



## INFORMS Journal on Computing

Publication details, including instructions for authors and subscription information:  
<http://pubsonline.informs.org>

### Note from the Editor

Alice E. Smith

To cite this article:

Alice E. Smith (2025) Note from the Editor. INFORMS Journal on Computing 37(6):iii-iv. <https://doi.org/10.1287/ijoc.2025.ed.v37.n6>

Full terms and conditions of use: <https://pubsonline.informs.org/Publications/Librarians-Portal/PubsOnLine-Terms-and-Conditions>

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact [permissions@informs.org](mailto:permissions@informs.org).

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2025, INFORMS

Please scroll down for article—it is on subsequent pages



With 12,500 members from nearly 90 countries, INFORMS is the largest international association of operations research (O.R.) and analytics professionals and students. INFORMS provides unique networking and learning opportunities for individual professionals, and organizations of all types and sizes, to better understand and use O.R. and analytics tools and methods to transform strategic visions and achieve better outcomes.

For more information on INFORMS, its publications, membership, or meetings visit <http://www.informs.org>

## Note from the Editor

Alice E. Smith<sup>a</sup>

<sup>a</sup>Department of Industrial and Systems Engineering, Auburn University, Auburn, Alabama 36849

Contact: [smithae@auburn.edu](mailto:smithae@auburn.edu),  <https://orcid.org/0000-0001-8808-0663> (AES)

---

<https://doi.org/10.1287/ijoc.2025.ed.v37.n6>

Copyright: © 2025 INFORMS

---

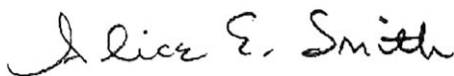
### Introduction

A reminder to you all that our new technical section, Quantum Computing, is fully implemented and open for business. Please help us publicize this widely by sharing with your networks and colleagues. We aim to be the premier publication venue of research emerging at the intersection of quantum computing and operations research.

I am so pleased to continue to announce the winners of *INFORMS Journal on Computing* (IJOC) Test of Time Paper Award to cover the backlog of awards since the journal's inception. The energetic and able committee, chaired by John Chinneck with members Bill Cook, Bruce Golden, Karla Hoffman, and David Woodruff, has selected the awardee covering the period 2006–2010. What follows is the citation from the award committee and then a reflection from the author concerning his paper.

I want to thank the committee for their superb efforts and am very pleased to share this recognition of the impactful heritage of our journal.

All my best,



The Test of Time Award for papers published in the *INFORMS Journal on Computing* in the years 2006–2010 is awarded to

### An Evolutionary Algorithm for Polishing Mixed Integer Programming Solutions

Edward Rothberg

*INFORMS Journal on Computing* 19(4):534–541, Fall 2007

<https://pubsonline.informs.org/doi/10.1287/ijoc.1060.0189>

### Test of Time Award Citation 2006–2010

This paper provides an elegant and computationally efficient method for applying evolutionary algorithms to general mixed integer linear optimization problems (MIPs). It integrates the ideas of reproduction, crossover, and mutation into general MIP methodology. The novelty is that it chooses parents by collecting feasible solutions found during the solution process, ranks the parents based on solution quality, and then applies the evolutionary properties of mating and mutation to determine the next generation of solutions. The technique requires no information about the underlying structure of the problem and is, therefore, applicable to any MIP problem. The impact that this paper has had on MIP solution technology through its implementation in both commercial and open-source solvers is enormous. The paper continues to be highly cited.

Below is the retrospective from the author, Edward Rothberg, concerning his landmark paper.

### Reflections on “An Evolutionary Algorithm for Polishing Mixed Integer Programming Solutions” by Edward Rothberg (2007)

I am honored that my paper, “An Evolutionary Algorithm for Polishing Mixed Integer Programming Solutions,” was selected for the IJOC Test of Time Paper Award. I would like to thank IJOC and the award committee for their commendable work on shining a light on papers from the past.

My research for the selected paper came out of my efforts to expand the capabilities of the heuristics in CPLEX back in the mid-2000s. Users had made it quite clear that optimality proofs were nice when you could get them but that it was often necessary to stop the MIP search well before such proofs were available. Exploring the branch-and-bound tree eventually produces high-quality solutions, but good heuristics were typically essential for finding them early. It was not unusual for heuristics to find extremely high-quality solutions to difficult MIP models at the root node.

In light of this, one significant thrust of MIP research efforts at the time was to consider heuristics that performed substantial computation. Although people were hesitant to expend too much computational effort on a single heuristic invocation, especially because most did not actually produce an improved solution and thus, represented wasted effort, the benefit of obtaining better solutions earlier in the search, even if it slowed the optimality proof, could not be ignored. A few notable examples of methods that came out of this work were local branching (Fischetti and Lodi 2003), the feasibility pump (Fischetti et al. 2005), the objective feasibility pump (Achterberg and Berthold 2007), and relaxation-induced neighborhood search (RINS) (Danna et al. 2005). The common theme among these was that they solve either a series of LP models or an MIP model to explore a neighborhood of an LP relaxation. RINS eventually emerged as the most effective heuristic in CPLEX.

As MIP solvers got better exploring neighborhoods of the relaxation, it quickly became apparent that relaxations were not always appropriate for this purpose, either because they took too long to compute or because they did not identify productive neighborhoods to explore. My favorite example is MIPLIB model t1717—a modest-sized set partitioning model with 551 constraints and 16,428 binary variables. The best-known solution to this problem chooses 92 variables of the 16,428 to set to one. In our efforts to explore a branch-and-bound tree to guide the search, we found that 31 of those 92 variables took a value of zero in every one of the first 100,000 node relaxation solutions. A heuristic can often recover a good solution from a starting point that differs in a handful of critical variables, but a starting point that differs in more than a third is unlikely to guide even the most effective heuristic to the desired solution.

That observation led me to consider what sort of neighborhood search might be effective in situations where the relaxation did not provide useful guidance. A combination of three key ingredients led me to the scheme described in this paper. The first came out of our work on RINS, where we realized that the combinatorial explosion that results from growth in the size of an MIP model also had an upside. If the difficulty of the problem grows dramatically as a result of small increases in the problem size, then by contrast, the difficulty diminishes greatly with a small decrease. The practical impact is that you can typically solve hundreds or even thousands of slightly simplified MIP models in the time that it takes to solve the original problem. That fact is what made RINS practical.

The second ingredient came from a talk that I heard by Paul Shaw of ILOG about his large neighborhood search technique in constraint programming. Constraint programming does not have relaxation, which gave me optimism that a method that was effective in this context was likely to also be effective for the problem that I was trying to solve.

The third ingredient was an investigation of metaheuristics and evolutionary algorithms. RINS had a scheme for diversifying the search baked in; one of the two inputs to an invocation of RINS was the current node relaxation solution, so if you invoke it at different nodes in the branch-and-bound tree, you get different neighborhoods to explore. I often wondered if that was enough, however, because a branch in the search tree only changes a single variable bound. I had been exploring the possibility of using a more aggressive diversification scheme, possibly drawn from the metaheuristic literature. That never panned out for RINS, but the general strategy proved useful in this context.

After combining these three ingredients with a bit of engineering, the result was a strategy called solution polishing, where tens of thousands of randomly selected neighborhoods of the current incumbent were explored. Although the strategy did not pay off for most problems, that was more of a comment on the quality of the relaxation neighborhoods for most problems than it was a comment on the effectiveness of the new method. The primary impact of solution polishing was that it enabled an MIP solver to find high-quality solutions to significant classes of models that were often unapproachable with traditional tree search.

## References

- Achterberg T, Berthold T (2007) Improving the feasibility pump. *Discrete Optim.* 4(1):77–86.  
Danna E, Rothberg EE, Pape CL (2005) Exploring relaxation induced neighborhoods to improve MIP solutions. *Math. Programming* 102:71–90.  
Fischetti M, Lodi A (2003) Local branching. *Math. Programming* 98:23–47.  
Fischetti M, Glover FW, Lodi A (2005) The feasibility pump. *Math. Programming* 104:91–104.  
Rothberg E (2007) An evolutionary algorithm for polishing mixed integer programming solutions. *INFORMS J. Comput.* 19(4):534–541.