OMEGA The Int. Jl of Mgmt Sci., Vol. 11, No. 5, pp. 507–512, 1983 0305-0483/83 $3.00 + 0.00

Combinatorial Analysis for Route First–Cluster Second Vehicle Routing

RH MOLE

DG JOHNSON

K WELLS

Loughborough University of Technology, UK

(*Received January 1983; in revised form March 1983*)

Two Route first–cluster second vehicle routing algorithms are contrasted in the first section of the paper. Next, the 'large' number of feasible solutions to a multiple travelling salesman problem is established given that each salesman can visit any number of customers in a stated range. An approximate expression is given for the 'small' fraction of this solution space searched by a route first–cluster second vehicle routing heuristic. Nevertheless, this heuristic is seen to be a very efficient means of searching its solution space.

INTRODUCTION

THERE ARE several common methods for engineering a sub-optimum solution to an intractable optimisation problem. For instance one can first artificially restrict the solution space and then make a complete search for a preferred solution. This heuristic approach may yield an acceptable sub-optimum solution to the original problem, particularly if the restriction on the solution space was well chosen and if the search can be conducted efficiently. This approach can be seen at work in several papers concerned with vehicle routing problems.

The first section of this paper is devoted to a very brief contrast of the methods reported by Beasley [1] and Foster & Ryan [2]; the reader is referred to these sources for further details. However, both these papers are essentially reports of empirical work. This paper reports on an investigation of the reduction in the number of feasible solutions which follows the imposition of artificial constraints of the sort employed in route first–cluster second methods.

TWO ROUTE FIRST–CLUSTER SECOND METHODS

Beasley [1] has recently described the performance of a route first–cluster second vehicle routing method. A particular travelling salesman tour (TST) around the entire set of N customers is first chosen in some manner, which defines in effect an ordered linear list of customers from 'first' to 'last'. This list is partitioned as variously as possible into v 'clusters' of customers, one cluster for each of the v vehicles available. A vehicle is to start from a depot and go on to visit each customer in the cluster before returning to the depot. Route costs can therefore be calculated and infeasible routes, for whatever reason, are assigned a very high cost.

The ordered list of customers from the TST is used to define the order of the N vertices of a directed path graph on $N + 1$ vertices, which also includes an initial vertex. The routing cost for a cluster of customers is associated with the weight of that arc added to the graph from the last customer of the preceding cluster to the last customer of the cluster in question. A sub-optimal vehicle routing solution corresponds to the shortest path from the initial vertex to the last vertex of the Digraph.

Better solutions follow the recognition of the essentially cyclic rather than linear customer ordering supplied by the TST around the N customers. That is to say that any of the N customers can be chosen as the 'first' of the tour. This leads in turn to a larger number of feasible partitions, or clusters, which preserve the cyclic ordering of the TST. The shortest cycle is then found from an augmented cycle graph. The added arcs represent the cost of a vehicle trip as before, and an optimum or near optimum travelling salesman route can be devised for visiting the customers in each cluster.

Some other authors of vehicle routing methods have followed the same broad methodology. For example, Foster & Ryan [2] worked primarily with 'petal' clusters—that is, customer clusters with non-overlapping petal shaped envelopes when drawn on a map. Petal clusters may be quickly obtained by the radial ordering of the customers around the depot prior to partitioning. As before, vehicle routing costs around the customers in each petal cluster can be calculated for later use.

Foster & Ryan [2] employed an integer linear programming approach to the selection of a preferred set of v petal clusters which together result in as many customers as possible receiving a single delivery. They then allowed a slight distortion of petal structure to include a minimum degree of petal overlap, thus locally weakening the primary restriction in the neighbourhood of the solution. This usually improved the results, but a general weakening in petal structure was not favourably reported upon. Slight distortion in structure could possibly be implemented on the Beasley principle. Suppose two adjacent petals overlapped each other. Then the combined costs of both routes could be associated with a single arc spanning the two adjacent arcs associated with the same customer set when distributed between two non-overlapping petals.

Both methods proceed through the following four phases (a) to (d) and a further phase (e) is included by Foster & Ryan.

(a) The cyclic ordering of the customer set.

(b) The rules for determining which clusters to consider.

(c) The methods (heuristic or exact) for the TST within a cluster.

(d) The method for optimally selecting v clusters.

(e) Whether and how to relax the phase (a) choice to allow clusters to overlap (and whether to relax the rules in phase (b)).

The quality of the solution will not depend upon choice in phase (d). But it is impossible to judge the effects of choices in the first three phases, separately or in partial combination, on the basis of the published results. Beasley reports on results of 1,5,10,25 trials with different randomly chosen cyclic orders. Even so, Foster & Ryan's solutions utilising phase (e) are always at least as good, but they report many infeasible solutions without phase (e).

The computational efficiency would seem to be relatively insensitive to phase (a) in both methods and could be expected to be broadly similar in phase (b). Both sets of authors used heuristic methods in phase (c). Beasley's method has a predictable computational burden for phase (d) which is later shown to be essentially $0(N^2)$. Phase (e) relaxations were not attempted possibly because they fit rather uneasily into the shortest path approach. Foster & Ryan provide some limited empirical evidence on the overall computational burden for phases (a) through (d) and also for relaxations in phase (e).

A comparison suggests that the best choices remain open questions for all but phase (d) of cyclic first–route second methods. But the results of these very different implementations of this same general approach are sufficiently good to warrant a general investigation. This paper considers the reduction in the solution space due to the imposition of an arbitrary choice in phase (a), a plausible choice for phase (b) and either ignoring phase (c) or optimally reordering customers.

THE SYMMETRIC MULTIPLE TRAVELLING SALESMAN PROBLEM (MTSP)

Consider the 'symmetric' MTSP where the

cost of travel between two customers is independent of the direction travelled. Suppose that a minimum number, m, of customers is to be visited by each of v salesman and that each of the N customers is to be visited exactly once. The number of feasible solutions, $f_m(N, v)$ is defined for the case where all the salesmen share a common base and are in all respects indistinguishable. The following recursion can be established when $N \geqslant m(v + 1)$:

$$f_m(N, v + 1) = (v + 1)^{-1} \sum_{i=m}^{N-mv} \times \binom{N}{i} f_i(i, 1) f_m(N - i, v). \quad (1)$$

This follows the recognition that a cluster of i customers may be visited by any one of the $(v + 1)$ salesmen, leaving $N - i$ customers to be visited by the remaining v salesmen; the cluster can be chosen in

$$\binom{N}{i}$$

ways and the conditions on the range of i preserve the minimum number m of customers on a salesman's route.

Now for $m \geqslant 2$ we know that the number of distinguishable TSP tours around $i \geqslant m \geqslant 2$ customers for a salesman from a remote base is $f_i(i, 1) = \frac{1}{2}i!$ On substitution into equation (1) we find:

$$f_m(N, v + 1) = N![2(v + 1)]^{-1} \times \sum_{i=m}^{N-mv} [(N - i)!]^{-1} f_m(N - i, v). \quad (2)$$

By induction or otherwise it can be shown that:

$$f_m(N, v) = N![2^v v!]^{-1} \times \binom{N - (m - 1)v - 1}{v - 1} \quad \text{for} \quad m \geqslant 2, \quad (3a)$$

and

$$f_1(N, v) = \sum_{j=0}^{v-1} (v - j)! [v!]^{-1} \binom{N}{j} f_2(N - j, v - j). \quad (3b)$$

The result in equation (3a) can be verified by substitution into the right hand side of equation (2) with the use of the well-known identity:

$$\sum_{k=0}^{p-v} \binom{p - k - 1}{v - 1} = \binom{p}{v}$$

where

$$p = N - (m - 1)v - m.$$

The result (3a) has an instructive interpretation which is revealed by writing:

$$f_m(N, v) = N![2^v v!]^{-1} \phi_m(N, v). \quad (4)$$

The denominator in equation (4) removes the double counting associated with the equivalence of each individual salesman's route when travelled in either direction and the immaterial ordering of the v parts of any partition. The quantity $\phi_m(N, v)$ will be shown below to be the number of partitions of an ordered list of N customers into v parts, each ordered part being a route for a salesman visiting at least m customers. This is seen if we write:

$$\phi_m(N, v) = \phi_0(S, v) = \binom{S + v - 1}{v - 1} \quad \text{for} \quad S = N - mv,$$

so that

$$\phi_m(N, v) = \binom{N - (m - 1)v - 1}{v - 1} \text{ as required.}$$

Suppose that there is an upper limit M on the number of customers that a salesman can visit. The smaller number of feasible solutions, say $f_m^M(N, v)$, to this further restricted MTSP depends upon $\phi_m^M(N, v)$ the smaller number of partitions such that each part contains between m and M customers. That is:

$$f_m^M(N, v) = N![2^v v!]^{-1} \phi_m^M(N, v) = N![2^v v!]^{-1} \phi_0^{M-m}(S, v). \quad (5)$$

The number $\phi_0^{M-m}(S, v)$ is derived in Appendix A and, on substitution:

$$f_m^M(N, v) = N![2^v v!]^{-1} \sum_{i=0}^{\lfloor (N-mv)/(M-m+1) \rfloor} (-1)^i \binom{v}{i} \times \binom{N - (m - 1)v - 1 - i(M - m + 1)}{v - 1} \quad (6)$$

for $m \geqslant 2$ where $\lfloor \quad \rfloor$ is the integer part.

The ratio $R = f_m^M(N, v)/f_N^N(N,1)$ gives a comparative measure of the extent of the MTSP solution space relative to that of the TSP on the same customer set.

When $M = m = N/v$ it can be seen that $R = (2^{v-1} v!)^{-1}$ but it is also clear that there are substantial areas of the N, v, M, m space for which $R > 1$. Indeed, for given N, M, m there is a 'worst' value of v which gives a maximum $f_m^M(N, v)$.

ROUTE FIRST–CLUSTER SECOND METHODS

It is entirely realistic to presume a minimum number m and maximum number M of customers on a vehicle route. These parameters can be estimated by a variety of tests which include the following very simple relationships:

(i) $M = \min\,[N - m(v-1);\ I]$ where I is the largest number of adjacent customers in the initial tour whose combined delivery requirement does not exceed the vehicle capacity.

(ii) $m = \max\,[N - M(v-1);\ J]$ where J is the smallest number of adjacent customers in the initial tour whose combined delivery requirements first exceed a stated proportion q of vehicle capacity.

Thus, if the total of the delivery requirements of all the customers, numerically equal to $(v-1)+r$ vehicle loads say, is rounded up to the next integer multiple v of vehicle capacity then a solution involving v vehicles must be such that no vehicle carries less than the fractional proportion r of its capacity [2]. One might also observe that secondary criteria, such as the desire to avoid very dissimilar vehicle loadings and to avoid very dissimilar numbers of vehicle drops per route, could be satisfied by suitable choices of the parameters q, M, m. These choices will be a major determinant of the number of clusters produced in phase (b) and of course parameter M will be a major influence upon the computational burden in phase (c).

In Beasley's method, for example, there will be $(M-m+1)$ arcs incident to each of the N vertices in the graph and so one would expect $(M-m+1)N$ clusters, each defining a small scale TSP. In practice, then, the number of operations required to identify the clusters is likely to be closer to $0(N)$ than $0(N^2)$ since M and m are not so much dependent upon N as upon vehicle and customer characteristics. However, the computational requirements in phases (c) and (d) are much greater, being $0((M-m+1)M^x)$ and $0((M-m+1)N^2)$ respectively, for handling the small scale TSTs and for determining the best cluster combination where it has been assumed that a heuristic method for solving a TST on M customers runs in a time proportional to M^x.

Both methods implicitly search through $\psi_m^M(N, v)$ distinct solutions to the vehicle routing problem where:

$$\psi_m^M(N, v) = \sum_{i=m}^{M} i\,\phi_m^M(N-i, v-1) \tag{7}$$

where in the absence of phase (c) the vehicle routes correspond to the original cyclic ordering of customers in phase (a).

It is assumed in equation (7) that a specified customer can appear as, say, the first in a cluster of i customers in only one partition of the initial cyclic ordering. If, on the other hand, the optimum TST is selected for each vehicle route then:

$$\psi_m^M(N, v) \geq [(Nv^{-1})!]^v\, 2^{-v} \sum_{i=m}^{M} i\phi_m^M(N-i, v-1). \tag{8}$$

There are Nv^{-1} customers per vehicle on average and $[(Nv^{-1})!]^v$ is a lower bound on the product of the numbers of TSTs searched for each vehicle route, irrespective of the actual distribution of customer numbers to vehicles.

The right hand side of equation (8) is dominated by the first term. For instance when $M = m = Nv^{-1}$ then:

$$\psi \geq [(Nv^{-1})!]^v\, 2^{-v}\, Nv^{-1}.$$

Therefore it might be sensible to choose a heuristic method for optimally solving the TST around a maximum of $(M+1)$ points with a high probability even if the corresponding computational requirement were to exceed that of phase (d).

Obviously the numbers ψ and f can take enormous values even for small problems. Taking the most restricted case $m = M = N/v$ one would still expect N clusters and a search through $\psi = Nv^{-1}2^{-v}((N/v)!)^v$ feasible vehicle routes (from equation (8)) out of the total of $f = N!\,2^{-v}(v!)^{-1}$ (from equation (5)). These numbers are presented in Table 1 for three convenient combinations of N and v associated with problem numbers 1, 2 and 8 described by

TABLE 1

N	v	Number of clusters N	Solutions searched	All solutions f
6	2	6	27	90
12	4	12	243	1,247,400
50	5	50	2.10^{32}	8.10^{60}

Beasley. From these simple examples it is very clear that a cyclic route first–cluster second vehicle routing method employing phases (a) to (d) can be used to search a large number of solutions with commendable efficiency, and yet it considers only a minute proportion of the total number of solutions. Indeed for 'large' N/v the proportion ψ/f is shown to be Nv^{v-N-1} in Appendix B.

CONCLUSIONS

Some general characteristics of cyclic route first–cluster second vehicle routing algorithms have been established under entirely plausible restrictions upon the minimum and the maximum numbers of customers on individual vehicle routes. The computational burden is likely to be better than $0(N^3)$ for a single run and yet a search can be implicitly made over $0(((Nv^{-1})!)^v)$ feasible solutions, which although very large is nonetheless a small fraction of the total possible.

Further work is required to investigate separately and in combination phases:

(a) the initial cyclic ordering of the customer set,

(e) whether and how to relax the cyclic ordering to allow cluster overlap.

Foster & Ryan's comparative success with mildly distorted petal structures suggests that characterisations of some sort are likely to be superior to random choices in phase (a). One might investigate, for example, rules which allow for mild distortion of petal structure remote from the depot and progressively allow more distortion in proximity to the depot. The balance of advantage as between multiple runs with minimal relaxation in phase (e) or fewer runs with greater relaxation is probably best established empirically.

REFERENCES

1. BEASLEY JE (1983) Route first–cluster second methods for vehicle routing. *Omega* **11**(4), 403–408.
2. FOSTER BA & RYAN DM (1976) An integer programming approach to the vehicle scheduling problem. *Opl Res. Q.* **27**, 267–385.

ADDRESS FOR CORRESPONDENCE: *Dr Richard H Mole, Department of Management Studies, Loughborough University of Technology, Loughborough, Leicestershire LE11 3TU, UK.*

APPENDIX A

DERIVATION OF $\phi_0^a(S, v)$

Consider a sample of v random integers in the range $(0, a)$ having probability density $P(x) = (a+1)^{-1}$ for $x = 0, 1, \ldots, a$. The moment generating function of each integer is given by:

$$G(t) = (a+1)^{-1}(1 + e^{it} + e^{2it} + \ldots e^{ait})$$
$$= (1 - e^{(a+1)it})/((a+1)\cdot(1 - e^{it})).$$

The m.g.f. of a sum of v such integers is:

$G'(t) = (G(t))^v$ and on putting $x = e^{it}$

$$(a+1)^v G'(t) = (1 - x^{a+1})^v (1-x)^{-v}$$
$$= 1 + \binom{v}{1}x + \binom{v+1}{2}x^2 + \cdots$$
$$+ \left[\binom{v+a}{a+1} - \binom{v}{1}\right]x^{a+1}$$
$$+ \left[\binom{v+a+1}{a+2} - \binom{v}{1}\binom{v}{1}\right]x^{a+2}$$
$$+ \cdots$$
$$+ \left[\binom{v+2a+1}{2a+2} - \binom{v}{1}\binom{v+a}{a+1} + \binom{v}{2}\right]x^{2a+2} + \cdots.$$

The coefficient of x^S in $G'(t)$ is the probability that the sum of the v integers equals S. We can therefore recognise how many of the $(a+1)^v$ different permutations will total S as the coefficient of x^S in the expansion of $(a+1)^v\, G'(t)$. That is:

$$\phi_0^a(S, v) = \sum_{i=0}^{|S/(a+1)|} (-1)^i \binom{v}{i} \times \binom{v - 1 + S - i(a+1)}{S - i(a+1)}$$

where $|S/(a+1)|$ denotes the integer part of $S/(a+1)$.

APPENDIX B

DERIVATION OF ψ/f

In the case where $m = M = N/v$:

$\psi = Nv^{-1}2^{-v}[(N/v)!]^v$ from equation (8)

$f = 2^{-v}N!\,(v!)^{-1}$ from equation (5).

The proportion R' of routes searched is therefore:

$$R' = \psi/f = (N/v)v!/N!((N/v)!)^v$$

But for 'large' x we know that:

$$\log_e x! \doteqdot x \log_e x - x + 1,$$

so that for 'large' N/v we find that:

$$\begin{aligned}\log(R'/(N(v-1)!)) &= v\log((N/v)!) - \log N! \\ &= v(N/v(\log N - \log v) \\ &\quad - N/v + 1) - N\log N \\ &\quad + N - 1 \\ &= -N\log v + v - 1 \\ &= v\log v - N\log v \\ &\quad - (v\log v - v + 1) \\ &= \log v^{-(N-v)} - \log v! \\ &= \log((v^{N-v}v!)^{-1}).\end{aligned}$$

Thus

$$\begin{aligned}R' &= N(v-1)!/(v^{N-v}v!) \\ &= Nv^{v-N-1}.\end{aligned}$$