Large-Scale Linear Algebra and Its Role in Optimization

Michael Saunders

MS&E and ICME, Stanford University

SIAM Conference on Optimization (SIOPT17)

Vancouver, BC May 22–25, 2017
Abstract

Numerical optimization has always needed reliable numerical linear algebra. Some worries go away when we use quad-precision floating-point (as is now possible with the GCC and Intel compilers). quad-MINOS has proved useful in systems biology for optimizing multiscale metabolic networks, and in computer experimental design for optimizing the IMSPE function with finite-difference gradients (confirming the existence of twin points). Recently quad-MINOS gave a surprising solution to the optimal control problem spring200.

Quad versions of SNOPT and PDCO have been developed in f90 and C++ respectively. Changing a single line of source code leads to double-SNOPT or quad-SNOPT. The C++ PDCO can switch from double to quad at runtime.

Partially supported by the National Institute of General Medical Sciences of the National Institutes of Health (NIH) Award U01GM102098.
Pre-history
Simplex via Cholesky

\[ B = LQ, \quad Q \text{ not kept, replace one col of } B \]

\[ LL^T \leftarrow LL^T + vv^T - ww^T \]

or \[ LL^T - ww^T + vv^T \]

- **Gill and Murray (1973)** A numerically stable form of the simplex method
- **Saunders (1972)** Large-scale Linear Programming using the Cholesky Factorization

Perhaps a modern version is possible via \( LL^T \) or \( QR \):

- **Chen, Davis, Hager and Rajamanickam (2008)** Algorithm 887, CHOLMOD
  Supernodal Sparse Cholesky Factorization and Update/Downdate

- **Davis (2011)** Algorithm 915, SuiteSparseQR
  Multifrontal multithreaded rank-revealing sparse QR factorization
LU is more sparse than LQ

Early linear programming systems assumed $B$ is close to triangular.

- Hellerman and Rarick (1971, 1972) The (partitioned) preassigned pivot procedure $P^3$, $P^4$
- Saunders (1976) The complexity of LU updating in the simplex method, MINOS

\[
B = \begin{bmatrix}
  x & x & x & * & * \\
  x & x & x & * & * \\
  x & x & x & * & * \\
  x & x & x & * & * \\
  x & x & x & x & x \\
  x & x & x & x & x \\
\end{bmatrix}
\]
Markowitz LU is more sparse than P^4

Early 1980s: Rob Burchett, General Electric
Basis matrices were close to symmetric

Sparse LU with Markowitz merit function

- Duff and Reid (1977) Fortran subroutines for sparse unsymmetric linear equations, MA28
- Reid (1982) A sparsity-exploiting variant of the Bartels-Golub decomposition, LA05
- Gill, Murray, S, & Wright (1987) Maintaining LU factors of a general sparse matrix, LUSOL

LUSOL does the linear algebra for MINOS, SQOPT, SNOPT, MILES, PATH, lp_solve
McDonnell Douglas

Huntington Beach, CA

SQP methods

NPSOL, SNOPT
Aerospace Applications of NPSOL and SNOPT

OTIS #1
DC-Y single-stage-to-orbit

SSTO

A reusable, single-stage-to-orbit-and-return space transportation system

Delta Clipper's robust vehicle design, streamlined ground turnaround, and autonomous flight operations are the keys to reliable, low-cost routine space transportation.
Pre-history  McDonnell Douglas  Block-LU updates  Quad Precision  DQQ procedure  Quad NLP  Finite-difference gradients  PDCO  spring200

Linear Algebra and Optimization  SIOPT17 Vancouver  10/40

OTIS

DC-Y Landing Maneuver

Retract air brakes at
2800 ft
420 mph
Control engineer
Feasible solution
1 year

2000 x 1200 NLP

NPSOL
Optimal solution
50% of fuel saved
DC-Y landing, 2nd OTIS/NPSOL optimization

- 1st optimization: starting altitude = 2800ft
- 2nd optimization: starting altitude = variable
- New constraint needed: Don’t exceed 3g

Optimum starting altitude = 1400ft(!)

Come back Alan Shephard!
DC-X flying model
1/3 scale = 40ft tall

- 1993-95: DC-X made 8 flights
  Flight 8: demonstrated turnaround maneuver; hard landing damaged aeroshell

- 1996: DC-XA made 4 flights
  Flight 3: demonstrated 26-hour turnaround
  Flight 4: landing strut failed to extend; tipped over and exploded

- 1997: McDonnell Douglas merges with Boeing
  Huntington Beach campus becomes part of Boeing
  Philip continued 5-to-8 days for several years (till Rocky Nelson retired)
McDonnell Douglas motivation

The aerospace problems kept getting bigger

SQP needs Hessian $H$ for QP subproblems and null-space operator $Z$ for constraints

**NPSOL**
- dense quasi-Newton $H = R^T R$
- dense $Z$ from $J^T = QR$

**SNOPT**
- limited-memory $H$
- $Z$ from sparse $B = LU$ (reduced-gradient method)
- SQIC can switch from $B = LU$ to block-LU updates of $K$
Block-LU updates
Block-LU updates

- Bisschop and Meeraus (1977) Matrix augmentation and partitioning in updating the basis inverse
- Eldersveld and S (1992) A block-LU update for large-scale linear programming, MINOS/SC
- Huynh (2008) A large-scale QP solver based on block-LU updates of the KKT system, QPBLU
- Maes (2010) Maintaining LU factors of a general sparse matrix, QPBLUR
- Wong (2011) Active-set methods for quadratic programming, icQP
- Gill and Wong (2014) Software for large-scale quadratic programming, SQIC

\[
B_0 = L_0 U_0 \quad \text{LUSOL, BG updates}
\]
\[
B_k \equiv \begin{pmatrix} B_0 & V_k \\ E_k & \end{pmatrix} \quad \text{not implemented}
\]
\[
K_0 = L_0 U_0 \quad \text{LUSOL, MA57, MA97}
\]
\[
K_k \equiv \begin{pmatrix} K_0 & V_k \\ V_k^T & D_k \end{pmatrix} \quad \text{SuperLU, UMFPACK}
\]
Quad Precision
“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, 2011
<table>
<thead>
<tr>
<th>Double MINOS</th>
<th>Quad MINOS</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>real(8)</code></td>
<td><code>real(16)</code></td>
</tr>
<tr>
<td><code>eps = 2.22e-16</code></td>
<td><code>eps = 1.93e-34</code></td>
</tr>
</tbody>
</table>

We use this humane approach to Quad implementation

2 source codes  2 programs
snopt9 = Double or Quad SQOPT, SNOPT

snPrecision.f90

module snModulePrecision
    integer(4), parameter :: ip = 4, rp = 8 ! double
    ! integer(4), parameter :: ip = 8, rp = 16 ! quad
end module snModulePrecision

module sn50lp
    use snModulePrecision, only : ip, rp
    subroutine s5solveLP ( x, y )
    real(rp), intent(inout) :: x(nb), y(nb)
end subroutine s5solveLP
DQQ procedure
for multiscale LP and NLP

Developed for systems biology models of metabolism
Ding Ma, Laurence Yang, MS, et al. (2017) Reliable and efficient solution of genome-scale models of Metabolism and macromolecular Expression Double and Quad MINOS
## Meszaros “problematic” LP test set

<table>
<thead>
<tr>
<th></th>
<th>Itns</th>
<th>Times</th>
<th>Final objective</th>
<th>Pinf</th>
<th>Dinf</th>
</tr>
</thead>
<tbody>
<tr>
<td>gen1</td>
<td>369502</td>
<td>205.3</td>
<td>$-1.6903658594e-08$</td>
<td>$-06$</td>
<td>$-12$</td>
</tr>
<tr>
<td></td>
<td>246428</td>
<td>9331.3</td>
<td>$1.2935699163e-06$</td>
<td>$-12$</td>
<td>$-31$</td>
</tr>
<tr>
<td></td>
<td>2394</td>
<td>81.6</td>
<td>$1.2953925804e-06$</td>
<td>$-45$</td>
<td>$-30$</td>
</tr>
<tr>
<td></td>
<td>44073</td>
<td>60.0</td>
<td>$3.2927907828e+00$</td>
<td>$-04$</td>
<td>$-11$</td>
</tr>
<tr>
<td></td>
<td>1599</td>
<td>359.9</td>
<td>$3.2927907840e+00$</td>
<td>$-29$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>10.4</td>
<td>$3.2927907840e+00$</td>
<td>$-32$</td>
<td></td>
</tr>
<tr>
<td>gen2</td>
<td>45369</td>
<td>212.4</td>
<td>$1.5793970394e-07$</td>
<td>$-06$</td>
<td>$-10$</td>
</tr>
<tr>
<td></td>
<td>53849</td>
<td>14812.5</td>
<td>$2.8932268196e-06$</td>
<td>$-12$</td>
<td>$-30$</td>
</tr>
<tr>
<td></td>
<td>37</td>
<td>10.4</td>
<td>$2.8933064888e-06$</td>
<td>$-54$</td>
<td>$-30$</td>
</tr>
<tr>
<td>gen4</td>
<td>1229326</td>
<td>876.7</td>
<td>$9.5266141574e-01$</td>
<td>$-10$</td>
<td>$-09$</td>
</tr>
<tr>
<td></td>
<td>275287</td>
<td>7507.1</td>
<td>$-7.5190273434e-26$</td>
<td>$-25$</td>
<td>$-32$</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0.2</td>
<td>$-4.2586876849e-24$</td>
<td>$-24$</td>
<td>$-33$</td>
</tr>
</tbody>
</table>

$P_{inf}, D_{inf} = \log_{10} \text{Primal/Dual infeasibilities}$
Systems biology multiscale LP modes

<table>
<thead>
<tr>
<th></th>
<th>Itns</th>
<th>Times</th>
<th>Final objective</th>
<th>Pinf</th>
<th>Dinf</th>
</tr>
</thead>
<tbody>
<tr>
<td>TMA_ME</td>
<td>12225</td>
<td>37.1</td>
<td>8.0051076669e−07</td>
<td>−06</td>
<td>−05</td>
</tr>
<tr>
<td></td>
<td>685</td>
<td>61.5</td>
<td>8.7036315385e−07</td>
<td>−24</td>
<td>−30</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>6.7</td>
<td>8.7036315385e−07</td>
<td>−</td>
<td>−31</td>
</tr>
<tr>
<td>GlcAerWT</td>
<td>62856</td>
<td>9707.3</td>
<td>−2.4489880182e+04</td>
<td>+04</td>
<td>−05</td>
</tr>
<tr>
<td></td>
<td>5580</td>
<td>3995.6</td>
<td>−7.0382449681e+05</td>
<td>−07</td>
<td>−26</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>60.1</td>
<td>−7.0382449681e+05</td>
<td>−19</td>
<td>−21</td>
</tr>
<tr>
<td>GlcAlift</td>
<td>134693</td>
<td>14552.8</td>
<td>−5.1613878666e+05</td>
<td>−03</td>
<td>−01</td>
</tr>
<tr>
<td></td>
<td>3258</td>
<td>1067.1</td>
<td>−7.0434008750e+05</td>
<td>−09</td>
<td>−26</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>48.1</td>
<td>−7.0434008750e+05</td>
<td>−20</td>
<td>−22</td>
</tr>
</tbody>
</table>

Final $P_{\infty}/\|x^*\|_\infty$ and $D_{\infty}/\|y^*\|_\infty$ are $O(10^{-30})$
Quad NLP

Metabolic models and macromolecular expression (ME models)

Laurence Yang, UC San Diego
Quadratic convergence of major iterations (Robinson 1972)

quadMINOS  5.6  (Nov 2014)

Begin tinyME-NLP  cold start NLP with mu = mu0
Itn 304 -- linear constraints satisfied.
Calling funcon.  mu = 0.800000000000000000000000000000004
nnCon, nnJac, neJac 1073 1755 2681
funcon sets 2681 out of 2681 constraint gradients.
funobj sets 1 out of 1 objective gradients.

<table>
<thead>
<tr>
<th>Major</th>
<th>minor</th>
<th>step</th>
<th>objective</th>
<th>Feasible</th>
<th>Optimal</th>
<th>nsb</th>
<th>ncon</th>
<th>penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>304T</td>
<td>0.0E+00</td>
<td>8.000000E-01</td>
<td>6.1E-03</td>
<td>2.1E+03</td>
<td>0</td>
<td>4 1.0E+02</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>561T</td>
<td>1.0E+00</td>
<td>8.000000E-01</td>
<td>2.6E-14</td>
<td>3.2E-04</td>
<td>0</td>
<td>46 1.0E+02</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>40T</td>
<td>1.0E+00</td>
<td>8.28869E-01</td>
<td>5.4E-05</td>
<td>3.6E-05</td>
<td>0</td>
<td>87 1.0E+02</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>7</td>
<td>1.0E+00</td>
<td>8.46923E-01</td>
<td>1.2E-05</td>
<td>2.9E-06</td>
<td>0</td>
<td>96 1.0E+02</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>0</td>
<td>1.0E+00</td>
<td>8.46948E-01</td>
<td>4.2E-10</td>
<td>2.6E-10</td>
<td>0</td>
<td>97 1.0E+02</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>0</td>
<td>1.0E+00</td>
<td>8.46948E-01</td>
<td>7.9E-23</td>
<td>1.2E-20</td>
<td>0</td>
<td>98 1.0E+01</td>
</tr>
</tbody>
</table>

EXIT -- optimal solution found 13.5 secs
Quasi-Newton optimization with finite-difference gradients
Design of computer experiments  

Selden Crary, indie-physicist, 2015

\[ n = 11 \text{ points } (x_i, y_i) \text{ on } [-1, 1] \text{ square (one twin-point)} \]

\[ [d, n, p, \theta_1, \theta_2] = [2, 11, 2, 0.128, 0.069] \]
Design of computer experiments

Selden Crary, physicist, 2015

**IMSPE-optimal designs (integrated mean-squared prediction error)**

\[
\min 1 - \text{trace}(B^{-1}A)
\]

A and B: symmetric matrices of order \(n + 1\)

B increasingly ill-conditioned if points approach each other

**2D, Gaussian covariance parameters \(\sigma, \theta_1, \theta_2\)**

\[
A \propto \int_{-1}^{1} \int_{-1}^{1} \begin{bmatrix} 1 \\ \nu \end{bmatrix} \begin{bmatrix} 1 & \nu^T \end{bmatrix} \, dx \, dy
\]

\[
B = \begin{bmatrix} 0 & 1^T \\ 1 & V \end{bmatrix}
\]

\(\nu_i\) functions of \(\exp(\cdot)\) and \(\text{erf}(\cdot)\),

\[
V_{ij} = \sigma^2 e^{-\theta_1(x_i-x_j)^2-\theta_2(y_i-y_j)^2}
\]
$C^\infty$ a.e.

"Post", not pole

Need multistarts

With Maple, Selden has found twin-points, triple-points, ... and a new class of rational functions (the Nu class)

Smooth sailing through the black-hole/white-hole event horizon

No string theory

No complex numbers in quantum mechanics

...
**IMSPE, 2D, n = 11, θ = (0.128, 0.069)**

Quad MINOS unconstrained optimization \(\in \mathbb{R}^{22}\) without gradients 6 secs

<table>
<thead>
<tr>
<th>Itn</th>
<th>ph</th>
<th>pp</th>
<th>rg</th>
<th>step</th>
<th>objective</th>
<th>nobj</th>
<th>nsb</th>
<th>cond(H)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>0</td>
<td>3.9E-05</td>
<td>1.0E+03</td>
<td>2.47305090E-05</td>
<td>57</td>
<td>22</td>
<td>4.5E+01</td>
</tr>
<tr>
<td>50</td>
<td>4</td>
<td>1</td>
<td>6.4E-07</td>
<td>6.2E+00</td>
<td>6.01181966E-06</td>
<td>1384</td>
<td>22</td>
<td>2.1E+04</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>1</td>
<td>-7.6E-08</td>
<td>1.1E+00</td>
<td>5.65611811E-06</td>
<td>2726</td>
<td>22</td>
<td>1.4E+05</td>
</tr>
<tr>
<td>150</td>
<td>4</td>
<td>1</td>
<td>8.5E-07</td>
<td>6.0E+00</td>
<td>5.11053080E-06</td>
<td>4102</td>
<td>22</td>
<td>8.2E+03</td>
</tr>
<tr>
<td>200</td>
<td>4</td>
<td>1</td>
<td>2.6E-08</td>
<td>1.1E-01</td>
<td>5.02762155E-06</td>
<td>5464</td>
<td>22</td>
<td>1.0E+07</td>
</tr>
<tr>
<td>239</td>
<td>4</td>
<td>1</td>
<td>1.1E-07</td>
<td>1.5E-06</td>
<td>5.02762154E-06</td>
<td>7478</td>
<td>22</td>
<td>1.0E+10</td>
</tr>
</tbody>
</table>

Search exit 7 -- too many functions.

EXIT -- optimal solution found

No. of iterations 239 Objective value 5.0276215358E-06
No. of calls to funobj 7538 Calls with mode=2 (f, known g) 244
Calls for forward differencing 4466 Calls for central differencing 1716
Max Primal infeas 0 0.0E+00 Max Dual infeas 2 1.1E-07

not great ↑
17 points on $[-1, 1]^2$
(two twin-points)

Quad MINOS design refined by Selden via MAPLE

After further downhill searching, both sets of twins are now close, confirming the conjecture of a true double–twin-point design. 20161106

\[ \theta_2 = 0.006 \]
\[ \text{max } 2\delta_i = 0.009 \]
Linear algebra question

\[ A, B \text{ real, symmetric, indefinite, ill-conditioned} \]

\[ \min \text{ IMSPE} = 1 - \text{trace}(B^{-1}A) \]

\[ \text{trace}(B^{-1}A) = \sum \lambda_i \]

- QZ algorithm ignores symmetry but avoids ill-conditioned \( B^{-1} \)
- Will QZ compute real \( \lambda_i \)?

Yuji Nakatsukasa (Oxford) is developing \texttt{qdwhgep.m} for \( Ax = \lambda Bx \) (real, symmetric)

- Congruence transformations are real \( P^TAPy = \lambda P^TPPy \)
- Eigenvalues can be complex conjugate pairs
- \( \text{trace}(B^{-1}A) = \sum \lambda_i \) will be real
PDCO in C++

Ron Estrin, UBC → Stanford
PDCO in C++

Matlab PDCO: regularized convex optimization \((D_1, D_2 \succ 0, \text{diagonal})\)

\[
\begin{align*}
\text{minimize} \quad & \phi(x) + \frac{1}{2} \|D_1 x\|^2 + \frac{1}{2} \|r\|^2 \\
\text{subject to} \quad & Ax + D_2 r = b, \quad \ell \leq x \leq u,
\end{align*}
\]

- Needed for huge metabolic LP models that are near-block-diagonal
- C++ has **double** and **float128** data-types
- Compiler generates multiple codes
- Switch from **double** to **quad** at run-time

1 source code 1 program
spring200

An optimal control problem modeling a spring/mass/damper
spring200

\[ \min f(y, z, u) = \frac{1}{2} \sum_{t=0}^{T} z_t^2 \]

\[
y_{t+1} = y_t - 0.01y_t^2 - 0.004z_t + 0.2u_t
\]

\[
z_{t+1} = z_t + 0.2y_t \quad t = 0, \ldots, T - 1
\]

\[-1 \leq y_t \leq 0.2 \leq u_t \leq 0.2 \]

\[
y_0 = 0 \quad y_T = 0 \quad z_0 = 10
\]

<table>
<thead>
<tr>
<th>Opt tol</th>
<th>Majors</th>
<th>Minors</th>
<th>Superbasics</th>
<th>Objective</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>double</td>
<td>1e-06</td>
<td>13</td>
<td>576</td>
<td>18</td>
<td>1186.3839</td>
</tr>
<tr>
<td>quad</td>
<td>1e-15</td>
<td>31</td>
<td>1282</td>
<td>113</td>
<td>1186.3820</td>
</tr>
</tbody>
</table>

quad-MINOS gives an unexpectedly “clean” solution
(many variables exactly zero, including control variables \( u_t \))
double-MINOS
quad-MINOS
PEG: Tireless teacher, author, implementer

- Let’s get things nice and sparkling clear.
- I’ve taught you much, my little droogies.
- It had been a wonderful evening and what I needed now, to give it the perfect ending, was a little of the Ludwig Van.

— Alex, in “Clockwork Orange”

HAPPY ROUND NUMBER droogie!