OMEGA The Int. Jl of Mgmt Sci. Vol. 9, No. 4, pp. 439 to 445, 1981
0305-0483/81/040439-07$02.00/0

Two Generalizations of the Traveling Salesman Problem

INTRODUCTION

THERE ARE numerous generalizations of the traveling salesman problem (TSP). For example, Golden *et al.* [4] explore some generalizations which include vehicle capacity restrictions, Lokin [7] considers extensions in which precedence relationships are imposed on some of the points, and Kao [5] studies a traveling salesman problem with stochastic travel times. In this note, we introduce two interesting combinatorial problems which each include the TSP as a special case. We define the problems, describe heuristic solution strategies which are straightforward and very intuitive, and illustrate the procedures via examples. Finally, we mention some computational results.

THE TRAVELING PURCHASER PROBLEM

The traveling purchaser problem is a difficult combinatorial problem of wide applicability. Recently, Ramesh [8] presented a lexicographic search procedure for solving the problem exactly. In this note we take a different approach. We describe a heuristic solution strategy and provide an example which illustrates the procedure.

Given a domicile denoted by s, a set $I = \{1, 2, \ldots, m\}$ of markets, and a set $K = \{1, 2, \ldots, n\}$ of items, the traveling purchaser problem is to generate a cycle through a subset of the m markets and the domicile and purchase each of n specific items at one of these markets in such a way that the total of travel and purchase costs is minimized.

It is assumed that each item is available in at least one market, that the traveler may pass through a market any number of times without purchasing an item there, that a traveler may purchase as many items as there are available at each market, and that no items are available at the domicile. Also, we are given the matrices D and C where

$$d(i, k) = \text{the cost of item } k \text{ at market } i, \text{ and}$$

$$c(i, j) = \text{the cost of travel from } i \text{ to } j.$$

If item k is not available in market i, we define $d(i, k) = M$ where $M \gg \max\{\max\{d(i, k)\}, \max\{c(i, j)\}\}$. In the case where $m = n$ and each market carries only one item, the traveling purchaser problem reduces to a TSP.

A savings algorithm for the traveling purchaser problem is described below:

Procedure TPP.

Step 1. Solve the all-pairs shortest-path problem to obtain the shortest distance matrix $\bar{C} = [\bar{c}(i, j)]$.

Step 2. Find the market i^* which sells more items than any other market at the cheapest price. Resolve ties by choosing

$$i^* \text{ to minimize } \sum_{k=1}^{n} d(i, k).$$

Form initial cycle s–i^*–s. Call this α.

Step 3. Compute $f(\alpha, l) = \min_{i \in \alpha} \{d(i, l)\}$ for all l, and

$g(\alpha, p, l) = \max\{f(\alpha, l) - d(p, l), 0\}$ for all l and $p \notin \alpha$.

Step 4. Find the market $p^* \notin \alpha$ and adjacent markets i^*, $j^* \in \alpha$ such that

$$S(i, j, p) = \bar{c}(i, j) - \bar{c}(i, p) - \bar{c}(p, j) + \sum_{l=1}^{n} g(\alpha, p, l)$$

is maximized. Suppose

$$S(i^*, j^*, p^*) = \max\{S(i, j, p)\}.$$

Step 5. If $S(i^*, j^*, p^*) > 0$, insert p^* between i^* and j^*. Update α, $f(\alpha, l)$ for all l, and $g(\alpha, p, l)$ for all l and $p \notin \alpha$. Go to Step 4. If $S(i^*, j^*, p^*) \leqslant 0$, terminate.

The algorithm is straightforward to perform; it is also easy to see why it should work well. In Step 1 the shortest distance matrix is computed. An initial cycle is chosen in Step 2. In Step 3, $f(\alpha, l)$ is the cost of item l given the current cycle α and $g(\alpha, p, l)$ is the decrease in the cost of item l if market p is next inserted into the cycle. Savings or total decreases in travel and purchase costs when market p is inserted between markets i and j are determined in Step 4. If there exists a positive savings, the largest one defines the next insertion. Then, the cycle is updated and we search to see if another market can be added to the cycle.

After Step 1, the most demanding steps in the procedure involve the computation of $g(\alpha, p, l)$ and $S(i, j, p)$. Any algorithm for finding the shortest distance matrix requires on the order of m^3 operations (see [2, 6] for details). In the worst case, where each market is inserted into the cycle, the computation of $g(\alpha, p, l)$ requires a total of about $n\{(m-1) + (m-2) + \ldots + 1\}$ additions and comparisons, whereas the calculation of $S(i, j, p)$ requires a total of about $n\{2(m-1) + 3(m-2) + \ldots + m(1)\}$ operations. Since this last expression is equivalent to

$$n\left\{\frac{m(m+1)(m+2)}{6} - m\right\},$$

our implementation of this savings algorithm is $O(nm^3)$ in time complexity (it is quite possible that the order of complexity can be reduced).

We briefly illustrate the algorithm via an example. The 4 market—8 item problem displayed in Table 1 is taken from Ramesh [8].

In performing the steps of the algorithm, the cycles and savings obtained are

$$\begin{aligned}
& & \alpha_1 &= s - 3 - s, \\
S(s, 3, 4) &= M + 48 & \text{and } \alpha_2 &= s - 4 - 3 - s, \\
S(s, 4, 1) &= 104 & \text{and } \alpha_3 &= s - 1 - 4 - 3 - s, \\
S(1, 4, 2) &= 13 & \text{and } \alpha_4 &= s - 1 - 2 - 4 - 3 - s.
\end{aligned}$$

The final cycle has travel cost = 108, purchase cost = 135, and total cost = 243. Furthermore, it is the

TABLE 1

	c(i, j)						d(i, k)								
	s	1	2	3	4		1	2	3	4	5	6	7	8
s	—	32	68	47	51	1	M	M	17	26	85	M	32	M
1	18	—	32	71	16	2	42	82	63	94	60	M	12	55
2	39	82	—	38	19	3	60	10	57	8	M	95	96	9
3	56	8	63	—	28	4	M	M	M	80	21	16	M	77
4	55	22	6	9	—									

optimal solution. Although, in this example, the final cycle includes all of the markets, this will not be the case in general. For larger problems, we expect the savings procedure to yield near-optimal solutions in a reasonable amount of computing time.

We remark that from a practical point of view the procedure has been coded in FORTRAN and is extremely efficient.

THE TIME-CONSTRAINED TRAVELING SALESMAN PROBLEM

Suppose that each ordered pair of points (i, j) has associated with it a net profit and a travel time. In the time-constrained TSP, the objective is to find a subtour that begins and ends at the origin (point 0), which maximizes profit while requiring less than τ units of time. This is a much more realistic statement of the problem a traveling salesman faces and it has been the focus of research attention by Cloonan [1] and, more recently, Gensch [3].

In particular, let $p = [p(i, j)]$ be the net profit matrix and $t = [t(i, j)]$ be the time matrix and let both of these be $N \times N$. We now present an iterative heuristic solution procedure. At each iteration l, we let P and T denote the total profit and total time of the subtour just generated. Also, ΔP and ΔT are the changes in total profit and time associated with each permissible insertion. Our initial idea was to use the ratio $\Delta P/\Delta T$ as a measure of the attractiveness of an insertion. However, since these changes might be negative, we sought an alternative approach.

Procedure TCTSP.

Step 0. Set l, P, and T to 0. Initialize R_l. Choose α.

Step 1. For each point i, compute

$$\Delta T = t(0, i) + t(i, 0),$$

and

$$\Delta P - R_0 \Delta T = \{p(0, i) + p(i, 0)\} - R_0\{t(0, i) + t(i, 0)\}.$$

Find the point i^* such that $\Delta T \leqslant \tau$ and $\Delta P - R_0\Delta T$ is maximized. If no such point exists, stop. Otherwise, form the subtour 0–i^*–0 and record it. Set P to $P + \Delta P$ and T to $T + \Delta T$.

Step 2. $l \leftarrow l + 1$

$$R_l = \alpha(P/T) + (1 - \alpha)R_{l-1}.$$

Step 3. For each point k not in the present subtour and adjacent points i and j in the subtour, compute

$$\Delta T = t(i, k) + t(k, j) - t(i, j),$$

and

$$\Delta P - R_l\Delta T = \{p(i, k) + p(k, j) - p(i, j)\} - R_l\{t(i, k) + t(k, j) - t(i, j)\}.$$

Find the i, k, j triple i^*, k^*, j^* such that $T + \Delta T \leqslant \tau$ and $\Delta P - R_l\Delta T$ is maximized. If no such triple exists, go to Step 5. Otherwise, insert k^* between i^* and j^* in the subtour and record the subtour. Set P to $P + \Delta P$ and T to $T + \Delta T$.

Step 4. Go to Step 2.

Step 5. From all the subtours recorded, select the one with the largest P.

Several points still need clarification. First, the ratio P/T is the worth of a unit of time in the current subtour. R_l is the best estimate of the worth of a unit of time at iteration l. That is, R_l is an estimate which takes into account all previous ratios P/T but weights the more recent ones more heavily. R_0 should represent an educated guess (possibly based on preliminary analysis) of the profit to time ratio for the optimal subtour. The expression for R_l is the standard exponential smoothing model that is used in a variety of forecasting situations. Finally, in order to promote flexibility and generate a number of heuristic solutions, we can vary α between 0 and 1 and select different start values for R_0. We remark that each application of this insertion procedure requires on the order of N^3 operations in the worst case.

Consider a traditional traveling salesman problem with arc lengths $l(i, j)$, in which we seek a minimum-length Hamiltonian tour over the points. Procedure TCTSP will generate a near-optimal solution to the TSP if we set

$$p(i, j) = \tau - l(i, j), \text{ and}$$
$$t(i, j) = 0 \text{ for all } i \text{ and } j.$$

TABLE 2

	p(i, j)						t(i, j)				
	0	1	2	3	4		0	1	2	3	4
0	—	160	125	35	53	0	—	26	26	15	14
1	−48	—	86	2	58	1	8	—	30	19	16
2	−35	106	—	80	65	2	6	28	—	12	16
3	−50	97	152	—	60	3	8	30	25	—	16
4	−37	148	125	55	—	4	4	24	26	13	—

TABLE 3.

α \ R_0	5	10	15	20	25	30
0.1	A	A	B	B	B	B
0.2	A	A	B	B	B	B
0.3	A	A	A	B	B	B
0.4	A	A	A	B	B	B
0.5	A	A	A	A	B	B
0.6	A	A	A	A	A	B
0.7	A	A	A	A	A	A
0.8	A	A	A	A	A	A
0.9	A	A	A	A	A	A
1.0	A	A	A	A	A	A

where $\tau \geqslant 2\max\{l(i,j)\}$. In this case, the heuristic becomes the well-known cheapest-insertion algorithm. The transformation from $l(i,j)$ to $p(i,j)$ is needed in order to ensure that each point will be included in the subtour obtained.

An example of a time-constrained TSP (TCTSP) with $\tau = 72$ is given in Table 2.

If we let $R_0 = 1$ and $\alpha = 1$, Procedure TCTSP determines the subtour 0–1–4–3–0 with a total profit of 223 and a total time of 63. The intermediate subtours are 0–1–0 and 0–1–4–0 with total profits of 112 and 181 and total travel times of 34 and 46 units. The *optimal* solution is 0–2–3–4–0 with a total profit of 228 and a total travel time of 58.

We have written and tested a FORTRAN program to execute Procedure TCTSP. It is capable of solving problems with as many as 100 points in under five seconds on a Univac 1108. An interesting computational issue is whether varying R_0 and α leads to improvements over the solution obtained when $R_0 = 1$ and $\alpha = 1$. We have, at least tentatively, answered the question in the affirmative. From the previous example, let A and B denote the subtours listed with total profits of 223 and 228, respectively. In Table 3, we exhibit the resulting subtour for each α, R_0 combination.

The pattern displayed in Table 3 is most curious but not atypical of other results that we have obtained. We hope to study it in more detail in future computational research. A number of additional problems that we have experimented with and corresponding heuristic solutions are available upon request from the authors.

REFERENCES

1. CLOONAN J (1966) A heuristic approach to some sales territory problems. In *Proceedings of the Fourth International Conference on Operations Research* (Ed. LITTLE JDC), 81–84. MIT Press, Cambridge, Massachusetts, USA.
2. DREYFUS S (1969) An appraisal of some shortest-path algorithms. *Ops Res.* **17**, 395–412.
3. GENSCH D (1978) An industrial application of the traveling salesman's subtour problem. *AIIE Trans* **10**, 362–370.
4. GOLDEN B, MAGNANTI T & NGUYEN H (1977) Implementing vehicle routing algorithms. *Networks* **7**, 113–148.
5. KAO E (1978) A preference order dynamic program for a stochastic traveling salesman problem. *Ops Res.* **26**, 1033–1045.
6. KELTON W & LAW A (1978) A mean-time comparison of algorithms for the all-pairs shortest-path problem with arbitrary arc lengths. *Networks* **8**, 97–106.
7. LOKIN F (1979) Procedures for traveling salesman problems with additional constraints. *Eur. Jl Opl Res.* **3**, 135–141.
8. RAMESH T (1978) Traveling Purchaser Problem. Unpublished manuscript.

Bruce Golden
Larry Levy
Roy Dahl

(December 1980)

College of Business & Management
University of Maryland at College Park
MD 20742
USA