SIAG/OPT Views-and-News

A Forum for the SIAM Activity Group on Optimization

Volume 14 Number 1 April 2003

Contents

Large Scale Nonconvex Optimization

Introduction by the Guest Editors

This issue of SIAG/OPT Views-and-News is devoted to Large Scale Nonconvex Optimization. There has been remarkable progress in nonlinear optimization algorithms and software in the last few years. The supremacy of SQP and augmented Lagrangian methods has been challenged by interior methods, and the competition between these approaches has revitalized the area. The article by Gould gives a perspective of the current state of nonlinear programming algorithms and raises a variety of questions that require investigation.

There has been much work in extending the areas of application of nonlinear programming. The articles by Sachs, and Biegler and Waechter discuss the formulation and solution of optimization problems whose constraints are differential (or differential-algebraic) equations. This area, which has been studied since the 1970s, is now drawing much attention thanks to the robustness, flexibility and efficiency of differential equation solvers, and the development of a wide range of practical applications.

General-purpose nonlinear programming algorithms are having a profound effect in the area of mathematical programming with complementarity constraints (MPCC). This area has been growing in importance in the last 10 years with the development of its theoretical foundations and the recognition that many economic and engineering applications can be formulated as MPCCs. The survey by Leyffer provides an overview of recent developments in this field.

Bussieck and Pruessner discuss one of the most challenging problems in nonlinear optimization, namely the solution of problems involving both discrete and continuous variables. Problems of this type arise naturally in many engineering and management applications. They combine the difficulties of nonlinear optimization with the combinatorial nature of integer optimization. The authors argue that integer nonlinear optimization is a wide open field

with opportunities for novel approaches.

We hope that the five papers contained in this volume encourage researchers working in nonlinear optimization to branch out into some of these new emerging areas.

Sven Leyffer and Jorge Nocedal March 10, 2003

Some Reflections on the Current State of Active-Set and Interior-Point Methods for Constrained Optimization

Nicholas I. M. Gould
Computational Science & Engineering Department,
Rutherford Appleton Laboratory, Chilton, Didcot,
Oxfordshire, UK, OX11 0QX.

1. Introduction

This is an exciting time to be working in constrained nonlinear optimization. New ideas abound. Collaborations and alliances are forged, rivalry is intense, competition fierce. Why should this be? After all, surely the importance of optimization was recognised many decades ago. So why, now, should there be so much activity and why did it take so long?

I believe that the answer is complicated, but certainly one of the main reasons is that, finally, we really are starting to believe that we have the right (theoretical and practical) tools to tackle the problems we have long been asked to solve. What was the stimulus for this? Well, without doubt in part what has been called the "interior-point" revolution. But also the fight-back from the traditionalists, those who promote earlier "active-set" approaches. And finally, the recognition from practitioners that, yes indeed, we can now solve sizable nonlinear programming problems, so that there has been a shift away from linear models and the thinking that lead to these.

In this short article, I hope to explain the salient points of both approaches, the symbiosis that has arisen, and how both approaches have impacted on nonlinear optimization. But I also want to look to the future, and to see how things may develop.

2. History

2.1 Active-set methods

In the beginning, there was linear programming: as simple an approximation to the real world as one could possibly make, but nonetheless one of the most important (and most studied) problems in the history of computational mathematics. As we all know, linear programming is concerned with (say) minimizing a linear function of n unknown parameters (variables) over a feasible region described by m linear equations and/or inequalities. A solution will (almost always) occur at a vertex of the feasible region, and the archetypical *active-set* solution algorithm, the Simplex method, aims to find such a solution by moving through a sequence of objective-improving, feasible, adjacent vertices. Thus, the search is to determine which of the constraints "define" the solution (the *active* ones), and which may be safely discarded, and this defining characteristic extends easily to more general constrained optimization problems. Such an algorithm may explore an exponential (in terms of $m - n$) number of active "sets" in the worst case, is known to depend linearly on these parameters on the average, and in practice really seems to behave just as its average-case performance predicts. Thus for a problem involving, say, a million degrees of freedom, it is reasonable to expect a few millions iterations. While this might at first sound impractical, it is vital to recognise that for linear constraints the dominant cost per iteration is usually the solution of a system of linear equations, and that each system is a rank-one modification of its predecessor. Thus the cost per iteration is often very small, and it is this feature that has kept the Simplex method for linear programming competitive over the past 50 years. Most commercial linear programming systems (such as CPLEX, Xpress and OSL) still have Simplex components (albeit with numerous enhancements such as advanced crash and pre-solve procedures, steepest edge exchange rules, and hyper-sparsity exploiting linear solvers, etc.), and the active-set paradigm also extends into the nonlinear world by virtue of successful and widely-used packages like MINOS [25] and SNOPT [14]. Our experience is that to build a successful active-set method requires considerable care, since rounding errors have ample opportunities to build up and

cause havoc over the large number of iterations that occur, even in the linear and quadratic programming cases.

2.2 Interior-point methods

Knowing that the Simplex method might take an exponential number of steps started a race to find alternatives whose worst-case complexity was polynomial in $m-n$. The first-reported polynomial algorithm, the ellipsoid method, has alas not turned out to be effective in practice [1]. Fortunately the next competitor, Karmarkar's method [21], proved to be a major advance, and started a frenetic research feeding-frenzy on interior-point methods which has continued to this day. Karmarkar's genius was to produce a *nonlinear* iteration that attempted to stay well away from the boundary of the feasible region (and thus avoid the influence of the myriad of vertices) until it approached optimality. It was soon recognised that the method (and many of its successors) may be interpreted as the approximate minimization of a sequence of logarithmic barrier functions—or, if we prefer, as following the "central path" defined as the trajectory of such minimizers as a function of the barrier parameter—and these perspectives have obvious and important consequences for its use for nonlinear problems.

The current state for linear (and many convex) problems is primal-dual variants (in which, as the name suggest, duality plays a strong role and primal and dual variables are treated equally) whose worst-case behaviour to achieve a close-to optimal solution varies like $O(\sqrt{m-n})$ in theory and significantly better (perhaps $O(\log(m-n))$?) in practice [26, 38]. All of the major commercial systems contain interior-point solvers (again with a large number of enhancements). It is interesting to note that although such methods require considerably fewer iterations than their active-set rivals, the cost per iteration is significantly higher—there is, in general, no rank-one update for the crucial linear systems—so that there is still fierce and healthy competition between the two competing ideologies. Certainly, active-set interior-point hybrids are now popular and successful. It remains to be seen that, if in the long term as problem sizes grow, the superior complexity bounds for interior-point methods proves decisive, but I believe this will be the case.

3. Where are we now?

Thus far, all seems perfect. But how do these ideas extend into the nonlinear, nonconvex world?

3.1 The trouble with SQP ...

Extending active-set methods would at a first glance appear to be easy, simply replacing the solution over the whole feasible set by that over a sequence of active sets in which the inactive inequalities are discarded. However, the resulting subproblems are still nonlinear, and thus in principle will each require an infinite iteration. Early attempts to "truncate" such subproblems suffered from a nasty phenomenon known as zig-zagging in which constraints continually entered and left the active set.

A more successful idea is to replace the general problem by a sequence of "simple", tractable approximations. For instance, one might replace the objective and constraints by linear approximations (the so-called Successive Linear Programming or SLP approach [11]) or perhaps the objective by a quadratic approximation (the Successive Quadratic Programming or SQP approach [2, 19]). The advantage here is that the subproblem (a linear or quadratic program) is significantly easier to solve than the nonlinear approximation of the previous paragraph. Indeed if the quadratic approximation is convex (or a linear approximation used), we have polynomial-time subproblem-solution methods at our disposal. Having solved the subproblem, we can use its solution as the next trial iterate, and we might embed such a scheme within a linesearch, a trust-region or a filter globalization scheme. But caution is needed here, since there are a number of potential pitfalls.

Firstly, it is well known that the globalization scheme may interfere catastrophically with the SLP/SQP step (the Maratos effect) and avoiding action may result in extra computation [2, 19].

Secondly, to obtain fast ultimate convergence, it is usually vitally important to use some 2nd derivative information/approximation (and thus ultimately some form of SQP iteration). If we are "lucky" enough to have (and use) exact 2nd derivatives, the resulting nonconvex QP may have a num-

ber of local minimizers, some of which may not be consistent with our overall globalization scheme (the SQP step may be "uphill"). Although many active-set QP solvers can ensure that the step is downhill, I do not currently know how to guarantee this for interior-point QP solvers. If we must resort to "approximate" (say secant-approximation) 2nd derivatives, it is known that requirements of sparsity and positive-definiteness together conflict with stability [28], so we may be restricted to dense updates, and thus problems with few degrees of freedom—it is worth noticing that all of the successful SQP and SQP-like methods we are aware of (such as MINOS, SNOPT and filterSQP[10]) rely on having relatively few degrees of freedom.

Thirdly, if there is one lesson we should have learned from large-scale unconstrained minimization, it is to aim to solve the subproblem as *inaccurately* as possible consistent with overall convergence—the truncated Newton approach [8], along with its practical manifestation via the linear (preconditioned) conjugate-gradient method, is one of the key ideas to have evolved in the unconstrained case during the 20th century. So it is clearly desirable to truncate the LP/QP solution process. But how? We are aware of almost no work in this area (but see [24] for an exception), and it is of vital practical importance. Again, it would seem easier to stop "early" with an active-set QP solver than with an interior-point one.

Finally, we would ultimately expect that the active sets for our LP/QP subproblems will settle down as we approach the solution to the overall problem [27]. This suggests that we should be exploiting *a priori* information about candidate active sets to warm start subsequent subproblem solves. This would seem to be one area in which active-set methods have a clear edge, since the ability to warm start interior-point methods is in its infancy—there has been some work in the LP case [15, 40], but to our knowledge none for QPs. In practice, by contrast, we have observed that it is still sometimes faster (especially in the degenerate case) to "cold-start" an interior-point QP than "warm start" active set QP code, simply because even slightly incorrect active set predictions can have dramatic undesirable consequences for active-set methods [20].

We have currently suspended development of the large-scale SQP method that we had intended including in GALAHAD [18] despite having produced both effective active-set and interior-point QP solvers. Our experience has been that without QP truncation, the cost of the QP solution so dominates that other non-SQP approaches (such as IPOPT [33], KNITRO [4] and LOQO [32]), in which truncation is possible, have made significant progress even before our QP code had solved its first subproblem!—see also [23] for further evidence that interior-point methods appear to scale better than SQP ones. We are more enthusiastic about an SLP-QP approach we are currently developing [3], since LP truncation is in principle easier and since the QP phase is restricted to a problem with equality constraints for which a truncated conjugate-gradient iteration is possible.

3.2 Whither interior-point methods . . . ?

As I mentioned above, we produced two (nonconvex) quadratic programming packages for GALAHAD. Considerable numerical experience has indicated to us that the interior-point version, `QPB` is almost always vastly superior for large problems [7]. Since we have now all but given up our SQP developments, we have now turned to what we consider to be the other possibility, namely to solve general constrained optimization problems by sequential barrier-function minimization, using the lessons learned when designing and evaluating `QPB`.

We were warned as children that barrier-function methods are beastly because of the effects the barrier has close to the boundary. It later turned out that these fears were almost groundless, and that actually observed inefficiencies were to a large degree due to using the wrong dual variable updates following a reduction in the barrier parameter [36]. Without doubt, the problem does become very poorly conditioned near the solution, but this itself does not cause failure since even search direction calculations that might result in large numerical errors do not, because all such errors are confined to uninteresting subspaces [16, 37, 39]. But being prematurely close to the boundary certainly is bad in that it can be painfully slow to escape. For example, if we wish to minimize $-x$ for $x \in [0, 1]$ and start with x_0 very close to zero, the Newton barrier correction (with

modest barrier parameter) results in a new point $x_1 \approx 2x_0$. Thus an initial point $x_0 = 2^{-40} \approx 10^{-12}$ will take roughly 40 iterations to move to the centre of the feasible region. The lesson here is, I believe, to stay away from the boundary unless there are good reasons to get close (such as if a particular constraint is active at optimality). I strongly believe that it pays to stay close to "the" central path since this avoids as best as possible premature contact with the boundary, although since different scalings result in different central paths, it is far from obvious which path is actually the one to follow!

An important question if we are to use an interior-point approach is how we should handle equality constraints. To a certain extent, I suggest this depends on quite what sort of constraints they are. If they (or some of them) are linear, I believe that it often pays to use a "phase-1" procedure to find a "well-centred" feasible point for these constraints, and thereafter to ensure that they remain satisfied on all subsequent iterations. The reasoning is simply that dealing with a nonlinear objective is tricky enough without having to cope with non-convexity in subspaces that the constraints rule out—we certainly have seen the advantages even in the "simple" case of nonconvex quadratic programming of using feasible rather than infeasible interior point methods.

Nonlinear equality constraints are altogether more tricky, and it is in this area that the most significant differences between competing modern interior point methods occur. Some methods (like IPOPT, KNITRO, LOQO and INTOPT [22]) prefer to treat them explicitly by gradually moving towards feasibility but balancing the requirements of optimality using a penalty function or filter. Others like [29] and the method we are developing for GALAHAD replace equalities by one-sided inequalities (which are then handled using interior-point technology) and other-sided penalization. At this stage I do not think we know which of this approaches is best, but it is likely that actually there is very little difference.

There are two important side issues here, though. Firstly, if we really believe we have good methods for handling equations, is it better to treat inequalities by converting them to equations using slack variables and then simply treat the slack variables using interior-point methods? From a linear-algebraic perspective there is little difference, but there seem to be ardent devotees of both schools of thought [6], so I do not really believe we have exhausted or settled this question. Secondly, if we plan to use equality constraints explicitly, it is vital that there is some coherence between the search direction employed and the merit function used to ensure their ultimate satisfaction. Several cautionary examples [5, 34] attest to the pitfalls that may befall the unwary.

The asymptotic behaviour of interior-point methods is relatively well understood even in the non-convex case, at least under non-degeneracy assumptions: the barrier parameter may be reduced at a superlinear rate so that the overall iteration converges superlinearly for primal-dual methods [17] and 2-step superlinearly for primal-only methods [9], although the latter requires some care when reducing the barrier parameter. Some progress has been made in the degenerate case, but we do not currently have as complete an understanding as in the linear programming case where degeneracy does not hinder convergence to a well-defined point in the centre of the face of optimally active constraints. In practice, asymptotic convergence behaviour appears to behave just as one would hope from the linear programming experience.

So what are the outstanding issues? The effects of constraint scaling, and just how one might re-scale to improve convergence are not well understood. Just as importantly, since as we have hinted we really wish to truncate the calculation of the Newton-barrier search direction, we need to discover how to precondition the conjugate-gradient scheme that we will undoubtedly use; it is already clear that any preconditioner has to reflect the dominant barrier terms in the Hessian matrix of the barrier function, but just how much more is needed is unknown. Finally, another area where there is room for improvement is in extrapolation for better points on the central path. This has proved to be most useful in the linear programming case, but things are certainly more complicated for nonlinear problems because of possible bizarre behaviour [13] of the central path (multiplicity, bifurcation, and even non-existence).

4. Conclusions

I hope I have persuaded the reader that these are indeed exciting times in nonlinear optimization. With interior-point and (to some extent) active set approaches we now have a realistic chance of solving very large nonlinear programming problems. Of course there are difficulties, but the ingenuity and vigour with which the research community is currently addressing such challenges makes me extremely optimistic about future progress. Even in the last few months we have heard of a number of new and interesting proposals [12, 29, 35, 30, 31], and we eagerly await to see how these complement the already large corpus of algorithms and software.

REFERENCES

[1] R. G. Bland, D. Goldfarb, and M.J. Todd. The ellipsoid method: A survey. *Operations Research*, 29:1039–1091, 1981.

[2] P. T. Boggs and J. W. Tolle. Sequential quadratic programming. *Acta Numerica*, 4:1–51, 1995.

[3] R. H. Byrd, N. I. M. Gould, J. Nocedal, and R. A. Waltz. An active set algorithm for nonlinear programming using linear programming and equality constrained subproblems. Technical Report RAL-TR-2002-032, Rutherford Appleton Laboratory, Chilton, Oxfordshire, England, 2002.

[4] R. H. Byrd, M. E. Hribar, and J. Nocedal. An interior point algorithm for large scale nonlinear programming. *SIAM Journal on Optimization*, 9(4):877–900, 2000.

[5] R. H. Byrd, M. Marazzi, and J. Nocedal. On the convergence of Newton iterations to non-stationary points. Technical Report 2001/7, Optimization Technology Center, Northwestern University, Evanston, Illinois, USA, 2001. To appear, Math. Programming.

[6] R. H. Byrd, J. Nocedal, and R. A. Waltz. Feasible interior methods using slacks for nonlinear optimization. Technical Report 11, Optimization Technology Center, Argonnne National Laboratory, Argonne, Illinois, USA, 2000. To appear, Comp. Optim. & Apps.

[7] A. R. Conn, N. I. M. Gould, D. Orban, and Ph. L. Toint. A primal-dual trust-region algorithm for nonconvex nonlinear programming. *Mathematical Programming*, 87(2):215–249, 2000.

[8] R. S. Dembo and T. Steihaug. Truncated-Newton algorithms for large-scale unconstrained optimization. *Mathematical Programming*, 26(2):190–212, 1983.

[9] J.-P. Dussault. Numerical stability and efficiency of penalty algorithms. *SIAM Journal on Numerical Analysis*, 32(1):296–317, 1995.

[10] R. Fletcher and S. Leyffer. Nonlinear programming without a penalty function. *Mathematical Programming*, 91(2):239–269, 2002.

[11] R. Fletcher and E. Sainz de la Maza. Nonlinear programming and nonsmooth optimization by successive linear programming. *Mathematical Programming*, 43(3):235–256, 1989.

[12] E. M. Gertz and P. E. Gill. A primal-dual trust region algorithm for nonlinear optimization. Optimization Technical Report 02-09, Computer Science Department, University of Wisconsin, Madison, USA, 2002.

[13] J. Ch. Gilbert, C.C. Gonzaga, and E. Karas. Examples of ill-behaved central paths in convex optimization. Technical Report 4179, INRIA, Rocquencourt, France, 2001. To appear, Math. Programming.

[14] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Journal on Optimization*, 12(4):979–1006, 2002.

[15] J. Gondzio and A. Grothey. Re-optimization with the primal-dual interior point method. Technical Report MS-01-004, Department of Mathematics and Statistics, University of Edinburgh, Scotland, 2001. To appear, SIAM J. Optimization.

[16] N. I. M. Gould. On the accurate determination of search directions for simple differentiable penalty functions. *IMA Journal of Numerical Analysis*, 6:357–372, 1986.

[17] N. I. M. Gould, D. Orban, A. Sartenaer, and Ph. L. Toint. Superlinear convergence of primal-dual interior point algorithms for nonlinear programming. *SIAM Journal on Optimization*, 11(4):974–1002, 2001.

[18] N. I. M. Gould, D. Orban, and Ph. L. Toint. GALAHAD—a library of thread-safe fortran 90 packages for large-scale nonlinear optimization. Technical Report RAL-TR-2002-014, Rutherford Appleton Laboratory, Chilton, Oxfordshire, England, 2002.

[19] N. I. M. Gould and Ph. L. Toint. SQP methods for large-scale nonlinear programming. In M. J. D. Powell and S. Scholtes, editors, *System Modelling and Optimization, Methods, Theory and Applications*, pages 149–178, Dordrecht, The Netherlands, 2000. Kluwer Academic Publishers.

[20] N. I. M. Gould and Ph. L. Toint. Numerical methods for large-scale non-convex quadratic programming. In A. H. Siddiqi and M. Kočvara, editors, *Trends in Industrial and Applied Mathematics*, pages 149–179, Dordrecht, The Netherlands, 2002. Kluwer Academic Publishers.

[21] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395, 1984.

[22] L. S. Lasdon, J. Plummer, and G. Yu. Primal-dual and primal interior point algorithms for general nonlinear programs. *ORSA Journal on Computing*, 7(3):321–332, 1995.

[23] J. L. Morales, J. Nocedal, R. Waltz, G. Liu, and J. P. Goux. Assessing the potential of interior methods for nonlinear optimization. In O. Ghattas, editor, *Proceedings of the First Sandia Workshop on*

Large-Scale PDE-Constrained Optimization, Heidelberg, Berlin, New York, 2002. Springer Verlag.

[24] W. Murray and F. J. Prieto. A sequential quadratic programming algorithm using an incomplete solution of the subproblem. *SIAM Journal on Optimization*, 5(3):590–640, 1995.

[25] B. A. Murtagh and M. A. Saunders. A projected Lagrangian algorithm and its implementation for sparse non-linear constraints. *Mathematical Programming Studies*, 16:84–117, 1982.

[26] J. Renegar. *A Mathematical View of Interior-Point Methods in Convex Optimization*. SIAM, Philadelphia, 2001.

[27] S. M. Robinson. Perturbed Kuhn-Tucker points and rates of convergence for a class of nonlinear programming algorithms. *Mathematical Programming*, 7(1):1–16, 1974.

[28] D. C. Sorensen. An example concerning quasi-Newton estimates of a sparse Hessian. *SIGNUM Newsletter*, 16(2):8–10, 1981.

[29] A. L. Tits, A. Wächter, S. Bakhtiari, T. J. Urban, and C. T. Lawrence. A primal-dual interior-point method for nonlinear programming with strong global and local convergence properties. Technical Report TR 2002-29, Institute for Systems Research, University of Maryland, College Park, MD 20742, USA, 2002.

[30] M. Ulbrich and S. Ulbrich. Non-monotone trust region methods for nonlinear equality constrained optimization without a penalty function. *Mathematical Programming*, Published Online, 2002.

[31] M. Ulbrich, S. Ulbrich, and L. Vicente. A globally convergent primal-dual interior point filter method for nonconvex nonlinear programming. Technical Report TR00-11, Department of Mathematics, University of Coimbra, Coimbra, Portugal, 2000.

[32] R. J. Vanderbei and D. F. Shanno. An interior point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications*, 13:231–252, 1999.

[33] A. Wächter. *An Interior Point Algorithm for Large-Scale Nonlinear Optimization with Applications in Process Engineering*. PhD thesis, Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, USA, 2002.

[34] A. Wächter and L. T. Biegler. Failure of global convergence for a class of interior point methods for nonlinear programming. *Mathematical Programming*, 88(3):565–574, 2000.

[35] A. Wächter and L. T. Biegler. Global and local convergence of line search filter methods for nonlinear programming. CAPD Technical Report B-01-09, Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, Philadelphia, USA, 2001.

[36] M. H. Wright. Why a pure primal Newton barrier step may be infeasible. *SIAM Journal on Optimization*, 5(1):1–12, 1995.

[37] M. H. Wright. Ill-conditioning and computational error in interior methods for nonlinear programming. *SIAM Journal on Optimization*, 9(1):84–111, 1999.

[38] S. J. Wright. *Primal-Dual Interior-Point Methods*. SIAM, Philadelphia, USA, 1997.

[39] S. J. Wright. Effects of finite-precision arithmetic on interior-point methods for nonlinear programming. *SIAM Journal on Optimization*, 12(1):36–78, 2001.

[40] E. A. Yildirim and S. J. Wright. Warm-start strategies in interior-point methods for linear programming. *SIAM Journal on Optimization*, 12(3):782–810, 2002.

PDE Constrained Optimization

Ekkehard W. Sachs

FB IV - Mathematik
Universität Trier
54296 Trier, Germany

and

Interdisc. Center for Appl. Mathematics
Department of Mathematics
Virginia Tech
Blacksburg, VA 24061, USA

sachs@uni-trier.de

"PDE Constrained Optimization" and "Optimization of PDEs" were the themes of several conferences in the past two years, that took place in Santa Fe, New Mexico, the Weierstrass Institute in Berlin, the University of Heidelberg, and at the Mathematical Research Institute in Oberwolfach, Germany. This field also played an important role at a recent workshop at the IMA in Minneapolis during the optimization year. Since there is so much interest in this field, the question arises: Is this a hot new area or is there already some tradition in this field ? The answer is: both is true.

Back in the seventies ...

Back in the early 70s, when the author was a starting Ph.D. student at the Technical University of Darmstadt, Germany, we had a seminar on a new topic. No longer optimization in finite dimensions or with ordinary differential equations but instead "Optimal Control of Partial Differential Equations". The standard applications in those days were the control of the heat equation or the wave equation, that still serve as work horses in standard test examples for numerical methods. Hence already decades ago, there was research performed in the area of optimization of PDEs. Actually, there were also text books

and conference proceedings available that dealt with the subject of optimal control of PDEs, like e.g. the books by Butkovski [**?**] and Lions [**?**] published around 1970 and later the proceedings by Ray and Lainiotis [**?**].

In the following decades, various schools around the world were established, which worked on different aspects of these problems like the existence of optimal controls, necessary optimality conditions, duality theory, controllability and observability, feedback control laws, numerical methods and applications.

PDE constrained optimization is more ...
In today's research environment, the term "PDE-constrained optimization" includes more than just "Optimal Control of PDEs". For example, there is a large community, which solves optimal design problems. These optimization problems often include also systems described by partial differential equations, but they depend on a finite number of design variables, in contrast to optimal control problems, where the control variable usually lies in a function space. Another example of a large area of PDE constrained optimization problems are parameter estimation problems for PDEs. In this case, certain parameters in a PDE are the variables and they are adjusted in such a way that the output given by the PDE solution matches certain measured data.

In this note, I want to focus on the numerical aspects of PDE constrained optimization, since there is a lot of interest in this aspect by the people coming from optimization or the PDE community.
• What is different with PDE constrained optimization compared to finite dimensional optimization ?
• Do we need to care about this topic at all, if the PDE is discretized and we finally come up with a finite dimensional optimization problem ?
These are some typical questions that arise, when an optimizer looks at PDE constrained optimization. For a list of citations and literature, I refer to the proceedings of the workshops in Santa Fe [**?**] and Berlin [**?**], that have been published or are under way.

Discretization leads to finite dimensional optimization ... what's new ?
Obviously the resulting finite dimensional optimization problem is large scale or very large scale, when a discretized PDE is included in the constraints. Although the progress in using direct solvers for the linearized state equation has been substantial in the past decade, one tends to use iterative solvers for the inner iteration of an SQP method or an interior point method for very large problems. This leads to the question of inexact solves of the subproblems and how this fits into the convergence theories and algorithms. This has been already investigated by many researchers and is a well developed research area.

In the discretization process, however, it is crucial which type of discretization is being used. Unfortunately, there is no all-purpose discretization method for all types of PDEs. The type of discretization and solution technique for the PDE can be quite different depending on the problem, e.g. finite differences, finite elements, Ritz-Galerkin, discontinuous Galerkin, wavelets, multigrid, hierarchical basis, etc. This can affect the optimization method being used. Often the cost of solving the PDE numerically is the dominating factor in overall computational cost and using an efficient method for solving the PDE can save a lot of computing time. However, in the case of using a sophisticated PDE solver, often the optimization method has to be adapted to this special method. A good example are multigrid methods, where first control problems without constraints have been solved efficiently, a trend that started with Hackbusch's papers in the early 80s. In the past years various authors have weaved multigrid methods into optimization routines or vice versa. These methods can be used for control constrained problems or even more generally constrained optimization problems.

If iterative solvers for the resulting subproblems of optimization routines, often linear systems, are used, like KKT systems, it is mandatory for an efficient code to care about preconditioning, if PDEs are involved. Rather than using a preconditioner for the whole KKT system, various research groups have tried to utilize the structure of the KKT system in a clever way for better preconditioners, among others see [**?**]. This way, it also easier to incorporate preconditioners from the PDE community directly into the optimization code.

First discretize, then optimize ... or vice versa
Another point of discussion that has been raised over

the past decades and resurfaces in periodic time intervals is the following: PDE constrained optimization problems are posed in function spaces. One approach is to discretize the problem first, and then to optimize the finite-dimensional version. An alternative approach is to optimize first and then to discretize. This would mean, e.g., to derive the necessary optimality conditions for the infinite dimensional problem using the function space formulation and then to discretize these. One could go further by deriving even the Newton or SQP step in infinite dimensions and then to discretize this step. Which approach is better and more efficient was a topic of many discussions at each of the conferences mentioned above - and also in the seminar in the seventies mentioned in the beginning.

The frequency of the discussion and lack of a decisive argument point to the conclusion that there is no clear answer to this question and the best way to answer it, is problem dependent. The combination of SQP methods and multiple shooting methods for the solution of optimization problems with ODE constraints, where the optimal control has a complicated switching structure, shows that both approaches sometimes have to be combined.

Think infinite dimensional ... at least sometimes

One advantage of keeping the infinite dimensional flavor of the problem as long as possible is that certain infinite dimensional effects are better retained than in the discretized version. The necessary optimality conditions, e.g. for a hyperbolic control problem, can lead to an adjoint differential equation, which has a solution in a quite different space than the state equation. Therefore it might be efficient to use a different discretization scheme for the adjoint equation. Since this plays the role of a Lagrange multiplier, it could result in a Newton-SQP step with a nonsymmetric KKT matrix. This will not occur, if the discretization takes place at the level of the original optimization problem. But then the Lagrange multiplier has to approximate the solution of the adjoint equation with a discretization scheme, that is not suitable for the adjoint and could result in unnecessarily fine mesh refinements.

Another instance, where it pays to discretize late, is the regularity of the Lagrange multipliers. If the PDE constrained optimization problem exhibits state constraints, then the corresponding Lagrange multipliers in the infinite dimensional case are often very irregular, sometimes represented as discontinuous functions or even as distributions. If the user is aware of this, he can use appropriate discretization schemes for these multipliers to take into account their nonsmoothness.

A further example, how the infinite dimensional formulation can influence the numerical behavior, came up in quasi-Newton methods. It is well known, that these methods converge locally superlinearly under the usual regularity assumptions on the Hessian in finite dimensions. However, in infinite dimensions an additional assumption shows up. The initial approximation of the Hessian and the true Hessian do not only need to be close, but they are required to differ only by a compact operator. There are many optimal control problems, where this property can be checked. If one looks at the discretized problem, it seems as if this would not play any role, since everything is finite dimensional (and hence compact). But one can see numerically, that this effect indeed shows up in the local convergence behavior. Hence one can take extra care in choosing the initial quasi-Newton matrix using the knowledge from the infinite dimensional problem.

Other difficulties ...

A more obvious difference between an infinite and finite dimensional formulation is the identification of active indices. In any finite dimensional optimization problem with inequality constraints, the smallest absolute value of all inactive constraints is a positive number. This is not the case, if one has a continuum of inequality constraints, which depend continuously on the index. There has been work by Dunn in the late seventies where an additional assumption is imposed on the slope of the constraint function when it passes from the inactive to the active set. This has been used in the context of convergence analyses for algorithms. These type of conditions have shown up here and there in various research papers analyzing convergence of numerical methods for optimization problems over function space and in particular with PDEs.

On the more theoretical side occurs a phenomenon which is called the 2-norm-discrepancy and should have also some effect on the numerical behavior, but this has not been investigated too closely from a nu-

merical point of view. If the solution of the PDE is differentiated in the Fréchet sense with respect to controls, which are also functions, it is often necessary to use supremum norms for the function spaces. On the other hand, if one deals with second order sufficiency conditions, then this involves positive definiteness of the Hessian on certain subspaces. This is formulated in an inner product or L_2 norm setting. This yields a gap in the proper formulation of the function spaces which sometimes can be patched with a smoothness argument arising from the PDE formulation.

A few words about applications ...
Let me close this note with a few comments on applications and a biased selection. For a more complete picture check the proceedings mentioned before. There are many systems in engineering and economics that are described by partial differential equations and almost canonically there are also optimization problems associated with them.

A standard case where this occurs is when mathematical modelling technique yields a certain type of PDE like a diffusion process, but within the PDEs there are various coefficients not exactly known. The area of parameter identification or inverse problems deals with the problem to determine the unknown parameters from measurements taken. This is an optimization problem with a PDE and it is a fairly difficult one, although it exhibits quite a bit of structure.

Mathematical finance is often referred to as a hot new area. The value of an option for a certain underlying like a stock or stock index can be described by the Black-Scholes model. This involves a partial differential equation and if one tries to adapt some of its parameters so that the market prices are met better, then one arrives at a parameter identification problem. A portfolio optimization involving options leads to an optimization problem with PDE constraints, given by the Black-Scholes model.

Another example is from food sterilization. Here the product quality is optimized under the constraint, that sterility of the product is preserved. This leads to a nonlinear optimization problem with inequality and equality constraints coming from an optimal control problem with PDEs and control and state constraints.

A fairly recent development is the use of simpler models replacing the PDE in the optimization. Some of the technical terms are multilevel optimization, reduced order models, proper orthogonal decomposition, neural networks, etc. Although the PDE itself disappeared from the numerical solution of the optimization problem, it is important to investigate, for example, the quality of approximation by these surrogate models, their construction and efficiency etc.

This article is a review of some of the aspects of PDE constrained optimization. Obviously it is not complete and biased, but hopefully serves the purpose to motivate a novice in this area to take a closer look at this interesting and challenging field.

DAE-Constrained Optimization

Lorenz T. Biegler
Department of Chemical Engineering, Carnegie Mellon University, e-mail: lb01@andrew.cmu.edu
Andreas Wächter
Department of Mathematical Sciences IBM T.J. Watson Research Center, e-mail: andreasw@watson.ibm.com

1. Introduction

The dynamic behavior of many physical systems can be described by ordinary and partial differential equations, which express, for example, conservation laws on momentum, mass and energy coupled with algebraic equations, which represent constitutive relations or design specifications. In particular, the development of powerful commercial modeling and simulation tools for systems described by ordinary differential and algebraic equations (DAEs) has led designers, engineers, and researchers to consider DAE optimization as a natural extension of these simulation tools in order to systematize and automate decisions required for science and engineering studies. In the past decade new numerical algorithms for solving those problems have been proposed, but the design of powerful methods remains an active research area. In this article, we discuss some of the currently most popular solution approaches and finish with some open research questions.

The DAE optimization problem can be stated in a fairly general form as follows:

$$\min_{z(t),y(t),u(t),t_f,p} \varphi(z(t_f), y(t_f), u(t_f), t_f, p) \quad (1)$$

$$s.t. \quad \frac{dz(t)}{dt} = F\left(z(t), y(t), u(t), t, p\right) \quad (2)$$

$$0 = G\left(z(t), y(t), u(t), t, p\right) \quad (3)$$

$$z(t_0) = z_0 \quad (4)$$

$$H_s\left(z(t_s), y(t_s), u(t_s), t_s, p\right) = 0 \quad \text{for } s \in \{1, \ldots, n_S\} \quad (5)$$

$$(z^L, y^L, u^L) \leq (z(t), y(t), u(t)) \leq (z^U, y^U, u^U) \quad (6)$$

$$(p^L, t_f^L) \leq (p, t_f) \leq (p^U, t_f^U) \quad (7)$$

The "unknowns" in this optimization problem are the *differential state variables* $z(t)$, *algebraic state variables* $y(t)$, and *control variables* $u(t)$, all functions of the scalar "time" parameter $t \in [t_0, t_f]$, as well as *time-independent parameters* p, and possibly the *final time* t_f. As constraints the above formulation includes bounds on all those variables (6)-(7), the differential and algebraic equations (2)-(3) with initial conditions (4), and additional point constraints (5) for a finite number of time points t_s.

Note that the DAE system (2)-(3) is given without loss of generality in a semi-explicit form. Here, we assume that this system is index one, which implies that the matrix $\partial G/\partial y$ is nonsingular for all values of (z, y, u, p). Since a number of standard procedures can be applied to reduce high index algebraic equations to index one [2], this is not really a severe restriction.

Practical applications of dynamic optimization include:

- Design and control of chemical and mechanical processes for the handling of abnormal operations, severe disturbances and transitions to different operating points [4];
- Data assimilation, system identification and model predictive control requiring the on-line solution of a dynamic optimization problem with a nonlinear dynamic process model [1];
- Design and operation of aerospace applications including trajectories for aircraft, rockets and satellites [6];
- Conflict resolution of trajectories for multiple aircraft in a cooperative or noncooperative setting [30];
- Analysis and tuning of electronic circuits [27].

2. Dynamic Optimization Methods

To solve the dynamic optimization problem (1)-(7) a number of approaches can be taken. In particular, DAE optimization problems can be solved using a variational approach [29] or by various strategies that apply a Nonlinear Programming (NLP) solver to the DAE model. The indirect or *variational approach* is based on the solution of the first order necessary conditions for optimality obtained from Pontryagin's Maximum Principle [29]. For problems without inequality constraints, the optimality conditions can be formulated as a set of differential-algebraic equations. The resulting two-point boundary value problem has been addressed by single shooting, invariant embedding, multiple shooting or some discretization method such as collocation on finite elements or finite differences. A review of these approaches can be found in [14, 21].

On the other hand, if the problem requires the handling of active inequality constraints, finding the correct switching points as well as suitable boundary conditions for state and adjoint variables is difficult and NLP strategies must be used. Methods that apply NLP solvers can be separated into three groups: the sequential and simultaneous strategies and an intermediate strategy based on multiple shooting.

2.1 Sequential Methods

In a *sequential method*, the control variables are often represented as piecewise polynomials [4, 18], and optimization is performed with respect to the polynomial coefficients. Given initial conditions and a guess of the discretized optimal control profile, the model (2)-(4) is integrated with a DAE solver at each iteration. This produces values of the objective function (1) and constraints (5)-(6) (at fixed points t_s in time) which are used by a NLP solver to find the optimal coefficient values in the control discretization. The gradients of the objective and constraint functions with respect to the control coefficients and

parameters are calculated from sensitivity equations of the DAE system.

The sequential approach is a feasible path method; in every iteration the DAE system has to be solved. However, this procedure is stable only if the system does not contain increasing forward modes[1]. Otherwise, solving the DAE system for a given set of control parameters may be difficult or impossible. See [14, 21] for a review of these methods.

The data seen by the NLP solver corresponds to a small dense NLP with typically $O(100)$ variables and $O(1000)$ (inequality) constraints. The dominant cost for the optimization is due to computing the state equations and their sensitivities (using a direct [26] or adjoint [13, 22] approach). Because of this cost, second derivatives are rarely available and dense quasi-Newton SQP methods (such as [20]) are best suited for this approach.

2.2 Multiple shooting methods

These methods serve as a bridge between sequential approaches and a complete discretization of the state and control variables. Here, the time domain is partitioned into a moderate number (say, $O(50)$) of time elements, $t_0 < t_1 < \ldots < t_i < \ldots < t_f$, and the DAE model is integrated separately in each element [11, 24]. Control variables and path constraints (5)-(6) are treated in the same manner as in the sequential approach, and equality constraints are added to the NLP in order to link the elements and ensure that the states are continuous across each element boundary. Sensitivities are obtained with respect to both the control variable coefficients as well as the initial conditions, $z_{0,i}$, of the states in each element.

As with sequential approaches, accuracy of the state profiles is governed by the DAE solver (not seen by the optimizer) and the cost of the optimization is dominated by the DAE state and sensitivity calculations. The resulting NLP problem is moderately large (with $O(10000)$ variables) and reduced space SQP methods (see [19, 24, 34]) appear best suited for these approaches. Compared to sequential approaches, they can better handle DAEs with forward increasing modes. On the other hand, multiple shooting approaches lead to larger, more structured NLPs with dense sensitivity blocks.

[1] We define increasing forward modes as profiles whose magnitudes become unbounded as time increases.

2.3 Simultaneous methods

Simultaneous methods fully discretize the DAE system by approximating not only the control variables, but also the state variables by piecewise polynomial functions over *finite elements*, $t_0 < t_1 < \ldots < t_f$, often by *implicit Runge-Kutta methods*. In [6] this approximation is given by Hermite-Simpson polynomials and leads to sparse equations of relatively low order approximation errors. Alternatively, one can apply *collocation on finite elements* and use a monomial basis representation [3] for the differential profiles. Algebraic states and control profiles are represented analogously by Lagrange interpolation profiles. The NLP formulation now has as variables all the polynomial coefficients and as constraints the DAE (2)-(3) enforced at all collocation points (intermediate time points within the finite elements). Further constraints are on continuity of the differential state variables at element boundaries. Control and algebraic state profiles are allowed to have discontinuities at those boundaries.

Simultaneous methods directly couple the solution of the DAE system with the optimization problem; the DAE system is solved only once, at the optimal point. Moreover, simultaneous approaches have advantages for problems with path constraints and with instabilities that occur for a range of inputs. Because they can be seen as extensions of boundary value solvers, they are able to "pin down" increasing modes in the forward direction by enforcing the appropriate boundary condition. In addition, these methods allow the direct enforcement of state and control variable bounds (6), at the same level of discretization as the state variables of the DAE system.

The degree of accuracy of the solution of the DAE system is now determined by the number of the finite elements, which could be $O(100)$ or larger, thus leading to very large, sparse, and structured NLPs with $O(10^6)$ variables. On the other hand, exact first and second derivatives are straightforward to evaluate and an additional DAE solver is not needed.

These NLPs are usually solved using variations of Successive Quadratic Programming (SQP) or Interior Point (IP) methods. In both cases, *full-space* approaches take advantage of the DAE optimization

problem structure and the sparsity of the model during the computation of the optimization step, obtained by the solution of a QP. They are best suited for problems where the ratio of state variables to control variables is small, say $O(10)$, [8, 7]. Here, second derivatives of the objective function and constraints are desired, as are measures to deal with directions of negative curvature in the Hessian matrix [8, 17]. A recent monograph [6] provides a detailed description of the simultaneous approach with full-space methods, along with mesh refinement strategies and case studies in mechanics and aerospace.

However, if the ratio of state to control variables is large as in many process control applications, a reduced space approach might be preferrable, where the (linearized) equality constraints and associated variables are eliminated. Often these correspond to the collocation equations and the state variable coefficients. In an active set approach, a *reduced QP* in the space of the remaining variables (often the control variables) is solved. Alternatively, in an interior point approach, a dense linear system is solved in the same reduced space, and for very large problems, iterative methods can be applied here in order to avoid having to construct the reduced Hessian explicitly. In this way, dynamic optimization problems with up to $2,000,000$ variables and $4,500$ degrees of freedom have been solved [9].

3. Open Problems

We close with a brief discussion of challenges in dynamic optimization, focusing on the simultaneous approach, which puts the strongest demands on the NLP algorithm.

Dynamic Optimization Formulations

For optimal control problems where control variables are discretized at the same level as the state variables, a number of open questions relate to convergence to the solution of the original variational problem. A number of studies have shown (e.g., [31, 33, 28]) that the Karush Kuhn Tucker (KKT) conditions of the simultaneous NLP can be made consistent with the optimality conditions of the variational problem as the maximum element size approaches zero. Nevertheless, these consistency properties alone do not imply that the sequence of approximate solutions will converge at all, and several studies report stability problems due to poor discretizations, high index inequality constraints and singular arcs [25, 15, 5]. Special cases of these have been analyzed rigorously in [10, 32, 16]. Related to this question is the optimal placement of finite elements. Optimization formulations that facilitate movement and addition of finite elements in order to satisfy accuracy and consistency properties are reported in [9], but a more rigorous analysis is needed to put this strategy on a more fundamental basis.

Improvements to NLP Algorithms

Since the heart of the simultaneous approach is a robust and efficient large-scale NLP solver, improvement in the NLP algorithm will immediately lead to advances in the DAE optimization method. Over the past few years interior point NLP methods [36, 12, 35] have pushed the size of problems that could be solved, because they avoid the combinatorial bottleneck of identifying the active inequality constraints, and because they allow a straightfoward exploitation of the structure of the KKT matrix and the direct use of second derivatives. In particular, our experience with our interior point code, IPOPT [37], is very promising, although many open questions regarding general purpose IP-NLP solvers remain, including the following issues.

Warm starts are needed for nonlinear model predictive control (NMPC) and other applications where DAE optimization must be performed repeatedly with only slightly perturbed data. Whereas this is naturally handled in active set SQP methods, better warm start strategies need to be developed for IP algorithms. Some experience along these lines is reported in [23].

Efficient preconditioners for iterative linear solvers, such as conjugate gradient methods, are still needed within a reduced space approach, particularly within an IP framework where the barrier Hessian becomes increasingly ill-conditioned. In [9] we observed that separating the reduced Hessian of the barrier term from the reduced Hessian of the original problem leads to a significant reduction in CG iterations, but at the expense of constructing this additional term. Finally, because second derivatives can be incorporated directly or can be

differenced along conjugate gradient steps, improved provisions must be made to deal with nonpositive eigenvalues in the reduced space of the constraint set. Here we especially need to consider singular control problems, which have characteristics similar to inverse problems.

As a result, the development of algorithms for DAE constrained optimization remains an active and interesting research area, with close ties to the development of efficient and robust NLP methods.

REFERENCES

[1] F. Allgöwer and A. Zheng, editors. *Nonlinear Model Predictive Control*. Birkhäuser, Basel, 2000.

[2] U. M. Ascher and L. R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM, Philadelphia, PA, USA, 1998.

[3] G. Bader and U. Ascher. A new basis implementation for mixed order boundary value ODE solver. *SIAM Journal on Scientific Computing*, 8:483–500, 1987.

[4] P. I. Barton, R. J. Allgor, W. F. Feehery, and S. Galan. Dynamic optimization in a discontinuous world. *Industrial and Engineering Chemistry Research*, 37:966–981, 1998.

[5] J. Bausa and G. Tsatsaronis. Discretization methods for direct collocation of optimal control problems. *Computers and Chemical Engineering*, 25(2-3):359–370, 2001.

[6] J. Betts. *Practical Methods for Optimal Control Using Nonlinear Programming*. SIAM Series on Advances in Design and Control. SIAM, Philadelphia, PA, USA, 2001.

[7] J. T. Betts and P. D. Frank. A sparse nonlinear optimization algorithm. *Journal of Optimization Theory and Application*, 82(3):519–541, 1994.

[8] J. T. Betts and W. P. Huffman. Application of sparse nonlinear programming to trajectory optimization. *J. Guidance, Dynamics and Control*, 15:198–210, 1992.

[9] L. T. Biegler, A. M. Cervantes, and A. Wächter. Advances in simultaneous strategies for dynamic process optimization. *Chemical Engineering Science*, 57(4):575–593, 2002.

[10] N. Biehn. *Implicit Runge-Kutta Methods for Stiff and State Constrainted Optimal Control Problems*. PhD thesis, Department of Mathematics, North Carolina State University, Raliegh, NC, 2001.

[11] H. G. Bock and K. J. Plitt. A multiple shooting algorithm for direct solution of optimal control problems. In *9th IFAC World Congress*, Budapest, 1984.

[12] R. H. Byrd, M. E. Hribar, and J. Nocedal. An interior point algorithm for large-scale nonlinear programming. *SIAM Journal on Optimization*, 9(4):877–900, 1999.

[13] Y. Cao, S. Li, and L. R. Petzold. Adjoint sensitivity analysis for differential-algebraic equations: Algorithms and software. *Journal of Computational and Appied Mathematics*, to appear.

[14] A. M. Cervantes and L. T. Biegler. Optimization strategies for dynamic systems. In C. Floudas and P. Pardalos, editors, *Encyclopedia of Optimization*. Kluwer, 1999.

[15] A. M. Cervantes, A. Wächter, R. H. Tütüncü, and L. T. Biegler. A reduced space interior point strategy for optimization of differential algebraic systems. *Computers and Chemical Engineering*, 24(1):39–51, 2000.

[16] A. Dontchev, W. Hager, and V. Veliov. Second order Runge Kutta approximations in constrained optimal control. *SIAM Journal on Numerical Analysis*, 38:202–220, 2000.

[17] A. Forsgren and P. E. Gill. Primal-dual interior methods for nonconvex nonlinear programming. *SIAM Journal on Optimization*, 8(4):1132–1152, 1998.

[18] P. E. Gill, L. Jay, M. Leonard, L. Petzold, and V. Sharma. An SQP method for the optimal control of large-scale dynamical systems. *Journal of Computational and Appied Mathematics*, 120:197–213, 2000.

[19] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Journal on Optimization*, 12(4):979–1006, 2002.

[20] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. User's guide for NPSOL (version 4.0): A Fortran package for nonlinear programming. Technical report, Systems Optimization Laboriatory, Stanford University, Stanford, CA, USA, 1986.

[21] M. Groetschel, S. O. Krumke, and J. Rambau, editors. *Online Optimization of Large Scale Systems*. Springer, Berlin, 2001.

[22] L. Hasdorff. *Gradient Optimization and Nonlinear Control*. Wiley, New York, 1976.

[23] T. Jockenhövel, L. T. Biegler, and A. Wächter. Dynamic optimization of the tennessee eastman process using the optcontrolcentre. Technical Report CAPD B-02-11, Department of Chemical Engineering, Carnegie Mellon University, 2002.

[24] D. B. Leineweber. *Efficient Reduced SQP Methods for the Optimization of Chemical Processes Described by Large Sparse DAE Models*. PhD thesis, University of Heidelberg, Heidelberg, Germany, 1999.

[25] J. S. Logsdon and L. T. Biegler. On the accurate solution of differential-algebraic optimization problems. *Industrial and Engineering Chemistry Research*, 28:1628–1640, 1989.

[26] T. Maly and L. Petzold. Numerical methods and software for sensitivity analysis of differential algebraic equations. *Applied Numerical Mathematics*, 20:57–79, 1996.

[27] W. Nye, D. C. Riley, A. Sangiovanni-Vincentelli, and A. L. Tits. DELIGHT.SPICE: An optimization-based system for the design of integrated circuits. *IEEE Transactions on Computer-Aided Design of*

ICs and Systems, CAD-7(4):501–519, April 1988.

[28] E. Polak. *Optimization: Algorithms and Consistent Approximations*. Springer, New York, NY, 1997.

[29] V. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mishchenko. *The Mathematical Theory of Optimal Processes*. Interscience Publishers Inc., New York, NY, 1962.

[30] A. Raghunathan, V. Gopal, D. Subramanian, and L. T. Biegler. Dynamic optimization strategies for 3D conflict resolution of multiple aircrafts. Technical Report CAPD B-02-12, Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA, December 2002.

[31] G. W. Reddien. Collocation at Gauss points as a discretization in optimal control. *SIAM Journal on Control and Optimization*, 17:298–299, 1979.

[32] A. Schwartz. *Theory and Implementation of Numerical Methods Based on Runge-Kutta Integration for Solving Optimal Control Problems*. PhD thesis, Department of Electrical Engineering and Computer Science, University of California-Berkeley, Berkeley, CA, 1996.

[33] A. Schwartz and E. Polak. Consistent approximations for optimal control problems based on Runge-Kutta integration. *SIAM Journal on Control and Optimization*, 34(4):1235–1269, 1996.

[34] D. J. Ternet and L. T. Biegler. Recent improvements to a multiplier free reduced hessian successive quadratic programming algorithm. *Computers and Chemical Engineering*, 22(7-8):963–991, 1998.

[35] R. J. Vanderbei and D. F. Shanno. An interior-point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications*, 13:231–252, 1999.

[36] A. Wächter and L. T. Biegler. Global and local convergence of line search filter methods for nonlinear programming. Technical Report CAPD B-01-09, Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA, August 2001.

[37] COIN-OR web page. `http://www.coin-or.org/Ipopt`.

Mathematical Programs with Complementarity Constraints

Sven Leyffer

Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, USA.

(`leyffer@mcs.anl.gov`)

1. Introduction

An exciting new application of nonlinear programming techniques is mathematical programs with complementarity constraints (MPCC),

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & c(x) \geq 0 \\ & 0 \leq x_1 \ \perp \ x_2 \geq 0, \end{array} \tag{1}$$

where $x = (x_0, x_1, x_2)$ and $\perp$ is the complementarity operator, which requires that either a component $x_{1i} = 0$ or the corresponding component $x_{2i} = 0$. It is straightforward to include equality constraints in (1). Problems of this type arise in many engineering and economic applications; see the survey [6], the monographs [12, 13], and the growing collections of test problems [9, 4].

One attractive way of solving (1) is to replace the complementarity condition by a set of nonlinear inequalities, such as $X_1 x_2 \leq 0$, and then solve the equivalent nonlinear program (NLP),

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & c(x) \geq 0 \\ & x_1, \ x_2 \geq 0, \ X_1 x_2 \leq 0, \end{array} \tag{2}$$

where $X_1 = \text{diag}(x_1)$. Unfortunately, it has been shown [15] that (2) violates the Mangasarian-Fromovitz constraint qualification (MFCQ) at *any* feasible point. This failure of MFCQ implies that the multiplier set is unbounded, the central path fails to exist, the active constraint normals are linearly dependent, and linearizations of (2) can become inconsistent *arbitrarily close* to a solution. In addition, early numerical experience with this approach has been disappointing [2]. As a consequence, solving MPCCs via NLPs such as (2) has been commonly regarded as numerically unsafe.

The failure of MFCQ in (2) can be traced to the formulation of the complementarity constraint as $X_1 x_2 \leq 0$. Consequently, algorithmic approaches have focused on avoiding this formulation. Instead, researchers have developed special purpose algorithms for MPCCs, such as branch-and-bound methods [2], implicit nonsmooth approaches [13], piecewise SQP methods [12], and perturbation and penalization approaches [5] analyzed in [16]. All of these techniques, however, require significantly more work than a standard NLP approach to (2).

Recently, exciting new developments have demonstrated that the gloomy prognosis about the use of (2) may have been premature. Standard NLP solvers have been used to solve a large class of MPCCs,

written as NLPs, reliably and efficiently. This short note surveys these novel developments and summarizes open questions and possible extensions of these ideas.

The remainder is organized as follows. The next section provides a summary of certain stationarity concepts for MPCCs and establishes an important relationship with the Karush-Kuhn-Tucker (KKT) conditions of (2). This relationship is pivotal in the success of NLP solvers. The development of two important classes of solvers, sequential quadratic programming (SQP) and interior-point methods (IPMs), is charted in the subsequent two sections. The note concludes by providing a brief description of open problems.

2. The NLP Revolution

The resurgence of interest in the analysis of NLP solvers applied to (2) is motivated by the success of SQP methods in particular. A simple but key observation of Scholtes is that strong stationarity is equivalent to the KKT conditions of (2).

A point x^* is called *strongly stationary* if and only if there exist multipliers $\lambda \geq 0$, $\hat{\nu}_1$, and $\hat{\nu}_2$ such that

$$\begin{aligned} \nabla f^* - \nabla c^{*^T}\lambda - \begin{pmatrix} 0 \\ \hat{\nu}_1 \\ \hat{\nu}_2 \end{pmatrix} &= 0 \\ c(x^*) &\geq 0 \\ x_1^*, x_2^* &\geq 0 \\ x_{1j}^* = 0 \ \text{ or } \ x_{2j}^* &= 0 \\ c_i^*\lambda_i = 0,\ x_{1j}^*\hat{\nu}_{1j} = 0,\ x_{2j}^*\hat{\nu}_{2j} &= 0 \\ \hat{\nu}_{1j} \geq 0,\ \hat{\nu}_{2j} \geq 0, \text{ if } x_{1j}^* = x_{2j}^* &= 0. \end{aligned} \tag{1}$$

These are the KKT conditions of the *relaxed NLP* [15], which contains no complementarity condition and is therefore well behaved.

The KKT conditions of (2) are similar to (1), and this similarity will be exploited. Formally, a point x^* is called is a KKT point of (2) if and only if there exist multipliers $\lambda \geq 0$, $\nu_1 \geq 0$, $\nu_2 \geq 0$, and $\xi \geq 0$, such that

$$\begin{aligned} \nabla f^* - \nabla c^{*^T}\lambda - \begin{pmatrix} 0 \\ \nu_1 - X_2\xi \\ \nu_2 - X_1\xi \end{pmatrix} &= 0 \\ c(x^*) &\geq 0 \\ x_1^*, x_2^* &\geq 0 \\ X_1^* x_2 &\leq 0 \\ c_i(x)\lambda_i = 0,\ x_{1j}\nu_{1j} = 0,\ x_{2j}\nu_{2j} &= 0. \end{aligned} \tag{2}$$

Note that complementarity between ξ and $X_1^* x_2 \leq 0$ follows trivially. Now observe that (1) and (2) are equivalent if we set

$$\hat{\nu}_1 = \nu_1 - X_2\xi \tag{3}$$

$$\hat{\nu}_2 = \nu_2 - X_1\xi. \tag{4}$$

Hence there exists a *minimal value* of ξ, namely,

$$\xi_i = \begin{cases} 0 & \text{if } x_{1i}^* = x_{2i}^* = 0 \\ \max\left(0, \dfrac{-\hat{\nu}_{1i}}{x_{2i}^*}\right) & \text{if } x_{2i}^* > 0 \\ \\ \max\left(0, \dfrac{-\hat{\nu}_{2i}}{x_{1i}^*}\right) & \text{if } x_{1i}^* > 0, \end{cases} \tag{5}$$

from which it follows that the unboundedness of the multipliers of (2) has a very special structure: the multipliers form a ray.

The fact that the KKT conditions of (2) are equivalent to strong stationarity implies the existence of bounded multipliers. This can be exploited in the analysis of SQP methods and in the design of robust IPM methods for MPECs.

3. SQP Lead the Way

SQP methods have recently been shown to solve MPCCs reliably as NLPs, despite the common folklore that this approach is doomed. Over 150 problems were solved, and the SQP solver obtained *quadratic* convergence for all but two problems [7].

This success of SQP methods has motivated renewed interest in the theoretical properties of SQP methods. In [1] it is shown that an SQP method with *elastic mode* converges locally. The key idea is to consider a penalized version of (2). The penalty problem satisfies MFCQ; and near a strongly stationary point, a sufficiently large penalty parameter

can be found, similar to (5). Convergence can thus be established by using standard techniques.

In [8] it is shown that SQP converges superlinearly near a strongly stationary point. The proof is divided into two parts. First, it is shown that if $x_1^{(k)^T} x_2^{(k)} = 0$ at some iteration k, then the SQP approximation of (2) about this point is equivalent to the SQP approximation of the relaxed NLP. Since the latter is a well behaved problem, superlinear convergence follows. The second part of the proof assumes that $x_1^{(k)^T} x_2^{(k)} > 0$, and it is shown that each QP basis remains bounded away from singularity. Again, convergence can be established by using standard techniques.

One undesirable assumption in [8] is that all QP approximations are consistent. This is trivially true if $x_1^{(k)^T} x_2^{(k)} = 0$ for some k, and it can be shown to hold if the lower-level problem satisfies a certain mixed-P property [12]. In practice [7], a simple heuristic is implemented that relaxes the linearization of the complementarity constraint.

4. Interior-Point Methods

In contrast to SQP methods, interior-point methods (IPMs) are not as robust at solving MPCCs. Using default settings, they solve about 80% of MPCCs. This is still remarkable, however, considering that the constraint gradients are dependent and the central path fails to exist.

The reason for the nonexistence of the central path is the complementarity constraint. Clearly,

$$x_1 \geq 0\ ,\ x_2 \geq 0,\ \text{ and }\ X_1 x_2 \leq 0$$

have no interior. As a consequence, multipliers can become unbounded, resulting in slow progress (if any) toward the solution.

Three approaches to remedy this situation are being investigated. The first two are related to [16] and either relax the complementarity constraint or penalize it. The third approach mixes a simple active set heuristic with the IPM to identify and remove indices of bi-active constraints ($x_{1i} = 0 = x_{2i}$) [3]. It is not clear at present, however, what convergence properties this approach possesses.

The relaxation scheme [11, 14] introduces a parameter $\tau > 0$ and relaxes the complementarity constraint to

$$x_1 \geq 0\ ,\ x_2 \geq 0\ \text{ and }\ X_1 x_2 \leq \tau.$$

A standard primal-dual method is then applied, and the parameter τ is controlled in conjunction with the barrier parameter. It can be shown that near a strongly stationary point, the multipliers remain bounded and the central path exists.

An alternative to relaxation is to introduce an ℓ_1 penalty for the complementarity constraint and add $\rho x_1^T x_2$ to the objective. The resulting penalized NLP satisfies MFCQ and is well behaved. In addition, since $x_1, x_2 \geq 0$, no absolute values are required, and the problem is smooth. Near a strongly stationary point, a sufficiently large (but finite) penalty parameter exists, and any IPM method converges to this stationary point. We are investigating techniques for updating the penalty parameter, ρ.

The two schemes can be shown to be equivalent in the sense that for every relaxation τ there exists a penalty parameter ρ such that both approaches give the same solution. We prefer to control the penalty parameter, however, as it allows us to control the multipliers directly.

Interior-point methods for MPCCs have also been considered in [12], for example, the penalty interior-point algorithm (PIPA). This is a hybrid SQP-IPM method that aims to remain interior *only* with respect to the variables in the complementarity constraint, by perturbing it to

$$x_1 \geq 0\ ,\ x_2 \geq 0,\ \text{ and }\ X_1 x_2 = \tau.$$

Note that the last constraint is an *equation*. It is possible to construct a simple example where $x_1^* = x_2^* = 0$ and the central path fails to exist. Thus, this perturbation is suitable only for problems without bi-active constraints. In [10] another example is constructed that shows that PIPA may fail to converge, even when strict complementarity ($x_1^* + x_2^* > 0$) holds. The reason for this adverse behavior is the trustregion used in PIPA, which is controlled by the norm of the infeasibility.

5. Conclusion and Outlook

The underlying theme of the preceding two sections has been to show that *small modifications* enable

NLP solvers to work for MPCCs. Both SQP methods and IPM solvers either perturb or penalize the complementarity constraint. The key to proving convergence in both cases is the equivalence between strong stationarity (1) and the KKT conditions (2).

The robust solution of MPCCs as NLPs has harnessed the power of large-scale NLP solvers to this new and exciting class of problem. Despite this success, however, there still remain some open questions.

An important open question is whether global convergence results can be established and—more important—whether these results can be strengthened to provide convergence to B-stationary points [15]. For instance, it is easy to construct examples for which the NLP approaches converge to a feasible C-stationary point. Unfortunately, C-stationary points allow trivial first-order descent directions (and are really a misnomer!). Convergence to B-stationary points that are not strongly stationary can be observed in practice, even though the multiplier of the complementarity constraint ξ diverges to infinity.

Some MPCCs require global solutions to be obtained. For instance, in the context of brittle fracture identification, the global minimum corresponds to the first structural failure. Local minima are physically meaningless in this case. Finding global minima for large NLPs is a challenging problem, and success is likely to involve the use of robust NLP techniques in conjunction with complementarity solvers.

Acknowledgements

This work was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

REFERENCES

[1] M. Anitescu. On solving mathematical programs with complementarity constraints as nonlinear programs. Preprint ANL/MCS-P864-1200, MCS Division, Argonne National Laboratory, Argonne, IL, USA, 2000.

[2] J. F. Bard. Convex two-level optimization. *Mathematical Programming*, 40(1):15–27, 1988.

[3] H. Benson, D. F. Shanno, and R. V. D. Vanderbei. Interior-point methods for nonconvex nonlinear programming: Complementarity constraints. Technical Report ORFE 02-02, Princeton University, Operations Research and Financial Engineering, 2002.

[4] S. P. Dirkse. MPEC world. Webpage, GAMS Development Corporation, http://www.gamsworld.org/mpec/, 2001.

[5] S. P. Dirkse, M. C. Ferris, and A. Meeraus. Mathematical programs with equilibrium constraints: Automatic reformulation and solution via constraint optimization. Technical Report NA-02/11, Oxford University Computing Laboratory, July 2002.

[6] M. C. Ferris and J. S. Pang. Engineering and economic applications of complementarity problems. *SIAM Review*, 39(4):669–713, 1997.

[7] R. Fletcher and S. Leyffer. Numerical experience with solving MPECs as NLPs. Numerical Analysis Report NA/210, Department of Mathematics, University of Dundee, Dundee, UK, 2002.

[8] R. Fletcher, S. Leyffer, D. Ralph, and S. Scholtes. Local convergence of SQP methods for mathematical programs with equilibrium constraints. Numerical Analysis Report NA/209, Department of Mathematics, University of Dundee, Dundee, UK, 2002.

[9] S. Leyffer. MacMPEC: AMPL collection of MPECs. Webpage, http://www-unix.mcs.anl.gov/~leyffer/MacMPEC/, 2000.

[10] S. Leyffer. The penalty interior point method fails to converge for mathematical programs with equilibrium constraints. Numerical Analysis Report NA/208, Department of Mathematics, University of Dundee, Dundee, UK, February 2002.

[11] X. Liu and J. Sun. Generalized stationary points and an interior point method for mathematical programs with equilibrium constraints. Preprint, Singapore, School of Business, 2002.

[12] Z.-Q. Luo, J.-S. Pang, and D. Ralph. *Mathematical Programs with Equilibrium Constraints*. Cambridge University Press, Cambridge, UK, 1996.

[13] J. Outrata, M. Kocvara, and J. Zowe. *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints*. Kluwer Academic Publishers, Dordrecht, 1998.

[14] A. Raghunathan and L. T. Biegler. Barrier methods for mathematical programs with complementarity constraints (MPCCs). Technical report, Carnegie Mellon University, Department of Chemical Engineering, Pittsburgh, PA 15213, December 2002.

[15] H. Scheel and S. Scholtes. Mathematical program with complementarity constraints: Stationarity, optimality and sensitivity. *Mathematics of Operations Research*, 25:1–22, 2000.

[16] S. Scholtes. Convergence properties of regularization schemes for mathematical programs with complementarity constraints. *SIAM Journal on Optimization*, 11(4):918–936, 2001.

Mixed-Integer Nonlinear Programming

Michael R. Bussieck and Armin Pruessner
GAMS, Development Corporation, 1217 Potomac St, NW
Washington, DC 20007. (MBussieck@gams.com, APruessner@gams.com).

*R*ecently, the area of Mixed Integer Nonlinear Programming (MINLP) has experienced tremendous growth and a flourish of research activity. In this article we will give a brief overview of past developments in the MINLP arena and discuss some of the future work that can foster the development of MINLP in general and, in particular, robust solver technology for the practical solution of problems.

1. Introduction

Mixed Integer Nonlinear Programming (MINLP) refers to mathematical programming with continuous and discrete variables and nonlinearities in the objective function and constraints. The use of MINLP is a natural approach of formulating problems where it is necessary to simultaneously optimize the system structure (discrete) and parameters (continuous).

MINLPs have been used in various applications, including the process industry and the financial, engineering, management science and operations research sectors. It includes problems in process flow sheets, portfolio selection, batch processing in chemical engineering (consisting of mixing, reaction, and centrifuge separation), and optimal design of gas or water transmission networks. Other areas of interest include the automobile, aircraft, and VLSI manufacturing areas. An impressive collection of MINLP applications can be found in [14] and [15]. The needs in such diverse areas have motivated research and development in MINLP solver technology, particularly in algorithms for handling large-scale, highly combinatorial and highly nonlinear problems.

The general form of a MINLP is

$$\begin{array}{lll} \text{minimize} & f(x,y) & \\ \text{subject to} & g(x,y) \leq 0 & \\ & x \in X & \\ & y \in Y & \text{integer} \end{array} \tag{1}$$

The function $f(x,y)$ is a nonlinear objective function and $g(x,y)$ a nonlinear constraint function. The variables x, y are the decision variables, where y is required to be integer[1] valued. X and Y are bounding-box-type restrictions on the variables. We refer to [9] for more information about MINLP fundamentals in textbook format.

2. Algorithms

MINLP problems are precisely so difficult to solve, because they combine all the difficulties of both of their subclasses: the combinatorial nature of mixed integer programs (MIP) and the difficulty in solving nonconvex (and even convex) nonlinear programs (NLP). Because subclasses MIP and NLP are among the class of theoretically difficult problems (*NP-complete*), so it is not surprising that solving MINLP can be a challenging and daring venture. Fortunately, the component structure of MIP and NLP within MINLP provides a collection of natural algorithmic approaches, exploiting the structure of each of the subcomponents.

Solution Approaches

Methods for solving MINLPs include innovative approaches and related techniques taken and extended from MIP. Outer Approximation (OA) methods [5, 6], Branch-and-Bound (B&B) [16, 23], Extended Cutting Plane methods [33], and Generalized Bender's Decomposition (GBD) [13] for solving MINLPs have been discussed in the literature since the early 1980's. These approaches generally rely on the successive solutions of closely related NLP problems. For example, B&B starts out forming a pure continuous NLP problem by dropping the integrality requirements of the discrete variables (often called the relaxed MINLP or RMINLP). Moreover, each node of the emerging B&B tree represents a solution of the RMINLP with adjusted bounds on the discrete variables.

In addition, OA and GBD require the successive solution of a related MIP problem. Both algorithms decompose the MINLP into an NLP sub-

[1] Other special types of discrete variables known from the "linear world" such as SOS, semi-continuous, and semi-integer variables can also be handled by most algorithms.

problem that has the discrete variables fixed and a linear MIP master problem. The main difference between GBD and OA is in the definition of the MIP master problem. OA relies on tangential planes (or linearizations), effectively reducing each subproblem to a smaller feasible set, whereas the master MIP problem generated by GBD is given by a dual representation of the continuous space.

The approaches described above only guarantee global optimality under (generalized) convexity. Deterministic algorithms for global optimization of *nonconvex problems* require the solution of subproblems obtained via *convex relaxations* of the original problem in a branch-and-bound context, and have been quite successful in solving MINLPs [7, 29].

3. Software

Although theoretical algorithmic ideas for solving MINLP have been around for a while, the practical implementation of such concepts is much more difficult. Memory limitations, efficient numerical linear algebra routines, suitable algorithmic tolerances, and determining default solver options are some of the key issues faced when extending algorithms to large-scale, general-purpose software. In this section we give a brief and possibly incomplete historical overview of practical general purpose MINLP software.

Commercial MINLP Software Packages

Best to our knowledge, the earliest commercial software package that could solve MINLP problems was SCICONIC [10, 27] in the mid 1970's. Rather than handling nonlinearities directly, linked SOS variables provided a mechanism to represent discretized nonlinear functions and allowed solving the problem via MIP. In the mid 1980's Grossman and Kocis [17] developed GAMS/DICOPT [12], a general purpose MINLP algorithm based on the outer approximation method. In the early 1990's LINDOs [25] and What's Best [24] B&B code using the Generalized Reduced Gradient (GRG) code for subproblems was extended to solve MINLPs.

Since then a number of excellent academic as well as commercial codes have surfaced, including alphaECP [34] and mittlp [28], both of which are based on extended cutting plane methods, and MINLP_BB [19] and SBB [12], which use branch-and-bound to solve relaxed NLP subproblems. Even on the frontier of global MINLP, reliable and large-scale packages have materialized including alphaBB [1] and BARON [29], which use convex relaxations in a branch-and-bound framework.

Modeling Languages

The emergence of *algebraic modeling languages* in the mid to late 1980's and early 1990's has greatly simplified the process of modeling, in particular the formulation of MINLP type problems. Also, from a MINLP solver perspective, a modeling system delivers reliable black-box-type function evaluations and first and second order derivative information. Finally, the common solver interface of a modeling system allows MINLP algorithms to deploy existing NLP and MIP solvers to solve subproblems in a seamless way. A collection of MINLP models can be found in libraries such as MacMINLP [18] (AMPL [11] models), chapter 12 of [8] (GAMS [4] models) and as a superset MINLPLib [4] (GAMS models). The latter is available as part of the *MINLP World*. MINLP World is a forum for discussion and dissemination of information about all aspects of MINLP [20].

4. Recent Developments

With the recent progress made in global optimization, the importance of modeling systems has taken on a more significant role. In particular, most global solvers require more than black-box function evaluations. These solvers need *structural information* of algebraic expressions to build convex relaxations. AlphaBB and the modeling language MINOPT [26], as well as the recent release of GAMS/BARON [29] have shown the feasibility of this concept.

Another important advancement is the implementation of *open algorithms*. AIMMS-OA [2] is an outer approximation method similar to GAMS/DICOPT, but with the distinct feature that it allows user modification for fine-tuning the method for a particular problem. Such an open approach allows advanced users to adjust the algorithm to suit the problem at hand.

Recent research has also focused on combining of Random Search (RS), such as Tabu, Scatter Search, Simulated Annealing or Genetic Algorithms, with NLP methods. Recent implementations like OQNLP [12, 30] and LaGO [21, 22] have proven to be quite successful.

Finally, the area of Disjunctive Programming uses disjunctions and logic propositions to represent the discrete decisions in the continuous and discrete space respectively. Disjunctive programs, conveniently modeled and automatically reformulated in big M or convex region models, give access to a rich area of applications. Widespread interest in such models has spawned a new computing environment (LogMIP [31]), developed specifically for generalized disjunctive programming.

5. Future Directions

Progress in the MINLP arena has been significant in recent years, and we are now able to solve large-scale problems efficiently using a wide variety of approaches. However, MINLP has yet to reach the level of maturity that MIP has achieved. While the MIP community has benefited greatly from *preprocessing* to reduce model sizes and to detect special structure, MINLP technology is still lagging behind. NLP and MINLP preprocessing, similar to global methods, will require the delivery of structural information from the modeling languages. Progress on reliable large-scale NLP codes with restarting capabilities will have an immediate impact on MINLP. Furthermore, combining individual algorithms (e.g. branch-and-bound and extended cutting plane method) with sophisticated search strategies (e.g. non-trivial B&B selection strategies) and heuristics to quickly determine integer solutions will help to close the gap. If research and development continues at the current level of activity, MINLP will soon achieve a stage of maturity enjoyed by the other areas in mathematical programming.

REFERENCES

[1] I.P. Androulakis, C.D. Maranas, and C.A. Floudas (1995). *A Global Optimization Method for Generalized Constrained Nonconvex Problems*, J. Global Optim., **7**, 337–363.

[2] J. Bisschop and M. Roelofs (2002). *AIMMS - The User's Guide*, Paragon Decision Technology B.V., Haarlem, The Netherlands, http://www.aimms.com.

[3] A. Brooke, D. Kendrick, and A. Meeraus (1988). *GAMS: A User's Guide*, The Scientific Press, San Francisco, CA.

[4] M.R. Bussieck, A.S. Drud, and A. Meeraus (2002). *MINLPLib - A Collection of Test Models for Mixed-Integer Nonlinear Programming*, INFORMS J. Comput., to appear.

[5] M.A. Duran and I.E. Grossmann (1986). *An Outer-Approximation Algorithm for a Class of Mixed-integer Nonlinear Programs*, Math. Program., **36**, 307–339.

[6] R. Fletcher and S. Leyffer (1994). *Solving Mixed Integer Programs by Outer Approximation*, Math. Program., **66**, 327–349.

[7] C.A. Floudas (2000). *Deterministic Global Optimization: Theory, Methods and Applications*, Nonconvex Optimization And Its Applications series, **37**, Kluwer Academic Publishers, Boston MA.

[8] C.A. Floudas, P.M. Pardalos et. al. (1999). *Handbook of Test Problems in Local and Global Optimization*, Nonconvex Optimization And Its Applications series, **33**, Kluwer Academic Publishers, Boston MA.

[9] C.A. Floudas (1995). *onlinear and Mixed Integer Optimization : Fundamentals and Applications*, Oxford University Press, New York NY.

[10] J.J.H. Forrest. (2003). personal communication.

[11] R. Fourer, D.M. Gay, and B.W. Kernighan (2003). *AMPL: A Modeling Language for Mathematical Programming*, Duxbury Press, Brooks/Cole-Thomson Publishing Company, Pacific Grove, CA.

[12] GAMS Development Corp. (2003). *GAMS - The Solver Manuals*, GAMS Development Corp., Washington, DC.

[13] A.M. Geoffrion (1972). *A Generalized Benders Decomposition*, J. Optim. Theory Appl., **10** (4), 237–260.

[14] I.E. Grossmann and N.V. Sahinidis (eds) (2002). *Special Issue on Mixed-integer Programming and it's Application to Engineering, Part I*, Optim. Eng., **3** (4), Kluwer Academic Publishers, Netherlands.

[15] I.E. Grossmann and N.V. Sahinidis (eds) (2002). *Special Issue on Mixed-integer Programming and it's Application to Engineering, Part II*, Optim. Eng., **4** (1), Kluwer Academic Publishers, Netherlands.

[16] O.K. Gupta and A. Ravindran (1985). *Branch and Bound Experiments in Convex Nonlinear Integer Programming*, Manage Sci., **31** (12), 1533–1546.

[17] G.R. Kocis and I.E. Grossmann (1989). *Computational Experience with DICOPT: Solving MINLP Problems in Process Systems Engineering*, Computers Chem. Eng., **13**, 307–315.

[18] S. Leyffer (2003). *MacMINLP: Test Problems for Mixed Integer Nonlinear Programming*, http://www-unix.mcs.anl.gov/~leyffer/MacMINLP.

[19] S. Leyffer (1999). *User Manual for MINLP_BB*, University of Dundee Numerical Analysis Report NA/XXX.

[20] MINLP World (2003). http://www.gamsworld.org/minlp.

[21] I. Nowak (2003). *LaGO: A Lagrangian Global Optimizer*, http://www-iam.mathematik.hu-berlin.de/~eopt/index_en.html.

[22] I. Nowak, H. Alperin and S. Vigerske (2003). *LaGO - An object oriented library for solving MINLPs*, submitted for publication in COCOS'02 Conference Proceedings.

[23] I. Quesada and I.E. Grossmann (1992). *An LP/NLP Based Branch and Bound Algorithm for Convex MINLP Optimization Problems*, Computers Chem. Eng., **16** (10/11), 937–947.

[24] L.E. Schrage (2003). personal communication.

[25] L.E. Schrage (1991). *LINDO: An Optimization Modeling System*, Scientific Press series; Fourth ed. Boyd and Fraser, Danvers, Mass.

[26] C.A. Schweiger and C.A. Floudas (1998). *MINOPT: A Modeling Language and Algorithmic Framework for Linear, Mixed-Integer, Nonlinear, Dynamic, and Mixed-Integer Nonlinear Optimization, Version 3.1*, http://titan.princeton.edu/MINOPT/manual.pdf.

[27] SCICON Ltd. (1989). *SCICONIC User Guide Version 1.40*, Scicon Ltd., Milton Keynes, UK.

[28] H. Skrifvars (2003). *mittlp*, http://at8.abo.fi/~hasku/.

[29] M. Tawarmalani, and N.V. Sahinidis (2002). *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications*, Nonconvex Optimization And Its Applications series, **65**, Kluwer Academic Publishers, Boston MA.

[30] Z. Ugray, L. Lasdon, J. Plummer, F. Glover, J. Kelly, and R. Marti (2002). *A Multistart Scatter Search Heuristic for Smooth NLP and MINLP Problems*, INFORMS J. Comput., to appear.

[31] A. Vecchietti, J.J. Gill, L. Catania, and I.E. Grossmann (2003). *GAMS/LogMIP1.0 User's Manual*, http://www.ceride.gov.ar/logmip/eng/documentation/logmip_manual.pdf.

[32] V. Visweswaran and C.A. Floudas (1990). *A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLP's II*, Application of Theory and Test Problems, Computers Chem. Eng., **14** (12), 1417–1434.

[33] T. Westerlund and F. Petersson (1995). *A Cutting Plane Method for Solving Convex MINLP Problems*, Computers Chem. Eng., **19**, 131–136.

[34] T. Westerlund and K. Lundqvist (2003). *Alpha-ECP, Version 5.04. An Interactive MINLP-Solver Based on the Extended Cutting Plane Method*, Report 01-178-A, Process Design Laboratory, Abo Akademi Univeristy, Abo, Finland, http://www.abo.fi/~twesterl/A-ECPManual.pdf.

Bulletin

1. Workshop Announcement

3rd Annual McMaster Optimization Conference: Theory and Applications (MOPTA 03) July 30 - August 1, 2003, McMaster University Hamilton, Ontario, Canada http://www.cas.mcmaster.ca/~mopta

The 3rd annual McMaster Optimization Conference (MOPTA 03) will be held at the campus of McMaster University. It will be hosted by the Advanced Optimization Lab at the Department of Computing and Software and it is co-sponsored by the Fields Institute and MITACS.

SCOPE: The conference aims to bring together a diverse group of people from both discrete and continuous optimization, working on both theoretical and applied aspects. We aim to bring together researchers from both the theoretical and applied communities who do not usually get the chance to interact in the framework of a medium-scale event.

Distinguished researchers will give one-hour long invited talks on topics of wide interest. Invited speakers include:

Laurent El Ghaoui, University of California, Berkeley, CA Lisa K. Fleischer, Carnegie Mellon University, Pittsburg, PA Minyue Fu, University of Newcastle, NSW, Australia Masakazu Kojima, Tokyo Institute of Technology, Tokyo, Japan George Nemhauser, Georgia Institute of Technology, Atlanta, GA Arkadi Nemirovski, TECHNION, Haifa, Israel Stratos Pistikopoulos, Imperial College, London, UK Margaret H. Wright, Courant Institute, New York University, NY

CONTRIBUTED TALKS Each accepted paper will be allotted a 25 minute talk. Authors wishing to speak should submit an abstract via the conference WEB page in ASCII or LaTex source, to terlaky@mcmaster.ca by April 30, 2003. Please use "MOPTA 03" in the email subject line. Notification of acceptance / Program available: May 31, 2003. Deadline for early registration: June 30, 2003.

On behalf of the Organizing Committee Tams Terlaky, terlaky@mcmaster.ca (Chair, McMaster University)

Further information is available at http://www.cas.mcmaster.ca/m̃opta/

2. Workshop Announcement

High Performance Methods for Mathematical Optimization 2003
HPMMO2003
Monday June 23, 2003, Tilburg University, The Netherlands

http://fewcal.uvt.nl/sturm/hpmmo2003/

This short workshop has an informal character. Discussion is stimulated. The main discussion topics are: semidefinite programming, interior point method, spectral bundle method, nonnegative polynomials and sum-of-squares decompositions

3. Call for papers

A Special Issue of the INFORMS Journal on Computing:
http://joc.pubs.informs.org/CallSpecialIssueCompBio.html

Computational Molecular Biology/Bioinformatics
Guest Editor: Harvey J. Greenberg,
Harvey.Greenberg@cudenver.edu
University of Colorado at Denver
Associate Editors:
Dan Gusfield, gusfield@cs.ucdavis.edu,
University of California, Davis
William Hart, wehart@sandia.gov
Sandia National Labs
Giuseppe Lancia, lancia@dei.unipd.it
University of Udine
David Rocke, dmrocke@ucdavis.edu
University of California, Davis
Ying Xu, xyn@ornl.gov
Oak Ridge National Labs

Computational biology, or bioinformatics, has emerged from the Human Genome Project as one of the new frontiers for biomedical research. As problems become better defined, it becomes evident that operations research techniques can be applied with great success. Although some computational, mathematical, and statistical techniques have been used for more than a century, it is the recent explosion of data that has brought this to an information science that requires new algorithms and models to understand complex biological systems. We invite research papers that apply OR to these problems. Some examples and elaboration are at http://joc.pubs.informs.org/CallSpecialIssueCompBio.html

Deadline: August 1, 2003 Expected publication: Fall 2004

You can submit electronically a postscript or pdf file to the Editor, or to any Associate Editor with cc: Harvey.Greenberg@cudenver.edu.

4. Book announcement

Four Colors Suffice:
How the Map Problem Was Solved

Robin Wilson

On October 23, 1852, Professor Augustus De Morgan wrote a letter to a colleague, unaware that he was launching one of the most famous mathematical conundrums in history–one that would confound thousands of puzzlers for more than a century. This is the amazing story of how the "map problem" was solved.

Cloth — 2003 — 24.95 — ISBN: 0-691-11533-8
280 pp. — 5 x 8 — 24 halftones. 173 line illus.

A sample chapter is available at http://pup.princeton.edu/titles/7495.html

5. Book announcement

Least Squares Support Vector Machines

J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle

This book focuses on Least Squares Support Vector Machines (LS-SVMs) which are reformulations to standard SVMs. LS-SVMs are closely related to regularization networks and Gaussian processes but additionally emphasize and exploit primal-dual interpretations from optimization theory. The authors explain the natural links between LS-SVM classifiers and kernel Fisher discriminant analysis. Bayesian inference of LS-SVM models is discussed, together with methods for imposing sparseness and employing robust statistics.

The framework is further extended towards unsupervised learning by considering PCA analysis and its kernel version as a one-class modelling problem.

This leads to new primal-dual support vector machine formulations for kernel PCA and kernel CCA analysis. Furthermore, LS-SVM formulations are given for recurrent networks and control. In general, support vector machines may pose heavy computational challenges for large data sets. For this purpose, a method of fixed size LS-SVM is proposed where the estimation is done in the primal space in relation to a Nystrm sampling with active selection of support vectors. The methods are illustrated with several examples.

Readership: Graduate students and researchers in neural networks; machine learning; data-mining; signal processing; circuit, systems and control theory; pattern recognition; and statistics.

World Scientific, 308pp., Nov. 2002, ISBN 981-238-151-1

http://www.wspc.com/books/compsci/5089.html
http://www.esat.kuleuven.ac.be/sista/lssvmlab/book.html

6. Software Announcement

LS-SVMlab:
Least Squares - Support Vector Machines Matlab/C Toolbox

http://www.esat.kuleuven.ac.be/sista/lssvmlab/

Toolbox: Matlab LS-SVMlab1.4 - Linux and Windows Matlab/C code. Basic and advanced versions. Functional and object oriented interface.

Tutorial User's Guide (100pp.): Examples and demos. Matlab functions with help.

Solving and handling: Classification, Regression; Tuning, cross-validation, fast loo, receiver operating characteristic (ROC) curves; Small and unbalanced data sets; High dimensional input data; Bayesian framework with three levels of inference; Probabilistic interpretations, error bars; hyperparameter selection, automatic relevance determination (ARD); input selection, model comparison; Multi-class encoding/decoding; Sparseness; Robustness, robust weighting, robust cross-validation; Time series prediction; Fixed size LS-SVM, Nystrom method; kernel principal component analayis (kPCA), ridge regression; Unsupervised learning; Large scale problems

Related links, publications, presentations and book: http://www.esat.kuleuven.ac.be/sista/lssvmlab/

Chairman's Column

First, I would like to thank our new editor Jos Sturm and the guest editors Sven Leyffer and Jorge Nocedal for this excellent edition of Views-and-News. This is an exciting and interesting issue on the state-of-the-art of several areas of optimization.

The previous SIAG/OPT Views-and-News dated Feb./2002 contained comments on 9/11 and on the quote *the world has changed*. Since then we have had an extremely successful Seventh SIAM Conference on Optimization in Toronto, Canada. Worries about lack of participants were unfounded. Here is the history of total attendance at our meetings:

- June 12-14, 1984, Boulder, CO, attendance 248
- May 17-20, 1987, Houston, TX, attendance 399
- April 3-5, 1989, Boston, MA, attendance 468
- May 11-13, 1992, Chicago, IL, attendance 440
- May 20-22, 1996, Victoria, BC, Canada, attendance 465
- May 10-12, 1999, Atlanta, GA, attendance 442
- May 20-22, 2002, Toronto, Canada , attendance 414.

Plenary lectures and other information are available at the conference web site.

However, we are now in the middle of one of the consequences of 9/11, war in Iraq. Changes in our world are rapid. As this article is being written, the war appears to be over. New alliances appear to be forming. Who would believe that France-Germany-Russia would meet to discuss strategy to counter an American-British led alliance? However, this is not a forum to discuss politics. But, as we see divisions grow between many countries, history suggests that the global openness and freedom that we are enjoying now may be threatened. Border restrictions are threatening international travel. What does this mean for our community? Will it threaten scientific cooperation between countries? Will sanctions and boycotts come into the scientific community?

My hope is that it does not. I hope that scientific cooperation and travel between all countries remains open. I hope that SIAM continues to have international meetings at locations around the world.

The next big meeting in Optimization is in Copenhagen, 18th International Symposium on Mathematical Programming, ISMP 2003. From the preliminary list of registrants it appears that this meeting will be large and international. Let us hope that this is an indicator of the future.

Henry Wolkowicz, SIAG/OPT Chair
University of Waterloo
Department of Combinatorics and Optimization
Waterloo, Ontario
CANADA N2L 3G1
hwolkowicz@uwaterloo.ca
http://orion.math.uwaterloo.ca/~hwolkowi

Comments from the editor

This is the first Views-and-News issue that appears under my editorship. I have maintained the style as developed by Juan Meza and his predecessors.

We have negotiated with SIAM that printed issues of the newsletter continue to be shipped to our SIAG's members as before. If you are reading this newsletter in its printed form, please also take a look at the electronic version, available from http://fewcal.uvt.nl/sturm/siagopt/.Regardless of whether you prefer reading in print or not, I believe that we will all appreciate the unique benefits of the electronic version. The newsletter has been compiled using hyperref, a LaTeX package producing clickable links to resources, webpages, and references.

I would like to thank the guest editors of this thematic issue on **Large Scale Non-Convex Optimization**, Sven Leyffer and Jorge Nocedal. They have really delivered an issue that is of high interest and also very accessible to the readership of *Views-and-News*. I expect that this issue will both increase awareness of new developments in this area, and attract new researchers to work on non-convex optimization. For this, I also like to thank the contributing authors.

I hope that you share the enthusiasm regarding this issue, and keep sending me material for publication in this newsletter. In particular, I am much less in touch with the applied world than the previous editor, and I very much welcome short contributed articles that outline a case study in which optimization played a crucial role. I need your help with this.

Jos F. Sturm, Editor
Tilburg University
P.O. Box 90.153
NL-5000 LE Tilburg
The Netherlands
j.f.sturm@uvt.nl
http://fewcal.uvt.nl/sturm