

The State of the Art in Software for SDP&SOCP and Applications

*INFORMS Annual Meeting
Denver, CO
26 October 2004*

H. D. Mittelmann
Dept of Math and Stats
Arizona State University

NOTE: some of the results have been corrected/updated after the conference. Check the benchmark page for latest results.

Services we provide

- Guide to Software: "**Decision Tree**"
- <http://plato.asu.edu/guide.html>
- Software Archive
- Software Evaluation: "**Benchmarks**"
- Archive of Testproblems
- Web-based Solvers (**1/3 of NEOS**)

From the NEOS solvers webpage

- * Semidefinite & Second Order Cone Programming
- * CSDP [Matlab Binary Input] [Sparse SDPA Input]
- * DSDP [Sparse SDPA Input]
- * PENBMI [Matlab Binary Input] [Matlab input]
- * PENSDP [Matlab Binary Input] [Sparse SDPA Input]
- * SDPA [Matlab Binary Input] [Sparse SDPA Input]
- * SDPLR [Matlab Binary Input] [Sparse SDPA Input] [SDPLR input]
- * SDPT3 [Matlab Binary Input] [Sparse SDPA Input]
- * SeDuMi [Matlab Binary Input] [Sparse SDPA Input]

Overview of Talk

- Current Benchmarks
- SDP/SOCP Benchmarks
 - SDPLIB Benchmark
 - DIMACS Update (SDP&SOCP)
 - SOCP Benchmark (FIR)
 - Large Sparse SDP Benchmark
- Conclusion, Future Work

From my benchmark page

<http://plato.asu.edu/bench.html>

SEMIDEFINITE/SQL PROGRAMMING

Several SDP codes on problems from SDPLIB (10-14-2004)

SQL problems from the 7th DIMACS Challenge (8-8-2002)

Newer SDP/SOCP-codes on the 7th DIMACS Challenge problems (10-14-2004)

Several SDP codes on sparse and other SDP problems (10-14-2004)

SOCP (second-order cone programming) Benchmark (9-17-2004)

The only other ones updated: AMPL-NLP, LP

The following codes were used to solve problems from SDPLIB.

The logfiles are at: ftp://plato.asu.edu/pub/sdplib_logs/

CSDP-4.9: <http://www.nmt.edu/~borchers/csdp.html>

SDPA-6.00: <http://grid.r.dendai.ac.jp/sdpa/index.html>

SDPT3-3.1: <http://www.math.nus.edu.sg/~mattohkc/>

SeDuMi-1.05: <http://fewcal.kub.nl/sturm/software/sedumi.html>

DSDP-5.1: <http://www-unix.mcs.anl.gov/~benson/>

PENNON-2.01 <http://www2.am.uni-erlangen.de/~kocvara/pennon/>

SBmethod-1.1.2 <http://www-user.tu-chemnitz.de/~helmberg/SBmethod/>

SDPLR-1.01: <http://dollar.biz.uiowa.edu/~burer/software/SDPLR/>

SDPLIB-1.2: <http://www.nmt.edu/~borchers/sdplib.html>

The codes were run in default mode except for increased iteration limits on a 3.2 GHz P4 with 4GB RDRAM under Linux. Given are total CPU seconds.

Error measures are given (exc for SBm) as defined in the 7th DIMACS Challenge benchmark paper Math Prog 95,407-430(2003), <http://plato.asu.edu/dimacs/node3.html>,

problem	CSDP	DSDP	SDPA	SDPT3	SeDuMi	PENNON	SBmeth	SDPLR
arch8	3	4	3	6	20	4		>1100
control7	33	32	40	91	42	62		>3000
control10	151	189	199	124	241	500		>3000
control11	242	311	316	197	389	740		>3000
equalG11	80	444	63	126	1042	66		13!
equalG51	207	964	127	249	1847	235		64!
gpp250-4	6	2	2	9	68	3		17!
gpp500-4	34	19	18	41	152	16		8!
maxG11	42	13	88	40	586	18	13	9
maxG32	391	136	1378	483	9405	185	45	40
maxG51	105	35	83	96	1337	127	8	19
mcp500-1	11	4	13	14	162	5		1

"!": SDPLR format input

problem	CSDP	DSDP	SDPA	SDPT3	SeDuMi	PENNON	SBmeth	SDPLR
qap9	2	6	2	5	13	7		175!
qap10	4	13	5	10	32	11		385!
qpG11	314	30	332	214	4562	55		17
qpG51	459	163	693	427	12113	218		4
ss30	12	19	33	12	63	16		12
theta3	4	13	5	8	27	11		49!
theta4	18	53	22	28	128	49		207!
theta5	57	165	68	88	474	165		358!
theta6	152	468	180	230	1440	499		554!
thetaG11	100	269	129	131	941	229	1	10!
thetaG51	1614	9483	1455	2809	8087	5183	13975	3890
truss7	1	1	1	1	1	1		>1000
truss8	3	5	17	1	5	10		3205

=====
Newer SDP/SOCP-codes on the 7th DIMACS Challenge problems
=====

14 Oct 2004

Logfiles for these runs at: ftp://plato.asu.edu/pub/dimacs_logs/

CSDP-4.9 (SDPA): www.nmt.edu/~borchers/csdp.html

MOSEK-3.1.1.30 (MPS): www.mosek.com

PENNON-2.01 (SDPA): www2.am.uni-erlangen.de/~kocvara/pennon/

SDPT3-3.1 (SeDuMi): www.math.nus.edu.sg/~mattohkc/

SeDuMi-1.05R5 (SeDuMi): fewcal.kub.nl/sturm/software/sedumi.html

DSDP-5.1 (SDPA): www-unix.mcs.anl.gov/~benson/

SDPA-6.0 (SDPA): grid.r.dendai.ac.jp/sdpa/index.html

SDPLR-1.01 (SDPA/LR): dollar.biz.uiowa.edu/~burer/software/SDPLR/

SBmethod-1.1.2 (graph): www-user.tu-chemnitz.de/~helmborg/SBmethod/

LOQO-6.06 (AMPL): orfe.princeton.edu/~loqo/

SDP prob	PENNON	SeDuMi	SDPT3	CSDP	DSDP	SDPA	SDPLR	SBmth
bm1	87	2734	165	459	482	73	32!	10
biomed	m	m	m	m	m	m	23496!	2304
copo14	19	140	8	5	10	42	463	na
copo23	1201	22370	482	445	582	5168	24021	na
filter48	216	5881	135	110	929	274	6435	na
filtinf1	131	6357	236	71	110	324	6244	na
ham_9_8	100	225	47	51	93	23	18	1
ham_8_3_4	15476	m	m	5309	1308	5896	6	1
ham_9_5_6	m	m	m	m	m	m	24	1
ham_10_2	m	m	m	m	m	m	216	35
ham_11_2	m	m	m	m	m	m	1529	77
torusg3-15	3694	54074	2249	2111	952	8849	279	78
torusp*15*	2493	47945	2257	1949	902	2839	192	76

SOCP problem	MOSEK	SDPT3	SeDuMi	LOQO
nb	3	7	11	8
nb_L1	3	12	9	6
nb_L2	8	11	19	12
nb_L2_bessel	2	8	12	5
nql60	7	17	7	118
nql180	146	222	101	m
qssp60	4	21	11	177
qssp180	76	391	390	m
sched_100_100_orig	3	18	21	21
sched_200_100_orig	9	70	42	321
sched_100_100_scaled	3	19	23	23
sched_200_100_scaled	9	54	73	324

=====
SOCP (second-order cone programming) Benchmark
=====

17 Sep 2004

SDPT3-3.1: <http://www.math.nus.edu.sg/~mattohkc/>
SeDuMi-1.05R5: <http://fewcal.kub.nl/sturm/software/sedumi.html>
LOQO-6.06 <http://orfe.princeton.edu/~loqo/>
MOSEK-3.1.1.30 <http://www.mosek.com/>

Input: SDPT3&SeDuMi: SeDuMi-format, MOSEK: ext MPS, LOQO: AMPL

These codes were used to solve SOCP problems generated with the Matlab FIR toolbox (<http://www.csee.umbc.edu/~dschol2/opt.html>)

We had solicited this testsuite. It is included in the toolbox.

Problem Statistics (as reported by SeDuMi)

```

=====
problem          variables    constraints    nonzeros      blocks
=====
firL1            15805        8302           6321201       7903
firL1Linfalph   31609        8303           12642402      15805
firL1Linfeps    15450        11785          1548465       7725
firL2L1alph     7848         11968          1592132       3924
firL2L1eps      15807        8303           6401802       7904
firL2Linfalph   11857        402            4821802       5929
firL2Linfeps    8857         9155           2673609       4429
firL2a           3            3001           4504501        2
firLinf         29857        1001           19903001      9953
dsNRL           10509        406            4309501       5255
wbNRL           17196        460            8061190        10
=====

```

```

=====
problem          K      MOSEK      SeDuMi*      SDPT3*      LOQO
=====
firL1            400        98          664          1474         716
firL1Linfalph   400        296         2059         1486        fail
firL1Linfeps    198         18          409           m          828
firL2L1alph     201         6           103           116         413
firL2L1eps      400         96          627          1606        1004
firL2Linfalph   400        111          752           245        fail
firL2Linfeps    300         35          492           311         386
firL2a          3000        179          247           614         434
firLinf         1000        237         5506           348           m
dsNRL           385         808          975           340
wbNRL           304         719          700           885
=====

```

"m": memory exceeded, "*" : DIMACS error measures

=====
Several SDP-codes on sparse and other SDP problems
=====

- CSDP-4.9: <http://www.nmt.edu/~borchers/csdp.html>
- PENSDP-1.2: <http://www2.am.uni-erlangen.de/~kocvara/pennon/>
- SDPT3-3.1: <http://www.math.nus.edu.sg/~mattohkc/sdpt3.html>
- SeDuMi-1.05R5: <http://fewcal.kub.nl/sturm/software/sedumi.html>
- DSDP-5.1: <http://www-unix.mcs.anl.gov/~benson/dsdp/>
- SDPA-6.0: <http://grid.r.dendai.ac.jp/sdpa/index.html>
- SDPLR-1.0: <http://dollar.biz.uiowa.edu/~burer/software/SDPLR/>

These codes were used to solve SDP problems from

- <ftp://plato.asu.edu/pub/sdp/..qap/> (see also README)
- <http://www2.am.uni-erlangen.de/~kocvara/pennon/problems.html>
- <http://www.cims.nyu.edu/~mituhiro/software.html>

In addition to taking problems from existing collections we have used the following application programs to generate instances:

GLOPTIPOLY-2.2f: www.laas.fr/~henrion/software/gloptipoly/

SOSTOOLS-2.00: control.ee.ethz.ch/~parrilo/sostools/index.html

SeDuMiInt-1.04: www.laas.fr/~peaucell/SeDuMiInt.html

SDE: www.seas.upenn.edu/~kilianw/sde/

YALMIP: control.ee.ethz.ch/~joloef/yalmip.msql

We have solicited problems from the Computer Vision and Pattern Recognition Group, University of Mannheim/Germany

=====
Several SDP-codes on sparse and other SDP problems
=====

14 Oct 2004

Total: 62 problems

Graph relaxations:	5
Kocvara collection:	16
Fukuda collection:	9
Pattern recognition:	6
GLOPTIPOLY:	11
SOSTOOLS:	8
SDE:	3
misc:	5

Kocvara collection, dimensions

```

=====
problem      variables    constraints    nonzeros      blocks
=====
buck-4        1,200        2,546          17,509         4
buck-5        3,280        6,802          48,385         4
mater-3       1,439        3,588          45,189        329
mater-4       4,807       12,498         157,779       1139
mater-5      10,143       26,820         338,757       2441
mater-6      20,463       56,311         690,149       4969
shmup4         800         4,962          36,477         4
shmup5        1,800       11,042          82,317         4
trto-4        1,200        1,874           8,734         3
trto-5        3,280        5,042          24,158         3
vibra-3         544         1,186           7,831         4
vibra-4        1,200        2,546          17,509         4
vibra-5        3,280        6,802          48,385         4

```

Kocvara collection, times ("m": memory exceeded)

```

=====
problem    PENSDP    SeDuMi    SDPT3     CSDP     DSDP     SDPA     SDPLR
=====
buck-4      174     14544     168      536     927      248     3411
buck-5     2445         m     2165a    5005    fail     4960    57018
mater-3       5       17       26      12       8     1044 >26000
mater-4      23       50      131     290      46    138452
mater-5      60      137     294    3647     176         m
mater-6     178     281     929         m     731         m
shmup4     1655    109670    1681    3774    2752     1988     1866
shmup5    10994         m    15147   52295   23498         m    61794
trto-4       83    11260      86     238     111     125    13168
trto-5     1149         m    1090    2671    2311     1963
vibra-4     148    16199     191     774    2968     269    11206
vibra-5     2107         3490    5984    fail     4740 >153000
=====

```

Fukuda collection, dimensions

```

=====
problem      variables  constraints  nonzeros    blocks
=====
BH2.r14      2,167      1,743        142,919     23
CH2_1.r14    2,167      1,743        142,919     23
CH2_3.r14    2,167      1,743        142,919     23
H2O+.r14     2,167      1,743        142,919     23
H2O_.r14     2,167      1,743        142,919     23
NH2.r14      2,167      1,743        142,919     23
H30_.r16     3,163      2,964        279,048     23
NH3.r16      3,163      2,964        279,048     23
AlH.r20      5,991      7,230        855,800     23
=====

```

Fukuda collection, times ("a": reduced accuracy)

```

=====
problem    PENSDP    SeDuMi    SDPT3    CSDP    DSDP    SDPA    SDPLR
=====
BH2.r14    3612      2010      809a    1995    12935   1495    2151
C._1.r14   3309      1566      1719    1861    12705   1393    2117
C._3.r14   3764      1487      805a    1958    11868   1592     715
H20+.r14   4177      1378      1764    1807    13296   1591    1976
H20_.r14   4127      1149      1655    1599    11221   1492     863
NH2.r14    4033      1470      1314    1747    13534   1496    1326
H30_.r16   23532     4477     14687   16087   60947   15189   2567
NH3.r16    26634     4481     9423a   16917   57817   15190   3738
AlH.r20           57059                                42296
=====

```

Pattern recognition and SDE, dimensions

problem	variables	constraints	nonzeros	blocks
cancer	570	10,470	10,569	2
checker	3,971	3,971	3,970	2
foot	2,209	2,209	2,440,944	2
hand	1,297	1,297	841,752	2
ice_2.0	8,114	8,113	8,113	2
p_auss2	9,116	9,115	9,115	2

inc_600	3,115	2,515	190,356	2
inc_1200	6,375	5,175	741,296	2
swissroll	801	3,380	330,537	2

Pattern recognition and SDE, times

problem	PENSDP	SeDuMi	SDPT3	CSDP	DSDP	SDPA	SDPLR
cancer	19349	m	m	2539	3068	1686	>60000
checker	1529	m	4908	9915	1726	m	231
foot	1706	m	3618	4019	44197	2156	>60000
hand	278	fail	555	452	5300	253	2270
ice_2.0	9060	m	m	m	16547	m	1152
p_auss2	8635	m	m	m	25865	m	1211
inc_600	1042	1424	377a	549	429	604	62504
inc_1200	10840	12107	fail	2937	3675	fail	
swissroll		3646a	1304a	859a	1271a	fail	

GLOPTIPOLY, dimensions

problem	variables	constraints	nonzeros	blocks
biggs	703	1,819	14,431	49
neu1g	253	3,002	31,877	2
neu2g	253	3,002	31,877	2
neu2c	1,256	3,002	158,098	15
neu3g	463	8,007	106,952	2
rose13	106	2,379	5,564	2
taha1a	1,681	3,002	177,420	15
taha1b	1,610	8,007	107,373	25
butcher	22,843	6,434	206,992	2
rabmo	6,827	5,004	60,287	2
reimer5	102,607	6,187	719,806	2

GLOPTIPOLY, times

problem	PENSDP	SeDuMi	SDPT3	CSDP	DSDP	SDPA	SDPLR
biggs	828	193	122	87	fail	fail	>30000
neu1g	712	781	517	835	fail	279	255
neu2g	1981	946	601	516	fail	fail	14976
neu2c	2722	1415	fail	2531	1964	1082	>11700
neu3g	44451	12572	933	10366	>70000	7070	580
rose13	140	420	143	104	778	72	141
taha1a	2810	838	fail	1440	fail	fail	1719
taha1b	9520	13646	m	3297	13617	10886	10887
butcher	9369	7769	2137	4032	25186a	23864a	>40000
rabmo	2419	2603	580	615	6105	7199	2735
reimer5	m	4680	3762	14436	26808	m	55148

SOSTOOLS and SDE, dimensions

problem	variables	constraints	nonzeros	blocks
cnhil8	121	1,716	7,260	2
cnhil10	221	5,005	24,310	2
cphil10	221	5,005	24,310	2
cphil12	364	12,376	66,429	2
neu1	255	3,003	31,880	2
neu2	255	3,003	31,880	2
neu3	421	7,364	87,573	3
rose15	138	3,860	9,182	2

inc_600	3,115	2,515	190,356	2
inc_1200	6,375	5,175	741,296	2
swissroll	801	3.380	330,537	2

SOSTOOLS and SDE, times

problem	PENSDP	SeDuMi	SDPT3	CSDP	DSDP	SDPA	SDPLR
cnhil8	876	103	11	22	28	31	73
cnhil10	fail	2323	113	398	635	505	161
cphil10	660	725	37	93	469	337	25
cphil12	7468	m	m	1078	10596	3825	71
neu1	1683	795	fail	581	fail	341	242
neu2	fail	1024	fail	852	fail	fail	14937
neu3	fail	8993	fail	5807	>70000	13938	314
rose15	fail	1800	fail	634	2708	293	473
inc_600	1042	1424	377a	549	429	604	62504
inc_1200	10840	12107	fail	2937	3675	fail	
swissroll		3646a	1304a	859a	1271a	fail	

Statistics

item	PENSDP	SeDuMi	SDPT3	CSDP	DSDP	SDPA	SDPLR
fastest	10	4	8	7	2	9	21
second	10	9	13	11	8	6	4
slowest	5	14	0	2	13	1	21
fails	5	2	7	0	8	6	0
red acc	0	1	6	1	2	1	0
mem exc	1*	11	7	3*	0	9\$	0
parallel	n	n	n	y	y	y	n

* none in 64-bit version

\$ none in parallel version

Conclusions

The state of the art of SDP&SOCP solvers for large (but not huge) problems is **quite** satisfactory

Strong SQL solvers: SDPT3, SeDuMi

Strong SDP solvers:

general: CSDP, PENNON, SDPA

specialty: SDPLR, DSDP

Strong SOCP solvers: MOSEK, LOQO

To do: evaluate parallel versions, Matlab 7.0.1, 64-bit Matlab