Nº75

OPTIMA

Mathematical Programming Society Newsletter

DECEMBER 2007

# Rapid Development of an Open-source Minlp Solver with COIN-OR

**Pierre Bonami, John J. Forrest, Jon Lee and Andreas Wächter**

## Abstract

We describe the rapid development of Bonmin (Basic Open-source Nonlinear Mixed INteger programming) using the community and components of COIN-OR (COmputational INfrastructure for Operations Research).

## 1. Introduction

As famous examples like Linux have proven, open-source communities can produce rapidly-developing high-quality software, often competitive with expensive commercial solutions. COIN-OR (COmputational INfrastructure for Operations Research), originally founded by IBM in 2000 and a non-profit foundation since 2004, provides such a platform for us optimization folks.

Since its conception, a number of optimization-algorithm implementations and related software projects have been contributed to COIN-OR. From a web server hosted by INFORMS, at the URL www.COIN-OR.org, COIN-OR offers online repositories for sharing source code, web sites with information, documentation and bug tracking systems, as well as mailing lists for users and developers. Many of the optimization codes have been used extensively by academics and in commercial products.

Currently, COIN-OR hosts many optimization-related codes:

- Clp, DyLP, and Vol are solvers for Linear Programming
- Bcp, Cbc, Cgl, Cops, and Symphony are solvers, frameworks and tools for (serial and parallel) Mixed-Integer Linear Programming,
- Csdp, Dfo, Ipopt, LaGO, and Svm-QP are solvers for different types of Nonlinear Optimization (semidefinite, derivative-free, general large-scale interior point, global optimization, QP solver for support vector machines)
- CoinMP, FlopC++, GAMSlinks, NLPAPI, Osi, OS and Smi are problem modeling and interface tools.

Even more, COIN-OR has become the platform that facilitates the collaboration of optimization researchers to start entirely new projects, exploring and developing new algorithmic ideas, by providing both a sound software basis that can be re-used, as well as the technical forum to coordinate the effort.

In order to clarify a common misconception, we want to emphasize that the term "open source" does not simply refer to software that is made available to some users in source-code format. According to the definition put forth by the non-profit corporation Open Source Initiative (OSI) [26], a number of conditions must be met for a software license to be an "open-source" license, including the right for free distribution and usage (including for commercial purposes), as well as the right to modify the code. There are many OSI-certified open-source licenses. The one preferred on COIN-OR is the CPL (Common Public License [12]), which was chosen for its very liberal terms.

In this article, we tell the on-going success story of one such project. In 2004, IBM and Carnegie Mellon University initiated a joint study aimed at Mixed Integer Nonlinear Programming (MINLP), with the goal of releasing the resulting software under the CPL on COIN-OR. In 2006, within a new Open Collaborative Research (OCR) initiative at IBM, this relationship with CMU was extended. This OCR program builds on the Open Collaboration Principles established in 2005 by IBM and eleven other institutions to accelerate innovation and foster open-source research.

Besides the joint activity of CMU and IBM, the subject of MINLP has received a lot of recent attention. For example,

- The 2006 Beale-Orchard Hays Prize for Excellence in Computational Mathematical Programming was awarded by the Mathematical Programming Society to Nick Sahinidis and Mohit Tawarmalani for their work on global optimization which culminated in their software Baron [3, 27]; global optimization is one of the key approaches to MINLP, and Baron can be directly applied to MINLP problems.
- The 2006 CORE Lecture Series consisted of a series of lectures by Robert Weismantel on MINLP.

Our goal was to develop algorithms and software for this important class of problems. We undertook to carry this out in the context of COIN-OR. Our original goal was to use existing software components of COIN-OR to produce a simple NLP-based branch-and-bound code aimed at MINLP problems that have convex relaxations. As we progressed, we expanded our goal and developed instantiations of several MINLP methods, aimed mostly at problems having convex relaxations. We were able to carry this out rather quickly, as we had at our disposal the source code to manage a branch-and-bound tree and to solve MILP and NLP subproblems.

In Section 2, we describe the setting of MINLP. In Section 3, we discuss the COIN-OR NLP solver Ipopt and the COIN-OR MILP solver Cbc, which were the starting points of our implementation. In Section 4, we describe the capabilities of our MINLP solver Bonmin (Basic Open-source Nonlinear Mixed INteger programming) and how we constructed it using other components of COIN-OR. Finally, in Section 5, we discuss ongoing and future plans for Bonmin.

## 2. MINLP

We consider MINLP problems of the form

$$\min \quad f(x)$$
$$g(x) \le 0,$$
$$x \in \mathbb{R}^{n-k} \times \mathbb{Z}^{k},$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}$ and $g : \mathbb{R}^n \mapsto \mathbb{R}^m$ are twice continuously differentiable.

MINLP is the *mother of all deterministic optimization problems*. Many business and engineering optimization problems are naturally formulated as mathematical programs involving both discrete and continuous variables, as often decisions have to be made that involve making discrete decisions (e.g., yes/no) and setting continuous levels (e.g., flowrates). Natural phenomena often involve relationships between variables which, while often smooth, may be rather nonlinear. Accurate models need to allow for all of these.

A common approach to such problems is to reformulate them as Mixed Integer Linear Programming (MILP) problems, usually through the use of piecewise-linear functions approximating the true functions. In the presence of non-convexity, this leads to the introduction of additional binary variables or non-standard branching rules. The inaccuracies that are introduced and the added computational effort induced by accommodating the piecewise-linear functions may make the value of such a solution approach rather limited.

Another approach is to relax the MINLP to an NLP, apply an NLP solver, and then use some heuristic method to reach a nearby feasible point that respects the integrality restrictions. Just as for MILP, this technique is not very robust, often leading to very poor solutions.

The alternative to methods that are based strictly on either MILP and NLP is to develop methods that are directly aimed at MINLP formulations. In effect, one shifts some of the burden from the modeler to the solution software. Still, just as we know from MILP and NLP, there are always significant issues that cannot be ignored by a modeler who seeks to solve difficult instances.

It is useful to first consider methods that are aimed at MINLP problems having convex relaxations. In this sub-domain, there are natural methods that generalize MILP methods (e.g., branch-and-bound, using NLP subproblems) as well as more sophisticated methods like outer approximation (see [13]), generalized Bender's decomposition (see [18]) and hybrid methods (see [8], for example). Even in this sub-domain, problems can be much harder than the MILP counterparts. A main reason is that in the space of the natural variables, the solution of a relaxation may be well inside the interior of the feasible region, and so the cutting planes that have been so successful for MILP may have very limited value (see [24], for example).

If the relaxation of the MINLP problem is not convex, we are really in a different world. There are several approaches. There are ad hoc heuristics, which suffer from being well suited to only very specific models (see [23], for example). There is the possibility of taking methods designed for problems having convex relaxations and heuristically adapting them to problems that do not satisfy this restriction (e.g., the branch-and-bound option of Bonmin; see [7]). There is the approach of global optimization typified by the method of spatial branch-and-bound wherein one repeatedly subdivides the feasible region, with the goal of producing better and better convex under-estimators (e.g., Baron; see [28]). Finally, there is the approach of systematically remodeling a broad subclass of MINLP problems as MILPs using specialized techniques (e.g., see [16] for such a method using linked ordered sets)

## 3. Building Blocks

As MINLP combines the worlds of MILP and of NLP, a logical start for the Bonmin project was to look at the existing codes and libraries in COIN-OR related to MILP and NLP and to see how they could be integrated.

Cbc (Coin Branch and Cut) is a powerful, industrial-strength branch-and-cut code for MILP problems. It is a component of COIN-OR, and it uses several other components of COIN-OR. It uses Clp (the Coin Linear Programming solver) through the Osi (Open Solver Interface) library. The Osi is a layer used to insulate an application that makes use of an LP solver from the specific LP solver employed. Clp includes primal and dual simplex solvers as well as an interior-point method. As a full-fledged branch-and-cut code, Cbc strengthens relaxations through use of the COIN Cgl (Cut Generation Library), accommodates different node-selection and branching rules (e.g., via pseudocosts and strong branching), and allows for SOS Type-1 and Type-2 modeling and branching. Cgl includes a

*Date: November 2, 2007.*

wide variety of cutting-plane generators (including: knapsack cover, odd hole, flow cover, lift and project, Gomory, reduce and split, and MIR). Our interest in Cbc was twofold. We sought to re-use infrastructure that we would need (for example, maintaining any needed branch-and-bound enumeration trees, handling the branching and fathoming decisions, etc.), but we also simply wanted the Cbc functionality for solving any MILP subproblems that we might create.

To address the nonlinear aspect of the problem, the Ipopt NLP solver from COIN-OR was used. Ipopt implements an algorithm for continuous nonlinear optimization (see [29]). Its interior point approach, together with a third-party sparse linear algebra routine such as MA27 [20], allows Ipopt to efficiently solve large-scale problems with up to millions of variables. Global convergence (i.e.,convergence to a stationary point from any starting point) is achieved by a filter line-search procedure. Here, nonconvexities in the problem formulation are tackled by possibly regularizing the Hessian matrix of the Lagrangian to ensure that the search directions are descent directions in the appropriate sense, so that convergence to non-minimizers is less likely. Since Ipopt is using second derivatives, fast local convergence can be achieved. However, if second derivatives are not available, they are approximated by a quasi-Newton approach. The Ipopt code was originally written in Fortran, but a complete rewrite in C++, following an object-oriented approach, was released on COIN-OR in 2004. The Ipopt library includes a few utility functionalities, such as output logging and algorithmic options handling, which can be used by other codes as well.

All of these basic components from COIN are being actively developed and have a wide user basis.

## 4. Bonmin

Using the building blocks described in Section 3, we began to develop Bonmin (Basic open-source nonlinear mixed integer programming). In the first phase, we sought to implement an MINLP code targeted at problems with convex relaxations. A straightforward approach employing available components from COIN-OR uses the Cbc framework to organize a branch-and-bound search, where now all the nodes are NLPs. The nonlinear relaxations are solved by Ipopt, which has been connected to Cbc via an Osi solver object. Attempts were also made to handle nonconvex problem instances by some heuristics, such as negative cut-off, multiple local optimization runs for the node NLPs from randomly chosen starting points, and continuing branching even at (locally) infeasible nodes. The integration of these features, which helped to obtain better solutions, would not have been possible if the source code of the components had not have been available.

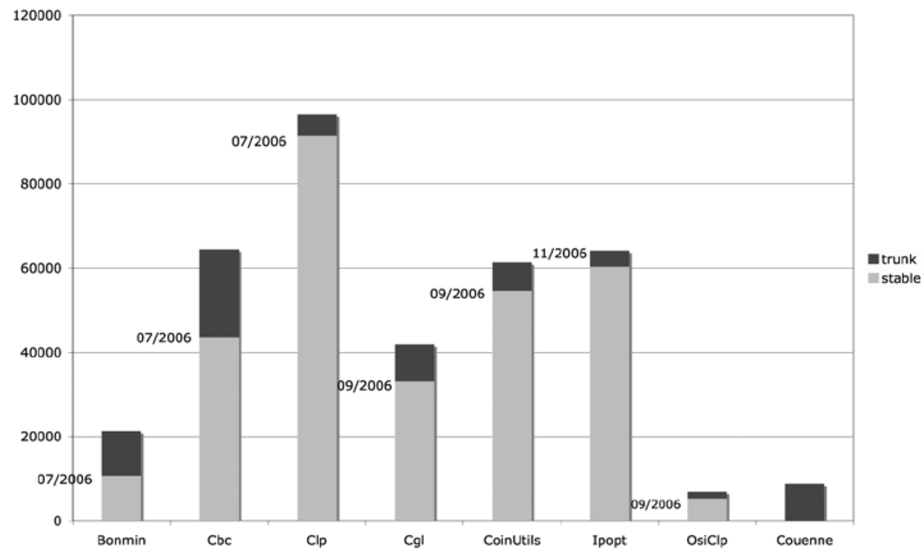

Figure 1. Code re-use by Bonmin and growth

Another popular strategy for solving convex MINLPs uses Outer Approximation (OA). The basic idea is to approximate the original nonlinear problem by a sequence of MILPs, where, successively, linear cuts are added based on linearizations of the nonlinear functions. The OA algorithm alternatingly solves the current MILP approximation and an NLP (the MINLP with all integer variables fixed). It can be shown that this procedure with appropriate linearization points converges to the optimal solution of the MINLP in a finite number of iterations (see [13]). Using Cbc with Clp as the MILP solver and Ipopt as the NLP solver, we could implement the OA algorithm in Bonmin with existing COIN-OR components.

An intermediate hybrid approach between a pure branch-and-bound and a pure OA algorithm has also been developed and implemented in Bonmin [8].

The success of the project is facilitated by the fact that all essential components were available as open-source code, so that they could be changed (and changes could be contributed back to the project maintainers), and by the object-oriented design of the codes. Figure 1 indicates how Bonmin makes significant re-use of COIN-OR components. The ordinate indicates the number of lines of code for the various components. In the totals, we do not count parts of some codes that we definitively do not use. The bar for Bonmin refers to the new code written specifically for the MINLP project. Besides demonstrating how Bonmin is re-using COIN-OR components, Figure 1 also indicates how all of these components (including Bonmin) are growing. For each component, we indicate the number of (relevant) lines of code as of the most-recent stable release on COIN-OR, as well as the current incremental size (as of 21 May, 2007) in the \trunk“ (i.e., development branch) of each project.

Bonmin can be accessed as a C++ library, as a stand-alone code that can accept an input file in .nl format [17] from the mathematical-programming modeling and scripting language Ampl [1], and via the NEOS Server for Optimization [25]. Similar to Ampl, Gams is another widely used scripting and modeling language for manipulating mathematical programs. The recently added COIN-OR project, GAMSlinks, managed by Stefan Vigerske, provides an interface to Bonmin (as well as the COIN- OR solvers Ipopt and Cbc).

## 5. Ongoing and future work

We are presently pursuing several initiatives with the goal of extending the applicability of our work.

One ongoing project involves using the parallel MILP framework of the COIN-OR code Bcp (Branch-Cut-and Price) to develop a parallel version of Bonmin. Our goal is to exploit parallelism on a small scale as well as on massively parallel architectures (like BlueGene; see [5]). Alternatively, we might consider use of the relatively new COIN-OR Cops (COIN-OR Open Parallel Search) framework, to manage the parallelization. The use of object-oriented design for the basic Bonmin code facilitates the easy adaption into different search tree management frameworks (e.g., Cbc, Bcp, Cops).

Another project jointly explored with Carnegie Mellon University, involves the use of global-optimization ideas and software. We are experimenting with the usage of rigorous convex under-estimators and spatial decomposition. The resulting software will be available on COIN-OR as Couenne (Convex Over/Under ENvelopes for Nonlinear Estimation). Similarly, we are exploring the possibility to re-use some globalization cuts from the recent COIN addition LaGO (Lagrangian Global Optimizer), authored by Ivo Nowak and Stefan Vigerske. The LaGO code itself is a MINLP solver that makes use of Clp, Cgl, and Ipopt, and is based on some heuristics.

However, for many MINLP instances, it is impractical to expect to be able to reliably find the global optimum. So, an extremely important area is the development of general-purpose heuristics for MINLP. Already, in the context of the development of the Bonmin project, some success in this direction has been achieved (see [6]). Still, there is much more work to do in this direction.

We are also investigating the use of other NLP solvers (e.g., active-set approaches like that of FilterSQP [14]) as an alternative to or in conjunction with Ipopt.

On the theoretical side, we are exploring the use of curvature information from the NLP in making branching decisions in the branch-and-bound MINLP frame-work. We are already finding this approach to be useful for certain nonlinear facility-location problems as well as certain portfolio-optimization problems.

We are also investigating the strengthening of MINLP formulations, following the success of such methods for MILP. Already we have some success for structured problems (see [19]) and we are currently exploring more generic techniques.

Finally, there are other components of COIN-OR that we may consider taking advantage of. For example, Csdp (a semidefinite-programming code authored by Brian Borchers) and Dfo (a derivative-free NLP code authored by Katya Scheinberg) could be exploited in an attempt to solve broader classes of MINLP problems than we currently consider.

It is our fondest desire that this entire manuscript be already obsolete by the time that you are reading this. Indeed, for our project to be succeeding, that should be the case. The very latest information concerning our efforts should be available from the Bonmin web page [9] and the IBM MINLP web page [21].

## 6. Acknowledgments

Thanks are due to the others who have participated in the CMU/IBM collaboration, namely Pietro Belotti, Larry Biegler, Andrew Conn, Gérard Cornuéjols, Ignacio Grossman, Carl Laird, Andrea Lodi, Françcois Margot and Nick Sawaya.

## References

1. Ampl, www.ampl.com
2. Ampl Solver Library, www.netlib.org/ampl/solvers
3. Baron, archimedes.scs.uiuc.edu/baron.html
4. Blas, www.netlib.org/blas
5. BlueGene, IBM Journal of Research and Development, Volume 49, Number 2/3, 2005. www.research.ibm.com/journal/rd49-23.html
6. P. Bonami, G. Cornuéjols, A. Lodi and F. Margot. A Feasibility Pump for Mixed Integer Nonlinear Programs. IBM Research Report RC23862, 2006.
7. P. Bonami and J. Lee, BONMIN Users' Manual. projects.COIN-OR.org/Bonmin
8. P. Bonami, A. Wächter, L.T. Biegler, A.R. Conn, G. Cornuéjols, I.E. Grossmann, C.D. Laird, J. Lee, A. Lodi, F. Margot and N. Sawaya, An algorithmic framework for convex mixed integer nonlinear programs. IBM Research Report RC23771, October 2005. To appear in: Discrete Optimization.
9. Bonmin, projects.COIN-OR.org/Bonmin
10. C. Bragalli, C. D'Ambrosio, J. Lee, A. Lodi and P. Toth, An MINLP model and solution method for a water network optimization problem. In Algorithms - ESA 2006, Y. Azar and T. Erlebach, Eds., pages 696-707. Springer, 2006.
11. COIN-OR, www.COIN-OR.org
12. Common Public License, www.opensource.org/licenses/cpl1.0.php
13. M. Duran and I.E. Grossmann. An outer-approximation algorithm for a class of mixed-integer nonlinear programs. Mathematical Programming, 36:307{339, 1986.
14. FilterSQP, www-unix.mcs.anl.gov/~leyffer/solvers.html
15. C.A. Floudas, Deterministic global optimization, Kluwer Acad. Publ., Dordrecht, 2000.
16. J. Forrest and J. Lee (in preparation).
17. D.M. Gay, Writing .nl files, Technical Report 2005-7907P, Sandia National Laboratories, Albuquerque, NM, 2005. www.cs.sandia.gov/~dmgay/nlwrite.pdf
18. A.M Geoffrion. Generalized Benders decomposition. Journal of Optimization Theory and Applications, 10:237{260, 1972.
19. O. Günlük, J. Lee and R. Weismantel. MINLP Strengthening for Separable Convex Quadratic Transportation-Cost UFL, IBM Research Report RC24213, March 2007.
20. Harwell Subroutine Library, www.cse.clrc.ac.uk/nag/hsl
21. IBM MINLP, domino.research.ibm.com/comm/research projects.nsf/pages/minlp.index.html
22. Lapack, www.netlib.org/lapack
23. J. Lee, In situ column generation for a cutting-stock problem. Computers & Operations Research, 34(8):2345-2358, 2007.
24. J. Lee, Mixed integer nonlinear programming: Some modeling and solution issues, to appear in the IBM Journal of Research and Development, 2007.
25. NEOS Server for Optimization, www-neos.mcs.anl.gov
26. Open Source Initiative, www.opensource.org
27. N. Sahinidis and M. Tawarmalani, A polyhedral branch-and-cut approach to global optimization, Mathematical Programming, 103(2), pp. 225-249, 2005.
28. M. Tawarmalani and N.V. Sahinidis, Convexification ▶

# Optimization Research Symposium Recognizing Professor George Nemhauser's Contributions to the Field of Operations Research

July 27, 2007 marked the 70th birthday of George Nemhauser. George has been a long-time member of MPS, served as its president from 1989-1992 and co-chaired the ISMP at Georgia Tech in 2000. To mark the occasion of George's birthday, a two-day symposium was held in his honor at the Georgia Institute of Technology in Atlanta, GA. The symposium included a mix of serious scientific lectures and, sometimes less serious presentations on George's career and his contributions to various fields and institutions. There were 17 invited speakers and close to 100 attendees. The symposium was organized by Mike Ball and Martin Savelsbergh.

The Thursday afternoon program included one session on George's scientific contributions to four disciplines: integer programming, presented by Bob Bixby, combinatorial optimization presented by Gerard Cornuéjols, airline optimization presented by David Ryan and sports scheduling presented by Mike Trick. A second session covered George's contributions to four institutions: Johns Hopkins University presented by Mike Thomas, Cornell University presented by Dave Goldsman, CORE presented by Lawrence Wolsey and Georgia Tech presented by John Jarvis. There was a special bonus presentation by Martin Grötschel and the evening dinner included remarks by Bill Pulleyblank, Don Ratliff and Mike Ball.

Several presentations included gifts such as a selection of red and white New Zealand wines by Ryan and Trick and the official naming of a prime number in George's honor by Groetschel. While Gerard Cornuejols presented George with one of his integer programming homework assignments from Cornell, it did come with the request that George regrade a problem he had incorrectly graded 30 years ago. Although Bob Bixby demonstrated that certain mixed integer cuts developed by Nemhauser and his co-authors were the basis for a substantial speed improvement in Cplex, no associated stock options were provided. The institutional presentations chronicled George's significant contributions to four great institutions and also demonstrated George's ability both to solve and create problems. The dinner presentations were lighter in tone and ended with a presentation of a compilation of the title pages and table of contents from the 50 PhD dissertations supervised by George.

The Friday program included five scientific presentations: "Retail Assortment Optimization" by Marshall Fisher, "Airline Optimization: If we are so good at it, why are things so bad?" by Cindy Barnhart, "Corner Polyhedra and Two-Equation Cutting Planes" by Ralph Gomory, "Minimizing with Concave Costs?" by Tom Magnanti, and "Integer Programming and Branch-Width" by Bill Cunningham. Each of the speakers has had close ties to George over the years but more significantly the body of results presented had numerous connections to his work.

Overall the symposium proved to be both intellectually stimulating and, at times, highly entertaining. By examining the contributions of a great individual in this way, one can gain an appreciation for the substantial positive impact one can have through a career in mathematical programming. The symposium was certainly a fitting tribute to George Nemhauser for providing such a positive role model to our community.

Michael O. Ball
*Orkand Corporation Professor of Management Science*
*Robert H. Smith School of Business and Institute for Systems Research*
University of Maryland
College Park, MD 20742

and global optimization in continuous and mixed-integer nonlinear programming, Kluwer Acad. Publ., Dordrecht, 2002.

29. A. Wächter and L. T. Biegler, On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming, Mathematical Programming 106(1), pp. 25-57, 2006.

IBM T.J. Watson Research Center,
Yorktown Heights, New York. USA
e-mail address: fpbonami,jjforre,jonlee,andreasw@us.ibm.com

# MPS Chair's Optima Column – 3rd issue of 2007

By Steve Wright

I am delighted to contribute the inaugural MPS Chair's column for Optima. As you can see, Optima is being published on a regular schedule once again. Under the leadership of editor Andrea Lodi, the design of our Society's newsletter is being reworked, and we look forward to exciting innovations in the months ahead.

MPS has recently had a large increase in membership due to the 2006 ISMP in Rio de Janeiro and the 2007 ICCOPT in Hamilton. Renewal notices for 2008 have recently gone out. I hope that our new members will stay involved in the Society and help it play an ever more important role in sustaining and building our international community of optimizers and mathematical programmers.

Preparations for the 2009 ISMP in Chicago gathering momentum under general chair John Birge. The symposium will be held in the week of August 23-28, 2009, with most sessions taking place at the Gleacher Center (the downtown campus of the University of Chicago Business School) and the nearby Marriot Hotel. Both locations are close to Millennium Park, around which the recent revival of downtown Chicago has been centered. The meeting will include a commemoration of the 60th anniversary of the "zero-th" ISMP, which was held in Chicago in 1949. I hope that all of you will be able to take advantage of Chicago's pleasant summer weather and its location as a global air hub to attend a wonderful symposium and witness the exciting recent changes to this great American city.

We welcome new editors of our Society's journal Mathematical Programming: Kurt Anstreicher (Series A) and Danny Ralph (Series B). The number of published issues will increase from nine to ten in 2008 and later years, a reflection of the robust health of the journal. As always, subscription to the print version of the journal is included in MPS membership. The online version, along with complete archives of the journal and Mathematical Programming Studies, is also free to members. We also welcome Philippe Toint, the new editor of the MPS-SIAM book series. Please contact Philippe or one of his editorial board members with any book-writing ideas you may have.

I take this opportunity to remind you of the generous discounts available to MPS members on all books published by Springer and SIAM. A new SpringerToken will be issued in 2008; look for it in your email inbox.

Another snippet of news is that MPS recently became a member of the International Council of Industrial and Applied Mathematics (ICIAM), a society of societies whose main function is to organize a congress every four years. I was privileged to attend the most recent congress in Zurich along with 3100 colleagues. The next congress is planned for Vancouver in 2011. MPS members have already played key roles in the last two congresses, as organizers and speakers, and our new status will formalize and solidify our involvement.

I wish to thank the organizers of several recent meetings sponsored by MPS in 2007. IPCO continued its tradition as a key meeting of the integer programming and combinatorial optimization community with a highly successful meeting in June at Cornell. Thanks to David Williamson and Matteo Fischetti and all involved with the organization and selection of papers for this event. ICCOPT at McMaster University in August was an outstanding success, under general chair Tamas Terlaky and his large and excellent cast of organizers. The 420 attendees enjoyed good weather, an excellent venue and program, and a wonderful roster of social events.

Finally, I thank the officers and councilors of MPS whose terms have ended recently. Former Chair Rolf Möhring continues as Vice-Chair, so we will continue to have the benefit of his wisdom and experience. Karen Aardal is stepping down as Chair of the Executive Committee. Karen has served MPS tirelessly for over ten years as Councillor and Publications Committee Chair, in addition to her most recent role. Bill Cook did great work as editor of Mathematical Programming Series B, then Series A, maintaining the high standards of our journal over a 7-year span. Special thanks go to these three colleagues!

## Application for Membership

*I wish to enroll as a member of the Society.*

My subscription is for my personal use and not for the benefit of any library or institution.

☐ I will pay my membership dues on receipt of your invoice.

☐ I wish to pay by credit card (Master/Euro or Visa).

CREDIT CARD NO. ____________________ EXPIRATION DATE ____________________

FAMILY NAME ____________________

MAILING ADDRESS ____________________

____________________

TELEPHONE NO. ____________________ TELEFAX NO. ____________________

E-MAIL ____________________

SIGNATURE ▶ ____________________

*Mail to:*

Mathematical Programming Society
3600 University City Sciences Center
Philadelphia, PA 19104-2688 USA

Cheques or money orders should be made payable to The Mathematical Programming Society, Inc. Dues for 2007, including subscription to the journal *Mathematical Programming*, are US 80. Retired are 50. Student applications: Dues are 25. Have a faculty member verify your student status and send application with dues to above address.

Faculty verifying status ____________________

Institution ____________________

# Prizes and Conferences

by Katya Scheinberg

The Second Mathematical Programming Society International Conference on Continuous Optimization (ICCOPT II) was held at McMaster University on August 13-16, 2007. Even though only the second of its kind this conference was a smashing success. Instead of writing about it ourselves, we let the professional speak for us.

Here is a excerpt from the Hamilton Spectator:
“There is an endless number of optimization applications”, Terlaky says. Specialists develop algorithms and use computer networks to simulate, model and optimize solutions to various problems, as well as design new products and services. “Mostly, you don't see it because it's behind the scenes“, Terlaky says. “But without it, everything would be more expensive and less efficient.“ It all has to do with a field of research called continuous optimization, or more formally, mathematical programming — finding the most efficient and effective means to an end. And big players like NASA, Boeing, General Motors, which used to have their own mathematics department, Microsoft and Exxon Mobil rely on it to improve their products or in designing new ones. The companies have representatives in Hamilton this week to attend a major international conference on optimization at McMaster. The conference has also attracted NASA representatives, PhD students, professors and researchers from around the world, representing Ivy League institutions such as Princeton, Oxford, Stanford and MIT. About 450 specialists are attending from 36 countries, including New Zealand, Singapore, Brazil and Chile.

The Ivy Leagueness of Oxford, Stanford and MIT is debatable, but everything else is true. From ourselves we can add that with 22 streams on numerous subjects there was never a dull moment at this well attended conference. The quality of talks and attendance was high and discussions lively. One of the highlights of the conference was the Young Researchers Competition in continuous optimization. The committee consisted of Kees Roos (chair), Arnold Neumaier, Levent Tuncel, Yinyu Ye and Akiko Yoshise.

Out of 23 submissions three finalists were selected:
Alexandre Belloni (Duke University),
Mung Chiang (Princeton University)
Eissa Nematollahi (McMaster University)
who presented their work in a special session during the conference. The prize was awarded to Alexandre Belloni for his paper “Norm-induced Densities and Testing the Boundedness of a Convex Set”.

Here is how Kees Roos presented the prize at the conference banquet:
Ludwig Wittgenstein, the most famous logician and philosopher of language, was awarded a PhD at the university of Cambridge (UK), in 1929. As his PhD thesis served the small booklet ’ ‘Tractatus logico-philosophicus’, a work that had been in print already for seven years and that was regarded at that time by many as a philosophical classic. The examiners were George E. Moore and Bertrand Russell. The PhD defense was set for 18 June, 1929, and was conducted with an air of farcical ritual. As Russell walked into the examination room with Moore, he smiled and said: ‘I have never known anything so absurd in my life’. The examination began with a chat between old friends. Then Russell, relishing the absurdity of the situation, said to Moore: ‘Go on, you’ve got to ask him some questions – you’re the professor’. Then followed a short discussion in which Russell advanced his view that Wittgenstein was inconstant in claiming to have expressed unassailable truths by means of meaningless propositions. He was, of course, unable to convince Wittgenstein, who brought the proceeding to an end by clapping each of his examiners on the shoulder and remarking consolingly: ‘Don’t worry, I know you’ll never understand it’.

In his examiner’s report, Moore stated: ‘It is my personal opinion that Mr. Wittgenstein’s thesis is a work of a genius, but, be that as it may, it is certainly well up to the standard for the Cambridge degree of Doctor of Philosophy’. (See Ray Monk, Ludwig Wittgenstein. The duty of genius. Penguin Books, 1990).

The last sentence, slightly modified, well expresses the committee’s feelings on the participants in this year’s Young Researcher Competition. Most of the 23 submissions were of high quality. Also, the submissions of the three finalists, Alex Belloni, Mung Chiang and Eissa Nematollahi, who presented their papers this afternoon, where convincing enough for us to agree that they are up to the standard in the optimization community for being finalists in the competition. But our committee also agreed unanimously that the work of Belloni certainly meets the standard for being the winner of the competition.

# Uneasy Relations by Michael Bartholomew-Biggs: a Book Review

by Andrea Lodi

Uneasy Relations contains poems - mainly in haiku form - which play with ideas from computational mathematics. Non-specialists may choose to murmur them as zen-like meditations. For readers of a curious disposition, notes are provided that may be no less informative than those at the end of The Waste Land.

Our favorite?

**AUXILIARIES**

*Lagrange Multipliers*

These are our shadows,
dual personalities,
who know what we don't —
the consequences
of pushing our boundaries
or pulling them in.

Michael Bartholomew-Biggs has spent most of his working life as a mathematician at the University of Hertfordshire. As the author himself says, his poetic ambitions arrived as part of a mid-life crisis and this, his fourth poetry chapbook, is an attempt to unite both halves of his brain.

Michael Bartholomew-Biggs, Uneasy Relations, Hearing Eye London (UK), 2007. ISBN: 978-1-905082-31-5.

O P T I M A

MATHEMATICAL PROGRAMMING SOCIETY

UF UNIVERSITY of FLORIDA

Center for Applied Optimization
401 Weil
PO Box 116595
Gainesville, FL 32611-6595 USA

FIRST CLASS MAIL

EDITOR:
Andrea Lodi
DEIS University of Bologna,
Viale Risorgimento 2,
I - 40136 Bologna, Italy
e-mail: andrea.lodi@unibo.it

CO-EDITORS:
Alberto Caprara
DEIS University of Bologna,
Viale Risorgimento 2,
I - 40136 Bologna, Italy
e-mail: acaprara@deis.unibo.it

Katya Scheinberg
IBM T.J. Watson Research Center
PO Box 218
Yorktown Heights, NY 10598, USA
katyas@us.ibm.com

FOUNDING EDITOR:
Donald W. Hearn

PUBLISHED BY THE
MATHEMATICAL PROGRAMMING SOCIETY &
University of Florida

***Journal contents are subject to change by the publisher.***