The 7th Balkan Conference on Operational Research
"BACOR 05"
Constanta, May 2005, Romania

THE GENERALIZATION OF THE INTERIOR-POINT METHOD FOR SOLVING THE CONVEX QUADRATIC PROGRAMMING PROBLEMS

POPOVICIU IOAN
COLȚESCU ION
CIOBANU CAMELIA
VASILIU PAUL

Naval Academy "Mircea cel Batran", Constanta, Romania

Abstract
In this paper is described how to efficiently solve a convex quadratic programming problems using a generalization of the interior –point method.

Keywords*: interior-point, convex, quadratic, optimal control*

The interior-point method finds a solution for the Karush-Kuhn-Tucker (KKT) conditions of the quadratic problem, being a alternative to Newton method. In the interior-point method at each iteration to compute the searching direction which leads to solving a linear system.

M is a square, positive semidefinite matrix, $M \in R^{n \times n}$ and a is a vector $q \in R^n$. The problem is to find vectors z, x and s such that

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} z \\ x \end{bmatrix} + \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 0 \\ s \end{bmatrix}, \quad (1)$$

$$x \geq 0, s \geq 0, \; x^T s = 0 . \quad (2)$$

The name of the problem is *realizable interior-point method* **(MRPI)**:

Here M_{11} and M_{22} are square submatrices of M with dimensions n_1 and n_2 respectively, and the vector q is partitioned accordingly. The condition (1) is the condition of *infeasible,* and $x^T s = 0$ is the condition of *complementarity*..

The realizable interior-point method starts at point (z^0, x^0, s^0), where $x^0 > 0$, $s^0 > 0$, but possibly infeasible with respect the constraints (1). All iterates

(z^k, x^k, s^k) retain the positivy properties $x^k > 0,\ \ s^k > 0$, but the infeasibilities and the complementary gap defined by:

$$\mu_k = (x^k)^T s^k / n_2 \tag{3}$$

are gradually reduced to zo zero as $k \to \infty$. Each step of the algorithm is a modified Newton step for the system of nonlinear equations defined feasibility conditions (1) and the complementarity $x_i s_i = 0, i = 1,2,\ldots,n_2$.

We can write this system as:

$$F(z,x,s) = \begin{bmatrix} M_{11}z + M_{12}x + q_1 \\ M_{21}z + M_{22}x - s + q_2 \\ XSe \end{bmatrix} = 0, \tag{4}$$

where we have used the notation

$$X = \operatorname{diag}(x_1, x_2, \ldots, x_{n_2}),\ S = \operatorname{diag}(s_1, s_2, \ldots, s_{n_2}).$$

The algorithm has the following form:

Given: (z^0, x^0, s^0), with $x^0 > 0, s^0 > 0$.

for $k = 0,1,2,\ldots$

- For some $\sigma_k \in (0,1)$ solve:

$$\begin{bmatrix} M_{11} & M_{12} & 0 \\ M_{21} & M_{22} & -I \\ 0 & S^k & X^k \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -r_1^k \\ -r_2^k \\ -X^k S^k e + \sigma_k \mu_k e \end{bmatrix} \tag{5}$$

to obtain $(\Delta z^k, \Delta x^k, \Delta s^k)$, where

$$r_1^k = M_{11}z^k + M_{12}x^k + q_1,$$
$$r_2^k = M_{21}z^k + M_{22}x^k - s^k + q_2,$$
$$e = (1,1,\ldots,1)^T,$$

- set:

$$(z^{k+1}, x^{k+1}, s^{k+1}) = (z^k, x^k, s^k) + \alpha_k (\Delta z^k, \Delta x^k, \Delta s^k), \tag{6}$$

for some $\alpha_k \in (0,1]$ that retains $(x^{k+1}, s^{k+1}) > 0$.

repeat

Note that (5) differs from the pure Newton step for (4) only because of the term $\sigma_k \mu_k e$ on right-hand side. This term plays a stabilizing role, ensuring that the algorithm converges steadily to a solution (1)-(2). The only two parameters to choose in algorithm are the scalars σ_k and α_k, .The convergence analysis leaves the choice of σ_k relatively unfettered (it is often confined to the range $[\sigma, 0.8]$, where σ is a fixed parameter, typically $\sigma = 10^{-3}$) but α_k is required to satisfy the following conditions.

a) The ratios $\|r_1^k\| / \mu_k$ and $\|r_2^k\| / \mu_k$ should decrease monotonically with k.

b) $x_i s_i$, $i = 1,2,\ldots,n_2$ should remain bounded away from zero for all i and all k.

c) μ_k should decrease at each iterate.

In practical implementations of algorithm, α_k often is chosen via the following simple heuristic. First we set $\alpha_k^{\max}$ to be the supremum of the following set:

$$\{\alpha \in (0,1] \mid (z^k, x^k, s^k) + \alpha(\Delta z^k, \Delta x^k, \Delta s^k) > 0\}. \tag{7}$$

Then we set

$$\alpha_k = \min(1, 0.995 * \alpha_k^{\max}). \tag{8}$$

The major operation to be performed at each step of algorithm is the solution of the linear system (5). The matrix in this system obviously has a lot of structure due to the presence of the zero blocks and the diagonal components I, S^k and X^k. Additionally, the matrix M is sparse in most cases of practical interest, including our motivating problem (1) , so sparse matrix factorizations are called for.

The first step in solving (5) is to eliminate the Δs component. Since the diagonal elements of X^k are positive, we can rearrange the last block in (5) to obtain

$$\Delta s = (X^k)^{-1}(-X^k S^k e + \sigma_k \mu_k e - S^k \Delta x^k) = -s_k + (X^k)^{-1}(\sigma_k \mu_k e - S^k \Delta x^k).$$

By substituting into the first two rows in (5), we obtain:

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} + (X^k)^{-1} S^K \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta x \end{bmatrix} = \begin{bmatrix} -r_1^k \\ -r_2^k - s_k + \sigma_k \mu_k (X^k)^{-1} e \end{bmatrix}. \tag{9}$$

In most cases, some of the partitions $M_{11}, M_{12}, M_{21}, M_{22}$ are zero or diagonal or have some other simple structure, so further reduction of the system (9) is usually possible. This phenomenon happens, for instance, when (1), (2) is derived from a linear or quadratic program.

We now show how convex quadratic programming problems can be expressed in the form (1), (2) and solved via algorithm MRPI. We consider the following general convex quadratic program:

$$\min_z \frac{1}{2} z^T Q z + c^T z \tag{10}$$

$$Hz = h,$$

$$Gz \le g,$$

where Q is a symmetric positive semidefinite matrix.

The KKT conditions for this system are:

$$Qz + H^T \zeta + G^T \lambda = -c, \tag{11}$$

$$Hz = h,$$

$$Gz + t = g,$$

$$t^T \lambda = 0,$$

$$t \ge 0, \lambda \ge 0.$$

The following identifications confirm that the system (11) can be expressed in the form (1), (2):

$$M_{11} = \begin{bmatrix} Q & H^T \\ -H & 0 \end{bmatrix}, M_{12} = \begin{bmatrix} G^T \\ 0 \end{bmatrix}, M_{21} = \begin{bmatrix} -G & 0 \end{bmatrix}, M_{22} = O ,$$

$$q_1 = \begin{bmatrix} c \\ h \end{bmatrix}, q_2 = g ,$$

$$z \leftarrow \begin{bmatrix} z \\ \zeta \end{bmatrix}, x \leftarrow \lambda, s \leftarrow t .$$

The reduced form (9) of the linear system to be solved at each iteration of algorithm is

$$\begin{bmatrix} Q & H^T & G^T \\ -H & 0 & 0 \\ -G & 0 & (\Lambda^k)^{-1} T^k \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \zeta \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -r_c^k \\ -r_h^k \\ -r_g^k - t^k + \sigma_k \mu_k (\lambda^k)^{-1} e \end{bmatrix}, \tag{12}$$

where $\mu_k = (t^k)^T \lambda^k / m$, m is a number of inequality constraints in (10). It is customary to multiply the last two block rows in (12) by –1, so that the coefficient matrix is symmetric indefinite.

$$\begin{bmatrix} Q & H^T & G^T \\ H & 0 & 0 \\ G & 0 & -(\Lambda^k)^{-1} T^k \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \zeta \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -r_c^k \\ r_h^k \\ r_g^k + t^k - \sigma_k \mu_k (\lambda^k)^{-1} e \end{bmatrix}. \tag{13}$$

Since $(\Lambda^k)^{-1} T^k$ is diagonal with positive diagonal elements, we can eliminate $\Delta\lambda$ from (13) to obtain an even more compact form:

$$\begin{bmatrix} Q + G^T \Lambda^k (T^k)^{-1} G & H^T \\ H & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \zeta \end{bmatrix} = \begin{bmatrix} -r_c^k + G^T \left[\Lambda^k (T^k)^{-1} r_g^k + \lambda^k - \sigma_k \mu_k (T^k)^{-1} e \right] \\ r_h^k \end{bmatrix}. \tag{14}$$

EXAMPLE

We consider the problem with the following transfer function:

$$\mathrm{H}(\mathrm{s}) = \frac{0.0011}{\mathrm{s}^4 + 0.3466\mathrm{s}^3 + 0.1155\mathrm{s}^2 + 0.0083\mathrm{s} + 0.0001}$$

with the poles: -0.0158, -0.0718 şi -0.1295 ± 0.2736i.

For transfer function H(s) results the following standard realisation:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.0001 & -0.0083 & -0.1155 & -0.3466 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad c = \begin{bmatrix} 0.0011 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

coresponding to system:

$$\dot{x} = Ax + bu$$

$$y = c^T x$$

with the performance criterion

$$J = \frac{1}{2} u^2 ,$$

To obtain the discret model we calculate the matrix Φ and Γ with the sampling period T=2s.

$$\Phi = e^{AT} \approx I + AT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + 2 \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.0001 & -0.0083 & -0.1155 & -0.3466 \end{bmatrix},$$

$$\Phi = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 1 & 2 \\ -0.0002 & -0.0166 & -0.1310 & 0.3068 \end{bmatrix},$$

$$\Gamma = \int_0^T e^{Ap} b dp = \int_0^2 \begin{bmatrix} p & 2p & 0 & 0 \\ 0 & p & 2p & 0 \\ 0 & 0 & p & 2p \\ -0.0002p & -0.0166p & -0.1310p & 0.3068p \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} dp$$

$$= \int_0^2 \begin{bmatrix} 0 \\ 0 \\ 2p \\ 0.3068p \end{bmatrix} dp = \begin{bmatrix} 0 \\ 0 \\ 4 \\ 0.6136 \end{bmatrix}.$$

Then the discrete model of problem becomes:

$$x(k+1) = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 1 & 2 \\ -0.0002 & -0.0166 & -0.1310 & 0.3068 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 0 \\ 4 \\ 0.6136 \end{bmatrix} u(k),$$

$$y(k) = \begin{bmatrix} 0 & 0 & 0 & 0.0011 \end{bmatrix} x(k) \;,$$

with the performance criterion:

$$J = \frac{1}{2} \sum_{k=0}^{N-1} u^2(k) .$$

The command is restricted with:

$$\frac{1}{2N}\sum_{k=0}^{N-1} u^2(k) \le 0.007 \, .$$

Because the matrix $(\Gamma \quad \Phi\Gamma)$ is nonsingular we can formulate the equivalent quadratic programming problem. We define the vectors and matrix as below:

$$z_1 = \begin{bmatrix} u_0 \\ x_1 \\ u_1 \\ \vdots \\ x_{N-1} \\ u_{N-1} \\ x_N \end{bmatrix}, \quad z_2 = \begin{bmatrix} y_0 \\ x_0 \\ y_1 \\ x_1 \\ \vdots \\ y_{N-1} \\ x_{N-1} \end{bmatrix}, \quad h_1 = \begin{bmatrix} -\Phi x_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad h_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$H_1 = \begin{bmatrix} \Gamma & -I & & & & & \\ & \Phi & \Gamma & -I & & & \\ & & & \ddots & & & \\ & & & & \Phi & \Gamma & -I \end{bmatrix}, H_2 = \begin{bmatrix} I & -c & & & & \\ & & I & -c & & \\ & & & \ddots & & \\ & & & & I & -c \end{bmatrix},$$

$$z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}, \quad H = \begin{bmatrix} H_1 & 0 \\ 0 & H_2 \end{bmatrix}, \quad h = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix},$$

$$\tilde{Q} = \begin{bmatrix} I & & & & & & \\ & 0 & & & & & \\ & & I & & & & \\ & & & 0 & & & \\ & & & & \ddots & & \\ & & & & & I & \\ & & & & & & 0 \end{bmatrix}, \quad Q = \begin{bmatrix} \tilde{Q} & 0 \\ 0 & 0 \end{bmatrix},$$

$$\tilde{G} = \begin{bmatrix} I & & & & & & \\ & 0 & & & & & \\ & & I & & & & \\ & & & 0 & & & \\ & & & & \ddots & & \\ & & & & & I & \\ & & & & & & 0 \end{bmatrix}, \quad G = \begin{bmatrix} \tilde{G} & 0 \\ 0 & 0 \end{bmatrix}, \quad g = \begin{bmatrix} 0.007 * N \\ 0.007 * N \\ 0.007 * N \\ \vdots \\ 0.007 * N \\ 0.007 * N \\ 0.007 * N \end{bmatrix}$$

Then, the discrete model is equivalent with the following quadratic problem:

$$\min_z \frac{1}{2} z^T Q z$$
$$Hz = h$$
$$Gz \le g\ .$$

The KKT conditions for this problem are:

$$Qz + H^T\zeta + G^T\lambda = 0\,,$$
$$Hz = h\,,$$
$$Gz + t = g\ ,$$
$$t \ge 0,\ \lambda \ge 0,\quad t^T\lambda = 0\,,$$

and the following substitutions:

$$M_{11} = \begin{bmatrix} Q & H^T \\ -H & 0 \end{bmatrix},\quad M_{12} = \begin{bmatrix} G^T \\ 0 \end{bmatrix},\quad M_{21} = \begin{bmatrix} -G & 0 \end{bmatrix},\quad M_{22} = 0\ ,$$
$$q_1 = \begin{bmatrix} 0 \\ h \end{bmatrix},\quad q_2 = g\,,\quad z \leftarrow \begin{bmatrix} z \\ \zeta \end{bmatrix},\quad x \leftarrow \lambda\,,\quad s \leftarrow t\,,$$

shows that the KKT conditions system is a problem which can be solve with the MRPI algorithm.

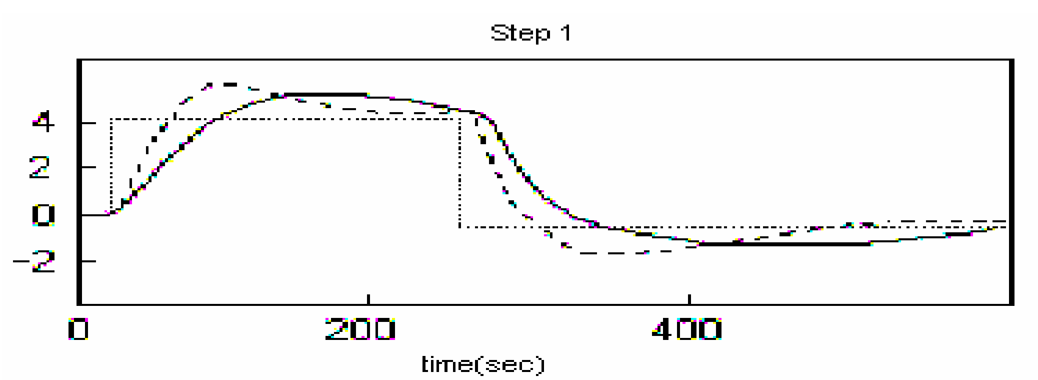
The linear system which has to be solved at each iteration is:

$$\begin{bmatrix} Q & H^T & G^T \\ -H & 0 & 0 \\ -G & 0 & (\Lambda^k)^{-1}T^k \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta\zeta \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} -r_c^k \\ -r_h^k \\ -r_g^k - t^k + \sigma_k\mu_k(\lambda_k)^{-1}e \end{bmatrix}.$$

By eliminating $\Delta\lambda$ result the linear system:

$$\begin{bmatrix} Q + G^T\Lambda^k(T^k)^{-1}G & H^T \\ H & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta\zeta \end{bmatrix} = \begin{bmatrix} -r_c^k + G^T\left[\Lambda^k(T^k)^{-1}r_g^k + \lambda^k - \sigma_k\mu_k(T^k)^{-1}e\right] \\ r_h^k \end{bmatrix}$$

The solution after few iterations it is shown in Chart 1.



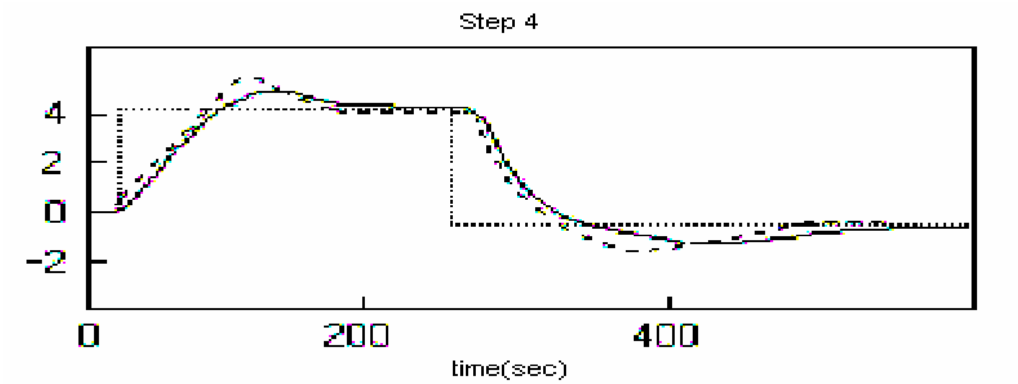


Chart 1 After five iterations the solution is near to the wanted value

BIBLIOGRAPHY

[1] Bini D.,Pan V., Polynomial and Matrix Computations: Fundamental Algorithms, Birkhauser, Boston, 1994;

[2] Chmielewski D., Manousiouthakis V., On constrained infinite-time linear quadratic optimal control, System & Control Letters, 29:121-129, 1996;

[3] Drexler M., Application of iterative methods in interior-point methods for robust optimal control of linear discrete-time systems, SIAM, 1998;

[4] Glowinski R., Lions J.L., Exact and approximate contollability for distributed parameters systems, Acta numerica, 159-333, Cambridge University Press, 1996;

[5] Lazar C., Pastravan O., Poli E., Schonberger F., Conducerea asistata de calculator a proceselor tehnice, Ed. Matrix, Bucuresti, 1996;

[6] Lions J.L., Optimal Control of Systems Governed by Partial Differential Equations, Springer Verlag, Berlin, 1971;

[7] Mangasarin O.L., Nonlinear Programming, McGraw-Hill, New York, 1969;

[8] Mehrotra S., On the implementation of a primal-dual interior point method, SIAM Journal on Optimization, 2:576-601, 1992;

[9] Nesterov Y., Nemirovsky A., Interior-Point Polynomial Methods in Convex Programming, SIAM, Philadelphia, 1994;

[10] Rao C.V., Wright S.J., Rawlings J.B., Application of interior-point methods to model predictive control, Preprint ANL/MCS-P664-0597, Argonne National Laboratory, 1997;

[11] Wright S.J., Interior-point methods for optimal control of discrete-time systems, Journal of Optimization Theory and Applications, 77:161-187, 1993;

[12] BPMPD user’s manual, Version 2.20, Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, 2000;