Interval Methods for Accelerated Global Search in the Microsoft Excel Solver

Ivo P. Nenov and Daniel H. Fylstra
Frontline Systems Inc.

This paper describes advanced interval methods for finding a verified global optimum and finding all solutions of a system of nonlinear equations, as implemented in the Premium Solver Platform, an extension of the Solver bundled with Microsoft Excel. It also describes the underlying tools that allow Excel spreadsheets to be evaluated over reals and intervals, with fast computation of real gradients and interval gradients. The advanced interval methods described include mean value (MV) and generalized interval (GI) representations for functions, constraint propagation for both the MV and GI forms, and a linear programming test for the GI form, in the context of an overall interval branch and bound algorithm. Numerical results for a set of sample problems demonstrate a significant speed advantage for the GI techniques, compared to alternative methods.

1 Introduction

The Solver bundled with Microsoft Excel, developed by Frontline Systems for Microsoft, is among the most widely used tools for optimization and equation solving. It is capable of solving small-scale linear programming (LP), smooth nonlinear programming (NLP), and mixed integer programming (MIP) problems. Included in nearly 100 million copies of Microsoft Excel, it offers Excel spreadsheet users an easy introduction to classical methods of optimization. An upgraded Premium Solver for Education, now bundled with more than a dozen textbooks, is used in a wide range of MBA and engineering courses.

The Premium Solver Platform is a compatible upgrade that extends the functionality, capacity and speed of the Microsoft Excel Solver to handle industrial-scale problems, including LP problems of over 100,000 variables and constraints; NLP problems with tens of thousands of variables and constraints; challenging mixed-integer problems; global optimization problems using multi-start or clustering methods; and non-smooth problems using methods based on genetic and evolutionary algorithms and tabu search.

In the past two years, we have sought to greatly extend the capabilities of the Premium Solver Platform for verified global optimization and solution of systems of equations, using interval methods. Since Microsoft Excel is designed to evaluate spreadsheet models only over real numbers, we built a new parser and interpreter for Excel formulas that can evaluate models over several domains including intervals, as outlined below. We then implemented and tested a variety of interval-based techniques for global optimization and solution of systems of equations, including techniques implemented in other interval solvers and some new techniques implemented for the first time, to our knowledge, in the Premium Solver Platform. Numerical results suggest that these new techniques offer significant speed advantages over previously described methods.

2 Parser/Interpreter and Automatic Differentiation

Our parser and interpreter for Excel formulas take the place of the standard facility in Microsoft Excel for function evaluation or "recalculation." These tools can be used to evaluate Excel formulas over real and interval numbers; real gradients and interval gradients, using the techniques of automatic differentiation; and a special "diagnostic number type" that identifies sparsity in models to save memory, and classifies models, functions, and individual variable occurrences as linear, smooth nonlinear, or non-smooth.

Since Microsoft Excel supports a wide range of arithmetic (and non-arithmetic) operators and several hundred built-in functions, and we wished to evaluate all of these functions over several domains, we broke down the task by (i) defining a small number of basic functions, and implementing all other functions in terms of these, and (ii) making extensive use of operator overloading in the C++ programming language. Operator overloading allows us to define composite functions that can be evaluated over reals and intervals, real and interval gradients, G-intervals, and the "diagnostic number type."

We have implemented both forward and reverse mode automatic differentiation for both real gradients and interval gradients, using certain memory efficient techniques.

3 Interval Methods for Global Optimization and Equation Solving

We describe a framework for finding the global optimum of a constrained optimization problem, and finding all solutions of a system of equations. We begin with the basic interval branch and bound algorithm, which processes a list of "boxes" and seeks to reduce their size, splitting them as necessary, until a list of sufficiently small boxes enclosing the solution(s) is obtained.

We *represent the function* to be minimized and the equality and inequality constraints using the natural interval extension form, the mean value (MV) form (a quadratic outer approximation), and the generalized interval (GI) or linear enclosure form. The MV and GI forms are effectively first-order methods that require more computation than the natural interval extension, but they permit more rapid reduction of box sizes and elimination of boxes in the overall branch and bound algorithm.

We then describe *constraint propagation techniques* as a way of speeding up the process of box size reduction. These techniques can be applied to single equations (or equality constraints) at a time, and hence require less memory and computation than the general MV and GI box reduction techniques.

Finally, we describe a *linear programming test* that can be applied to the GI form to rapidly eliminate a box where the (dual) Simplex method finds no feasible solutions for the linear enclosures of the equality constraints.

4 Numerical Results and Conclusions

We present numerical results for a few test problems, including an electronic circuit analysis problem and a problem used by van Hentenryck to illustrate performance of the Numerica system. We report CPU time and the number of function evaluations, gradient evaluations, and box splits for three interval method implementations: using MV form techniques, using GI form techniques, and adding a LP test to the second method. The GI form techniques including the LP test dramatically outperform the natural interval extension and the MV form techniques in all reported cases, with several-fold reductions in CPU time, function evaluations, and gradient evaluations.

We conclude that the interval methods presented for global optimization and equation solving show promise for practical problems, especially where a verified rather than an approximate or merely "good enough" solution is desired, and that the availability of these methods in an easy-to-use, highly accessible form in Microsoft Excel should result in more widespread use and appreciation for the power of these methods.