Christopher Maes

cmmaes@gmail.com
(617) 233-1075

Objective

To apply my experience in numerical optimization and linear algebra to interesting applications.

Education

Stanford University Stanford, CA

Degree: Ph.D. Computational and Mathematical Engineering *January 2011*

A Regularized Active-set Method for Sparse Convex Quadratic Programming

Developed QPBLUR, a quadratic programming solver with block-LU updates and regularization. QPBLUR is used as a subproblem solver in SNOPT, a sequential quadratic programming algorithm for constrained nonlinear optimization.

http://www.stanford.edu/group/SOL/dissertations/maes-thesis.pdf

Advisor: *Michael Saunders*

Massachusetts Institute of Technology Cambridge, MA

Degree: S.B. Applied Mathematics *June 2005*

Work Experience

Gurobi Optimization, Inc. Boston, MA

Senior Software Developer *January 2012 - January 2016*

Developed a MIP heuristic, sparse LU factorization, automatic tuning tool, and MATLAB interface that appeared in versions 5.0 through 6.5 of the Gurobi Optimizer. Updated LP presolve and achieved a 5% performance improvement in the fastest LP solver in the world. Lead the development and launch of the Gurobi Instant Cloud product.

MIT Operations Research Center Cambridge, MA

Postdoctoral Associate *October 2010 - December 2011*

Developed a large-scale implementation of an MIP model for routing air traffic in the presence of convective weather. Designed algorithms for computing adaptive policies for two-stage linear programming under uncertainty.

Stanford University Stanford, CA

Research Assistant *January 2008 - August 2010*

Worked with an interdisciplinary team of mathematicians and systems biologists to devise an optimization-based model of the metabolic system of unicellular organisms. The team received a three-year grant from the Department of Energy to explore the model's use in biological hydrogen production.

Google Summer of Code: Scilab Consortium Palo Alto, CA

Open-source developer *June 2009 - August 2009*

Developed SPARTAN, a sparse trust-region algorithm for nonlinear equations for use in the open-source Matlab clone Scilab. SPARTAN is designed to solve large systems of nonlinear equations with sparse derivatives.

https://github.com/cmaes/spartan

Stanford University Stanford, CA

Research Assistant *September 2008 - January 2008*

Modified PDCO, a primal-dual interior method for convex objectives, to handle nonseparable objective functions, and use sparse symmetric quasi-definite factors of the KKT system to compute search directions.

`http://www.stanford.edu/group/SOL/software/pdco/`

Electric Power Research Institute Palo Alto, CA

Research Assistant *June 2007 - November 2007*

Formulated and solved global optimization problems with bilinear objectives and equilibrium constraints to compute optimal bidding strategies in an electrical power market.

Wolfram Research Boston, MA

Research Fellow *July 2006 - September 2006*

Constructed numerical analysis demonstrations in Mathematica. Developed a statistics module for Wolfram Alpha.

Wolfram Research Champaign, IL

Intern *June 2004 - August 2004*

Implemented an add-on package to Mathematica for nonuniform B-Spline interpolation.

Teaching Experience

Parallel Methods in Numerical Analysis: *Prof. Juan Alonso*	*Spring 2009*
Introduction to Large-scale Computing in Engineering: *Prof. James Lambers*	*Winter 2008*
Numerical Linear Algebra: *Prof. Gene Golub*	*Fall 2007*
Constructing Scientific Simulation Codes: *Patrick Miller*	*Spring 2007*

Publications

[1] R. M. T. Fleming, C. M. Maes, M. A. Saunders, Y. Ye, B. Ø. Plasson. A variational principle for computing nonequilibrium fluxes and potentials in genome-scale biochemical networks. *J. Theoret. Biol.* 292, 71–77, 2012.

[2] C. M. Maes. Optimal Bidding in Power Markets: Test for Bilinear Bilevel Programs. Program on Technology Innovation 1016228, `http://www.epriweb.com/public/000000000001016228.pdf` EPRI, Electricité de France, 2007.

[3] C. M. Maes. Reversal Addition Maps. *MIT Undergraduate Journal of Mathematics*, 7:101–112, 2005.

Programming Languages

C, Python, Javascript, Matlab, Mathematica, OCAML, Fortran 95, CUDA (NVIDIA's GPU language), Haskell, Perl, Scheme, Intel 8051 Assembly, R.