Benchmarking Optimization Software
a (Hi)Story

Hans D Mittelmann

School of Mathematical and Statistical Sciences
Arizona State University

People's Republic of China
April 2019

Outline

Background

Our Service and the Rationale for Benchmarking

The History of our Benchmarking

Very Early History [1997 - 2002]

Early History [2003 - 2009]

Intermediate History [2010 - 2017]

Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

What did we learn?

What will we do?

What are the others doing?

Outline

Background

- Our Service and the Rationale for Benchmarking

The History of our Benchmarking

- Very Early History [1997 - 2002]
- Early History [2003 - 2009]
- Intermediate History [2010 - 2017]
- Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

- What did we learn?
- What will we do?
- What are the others doing?

Our Service and the Rationale for Benchmarking

our "community service, part I"

- about 1996 Decision Tree started (with Peter Spellucci)
- soon after Benchmarks added
- first no commercial software, later selected codes
- extensive, very frequently updated
- lead to more transparency and competition
- both open source and commercial developers use benchmarks for advertising

DECISION TREE FOR OPTIMIZATION SOFTWARE

Welcome! This site aims at helping you identify ready to use solutions for your optimization problem, or at least to find some way to bui solution using work done by others. If you know of useful sources not listed here, please let us know. If something is found to be erroneous us know, too. Where possible, public domain software is listed here.

In any case, observe the expressed or implied LICENSE conditions ! In most cases, these accompany the source code. As a rule, most cod for research. This means free for academic research and teaching or for trying whether it serves your needs. Commercial uses (either di indirect) require licensing, as a rule.

We do not aim at giving an overview over existing commercial products and recommend one of the other guides for that. We have struct information in the way you can see on the left. Clicking on the corresponding part takes you there. The contents are as follows:

- *Problem & Software:* software sorted by problem to be solved
- *Benchmarks:* collection of testresults and performance tests, made by us or others
- *Testcases:* example files ready to use with existing software, in different formats
- *Books & Tutorials:* a short list of introductory texts, some online
- *Tools:* software which helps formulating an optimization problem or simplifyin solution
- *WebSubmission:* some software can be used directly via the net thanks to implementors w their computing facilities available to you
- *Other sources:* for more information provided by others

Hans D. Mittelmann,
School of Math & Stats
Arizona State University
✉ mittelmann@asu.edu

DECISION TREE FOR OPTIMIZATION SOFTWARE

BENCHMARKS FOR OPTIMIZATION SOFTWARE

By Hans Mittelmann (mittelmann at asu.edu)

Note that on top of the benchmarks a link to logfiles is given!

NOTE ALSO THAT WE DO NOT USE PERFORMANCE PROFILES. SEE THIS PAPER

WE USE INSTEAD THE SHIFTED GEOMETRIC MEAN

COMBINATORIAL OPTIMIZATION

- *Concorde-TSP with different LP solvers (12-20-2017)*

LINEAR PROGRAMMING

- *Benchmark of Simplex LP solvers (10-17-2018)*
- *Benchmark of commercial LP solvers (10-17-2018)*
- *Parallel Barrier Solvers on Large LP/QP problems (10-17-2018)*
- *Large Network-LP Benchmark (commercial vs free) (10-17-2018)*

MIXED INTEGER LINEAR PROGRAMMING

- *MILP Benchmark - MIPLIB2010 (10-21-2018)*
- *MILP cases that are slightly pathological (10-30-2018)*

- *Feasibility Benchmark (11-1-2018) (MIPLIB2010)*
- *Infeasibility Detection for MILP Problems (11-1-2018)*

SEMIDEFINITE/SQL PROGRAMMING

- *SQL problems from the 7th DIMACS Challenge (8-8-2002)*
- *Several SDP codes on sparse and other SDP problems (10-25-2018)*
- *Infeasible SDP Benchmark (5-9-2018)*
- *Large SOCP Benchmark (10-17-2018)*
- *MISOCP Benchmark (10-17-2018)*

NONLINEAR PROGRAMMING

- *AMPL-NLP Benchmark (10-30-2018)*

MIXED INTEGER QPS AND QCPS

- *Non-commercial convex QP Benchmark (11-16-2018)*
- *Binary QPLIB Benchmark (10-30-2018)*
- *QPLIB-QCQP Benchmark (7-27-2018)*
- *Convex Discrete QPLIB Benchmark (10-30-2018)*

MIXED INTEGER NONLINEAR PROGRAMMING

- *MINLP Benchmark (6-14-2018)*

PROBLEMS WITH EQUILIBRIUM CONSTRAINTS

- *MPEC Benchmark (4-17-2018)*

Our Service and the Rationale for Benchmarking

our "community service, part II"

- after benchmarks, NEOS solvers were added
- NEOS (network-enabled optimization solver) provides large number of interactively usable optimization programs
- about 1/3 run on our computers, NEOS only gateway
- needs to be demonstrated to give impression
- additional archives developed over time: software, test problems
- both service components benefit (our) research and teaching

(https://neos-server.org/neos)

NEOS Server: State-of-the-Art Solvers for Numerical Optimization

The **NEOS Server** is a free internet-based service for solving numerical optimization problems. Hosted by the Wisconsin Institute for Discovery at the University of Wisconsin in Madison (http://www.discovery.wisc.edu), the NEOS Server provides access to more than 60 state-of-the-art solvers in more than a dozen optimization categories. Solvers hosted by the University of Wisconsin in Madison run on distributed high-performance machines enabled by the HTCondor software (http://research.cs.wisc.edu/htcondor/); remote solvers run on machines at Arizona State University (http://www.asu.edu), the University of Klagenfurt (http://www.uni-klu.ac.at/english) in Austria, and the University of Minho (http://www.uminho.pt/en) in Portugal.

The **NEOS Guide** (https://neos-guide.org) website complements the NEOS Server, showcasing optimization case studies (https://neos-guide.org/Case-Studies), presenting optimization information and resources (https://neos-guide.org/Optimization-Guide), and providing background information (https://neos-guide.org/NEOS-Server) on the NEOS Server.

NEOS Server (https://neos-server.org/neos/solvers/index.html)

- Submit a job to NEOS (https://neos-server.org/neos/solvers/index.html)
- View Job Queue and Job Results (https://neos-server.org/neos/admin.html)
- User's Guide to the NEOS Server (http://neos-guide.org/content/users-guide)
- NEOS Server FAQ (http://neos-guide.org/content/FAQ)
- NEOS Support (http://neos-guide.org/content/contact-us)

DECISION TREE FOR OPTIMIZATION SOFTWARE

WEB-SUBMISSION

Thanks to the generosity of their providers at some sites you can try optimization software directly without any need to install it yourself.

VARIOUS SOLVERS AND SUBMISSION FORMS

The following are NEOS solvers we have installed

BNBS, BPMPD, BPMPD-AMPL, Concorde, CONDOR, CSDP, DDSIP, FEASPUMP, FEASPUMP-AMPL, ICOS, MOSEK, NSIPS, PENBMI, PROXY, PENSDP, QSOPT_EX, SCIP-G, SCIP-L, SCIP-N, SCIP-AMPL, SCIPSDP, SD, SDPA, SDPLR, SDPT3, SeDuMi, SoPlex80bit

NumaWWW	interactive use of methods for numerical mathematics and optimization
TryAMPL	AMPL student solver (max 300 variables, max 300 constraints, 10 different solvers)

Our Service and the Rationale for Benchmarking

The Rationale for Benchmarking

- Optimization is ubiquitous
- Most number-crunching computing is done in optimization
- While mathematically most optimization is not hard, writing efficient and robust programs is
- Users of optimization are well advised to try not one but several programs on their problems
- Even some powerful commercial software is available for use: NEOS (everyone), source/binaries (certain groups)

Outline

Background

Our Service and the Rationale for Benchmarking

The History of our Benchmarking

Very Early History [1997 - 2002]

Early History [2003 - 2009]

Intermediate History [2010 - 2017]

Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

What did we learn?

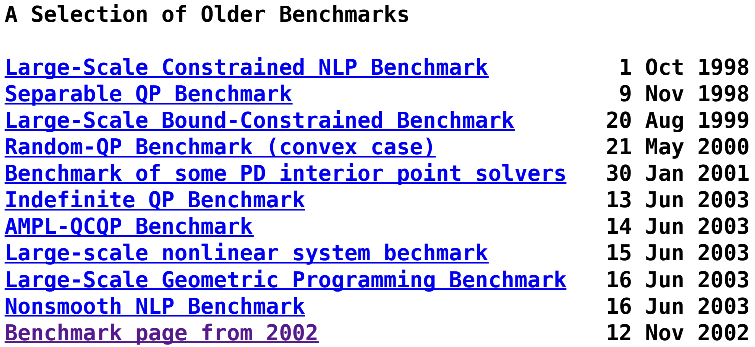
What will we do?

What are the others doing?

From the Beginning to about 2002

Some leftover files

http://plato.asu.edu/ftp/older_benchmarks/

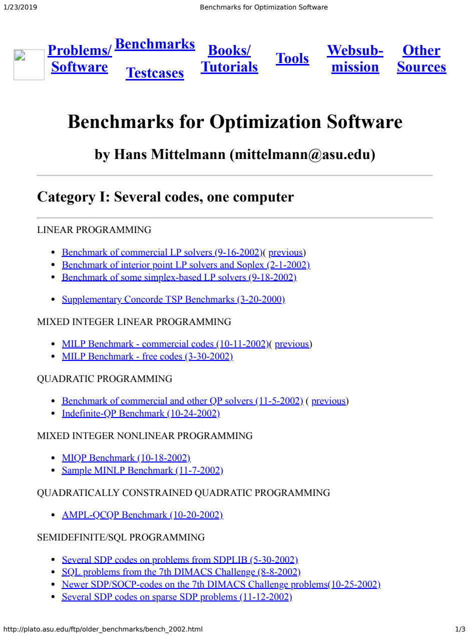


A Selection of Older Benchmarks

Large-Scale Constrained NLP Benchmark	1 Oct 1998
Separable QP Benchmark	9 Nov 1998
Large-Scale Bound-Constrained Benchmark	20 Aug 1999
Random-QP Benchmark (convex case)	21 May 2000
Benchmark of some PD interior point solvers	30 Jan 2001
Indefinite QP Benchmark	13 Jun 2003
AMPL-QCQP Benchmark	14 Jun 2003
Large-scale nonlinear system bechmark	15 Jun 2003
Large-Scale Geometric Programming Benchmark	16 Jun 2003
Nonsmooth NLP Benchmark	16 Jun 2003
Benchmark page from 2002	12 Nov 2002

Snapshot of 2002 Benchmark page

How extensive it is



1/23/2019 Benchmarks for Optimization Software

Problems/ Software | Benchmarks Testcases | Books/ Tutorials | Tools | Websub- mission | Other Sources

Benchmarks for Optimization Software

by Hans Mittelmann (mittelmann@asu.edu)

Category I: Several codes, one computer

LINEAR PROGRAMMING

- Benchmark of commercial LP solvers (9-16-2002)(previous)
- Benchmark of interior point LP solvers and Soplex (2-1-2002)
- Benchmark of some simplex-based LP solvers (9-18-2002)

- Supplementary Concorde TSP Benchmarks (3-20-2000)

MIXED INTEGER LINEAR PROGRAMMING

- MILP Benchmark - commercial codes (10-11-2002)(previous)
- MILP Benchmark - free codes (3-30-2002)

QUADRATIC PROGRAMMING

- Benchmark of commercial and other QP solvers (11-5-2002) (previous)
- Indefinite-QP Benchmark (10-24-2002)

MIXED INTEGER NONLINEAR PROGRAMMING

- MIQP Benchmark (10-18-2002)
- Sample MINLP Benchmark (11-7-2002)

QUADRATICALLY CONSTRAINED QUADRATIC PROGRAMMING

- AMPL-QCQP Benchmark (10-20-2002)

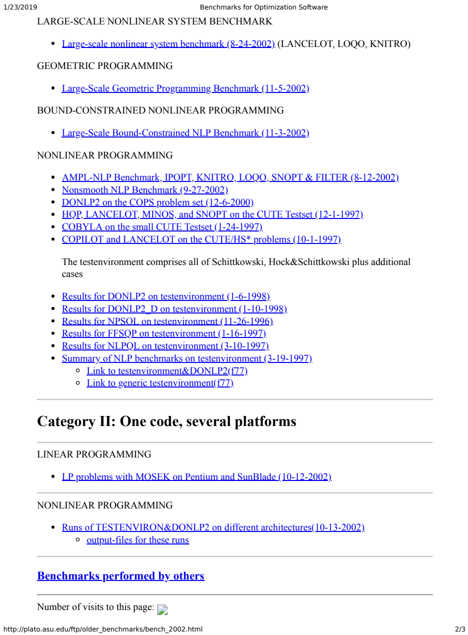
SEMIDEFINITE/SQL PROGRAMMING

- Several SDP codes on problems from SDPLIB (5-30-2002)
- SQL problems from the 7th DIMACS Challenge (8-8-2002)
- Newer SDP/SOCP-codes on the 7th DIMACS Challenge problems(10-25-2002)
- Several SDP codes on sparse SDP problems (11-12-2002)

http://plato.asu.edu/ftp/older_benchmarks/bench_2002.html 1/3

Snapshot of 2002 Benchmark page

How extensive it is



1/23/2019 Benchmarks for Optimization Software

LARGE-SCALE NONLINEAR SYSTEM BENCHMARK

- Large-scale nonlinear system benchmark (8-24-2002) (LANCELOT, LOQO, KNITRO)

GEOMETRIC PROGRAMMING

- Large-Scale Geometric Programming Benchmark (11-5-2002)

BOUND-CONSTRAINED NONLINEAR PROGRAMMING

- Large-Scale Bound-Constrained NLP Benchmark (11-3-2002)

NONLINEAR PROGRAMMING

- AMPL-NLP Benchmark, IPOPT, KNITRO, LOQO, SNOPT & FILTER (8-12-2002)
- Nonsmooth NLP Benchmark (9-27-2002)
- DONLP2 on the COPS problem set (12-6-2000)
- HQP, LANCELOT, MINOS, and SNOPT on the CUTE Testset (12-1-1997)
- COBYLA on the small CUTE Testset (1-24-1997)
- COPILOT and LANCELOT on the CUTE/HS* problems (10-1-1997)

The testenvironment comprises all of Schittkowski, Hock&Schittkowski plus additional cases

- Results for DONLP2 on testenvironment (1-6-1998)
- Results for DONLP2_D on testenvironment (1-10-1998)
- Results for NPSOL on testenvironment (11-26-1996)
- Results for FFSQP on testenvironment (1-16-1997)
- Results for NLPQL on testenvironment (3-10-1997)
- Summary of NLP benchmarks on testenvironment (3-19-1997)
    - Link to testenvironment&DONLP2(f77)
    - Link to generic testenvironment(f77)

Category II: One code, several platforms

LINEAR PROGRAMMING

- LP problems with MOSEK on Pentium and SunBlade (10-12-2002)

NONLINEAR PROGRAMMING

- Runs of TESTENVIRON&DONLP2 on different architectures(10-13-2002)
    - output-files for these runs

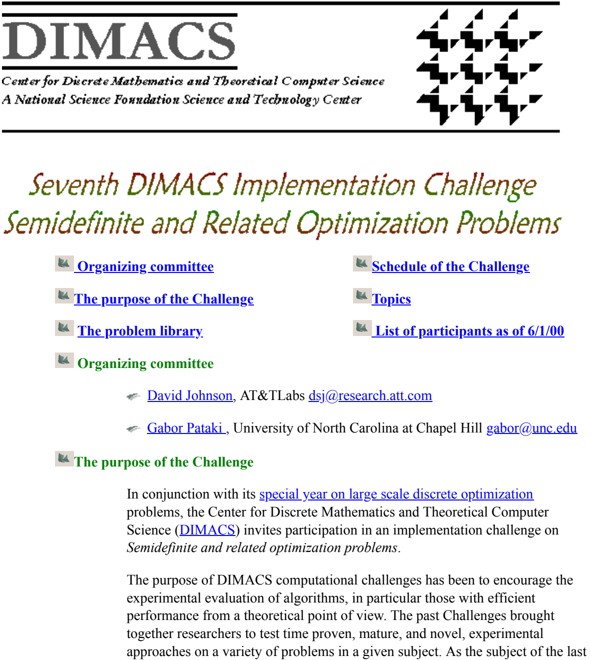
Benchmarks performed by others

Number of visits to this page:

http://plato.asu.edu/ftp/older_benchmarks/bench_2002.html 2/3

An independent benchmark from the early 2000s

First time this happened



DIMACS

Center for Discrete Mathematics and Theoretical Computer Science
A National Science Foundation Science and Technology Center

Seventh DIMACS Implementation Challenge
Semidefinite and Related Optimization Problems

- Organizing committee
- The purpose of the Challenge
- The problem library
- Schedule of the Challenge
- Topics
- List of participants as of 6/1/00

Organizing committee

- David Johnson, AT&TLabs dsj@research.att.com
- Gabor Pataki , University of North Carolina at Chapel Hill gabor@unc.edu

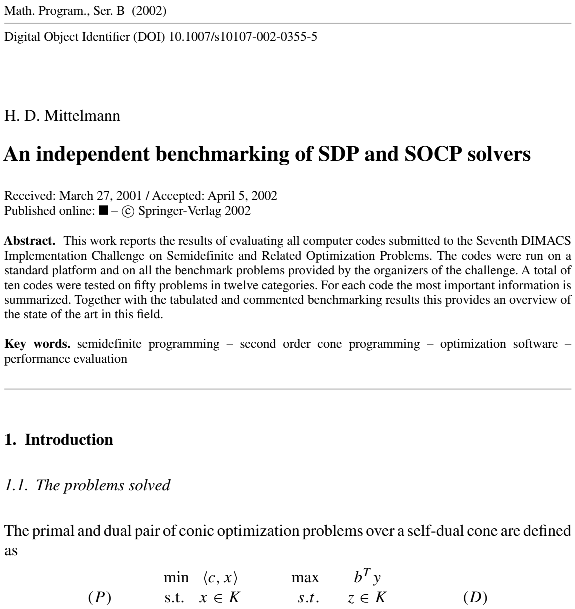
The purpose of the Challenge

In conjunction with its special year on large scale discrete optimization problems, the Center for Discrete Mathematics and Theoretical Computer Science (DIMACS) invites participation in an implementation challenge on *Semidefinite and related optimization problems*.

The purpose of DIMACS computational challenges has been to encourage the experimental evaluation of algorithms, in particular those with efficient performance from a theoretical point of view. The past Challenges brought together researchers to test time proven, mature, and novel, experimental approaches on a variety of problems in a given subject. As the subject of the last

An independent benchmark from the early 2000s

The benchmark paper



Math. Program., Ser. B (2002)

Digital Object Identifier (DOI) 10.1007/s10107-002-0355-5

H. D. Mittelmann

An independent benchmarking of SDP and SOCP solvers

Received: March 27, 2001 / Accepted: April 5, 2002
Published online: ■ – © Springer-Verlag 2002

Abstract. This work reports the results of evaluating all computer codes submitted to the Seventh DIMACS Implementation Challenge on Semidefinite and Related Optimization Problems. The codes were run on a standard platform and on all the benchmark problems provided by the organizers of the challenge. A total of ten codes were tested on fifty problems in twelve categories. For each code the most important information is summarized. Together with the tabulated and commented benchmarking results this provides an overview of the state of the art in this field.

Key words. semidefinite programming – second order cone programming – optimization software – performance evaluation

1. Introduction

1.1. The problems solved

The primal and dual pair of conic optimization problems over a self-dual cone are defined as

$$
\begin{array}{llllll}
 & \min & \langle c, x\rangle & \max & b^T y & \\
(P) & \text{s.t.} & x \in K & s.t. & z \in K & (D)
\end{array}
$$

Was our work ever supported?

Very temporarily

- We had a small NSF grant to support benchmarking from 2000 to 2003

- A follow-up proposal was not approved

- At the ISMP 2003 meeting commercial developers discussed creating a fund to support our work

- CPLEX was not in favor

- We decided to continue without support (up to the present)

What will be shown next

- Initially we had chosen all benchmark problems ourselves
- Later various libraries were created:
  MIPLIB2010/17, CBLIB14, QPLIB17
- To allow tracking of development over time we archived our benchmark talks starting in 2002. From them the history will be documented
- In view of the very latest developments mostly MILP results are presented, in particular for the "BIG THREE"
  CPLEX, Gurobi, XPRESS
- Note that historic MILP speedup is 10^{12} (one trillion)

Outline

Background

Our Service and the Rationale for Benchmarking

The History of our Benchmarking

Very Early History [1997 - 2002]

Early History [2003 - 2009]

Intermediate History [2010 - 2017]

Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

What did we learn?

What will we do?

What are the others doing?

Early History

first parallel computations, AMD

```
9 Sep 2006      ==============================
                 Parallel CPLEX on MIP problems
                 ==============================
     elapsed CPU seconds on 2.4GHz Opteron (64-bit, Linux)

========================================================
 class    problem       Opter-1       Opter-2      Opt-dual
========================================================
 MILP      bienst2        2529           608           762
               lrn         114            85           356
             mas74         897           441           483
            neos13        2073          1694          2266
             neos5        1169        >40000
          seymour1         669           449           526
--------------------------------------------------------
```

Early History

first parallel computations, AMD

```
9 Sep 2006      ==============================
                 Parallel CPLEX on MIP problems
                 ==============================
            "c":  problem convex
--------------------------------------------------------------
 MIQP     ibienst1           2742          1330          1105
            inug08   c       7973          4761         10209
            iqap10           1679           457           687
              isqp   c       4755          2824          8827
--------------------------------------------------------------
 MIQQP    ibienst1           3132          1878          2644
           imisc07           6460          3255          3445
           imod011   c       7348          9463         10014
        inug06-3rd   c       6588          6890          7833
            inug08   c       4221          2336          2768
         iran13x13           8756          3876          4278
         CLay0304M   c       1278           630          1329
--------------------------------------------------------------
```

Early History

Intel vs AMD

```
27 Oct 2007      ==============================
                 Parallel CPLEX on MIP problems
                 ==============================
Logiles at http://plato.asu.edu/ftp/ser_par_logs/

CPLEX-11.0 was run in default mode on a single and on a
2-processor 2.4GHz Opteron (64-bit, Linux), as well as on 1,2,4
processors of a 2.667GHz Intel Core 2 Quad on problems from

http://plato.asu.edu/ftp/milpf.html
http://plato.asu.edu/ftp/miqp.html

Times given are elapsed CPU times in seconds.
```

Early History

Intel vs AMD

```
27 Oct 2007      ===============================
                 Parallel CPLEX on MILP problems
                 ===============================
elapsed CPU sec on AMD Opteron resp Intel Core2 (64-bit, Linux)
"c": problem convex

===========================================================
class  problem   c  Opter-1 Opter-2 Intel-1 Intel-2 Intel-4
===========================================================
MILP    bienst2  y     203      83     154      70      34
            lrn  y     101      51      54      25      26
          mas74  y     467     365     294     131      71
         neos13  y     154     524      67      91     245
          neos5  y     251     207     185     117      40
       seymour1  y     284     204     158     114      71
-----------------------------------------------------------
```

Early History

Intel vs AMD

```
27 Oct 2007      =========================================
                 Parallel CPLEX on MIQP and MIQQP problems
                 =========================================


===========================================================
class  problem   c  Opter-1 Opter-2 Intel-1 Intel-2 Intel-4
===========================================================
MIQP   ibienst1       1612    1447    1052     466     313
         inug08  y    7954    4940    2820    1593    1844
         iqap10       1560     467     599     185     180
           isqp  y    5847    3994    1790    1043    1970
-----------------------------------------------------------
MIQQP  ibienst1        331     112     247     105      51
        imisc07         83      46      54      31      22
        imod011  y    9353  >10200    5025    3571    2916
     inug06-3rd  y    8016  >15600    4251    3230    3582
         inug08  y    4281    4021    2598    1473    1068
      iran13x13         40      50      29      13       7
===========================================================
```

Early History

more Intel vs AMD

```
11 Jun 2008        ==============================
                   Parallel CPLEX on MIP problems
                   ==============================

Logiles at http://plato.asu.edu/ftp/ser_par_logs/

CPLEX-11.1 was run in opportunistic and deterministic parallel mod
on 4 and 8 processors of a dual-quad 2.2GHz Opteron (64-bit, Linux
as well as on 1, 2, 4 processors of a 2.667GHz Intel Core 2 Quad
(64-bit, Linux) on problems from the benchmarks:

http://plato.asu.edu/ftp/milpf.html
http://plato.asu.edu/ftp/miqp.html

Times given are elapsed CPU times in seconds.
```

Early History

more Intel vs AMD

```
10 Apr 2008      ===============================
                 Parallel CPLEX on MILP problems
                 ===============================
elapsed CPU sec on AMD Opteron resp Intel Core2 (64-bit, Linux)
===============================================================
problem   Opt4o Opt4d Opt8o Opt8d Intl1 Int2o Int2d Int4o Int4d
===============================================================
bienst2      59   119    34    64   156    71    97    40    89
    lrn      41    58    39    55    38    27    44    49    39
  mas74     120   131    91   109   237   116   182    65   105
 neos13     236   290   214   127    72    98    90   126   282
  neos5      57   202    40   117   189    64   247    21   150
seymour1     91   123    67   101   166   100   114    65    84
===============================================================
"o"  opportunistic parallelism
"d"  deterministic parallelism
```

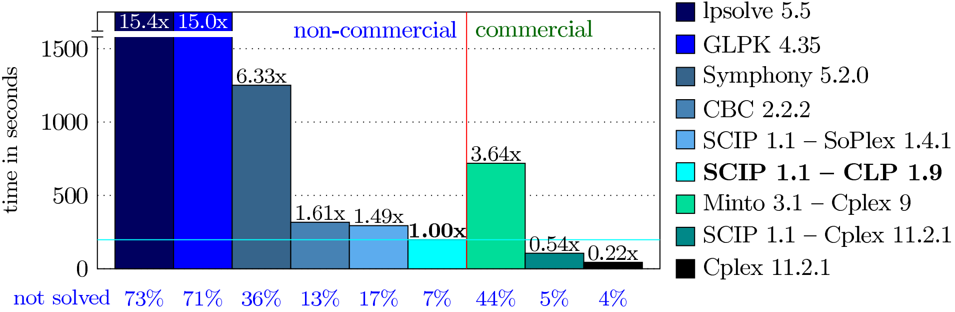
Early History

more Intel vs AMD

```
     Parallel CPLEX on MIQP and MIQCQP problems ("c" convex)

problem  c Opt4o Opt4d Opt8o Opt8d Intl1 Int2o Int2d Int4o Int4d
=================================================================
ibienst1       236   421   121   458  1174   453   584   295   298
  inug08 y  1989  1852  1483  1695  3113  1632  1570  1732  1652
  iqap10       350   480   347   543   664   191   222   179   267
    isqp y  1735  2029  1690  1954  1798  1061  1140  2193  2033
-----------------------------------------------------------------
ibienst1        49   317    23   311   253    75   480    34   241
 imisc07        40   113    42    57   110    75   236    48    76
 imod011 y  3481  7705  3736  8021  5292  3822  5300  3123  5303
inug06-3 y  4301  6284  4211  6483  4317  3280  4371  3130  4280
  inug08 y  2520  2007  1035  1559  2762  1576  1941  1068  1357
iran13x13        8    94     7   103    27    12   100     7    64
=================================================================
```

Early History

From the SCIP webpage 2/28/2009



What happened in the early history?

- Multicore computing becomes the standard
- After publishing CPLEX vs. XPRESS in a benchmark in 2007, XPRESS(Dash) asks not to be included
- In late 2008 at INFORMS Washington/DC Bixby/Gurobi presents first results after 18 months, during 9 of which code development by Gu and Rothberg
- Later Gurobi makes code available to academics; this forces CPLEX to make it available as well; we include Gurobi starting 2010
- FICO buys XPRESS. In 2010 they want to be included again

Outline

Background

Our Service and the Rationale for Benchmarking

The History of our Benchmarking

Very Early History [1997 - 2002]

Early History [2003 - 2009]

Intermediate History [2010 - 2017]

Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

What did we learn?

What will we do?

What are the others doing?

Intermediate History

Our initial selection of difficult problems

```
15 Jun 2010      ==========================================
                 MILP cases that are difficult for some codes
                 ==========================================


CPLEX-12.1    GUROBI-3.0.0    CBC-2.4.1    MOSEK-6.0.0.78
SCIP-1.2.0 (CPLEX or CLP as LP solver)
```

problem	CPLEX4	GUROBI4	SCIPC	CBC4	MOSEK	SCIPL
bc	>50000	232	7681	>40000	>40000	6564
neos-849702	209	19583	1295	1864	>40000	3004
ns1952667	147	>60000	811	>60000	>40000	503
ns2017839	66	251	112	6902	18106	58
ns2034125	>65000	3501	>65000	>65000	>40000	fail
ns2070961	>80000	>40000	18279	>40000	>40000	>40000
ns2071214	>72000	32042	f	>40000	>40000	8260
ns2081729	>60000	363	11649	>40000	>40000	14329
ns2082664	5	4	5164	>40000	1	21
ns2082847	1	1	>5000	24	>40000	1

Intermediate History

```
11 Nov 2011 Mixed Integer Linear Programming Benchmark (MIPLIB2010)

        Scaled shifted geometric means of times, 87 problems total

threads CBC  CPLEX  GLPK  GUROBI LPSOLVE SCIPC SCIPL SCIPS XPRESS
-------------------------------------------------------------------
   1     8.82  1.25  19.14    1     16.8     3.19   5.3  4.88  1.12
solved   41    73      3     78       5       61    52    56    74
-------------------------------------------------------------------

threads   CBC      CPLEX      FSCIPC     FSCIPS     GUROBI     XPRESS
-------------------------------------------------------------------
   4     10.27       1          5.78       9.41       1.06       1.23
solved    52        84          66         64         84         79
-------------------------------------------------------------------

threads   CBC      CPLEX      FSCIPC     FSCIPS     GUROBI     XPRESS
------------------------------------------------------------------
  12      11.1       1          8.08       11.9        1.07      1.21
solved    56        84          68          65        87          83
-------------------------------------------------------------------
```

What is the shifted geometric mean?

- There are huge problems in using the performance profiles for several codes in one graph
- One would need to do $N - 1$ graphs for N codes
- Commercial code developers use the shifted geometric mean
- If c_i is the compute time for instance i then one computes
- $(\prod_{i=1}^{N}[c_i + \text{shift}])^{\frac{1}{N}} - \text{shift}$
- For the shift typically 10 [secds] is used to avoid skewing from relatively very small c_i
- This provides a balanced averaging

Intermediate History

```
9 Aug 2012  Mixed Integer Linear Programming Benchmark (MIPLIB2010)

threads   CBC  CPLEX   GLPK  GUROBI  LPSOLVE  SCIPC  SCIPL  SCIPS  XPRESS
--------------------------------------------------------------------------
   1      10.1  1.26   21.6     1       18.9     3.37   5.30   5.00   1.09
solved     41    75      3     77        5        64     55     58     76
--------------------------------------------------------------------------


threads    CBC      CPLEX      FSCIPC    FSCIPS    GUROBI    XPRESS
-------------------------------------------------------------------
   4       11.6      1.13       6.03      10.2       1        1.17
solved      52        84         69        65        83        81
-------------------------------------------------------------------


threads    CBC      CPLEX      FSCIPC    FSCIPS    GUROBI    XPRESS
-------------------------------------------------------------------
  12       13.4       1.2       9.51      15.6       1        1.25
solved      55        84         71        66        87         82
-------------------------------------------------------------------
```

Intermediate History

```
31 May 2013        MILP cases that are slightly pathological

CPLEX-12.5.1pre CPLEX
GUROBI-5.5.0: GUROBI
ug[SCIP/cpx]: FSCIP-Parallel development version of SCIP
CBC-2.8.0: CBC
XPRESS-7.5.0: XPRESS
SCIP-3.0.1: serial SCIP with CPLEX

Table for 12 threads, Result files per solver, Log files per solver

Scaled shifted geometric mean of runtimes and problems solved (25 total)
----------------------------------------------------------------------
   CBC    CPLEX    FSCIP   GUROBI   SCIP   XPRESS  CPLEX-5   GUROBI-5
----------------------------------------------------------------------
   8.79      1      9.27     1.65    7.64    2.53    0.69      0.75
    10      23       14       24      15      17      25         24
----------------------------------------------------------------------
GUROBI/CPLEX-5: Best of 5 runs with random seeds 1001-1005
```

Intermediate History

```
 8 Jul 2015         ======================================
                    The EASY MIPLIB Instances (MIPLIB2010)
                    ======================================
                      H. Mittelmann (mittelmann@asu.edu)

CBC-2.9.4: CBC
CPLEX-12.6.2: CPLEX
GUROBI-6.0.0: GUROBI
XPRESS-7.9.0: XPRESS
FiberSCIP[cpx]-3.1.1: Parallel development version of SCIP

Table for all solvers, Result files per solver, Log files per solver

++++++++++++++++++++++++++++++++++++++++++++++
Shifted geometric means of times


 no. of probs     CBC     CPLEX    GUROBI     XPRESS     FSCIP
-----------------------------------------------------------------
    205           12      1.05        1       1.74        7.64
  solved          115      194     194        170         139
-----------------------------------------------------------------
```

Intermediate History

```
11 Nov 2016        ==========================================
                   The Solvable MIPLIB Instances (MIPLIB2010)
                   ==========================================

CBC-2.9.8: CBC
CPLEX-12.7.0: CPLEX
GUROBI-7.0.0: GUROBI
XPRESS-8.0.0: XPRESS
FiberSCIP[cpx]-3.2.0: Parallel development version of SCIP

 no. of probs     CBC     CPLEX    GUROBI     XPRESS     FSCIP
----------------------------------------------------------------
12 threads       1183     85.7       76        158         727
    211          15.5     1.13        1        2.07       9.56
  solved          118      201      207        178        142
----------------------------------------------------------------

 no. of probs   CPLEX    GUROBI     XPRESS
-------------------------------------------
48 threads       79.9     69.3        139
    213          1.19       1        2.07
  solved          206      210        181
-------------------------------------------
```

Intermediate History

Updated versions of codes

```
10 Sep 2017 =======================================================
            Mixed Integer Linear Programming Benchmark (MIPLIB2010)
            =======================================================
                      H. Mittelmann (mittelmann@asu.edu)

CPLEX-12.7.1: CPLEX
GUROBI-7.5.0 GUROBI
ug[SCIP/cpx/spx]-4.0.0:
Parallel development version of SCIP (SCIP+CPLEX/SOPLEX on 1 thread)
CBC-2.9.8: CBC
XPRESS-8.2.1: XPRESS
MATLAB-2017a: MATLAB (intlinprog)
MIPCL-1.4.0: MIPCL
```

Intermediate History

Gurobi clearly ahead

1 thr	CBC	CPLEX	GUROBI	SCIPC	SCIPS	XPRESS	MATLAB
unscal	1639	66.7	50.8	435	473	97	2834
scaled	32	1.31	1	8.56	9.32	1.91	56
solved	53	87	87	74	71	85	36

4 thr	CBC	CPLEX	FSCIPC	FSCIPS	GUROBI	XPRESS	MIPCL⋆
unscal	843	41.1	278	355	30	47.9	252
scaled	28.2	1.37	9.28	11.9	1	1.60	8.41
solved	66	86	74	74	87	85	79

⋆ 8 threads

12 thr	CBC	CPLEX	FSCIPC	FSCIPS	GUROBI	XPRESS	MIPCL
unscal	668	32.8	286	448	27.9	40.9	209
scaled	24	1.17	10.2	16	1	1.46	7.48
solved	69	86	73	69	87	86	79

What happened in the intermediate history?

- MIPLIB2010 was released
  - 361 instances, benchmark set 87, still unsolved 70
- We introduce the shifted geometric mean
- Gurobi surpasses CPLEX, XPRESS falls behind
- Standard benchmark set becomes too easy
- A new benchmark in 2013: SOCP and MISOCP (not shown, from CBLIB)
- A new code appears out of nowhere: MIPCL

Outline

Background

Our Service and the Rationale for Benchmarking

The History of our Benchmarking

Very Early History [1997 - 2002]

Early History [2003 - 2009]

Intermediate History [2010 - 2017]

Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

What did we learn?

What will we do?

What are the others doing?

Latest (Hi)Story

Pre INFORMS 2018

```
21 Jun 2018        ==========================================
                   The Solvable MIPLIB Instances (MIPLIB2010)
                   ==========================================
                     H. Mittelmann (mittelmann@asu.edu)

The following codes were run on the "green" problems from MIPLIB2010 with the
MIPLIB2010 scripts on an Intel Xeon X5680 (32GB, Linux, 64 bits, 2*6 cores) and
with 40 threads on an Intel Xeon Gold 6138, 40 cores, 256GB, 2.00GHz.

CBC-2.9.8, CPLEX-12.8.0, GUROBI-8.0.0, XPRESS-8.5.1, FiberSCIP[cpx]-4.0.0,
ODH-3.3.6, SAS-OR-14.3

 no. of probs     CBC     CPLEX    GUROBI     XPRESS     FSCIP        SAS
--------------------------------------------------------------------------
12 threads       1266     73.4      60.9       95.3       746         256
    220          20.8     1.20         1       1.56       12.2        4.21
  solved          119      211      213        207        140         171
--------------------------------------------------------------------------

 no. of probs   CPLEX    GUROBI     XPRESS       SAS        ODH
----------------------------------------------------------------
40 threads       54.0     44.2       64.7        197       54.9
    220          1.22        1       1.46       4.46       1.24
  solved          211      216        208        183        212
----------------------------------------------------------------

unscaled and scaled shifted geometric means of runtimes
```

In how many benchmarks are the BIG THREE?

- Pre INFORMS 2018
  - CPLEX is in 15 of 22 of our benchmarks
  - Gurobi and XPRESS are in 13 of our benchmarks (not TSP, not QCQP)
- Post INFORMS 2018
  - CPLEX, Gurobi, XPRESS are in NONE of our benchmarks
- What happened?
- This is finally the Story
  - Gurobi advertised aggressively
  - CPLEX (IBM) and XPRESS (FICO) reacted

This is what happened at INFORMS2018

The Story part I

- Over many years Gurobi had used our benchmark results for advertising making bargraphs from the tables
- At INFORMS 2018 the library MIPLIB2017 was released. We had just used it in our benchmark. It has 240 instances and only the full set is a benchmark set
- Instance selection of MIPLIB2017 uses a sophisticated computer program
- Gurobi was represented on the MIPLIB2017 committee
- At INFORMS2018 Gurobi claimed that we had used certain 99 MIPLIB2017 instances in our benchmark showing they are 2.69 times faster than CPLEX and 5.51 times faster than XPRESS

This is what happened at INFORMS2018

The Story part II

- On the last day of the conference in our session Gurobi apologized to IBM, FICO, ourselves and the community
- Tobias Achterberg and Zonghao Gu draft a paper analyzing what had happened
- After INFORMS2018 both IBM and FICO request that we remove their numbers from all benchmarks
- We decide to also omit the Gurobi numbers
- See the following slides documenting these developments

Gurobi Optimizer 8.1: The Fastest Solver in the World

2.69X

Faster than CPLEX

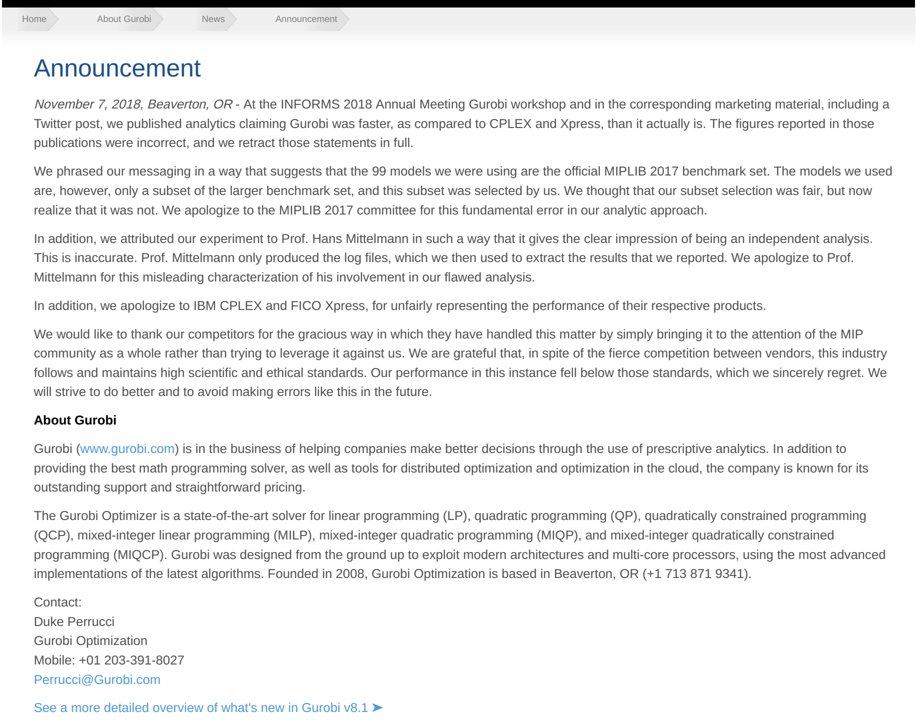
5.51X

Faster than Xpress

“Benchmarks on the 99 models in the new 2017 MIPLIB demonstrate the purest objective comparison of speed.”

Independent performance tests performed by Professor Hans Mittelmann using all new models from the recently released MIPLIB 2017 benchmark set show that Gurobi Optimizer 8.1.0 is 2.69X faster than IBM® CPLEX 12.8.0 and 5.51X faster than FICO® Xpress 8.5.1.

- ✓ The new 2017 MIPLIB is a standard test set used to compare the performance of Mixed-Integer Programming (MIP) solvers.
- ✓ These results look at performance on all 99 new models in the set.
- ✓ Considering only the newest models in the set gives the fairest, most objective speed comparison, since none of the vendors have had a chance to tune to these models.
- ✓ These numbers show geometric mean runtime ratios, calculated using the standard PAR-10 performance testing methodology.
- ✓ These results confirm Gurobi Optimizer’s position as the world’s fastest math programming solver.



Announcement

November 7, 2018, Beaverton, OR - At the INFORMS 2018 Annual Meeting Gurobi workshop and in the corresponding marketing material, including a Twitter post, we published analytics claiming Gurobi was faster, as compared to CPLEX and Xpress, than it actually is. The figures reported in those publications were incorrect, and we retract those statements in full.

We phrased our messaging in a way that suggests that the 99 models we were using are the official MIPLIB 2017 benchmark set. The models we used are, however, only a subset of the larger benchmark set, and this subset was selected by us. We thought that our subset selection was fair, but now realize that it was not. We apologize to the MIPLIB 2017 committee for this fundamental error in our analytic approach.

In addition, we attributed our experiment to Prof. Hans Mittelmann in such a way that it gives the clear impression of being an independent analysis. This is inaccurate. Prof. Mittelmann only produced the log files, which we then used to extract the results that we reported. We apologize to Prof. Mittelmann for this misleading characterization of his involvement in our flawed analysis.

In addition, we apologize to IBM CPLEX and FICO Xpress, for unfairly representing the performance of their respective products.

We would like to thank our competitors for the gracious way in which they have handled this matter by simply bringing it to the attention of the MIP community as a whole rather than trying to leverage it against us. We are grateful that, in spite of the fierce competition between vendors, this industry follows and maintains high scientific and ethical standards. Our performance in this instance fell below those standards, which we sincerely regret. We will strive to do better and to avoid making errors like this in the future.

About Gurobi

Gurobi (www.gurobi.com) is in the business of helping companies make better decisions through the use of prescriptive analytics. In addition to providing the best math programming solver, as well as tools for distributed optimization and optimization in the cloud, the company is known for its outstanding support and straightforward pricing.

The Gurobi Optimizer is a state-of-the-art solver for linear programming (LP), quadratic programming (QP), quadratically constrained programming (QCP), mixed-integer linear programming (MILP), mixed-integer quadratic programming (MIQP), and mixed-integer quadratically constrained programming (MIQCP). Gurobi was designed from the ground up to exploit modern architectures and multi-core processors, using the most advanced implementations of the latest algorithms. Founded in 2008, Gurobi Optimization is based in Beaverton, OR (+1 713 871 9341).

Contact:
Duke Perrucci
Gurobi Optimization
Mobile: +01 203-391-8027
Perrucci@Gurobi.com

See a more detailed overview of what's new in Gurobi v8.1 ➤

Good Benchmarking Practices – And What Happens If They Are Ignored

Tobias Achterberg[*], Zonghao Gu[†], and Michael Winkler[‡]

Gurobi Optimization

13 December 2018

Abstract

Conducting computational experiments to evaluate the performance of solvers for an optimization problem is a very challenging task. In this paper, we outline good practices regarding test set selection and benchmarking methodology. Moreover, we present a concrete example in our context of mixed integer linear programming solvers, where failure to adhere to these guidelines results in wrong conclusions.

1 Introduction

Gurobi is one of today's fastest solvers for mixed integer linear programming. In the development of such a software, one of the key aspects is to be able to assess whether a new component or a change to some existing algorithm improves the overall performance of the solver. Moreover, for competitive reasons, it is interesting to know how the performance of ones own solver compares against the competition. Such questions are usually answered by conducting benchmark runs on a set of test problems. Then, the running times of the different solvers or solver versions are compared in order to draw qualitative and quantitative conclusions about their performance. It is, however, not easy to perform this evaluation in a reasonable way. If done wrong, the conclusions drawn from the

MIPLIB 2017: a Data-Driven Compilation of the 6th Mixed Integer Programming Library

Ambros Gleixner Gregor Hendel Gerald Gamrath
Tobias Achterberg Michael Bastubbe Timo Berthold
Philipp Christophel Kati Jarck Thorsten Koch
Jeff Linderoth Marco Lübbecke Hans Mittelmann
Ted Ralphs Domenico Salvagnin Yuji Shinano

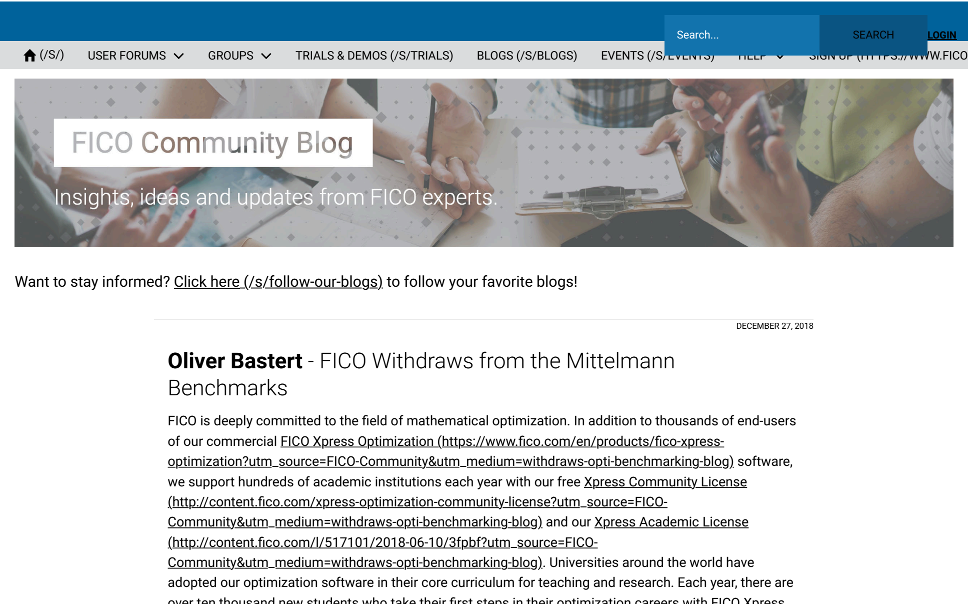
March 4, 2019

List of symbols

D	Total dissimilarity
$\mathcal{E}$	Set of excluded instances
ε	Feasibility tolerance
F	Feature matrix
$\mathcal{F}$	Instance clustering
$\mathcal{G}$	Set of model groups
$\mathcal{I}$	Set of instances
$\mathcal{I}$	Set of submitters
$\mathcal{P}$	Performance clustering
Q	Dimension of static feature space
R	Cluster count
r	Ranking
$\mathcal{S}$	Set of solvers
σ	shift value in geometric mean computation
T	The time limit
t	running time in seconds
$t^{\mathbf{rel}}$	performance matrix
ω	weight (objective coefficient) of each instance

Latest (Hi)Story

After INFORMS 2018



Want to stay informed? Click here (/s/follow-our-blogs) to follow your favorite blogs!

DECEMBER 27, 2018

Oliver Bastert - FICO Withdraws from the Mittelmann Benchmarks

FICO is deeply committed to the field of mathematical optimization. In addition to thousands of end-users of our commercial FICO Xpress Optimization (https://www.fico.com/en/products/fico-xpress-optimization?utm_source=FICO-Community&utm_medium=withdraws-opti-benchmarking-blog) software, we support hundreds of academic institutions each year with our free Xpress Community License (http://content.fico.com/xpress-optimization-community-license?utm_source=FICO-Community&utm_medium=withdraws-opti-benchmarking-blog) and our Xpress Academic License (http://content.fico.com/l/517101/2018-06-10/3fpbf?utm_source=FICO-Community&utm_medium=withdraws-opti-benchmarking-blog). Universities around the world have adopted our optimization software in their core curriculum for teaching and research. Each year, there are over ten thousand new students who take their first steps in their optimization careers with FICO Xpress

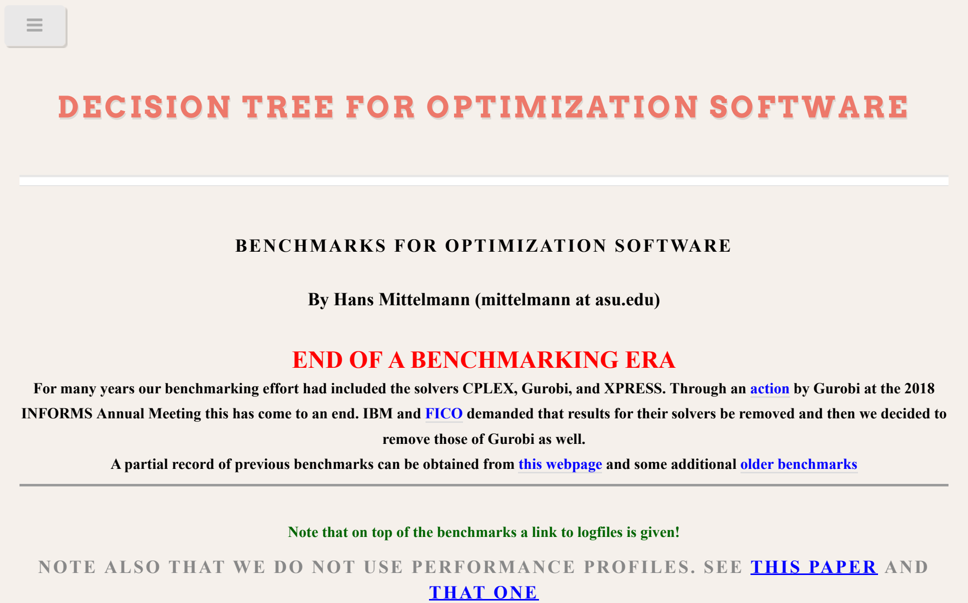
Latest (Hi)Story

At INFORMS 2018

```
 1 Nov 2018 =======================================================
           Mixed Integer Linear Programming Benchmark (MIPLIB2017)
           =======================================================
                     H. Mittelmann (mittelmann@asu.edu)

The following codes were run on the benchmark instances of the forthcoming
MIPLIB2017 on an Intel Xeon X5680 (32GB, Linux, 64 bits, 2*6 cores) and with
48 threads on an Intel Xeon E5-4657L, 48 cores, 512GB, 2.40GHz
(available memory 256GB). 2/1 hours max. More codes to be added later.

               CPLEX-12.8.0, GUROBI-8.1.0, XPRESS-8.5.1


 no. of probs     CPLEX    GUROBI     XPRESS
----------------------------------------------
12 threads         307       207        416
    240           1.48         1        2.01
  solved           195       212        180
-----------------------------------------------


 no. of probs   CPLEX    GUROBI     XPRESS
--------------------------------------------
48 threads        238       176        336
    240          1.35         1       1.90
  solved          199       211        180
--------------------------------------------

unscaled and scaled shifted geometric means of runtimes
```

Latest (Hi)Story

After INFORMS 2018



DECISION TREE FOR OPTIMIZATION SOFTWARE

BENCHMARKS FOR OPTIMIZATION SOFTWARE

By Hans Mittelmann (mittelmann at asu.edu)

END OF A BENCHMARKING ERA

For many years our benchmarking effort had included the solvers CPLEX, Gurobi, and XPRESS. Through an action by Gurobi at the 2018 INFORMS Annual Meeting this has come to an end. IBM and FICO demanded that results for their solvers be removed and then we decided to remove those of Gurobi as well.

A partial record of previous benchmarks can be obtained from this webpage and some additional older benchmarks

Note that on top of the benchmarks a link to logfiles is given!

NOTE ALSO THAT WE DO NOT USE PERFORMANCE PROFILES. SEE THIS PAPER AND THAT ONE

Outline

Background

Our Service and the Rationale for Benchmarking

The History of our Benchmarking

Very Early History [1997 - 2002]

Early History [2003 - 2009]

Intermediate History [2010 - 2017]

Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

What did we learn?

What will we do?

What are the others doing?

What did we learn?

- Optimization Software is a cutthroat business
- IBM and FICO claim that Gurobi had their licenses for years while refusing to grant them a license for Gurobi
- Sometimes even very smart people overstep the mark
- Now users have to benchmark themselves again
- Our benchmarks are less exciting but to make up a bit for the loss we list ballpark geomeans for best commercial codes

Outline

Background

Our Service and the Rationale for Benchmarking

The History of our Benchmarking

Very Early History [1997 - 2002]

Early History [2003 - 2009]

Intermediate History [2010 - 2017]

Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

What did we learn?

What will we do?

What are the others doing?

What will we do?

- We will travel to China for three conferences
- We will benchmark noncommercial and some commercial codes
- We will have less stress before INFORMS meetings
- We will observe the development and provide free advertising for "free" codes

Outline

Background

Our Service and the Rationale for Benchmarking

The History of our Benchmarking

Very Early History [1997 - 2002]

Early History [2003 - 2009]

Intermediate History [2010 - 2017]

Latest (Hi)Story [2018 - 2019]

The Situation Now and in the Future

What did we learn?

What will we do?

What are the others doing?

What are the others doing?

They are advertising they best they can

- Gurobi: The Fastest Mathematical Programming Solver

- CPLEX: The Most Robust and Reliable Solver

- XPRESS: Fast and Reliable ... Solvers and Optimization Technologies

THE END

Thank you for your attention

Questions or Remarks?

slides of talk at:
http://plato.asu.edu/talks/china2019.pdf

our benchmarks at:
http://plato.asu.edu/bench.html

decision tree guide at:
http://plato.asu.edu/guide.html