High Performance Software for Nonlinear Optimization
Status and Perspectives
Ischia, Italy
June 4–6, 1997

The NEOS Server
Optimization, Environments, Network Computing

Jorge J. Moré

Mathematics and Computer Science Division
Argonne National Laboratory
http://www.mcs.anl.gov/home/more/

NEOS (Network-Enabled-Optimization-System) Server

A novel environment for solving optimization problems.

- Internet-based access to a library of optimization software.
- Submissions via email, a browser, or the `neos-submit` tool.
- Users only need to submit a specification of the problem.
- The NEOS Server provides all other information required by the solver.

NEOS (Network-Enabled-Optimization-System) Server

Optimization software

Software tools

- Perl
- Tcl/Tk

Automatic differentiation

- ADIFOR
- ADOL-C

Heterogeneous distributed computing

- Condor
- Globus

Typical Environment for Nonlinear Optimization

$$\min \{ f(x) : x_l \le x \le x_u, \ c_l \le c(x) \le c_u \}$$

Tasks

- Develop code to evaluate the objective function $f : \mathbb{R}^n \mapsto \mathbb{R}$ and constraint $c : \mathbb{R}^n \mapsto \mathbb{R}^m$
- Decide on the software
- Retrieve the software
- Hand-code the gradients, Hessians, sparsity patterns.
- Compile, debug, compile, debug, ... run
- Interpret the results
- Repeat the process

The NEOS Way

- Develop code to evaluate the objective function $f : \mathbb{R}^n \mapsto \mathbb{R}$ and constraint $c : \mathbb{R}^n \mapsto \mathbb{R}^m$
- Submit problem to the **NEOS** Server
- Interpret the results
- Repeat the process

Advantages

Easy to use
No need to choose or retrieve software
No need to compute derivatives or sparsity patterns

NEOS

A research effort of the Optimization Technology Center, initiated by Argonne National Laboratory and Northwestern University.

NEOS Guide: A Web-based guide to optimization theory and practice.

NEOS Server: An Internet-based client/server application that provides access to a library of optimization software.

NEOS Guide

- Optimization Software Guide
- Optimization Tree
- Case Studies
- Frequently Asked Questions

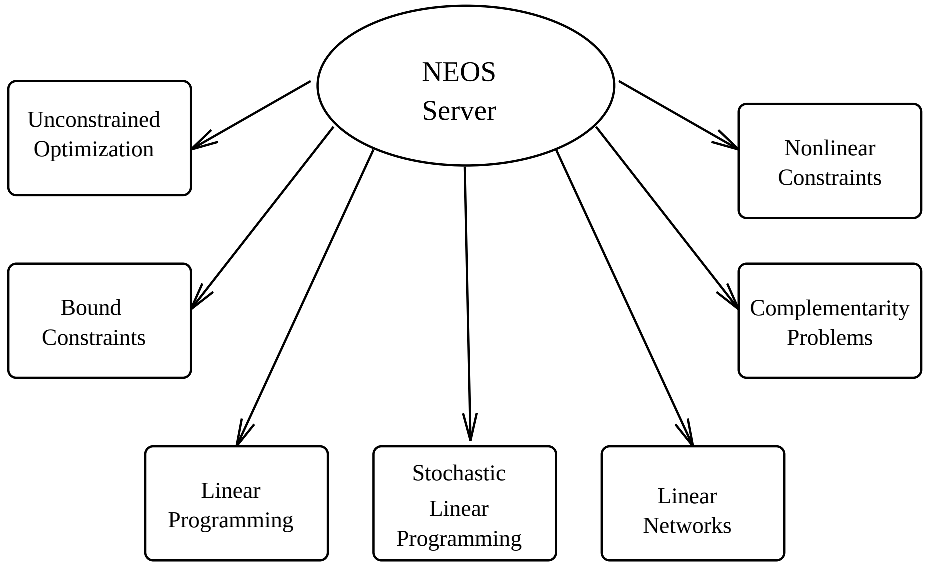
NEOS Collaborations

- Michael Ferris, University of Wisconsin
- Nick Gould, Rutherford Appleton Laboratory
- David Serafini, Rice University
- Jorge Nocedal, Northwestern University
- Joe Czyzyk and Steve Wright, Argonne National Laboratory
- Dan Bienstock, Columbia University
- Phil Gill, University of California, San Diego
- Jorge Moré, Argonne National Laboratory

Outline

- What optimization problems can we solve?
- How are optimization problems specified?
- How are optimization problems submitted?
- How are optimization problems solved automatically?
- How do we add optimization solvers to the NEOS Server?

NEOS Server Solvers



NEOS Server Submissions

NEOS Submission tool

- Socket-based; results returned via `neos-submit`.
- `http://www.mcs.anl.gov/otc/Server/neos/subtool.html`

Web Browser

- Web-based; results returned via browser or email.
- `http://www.mcs.anl.gov/otc/Server/`

Email

- Email-based; results returned via email.
- `neos@mcs.anl.gov`

Specifying Constrained Optimization Problems

$$\min\left\{f(x) : x_l \leq x \leq x_u,\ c_l \leq c(x) \leq c_u\right\}$$

$$f(x) = \sum_{i=1}^{n_f} f_i(x)$$

Dimensions: `n, m`

Number of element functions: `nf`

Initial point: `initpt(n,x)`

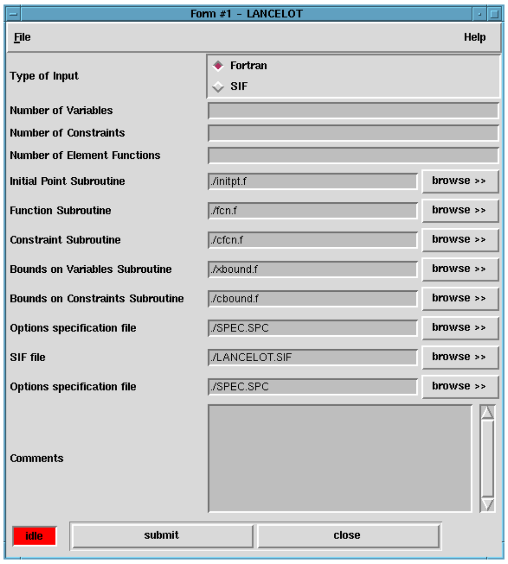
Objective function: `fcn(n,x,nf,fvec)`

Constraint function: `cfcn(n,x,m,c)`

Bounds on the variables: `xbound(n,xl,xu)`

Bounds on the general constraints: `cbound(m,cl,cu)`

NEOS Submission Tool – LANCELOT



NEOS Email Submission – NMTR

Send the following message to `neos@mcs.anl.gov`

```
type UCO
solver NMTR
n = 100
begin.initpt
```

Subroutine `initpt(n,x)` to evaluate the starting point

```
end.initpt
begin.fcn
```

Subroutine `fcn(n,x,f)` to evaluate $f : \mathbb{R}^n \mapsto \mathbb{R}$

```
end.fcn
```

NEOS Submission – `initpt`

```
subroutine initpt(n,x)
integer n
double precision x(n)

integer nx, ny
nx = 10
ny = 10
call odc_init(nx,ny,x)
end

subroutine odc_init(nx,ny,x)

.....

end
```

NEOS Submission – `fcn`

```
subroutine fcn(n,x,f)
integer n
double precision f
double precision x(n)

integer nx, ny
double precision lambda
lambda = 0.008
nx = 10
ny = 10
call dodc(nx,ny,x,f,lambda)
end

subroutine dodc(nx,ny,x,f,lambda)

.....

end
```

Performance: Optimal Design with Composites Problem

Determine a minimizer $v : \mathbb{R}^2 \mapsto \mathbb{R}$ of the problem

$$\min\{f_\lambda(v) : v \in H_0^1(\mathcal{D})\},$$

where

$$f_\lambda(v) = \int_{\mathcal{D}} \left\{ \psi_\lambda \left(\|\nabla v(x)\| \right) + v(x) \right\} dx$$

$$\psi_\lambda(t) = \begin{cases} \frac{1}{2}\mu_2 t^2, & 0 \le t \le t_1, \\ \mu_2 t_1 (t - \frac{1}{2} t_1), & t_1 \le t \le t_2, \\ \frac{1}{2}\mu_1 (t^2 - t_2^2) + \mu_2 t_1 (t_2 - \frac{1}{2} t_1), & t_2 \le t, \end{cases}$$

$$t_1 = \left(2\lambda \frac{\mu_1}{\mu_2} \right)^{\frac{1}{2}}, \qquad t_2 = \left(2\lambda \frac{\mu_2}{\mu_1} \right)^{\frac{1}{2}}.$$

Discretization of the ODC Problem

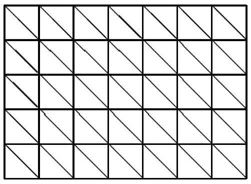
A finite element formulation leads to

$$\min \{ f(x) : x \in \mathbb{R}^n \}$$

with `nx*ny` variables. The mapping $f : \mathbb{R}^n \to \mathbb{R}$ is a partially separable function

$$f(x) = \sum_{i=1}^{m} f_i(x),$$

where $f_i : \mathbb{R}^n \to \mathbb{R}$ only depends on 3 variables.



Performance of the NEOS Solver vmlm

Optimal design problem

`dodc(nx,ny,x,nf,fvec,lambda)`

	Function			Function and Gradient		
n	`niters`	`nfgev`	`time`	`niters`	`nfgev`	`time`
2,500	230	237	139	232	239	22
10,000	427	436	1042	427	433	165
40,000	865	871	8399	877	885	1461

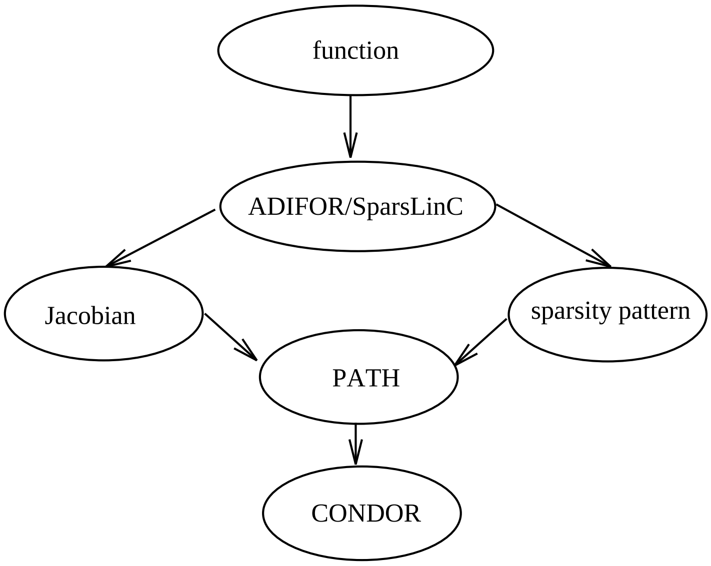
Computing time for the function/gradient grows linearly with n

Issues for Problem-Solving Environments

What if ...

- A subroutine does not compile?
- The calling sequence is wrong?
- The function evaluation blows up at the starting point?
- The function evaluation blows up at an iterate?
- The function evaluation is garbage?
- The function is non-differentiable?
- The user wants to write out information?

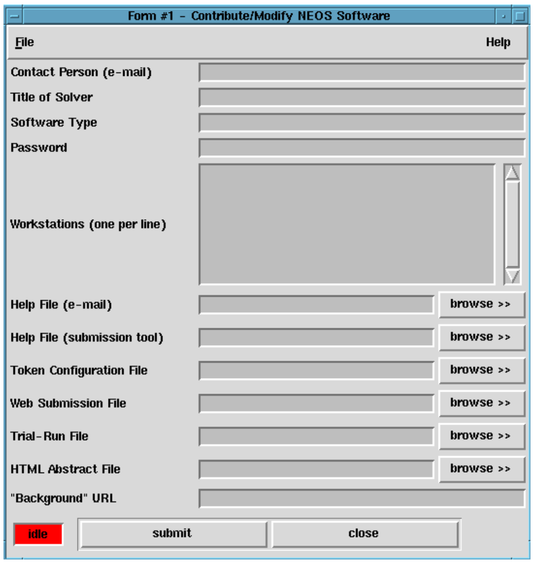
Handling Complementarity Problems: PATH



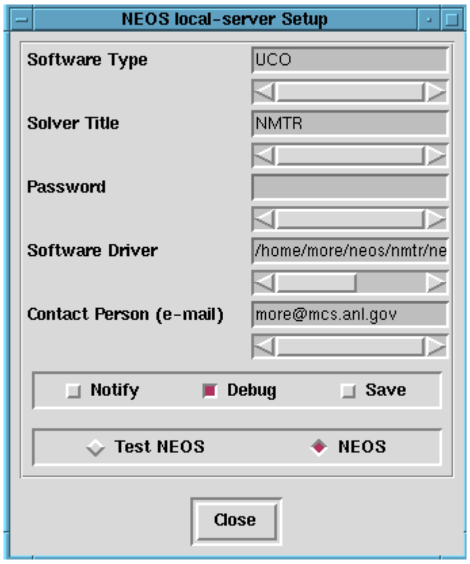
NEOS Server Issues

- Registering software – The NEOS Submission Tool
- Registering software – The NEOS Communications Tool
- Monitoring traffic – The NEOS Monitor
  - Job number
  - Type of job and interface used to submit job
  - Date and user name

Registration of Software: NEOS Submission tool



Registration of Software: NEOS Communications tool

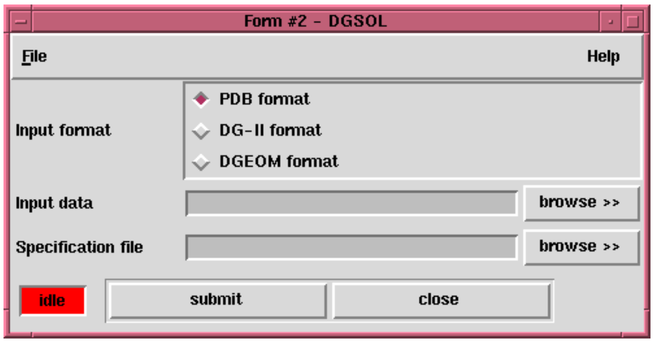


Registration of Software: Adding Solvers

Problem: Determine $x_1, \ldots, x_m$ in $\mathbb{R}^3$ such that

$$l_{i,j} \le \|x_i - x_j\| \le u_{i,j}, \qquad (i,j) \in \mathcal{S}$$

Data: $(i, j, l_{i,j}, u_{i,j}), \qquad (i,j) \in \mathcal{S}$



Uses of the NEOS Server

- Solving large optimization problems
- Education and information on optimization technology
- Classroom exercises
- Testing optimization software
- Comparing optimization software
- Prototype problem solving environment

Future Directions for the NEOS Server

- Handling C-programs (ADOL-C and ADIC)
- Distributed computing (remote procedure calls)
- Global optimization
- Problem-solving environment

Additional Information

J. Czyzyk, M. Mesnier, and J. Moré,
The Network-Enabled Optimization System (NEOS) Server,
Preprint MCS-P615-0996, September 1996.

W. Gropp and J. Moré,
Optimization environments and the NEOS Server, Preprint
MCS-P654-0397, March 1997.

NEOS Server `http://www.mcs.anl.gov/otc/Server/`
NEOS Guide `http://www.mcs.anl.gov/otc/Guide/`