Experiments with linear and nonlinear optimization using Quad precision

Ding Ma and Michael Saunders

Stanford University

INFORMS Annual Meeting 2014
San Francisco, CA

Abstract

Systems biologists are developing increasingly large models of metabolism and integrated models of metabolism and macromolecular expression. Standard LP solvers do not give sufficiently accurate solutions, and exact simplex solvers are extremely slow. On a range of multiscale examples we find that 34-digit Quad floating-point achieves exceptionally small primal and dual infeasibilities (of order 10^{-30}) when no more than 10^{-15} is requested.

Partially supported by the
National Institute of General Medical Sciences
of the National Institutes of Health (NIH)
Award U01GM102098

1. Motivation
2. System and Methods
3. Algorithm and Implementation
4. 62 LPnetlib test problems
5. Conclusions

Motivation

In the Constraint Based Reconstruction and Analysis (COBRA), a biochemical network, which is inherently multiscale, is represented by a stoichiometric matrix S with m rows corresponding to metabolites (chemicals) and n columns representing reactions. Mathematically, S is part of the ordinary differential equation that governs the time-evolution of concentrations in the network:

$$\frac{d}{dt}x(t) = Sv(t), \tag{1}$$

where $x(t) \in \mathbf{R}^m$ is a vector of time-dependent concentrations and $v(t) \in \mathbf{R}^n$ is a vector of reaction fluxes. With the objective of maximizing the growth rate at the steady state, the following LP is constructed:

$$\begin{aligned} \max_{v} \quad & c^T v && \text{(2a)} \\ \text{s.t.} \quad & Sv = 0, && \text{(2b)} \\ & l \le v \le u, && \text{(2c)} \end{aligned}$$

where growth is defined as the biosynthetic requirements of experimentally determined biomass composition, and biomass generation is a set of reaction fluxes linked in the appropriate ratios.

FBA with coupling constraints

FBA has been used by Ines2012ME for the first integrated stoichiometric multiscale model of metabolism and macromolecular synthesis for *Escherichia coli* K12 MG1655. The model modifies (2) by adding constraints that couple enzyme synthesis and catalysis reactions to (2b). Coupling constraints of the form

$$c_{\min} \le \frac{v_i}{v_j} \le c_{\max} \tag{3}$$

become linear constraints

$$c_{\min} v_j \le v_i, \quad v_i \le c_{\max} v_j \tag{4}$$

for various pairs of fluxes v_i, v_j. They are linear approximations of nonlinear constraints and make S in (2b) even less well-scaled because of large variations in reaction rates. Quad precision is evidently more appealing in this case.

Coupling constraints

For example, two fluxes could be related by

$$0.0001 \le \frac{v_1}{v_2} \le 10000. \tag{5}$$

As before, we can decompose these constraints into sequences of constraints involving auxiliary variables with reasonable coefficients. If the second inequality in (5) were presented to our implementation as $v_1 \le 10000 v_2$, we would transform it to two constraints involving an auxiliary variable s_1:

$$v_1 \le 100 s_1, \qquad s_1 \le 100 v_2. \tag{6}$$

If the first inequality in (5) were presented as $v_1 \ge 0.0001 v_2$, we would leave it alone, but the equivalent inequality $10000 v_1 \ge v_2$ would be transformed to

$$v_2 \le 100 s_2, \qquad s_2 \le 100 v_1.$$

“Carrying somewhat more precision in the arithmetic than twice the precision carried in the data and available for the result will vastly reduce embarrassment due to roundoff-induced anomalies.”

“Default evaluation in Quad is the humane option.”

— *William Kahan*

System and Methods

On today's machines, Double is implemented in hardware, while Quad (if available) is typically a software library.

Fortunately, the GCC Fortran compiler now makes Quad available via the real(16) data type. We have therefore been able to make a Quad version of the Fortran 77 linear and nonlinear optimization solver MINOS using the gfortran compiler.

Our aim is to explore combined use of the Double and Quad MINOS simplex solvers for the solution of large multiscale linear programs. We seek greater efficiency than is normally possible with exact simplex solvers.

The primal simplex solver in MINOS includes

- geometric-mean scaling of the constraint matrix
- the EXPAND anti-degeneracy procedure
- partial pricing (but no steepest-edge pricing, which would generally reduce total iterations and time)
- Basis LU factorizations and updates via LUSOL

NEOS Statistics

NEOS

Free optimization solvers
via Argonne National Lab
(now Univ of Madison, Wisconsin)

NEOS Solver Statistics for 2 years 1 Jan 2012 -- 1 Jan 2014

Total Jobs 2218537

Solver	Submissions	Solver	Submissions	Solver	Submissions	Solver	Submissions
MINOS	774695	filter	8123	PATHNLP	1423	PGAPack	350
MINLP	514475	Couenne	7996	L-BFGS-B	1351	sd	124
KNITRO	276896	BDMLP	6691	ASA	1326	xpress	123
Gurobi	130334	PATH	6298	NLPEC	1281	Cplex	32
SNOPT	48281	bpmpd	6121	RELAX4	1265	DONLP2	3
Ipopt	46305	BLMVM	6005	condor	993	LGO	3
CONOPT	38331	NMTR	5248	SYMPHONY	871		
XpressMP	32688	AlphaECP	5201	sedumi	833		
MINTO	30367	OOQP	5147	icos	808		
csdp	28662	LANCELOT	5045	DSDP	805		
DICOPT	25524	MUSCOD-II	4973	Glpk	785		
BARON	25138	FilMINT	4523	PSwarm	784		
Cbc	23752	feaspump	3731	sdplr	741		
scip	21529	TRON	2237	Clp	735		
SBB	21466	MILES	1853	penbmi	573		
MOSEK	21192	LRAMBO	1774	bnbs	547		
Bonmin	19144	qsopt_ex	1718	nsips	516		
LOQO	16095	SDPA	1669	FortMP	492		
concorde	9652	sdpt3	1582	ddsip	489		
LINDOGlobal	8459	filterMPEC	1438	pensdp	447		

NEOS Solver Statistics for 2 years 1 Jan 2012 -- 1 Jan 2014

Total Jobs 2218537

Category	Submissions
nco	1170088
kestrel	533865
milp	190822
minco	117723
lp	81472
sdp	35312
go	29246
cp	23210
co	9676
bco	9585
uco	5248
miocp	4973
lno	4155
slp	1160
ndo	993
sio	516
socp	206

Input	Submissions
AMPL	1850882
GAMS	274585
SPARSE_SDPA	31266
MPS	15319
TSP	9652
Fortran	7811
CPLEX	7396
C	7375
MOSEL	4998
MATLAB_BINARY	2364
LP	1496
DIMACS	1148
ZIMPL	1078
SDPA	805
SMPS	671
MATLAB	402
SDPLR	332

Algorithm and Implementation

3-step procedure

1. Cold start Double MINOS with scaling and somewhat strict settings, save basis
2. Warm start Quad MINOS with scaling and tighter Feasibility and Optimality tols, save basis
3. Warm start Quad MINOS without scaling but tighter LU tols

MINOS runtime options for Steps 1–3

	Default Double	Step1 Double	Step2 Quad	Step3 Quad
Scale option	2	2	2	0
Feasibility tol	1e-6	1e-7	1e-15	1e-15
Optimality tol	1e-6	1e-7	1e-15	1e-15
LU Factor tol	100.0	10.0	10.0	5.0
LU Update tol	10.0	10.0	10.0	5.0
Expand frequency	10000	100000	100000	100000

Table: Three pilot models from Netlib, eight Mészáros *problematic* LPs, and three ME biochemical network models. Dimensions of $m \times n$ constraint matrices A and size of the largest optimal primal and dual variables x^*, y^*.

model	m	n	nnz(A)	max $\lvert A_{ij}\rvert$	$\lVert x^*\rVert_\infty$	$\lVert y^*\rVert_\infty$
pilot4	411	1000	5145	3e+04	1e+05	3e+02
pilot	1442	3652	43220	2e+02	4e+03	2e+02
pilot87	2031	4883	73804	1e+03	2e+04	1e+01
de063155	853	1488	5405	8e+11	3e+13	6e+04
de063157	937	1488	5551	2e+18	2e+17	6e+04
de080285	937	1488	5471	1e+03	1e+02	3e+01
gen1	770	2560	64621	1e+00	3e+00	1e+00
gen2	1122	3264	84095	1e+00	3e+00	1e+00
gen4	1538	4297	110174	1e+00	3e+00	1e+00
l30	2702	15380	64790	1e+00	1e+09	4e+00
iprob	3002	3001	12000	1e+04	3e+02	1e+00
TMA_ME	18210	17535	336302	2e+04	6e+00	1e+00
GlcAerWT	68300	76664	926357	8e+05	6e+07	2e+07
GlcAlift	69529	77893	928815	3e+05	6e+07	2e+07

Table: Itns and runtimes in secs for Step 1 (Double MINOS) and Steps 2–3 (Quad MINOS). Pinf and Dinf = $\log_{10}$ final maximum primal and dual infeasibilities. Problem iprob is infeasible. Bold figures show Pinf and Dinf at the end of Step 3. Pinf = **−99** means Pinf = 0. Pinf/ $\|x^*\|_\infty$ and Dinf/ $\|y^*\|_\infty$ are all $O(10^{-30})$ or smaller, even though only $O(10^{-15})$ was requested. This is an unexpectedly favorable empirical finding.

model	Itns	Times	Final objective	Pinf	Dinf
pilot4	1571	0.1	−2.5811392602e+03	−05	−13
	6	0.0	−2.5811392589e+03	−39	−31
	0	0.0	−2.5811392589e+03	**−99**	**−30**
pilot	16060	5.7	−5.5739887685e+02	−06	−03
	29	0.7	−5.5748972928e+02	−99	−27
	0	0.2	−5.5748972928e+02	**−99**	**−32**
pilot87	19340	15.1	3.0171038489e+02	−09	−06
	32	2.2	3.0171034733e+02	−99	−33
	0	1.2	3.0171034733e+02	**−99**	**−33**

model	Itns	Times	Final objective	Pinf	Dinf
de063155	921	0.0	1.8968704286e+10	−13	+03
	78	0.1	9.8830944565e+09	−99	−17
	0	0.0	9.8830944565e+09	**−99**	**−24**
de063157	488	0.0	1.4561118445e+11	+20	+18
	476	0.5	2.1528501109e+07	−27	−12
	0	0.0	2.1528501109e+07	**−99**	**−12**
de080285	418	0.0	1.4495817688e+01	−09	−02
	132	0.1	1.3924732864e+01	−35	−32
	0	0.0	1.3924732864e+01	**−99**	**−32**
gen1	369502	205.3	−1.6903658594e−08	−06	−12
	246428	9331.3	1.2935699163e−06	−12	−31
	2394	81.6	1.2953925804e−06	**−45**	**−30**
gen2	44073	60.0	3.2927907828e+00	−04	−11
	1599	359.9	3.2927907840e+00	−99	−29
	0	10.4	3.2927907840e+00	**−99**	**−32**
gen4	45369	212.4	1.5793970394e−07	−06	−10
	53849	14812.5	2.8932268196e−06	−12	−30
	37	10.4	2.8933064888e−06	**−54**	**−30**

model	Itns	Times	Final objective	Pinf	Dinf
l30	1229326	876.7	9.5266141574e−01	−10	−09
	275287	7507.1	−7.5190273434e−26	−25	−32
	0	0.2	−4.2586876849e−24	**−24**	**−33**
iprob	1087	0.2	2.6891551285e+03	+02	−11
	0	0.0	2.6891551285e+03	+02	−31
	0	0.0	2.6891551285e+03	+02	**−28**
TMA_ME	12225	37.1	8.0051076669e−07	−06	−05
	685	61.5	8.7036315385e−07	−24	−30
	0	6.7	8.7036315385e−07	**−99**	**−31**
GlcAerWT	62856	9707.3	−2.4489880182e+04	+04	−05
	5580	3995.6	−7.0382449681e+05	−07	−26
	4	60.1	−7.0382449681e+05	**−19**	**−21**
GlcAlift	134693	14552.8	−5.1613878666e+05	−03	−01
	3258	1067.1	−7.0434008750e+05	−09	−26
	2	48.1	−7.0434008750e+05	**−20**	**−22**

Multiscale NLPs

Systems biology FBA problems with variable μ

Analog filter design for a personalized hearing aid
(Jon Dattorro, Stanford)

FBA with nonlinear constraints

As coupling constraints are often functions of the organism's growth rate μ, Lerman et al. (UCSD) consider growth-rate optimization nonlinearly with the single μ as the objective instead of via a linear biomass objective function. Nonlinear constraints of the form

$$\frac{v_i}{v_j} \leq \mu \tag{7}$$

represented as

$$v_i \leq \mu v_j \tag{8}$$

are added to (2b), where v_i, v_j, μ are all variables. Constraints (8) are linear if μ is fixed at a specific value μ_k. Lerman et al. employ a binary search to find the largest $\mu_k \in [\mu_{\text{min}}, \mu_{\text{max}}]$ that keeps the associated LP feasible. Thus, the procedure requires reliable solution of a sequence of related LPs.

Analog filter design

NLP1	$\underset{\beta\ge 1,\,U,V,u,v\ge 0}{\text{minimize}}$	β	
	subject to	$\dfrac{1}{\beta} \le g_i^2 \dfrac{U_i}{V_i} \le \beta\,,$	$\omega_i \in \Omega$

where

$$U_i(u) \equiv 1 + u_1\omega_i^2 + u_2\omega_i^4$$
$$V_i(v) \equiv 1 + v_1\omega_i^2 + v_2\omega_i^4$$

19 frequencies ω_i (Hz):

$$\omega = 2\pi[\,30 \;\; 45 \;\; 60 \;\; 90 \;\; 125 \;\; 187 \;\; 250 \;\; 375 \;\; 500 \;\; 750 \;\; \ldots \;\; 1000 \;\; 1500 \;\; 2000 \;\; 3000 \;\; 4000 \;\; 6000 \;\; 8000 \;\; 12000 \;\; 16000\,]^{\mathrm{T}}$$

19 filter magnitudes:

$$g = [\,1. \;\; 1.2589 \;\; 2.2387 \;\; 2.5119 \;\; 2.8184 \;\; 5.0119 \;\; 5.0119 \;\; 7.9433 \;\; 10. \;\; 6.3096 \;\; \ldots \;\; 6.3096 \;\; 4.4668 \;\; 6.3096 \;\; 10. \;\; 7.9433 \;\; 14.125 \;\; 25.119 \;\; 44.668 \;\; 79.433\,]^{\mathrm{T}}$$

Analog filter design

NLP2

$$\begin{aligned} \underset{\beta\ge 1,\, U, V, u, v \ge 0}{\text{minimize}} \quad & \beta \\ \text{subject to} \quad & \beta V_i - \gamma_i U_i \ge 0\,, \qquad \omega_i \in \Omega \\ & \beta U_i - \gamma_i^{-1} V_i \ge 0 \\ & U_i - \omega_i^2 u_1 - \omega_i^4 u_2 = 1 \\ & V_i - \omega_i^2 v_1 - \omega_i^4 v_2 = 1 \end{aligned}$$

$$\omega = 2\pi[\,30\ \ 45\ \ 60\ \ 90\ \ 125\ \ 187\ \ 250\ \ 375\ \ 500\ \ 750\ \ \ldots \\ 1000\ \ 1500\ \ 2000\ \ 3000\ \ 4000\ \ 6000\ \ 8000\ \ 12000\ \ 16000\,]^{\mathrm{T}}$$

$$g = [\,1.\ \ 1.2589\ \ 2.2387\ \ 2.5119\ \ 2.8184\ \ 5.0119\ \ 5.0119\ \ 7.9433\ \ 10.\ \ 6.3096\ \ \ldots \\ 6.3096\ \ 4.4668\ \ 6.3096\ \ 10.\ \ 7.9433\ \ 14.125\ \ 25.119\ \ 44.668\ \ 79.433\,]^{\mathrm{T}}$$

$\gamma_i \equiv g_i^2$ β, U_i, V_i appear nonlinearly

Analog filter design results

With $\beta \equiv \beta_0 = 5.0$ fixed, the problem is effectively an LP.

With scaling, the "LP" and then NLP2 solve as follows:

	major itns	minor itns	f/g evaluations	Pinf	Dinf
LP	3	9	7		
NLP2	13	33	79	0.0	5×10^{-23}

$$\beta = 2.7837077182, \quad u_1 = 1.333433 \times 10^{-6}, \quad u_2 = 0.0$$
$$v_1 = 4.853544 \times 10^{-5}, \quad v_2 = 2.942739 \times 10^{-13}$$

Improvement if the frequencies ω_i are measured in kHz instead of Hz:

	major itns	minor itns	f/g evaluations	Pinf	Dinf
LP	2	8	5		
NLP2	12	19	39	0.0	5×10^{-31}

$$\beta = 2.7837077182, \quad u_1 = 1.333433 \times 10^{-0}, \quad u_2 = 0.0$$
$$v_1 = 4.853544 \times 10^{+1}, \quad v_2 = 2.942739 \times 10^{-1}$$

LPnetlib test problems

Unexpectedly high accuracy in Double and Quad

62 classic LP problems (ordered by file size)

afiro
stocfor1
adlittle
scagr7
sc205
share2b
recipe
vtpbase
share1b
bore3d
scorpion
capri
brandy
scagr25
sctap1
israel
scfxm1
bandm
e226
grow7
etamacro
agg
scsd1
standata
beaconfd
gfrdpnc
stair
scrs8
shell
scfxm2
pilot4
scsd6
ship04s
seba
grow15
fffff800
scfxm3
ship04l
ganges
sctap2
grow22
ship08s
stocfor2
pilotwe
ship12s
25fv47
sierra
czprob
pilotja
ship08l
nesm
ship12l
cycle
greenbea
greenbeb
80bau3b
d2q06c
woodw
d6cube
pilot
wood1p
pilot87

LP experiment

MINOS double precision `real(8)` $\epsilon = $ 2.2e-16

```
Feasibility tol = 1e-8
Optimality  tol = 1e-8
```

- Cold start with scaling and other defaults
- Warm start, no scaling, LU rook pivoting
- Plot max primal and dual infeasibilities $\log_{10} \frac{\text{Pinf}}{\|x^*\|_\infty}, \quad \log_{10} \frac{\text{Dinf}}{\|y^*\|_\infty}$

Compare with MINOS quad precision `real(16)` $\epsilon = $ 1.9e-35

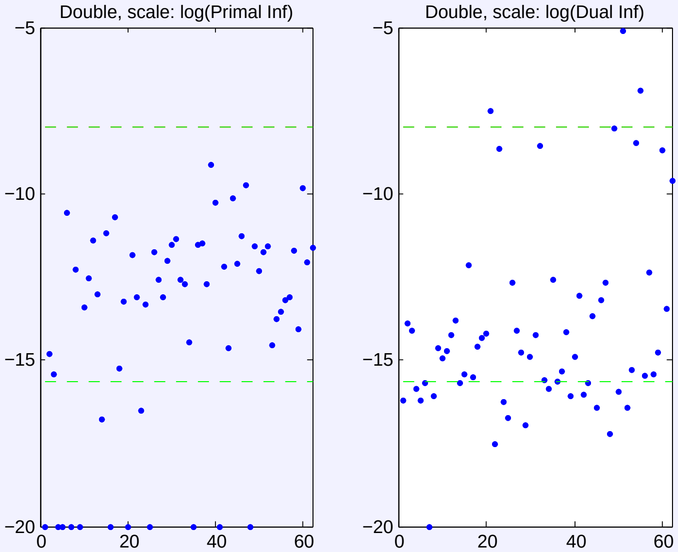
```
Feasibility tol = 1e-17
Optimality  tol = 1e-17
```

Double precision, cold start: Max primal/dual infeas

```
Scale option 2
Feasibility tol 1e-8
Optimality  tol 1e-8

LU Partial Pivoting
LU Factor tol  100.0
LU Update tol   10.0
```

$\epsilon = 2.2\text{e-}16$

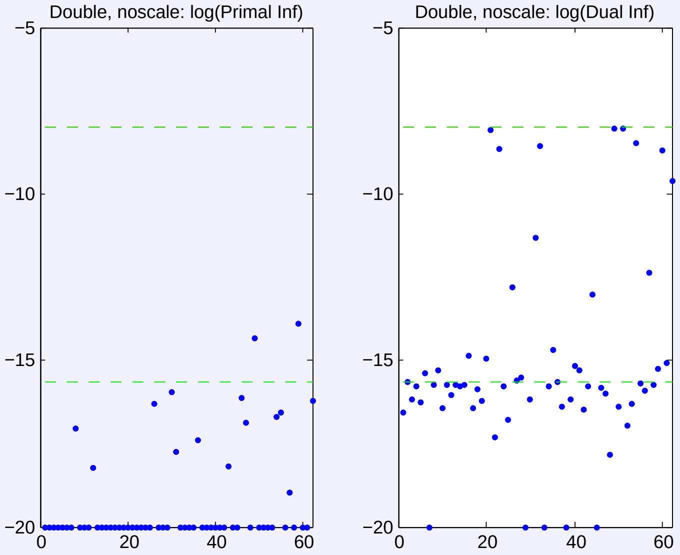


Double precision, warm start: Max primal/dual infeas

```
Scale option 0
Feasibility tol 1e-8
Optimality  tol 1e-8

LU Rook Pivoting
LU Factor tol     4.0
LU Update tol     4.0
```

$\epsilon = 2.2\text{e-}16$

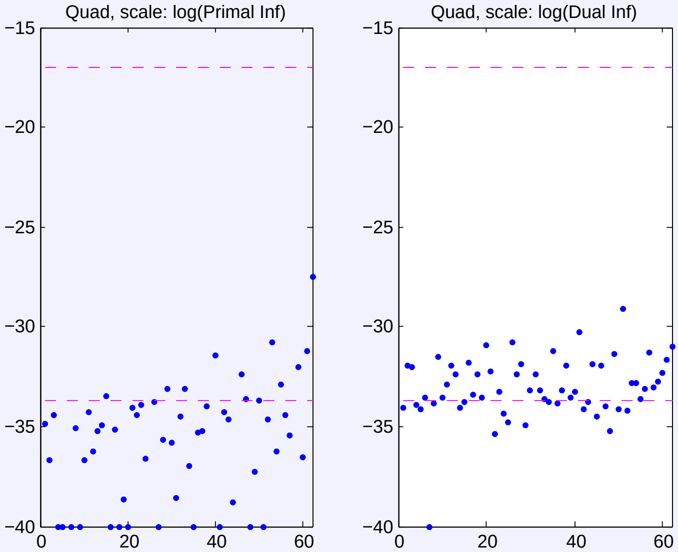


Quad precision, cold start: Max primal/dual infeas

```
Scale option 2
Feasibility tol 1e-17
Optimality  tol 1e-17

LU Partial Pivoting
LU Factor tol   100.0
LU Update tol    10.0
```

$\epsilon = 1.9\text{e-}35$

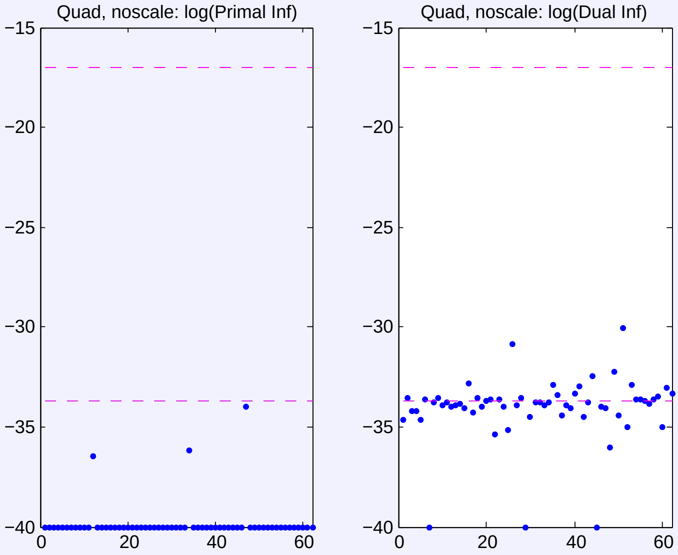


Quad precision, warm start: Max primal/dual infeas

```
Scale option 0
Feasibility tol 1e-17
Optimality  tol 1e-17

LU Rook Pivoting
LU Factor tol    4.0
LU Update tol    4.0
```

$\epsilon = 1.9\text{e-}35$



Conclusions

Conclusions

Just as double-precision floating-point hardware revolutionized scientific computing in the 1960s, the advent of quad-precision data types (even in software) brings us to a new era of greatly improved reliability in optimization solvers.

— Michael Saunders

Reference

Ding Ma and Michael Saunders (2014). Solving multiscale linear programs using the simplex method in quadruple precision. http://stanford.edu/group/SOL/multiscale/papers/quadLP3.pdf

Special thanks

- George Dantzig, born 100 years ago yesterday
- William Kahan, IEEE floating-point standard, including Quad
- Ronan Fleming, Ines Thiele (Luxembourg)
- Bernhard Palsson, Josh Lerman, Teddy O'Brien, Laurence Yang (UCSD)
- Ed Klotz (IBM CPLEX), Yuekai Sun, Jon Dattorro (Stanford)

FAQ

FAQ

- Is quadMINOS available? Yes, free to academics
- Can quadMINOS be called from Matlab or Tomlab? No, Matlab uses an old GCC
- Is quadMINOS available in GAMS? Soon Yes for LP
- How about AMPL? No, but should be feasible for LP
- Is there a quadSNOPT? Yes, in f90 snopt9 we can change 1 line
- Can CPLEX / Gurobi / Mosek / ... help? Yes, they can provide Presolve and Warm start, especially from GAMS
- Will Quad hardware eventually be standard? We hope so