# The Common Optimization Interface for Operations Research (COIN-OR) Project

Ted Ralphs
Industrial and Systems Engineering
Lehigh University
http://www.lehigh.edu/~tkr2

International Federation of Operational Research Societies Conference, Tuesday July 9, 2002

# Outline of Talk

- Introduction
- Project Goals
- Open Source Primer
- Components of COIN-OR
- The Future of COIN-OR
- How to Help
- Resources

# The State of Computational Research

- The majority of the papers being written in OR today are computational in nature or have a computational component.
- Despite this, the pace of computational research is slow and the transfer of knowledge to practitioners is even slower.
- Results of research are generally not reproducible.
- Research codes are buggy, narrowly focused, and lack robustness.
- There are few rewards for publishing software outside of archival journals.
- There is no peer review process for software and referees of computational papers have little to go on.
- Building on previous results is difficult and time-consuming.
- Interoperating with other software libraries (such as LP solvers) is difficult.
- The paradigm encouraged by archival journals does not work for computational research.

# Lifecycle of a Typical Software Package

1. Idea is developed (the easy part).
2. Software is developed containing numerous tweaks and hacks that make the idea work (the time-consuming part).
3. Because of space constraints, the tweaks and hacks are not described in the paper.
4. Afterward, the software is not documented or maintained because there is no incentive for doing so.
5. The only thing archived is what was written in the paper, which is not enough to reproduce the results.
6. If someone else wants to use the idea, they have to repeat 90% of the work again.

# What is COIN-OR?

- The COIN-OR Project
  - A consortium of researchers in both industry and academia dedicated to improving the state of computational research in OR.
  - An initiative promoting the development and use of interoperable, open-source software for operations research.
- The COIN-OR Repository
  - A library of interoperable software tools for building optimization codes, as well as a few stand alone packages.
  - A venue for peer review of OR software tools.
  - A development platform for open source projects, including a CVS repository.
- See `www.coin-or.org` for more information.

# Who is COIN-OR?

List of primary contributors:

- Francisco Barahona
- Andrew R Conn
- Marta Eso
- JP Fasano
- John Forrest
- Lou Hafer
- Laszlo Ladanyi
- Robin Lougee-Heimer
- Tobias Pfender
- Ted Ralphs
- Matthew J Saltzman
- Katya Scheinberg
- Andreas Waechter

# Our Agenda

- Accelerate the pace of research in computational OR.
  - Reuse instead of reinvent.
  - Reduce development time and increase robustness.
  - Increase interoperability (standards and interfaces).
- Provide for software what the open literature provides for theory.
  - Peer review of software.
  - Free distribution of ideas.
  - Adherence to the principles of good scientific research.
- Define standards and interfaces that allow software components to interoperate.
- Increase synergy between various development projects.
- Provide robust, open-source tools for practitioners.

# What is Open Source?

- A coding paradigm in which development is done in a cooperative and distributed fashion.
- An economic model used by some "for-profit" software ventures.
- This model is followed by a number of well-known software projects.
  - Linux (Red Hat, etc.)
  - Netscape/Mozilla (AOL)
  - Star Office/Open Office (Sun)
  - Apache
- A type of software license (described on the next slide).
- To find out more, see the writings of Eric S. Raymond, such as *The Cathedral and the Bazaar*.

# Open Source Licenses

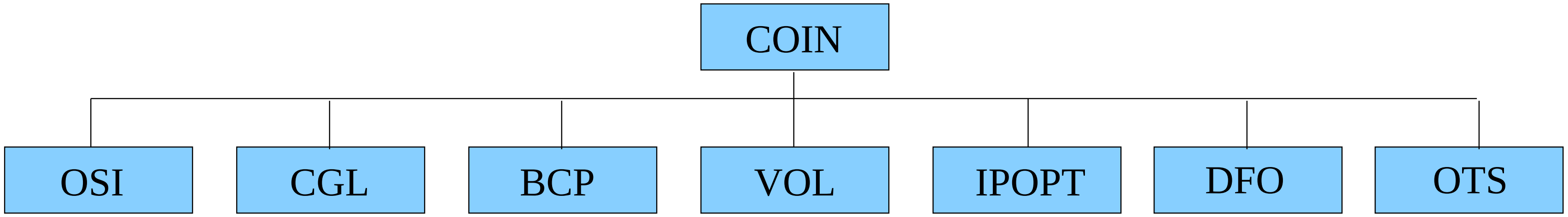
- Strictly speaking, an open source license must satisfy the requirements of the *Open Source Definition*.
- A license cannot call itself "open source" until it is approved by the Open Source Initiative.
- Basic properties of an open source license
  - Access to source code.
  - The right to redistribute.
  - The right to modify.
- The license may require that modifications also be kept open.

# Why Open Source?

- Increases the pace of development.
- Produces more robust code.
- Introduces an inherent peer review process.
- Creates an informal reward structure.
- Creates an impetus for good documentation.
- Increases the use and distribution of code.
- Prevents obsolescence.
- Promotes reuse over reimplementation.

"Given enough eyeballs, all bugs are shallow" –ESR

# Components of the COIN-OR Library




- OSI: Open Solver Interface
- CGL: Cut Generator Library
- BCP: Branch, Cut, and Price Library
- VOL: Volume Algorithm
- IPOPT: Interior Point Optimization (Nonlinear)
- DFO: Derivative Free Optimization
- OTS: Open Tabu Search

# Open Solver Interface

- Uniform API to solvers.
- Currently, linear and integer programming are supported.
- Interface to commercial codes
  - CPLEX
  - OSL
  - XPRESS-MP
- Interfaces to academic codes
  - Volume
  - DYLP
  - SOPLEX

# OSI Classes

- Sparse vector/matrix
- Cuts
- MPS reader
- Factorization
- Solver interface methods
  - create an LP formulation,
  - modify the formulation directly by adding rows/columns,
  - modify the formulation by adding cutting planes provided by CGL,
  - solve the formulation (and resolve after modification),
  - extract solution information, or
  - invoke the underlying solver's branch-and-bound component.

# Cut Generator Library

- Methods to generate cuts valid for mixed integer programs.
- Uses the OSI to access an LP relaxation and return violated cuts.
- Cut classes currently in the CGL
  - Gomory
  - Knapsack Covers
  - Lift and Project
  - Odd Holes
  - Probing
  - Simple Rounding

# Branch, Cut, and Price Library

- A state of the art parallel framework for branch, cut, and price.
- Manages LP-based branch and bound with dynamic generation of cuts and variables.
- Interfaces with the LP solver through the OSI.
- Can also interface with the CGL.
- Subproblems are collections of cuts and variables.
- User derives C++ class corresponding to each class of cuts and/or variables.
- User can override default methods to completely customize the behavior of the solver.

# Volume Algorithm

- A subgradient method that produces both primal and dual solutions.
- Can be used to obtain approximate primal solutions to linear programs quickly.
- Has an OSI interface.
- Can be warm started, just as with the simplex algorithm.
- Can be used within an LP-based branch and bound framework.
- Effective for combinatorial problems where the LP relaxations are especially difficult to solve.
- Can also be used in a Lagrangian relaxation setting.

# Interior Point Optimization

- IPOPT implements a primal-dual interior point algorithm for large-scale nonlinear optimization.
- It is a general purpose sover that does not focus on problems with a particular structure.
- It is capable of handling very large-scale instances ($O(10^6)$ variables).
- Has both AMPL and CUTEr interfaces.
- Hans Mittelmann lists IPOPT as one of the best available codes for nonlinear optimization (see `plato.la.asu.edu/bench.html`).

# Derivative Free Optimization

- DFO is used for solving general nonlinear optimization problems that have these characteristics:
  - the model is relatively small (less than 100 variables),
  - the objective function is relatively expensive to compute,
  - derivatives are not available and cannot be estimated efficiently.
- There may also be some noise in the function evaluation procedures.
- Such models arise in engineering design, where the objective function evaluation is a simulation package treated as a black box.

# Open Tabu Search

- An object-oriented tabu search framework.
- User defines the following.
  - Solution structure
  - Objective function
  - Tabu list
  - Move
  - Move manager
- OTS takes care of the rest.

# Standards and Interfaces

- New MPS standard (using XML)
- A modeling standard for stochastic programming
- Open Solver Interface (OSI)
- Cut Generator Library (CGL)

# Pending Contributions

There are a number of contributions "in the pipeline."

- Proposed
  - SYMPHONY: Single- or Multi-Process Optimization over Networks.
- Awaiting Approval
  - COIN LP Solver: Built on OSI factorization class.
  - Manifold: Package for computing implicitly defined surfaces
- Under Development
  - Advanced Library for Parallel Search (ALPS)
  - Branch, Constrain, and Price Software (BiCePS)
  - BiCePS Linear Integer Solver (BLIS)
- We are also having discussions with a number of other authors.

# The Future of COIN-OR

- COIN-OR has received a great deal of support from IBM research.
- The idea has been to try to grow it into a community-based project.
- Currently, the COIN-OR Web site and repository are hosted by IBM.
- We are looking at transferring hosting outside of IBM in 2003.
- There has been tremendous interest in the project on the part of the OR community.
- Ultimately, however, the future of COIN-OR depends on how well the community supports it.

# How You Can Help

- Contribute your software.
- Use the software that's in the repository already.
  - Help document.
  - Contribute bug fixes.
- Help with development
  - New cut classes for CGL.
  - New interface classes for OSI.
- Visit `www.coin-or.org` for more information.