ORSA Journal on Computing

Rejoinder—Anchoring and Cross-Fertilization

Christopher V. Jones,

ORSA Journal on Computing
Vol 6, No. 3, Summer 1994

0899-1499/94/0603-0278 $01.25

REJOINDER

Anchoring and Cross-Fertilization

CHRISTOPHER V. JONES / *Faculty of Business Administration, Simon Fraser University, Burnaby, BC V5A 1S6, Canada; Email: chris_jones@sfu.ca*

I would like to thank the commentators who produced such thought-provoking comments on the feature article. It is a pleasure to be able to respond.

In reading the commentaries, I was continually struck by two opposing themes. On the one hand, the commentators emphasize the need to use visualization to support how decision makers and practitioners currently do their work; on the other hand, they emphasize the need to learn from other areas that have experience in visualization. I call these two opposing ideas *anchoring* and *cross-fertilization.* I discuss these two themes in more detail in the next two sections.

1. Anchoring

Anchoring is a well-known phenomenon studied in the decision theory literature (Kahneman, Slovic, and Tversky[7]), wherein people's decisions can be influenced (anchored) by an initial value. In the visualization area, I am aware of few studies that have addressed to what extent people prefer the representations they learn first (with the notable exception of Ghani[4]). From reading the commentaries, as well as from personal experience, though, I am sure that the phenomenon is common.

For example, mathematical programmers are generally trained to formulate their problems using algebraic notation. Since mathematical programmers are already familiar with that representation style, it makes sense to develop tools based on algebraic notation. Many commercial mathematical programming languages now exist that support algebraic notation. Yet not all mathematical programmers are anchored to algebraic notation. As noted in the comments from Greenberg and Murphy, for example, the first commercial mathematical programming systems provided representations closer to the traditions of the first large-scale users, the process industry.

More general than the algebraic notation used for mathematical programming, symbolic mathematical notation has evolved over hundreds of years to provide a powerful, concise tool for mathematicians. Again, given the anchoring of mathematicians to symbolic notation, it makes sense to develop tools based on that notation. Mathematica® is an elegant example of such a system.

Such symbolic, algebraic representations enjoy a large and productive audience. Many people find such representations impenetrable, however. For example, many of my MBA students cringe at any use of algebraic notation. Most MBA students, though, are quite comfortable with very complex spreadsheets. In short, they are more tightly anchored to spreadsheets than to algebraic notation. Some MBA students take their anchoring to spreadsheets to the extreme; one successful MBA I taught did all his word processing using Lotus 1-2-3 because he claimed that word processors were "beyond him." Through spreadsheets, even with their flaws (as noted by Keiper and Wickham-Jones), we can hope to see our mathematical tools used by a larger audience than just those people who are comfortable with symbolic mathematical notation.

The comments by Bodin and Levy provide a welcome survey of how visualization has been used to support vehicle routing. The field of vehicle routing is (quite properly) anchored to a particular set of visualizations, often based on graphs. Other problem domains such as production scheduling and telecommunications are similarly anchored to their own style of representation. These areas have also devoted efforts to developing appropriate computer-based implementations.

Appropriate visualizations for other problem domains, as Bell noted in his comments, are less obvious. Bell uses the example of senior decision makers, in particular a vice president of marketing. What would be the appropriate representations for her? For certain people, there may be no obviously appropriate representations. Ideas for such representations might be found, though, in an Executive Information System (EIS) or Executive Support System (ESS) (Rockart and De Long[13]), which deliver information in a congenial way to high-level officers in a firm. Although OR models are rarely used in an EIS (they largely provide only data query capabilities), they might provide a useful entry point for delivering such models.

In short, effective representations for problem solving often come from the representations to which people are anchored. The idea is really rather simple: let people work with their problems in the way that they are used to.

2. Cross-Fertilization

Anchoring, however, also has a darker side. By sticking with the tried-and-true, one can become anchored to outmoded and unhealthy habits. Moreover, one may be blinded to foreign problem domains and areas that may have successfully attacked problems that still dog one's native area. I agree with Bell, who stated that developing new visualization techniques should be an important research area. Many of the "new" techniques may come from cross-fertilization with other areas.

As Greenberg and Murphy stated, though, the relevant "professional communities do not appear to communicate with each other." For example, the graph-based representations used for vehicle routing are probably very applicable to telecommunications problems, and vice versa. Many of the problems in trying to draw networks cited by Bodin and Levy as a continuing challenge to vehicle routing have been extensively studied in areas such as VLSI layout (Lengauer[8]) as well as computer graphics and human-computer interaction (Fairchild, Poltrock, and Furnas[2]; Ganser, North, and Vo[3]; Henry and Hudson[5]; Messigner, Rowe, and Henry[10]; Roberston, Mackinlay, and Card[12]; Rowe et al.[14]; Sarkar and Brown[15]; Sarkar et al.[16]; Schaffer et al.[17]; Shneiderman[18]; Tamassia, Di Battista, and Batini[19]; Vaucher[20]).

Similarly, "visualization in three dimensions"—the third dimension being time—was identified in Greenberg and Murphy's comments and Bodin and Levy's comments as an important area for research. Research on such visualizations is emerging from the computer graphics and human-computer interaction communities (Brown and Najork[1]; Mackinlay, Robertson, and Card[9]). This has taken the form of visualizations where time is represented through three-dimensional algorithm animation[1] as well as where time is represented as a spatial dimension.[9]

Cross-fertilization among different areas, as noted in the feature article, is enhanced by the increasingly easy ability to link together different computer systems. The MathLink feature of Mathematica, which I personally have used, represents a notable example. MathLink links Mathematica to and from spreadsheets and other systems. Traditional optimization software has yet to provide such capabilities.

One can argue that the feature article itself was overly anchored to the realm of optimization. Visualization techniques from other OR disciplines such as simulation, model management, stochastic analysis, decision theory, mathematics, and cognitive psychology, as noted by Greenberg and Murphy, have much to contribute. Not only optimization but the entire field of OR has much to learn from the computer graphics and human-computer interaction fields. These fields have developed techniques and ideas that can be put to work to help deliver OR technology, but more importantly, to help people solve their problems. The converse is also true, as witnessed by the increasing use of optimization techniques to produce visualizations.

3. Conclusions

OR, not just optimization, tries to solve difficult problems. It tries to shed light on impenetrable muddles. Yet the techniques used to penetrate the darkness paradoxically are often themselves impenetrable to the people who are faced with the problem to solve. By concentrating on how to deliver the technology effectively to both decision makers *and* technical specialists, in addition to developing fast, clever algorithms, one should be able to do a better job at solving real problems than using just the algorithms alone.

The field of OR is anchored in mathematics. That anchoring has produced marvelous, powerful, and effective techniques. But the field should not let that anchoring blind it from other techniques that can be used to help solve difficult problems. In fact, mathematics is increasingly embracing visualization as a useful tool (Horgan[6]).

Keiper and Wickham-Jones seem to imply that the field of OR will not lack prominence if it concentrates solely on developing clever algorithms. Perhaps, but the field of OR has always aspired to more. As discussed in a recent article by an OR pioneer, Hugh Miser,[11] the field originally sought to embrace any technique that could help shed some insight on a difficult problem. He laments the narrowing of the field to its current suite of techniques. I agree with Miser. Let's concentrate on the goal—to solve problems—and accept the premise that multiple techniques, many outside the existing traditions of OR, can be successfully used to attain the goal. Visualization is one of those techniques.

References

1. M.H. Brown and M.A. Najork, 1993. Algorithm Animation using 3D Interactive Graphics, *Proceedings of the ACM Symposium on User Interface Software and Technology*, ACM Press, New York, 93–100.
2. K.M. Fairchild, S.E. Poltrock and G.W. Furnas, 1988. SemNet: Three-Dimensional Graphic Representations of Large Knowledge Bases, in *Cognitive Science and its Application for Human Computer Interface*, R. Guindon (ed.), Elsevier, Amsterdam, 201–233.
3. E.R. Ganser, S.C. North and K.P. Vo, 1988. DAG—A Program that Draws Directed Graphs, *Software-Practice and Experience 18:11*, 1047–1062.
4. J. Ghani, 1981. The Effects of Information Representation and Modification on Decision Performance, Unpublished Ph.D. Dissertation, Decision Sciences Working Paper 81-08-01, The Wharton School, The University of Pennsylvania, Philadelphia, PA.
5. T.R. Henry and S.E. Hudson, 1991. Interactive Graph Layout, *Proceedings of the ACM Symposium on User Interface Software and Technology*, ACM Press, New York, 55–64.
6. J. Horgan, 1993. The Death of Proof, *Scientific American 269:4*, 92–103.
7. D. Kahneman, P. Slovic and A. Tversky (eds.), 1982. *Judgement Under Uncertainty: Heuristics and Biases*. Cambridge University Press, Cambridge, UK.
8. T. Lengauer, 1990. *Combinatorial Algorithms for Integrated Circuit Layout*, Wiley, New York.
9. J.D. Mackinlay, G.G. Robertson and S.K. Card, 1991. The Perspective Wall: Detail and Context Smoothly Integrated, in *Proceedings ACM Computer-Human Interaction '91 Conference on Human Factors in Computing Systems*, S.P. Robertson, G.M. Olson and J.S. Olson (eds.), ACM Press, New York, 173–179.
10. E.B. Messinger, L.A. Rowe and R.R. Henry, 1991. A Divide-and-Conquer Algorithm for the Automatic Layout of Large Directed Graphs, *IEEE Transactions on Systems, Man and Cybernetics 21:1*, 1–11.
11. H.J. Miser, 1993. The Need for Inclusion, *OR/MS Today 20:6*, 38–42.
12. G.G. Robertson, J.D. Mackinlay and S.K. Card, 1991. Cone Trees: Animated 3D Visualizations of Hierarchical Information, *Proceedings ACM Computer-Human Interaction '91 Conference on Human Factors in Computing Systems*, S.P. Robertson, G.M. Olson and J.S. Olson (eds.), ACM Press, New York, 189–194.
13. J.F. Rockart and D.W. De Long, 1988. *Executive Support Systems*, Dow Jones-Irwin, Homewood, IL.
14. L.A. Rowe, M Davis, E. Messinger, C. Meyer, C. Spirakis and A. Tuan, 1987. A Browser for Directed Graphs, *Software-Practice and Experience 17:1*, 61–76.
15. M. Sarkar and M.H. Brown, 1992. Graphical Fisheye Views of Graphs, in *Proceedings ACM Computer-Human Interaction '92*

Conference on Human Factors in Computer Systems, ACM Press, New York, 83–91.

16. M. SARKAR, S.S. SNIBBE, O.J. TVERSKY and S.P. REISS, 1993. Stretching the Rubber Sheet: A Metaphor for Viewing Large Layouts on Small Screens, in *Proceedings of the ACM Symposium on User Interface Software and Technology*, ACM Press, New York, 81–92.
17. D. SCHAFFER, Z. ZUO, L. BARTRAM, J. DILL, S. DUBS, S. GREENBERG and M. ROSEMAN, 1993. Comparing Fisheye and Full-Zoom Techniques for Navigation of Hierarchically Clustered Networks, *Proceedings of Graphics Interface '93*, ACM Press, New York, 87–96.
18. B. SHNEIDERMAN, 1992. Tree Visualization with Tree Maps: 2-D Space-Filling Approach, *ACM Transactions on Graphics 11:1*, 92–99.
19. R. TAMASSIA, G. DI BATTISTA and C. BATINI, 1988. Automatic Graph Drawing and Readability of Diagrams, *IEEE Transactions on Systems, Man and Cybernetics 18:1*, 61–79.
20. J.G. VAUCHER, 1980. Pretty-Printing of Trees, *Software-Practice and Experience 10*, 553–561.