Benchmarking Optimization Software a (Hi)Story

Hans D Mittelmann

School of Mathematical and Statistical Sciences
Arizona State University

EURO 2019
Dublin - Ireland
25 June 2019

Outline

Outline

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

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

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

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)

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

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

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	Int1	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

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

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
- $$\left(\prod_{i=1}^{N} [c_i + \text{shift}]\right)^{\frac{1}{N}} - \text{shift}$$
- 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

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 from me to 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

What did we learn?

- Optimization Software is a cutthroat business
- IBM claims that Gurobi had their license for years while refusing to grant them a license for Gurobi
- Gurobi has similar accusations against the others
- 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

What are the BIG THREE 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/euro2019.pdf

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

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