Heuristic “Optimization”: Why, When, and How to Use It

Stelios H. Zanakis, James R. Evans,

INTERFACES
Vol. 11, No. 5, October 1981

0092-2102/81/1105/0084$01.25

HEURISTIC "OPTIMIZATION": WHY, WHEN, AND HOW TO USE IT

Stelios H. Zanakis

Florida International University, College of Business Administration, Department of Management, Miami, Florida 33199

and

James R. Evans

University of Cincinnati, College of Business Administration, Department of Quantitative Analysis, Cincinnati, Ohio 45221

ABSTRACT. The pressing need of real-world problems for quick, simple, and implementable solutions coupled with a recently increased research productivity on improved and rigorously evaluated heuristic methods are rapidly increasing the need, usage, and respect for heuristics. This paper presents a framework for heuristic "optimization" by systematically examining this change in attitudes towards heuristics, their desirable features, and proper usage.

The word "heuristic" derives from the Greek "heuriskein," meaning "to discover." In that sense, a heuristic aims at studying the methods and rules of discovery [Polya, 1947, pp. 112–113] or assisting in problem solving, which is a process of systematically trying to attain a preconceived but not immediately attainable aim [Polya, 1962, 1963]. Operations Researchers have seen heuristics as procedures to reduce search in problem-solving activities [Tonge, 1961] or a means to obtain acceptable solutions within a limited computing time [Lin, 1975]. To practitioners, heuristics are simple procedures, often guided by common sense, that are meant to provide good but not necessarily optimal solutions to difficult problems, easily and quickly.

This branch of study is "not very clearly circumscribed, . . . often outlined and seldom presented in detail [Polya, 1947]," quick and dirty [Woolsey, 1975a], not abiding by the mathematical etiquette (proofs, theorems, convergence, optimality, etc.), in contrast to algorithms that "are conceived in analytic purity in the high citadels of academic research, heuristics are midwifed by expediency in the dark corners of the practitioner's lair, . . . and are accorded lower status [Glover, 1977]." For many years heuristics have encountered the hostility, scorn, and condemnation of high academic priests, "treating them as the poor man's tools of the trade, as expediencies that should be regarded with deep suspicion [Eilon, 1977]."

Recently, however, this picture started to change, as evidenced by the growing number of related articles in scholarly reputable journals. The Operations Research Society of America awarded the 1977 Lancaster Prize to two papers on heuristic optimization [Cornuejols, Fisher, and Nemhauser, 1977; and Karp, 1977]. The American Institute for Decision Sciences 1980 Innovative Education Award went to a paper on heuristic problem solving [Brightman, 1980]. One may naturally wonder: Why then this change?

OPTIMIZATION; HEURISTICS

We feel that this is caused by a practitioner-researcher-practitioner slow cyclic reaction that is now in a stage of rapid acceleration. Practitioners are more pressured by declining productivity to analyze complex management problems with limited available resources, creating an increased need for good heuristics. Open-minded researchers in academia eventually realize that mathematical rigor does not guarantee success of OR/MS systems in practice, and channel some of their efforts towards the development of efficient heuristics that are then eagerly adopted by practitioners. Undoubtedly, the excellent work of R. Karp [1975a, 1975b, 1976, 1977] accelerated this reaction in two ways: (1) by proving that many practical combinatorial problems, termed *NP*-complete, cannot be solved efficiently by exact algorithms, whose running times increase exponentially with the problem size; and (2) by establishing a framework for probabilistic analysis of heuristic methods that elevated them from poor relatives to a hot topic of sophisticated analysis now respected in the OR/MS academic community.

As more and more researchers take heuristics more seriously (as M. Fisher [1980] admits is happening now), improved heuristics and documented performance analysis will increase their proper and successful use in practice, thus leading to additional developments (cyclic reaction). An increased emphasis on heuristic optimization should be expected in both academia and businesses over the next several years.

In this paper, we attempt to provide a unifying framework on the advantages/disadvantages, uses/misuses, do's/don'ts, of heuristic optimization. Some of the general issues have been previously addressed by other writers [e.g., Evans, 1979; Ignizio, 1980; Silver, Vidal, and DeWerra, 1980; Woolsey, 1975b; Zanakis, 1977; Zanakis, Austin, Nowading, and Silver, 1980]. Our emphasis here is to present a summary of practical guidelines on the use of heuristics for optimization. Thus we prefer the term "heuristic optimization," although the words seem inconsistent. Alternatively, one may use "heuristic programming" (more restrictive) or "heuristic procedures/algorithms" (more general), which could include special classes of problems, such as scheduling.

WHY AND WHEN TO USE HEURISTICS

There are several instances where the use of heuristics is desirable and advantageous:

(1) *Inexact or limited data* used to estimate model parameters may inherently contain errors much larger than the "suboptimality" of a good heuristic. We have seen monumental efforts in academia and industry to develop and/or use ILP codes, for example, to solve large-scale capital budgeting problems with intangible benefit estimates or risk analysis. In one case [Zanakis, 1977], an existing 0-1 heuristic was easily modified within a day and successfully run to produce in 1/10 of CPU time 99.7% of the "optimal" solution to a 308×372 capital budgeting problem with intangible benefits, which had been solved "optimally" elsewhere with a special purpose algorithm after a six-month developmental effort.

(2) *A simplified model* is used, which is already an inaccurate representation of the real problem, thus making the "optimal" solution only academic. In both cases 1 and 2, a fast near-optimal solution makes more sense than a time-consuming exact solution to an inexact problem.

(3) *A reliable exact method is not available*. Small companies may not have access to a good optimizing code. One of our students, working on an actual problem for a utility company, found that a classroom heuristic produced a better solution than the optimization code the company had been using! And there are problem solvers in this world that, without questioning, accept a code's final solution as optimal.*

(4) *An exact method is available, but it is computationally unattractive* due to excessive time and/or storage requirements. Large real-world complex problems may prevent an optimizer from finding an optimal or even a feasible solution within a reasonable effort. Heuristics, on the other hand, can produce at least feasible solutions with minimal time and storage requirements.

(5) *To improve the performance of an optimizer*, heuristics have been used to provide starting solutions and/or guide the search and reduce the number of candidate solutions (e.g., in branch-and-bound).

(6) *Repeated need to solve the same problem frequently* or on a real-time basis, which may translate a heuristic's speed into significant computing savings.

(7) *A heuristic solution may be good enough* for a manager if it produces results better than those currently realized.

(8) Heuristics are *simple* so users can *understand* them, gain confidence in the approach, and therefore be more likely to *implement* the recommended changes.

(9) As a *learning device,* heuristics may help in gaining insight into a complex problem, which then may be modeled and solved more pragmatically.

(10) Other *resource limitations* (e.g., project time and budget, manpower availability and knowledge, etc.) may enforce use of a heuristic.

Most of the above reasons have been advocated by practice-oriented persons and are epigramatically summarized elsewhere: "Managers want to improve their current operation as cheaply and quickly as possible, care little about an optimal solution to a problem with usually inexact data, and will not accept a new solution they do not understand [Zanakis, Austin, Nowading, and Silver, 1980]." As T. Levitt [1978] noted, *simplicity* is the only way to manage anything, even in today's complex world.

We are not advocating that optimizing methods be dispensed with; indeed, if one can effectively apply an optimization algorithm to solve a problem, then by all means do. In large-scale applications, such as facility location for instance, the difference of a percent or two from optimality may mean several hundred thousand dollars to the decision maker. For such capital-intensive decisions, more care would most likely go into data collection and model building; therefore an optimizing technique would be appropriate. However, for many real-world applications, the reasons outlined above often suggest a heuristic method.

FEATURES OF GOOD HEURISTICS

In general, and without regard to a specific problem, a good heuristic should have the following properties or features:

(1) *Simplicity,* which facilitates user understanding and acceptance.

(2) Reasonable *core* storage requirements.

(3) *Speed,* i.e., reasonable computing times that do not grow polynomially or even exponentially as problem size increases.

**Ed. Note:* I observed this 10 years ago in *Management Science,* 1971, Vol. 17, No. 7, p. 500.

(4) *Accuracy,* i.e., small average and mean square errors; these will increase (decrease) the chances of a good (poor) solution. The degree of acceptable accuracy should be dictated by the problem and the user.

(5) *Robustness,* i.e., the method should obtain good solutions, in reasonable times, for a wide variety of problems and not be too sensitive to changes in parameters. This is not true of many branch-and-bound methods where a small change in a parameter may double or triple running times.

(6) Accept *multiple starting points,* not necessarily feasible — a formidable task in large and severely constrained problems.

(7) Produce *multiple solutions* (ideally in a single run) by properly selecting input parameters or perturbing the final solution. This allows the user to select the result that is most accurate or ''satisficing'' (in case of unquantifiable intangibles). A new perturbation heuristic, added to several existing 0-1 LP heuristics, captured about 65% of possible improvement to optimality, thus increasing average accuracy to 99.8% [Zanakis and Gillenwater, 1981].

(8) Good *stopping criteria* that take advantage of search ''learning'' and avoid ''stagnation.'' These are in general some ratio measure of objective function improvement per additional computing effort.

(9) *Statistical estimation* of the true optimum from a series of iterative solutions (see next section and appendix).

(10) *Interactive* ability with the analyst and decision maker.

An ideal heuristic is one that has all of the above properties. It should be noted that problem-dependent heuristics, that take advantage of the special structure of a problem [e.g., Evans and Cullen, 1977], are more efficient than general mathematical programming heuristics, but their use is limited only to a specific class of problems. A classical example is scheduling heuristics vs integer linear programming heuristics applied to scheduling problems.

HOW TO USE HEURISTICS

Many people may argue that proper use of heuristics is more of an art than a science. It is certainly based on common sense, but some form of analytical work is needed to increase confidence in the heuristic results (current and future). A few general guidelines are presented below:

(1) Use heuristics carefully and cautiously; understand their capabilities and limitations. Don't fall in love with them (a tendency of many analysts towards their intellectual baby).

(2) Decide whether a heuristic solution on the total problem is preferable to an exact solution of heuristically obtained subproblem(s), i.e., solution space reduction vs problem simplification (decomposition, partitioning, aggregation, and structural approximation). A more detailed discussion of these heuristic strategies is provided by Evans [1979]. Random, biased, and improvement sampling for generating solutions to large combinatorial problems is another heuristic approach [Mabert and Whybark, 1977].

(3) ''Clean'' the model before solution; i.e., check data accuracy, rearrange (or eliminate redundant) variables and constraints, etc. We have often noticed improved performance of heuristics when such redundancy was eliminated.

(4) Be aware of the pitfalls of the common practice of rounding continuous optimal solutions in discrete-variable problems [Glover and Sommer, 1975].

(5) Establish appropriate measures of performance, such as CPU time, number of iterations, percent of times optimal, average (percent) error, mean-squared error, maximum error (not necessarily correlated with average error), etc.

(6) Solve a sufficient number of literature *and* large-scale randomly generated test problems of the type for which the method was designed. The challenging difficulty of the former is usually structure, while for the latter it is size. Literature test problems are hard-to-solve benchmark problems, usually with known optimal solutions — thus of small size. Of great importance is the generation of "realistic" large-size problems with *known* optimal solutions (which cannot be determined with an exact method). This has been achieved for linear programming [Charnes *et al.*, 1974], nonlinear programming [Rosen and Suzuki, 1965], and more recently for integer programming test problems [Fleisher and Meyer, 1976; Rardin and Lin, 1980].

(7) Validate solution results statistically. Use analysis of variance [Zanakis, 1977] and experimental design approaches [Lin and Rardin, 1979] to understand the relationships between problem characteristics and method performance, i.e., when to use or not use a certain heuristic.

(8) Predict or bound the performance of a heuristic by using one or more of the following approaches:

(a) *Problem relaxation* to establish upper and lower bounds on the optimum by relaxing some problem requirement, e.g., integral variables, some constraint(s), etc.

(b) *Worst-case study* to establish analytically the maximum possible error of a heuristic on a certain class of problems [Christofides, 1976; Cornuejols *et al.*, 1977; Fisher, 1980].

(c) *Probabilistic analysis,* where one assumes a probability distribution of problem data to establish statistical properties of a heuristic [Karp, 1976, 1977]. Only simple density functions can provide tractable results, which is a limitation of this approach.

(d) *Statistical inference* to determine point and interval estimates of the true (unknown) optimal value based on a series (sample) of iterative solutions. The idea of point estimation for large combinatorial problems was proposed earlier by Swart [1967] and later extended by others [e.g., Golden, 1977; Golden and Alt, 1979; Dannenbring, 1977; Zanakis and Gillenwater, 1981]. These procedures are based on the statistical property of the smallest (or largest) sample observation to follow an extreme-value or a three-parameter Weibull distribution, when the sample size gets large. Because of its simplicity and the fact that it has not been publicized widely, this estimation procedure is summarized in an appendix.

CONCLUSIONS

The need for and usage of good heuristics in both academia and businesses will continue increasing fast. When confronted with real world problems, a researcher in academia experiences at least once the painful disappointment of seeing his product, a theoretically sound and mathematically "respectable" procedure, not used by its ultimate user. This has encouraged researchers to develop new improved heuristics and rigorously evaluate their performance, thus spreading further their usage in practice, where heuristics have been advocated for a long time. More work is needed in the area of heuristic optimization, particularly in the multiobjective case [Zanakis, 1981], where the available codes seriously fail to meet the real world problem needs.

References

Brightman, H., 1980, "Teaching Quantitative Techniques by General Problem Solving Heuristics," *Proceedings AIDS 12th National Conference*, pp. 116–118.

Charnes, A., *et al.*, 1974, "On Generation of Test Problems for Linear Programming Codes," *Communications of the ACM*, Vol. 17, No. 10, pp. 583–586.

Christofides, N., 1976, "Worst-Case Analysis of a New Heuristic for the Traveling-Salesman Problem," MS Report No. 388, Carnegie-Mellon University, February.

Cooke, P. 1979, "Statistical Inference for Bounds of Random Variables," *Biometrika* Vol. 66, pp. 367–374.

Cornuejols, G., Fisher, M. L., and Nemhauser, G. L., 1977, 1979, "Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms," *Management Science* Vol. 23, pp. 789–810 (Errata in *Management Science* Vol. 25, pp. 808–809).

Dannenbring, D. G., 1977, "Procedures for Estimating Optimal Solution Values for Large Combinatorial Problems," *Management Science* Vol. 23, pp. 1273–1283.

Eilon, S., 1977, "More Against Optimization," *Omega* Vol. 5, pp. 627–633.

Evans, J. R., 1979, "Toward a Framework for Heuristic Optimization," *Proceedings AIIE Annual Conference, Spring 1979*, pp. 284–292.

Evans, J. R. and Cullen, F. H., 1977, "The Segregated Storage Problem: Some Properties and an Effective Heuristic," *AIIE Transactions* Vol. 9, pp. 409–413.

Fisher, M. L., 1980, "Worst-Case Analysis of Heuristic Algorithms," *Management Science* Vol. 26, pp. 1–17.

Fleisher, J. M. and Meyer, R. R., 1976, "New Sufficient Optimality Conditions for Integer Programming and Their Application," Computer Science Technical Report No. 278, University of Wisconsin, Madison, October.

Garey, M. R. and Johnson, D. S., 1976, "Approximation Algorithms for Combinational Problems: An Annotated Bibliography," in *Algorithms and Complexity*, J. F. Traub, ed., Academic Press.

Glover, F. and Sommer, D. C., 1975, "Pitfalls of Rounding in Discrete Management Decision Problems," *Decision Sciences* Vol. 6, pp. 211–220.

Glover, F., 1977, "Heuristics for Integer Programming Using Surrogate Constraints," *Decision Sciences* Vol. 8, pp. 156–166.

Golden, B. L., 1977, "A Statistical Approach to the TSP," *Networks* Vol. 7, pp. 209–225.

Golden, B. L. and Alt, F. B., 1979, "Interval Estimation of a Global Optimum for Large Combinatorial Problems," *Naval Research Logistics Quarterly*, pp. 69–77.

Ignizio, J. P., 1980, "Solving Large-Scale Problems: A Venture into a New Dimension," *Journal of Operational Research Society* Vol. 31, pp. 217–225.

Karp, R., 1975a, "On the Computational Complexity of Combinational Problems," *Networks* Vol. 5, pp. 45–68.

Karp, R., 1975b, "The Fast Approximate Solution of Hard Combinatorial Problems," *Proceedings 6th SE Conference Combinatorics, Graph Theory, & Computing*, pp. 15–31.

Karp, R., 1976, "The Probabilistic Analysis of Some Combinatorial Search Algorithms," in *Algorithms and Complexity* (J. F. Traub, ed.), Academic Press.

Karp, R. M., 1977, "Probabilistic Analysis of Partitioning Algorithms for the Traveling-Salesman Problem in the Plane," *Mathematics of Operations Research* Vol. 2, pp. 209–224.

Levitt, T., 1978, "A Heretical View of Management 'Science'," *Fortune*, December 18, pp. 50–52.

Lin, S., 1975, "Heuristic Programming as an Aid to Network Design," *Networks* Vol. 5, pp. 33–43.

Lin, B. W. and Rardin, R. L., 1979, "Controlled Experimental Design for Statistical Comparison of Integer Programming Algorithms," *Management Science* Vol. 25, pp. 1258–1270.

Mabert, V. A. and Whybark, D. C., 1977, "Sampling as a Solution Methodology," *Decision Sciences* Vol. 8, pp. 167–177.

Polya, G., 1947, *How to Solve It*, Princeton University Press, Princeton.

Polya, G., 1962, *Mathematical Discovery*, Vol. I, Wiley, New York.

Polya, G., 1965, *Mathematical Discovery*, Vol. II, Wiley, New York.

Rardin, R. L. and Lin, B. W., 1980, "The RIP Random Integer Programming Test Problem Generator," School of Industrial & Systems Engineering, Georgia Institute of Technology, June.

Rosen, J. B. and Suzuki, S., 1965, "Construction of Nonlinear Programming Test Problems," *Communications of the ACM* Vol. 8, No. 2, p. 113.

Silver, E. A., Vidal, R. V. V., and De Werra, D., 1980, "An Introduction to Heuristic Methods," *European Journal of Operational Research* Vol. 5, pp. 153–162.

Swart, W. W., 1967, "Optimal Value Estimation Applied to the Redistricting Problem," unpublished MSIE Thesis, Georgia Institute of Technology.

Tonge, F. M., 1961, "The Use of Heuristic Programming in Management Science," *Management Science* Vol. 7, pp. 231–237.

Woolsey, R. E. D. and Swanson, H. S., 1975a, *Operations Research for Immediate Application: A Quick and Dirty Manual*, Harper & Row, New York.

Woolsey, R. E. D., 1975b, "How to Do Integer Programming in the Real World," in *Integer Programming*, H. M. Salkin, ed., Addison-Wesley, Chap. 13.

Zanakis, S. H., 1977, "Heuristic 0-1 Linear Programming: An Experimental Comparison of Three Methods," *Management Science* Vol. 24, pp. 91–104.

Zanakis, S. H., 1979a, "A Simulation Study of Some Simple Estimators of the Three-Parameter Weibull Distribution," *Journal of Statistical Computation and Simulation* Vol. 9, pp. 101–116.

Zanakis, S. H., 1979b, "Extended Pattern Search with Transformations for the Three-Parameter Weibull MLE Problem," *Management Science* Vol. 25, pp. 1149–1161.

Zanakis, S. H., Austin, L., Nowading, D., and Silver, E., 1980, "From Teaching to Implementing Inventory Management: Problems of Translation," *Interfaces* Vol. 10, No. 6, pp. 103–110.

Zanakis, S. H., 1981, "A Large-Scale Integer Goal Programming Method with an Application to a Facility Location/Allocation Problem," *Multicriteria Decision Making*, in *Organizations: Multiple Agents with Multiple Criteria*, J. Morse, ed. Springer-Verlag.

Zanakis, S. H., and Gillenwater, J., 1981, Improved Heuristics for Integer and 0-1 Programming, Working Paper, West Virginia College of Graduate Studies.

APPENDIX

Let $Z_1 \leqslant Z_2 \leqslant \ldots \leqslant Z_n$ be an ordered random sample of iterative solutions (objective function values) to a large minimization problem. (For a maximization problem, simply replace Z_i by $-Z_{n+1-i}$, $i=1, 2, \ldots, n$.) A point estimate of the unknown true optimum can be obtained by one of the following simple estimators:

$$\tilde{Z} = 2Z_1 - Z_2 \tag{1}$$

used by Dannenbring [1977] and Golden [1977], or

$$\hat{Z} = (Z_1 \cdot Z_n - Z_2^2)/(Z_1 + Z_n - 2Z_2) \tag{2}$$

used by Zanakis [1979a] and Zanakis and Gillenwater [1981], or

$$Z^* = 2Z_1 - (e-1) \sum_{i=1}^{n} Z_i / e^i \tag{3}$$

proposed by Cooke [1979]. Note that these estimators do not require any restrictive assumption and apply to *any* distribution with a location or threshold parameter. From a storage requirement viewpoint their order of preference is $\tilde{Z}$, $\hat{Z}$, Z^* but their accuracy (MSE) is in the reverse order. This is expected because of the different number of observations considered by each estimator.

The following very simple and robust $100(1-e^{-n})$% confidence interval on the unknown true optimum value, $Z_{opt}=a$, was proposed by Golden and Alt [1979]:

$$Z_1 \leqslant a \leqslant Z_1 - \tilde{b}, \tag{4}$$

where

$$\tilde{b} = Z_{[0.63n]+1} - \tilde{a}. \tag{5}$$

Here, [ ] denotes rounding down and $\tilde{a} = \tilde{Z}$ ($\hat{Z}$ or Z^* will be more accurate). This requires the (empirically justified) assumption that the random sample came from a three-parameter Weibull cumulative distribution

$$F(Z) = 1 - \exp\{-[(Z-a)/b]^c\}, \qquad Z \geqslant a, \tag{6}$$

with parameters a (location), b (scale), and c (shape). Golden and Alt [1979] recommend refining this confidence interval by using maximum likelihood estimates (MLE). The added accuracy, however, will not justify the incremental effort because: (1) MLE's do not have a closed form, thus requiring an iterative procedure on a computer; (2) it will require the additional estimation of the shape parameter c, which is the most difficult of the three parameters to estimate; (3) estimator $\tilde{b}$ is a very good simple Weibull scale estimator, even better than MLE's when c is small [Zanakis, 1979a and 1979b]; and (4) using $\tilde{a} = Z^*$, or even $\hat{Z}$, instead of $\tilde{Z}$, will improve the above confidence interval. Therefore, the simple confidence interval (4) should suffice for most simple heuristic optimization methods.