Corrected Copy

DISCRETE OPTIMIZING

—STANLEY REITER AND GORDON SHERMAN

2-sided

REPRINTED FROM
JOURNAL OF SIAM
SEPTEMBER 1965—VOL. 13, NO. 3

PURDUE UNIVERSITY

KRANNERT SCHOOL OF INDUSTRIAL ADMINISTRATION REPRINT SERIES

(continued)

78. NATHAN ROSENBERG, "Technological Change in the Machine Tool Industry, 1840-1910," *The Journal of Economic History*, Volume XXIII, No. 4, Dec. 1963.

79. PAUL V. JOHNSON, "Use of Group Projects in College Teaching," *Collegiate News and Views*, December, 1963, South-Western Publishing Company.

80. R. L. BASMANN, "Remarks Concerning the Application of Exact Finite Sample Distribution Functions of GCL Estimators in Econometric Statistical Inference," *The Journal of the American Statistical Association*, Vol. 58, No. 304, December 1963.

81. GEORGE HORWICH, "Effective Reserves, Credit, and Causality in the Banking System of the Thirties," *Banking and Monetary Studies*, Edited by D. Carson.

82. NATHAN ROSENBERG, "Capital Goods, Technology, and Economic Growth," *Oxford Economic Papers*, Vol. 15, No. 3, Nov. 1963.

83. FRANK M. BASS, "A Dynamic Model of Market Share and Sales Behavior," *Proceedings of the Winter Conference of the American Marketing Association*, December 1963, *Toward Scientific Marketing*.

84. J. R. T. HUGHES, "Measuring British Economic Growth," *The Journal of Economic History*, Volume XXIV, No. 1, March 1964.

85. NATHAN ROSENBERG, "Neglected Dimensions in the Analysis of Economic Change," *Oxford Institute of Economics and Statistics Bulletin*, Vol. 26, No. 1, Feb. 1964.

86. JOHN M. DUTTON AND WILLIAM H. STARBUCK, "On Managers and Theories," *Management International*, No. 6, 1963.

87. M. JUNE FLANDERS, "The Economics of Prebisch and Ecla: A Comment," *Economic Development and Cultural Change*, Vol. XII, No. 3, April 1964.

88. JAMES A. PAPKE, "Research and State Tax Reform," *Proceedings of the Fifty-Sixth Annual Conference on Taxation, National Tax Association*, November 1963, Milwaukee, Wisconsin.

89. EDGAR A. PESSEMIER, "Forecasting Brand Performance Through Simulation Experiments," *Journal of Marketing*, Vol. 28, No. 2, April 1964.

90. RENE P. MANES, "The Quality and Pricing of Crude Oil: The American Experience," *Journal of Industrial Economics*, Volume XII, No. 2, March 1964.

91. PATRIC H. HENDERSHOTT AND JAMES L. MURPHY, "The Monetary Cycle and the Business Cycle: The Flow of Funds Re-examined," *The National Banking Review*, Volume I, No. 4, June 1964.

92. M. JUNE FLANDERS, "Prebisch on Protectionism: An Evaluation," *The Economic Journal*, Volume LXXIV, No. 294, June 1964.

93. CLIFF LLOYD, "The Real-Balance Effect and the Slutsky Equation," *The Journal of Political Economy*, Volume LXXII, No. 3, June 1964.

94. VERNON L. SMITH, "Effect of Market Organization on Competitive Equilibrium," *The Quarterly Journal of Economics*, Volume LXXVIII, May 1964.

95. JAM[illegible], "Looking Forward in State and Local Revenues: A Prospective Analysis," *The [illegible] Seventh National Conference on School Finance*, 1964.

96. DONA[illegible] AND VERNON L. SMITH, "Nature, The Experimental Laboratory, and the C[illegible] of Hypotheses," *Behavioral Science*, Volume 9, No. 3, July 1964.

97. RENE P. MANES, "The Grant-in-Aid System for Interstate Highway Construction: An Accounting or Economic Problem?" *The Accounting Review*, Volume XXXIX, No. 3, July 1964.

98. JONATHAN R. T. HUGHES, "Eight Tycoons: The Entrepreneur and American History," *Explorations in Entrepreneurial History/Second Series*, Volume 1, No. 3, Spring/Summer 1964.

99. JAMES A. PAPKE, "Indiana Tax Policy: Revision, Reform, Reconstruction," *National Tax Journal*, Volume XVII, No. 2, June 1964.

100. J. M. DUTTON AND R. E. WALTON, "Operational Research and the Behavioural Sciences," *Operational Research Quarterly*, Volume 15, 1964.

(Continued on inside back cover)

J. Soc. Indust. Appl. Math.
Vol. 13, No. 3, September, 1965
Printed in U.S.A.

DISCRETE OPTIMIZING*

STANLEY REITER† AND GORDON SHERMAN‡

Introduction. In this paper we present a new approach to optimization problems, and report the results of applying it to a variety of traveling salesman problems. We give a formulation of the general optimizing problem in terms of a few very simple ideas which perhaps surprisingly make it quite easy, indeed almost routine, to construct very powerful procedures for handling large scale problems. While in this paper we report only some of the results obtained for traveling salesman problems, this approach has already been applied with good success to several other types of problems, including the trim problem, the problem of choosing an optimal partition of a finite set, and others. These methods have produced results in these problems clearly superior to any yet obtained by other techniques.

We shall present our approach in terms of the problem of maximizing (minimizing) a function defined on a finite set. The traveling salesman problem, the job shop scheduling problem, and the plant location problem are well-known examples of such problems. Many others arise in connection with scheduling industrial or other operations or with grouping or arranging entities of some sort. There is a large class of such problems for which no useful techniques of solution are known. In some cases for which algorithms are known, the computations required exceed the memory capacity of available computing machinery and/or require unreasonable amounts of computer time. In other cases the only "theory" available is the proposition that complete enumeration of alternatives would, if carried out, produce solutions. Often such enumeration is not feasible. While this situation is, of course, widely understood and appreciated, it nevertheless remains true that problems are formulated in such terms that complete enumeration or other algorithms are formally correct methods of solution even though inapplicable. It is obvious that this curious situation persists

* Received by the editors December 10, 1963, and in final revised form October 7, 1964. The authors are indebted to Herman Chernoff, Stanford University, for valuable comments on an earlier version of this paper.

Because of space limitations, the tables of numerical results shown in this paper are incomplete. The complete tables may be obtained from the authors.

† Department of Economics, Purdue University, Lafayette, Indiana. The work of the first-named author was done in part during 1961–1962 when he was a John Simon Guggenheim Memorial Fellow, and the work has also been supported by the Office of Naval Research.

‡ University Computing Center, University of Tennessee, Knoxville, Tennessee. The research of the second-named author has been supported in part by the National Aeronautics and Space Administration under Contract NAS8-11189.

because costs of computing are not included in the formal structure of the problem. Once computing costs (here understood in a sense general enough to cover such things as capacity restrictions and costs of delay) are considered explicitly, it is natural to try to distinguish those objectives in problem-solving whose value is worth the cost of achieving them.

It is possible to distinguish three types of approaches to optimizing problems. The first is to try to construct an algorithm which is sure to produce an exact solution, or a controlled approximation to such a solution. Such algorithms or deterministic procedures may be thought of as intelligently directed search. As we have noted already, such algorithms or approximations are not always feasible for large problems. The second is roughly equivalent to complete enumeration in the sense that the number of observations required to provide a satisfactory probability of observing optimal points may well be as large as the (finite) number of points in the set being searched. More recently there are so-called heuristic methods, which can be considered to constitute intelligently directed search procedures but for which no assurance can be given about the results obtained. They can be thought of as giving approximations to solutions but with uncontrolled error.

The basic idea of our approach is to combine intelligent search with random search; to use such intelligently directed search as we can afford, given the costs of computing, and to rely on random search to avoid being trapped in nonoptimal local optima. Our method is designed to allow us to use whatever information we may have about the structure of the problem to be solved, even if that information is insufficient to permit construction of a feasible algorithm. Our approach permits us to achieve an approximation to an optimal solution whose error is under control. In the extreme case of minimum information about the structure of the problem we can control the probability of error; additional information can be used to give more precise control of the error. Moreover the cost of computing is used in relation to the value of results to decide how much computing to do.

Briefly our procedure is this. Starting with any point we use a suitably chosen local search technique to move to better points until eventually no further improvement (using that particular search method) is possible. We are then at a locally optimal point relative to our procedure for making improvements. Generally there will be many such locally optimal points. We sample starting points according to some suitable probability distribution, a procedure which induces a probability distribution over the set of local optima. In effect we sample sequentially from the distribution of local optima according to a plan which stops when the expected return from further sampling is not sufficient to pay the cost of further sampling.

Our approach has been tried out on a number of problems with good success. In this paper we describe in detail the application of our method to a number of traveling salesman problems together with the numerical results obtained. Briefly, we tried all the specific traveling salesman problems we could find. These include 17 problems ranging in size from 9 to 57 cities. In every case, we found solutions at least as good as the best solutions known before. In cases where the optimal solution is known, it was also found by our methods; in several other cases, solutions were found which were better than the best known solutions. The detailed results, including computing times, are described in §2 of this paper.

1. Theory: maximizing a function on a finite set. In this section we present a way of structuring optimization problems which permits us to encompass in one formulation all of the logical decisions and computational procedures used in a wide variety of methods of attacking such problems. These include exact algorithms, Monte Carlo methods, heuristic procedures, and "learning" methods. A few simple ideas are sufficient to describe this structure.

We first specify the class of problems to be considered. Let f be an integer valued function defined on a finite set $\mathcal{P} = \{p\}$ with upper bound U and lower bound L.[1] The range of f will be denoted by r $(= U - L + 1)$. It will be convenient to denote the integers between U and L by $Z = \{Z_i : i = 1, 2, \cdots, r\}$, where

$$Z_i = L + i - 1.$$

We assume in what follows that f is optimized on $\mathcal{P}$ by maximization. The procedures for minimizing f on $\mathcal{P}$ are similar except for the obvious reversals of the inequalities.

Definition. A *successor structure* or *tree structure* on $\mathcal{P}$ is a pair $(\mathfrak{N}, s)$ where

(i) $\mathfrak{N}$ is a collection of subsets $\{\mathfrak{N}(p) : p \in \mathcal{P}\}$ such that
 a. for each $p \in \mathcal{P}$ there is precisely one $\mathfrak{N}(p) \in \mathfrak{N}$,
 b. for each $p \in \mathcal{P}$, $p \in \mathfrak{N}(p)$;

(ii) s is a function defined on $\mathcal{P}$ satisfying
 a. $s(p) \in \mathfrak{N}(p)$,
 b. $f(s(p)) \geqq f(p)$,
 c. $f(s(p)) = f(p) \Rightarrow s(p) = p$,
 d. $s(p) = p \Rightarrow f(p') \leqq f(p)$ for all $p' \in \mathfrak{N}(p)$.

The collection $\mathfrak{N}$ is called a ***neighborhood structure on*** $\mathcal{P}$, $\mathfrak{N}(p)$ being the ***neighborhood of*** p; the function s is called a *successor function on* $\mathcal{P}$.

A successor structure amounts to a monotone nondecreasing (local) search procedure. That is, the successor function associates to each point a point in its neighborhood at which the function is at least as large. It is possible to define the successor function without reference to neighborhoods. However, the neighborhood structure permits us to describe the particular local search pattern in a convenient way.

[1] We take f to be integer valued only for convenience of exposition. Any discrete set of values could be handled with no real change in what follows.

We give some examples of successor structures.

Examples. (1) Let $\mathcal{P}$ be the set of all permutations of the integers 1, 2, $\cdots$, n, let f be a real-valued function defined on $\mathcal{P}$, and let $p = (i_1, i_2, \cdots, i_n)$,

$$\mathfrak{N}(p) = \{(i_2, i_1, i_3, \cdots, i_n), (i_2, i_3, i_1, \cdots, i_n), \cdots, (i_2, i_3, \cdots, i_n, i_1), (i_1, i_2, \cdots, i_n)\}.$$

Thus, $\mathfrak{N}(p)$ is ordered. Let p' be the first element of $\mathfrak{N}(p)$ such that $f(p') > f(p)$. Then

$$s(p) = p'.$$

If there is no such p',

$$s(p) = p,$$

i.e., the neighborhood of a permutation p consists of the permutations obtained by shifting the first element i_1 of the original permutation to the second, third, etc., positions successively. This defines an ordered set of permutations. The successor of p is the first permutation so arrived at where the function is larger. If there is no such permutation, p is locally maximal and is its own successor.

(2) For the same set $\mathcal{P}$ as in (1), with $p = (i_1, i_2, \cdots, i_n)$,

$$\mathfrak{N}(p) = \{(i_3, i_1, i_2, i_4, \cdots, i_n), (i_3, i_4, i_1, i_2, i_5, \cdots, i_n), \cdots, (i_1, i_2, \cdots, i_n)\},$$

with s defined as in (1).

Here we have the same procedure as in Example (1) except that we move the first two elements i_1, i_2 of the permutation p as a pair.

(3) Let $\mathcal{P}$ be the set of all combinations of the integers 1, 2, $\cdots$, n, i.e., the set of all partitions of 1, $\cdots$, n into two subsets. For

$$p = (\{i_1, i_2, \cdots, i_r\}, \{i_{r+1}, \cdots, i_n\}),$$

let $\mathfrak{N}(p)$ consist of all combinations obtainable from p by transfer of a single point from one subset to the other. The function s is defined by the condition

$$f(s(p)) - f(p) = \max f(p') - f(p), \qquad p' \in \mathfrak{N}(p).$$

In case of tie, choose $s(p)$ so it contains 1; if there is still a tie choose $s(p)$ so it contains 2, and so on until the tie is broken.

In this example, the neighborhood of a given combination consists of all those combinations obtainable from it by shift of a single element from one subset to the other. The successor function moves in the direction of steepest ascent.

(4) Let $\mathcal{P} = \left\{0, \frac{1}{n}, \frac{2}{n}, \cdots \frac{n-1}{n}, 1\right\}$ and for $p = \frac{r}{n}$, where $r = 1, 2, \cdots, n-1$, let

$$\mathfrak{N}(p) = \left\{\frac{r-1}{n}, \frac{r}{n}, \frac{r+n}{n}\right\},$$

$$\mathfrak{N}(0) = \left\{0, \frac{1}{n}\right\}, \qquad \mathfrak{N}(1) = \left\{\frac{n-1}{n}, 1\right\},$$

with $s(p)$ defined as in (3) with ties being broken in any deterministic manner.

Here we have a function defined as $n + 1$ evenly spaced points in the closed unit interval. The neighborhood of a given point r/n consists of itself together with the two adjacent points. The successor function gives the direction of steepest ascent.

Definition. (a) The point $p \in \mathcal{P}$ is a *locally optimal point* (with respect to $(\mathfrak{N}, s)$) and $f(p)$ a *local optimum* if $s(p) = p$.

(b) Let $s^n(p)$ denote the nth successor of p, defined by $s^n(p) = s(s^{n-1}(p))$, $n = 2, 3, \cdots$.

(c) A point $p^* \in \mathcal{P}$ such that $f(p^*) \geq f(p)$ for all $p \in \mathcal{P}$ is called an *optimal point.*

The foregoing definitions describe what often happens when a "near optimizing" algorithm is used on the type of problem to which we are addressing ourselves. The computer starts with some feasible solution and then searches for an improvement from among "neighboring feasible solutions." Usually, "neighboring feasible solutions" are determined by their computational convenience. This process continues until the algorithm (or successor function) can no longer find a better solution in the neighborhood of the best solution already found.

The following remarks are easily proved.

(1) For any $(\mathfrak{N}, s)$, there are locally maximal points in $\mathcal{P}$. In particular an optimal point p^* is a locally maximal point with respect to every $(\mathfrak{N}, s)$.

(2) For any successor structure and each $p \in \mathcal{P}$ and for some n, depending on p, $s^n(p) = s^{n+1}(p)$.

Definition. For each $p \in \mathcal{P}$, $\hat{p}$ is called the *ultimate successor* of p, if for some n and all integer-valued $t \geq 0$, $s^{n+t}(p) = \hat{p}$.

It follows from (2) and the definitions of successor function and local maximum that:

(3) Every point in $\mathcal{P}$ has a unique ultimate successor.

(4) The set of locally maximal points is determined by $\mathfrak{N}$, independently of s, i.e., if $(\mathfrak{N}, s)$ and $(\mathfrak{N}', s')$ are successor structures with Q and Q' the corresponding sets of locally maximal points, then $\mathfrak{N} = \mathfrak{N}'$ implies $Q = Q'$.[2]

One further assumption will be made here which simplifies the presentation but is no restriction upon the theoretical structure. We will assume that different locally optimal points give different local optimums, i.e., no two locally optimal points give the same value of the function f.

Definition. For each $p \in \mathcal{P}$ let

$$T(p) = \{q \in \mathcal{P}: \hat{q} = \hat{p}\}.$$

The set $T(p)$ then is the set of points in $\mathcal{P}$ which have the same ultimate successor as p.

Further, let

$$T_i = \{p \in \mathcal{P}: f(\hat{p}) = Z_i\}.$$

The set T_i is the set of points in $\mathcal{P}$ whose ultimate successor has an f function value equal to Z_i.

We may interpret the successor structure in terms of a graph. The points of $\mathcal{P}$ are nodes; an arc is directed from p to p' if p' is the successor of p. The resulting graph consists of a number of disjoint (inverted) trees, each of which has a locally optimal point in $\mathcal{P}$ at its top or apex. Choosing a point p to start with, the successor structure leads from p to a unique locally optimal point, namely, the one at the top of the (unique) tree containing p.

1.1. Probability structure. If a point $p \in \mathcal{P}$ is selected according to some probability law F, then the successor structure, together with the properties of f and $\mathcal{P}$, determine a probability distribution $\xi = (\xi_1, \xi_2, \xi_3, \cdots, \xi_r)$ as follows:

Definition. (i) Let F be a probability density on $\mathcal{P}$. Then

$$\xi_i = \sum_{p \in T_i} F(p), \qquad i = 1, 2, 3, \cdots, r.$$

[2] Because, as we remarked above, the successor function can be defined without reference to a neighborhood structure, it is also true that the set of locally maximal points is determined by s.

That is, ξ_i is the probability that the probability law F selects a point in $\mathcal{P}$ whose ultimate successor has a functional value equal to Z_i.

(ii) In particular, if F^0 is the uniform distribution on $\mathcal{P}$,

$$\xi_i = \frac{|T_i|}{|P|}, \qquad i = 1, 2, 3, \cdots, r.^3$$

(5) The optimal point p^* is that point such that if

$$f(p^*) = Z_i, \qquad \xi_i > 0,$$

then for all

$$j > i, \qquad \xi_j = 0.$$

The probability vector ξ summarizes the effect of imposing the successor structure $(\mathfrak{N}, s)$ on the given problem $(\mathcal{P}, f)$.

It is interesting to note that an exact algorithm has a successor structure which determines a unique local maximum which is the optimum, and therefore is described by a probability vector ξ which gives probability 1 to the optimum and 0 elsewhere.

The Monte Carlo method is described by a successor structure in which the neighborhood of each point consists of that point alone. Hence, each point at which the function is defined is locally maximal. The probability ξ_i, in the distribution ξ induced by uniform sampling, is the relative frequency of points at which the function f takes the value Z_i.

We have seen that the choice of a computational procedure, whether it be an algorithm, a so-called heuristic procedure or a probabilistic method, amounts to determining a probability distribution ξ over local maxima. A numerical solution of a problem is obtained in a particular instance by sampling from this distribution. We are therefore led to the problem of how to sample. It is easy to see that this problem depends in an essential way on the information we may have about the probability ξ. For example, if we know we have constructed an exact algorithm, then we know ξ is a one-point distribution, and therefore a single observation from ξ will be sufficient. However, in general ξ will distribute positive probability over several points and our information about ξ may be very limited indeed. Further, rational choice of a sampling procedure must take into consideration the cost of sampling, and the value of the results obtained.

At present our study of the choice of a sampling plan is incomplete. We have derived Bayes-optimal fixed sample size and sequential sampling plan when the *a priori* distribution is rectangular. We also have studied a sequential sampling plan which permits the experimenter to control the

[3] If A is a set, then $|A|$ denotes the cardinal number of A.

probability of achieving a given percentile of the distribution ξ when no *a priori* information about ξ is assumed. Let the number $R(i)$ represent the *utility* or *return* due to accepting as optimal a point p for which $f(p) = Z_i$. The $R(i)$ such that $f(p^*) = Z_i$ is called the *optimal utility* and is denoted by R^*. We assume $R(1) \geqq 0$ and that $i > j$ implies $R(i) \geqq R(j)$.

The cost of observing a local maximum depends on the amount of computation performed in going from the starting point p to its ultimate successor $\hat{p}$. This cost is therefore a random variable whose distribution depends on the probability law F used for sampling, the successor functions s and the computer used. Thus, for each F and s, there is an associated probability distribution of costs. We consider only the expected cost $C_{F,s}$ per observation and suppress the subscripts F and s.

It can be shown that the Bayes-optimal sequential sampling plan against the rectangular *a priori* distribution over the space Ξ_r of r-nomial distributions ξ, is to take one observation and continue sampling as long as the expected gain V_m from one more observation, given that Z_{i_m} is the best value observed in the m observation so far taken, exceeds the expected cost C of one more observation. The expected gain V_m is given by

$$V_m = \sum_{i > i_m} \frac{1}{m + r} [R(i) - R(i_m)].$$

It can also be shown that the number of observations to a stop is bounded by

$$\frac{1}{C} \sum_{i=2}^{r} R(i) - \frac{r - 1}{C} R(1) - r.$$

The Bayes optimal fixed sample size for the rectangular *a priori* distribution is given by the integer value of m which most nearly satisfies

$$\sum_{i=1}^{r-1} [R(i) - R(i + 1)] \frac{(r - 1)!\,(m + i - 1)!}{(i - 1)!\,(m + r - 1)!} = C.$$

Alternatively, we may not have enough information to specify an *a priori* distribution. In this case it is still possible to sample so as to control the probability that a better value remains unobserved. We sample until a run of $K(i)$ observations appears without improvement of the best value Z_i observed. We can specify a desired error probability α and choose

$$K(i) = \log (1 - \alpha) \Big/ \log \left(\frac{C}{R^* - R(i)} \right),$$

thereby ensuring that the probability is smaller than α that a run of $K(i)$ observations without improvement will occur when the probability of improvement is greater than the allowable quantity $C/(R^* - R(i))$.

1.2. Choice of successor structure. We have so far left open the question of how to choose a successor structure. This question amounts to asking what method of solution, algorithm or local search technique should be used on the problem at hand. The answer to this question depends of course on the characteristics of the particular problem to be solved. It is therefore surprising that anything at all useful can be said about the choice of successor structure when nothing more is assumed about the characteristics of the problem $(\mathcal{P}, f)$. Nevertheless some generalizations are possible; moreover these are supported by the computational experience so far accumulated. Without presenting any of that material here, we summarize the practical implications drawn from it. These are that one should look for ways of search which are cheap and easy to carry out with the computing equipment available and then to extend such ways of search until one encounters sharply increasing costs of computing. The procedures developed for the traveling salesman problem, especially those called Algo IV(r), illustrate this suggestion.

Three remarks enlarging the set of strategies. In this section we indicate three important ways in which the set of strategies open to the problem-solver can be enlarged. These are:

(i) A non-uniform but fixed distribution F can be used to select starting points in $\mathcal{P}$.

(ii) The distribution F of starting points can be made a conditional distribution, depending on what has been observed.

(iii) The successor function s can be made probabilistic.

Non-uniform starting. It is essential for the theory presented above that optimal points of $\mathcal{P}$ be among the locally maximal points and that such points be sampled with positive probability. In the absence of information about $\mathcal{P}$ and f (other than that $\mathcal{P}$ is finite and f restricted to a known interval), only a distribution which is positive everywhere on $\mathcal{P}$ can guarantee that optimal points have positive probability of being observed. In the absence of further information, the uniform distribution F^0 is an "obvious" choice. However, if other information is available it should obviously be used. If it is suspected that optimal points are more likely to be found in one part of $\mathcal{P}$ rather than another, a distribution F should be used which gives more probability to the former region than to the latter.

Learning. Consider that we begin sampling using the uniform distribution F^0 on $\mathcal{P}$. As sampling proceeds we have the possibility of learning something about the structure of $\mathcal{P}$ and f from the observations so far made. We may be able to use such knowledge to make further observation more efficient. We can "learn" about $\mathcal{P}$ and f in several different ways:

1. We can look for characteristics associated with locally maximal points:

2. We can look for characteristics associated with locally maximal points at which f is relatively large.

3. We can look for characteristics associated with points whose ultimate successors are locally maximal points at which f is relatively large.

Let $\mathcal{Y}$ be a list of characteristics likely to be associated with locally maximal points and for $Y \subset \mathcal{Y}$, let $h(Y)$ be those points in $\mathcal{P}$ which have at least one of the properties Y. Then, we can sample more heavily from the set of points in $\mathcal{P}$

$$h(Y) \subset \mathcal{P}$$

with the desired characteristics. We can give more attention to those points in sampling, e.g., by using for choice of starting points the conditional distribution

$$F(p \mid Y, \lambda) = \lambda F^0(p) + (1 - \lambda)F^0(p \mid Y), \qquad \lambda(t) > 0,$$

where $F^0(p \mid Y)$ is the uniform distribution over the set $h(Y)$. The function λ can be made to depend on the number of observations so far available so as to express the problem-solver's willingness to give weight to what he has so far "learned." If the conditional probability of values of f, given the characteristics Y depends strongly on Y, then h is quite informative about f, and can be expected to identify points in $\mathcal{P}$ more sharply as observations accumulate.

Because $F(p \mid Y, \lambda)$ is positive everywhere on $\mathcal{P}$, the theory presented earlier remains applicable even when this type of "learning" is used; the "learning" may help to reduce the number of observations needed to attain a given level of confidence at the end.

This type of learning has been used with notably good effect in several problems to be reported elsewhere.

Randomized successor functions. We can replace deterministic search of the neighborhood of a point by a probabilistic search procedure without changing any essential part of the theory so far developed. Probabilistic search can be expressed in terms of randomized successor functions, thus enlarging the class of successor structures available to the problem-solver. Randomized successor functions may in some cases be more efficient ways of exploring than are deterministic ones. They make the number of points examined in a neighborhood independent of the configuration of function values and also have the virtue of guarding against possibly unfavorable problem configurations, an especially comforting feature when nothing is known about the problem. These procedures are very much in the spirit of our method; they appear here at the end not because we regard them as being less important, but only because the theory has a simpler expression in the case of deterministic successor functions. The generalization to randomized successor functions is straightforward.

2. Traveling salesman problems. The traveling salesman problem (or optimal sequencing problem) can be stated as follows:

There are n cities with known distances between each pair of cities. A salesman must visit each city once on a proposed trip and return to his home city. The salesman's problem is to find the tour (i.e. the order in which he visits the cities) which minimizes the total distance he must travel.

More formally, the problem can be stated as follows:

Let the real symmetric matrix $A = (a_{ij})$, $i, j = 1, 2, \cdots, n$, give the distances between all possible pairs of cities and let $\mathcal{P} = \{p\}$ be the set of all permutations of the first n integers. Then denoting a permutation p by $(p_1, p_2, \cdots p_n)$, the distance traveled by taking tour p is given by

$$f(p) = \sum_{i=1}^{n-1} a_{p_i p_{i+1}} + a_{p_n p_1}.$$

The problem then is to find p_0 such that $f(p_0) \leq f(p)$ for all $p \in \mathcal{P}$. These problems have been extensively studied and various algorithms have been devised for attacking them. Although these problems do not lend themselves to easy solution a number of examples exist which do have (or were thought to have) known solutions. Our purpose was to use these problems with known solutions to gain some insight into the worthiness of our approach.

The problems on which we have worked include: Ten 9-city problems given in [1], a 15-city problem, a 20-city problem [2], a 25-city problem [3], a 33-city problem [5], a 42-city problem [4], a 48-city problem [3], and a 57-city problem [5].

A family of algorithms were developed and applied to the problems mentioned above. These algorithms which we have labeled Algo I, Algo II, Algo III, Algo IV and Algo IV(r), $r = 1, 2, \cdots, n - 1$, are described below. They were designed for computational convenience. Algo I, Algo II and Algo III were used in this study only to a limited extent; the Algo IV(r) series was used on all of the problems mentioned above for several values of r. Most of our tabulated results are for the Algo IV(r) experiments. In each case an estimate is made of the value of r which would have given the best performance. This, of course, involves hindsight and cannot always be expected to carry over to different traveling salesman problems. We maintain, however, that the evidence presented here indicates that in practical situations, our methods are likely to be valuable.

To relate the concepts of §1 to §2, *the function f (of §1) is the function which measures tour length.* Optimality then, is equivalent to minimization.

The cost of sampling was taken to be the average amount of computer time necessary to make an observation, i.e., the computing time taken to find a locally minimal tour after starting with a randomly chosen tour.

2.1. The algorithms. *Algo I.* Algo I proceeds as follows: A random permutation p of $\{1, 2, \cdots, n\}$ is selected and $f(p)$, the length of tour p, is computed. The permutation p' is derived from p by inverting the first and second elements of p. Then $f(p')$ is compared with $f(p)$; if $f(p')$ is less than $f(p)$, p' replaces p and $f(p')$ replaces $f(p)$. Then the second and third elements of the resulting permutation are inverted to form a new permutation p'' and again comparison is made. This process is continued until n consecutive interchanges have been checked without a change in the permutation, i.e., without reducing the length of the tour. At this point a locally minimal tour has been found. The tour $\hat{p}$ and its length $f(\hat{p})$ are recorded, another random starting permutation is chosen, and the process repeated.

Algo II. Algo II is similar to Algo I except that instead of comparing two permutations (i.e., by considering the two permutations of a given pair of adjacent points), six permutations are compared. The six permutations are those obtained from the six permutations of $i, i + 1, i + 2 \pmod n$ for $i = 1, 2, \cdots, n$. A locally minimal permutation is one such that no better permutation (i.e., one with a shorter tour length) can be found by permuting any three adjacent points in that permutation.

Algo III. Let $p^1 = (i_1, i_2, \cdots, i_n)$ be a (randomly selected) permutation of the integers $1, 2, \cdots n$. Now form a new permutation, say p^2, by interchanging the values of i_1 and i_2 to form p^2 (i.e., i_1 in p^2 is the i_2 in p^1 and the i_2 in p^2 is i_1 in p^1). Then compute the tour length $f(p^2)$ and record its value. Next interchange i_2 of p^2 and i_3 of p^2 to form a new permutation p^3. Compute $f(p^3)$ and record. Then interchange i_3 of p^3 and i_4 of p^3 to form p^4. Compute $f(p^4)$ and record, etc. This process continues until $f(p^1), f(p^2), \cdots, f(p^{n-1}), f(p^n) = f(p)$ have all been computed. A permutation p_a such that $f(p_a) \leq f(p^i)$, $i = 1, 2, \cdots, n$, is determined and is used as a starting point for another set of comparisons. The next set of comparisons is carried out by starting with the second point in the permutation p_a. This can easily be done by circling the elements of p_a one position to the left and then operating on the resulting permutation in the same manner as described above. The process is continued until n sets of $(n - 1)$ permutations have been examined without decreasing f. At this point a locally minimal permutation has been found. Another random permutation is then chosen and the algorithm proceeds to find another local minimal permutation.

Algo IV(r). Algo IV(1) is just Algo III. Algo IV(2) is an extension of Algo III. When a locally minimal permutation, p, has been found by means of Algo III, two adjacent points in p are moved as a pair, being placed in that position and orientation which minimize the length of the tour. This procedure of moving two adjacent points together continues until no pair of points can be moved so as to decrease the tour length. Algo IV(3) is similar process for triples of adjacent points after Algo IV(2) can find no further improvements, while Algo IV(r) continues until no advantageous move can be made by applying Algo III, Algo IV(2), Algo IV(3), $\cdots$, Algo IV($r - 1$).[4]

2.2. The computational experiments. The computing results presented here are results of a random sampling experiment. As such they are subject to random errors. Sample sizes here were restricted by the size and speed of the computing facilities available to us. Every known precaution was made to insure against nonrandomness.

It is noticeable that the relative frequency with which the optimal solution is observed does not always vary in the way which one might expect. An extreme example of this occurs in the 15-city problem, the relative frequency of the optimal solution here was (.354) for Algo IV(5) but only (.307) for Algo IV(7). We might have expected the second figure to be larger than the first because Algo IV(7) uses larger neighborhoods than does Algo IV(5). The difference here could be simply a matter of a chance observation of an event whose probability is small.

The tables describe the results of our experiments; they show the frequencies with which the locally minimal tours were found. The corresponding relative frequences appear in parentheses next to some of the frequency entries. There are also data which estimate the amount of computing time necessary to find an optimal solution. It is assumed that the same computer programs are used on the same type of machine.[5] Finally, for purposes of illustration, some calculations are presented using data from the 15-city problem which shows how one might arrive at a sample plan for that particular problem.

2.3. The calculation of neighborhood sizes for the Algo IV(r) series. Let n be the number of cities in a traveling salesman problem and p be a permutation of the integers $1, 2, \cdots, n$. A neighborhood of p is regarded here to be the total number of permutations which the algorithm would evaluate before determining that p is locally minimal. Since the Algo IV(r) series treats all permutations alike, all neighborhoods have the same size.

If the size of the neighborhood, under Algo IV(r) is denoted by $N(r)$, then $N(1) = n^2 - 3n + 1$ and $N(2) = 3n^2 + 3n + 1$. We also have

[4] Flow charts for these algorithms have been drawn and are available from the authors upon request.

[5] IBM 1620 Model 1.

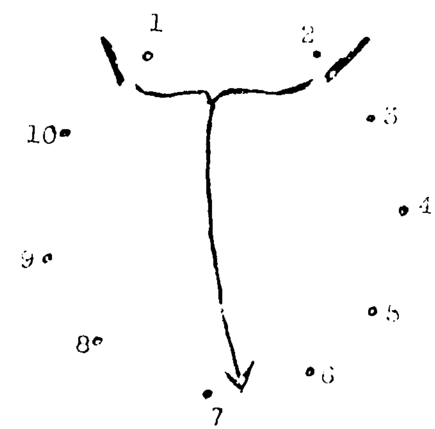

Fig. 1.

$N(r) = N(r - 1) + E(r)$ where $E(r)$ denotes the number of permutations which are examined by Algo IV(r) but not by Algo IV($r - 1$). To evaluate $E(r)$, consider the path defined by a particular permutation and the number of new permutations which can be made by moving any sub-tour of length r (i.e., any r adjacent cities in the permutation) into another position between any 2 adjacent cities which are not included in the sub-tour. For example let $n = 10$, $r = 2$, and the permutation $p = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)$. The sub-tour of length 2 consisting of cities (1, 2) might be moved into a position between cities 6 and 7 as pictured in Fig. 1.

A point in the neighborhood of p for Algo IV(2) is any permutation which can be reached by Algo IV(1) or by moving any 2 adjacent points (e.g., 1, 2) in between any other pair of adjacent points (e.g., 6 and 7) in either orientation (e.g., the permutation (3, 4, 5, 6, 1, 2, 7, 8, 9, 10) of (3, 4, 5, 6, 2, 1, 7, 8, 9, 10)).

If the sub-tour length is r, there are a total of $n - 3r - 1$ positions (provided of course that $n \geqq 3r + 1$) to which the sub-tour could be moved which are greater than r positions from the sub-tour. For each of these $n - 3r - 1$ positions, two new permutations can be formed by considering both orientations of the sub-tour. This gives a total of $2(n - 3r - 1)$ new tours.

If the sub-tour is moved less than r positions in either direction, only one new permutation can result since one of the two orientations involved could have been effected by moving a sub tour of length less than r. There are a total of $2(r - 1)$ of these types of positions in the tour. In addition to these two types of new permutations, there are the permutations which can be arrived at by considering the two positions which lie exactly r

positions away from the sub-tour. Since there are a total of n sub-tours of length r, and 2 of the 4 permutations arrived at by moving a sub-tour exactly r positions are counted by 2 of the sub-tours, there are a total of $n[(2(n-3r-1)+2(r-1)+3] = 2n^2-4rn-n\ (=E(r))$ new permutations formed by Algo IV(r).

From this reasoning we have that $E(r) = 2n^2 - 4nr - n$ if $n-3r-1 \geqq 0$ or $r \leqq \frac{1}{3}(n-1)$. If $r > \frac{1}{3}(n-1)$, a sub-tour of length r can move to a total of $n-r-1$ new positions but only one orientation for each of these positions will not have been reached by Algo IV(r') for some $r' < r$. Since there are a total of n sub-tours of length r, $E(r)$ would equal $n(n-r-1) = n^2 - rn - n$. In summary then we have

$$N(r) = N(r-1) + 2n^2 - 4rn - n, \qquad \text{if } 2 \leqq r \leqq \tfrac{1}{3}(n-1),$$

$$N(r) = N(r-1) + n^2 - rn - n, \qquad \text{if } \tfrac{1}{3}(n-1) < r < (n-1),$$

$$N(1) = n^2 - 3n + 1.$$

Summing these recurrence relations we get

$$N(r) = n(r-1)[2n-2r-5] + n^2 - 3n + 1, \quad \text{for } 2 \leqq r \leqq \tfrac{1}{3}(n-1),$$

$$N(r) = \tfrac{1}{9}n(n-4)(4n-13) + n^2 - 3n + 1$$
$$+\, n(r - \tfrac{1}{3}n + \tfrac{4}{3})(\tfrac{7}{6}n + \tfrac{1}{2}r - \tfrac{7}{6}), \quad \text{for } \tfrac{1}{3}(n-1) \leqq r \leqq (n-1).$$

The 9-city problems. The experiments in Tables 1 and 2 are a series of 9-city problems which were constructed by Robacker [1] who determined the optimal solutions by deterministic methods. In every case our solutions are at least as good as those found in [1]. Our solution is better on Problems 4 and 8. The computing time involved for each observation was essentially limited only by the speed of the card punch on the computer. These items are interesting in part because they show the variability of performance of algorithms on different problems from the same source (i.e., in this case 9-city problems whose distances were randomly generated). For example, with Algo III, 39 locally minimal tours were found on Problem 8 and only 10 on Problem 4 even though more computing time was spent on Problem 4 than on Problem 8. The optimal tour for Problem 2 was observed with a relative frequency of .476 while that of Problem 5 was observed only once in 70 trials. Problem 5 remained the toughest to solve with Algo IV(2), Algo IV(3) and Algo IV(4).

A 15-city problem.[6] For this problem it is believed that the optimal tour has $f = 416$ (see Table 3), but this has not been proved.

Our best tour (length 416) was shorter than the tour (length 417) which had been previously conjectured as optimal. The cities, given in order

[6] Communicated to the authors in private correspondence.

TABLE 1. *Algo III.**

								Problem 1									
f	232		257	258	271	272	289	290	305	317							
FO	20	(.308)	7	11	6	3	3	10	1	4							(Σ = 65)
								Problem 2									
f	204		225	232	243	254											
FO	40	(.476)	18	5	7	14											(Σ = 84)
								Problem 3									
f	178		184	192	193	197	214	216	223	226	234	250					
FO	16	(.211)	20	6	5	7	15	2	1	1	2	1					(Σ = 76)
								Problem 4									
f	181		192	201	205	212	219	222	224	229	230						
FO	24	(.209)	21	12	27	6	8	2	7	3	5						(Σ = 115)
								Problem 5									
f	283		288	297	299	302	305	310	314	338							
FO	1	(.013)	20	17	8	8	12	2	9	2							(Σ = 79)
								Problem 6									
f	150		159	173	176	177	179	180	182	188	193	194	199	214			
FO	5	(.038)	17	8	20	11	22	11	14	8	6	3	4	1			(Σ = 130)
								Problem 7									
f	185		186	193	199	200	201	202	208	209	211	214	216	··· ···	255		
FO	12	(.065)	6	28	2	17	25	10	7	16	7	3	15	··· (*35, 8*) ···	1		(Σ = 181)
								Problem 8									
f	159		165	168	169	174	175	173	182	184	185	189	··· ···	265			
FO	14	(.133)	5	5	1	1	1	1	2	6	3	2	··· (*59, 6*) ···	2			(Σ = 105)
								Problem 9									
f	266		270	279	281	285	292	293	295	296	298	300	301	303	305	306	
FO	46	(.161)	57	14	20	19	18	25	9	21	4	9	1	10	15	11	(Σ = 285)
								Problem 10									
f	236		239	240	244	266	267	279	286	293							
FO	6	(.063)	26	15	22	6	9	3	5	3							(Σ = 95)

* f is the tour distance; FO is the frequency of observations. Numbers in parentheses indicate the relative frequency of the observation; Σ is the total number of observations. Where data are omitted, the number of omitted observations is indicated by the first of the italicized pair in parentheses; the second of the pair is the maximum number of observations for a single value of f.

of the best known tour, are Chicago, Colon, Caracas, Buenos Aires, Capetown, Auckland, Honolulu, Khabarovsk, Hong Kong, Calcutta, Bombay, Cairo, Istanbul, Berlin, Dakar. The distances have been scaled down from nautical miles as a matter of convenience.

According to the results (Table 3), the algorithms with relatively small neighborhood sizes worked best on this problem. Algo IV(2), for example, gave the best performance. For Algo III, there are an additional 38 observations distributed over 20 values of f ranging in value to $f = 555$ for a total of 331 observations (average computing time per observation = 8 seconds).

The 20-city problem. The 20-city problem, which has been solved by deterministic methods [2], seemed to be more difficult than might have been anticipated from consideration of size alone. Algo III was used for a total of 928 observations without ever finding the optimal tour. The range of the lengths of locally minimal tours found by using Algo III was 261 to 494.

Table 2. *9 city problems, Algo IV(r)*

Problem	f	Frequency of Observations: $r = 2$	$r = 3$	$r = 4$
1	232	96 (.97)	79 (1.0)	87 (1.0)
	294	3 (.03)		
		Σ = 99	Σ = 79	Σ = 87
2	204	93 (.91)	75 (1.0)	79 (1.0)
	220	5 (.05)		
	225	4 (.04)		
		102	75	79
3	178	46 (.46)	78 (1.0)	76 (1.0)
	184	54 (.53)		
	191	1 (.01)		
		101	78	76
4	181	77 (.79)	61 (.77)	59 (.70)
	192	19 (.19)	18 (.23)	25 (.30)
	212	2 (.02)		
		98	79	84
5	283	27 (.25)	35 (.44)	47 (.59)
	288	52 (.48)	45 (.56)	32 (.41)
	297	30 (.28)		
		109	80	79
6	150	72 (.80)	63 (.87)	66 (.10)
	177	18 (.20)	9 (.12)	
		90	72	66
7	185	59 (.58)	44 (.55)	58 (.71)
	193	26 (.25)	34 (.42)	23 (.28)
	199	9 (.09)	2 (.02)	
	201	7 (.07)		
	204	1 (.01)		
		102	80	81
8	159	72 (.67)	56 (.65)	59 (.68)
	168	35 (.33)	29 (.34)	28 (.32)
		107	85	87
9	266	67 (.63)	66 (1.0)	70 (1.0)
	276	7 (.07)		
	281	16 (.15)		
	292	16 (.15)		
		106	66	70
10	236	67 (.63)	69 (1.0)	68 (1.0)
	244	36 (.34)		
	282	3 (.03)		
		106	69	68

Table 3. *15-city problem**

f	Frequency of Observations: III†	IV (2)	IV (3)	IV (4)	IV (5)	IV (6)	IV (7)
416	36 (.112)	80 (.286)	20 (.256)	26 (.317)	63 (.351)	36 (.346)	54 (.307)
417	30 (.094)	45 (.161)	54 (.692)	52 (.634)	106 (.596)	61 (.587)	113 (.642)
421	58 (.181)	86 (.307)					
430	12 (.037)						
452	2 (.006)	5 (.018)					
453	11 (.034)	10 (.036)					
455	21 (.065)						
456	15 (.047)	22 (.079)	4 (.051)	4 (.049)	8 (.045)	5 (.048)	9 (.051)
457	1 (.003)						
459	2 (.006)						
460	4 (.012)						
461	2 (.006)						
462	13 (.041)						
464		2					
⋮	⋮	⋮					
	(76, 15)	*(26, 5)*					
⋮	⋮	⋮					
476					1	2 (.019)	
491	10 (.030)						
508		2					
Σ	331	280	78	82	178	104	176
$\bar{t}$	.120	.133	.218	.232	.273	.308	.333
$\mathcal{E}_n$	9.20	3.50	3.91	3.15	2.75	2.89	4.17
$\mathcal{E}_t$	1.10	.47	.85	.73	.75	.89	1.39

* $\bar{t}$ (in minutes) is the average time per observation; $\mathcal{E}_n$ is the expected number of observations before observing the optimal tour for the first time; $\mathcal{E}_t$ is the expected time to observe the optimal tour the first time.

† 38 additional observations ranging from $f = 509$ to $f = 555$ are not shown.

Algo IV(1) was run for a total of 1272 observations and it found the optimal tour only twice. The locally minimal tour lengths ranged from 246 (which here is the shortest tour length) to 361 for Algo IV(r), $r = 2, \cdots, 10$. The results of the experiments with the Algo IV(r) series are shown in Table 4.

It appears that for this particular problem it would pay to use algorithms with even larger neighborhoods than those we used here. Our computer program, however, only allows the parameter r (from the series Algo IV(r)) to take values up to $\frac{1}{2}n$. The unusually "good" performance of Algo IV(5) is rather striking; a sudden increase in the probability of finding the optimal solution can be noted when proceeding from Algo IV(4) to Algo IV(5).

The 25-city problem. The 25-city problem seems to be relatively easier

TABLE 4. *20-city problem*

f	Frequency of Observations IV (2)	IV (3)	IV (4)	IV (5)	IV (6)	IV (7)	IV (8)	IV (10)
246	1 (.010)	3 (.030)	4 (.040)	30 (.151)	18 (.180)	19 (.188)	32 (.302)	87 (.343)
251	1 (.010)	2 (.020)	16 (.162)	21 (.106)	18 (.180)	24 (.238)	12 (.113)	31 (.122)
252	0	1 (.010)	6 (.061)	21 (.106)	19 (.190)	20 (.198)	28 (.264)	90 (.354)
253	1 (.010)	6 (.060)	9 (.091)	29 (.146)	4 (.040)	6 (.059)	3 (.128)	
255	1 (.010)	4 (.040)	3 (.030)	14 (.070)	9 (.090)	3 (.030)	7 (.066)	31 (.122)
256	0	1 (.010)	2 (.020)	4 (.020)	0	0		
257	1 (.010)	10 (.100)	6 (.061)	5 (.025)	6 (.060)	3 (.030)		
258	3 (.030)	2 (.020)	0	6 (.030)	2 (.020)	2 (.020)		
259	2 (.020)	3 (.030)	7 (.071)	3 (.015)	1 (.010)	3	1 (.009)	
260	2 (.020)	8 (.080)	2 (.020)	11 (.055)	7 (.070)	5 (.049)	7 (.066)	6 (.024)
261	0	1 (.010)	2 (.020)	3 (.015)	1 (.010)	2	0	0
262	2 (.020)	2 (.020)	6 (.061)	4 (.020)	1 (.010)	4 (.040)	9 (.085)	6 (.024)
263	3 (.030)	1 (.010)	5 (.050)	9 (.045)	4 (.040)	1	5 (.047)	2 (.008)
264	2 (.020)	0	4 (.040)	2				
265	2 (.020)	6 (.060)	3	6				
266	3 (.030)	3 (.030)	3	2				
267	1 (.010)	1	0	2				
268	2 (.020)	0	4	5	3 (.030)	2		
269	2 (.020)	1	1	1	1	2		
270	0	5 (.050)	2	4		1		
271	1 (.010)	2	1	0	0	1		
272	4 (.040)	4	2	1	1			
273				1	1	1	2 (.019)	1 (.004)
274	4 (.040)	2	1	2	0	2	0	
275	1 (.010)	2 (.020)	0	1				
276	2 (.020)	4 (.040)	1	2				
277	2	1		1				
278	4	3	1	1	1			
279	2	1		1				
280	3	0	3	1				
281	5 (.050)	1	1	1				
282	2	1		2	2			
283	0	1		1				
284	1	0		1				
285	2	1		1				
286	1	0						
Others	38 (.3762)	2	4	2	1			
Σ	101	100	99	199	100	101	106	254
$\bar{t}$ (min.)	.35	.59	.63	.71	.93	.93	1.08	1.15
ε_n	101.0	33.3	24.7	6.6	5.5	5.3	3.3	2.9
ε_t (min.)	35.4	19.6	15.6	4.7	5.1	4.9	3.6	3.3

for our algorithms than the 20-city problem (see Table 5). Algo III was applied for a total of 1186 observations (1100 of these observations distributed over values ranging to 2493 are not shown). Algo IV(r) was applied for 182 observations.

The value $f = 1711$ is known to be optimal for this problem (solved by Held and Karp [3]).

The 33-city problem. This problem was used in a national contest spon-

TABLE 5. *25-city problem*

f	Frequency of Observations III	IV (2)	IV (3)	IV (4)	IV (5)	IV (8)	IV (10)	IV (11)	IV (12)
1711	21 (.017)	22 (.122)	20 (.267)	28 (.373)	21 (.280)	21 (.280)	21 (.320)	31 (.408)	33 (.440)
1723	11	23 (.127)	10 (.133)	15 (.200)	10 (.133)	14 (.187)	18 (.240)	19 (.253)	12 (.160)
1736	21	11 (.061)	3 (.040)	5 (.067)	9 (.120)	9 (.120)	8 (.107)	3 (.040)	10 (.133)
1748	25	20 (.110)							
1751	8								
1760		4 (.022)							
1763		18 (.099)	12 (.160)	13 (.173)	14 (.187)	17 (.227)	17 (.224)	16 (.213)	11 (.147)
1770		4 (.022)		2	9 (.120)	10 (.133)	6 (.080)	5 (.067)	8 (.107)
1779			2 (.027)	1					
1781		3 (.016)	1 (.013)	2					
1797		6 (.033)	6 (.080)	4	3	3			
1802		3 (.016)	4 (.053)	2	4	1	3	2 (.027)	1
1803		1	1						
1829		2							
1847		3	2	3					
1851					1				
1857		4			1				
1873		3							
1887		9							
1892		1							
1896		1	1		2				
1902			1						
1909		1	1						
1917		2							
1924		1							
1926		1							
1981			2						
1985		11	4						
1993		10	2						
⋮		⋮ (17, 4) ⋮	⋮ (2, 1) ⋮						
2008					1				
2030			1						
2050		1							
Σ	86	182	75	75	75	75	76	76	75
$\bar{t}$	.175	.39	.60	.80	.97	1.19	1.48	1.64	1.71
ε_n	56.5	8.30	3.75	2.68	3.58	3.58	3.12	2.42	2.27
ε_t	9.89	3.24	2.25	2.14	3.47	4.26	4.62	3.97	3.95

sored by a soap manufacturer. It has been worked on by several people[7] and the tour whose distance is 10861 is strongly believed to be the optimal tour. This problem seems fairly easy to solve with our methods (Table 6). Algo IV(3) or Algo IV(4) are probably best. The increased neighborhood sizes have not increased the probability of observing the optimal solution enough to offset the extra computational time necessary to use them. In

[7] For example, Prof. G. Thompson, Carnegie Institute of Technology, who furnished the authors with data concerning the ostensible optimal tour.

TABLE 6. *33-city problem*

f	Frequency of Observations					
	IV (3)	IV (4)	IV (5)	IV (8)	IV (10)	IV (15)
10861	13 (.167)	17 (.226)	17 (.230)	26 (.228)	17 (.217)	23 (.334)
10929			1			
10944	12 (.154)	20 (.267)	27 (.365)	42 (.368)	26 (.334)	34 (.493)
10998	7 (.089)	7 (.093)	8 (.108)	9 (.079)	7 (.090)	5 (.072)
11043				1	3	
11075	6	4	3	9 (.079)	11	
11122		1		3		3
11133	4	3	3	3	2	
11134	2	2	1		1	
11142	4	2	1	1	2	4
11143	3	1	2	3	1	
11147		1				
11212	1	4	2			
11222	2					
11268		4	1	8	3	
⋮	⋮ (22, 3) ⋮	⋮ (9, 3) ⋮	⋮ (7, 6) ⋮	⋮ (8, 3) ⋮	⋮ (4, 3) ⋮	
11570					1	
11590			1			
11847				1		
13606	1					
Σ	77	75	74	114	78	69
$\bar{l}$	1.30	1.67	2.14	2.79	3.57	5.63
$\mathcal{E}_m$	6.0	4.4	4.3	4.4	4.6	3.0
$\mathcal{E}_t$	7.80	7.35	9.20	12.28	16.42	16.89

fact there does not seem to be much change in the probability of observing the optimal solution between Algo IV(4) and Algo IV(10).

The 42-city problem. The 42-city problem has been worked on by several people and seems to be a relatively easy problem considering its size. The optimal solution was worked out by Dantzig, et al. [4]. Algo I, Algo II and Algo III were each applied to this problem but without satisfactory results. However, as our theory leads us to expect, there was a marked improvement in the performance of Algo II over Algo I and Algo III over Algo II. The range of the lengths of the locally minimal tours found with Algo I was 1204–2291, with Algo II it was 605–1178, and with Algo III it was 204–518. The optimal tour length is 167.5 and was found by all of the Algo IV(r) series (see Table 7).

Large neighborhoods seem best for this problem. Algo IV(7) was quite

TABLE 7. *42-city problem*

f	Frequency of Observations												
	IV (2)	IV (3)	IV (4)	IV (5)	IV (6)	IV (7)	IV (8)	IV (9)	IV (10)	IV (12)	IV (13)	IV (14)	IV (15)
1675	2 (.011)	2 (.035)	5 (.075)	6 (.087)	8 (.116)	12 (.169)	8 (.133)	9 (.111)	15 (.220)	14 (.206)	13 (.265)	18 (.261)	18 (.286)
1725	3 (.017)	4 (.069)	2 (.030)	1 (.015)	6 (.087)	6 (.084)	9 (.150)	9 (.111)	7 (.103)	5 (.074)	6 (.116)	8 (.116)	12 (.191)
1735								1 (.012)			1 (.022)	1 (.015)	
1755	6 (.033)	7 (.121)	8 (.120)	5 (.073)	5 (.073)	11 (.155)	13 (.217)	11 (.136)	9 (.132)	6 (.088)	4 (.082)	11 (.159)	7 (.111)
1765	3 (.017)	1	3	3 (.043)	6 (.087)	2	3	4	2	2	2	3 (.044)	6 (.095)
1775	4												
1805	3	3	3	8 (.116)	6	6	4	8	6	4	2	4 (.058)	3
1815	0	2	3	3		4		2	4	2	1	3 (.044)	
1825	2	1	1	1	2	1		3	3	1	1		
1835			1		2		1	3		1	1	4	1
1845	1			1					1			2	
1855	1	2											
1885	1		2	1	1	1	2	2		2			
1895	1	1		1	2	3		3	3	1		1	2
1905	3				1					1			
1935	1	1	3	4	1	3	3			2	5	4	2
1965					1								
1975	0	1	1	1	1					2			
1985				2	2		1	4		2		1	1
1995	1	0	1	1	1	2			5	1		1	
2005	3			2			3	4		2			1
⋮	⋮ (10, 4) ⋮	⋮ (32, 2) ⋮	⋮ (33, 6) ⋮	⋮ (26, 5) ⋮	⋮ (24, 5) ⋮	⋮ (15, 4) ⋮	⋮ (12, 3) ⋮	⋮ (15, 6) ⋮	⋮ (12, 3) ⋮	⋮ (19, 5) ⋮	⋮ (11, 4) ⋮	⋮ (7, 2) ⋮	⋮ (9, 4) ⋮
2145								1					
2155										1	2		
2195													1
2235									1				
2275							1						
2285												1	
2565						2							
2585					1								
2760				1									
2925			1										
3735		1											
4585	1												
Σ	158	58	67	69	70	71	60	80	68	68	49	69	63
$\bar{l}$	2.37	2.52	3.27	3.65	4.05	4.85	5.05	5.46	6.25	6.85	7.76	7.99	8.43
$\mathcal{E}_m$	79.0	29.0	13.4	11.5	8.8	5.92	7.50	9.00	4.53	4.86	3.77	3.83	3.50
$\mathcal{E}_t$	187.2	73.1	42.8	42.0	35.6	28.7	37.9	49.1	28.3	33.3	29.3	30.6	29.5

TABLE 8. *48-city problem*

f	Frequency of Observations			
	IV (12)	IV (15)	IV (18)	IV (21)
11461	2 (.017)	4 (.047)	5 (.045)	7 (.081)
11470	2 (.017)	4 (.047)	7 (.062)	5 (.058)
11474	2 (.017)	1 (.012)	1 (.009)	
11508	2 (.017)	1 (.012)	4 (.036)	3 (.035)
11533		2 (.024)	1	2 (.023)
11545		3 (.036)	2	
11550	7	1	4	1
11556	4	1	4	2
11574	4	2	9	8
11592			1	
11611	1	2	1	3
11631				1
11634			1	
11640	1	2	1	
⋮	⋮	⋮	⋮	⋮
	(88, 6)	(62, 3)	(71, 5)	(54, 6)
⋮	⋮	⋮	⋮	⋮
12249				1
12289			1	
12474	1			
12524		1		
Σ	114	86	113	87
$\bar{t}$	9.3	11.8	12.8	16.4
$\mathcal{E}_n$	56.5	21.5	22.4	12.3
$\mathcal{E}_t$	525.5	253.7	286.7	201.7

successful here but the sample sizes for this problem were not large enough to give reasonable assurance that the results are due to chance variation.

The 48-city problem. This problem was taken from [3]. Our best answer is **11461** (see Table 8); it is better than the authors' conjectured optimum tour which has f = **11470**.

The expected computing time seems to be decreasing as the parameter r in Algo IV(r) increases—at least within the parameter range of our experiments. Due to a lack of available computer time on a computer identical to that used in our other experiments we were unable to gather data for values of r greater than 21.

The 57-city problem. The 57-city problem proved to be quite challenging to the Algo IV(r) series. Although it has not been proved, we feel confident that the optimal tour was found (see Table 9). Extensive work on this problem by others has led to good tours which are not as short as ours.[8]

TABLE 9. *57-city problem*

f	Frequency of Observations					
	IV (12)	IV (15)	IV (18)	IV (21)	IV (22)	IV (28)
12955					5 (.041)	
12967					7 (.058)	2
12985			2		3 (.025)	1
12986		2		1	6 (.050)	2
12998					1 (.033)	4
13013					4 (.033)	1
13034		1			2 (.016)	
13042					5 (.041)	
⋮	⋮	⋮	⋮	⋮	⋮	⋮
	(13, 1)	(21, 1)	(16, 1)	(15, 1)	(85, 2)	(54, 2)
⋮	⋮	⋮	⋮	⋮	⋮	⋮
13751	*					
13785			*			
13811				*		
13861					*	
13938						*
13967		*				
Σ	13	21	18	16	121	64
$\bar{t}$	14.4	16.5	18.8	21.9	21.5	26.1
$\mathcal{E}_n$					24.2	
$\mathcal{E}_t$					520.3	

This problem is probably about as large as we could attack with the computing equipment now available to us. Algo IV(22) was run more often than the others of this series. We also ran Algo IV(28) a number of times hoping it would perform better than Algo IV(22). It appears, however, that Algo **IV**(22) is better.

2.4. Calculating an optimal sample size: An example. For illustrative purposes we will consider the 15-city problem described earlier. From the matrix of distances between pairs of cities a crude lower bound on the optimal tour length was calculated as 351. Any local minimal tour could be used for an upper bound. We will use the tour of length 508. Then r = $508 - 351 + 1 = 158$. We'll assume a rectangular *a priori* probability

[8] Professor G. L. Thompson of Carnegie Institute of Technology has applied his heuristic program to this problem. The shortest tour he found was 12,985 miles. We have found two tours shorter than this, as shown in Table 9.

distribution over all 158-nomial probability distributions and then compute the best fixed sample size for this problem with respect to our assumption.

We are dealing with a minimization problem here. Therefore our object will be to minimize our expected cost. In this case, expected cost for a sample of size m is given by:

$$G(m) = \frac{1}{\binom{m+r-1}{m}} \sum_{i=1}^{r} R(r-i+1)\binom{m+i-2}{m-1} + Cm.$$

For convenience we will assume that C (i.e., the cost of one sample observation) is equal to 1 and the cost of a tour of length $L + i$ is $R(i) = K(L + i) = K(351 + i)$ where K is a suitably chosen factor relating the cost of traveling one unit to the cost of sampling. Filling in the data for the 15-city problem we get:

$$G(m)$$
$$= \frac{1}{\binom{m+r-1}{m}} \sum_{i=1}^{158} K(510-i)\binom{m+i-2}{m-1} + m$$
$$= \frac{510K}{\binom{m+157}{m}} \sum_{i=1}^{158} \binom{m+i-2}{m-1} - \frac{K}{\binom{m+157}{m}} \sum_{i=1}^{158} i\binom{m+i-2}{m-1} + m$$
$$= 510K - \frac{K}{\binom{m+157}{m}} \sum_{i=1}^{158} i\binom{m+i-2}{m-1} + m.$$

Approximating the second term in the above expression by $158Km/(m+1)$ and using elementary calculus, an approximation to the best sample size for the 15 city problem is calculated to be

$$m \cong \sqrt{158K}.$$

This implies, for example, that if the cost of sampling were one dollar and the cost for traveling an extra unit on the tour were 1000 dollars (which would not be unlikely, since each unit in this problem represents about 500 miles and would be traveled many times), then the most economical sample size would be about 400. In this case the optimal tour would almost certainly be observed by any of the algorithms which we used on the 15-city problem.

PURDUE UNIVERSITY
KRANNERT SCHOOL OF INDUSTRIAL ADMINISTRATION REPRINT SERIES
(continued)

101. **John A. Carlson**, "Demand Obsolescence in a Natural Monopoly," ***Land Economics***, Volume XL, No. 3, August 1964.

102. **Edgar A. Pessemier**, "Models Using Experimental Data to Appraise Marketing Strategy," ***Marketing Models, Quantitative and Behavioral*** (published by International Textbook Co.), 1964.

103. **Yasusuke Murakami and Takashi Negishi**, "A Note on a Formulation of [illegible] Economy," ***International Economic Review***, Volume 5, No. 3, September 1964.

104. **John M. Dutton**, "Computers and Production Management," ***The [illegible] Engineer***, Volume 43, No. 9, September 1964.

105. **R. P. Manes and V. L. Smith**, "Economic Joint Cost Theory [illegible]," ***The Accounting Review***, Volume XL, No. 1, January 1965.

106. **Richard G. Swensson**, "Incentive Shifts in a Three-Choice [illegible]," ***Psychonomic Science***, Volume 2, 1965.

107. **John J. Sherwood**, "Self Identity and Referent Others," ***Sociometry***, Volume 28, No. 1, March 1965.

108. **J. M. Dutton**, "Simulation of an Actual Production Scheduling and Work Flow Control System," ***The International Journal of Production Research***, 1963.

109. **William H. Starbuck**, "The Aspiration Mechanism," ***General Systems Journal***, Volume 9, 1964.

110. **Arnold C. Cooper**, "R & D Is More Efficient in Small Companies," ***Harvard Business Review***, May-June 1964.

111. **Robert L. Basmann**, "The Role of the Economic Historian in Predictive Testing of Proffered 'Economic Laws'," ***Explorations in Entrepreneurial History/Second Series***, Volume 2, No. 1, Spring/Summer 1965.

112. **Robert L. Basmann**, "Causal Systems and Stability: A Reply," ***Econometrica***, Volume 33, No. 1, January 1965.

113. **John M. Dutton**, "Production Scheduling—A Behavioral Model," ***The International Journal of Production Research***, 1964.

114. **Richard E. Walton**, "Two Strategies of Social Change and their Dilemmas," ***The Journal of Applied Behavioral Science***, Volume 1, No. 2, April-May-June 1965.

115. **M. June Flanders**, "Measuring Protectionism and Predicting Trade Diversion," ***The Journal of Political Economy***, Volume LXXIII, No. 2, April 1965.

116. **T. Y. Hans Tjian**, "Tabulation of Percentage Points and Properties of h_n^* Distribution," ***Skandinavisk Aktuarietidskrift***, 1964.

117. **R. L. Basmann**, "A Tchebychev Inequality for the Convergence of a Generalized Classical Linear Estimator, Sample Size Being Fixed," ***Econometrica***, Volume 33, No. 3, July 1965.

118. **Donald C. King and H. Neil Rude**, "Implications of Interest Patterns for Vocational Counseling and Retraining Programs," ***The Personnel Journal***, Volume 43, No. 11, December 1964.

119. **Charles W. King**, "Fashion Adoption: A Rebuttal to the 'Trickle Down' Theory," ***Proceedings of the Winter Conference of the American Marketing Association***, December 1963, ***Toward Scientific Marketing***.

120. **Cliff L. Lloyd**, "Lord Preference and Lord Funds," ***The Economic Journal***, Volume 74, No. 295, September 1964.

121. **Charles W. King**, "The Innovator in the Fashion Adoption Process," ***Proceedings of the Winter Conference of the American Marketing Association***, December 1964, ***Reflections on Progress in Marketing***.

(Continued on back cover)

122. EDGAR A. PESSEMIER, "Analyzing the Economic Potential of a Consumer Product," *Proceedings of the Winter Conference of the American Marketing Association*, December 1964, *Reflections on Progress in Marketing*.

123. VERNON L. SMITH, "Experimental Auction Markets and the Walrasian Hypothesis," *The Journal of Political Economy*, Volume 73, No. 4, August 1965.

124. EDWARD AMES AND NATHAN ROSENBERG, "The Progressive Division and Specialization of Industries," *The Journal of Developmental Studies*, Volume 1, No. 4, July 1965.

125. JOHN J. SHERWOOD, "A Relation between Arousal and Performance," *The American Journal of Psychology*, Volume 78, No. 3, September 1965.

126. NATHAN ROSENBERG, "Adam Smith on the Division of Labour: Two Views or One?," *Economica*, Volume 32, No. 126, May 1965.