Interior-Point Algorithms: First Year Progress Report on NSF Grant DDM-9207347

Yinyu Ye

October 19, 1995

1 Introduction

This document is the first year progress report on an optimization project funded by NSF Grant DDM-9207347. The goal of the project is to analyze and develop interior-point algorithms for solving large-scale linear, quadratic, and convex optimization problems that arise in production management and resource allocation. It is a continuation of our previous projects, and it is motivated by the demand to resolve several theoretical and practical issues concerning interior-point algorithms. In this project we do research to strengthen some results and to explore new techniques for analyzing and improving interior-point algorithms. We also do research on implementation and application for solving problems in production scheduling and network planning. Our research topics include the following.

1. Termination and Local Convergence: To develop efficient termination techniques to obtain an (exact) optimal solution, and to design superlinearly and quadratically convergent interior-point algorithms.

2. Condition-Based Complexity: To study complexity analysis of interior-point algorithms for solving linear programming (LP) problems under the real number model, and to analyze "condition" numbers that determine the degree of difficulty of these problems.

3. Strongly Polynomial Algorithm: To improve the "condition" of LP problems, and to develop strongly polynomial interior-point algorithms for solving some LP problems.

4. Average Complexity Analysis: To analyze interior-point algorithms for solving randomly generated LP problems, and to explain the excellent practical behavior of these algorithms.

5. Decomposition and Column Generation: To develop various decomposition and column generation schemes for solving large-scale or infinite-dimensional practical optimization problems.

6. Implementation and Application: To implement the state of the art interior-point algorithm, and to apply the algorithm to solving real-world problems.

Progresses have been made in each of these areas. The project is going well in general. Most of the first-year objectives have been met. Since the last report, several new working papers have been resulted from the project: 5 of them have been accepted for publication in *Mathematical Programming* or *SIAM J. on Optimization*, or *Mathematics of OR*. A implementation code is also developed and some computational experiments are conducted. Below, I provide some partial results and discuss ongoing research activities.

2 Project Progress

2.1 Progress in Local Convergence

Quadratic convergence for general LP, assuming neither convergence of the iteration sequence nor nondegeneracy, was first established by Ye, Güler, Tapia and Zhang [21]. The algorithm was based on the predictor-corrector algorithm of Mizuno et al. [9]. A stronger result was developed later where we showed how to construct an $O(\sqrt{n}L)$ algorithm whose Q-order of convergence exactly equals 2. These results are not trivial since the Jacobian matrix of the KKT equations for LP tends towards singular in solving degenerate LP problems.

While almost every result concerning interior-point algorithms for LP has been immediately extended to the monotone linear complementarity problem (LCP) or convex quadratic programming (QP), the extension of the quadratic convergence result seems very hard. The main reason is a fundamental difference between LP and LCP: LCP (or QP) may not have a strict complementarity solution but LP always has, and the existence of a strict complementarity solution plays an important role in the local convergence analysis.

This fact presents certain challenges to developing an efficient termination criterion or superlinearly convergent algorithm for LCP or QP.

But this problem has been completely resolved by a recent work of Anstreicher and myself [20], where we proved a quadratically convergent predictor-corrector algorithm for the LCP possessing a strictly complementary solution. Moreover, we proved it is impossible for the available polynomial interior-point algorithms to achieve superlinear convergence for the LCP that does not possess a strictly complementary solution.

2.2 Progress in Condition-Based Complexity

Vavasis and myself [W2] developed a condition-based complexity analysis of homogeneous polyhedra with real number data. We analyzed the dependency of primal-dual interior point algorithm efficiency on this condition number for finding a feasible point in a polyhedron. Consider the following polyhedron:

$$\mathcal{P} = \{\mathbf{x} : \ A\mathbf{x} = \mathbf{0}, \ \mathbf{e}^T\mathbf{x} = 1, \ \mathbf{x} \geq \mathbf{0}\},$$

where $A \in \mathbf{R}^{m\times n}$ with rank m is given, $\mathbf{e}$ is the vector of all ones, and ^{T} denotes transpose. This is the homogeneous form proposed by Karmarkar [3]. $\mathcal{P}$ is said to be feasible if and only if $\mathcal{P} \neq \emptyset$. Given an A, there is unique partition of the columns of A, $A = (B, N)$, such that the set

$$\mathcal{P}_p = \{\mathbf{x}_B : \ B\mathbf{x}_B = \mathbf{0}, \ \mathbf{e}^T\mathbf{x}_B = 1, \ \mathbf{x}_B \geq \mathbf{0}\},$$

has a strictly feasible point or an interior point in the positive orthant, and the dual set

$$\mathcal{P}_d = \{(\mathbf{y}, \mathbf{s}) : \ B^T\mathbf{y} = \mathbf{0}, \ N^T\mathbf{y} \leq \mathbf{0}, \ \mathbf{e}^T\mathbf{s}_N = 1, \ \mathbf{s} = -A^T\mathbf{y}\},$$

has a strictly feasible point, i.e., a feasible $(\mathbf{y}, \mathbf{s})$ with $\mathbf{s}_N = -N^T\mathbf{y} > \mathbf{0}$. (We use $\mathbf{s}$, the slack vector, to simplify notations.) It is also known that $\mathbf{x}_N = \mathbf{0}$ for any $\mathbf{x} \in \mathcal{P}$. Thus, $\mathcal{P}$ is infeasible if and only if $A = N$.

Suppose we want to answer the following question using an interior point method:

$$(P) \quad \text{Is} \quad \mathcal{P} \quad \text{feasible?}$$

Consider a condition number of A, defined as

$$\begin{array}{rll} \sigma_p & = \min_{j\in B}\{\max \quad x_j : & \mathbf{x}_B \in \mathcal{P}_p\} \\ \sigma_d & = \min_{j\in N}\{\max \quad s_j : & (\mathbf{y}, \mathbf{s}) \in \mathcal{P}_d\} \\ \sigma(A) & = \min(\sigma_p, \sigma_d). & \end{array} \qquad (1)$$

(we assign σ_p (σ_d) to 1 if B (N) is null.) It has been shown that the interior-point algorithm, such as [2][4][10][11][13] generates a sequence of partitions $(B^k, N^k) = A$ such that, after $O(\sqrt{n}(|\log\sigma(A)| + \log n))$ iterations, we have convergence to $B^k = B$ and $N^k = N$ (see [18]). It seems that this result could be used to answer question (P) to determine whether $A = N$ or not. However, this requires $\sigma(A)$ as *a priori* information. Without knowledge of $\sigma(A)$, we have to provide a witness whether $A = N$ or not.

Also in [18], we developed a least squares projection technique to demonstrate that a witness that (P) is infeasible can be also found in $O(\sqrt{n}(|\log\sigma(A)| + \log n))$ iterations. Similarly, the case $A = B$ can be proved in the same number of iterations. In fact, all other cases can be completed in about the same number of iterations to obtain (B, N), and to generate feasible points in $\mathcal{P}_p$ and $\mathcal{P}_d$, respectively. Thus, $\sigma(A)$ represents a measure of difficulty in solving (P): the smaller $\sigma(A)$, the harder the problem.

We also relate $\sigma(A)$ to another widely used condition number of A. Let A be an $m \times n$ matrix, and let $\|\cdot\|$ be some p-norm. Let $\mathcal{D}$ be the set of all positive definite $n \times n$ diagonal matrices. Let

$$S = \{\mathbf{s} \in \mathbf{R}^n : \|\mathbf{s}\| = 1 \text{ and } \mathbf{s} = A^T\mathbf{y} \text{ for some } \mathbf{y} \in \mathbf{R}^m\}.$$

Let

$$X = \{\mathbf{x} \in \mathbb{R}^n : AD\mathbf{x} = \mathbf{0} \text{ for some } D \in \mathcal{D}\}.$$

Define

$$\rho_0(A) = \inf\{\|\mathbf{s} - \mathbf{x}\| : \mathbf{x} \in X, \mathbf{s} \in S\}. \tag{2}$$

Stewart [15] has shown that for any nonzero matrix A, $\rho_0(A) > 0$.

We now define $\bar{\chi}(A) = 1/\rho_0(A)$. In the case that A is full rank, there is an alternative definition by Stewart [15], O'Leary [12]: Let A be an $m \times n$ matrix of rank n. Then

$$\bar{\chi}(A) = \sup\{\|A^T(ADA^T)^{-1}AD\| : D \in \mathcal{D}\}. \tag{3}$$

This quantity has been independently analyzed by Todd [16].

We explore the relation between $\sigma(A)$ and $\bar{\chi}(A)$. More specifically, we have shown

$$\sigma(A) \geq \frac{1}{\bar{\chi}(A) + 1}. \tag{4}$$

Therefore, to solve problem (P) we need at most $O(\sqrt{n}(\log\bar{\chi}(A) + \log n))$ interior-point algorithm iterations.

Finally, if A is rational, we provide a bound for $\bar{\chi}(A)$ in terms of size of A: Let A be rational and L be its bit size. Then

$$\bar{\chi}(A) \leq 2^{O(L)}.$$

This establishes that rational LP problems can be solved in polynomial time in the size of the input data. Furthermore, $\bar{\chi}(A)$ is bounded above by a polynomial in the dimension for some LP problems, such as network flow problems. Thus, these problems can be solved in strongly polynomial time by certain interior-point algorithms.

2.3 Progress in Strongly Polynomial Algorithm

In the previous topic, we have shown that the complexity of finding an interior point for a homogeneous polyhedron is bounded by $\log \bar{\chi}(A)$. This result can be generalized to nonhomogeneous polyhedra. The difficulty with this generalization is that appending $\mathbf{b}$ as a column of A could increase the value of $\bar{\chi}(A)$. This increase is particularly undesirable for problems like flow problems, in which the constraints have the form $A\mathbf{x} = \mathbf{b}$ with small integers entries for A but arbitrary real numbers in the right-hand side vector. It is not hard to construct nonhomogeneous problems with near-degeneracies that make σ_p and σ_d arbitrarily small, independent of A.

Recently, Vavasis and I [W1] develop a "layered-step" interior point (LIP) algorithm for linear programming. This algorithm follows the central path, either with short steps or with a new type of step called a "layered least squares" (LLS) step. The algorithm returns the exact global minimum after a finite number of steps—in particular, after $O(n^{3.5}c(A))$ iterations, where $c(A)$ is a function of the coefficient matrix. The LLS steps can be thought of as accelerating a path-following interior point method whenever near-degeneracies occur. One consequence of the new method is a new characterization of the central path: we show that it composed of at most n^2 alternating straight and curved segments. This is the first such result for LP under the real number model. If the LIP algorithm is applied to integer data, we get as another corollary a new proof of a well-known theorem by Tardos that linear programming can be solved in strongly polynomial time provided that A contains small-integer entries.

2.4 Progress in Average Analysis

Anstreicher, Ji, Potra and myself [P1][19] have developed a rigorous average analysis of interior-point algorithms. We have shown that the expected number of iterations required to solve a random LP problem is bounded above by $O(\sqrt{n}\log(n))$, where n is the number of variables in the standard LP form:

$$\min \quad c^T x, \quad s.t. \quad Ax = b, \quad x \geq 0.$$

To this date our probabilistic analysis is essentially focused on the termination criteria for interior-point algorithms. No effort is made to evaluate the improvement on every iteration of an interior-point algorithm for random LP problems. As we mentioned before, some non-rigorous probabilistic analysis of the reduction of the objective function is discussed in Mizuno et al. [9], where the result holds rigorously for just the first iteration. We plan to expand our study to include the average reduction of the objective function for every iteration.

2.5 Progress in Column Generation

Methods, which allow for column generation—the addition of columns or constraints to the system only when needed—permit a great deal of flexibility for solving optimization problems in which the number of columns is either very large or not explicitly known. Thus, we need to develop interior-point algorithms that allow decomposition or column generation and are implementable with practical efficiency. This will advance the practical solvability for a large class of optimization problems, such as semi-infinite programming, convex nonlinear programming, and combinatorial optimization.

Several approaches for column generation or decomposition were proposed and developed. One method, the analytic center cutting plane method was developed and analyzed by several people, including myself, (see [1][14][17]). However, no complexity or convergence result has ever been established for this method in solving general convex problems. In practice, Goffin et al. [1] and Mitchell[8] have reported encouraging computational results in a decomposition algorithm similar to ours. The number of inequality constraints generated is virtually independent of the total number of inequality constraints in the original problem.

Recently, Goffin, Luo and myself [W3] finally established a convergence result for this method. In fact, we proved that the algorithm is a pseudo-polynomial algorithm for solving general convex optimization problems.

2.6 Progress in Implementation and Application

In [W4], we have developed a simplification and generalization of the recent Ye-Todd-Mizuno homogeneous and self-dual linear programming (LP) algorithm [P2][P3]. The simplified and generalized algorithm does not use any Big-M initial point while achieves $O(\sqrt{n}L)$-iteration, the theoretically best, complexity for LP. It solves the linear programming problem without any regularity assumption concerning the existence of optimal, feasible, or interior feasible solutions. It also detects LP infeasibility based on a provable criterion. Its preliminary implementation with a simple predictor and corrector technique results in an efficient computer code in practice. In contrast to other interior-point methods, our code solves NETLIB problems, feasible or infeasible, starting simply from $x = e$ (primal variables), $y = 0$ (dual variables), $z = e$ (dual slack variables), where e is the vector of all ones.

We describe our computational results in solving NETLIB feasible problems, with a comparison to OB1.60, the state of the art implementation of interior-point algorithms [5] [7]. Very recently, a set of infeasible LP problems has joined NETLIB collection. Some of them are quite interesting such as CPLEX2, an almost-feasible problem contributed by CPLEX Optimization Inc. It is a major problematical issue to detect infeasibility for practical interior-point methods. Some theoretical results hold only for feasible cases. Other approaches to detect infeasibility are somewhat difficult to implement in practice according to Lustig et al. [6]. Thus, in addition to solving NETLIB feasible problems, we also test our code in solving these infeasible problems.

Starting simply from $x = e$, $y = 0$, $z = e$ (slack of dual) and using the default values for program parameters, our code successfully solves or detects infeasibility of all encountered test problems in NETLIB. The technical details of our work are included in a technical report [W4].

We run both OB1 and our implementation code on a APOLLO 720 workstation. The Computational results are reported in the two Tables of [W4], where "Rows" and "Columns" represent the number of rows and columns of the constraint matrix of the named LP problem, respectively. "Nonzeros" represent the number of nonzero entries in the matrix. "B" indicates that bounded (from both sides) variables exist; "R" indicates that bounded (from both sides) constraints exist.

The reported cpu time (in seconds) of OB1 includes preprocessing, such as matrix cleanup, ordering and symbolic factorization, and arithmetic operations till solver terminates. However, we exclude any efforts required by OB1 to translate files such as MPS input and output, either at the beginning or in the middle of the run. Default parameters are used (except for WOOD1P and WOODW). The reported cpu time (in seconds) of our code includes all operations, except for MPS file input and output. All default parameters are used. All of NETLIB feasible problems are successfully solved except

DFL001 and FIT2P. The sizes of these two problems are too large for our machine memory capacity.

Table I contains detailed computational results for solving NETLIB feasible problems. I_O and I_H represent the number of iterations used by OB1 and our code, called HLF, respectively; T_O and T_H are cpu times used by OB1 and HLF, respectively. I_{H-O} is the difference in the number of iterations between OB1 and HLF. Positive sign indicates that OB1 requires less number of iterations, while negative sign means that HLF uses less. The last column is the cpu time comparison. Again, positive rate represents how much OB1 is faster than HLF, and negative rate indicates how much HLF is faster than OB1.

For several problems, such as FIT1D, FIT1P, FIT2D, our code takes much more time per iteration. This is because these problems have many variables with upper bounds. OB1 handles upper bound implicitly and intelligently. But our technique, i.e, explicitly including upper bounds into constraints, results in a very big constraint matrix in contrast to the original A, and results in slow performance. For problems WOOD1P and WOODW, as suggested by Lustig (private communication), we reset "DEPENDENT NO" in the specification input file for OB1. Otherwise, due to data structure of their coefficient matrix, it will take significant cpu time to perform Gauss elimination, which is defaultly employed by OB1.60 to detect dependent rows. Among 91 problems, our code performs better than OB1 in cpu time for 63 problems.

Table II includes computational results for solving NETLIB infeasible problems. There are total 30 problems in the set of NETLIB infeasible problems. Two of them, CERIA3D and CPLEX1, have dense columns of near to or more than thousand nonzero entries, which causes space trouble for our code to handle $A\Theta A^T$. (The dense column treatment technique has not been employed in our current implementation.) For all other problems, infeasibility is successfully detected. Particularly, CPLEX2, an almost feasible problem, contributed by CPLEX Optimization Inc., is a very interesting problem for testing infeasibility detection capability. At iteration 15 of our run, both primal and dual infeasibility are less than 10^{-7}, and primal and dual objective values share same first 6 digits. Finally, its infeasibility is detected at iteration 44.

It is well known that preprocessing (or presolve) does an important job in many up-to-date LP codes such as simplex code CPLEX and interior-point code OB1. For instance, the presolver employed in OB1.60 can remove more than 1/3 rows and columns of problem AGG. However, almost no preprocessing is employed in our implementation at this moment. Thus, our results are very preliminary. By employing data preprocessing, as well as a technique for bounded and free variables, a numerical procedure for dense columns, and a finite termination scheme for optimal bases, we expect a powerful and complete implementation for solving LP in the near future.

We plan to use this code for solving some LP problems arising in our joint project with MCI. We have made a preliminary progress in this application project.

3 Education and Human Resources

Some beneficiaries, educationally, of the project are the Ph.D. students at the University of Iowa. We have supported several Ph.D. students at UI from this NSF grant. Some results mentioned above are part of their Ph.D. theses.

Another outgrowth of our research has been a new seminar, meeting once a week, among graduate students and faculty members from several departments. We have also brought some of the new research topics and results to the optimization and mathematical programming courses at the graduate level.

4 Related Publication and Working Papers

4.1 New Publication

[P1] K. Anstreicher, J. Ji, F. Potra, and Y. Ye, "Average performance of a self-dual interior-point algorithm for linear programming," in P. Pardalos eds., *Complexity in Numerical Optimization* (World Scientific, New Jersey, 1993) pp. 1-15.

[P2] Y. Ye, M. J. Todd, and S. Mizuno, "An $O(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm," Working Paper No. 92-11, College of Business Administration, The University of Iowa (1992), to appear in *Mathematics of Operations Research.*

[P3] S. Mizuno, M. Todd, and Y. Ye, "A surface of analytic centers and infeasible-interior-point algorithms for linear programming," Technical Report No. 1037, School of ORIE, Cornell University (1993), to appear in *Mathematics of Operations Research.*

[P4] J. Ji and Y. Ye, "A complexity analysis for interior-point algorithms based on Karmarkar's potential functions," Working Paper No. 90-5, College

of Business Administration, The University of Iowa (1990), to appear in *SIAM J. on Optimization.*
[P5] Y. Ye, "Combining binary search and Newton's method to compute real roots for a class of real functions," Working Paper No. 91-7, College of Business Administration, The University of Iowa (1991), to appear in *Journal of Complexity.*

4.2 New Working Papers

[W1] S. Vavasis and Y. Ye, "An accelerated interior-point method whose running time depends only on A," TR 93-1391, Department of Computer Science, Cornell University (Ithaca, NY, 1993).
[W2] S. Vavasis and Y. Ye, "Condition numbers for polyhedra with real number data," (S. Vavasis and Y. Ye), TR 93-1398, Department of Computer Science, Cornell University (Ithaca, NY, 1993).
[W3] J. Goffin, Z. Luo and Y. Ye, "Complexity analysis of a potential reduction and column generation algorithm for convex programming," Technical Report, Faculty Management, McGill University (Montreal, Canada, 1993).
[W4] X. Xu, P. Hung, and Y. Ye, "A simplified homogeneous and self-dual linear programming algorithm and its implementation," Working Paper, College of Business Administration, The University of Iowa (Iowa City, IA, 1993),

References

[1] J.-L. Goffin, A. Haurie, and J.-P. Vial, "Decomposition and nondifferentiable optimization with the projective algorithm", *Management Science* **38** (1992) 284–302.

[2] C. C. Gonzaga, An algorithm for solving linear programming problems in $O(n^3L)$ operations, In N. Megiddo (ed.): *Progress in Mathematical Programming* (Springer-Verlag, New York, 1989).

[3] N. Karmarkar, A new polynomial-time algorithm for linear programming, *Combinatorica* 4 (1984) 373-395.

[4] M. Kojima, S. Mizuno and A. Yoshise, A polynomial-time algorithm for a class of linear complementarity problems, *Mathematical Programming* 44 (1989) 1-26.

[5] I. J., Lustig, R. E. Marsten, and D. F. Shanno, "On implementing Mehrotra's predictor-corrector interior point method for linear programming" *SIAM Journal on Optimization* 2 (1992) 435-449.

[6] I. J., Lustig, R. E. Marsten, and D. F. Shanno, "Computational Experience with a Globally Convergent Primal-Dual Predictor-Corrector Algorithm for Linear Programming" Technical Report SOR 92-10, Dept. of Civil Engineering and Operations Research, Princeton University (Princeton, NJ, 1992).

[7] S. Mehrotra "On the implementation of a (primal-dual) interior point method," *SIAM Journal on Optimization* 2 (1992) 575-601

[8] J. E. Mitchell, "Karmarkar's algorithm and combinatorial optimization problems," Ph.D. Thesis, Department of ORIE, Cornell University (Ithaca, NY, 1988).

[9] S. Mizuno, M. J. Todd and Y. Ye, "On adaptive-step primal-dual interior-point algorithms for linear programming," *Mathematics of Operations Research* 18 (1993) 964-981.

[10] R. C. Monteiro and I. Adler, Interior path following primal-dual algorithm. Part I: Linear programming, *Mathematical Programming* 44 (1989) 27-42.

[11] Ju. E. Nesterov and A. S. Nemirovsky. *Self-concordant functions and polynomial-time methods in convex programming*, USSR Academy of Sciences, Central Economic and Mathematics Institute (Moscow, USSR, 1989).

[12] D. P. O'Leary, "On bounds for scaled projections and pseudoinverses," *Linear Algebra and its Applications* 132 (1990) 115-117.

[13] J. Renegar, A polynomial-time algorithm, based on Newton's method, for linear programming, *Mathematical Programming* 40 (1988) 59-93.

[14] G. Sonnevend, "New algorithms in convex programming based on a notion of "centre" (for systems of analytic inequalities) and on rational extrapolation", in K. H. Hoffmann, J. B. Hiriat-Urruty, C. Lemarechal, and J. Zowe, editors, *Trends in Mathematical Optimization: Proceedings of the 4th French–German Conference on Optimization in Irsee,*

West–Germany, April 1986, volume 84 of *International Series of Numerical Mathematics*, pages 311–327. Birkhäuser Verlag, Basel, Switzerland, 1988.

[15] G. W. Stewart, "On scaled projections and pseudoinverses," *Linear Algebra and its Applications* 112 (1989) 189-193.

[16] M. J. Todd, "A Dantzig-Wolfe-like variant of Karmarkar's interior-point linear programming algorithm," *Operations Research* 38 (1990) 1006-1018.

[17] Y. Ye, "A potential reduction algorithm allowing column generation," *SIAM J. on Optimization* 2 (1992) 7-20.

[18] Y. Ye, "On the finite convergence of interior-point algorithms for linear programming," *Mathematical Programming* 57 (1992) 325-335.

[19] Y. Ye, "Toward probabilistic analysis of interior-point algorithms for linear programming," Working Paper, Department of Management Science, The University of Iowa (Iowa City, IA, 1991), to appear in *Mathematics of Operations Research.*

[20] Y. Ye and K. Anstreicher, "On quadratic and $O(\sqrt{n}L)$ convergence of a predictor- corrector algorithm for LCP," Working Paper No. 91-20, College of Business Administration, The University of Iowa (1991), to appear in *Mathematical Programming.*

[21] Y. Ye, O. Güler, R. A. Tapia and Y. Zhang, "A quadratically convergent $O(\sqrt{n}L)$-iteration algorithm for linear programming," *Mathematical Programming* 59 (1993) 151-162.