ELSEVIER

European Journal of Operational Research 97 (1997) 75–86

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

Theory and Methodology

Heuristic concentration: two stage solution construction

K.E. Rosing [a,*], C.S. ReVelle [b]

[a] Vakgroep Toegepaste Economie, Sectie Economische Geografie, Erasmus Universiteit Rotterdam, P.O.Box 1738, NL-3000 DR Rotterdam, The Netherlands

[b] Department of Geography and Environmental Engineering, Johns Hopkins University, 514 Ames Hall, Baltimore, MD 21218-2686, USA

Received 1 October 1995; accepted 1 March 1996

Abstract

By utilizing information from multiple runs of an interchange heuristic we construct a new solution that is generally better than the best local optimum previously found. This new, two stage, approach to combinatorial optimization is demonstrated in the context of the p-median problem. Two layers of optimization are superimposed. The first layer is a conventional heuristic the second is a heuristic or exact procedure which draws on the concentrated solution set generated by the initial heuristic. The intention is to provide an alternative heuristic procedure which, when dealing with large problems, has a higher probability of producing optimal solutions than existing methods. The procedure is fairly general and appears to be applicable to combinatorial problems in a number of contexts.

Keywords: Combinatorial optimization; Heuristics; Integer programming; Location; Heuristic concentration

1. Introduction

Many heuristics employ an interchange principle together with steepest descent and produce, or can produce, myriad locally optimal solutions to a given problem. Specific instances of the metaheuristics Simulated Annealing, Tabu Search, Genetic Algorithms and Neural Networks all share this characteristic (Pirlot, 1992) as do simpler vertex substitution heuristics (Cornuejols et al., 1977). Each run of any of these heuristics results in not just a functional value but the basis of the solution as given by the assignment characteristics of the nodes of the network or vertices of the graph.

Multiple-random trials of interchange heuristics have been used for a generation now in the context of facility siting as well as other problem areas. Solution methods choose the solution with the minimum functional value from among all the local optima generated and report this as the "best found" solution. In facility siting problems, each solution from an interchange heuristic which differs in functional value from others must also have differences in the set of facilities composing the solution set. It is generally true that solutions whose functional values differ little also are derived from largely identical solution sets.

The present study demonstrates how advantage can be taken of these characteristics to build, in a first stage, a Concentration Set (CS) which has a high probability of containing, within its limited membership, the facilities which comprise the still

* Corresponding author.

PII S0377-2217(96)00100-2

smaller set of the optimal solution to the original problem. In a second stage the best solution of this subproblem, (that restricted to the CS) is found. The selection of the members of the CS is at the heart of the heuristic proposed here; and since the selection is done heuristically, we call the methodology Heuristic Concentration (HC).

The contribution of this work is *not* a faster algorithm for the p-median or, for that matter, for any combinatorial problem. Like the metaheuristics of simulated annealing, tabu search and genetic algorithms, the methodology presented here is designed to escape the traps of local optima which tend to be found by some base heuristic technique. No claim is made for an increased *efficiency*, which we take to mean speed of convergence. A claim is made, however, for increased *effectiveness*; by this we mean that decidedly improved solutions are the norm.

Section 2 defines the location-allocation model, the p-median, used in this empirical demonstration of HC. Section 3 introduces the specific interchange heuristic used here for the concentration step, the method of Teitz and Bart (1968), an example of a Vertex Substitution Heuristic (VSH). Section 4 discusses the 90 network problems, created by varying parameters, for computational experience. In Section 5 some summary and descriptive statistics are presented which illustrate characteristics of the CS and why certain parameters obtain the settings we assign. Section 6 presents two alternative mathematical programmes which are used to operate on the CS as well as a final heuristic step. Finally, Section 7 indicates the level of success, some provisos, and suggests directions which further research can follow. We now turn our attention specifically to location studies on a network to demonstrate and clarify this introduction.

2. The *p*-median problem

The p-median problem is probably the most common and most studied problem in location decision analysis. For this reason we choose it to demonstrate HC. The p-median problem is to find some number (p) medians in a graph or network which, as a set, minimize the weighted distance from all the nodes (n) of the network or vertices of the graph when each node or vertex is assigned (exclusively) to its closest median. For convenience we utilize the location-allocation terminology and call the p medians "facilities" and the n nodes or vertices "demand nodes".

Hakimi (1964,1965) has proven that there exists an optimal solution for any network or graph in which the locations of the facilities coincide with the locations of p selected demand nodes. Balinski (1965) described the plant location problem, a close relative of the p-median, and stated a crucial constraint ((3), below). It was however ReVelle and Swain (1970) who introduced the integer linear programming (ILP) formulation for the p-median problem into the literature. The problem may be stated as:

Minimize

$$Z = \sum_{i=1}^{n} \sum_{j=1}^{n} a_i d_{ij} X_{ij} \tag{1}$$

Subject to:

$$\sum_{j=1}^{n} X_{ij} = 1, \quad \text{for all } i \tag{2}$$

$$X_{jj} - X_{ij} \geq 0, \quad \text{for all } i, j, \quad i \neq j \tag{3}$$

$$\sum_{j=1}^{n} X_{jj} = p \tag{4}$$

$$X_{ij} = 0 \text{ or } 1, \quad \text{for all } i, j \tag{5}$$

Where:

d_{ij} = the distance i to j;
a_i = the weight associated with demand node i;
i = the index of demand nodes;
j = the index of potential facility sites;
X_{ij} = 1 if the ith demand node assigns to the jth facility and 0 otherwise.

In order to solve this as a linear programme constraint (5) must be relaxed to

$$X_{ij} \geq 0, \quad \text{for all } i, j. \tag{6}$$

This is an example of an integer-friendly programme (ReVelle, 1993); experience has shown that in over 95 percent of problems the relaxed version terminates fully integer (Morris, 1978). If, however, a particular instance yields a fractional solution the fractions can be resolved quickly by branch and bound; branching on the fractional X_{jj}'s (Rosing et al., 1979c).

This programme can be solved optimally but the size of its matrix increases as a function of n^2. As a result, large problems outstrip the potential of optimal methods and generally require heuristics. While a number of different heuristic methods have been applied (see Densham and Rushton, 1992b) the most commonly used heuristic is still that of Teitz and Bart (1968).

3. Interchange in location-allocation

The Teitz and Bart (1968) heuristic for the p-median problem is the prototype VSH. As such it is widely available and much studied. For this reason, and in the absence of evidence that any other consistently returns better solutions (see Densham and Rushton, 1992b, Table 2, p. 326) we choose it for this demonstration. Alternative heuristics are faster, such as that of Densham and Rushton (1992b) but we choose for generality of result over batch computer time.

Like any classical interchange heuristic the Teitz and Bart heuristic is started by supplying either a set deliberately chosen or a set of random nodes (a "current solution"). Each potential facility site (node where a facility could be sited) not in the current solution is substituted for the one facility in the current solution which is under consideration. If a substitution makes an improvement, the current solution is updated and testing that same facility continues. When one member of the current solution has been tested against all potential sites (and perhaps substituted one or more times) that one facility is in the best possible position given the positions of all others at the time of its testing. Once all potential facility sites have been tested to become a replacement for each and every facility in the current solution, one iteration is finished. The current solution then has each facility in the best place it could be given the positions of the other facilities at the time it was being tested. Additional iterations may further improve the solution.

When one full iteration is completed without any substitutions, the algorithm terminates with the best solution that can be reached by one-at-a-time exchanges from the initial set of facilities supplied. This condition is termed a stable partitioning pattern (SPP). Attainment of such a pattern satisfies the stopping rule of any one-at-a-time VSH. The term SPP is preferred here to local or suboptimal solution since termination of the algorithm is dependent upon the stopping rule of the heuristic and totally unrelated to the gradient of the objective function. We do, however, differentiate between globally optimal SPPs and nonoptimal SPPs because for the former an additional characteristic obtains; that is, that no change of strategy or stopping rule could improve it — there is no better.

Leaving aside papers which deal with applications, studies of the Teitz and Bart heuristic have concentrated upon speeding up the heuristic (Densham and Rushton, 1992a,b) on extending it to other problems (Church and ReVelle, 1976; Hillsman, 1984; Hodgson et al., forthcoming) or improving the search strategy (Goodchild and Noronha, 1983; Densham and Rushton, 1992b). Other studies have dealt with the number of trials necessary to reach a reliable solution (Rosing and Van Dijk, 1993) and judging the robustness of the heuristic (Rosing et al., 1979a,b; Rosing, forthcoming). These studies concentrate on the algorithm, on the search strategy, or on functional value(s) but all ignore the actual nodes selected to be facilities in the SPPs found. The present study differs by concentrating on the lists of facilities associated with these different SPPs.

4. The example problems

For each problem in this study 200 different random starts were used and the objective value and the solution set (the list of selected facilities at termination) corresponding to each start recorded. A series of ninety problems were solved optimally and heuristically. The optimal solutions were used as reference points for judgement of the 18000 heuristic solutions.

All combinations of $n = 100, 125, 150, \ldots, 300$ (the number of demand nodes) and $p = 5, 10, 15, \ldots, 50$ (the number of facilities) define the instances solved, creating a sort of "crosstable" (see Table 1 as an example of the organization) of parameter variation. Three hundred random coordinate pairs were generated and the Euclidian internode distances calculated. Equal weights ($a_i = 1$) were applied. Further details can be found in Rosing (forthcoming). The work presented in that study showed that the solution quality of the VSH described above degrades as either n or p or both increase, at least in random networks with no spatial structure. It was this finding that directed attention to attempting to find an improved (in the sense of more likely to be optimal) heuristic method, particularly for larger problems. The generality of that empirical study and of this one in the presence of a clear spatial structure is, at this time, unknown; though we hypothesis that the behaviour will be similar.

5. Stage one: finding the CS

In each problem each SPP differs from each other SPP in having at least two nodes chosen as facilities which are not so chosen in the other SPP. In addition each nonoptimal pattern differs from the globally optimal solution in having at least two nodes which are not in the optimal solution. If this condition were not true, the inferior SPP would iterate to the superior. It is also likely, due to the search strategy, that SPPs which are similar in functional value are also similar in much of the membership of their solution sets, i.e. that they have few differences in chosen facilities. Based upon similarity of nodes in nonoptimal solutions we propose to find the CS by a direct comparison and tabulation of nodes occurring in a selection of these solutions. We have tried other, more complex, systems of analysis of the nonoptimal solutions, to a limited extent, but failed, in general, to obtain a better result.

5.1. Stable partitioning patterns

The total number of partitioning patterns is upper bounded by:

$$\binom{n}{p} = \frac{n!}{p!(n-p)!} \tag{7}$$

which is the number of different ways of drawing p items from a population of size n; and thus the number of potential starting positions for the VSH. Some number of these will also be sets of facilities corresponding to SPPs. Since our 200 different starting node lists, for each problem, are randomly drawn they are, in each case, a sample of the large number of partitioning patterns available. It follows that the SPPs actually found are also a sample of the SPPs available. It seems reasonable then to suppose that the number of available SPPs grows as a function of n and p (see also: Rosing, forthcoming) just as the total number of partitioning patterns does (up to $p = n/2$). Certainly the computational experience acquired from this study shows that 1). the number of SPPs found increases and 2). the difference between their functional values decreases as a function of n and p. The number of different SPPs found in the 90 cases correlate with the percent of runs of the heuristic terminating with the optimal solution at $r = -0.75$ ($r^2 = 0.56$) with an F statistic of 110.27, significant at 0.001. This provides a strong indication that the failure to find the optimal or a truly "good" solution in cases involving a large n and p are a function of the increase in the availability of SPPs from which to choose.

Consider, for simplicity, a minimal, two-node, difference between the optimal SPP and the next best SPP. The two suboptimally located facilities constituting the difference must be close together (in a relative sense) since they must distort the assignment of one another's demand nodes causing this pattern to be different from that of the optimal solution's pair (which must also be close together) assignments. If they are not close together other, optimally located, facilities and their respective partitions would be between them and they could not affect one another — which they must do to be suboptimally located.

Each of these partially nonoptimal lists of facilities can be thought of as an information source, each providing information about the structure of a portion the network and the optimal solution. The hope is that they provide information about different portions of the network. The quality of this information is inspected in the following section.

5.2. The differences between "good" facility lists

To facilitate this work each list of 200 heuristic solutions was sorted into ascending order by functional value. This places the optimal, and any equal optima at the top of the list of heuristic solutions. In addition equal nonoptimal SPPs are grouped together and the groups of individual SPPs are listed in increasing order of functional value; those closest, in functional value, to the optimal at the top of the list. The assumption is that the lists of chosen facilities in the nonoptimal SPPs with smallest functional values will be those most similar to the optimal list ie. convey the most information about it.

In Table 1 some statistics about the differences in facility lists, optimal and first- and second-best SPPs are summarized. On the lower line of each cell, labelled "%in sub", is shown the percent of identical nodes in the facility list of the two best, but not optimal, patterns. The best, as judged by functional value, nonoptimal SPP is shown on the left of each cell and the second best, which must differ from each the best and the optimal each by at least two nodes, on the right. For example, in the cell for $n = 150$, $p = 40$ the entry in the lower left, 95.0%, means that 38 facilities of the minimum functional value nonoptimal SPP and the optimal solution are identical, the minimum necessary difference. The entry in the lower right, 92.5%, means that 37 of the members of the facility list of the second best nonoptimal SPP are identical to the facility list of the optimal. The 100.0% on the top line (labelled "% in opt.") of the cell 150/40 indicates that all the facilities in the global optimum are contained in the union of the lists of facilities of the two best SPPs.

En passant, it is interesting to note that the solution set of the second best SPP, judged by functional value, can, and often does, match the solution set of the optimal better than does the solution set of the best nonoptimal. See for example the cell corresponding to $n = 275$, $p = 40$. Here 80.0% (32 nodes) of the nodes in the solution set of the best is identical to the optimal solution set while 87.5% (35 nodes) of the nodes in the solution set of the second best are identical. In the table as a whole the best nonoptimal is best match 37 times, the second best is best match 21 times and the number of nodes of the best and second best matching an optimal is equal 32 times.

Table 1
Percent of identical nodes in optimal solution and best two nonoptimal SPPs

p	n	100		125		150		175		200		225		250		275		300	
50	%in opt	98.0		100.0		90.0		98.0		98.0		96.0		100.0		94.0		96.0	
	%in sub	96.0	96.0	96.0	86.0	80.0	84.0	96.0	96.0	96.0	96.0	94.0	94.0	94.0	94.0	94.0	90.0	92.0	88.0
45	%in opt	100.0		100.0		97.8		93.3		97.8		95.6		97.8		97.8		91.1	
	%in sub	95.6	95.6	97.8	86.7	88.9	86.7	88.9	91.1	95.6	91.1	93.3	93.3	95.6	86.7	93.3	93.3	84.4	84.4
40	%in opt	92.5		92.5		100.0		92.5		100. 0		97.5		100.0		97.5		92.5	
	%in sub	87.5	80.0	87.5	92.5	95.0	92.5	90.0	85.0	92.5	90.0	82.5	90.0	95.0	80.0	80.0	87.5	85.0	87.5
35	%in opt	97.1		97.1		97.1		94.3		94.3		100.0		88.6		94.3		97.1	
	%in sub	88.6	94.3	94.3	91.4	91.4	88.6	91.4	91.4	82.9	88.6	91.4	91.4	88.6	82.9	94.3	82.9	80.0	91.4
30	%in opt	96.7		93.3		93.3		96.7		93.3		93.3		100.0		93.3		100.0	
	%in sub	90.0	93.3	900.	93.3	90.0	86.7	93.3	93.3	93.3	83.3	76.7	90.0	93.3	93.3	90.0	76.7	86.7	93.3
25	%in opt	100.0		96.0		100.0		100.0		100.0		88.0		100.0		96.0		92.0	
	%in sub	92.0	92.0	88.0	88.0	88.0	88.0	92.0	88.0	92.0	92.0	80.0	88.0	92.0	92.0	88.0	88.0	80.0	92.0
20	%in opt	80.0		100.0		85.0		100.0		100.0		85.0		85.0		90.0		100.0	
	%in sub	80.0	75.0	85.0	85.0	80.0	80.0	90.0	85.0	90.0	90.0	80.0	80.0	65.0	85.0	90.0	85.0	90.0	90.0
15	%in opt	86.7		93.3		86.7		80.0		86.7		86.7		86.7		93.3		86.7	
	%in sub	86.7	73.3	80.0	73.3	86.7	80.0	80.0	53.3	86.7	60.0	86.7	60.0	66.7	53.3	86.7	80.0	80.0	80.0
10	%in opt	100.0		90.0		80.0		80.0		80.0		80.0		70.0		60.0		90.0	
	%in sub	80.0	80.0	70.0	60.0	70.0	80.0	80.0	50.0	80.0	70.0	80.0	60.0	70.0	40.0	60.0	40.0	60.0	70.0
5	%in opt	20.0		40.0		40.0		40.0		60.0		0.0		40.0		60.0		0.0	
	%in sub	0.0	20.0	20.0	20.0	20.0	20.0	40.0	40.0	20.0	60.0	0.0	0.0	40.0	40.0	60.0	20.0	0.0	0.0

This raises questions, which go beyond the scope of this paper (see however Section 7, end), concerning judging similarity of solution sets by means of similarity of functional value. This *en passant* comment does not, however, invalidate the general observation that a *group* of the best are more similar to the optimal than a *group* of inferior solutions when quality is judged by functional value.

Above and to the left of the dark line in Table 1, marking off 21 cells from the other 69, certain difficulties were encountered in tabulating. This was because the p/n ratio in this part of the table is so high that there are many equal optimal SPPs and equal nonoptimal SPPs as well. The rule followed is the percent referring to the optimal is for that one equal optimal that best matched the two nonoptimal SPPs selected. For each of the two best nonoptimal SPPs, all nodes in all equal nonoptima, were counted when matching. Below and to the right of the dark line no equal optimal solutions were encountered.

One optimal solution in each of the 90 problems comes from the ILP. In addition other equal optima have been found, in some cases, by the VSH. Since our process of finding solutions, eg. running the heuristic, is not exhaustive but rather a sample; it is possible that another, not found, optimal solution exists which is more similar than that referred to by the percent on the top line of each cell. These percents on the top line are then a lower bound on the similarity of the nonoptimal SPPs we found and the optimal. It is also quite possible that there are other nonoptimal SPPs whose solution sets are more similar to the optimal than those found.

From the evidence presented in Table 1 it should be apparent that the majority of the nodes selected to be facilities in "good" SPPs are common to the optimal solution; furthermore, as noted above, non-common facilities must, in each nonoptimal, affect a relatively small portion of the network. Conditions causing the operation of the heuristic's stopping rule are relatively rare, small scale and restricted in spatial occurrence in any one instance. Since each sub-optimal SPP must differ from each other optimal or nonoptimal SPP by at least two nodes various "good" solutions with differing functional values would appear, in most cases, to tend to "get stuck" in different portions of the network while attaining optimal positioning in most of the network.

Inspection of Table 1 reveals, however, that similarity in solutions deteriorates with decreasing values of p. Remembering that (Section 5.1) low p problems are those most likely to reach optimality with the original heuristic mitigates this difficulty. Even though these low p cases are most probably already optimally solved, the relationship of the optimal SPP and good nonoptimal SPPs in these cases must be examined.

In the $p = 5$ case the "best" nonoptimal must have at least 40% dissimilarity (two nodes) from the optimal; the same is true of the "second best". With low values of p however a nonoptimal placement of two nodes is more likely to result in a largely or completely different set of facilities. With higher values of p the effect of nonoptimal placements is more local; there is more "inertia" in the pattern of the optimal solution. But with a small p sudden, nearly complete, shifts in the full facility choice are the rule rather than the exception.

5.3. The number of nonoptimal solutions which must be inspected

In this section we attempt to answer the question of how many nonoptimal SPPs must be inspected to ensure at least one optimal solution is contained in the CS and the amount of information (number of potential facility nodes) this results in. This is done by crosschecking all known optimal solutions with the nonoptimal SPPs until all nodes in at least one optimal solution set have been found and placed into the CS. Table 2 displays the results.

To construct this table the optimal solution from the ILP is read in and matched with the list of heuristic solutions (which is in ascending order by functional value). Any equal optima from the list of heuristic solutions are also recorded, each separately. The nonoptimal SPPs are now considered one by one and a list of all nodes chosen as facilities in the nonoptimal SPPs considered is made up. The examination of heuristic solutions stops when all nodes chosen as facilities in at least one equal optimal solution have been found in nonoptimal SPPs. The top line (labelled "#non-opts") of each of the 90 cells of Table 2 shows the number of "good" (in terms of functional value) nonoptimal SPPs which must be inspected before matching is complete. This

Table 2
Number of nonoptimal SPPs investigated and numbers of nodes

P	n	100		125		150		175		200		225		250		275		300		central tendency	
50	#non-opts	2		2		6		5		2		5		2		3		4		3.3	3
	#nodes	28	44	41	18	30	42	33	36	44	12	41	20	44	12	43	14	36	27	62.8	1.3
45	#non-opts	2		2		3		4		3		11		4		3		4		3.9	3
	#nodes	32	26	34	23	31	29	32	28	38	16	27	39	35	20	37	18	32	28	58.3	1.3
40	#non-opts	4		4		2		5		2		3		2		3		22		5.2	3
	#nodes	25	31	30	20	32	17	28	28	33	14	30	19	30	20	27	28	21	46	53.3	1.3
35	#non-opts	4		4		3		3		3		2		3		3		4		3.2	3
	#nodes	24	23	28	15	25	20	29	14	25	20	29	12	28	15	28	15	22	26	43.3	1.2
30	#non-opts	3		6		6		4		3		5		2		5		2		4.0	4
	#nodes	20	20	22	17	19	22	23	15	25	12	25	12	26	8	19	23	24	12	38.3	1.3
25	#non-opts	2		4		2		2		2		3		2		3		5		2.8	2
	#nodes	21	8	19	13	19	12	20	10	22	6	18	15	21	8	16	19	10	33	32.2	1.3
20	#non-opts	3		2		5		2		2		5		3		3		2		3.0	3
	#nodes	13	15	14	12	10	25	15	10	16	8	7	25	11	19	12	17	16	8	28.1	1.4
15	#non-opts	8 *		3		5		5		56		4		3		6		3		10.6	4.5
	#nodes	0	29	8	15	6	24	6	22	1	56	7	18	4	23	8	22	4	25	31.1	2.1
10	#non-opts	2		3		3		24 *		3		6		12		64 *		10		5.6	4
	#nodes	6	8	3	16	4	13	0	51	3	17	1	23	1	37	0	80	2	30	22.9	2.3
5	#non-opts	10 *		5 *		11 *		10 *		7 *		11 *		14 *		5		8 *			
	#nodes	0	19	0	17	0	32	0	29	0		24	0	27	0	33	0	18	0	26	
	central tendency	2.6 43.0	2.5	3.2 38.7	3	3.9 42.3	3	3.8 43.8	4	8.4 40.0	3	4.9 40.9	5	3.7 40.2	3	3.6 43.3	3	6.2 44.7	4		

* Not all nodes in the optimal solution were found in nonoptimal solutions. The counts in these cells were *not* included in calculating the marginal statistics.

number, m, is the number of solutions to be inspected in that case to ensure that all nodes in the optimal solution are contained in the CS.

A "*" in the cell indicates this is not possible; i.e. the optimal solution contains at least one node not appearing in any nonoptimal SPP found. The number on the top line of those cells is then the number of nonoptimal SPPs found which have different functional values. Stating this in another way; it is the size of the sample of nonoptimal SPPs found when all equal, in functional value, nonoptimal SPPs are considered to be the same.

The bottom line (labelled "#nodes") of each of the 90 cells shows (on the left) the total number of nodes, selected to be facilities, in *all m* nonoptimal SPPs investigated. The number on the right on the bottom line is the number of nodes which appear in *at least one but less than m* solutions. The size of the CS, formed by the union of the m solutions sets, is the sum of these two numbers.

Considering the body of Table 2, the row for $p = 5$ indicates that, as was to be expected, heuristic concentration does not work for smaller values of p. Every cell has a "*" and the top number is therefore the total number of nonoptimal SPPs which were found in 200 trials. This effect extends itself to $p = 10$ (two cells with "*"s) and $p = 15$ (one cell with a "*"). The good news is that this appears, as one would suspect, to be a direct function of p irrespective of n. For moderate or large values of p the number of nonoptimal SPPs required is generally very small.

Table 2 also has a marginal column and row labelled in each case "central tendency." Since for $p = 5$, 10, and 15 the numbers in the cells with "*"s represent something rather different from the rest, cells with "*"s are excluded in calculations for the marginal cells. The bottom line of each marginal cell gives, on the left, the average size of the CS. Examination of the marginal *row* shows that the size of the CS is unrelated to n. The marginal *column* however shows, again as one would expect, a close relationship between p and the size of the CS. The marginal column also shows (bottom line, right) the average number of nodes per facility required. Although this number begins, for $p = 10$, at nearly 2.5 it rapidly decreases and stabilizes at about 1.3. In other words for a moderate or large value of p approximately 30% more nodes are needed in the CS than the number of facilities required to be in the solution set.

On the top line of each marginal cell the left number is the arithmetic mean of the number of nonoptimal solutions examined ("*" marked cells excluded) and the right number the median value. No median and very few arithmetic means exceed five. This indicates that in the majority of cases examining five nonoptimal solutions will provide a CS containing an optimal set of facilities.

6. Stage two: selection from the CS

The CS constitutes our restricted set of potential facility sites. We chose, in this example, to employ an optimal method to select the best solution set from the CS — an ILP. The mathematical programme to choose the optimal (in terms of this restricted problem) solution from the potential facilities in CS is then:

Minimize

$$Z = \sum_{i=1}^{n} \sum_{j \in CS} a_i d_{ij} X_{ij} \tag{8}$$

Subject to:

$$\sum_{j \in CS} X_{ij} = 1, \quad \text{for all } i \tag{9}$$

$$X_{jj} - X_{ij} \geq 0, \quad \text{for all } i, \text{for all } j \in CS, i \neq j \tag{10}$$

$$\sum_{j \in CS} X_{jj} = p \tag{11}$$

$$X_{jj} = 0, 1, \quad \text{for all } j \in CS \tag{12}$$

Where variables are as defined above and CS = the concentration set.

The parameter m is the number of good solutions, with different functional values, taken in order from the top of the list (ranked in ascending order by functional value) to be inspected to define the CS. Since different solution sets may have identical functional values the number of solutions inspected in a particular problem may actually be greater. For this demonstration the value for m is chosen to be five. This should give the authors an ego-satisfying success rate and yet leave some failures which can be usefully examined. Looking again at Table 2, any problem represented by a cell in which the number on the top line is a five or smaller we can expect to be solved optimally. Equally for any cell with a "*" we can expect failure. Interesting cells will be those with values of m greater than five. Three sorts of results can be expected for instances which do not reach optimality (remember, in this demonstration the optimal is known). 1). The ILP described above, which uses the CS as its base, will terminate with the best solution from the 200 trials of the heuristic with the same or different solution set. 2). The (known) optimal or an equal-optimal solution to the original unrestricted problem will be found. 3). A local optimal will be found which is better than any SPP found while running the heuristic.

The size of the CS, in cases where the number on the top line of a cell of Table 2 is five, is the sum of the two numbers on the lower line of that cell. If the number on the top line is less than five the CS will contain the number of nodes indicated by this sum or slightly more; if it is more than five the CS contains the number of nodes denoted by the sum or slightly less. The ILP tableau will have n × this sum (size of the CS) columns and $(n-1)\times$ size of CS $+ n + 2$ rows. A comparatively small programme.

A second, even smaller, mathematical programme can be constructed. For this programme an additional assumption has to be made; to wit, that the nodes which appear in the solution set of all examined heuristic solutions to be facilities *really are* facilities while other nodes which appear as members of the solution set in only some of the examined heuristics *may be or may not be* facilities. The former would represent those portions of the network which are relatively uncomplicated and where "traps" do not exist but rather the heuristic always iterates directly, in this portion of the network, to the optimal position. The latter would represent difficult portions of the network where nonoptimal pairs, triplets, whatever, are more likely to be chosen.

To implement this, the set CS must be partitioned into two new sets, namely $\mathrm{CS_o}$, *CS open*, for those members of the set CS which appear in all examined solutions (corresponding to the lower left hand number of each cell, Table 2) and $\mathrm{CS_f}$, *CS free*, for those members of CS which appear in at least one but not all solutions (corresponding to the lower right hand number of each cell, Table 2). Since each demand node not in the set CS must assign to the closest chosen facility site constraints and variables are needed only for that one member of $\mathrm{CS_o}$ which is closest to the particular demand node and only for those members of the set $\mathrm{CS_f}$ which are closer than that one member of $\mathrm{CS_o}$. This second programme is then:

Minimize

$$Z = \sum_{i=1}^{n} \sum_{j \in R_i} a_i d_{ij} X_{ij} \tag{13}$$

Subject to:

$$\sum_{j \in R_i} X_{ij} = 1, \quad \text{for all } i \neq j \in \mathrm{CS_o} \tag{14}$$

$$X_{jj} = 1, \quad \text{for all } j \in \mathrm{CS_o} \tag{15}$$

$$X_{jj} - X_{ij} \geq 0, \quad \text{for all } i, \text{for all } j \in R_i, i \neq j \tag{16}$$

$$\sum_{j \in \mathrm{CS}} X_{jj} = p \tag{17}$$

$$X_{jj} = 0, 1, \quad \text{for all } j \in \mathrm{CS_f} \tag{18}$$

Where variables are as defined above and

$$r_i = \left\{ j \mid \min\left(d_{ij,\, j \in \mathrm{CS_o}}\right)\right\}, \quad \text{for all } i \neq j \in \mathrm{CS_o} \tag{19}$$

$$R_i = \{ j \in r_i \} \cup \left\{ j \in \mathrm{CS_f} \mid d_{ij} < d_{ir_i} \right\},$$
$$\text{for all } i \neq j \in \mathrm{CS_o} \tag{20}$$

As Eq. (19) states for each i the set r_i contains the one member of the set $\mathrm{CS_o}$ which is closest to it. The meaning of Eq. (20) is then that for each i the set R_i contains that one member of r_i and any members of the set $\mathrm{CS_f}$ which are closer than r_i. One of these potential facilities must serve i in the best solution. Members of the set $\mathrm{CS_o}$ are not allowed to assign away but must be facilities (constraint (15)). Members of the set CS_f may assign away or may be facilities Eqs. (contraints (14) and (16)). Nodes not in either of these sets must assign to one member of either $\mathrm{CS_o}$ or $\mathrm{CS_f}$ (constraint (14)). The formulations are both very integer friendly (ReVelle, 1993). Eleven of the 90 problems terminated fractionally; but only one required more than seven nodes to resolve (it required ten).

The particular size of the tableau of this second programme will depend upon the number of mem-

Table 3
Tableau size, problem $n = 300$, $p = 50$

Model	Rows	Columns	Iterations	CPU seconds
Fully specified p-median	90002	90000	9997	2726.8
CS ($m = 5$)	19139	18900	907	42.2
CS_o and CS_f ($m = 5$)	798	560	263	0.7

bers of CS_o, CS_f and the geometric arrangement of them and of the demand nodes. Table 3 gives the actual sizes of the tableau and solution statistics for: the fully specified p-median, the programme involving set CS, and the programme with CS_o and CS_f for one instance: $n = 300$, $p = 50$. The small size of the latter programme (CS_o, CS_f) makes it extremely attractive — if it is successful. The CPU times and the number of iterations are for CPLEX, 1989-1994, 3.0 running on a Sun Sparcserver 20 with 96 megabytes of CPU under SunOS 4.1.2_U1. These statistics are indicative of the possibility of solving much much larger problems than can be attempted with the standard p-median ILP formulation (1)–(5).

While the emphasise of this work is effectiveness not efficiency the CPU time is indicated in Table 3. The appropriate question is however not the LP time but "At what total cost are better solutions found?" The answer is that the cost is in the noise. If a (large) number of runs of the base heuristic were to be made in any case and the best of these runs reported as the solution, the cost of applying the HC metaheuristic is the cost of assembling the CS and solving the relatively small (see Table 3) integer linear programme as a relaxed linear programme. The times for assembly of the CS and solution by LP of the reduced problem are minor compared to the time spent running the base heuristic.

In cases where the solution of this programme (8)–(12) or (13)–(18) is better than the best nonoptimal SPP from the heuristic but less than the global optimal of the original ILP the solution set (chosen from the CS) may not be stable in the sense of a VSH. Re-applying an interchange heuristic to this solution may result in further improvement. This is done by submitting the fixed solution set chosen from the CS to the interchange heuristic and allowing it full freedom to move facilities. In all cases where one facility from the CS is nonoptimal optimality will always be reached by this step. In cases where two or more are nonoptimal it may be reached.

Table 4
The results of HC for the ninety problems

$p \backslash n$	100	125	150	175	200	225	250	275	300
p									
50	■■■	■■■	■A■	■■■	■■■	■■■	■■■	■■■	■■■
45	■■■	■■■	■■■	■■■	■■■	S	■★■	■★■	■■■
40	■★■	■■■	■■■	■■■	■★■	■■■	■★■	■★■	B
35	■■■	■■■	■■■	■■■	■★■	■■■	■★■	■■■	■★■
30	■■■	S	■A■	■■■	■■■	■■■	■★■	■■■	■■■
25	■■■	■■■	■■■	■■■	■C■	■■■	■★■	■■■	■■■
20	■■■	■■■	■■■	■■■	■■■	■■■	■■■	■■■	■★■
15	*	■■■	■■■	■■■	S	■■■	■■■	S	■■■
10	■■■	■■■	■■■	*	■■■	S	■T■	*	S
5	*	*	*	*	*	*	*	*	*

■★■ CS_o, CS_f Optimal, no optimals found in 200 runs of heuristic.
■■■ CS_o, CS_f Optimal.
■A■ CS_o, CS_f Alternate optimal found.
■C■ CS Optimal, CS_o, CS_f best SPP.
■T■ CS_o, CS_f and CS 66.7% better, Teitz and Bart to optimal.
B CS_o, CS_f and CS 1.6% better, Teitz and Bart no improvement.
S CS_o, CS_f and CS no improvement, best SPP found.
* Expected failures.

7. Results and conclusions

See Table 4 for the results. All 90 instances were solved using the CS_o, CS_f formulation in order to see if the additional assumption affected the solution. In cases where this formulation terminated with a functional value less than global optimum the instance was then solved again using the first ILP (CS model). As was to be expected no problem which earned a "*" in its cell on Table 2 solved optimally with either formulation.

In all cases corresponding to cells marked with a black bar, combined or not with another symbol, HC found the optimal solution — 78.9% of all 90 cases. As earlier noted HC should not be used for small values of p. If we exclude $p = 5$ the success rate raises to 87.7% of the remaining 81 cases. In two of these cases (marked by a bar and an "A") an alternate optimal was found. In one case (marked by a bar and a "C") the CS_o, CS_f model made no improvement while the CS model found the optimal solution. This is because one node, not in the optimal solution was included in the set CS_o and was therefore fixed "open". Once it was a choice facility (CS model) that facility was substituted and optimality was achieved.

In two cases HC (both models) found a nonoptimal solution better than the best nonoptimal SPP. In one case (marked by a bar and a "T") the functional value of the solution was 66.7% better (measured as percent of the range, best, lowest functional value, SPP to optimal solution) than the functional value of the best nonoptimal SPP. Application of the Teitz and Bart algorithm from that solution as a fixed starting point resulted in the optimal solution. In the other case (marked by a letter "B") the solution was 1.6% better (measured in the same way) and re-application of the heuristic made no further improvement. The HC solution was itself another, until that time unidentified, SPP better than the best achieved by the random runs. In six cells (marked by a letter "S") HC made no improvement, terminating with the best SPP used in constructing the CS.

It is well known that particular combinations of n and p and particular geometries make some problems "hard" and others "less hard" for an interchange heuristic. Once a starting set is determined the mechanistic nature of the heuristic leads to a fixed outcome. Determining a starting set determines the solution set — even though it is not know until execution of the algorithm. From any particular starting point to the corresponding solution set there is a fixed "path" or pattern of interchanges. These paths, from different starting sets, can merge but never diverge. Consider a case where paths merge and move to various suboptimal SPPs. The optimal solution can be thought of as "defended" by suboptimal bulwarks. An interchange heuristic has then a very low probability of penetrating these "defenses" and reaching optimality.

Something like this occurs in 13 of our 90 cases. Two hundred runs of the Teitz and Bar heuristic, in each case, were insufficient to find any optimal solution at all (in 2600 total runs). A failure rate for the Teitz and Bart of 14.4%. Of these 13 one (marked by an "S", cell 225/45) was also a failure for HC. In the other 12 cases however the cells are marked by a bar combined with a "★". In these cases HC (both models) found the optimal solution. A success rate of 92.3% for the 13 cases where Teitz and Bart fails. It is this success in cases of interchange failure which we believe makes HC attractive. HC can never do worse than the best SPP found and it can, and regularly does find the optimal solution.

To summarize:

1. In general, heuristic concentration works well, at least in this problem and probably has potential for others as well, for moderate and large values of p. For small values of p it appears to be inappropriate. A large proportion of the runs of the interchange heuristic are already optimal for these values of p in any case.
2. In our experiment $m = 5$ is too restrictive and a higher value of m would have provided better results. We knew this but wished to have failures to analyze.
3. Considering the minor improvement in results from programme CS the CS_o, CS_f programme is, generally, to be preferred particularly since a larger m value will also push any "extra" nodes from CS_o into CS_f and considering the size of problem whose solution it allows.
4. Heuristic Concentration can provide solutions to problems with better objective functions than an interchange heuristic alone.

Given the extremely compact form of the tableau resulting from HC one might speculate that this method, with programme CS_o, CS_f, could be employed on problems one or perhaps two orders of magnitude (in terms of n) larger.

Several important questions remain open. These questions relate not just to the p-median — it is only the example demonstrated here — but any application of HC; a procedure which should have a general application to a number of areas of combinatorial optimisation.

● One question is: does the method work well with a much more limited number of runs of the heuristic? The heuristic was run 200 times and the best five were inspected. The 200 runs used here are a lot; that number of runs related to the need to calculate probabilities in a piece of research reported elsewhere (Rosing, forthcoming). What would be the effect on the CS, on CS_o, CS_f of a smaller number of random starts?

● Another open question is: the setting and nature of the parameter m. Is m highly dependent upon the structure of the individual problem/data or is a good m carved in stone? The value for m may also be related to the number of runs of the heuristic. Just what is a good value for m? As m goes up the set CS becomes larger and, in more detail CS_0 decreases in size while CS_f increases in size. These changes directly affect the size of the ILP tableau.

● Finally: is the ranking by functional value the best way to chose solutions which approach the optimal. Might a more considered analysis of the solution sets reveal, in some cases at least, significant sets of groupings of nodes leading towards different, nearly equal, solutions? If so could these be partitioned and the process of stage two applied to the different Concentration Sets.

We hope to report on these points (and others) in the near future.

References

Balinski, M. (1965). "Integer Programming: Methods, Uses and Computations." *Management Science* 12, 253–313.

Church, R.L. and ReVelle, C.S. (1976). "Theoretical and Computational Links Between the p-Median, Location Set-covering, and the Maximal Covering Location Problem." *Geographical Analysis* 8, 406–15.

Cornuejols, G., Fisher, M.L. and Nemhauser, G.L. (1977). "Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms." *Management Science* 23, 789–810.

CPLEX (1989–1994). *Using the CPLEX Callable Library* (Incline Village, NV: CPLEX Optimization, Inc.).

Densham, P.J. and Rushton, G. (1992a). "Strategies for Solving Large Location-Allocation Problems by Heuristic Methods." *Environment and Planning, Series A* 24, 289–304.

Densham, P.J. and Rushton, G. (1992b). "A More Efficient Heuristic for Solving Large P-Median Problems." *Papers in Regional Science* 71, 307–329.

Goodchild, M.F., and Noronha, V. (1983). *Location-Allocation for Small Computers.* Monograph No. 8, Iowa City, Iowa: Department of Geography, University of Iowa.

Hakimi, S.L. (1964). "Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph." *Operations Research* 12, 450–459.

Hakimi, S.L. (1965). "Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems." *Operations Research* 13, 462–475.

Hillsman, E.L. (1984). "The p-Median Structure as a Unified Linear Model for Location-Allocation Analysis." *Environment and Planning, Series A* 16, 305–18.

Hodgson, M.J., Rosing, K.E. and Storrier, A.L.G. (forthcoming). "Applying the Flow-Capturing Location-Allocation Model to an Authentic Network: Edmonton, Canada." *European Journal of Operational Research.*

Morris, J.D. (1978). "On the Extent to Which Certain Fixed-Charge Depot Location Problems can be Solved by LP." *Journal of the Operational Research Society* 29, 71–76.

Pirlot, M. (1992). "General Local Search Heuristics in Combinatorial Optimization: a Tutorial." *Belgian Journal of Operations Research, Statistics and Computer Science* 32, 7–67.

ReVelle, C.S. and Swain, R. (1970). "Central Facilities Location." *Geographical Analysis* 2, 30–42.

ReVelle, C.S. (1993). "Facility Siting and Integer-Friendly Programming." *European Journal of Operational Research* 65, 147–158.

Rosing, K.E. (forthcoming). "An Empirical Investigation of the Power of a Vertex Substitution Heuristic." *Environment and Planning, B.*

Rosing, K.E., Hillsman, E.L. and Rosing-Vogelaar, H. (1979a). "A Note Comparing Optimal and Heuristic Solutions to the p-Median Problem." *Geographical Analysis* 11, 86–89.

Rosing, K.E., Hillsman, E.L. and Rosing-Vogelaar, H. (1979b). "The Robustness of two Common Heuristics for the p-Median Problem." *Environment and Planning A*, 11, 373–380.

Rosing, K.E., ReVelle, C.S. and Rosing-Vogelaar, H. (1979c). "The p-Median and its Linear Programming Relaxation: An Approach to Large Problems." *Journal of the Operational Research Society* 30, 815–823.

Rosing, K.E. and Van Dijk, J.J. (1993). "Estimating the Probability of Heuristic Improvement: A Large Scale Application of Extreme Value Theory." *Studies in Locational Analysis* 4, 301–304.

Teitz, M.B. and Bart, P. (1968). "Heuristic Methods for Estimating the Generalized Vertex Median of a Weighted Graph." *Operations Research* 16, 955–961.