is significant, it is important to note that the firm's existing designs had been developed under the above-mentioned punch and die technology when a different die was designed and produced for each part. Standardizing hole sizes between parts would have made the dies no less difficult or expensive to manufacture, and hence the firm previously had no compelling financial incentive to standardize. Indeed, when we examined data for existing designs of the firm's parts (over 3000 in total), we found that existing designs required over 900 different specifications for round holes alone, where a specification is a combination of hole diameter, tolerance, and material thickness. (Material thickness is included because different material thicknesses require different tools, even when hole diameters are identical.) Further, ignoring material thickness, there were over 480 different combinations of hole diameter and tolerances with over 380 different nominal diameters alone.

With the new laser punch press, the situation can be quite different. The same tool can (and will) punch holes of the same size in different parts. Further, the tool rack on the new punch press holds a limited number of tools (the press considered by the above-referenced company had a tool capacity of 200 tools). Thus, with the new technology, standardization is an important economic issue for two reasons. First, tooling is very expensive, even using standard tools in stock sizes. If custom tooling is required, it is even more costly. Thus, using fewer tools and eliminating as much custom tooling as possible can yield significant savings. Second, if the 200-tool capacity of the tool rack is exceeded, set-ups are required to mount different tools in the rack and to reprogram the system. Because eliminating set-ups is a primary advantage of the new system, implementing it in a fashion that requires frequent set-ups is an ineffective strategy.

The most obvious way to purchase tooling for producing existing designs is to order a tool corresponding to each different combination of material thickness and hole diameter. This obvious approach, however, was ineffective for the firm under consideration. (As mentioned above, this approach would have required the firm to purchase over 900 different tools for round holes alone. Furthermore, most of these tools would have to be custom-made.)

Thus, introducing new technology into an already existing production environment can be considerably more difficult than it would be in a 'clean-state' situation. A previous paper (Daskin *et al.*, 1990) developed effective algorithms for minimizing the tooling required to punch existing round-hole specifications as well as maximizing the number of holes that could be punched with a specified number of tools, and demonstrated these algorithms by using data from the firm mentioned above. This paper extends our previous work by analyzing the question of which tools to select so as to optimize *part* production when the tool rack capacity is limited. Since a given part may, in general, require holes of different sizes, optimizing part production is different from optimizing hole production. Only if all holes required by a part are punched can the part be produced.

We now give an overview of the remainder of the paper. Section 2 presents a more detailed statement of the problem and formulates the problem as an optimization model. Section 3 discusses the Lagrangian relaxation of the model, derives properties of the optimal multipliers, and presents a subgradient optimization algorithm. Section 4 presents the results of computational studies, showing that, even though examples with arbitrarily large duality gaps exist, problems using real-world data can be solved very effectively with the methods presented in Section 3. In Section 5, we show that the repair kit problem, a problem that has received considerable attention in the literature, is a special case of the problem we consider in this paper. We provide concluding remarks in Section 6.

2. Tool selection model

To formulate the problem properly, we must select a *production period*, and select model parameters based on the period. The production period might correspond to one work shift, one week, etc. In our formulation we include the cost of selecting tooling and include a penalty cost for not punching a given hole-type. Recall that a part is not completed within a production period if one or more holes in that part are not punched by the selected tools. In addition to a penalty cost for not punching a given hole, we include a fixed cost for each incomplete part regardless of the number of its unpunched holes. Such a cost may include the cost of removing the part from the machine, rescheduling and reloading the part in another shift of production, or even subcontracting the part to other vendors.

We will use the following notation in our formulation: suppose there are m holes, n tools, and s products (parts). Let

c_j = the cost of selecting tool j, $j = 1, 2, \ldots, n$;
b_i = the penalty cost for not punching hole i, $i = 1, 2, \ldots, m$;
t_k = the penalty cost for not completing part k, $k = 1, 2, \ldots, s$;
S_k = the index set for holes contained in part k, $k = 1, 2, \ldots, s$;
T_i = the index set for parts containing hole i, $i = 1, 2, \ldots, m$;
n_k = $|S_k|$ = number of distinct holes in part k, $k = 1, 2, \ldots, s$.

We define an $m \times n$, 0–1 matrix $A = (a_{ij})$, whose rows correspond to holes and columns correspond to tools, where

$$a_{ij} = \begin{cases} 1, & \text{if tool } j \text{ can punch hole } i; \\ 0, & \text{otherwise.} \end{cases}$$

We say that tool j *covers* hole i if $a_{ij} = 1$. We also define 0–1 decision variables as follows:

$$x_j = \begin{cases} 1, & \text{if tool } j \text{ is selected;} \\ 0, & \text{otherwise;} \end{cases}$$

$$z_i = \begin{cases} 1, & \text{if hole } i \text{ is not covered by any of the selected tools;} \\ 0, & \text{otherwise;} \end{cases}$$

$$y_k = \begin{cases} 1, & \text{if part } k \text{ cannot be completed by the selected tools;} \\ 0, & \text{otherwise.} \end{cases}$$

We remark that a given column of the matrix A may contain more than one '1' entry. This occurs, for example for column j, if tool j can punch more than one hole-type, i.e. if the *nominal diameters* and *tolerances* of several holes are such that tool j can safely punch each such hole within specifications. We refer the reader to Daskin *et al.* (1990) for detailed discussion on this issue. We can now formulate our tool selection problem:

(PART)

$$Z^* = \text{minimize} \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} b_i z_i + \sum_{k=1}^{s} t_k y_k$$

$$\text{subject to}: \sum_{j=1}^{n} a_{ij} x_j + z_i \geq 1, \quad i = 1, 2, \ldots, m; \quad (1)$$

$$\sum_{j=1}^{n} x_j \leq p; \quad (2)$$

$$z_i \leq y_k, \quad i \in S_k, \quad k = 1, 2, \ldots, s; \quad (3)$$

$$x, z, y \in \{0, 1\}. \quad (4)$$

Note that in constraint (1), if a hole i cannot be punched by any of the selected tools, then $z_i = 1$ and thus a penalty of b_i occurs in the objective. Constraint (2) is a tool magazine capacity constraint, where p is the upper bound on the number of selected tools (e.g., $p = 200$ in the application refered to earlier). Constraint (3) says that a part is not completed if any one of the holes in the part is not punched by one of the selected tools. (PART) may be thought of as a nested covering problem in the sense that hole covering is nested within the part covering problem.

In addition to the cost of selecting tools, i.e., the cost of using the selected tools in the production period, the objective also includes two types of penalty costs for work that cannot be accomplished by the selected tools in the current production period. The first type of penalty costs (the t_k's) are the fixed charges for processing (off-line) incomplete parts. The second type of penalty costs (the b_i's) are the variable costs of the unpunched holes. For a given hole i, the b_i value would depend upon the number of times hole i needed to be punched during the production period. Thus, if $z_i = 1$, $b_i z_i$ measures the total penalty cost over the production period for not punching *any* of the type i holes in the parts.

Problem (PART) is a mixed integer linear program (MILP) that can be shown to be NP-hard. Certainly general-purpose MILP algorithms can solve such problems exactly, but large instances of (PART) would take a considerable amount of time to solve via these algorithms. Also, a company may solve (PART) several times, each time changing some elements of the part–hole relationships. This would happen if the company was examining the *redesign* of some of its products. Each design scenario would give rise to a different instance of (PART) and it would be necessary to solve each instance to evaluate scenario quality.

As an alternative to exact (direct) solutions to (PART), we now focus on a dual-based approach (via Lagrangian relaxation) to the problem. Such methods have found considerable success in solving large mixed integer programs.

3. The dual-based approach

As is emphasized by Fisher (1981), keys to the success of the Lagrangian relaxation method include (a) tightness of dual bound (the dual objective function value in relating to primal solution), (b) ease of solving dual subproblems, and (c) ability to recover good primal solutions from dual solutions. In what follows, we show that our approach satisfies (b) and (c) above. The quality of our duality bounds ((a) above) will be demonstrated via our computational experience, reported in Section 4.

We now develop our Lagrangian relaxation approach, and provide a result (Theorem 1) regarding the multipliers that we exploit in our algorithm. Relaxing (3) in (PART) with Lagrangian multipliers $\lambda_{ik} \geq 0$ where $i \in S_k$, the Lagrangian problem is:

$$Z_D(\lambda) = \text{minimize} \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} \Big(b_i + \sum_{k \in T_i} \lambda_{ik}\Big) z_i + \sum_{k=1}^{s} \Big(t_k - \sum_{i \in S_k} \lambda_{ik}\Big) y_k$$

$$\text{subject to}: (1),\ (2) \text{ and } (4).$$

We note that $Z_D(\lambda) = F(\lambda) + G(\lambda)$, where

$$\left.\begin{aligned} F(\lambda) = \text{minimize} \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} \Big(b_i + \sum_{k \in T_i} \lambda_{ik}\Big) z_i \\ \text{subject to}: (1),\ (2) \text{ and } x,\ z \in \{0, 1\}. \end{aligned}\right\} \quad (5)$$

and

$$\left.\begin{aligned} G(\lambda) = \text{minimize} \sum_{k=1}^{s} \Big(t_k - \sum_{i \in S_k} \lambda_{ik}\Big) y_k \\ \text{subject to}: y \in \{0, 1\}. \end{aligned}\right\} \quad (6)$$

The dual problem corresponding to $Z_D(\lambda)$ is:

(PART-D)

$$Z_D = \max_{\lambda \geq 0} Z_D(\lambda).$$

We now have

Theorem 1. *For problem (PART-D) there exists a set of optimal multipliers* $\{\lambda^*_{ik}\}$ *satisfying*

$$\sum_{i \in S_k} \lambda^*_{ik} = t_k, \quad k = 1, \ldots, s.$$

Proof: First, we note that with $t_k > 0\ \forall k$, and from (3) and the fact that $z_i \in \{0, 1\}$, we can allow y_k to be unrestricted in sign. Letting CH denote the convex hull (expressed as a set of linear constraints) of (1), (2) and $x, z \in \{0, 1\}$, problem (PART) is equivalent to the following linear program:

(PART*)

$$\text{minimize} \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} b_i z_i + \sum_{k=1}^{s} t_k y_k$$

$$\left.\begin{aligned} &\text{subject to}: x, z \in CH; \\ &\quad z_i \le y_k,\ i \in S_k,\ k = 1,2,\dots,s; \\ &\quad y_k \text{ unrestricted},\ k = 1,2,\dots,s. \end{aligned}\right\} \quad (7)$$

We remark that it may be difficult to construct the linear constraints for CH, but in principle we can always do this. The equivalence of (PART*) and (PART) follows from the fact that the objective function is linear.

Letting $\{\bar{\lambda}_{ik}\}$ be the dual variables associated with the constraints (7) of (PART*), we know that with y_k unrestricted in sign, the optimal dual variables will satisfy

$$\sum_{i=1}^{m} \bar{\lambda}^*_{ik} = t_k. \quad (8)$$

Based upon Geoffrion (1974), the optimal dual variables $\{\lambda^*_{ik}\}$ for (PART-D) are identical to $\{\bar{\lambda}^*_{ik}\}$ and so from (8) the Theorem follows. ■

Theorem 1 suggests that Lagrangian dual problem (PART-D) is equivalent to the following problem:

(PART-D′)

$$\text{maximize } F(\lambda) + G(\lambda)$$

$$\text{subject to}: \sum_{i \in S_k} \lambda_{ik} = t_k,\ k = 1,\dots,s; \quad (9)$$

$$\lambda \ge 0. \quad (10)$$

From (9) in problem (PART-D′) we see that for $i \in S_k$ the optimal Lagrangian multiplier λ_{ik} can be interpreted as the proportion of the penalty cost of incomplete part k allocated to the unpunched hole i. The procedure of adjusting multipliers can be viewed as a procedure to find an appropriate estimate of such proportions for the unpunched holes.

Theorem 1 motivates a Lagrangian multiplier projection procedure. With the multipliers projected to the hyperplanes (9), the second term of the objective, in problem (PART-D′), satisfies $G(\lambda) = 0$, and thus it is unnecessary to solve (6). Problem (5), which defines $F(\lambda)$, is in general an NP-hard problem. However, owing to the special structure of the problem, it can be solved in polynomial time when all holes in question are round holes. Daskin *et al.* (1990) have shown that with reasonable assumptions on tools and holes, the equivalent of problem $F(\lambda)$, can be solved very efficiently via dynamic programming. Thus, feature (b) (ease of solving subproblem) is satisfied by our approach. We now address feature (c) (ease of recovering primal solutions).

From any feasible solution λ of (PART-D′), we can easily recover a feasible solution for the primal problem (PART). Suppose x, z solve $F(\lambda)$. Thus x, z satisfy (1), (2) and (4). Note that since λ satisfies (9), all y_i can be set to either 0 or 1 without affecting the objective function value of (PART-D′). We set

$$y_k = \begin{cases} 0, & \text{if } z_i = 0\ \forall\, i \in S_k; \\ 1, & \text{otherwise.} \end{cases} \quad (11)$$

Then x, z, and y satisfy (1), (2), (3) and (4), and thus represent a feasible solution to problem (PART).

We now give our Lagrangian relaxation procedure to solve problem (PART). In this procedure, we initialize the Lagrangian multipliers to satisfy (9), and use a projected subgradient method to update the multipliers. We outline the Lagrangian relaxation procedure as follows.

Algorithm:

0. (Initialization) Set $N = 0$, LB $= 0$ and UB $= M$, where M is a large number. Set $\lambda^N_{ik} = t_k/n_k$ for every $i \in S_k$, $k = 1,\dots,s$.
1. Solve $F(\lambda^N)$. Then recover (via (11)) a feasible solution $\{x^N, z^N, y^N\}$ and the corresponding primal objective function value OFV(N) for the primal problem (PART). If $F(\lambda^N) >$ LB, then set LB $= F(\lambda^N)$. If OFV(N) $<$ UB, then set UB $=$ OFV(N).
2. If (UB − LB)/LB $< \epsilon$ (a specified stopping criterion); or if $N > T$ (a specified maximum iteration number), stop.
3. Set $N = N + 1$. Update the multipliers

   $$\lambda^N_{ik} = \max\{0,\ \lambda^{N-1}_{ik} + \theta_N(z^{N-1}_i - y^{N-1}_k)\},$$
   $$i \in S_k, \quad k = 1,\dots,s.$$

   Then set $\lambda^N_{ik} = P(\lambda^N_{ik})$, where P is a projection operator that projects Lagrangian multipliers onto the hyperplanes (9). Go to Step 1.

In Step 3 we use a projected subgradient method (Held *et al.*, 1974) to update the Lagrangian multipliers, where $(z^{N-1}_i - y^{N-1}_k)$ is the subgradient at λ^{N-1} and θ_N is a step size. In our computational implementation of the algorithm, we set

$$\theta_N = \frac{t^*(\text{UB} - F(\lambda^N))}{\sum_{k=1}^{s} \sum_{i \in S_k} (z^{N-1}_i - y^{N-1}_k)^2},$$

where t^* is a factor that is initialized at 2 and is reduced by one half if LB or UB are not updated in Step 1 within five iterations. Note that given a pair of hole–part indices (i, k), $i \in S_k$, if $z^{N-1}_i = 1$, then $y^{N-1}_k = 1$. Thus, the denominator of θ_N is equal to the number of hole–tool pairs (i, k), $i \in S_k$, with $y^{N-1}_k = 1$ and $z^{N-1}_i = 0$.

To conclude this section, we give a result that gives rise to an economic interpretation of any duality gap that may exist in an instance of (PART). In addition, we give a simple example problem that demonstrates that the duality gap can be arbitrarily large for some instances.

Let λ^+ be a set of multipliers satisfying (9) and (10). Denote x^+, z^+ and y^+ as recovered primal feasible variables, i.e. x^+ and z^+ are found from $F(\lambda^+)$, and y^+ is found from (11). Let OFV^+ be the primal objective function value obtained by x^+, z^+ and y^+. Finally, let Z^* be the optimal objective function value of (PART) and let λ^* be an optimal solution to (PART-D′). A bound on the duality gap is given by the following theorem:

Theorem 2.

$$Z^* - F(\lambda^*) \le OFV^+ - F(\lambda^+) = \sum_{y_k^+ = 1} \sum_{\substack{z_i^+ = 0 \\ i \in S_k}} \lambda_{ik}^+.$$

Proof: Note that

$$OFV^+ - F(\lambda^+) = \sum_{k=1}^{s} t_k y_k^+ - \sum_{i=1}^{m} \Big(\sum_{k \in T_i} \lambda_{ik}^+ \Big) z_i^+$$
$$= \sum_{y_k^+ = 1} t_k - \sum_{z_i^+ = 1} \sum_{k \in T_i} \lambda_{ik}^+.$$

Also, if $z_i^+ = 1$ and $i \in S_k$, then $y_k^+ = 1$. We then have

$$OFV^+ - F(\lambda^+) = \sum_{y_k^+ = 1} \Big(t_k - \sum_{\substack{z_i^+ = 1 \\ i \in S_k}} \lambda_{ik}^+ \Big)$$
$$= \sum_{y_k^+ = 1} \sum_{\substack{z_i^+ = 0 \\ i \in S_k}} \lambda_{ik}^+ \quad \text{(from (9))}.$$

$F(\lambda^*) \ge F(\lambda^+)$ and $Z^* \le OFV^+$ complete the proof. ■

Recall that a part k is not completed ($y_k^+ = 1$) because one or more of its holes cannot be punched by the selected tools. However, some holes in part k may be punched by the selected tools. Whenever there is an incomplete part ($y_k^+ = 1$), but there is a hole i in part k ($i \in S_k$) that is punched ($z_i^+ = 0$), then if $\lambda_{ik}^+ > 0$, this multiplier contributes a positive amount to the gap bound $OFV^+ - F(\lambda^+)$.

Based on the above observations, it is easy to construct a simple example that can have an arbitrarily large duality gap. Suppose there are two tools, two holes and one part. The cost of selecting any tool is one unit. The penalty cost of any unpunched hole is t units (where $t > 1$). The cost of not completing the part is $2t$ units. Suppose tool i can punch only hole i, $i = 1, 2$; the part contains both holes, and the tool magazine capacity is 1. The optimal solution to the primal problem (PART) is: choose one tool, leave one hole unpunched, so that the part is not completed. The primal objective function value is $3t + 1$. The optimal solution to the Lagrangian dual problem is obtained at $\lambda_1 = \lambda_2 = t$, because of the symmetric cost of holes. The optimal dual objective function value is $2t + 1$. Thus the duality gap is t. Note that as t gets larger, the duality gap increases accordingly.

The example above shows that the Lagrangian heuristics proposed in the last section do not provide optimal solutions to *all* instances of our model. However, such a specially constructed example is not realistic in real-world applications. Our computational experiences based on real data from industry (hereafter referred to as DATA SET) appear to be very encouraging.

4. Computational studies

Before showing our computational results, we first describe DATA SET. We summarize the information regarding tools, holes and parts from the DATA SET as follows.

(A) Tools: there are 498 tools, of which 117 are standard tools and 381 are custom tools. Each tool can punch round holes with tolerances of +0.0015 inch and −0.0000 inch.

(B) Holes (we consider only round holes): there are a total of 481 round holes. Each hole i is specified by its nominal diameter and tolerances.

(C) Parts: we assume that a part contains *only* round holes. There are a total of 569 parts. Each part k has an associated production rate p_k over the production period and a set (S_k) of holes it contains. For every hole i in part k, i.e. $i \in S_k$, DATA SET also gives h_{ik}, which is the number of times hole i must be punched in each unit of part k. Thus the total number of type i holes that must be punched over the production period is $\sum_{k \in T_i} h_{ik} p_k$, where T_i is the index set of parts containing hole i.

In our preliminary analysis of DATA SET, we found that

(1) There were 32 holes that cannot be punched by any tool specified in DATA SET. Thus, all parts containing any one of these 32 holes cannot be completed. Also, there were 36 tools each of which cannot punch any of the specified holes. In our computations, the input files of the test problems excluded the above tools, holes and parts. After elimination, the total number of tools, holes and parts were reduced to 462, 449 and 455 respectively.

(2) In the remaining data, a tool can cover as many as 19 holes (on average, each tool covers 7.94 holes). A hole can appear in as many as 44 parts (3.43 parts on average). A part can contain as many as 47 holes (3.26 holes on average).

The fact that DATA SET included a few holes that could not be punched (with tools provided by a vendor) and tools that could not punch any hole, was a consequence of the existing hole design specifications of the firm, e.g., some tolerances were too tight. Recall that the hole specifications were developed *before* consideration of using a laser punch press to complete the parts.

In addition to DATA SET, we also create a subset

Table 1. Computational results from DATA SET

(1) Problem number	(2) Cost structure	(3) p	(4) Iteration at which GAP < 2%	(5) CPU sec when GAP < 2%	(6) GAP(%) at 200th iteration	(7) Iteration at which the best primal solution found
1	L L H	50	55	792.1	0.6	192
2	L H H	50	4	58.9	0.04	5
3	H L L	50	37	545.9	0.2	200
4	L L L	100	16	429.8	0.1	42
5	L H L	100	8	209.9	0.09	41
6	H H L	100	4	101.9	0.03	5
7	H L L	150	5	163.5	0.04	21
8	L H H	150	14	487.3	0.5	78
9	L H H	150	21	737.5	0.1	79
10	L H L	200	7	293.3	0.07	15
11	L L H	200	27	1100.9	0.4	171
12	L H H	200	3	152.2	0.0[a]	3

[a] Optimal solution found at iteration 3.

(SUB SET) of DATA SET by randomly selecting some records of tools, holes and parts from DATA SET. SUB SET has 71 tools, 94 holes and 243 parts.

We tested the dual-based approach developed in the last section based on test problems generated from DATA SET and SUB SET. The test problems from the same data set (DATA SET or SUB SET) differ only with respect to the tool magazine capacity p, the cost of tooling (c_j's), and the penalty costs of holes (b_i's) and parts (t_k's). The c_j's, b_i's and t_k's were randomly drawn from different specified ranges with uniform distributions. We evaluated the Lagrangian subproblem $F(\lambda)$ by using the dynamic programming algorithm of Daskin *et al.*

At this point it may be appropriate to comment on the fact that, although we have used actual industrial data for the parts, holes, and tool diameters, we have used simulated cost data for tooling costs as well as for penalty costs. It is clear that the actual user of industrial machinery who is purchasing tooling from the manufacturer can obtain accurate data on tooling costs from the manufacturer.

The cost of tooling increases with increasing punch diameter, and the cost of custom tooling, according to some industrial contacts, can run at up to 10 times as much as standard tooling of similar size. We found, however, that the manufacturer of the laser punch press installed in the example presented in this paper was not willing to provide extensive tooling cost data to academics who had no intention of actually buying any of it.

Obtaining accurate penalty cost data, even with full access to industrial data, is obviously subject to some degree of error. Furthermore, in many cases, appropriate penalty costs will depend upon managerial decisions. If, for example, not having the tooling required to punch a particular part means that the part is outsourced, then an appropriate penalty cost might be the difference between the cost of outsourcing the part and the in-house (less overhead) manufactured cost. On the other hand, if another in-house process is used to produce the part, an appropriate penalty cost might be the difference in cost between the two processes. It is clear that obtaining penalty cost data is more involved than the process of obtaining data on tooling costs. However, in an industrial environment reasonable estimates can be made, albeit with some effort. The company with which we worked was able to develop what its management felt were reasonable cost estimates (based upon using other in-house processes), but did not wish to disclose them.

Because of the difficulties associated with obtaining actual industrial cost data for a research paper, we have chosen to use simulated cost data for the purpose of illustrating the algorithm. We do not believe that using actual cost data would have influenced the algorithm's performance substantially.

We have tested 24 problems, 12 from the DATA SET and 12 from SUB SET, with various choices of p and different ranges of tool, hole and part costs. We obtained very good primal and dual solutions in every instance with a relative duality gap of less than 2% within 60 iterations in each test problem. By relative duality gap (GAP) we mean the ratio of the difference between the current (best) upper bound (UB) and the current (best) lower bound (LB) divided by LB, i.e., GAP $=$ (UB $-$ LB)/LB. We summarize our computational results in Tables 1 and 2, where Table 1 (2) corresponds to problems derived from DATA SET (SUB SET).

Each table contains seven columns. The first column corresponds to the problem number. The second column, labeled cost structure, refers to relative values of unit costs in the corresponding problem. Recall that cost

Table 2. Computational results from SUB SET

(1) Problem number	*(2)* Cost structure	*(3)* p	*(4)* Iteration at which GAP < 2%	*(5)* CPU sec when GAP < 2%	*(6)* GAP(%) at 200th iteration	*(7)* Iteration at which the best primal solution found
1	L H L	10	4	0.29	0.001	1
2	H L L	10	3	0.25	0.0[a]	1
3	H L H	10	7	0.47	0.3	1
4	H L L	20	2	0.24	0.001	1
5	H L H	20	7	0.68	0.002	2
6	L H H	20	7	0.66	0.0[b]	3
7	H L L	50	4	0.56	0.002	5
8	H H L	50	3	0.45	0.0[c]	25
9	L H H	50	4	0.61	0.0[d]	4
10	H L H	70	7	0.95	0.002	4
11	H H L	70	2	0.30	0.0[e]	1
12	H L H	70	4	0.56	0.0[f]	4

[a] Optimal solution found at iteration 130. [b] Optimal solution found at iteration 150. [c] Optimal solution found at iteration 53. [d] Optimal solution found at iteration 121. [e] Optimal solution found at iteration 74. [f] Optimal solution found at iteration 98.

values are generated from a uniform distribution. The symbol L (H) refers to a value selected from the range [50–200] ([200–1200]). In column (2), the first of the three symbols (L or H) refers to the range for selection of tool costs (the c_j's), the second symbol gives the range for hole penalty costs (the b_i's), and the third symbol is the range for part penalty costs (the t_k's). Regarding the tool costs, it should be noted that when generating the c_j value for a custom tool (custom tools were identified in DATA SET), we multiplied the randomly generated number by a factor of 10. We did this because we have been told that custom tools can cost up to 10 times as much as standard tools. Column (3) corresponds to the tool rack capacity p. Note that in each problem set, we solved three problems for fixed values of p. In each case, the problems differed owing to relative cost ranges (column (2)).

All problems were run on a VAX 6700-430 computer. The stopping criteria were set to be GAP $< \epsilon = 9 \times 10^{-6}$ or $T = 200$ iterations, whichever occurred first. To test the effectiveness of finding near-optimal solutions, we recorded the iteration number (column (4)) when GAP first fell below 2%. Also, the CPU time for this first occurrence (GAP < 2%) appears in column (5). The CPU time does not include problem input and output time.

To demonstrate that the algorithm is capable of finding an almost optimal solution for the test problems, we include in column (6) the GAP after 200 iterations. Entries with footnotes in column (6) are the problems with either the optimal solution found, or GAP fell below ϵ before 200 iterations. We also report in column (7) the iteration at which the best primal solution was found.

5. The repair kit problem: a special case

In this section we describe a problem referred to in the literature as the repair kit problem (Brumelle and Granot, 1993; Graves, 1982; Mamer and Smith, 1982; Smith *et al.*, 1980). We show that an important version of the repair kit problem is a special instance of problem (PART) and thus may be solved with the Lagrangian-based heuristics developed in this paper.

Consider a situation in which equipment at various field locations is repaired and maintained. Field repair crews carry a standard kit of tools and parts to the equipment sites and perform a wide range of diagnosis and repair jobs. The kit can be restocked between jobs, but its specified contents are fixed. A job requires various parts and tools. A 'broken' or incomplete job occurs whenever one or more of the job's required parts (tools) are not included in the field repair kit. Incomplete jobs may be costly because of production interruption and the requirement for the repair crew to make extra trips for parts. On the other hand, there is an inventory-carrying cost for every part stocked in the repair kit. A repair kit problem is to determine the kit of parts so as to optimize a given objective (to minimize costs, for example) subject to some production and technical constraints.

The repair kit problem was originally proposed by Smith *et al.* (1980) and subsequently studied by Graves (1982), Mamer and Smith (1982), and Brumelle and Granot (1993). Graves (1982) considers the problem of minimizing part and tool inventory-carrying costs subject to a constraint on job completion rate. The models studied in Smith *et al.* (1980), and Mamer and Smith (1982) minimize the sum of inventory-carrying costs plus the expected cost of incomplete jobs. Brumelle and

Granot (1993) consider models in which the objective function consists of the carrying costs of parts and the penalty costs of incomplete jobs as well as other parameters, for instance, weights of the above two types of costs.

The basic repair kit problem is described as follows. Suppose we have m jobs indexed by k, and n required parts indexed by i. Let B_k be the set of indices of parts required by job k, $k = 1, 2, \ldots, m$. A stocking policy for the repair kit, i.e., a set of parts to be included in the kit, is denoted by M, which is a subset of $\{1, 2, \cdots, n\}$. Note that a job k can be done if and only if $B_k \subseteq M$. Let H_i be the annual inventory-carrying cost for part i, $i = 1, 2, \ldots, n$; and let L_k be the expected annual cost for the incomplete job k, $k = 1, 2, \ldots, m$. The repair kit problem as proposed in Mamer and Smith (1982) is:

(R)

$$\min_M \sum_{i \in M} H_i + \sum_{\{k | B_k \not\subseteq M\}} L_k.$$

Problem (R) determines the optimal kit of parts to minimize the sum of the annual inventory-carrying costs of stocked parts and the expected costs of the incomplete jobs. It is solved via a network flow approach (i.e., maximum flow algorithm) in $O((m + n + A)(m + n + 2)^2)$ time, where $A = \sum_{k=1}^{m} |B_k|$.

Graves (1982) suggests modifying his model to include a restriction on the capacity of a repair kit, i.e., the limited number of parts a repair kit can hold. Such a constraint is not unusual because it could be *impossible* for a repair crew to carry all the spare parts and tools that may be needed. The modified model is a two-dimensional knapsack problem, which is difficult to solve. In what follows we will formulate the repair kit problem with the capacity constraint as a linear integer programming model. We will show that the formulation is a special instance of problem (PART).

Define

$$x_i = \begin{cases} 1, & \text{if part } i \text{ is included in the repair kit;} \\ 0, & \text{otherwise;} \end{cases}$$

$$y_k = \begin{cases} 1, & \text{if job } k \text{ is not completed;} \\ 0, & \text{otherwise.} \end{cases}$$

We have the following repair kit problem:

(RC)

$$\text{minimize} \sum_{i=1}^{n} H_i x_i + \sum_{k=1}^{m} L_k y_k \tag{12}$$

$$\text{subject to}: x_i + y_k \geq 1, \quad i \in B_k, \quad k = 1, 2, \ldots, m; \tag{13}$$

$$\sum_{i=1}^{m} x_i \leq p; \tag{14}$$

$$x, \ y \in \{0, 1\}. \tag{15}$$

In (RC), constraint (13) says that if a part i required by a job k is not stocked in the repair kit, then job k is not completed. Constraint (14) is the repair kit capacity constraint, allowing no more than p parts to be stocked in a repair kit. It is easy to see that the problem without constraint (14), i.e. the problem given by (12), (13) and (15), is equivalent to formulation (R).

Letting $z_i = 1 - x_i$, problem (RC) can be rewritten as:

(RC′)

$$\text{minimize} \sum_{i=1}^{n} H_i x_i + \sum_{k=1}^{m} L_k y_k \tag{16}$$

$$\text{subject to}: x_i + z_i = 1, \quad i = 1, 2, \ldots, n; \tag{17}$$

$$z_i \leq y_k, \quad i \in B_k, \quad k = 1, 2, \ldots, m; \tag{18}$$

$$\sum_{i=1}^{m} x_i \leq p; \tag{19}$$

$$x, \ z, \ y \in \{0, 1\}. \tag{20}$$

It is clear that (RC′) is a special instance of (PART) with zero objective function coefficients for all z_i variables and a covering matrix $A = I$, where I is an $n \times n$ identity matrix.

Relaxing (18) with non-negative multipliers $\{\lambda_{ik}\}$, and replacing z_i with $1 - x_i$, we have the following Lagrangian dual problem to (RC′):

(RC′-D)

$$\max_{\lambda \geq 0} (F(\lambda) + G(\lambda)),$$

where

$$F(\lambda) = \text{minimize} \sum_{i=1}^{n} (H_i - \sum_{k \in T_i} \lambda_{ik}) x_i + \sum_{i=1}^{n} \sum_{k \in T_i} \lambda_{ik}$$

$$\text{subject to}: (19) \text{ and } x \in \{0, 1\},$$

and

$$G(\lambda) = \text{minimize} \sum_{k=1}^{m} (L_k - \sum_{i \in B_k} \lambda_{ik}) y_k$$

$$\text{subject to}: y \in \{0, 1\}.$$

Note that Theorem 1 is true for problem (RC′-D). Thus, (RC′-D) is equivalent to

(RC′-D′)

$$\text{minimize } F^*(\lambda) \tag{21}$$

$$\text{subject to}: \sum_{i \in B_k} \lambda_{ik} = L_k, \quad k = 1, 2, \ldots, m; \tag{22}$$

$$\lambda \geq 0, \tag{23}$$

where

$$F^*(\lambda) = \text{minimize} \sum_{i=1}^{n}(H_i - \sum_{k \in T_i} \lambda_{ik})x_i$$
$$\text{subject to : (19) and } x \in \{0, 1\}.$$

The Lagrangian subproblem in this case is a trivial problem, which can be solved as follows: Let

$$\delta_i = H_i - \sum_{k \in T_i} \lambda_{ik}, \quad i = 1, 2, \ldots, n.$$

If there are more than p indices i such that $\delta_i \leq 0$, then set $x_i^* = 1$ for the p indices i with smallest δ_i values and set $x_i^* = 0$ for all other i; otherwise, set $x_i^* = 1 \ \forall\, i$ with $\delta_i \leq 0$ and set $x_i^* = 0$ for all other i. A primal feasible solution is easily recovered by setting $y_k^* = 0$ if $x_i^* = 1 \quad \forall\, i \in B_k$; $y_k^* = 1$ otherwise.

Note that since $F^*(\lambda)$ has the integrality property (Fisher, 1981; Geoffrion, 1974), the lower bound given by the optimal value of (RC′-D) is the same with the optimal value of the LP relaxation of (RC). We propose the above Lagrangian relaxation approach because the structural properties of the Lagrangian multipliers will enable us to design a dual-ascent procedure to obtain good Lagrangian dual solutions quickly. We are now investigating dual-ascent procedures for solving this problem.

6. Conclusions

This paper has presented a nested covering model for the problem of selecting tooling that minimizes the cost of parts produced on a laser punch press and has developed a Lagrangian relaxation approach to solve the model. Computational results on a data set obtained from a Chicago-area firm indicate that the approach quickly computes optimal or near-optimal solutions to relatively large problems. Additionally, we have shown that a variant of the repair kit problem incorporating an upper bound on the number of parts (tools) to be included in the repair kit can be solved as a special case of the nested covering model developed in this paper.

Acknowledgement

We wish to thank two referees and the Associate Editor for helpful suggestions on a earlier version of this paper. Their insightful comments have led to an improved presentation.

References

Brumelle, S. and Granot, D. (1993) The repair kit problem revisited. *Operations Research*, **41**, 994–1006.

Daskin, M., Jones, P. and Lowe, T. (1990) Rationalizing tool selection in a flexible manufacturing system for sheet-metal products. *Operations Research*, **38**, 1104–1115.

Fisher, M. (1981) The Lagrangian relaxation method for solving integer programming problems. *Management Science*, **27**, 1–18.

Geoffrion, A. M. (1974) Lagrangian relaxation for integer programming. *Mathematical Programming Study*, **2**, 82–114.

Graves, S. C. (1982) A multiple-item inventory model with a job completion criterion. *Management Science*, **28**, 1334–1337.

Held, M., Wolfe, P. and Crowder, H. P. (1974) Validation of subgradient optimization. *Mathematical Programming*, **6**, 62–88.

Mamer, J. W. and Smith, S. A. (1982) Optimizing field repair kits based on job completion rate. *Management Science*, **28**, 1328–1333.

Smith, S. A., Chambers, J. C. and Shlifer, E. (1980) Optimal inventories based on job completion rate for repairs requiring multiple items. *Management Science*, **26**, 849–852.

Biographies

Vernon Ning Hsu is an assistant professor at the School of Business Administration, George Mason University. He received his Ph.D. in Operations Management from the University of Iowa in 1993. He also has an M.S. in Computer Science and an M.A. in Economics both from the University of Iowa. He had been a Visiting Lecturer at the University of Iowa and taught MBA courses in Operations Management and Operations Research. His current research interests are Manufacturing/Operations Management, Combinatorial Optimization, and Location Models.

Mark S. Daskin is a Professor of Civil Engineering and Industrial Engineering at Northwestern University. He also holds a joint appointment in the Transportation Center at Northwestern. He taught at the University of Texas at Austin for one and a half years before joining the faculty at Northwestern in January of 1980. He received his Ph.D. in 1978 from the Massachusetts Institute of Technology, a Certificate of Post-Graduate Study in Engineering from Cambridge University, England, in 1975, and a B.S.C.E. from M.I.T. in 1974. He has received a number of awards including a Presidential Young Investigator Award from the National Science Foundation and a Fulbright Research Award which enabled him to spend the 1989-90 academic year in Israel. He is on the editorial boards of *Transportation Science*, *Location Science* and *The International Journal of Logistics Management*. His primary areas of research are logistics, location modelling and theory, transportation planning, and manufacturing and production planning.

Philip C. Jones is George Daly Professor of Manufacturing Productivity at the College of Business, University of Iowa. Previously, he held a joint appointment in the McCormick School of Engineering and the Kellogg Graduate School of Management where he served as founding director of Northwestern University's Master of Management in Manufacturing (MMM) program. Before entering academia, he was a senior engineer at EG&G Idaho, Inc. and an assistant economist at Argonne National Laboratory. He holds a B.I.E. degree from Georgia Tech, an M.S. in Operations Research from Stanford, and an M.A. in Economics as well as a Ph.D. in Industrial Engineering from the University of California, Berkeley. He is an area editor for *Engineering Economist* and conducts research in manufacturing and logistics.

Timothy J. Lowe is the C. Maxwell Stanley Professor of Production Management at the College of Business, University of Iowa. At the University of Iowa, he also serves as Director of the Manufacturing Productivity Center. He received his Ph.D. from Northwestern University in 1973. His previous academic appointments were at the Industrial and Systems Engineering Department, University of Florida, and the Krannert School of Management, Purdue University. His research interests are in applied operations research, manufacturing, and operations management. He serves as an Associate Editor for *Operations Research* and *Location Science*. He was formerly a Departmental Editor for *IIE Transactions*.