Objective Value Estimation for Simple Bound Constrained QP

June 10, 2008

Stanley Chan
h5chan@ucsd.edu

1 Introduction

We consider a type of least square problem whose constraints are upper and lower bounds on the variables. Formally, the problem is of the form:

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & \|Cx - d\|_2^2 \\ \text{subject to} \quad & l \leq x \leq u, \end{aligned}$$

where C is an $m \times n$ matrix, d is a $m \times 1$ vector, l and u are $n \times 1$ vector of lower and upper bounds such that $l_i \leq x \leq u_i$, for $i = 1, \ldots, n$. Expanding the objective function as $(Cx - b)^T(Cx - b)$, drop the constant term $d^T d$ and scale the equation down by a factor of 2, the above problem can be restated as

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & -d^T Cx + \frac{1}{2} x^T C^T Cx \\ \text{subject to} \quad & \begin{pmatrix} I_n \\ -I_n \end{pmatrix} x \geq \begin{pmatrix} l \\ -u \end{pmatrix}, \end{aligned}$$

which is equivalent to the standard quadratic programming (QP) subject to inequality constraints:

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & \varphi(x) = c^T x + \frac{1}{2} x^T H x \\ \text{subject to} \quad & Ax \geq b, \end{aligned}$$

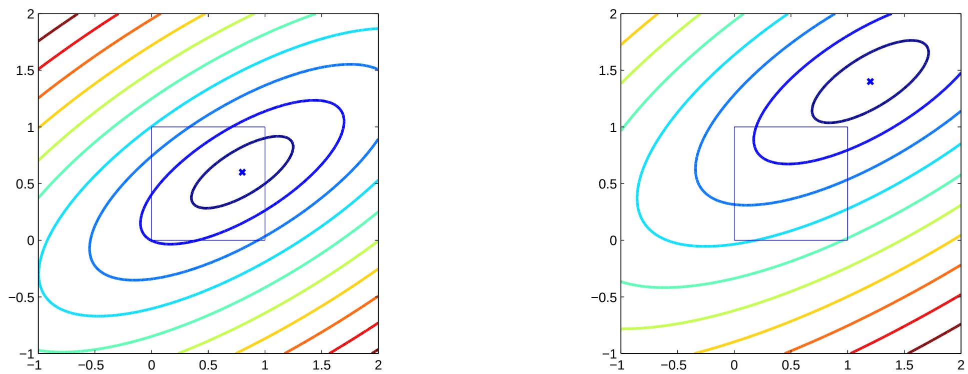
where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^{m \times 1}$, $c \in \mathbb{R}^{n \times 1}$, $H \in \mathbb{R}^{n \times n}$.

We have presented two iterative methods in solving the above QP problem in previous reports: (i) by steepest descent with projection onto active sets; (ii) by primal dual interior point methods. In this report, we address a question: Can we estimate the range of the optimal objective value, with relatively low computation costs?

Such question arises in places where the problems are huge. For example, in designing photo-masks in optical lithography, it is always desired to verify the hardware configurations before going to optical proximity correction (OPC) iterations, because a complete cycle of OPC iterations often take days. So if one can estimate the range of the optimal objective value, then engineers can decide whether to start the iteration, or reconfigure the setup (e.g. laser source wavelength, lens curvature).

Another application is in video processing. In LCD deblur, we need to repeatedly solving QPs for each frame. If it happens that the global motion estimation is way off the actual value, then it is meaningless to actually solve the QPs. Therefore, if one can give the range of optimal objective value, and if the lower bound is too high, then we can go back and double check the global motion estimation. This avoids wasting time in solving QPs.

In this report we present a method to estimate the upper and lower bound of the optimal objective value, via a low cost equality constrained QP. Although the problem we are looking at is simple bound constrained convex QP, we foresee that the method can be extended to general convex QP. In the followings we first describe the theory in section 2; implementation in section 3; and numerical results in section 4.



(a) $\hat{x} \in \mathbb{F}$

(b) $\hat{x} \notin \mathbb{F}$

Figure 1: The two cases: (1) the unconstrained solution $\hat{x} \in \mathbb{F}$; in this case the optimal solution is $\hat{x}$. (2) the unconstrained solution $\hat{x} \notin \mathbb{F}$; in this case the optimal solution is on the boundary of the feasible set.

2 Bound Estimation

Suppose we want to minimize a quadratic objective function

$$\varphi(x) = c^T x + \frac{1}{2} x^T H x,$$

subject to a set of simple bounded constraints

$$Ax \geq b.$$

We assume that H is positive definite, so the problem is convex. Let's first define a few terminologies:

Definition 2.1 (unconstrained solution)**.** *$\hat{x} \in \mathbb{R}^n$ is an unconstrained solution if for any $x \in \mathbb{R}^n$, $\varphi(\hat{x}) \leq \varphi(x)$.*

Definition 2.2 (feasible set)**.** *Define $\mathbb{F} := \{x \ : \ Ax \geq b\}$ be the feasible set.*

Definition 2.3 (active constraint)**.** *A constraint $a_i^T x \geq b_i$ is active if $a_i^T x = b_i$.*

Definition 2.4 (interior)**.** *For any set $\mathbb{F}$, the interior $\mathbb{F}^\circ$ is a subset of $\mathbb{F}$ s.t. for any $x \in \mathbb{F}^\circ$, there exists an open ball $B_\delta(x) := \{y : \|x - y\| < \delta\} \subset \mathbb{F}$, for some $\delta > 0$.*

Definition 2.5 (boundary)**.** *The boundary of $\mathbb{F}$, $\partial\mathbb{F}$, is $\mathbb{F}\backslash\mathbb{F}^\circ$.*

Definition 2.6 (open ball)**.** *We define an open ball centered at x_0 with radius δ to be the set $B_\delta(x_0) := \{x : \|x - x_0\| < \delta\}$.*

Then, there are two possibilities:
(i) the unconstrained solution $\hat{x}$ lies inside the feasible set, $\hat{x} \in \mathbb{F}$ (see Figure 1a);
(ii) the unconstrained solution $\hat{x}$ lies outside the feasible set, $\hat{x} \notin \mathbb{F}$ (see Figure 1b).

Obviously, if $\hat{x} \in \mathbb{F}$, then the optimal solution $x^* = \hat{x}$, and so the optimal value to the original problem: $\min \|Cx - d\|_2^2$, s.t. $l \leq x \leq u$ is zero. In this case the upper and lower bounds are trivially given by the unbounded solution $\hat{x}$.

Let's assume $\hat{x} \notin \mathbb{F}$. We first prove that the optimal solution has to lie on the boundary.

Proposition 2.1. *For a simple bound constrained convex QP, if the unconstrained solution $\hat{x} \notin \mathbb{F}$, then there exists an index s such that $a_s^T x^* = b_s$, where x^* is the optimal solution, a_i^T is the i^{th} row of A, and b_i is the i^{th} element of b.*

Proof. We prove by contradiction. Suppose there does not exists such index s, then for all i, $a_i^T x^* > b_i$, so $x^* \in \mathbb{F}^\circ$ (for otherwise x^* is infeasible). Since $x^* \in \mathbb{F}^\circ$, there there exists δ such that $\forall z \in B_\delta(x^*)$, $\varphi(x^*) < \varphi(x)$. Suppose that $\hat{x}$ is another optimal point, so $\varphi(x^*) = \varphi(\hat{x})$. Now for any $0 < \lambda < 1$, let $z = \lambda x^* + (1 - \lambda)\hat{x}$. By convexity of φ, we have $\varphi(z) \leq \lambda\varphi(\hat{x}) + (1 - \lambda)\varphi(x^*) = \varphi(x^*)$. Contradiction, because x^* by definition is the optimal solution. □

Next we want to show that by projecting the unbounded solution onto the closest active constraint(s), we can estimate the upper bound of the optimal solution.

Proposition 2.2. *For a simple bound constrained convex QP, let $\hat{x}$ be the unconstrained solution s.t. $\hat{x} \notin \mathbb{F}$. Then there exists a projected solution $x_p \in \mathbb{F}$ such that $\|x_p - \hat{x}\| \leq \|x - \hat{x}\|$ for all $x \in \mathbb{F}$, and $\varphi(x^*) \leq \varphi(x_p)$.*

Proof. Since $\mathbb{F}$ is a compact subset of $\mathbb{R}^n$, and $\|\cdot\|$ is a continuous function, $\inf\{\|z - \hat{x}\| : z \in \mathbb{F}\}$ exists. Thus, there exists $x_p \in \mathbb{F}$ s.t. $\|x_p - \hat{x}\| \leq \|x - \hat{x}\|$ for all $x \in \mathbb{F}$.

Since $\mathbb{F}$ is closed, must have $x_p \in \partial\mathbb{F}$. So by Proposition 2.1, as $x^* \in \partial\mathbb{F}$ and is the optimal point, we must have $\varphi(x^*) \leq \varphi(x_p)$. □

Before we go into the lower bound of objective value, we want to show that the true optimal point x^* lies on the same active constraint as the projected solution x_p.

Proposition 2.3. *For a simple bound constrained convex QP, let $\hat{x}$ be the unconstrained solution s.t. $\hat{x} \notin \mathbb{F}$. Let x_p be the projected solution defined in Proposition 2.2. Then the true constrained solution x^* lies on the same active constraint as x_p, i.e. if $a_s^T x_p = b_s$, then $a_s^T x^* = b_s$.*

Proof. Let $\mathcal{H}_s := \{x : a_s^T x \geq b_s\}$ be the hyperplane of the active constraint $a_s^T x_p = b_s$. We first claim that there exists a unique minimum point $x_e \in \mathcal{H}_s$ such that $\varphi(x_e) \leq \varphi(x)$, $\forall x \in \mathcal{H}_s$. But this simple, for existence of x_e follows from minimizing a convex function φ over a convex set $\mathcal{H}_s$, and uniqueness follows from the convexity. An immediate consequence of this is that $\varphi(x_e) \leq \varphi(x)$, $\forall x \in B_\delta(x_e) \bigcap \mathcal{H}_s$.

Now, we claim that x^* satisfies $a_s^T x^* = b_s$. Suppose not, i.e. $x^* \in \mathcal{H}_s^\circ$, let $z = (1-\lambda)x^* + \lambda x_e$, where $0 < \lambda < 1$. Since $x^* \neq x_e$ (for otherwise there is nothing to prove!), we have either $\varphi(z) \leq \varphi(x_e)$ or $\varphi(z) \leq \varphi(x^*)$. The first case leads to contradiction, because we have just proven that x_e is the unique minimizer over $B_\delta(x_e) \bigcap \mathcal{H}_s$. The second case also leads to contradiction, because by definition x^* is the true optimal solution in $\mathbb{F} \subset \mathcal{H}_s$. Thus, we must have x^* satisfying $a_s^T x^* = b_s$.

□

The following proposition concerns with the lower bound of the $\varphi(x^*)$. We claim that the lower bound of $\varphi(x^*)$ can be found by solving an equality constrained QP.

Proposition 2.4. *For a simple bound constrained convex QP, let $\hat{x}$ be the unconstrained solution s.t. $\hat{x} \notin \mathbb{F}$. Let x_p be the projected solution defined in Proposition 2.2. If x_e is the solution to the equality constrained problem*

$$\begin{aligned} \mathop{\mathit{minimize}}_{x\in\mathbb{R}^n} \quad & \varphi(x) = c^T x + \frac{1}{2} x^T H x \\ \mathit{subject\ to} \quad & a_s^T x = b_s, \end{aligned}$$

where $a_s^T x = b_s$ is an active constraint at the projected solution x_p, then $\varphi(x_e) \leq \varphi(x^*)$.

Proof. From Proposition 2.3, x^*, x_p and x_e are in the set $\{x : a_s^T x = b_s\}$. By definition x_e is the minimizer for $\varphi(x)$ along the active constraint $a_s^T x = b_s$. So, must have $\varphi(x_e) \leq \varphi(x^*)$. □

As a quick summary, we conclude that the upper bound of the objective value can be found projecting the unconstrained solution to its closest constraint. Then along the active constraint (any one of the active constraints), the solution that minimizes the objective function gives the lower bound of the objective value.

Example

Let's illustrate the idea with an example. Suppose we want to find the upper and lower bound for the problem:

$$\begin{aligned} \mathop{\text{minimize}}_{x\in\mathbb{R}^2} \quad & \phi(x) = \|Cx - d\|^2 \\ \text{subject to} \quad & 0 \leq x \leq 1, \end{aligned}$$

where $C = \begin{pmatrix} 3 & -1 \\ -1 & 2 \end{pmatrix}$, and $d = \begin{pmatrix} 4 \\ -1 \end{pmatrix}$.

Put it into standard QP format, we have

$$\begin{aligned} \mathop{\text{minimize}}_{x\in\mathbb{R}^2} \quad & \varphi(x) = c^T x + \frac{1}{2} x^T H x \\ \text{subject to} \quad & Ax \geq b, \end{aligned}$$

where $c = \begin{pmatrix} -13 \\ 6 \end{pmatrix}$, $H = \begin{pmatrix} 10 & -5 \\ -5 & 5 \end{pmatrix}$, $A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}$, and $b = \begin{pmatrix} 0 \\ 0 \\ -1 \\ -1 \end{pmatrix}$.

The unconstrained solution $\hat{x}$ can be found by solving the problem $\min c^T x + \frac{1}{2} x^T H x$. In particular, we can solve

$$H\hat{x} = -c,$$

using LSQR [3] and can find that

$$\hat{x} = \begin{pmatrix} 1.4 \\ 0.2 \end{pmatrix}.$$

Since $\hat{x}_1 = 1.4 > 1$, we need to project this solution onto the closest constraint, namely $\hat{x}_1 \leq 1$ (which corresponds to $a_3^T x \geq b_3$). By doing that we get the projected solution

$$x_p = \begin{pmatrix} 1.0 \\ 0.2 \end{pmatrix}.$$

Then we can find a point along the active constraint $a_3^T x = b_3$ such that the objective function is minimized. That is

$$\begin{aligned} \underset{x \in \mathbb{R}^2}{\text{minimize}} \quad & c^T x + \frac{1}{2} x^T H x \\ \text{subject to} \quad & a_3^T x = b_3, \end{aligned}$$

where $a_3 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$, $b_3 = -1$.

This equality constrained problem can solved by considering the optimality condition, namely the KKT equation. Thus we want to find p such that

$$\begin{pmatrix} H & a_3^T \\ a_3 & 0 \end{pmatrix} \begin{pmatrix} p \\ \lambda \end{pmatrix} = - \begin{pmatrix} g_0 \\ 0 \end{pmatrix},$$

where $g_0 = c + Hx_p$, and λ is the dual multiplier [1]. Then the solution x_e of this problem is

$$x_e = x_p + \alpha p,$$

where $\alpha = -\frac{g_0^T p}{p^T H p}$. (for more details please refer to [1]).

By doing that we can find $g_0 = \begin{pmatrix} -4 \\ 2 \end{pmatrix}$, $p = \begin{pmatrix} 0 \\ -0.4 \end{pmatrix}$, $\alpha = 1$, so

$$x_e = \begin{pmatrix} 1.0 \\ 0.2 \end{pmatrix} + 1 \begin{pmatrix} 0 \\ -0.4 \end{pmatrix} = \begin{pmatrix} 1 \\ -0.2 \end{pmatrix}.$$

Therefore, the lower and upper bounds for the problem is given by

$$\phi(x_e) \leq \phi(x^*) \leq \phi(x_p),$$

i.e.

$$0.4 \leq \phi(x^*) \leq 0.8.$$

Indeed the true optimal is $x^* = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, and $\phi(x^*) = 0.5$.

Figure 2 illustrates this example.

3 Implementation

We now present numerical implementation for the above proposed idea.

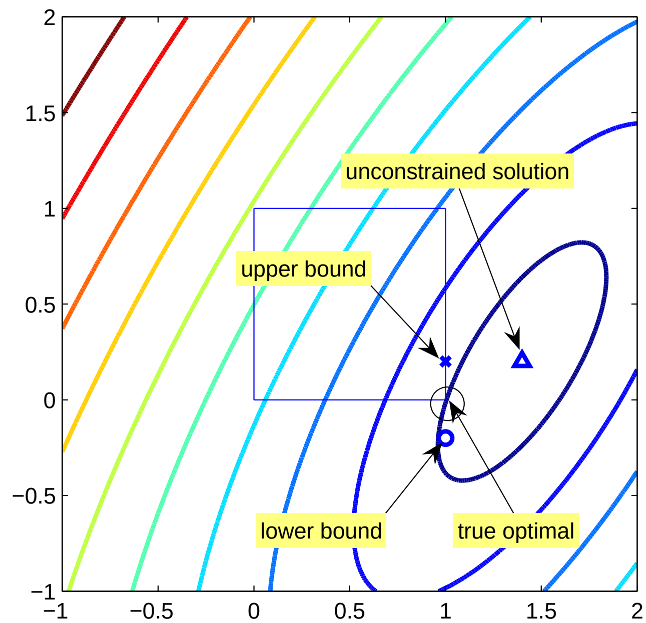


Figure 2: Geometric illustration of determining the upper and lower bound for the QP.

3.1 Unconstrained solution

To compute the unconstrained solution $\hat{x}$, we have to solve the unconstrained problem

$$\min c^T x + \frac{1}{2} x^T H x,$$

or equivalently the least square problem

$$\min \|Cx - d\|_2^2.$$

In case of large problems with sparse matrix C, we can solve the problem using standard large scale packages, e.g. LSQR. In MATLAB, we can call the following routine to perform the task:

```
sol_unconstrained = lsqr(C, d, TOL, max_itn);
```

where `TOL` is the tolerance level (say 1×10^{-10}), `max_itn` is the maximum number of conjugate gradient iterations of LSQR.

3.2 Projected solution

To compute the projected solution for simple bound constrained problems, we can set elements of `sol_unconstrained` to its closest bound if it is out of bound. More precisely, we implement the following in MATLAB:

```
sol_project = sol_unconstrained;
sol_project(sol_project>=1)=1;
sol_project(sol_project<=0)=0;
```

3.3 EQP solution

To compute the EQP solution, we first have to determine which of the active constraints to be picked, because in our proposal we only use one of them. In our simple bound constrained problem, there are n constraints associated with the upper bound $x \leq 1$, and anther n associated with the lower bound $0 \leq x$. Since we are only interested in picking one out of the $2n$ constraints, we can pick the one with least index. In MATLAB, we can do:

```
active_set_p = [(sol_project>=1); false(n,1)];
active_set_n = [false(n,1); (sol_project<=0)];
idx_p        = find(active_set_p, 1, 'first');
idx_n        = find(active_set_n, 1, 'first');
```

Of course, if both `idx_p` and `idx_n` are empty, (i.e. none of the constraints are active, so the unconstrained solution is inside the feasible set), then the unconstrained solution is the true solution. For otherwise, we have to setup the KKT equations and solve the system of linear equations. In MATLAB, we can do:

```
x0 = sol_project;
if (isempty(idx_p))&&(isempty(idx_n))
     KKT     = sparse(H);
     g0      = c + H*x0;
     rhs     = -sparse(g0);
else
    if isempty(idx_p)
        Aactive  = A(idx_n, :);
    else
        Aactive  = A(idx_p, :);
    end
    [ma na] = size(Aactive);
    KKT     = sparse([H           Aactive';
                     Aactive      zeros(ma,ma);]);
    g0      = c + H*x0;
    rhs     = -sparse([  g0;
                        zeros(ma,1)  ;]);
end

dv      = lsqr(KKT, rhs, 1e-10, 500);
```

After computing the search direction `dv`, we can compute the minimizer by doing:

```
p       = dv(1:n);
alpha   = -g0'*p/(p'*H*p);
sol_eq  = x0 + alpha*p;
```

3.4 Upper and Lower bounds

To compute the upper and lower bounds, we can simply plug `sol_eq` and `sol_project` into the objective function

```
x_L = sol_eq;
x_U = sol_project;
UpBound    = (C*x_U - d)'*(C*x_U - d)/prod(size(x_U));
LowBound   = (C*x_L - d)'*(C*x_L - d)/prod(size(x_L));
```

4 Demonstration

We demonstrate here an inverse signal synthesis problem. Given a target signal d, and a transfer function C, we would like to prewarp the input signal x so that the forward error $\|Cx - d\|_2^2$ is minimized. In this experiment, we set

$$d = \begin{cases} 0, & 1 \le k \le 30 \\ 0.5, & 31 \le k \le 50 \\ 1, & 51 \le k \le 70 \\ 0, & 71 \le k \le 100 \end{cases},$$

and

$$\mathbf{C} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 \end{pmatrix}.$$

As usual, we need to bound the solution so that $0 \le x \le 1$.

Now, we can use either steepest descent algorithm (with projection onto active sets) or primal dual interior point method [2] to solve the problem. Here are their numerical performance (to TOL of 1×10^{-4}):

```
                              steepest descent
itn     norm(g)        obj_val
  1  4.212410e+000 7.429693e-003
  2  2.418249e-001 5.971998e-003
  3  1.367340e-001 5.700149e-003
  4  6.845258e-002 5.518844e-003
  ...
194  1.027071e-004 5.040010e-003
195  1.588348e-004 5.040010e-003
196  9.896364e-005 5.040010e-003
```

```
                              interior point method
itn      mu         obj_val
  1 1.000000e-001 1.278012e-001
  2 1.000000e-002 1.091769e-001
  3 1.000000e-003 4.563403e-002
  4 1.000000e-004 2.105182e-002
  ...
 17 1.000000e-017 5.040001e-003
 18 1.000000e-018 5.040000e-003
 19 1.000000e-019 5.040000e-003
```

Clearly, both algorithm converges to a solution with objective value 5.04×10^{-3}. Now, let's compute the upper and lower bounds for this problem. Using the proposed method discussed above, we find that

$$3.2781 \times 10^{-5} \le \varphi(x^*) \le 0.1585.$$

Therefore, the upper and lower bounds computed are valid.

Figure 3 below shows the plots for this particular problem.

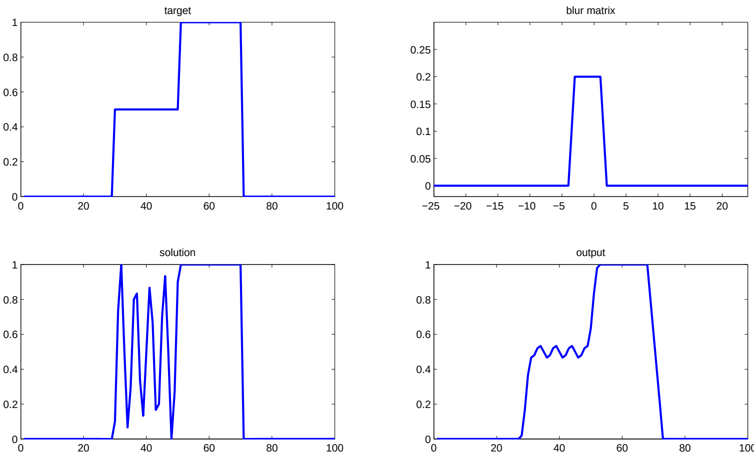


Figure 3: Plots for the demonstration problem. Top left: the target signal d; Top right: the transfer function in making the matrix C; Bottom left: optimal solution x^* (from interior point method); Bottom right: the output Cx^*.

5 Conclusion

We proposed a computationally cheap method in determining the upper and lower bounds of the objective value for a convex simple bound constrained quadratic programming problem. The proposed method first solves the unconstrained solution using LSQR, then project the unconstrained solution onto its closest bounds to obtain the upper bound. Determining the active constraint at the projected solution, we solve an equality constrained QP to obtain the lower bounds. We provide theoretical proofs and numerical demonstrations to verify the results. Although this report focused on simple bound constrained problems, the method can be generalized to any linear inequality constrained convex QP.

References

[1] P. E. Gill, W. Murray and M. H. Wright, *Practical Optimization*, Academic Press, 1986.

[2] P. E. Gill, A. Forsgren and M. H. Wright, *Interior methods for nonlinear optimization*, SIAM Review 44 (2002), 525-597.

[3] C. C. Paige, and M. A. Saunders, *LSQR: An Algorithm for Sparse Linear Equations And Sparse Least Squares*, ACM Trans. Math. Soft., Vol.8, 1982, pp. 43-71.