ARGONNE NATIONAL LABORATORY
Mathematics and Computer Science Division

9700 South Cass Avenue
Argonne, Illinois 60439

# The NEOS Server for Optimization
# Version 4 and Beyond

**Elizabeth D. Dolan, Robert Fourer, Jorge J. Moré, and Todd S. Munson**

Preprint ANL/MCS-P947-0202

February 2002

This work was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing, U.S. Department of Energy, under Contract W-31-109-Eng-38, and by the National Science Foundation (Information Technology Research) grant CCR-0082807.

# Contents

**The NEOS Server for Optimization: Version 4 and Beyond[1]**

**Elizabeth D. Dolan,[2] Robert Fourer,[3] Jorge J. Moré,[4] and Todd S. Munson[4]**

**Abstract**

We describe developments associated with Version 4 of the NEOS Server and note that these developments have led to an exponential growth in the number of job submissions. We also provide an overview of some of the research and educational uses for the NEOS Server and discuss future research challenges.

# 1 Introduction

The NEOS Server [14] eliminates most of the time-consuming work conventionally associated with optimization applications, and allows the user to concentrate on the formulation of the problem. Solving optimization problems with NEOS requires the following steps:

Develop code to define the optimization problem

Submit the problem to NEOS

Interpret the results

In contrast, the solution of optimization applications in conventional computational environments requires several additional (time-consuming!) steps:

Identify and purchase software to solve the optimization problem.

Provide auxiliary information required by the solver.

Write code to link the solver, the auxiliary information, and the optimization problem.

Debug the linking process.

Execute the program on the machine where the solver is installed.

Providing auxiliary information for a solver can be especially time-consuming. For example, nonlinear problems often require derivatives and sparsity patterns.

The advantages of the NEOS approach are considerable, and users are often able to produce solutions to applications in a matter of hours instead of days or weeks. NEOS users do not need to purchase and install any optimization software; derivatives and other auxiliary

[1]This work was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing, U.S. Department of Energy, under Contract W-31-109-Eng-38, and by the National Science Foundation (Information Technology Research) grant CCR-0082807.

[2]Department of Industrial Engineering & Management Sciences, Northwestern University, and Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Illinois 60439 (`dolan@mcs.anl.gov`).

[3]Department of Industrial Engineering & Management Sciences, Northwestern University, Evanston, Illinois 60208 (`4er@iems.nwu.edu`).

[4]Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Illinois 60439 ({`more,munson`}`@mcs.anl.gov`).

information are computed automatically; the optimization problem is linked with the optimization software by the NEOS solver; and computing resources are provided through the NEOS system. The NEOS approach eliminates nonessential details of the solution process.

The NEOS project was launched in 1994 by the Optimization Technology Center [15] with support from the Department of Energy and Northwestern University. An initial research direction centered on showing that automatic differentiation tools can be used in optimization software to solve large problems while requiring only that the user provide a formulation of the problem. A later project, metaNEOS [11], investigated the use of metacomputing platforms to solve very large optimization problems cheaply. Current research centers on the use of advanced application service provider technologies for large-scale optimization.

A preliminary version of the NEOS Server was shown as a demo at Supercomputing 94 in November 1994, and version 1 of the Server became available in September 1995. It quickly evolved into an outstanding example of what can be done in the areas of Internet-based computing, computational servers, and collaborational technology. Major improvements to the Server became available in May 1997 (Version 2) and March 1999 (Version 3).

The NEOS Server is a collaborative project that represents the efforts of the optimization community by providing access to a large number (over 50) of solvers from both academic and commercial researchers. Optimization problems can be submitted in a programming language (Fortran, C), a modeling language (AMPL [1], GAMS [9]), or a wide variety of data formats. In short, the NEOS Server is a window into the complexity of the optimization world.

We provide an overview of developments that led to Version 4 of the NEOS Server. Our overview is highly focused and mainly nontechnical. See [8] for a comprehensive view of other Internet-based optimization projects. We describe the NEOS model in Section 2 with an emphasis on the components that are needed to provide an effective computational environment. Section 3 describes recent developments that have been introduced in Version 4 in order to make the Server faster and easier to use. These improvements include new interfaces, solvers, and computing resources.

The impact of the NEOS Server as a computational environment for the solution of optimization problems is reviewed in Section 4. We outline some of the applications that have been done with the NEOS Server in 2000–2001, with additional details in Appendix A. We also mention the use of the NEOS Server as an educational tool.

In Section 5 we discuss items of interest that might be gleaned from the accumulated records of the NEOS Server. Having processed tens of thousands of optimization submissions, we present examples of the kind of data that might appeal to the optimization community in general. We conclude in Section 6 with a brief discussion of possible future research directions.

## 2 The NEOS Model

Developing an effective environment for solving optimization problems over the Internet requires a multifaceted design. The components of the NEOS model, shown in Figure 2.1, implement this design. We now examine these components and outline some of the research issues that have to be addressed in order to provide an effective optimization environment.

Given a formulation of an optimization problem, the NEOS Server offers four interfaces for submitting optimization problems. These four interfaces, as shown in Figure 2.1, are email, a Web browser, the NEOS Submission Tool (NST), and Kestrel. In all cases, the problems are processed in the same way; just the method of submission changes.

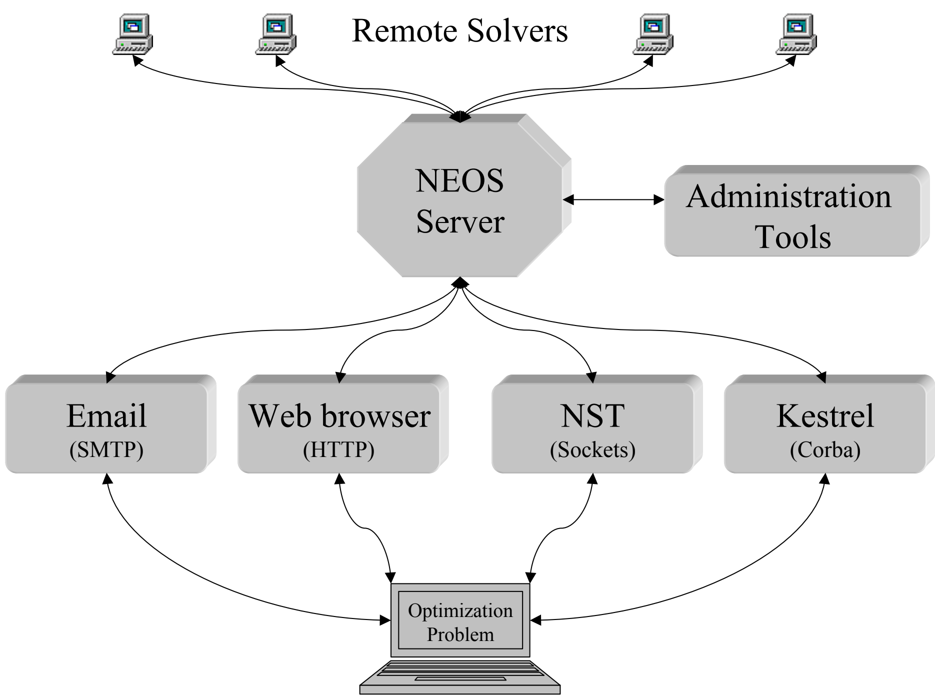


Figure 2.1: The NEOS Server

Email is efficient and basic. Many users who have slow Internet connections or who do not wish to remain online through a long job rely on this interface.

The most popular interface for problem submission is a Web browser; this interface is not, however, necessarily the fastest.

Submitting problems with NST is fast and convenient. Our first NST was developed in Tcl/Tk. This submission tool worked well on Unix systems, but many users wanted job submissions from non-Unix machines. To address this need, we developed a Java application. Many experienced users prefer the Java-enabled NST for job submissions because it is platform–independent and fast and allows users to save submissions for reuse.

Our most recent interface to the NEOS Server is Kestrel [4, 6]. Through this interface, the user can send optimization jobs from within a modeling language environment (AMPL [1] or GAMS [9]) to the NEOS Server. The user may choose a solver from the Server's list of appropriate solvers. Once the problem is solved on a remote system, results are returned to the user in the original (native) format. This is a key feature of the Kestrel interface: The native format enables the user's modeling language to interpret and manipulate the results directly, without having to parse the traditional NEOS text.

The NEOS Server received about 74,000 job submissions in 2001. This number translates into an average of more than 6,000 job submissions per month. Percentages of jobs submissions for each mode of submission are as follows: Email–11% , Kestrel–12%, NST–19%, and Web–58%. As expected, most of our users relied on a web browser for submitting their jobs. We suspect that users are reluctant to try NST because it requires downloading and installing an additional piece of software. In our view, the Java-enabled NST offers the best submission mode for general problems, but Kestrel offers the ultimate in convenience in terms of solving optimization problems within a modeling language. We suspect that the popularity of AMPL and GAMS for formulating optimization problems will lead to a sharp rise in problem submissions with Kestrel.

The NEOS Server, as shown in Figure 2.1, is the centerpiece of the NEOS model. Once the optimization problem is submitted to the NEOS Server, the submission is processed and dispatched for execution at a remote solver. The NEOS Server maintains communication with the remote solver and returns results to the user. The solution process raises several research issues:

How are problems specified?

How are problems submitted?

How are problems scheduled for solution?

How do we add solvers?

How are the problems solved?

Where are the problems solved?

Our solutions to these issues follow the design guidelines [2] of the NEOS Server: minimal problem specification, uniform and automated software registration, fault tolerance, and minimal maintenance. See [10] for an early discussion of network computing issues and [7] for a discussion of how Condor has been used in connection with the NEOS Server.

A user of the NEOS Server is concerned mainly with the first two issues. The user needs to know how the problems are specified and submitted. The other four issues concern the efficiency of the solution process, and, as long as the process is efficient, the typical user is not concerned with these issues. From an administrative viewpoint, however, those four issues are of paramount importance.

A description of the design and implementation of the NEOS Server is beyond the scope of this paper. Interested readers should consult the NEOS Administrative Guide [3] for details. This guide covers a variety of issues, including installing the Server, monitoring usage, adding solvers, and scheduling. There is also a chapter on the implementation of the Server.

Maintaining the NEOS Server requires several administrative tools. The main tools are the NEOS monitor and the NEOS communications tool. The NEOS monitor provides real-time information on the jobs submitted to the Server and their status. The NEOS monitor is also able to query a database of all jobs submitted to the Server and retrieve jobs with given attributes. The NEOS communications tool is used to enable communications between remote solver stations and the Server. New solvers can be registered through an NST, Web browser, or email.

Remote solvers are distributed worldwide. Once the submission arrives at a remote station, the submission is unpacked, a (typically Perl) script processes the submission through an optimization solver, and the results are returned to the NEOS Server. At present, remote solvers are available at the following sites:

| | |
|---|---|
| Aachen University of Technology | National Taiwan University |
| Argonne National Laboratory | Northwestern University |
| Arizona State University | University of Wisconsin |

Computations may take place on dedicated machines or on machines that have multiple uses. Computing times are fast because the NEOS machines tend to have powerful processors and large amounts of memory. Indeed, many users prefer to use NEOS machines for solving larger versions of their problems. See, for example, several of the reports in Appendix A.

Versions 1 and 2 of the NEOS Server were not portable, but version 3 was designed to be portable across machines, Web servers, and email servers. All that is required is a computer running Unix. This portability is a distinguishing feature of the NEOS Server; other computational servers are invariably dependent on the software environment and cannot be readily moved. Section 3 describes the new improvements that are available in version 4.

Improvements to the NEOS Server have allowed us to handle an exponentially increasing number of submissions without a heavy investment in either maintenance or machines. Over the past three years, the average number of job submissions per month for the NEOS Server has been as follows:

| | |
|---|---|
| 1999: | 1400 jobs/month |
| 2000: | 3100 jobs/month |
| 2001: | 6200 jobs/month. |

As the data indicates, the number of jobs roughly has doubled each year during this period. These jobs cover a wide variety of optimization problems. Some of the projects are outlined in Section 4, but the interested reader should examine Appendix A for details.

# 3 Recent Developments

Version 4 of the NEOS Server reveals significant improvements over previous versions. In this section we provide a brief overview of these enhancements. The aim is to give an idea of how the Server is evolving rather than to provide complete technical descriptions.

### Java submission client

We wrote a platform-independent NEOS Submission Tool (NST) GUI in Java to allow users of non-Unix platforms the features already enjoyed by our users of the Tcl/Tk NST. These features include fast socket connections with the NEOS Server and the ability to save submissions for reuse. One user took the Java class that specifies socket communications between client and Server and successfully plugged it into his own script to automate job submission.

### Kestrel interface

The Kestrel interface enables users of the AMPL or GAMS modeling environments to send their jobs to one of the many appropriate solvers on the NEOS Server and receive results in the native format for their modeling environment as if they were solving the jobs locally.

### Communication fault tolerance

We make better use of the Internet Protocol to detect communication errors and return information to the user. Responding to communication faults, we can overcome a temporary loss of connectivity between the Server and solver communications handler and return completed results to the user as if no lapse had occurred.

### Shift in Web technology

Our replacement of client-pull with server-push Web technology allows more NEOS jobs to run simultaneously without overwhelming the Web server with requests for information that may not be available. The former client-pull structure forced clients to query the Web server for intermediate job results every few seconds, resulting in tens of thousands of useless requests per month. Server-push techniques allow the Web server to transfer data as it becomes available over a sustained connection with the client. This technique has allowed the NEOS Server to process increasing numbers of jobs with only a fraction of the Web hits and subsequent load on the Web server machine as under the client-pull system.

### Web access to job status reports and job results

Web access to current queues and job intermediate and final results via job passwords allows users of any NEOS interface to access intermediate data and offers an alternative result retrieval mechanism to users who lose connectivity with the NEOS site's Web server, the

NEOS socket server, or the Kestrel server (although Kestrel users do have other mechanisms of retrieving results by reconnecting with the Kestrel server).

### Job termination

The Kill-Job administrative solver enables users to enter their job number and password to request that a job be terminated. If the job is waiting to run, it is removed from the solver queue. If the job is already running on a remote solver station, the Kill-Job solver connects to the NEOS communications daemon on that station and requests that the job be killed.

### Compressed file submission

Users can now upload compressed files via the Web or NST interfaces. The Server handles standard deflation of file types `.z`, `.Z`, `.gz`, and `.zip`.

### Binary file submission

The NEOS Server's newly refined ability to handle binary input files supports the Kestrel interface to AMPL and GAMS as well as Matlab binary submissions through the Web or NST.

### Solver flexibility

We appreciate the efforts of collaborative institutions to make some of their resources available for NEOS jobs. To ensure that these jobs do not overwhelm their systems, we have added numerous options to increase flexibility in setting limits on jobs. Examples of these options include file size limits, job time limits, and limits on the number of jobs that may run on a particular workstation at one time.

### Scheduler improvements

The scheduler has been rewritten to take some of the solver options into account. It also takes care to cull any bad jobs from a solver queue to make room for legitimate submissions.

### New solvers/resources

We have continued to add solvers to our Server. Currently over fifty solvers are available, some of them now running on dedicated machines at Northwestern University donated by Sun Microsystems or on part of the Linux cluster at Aachen University of Technology. New machines have also been made available at Arizona State University, essentially dedicated to the semidefinite and second–order cone programming area.

### GAMS to AMPL translator

One of our latest solvers harnesses the translator created by GAMS Development Corporation to solve GAMS submissions on any of our available AMPL solvers and return the results to users in GAMS format.

## 4 NEOS Users and Uses

Current NEOS users are attacking a wide range of optimization applications, demonstrating the power of the NEOS Server to solve business and scientific applications. Some of these applications are mentioned in this section, where we have divided applications into two groups depending on the reporting period.

As shown in Figure 4.1, the number of problems solved by the NEOS Server has increased dramatically over the past three years, with the current submission rate averaging above 5,000 job requests per month. An aspect of these results that is not shown in Figure 4.1 is that the increase in jobs submitted has been exponential over the past three years.

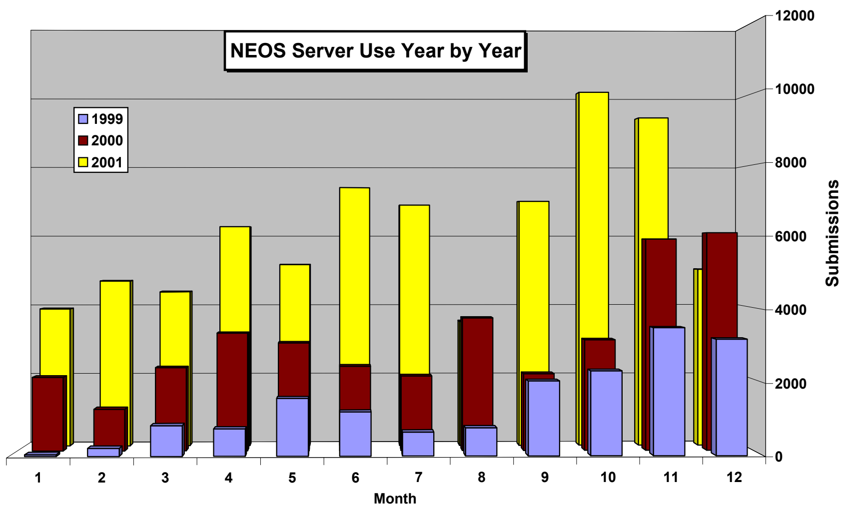


Figure 4.1: NEOS Jobs, 1999–2001

Although the increase in number of jobs processed by NEOS is impressive, even more impressive is the number of applications that are being solved with the help of NEOS. Obtaining information on these applications is not straightforward, since there is no automatic mechanism for collecting this information. We rely instead on users' responses to a pointer on the main NEOS Web page or to a notice included with their results. The applications listed below for 2001 were obtained mainly as a result of an email requesting feedback sent in May 2001. Additional information on most of the applications reported in 2001 can be found in Appendix A.

In the listing below we have noted only the business and scientific applications, but the NEOS Server is also being used for educational purposes. Indeed, about half of the responses are from academic institutions that are using our facilities to teach (usually engineering) courses that use optimization as a modeling tool for scientific and business applications. There is also heavy use by classes that specialize in optimization techniques.

### NEOS Applications: 2000

- ◇ Quality–of–service management in multimedia database systems.
- ◇ Estimating the value at risk of financial institutions.
- ◇ Image reconstruction of space objects.
- ◇ Design of a power cycle consisting of four heat exchangers, a turbine, and a pump.
- ◇ Transistor sizing for timing and noise reduction.
- ◇ Design of communication networks for large distributed networks.
- ◇ Modeling seat assignment problems.
- ◇ Unilateral contact problems in engineering mechanics and robotics.
- ◇ Distance geometry and multidimensional scaling.
- ◇ Forecasting future purchasing behavior for telecommunications services.
- ◇ Finding energy functions for molecular structures.
- ◇ Evaluating the capacity of the binary-multiplying channel.
- ◇ Portfolio selection and scheduling.
- ◇ Designing a yagi antenna.
- ◇ Resource requirements for broadband networks.

### NEOS Applications: 2001

- ◇ Circuit simulation with a large number of complex constraints.
- ◇ Predicting recent observations of Bose-Einstein condensates.
- ◇ Predicting globular protein folding.
- ◇ Low-power VLSI design.
- ◇ Settling conjectures for multi-person co-operation.
- ◇ Supply chain optimization.
- ◇ Protein structure prediction.
- ◇ Limit-state analyses of large dams.
- ◇ Studying the brain's representation system.

- ◇ Modeling electricity markets.
- ◇ Designing fractional delay filters for LAN/WAN's.
- ◇ Scheduling thermo– and hydro–energy resources.
- ◇ Building a crew-scheduling system for an airline in Indonesia.
- ◇ Solving the modified Cahn-Hilliard free energy equation.
- ◇ Applying optimization to farming in Switzerland.
- ◇ Route selection in packet switched networks.
- ◇ Duty-scheduling for ground handling activities at a regional airport.
- ◇ Finding bounds for the maximum cut value in sparse graphs.

## 5 Data Mining the NEOS Database

By the end of 2001, the NEOS Server had processed over 120,000 optimization jobs, with the submission rate still increasing. The data that has been collected during the life of the Server is of interest to the optimization community at large, and in this section we present some of this data. In particular, Figures 5.1 and 5.2 display sample compilations of NEOS submission characteristics for 2001.

Figure 5.1 shows the number of NEOS submissions per solver area. The recently separated areas of mixed-integer nonlinearly constrained optimization and mixed-integer linear programming are recombined into integer programming for the chart. This figure clearly shows that most of the jobs received by the Server are nonlinearly constrained optimization (NCO) problems, closely followed by integer programming (IP) problems. We caution, however, that these statistics can be misleading. For example, a user with a linear programming problem can easily submit the problem to a nonlinearly constrained optimization solver.

Nevertheless, we find it interesting that we are getting relatively few submissions of semidefinite programming (SDP) problems, since there is significant interest in the semidefinite programming area within the optimization community. We speculate that this is the fate of most new areas. Users have yet to understand how to pose their problems as semidefinite programs, and until that happens, we are likely to see interest center on more established areas.

Figure 5.2 gives an idea of the job submissions to some of the most popular solvers on NEOS. The submission numbers are a combination of all submissions to a solver at any of its various interfaces, including different input formats and optimization areas. We again warn about drawing conclusions from this data.

The data that we have presented raises many issues. For example, what type of the problem is being sent to each of the solvers? Are most of the problems small? If the problems are large, what are the characteristics of these problems?

We have developed a problem analyzer that provides information on the characteristics of the problems submitted, and we plan to use the information gathered in future work.

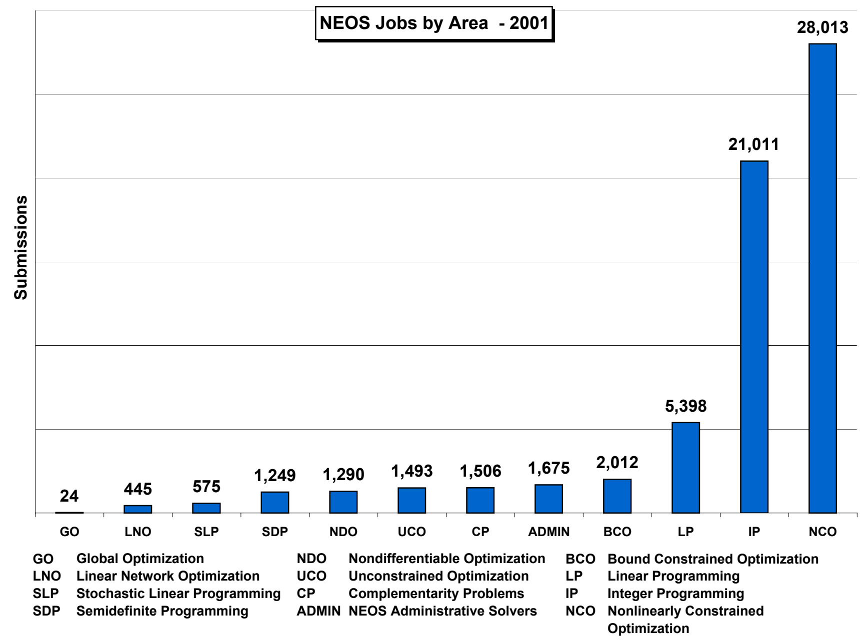


Figure 5.1: NEOS Jobs by Area

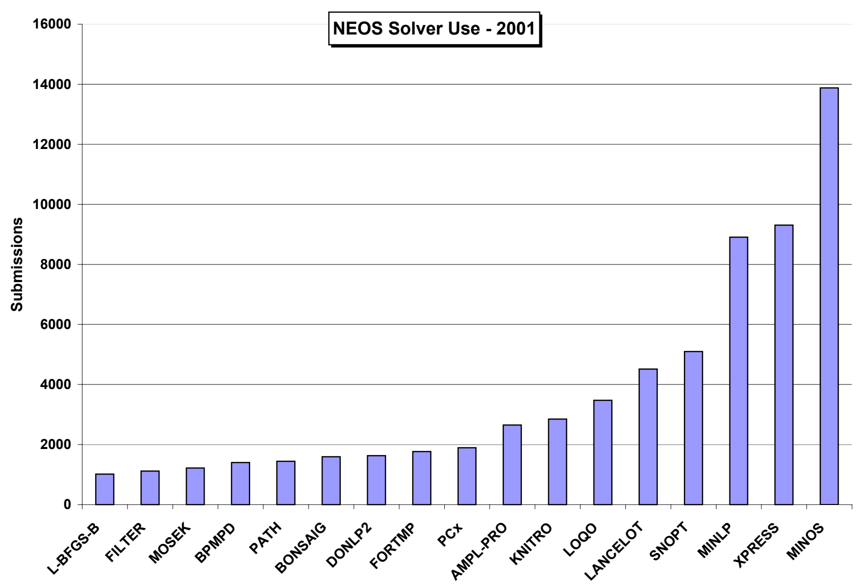


Figure 5.2: Most-used NEOS Solvers of 2001

Users should find a problem analyzer useful for understanding their models, and we may be able to use the analyzer's information to tune parameter choices for an optimization algorithm.

# 6 Future Developments

The NEOS Server was designed to provide optimization solutions to scientific computing problems, but it has found a niche in other areas as well. In this section we outline future research directions that have motivated by these developments.

Analyzing the data contained in all the problems that have been sent to the NEOS Server represents a major challenge. We have already addressed some of these issues in Section 5, but we have only scratched the surface. For example, we are frequently asked whether we can use a subset of these problems to develop a benchmarking collection. This is an appealing idea, but there do not seem to be any simple solutions.

As shown in Appendix A, the NEOS Server is being used for classroom projects. The ability to use an array of solvers on an optimization problem is clearly useful since students are able to concentrate on the modeling aspects of the project.

The NEOS Server is also being used to benchmark optimization solvers. Since users do not have a choice over the architecture used by the solver, however, timing information is unreliable. In addition, the information that solvers return is not consistent among solvers, so users are not able to compare the accuracy obtained by each solver. As a response to a growing interest in benchmarking [13, 12], we are developing a solver that will facilitate comparing various solvers. This interest is also partially responsible for our development of performance profiles [5] for benchmarking.

We anticipate adding solvers in new areas that are not currently represented. We also hope to add more global optimization solvers. Global optimization, in particular, is likely to raise scheduling and communication issues because solution times can be large.

Another possible direction for research is the addition of application–specific solvers. For example, we get many inquiries from users on cutting stock problems. This is a standard integer programming problem, but users are often not aware how this problem can be posed as an optimization problem. A solver directly connected with cutting stock problems would facilitate the solution of these problems.

## Acknowledgments

The NEOS Server is a collaborative project of the optimization community. Researchers associated with the Optimization Technology Center deserve special mention, in particular, Jean-Pierre Goux and Jeff Linderoth. A (we hope) complete list of collaborators can be found in the NEOS Server [14] site, but several efforts stand out:

Hans Mittelmann developed the semidefinite and second order cone programming area on the NEOS Server. He is also the solver administrator for more than a third of the solvers.

David Gay provides constant support for the AMPL solvers.

The GAMS development Corporation, in addition to providing GAMS support, developed the GAMS to AMPL translator.

Companies provide popular commercial solvers for access through the NEOS Server (Dash Associates and Dash Optimization's XPRESS, EKA Consulting's Mosek, and Optirisk Systems' FortMP).

The work of Mittelmann is noteworthy because the solvers that he manages are distributed over four machines at Arizona State University, with some of the solvers configured to use the full (64–bit) memory resources of the machines.

Of course, in many ways the most important collaborators are the developers of software that appear in our list of solvers that appears in Appendix B. Without their research, none of this would be possible.

# A User Feedback

The following are testimonials and explanations of use that arrived as a result of our May 2001 call for user feedback.

### Robert Meyer, University of Wisconsin

One of the main motivations [for using the NEOS Server] was the ability to run very large (more than 1M arc) network flow problems. Student accounts here have rather limited memory allocations, and many students had problems just storing the input data files for 1M arc problems (and the storage for the temporary files used by the algorithms is several times larger than the input). I wanted students to develop an awareness of the size of the problems that network codes can solve, and this would have been hard to do locally.

Additionally, I wanted students to develop a feeling for the relative speed of various network and general LP codes. NEOS provides easy access to 2 different network algorithms (both of which I teach in my course) as well as LP codes. This aspect I could have handled locally (with problems much smaller than 1M arcs), but students are generally reluctant to learn multiple interfaces, so it was very convenient to do it all via NEOS.

### Mahesh Ketkar, University of Minnesota

I have been using NEOS for circuit optimization problems involving a large number of complex constraints, and my experience with NEOS has been a very pleasant one. In particular I have used MINLP and SNOPT. The constant monitoring of the optimization problems being carried out is noteworthy and I especially thank the staff for killing my jobs from time to time and for making Kill Job facility available to users.

### Kazuhiro Saitou, University of Michigan

I have been using MINLP (AMPL version) at NEOS for my graduate class "Discrete Design Optimization." The class is offered every Fall semester (Sept – Dec) and enrolled by 20-30 students. Without the NEOS Server, I would be in big trouble...

For my class, it would be nice to have other solvers with AMPL input for mixed-integer programming (lp-solve with AMPL input?), genetic algorithms and simulated annealing....

### Richard Brewer, IBM Almaden Research Center and Stanford University

I am using the LOQO solver and code written in AMPL to perform numerical optimization of a spinor Bose-Einstein condensate. This yields the ground state energy of the atomic system and the corresponding wave functions as well as a variety of other properties of superfluids. The aim is to develop a fundamental theory which can explain the recent observations of Bose-Einstein condensates.

### Klaus Langohr, Universitat Politècnica de Catalunya, Spain

I'm a PhD Student here in Spain. I have been using the NEOS solvers frequently since the beginning of this year, when I was told that they could be useful in maximizing a likelihood function which has not been implemented in any statistical package, so far. Indeed, they are very USEFUL and have helped me a lot. Colleagues have already consulted me in order to get to know the possibilities you offer via the internet. For the moment, I normally use the solver SNOPT for my programmes written in AMPL, but I have to admit that, being a statistician, I need to reinforce more my knowledge about the other solvers you offer.

Summa summarum: THANK you very much!!!

### Stefano Morosetti, University of Rome, Italy

I have used your solver PCx extensively to answer the question: can a linear combination of knowledge-based potentials be used as fitness function in a genetic algorithm to predict the native folding of a globular protein? The NEOS Server is very useful because it allows [me] to save the time necessary to programme optimisation methods.

### Deborah Dent, University of Southern Mississippi

I am a part-time graduate student, and my research area is in solvers for systems of algebraic nonlinear equations. I first used MINPACK, and have added LANCELOT (SIF), LANCELOT (AMPL) and MINOS (AMPL) to the solvers included in my studies. I have had success with LANCELOT in solving a problem that I could not with other solvers, and I also use your solvers to verify the correctness of algorithms and other solvers that I am testing. I constantly use your NEOS Guide, and the links that your sites provide have been very useful in all of my studies. I really appreciate the work that you have put into making your site extremely user friendly and you sharing all of your technical expertise.

### Chandramouli Gopalakrishnan, University of South Florida

It was a very good experience working with the NEOS Server. I would highly compliment the stability of the server. In the one year I used the WWW version of the server, it might have crashed twice. I am indeed impressed.

I had been using XPRESS-MP MILP solver to obtain power optimal equivalent inputs to logic circuits. I work in the field of Low Power VLSI Design. I formulated large circuits as ILP problems, submitted them through the WWW interface, and obtained good results.

In the later part of my research, I had to solve a large number of similar problems. So I used the email option of the Server. I had written a shell script to mail my problem formulation as .mod files to the Server and waited for the results, which I parsed from the inbox input and processed.

On the whole, the XPRESS-MP solver as part of the NEOS Server was indeed a pleasure to work with.

I'd like to make special mention of the technical support behind the NEOS Server. I received outstanding support from them.

### Jenny Edwards, University of Technology, Sydney

I had a large very non linear problem. I used NEOS because of the AMPL interface and the access to various solvers. Originally I had no luck because the problem was just too large. I condensed it to one with about 11000 variables and 11000 constraints, of which about 10000 variables and 3000 constraints were nonlinear. After considerable work on the parameters, MINOS solved many variations of the problem in about ten minutes elapsed time each. I'd like to thank the NEOS team for their prompt and helpful attention to some queries I had. It was great to have the opportunity to try my problems on a number of different solvers; although, in the end, MINOS proved the most useful solver for this set.

I also found the web interface and email results made it easy to run a large project remotely and the overall documentation is an excellent introduction for beginners to math programming.

### Nick Gould, Rutherford Appleton Laboratory, England

I am a great fan of NEOS and try to spread the good news whenever I can. My most recent success was for an economics grad student from Oxford, whose thesis was to do with multi-person co-operation. He had a conjecture that something (I can't claim to understand the details!) that was true for two-person co-operation, might not be true for three or more. He managed to formulate this as a nonlinear program, that is if the NLP had a certain solution, his conjecture was true (the NLP was actually the discretized version of a continuous problem, and he wanted a fine discretization, that is, a large NLP). Anyway, he struggled on his own ... until I pointed him to NEOS. Within a few hours, he had formulated his problem in AMPL, sent it to a variety of solvers (LANCELOT, MINOS, KNITRO, LOQO and SNOPT), interpreted the results, et voila, his conjecture was true. I believe he has just successfully defended his thesis!

Quite apart from the fact that NEOS is extraordinarily useful for people wanting to solve real problems, it is equally wonderful for those of us who try to design new algorithms. I adore the gladiatorial aspect: put your solver up on NEOS, and let it fight, naked, against the world's finest (of course, I can't pretend that the humiliating defeats for LANCELOT don't hurt, but the experience is invaluable for our next-generation solvers).

### Giacomo Patrizi, Università degli Studi "La Sapienza", Rome, Italy

My students, all fourth year majors in OR, take my course in Mathematical Programming and are using NEOS to familiarize themselves with the best software available, which is invariably on your site. I would judge, but you should have machine data on this, that the favoured software is Lancelot for constrained problems and CGplus, NMTR for the unconstrained case, the latter being a lot faster. The input into Lancelot which requires elemental and group function decomposition is a bit troublesome and it would be preferable to submit the functions in natural form. Probably this can be done, but we may have overlooked something.

We are also using NEOS for research and for thesis development. This is very important to us, as we can use the abundant routines given to try and document how the other routines fare, as you will notice, from our user statistics. In this we have a bit of a problem in finding one input format, but we expect that the only common input to most routines is AMPL, so we shall have to learn it.

We will of course give due credit to NEOS as the papers that we develop appear in print, since we are extremely grateful to your institutions for such a great service to science and to research. I have neither kept a research diary regarding [our] use of NEOS, nor a didactic diary for my course, but as I find that NEOS is a great idea and extremely well implemented, I would be glad to do anything that might be useful to you to help the project.

### David Bogle, University College London, England

We are working on applied optimisation in Chemical Engineering with projects on global optimisation, supply chain optimisation, and a project about techniques for mining the data produced during an optimisation to provide more richness about the quality of the solution.

### Henry Wolkowicz, University of Waterloo, Canada

We (my students and I) have been solving large semidefinite programs (SDPs) using both SeDumi and CSDP solvers. Both have worked excellently. We have been able to solve some very large problems. These problems arise from relaxations of quadratic assignment problems.

### Alberto Maria Segre, University of Iowa

For the past several years I have been working on a system for protein structure prediction. As part of my system, I had need to incorporate a nonlinear programming solver to handle packing of sidechain atoms in the protein (a summary of my project can be found on my web site: `http://dollar.biz.uiowa.edu/~segre/`).

On the suggestion of both Kurt Anstreicher and Margaret Wright, I modified my code to output packing subproblems directly in AMPL, which I could then submit to NEOS. In this fashion, I was able to evaluate solutions provided by different nonlinear solvers (I tried DONLP2, LOQO, FILTER, KNITRO, and LANCELOT), in effect testing the code before deciding which to acquire and attempt to link into my system.

NEOS is a great resource for people like me. Keep up the good work!

### Miguel Anjos, University of Waterloo, Canada

I used the solver CSDP via NEOS to solve several dozen semidefinite optimization problems. These SDPs were very large, typically with over 100 variables and several thousands of constraints.

I found that the NEOS system was easy to use and quite reliable. I would certainly consider using it again in the future, as well as pointing it out to other people whose work also requires solving optimization problems.

Thank you again for setting up NEOS and for keeping it running!

### Raffaele Ardito, Technical University of Milan, Italy

I use the NEOS Server for academic purposes. In my research activities I have encountered some challenging problems in Mathematical Programming. In detail, I am dealing with ultimate limit-state analyses of large dams by means of a non-standard approach ("direct method"); this requires solving problems of linear and non-linear programming. The NEOS Server is an extraordinary tool to perform parametric tests on small models, in order to choose (also with the kind help of NEOS operators) the best suited solver. So far I have used CONOPT, MINOS and SNOPT, by way of the GAMS interface, for my non-linear, non-smooth, non-convex (very cumbersome!) problems, achieving very good results.

### Ariela Sofer, George Mason University

I have been using NEOS in my nonlinear programming course every spring semester. We use the student version of AMPL earlier in the semester, and then I assign a larger applied problem (that is too large to be handled by the student version), which the students are asked to model with AMPL and then solve. In the past I have used AMPL with MINOS, but in future semesters I plan to ask students to compare different solvers.

I have also used NEOS in assisting a non-optimization colleague of mine to solve optimization problems relating to database design.

### Sergio Lucero, University of California – Davis

I am a graduate student at the UC Davis applied math program, working under the supervision of Dr. Naoki Saito.

It has been our pleasure to discover the power of NEOS. Currently we are using it to solve the sparse representation problem that comes up in neuroscience (study of the brain's representation system):

$$\min \{\Sigma(\|BX_k\|) : B'B = I\}$$

or an appropriate relaxation thereof. The advantage of NEOS is that we can try different solvers to see which one is capable of solving our problem. Many thanks and keep it up!

### Ralph Mellor, Digital Impact Web Designs

I'm writing a business application with an optimization sub-problem. I've formulated the optimization sub-problem as a binary LP. On my 192MB ram PC, lp_solve handles up to 10,000 variables. I'm using NEOS to try out other solvers.

### Jim Bushnell, University of California Energy Institute

I've been using your server to solve complementarity models of electricity markets (using the PATH solver and AMPL). This is academic research. The following is an abstract of my paper. Your site has been invaluable for me. Keep up the good work.

> This paper presents a modeling framework for analyzing competition between multiple firms that each possess a mixture of hydroelectric and thermal generation resources. Based upon the Cournot concept of oligopoly competition, the model characterizes the Cournot equilibrium conditions of a multiperiod hydro-thermal scheduling problem. Using data from the western U.S. electricity market, this framework is implemented as a mixed linear complementarity model. The results show that some firms may find it profitable to allocate considerably more hydro production to off-peak periods relative to peak periods. This strategy is a marked contrast to the optimal hydro-schedules that would arise if no firms were acting strategically. These results highlight the need to explicitly consider profit maximizing behavior when examining the impact of regulatory and environmental policies on electricity market outcomes.

### Muzaffer Kal, Soft Mixed Signal Corporation

I have used NEOS with the LOQO solver to optimize an interpolator. The interpolator will be used in a fractional delay filter. This is for R&D purposes [relating to LAN/WAN systems].

### Fabio Schoen, Università degli Studi di Firenze, Italy

Just a comment: you are doing an excellent job and an extraordinary service to the academic community. I regularly suggest that my students use NEOS as soon as their projects in AMPL cannot be solved with the student edition. So they debug their AMPL models locally with AMPLPlus (which unfortunately is no longer supported from ILOG - this is a great damage to the operations research community) and then they run their real-life projects thanks to NEOS. In my laboratory we have full versions of AMPL, Cplex and MINOS, both on a PC and on a SUN server, but the students decidedly prefer to use NEOS from their lab or home PC. (If you are curious, you may find a short list of some of the projects at my home page, following the links teaching→cases, or going directly to `http://globopt.dsi.unifi.it/users/schoen/teaching/casi.html`).

So: many many THANKS!

### Steven Dirkse, GAMS Development Corporation

I've been using NEOS lately to compare the performance of various NLP algorithms that utilize second order information. We're a business of course, but this usage is a mixture of commercial and research work.

I've been testing Knitro and filter and will use other NLP codes as time and bandwidth permit.

Thanks for providing this useful service.

### Craig Kirkwood, Arizona State University

This is a great resource for those of us who need to demonstrate large scale optimization to students, but who do not have enough need for this (or funds!) to justify the purchase of a large scale optimization package.

### Rodrigo Fuentes, University of Waterloo, Canada

Lately I've been using the NEOS Server to solve large optimization problems formulated as semidefinite programs. Essentially, it is a combinatorial problem that models the scheduling of thermal and hydro energy resources. This is part of my research work at the University of Waterloo in Canada.

Because of the size of the test systems I've used, the NEOS Server has been quite useful.

### Jonathan Morris, Global Tech

We are in the process of trying to build a Web based crew scheduling system for an Airline in Indonesia (where we live!).

### Timothy J. Kosto II, Rensselaer Polytechnic Institute

I am a graduate student at RPI in Troy, NY working on non-linear polymer diffusion in confined spaces. Specifically, I am attempting to find solutions to the modified Cahn-Hilliard equation, a fourth order partial differential equation by techniques to minimize the total free energy of the system. While these problems are certainly not as complex as the large scale process optimization studies performed by other groups, the NEOS system using the Lancelot solver has allowed me to spend more time on experimental work that is also of interest to us since the optimization routines are made easier to use than if I had to write them myself. The submission process is straightforward and the availability of the NEOS solvers has allowed me to concentrate more on the physics of our problem than the mechanics of writing optimization codes. If programs like the IBM data visualization explorer and HDF were made as easy to use as the optimizers from ANL, I would be ecstatic.

As it is, I would encourage the continued funding of the NEOS Server at ANL and its related costs. The service has been a great help in my progress toward a doctorate.

### Dean Vrecko, University of British Columbia

I would like to thank all the people who provide and maintain NEOS. At the beginning of this summer I began a term of assisting a professor at my university with his research. The daunting task of solving nonlinear programming problems was put on my shoulders, and I felt as though I was in over my head. I didn't even know what nonlinear programming was, and after I discovered what it was, it became clear how enormous a task it would be to solve the problems assigned to me. I bought Maple 6 to try to help me, but the calculation times were huge, as well as the round off error. Then looking for a better solution, I found NEOS. With the great abundance of help given me by the caretaker(s) of NEOS, it took a short while to figure out how to set up the input file (I used AMPL format). I had extremely complicated objective functions, both convex and nonconvex, an arm-load of variables, and an arm-load of convex, nonconvex, equality and inequality constraints, but when I sent off the information via the web submission form, within seconds I received extremely accurate and consistent results. The results were used for verifying a certain theory in my professor's research, and so accuracy was extremely important.

It has been an invaluable service to me and I truly doubt that I could have done my job to the satisfaction of my professor without it. I would highly recommend NEOS to anyone needing constrained optimization calculations done quickly and accurately.

### Daniela Mencarelli Wieser, Swiss Federal Institute of Technology

The model I have been developing includes external costs of soil erosion and water pollution in farmers' optimisation problems. The model is dynamic, spatially differentiated, and includes some non-linear restrictions and binary variables. I have been using the MINLP (AMPL input) solver. It is the best non-linear solver that provides integer results I have found so far.

### Peter Whiting, University of Kansas

Excellent tool! I am using the NEOS Server for academic research regarding route selection in packet switched networks. I had no prior experience in the GAMS language but was able to quickly hack together a series of equations that the NLP solver was able to give me good results with.

### Ingrid Bohn, Universidad Tecnica Federico Santa Maria, Chile

I'm a student from Chile at the "Universidad Tecnica Federico Santa Maria", and I'm going to graduate [soon]. My thesis is about an Extension of the Eppen & Martin Variable

Redefinition for Multi-item Capacitated Lot-sizing Problems, and I have sent some examples to the NEOS Server.

### Rob Brans, P.W. Groep B.V., The Netherlands

We are using NEOS services for duty-scheduling for ground handling activities in [a] regional airport environment.

At this moment, duty-scheduling is a manual process. We are investigating how much can be gained in terms of "doing the same work with less employees" by using a more advanced scheduling mechanism. In the investigation stage "LP on demand" offers a very good solution to see how much can be gained. If the results are positive, we will consider buying the software tools.

We are using the XPRESS solver.

### Jouneau Frederic, Universite Charles-de-Gaulle Lille 3, France

I enjoyed working with NEOS mainly because it provides a non-commercial way to compute large MIP. My research interests are in econometrics. I am currently writing an article using the results NEOS just provided.

### Xinmin Hu, University of Melbourne, Australia

Recently, I have been using NEOS (solvers:PATH, DONLP2, FILTER, MINOS) to carry out some numerical tests on a game theory model arising in currently debated electricity restructuring.

A big thank you to NEOS staff for providing such a wonderful facility (free!).

### Frauke Liers, University of Cologne, Germany

We very much appreciate the NEOS Server service. We find not only that the solvers and the software are reliable, but also that the people maintaining and developing the codes are very helpful. The NEOS Server provides the possibility to run problems without going through the whole process of installing a software package, paired with the possibility of running instances on powerful machines.

At the university of Cologne we are interested in finding good bounds in the maximum cut value in sparse graphs. This problem has an application in theoretical physics and is interesting to the physicists' community. We calculate an upper bound on the exact value of the maximum cut by formulating it as a positive semidefinite program. We then add valid inequalities to tighten the relaxation. We solve the resulting problem exactly with the solvers CSDP and SDPA.

### Quan Wen, Insightful Corporation

I am using NEOS Servers for benchmark testing on one of the nonlinear optimization solvers. This is an important part of the decision process if we want to adopt the new optimizer. NEOS is very user-friendly, dependable and saves a lot of research time. I would continue to use NEOS Server for this purpose.

I appreciate all the work you have done to make this project available to the public.

### Robert R. Meyer, University of Wisconsin

NEOS has been a very valuable tool in the two graduate optimization courses that I regularly teach: CS 719 Network Flows and CS720 Integer Programming. NEOS allows students to see a broader variety of solvers than we have available in the Computer Sciences Department at the University of Wisconsin - Madison , and also helps to overcome a variety of time/storage limits that apply to student jobs here. For example, the student version of AMPL packaged with the text for CS720 allows at most 300 constraints and 300 variables, but several students in the course are doing course projects involving problems that are substantially larger than those limits.

### Anders Forsgren, Royal Institute of Technology, Sweden

I have been teaching a course on linear and integer programming over the past two months. Part of the course is based on projects where the students use GAMS. Previous years, they have used the student version of GAMS to model the project problems. This year, I have given them larger problems where they have sent their GAMS files to NEOS and used XPRESS-MP to solve the problems. The ability to use GAMS and not being limited in size by the student version has been extremely useful. I find NEOS a very useful tool for teaching this course. When the students leave the course, they know how to build a GAMS model, and they also know that they can solve large problems via NEOS.

PS I assume that my students were instrumental in making ".se" the number one user domain in terms of usage at some point during the autumn. I plan to give a similar course on nonlinear programming during January-March, so we are looking forward to making use of NEOS again.

Many thanks!

### Karthik Natarajan, MIT

I would like to complement on you an excellent site - that simplifies solving semidefinite programs a lot without installing software. I am interested in knowing if people use NEOS solvers in their computational experiments and actually quote computation times from here in academic papers - I find computation times here faster than from running codes on my laptop.

**Bee Jay, Bee Jay Enterprises**

We use NEOS to evaluate various solvers/algorithms on test problems of interest. The use is both educational and commercial. The application is the supply-chain optimization of global manufacturing enterprises where manufacturing activities take place across the globe around the clock.

The model is LP/MILP. Constraints are from transportation, manufacturing and warehousing capacities. Optimization is to achieve just-in-time production to meet demands on time considering the above constraints. Optimization with uncertainty is also under study. Typical problems are very large and hence solver performance is a critical factor. Solvers used are MOSEK, XPRESS-MP for LP/MILP.

## B NEOS Solvers

Mixed-integer Nonlinearly Constrained Optimization

- MINLP
- SBB

Mixed-integer Linear Programming

- BonsaiG
- FortMP
- GLPK
- XPRESS

Nonlinearly Constrained Optimization

- CONOPT
- DONLP2
- FILTER
- KNITRO
- LANCELOT
- LOQO
- MINOS
- MOSEK
- PATHNLP
- SNOPT

Linear Programming

- BDMLP
- BPMPD
- FORTMP
- HOPDM
- MOSEK
- PCx
- XPRESS-MP

Bound-Constrained Optimization

- BLMVM
- L-BFGS-B
- LANCELOT
- TRON

Unconstrained Optimization

- CGplus
- NMTR
- VMLM

Linear Newtork Optimization

- NETFLO
- RELAX4

Semidefinite and Second-order Cone Programming

- CSDP
- CirCut
- DSDP
- MOSEK
- PENNON
- SDPA
- SDPT3
- SeDuMi

Complementarity Problems

- MILES
- PATH

Nondifferentiable Optimization

- ACCPM
- APPS
- BT
- DFO
- NDA

Stochastic Linear Programming

- AUGMENTED
- MSLIP

Global Optimization

- GLOBMIN

## References


[1] *AMPL*. See `http://www.ampl.com/`.

[2] J. Czyzyk, M. P. Mesnier, and J. J. Moré, *The NEOS Server*, IEEE Computational Science & Engineering, 5 (1998), pp. 68–75.

[3] E. D. Dolan, *NEOS Server 4.0 administrative guide*, Technical Memorandum ANL/MCS-TM-250, Argonne National Laboratory, Argonne, Illinois, 2001.

[4] E. D. Dolan, R. Fourer, J.-P. Goux, and T. S. Munson, *The Kestrel interface to the NEOS Server*. In preparation.

[5] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Programming, 91 (2002), pp. 201–213.

[6] E. D. Dolan and T. S. Munson, *The Kestrel interface to the NEOS Server*, Technical Memorandum ANL/MCS-TM-248, Argonne National Laboratory, Argonne, Illinois, 2001.

[7] M. C. Ferris, M. P. Mesnier, and J. J. Moré, *NEOS and Condor: Solving optimization problems over the Internet*, ACM Trans. Math. Software, 26 (2000), pp. 1–18.

[8] R. Fourer and J.-P. Goux, *Optimization as an Internet resource*, Interfaces, 31 (2001), pp. 130–150.

[9] *GAMS*. See `http://www.gams.com/`.

[10] W. Gropp and J. J. Moré, *Optimization environments and the NEOS Server*, in Approximation Theory and Optimization, M. D. Buhmann and A. Iserles, eds., Cambridge University Press, 1997, pp. 167–182.

[11] *metaNEOS*. See `http://www.mcs.anl.gov/metaneos`.

[12] H. D. Mittelmann, *Benchmarks for optimization software*. See `http://plato.la.asu.edu/bench.html`.

[13] ——, *Benchmarking interior point LP/QP solvers*, Optim. Methods Software, 12 (1999), pp. 655–670.

[14] *NEOS Server for Optimization Problems*. See `http://www-neos.mcs.anl.gov/`.

[15] *Optimization Technology Center*. See `http://www.ece.nwu.edu/OTC`.