Journal of Computational Mathematics, Vol.21, No.1, 2003, ??–??.

LINEAR SYSTEMS ASSOCIATED WITH NUMERICAL METHODS FOR CONSTRAINED OPITMIZATION [*1)]

Y. Yuan
(*State Key Laboratory of Scientific and Engineering Computing, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, P. O. Box 2719, Beijing 100080, China*)

Abstract

Linear systems associated with numerical methods for constrained optimization are discussed in this paper. It is shown that the corresponding subproblems arise in most well-known methods, no matter line search methods or trust region methods for constrained optimization can be expressed as similar systems of linear equations. All these linear systems can be viewed as some kinds of approximation to the linear system derived by the Lagrange-Newton method. Some properties of these linear systems are analyzed.

Key words: constrained optimization, linear equations, Lagrange-Newton method, trust region, line search

1. Introduction

General nonlinear optimization problems have the form:

$$\min_{x\in\Re^n} f(x) \tag{1.1}$$

subject to

$$c_i(x) = 0, \qquad i = 1, 2, ...m_e, \tag{1.2}$$
$$c_i(x) \geq 0, \qquad i = m_e + 1, ..., m, \tag{1.3}$$

where $m \geq m_e \geq 0$ are two non-negative integers. From the Kuhn-Tucker theory, at a local solution x^* of (1.1)-(1.3), there exist Lagrange multipliers $\lambda_i (i = 1, 2...m)$ such that

$$\nabla f(x^*) - \sum_{i=1}^{m} \lambda_i \nabla c_i(x^*) \quad = \quad 0, \tag{1.4}$$
$$\lambda_i \geq 0, \quad \lambda_i c_i(x^*) \quad = \quad 0, \quad i = m_e, ..., m. \tag{1.5}$$

Let $\mathcal{E} = \{1, 2, ..., m_e\}$, and $\mathcal{I}^* = \{i| \quad c_i(x^*) = 0, i = m_e, ...m\}$ be the index set of all active inequality constraints. The first order necessary condition (1.4)-(1.5) can be written as

$$\nabla f(x^*) - \sum_{i\in\mathcal{E}\cup\mathcal{I}^*} \lambda_i \nabla c_i(x^*) = 0. \tag{1.6}$$

Thus, when the iterates are close to a solution, inequality constraints can be treated as equality constraints by applying the active set strategy. Therfore, for simplicity, some of the methods

[*] Received.

[1)] Research partially supported by Chinese NSF grants 19731010 and CAS knowledge innovation program.

we discussed in the paper are for equality constrained problem

$$\min_{x\in\Re^n} \quad f(x) \tag{1.7}$$
$$s. \quad t. \qquad c(x) = 0. \tag{1.8}$$

Some methods require the iterates staying in the interior of the feasible region, therefore only inequality constraints are considered. For these methods, we can only apply to inequality constrained problems:

$$\min_{x\in\Re^n} \quad f(x) \tag{1.9}$$
$$s. \ t. \qquad c(x) \geq 0. \tag{1.10}$$

Almost all numerical methods for nonlinear optimization are iterative. For a line search method, a search direction d_k will be generated and a suitable point $x_k + \alpha_k d_k$ is chosen so that a reduction in a merit function (which is a penalty function) will be obtained. For a trust region method, a trial step s_k is computed in a trust region, and some criterion will be used to decide whether the step s_k should be accepted.

For unconstrained problem ($m = m_e = 0$), the Newton's method is

$$x_{k+1} = x_k - (\nabla^2 f(x_k))^{-1}\nabla f(x_k), \tag{1.11}$$

which has a local quadratic convergence property if the Hessian matrix is positive definite at the solution. The Newton step $d = -(\nabla^2 f(x_k))^{-1}\nabla f(x_k)$ can be obtained by solving the following linear system

$$(\nabla^2 f(x_k))d = -\nabla f(x_k). \tag{1.12}$$

A very important class of methods for unconstrained optimization, quasi-Newton methods, define the search direction by solving

$$B_k d = -\nabla f(x_k), \tag{1.13}$$

where B_k is a quasi-Newton matrix. The linear system determines the next iterate, therefore play the essential role for the convergence rate of the method. It is well known([3]) that the superlinear convergence of quasi-Newton methods is equavalent to

$$\lim_{k\to\infty} \frac{\|(B_k - \nabla^2 f(x^k))d_k\|}{\|d_k\|} = 0. \tag{1.14}$$

For constrained optimization problems, the search directions or the trial steps are computed by solving some subproblems. These subproblems are some kinds of approximation to the orginal optimization problem. Most of these subproblems are simpler optimization problems. For example, the quadratic subproblem of the sequential quadratic programming method for (1.1)-(1.3) has the form

$$\min_{d\in\Re^n} d^T\nabla f(x_k) + \frac{1}{2}d^T B_k d \tag{1.15}$$

$$s. \quad t. \qquad c_i(x_k) + d^T\nabla c_i(x_k) = 0, \qquad i = 1, ..., m_e; \tag{1.16}$$
$$c_i(x_k) + d^T\nabla c_i(x_k) \geq 0, \qquad i = m_e + 1, ..., m, \tag{1.17}$$

where B_k is updated from iteration to iteration and is an approximation to the Hessian matrix of the Lagrange function. The first order necessary conditions for the above subproblem are

$$\nabla f(x_k) + B_k d_k = \sum_{i=1}^{m} \lambda_i \nabla c_i(x_k) \tag{1.18}$$

where $\lambda_i (i = 1, ..., m)$ are the Lagrange multipliers satisfying

$$\lambda_i \geq 0, \qquad i = m_e + 1, ..., m, \tag{1.19}$$
$$\lambda_i (c_i(x_k) + d_k^T \nabla c_i(x_k)) = 0 \qquad i = m_e + 1, ..., m. \tag{1.20}$$

Except the complementary condition (1.20), the search direction d_k is defined by the linear system (1.18). In general, subproblems for constrained optimization use quadratic models, implying that their first order conditions will be linear systems. Therefore for most numerical methods, the essential part that defines the search direction or the trial step depends on some linear systems. The main aim of this paper is to expose the corresponding such linear systems for various methods.

For simplicity, throughout this paper, we use the following notations:

$$g(x) = \nabla f(x), \tag{1.21}$$

$$A(x) = \nabla c(x)^T = [\nabla c_1(x), \nabla c_2(x), ..., \nabla c_m(x)], \tag{1.22}$$

$g_k = g(x_k)$, $c_k = c(x_k)$ and $A_k = A(x_k)$. We also use the notation

$$W(x, \lambda) = \nabla^2 f(x) - \sum_{i=1}^{m} (\lambda)_i \nabla^2 c_i(x) \tag{1.23}$$

to denote the Hessian matrix of the Lagrange function

$$L(x, \lambda) = f(x) - \lambda^T c(x). \tag{1.24}$$

2. Linear Systems Associated with Subproblems

In this section, we give a unified view to different methods, namely we try to write down the determining linear systems for various methods, even though the explicit descriptions of these algorithms are in the form of solving minimization subproblems.

2.1. Lagrange-Newton Step

Consider equality constrained problems. Based on the Kuhn-Tucker theory, a solution of x^* of the constrained optimization problem and its corresponding Lagrange multiplier λ^* consist a saddle point of the Lagrange function $L(x, \lambda)$. Namely, (x^*, λ^*) is a solution of the following system:

$$\nabla_x L(x, \lambda) \quad = \quad \nabla f(x) - A(x)\lambda = 0, \tag{2.1}$$
$$\nabla_\lambda L(x, \lambda) \quad = \quad c(x) = 0. \tag{2.2}$$

Let x_k be the current iterate point, and λ_k be an approximate Lagrange multiplier. The Newton step for the above nonlinear equations is

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k \\ -A_k^T & 0 \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k \\ c_k \end{pmatrix}, \tag{2.3}$$

where $W(x_k, \lambda_k)$ is defined by (1.23).

2.2. SQP step

For the sequential quadratic programming(SQP) method, when there are only equality constraints, the subproblem (1.15)-(1.16) can be expressed as

$$\min g_k^T d + \frac{1}{2} d^T B_k d \tag{2.4}$$

subject to

$$c_k + A_k^T d = 0, \tag{2.5}$$

Therefore, if we denote the multipliers of the QP problem (2.4)-(2.5) by η, the equality constrained QP is equivalent to the following linear system:

$$\begin{bmatrix} B_k & -A_k \\ -A_k^T & 0 \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k \\ c_k \end{pmatrix}. \tag{2.6}$$

Hence we can see that the linear system obtained from the SQP method is that same as that of the Lagrange-Newton method if the quasi-Newton matrix is the exact Hessian matrix of the Lagrange function at x_k.

2.3. Courant Penalty function

Consider the Courant penalty function ([4])

$$P(x) = f(x) + \frac{1}{2}\sigma \|c(x)\|_2^2. \tag{2.7}$$

At a minimizer of the Courant penalty function, the equality

$$g(x) + \sigma A(x) c(x) = 0 \tag{2.8}$$

should hold. The Newton's method for the above nonlinear equation would give

$$\left[\nabla^2 f(x_k) + \sigma A(x_k) A(x_k)^T + \sigma \sum_{i=1}^m c_i(x_k) \nabla^2 c_i(x_k)\right] d = -[g_k + \sigma A_k c_k]. \tag{2.9}$$

Define

$$\sigma[A_k^T d + c_k] = -\eta, \tag{2.10}$$

we obtain

$$\begin{bmatrix} W(x_k, -\sigma c_k) & -A_k \\ -A_k^T & -\frac{1}{\sigma} I \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k \\ c_k \end{pmatrix}. \tag{2.11}$$

2.4. Augmented Lagrange function

The augmented Lagrange function is

$$P(x) = f(x) - \lambda^T c(x) + \frac{1}{2}\sigma \|c(x)\|_2^2, \tag{2.12}$$

where $\lambda \in \Re^m$ is the Lagrange multiplier and $\sigma > 0$ is the penalty parameter. The condition for a stationary point of the augmented Lagrange function is

$$g(x) - A(x)\lambda + \sigma A(x) c(x) = 0. \tag{2.13}$$

Applying Newton's method gives

$$[\nabla^2 f(x_k) + \sigma A(x_k)A(x_k)^T + \sum_{i=1}^{m}[\sigma c_i(x_k) - (\lambda)_i]\nabla^2 c_i(x_k)]d = -[g_k + A_k(\sigma c_k - \lambda)]. \tag{2.14}$$

Using relation (2.10), we can rewrite the above equation as

$$\begin{bmatrix} W(x_k, \lambda - \sigma c_k) & -A_k \\ -A_k^T & -\frac{1}{\sigma}I \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k + A_k\lambda \\ c_k \end{pmatrix}. \tag{2.15}$$

2.5. Inverse barrier function

The inverse barrier function is used for inequality constrained problem, and it has the form:

$$f(x) + \frac{1}{\sigma}\sum_{i=1}^{m}\frac{1}{c_i(x)}. \tag{2.16}$$

A minimizer of the inverse barrier function satisfy the following condition

$$g(x) - \frac{1}{\sigma}\sum_{i=1}^{m}\frac{1}{c_i^2(x)}\nabla c_i(x) = 0, \tag{2.17}$$

which can be written as

$$g(x) - \frac{1}{\sigma}A(x)D(x)^{-3}c(x) = 0, \tag{2.18}$$

where $D(x)$ is defined by (2.25).

The Newton step for (2.17) is

$$\left[\nabla^2 f(x_k) + 2\frac{1}{\sigma}A_k D(x_k)^{-3}A_k^T - \frac{1}{\sigma}\sum_{i=1}^{m}\frac{1}{c_i^2(x_k)}\nabla^2 c_i(x_k)\right] d = -g_k + \frac{1}{\sigma}A_k D(x_k)^{-3}c_k. \tag{2.19}$$

Define

$$\frac{1}{\sigma}D(x_k)^{-3}(2A_k^T d - c_k) = -\eta, \tag{2.20}$$

the above system reduced to

$$\begin{bmatrix} W(x_k, \frac{1}{\sigma}D(x_k)^{-2}c_k) & -A_k \\ -A_k^T & -\frac{1}{2}\sigma D(x_k)^3 \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k \\ -\frac{1}{2}c_k \end{pmatrix}. \tag{2.21}$$

Please notice that the second term in the right hand side has the different sign from the "standard" system (2.3).

2.6. Log-barrier function

The log-barrier function is very similar to the inverse barrier function, and it is given by:

$$f(x) - \frac{1}{\sigma}\sum_{i=1}^{m}\log(c_i(x)). \tag{2.22}$$

The stationary condition for the log-barrier function is

$$g(x) - \frac{1}{\sigma}\sum_{i=1}^{m}\frac{1}{c_i(x)}\nabla c_i(x) = 0, \tag{2.23}$$

which can be written as

$$g(x) - \frac{1}{\sigma}A(x)D(x)^{-2}c(x) = 0, \tag{2.24}$$

where $D(x)$ is a diagonal matrix whose entries are $c_i(x)$, namely

$$D(x) = Diag(c_1(x), c_2(x), ..., c_m(x)). \tag{2.25}$$

The Newton step for (2.23) is

$$\left[\nabla^2 f(x_k) + \frac{1}{\sigma}A_k D(x_k)^{-2}A_k^T - \frac{1}{\sigma}\sum_{i=1}^{m}\frac{1}{c_i(x_k)}\nabla^2 c_i(x_k)\right] d = -g_k + \frac{1}{\sigma}A_k D(x_k)^{-2}c_k \tag{2.26}$$

Define

$$\frac{1}{\sigma}D(x_k)^{-2}(A_k^T d - c_k) = -\eta \tag{2.27}$$

the above system reduced to

$$\begin{bmatrix} W(x_k, \frac{1}{\sigma}D(x_k)^{-2}c_k) & -A_k \\ -A_k^T & -\sigma D(x_k)^2 \end{bmatrix}\begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k \\ -c_k \end{pmatrix}. \tag{2.28}$$

Again, as in the inverse barrier function, here the second term in the right hand side also has the different sign from the "standard" system (2.3).

2.7. A transformed log-barrier function method

Vanderbei and Shanno[9] transforms the inequality constrained problem (1.9)-(1.10) into

$$\min_{x\in\Re^n} f(x) \tag{2.29}$$

$$s.t. \quad c(x) - y = 0 \tag{2.30}$$

$$y \geq 0, \tag{2.31}$$

by adding the slack variables $y \in \Re^m$. Using the log-barrier penalty to the inequality constraints (2.31) for the above problem, we obtain that

$$\min_{x\in\Re^n} f(x) - \frac{1}{\sigma}\sum_{i=1}^{m} log(y_i) \tag{2.32}$$

$$s.t. \qquad c(x) - y = 0. \tag{2.33}$$

The first order conditions for the above problem can be written as

$$g(x) - A(x)\lambda = 0, \tag{2.34}$$

$$-\frac{1}{\sigma}\frac{1}{y_i} + \lambda_i = 0, \quad i = 1, ..., m, \tag{2.35}$$

$$c(x) - y = 0. \tag{2.36}$$

Replacing (2.35) by $y_i - \sigma^{-1}\frac{1}{\lambda_i} = 0$, we give the modified form of the first order conditions:

$$\begin{bmatrix} g(x) - A(x)\lambda \\ y - \sigma^{-1}\Lambda^{-1}e \\ -c(x) + y \end{bmatrix} = 0, \tag{2.37}$$

where Λ is the diagonal matrix whose entries are the elements of vector λ, namely $\Lambda = Diag((\lambda)_1, (\lambda)_2, ..., (\lambda_k))$, and $e = (1, 1, ..., 1) \in \Re^m$ is a vector with all entries being 1. The Newton step for the above nonlinear equation is defined by

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k^T & 0 \\ 0 & \sigma^{-1}\Lambda_k^{-2} & I \\ -A_k^T & 0 & I \end{bmatrix} \begin{bmatrix} d \\ \eta \\ \delta y \end{bmatrix} = - \begin{bmatrix} g_k - A_k\lambda_k \\ y_k - \sigma^{-1}\Lambda_k^{-1}e \\ -c_k + y_k \end{bmatrix}, \tag{2.38}$$

where

$$\Lambda_k = Diag((\lambda_k)_1, (\lambda_k)_2, ..., (\lambda_k)_m). \tag{2.39}$$

It follows from relation (2.38) that

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k^T \\ -A_k^T & -\sigma^{-1}\Lambda_k^{-2} \end{bmatrix} \begin{bmatrix} d \\ \eta \end{bmatrix} = \begin{bmatrix} -g_k + A_k\lambda_k \\ c_k - \sigma^{-1}\Lambda_k^{-1}e \end{bmatrix}. \tag{2.40}$$

In the original method of Vanderbei and Shanno[9], (2.35) is replaced by $\lambda_i y_i - \sigma^{-1} = 0$, the linear system becomes

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k^T \\ -A_k^T & -\Gamma_k\Lambda_k^{-1} \end{bmatrix} \begin{bmatrix} d \\ \eta \end{bmatrix} = \begin{bmatrix} -g_k + A_k\lambda_k \\ c_k - \sigma^{-1}\Lambda_k^{-1}e \end{bmatrix}, \tag{2.41}$$

where Γ_k is the diagonal matrix defined by

$$\Gamma_k = Diag[(y_k)_1, (y_k)_2, ..., (y_k)_m]. \tag{2.42}$$

More details can be found also in [8].

2.8. Affine Scaling interior point method

In Coleman and Li[2], an affine scaling interior point method was studied for linearly inequality constrained problem. Here we extend the method to the case when the constraints are general nonlinear functions. The subproblem of the affine scaling interior point method can be derived by considering the following first order necessary conditions of the constrained optimization problem:

$$g(x) - A(x)\lambda = 0, \tag{2.43}$$
$$-D(x)\lambda = 0, \tag{2.44}$$

where the primal and dual feasibility constraints are ignored as interior point methods always keep the iterate points in the feasible region. The Newton step for (2.43)-(2.44) is

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k \\ -\Lambda_k A_k^T & -D(x_k) \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k + A_k\lambda_k \\ D_k\lambda_k \end{pmatrix}, \tag{2.45}$$

where Λ_k is defined by (2.39). If Λ_k is nonsingular (which is the case if the approximate Lagrange multipliers $\lambda_k > 0$), (2.45) can be written as

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k \\ -A_k^T & -\Lambda_k^{-1}D(x_k) \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k + A_k\lambda_k \\ c_k \end{pmatrix}. \tag{2.46}$$

Because the Newton step obtained in this way may not be a descent direction, [2] suggests that Λ_k be replaced by $|\Lambda_k|$, which is defined by $Diag(|(\lambda_k)_1|, |(\lambda_k)_2|, ..., |(\lambda_k)_m|)$ in (2.45). The modified Newton step d, which satisfies the following system

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k \\ -|\Lambda_k| A_k^T & -D(x_k) \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k + A_k \lambda_k \\ D_k \lambda_k \end{pmatrix}, \tag{2.47}$$

is also a stationary point of the augmented quadratic function

$$\psi(d) = g_k^T d + \frac{1}{2} d^T W(x_k, \lambda_k) d + \frac{1}{2} d^T A D_k^{-1} |\Lambda_k| A_k^T d. \tag{2.48}$$

Thus a trust region subproblem can be defined by

$$\begin{aligned} \min_{d \in \Re^n} \quad & g_k^T d + \frac{1}{2} d^T W(x_k, \lambda_k) d + \frac{1}{2} d^T A D_k^{-1} |\Lambda_k| A_k^T d & (2.49) \\ s.t. \quad & \|(d; D_k^{-\frac{1}{2}} A_k^T d)\| \leq \Delta_k. & (2.50) \end{aligned}$$

Let $\tau \geq 0$ be the Lagrange multiplier of the above subproblem, it follows that

$$g_k + W(x_k, \lambda_k) d + A D_k^{-1} |\Lambda_k| A_k^T d + \tau (I + A_k D_k^{-1} A_k^T) d = 0, \tag{2.51}$$

$$\tau(\Delta_k - \|(d; D_k^{-\frac{1}{2}} A_k^T d)\|) = 0. \tag{2.52}$$

Defining $D_k^{-1}(|\Lambda_k| A_k^T d + \tau A_k^T d) + \lambda_k = -\eta$, (2.51) can be reduced to

$$\begin{bmatrix} W(x_k, \lambda_k) + \tau I & -A_k \\ -|\Lambda_k| A_k^T - \tau A_k^T & -D(x_k) \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k + A_k \lambda_k \\ D_k \lambda_k \end{pmatrix}, \tag{2.53}$$

which can be written as

$$\begin{bmatrix} W(x_k, \lambda_k) + \tau I & -A_k \\ -A_k^T & -(|\Lambda_k| + \tau I)^{-1} D(x_k) \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k + A_k \lambda_k \\ (|\Lambda_k| + \tau I)^{-1} \Lambda_k c_k \end{pmatrix}, \tag{2.54}$$

if $(|\Lambda_k| + \tau I)$ is nonsingular.

2.9. CDT subproblem

The CDT subproblem for equality constrained optimization is as follows

$$\min g_k^T d + \frac{1}{2} d^T B_k d \tag{2.55}$$

subject to

$$\begin{aligned} \|c_k + A_k^T d\|_2^2 &\leq \xi_k^2, & (2.56) \\ \|d\|_2 &\leq \Delta_k, & (2.57) \end{aligned}$$

where ξ_k is a parameter between $\min_{\|d\| \leq \Delta_k} \|c_k + A_k^T d\|$ and $\|c_k\|$ (for example, see [1] and [7]). The solution d_k of the CDT subproblem (2.55)-(2.57) would satisfy

$$\begin{aligned} B_k d_k + g_k + \tau d_k + \mu A_k (A_k^T d_k + c) &= 0, & (2.58) \\ \tau(\Delta_k - \|d_k\|) &= 0, & (2.59) \\ \mu(\xi_k^2 - \|c_k + A_k^T d\|_2^2) &= 0, & (2.60) \end{aligned}$$

for some $\tau \geq 0$ and $\mu \geq 0$. Define $\eta = -\mu(A_k^T d + c)$, we can see that

$$\begin{bmatrix} B_k + \tau I & -A_k \\ -\mu A_k^T & -I \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k \\ \mu c_k \end{pmatrix}. \tag{2.61}$$

If $\mu > 0$, the above equation can be written as

$$\begin{bmatrix} B_k + \tau I & -A_k \\ -A_k^T & -\frac{1}{\mu} I \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k \\ c_k \end{pmatrix}. \tag{2.62}$$

2.10. Reduced Hessian Method

The two sided reduced Hessian method computes the search direction d_k by decompositing it into the null space and the range space of A_k. Assume that the QR factorization of A_k is as follows

$$A_k = Q_k \begin{bmatrix} R_k \\ 0 \end{bmatrix} = [Y_k, \quad Z_k] \begin{bmatrix} R_k \\ 0 \end{bmatrix}, \tag{2.63}$$

then the search direction d can be expressed as

$$d_k = Y_k y_k + Z_k z_k. \tag{2.64}$$

The two sided reduced Hessian method use the reduced Hessian $Z_k^T W(x_k, \lambda_k) Z_k$ or its approximation. The steps in the reduced spaces can be computed separately:

$$\begin{aligned} y_k &= -R_k^T c_k, & (2.65) \\ z_k &= -(Z_k^T W(x_k, \lambda_k) Z_k)^{-1} g_k. & (2.66) \end{aligned}$$

One advantage of using two sided reduced Hessian is that only $(n-m) \times (n-m)$ matrices are used, which is very favorable when both n and m are very large while $n-m$ is relatively small. Detailed studies on reduced Hessian methods can be found in [6]. For some extremely large problems, using the full Hessian is nearly impossible due to storage and computational cost. And there are many practical problems that the variables are required to stay in a small dimensional subspace though the problem size is quite large. It follows (2.64)-(2.66) that

$$\begin{aligned} Z_k^T W(x_k, \lambda_k) d_k &= Z_k^T W(x_k, \lambda_k)(Y_k Y_k^T d_k + Z_k Z_k^T d_k) \\ &= -Z_k^T W(x_k, \lambda_k)(A_k^T)^+ c_k - Z_k^T g_k. \end{aligned} \tag{2.67}$$

The above relation indicates that there exists a vector $\eta \in \Re^m$ such that

$$W(x_k, \lambda_k) d_k = -W(x_k, \lambda_k)(A_k^T)^+ c_k - g_k + A_k \eta. \tag{2.68}$$

Thus, we can see the search direction obtained by the two sided reduced Hessian method satisfies the following linear system

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k \\ -A_k^T & 0 \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k - W(x_k, \lambda_k)(A_k^T)^+ c_k \\ c_k \end{pmatrix}. \tag{2.69}$$

3. Some Properties of the Linear Systems

From the previous section, we can easily see that all the linear systems corresponding the different methods are similar in form, and they can be express as

$$\begin{bmatrix} W(x_k, \lambda_k) + T_k & -A_k \\ -A_k^T & S_k \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k + \hat{\epsilon}_k \\ c_k + \bar{\epsilon}_k \end{pmatrix}, \tag{3.1}$$

where $T_k \in \Re^{n\times n}$ is a symmetric matrix, $S_k \in \Re^{m\times m}$ is a diagonal matrix, $\hat{\epsilon}_k \in \Re^n$ and $\bar{\epsilon}_k \in \Re^m$ are two vectors. Therefore we can view all these linear systems as perturbated system from the equation (2.3).

If x_k is close to a solution where the second order sufficient condition holds and if λ_k is close to the lagrange multiplier at the solution, we can show that

$$Z_k^T W(x_k \lambda_k) Z_k \tag{3.2}$$

is positive definite. Therefore throughout this section we use this assumption.

Assumption 3.1

1. *The matrix $Z_k T W(x_k \lambda_k) Z_k$ is positive definite for all k.*
2. *A_k is full column rank, namely Rank$(A_k) = m$ for all k.*

The following result is well known (for example, see Fletcher[4]).

Theorem 3.2 *Under the Assumption 3.1, the matrix*

$$\begin{bmatrix} W(x_k, \lambda_k) & -A_k \\ -A_k^T & 0 \end{bmatrix} \tag{3.3}$$

is nonsingular.

We found in the previous section, the perturbation matrix S_k is always a negative semi-definite matrix. Actually in all the cases, S_k is a diagonal matrix, whose diagonal elements are non-positive.

Before presenting our next theorem, we need to establish the following lemma:

Lemma 3.3 *Assume that $R \in \Re^{m\times m}$ is an nonsingular matrix, that $T \in \Re^{m\times m}$ is symmtric and positive semi-definite $S \in \Re^{m\times m}$ is symmetric and negative semi-definite, then the matrix*

$$\begin{bmatrix} T & R \\ R^T & S \end{bmatrix} \tag{3.4}$$

is nonsingular, and it has m positive eigenvalues and m negative eigenvalues.

Proof. Choose a positive number t such that $t^2 > \|S\|_2$. Consider all the vectors in $\Re^{2m}$ in the following form:

$$z = \begin{bmatrix} ty \\ \frac{1}{t}Ry \end{bmatrix}, \qquad \forall y \in \Re^m. \tag{3.5}$$

It is easy to see that

$$z^T \begin{bmatrix} T & R \\ R^T & S \end{bmatrix} z = t^2 y^T T y + 2\|Ry\|^2 + \frac{1}{t^2} y^T R^T S R y > 0. \tag{3.6}$$

for any nonzero vector $y \in \Re^m$. Because the vectors defined by (3.5) span an m-dimensional subspace in $\Re^{2m}$, it follows from (3.6) that the matrix (3.4) has at least m positive eigenvalues. Applying the same arguments to matrix

$$\begin{bmatrix} -T & -R \\ -R^T & -S \end{bmatrix}, \tag{3.7}$$

we can see that matrix (3.4) has at least m negative eigenvalues. □

The following theorem implies that the nonsingularity of the coefficient matrix of the linear systems given in the previous section.

Theorem 3.4 *Assume that $Rank(A_k) = m$, that $B_k \in \Re^{n\times n}$ is symmetric and positive semi-definite and that $Z_k^T B_k Z_k$ is positive definite, then for any symmetric and negative semidefinite matrix $S_k \in \Re^{m\times m}$ the matrix*

$$\begin{bmatrix} B_k & -A_k \\ -A_k^T & S_k \end{bmatrix} \tag{3.8}$$

is nonsingular. Moreover, the matrix (3.8) has exactly n positive eigenvalues and m negative eigenvalues.

Proof. Let the QR factorization of A_k is given by (2.63). Define the following orthogonal matrix

$$\bar{Q}_k = \begin{bmatrix} Q_k & 0 \\ 0 & I \end{bmatrix} \in \Re^{(n+m)\times(n+m)}. \tag{3.9}$$

It is easy to see that

$$\begin{aligned} \bar{Q}_k^T \begin{bmatrix} B_k & -A_k \\ -A_k^T & S_k \end{bmatrix} \bar{Q}_k &= \begin{bmatrix} Y_k^T B_k Y_k & Y_k^T B_k Z_k & -R_k \\ Z_k^T B_k Y_k & Z_k^T B_k Z_k & 0 \\ -R_k^T & 0 & S_k \end{bmatrix} \\ &= \hat{P}_k \begin{bmatrix} Y_k^T B_k Y_k - Y_k^T B_k Z_k (Z_k^T B_k Z_k)^{-1} Z_k^T B_k Y_k & 0 & -R_k \\ 0 & Z_k^T B_k Z_k & 0 \\ -R_k^T & 0 & S_k \end{bmatrix} \hat{P}_k^T, \end{aligned} \tag{3.10}$$

where

$$\hat{P}_k = \begin{bmatrix} I & Y_k^T B_k Z_k (Z_k^T B_k Z_k)^{-1} & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \tag{3.11}$$

is a nonsingular matrix. Because B_k is positive semi-definite and $Z_k^T B_k Z_k$ is positive definite, it is obvious that

$$Y_k^T B_k Y_k - Y_k^T B_k Z_k (Z_k^T B_k Z_k)^{-1} Z_k^T B_k Y_k \tag{3.12}$$

is also positive semi-definite. It follows from (3.10), the positive semi-definiteness of (3.12), the positive definiteness of $Z_k^T B_k Z_k$ and Lemma 3.3 that the matrix (3.8) has exactly n positive eigenvalues and m negative eigenvalues. □

Unfortunately the condition that B_k is positive semi-definite can not be relaxed. For example, in the linear system derived from the CDT subproblem, the coefficient matrix

$$\begin{bmatrix} B_k + \tau I & -A_k \\ -A_k^T & -\frac{1}{\mu} I \end{bmatrix} \tag{3.13}$$

has the same inertia (see [5]) as

$$\begin{bmatrix} B_k + \tau I + \mu A_k A_k^T & 0 \\ 0 & -\frac{1}{\mu} I \end{bmatrix}. \tag{3.14}$$

And it is showed by Yuan[10] that

$$B_k + \tau I + \mu A_k A_k^T \tag{3.15}$$

can have one negative eigenvalue even when the second order sufficient condition holds. This implies that (3.14) can have $m+1$ negative eigenvalue, which means that (3.13) can have more than m negative eigenvalue if B_k is not positive semidefinite.

If $S_k = 0$ in (3.1), the linear system is equivalent to

$$\begin{bmatrix} W(x_k, \lambda_k) + T_k & -A_k \\ -A_k^T & 0 \end{bmatrix} \begin{pmatrix} d \\ \eta - \hat{\lambda} \end{pmatrix} = \begin{pmatrix} -g_k + A_k^T \hat{\lambda} + \hat{\epsilon}_k \\ c_k + \bar{\epsilon}_k \end{pmatrix}, \tag{3.16}$$

for any $\hat{\lambda} \in \Re^m$. This shows that for linear systems that have a zero S_k, adding any vector in the range space of A_k does not alter the solution d. For example, it is convenient to choose $\hat{\lambda}$ that minimizes $\|g_k - A_k \lambda\|$ so that the right hand side of the linear system (3.16) is bounded above by $O(\|x_k - x^*\|)$ if x_k is close to a solution x^* and if $\|\hat{\epsilon}_k\| + \|\bar{\epsilon}_k\| = O(\|x_k - x^*\|)$.

If $S_k \neq 0$, it is normally a diagonal matrix whose diagonal elements are all negative. In that case, if we want to have $g_k - A_k \hat{\lambda}$ instead of g_k in the right hand of (3.1), the linear system can be written as

$$\begin{bmatrix} W(x_k, \lambda_k) + T_k & -A_k \\ -A_k^T & S_k \end{bmatrix} \begin{pmatrix} d \\ \eta - \hat{\lambda} \end{pmatrix} = \begin{pmatrix} -g_k + A_k^T \hat{\lambda} + \hat{\epsilon}_k \\ c_k - S_k \hat{\lambda} + \bar{\epsilon}_k \end{pmatrix}. \tag{3.17}$$

A reasonable choice for $\hat{\lambda}$ is that it minimizes

$$\|g_k - A_k^T \lambda\| + \|c_k - S_k \lambda\|. \tag{3.18}$$

Assuming that $\|\hat{\epsilon}_k\| + \|\bar{\epsilon}_k\| \to 0$, we should have that $S_k \to 0$ in order to ensure that the right hand side of (3.17). However, if $\|\bar{\epsilon}_k\| \to 0$ and if $\hat{\epsilon}_k = A_k \lambda_k + o(1)$, the right hand side of the linear system (3.1) converges to 0 provided λ_k is a good approximation to the Lagrange multiplier.

4. Update Linear Systems

In most methods, the Hessian of the Lagrange function will not be computed. Therefore, the linear system actually solved will be in the form

$$\begin{bmatrix} B_k & -A_k \\ -A_k^T & 0 \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} u_k \\ v_k \end{pmatrix}, \tag{4.1}$$

or

$$\begin{bmatrix} B_k + \tau I & -A_k \\ -A_k^T & \Sigma_k \end{bmatrix} \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} u_k \\ v_k \end{pmatrix}, \tag{4.2}$$

where $\tau \geq 0$ and $\Sigma_k = Diag[(\sigma_k)_1, (\sigma_k)_2, ..., (\sigma_k)_m]$ with $(\sigma_k)_i < 0$ for all i. B_k is a quasi-Newton matrix which is updated by either a rank-1 or a rank-2 formular. For example, the symmetric rank 1 update is given by

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)}{(y_k - B_k s_k)^T s_k}, \tag{4.3}$$

where

$$\begin{aligned} s_k &= x_{k+1} - x_k, & (4.4) \\ y_k &= g_{k+1} - g_k - \sum_{i=1}^{m} (\lambda_k)_i [\nabla c_i(x_{k+1}) - \nabla c_i(x_k)]. & (4.5) \end{aligned}$$

Here we only consider (4.1). Denote

$$H_k = \begin{bmatrix} B_k & -A_k \\ -A_k^T & 0 \end{bmatrix}. \tag{4.6}$$

Suppose that B_{k+1} is obtained by adding a rank-2 matrix to B_k. then H_{k+1} is a rank-2(m+1) changes from H_k. Namely we have

$$H_{k+1} - H_k = \begin{bmatrix} B_{k+1} - B_k & 0 \\ 0 & 0 \end{bmatrix} + \sum_{i=1}^{m} \left[\hat{y}_k e_i^T + e_i \hat{y}_k^T \right], \tag{4.7}$$

where

$$\hat{y}_k = \begin{pmatrix} 0 \\ \nabla c_i(x_{k+1}) - \nabla c_i(x_k) \end{pmatrix}. \tag{4.8}$$

Since the linear system (4.1) has to be solved in every iteration of an optimization algorithm, it is very natural for us to be very interested in the following question.

Problem 4.1 *If the linear system (4.1) has already been solved in the iteration k, how to solve the similar equation quickly for the next iteration using the rank-2(m+1) update relation (4.7)?*

In practical implementations, we may need to use some kind of pre-conditioning technique to solve (4.1). Namely we construct a matrix $P_k \in \Re^{(n+m)\times(n+m)}$ such that P_k is some kind of approximation to H_k^{-1}. Therefore it is very desirable to have a good answer to the following question.

Problem 4.2 *Suppose that we have a pre-conditioner matrix P_k, how can we quickly obtain a suitable pre-conditioner matrix P_{k+1} using relation (4.7)?*

One paticular answer to the above question is to use the Sherman-Morrison-Woodbury formula:

$$(A + UV^T)^{-1} = A^{-1} - A^{-1}U(I + V^T A^{-1} U)^{-1} V^T A^{-1}. \tag{4.9}$$

Since $P_k \approx H_k^{-1}$, it is reasonable to choose P_{k+1} by

$$P_{k+1}^{-1} = P_k^{-1} + [H_{k+1} - H_k]. \tag{4.10}$$

Because H_{k+1} is a rank-2(m+1) update from H_k, using the above relation we can apply the Sherman-Morrison-Woodbury formula to update P_{k+1} from P_k. However, it would be better if we can find an even better solution than this approach.

4. Conclusion

In this paper, we have discussed some linear systems associated with numerical methods for constrained optimization. It is interesting to study the properties of these linear systems, and try to construct new methods by proposing different linear systems which have similar structures. Indeed, if a search direction(in a line search type method) or a trial step (in a trust region method) is a superlinear convergence step, it must be some kind of approximation to the SQP step. Therefore, this step should satisfy (2.6) approximately. Thus, we may consider general methods that have the following form:

$$\left(\begin{bmatrix} B_k & -A_k \\ -A_k^T & 0 \end{bmatrix} + E_k \right) \begin{pmatrix} d \\ \eta \end{pmatrix} = \begin{pmatrix} -g_k + A_k \lambda_k \\ c_k \end{pmatrix}. \tag{5.1}$$

The matrix E_k should have the following properties: 1. it should converge to zero when the iterate points converge to a solution; 2. the linear system (5.1) should be better conditioned than the original system (2.6); 3. the solution d obtained from (5.1) should be easily verified to be an descent direction of certain merit function. We believe that to construct new efficient methods by proposing different E_k is an interesting subject to study.

References

[1] M.R. Celis, J.E. Dennis, Jr. and R.A. Tapia, A trust region algorithm for nonlinear equality constrained optimization, in: P.T. Boogs, R.H. Byrd and R.B. Schnabel, eds., *Numerical Optimization* (SIAM, Philadephia, 1985) pp. 71-82.

[2] T. F. Coleman and Y. Li, A trust region and affine scaling interior point method for nonconvex minimization with linear inequality constraints, *Math. Prog.* 88(2000) 1-32.

[3] J.E. Dennis and J.J. Moré, Quasi-Newton method, motivation and theory, *SIAM Review* 19(1977) 46-89.

[4] R. Fletcher, *Practical Methods of Optimization, 2nd. Ed.* (John Wiley and Sons, Chichester, 1987).

[5] G.H. Golub and C.F. Van Loan, *Matrix Computations, Third Ed.*, (Johns Hopkins, Baltimore and London, 1996).

[6] J. Nocedal and M.L. Overton, Projected Hessian update algorithms for nonlinear constrained optimization, *SIAM J. Numer. Anal.* 22(1985) 821-850.

[7] M.J.D. Powell and Y. Yuan, A trust region algorithm for equality constrained optimization, *Math. Prog.* 49(1991) 189-211.

[8] D.F. Shanno and R.J. Vanderbei, Interior-point methods for nonconvex nonlinear programming: orderings and higher-order methods, *Math. Prog.* 87(2000) 303-316.

[9] R.J. Vanderbei and D.F. Shanno, An interior point algorithm for nonconvex nonlinear programming, *Comput. Optim. Appl.* 13(1999) 231-252.

[10] Y. Yuan, On a subproblem of trust region algorithms for constrained optimization, *Math. Prog.* 47(1990) 53-63.