Modeling and Solving Nontraditional Optimization Problems

Session 1a: Background

Robert Fourer

Industrial Engineering & Management Sciences
Northwestern University

AMPL Optimization LLC

4er@northwestern.edu — 4er@ampl.com

Chiang Mai University International Conference Workshop
Chiang Mai, Thailand — 4-5 January 2011
Motivation

General-purpose optimization

- Minimize/maximize a function of decision variables
- Subject to equalities/inequalities constraining the values of the variables

General-purpose optimization software

- Solvers
  - apply algorithms to optimization problems
  - specialized to mathematical problem types
- Modeling systems
  - describe models to solvers using representations familiar to people
  - extended to problems & solvers of many types
Traditional Paradigms

Continuous optimization
- Interval domains for decision variables
- Smooth objective and constraint functions
- Locally optimal solutions

Discrete optimization
- Integer decision variables
  - Often zero-one decision variables
  - Often mixed with continuous decision variables
- Linear objective and constraint functions
  
  . . . diverse problem types converted to these forms
Non-Traditional Paradigms

Alternative problem types involving . . .
  - Logic operators
  - Complementarity conditions
  - Conic constraints
  - Globally optimal points

Varied solver strategies
  - Automated conversions
  - Extended traditional algorithms
  - New non-traditional algorithms
Sessions

1. Introduction
   a. Background
   b. Current features

2. Nontraditional specialized optimization
   a. Second-order conic constraints
   b. Complementarity conditions

3. Nontraditional discrete optimization
   a. Modeling support
   b. Solver support

4. Interfacing with nontraditional solvers
   a. Solver interfaces
   b. Solver selection
Session 1a: Background

Focus
- Survey of traditional systems
- Availability of software

Topics
- Solvers
- Modeling languages & systems
- Free solver sources
  - NEOS Server
  - COIN-OR
Solving: Problems & Algorithms

Linear programming

Integer programming

Quadratic programming

Nonlinear programming
Solving

Linear Programming

Algorithms

- **Scope**
  - Linear objective
  - Linear equations and inequalities

- **Methods**
  - Primal simplex
  - Dual simplex
  - Barrier (interior-point)

Solvers

- Module within nonlinear
  - MINOS
- Module within mixed-integer
  - *see next slide...*
Solving
Mixed-Integer Programming

**Algorithms**

- **Scope**: linear programming with
  - integer variables
  - zero-one variables
- **Method**: branch-and-bound
  - solve fractional subproblems by dual simplex
  - improve fractional solutions by cut generation
  - seek integer solutions by branching & heuristic search

**Solvers**

- **Commercial**
  - CPLEX, Gurobi, MOSEK, XA, Xpress
- **Open source**
  - CBC, GLPK, lp_solve
Solving

PSD Quadratic Programming

Algorithms

- Scope: linear with
  - objective minimizing $x^T Q x$
  - constraints of the form $x^T Q x \leq b$
  - ... where $Q$ is positive semi-definite ($x^T Q x \geq 0$ for all $x$)

- Methods: generalizations of linear algorithms
  - Simplex methods
  - Barrier methods
  - Branch-and-bound procedures

Solvers

- Commercial
  - CPLEX, Gurobi, MOSEK, Xpress
Solving

Nonlinear Programming

Algorithms

- Scope
  - Smooth nonlinear objective and constraints
  - First-order necessary conditions for local optimums
- Methods
  - Generalized reduced gradient
  - Sequential quadratic
  - Interior-point (barrier)

Solvers

- Commercial
  - CONOPT (GRG), KNITRO (IP/SQ), LOQO (IP), MOSEK (IP), MINOS (GRG), SNOPT (SQ)
- Open source
  - Ipopt (IP)
**Modeling: Languages & Systems**

**Goals**
- Express models in forms that are familiar and convenient to people
- Support the entire modeling process
  * Formulate
  * Solve
  * Analyze
  * Revise
  * Deploy

**Alternatives**
- Program your own specialized system
- Adapt an existing language or program
- Use a language & system designed for optimization
Modeling

Alternatives

Adaptations for optimization modeling

- Spreadsheets
  - Frontline Excel Solver, What’s Best
- Math modeling systems
  - MATLAB, Mathematica
- Object-oriented programming languages
  - C++: FLOPC++
  - Python: Pyomo, POAMS, CVXMOD

Algebraic modeling languages for optimization

- Captive
  - OPL (CPLEX), Mosel (Xpress), OPTMODEL (SAS)
- General-purpose
  - AIMMS, AMPL, GAMS, MPL
AMPL

Algebraic modeling language: *symbolic data*

```
set SHIFTS;               # shifts
param Nsched;             # number of schedules;
set SCHEDS = 1..Nsched;   # set of schedules
set SHIFT_LIST {SCHEDS} within SHIFTS;
param rate {SCHEDS} >= 0;      # pay rates
param required {SHIFTS} >= 0;  # staffing requirements
param least_assign >= 0;       # min workers on any schedule used
```
**AMPL**

*Algebraic modeling language: symbolic model*

```plaintext
var Work {SCHEDS} >= 0 integer;
var Use   {SCHEDS} >= 0 binary;

minimize Total_Cost:
    sum {j in SCHEDS} rate[j] * Work[j];

subject to Shift_Needs {i in SHIFTS}:
    sum {j in SCHEDS: i in SHIFT_LIST[j]} Work[j] >= required[i];

subject to Least_Use1 {j in SCHEDS}:
    least_assign * Use[j] <= Work[j];

subject to Least_Use2 {j in SCHEDS}:
    Work[j] <= (max {i in SHIFT_LIST[j]} required[i]) * Use[j];
```
AMPL

Explicit data independent of symbolic model

set SHIFTS := Mon1 Tue1 Wed1 Thu1 Fri1 Sat1
    Mon2 Tue2 Wed2 Thu2 Fri2 Sat2
    Mon3 Tue3 Wed3 Thu3 Fri3 ;

param Nsched := 126 ;

set SHIFT_LIST[1] := Mon1 Tue1 Wed1 Thu1 Fri1 ;
set SHIFT_LIST[2] := Mon1 Tue1 Wed1 Thu1 Fri2 ;
set SHIFT_LIST[3] := Mon1 Tue1 Wed1 Thu1 Fri3 ;
set SHIFT_LIST[4] := Mon1 Tue1 Wed1 Thu1 Sat1 ;
set SHIFT_LIST[5] := Mon1 Tue1 Wed1 Thu1 Sat2 ; .......

param required :=
    Mon1 100  Mon2 78  Mon3 52
    Tue1 100  Tue2 78  Tue3 52
    Wed1 100  Wed2 78  Wed3 52
    Thu1 100  Thu2 78  Thu3 52
    Fri1 100  Fri2 78  Fri3 52
    Sat1 100  Sat2 78 ;
AMPL

Solver independent of model & data

AMPL: model sched1.mod;
AMPL: data sched.dat;
AMPL: let least_assign := 7;
AMPL: option solver cplex;
AMPL: solve;

CPLEX 12.2.0.0: optimal integer solution; objective 266
419 MIP simplex iterations
39 branch-and-bound nodes

AMPL: option omit_zero_rows 1, display_1col 0;
AMPL: display Work;

Work [*] :=
  3  7  18  9  37  7  66  7  82  16  112  23  124  15
  6 21  20  7  41  9  72 13  91  20  118  29
16 13  29  7  53 13  78 20  94  9  122  21
;

Robert Fourer, Modeling & Solving Nontraditional Optimization Problems
Session 1a: Background — Chiang Mai, 4-5 January 2011
AMPL

Language independent of solver

```
ampl: option solver gurobi;
ampl: solve;

Gurobi 4.0.0: optimal solution; objective 266
857 simplex iterations
29 branch-and-cut nodes

ampl: display Work;
Work [*] :=
   1  21  21 36  52  7  89 29  94  7 109 16  124 36
   3  7  37 29  71 13  91 16  95 13 116 36
;
```
Nonlinear network example: *symbolic data*

```AMPL
set INTERS;
param EN symbolic;
param EX symbolic;
    check {EN,EX} not within INTERS;
set ROADS within {INTERS union {EN}, INTERS union {EX}};
param time {ROADS} > 0;
param cap {ROADS} > 0;
param sens {ROADS} > 0;
param through > 0;
```
AMPL

Algebraic modeling language: symbolic model

\[
\text{var } \text{Flow } \{(i,j) \text{ in ROADS}\} \geq 0, \leq 0.9999 \times \text{cap}[i,j]; \\
\text{var } \text{Time } \{\text{ROADS}\} \geq 0; \\
\text{minimize } \text{Avg\_Time}: \\
\quad (\text{sum } \{(i,j) \text{ in ROADS}\} \text{Time}[i,j] \times \text{Flow}[i,j]) / \text{through}; \\
\text{subject to } \text{Travel\_Time } \{(i,j) \text{ in ROADS}\}: \\
\quad \text{Time}[i,j] = \text{base}[i,j] + (\text{sens}[i,j] \times \text{Flow}[i,j]) / (1 - \text{Flow}[i,j] / \text{cap}[i,j]); \\
\text{subject to } \text{Balance\_Node } \{i \text{ in INTERS}\}: \\
\quad \text{sum}\{(i,j) \text{ in ROADS}\} \text{Flow}[i,j] = \text{sum}\{(j,i) \text{ in ROADS}\} \text{Flow}[j,i]; \\
\text{subject to } \text{Balance\_Flow}: \\
\quad \text{sum}\{(EN,j) \text{ in ROADS}\} \text{Flow}[EN,j] = \text{through};
\]
### AMPL

**Explicit data independent of symbolic model**

<table>
<thead>
<tr>
<th>set INTERS := b c ;</th>
</tr>
</thead>
<tbody>
<tr>
<td>param EN := a;</td>
</tr>
<tr>
<td>param EX := d;</td>
</tr>
<tr>
<td>param: ROADS: base cap sens :=</td>
</tr>
<tr>
<td>a b 5 10 .1</td>
</tr>
<tr>
<td>a c 1 30 .9</td>
</tr>
<tr>
<td>c b 2 10 .9</td>
</tr>
<tr>
<td>b d 1 30 .9</td>
</tr>
<tr>
<td>c d 5 10 .1 ;</td>
</tr>
<tr>
<td>param through := 4;</td>
</tr>
</tbody>
</table>
AMPL

Solver independent of model & data

```plaintext
amat: model traffic_c.mod;
amat: data traffic_c.dat;
amat: option solver minos;
amat: solve;

MINOS 5.51: optimal solution found.
7 iterations, objective 8.178571429
Nonlin evals: obj = 16, grad = 15, constrs = 16, Jac = 15.
amat: display Flow, Time;

:    Flow   Time    :=
a b   2 5.25
a c   2 2.92857
b d   2 2.92857
c b   0 2
b c   0 2
```
AMPL

Language independent of solver

```
AMPL: model traffic_c.mod;
AMPL: data traffic_c.dat;
AMPL: option solver knitro;
AMPL: solve;

KNITRO 6.0.0: Locally optimal solution.
objective 8.178571522; feasibility error 3.73e-07
4 iterations; 5 function evaluations

AMPL: display Flow, Time;
:        Flow        Time      :=
  a b   2             5.25
  a c   2             2.92857
  b d   2             2.92857
  c b  4.13927e-07   2
  c d   2             5.25
;
```
NEOS www.neos-server.org

Network Enabled Optimization System

- Guide
  - tutorials, case studies, test problems, FAQs
- Server
  - free Internet access to solvers
NEOS Server

Since 1995 . . .

- Hosted at Argonne National Laboratory (Illinois, USA)
- Developed through 5 major releases
  * many contributors @ Argonne, Northwestern & elsewhere
  * increasingly sophisticated as Web has matured
- 10-20,000 server submissions in a typical month
  . . . has handled over 100,000

A research project

- Currently free of charge
- Supported by grants & volunteer efforts
- Moved in December . . .
NEOS @ WID

*Wisconsin Institutes for Discovery* (discovery.wisc.edu)
- Wisconsin Institute for Discovery (public)
- Morgridge Institute for Research (private)

*Key participants*
- Michael Ferris
  - research theme leader, *optimization in biology & medicine*
  - coordinator of NEOS move
- Miron Livny
  - founder of the Condor distributed-computing project
  - coordinator of computing technology for WID
Design

Flexible architecture
- Central controller and scheduler machine
- Distributed solver sites

Standard formats
- Low-level formats: MPS, SIF, SDPA
- Programming languages: C/ADOL-C, Fortran/ADIFOR
- High-level modeling languages: AMPL, GAMS

Varied submission options
- E-mail
- Web form
- Direct call via XML-RPC
  - from AMPL or GAMS client (*Kestrel*)
  - from user’s client program using NEOS’s API

... server processes submissions of new solvers, too
**Using NEOS**

**Learn About Your Problem**

**The NEOS Guide**

- Optimization tree: Problem types
- Optimization software guide: Individual solvers
- Frequently asked questions: Varied listings & advice

---

The Optimization Tree is an online guide to the field of numerical optimization. It introduces the different subfields of optimization and includes outlines of the major algorithms in each area, with pointers to software packages where appropriate. The connections between the Tree’s web pages mirrors the relationships between these different areas. Follow the pathways through the tree to see how everything hangs together!

If you’d like to contribute a description of one of the areas that we don’t presently cover, please get in touch with us.

Material in the Tree can also be accessed through the search facility.

Text only version of the Optimization Tree.
Using NEOS

Investigate Solvers

NEOS Server home page

Our optimization solvers represent the state-of-the-art in optimization software. Optimization problems are solved automatically with minimal input from the user. Users only need a definition of the optimization problem. All additional information required by the optimization solver is determined automatically.

- User Feedback
- FAQ - NEOS Server
- Aknowledgments
- Collaboration

To submit your optimization job, first click on the NEOS Solvers icon to find a suitable solver.

NEOS Information

- Kestrel modeling language interface to the NEOS Server
- JAVA Submission Tool
- The NEOS Server 4.0 package
- NEOS Guide
- installNEOS
- Optimization Software Guide
- Frequently Asked Questions on Linear & Nonlinear Programming
- NEOS Server Web stats since January 1, 2003
- NEOS Server Web stats for the past month

Get neos-news!

Enter your email address: 

Subscribe to neos-news

neos-news provides occasional NEOS-related announcements
Using NEOS

Investigate Solvers

NEOS Server home page (new @ WID)
Using NEOS

Investigate Solvers

NEOS Server solver type listing

Choose the type of optimization problem that you want to solve to see a list of solvers. If you are not sure of the type of optimization problem, consult the Optimization Tree of the NEOS Guide for information on optimization problems. The choice of solver is then dictated by the language used to define the optimization problem. You can also choose your solver by input format.

Each solver has sample problems and background information on the solver. Be sure to submit a sample problem to get a feel for how to submit optimization problems to NEOS. If you encounter problems, consult the NEOS Server FAQ or send mail to us by clicking on the Comments and Questions icon at the bottom of the page.
### Investigate Solvers

**NEOS Server solver listing**

<table>
<thead>
<tr>
<th>Category</th>
<th>Solvers</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Nonlinearly Constrained Optimization</strong></td>
<td>CONOPT [GAMS Input]</td>
</tr>
<tr>
<td></td>
<td>DONLP2 [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>FILTER [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>KNTIRO [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>LANCELOT [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>LOQO [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>MINOS [AMPL Input] [GAMS Input]</td>
</tr>
<tr>
<td></td>
<td>MOSEK [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>PATHLP [GAMS Input]</td>
</tr>
<tr>
<td></td>
<td>SNOPT [Fortran Input] [AMPL Input] [GAMS Input]</td>
</tr>
<tr>
<td><strong>Semidefinite &amp; Second Order Cone Programming</strong></td>
<td>CSDDP [Matlab Binary Input] [Sparse SDPA Input]</td>
</tr>
<tr>
<td></td>
<td>CIRCUIT [Graph Input]</td>
</tr>
<tr>
<td></td>
<td>DDSDP [Sparse SDPA Input]</td>
</tr>
<tr>
<td></td>
<td>MOSEK [Matlab Binary Input] [MPS Input]</td>
</tr>
<tr>
<td></td>
<td>FENNON [Sparse SDPA Input]</td>
</tr>
<tr>
<td></td>
<td>SDPA [Graph Input]</td>
</tr>
<tr>
<td></td>
<td>SDPA [Sparse SDPA Input]</td>
</tr>
<tr>
<td></td>
<td>SDPT3 [Matlab Binary Input] [Sparse SDPA Input]</td>
</tr>
<tr>
<td></td>
<td>SEDUMI [Matlab Binary Input] [Sparse SDPA Input]</td>
</tr>
<tr>
<td><strong>Linear Programming</strong></td>
<td>BIBMPL [GAMS Input]</td>
</tr>
<tr>
<td></td>
<td>BMIPOP [LP Input] [MPS Input] [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>Fomsdp [MPS Input] [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>MOSEK [MPS Input] [AMPL Input] [LP Input]</td>
</tr>
<tr>
<td></td>
<td>OQPOP [AMPL Input] [MPS Input]</td>
</tr>
<tr>
<td></td>
<td>PCX [MPS Input] [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>XPRESS-Mp-BARRIER [MPS Input]</td>
</tr>
<tr>
<td></td>
<td>XPRESS-Mp-SIMPLEX [MPS Input]</td>
</tr>
<tr>
<td><strong>Semidefinite Constrained Optimization</strong></td>
<td>BLMVM [C Input] [Fortran Input] [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>L-BFGS-B [Fortran Input] [AMPL Input]</td>
</tr>
<tr>
<td></td>
<td>TRON [Fortran Input] [GAMS Input] [AMPL Input]</td>
</tr>
</tbody>
</table>
Using NEOS

Investigate Solvers

Individual solver listing

**KNITRO (AMPL input)**

The NEOS Server offers KNITRO (formerly NITRO) for the solution of nonlinearly constrained optimization problems in AMPL format. KNITRO is a primal-dual interior-point method which uses trust regions.


KNITRO was developed by Richard Byrd, Mary Beth Hribar, Jorge Nocedal and Richard Waltz with additional help from Guangxi Liu, Marcelo Martnez, Todd Plantenga and Jose Luis Morales.

Additional information on KNITRO (including information on how to obtain KNITRO) can be found at the [KNITRO website](http://www-neos.mcs.anl.gov/).
Using NEOS

Try a Solver: Web Interface

Sample submission form

- “Comments and Questions” button on every page
Using NEOS

Try a Solver: Web Interface

Submission form for your problem

WWW Interface

KNITRO (AMPL input)

The user must submit a model in AMPL format to solve a nonlinearly constrained optimization problem. Examples of models in AMPL format can be found in the model collection.

The model is specified by a model file, and optionally, a data file and a commands file. If the commands file is specified it must contain the AMPL solve commands.

The commands file can contain any AMPL command or set options for KNITRO. Printing directed to standard out is returned to the user with the output.

Enter the AMPL model.

Enter the AMPL data (optional).

Enter the AMPL commands (optional).
Using NEOS

Try a Solver: Web Interface

Start of your run
Try a Solver: Web Interface

Beginning of your solution listing

******************************************************************
NEOS Server Version 4.0

Job #: 240233
Solver: KENMERO (AMPL input)
Start: 03/25/2003 12:58:51
End: 03/25/2003 12:58:58
Host: ioncube.ieee.northwestern.edu

Restful Interface: check out our new client for sending jobs
To the NEOS Server from your AMPL or GAMS modeling session
and receiving results back into the session for further
computation. See http://www-neos.mcs.anl.gov/neos/kestrel.html

Feedback Requested!
Our funding agency has asked for a report on our work, and
feedback from our users would be greatly appreciated. The
information of interest is the type of work you are doing
on NEOS, whether for business, education, or other purposes,
and the solvers used.
Please send your comments to dwarm@anl.gov, mare@anl.gov.

Disclaimer:
This information is provided without any express or
implied warranty. In particular, there is no warranty
of any kind concerning the correctness of this
information for any particular purpose.

******************************************************************

***KENMERO (AMPL input)***

21 variables, all nonlinear
18 constraints, all linear; 42 onezeros
1 nonlinear objective; 21 onezeros.
Using NEOS

Try a Solver: Web Interface

End of your solution listing
Using NEOS

Try a Solver: Kestrel Interface

Kestrel client download page

Kestrel

The Kestrel client/server is a way of sending your optimization job to be solved via the NEOS Server from within your modeling environment and receiving results that can be interpreted directly by your modeler.

<table>
<thead>
<tr>
<th>Modeling Environment</th>
<th>Currently available versions of the Kestrel client</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMPL</td>
<td>linux, solana, windows</td>
</tr>
<tr>
<td>GAMS</td>
<td>linux, solana, windows</td>
</tr>
</tbody>
</table>

*Tested on Red Hat Linux 6.0, Solaris 7, Windows NT and Windows 98
*These Kestrel clients are tarred compressed with gzip.

A complete users guide to Kestrel is available in PostScript and PDF.

Short guide to using the Kestrel AMPL client

Installation:

- Give the binary for your system the name kestrel and make certain it is in your execution path.

Within the AMPL Environment:

- Design your model as you normally would. When choosing options, everything should remain as per usual with the following exceptions:
  - Choose option solver kestrel instead of the usual solver name.
  - Choose the solver you want with option kestrel options, solver=solvername.
Using NEOS

Try a Solver: Kestrel Interface

Applying a local solver to an AMPL model

```
sw: ampl -v
AMPL CPL beta Version 20020516 (MS VC++ 6.0)

ampl: model gs2000b.mod;
ampl: data gs2000b.dat;
ampl: option solver minos;
ampl: option show_stats 1;
ampl: solve;

Press value eliminates 100 constraints.
Adjusted problem:
4290 variables:
   4260 binary variables
   30 linear variables
733 constraints, all linear; 36340 nonzeros
1 linear objective; 30 nonzeros.

MINOS 5.5:
Sorry, the student edition is limited to 300 variables and
300 constraints. You have 4290 variables and 733 constraints.

exit code 1
<BREAK>
ampl: |
```
Using NEOS

Try a Solver: Kestrel Interface

Applying a NEOS solver to an AMPL model . . .

```ampl
sw: running ampl

sw: ampl -v
AMPL Version 20000016 <MS VC++ 6.0>

ampl: model gs2000b.mod;
ampl: data gs2000b.dat;

ampl: option solver kestrel;
ampl: option kestrel_options 'solver=logo';
ampl: option logo_options 'minlocfile outlo=1';

ampl: option show_stats 1;
ampl: solve;

Presolve eliminates 100 constraints.
Adjusted problem:
4290 variables:
   4260 binary variables
  30 linear variables
733 constraints, all linear; 36340 nonzeros
1 linear objective; 30 nonzeros.

Job has been submitted to Kestrel
Kestrel/NEOS Job number  : 115406
Kestrel/NEOS Job password : GKiKgUu

Check the following URL for progress report:
   http://www-neos.mcs.anl.gov/neos/neos-cgi/check-status.cgi?job=115406&pass=GKiKgUu

In case of problems, e-mail:
   neos-comments@mcs.anl.gov
```
Using NEOS

Try a Solver: Kestrel Interface

... and receiving a solution from the NEOS

Check the following URL for progress report:
http://www-neos.mcs.anl.gov/neos/neos-cgi/check-status.cgi?job=115406&pass=KskkXgJu

In case of problems, e-mail:
neos-comments@mcs.anl.gov

Intermediate Solver Output:
Executing algorithm...

LOQO 6.00: minlocfil
outlev=1

It's a QP.
ignoring integrality of 4260 variables

1  0.000000e+00  2.1e+02  -4.263593e+05  1.7e+03
2  2.839512e+03  1.1e+01  -4.286438e+05  8.8e+01
3  2.839512e+03  5.8e+01  -3.084429e+05  3.7e+00
4  1.804900e+03  7.0e-02  -2.965997e+04  1.4e-13
5  3.154594e+02  1.1e-02  -3.913235e+03  1.7e-13
6  3.771029e+01  1.2e-03  -2.281994e+02  4.6e-14
7  2.725864e+01  6.4e-04  -1.672544e+01  3.6e-14
8  1.780800e+01  3.1e-04  2.429905e+00  2.9e-14
9  1.536949e+01  1.4e-04  9.410129e+00  4.6e-14
10  1.546499e+01  4.4e-05  1.271394e+01  3.8e-14
11  1.495958e+01  2.4e-06  1.326864e+01  3.6e-14
12  1.400626e+01  1.4e-07  1.396300e+01  3.5e-14
13  1.400016e+01  7.3e-09  1.399815e+01  3.9e-14
14  1.400000e+01  3.6e-10  1.399991e+01  3.5e-14
15  1.400000e+01  1.8e-11  1.400000e+01  4.1e-14
16  1.400000e+01  9.1e-13  1.400000e+01  3.5e-14

Finished call

LOQO 6.00: optimal solution <16 QP iterations, 31 evaluations>
primal objective 14.000000002
dual objective 13.999999999

ampl: display MinNotDom, MaxNotDom;
        MinNotDom MaxNotDom ::
Office Americas       3     4

ampl:
Using NEOS

Try a Solver: Kestrel Interface

Web form for checking your run’s status

Welcome to NEOS!

Scheduling:
You are job #115407.
Solver Queue:

KESTREL AMPL:LOGO: 115407:

Jobs Executing:
job#108154 executing on utrims1.ics.uci.edu-NEOS-MINOS-AMPL.
job#115401 executing on calit2.edu-NEOS-MILP_SOLVER.
job#115402 executing on calit2.edu-NEOS-MILP_SOLVER.
Using NEOS

Try a Solver: Kestrel Interface

Intermediate status listing
Using NEOS

Try a Solver: Kestrel Interface

Final result listing

```plaintext
Job Status NEOS - Netscape

job_client.pl: alarm in 604000 seconds
job_client.pl: connecting to hermitage.cs.unc.edu:4012
job_client.pl: connected
job_client.pl: sending request
job_client.pl: 103146 bytes sent
job_client.pl: receiving data

connn-daemon.pl: downloading user data.............
connn-daemon.pl: uncompressing...
connn-daemon.pl: untarring...
connn-daemon.pl: launching KESTREL_AMPL:LOOQ driver...

------Begin Standard output/error------
Checking the AMPL files
Executing algorithm...
LOGO 6.0: minocfil
outlevel
It's a GP.
Inequality inequality of 4560 variables
1  1.4000000000  2.0e+03  2.655000e+05  1.7e+03
2  2.8415905e+03  1.1e+01  4.206150e+05  8.4e+01
3  2.366300e+03  5.8e-01  -3.076341e+05  3.8e+00
4  1.650500e+03  6.9e-02  -2.940097e+04  5.2e-04
5  3.025400e+02  1.1e-02  -2.915923e+03  1.8e-05
6  9.700100e+01  1.2e-03  -2.138846e+02  9.3e-07
7  2.223549e+01  6.4e-04  -1.050055e+01  4.5e-08
8  1.635979e+01  3.0e-04  2.502257e+00  1.8e-00
9  1.530956e+01  1.4e-04  9.557516e+00  4.5e-09
10 1.405694e+01  4.4e-05  1.264803e+01  9.2e-10
11 1.403725e+01  2.4e-06  1.333832e+01  4.1e-10
12 1.400313e+01  1.4e-07  1.396867e+01  2.0e-11
13 1.400164e+01  7.2e-09  1.399836e+01  1.0e-13
14 1.400013e+01  3.6e-10  1.399932e+01  6.6e-14
15 1.400000e+01  1.0e-11  1.400000e+01  4.0e-14
16 1.400000e+01  9.3e-13  1.400000e+01  3.7e-14

Finished call
--------End Standard output/error--------
```

Robert Fourer, Modeling & Solving Nontraditional Optimization Problems
Session 1a: Background — Chiang Mai, 4-5 January 2011
NEOS Frequently Asked Questions

Who uses it?
- Where are its users from?
- How much is it used?

What kinds of solvers does it offer?
- Who supplies them?
- Which are most heavily used?
- Where are they hosted?

How is it supported?
- Who answers user questions?
Who Uses NEOS? (*a sample*)

- We are using NEOS services for duty-scheduling for ground handling activities in a regional airport environment.
- We used NEOS to solve nonlinear optimization problems associated with models of physical properties in chemistry.
- Our company is working with various projects concerning R&D of internal combustion engines for cars and brakes for heavy vehicles.
- We are working on bi-dimensional modeling of earth's conductivity distribution.
- 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 the best suited solver.
- I have used NEOS with LOQO solver to optimize an interpolator. . . . My domain is digital receivers where the receiver clock is not changed to match the transmitter clock.
Who Uses NEOS? (more)

- I have been able to build and solve a prototype combinatorial auction MIP model using AMPL and NEOS in a fraction of the time it would have required me to do this had I needed to requisition a solver and install it locally.

- Our idea is trying to design antennas by using the computer. . . . We have tried various solvers on NEOS to see if this is possible at all.

- I am using the LOQO solver and code written in AMPL to perform numerical optimization of a spinor Bose-Einstein condensate.

- We are using the NEOS Server for solving linear and nonlinear complementarity problems in engineering mechanics and in robotics.

- I have been working on a system for protein structure prediction. . . . I had need to incorporate a nonlinear solver to handle packing of sidechain atoms in the protein.

... more at www-neos.mcs.anl.gov/neos/stories.html
**NEOS Users**

**Where are They From?**

*Standard domains*

![Bar Chart]

(2010 through October)
Where are They From?

Country domains (< 40000)

(2010 through October)
NEOS Users

Where are They From?

Country domains (< 4000)

(2010 through October)
Where are They From?

Country domains (< 400)

(2010 through October)
NEOS Users

How Much Do They Use It?

Monthly rates since 1999

20000/month ≈ 25/hour
NEOS Users

How Much Do They Use It?

Monthly rates for past year

20000/month ≈ 25/hour
What Solvers Does NEOS Offer?

For familiar problem types
- Linear programming
- Linear network optimization
- Linear integer programming
- Nonlinear programming
- Stochastic linear programming
- Complementarity problems

For emerging problem types
- Nondifferentiable optimization
- Semi-infinite optimization
- Global optimization
- Nonlinear integer programming
- Semidefinite & 2nd-order cone programming
- ... virtually every published semidefinite programming code
NEOS Solvers

Who Supplies Them?

Some commercial solver vendors
- Xpress-MP, MOSEK, FortMP (mixed integer)
- CONOPT, KNITRO, MOSEK (nonlinear)

Universities and their researchers
- BonsaiG (mixed integer)
- DONLP2, LANCELOT, LOQO, MINOS, SNOPT (nonlinear)

Open-Source Enthusiasts
- GLPK, CBC, Bonmin (mixed integer)

with thanks to . . .
- AMPL and GAMS developers
- Hans Mittelmann, Arizona State
**NEOS Solvers**

Which are Most Heavily Used?

**Solver submissions (< 50000)**

(2010 through October)
NEOS Solvers

Which are Most Heavily Used?

Solver submissions (< 5000)

(2010 through October)
NEOS Solvers

Which are Most Heavily Used?

Solver submissions (< 500)

(2010 through October)
Where are They Hosted?

Varied workstations at

- Aachen University of Technology, Germany
- Argonne National Laboratory
- Arizona State University
- Lehigh University
- National Taiwan University
- Universidade do Minho, Portugal
- University of Wisconsin at Madison

... new hosts readily added anywhere on the Internet
How is NEOS Supported?

**Grants**

- National Science Foundation, Operations Research Program, grant DMI-0322580
- National Science Foundation, Information Technology Research Program, grant CCR-0082807
- National Science Foundation, Challenges in Computational Science Program, grant CDA-9726385

**Donations**

- Processor cycles
- Many people’s time
Who Answers Users’ Questions?

Large mailing list for support questions
  - NEOS developers
  - Solver developers

Support request buttons on every page
NEOS Limitations

Limited choices for MIP
  ❖ But now offers Gurobi solver

Limited input standardization
  ❖ Some AMPL, some GAMS
  ❖ Varied low-level formats

Limited support
  ❖ Maintenance
  ❖ Computing power

Limited funding model
  ❖ Grants?
  ❖ User fees?

... recent move may change things!
To Learn More . . .

Websites

- www.neos-server.org

Overview


Modeling system interfaces

COIN-OR www.coin-or.org

Computational Infrastructure for Operations Research

- Repository for open-source software for optimization
- Mission:
  * Develop, manage & distribute
  * OR software, models, and data so that
  * OR professionals can benefit from
  * peer-reviewed, archived, openly-disseminated software
- Strongest in optimization

Since 2000 . . .

- Origins at IBM
- Transferred to nonprofit COIN-OR Foundation
- Hosted by INFORMS
Open-Source Software

Things to know

- Free, but subject to licensing restrictions
- Licenses vary considerably
- Equally available to all user classes
- Possibly owned (in part) by
  - Co-authors
  - Employer
  - Granting Agencies
  - Owner of the machines it was developed on

Examples

- GNU Public License, GNU Library Public License
- Common Public License, Eclipse Public License
- Mozilla Public License
- Apache License
- BSD Licenses
Open-Source Software at COIN-OR

Solvers
- Linear continuous & discrete
- Nonlinear continuous & discrete
- Semidefinite
- Stochastic

... source & binary

Infrastructures
- Developer tools
- Optimization utilities
- Interfaces
- Modeling systems and environments
Open-Source Software at COIN-OR