Les Cahiers du GERAD ISSN: 0711–2440

A collection of linear systems arising from interior-point methods for quadratic optimization

D. Orban

G–2015–117

November 2015

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

A collection of linear systems arising from interior-point methods for quadratic optimization

Dominique Orban

GERAD & Department of Mathematics and Industrial Engineering, Polytechnique Montréal, Montréal (Québec) Canada, H3C 3A7

dominique.orban@gerad.ca

November 2015

Les Cahiers du GERAD

G–2015–117

Abstract: We describe a collection of linear systems generated during the iterations of an interior-point method for convex quadratic optimization. As the iteration index grows, the systems may become increasingly ill conditioned. Each system represents the linearization of Newton's equations about an iterate and for a certain value of the barrier parameter. A salient feature of the collection is that each system comes in the form of its block components. Thus, it is possible to use the collection to benchmark direct or iterative solvers on various formulations of the linearized Newton equations. Matlab tools are supplied to facilitate forming those formulations.

Key Words: Interior-point methods, linear systems, quadratic optimization.

Résumé : Une collection de systèmes linéaires engendrés au cours des itérations d'une méthode de points intérieurs pour l'optimisation quadratique convexe est présentée. À mesure que le compteur d'itérations augmente, le conditionnement des systèmes se détériore. Chaque système est la linéarisation des équations de Newton autour d'un itéré pour une certaine valeur du paramètre barrière. Une caractéristique principale de la collection est que chaque système est fourni sous la forme des blocs qui le composent. Il est ainsi possible d'utiliser la collection pour évaluer la performance de méthodes directes et itératives sur diverses formulations des équations de Newton linéarisées. Plusieurs outils Matlab sont fournis pour assembler ces formulations.

Mots clés : Méthode de points intérieurs, systèmes linéaires, optimisation quadratique.

Acknowledgments: Research partially supported by NSERC Discovery Grant 299010-04.

1 Introduction

Interior-point methods for quadratic optimization give rise to linear systems that can be formulated so as to span a range of possibilities, each with its own advantages and challenges. A distinguishing feature of those methods is that the conditioning of the linear systems to be solved at each iteration worsens as a solution is approached. No existing collection of linear systems appears to reflect this feature. As a consequence, existing collections are of limited use in the study and design of factorization-free interior-point methods.

We propose a collection of systems generated during the iterations of an actual interior-point method for convex quadratic optimization. The collection possesses several unique features. The first is that, for a given optimization problem, several related systems are available, each generated at a different interior-point iteration. The second is that the collection features *systems* and not only matrices, i.e., accompanying right-hand sides are provided that reflect a realistic situation. A third feature is that the matrices are supplied in the form of their building blocks. This allows several formulations of the same system to be assembled.

The collection is available from https://dx.doi.org/10.5281/zenodo.34130 and may be cited as (Orban, 2015b). The source repository for the collection is located at https://github.com/optimizers/ip-systems.

Our hope is that this collection will aid in the design of robust factorization-free interior-point methods and of effective preconditioners. In the rest of this document, we cover the basics of interior-point methods for quadratic optimization and the components of the linear systems.

2 Interior-point methods and system formulations

Friedlander and Orban (2012) propose a general framework for the convex quadratic optimization problem in standard form that consists in introducing primal and dual regularization terms. The regularized problem reads

$$\begin{aligned} &\underset{x\in\mathbb{R}^n, r\in\mathbb{R}^m}{\text{minimize}} \ c^T x + \tfrac{1}{2}x^T H x + \tfrac{1}{2}\rho\|x - x_k\|^2 + \tfrac{1}{2}\delta\|r + y_k\|^2 \\ &\text{subject to } Jx + \delta r = b, \ x \geq 0, \end{aligned} \tag{1}$$

where $c \in \mathbb{R}^n$, $H = H^T \in \mathbb{R}^{n\times n}$ is positive semi-definite, $J \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, ρ and δ are positive regularization parameters, x_k is the current iterate, and y_k is the current Lagrange multiplier estimate associated to the equality constraints. The usual quadratic optimization problem results from choosing $\rho = \delta = 0$. Applying a primal-dual interior-point method to (1) results in solving a sequence of nonlinear systems of the form

$$\begin{bmatrix} c + Hx + \rho(x - x_k) - J^T y - z \\ Jx + \delta(y - y_k) - b \\ Xz - \mu e \end{bmatrix} = 0, \tag{2}$$

where $X := \operatorname{diag}(x)$, $\mu > 0$ is the barrier parameter, and e is the vector of ones, while maintaining $(x, z) > 0$. Computing a Newton step for (2) from $(x, y) = (x_k, y_k)$ yields the classic linear system

$$\begin{bmatrix} H + \rho I & J^T & -I \\ J & -\delta I & \\ Z & & X \end{bmatrix} \begin{bmatrix} \Delta x \\ -\Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} J^T y - c - Hx + z \\ b - Jx \\ \mu e - Xz \end{bmatrix}, \tag{3}$$

where I is the identity matrix and $Z := \operatorname{diag}(z)$. The matrix of (3) will be denoted K_3. Though it is not symmetric, it is structurally symmetric and is symmetrizable. A critical feature of (3) is that it remains well defined and uniformly nonsingular in the limit, even as some components of x and/or z approach zero, provided that *strict complementarity* is satisfied, i.e., each component of $x + z$ remains positive. If $\rho \geq 0$, if

$H + \rho I$ is positive definite, and if $\delta > 0$, the leading 2×2 block matrix

$$\begin{bmatrix} H + \rho I & J^T \\ J & -\delta I \end{bmatrix}$$

is symmetric and quasi definite (Vanderbei, 1995).

One possible symmetrization of (3) consists in using the similarity transformation

$$\begin{bmatrix} I & & \\ & I & \\ & & Z^{-\frac{1}{2}} \end{bmatrix} \begin{bmatrix} H + \rho I & J^T & -I \\ J & -\delta I & \\ Z & & X \end{bmatrix} \begin{bmatrix} I & & \\ & I & \\ & & Z^{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} \Delta x \\ -\Delta y \\ Z^{-\frac{1}{2}} \Delta z \end{bmatrix} = \begin{bmatrix} J^T y - c - Hx + z \\ b - Jx \\ Z^{-\frac{1}{2}}(\mu e - Xz) \end{bmatrix},$$

and changing the sign of the last block row, i.e.,

$$\begin{bmatrix} H + \rho I & J^T & -Z^{\frac{1}{2}} \\ J & -\delta I & \\ -Z^{\frac{1}{2}} & & -X \end{bmatrix} \begin{bmatrix} \Delta x \\ -\Delta y \\ Z^{-\frac{1}{2}} \Delta z \end{bmatrix} = \begin{bmatrix} J^T y - c - Hx + z \\ b - Jx \\ Z^{-\frac{1}{2}}(\mu e - Xz) \end{bmatrix}. \tag{4}$$

The matrix of (4) is called $K_{3.5}$.

Another possibility is to eliminate Δz from (3) to obtain

$$\begin{bmatrix} H + \rho I + X^{-1}Z & J^T \\ J & -\delta I \end{bmatrix} \begin{bmatrix} \Delta x \\ -\Delta y \end{bmatrix} = \begin{bmatrix} J^T y - c - Hx + \mu X^{-1} e \\ b - Jx \end{bmatrix}. \tag{5}$$

The matrix of (5) is called K_2. A potential advantage of (5) is that it is smaller than (3) and (4) while being symmetric. Unfortunately, the condition number of K_2 is unbounded if there is at least one component of x converging to zero such that the corresponding component of z does not converge to zero. Note that K_2 is symmetric and quasi definite if $\delta > 0$.

Other formulations are possible. In particular, it has historically been common to reduce (5) further, for instance in the case of linear programming or when H is diagonal, to

$$\big(J(H + \rho I + X^{-1}Z)^{-1} J^T + \delta I\big)\, \Delta y = (b - Jx) - J(H + \rho I + X^{-1}Z)^{-1}(J^T y - c - Hx + \mu X^{-1} e), \tag{6}$$

which involves the primal Schur complement $K_1 := J(H + \rho I + X^{-1}Z)^{-1} J^T + \delta I$. A potential disadvantage, even in the case of linear programming, is that K_1 may be unacceptably dense. There is of course the possibility of using the dual Schur complement when $\delta > 0$, i.e.,

$$\big(H + \rho I + X^{-1}Z + \delta^{-1} J^T J\big)\, \Delta x = J^T y - c - Hx + \mu X^{-1} e + \delta^{-1} J^T (b - Jx).$$

The latter is seldom used in practice, perhaps because it is larger than (6).

Greif et al. (2014) study eigenvalues bounds of K_3, $K_{3.5}$, K_2 and K_1, and conclude that the condition number of K_3 and $K_{3.5}$ remains uniformly bounded, even as convergence occurs, provided strict complementarity is satisfied, while that of K_2 typically diverges.

In practice, problems are not always in the standard form (1) but may possess linear inequality constraints, not all variables are subject to bounds, and some bounds may be two sided. All problems have been transformed to

$$\begin{aligned} \underset{x \in \mathbb{R}^n, r \in \mathbb{R}^m, s \in \mathbb{R}^p}{\text{minimize}} \quad & c^T x + \tfrac{1}{2} x^T H x + \tfrac{1}{2}\rho \|x - x_k\|^2 + \tfrac{1}{2}\rho \|s - s_k\|^2 + \tfrac{1}{2}\delta \|r + y_k\|^2 \\ \text{subject to} \quad & J_1 x + J_2 s + \delta r = b, \;\; s \geq 0, \end{aligned}$$

where s are slack variables, and whose bounds will be treated by the logarithmic barrier. Thus instead of, (3), (4), and (5), the corresponding systems have the following form. The unsymmetric K_3 system becomes

$$\begin{bmatrix} H+\rho I & & J_1^T & \\ & \rho I & J_2^T & -I \\ J_1 & J_2 & -\delta I & \\ & Z & & S \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s \\ -\Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} J_1^T y - c - Hx \\ J_2^T y + z \\ b - J_1 x - J_2 s \\ \mu e - Sz \end{bmatrix},$$

where $S := \operatorname{diag}(s)$. One way to view the difference between the latter system and (3) is that the Z block has become rectangular and now has the form $\begin{bmatrix} 0 & Z \end{bmatrix}$.

The symmetrized $K_{3.5}$ system (4) becomes

$$\begin{bmatrix} H+\rho I & & J_1^T & \\ & \rho I & J_2^T & -Z^{\frac{1}{2}} \\ J_1 & J_2 & -\delta I & \\ & -Z^{\frac{1}{2}} & & -S \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s \\ -\Delta y \\ Z^{-\frac{1}{2}}\Delta z \end{bmatrix} = \begin{bmatrix} J_1^T y - c - Hx \\ J_2^T y + z \\ b - J_1 x - J_2 s \\ Z^{-\frac{1}{2}}(\mu e - Sz) \end{bmatrix}.$$

The K_2 system becomes

$$\begin{bmatrix} H+\rho I & & J_1^T \\ & S^{-1}Z+\rho I & J_2^T \\ J_1 & J_2 & -\delta I \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s \\ -\Delta y \end{bmatrix} = \begin{bmatrix} J_1^T y - c - Hx \\ J_2^T y + \mu S^{-1} e \\ b - J_1 x - J_2 s \end{bmatrix}.$$

3 Systems in the collection

3.1 Description of the systems

The systems in the collection are generated during an interior-point method for (1) based on $K_{3.5}$ where (4) is solved with a direct method. All quadratic optimization problems come from the CUTEst collection (Gould et al., 2015). The systems are the same as those used by Orban (2015a) to demonstrate the effectiveness of a limited-memory LDL^T factorization, except that the blocks $H+\rho I$, J, Z and X are provided separately to facilitate the assembly of the various system formulations. Systems are generated at interior-point iterations 0, 5 and 10. The differences between systems are various iterations reside in the values of x, z, ρ, δ and μ. At iteration 0, $\rho = \delta = 1$, and (x, s, y, z) is determined using the initialization procedure of Mehrotra (1992). At iteration 5, $\rho = \delta = 10^{-5}$, while at iteration 10, $\rho = \delta = 10^{-8}$. Not all problems required 10 or even 5 iterations to solve using a direct method. Thus at the interior-point iteration increases, the number of systems diminishes.

Table 1 documents the system size for each problem and each formulation together with the overall system density at interior-point iteration 0. Because Matlab has no concept of a symmetric matrix, both triangles of K_2 and $K_{3.5}$ are assembled, and the density is always computed as the overall number of nonzeros over N^2, where N generically denotes the system size. When $\rho > 0$ and $\delta > 0$, there are no zeros on the diagonal of any system. Table 2 and Table 3 document the corresponding data for systems at interior-point iterations 5 and 10. The collection contains 76 systems at iteration 0, 58 systems at iteration 5 and 41 systems at iteration 10, a total of 175 systems, each of which can be formulated according to (3), (4), (5), (6), or any other appropriate formulation.

3.2 Matlab interface

A few Matlab functions and a script are available as part of the problem collection to conveniently cycle through the problems and assemble the desired formulation. All matrices are stored in MatrixMarket format[1].

[1] http://math.nist.gov/MatrixMarket

Thus a prerequisite is to obtain the `mmread.m` script to read MatrixMarket-formatted matrices. Each system is stored in a separate folder of the form `data/`*`problem`*`/3x3/iter_`*`k`*, where *`problem`* is the problem name and *`k`* is the interior-point iteration number. When changing to a folder of this form, the constituent blocks may be obtained by calling

```
[rho, delta, H, J, Z, X, rhs] = read_blocks(k)
```

which returns the values of ρ, δ, as well as

`H`: the lower triangle of
$$\begin{bmatrix} H + \rho I & \\ & \rho I \end{bmatrix},$$

`J`: the matrix $\begin{bmatrix} J_1 & J_2 \end{bmatrix}$,

`Z`: the matrix $\begin{bmatrix} 0 & Z^{\frac{1}{2}} \end{bmatrix}$,

`X`: the diagonal matrix S,

`rhs`: the right-hand side in the formulation (4).

It is important to note that the term ρI is already included in `H` and that `Z` and `rhs` correspond to the $K_{3.5}$ formulation. Thus the data requires further transformation if formulations other than $K_{3.5}$ are to be used. Several Matlab functions are provided to assemble each system:

```
[K, P, nz, rhs] = assembleK35(itnumber, varargin)
[K, P, nz, rhs] = assembleK3 (itnumber, varargin)
[K, P, nz, rhs] = assembleK2 (itnumber, varargin)
```

Each function returns the system requested in `K` and `rhs`. In addition, the assembly function can be made to return a preconditioner in `P` and a measure of the number of nonzero elements in the preconditioner in `nz`. By default, a sparse identity matrix is returned. Implementing user-defined preconditioners is described below.

The convenience function

```
[K, P, nz, rhs] = getK(assembler, problem, itnumber, varargin)
```

performs all of the above without the need to manually change folders, where `assembler` is the handle of a system assembly function, `problem` is a problem name, `itnumber` is the desired interior-point iteration, and the remaining optional arguments are related to user-defined preconditioners. For example, one might issue the call

```
[K, P, nz, rhs] = getK(@assembleK3, 'cvxqp1_s', 5)
```

which returns the unsymmetric K_3 system in `K` and `rhs`, a sparse identity matrix in `P` and the system size in `nz`.

The script `cycle_through_problems` is an example script that cycles through the problem list and reads problems at a given interior-point iterations. The script can be easily modified to suit the user's needs and is provided as a template driver.

We end this section with an example user-defined preconditioner. Suppose we wish to define the block-diagonal preconditioner
$$P = \begin{bmatrix} \mathrm{diag}(H) + \rho I & & \\ & S^{-1}Z + \rho I & \\ & & \delta I \end{bmatrix}$$
for the K_2 system formulation. This may be achieved by defining the function

```
function [P, nz] = blkdiagK2(rho, delta, H, J, Z, X, varargin)

  [m, n] = size(J);
  P = blkdiag(diag(H) + Z' * (X \ Z), delta * speye(m));
  nz = m + n;
end
```

A call to `getK()` may now look like

```
[K, P, nz, rhs] = getK(@assembleK2, 'cvxqp1_s', 5, @blkdiagK2)
```

Should a user-defined preconditioner require additional arguments, they can be supplied to `getK()` in order and will be passed unchanged to the preconditioner-defining function.

4 Discussion

Standard matrix collections, though crucial in the development of solvers and preconditioners, do not always provide families of related systems or even accompanying right-hand sides (other than the vector of ones). They also do not allow users to experiment with various formulations of a system easily. The development of robust factorization-free interior-point methods is still in its infancy, though interior-point methods themselves today are reasonably well understood. They are often the method of choice in convex and nonconvex optimization, but at their heart lies the solution of a difficult and increasingly ill-conditioned linear system. The present collection is provided in hopes to encourage a wider interest in the development of such methods.

Appendix A System specifications

Table 1: System specifications at interior-point iteration 0

	K_3			$K_{3.5}$			K_2		
Name	size	dens(%)	cond	size	dens(%)	cond	size	dens(%)	cond
aug2d	30200	2.4e−02	1.1e+01	30200	2.4e−02	9.9e+00	30200	2.4e−02	9.9e+00
aug2dc	30200	2.4e−02	1.1e+01	30200	2.4e−02	1.1e+01	30200	2.4e−02	1.1e+01
aug2dcqp	90800	7.1e−03	2.9e+02	90800	7.1e−03	2.9e+02	70600	9.3e−03	1.5e+01
aug2dqp	90800	7.1e−03	2.9e+02	90800	7.1e−03	2.9e+02	70600	9.3e−03	1.5e+01
aug3d	4873	1.5e−01	2.3e+01	4873	1.5e−01	2.3e+01	4873	1.5e−01	2.3e+01
aug3dc	4873	1.5e−01	2.6e+01	4873	1.5e−01	2.7e+01	4873	1.5e−01	2.7e+01
aug3dcqp	16492	3.9e−02	3.9e+01	16492	3.9e−02	3.9e+01	12619	5.2e−02	3.9e+01
aug3dqp	16492	3.9e−02	4.0e+01	16492	3.9e−02	4.0e+01	12619	5.2e−02	3.9e+01
cvxqp1_s	750	1.0e+00	3.4e+03	750	1.0e+00	3.6e+03	550	1.5e+00	3.5e+03
cvxqp1_m	7500	1.0e−01	5.1e+04	7500	1.0e−01	5.1e+04	5500	1.5e−01	5.4e+04
cvxqp1_l	75000	1.0e−02	7.6e+05	75000	1.0e−02	7.3e+05	55000	1.5e−02	7.7e+05
cvxqp2_s	725	1.0e+00	3.4e+03	725	1.0e+00	3.4e+03	525	1.5e+00	3.5e+03
cvxqp2_m	7250	1.0e−01	5.6e+04	7250	1.0e−01	5.1e+04	5250	1.5e−01	5.2e+04
cvxqp2_l	72500	1.0e−02	7.8e+05	72500	1.0e−02	7.5e+05	52500	1.5e−02	7.8e+05
cvxqp3_s	775	9.9e−01	3.0e+03	775	9.9e−01	2.8e+03	575	1.4e+00	3.5e+03
cvxqp3_m	7750	1.0e−01	4.7e+04	7750	1.0e−01	4.5e+04	5750	1.5e−01	4.9e+04
cvxqp3_l	77500	1.0e−02	7.5e+05	77500	1.0e−02	6.6e+05	57500	1.5e−02	7.4e+05
dual1	596	4.9e+00	1.8e+04	596	4.9e+00	6.2e+03	426	9.0e+00	4.5e+03
dual2	673	4.8e+00	1.4e+04	673	4.8e+00	5.6e+03	481	8.9e+00	3.9e+03
dual3	778	4.7e+00	3.4e+03	778	4.7e+00	7.2e+03	556	8.8e+00	7.1e+03
dual4	526	5.1e+00	1.5e+04	526	5.1e+00	8.0e+03	376	9.3e+00	4.9e+03
dualc1	706	2.2e+00	2.0e+07	706	2.2e+00	1.4e+07	474	4.4e+00	1.3e+07
dualc2	734	1.8e+00	5.1e+06	734	1.8e+00	2.4e+06	492	3.5e+00	2.4e+06
dualc5	888	1.7e+00	2.1e+05	888	1.7e+00	2.2e+05	595	3.2e+00	2.9e+05

Continued on next page

Table 1 – continued from previous page

Name	K_3			$K_{3.5}$			K_2		
	size	dens(%)	cond	size	dens(%)	cond	size	dens(%)	cond
dualc8	1563	9.6e−01	3.5e+07	1563	9.6e−01	3.4e+07	1045	1.9e+00	3.0e+07
genhs28	18	4.9e+01	4.2e+01	18	4.9e+01	4.2e+01	18	4.9e+01	4.2e+01
gouldqp2	5242	1.2e−01	2.7e+01	5242	1.2e−01	1.9e+01	3844	1.7e−01	2.3e+01
gouldqp3	5242	1.2e−01	8.1e+01	5242	1.2e−01	8.0e+01	3844	1.7e−01	3.1e+01
hs118	192	3.3e+00	3.2e+02	192	3.3e+00	3.2e+02	133	4.9e+00	2.8e+01
hs21	17	3.2e+01	1.1e+02	17	3.2e+01	1.1e+02	12	4.4e+01	8.0e+00
hs21mod	34	1.5e+01	9.1e+01	34	1.5e+01	1.0e+02	25	1.9e+01	7.9e+00
hs268	20	5.2e+01	1.6e+05	20	5.2e+01	1.3e+05	15	7.9e+01	1.5e+05
hs35	15	3.9e+01	2.2e+01	15	3.9e+01	2.4e+01	11	5.3e+01	1.5e+01
hs35mod	12	4.9e+01	4.3e+01	12	4.9e+01	4.6e+01	9	6.4e+01	3.0e+01
hs51	8	7.2e+01	4.1e+01	8	7.2e+01	4.1e+01	8	7.2e+01	3.9e+01
hs52	8	7.2e+01	5.0e+02	8	7.2e+01	5.0e+02	8	7.2e+01	4.6e+02
hs53	38	1.6e+01	4.3e+01	38	1.6e+01	4.2e+01	28	2.1e+01	2.9e+01
hs76	25	2.6e+01	1.3e+01	25	2.6e+01	1.4e+01	18	3.7e+01	1.0e+01
hues-mod	40002	1.7e−02	3.2e+05	40002	1.7e−02	3.2e+05	30002	2.4e−02	5.5e+04
huestis	40002	1.7e−02	3.1e+05	40002	1.7e−02	3.1e+05	30002	2.4e−02	5.4e+04
ksip	3023	1.0e+00	1.3e+04	3023	1.0e+00	1.1e+04	2022	2.1e+00	6.8e+03
liswet1	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet10	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet11	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet12	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet2	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet3	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet4	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet5	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet6	40002	1.7e−02	9.7e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet7	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet8	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
liswet9	40002	1.7e−02	9.8e+00	40002	1.7e−02	1.0e+01	30002	2.4e−02	9.7e+00
lotschd	55	1.5e+01	8.1e+01	55	1.5e+01	1.0e+02	43	2.1e+01	3.2e+01
mosarqp1	12100	5.0e−02	2.4e+01	12100	5.0e−02	2.2e+01	8900	6.9e−02	2.7e+01
mosarqp2	5400	1.3e−01	6.2e+01	5400	1.3e−01	6.3e+01	3900	1.9e−01	4.4e+01
powell20	40000	1.5e−02	1.0e+04	40000	1.5e−02	1.0e+04	30000	2.0e−02	6.6e+00
primal1	583	7.4e+00	1.3e+04	583	7.4e+00	6.3e+03	497	9.9e+00	4.1e+02
primal2	940	3.9e+00	1.9e+04	940	3.9e+00	8.0e+03	843	4.8e+00	4.6e+02
primal3	1081	7.6e+00	1.9e+04	1081	7.6e+00	8.4e+03	969	9.4e+00	9.8e+02
primal4	1717	2.3e+00	6.6e+02	1717	2.3e+00	1.6e+03	1641	2.5e+00	6.7e+02
primalc1	902	1.6e+00	3.3e+03	902	1.6e+00	3.3e+03	678	2.5e+00	3.5e+02
primalc2	939	1.3e+00	2.8e+02	939	1.3e+00	3.7e+02	703	2.0e+00	1.6e+02
primalc5	1145	1.1e+00	1.2e+04	1145	1.1e+00	1.2e+04	859	1.8e+00	1.0e+04
primalc8	2053	6.4e−01	3.7e+02	2053	6.4e−01	4.4e+02	1542	1.0e+00	9.2e+02
qpcblend	468	1.9e+00	6.1e+01	468	1.9e+00	6.2e+01	354	2.8e+00	5.6e+01
qpcboei1	3306	2.9e−01	1.8e+04	3306	2.9e−01	1.8e+04	2335	4.8e−01	6.0e+02
qpcboei2	1281	7.0e−01	4.6e+04	1281	7.0e−01	4.6e+04	903	1.1e+00	2.1e+02
qpcstair	2272	5.0e−01	4.4e+02	2272	5.0e−01	4.4e+02	1740	7.5e−01	4.4e+02
s268	20	5.2e+01	1.6e+05	20	5.2e+01	1.5e+05	15	7.9e+01	1.5e+05
stcqp1	30731	3.2e−02	3.5e+03	30731	3.2e−02	3.4e+03	22537	5.0e−02	3.7e+03
stcqp2	30731	3.2e−02	3.4e+03	30731	3.2e−02	3.5e+03	22537	5.0e−02	3.2e+03
tame	9	6.0e+01	1.0e+01	9	6.0e+01	7.9e+00	7	7.5e+01	1.1e+01
ubh1	66027	1.1e−02	1.2e+01	66027	1.1e−02	1.3e+01	54021	1.4e−02	1.1e+01
yao	8005	8.7e−02	3.7e+02	8005	8.7e−02	1.6e+02	6004	1.2e−01	9.8e+00
zecevic2	20	2.9e+01	2.9e+01	20	2.9e+01	2.9e+01	14	4.0e+01	1.5e+01

Table 2: System specifications at interior-point iteration 5

	K_3			$K_{3.5}$			K_2		
Name	size	dens(%)	cond	size	dens(%)	cond	size	dens(%)	cond
aug2d	30200	2.4e−02	4.0e+05	30200	2.4e−02	4.0e+05	30200	2.4e−02	4.0e+05
aug2dc	30200	2.4e−02	3.3e+03	30200	2.4e−02	3.3e+03	30200	2.4e−02	3.3e+03
aug2dcqp	90800	7.1e−03	3.3e+05	90800	7.1e−03	3.0e+04	70600	9.3e−03	1.5e+07
aug2dqp	90800	7.1e−03	4.5e+05	90800	7.1e−03	3.9e+04	70600	9.3e−03	2.9e+07
aug3dcqp	16492	3.9e−02	1.6e+03	16492	3.9e−02	1.0e+03	12619	5.2e−02	6.2e+04
aug3dqp	16492	3.9e−02	1.5e+04	16492	3.9e−02	1.6e+04	12619	5.2e−02	1.0e+06
cvxqp1_s	750	1.0e+00	4.5e+07	750	1.0e+00	4.4e+07	550	1.5e+00	4.7e+07
cvxqp1_m	7500	1.0e−01	3.6e+09	7500	1.0e−01	2.8e+09	5500	1.5e−01	3.5e+09
cvxqp1_l	75000	1.0e−02	8.2e+10	75000	1.0e−02	8.6e+10	55000	1.5e−02	8.2e+10
cvxqp2_s	725	1.0e+00	2.4e+06	725	1.0e+00	2.2e+06	525	1.5e+00	7.4e+06
cvxqp2_m	7250	1.0e−01	6.3e+08	7250	1.0e−01	6.1e+08	5250	1.5e−01	6.2e+08
cvxqp2_l	72500	1.0e−02	3.8e+10	72500	1.0e−02	3.4e+10	52500	1.5e−02	3.8e+10
cvxqp3_s	775	9.9e−01	1.5e+08	775	9.9e−01	1.7e+08	575	1.4e+00	4.2e+08
cvxqp3_m	7750	1.0e−01	3.8e+09	7750	1.0e−01	3.8e+09	5750	1.5e−01	3.8e+09
cvxqp3_l	77500	1.0e−02	9.4e+10	77500	1.0e−02	8.9e+10	57500	1.5e−02	8.9e+10
dual1	596	4.9e+00	3.3e+07	596	4.9e+00	1.1e+07	426	9.0e+00	9.2e+04
dual3	778	4.7e+00	5.1e+07	778	4.7e+00	2.3e+07	556	8.8e+00	6.5e+05
dual4	526	5.1e+00	1.1e+07	526	5.1e+00	5.6e+06	376	9.3e+00	2.9e+06
dualc1	706	2.2e+00	1.2e+12	706	2.2e+00	7.4e+11	474	4.4e+00	7.3e+11
dualc2	734	1.8e+00	3.4e+11	734	1.8e+00	1.7e+11	492	3.5e+00	7.8e+11
dualc5	888	1.7e+00	7.2e+09	888	1.7e+00	3.7e+09	595	3.2e+00	5.4e+09
dualc8	1563	9.6e−01	5.0e+11	1563	9.6e−01	4.6e+11	1045	1.9e+00	4.0e+11
gouldqp2	5242	1.2e−01	1.2e+05	5242	1.2e−01	1.7e+04	3844	1.7e−01	1.6e+05
gouldqp3	5242	1.2e−01	5.3e+04	5242	1.2e−01	5.2e+04	3844	1.7e−01	1.0e+07
hs118	192	3.3e+00	1.4e+05	192	3.3e+00	7.7e+04	133	4.9e+00	1.3e+04
hs21	17	3.2e+01	8.0e+03	17	3.2e+01	1.3e+03	12	4.4e+01	7.3e+01
hs21mod	34	1.5e+01	1.3e+04	34	1.5e+01	5.0e+03	25	1.9e+01	2.1e+08
hues-mod	40002	1.7e−02	1.8e+08	40002	1.7e−02	2.5e+08	30002	2.4e−02	7.0e+07
huestis	40002	1.7e−02	3.3e+05	40002	1.7e−02	2.2e+05	30002	2.4e−02	6.7e+04
ksip	3023	1.0e+00	3.7e+07	3023	1.0e+00	1.8e+07	2022	2.1e+00	2.8e+05
liswet1	40002	1.7e−02	7.5e+03	40002	1.7e−02	1.9e+03	30002	2.4e−02	5.3e+04
liswet10	40002	1.7e−02	7.4e+03	40002	1.7e−02	1.8e+03	30002	2.4e−02	5.2e+04
liswet11	40002	1.7e−02	7.4e+03	40002	1.7e−02	2.0e+03	30002	2.4e−02	5.2e+04
liswet12	40002	1.7e−02	7.4e+03	40002	1.7e−02	2.0e+03	30002	2.4e−02	4.6e+04
liswet2	40002	1.7e−02	7.5e+03	40002	1.7e−02	2.1e+03	30002	2.4e−02	5.2e+04
liswet3	40002	1.7e−02	7.6e+03	40002	1.7e−02	1.8e+03	30002	2.4e−02	5.2e+04
liswet4	40002	1.7e−02	7.5e+03	40002	1.7e−02	1.9e+03	30002	2.4e−02	5.2e+04
liswet5	40002	1.7e−02	7.5e+03	40002	1.7e−02	1.9e+03	30002	2.4e−02	5.2e+04
liswet6	40002	1.7e−02	7.4e+03	40002	1.7e−02	2.1e+03	30002	2.4e−02	5.2e+04
liswet7	40002	1.7e−02	7.5e+03	40002	1.7e−02	1.9e+03	30002	2.4e−02	4.6e+04
liswet8	40002	1.7e−02	7.5e+03	40002	1.7e−02	2.2e+03	30002	2.4e−02	5.2e+04
liswet9	40002	1.7e−02	8.0e+03	40002	1.7e−02	2.2e+03	30002	2.4e−02	5.2e+04
lotschd	55	1.5e+01	4.3e+03	55	1.5e+01	1.6e+03	43	2.1e+01	4.8e+05
mosarqp1	12100	5.0e−02	6.0e+04	12100	5.0e−02	3.4e+04	8900	6.9e−02	2.7e+08
mosarqp2	5400	1.3e−01	5.2e+03	5400	1.3e−01	3.3e+03	3900	1.9e−01	7.2e+05
powell20	40000	1.5e−02	2.3e+05	40000	1.5e−02	2.5e+04	30000	2.0e−02	4.0e+02
primal4	1717	2.3e+00	1.1e+07	1717	2.3e+00	4.5e+05	1641	2.5e+00	3.7e+05
primalc1	902	1.6e+00	7.9e+07	902	1.6e+00	7.9e+07	678	2.5e+00	2.1e+06
primalc2	939	1.3e+00	3.4e+08	939	1.3e+00	3.4e+08	703	2.0e+00	1.0e+08
primalc5	1145	1.1e+00	5.5e+08	1145	1.1e+00	3.6e+08	859	1.8e+00	9.7e+09
primalc8	2053	6.4e−01	1.4e+08	2053	6.4e−01	1.1e+08	1542	1.0e+00	7.6e+09
qpcblend	468	1.9e+00	3.7e+04	468	1.9e+00	3.1e+04	354	2.8e+00	1.3e+06
qpcboei1	3306	2.9e−01	4.1e+05	3306	2.9e−01	3.6e+05	2335	4.8e−01	1.0e+07
qpcboei2	1281	7.0e−01	2.5e+05	1281	7.0e−01	2.5e+05	903	1.1e+00	2.7e+03
qpcstair	2272	5.0e−01	3.5e+06	2272	5.0e−01	4.1e+05	1740	7.5e−01	1.0e+08
stcqp1	30731	3.2e−02	8.7e+08	30731	3.2e−02	8.7e+08	22537	5.0e−02	8.7e+08
stcqp2	30731	3.2e−02	1.8e+06	30731	3.2e−02	1.8e+06	22537	5.0e−02	5.1e+07
yao	8005	8.7e−02	6.9e+07	8005	8.7e−02	3.4e+07	6004	1.2e−01	2.5e+11

Table 3: System specifications at interior-point iteration 10

	K_3			$K_{3.5}$			K_2		
Name	size	dens(%)	cond	size	dens(%)	cond	size	dens(%)	cond
aug2dcqp	90800	7.1e−03	4.2e+06	90800	7.1e−03	1.2e+05	70600	9.3e−03	1.3e+08
aug2dqp	90800	7.1e−03	9.1e+06	90800	7.1e−03	1.9e+05	70600	9.3e−03	1.9e+09
cvxqp1_s	750	1.0e+00	7.6e+10	750	1.0e+00	7.3e+10	550	1.5e+00	1.9e+16
cvxqp1_m	7500	1.0e−01	3.8e+11	7500	1.0e−01	3.8e+11	5500	1.5e−01	7.2e+11
cvxqp1_l	75000	1.0e−02	4.6e+13	75000	1.0e−02	4.6e+13	55000	1.5e−02	4.6e+13
cvxqp2_s	725	1.0e+00	2.3e+07	725	1.0e+00	1.6e+07	525	1.5e+00	5.4e+09
cvxqp2_m	7250	1.0e−01	1.1e+09	7250	1.0e−01	1.0e+09	5250	1.5e−01	1.1e+09
cvxqp2_l	72500	1.0e−02	3.6e+11	72500	1.0e−02	3.6e+11	52500	1.5e−02	3.6e+11
cvxqp3_s	775	9.9e−01	6.0e+09	775	9.9e−01	5.8e+09	575	1.4e+00	2.5e+11
cvxqp3_m	7750	1.0e−01	2.2e+12	7750	1.0e−01	2.5e+12	5750	1.5e−01	9.0e+13
cvxqp3_l	77500	1.0e−02	5.1e+13	77500	1.0e−02	5.1e+13	57500	1.5e−02	5.1e+13
dualc1	706	2.2e+00	1.3e+14	706	2.2e+00	6.5e+13	474	4.4e+00	2.2e+14
dualc2	734	1.8e+00	1.4e+12	734	1.8e+00	7.0e+11	492	3.5e+00	1.1e+12
dualc8	1563	9.6e−01	5.0e+13	1563	9.6e−01	3.3e+13	1045	1.9e+00	3.2e+13
hs118	192	3.3e+00	8.6e+03	192	3.3e+00	8.3e+03	133	4.9e+00	2.4e+03
hues-mod	40002	1.7e−02	2.2e+08	40002	1.7e−02	3.1e+08	30002	2.4e−02	7.8e+07
huestis	40002	1.7e−02	3.3e+05	40002	1.7e−02	2.6e+05	30002	2.4e−02	2.2e+05
ksip	3023	1.0e+00	1.8e+09	3023	1.0e+00	6.9e+08	2022	2.1e+00	3.8e+09
liswet1	40002	1.7e−02	4.3e+05	40002	1.7e−02	1.8e+05	30002	2.4e−02	3.8e+08
liswet10	40002	1.7e−02	5.7e+05	40002	1.7e−02	1.9e+05	30002	2.4e−02	3.5e+08
liswet11	40002	1.7e−02	6.4e+05	40002	1.7e−02	2.4e+05	30002	2.4e−02	4.5e+08
liswet12	40002	1.7e−02	6.1e+05	40002	1.7e−02	2.3e+05	30002	2.4e−02	4.4e+08
liswet2	40002	1.7e−02	7.4e+05	40002	1.7e−02	2.6e+05	30002	2.4e−02	5.0e+08
liswet3	40002	1.7e−02	7.4e+05	40002	1.7e−02	2.6e+05	30002	2.4e−02	5.0e+08
liswet4	40002	1.7e−02	7.4e+05	40002	1.7e−02	2.6e+05	30002	2.4e−02	5.0e+08
liswet5	40002	1.7e−02	4.7e+05	40002	1.7e−02	1.9e+05	30002	2.4e−02	3.9e+08
liswet6	40002	1.7e−02	8.1e+05	40002	1.7e−02	2.8e+05	30002	2.4e−02	5.4e+08
liswet7	40002	1.7e−02	6.6e+05	40002	1.7e−02	2.2e+05	30002	2.4e−02	4.1e+08
liswet8	40002	1.7e−02	6.6e+05	40002	1.7e−02	2.2e+05	30002	2.4e−02	4.2e+08
liswet9	40002	1.7e−02	6.6e+05	40002	1.7e−02	2.2e+05	30002	2.4e−02	4.1e+08
powell20	40000	1.5e−02	9.4e+07	40000	1.5e−02	2.1e+05	30000	2.0e−02	4.0e+06
primal4	1717	2.3e+00	2.1e+05	1717	2.3e+00	1.0e+05	1641	2.5e+00	1.1e+05
primalc1	902	1.6e+00	4.3e+08	902	1.6e+00	2.1e+08	678	2.5e+00	1.7e+10
primalc2	939	1.3e+00	1.9e+08	939	1.3e+00	4.2e+06	703	2.0e+00	8.8e+10
primalc5	1145	1.1e+00	6.0e+08	1145	1.1e+00	3.2e+08	859	1.8e+00	3.8e+13
primalc8	2053	6.4e−01	1.5e+10	2053	6.4e−01	8.3e+09	1542	1.0e+00	6.8e+12
qpcblend	468	1.9e+00	1.7e+07	468	1.9e+00	1.7e+07	354	2.8e+00	1.9e+11
qpcboei1	3306	2.9e−01	3.9e+05	3306	2.9e−01	2.3e+05	2335	4.8e−01	5.9e+05
qpcboei2	1281	7.0e−01	3.6e+05	1281	7.0e−01	3.4e+05	903	1.1e+00	1.1e+05
qpcstair	2272	5.0e−01	1.8e+07	2272	5.0e−01	5.7e+05	1740	7.5e−01	3.8e+08
stcqp1	30731	3.2e−02	1.1e+12	30731	3.2e−02	1.1e+12	22537	5.0e−02	9.9e+12

References

M.P. Friedlander and D. Orban. A primal-dual regularized interior-point method for convex quadratic programs. Mathematical Programming Computation, 4(1):71–107, 2012. DOI: 10.1007/s12532-012-0035-2.

N.I.M. Gould, D. Orban, and Ph.L. Toint. CUTEst: a Constrained and Unconstrained Testing Environment with safe threads for Mathematical Optimization. Computational Optimization and Applications, 60:545–557, 2015. DOI: 10.1007/s10589-014-9687-3.

C. Greif, E. Moulding, and D. Orban. Bounds on the eigenvalues of matrices arising from interior-point methods. SIAM Journal on Optimization, 24(1):49–83, 2014. DOI: 10.1137/120890600.

S. Mehrotra. On the implementation of a primal-dual interior point method. SIAM Journal on Optimization, 2(4): 575–601, 1992. DOI: 10.1137/0802028.

D. Orban. Limited-memory LDL^T factorization of symmetric quasi-definite matrices with application to constrained optimization. Numerical Algorithms, 70(1):9–41, 2015a. DOI: 10.1007/s11075-014-9933-x.

D. Orban. A collection of linear systems arising from interior-point methods for quadratic optimization. Online data set. DOI: 10.5281/zenodo.34130, 2015b.

R. J. Vanderbei. Symmetric quasi-definite matrices. SIAM Journal on Optimization, 5(1):100–113, 1995. DOI: 10.1137/0805005.