A practical dual gradient-projection method for large-scale, strictly-convex quadratic programming

Nick Gould
STFC Rutherford Appleton Laboratory

with

Jonathan Hogg & Jennifer Scott

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2} x^T H x + g^T x \quad \text{subject to} \quad Ax \geq b$$

Manchester-NAG-RAL Workshop
The University of Manchester, 23rd October 2013

Summary of the talk

- optimization at RAL
- convex quadratic programming
- bound-constrained QP via gradient projection
- duality
- dual gradient-projection methods for QP
- DQP
- conclusions

Optimization at RAL

- V chapter in Harwell Subroutine Library (later HSL)
  - VA01 in HSL, Feb. 1963 (Powell)
  - Apollo 11 uses Mike Powell's code in 1969
  - a new V routine each month in 1970 (Fletcher,Powell)

Optimization at RAL

- V chapter in Harwell Subroutine Library (later HSL)
  - VA01 in HSL, Feb. 1963 (Powell)
  - Apollo 11 uses Mike Powell's code in 1969
  - a new V routine each month in 1970 (Fletcher,Powell)
- separate L (linear programming) and N (nonlinear equations) chapters from late 1960s (Reid)

Optimization at RAL

- `V` chapter in Harwell Subroutine Library (later HSL)
  - `VA01` in HSL, Feb. 1963 (Powell)
  - Apollo 11 uses Mike Powell's code in 1969
  - a new `V` routine each month in 1970 (Fletcher,Powell)
- separate `L` (linear programming) and `N` (nonlinear equations) chapters from late 1960s (Reid)
- large-scale NLP solver LANCELOT in 1990 (Conn, G., Toint)
  - CUTE optimization testing environment in 1992 (Bongartz, Conn, G., Toint)

Optimization at RAL

- V chapter in Harwell Subroutine Library (later HSL)
  - VA01 in HSL, Feb. 1963 (Powell)
  - Apollo 11 uses Mike Powell's code in 1969
  - a new V routine each month in 1970 (Fletcher,Powell)
- separate L (linear programming) and N (nonlinear equations) chapters from late 1960s (Reid)
- large-scale NLP solver LANCELOT in 1990 (Conn, G., Toint)
  - CUTE optimization testing environment in 1992 (Bongartz, Conn, G., Toint)
- GALAHAD library in 2003 replaces V chapter (G., Orban, Toint)
  - CUTEr in 2004, CUTEst in 2013 (G., Orban, Toint)

GALAHAD (G., Orban, Toint and friends)

Fully integrated, tested and documented Fortran 2003 large-scale optimization library of packages (many with Matlab interfaces):

GALAHAD (G., Orban, Toint and friends)

Fully integrated, tested and documented Fortran 2003 large-scale optimization library of packages (many with Matlab interfaces):

- general nonlinear optimization
  - LANCELOT B (augmented Lagrangian)
  - FASTR (SQP filter) and FUNNEL (SQP funnel), both forthcoming

GALAHAD (G., Orban, Toint and friends)

Fully integrated, tested and documented Fortran 2003 large-scale optimization library of packages (many with Matlab interfaces):

- general nonlinear optimization
  - LANCELOT B (augmented Lagrangian)
  - FASTR (SQP filter) and FUNNEL (SQP funnel), both forthcoming
- unconstrained optimization
  - TRU (trust-region) and ARC (cubic regularization)

GALAHAD (G., Orban, Toint and friends)

Fully integrated, tested and documented Fortran 2003 large-scale optimization library of packages (many with Matlab interfaces):

- general nonlinear optimization
  - LANCELOT B (augmented Lagrangian)
  - FASTR (SQP filter) and FUNNEL (SQP funnel), both forthcoming
- unconstrained optimization
  - TRU (trust-region) and ARC (cubic regularization)
- quadratic programming
  - QPA (nonconvex active set) and QPB (nonconvex interior point)
  - CQP (convex path-following) and DQP (convex gradient projection)

GALAHAD (G., Orban, Toint and friends)

Fully integrated, tested and documented Fortran 2003 large-scale optimization library of packages (many with Matlab interfaces):

- general nonlinear optimization
  - LANCELOT B (augmented Lagrangian)
  - FASTR (SQP filter) and FUNNEL (SQP funnel), both forthcoming
- unconstrained optimization
  - TRU (trust-region) and ARC (cubic regularization)
- quadratic programming
  - QPA (nonconvex active set) and QPB (nonconvex interior point)
  - CQP (convex path-following) and DQP (convex gradient projection)
- regularization subproblems
  - GLTR (trust-region) and GLRT (polynomial regularization)

GALAHAD (G., Orban, Toint and friends)

Fully integrated, tested and documented Fortran 2003 large-scale optimization library of packages (many with Matlab interfaces):

- general nonlinear optimization
  - LANCELOT B (augmented Lagrangian)
  - FASTR (SQP filter) and FUNNEL (SQP funnel), both forthcoming
- unconstrained optimization
  - TRU (trust-region) and ARC (cubic regularization)
- quadratic programming
  - QPA (nonconvex active set) and QPB (nonconvex interior point)
  - CQP (convex path-following) and DQP (convex gradient projection)
- regularization subproblems
  - GLTR (trust-region) and GLRT (polynomial regularization)
- generic linear system interfaces (via HSL and others)
  - SLS (symmetric), ULS (unsymmetric) and SBLS(saddle-point)

GALAHAD (G., Orban, Toint and friends)

Fully integrated, tested and documented Fortran 2003 large-scale optimization library of packages (many with Matlab interfaces):

- general nonlinear optimization
  - LANCELOT B (augmented Lagrangian)
  - FASTR (SQP filter) and FUNNEL (SQP funnel), both forthcoming
- unconstrained optimization
  - TRU (trust-region) and ARC (cubic regularization)
- quadratic programming
  - QPA (nonconvex active set) and QPB (nonconvex interior point)
  - CQP (convex path-following) and DQP (convex gradient projection)
- regularization subproblems
  - GLTR (trust-region) and GLRT (polynomial regularization)
- generic linear system interfaces (via HSL and others)
  - SLS (symmetric), ULS (unsymmetric) and SBLS(saddle-point)
- various support (root finding, scaling, presolving, preconditioning . . .)

Linear algebraic needs

Sparse linear equations $Ax = b$

- usually symmetric, often indefinite, sometimes block structured
- good preconditioners for generic and PDE-constrained problems
- singular systems and Fredholm alternatives
- sparse right-hand sides
- bandwidth reduction

Linear algebraic needs

Sparse linear equations $Ax = b$

- usually symmetric, often indefinite, sometimes block structured
- good preconditioners for generic and PDE-constrained problems
- singular systems and Fredholm alternatives
- sparse right-hand sides
- bandwidth reduction

Factorization updates rows and columns of A changed/appended

- explicit
- implicit (via Schur complement)

Linear algebraic needs

Sparse linear equations $Ax = b$

- usually symmetric, often indefinite, sometimes block structured
- good preconditioners for generic and PDE-constrained problems
- singular systems and Fredholm alternatives
- sparse right-hand sides
- bandwidth reduction

Factorization updates rows and columns of A changed/appended

- explicit
- implicit (via Schur complement)

Minimum-norm solution of over/under determined linear systems

- $\min \|Ax - b\|$ in ℓ_1, ℓ_2 and ℓ_∞-norms

Linear algebraic needs

Sparse linear equations $Ax = b$

- usually symmetric, often indefinite, sometimes block structured
- good preconditioners for generic and PDE-constrained problems
- singular systems and Fredholm alternatives
- sparse right-hand sides
- bandwidth reduction

Factorization updates rows and columns of A changed/appended

- explicit
- implicit (via Schur complement)

Minimum-norm solution of over/under determined linear systems

- $\min \|Ax - b\|$ in ℓ_1, ℓ_2 and ℓ_∞-norms

Regularization $(A + \lambda B)x(\lambda) = b$, with scalar “secular” equation $p(\lambda) = 0$ and definite B

Linear algebraic needs

Sparse linear equations $Ax = b$

- usually symmetric, often indefinite, sometimes block structured
- good preconditioners for generic and PDE-constrained problems
- singular systems and Fredholm alternatives
- sparse right-hand sides
- bandwidth reduction

Factorization updates rows and columns of A changed/appended

- explicit
- implicit (via Schur complement)

Minimum-norm solution of over/under determined linear systems

- $\min \|Ax - b\|$ in ℓ_1, ℓ_2 and ℓ_∞-norms

Regularization $(A + \lambda B)x(\lambda) = b$, with scalar "secular" equation $p(\lambda) = 0$ and definite B

Traditionally strong RAL group interactions on subjects such as these

Today's problem: quadratic programming

$$\textbf{QP:} \quad \underset{x \in \mathbb{R}^n}{\text{minimize}} \; q(x) = \tfrac{1}{2} x^T H x + g^T x \;\; \text{subject to} \;\; Ax \geq b$$

- assume that H positive definite $\Longrightarrow$ QP strictly convex

Today's problem: quadratic programming

QP: $\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; q(x) = \frac{1}{2} x^T H x + g^T x \;\; \text{subject to} \;\; Ax \geq b$

- assume that H positive definite $\Longrightarrow$ QP strictly convex
- aim to satisfy (KKT) **criticality conditions**

$$\begin{aligned} Ax_* &\geq b && \text{(primal feasibility)} \\ g + Hx_* - A^T y_* = 0 \;\&\; y_* &\geq 0 && \text{(dual feasibility)} \\ (Ax_* - b) \cdot y_* &= 0 && \text{(complementary slackness)} \end{aligned}$$

or to deduce that the problem is infeasible

Today's problem: quadratic programming

$$\textbf{QP:} \quad \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad q(x) = \tfrac{1}{2} x^T H x + g^T x \quad \text{subject to} \quad Ax \geq b$$

- assume that H positive definite $\Longrightarrow$ QP strictly convex
- aim to satisfy (KKT) **criticality conditions**

$$\begin{aligned} Ax_* &\geq b && \text{(primal feasibility)} \\ g + Hx_* - A^T y_* = 0 \;\&\; y_* &\geq 0 && \text{(dual feasibility)} \\ (Ax_* - b) \cdot y_* &= 0 && \text{(complementary slackness)} \end{aligned}$$

  or to deduce that the problem is infeasible
- interested in case where n is large and H and $A \in \Re^{m \times n}$ are sparse

Today's problem: quadratic programming

QP: $\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad q(x) = \frac{1}{2}x^T H x + g^T x \quad \text{subject to} \quad Ax \geq b$

- assume that H positive definite $\Longrightarrow$ QP strictly convex
- aim to satisfy (KKT) **criticality conditions**

$$
\begin{array}{cl}
Ax_* \geq b & \text{(primal feasibility)} \\
g + Hx_* - A^T y_* = 0 \ \& \ y_* \geq 0 & \text{(dual feasibility)} \\
(Ax_* - b) \cdot y_* = 0 & \text{(complementary slackness)}
\end{array}
$$

  or to deduce that the problem is infeasible

- interested in case where n is large and H and $A \in \Re^{m \times n}$ are sparse
- easy extension to more general constraint structures (equations, upper and both-sided bounds, simple bounds, ...)

Today's problem: quadratic programming

QP: $\underset{x \in \mathbb{R}^n}{\text{minimize}} \ \ q(x) = \frac{1}{2} x^T H x + g^T x \ \ \text{subject to} \ \ Ax \geq b$

- assume that H positive definite $\Longrightarrow$ QP strictly convex
- aim to satisfy (KKT) **criticality conditions**

$$\begin{aligned} Ax_* &\geq b && \text{(primal feasibility)} \\ g + Hx_* - A^T y_* = 0 \ \& \ y_* &\geq 0 && \text{(dual feasibility)} \\ (Ax_* - b) \cdot y_* &= 0 && \text{(complementary slackness)} \end{aligned}$$

or to deduce that the problem is infeasible

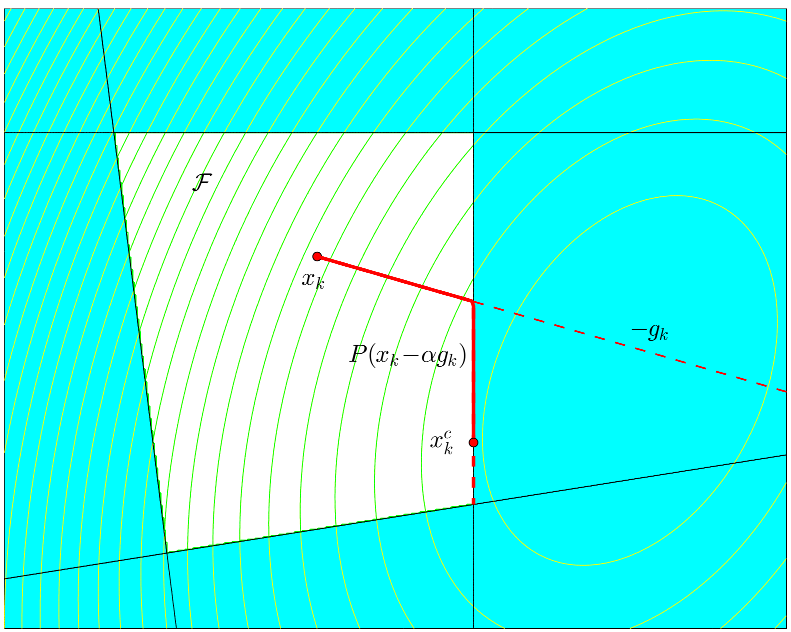
- interested in case where n is large and H and $A \in \Re^{m \times n}$ are sparse
- easy extension to more general constraint structures (equations, upper and both-sided bounds, simple bounds, ...)
- many real-world applications as well as SQP

Competing methods

- interior-point methods
  - usually very efficient
  - relatively poor at warm starting
- active-set methods
  - worst-case combinatorics due to pedestrian active-set changes
  - good at warm starting
- gradient projection methods
  - more rapid active-set changes
  - restricted to constraint sets for which projection is “easy”

Digression I: gradient projection

- convergence and active-set determination driven by projection
  - current iterate $x_k \in \mathcal{F} = \{x : Ax \geq b\}$
  - current gradient $g_k = Hx_k + g$
  - improved **Cauchy point** $x_k^c = P[x_k - \alpha_k g_k]$
  - projection $P[y] = \arg\min_{x \in \mathcal{F}} \|y - x\|$
  - step length $\alpha_k \approx \arg\min q(P[x_k - \alpha g_k])$ (Rosen,1960)



Accelerated gradient projection

- acceleration by subspace minimization
  - pick active set as subset of constraints $\mathcal{A}_k$ active at x_k^c
  - find (approximate) solution s_k to **equality constrained** QP

$$\textbf{EQP:} \quad \underset{s \in \mathbb{R}^n}{\text{minimize}} \;\; q(x_k^c + s) \;\; \text{subject to} \;\; A_{\mathcal{A}_k} s = 0$$

  - set $x_{k+1} \approx \arg\min q(P[x_k^c + \alpha s_k])$

Science & Technology Facilities Council

Accelerated gradient projection

- acceleration by subspace minimization
  - pick active set as subset of constraints $\mathcal{A}_k$ active at x_k^c
  - find (approximate) solution s_k to **equality constrained** QP

$$\textbf{EQP:} \quad \underset{s \in \mathbb{R}^n}{\text{minimize}} \quad q(x_k^c + s) \quad \text{subject to} \quad A_{\mathcal{A}_k} s = 0$$

  - set $x_{k+1} \approx \arg\min q(P[x_k^c + \alpha s_k])$
- solve EQP by
  - direct factorization (HSL, PARDISO, WSMP,...)

$$\begin{pmatrix} H & A_k^T \\ A_k & 0 \end{pmatrix} \begin{pmatrix} s_k \\ w_k \end{pmatrix} = - \begin{pmatrix} H x_k^c + g \\ 0 \end{pmatrix}$$

Accelerated gradient projection

- acceleration by subspace minimization
  - pick active set as subset of constraints $\mathcal{A}_k$ active at x_k^c
  - find (approximate) solution s_k to **equality constrained** QP

$$\textbf{EQP:} \quad \underset{s \in \mathbb{R}^n}{\text{minimize}} \; q(x_k^c + s) \;\; \text{subject to} \;\; A_{\mathcal{A}_k} s = 0$$

  - set $x_{k+1} \approx \arg\min q(P[x_k^c + \alpha s_k])$
- solve EQP by
  - direct factorization (HSL, PARDISO, WSMP,...)

$$\begin{pmatrix} H & A_k^T \\ A_k & 0 \end{pmatrix} \begin{pmatrix} s_k \\ w_k \end{pmatrix} = - \begin{pmatrix} H x_k^c + g \\ 0 \end{pmatrix}$$

  - factorization-free projected CG (G., Hribar & Nocedal, Luksan & Vlcek,90s...)

Projected search within simple bounds $x^L \leq x \leq x^U$

Find $\alpha^+ \approx \arg\min q(P[x + \alpha s])$ for $\alpha \geq 0$ (Conn, G. & Toint,1988)

Projected search within simple bounds $x^{\mathrm{L}} \leq x \leq x^{\mathrm{U}}$

Find $\alpha^{+} \approx \arg\min q(P[x + \alpha s])$ for $\alpha \geq 0$ (Conn, G. & Toint,1988)

- $P[x + \alpha s]$ piecewise linear, ordered breakpoints $\{0, \alpha_1, \ldots, \alpha_m\}$
- $q(P[x + \alpha s])$ piecewise quadratic $q_i(\alpha)$ for $\alpha \in [\alpha_i, \alpha_{i+1}]$
- consider each $q_i(\alpha)$ in turn until first local minimizer found

Projected search within simple bounds $x^L \leq x \leq x^U$

Find $\alpha^+ \approx \arg\min q(P[x + \alpha s])$ for $\alpha \geq 0$ (Conn, G. & Toint,1988)

- $P[x + \alpha s]$ piecewise linear, ordered breakpoints $\{0, \alpha_1, \ldots, \alpha_m\}$
- $q(P[x + \alpha s])$ piecewise quadratic $q_i(\alpha)$ for $\alpha \in [\alpha_i, \alpha_{i+1}]$
- consider each $q_i(\alpha)$ in turn until first local minimizer found
- for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $x_i = P[x_i + \alpha_i s]$:
  - $q_i(\alpha) = q(x_i) + \Delta\alpha(g^T s_i + x_i^T H s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H s_i$, where nonzero components of s_i are those of s "not fixed" at x_i

Projected search within simple bounds $x^{\mathrm{L}} \leq x \leq x^{\mathrm{U}}$

Find $\alpha^{+} \approx \arg\min q(P[x + \alpha s])$ for $\alpha \geq 0$ (Conn, G. & Toint,1988)

- $P[x + \alpha s]$ piecewise linear, ordered breakpoints $\{0, \alpha_1, \ldots, \alpha_m\}$
- $q(P[x + \alpha s])$ piecewise quadratic $q_i(\alpha)$ for $\alpha \in [\alpha_i, \alpha_{i+1}]$
- consider each $q_i(\alpha)$ in turn until first local minimizer found
- for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $x_i = P[x_i + \alpha_i s]$:
    - $q_i(\alpha) = q(x_i) + \Delta\alpha(g^T s_i + x_i^T H s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H s_i$, where nonzero components of s_i are those of s "not fixed" at x_i
    - $s_i = s_{i-1} - \Delta s_i$, where nonzero components of **very sparse** Δs_i are those of s_i "just fixed" at x_i

Projected search within simple bounds $x^{\mathrm{L}} \leq x \leq x^{\mathrm{U}}$

Find $\alpha^+ \approx \arg\min q(P[x + \alpha s])$ for $\alpha \geq 0$ (Conn, G. & Toint,1988)

- $P[x + \alpha s]$ piecewise linear, ordered breakpoints $\{0, \alpha_1, \ldots, \alpha_m\}$
- $q(P[x + \alpha s])$ piecewise quadratic $q_i(\alpha)$ for $\alpha \in [\alpha_i, \alpha_{i+1}]$
- consider each $q_i(\alpha)$ in turn until first local minimizer found
- for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $x_i = P[x_i + \alpha_i s]$:
  - $q_i(\alpha) = q(x_i) + \Delta\alpha(g^T s_i + x_i^T H s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H s_i$, where nonzero components of s_i are those of s "not fixed" at x_i
  - $s_i = s_{i-1} - \Delta s_i$, where nonzero components of **very sparse** Δs_i are those of s_i "just fixed" at x_i $\Longrightarrow$
  - $H s_i = H s_{i-1} - H \Delta s_i$ involves **likely very sparse** $H \Delta s_i$ if H is sparse

Projected search within simple bounds $x^{\mathrm{L}} \leq x \leq x^{\mathrm{U}}$

Find $\alpha^+ \approx \arg\min q(P[x + \alpha s])$ for $\alpha \geq 0$ (Conn, G. & Toint,1988)

- $P[x + \alpha s]$ piecewise linear, ordered breakpoints $\{0, \alpha_1, \ldots, \alpha_m\}$
- $q(P[x + \alpha s])$ piecewise quadratic $q_i(\alpha)$ for $\alpha \in [\alpha_i, \alpha_{i+1}]$
- consider each $q_i(\alpha)$ in turn until first local minimizer found
- for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $x_i = P[x_i + \alpha_i s]$:
  - $q_i(\alpha) = q(x_i) + \Delta\alpha(g^T s_i + x_i^T H s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H s_i$, where nonzero components of s_i are those of s "not fixed" at x_i
  - $s_i = s_{i-1} - \Delta s_i$, where nonzero components of **very sparse** Δs_i are those of s_i "just fixed" at x_i $\Longrightarrow$
  - $H s_i = H s_{i-1} - H \Delta s_i$ involves **likely very sparse** $H \Delta s_i$ if H is sparse $\Longrightarrow$
  - possible to recur required coefficients $g^T s_i$, $x_i^T H s_i$ and $s_i^T H s_i$ of $q_i(\alpha)$ very efficiently

Projected search within simple bounds $x^L \leq x \leq x^U$

Find $\alpha^+ \approx \arg\min q(P[x + \alpha s])$ for $\alpha \geq 0$ (Conn, G. & Toint,1988)

- $P[x + \alpha s]$ piecewise linear, ordered breakpoints $\{0, \alpha_1, \ldots, \alpha_m\}$
- $q(P[x + \alpha s])$ piecewise quadratic $q_i(\alpha)$ for $\alpha \in [\alpha_i, \alpha_{i+1}]$
- consider each $q_i(\alpha)$ in turn until first local minimizer found
- for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $x_i = P[x_i + \alpha_i s]$:
  - $q_i(\alpha) = q(x_i) + \Delta\alpha(g^T s_i + x_i^T H s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H s_i$, where nonzero components of s_i are those of s "not fixed" at x_i
  - $s_i = s_{i-1} - \Delta s_i$, where nonzero components of **very sparse** Δs_i are those of s_i "just fixed" at x_i $\Longrightarrow$
  - $H s_i = H s_{i-1} - H \Delta s_i$ involves **likely very sparse** $H \Delta s_i$ if H is sparse $\Longrightarrow$
  - possible to recur required coefficients $g^T s_i$, $x_i^T H s_i$ and $s_i^T H s_i$ of $q_i(\alpha)$ very efficiently
- approximate "Armijo"projected search also possible (Moré & Toraldo,Toint,90s)

Science & Technology Facilities Council

Anecdotal and empirical evidence

- large change possible in the active set per iteration
- often very effective in practice for convex bound-constrained QP
  - few overall iterations compared to active-set methods (Moré & Toraldo)
  - competitive with interior-point methods for such problems
- basis of LANCELOT (Conn, G. & Toint)
- generally impractical for general convex feasible regions as projection is too expensive
  - projection effectively requires the solution of a QP!

Anecdotal and empirical evidence

- large change possible in the active set per iteration
- often very effective in practice for convex bound-constrained QP
  - few overall iterations compared to active-set methods (Moré & Toraldo)
  - competitive with interior-point methods for such problems
- basis of LANCELOT (Conn, G. & Toint)
- generally impractical for general convex feasible regions as projection is too expensive
  - projection effectively requires the solution of a QP!

How might we apply such methods for QP over a general polyhedral feasible region?

Digression II: duality

$$\textbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2}x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\iff \quad \underset{x,s}{\text{minimize}} \; q(x) \text{ subject to } Ax - s = b \text{ and } s \geq 0$$

Digression II: duality

$$\textbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2} x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\Longleftrightarrow \quad \underset{x,s}{\text{minimize}} \; q(x) \text{ subject to } Ax - s = b \text{ and } s \geq 0 \Longrightarrow (\textbf{KKT})$$

$$\begin{pmatrix} Hx + g \\ 0 \end{pmatrix} - \begin{pmatrix} A^T \\ -I \end{pmatrix} y - \begin{pmatrix} 0 \\ I \end{pmatrix} z = 0, \;\; z \geq 0 \;\&\; s^T z = 0$$

Digression II: duality

$$\textbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2}x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\iff \underset{x,s}{\text{minimize}} \; q(x) \text{ subject to } Ax - s = b \text{ and } s \geq 0 \implies (\textbf{KKT})$$

$$\begin{pmatrix} Hx + g \\ 0 \end{pmatrix} - \begin{pmatrix} A^T \\ -I \end{pmatrix} y - \begin{pmatrix} 0 \\ I \end{pmatrix} z = 0, \; z \geq 0 \;\&\; s^T z = 0 \implies$$

$$g = A^T y - Hx, \; Ax = s + b, \; (s, y) \geq 0 \text{ and } s^T y = 0$$

Digression II: duality

$$\mathbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2}x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\Longleftrightarrow \quad \underset{x,s}{\text{minimize}} \; q(x) \text{ subject to } Ax - s = b \text{ and } s \geq 0 \Longrightarrow (\mathbf{KKT})$$

$$\begin{pmatrix} Hx + g \\ 0 \end{pmatrix} - \begin{pmatrix} A^T \\ -I \end{pmatrix} y - \begin{pmatrix} 0 \\ I \end{pmatrix} z = 0, \;\; z \geq 0 \;\&\; s^T z = 0 \Longrightarrow$$

$$g = A^T y - Hx, \;\; Ax = s + b, \;\; (s, y) \geq 0 \text{ and } s^T y = 0$$

Suppose $g = A^T y - Hx, \;\; Ax = s + b$ and $(s, y) \geq 0$

Digression II: duality

$$\textbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2}x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\Longleftrightarrow \quad \underset{x,s}{\text{minimize}} \; q(x) \text{ subject to } Ax - s = b \text{ and } s \geq 0 \Longrightarrow (\textbf{KKT})$$

$$\begin{pmatrix} Hx + g \\ 0 \end{pmatrix} - \begin{pmatrix} A^T \\ -I \end{pmatrix} y - \begin{pmatrix} 0 \\ I \end{pmatrix} z = 0, \; z \geq 0 \;\&\; s^T z = 0 \Longrightarrow$$

$$g = A^T y - Hx, \; Ax = s + b, \; (s, y) \geq 0 \text{ and } s^T y = 0$$

Suppose $g = A^T y - Hx, \; Ax = s + b$ and $(s, y) \geq 0 \Longrightarrow$

$$g^T x = y^T Ax - x^T Hx = y^T (s + b) - x^T Hx \geq y^T b - x^T Hx$$

Digression II: duality

$$\textbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2}x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\iff \quad \underset{x,s}{\text{minimize}}\ q(x) \text{ subject to } Ax - s = b \text{ and } s \geq 0 \implies (\textbf{KKT})$$

$$\begin{pmatrix} Hx + g \\ 0 \end{pmatrix} - \begin{pmatrix} A^T \\ -I \end{pmatrix} y - \begin{pmatrix} 0 \\ I \end{pmatrix} z = 0, \ z \geq 0 \ \& \ s^T z = 0 \implies$$

$$g = A^T y - Hx, \ Ax = s + b, \ (s, y) \geq 0 \text{ and } s^T y = 0$$

Suppose $g = A^T y - Hx, \ Ax = s + b$ and $(s, y) \geq 0 \implies$

$$g^T x = y^T Ax - x^T Hx = y^T(s + b) - x^T Hx \geq y^T b - x^T Hx$$

$$\implies q(x) \geq -\tfrac{1}{2}x^T Hx + b^T y$$

Digression II: duality

$$\textbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2}x^T Hx + g^T x \text{ subject to } Ax \geq b$$

$$\Longleftrightarrow \quad \underset{x,s}{\text{minimize}} \; q(x) \text{ subject to } Ax - s = b \text{ and } s \geq 0 \Longrightarrow (\textbf{KKT})$$

$$\begin{pmatrix} Hx + g \\ 0 \end{pmatrix} - \begin{pmatrix} A^T \\ -I \end{pmatrix} y - \begin{pmatrix} 0 \\ I \end{pmatrix} z = 0, \;\; z \geq 0 \;\&\; s^T z = 0 \Longrightarrow$$

$$g = A^T y - Hx, \;\; Ax = s + b, \;\; (s, y) \geq 0 \text{ and } s^T y = 0$$

Suppose $g = A^T y - Hx, \;\; Ax = s + b$ and $(s, y) \geq 0 \Longrightarrow$

$$g^T x = y^T Ax - x^T Hx = y^T (s + b) - x^T Hx \geq y^T b - x^T Hx$$

$\Longrightarrow \; q(x) \geq -\tfrac{1}{2}x^T Hx + b^T y \;\; \Longrightarrow$ equivalent **dual problem**

$$\textbf{DQP}: \quad \underset{x,y}{\text{maximize}} \; -\tfrac{1}{2}x^T Hx + b^T y \text{ s.t. } Hx - A^T y = -g \;\&\; y \geq 0$$

Duality II

$$\text{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2} x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\Longleftrightarrow$$

$$\text{DQP}: \quad \underset{x,y}{\text{maximize}} \; -\tfrac{1}{2} x^T H x + b^T y \text{ s.t. } Hx - A^T y = -g \;\&\; y \geq 0$$

Science & Technology Facilities Council

Duality II

$$\textbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2}x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\Longleftrightarrow$$

$$\textbf{DQP}: \quad \underset{x,y}{\text{maximize}} \; -\tfrac{1}{2}x^T H x + b^T y \text{ s.t. } Hx - A^T y = -g \;\&\; y \geq 0$$

$$\Longleftrightarrow$$

$$\textbf{DQP}: \quad \underset{x,y}{\text{minimize}} \; \tfrac{1}{2}x^T H x - b^T y \text{ s.t. } Hx - A^T y = -g \;\&\; y \geq 0$$

Duality II

$$\textbf{QP}: \quad \underset{x}{\text{minimize}} \quad q(x) = \tfrac{1}{2}x^T H x + g^T x \text{ subject to } Ax \geq b$$

$$\Longleftrightarrow$$

$$\textbf{DQP}: \quad \underset{x,y}{\text{maximize}} \; -\tfrac{1}{2}x^T H x + b^T y \text{ s.t. } Hx - A^T y = -g \;\&\; y \geq 0$$

$$\Longleftrightarrow$$

$$\textbf{DQP}: \quad \underset{x,y}{\text{minimize}} \; \tfrac{1}{2}x^T H x - b^T y \text{ s.t. } Hx - A^T y = -g \;\&\; y \geq 0$$

$\Longleftrightarrow$ (nonsingular H)

$$\textbf{DQP}: \quad \underset{y}{\text{minimize}} \; \tfrac{1}{2}(y^T A - g^T)H^{-1}(A^T y - g) - b^T y \text{ s.t. } y \geq 0$$

Dual gradient projection methods

$$\textbf{DQP}: \quad \underset{y}{\text{minimize}} \ \tfrac{1}{2}(y^T A - g^T) H^{-1} (A^T y - g) - b^T y \ \text{s.t.} \ y \geq 0$$

- for strictly-convex QP (i.e., H positive definite)
- dual objective $q_{\rm D}(y) = \frac{1}{2} y^T H_{\rm D} y + g_{\rm D}^T y$
  - $H_{\rm D} = A H^{-1} A^T$ and $g_{\rm D} = -A H^{-1} g - b$
- $H_{\rm D}$ may only be positive semi-definite
- since feasible region is simple, can use gradient projection to allow rapid changes in active set
- require **sparse factorization** $H = LL^T$ but everything else "matrix-free"

Dual gradient projection methods

DQP: $\underset{y}{\text{minimize}}\ \frac{1}{2}(y^T A - g^T) H^{-1} (A^T y - g) - b^T y$ s.t. $y \geq 0$

- for strictly-convex QP (i.e., H positive definite)
- dual objective $q_{\mathrm{D}}(y) = \frac{1}{2} y^T H_{\mathrm{D}} y + g_{\mathrm{D}}^T y$
  - $H_{\mathrm{D}} = A H^{-1} A^T$ and $g_{\mathrm{D}} = -A H^{-1} g - b$
- H_{D} may only be positive semi-definite
- since feasible region is simple, can use gradient projection to allow rapid changes in active set
- require **sparse factorization** $H = LL^T$ but everything else "matrix-free"

Questions:

- can we perform projected search efficiently?
- can we perform subspace minimization efficiently?

Dual projected search

- have $H_{\mathrm{D}} = AH^{-1}A^T$ and $g_{\mathrm{D}} = -AH^{-1}g - b$
- recall require $\alpha^+ \approx \arg\min q_{\mathrm{D}}(P[y + \alpha s])$ for $\alpha \geq 0$
- investigate for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $y_i = P[y_i + \alpha_i s]$:
  $q_i(\alpha) = q_{\mathrm{D}}(y_i) + \Delta\alpha(g_{\mathrm{D}}^T s_i + y_i^T H_{\mathrm{D}} s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H_{\mathrm{D}} s_i$
- recur via $H_{\mathrm{D}} s_i = H_{\mathrm{D}} s_{i-1} - H_{\mathrm{D}} \Delta s_i$ with very sparse Δs_i but likely **dense** H_{D} …looks expensive ☹

Science & Technology Facilities Council

Dual projected search

- have $H_{\mathrm{D}} = AH^{-1}A^T$ and $g_{\mathrm{D}} = -AH^{-1}g - b$
- recall require $\alpha^+ \approx \arg\min q_{\mathrm{D}}(P[y + \alpha s])$ for $\alpha \geq 0$
- investigate for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $y_i = P[y_i + \alpha_i s]$:
  $q_i(\alpha) = q_{\mathrm{D}}(y_i) + \Delta\alpha(g_{\mathrm{D}}^T s_i + y_i^T H_{\mathrm{D}} s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H_{\mathrm{D}} s_i$
- recur via $H_{\mathrm{D}} s_i = H_{\mathrm{D}} s_{i-1} - H_{\mathrm{D}} \Delta s_i$ with very sparse Δs_i but likely **dense** H_{D} . . . looks expensive ☹
- instead maintain $u_i = L^{-1}A^T s_i$ and $v_i = L^{-1}A^T y_i$

Dual projected search

- have $H_{\mathrm{D}} = AH^{-1}A^T$ and $g_{\mathrm{D}} = -AH^{-1}g - b$
- recall require $\alpha^+ \approx \arg\min q_{\mathrm{D}}(P[y + \alpha s])$ for $\alpha \geq 0$
- investigate for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $y_i = P[y_i + \alpha_i s]$:
  $q_i(\alpha) = q_{\mathrm{D}}(y_i) + \Delta\alpha(g_{\mathrm{D}}^T s_i + y_i^T H_{\mathrm{D}} s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H_{\mathrm{D}} s_i$
- recur via $H_{\mathrm{D}} s_i = H_{\mathrm{D}} s_{i-1} - H_{\mathrm{D}} \Delta s_i$ with very sparse Δs_i but likely **dense** H_{D} . . . looks expensive ☹
- instead maintain $u_i = L^{-1}A^T s_i$ and $v_i = L^{-1}A^T y_i \Longrightarrow$
- possible to recur required coefficients $g_{\mathrm{D}}^T s_i$ and $s_i^T H_{\mathrm{D}} s_i \equiv u_i^T u_i$ very efficiently via $u_i = u_{i-1} - \Delta u_i$, where $L\Delta u_i = A^T \Delta s_i$

Dual projected search

- have $H_{\rm D} = AH^{-1}A^T$ and $g_{\rm D} = -AH^{-1}g - b$
- recall require $\alpha^+ \approx \arg\min q_{\rm D}(P[y + \alpha s])$ for $\alpha \geq 0$
- investigate for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $y_i = P[y_i + \alpha_i s]$:
  $q_i(\alpha) = q_{\rm D}(y_i) + \Delta\alpha(g_{\rm D}^T s_i + y_i^T H_{\rm D} s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H_{\rm D} s_i$
- recur via $H_{\rm D} s_i = H_{\rm D} s_{i-1} - H_{\rm D} \Delta s_i$ with very sparse Δs_i but likely **dense** $H_{\rm D}$ ...looks expensive ☹
- instead maintain $u_i = L^{-1}A^T s_i$ and $v_i = L^{-1}A^T y_i \implies$
- possible to recur required coefficients $g_{\rm D}^T s_i$ and $s_i^T H_{\rm D} s_i \equiv u_i^T u_i$ very efficiently via $u_i = u_{i-1} - \Delta u_i$, where $L\Delta u_i = A^T \Delta s_i$
- as Δs_i, A and H are sparse, result of forward solve Δu_i is often **very sparse** ☺

Dual projected search

- have $H_{\text{D}} = AH^{-1}A^T$ and $g_{\text{D}} = -AH^{-1}g - b$
- recall require $\alpha^+ \approx \arg\min q_{\text{D}}(P[y + \alpha s])$ for $\alpha \geq 0$
- investigate for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $y_i = P[y_i + \alpha_i s]$:
  $q_i(\alpha) = q_{\text{D}}(y_i) + \Delta\alpha(g_{\text{D}}^T s_i + y_i^T H_{\text{D}} s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H_{\text{D}} s_i$
- recur via $H_{\text{D}} s_i = H_{\text{D}} s_{i-1} - H_{\text{D}} \Delta s_i$ with very sparse Δs_i but likely **dense** H_{D} … looks expensive ☹
- instead maintain $u_i = L^{-1}A^T s_i$ and $v_i = L^{-1}A^T y_i \implies$
- possible to recur required coefficients $g_{\text{D}}^T s_i$ and $s_i^T H_{\text{D}} s_i \equiv u_i^T u_i$ very efficiently via $u_i = u_{i-1} - \Delta u_i$, where $L\Delta u_i = A^T \Delta s_i$
- as Δs_i, A and H are sparse, result of forward solve Δu_i is often **very sparse** ☺
- required $y_i^T H_{\text{D}} s_i \equiv u_i^T v_i$ looks harder as update to v_i is dense … but can also be performed using inner-products involving Δu_i

Dual projected search

- have $H_{\mathrm{D}} = AH^{-1}A^T$ and $g_{\mathrm{D}} = -AH^{-1}g - b$
- recall require $\alpha^+ \approx \arg\min q_{\mathrm{D}}(P[y + \alpha s])$ for $\alpha \geq 0$
- investigate for $\alpha = \alpha_i + \Delta\alpha \leq \alpha_{i+1}$ and $y_i = P[y_i + \alpha_i s]$:
  $q_i(\alpha) = q_{\mathrm{D}}(y_i) + \Delta\alpha(g_{\mathrm{D}}^T s_i + y_i^T H_{\mathrm{D}} s_i) + \frac{1}{2}\Delta\alpha^2 s_i^T H_{\mathrm{D}} s_i$
- recur via $H_{\mathrm{D}} s_i = H_{\mathrm{D}} s_{i-1} - H_{\mathrm{D}} \Delta s_i$ with very sparse Δs_i but likely **dense** H_{D} …looks expensive ☹
- instead maintain $u_i = L^{-1}A^T s_i$ and $v_i = L^{-1}A^T y_i \implies$
- possible to recur required coefficients $g_{\mathrm{D}}^T s_i$ and $s_i^T H_{\mathrm{D}} s_i \equiv u_i^T u_i$ very efficiently via $u_i = u_{i-1} - \Delta u_i$, where $L\Delta u_i = A^T \Delta s_i$
- as Δs_i, A and H are sparse, result of forward solve Δu_i is often **very sparse** ☺
- required $y_i^T H_{\mathrm{D}} s_i \equiv u_i^T v_i$ looks harder as update to v_i is dense … but can also be performed using inner-products involving Δu_i
- sparse forward solves now available for HSL solvers `HSL_MA57/87/97` ☺

Dual subspace minimization

- acceleration by subspace minimization along $\boldsymbol{y}_k^c + \boldsymbol{s}$
  - partition variables $\boldsymbol{s}$ into active $\boldsymbol{s}_{\mathrm{A}_k}$ and free $\boldsymbol{s}_{\mathrm{F}_k}$ components according to status of $\boldsymbol{y}_k^c$
  - find (approximate) solution $\boldsymbol{s}_k$ to

$$\text{EQP}: \quad \underset{\boldsymbol{s} \in \mathbb{R}^m}{\text{minimize}} \quad q_{\mathrm{D}}(\boldsymbol{y}_k^c + \boldsymbol{s}) \quad \text{subject to} \quad \boldsymbol{s}_{\mathrm{A}_k} = \boldsymbol{0}$$

  - set $\boldsymbol{y}_{k+1} \approx \arg\min q_{\mathrm{D}}(\boldsymbol{P}[\boldsymbol{y}_k^c + \alpha \boldsymbol{s}_k])$

Dual subspace minimization

- acceleration by subspace minimization along $y_k^c + s$
  - partition variables s into active s_{A_k} and free s_{F_k} components according to status of y_k^c
  - find (approximate) solution s_k to

$$\text{EQP}: \quad \underset{s \in \mathbb{R}^m}{\text{minimize}} \quad q_{\mathrm{D}}(y_k^c + s) \quad \text{subject to} \quad s_{\mathrm{A}_k} = 0$$

  - set $y_{k+1} \approx \arg\min q_{\mathrm{D}}(P[y_k^c + \alpha s_k])$
- EQP equivalent to $\underset{s \in \mathbb{R}^{m_k}}{\text{minimize}} \quad \frac{1}{2} s^T H_k s + g_k^T s$
  - $H_k = A_k H^{-1} A_k^T$ and $g_k = -A_k H^{-1}(g - A_k^T y_k^c) - b_k$
  - A_k and b_k are respectively the rows of A and components of b corresponding to the m_k free components s_{F_k}
  - H_k is positive semi-definite but may be singular

Digression III: the Fredholm Alternative

$$\textbf{DEQP}: \quad \underset{s \in \mathbb{R}^{m_k}}{\text{minimize}} \quad q_k(s) = \tfrac{1}{2} s^T H_k s + g_k^T s$$

Two possibilities:

1. q_k has a finite critical point s_k for which

$$H_k s_k = -g_k$$

- always if H_k is positive definite
- true if $g_k \in \text{Range}(H_k)$

Digression III: the Fredholm Alternative

$$\textbf{DEQP}: \quad \underset{s \in \mathbb{R}^{m_k}}{\text{minimize}} \quad q_k(s) = \tfrac{1}{2} s^T H_k s + g_k^T s$$

Two possibilities:

1. q_k has a finite critical point s_k for which

$$H_k s_k = -g_k$$

- always if H_k is positive definite
- true if $g_k \in \text{Range}(H_k)$

2. q_k decreases linearly without bound along a direction s_k for which

$$H_k s_k = 0 \text{ and } g_k^T s_k < 0$$

- true if $g_k \notin \text{Range}(H_k)$

Digression III: the Fredholm Alternative

$$\textbf{DEQP}: \quad \underset{s \in \mathbb{R}^{m_k}}{\text{minimize}} \quad q_k(s) = \tfrac{1}{2} s^T H_k s + g_k^T s$$

Two possibilities:

1. q_k has a finite critical point s_k for which

$$H_k s_k = -g_k$$

- always if H_k is positive definite
- true if $g_k \in \text{Range}(H_k)$

2. q_k decreases linearly without bound along a direction s_k for which

$$H_k s_k = 0 \text{ and } g_k^T s_k < 0$$

- true if $g_k \notin \text{Range}(H_k)$

This is the **Fredholm Alternative** for the data $[H_k, g_k]$

The structured Fredholm Alternative

Seek Fredholm Alternative for data $[H_k, g_k]$ where

$$H_k = A_k H^{-1} A_k^T \text{ and } g_k = A_k H^{-1}(A_k^T y_k^c - g) - b_k$$

- $H_k s_k = -g_k$ equivalent to

$$\begin{pmatrix} H & A_k^T \\ A_k & 0 \end{pmatrix} \begin{pmatrix} t_k \\ -s_k \end{pmatrix} = \begin{pmatrix} A_k^T y_k^c - g \\ b_k \end{pmatrix}$$

  for auxiliary unknowns t_k

The structured Fredholm Alternative

Seek Fredholm Alternative for data $[H_k, g_k]$ where

$$H_k = A_k H^{-1} A_k^T \text{ and } g_k = A_k H^{-1}(A_k^T y_k^c - g) - b_k$$

- $H_k s_k = -g_k$ equivalent to

$$\begin{pmatrix} H & A_k^T \\ A_k & 0 \end{pmatrix} \begin{pmatrix} t_k \\ -s_k \end{pmatrix} = \begin{pmatrix} A_k^T y_k^c - g \\ b_k \end{pmatrix}$$

  for auxiliary unknowns t_k

- Fredholm Alternative for data

$$\left[\begin{pmatrix} H & A_k^T \\ A_k & 0 \end{pmatrix}, \begin{pmatrix} A_k^T y_k^c - g \\ b_k \end{pmatrix} \right]$$

  gives required alternative $H_k s_k = 0$ and $g_k^T s_k < 0$
  $\iff$ $[H_k, g_k]$ is inconsistent

The structured Fredholm Alternative

Seek Fredholm Alternative for data $[H_k, g_k]$ where

$$H_k = A_k H^{-1} A_k^T \text{ and } g_k = A_k H^{-1}(A_k^T y_k^c - g) - b_k$$

- $H_k s_k = -g_k$ equivalent to

$$\begin{pmatrix} H & A_k^T \\ A_k & 0 \end{pmatrix} \begin{pmatrix} t_k \\ -s_k \end{pmatrix} = \begin{pmatrix} A_k^T y_k^c - g \\ b_k \end{pmatrix}$$

  for auxiliary unknowns t_k

- Fredholm Alternative for data

$$\left[\begin{pmatrix} H & A_k^T \\ A_k & 0 \end{pmatrix}, \begin{pmatrix} A_k^T y_k^c - g \\ b_k \end{pmatrix} \right]$$

  gives required alternative $H_k s_k = 0$ and $g_k^T s_k < 0$
  $\iff$ $[H_k, g_k]$ is inconsistent

- HSL sparse solvers `HSL_MA57/86/97` now provide Fredholm Alternative

Alternative to the Fredholm Alternative

$$\textbf{DEQP}: \quad \underset{s \in \mathbb{R}^{m_k}}{\text{minimize}} \quad q_k(s) = \tfrac{1}{2} s^T H_k s + g_k^T s$$

$$H_k = A_k H^{-1} A_k^T \text{ and } g_k = A_k H^{-1}(A_k^T y_k^c - g) - b_k$$

- apply conjugate-gradient method with safeguards to detect steps to infinity
- each matrix-vector product $H_k p$ requires solve with H and sparse matrix-vector products with A_k and A_k^T
- preconditioning possible but no obvious simple preconditioner

An example

POWELL20: $n = 10000, m = 10000$

- solve problem using interior-point package CQP from GALAHAD
- perturb constraints and resolve by dual gradient-projection DQP

	CQP	size of perturbation before DQP solve					
		0	10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}
time	4.60	0.03	0.13	0.41	1.92	9.21	7.94
its		0	1	1	15	32	35
changes		0	1	8	594	3506	4763

An example

`POWELL20`: $\mathbf{n = 10000, m = 10000}$

- solve problem using interior-point package `CQP` from GALAHAD
- perturb constraints and resolve by dual gradient-projection `DQP`

		size of perturbation before `DQP` solve					
	`CQP`	0	$\mathbf{10^{-6}}$	$\mathbf{10^{-5}}$	$\mathbf{10^{-4}}$	$\mathbf{10^{-3}}$	$\mathbf{10^{-2}}$
time	4.60	0.03	0.13	0.41	1.92	9.21	7.94
its		0	1	1	15	32	35
changes		0	1	8	594	3506	4763

Active-set changes per iteration with perturbation $\mathbf{10^{-2}}$:

584	285	245	345	331	340	332	297	291	255
249	223	223	213	207	197	205	192	166	146
129	123	133	134	124	115	114	114	107	87
63	44	16	1	0					

Summary

- dual gradient-projection method for large-scale, strictly-convex QP
- requires sparse factorization of Hessian but otherwise can be used "factorization-free"
- allows rapid change to the "active set"
- particularly suited to "warm starting"
- efficient projected search
- extensive use of Fredholm alternative
- many technical details
- easily generalised for regularization problems in ℓ_1 and ℓ_∞ norms using appropriate simple projections onto boxes and simplices
- implemented as a fortran 2003 module `DQP` in GALAHAD (G., Hogg, Scott)