

# SANDIA REPORT

SAND99-8201

Unlimited Release

Printed November 1998

## Parallel Computational Chemistry Using Constraints: Final Report, LDRD 97-0301, Case 3504140000

Todd D. Plantenga

Prepared by

Sandia National Laboratories

Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,  
a Lockheed Martin Company, for the United States Department of  
Energy under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.



**Sandia National Laboratories**

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

**NOTICE:** This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from  
Office of Scientific and Technical Information  
P.O. Box 62  
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from  
National Technical Information Service  
U.S. Department of Commerce  
5285 Port Royal Rd  
Springfield, VA 22161

NTIS price codes  
Printed copy: A03  
Microfiche copy: A01



SAND99-8201  
Unlimited Release  
Printed November 1998

## Parallel Computational Chemistry Using Constraints: Final Report, LDRD 97-0301, Case 3504140000

Todd Plantenga\*  
Scientific Computing Department  
Sandia National Laboratories  
Livermore, CA

November 4, 1998

### ABSTRACT

Computer modeling to estimate material properties, design chem/bio sensors, and evaluate protein-protein interactions all require solving force field equations for molecular structures that contain tens of thousands of covalently connected atoms. Potential energy minimization is a key step in the calculation, but stiff covalent bonding forces make optimization difficult and expensive. This two-year LDRD developed two classes of advanced minimization algorithms that were specialized for chemistry applications and distributed computing machines. The project led to two successful algorithms that were implemented in three Sandia computational chemistry codes to support various users.

---

\*email: [tdplant@ca.sandia.gov](mailto:tdplant@ca.sandia.gov)

## Acknowledgment

The author acknowledges support by the Department of Energy through Sandia National Laboratories. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

# Parallel Computational Chemistry Using Constraints: Final Report, LDRD 97-0301, Case 3504140000

## 1 Introduction

This report summarizes the results of a two-year LDRD effort. The LDRD proposed to design and study new algorithms for providing a potential energy minimization capability in Sandia computational chemistry codes. The effort led to collaborations between computational chemists and optimization algorithm specialists. In the end, two advanced algorithms were implemented in three Sandia codes: LAMMPS [1], CCEMD [2], and DEMoS\*.

Technical accomplishments of this project are described in depth in other Sandia publications [3, 4, 5]. This report provides a comprehensive description of the diverse activities funded by the LDRD, and their likely impact on future projects. Recent project information can be viewed on the Internet at <http://midway.ca.sandia.gov/~tdplant/compchem/main.html>.

## 2 Technical Goals and Achievements

Energy minimization is an essential step for studying polymer material properties, chem/bio sensors, and protein-protein docking. Chemistry studies are usually based on dynamic simulations, and these require equilibrium structures of minimal energy before they can be started. For example, the rate of diffusion for water through a butyl rubber O-ring can be estimated by constructing butyl rubber samples and simulating the motion of water molecules down a concentration gradient [15]. Polymer samples are constructed from statistical rules and populated with randomly placed water molecules. The samples invariably have regions of compression or distortion that must be “relaxed” before a realistic simulation can be executed. Energy minimization is the most effective means of relaxing molecules into a natural conformation. Another important application of minimization is to characterize conformational possibilities between interacting molecules. Many chemical reaction properties depend on the structural shape of the interacting molecules. At room temperature the shapes change continually by hopping between energy minima that are not separated by high energy barriers. Thus, an interaction can be studied if relevant local energy minima are identified.

Potential energy minimization is computationally difficult when atoms are covalently connected, as is the case with all proteins and polymers. Covalent bonding forces are modeled as stiff mechanical springs, but much weaker long range coulomb forces are the prime determinant of three-dimensional structure. The force mismatch causes numerical ill-conditioning that becomes worse as molecules become larger. Standard minimization techniques do not perform well on ill-conditioned systems, and energy minimization is often viewed as the bottleneck in large-scale computational studies. Special minimization techniques for small molecules have been investigated by other researchers [6, 7, 8, 9, 10, 11, 12], but not extended to the larger and more ill-conditioned problems considered in this project.

The goals of this project were to develop state-of-the-art, large-scale optimization methods appropriate for ill-conditioned potential energy modeling, apply them to distributed machines,

---

\*Carl Melius, SNL/CA, Dept. 8130, email: melius@sandia.gov.

and provide a practical minimization capability for computational chemistry applications at Sandia. Two different algorithmic paths were taken to meet these objectives: a Newton-like method called TNCG which reduces ill-conditioning with indirect curvature information, and a constrained optimization approach called SQP that replaces covalent bond “springs” with semi-rigid rods. Together, the two algorithms provide complementary solutions for molecular calculations on various machines.

## 2.1 Unconstrained TNCG algorithm

The project originally addressed two Sandia codes for computational chemistry: CCEMD [2], a C language program for workstations, and LAMMPS [1], a Fortran program for data-parallel machines such as the Intel Teraflop and Paragon. CCEMD is used in Dept. 8130 to examine proteins for chem/bio sensors and drug docking applications. LAMMPS is used in Depts. 9221 and 9225 to study polymer materials and biological membranes. Originally, both codes relied on a “textbook” nonlinear conjugate gradient algorithm [13, §4.1] for energy minimization. This method is easy to implement and has small memory requirements (a major consideration for systems with thousands of atoms), but is predictably slow on ill-conditioned problems and occasionally fails to converge at all. CCEMD also contained a quasi-Newton, DFP algorithm that performed better, but required excessive memory on large problems.

A properly implemented Newton-like algorithm should converge faster and more reliably. Newton methods construct a quadratic model of the energy potential using first and second partial derivatives; however, an  $n$ -atom system has  $O(n^2)$  second derivatives, a prohibitively large number when  $n$  exceeds 1000. Conjugate gradient provides an indirect way of handling second derivatives without actually computing them. With proper safeguarding, the conjugate gradient inner iteration can be stopped after a few loops to save time. It produces an approximate Newton solution, and the algorithm is known as a “truncated” Newton method utilizing conjugate gradient (TNCG). Reliable convergence is obtained by adding a trust region constraint [14]. Table 2.1 provides an illustration of how much faster energy minimization was accomplished with the new method.

Versions of TNCG have appeared in the literature [6, 7, 8], but only for single processor machines. The TNCG algorithm is organized around vectors that correlate strongly with force vectors computed from the potential energy; hence, TNCG can be implemented on parallel

Table 2.1: Comparison of Algorithms on CCEMD

<i>Molecule</i>	<i>Num atoms</i>	<i>Old NLCG</i>	<i>Old DFP</i>	<i>New TNCG</i>
P21	172	3,600 sec	600 sec	770 sec
1ROP	455	15,000 sec	9,020 sec	4,990 sec
3CHY	979	49,700 sec	54,100 sec	12,100 sec
153L	1432	202,000 sec	†	33,600 sec

The new TNCG minimization algorithm developed under this project converges more quickly than the old algorithms in CCEMD. NLCG is nonlinear conjugate gradient, easy to implement but inefficient. DFP is the Davidon-Fletcher-Powell quasi-Newton method, which performs well on small molecules but requires  $O(n^2)$  memory; consequently, it runs much slower on large problems. TNCG has solved problems with 50,000 atoms in its distributed processing LAMMPS form.

† Still not converged after 400,000 seconds.

processors by following the force vector distribution pattern. This was done to create a parallel version of TNCG for LAMMPS. The algorithm was tuned for optimal performance on the Teraflop machine, and converged reliably while consuming only 5% of problem execution time. SAND report 98-8201 [3] describes this work in detail.

Recently, the TNCG algorithm was also incorporated into the next generation Sandia computational chemistry code, DEMoS (Distributed Extensible Molecular Simulation). The algorithm was redesigned as Java classes that interface with the InDEPS framework on which DEMoS is based. Acceptance testing is currently underway. The redesign effort was small (done by a summer intern), thanks in part to the use of object oriented software engineering principles. The TNCG algorithm now exists as a C program in CCEMD for workstations, a Fortran subroutine in LAMMPS for the Teraflop, and a Java code in DEMoS for heterogeneous distributed architectures.

TNCG is robust and scalable, and has enabled computations for polymers at the mesoscopic scale that were previously considered intractable [15]. Nevertheless, convergence to a minimum still takes a long time due to the extreme ill-conditioning in large problems. In 1998 a collaboration was established with the Optimization Technology Center (OTC) at Argonne National Laboratory and Northwestern University to explore enhancements based on limited memory BFGS approximations (L-BFGS). An experimental L-BFGS preconditioner [16] was added to the inner conjugate gradient loop of TNCG. Initial testing shows a 3- to 5-fold speedup, and the design should adapt easily for parallel computing [5]. Detailed testing of this option is continuing.

## 2.2 Constrained SQP algorithm

A second class of optimization algorithm was developed in the project to address ill-conditioning in a different way. As mentioned earlier, the conditioning results from a mismatch in force magnitudes between long range coulomb interactions and short range covalent bonds. Bonds are modeled as stiff mechanical springs that cause small, high speed oscillations during minimization. The shape of a minimum energy structure is ultimately determined by weaker long range forces, but their action is perturbed by the spring forces. The stiffness is such that springs in a minimum energy structure are rarely stretched more than a slight amount from their natural rest position; hence, a reasonable approximation is to replace the springs with rigid rods that fix the bonds in nominal positions. It turns out that even when all covalent bonds and valence angles are fixed, polymer molecules still have enough rotational degrees of freedom to reasonably model chemical interactions. Fixed bonds are routinely used in molecular dynamics simulations [17, §3.4], and they beneficially reduce the search space of possible conformations to a physically relevant subset [18].

Dynamics simulations commonly obtain only a 3-fold speedup, and it is usually assumed energy minimization can do no better. The new algorithm developed in this project achieved a 4- to 8-fold speedup because it allowed rigid rods to flex during intermediate stages of minimization. The new algorithm also dealt with scaling issues more successfully than conventional ideas [9, 10, 11, 12], though difficulties remain.

Mathematically, rigid rods translate to holonomic constraint equations that must be satisfied when minimization finishes. Thus, an optimization problem with quadratic constraints must be solved. The number of constraints is large: an  $n$ -atom protein has  $3n$  unknowns (the

Cartesian coordinates of each atom) and on the order of  $2.5n$  constraints. An advanced algorithm suitable for this class of problem is sequential quadratic programming (SQP), a Newton-like method. SQP algorithms invariably require solution of two or more linear systems, and it is crucial for scalability that these systems be kept sparse. A special implementation of SQP tailored to covalent molecules with distance constraints was able to accomplish these goals.

The linear system involving second derivatives was solved indirectly using the same trick as TNCG: an inner conjugate gradient iteration. Each inner iteration now requires “preconditioning” to make intermediate structures satisfy a linear approximation of the constraint equations. This preconditioner is a second linear system derived from the constraint Jacobian matrix. The Jacobian is very sparse and can be arranged into a nearly block diagonal form. Since this linear system is solved so many times, a sparse factorization is computed once per outer iteration and saved for reuse.

Testing revealed that another important factor for good algorithm performance is proper choice of the constraint set. There is usually considerable freedom in which distances to fix, and a poor choice can introduce new ill-conditioning into the constraint Jacobian. A satisfactory procedure was devised that uses QR factorizations to automatically choose constraints in proteins [4].

Together, the algorithmic enhancements to SQP resulted in a successful implementation in CCEMD. Figure 2.1 demonstrates the effect of eliminating ill-conditioning. SQP finds a constrained local energy minimum 4- to 8- times faster than TNCG. Of course, a constrained minimum is only an approximation to the unconstrained solution, but in many applications the approximation is justified. This work is described more fully in [4].

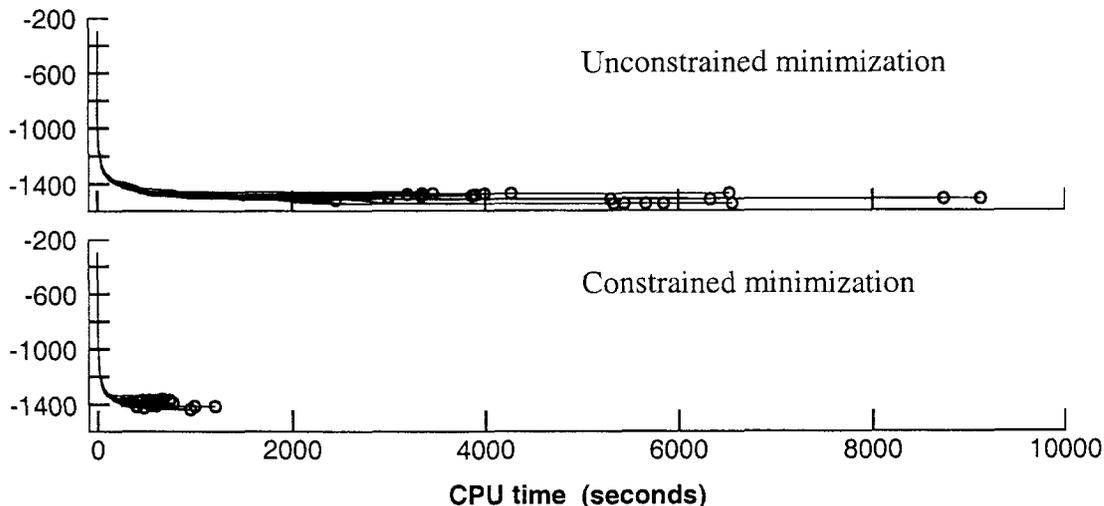


Figure 2.1: Constraints eliminate ill-conditioning

Potential energy reduction is plotted as a function of execution time for the unconstrained TNCG (top) and constrained SQP (bottom) algorithms. Each graph superimposes results from 20 different starting conformations of the 455-atom protein 1ROP. In general, each starting conformation leads to a different local minimum. The end of each algorithm run is shown by a circle. Unconstrained minimization exhibits long irregular “plateau” regions where little progress is made, which indicates ill-conditioning. Constraints eliminate the plateaus, but final energies (measured in kcal/mol) are not as low because bonds are fixed.

Attempts were made to port the SQP algorithm to LAMMPS for parallel processing. The intent was to solve the Jacobian linear system using Sandia's AZTEC software package [19]. AZTEC's solvers work efficiently, but require a distribution of matrix elements among processors that conflict with the distribution used by LAMMPS. Thus, there are two implementation options:

- Support the LAMMPS distribution for the Jacobian matrix and force AZTEC to use it. Saves on memory and communication costs, but AZTEC cannot achieve optimal load balancing.
- Support two distinct distributions for the Jacobian matrix. Gives good load balancing in AZTEC and LAMMPS, but requires twice as much memory, and significant communication overhead to keep the two distributions consistent.

Unfortunately, neither of these gave reasonable performance. Communication costs were very high, and the huge number of linear solves involving the Jacobian did not permit any compromise in AZTEC performance. To date, no version of the SQP algorithm has been released for LAMMPS.

### 3 Review of the Program

Customers for the project were identified as computational chemists at both SNL/CA and SNL/NM. Collaborations with both groups were established early in FY97: Richard Judson and Carl Melius (Dept. 8130), Steve Plimpton (Dept. 9221), and Jean-Loup Faulon (Dept. 9225). A representative set of ten test problems was obtained, ranging in size from 172 to 48,388 atoms. Minimization performance on these problems was used throughout the project as the yardstick for measuring progress.

The initial LDRD plan was to concentrate on the constrained optimization approach, but it turned out there was substantial customer demand for a reliable unconstrained minimizer. Thus, TNCG was developed first, in CCEMD and LAMMPS. In retrospect, this served as a useful introduction to the chemistry application source codes and supercomputer programming environments. The basic TNCG algorithm was delivered by Summer of 1997 and used immediately by the customers. Further TNCG work in FY98 was largely provided by student interns (Manoj Viswanathan, Genetha Gray, and Katherine Hegewisch), and OTC collaborators. This work included porting TNCG to the DEMoS code, and investigating advanced preconditioners.

The constrained optimization SQP algorithm was developed simultaneously beginning in Spring of 1997. The computational kernel was built around a prototype code known as ETR [20]. This Fortran code was first wrapped and connected to CCEMD, but later rewritten in C for cleaner interfacing. The single processor CCEMD implementation was used to develop algorithmic enhancements described in §2.2, and it performed well. The Fortran version of ETR was ported to LAMMPS in FY98, but never achieved a satisfactory performance level, as explained earlier. The project did succeed in identifying precise reasons for poor execution on distributed architectures, so a better approach can be designed for DEMoS.

OTC collaborations were begun in Spring of 1998. Progress was made on two different fronts: investigation of advanced optimization algorithms appropriate for energy minimization problems, and implementation of a novel client/server distributed computing framework for optimization problem solving. The algorithm work resulted in development of the L-BFGS

preconditioner described earlier. It also produced a general-purpose, public domain, nonlinear programming code built around the ETR kernel. The client/server framework exploits the nature of optimization methodologies to cleanly divide application code from algorithm code, allowing each to be executed on separate machines from separate object codes. This allows proprietary applications to access state-of-the-art algorithms without exposing source code or data structures. OTC researchers produced a final report summarizing the collaboration, included as Appendix A.

The project led to a number of publications and presentations to the scientific community. These included:

- “Parallel unconstrained minimization of potential energy in LAMMPS”, SAND report 98-8201
- “Fast energy minimization of large polymers using constrained optimization”, SAND report 98-8251
- “Optimization and parallel computing”, Graduate seminar on parallel computing, Northwestern University, Feb 1997
- “Parallel optimization of constrained molecular structures”, MCS department seminar, Argonne National Laboratory, Mar 1997
- “Parallel optimization of constrained molecular structures”, SIAM Annual Meeting, Stanford University, Jul 1997
- “Molecular energy minimization in parallel using constraints”, AMS Fall Western Sectional Meeting, University of New Mexico, Nov 1997
- Web pages at <http://midway.ca.sandia.gov/~tdplant/compchem/main.html>

The LDRD funded support for three student interns at Sandia, and two graduate students at Northwestern University:

Manoj Viswanathan (Spring 1998)

Chemical engineering undergraduate at University of California, Berkeley

Genetha Gray (Summer 1998)

Applied mathematics graduate student at Rice University

Katherine Hegewisch (Summer 1998)

Applied mathematics graduate student at Washington State

Leonardo Lopes (Summer 1998)

Industrial engineering graduate student at Northwestern University

Richard Waltz (Summer 1998)

Electrical and computer engineering graduate student at Northwestern University

Sandia employees who contributed to this project include Todd Plantenga (principal investigator), Len Napolitano (project manager), Juan Meza, Richard Judson, Steve Plimpton, Jean-Loup Faulon, Carl Melius, Mike Colvin, David Hobbs, Dianna Roe, Patricia Hough, Charles Tong, Chuck Bisson, and Randy Radmer.

## 4 Summary

The goals of this project were to design and study new algorithms for minimizing potential energy in Sandia computational chemistry codes. Two classes of algorithms were developed. The TNCG method proved faster and more reliable than existing algorithms for unconstrained minimization. It was coded in serial and parallel versions for CCEMD, LAMMPS, and DEMoS. The SQP method improved performance even more by partially fixing molecular covalent structure. The algorithm scaled well on a single workstation, but was difficult to parallelize for very large molecules.

The project started several collaborations between computational chemists and optimization algorithm specialists. Collaborations were fostered between National Laboratories and universities. The project's theories and results were disseminated through publications, presentations, personal contacts, student hires, and a web site.

Related work is continuing, including development of minimization algorithms for the new DEMoS software, and further research into L-BFGS preconditioners. Like most research projects, this LDRD led to new and unexpected inquiries while accomplishing its original goals.

## References

- [1] S. J. Plimpton, R. Pollock, and M. Stevens. Particle-mesh Ewald and rRESPA for parallel molecular dynamics simulations. In *Eighth SIAM Conference on Parallel Processing for Scientific Computing*, Minneapolis, MN, March 1997.
- [2] R. Judson, D. Barsky, T. Faulkner, D. McGarrah, C. Melius, J. Meza, E. Mori, T. Plantenga, and A. Windemuth. CCEMD - Center for Computational Engineering Molecular Dynamics: Theory and user's guide, version 2.2. Technical Report SAND95-8258, Sandia National Laboratories, Livermore, CA, 1995.
- [3] T. D. Plantenga. Parallel unconstrained minimization of potential energy in LAMMPS. Technical Report SAND98-8201, Sandia National Laboratories, Livermore, CA, 1997.
- [4] T. D. Plantenga. Fast energy minimization of large polymers using constrained optimization. Technical Report SAND98-8251, Sandia National Laboratories, Livermore, CA, 1998.
- [5] G. Gray, T. Plantenga, and M. Viswanathan. Effectiveness of Hessian-free unconstrained algorithms for minimization of molecular potential energy. In progress, 1998.
- [6] T. Schlick and M. Overton. A powerful truncated Newton method for potential energy minimization. *J. Comp. Chem.*, 8:1025-1039, 1987.
- [7] Tamar Schlick. Optimization methods in computational chemistry. In K. B. Lipkowitz and D. B. Boyd, editors, *Reviews in Computational Chemistry*, volume 3, pages 1-72. VCH Publishers, New York, 1992.
- [8] P. Derreumaux, G. Zhang, and T. Schlick. A truncated Newton minimizer adapted for CHARMM and biomolecular applications. *J. Comp. Chem.*, 15:532-552, 1994.
- [9] W. F. van Gunsteren and M. Karplus. A method for constrained energy minimization of macromolecules. *J. Comp. Chem.*, 1:266-274, 1980.
- [10] K. A. Palmer and H. A. Scheraga. Standard-geometry chains fitted to X-ray derived structures: Validation of the rigid-geometry approximation. *J. Comp. Chem.*, 12:505-526, 1991.
- [11] R. Abagyan, M. Totrov, and D. Kuznetsov. ICM - a new method for protein modeling and design: Applications to docking and structure prediction from the distorted native conformation. *J. Comp. Chem.*, 15:488-506, 1994.
- [12] Y. Duan, S. Kumar, J. M. Rosenberg, and P. A. Kollman. Gradient SHAKE: An improved method for constrained energy minimization in macromolecular simulations. *J. Comp. Chem.*, 16:1351-1356, 1995.
- [13] R. Fletcher. *Practical Methods of Optimization*. Wiley & Sons, Chichester, UK, second edition, 1990.
- [14] R. S. Dembo, S. C. Eisenstat, and T. Steihaug. Inexact Newton methods. *SIAM J. Numer. Anal.*, 19:400-408, 1982.
- [15] J.-L. Faulon. Massively parallel simulations of diffusion in dense polymeric structures. SC97 Super Computing Conference, San Jose, CA, November 1997.
- [16] J. L. Morales and J. Nocedal. Automatic preconditioning by limited memory quasi-Newton updating. Technical Report OTC 97/08, Optimization Technology Center, Argonne National Laboratory and Northwestern University, 1997.
- [17] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids*. Oxford UP, Oxford, U.K., 1987.
- [18] M. Saunders, K. N. Houk, Y.-D. Wu, W. C. Still, M. Lipton, G. Chang, and W. C. Guida. Conformations of cycloheptadecane. A comparison of methods for conformational searching. *J. Am. Chem. Soc.*, 112:1419-1427, 1990.
- [19] S. A. Hutchinson, J. N. Shadid, and R. S. Tuminaro. Aztec user's guide: Version 1.1. Technical Report SAND95-1559, Sandia National Laboratories, Albuquerque, NM, 1995.
- [20] Todd D. Plantenga. *Large-Scale Nonlinear Constrained Optimization Using Trust Regions*. PhD thesis, Dept. of Electrical Engineering and Computer Science, Northwestern University, 1994.

## Appendix A Report on Collaborations with the OTC

The LDRD funded a collaboration with the Optimization Technology Center (OTC)\*, a research enterprise with personnel at Argonne National Laboratory and Northwestern University. The project specifically supported three students at Northwestern during Summer 1998. They were supervised by Professor Jorge Nocedal, and contributed to the LDRD in two areas. Their final report is below.

### A.1 Interactive environment for nonlinear optimization

The recently developed NEOS Server ([www.mcs.anl.gov/home/otc](http://www.mcs.anl.gov/home/otc)) is an advanced demonstration of the Internet as a tool for computational optimization. It has been accessed by hundreds of users worldwide to solve a variety of optimization problems. Nevertheless, NEOS has several drawbacks. One of them is that it requires the user to send to the server a code that evaluates the objective and constraint functions. This mode of operation is often impractical, for either or both of two reasons:

- In many important applications this function evaluation is the result of a complex simulation involving a collection of routines, often written in various languages. It is therefore not possible or practical to send this code to the server.
- Some users are not willing to reveal their objective function and constraints, which describe their model and method of solution. One could think of ways of disguising the problem functions so that they would be difficult to interpret, but the user can never be sure that the NEOS operators won't be able to decipher their code and understand their model.

A further drawback is that NEOS will only interact with the user after completion of the optimization. In time-consuming applications, the user often wishes to monitor the solution process and stop if little progress is being made or if the computing time is too large.

One solution we have explored as part of this project is to leave the task of function evaluations in the user's machine and to access NEOS only to generate a better estimate of the solution. This is feasible in nonlinear programming, as we will discuss. To simplify the presentation of the ideas that follow, we will assume that the problem to be solved is the unconstrained minimization of a function,

$$\text{Minimize } f(x).$$

Nevertheless, the ideas we develop here are readily extended to constrained optimization.

We have focused on very large problems in which the Hessian matrix is difficult to evaluate, and have used the limited memory method (L-BFGS-B) [1] to perform the optimization. At every iterate  $x_k$ , this method computes a search direction  $d_k$  and a step length  $\alpha_k$ , and defines the new iterate as

$$x_{k+1} = x_k + \alpha_k d_k.$$

The crucial observation is that to compute the search direction  $d_k$ , these algorithms only require the current iterate  $x_k$  (an  $n$ -vector), the current function value  $f_k$  (a scalar) and the gradient

---

\* <http://www-c.mcs.anl.gov/home/otc/otc.html>

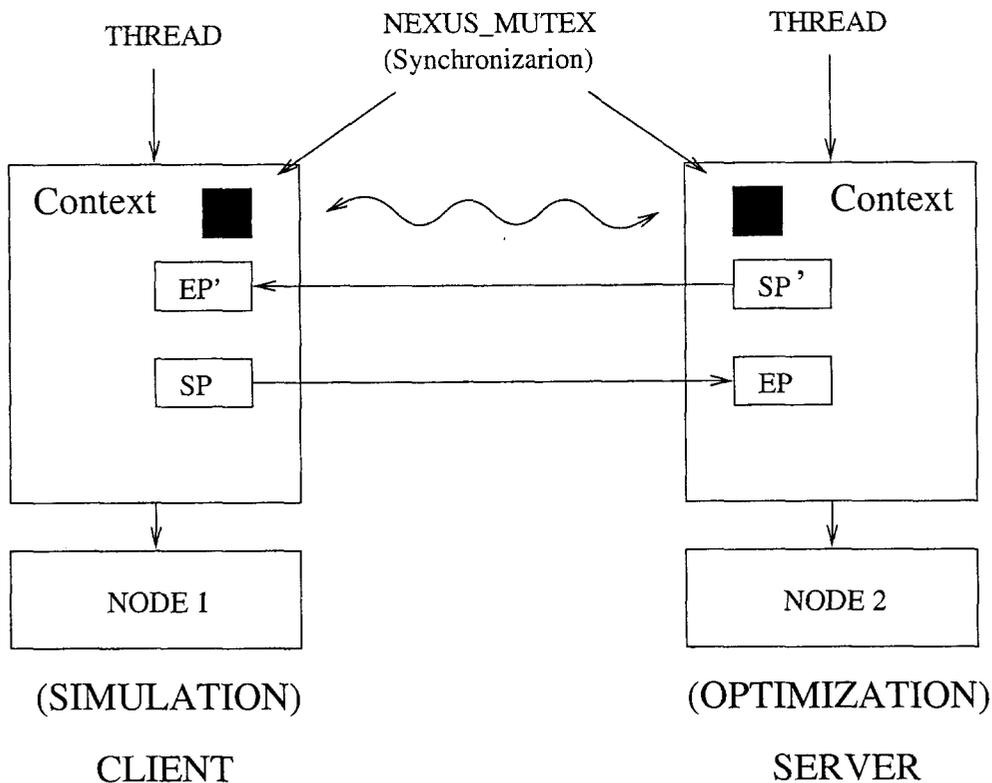
value at  $x_k$  (an  $n$ -vector). These are numerical quantities that do not reveal information about the model.

The interactive environment for nonlinear optimization that we developed operates as follows. The user chooses the initial vector  $x_0$ , evaluates the objective function  $f$  and gradient  $\nabla f$  at that point, and transmits this information to the appropriate NEOS server. The optimization server launches the solver, which computes a search direction and begins to perform a line search. During this line search new values of  $f$  and  $\nabla f$  may be required; if so NEOS sends the trial point to the user and requests the evaluation of the function and gradient at that point. After NEOS has received this information the algorithm proceeds with the line search.

An interactive session of this form continues until the solution is found. We should point out that most of the nonlinear optimization codes in NEOS, and in particular L-BFGS-B, are already written in a reverse communication form that makes this interaction simple to implement. The challenges lie in establishing a robust, reliable and fast interaction, and in creating the tools that will make access to NEOS transparent to the user. For this we have used the tools provided by Globus [2]. Communication between client and server is performed my means of Nexus [3]. Synchronization and check-pointing have been implemented using the mutex utilities. Authentication and security, as well as a heart-beat monitor are also provided by Globus, but have not yet been implemented in our environment. Figure A.1 illustrates our interactive environment using Nexus.

The interactive environment has been implemented using two Solaris machines. It is supported by the following software:

- `simulator.c` : the simulator exec (running on the user's machine); it contains the driver and the `main()` function.
- `optimizer.c` : the nonlinear optimization server (running on the remote machine). It contains all server functions, and Nexus handler functions which can be issued remotely by the client.
- `Makefile` : makes `simulator` and `optimizer`.
- `makefile_header`: the Globus header (required in `Makefile`).
- `reverse_com.h`: declares all data structures and memory allocation/deallocation functions.
- `reverse_internal.h` : declares all Nexus structures (client, monitor, (admin, server in the future...)) and functions for buffer management.
- `reverse_internal.c` : functions for buffer management.
- `reverse_memory.c` : memory allocation/deallocation functions.
- `reverse_lbfgs.h` : declares functions specific to L-BFGS.
- `reverse_lbfgs.c` : contains the client functions and Nexus handler functions which can be issued remotely by another Nexus node.



- 1) CLIENT ASKS FOR ATTACHMENT
- 2) SERVER GIVES AN SP TO THE CLIENT
- 3) CLIENT USES THE SP TO SEND SP' TO THE SERVER
- 4) THE REVERSE COMMUNICATION IS ESTABLISHED
- 5) THE NEXUS\_MUTEX ARE DOING THE SYNCHRONIZATION

Figure A.1: Interactive Nexus client/server environment

A working version of the interactive environment has been developed. We have demonstrated that the limited memory code L-BFGS-B can easily be accessed by a remote user in an interactive session supported by Nexus tools.

There are many areas of optimization where this interactive environment can be useful. Among them we should mention molecular design, geophysical inversion, neuron networks, chemical process analysis, and climate modeling. The advantage of our interactive environment is that the user can concentrate on the simulation and leave the task of optimization to NEOS, which is building an extensive array of solvers with which the user can experiment by simply clicking a box in a web page.

## A.2 Nonlinear optimization software

We have developed a Fortran code that implements the interior point algorithm for nonlinear programming described in [4, 5, 6]. It is designed to solve large optimization problems with equality and inequality constraints, and can be used as a feasible or infeasible method.

At the beginning of this project the mathematical algorithm had been fully developed

but several crucial details of implementation remained unresolved. One of them is the use of preconditioners for the conjugate gradient iteration that performs the step computation. We have coded three options for this preconditioner; they are all designed to remove the ill-conditioning created by the Hessian of the barrier term. In addition, we have implemented an adaptive heuristic in which the largest eigenvalue of the Hessian of the Lagrangian is estimated, and this value is used to scale the preconditioner. The objective is to properly weigh the relative contributions of the two terms in the barrier function: the portion corresponding to the primal variables, and the portion formed by the slacks.

The algorithm described in [4] is an infeasible method for optimization. This means that the algorithm can start from a point that does not satisfy the constraints, and that all subsequent iterates are allowed to move outside the feasible region of the problem. This gives the algorithm great freedom of movement. However, in many practical applications it is not admissible to generate points that violate certain constraints. This is because the objective function may not be defined at such points, or because the model is not valid. We have therefore developed a feasible version of the algorithm.

The fundamental idea of the feasible method is to let the merit function reject points that violate the constraints, and to leave all the logic of the algorithm unchanged. More precisely, suppose that an iterate violating a constraint is generated; then we simply declare the value of the merit function to be very large, and ask that the algorithm recompute the step with a smaller trust region. This simple change works well most of the time, but can give rise to the Maratos effect. This problem can be circumvented by the simple device of setting the slack variables to be the negative of the constraint values before the merit function is evaluated. An infeasible version of the algorithm based on this strategy has been extensively tested and has shown to be robust and efficient.

The Fortran code implementing the interior point algorithm, incorporating the new features described above, has now been written. It uses reverse communication to allow the user complete freedom in the evaluation of the function.

The step computation is done by a projected conjugate gradient iteration. The projections are performed by solving an augmented system, which is factored using the Harwell subroutine MA27. The same factorization of the augmented system is used to compute the Lagrange multiplier estimates. The code includes iterative refinement techniques to ensure that the projection operation is accurate.

Several drivers have been written to allow the user to choose one that fits his/her needs. The code has been tested using 135 problems from the CUTE collection. These include the well-known small problems collected by Hock and Schittkowsky as well as a variety of optimal control and nonlinear regression problems of large size. We have also tested the code on linear programs, which were successfully solved. Nevertheless, interior point codes designed exclusively to solve linear programs (such as PCx) appear to be more efficient. It is an open question whether our code can be made as efficient on linear programs as these codes.

Comparisons with various established codes have been made. Our algorithm requires far fewer function and constraint evaluations than LANCELOT. The active set method SNOPT has shown to be slightly more robust than our code, but appears to require more computing time – especially on problems where the number of degrees of freedom is not very small.

The code is now available for distribution in the public domain.

## Appendix References

- [1] D. C. Liu and J. Nocedal. On the limited-memory BFGS method for large scale optimization. *Mathematical Programming* 45:503-528, 1989.
- [2] I. Foster and C. Kesselman. Globus: A metacomputing infrastructure toolkit. *IJSA*, to appear.
- [3] I. Foster, C. Kesselman, and S. Tuecke. The Nexus approach to integrating multithreading and communication. *JPDC*, 37:70-82, 1996.
- [4] R.H. Byrd, J.C. Gilbert, and J. Nocedal. A trust region method based on interior point techniques for nonlinear programming. Technical Report OTC 96/02, Optimization Technology Center, Northwestern University, 1996.
- [5] R.H. Byrd, M.E. Hribar, and J. Nocedal. An interior point algorithm for large scale nonlinear programming. Technical Report OTC 97/05, Optimization Technology Center, Northwestern University, 1997.
- [6] R. H. Byrd, G. Liu, and J. Nocedal. On the local behavior of an interior point method nonlinear programming. *Numerical Analysis*, D. F. Griffiths, D. J. Higham and G. A. Watson eds., Longman, 1997.

DISTRIBUTION:

1	MS 9402	T. O. Hunter	8000
		Attn: J. B. Wright	2200
		J. F. Ney (A)	5200
		M. E. John	8100
		L. A. West	8200
		W. J. McLean	8300
		R. C. Wayne	8400
		P. N. Smith	8500
		P. E. Brewer	8600
		T. M. Dyer	8700
		L. A. Hiles	8800
1	MS 9003	D. R. Henson	8909
1	MS 9011	J. C. Meza	8950
1	MS 9214	L. M. Napolitano	8130
5	MS 9201	T. D. Plantenga	8950
1	MS 0188	D. L. Chavez	4100
1	MS 0899	Technical Library, 4916	
3	MS 9018	Central Technical Files, 8940-2	
1	MS 9021	Technical Communications Department, 8815/Technical Library, MS 0899, 4916	
2	MS 9021	Technical Communications Department, 8815, for DOE/OSTI	