Advanced Engineering Forum Online: 2011-12-22
ISSN: 2234-991X, Vols. 2-3, pp 801-806
doi:10.4028/www.scientific.net/AEF.2-3.801

Optimization Algorithm of Matrix Riccati Differential Equation and Application

Xianglin Hou[1,a], Desheng Huang[1,b], Cong Chen[1,c]

[1]School of Traffic & Mechanical Engineering, Shenyang Jianzhu University, Shenyang 110168
[a] drhouxl @ tom.com, [b] snow800827@yahoo.cn, [c] chencong253@126.com

Keywords: matrix Riccati differential equation, dynamic design Variables, optimization method, optimal control, linear quadratic control

Abstract. To the matrix Riccati differential equation, based on dynamic design Variables Optimization Method, making unknown element of Riccati matrix as design variables, square sum of defined summation matrix element as objective function, a kind of new optimization Method about element of Riccati matrix $n \times n$ orders is built. Universal program is formed. Practical examples are computed. Effectiveness is shown through result. The method is a new thinking for computing high order matrix Riccati Differential equation and obtaining control parameters.

Introduction

Matrix Riccati differential equation is an important equation acquiring feedback control parameter [1-2].

Many algorithm in both theoretical and applicable problem are studied [3-5] . Theoretically, only under the special condition, analytical solutions can be acquired [6-7]. Numerical solution of the elements in Riccati matrix differential equation could be obtained accurately [8-10]. Applying mathematics software and using numerical analysis theory, the matrix elements in Riccati differential equations could be computed by iterative method. Therefore, with calculating order and complication increase, calculating result becomes worse. A series of problems such as convergence and accuracy are existed and need be further studied.

Dynamic Design Variables Optimization Method [11] (DVOM) is put forward by Author. The method is successfully applied into computing unknown quantities of the restraint forces in any statics systems and the currents in any DC [11-13]. It is also used establish the solution to Matrix Riccati differential equation problems well [14]. In the following text, the precise program computation of matrix Riccati differential equation will be discussed.

Optimization principle of solving matrix Riccati differential equation

Matrix Riccati differential equation in control problem. In general the dynamic equation of a linear system is Expressed as: $\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases}$, here $x \in R^n, u \in R^r, A \in R^{n\times n}, B \in R^{n\times r}, C \in R^{n\times m}$.

According to maximum principle, through the optimal value analysis of state regulator's target function: $J = \int_{t_0}^{\infty} [x^T(t)Qx(t) + u^T(t)Ru(t)]dt$, here $Q \in R^{n\times n}, R \in R^{r\times r}$, a state feedback control $u^*(\vec{x},t) = -R^{-1}B^T P\vec{x}$ can be selected to make J access to a minimum, in which P is a symmetry Riccati matrix. P could be satisfied with matrix Riccati differential equation:

$$\dot{P} + PA + A^T P - PBR^{-1}B^T P + Q = 0 \tag{1}$$

In Eq.1 state matrix A, control matrix B, weighted matrix Q and R are known matrix. Elements of P matrix are unknown and changed with time. When time is given, Eq.1 is differential equations with $n(n+1)/2$. Dynamic design variables optimization can be used to compute solve Eq.1.

Outline of dynamic design variables optimization [11]. The core of dynamic design variable optimization is converting complex solving problem of unknown design variables about analyzed problem into optimization problem of computation and analysis about dynamic design variables. Compared with static optimization problem, dynamic design variables can be sorted and framework objective function is dynamically formed according to different input conditions. Dynamic design variable optimization method can effectively used to solve the theoretical and engineering practical problems with a large number of unknown variables, and the variables are subjected to certain distribution rules. The unknown elements of Riccati matrix is satisfied with certain distribution rules, so that the basic principle of the dynamic design variables optimization could be used to complete the unknown elements of matrix Riccati differential equation rapidly.

Dynamic design variables optimization about matrix Riccati differential equation. In matrix Riccati differential Eq.1, let $A=(A_1,A_2,\cdots,A_n), P=(P_1,P_2,\cdots,P_n)$, It can be converted into non-linear equations composed of n^2 equations:

$$\dot{P}_{i,j}+A_i^T P_j+P_i^T A_j-P_i^T BR^{-1}B^T P_j+Q_{i,j}=0,\quad i,j=1,2,\cdots,n \tag{2}$$

Difference principle can be used and $\dot{P}_{i,j}$ can be expressed:

$$\dot{P}_{i,j}\approx\frac{P_{i,j}(t)-P_{i,j}(t-\Delta t)}{\Delta t},i,j=1,2,\cdots n \tag{3}$$

Computing Eq. 2 is equivalent to searching for, $p_{i,j},i,j=1,2,\cdots n$ to satisfy:

$$Fp_{i,j}=\dot{P}_{i,j}+A_i^T P_j+P_i^T A_j-P_i^T BR^{-1}B^T P_j+Q_{i,j}=0,\quad i,j=1,2,\cdots,n \tag{4}$$

$\{Fp_{i,j}\}$ is defined as a summing matrix. $p_{i,j}$ is independent element and expressed as:

$$\begin{matrix} p_{1,1} & p_{1,2} & p_{1,3} & \cdots & p_{1,n} \\ & p_{2,2} & p_{2,3} & \cdots & p_{2,n} \\ & & p_{3,3} & \cdots & p_{3,n} \\ & & & \cdots & \cdots \\ & & & & p_{n,n} \end{matrix} \tag{5}$$

A total of independent element is: $N=n(n+1)/2$. Dynamic design variables optimization problem is established:

$$\min\{f(\vec{z})\} \tag{6}$$

Design variables are: $z_i, i=1,2,\cdots,N$. Objective function is:

$$f(\vec{z})=\sum_{i=1}^{n}\sum_{j=1}^{n}(Fp_{i,j})^2=\sum_{i=1}^{n}\sum_{j=1}^{n}(\dot{P}_{i,j}+A_i^T P_j+P_i^T A_j-P_i^T BR^{-1}B^T P_j+Q_{i,j})^2 \tag{7}$$

Sorting relation between elements of Riccati matrix and design variables. $p_{1,1},p_{1,2},p_{1,3},\cdots,$ $p_{1,n};p_{2,2},p_{2,3,}\cdots p_{2,n};p_{3,3}\cdots,p_{3,n},\cdots p_{n,n}$ are the independent elements in upper triangle matrix of P, $p_{i,j}=z_{n+(n-1)+[n-(i-2)]+(j-i)+1}=z_{\frac{(2n-i+2)(i-1)}{2}+(j-i)+1}, i=1,2,\cdots,n; j=i,i+1,\cdots,n$ could be obtained by sorting with $z_i, i=1,2,\cdots,N$. $p_{j,i}=p_{i,j}$ $,i=1,2,\cdots,n, j=1,2,\cdots,i-1$ could be obtained by the transpose of upper triangle matrix in P.

Dynamic process of objective functions. The objective functions are described in pseudo program method in Table1 when time is given.

Table 1 Pseudo program and meaning of dynamic process of objective functions

pseudo program	meaning
$f=0$	Setting up an initial function value
for $(i=1\ to\ n)$ { for$(j=i\ to\ n)$ $\{P(i,j)=z((2n-i+2)(i-1)/2+(j-i)+1)\}$ }	Sorting the elements of upper triangle P matrix and design variables
for $(i=1$ to $n)$ { for$(j=1$ to i-1) $\{P(i,j)=P(j,i)$ } }	Forming the elements of lower triangle P matrix
for $(i=1$ to $n)$ { for$(j=1$ to $n)$ $\{fp(i,j)=0$ }	Setting up initial function values for the summing matrix
for $(i=1$ to $n)$ { for$(j=1$ to $n)$ { for$(k=1$ to $n)$ $\{fp(i,j)=fp(i,j)+(P(i,j)-P0(i,j))/dt\}$ }}	Accumulating $\dot{P}$
for $(i=1$ to $n)$ { for$(j=1$ to $n)$ { for$(k=1$ to $n)$ $\{fp(i,j)=fp(i,j)+A(k,j)*P(k,j)\}$ }}	Accumulating $A^T P$
for $(i=1$ to $n)$ { for$(j=1$ to $n)$ { for$(k=1$ to $n)$ $\{fp(i,j)=fp(i,j)+P(i,k)*A(k,j)\}$ }}	Accumulating PA
for $(i=1$ to $n)$ { for$(j=1$ to $r)$ $\{PiB(i,j)=0$ { for$(k=1$ to $n)$ $\{PiB(i,j)=PiB(i,j)+P(i,k)*B(k,j)\}$ }} for $(i=1$ to $n)$ { for$(j=1$ to $r)$ $\{PiBR\_1(i,j)=0$ for$(k=1$ to $r)$ $\{PiBR\_1(i,j)=PiBR\_1(i,j)+PiB(i,k)*R\_1(k,j)\}$ for $(i=1$ to $n)$ {for$(j=1$ to $n)$ $\{PiBR\_1BT(i,j)=0$ for$(k=1$ to $r)$ $\{PiBR\_1BT(i,j)=PiBR\_1BT(i,j)+PiBR\_1(i,k)*B(j,k)\}$ } for $(i=1$ to $n)$ { for$(j=1$ to $n)$ {for$(k=1$ to $n)$ $\{fp(i,j)=fp(i,j)-PiBR\_1BT(i,k)*P(k,j)\}$ }}	Accumulating $PBR^{-1}B^T P$
for $(i=1$ to $n)$ { for$(j=1$ to $n)\{fp(i,j)=fp(i,j)+Q(i,j)\}$ }}	Accumulating Q
for $(i=1$ to $n)$ { for$(j=1$ to $n)\{f=f+(fp(i,j))^2\}$ }}	Accumulating the values of objective function

Constitutions of computing program and inputting conditions. Constitutions of computing program includes: Main program, Non-restraint optimal sub-programs of Powell, Sub-programs of advance and retreat algorithm of one-dimensional search and golden section method, and Sub-programs of objective functions with dynamic design variables.

Inputting data of program includes: the number of state variables n, the number of control variables r, state matrix A, control matrix B, weighted matrix Q and R.

Computing example analysis

$$\dot{x}=-\frac{1}{2}x+u, x(0)=x_0$$
$$J=\frac{1}{2}Sx^2(T)+\frac{1}{2}\int_0^T(2x^2+u^2)dt$$
, finding $P(t)$ and optimal control u^*, making J access to the minimum.

The example is the problem with single input and single output, and in which $n=1, r=1$, $A=-1/2, B=1, Q=2, R=1$. The number of design variables is $N=1(1+1)/2=1$, the accuracy of objective function is $e_2=10^{-6}$ and the accuracy of golden section is $e_2=10^{-4}$. Initial values of design variables are: $z_i=0.5, i=1$. Time step is $dt=T/1000=0.001$. The program is computed in the reverse time process. When $T=1$, $P(T)=S=10$, the analytical results and the computing results by the program of $p(t)$ are shown in Table 2, the time sequence of $p(t)$ are shown in Figure 1. When, $T=10$ $P(T)=S=10$, drawing the time sequence of $P(t)$ in Figure 2. The average absolute error of the function $p(t)$ is only 0.007 by the program computation, and the optimal

Table 2 program computed solution he and the analytical solution

Time t	P(t) program solution	P(t) Analytical solution	Time t	P(t) program solution	P(t) Analytical solution
1.000	10.000	10.000	0.401	1.429	1.426
0.998	9.790	9.788	0.300	1.306	1.303
0.996	9.589	9.585	0.200	1.221	1.219
0.994	9.396	9.390	0.100	1.161	1.159
0.992	9.210	9.203	…	…	…
0.990	9.031	9.023	0.012	1.122	1.121
…	…	…	0.010	1.121	1.120
0.900	4.769	4.751	0.008	1.120	1.119
0.800	3.111	3.099	0.006	1.120	1.119
0.700	2.324	2.316	0.004	1.119	1.118
0.600	1.881	1.875	0.002	1.118	1.117
0.500	1.607	1.603	0.000	1.118	1.116

control $K(t)=P(t)$。

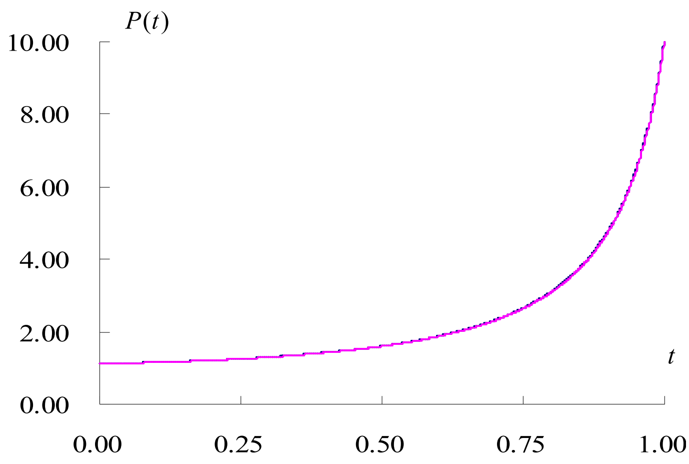

Fig.1. When $T=1, S=10$, the time sequence of P(t), the program computed solution is blue, the analytical results below is pink: $P(t)=-0.5+1.5\mathrm{cth}(-1.5t+1.643841)$

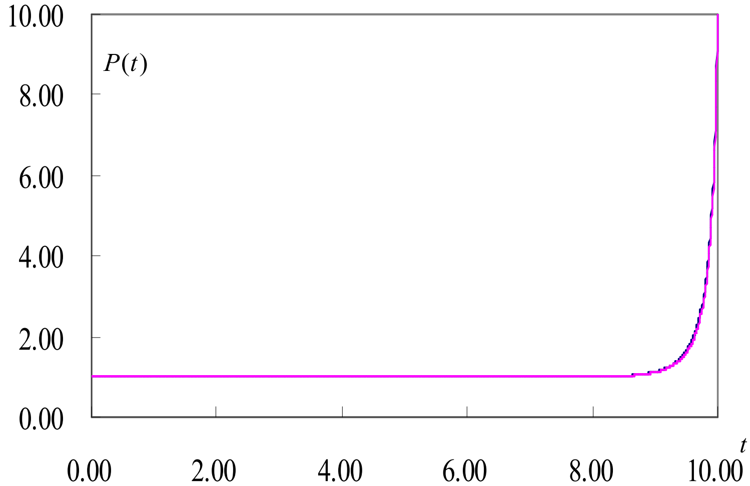

Fig.2. When $T=10, S=10$, the time sequence of p(t), the program computed solution is bule , the analytical results below is pink : $P(t)=-0.5+1.5\text{cth}(-1.5t+15.143841)$

Conclusion

In this paper, a new method is established to solve matrix Riccati differential equations, and a general computing program is designed. By comparing the solutions of two examples with the exact solutions, the effectiveness and feasibility are inspected and verified. It will be used to obtain the ideal feedback control in complex systems.

This work is supported by the National Natural Science Foundation of Liaoning Province (Grant No. 20072011) and Science research projects of Education Department of Liaoning Province (Grant No. L2010445)

References

[1] S.Y. Zhang. Modern control theory. Beijing：Metallurgical industry press.1994

[2] Q.K. Ye, Z.M. Wang. Computing method about optimization and optimal control. Beijing: Science Press, 1986

[3] P. Lancaster, L. Rodman. The algebraic Riccati equation. Oxford: Oxford University Press, 1995.

[4] L.Z. Lu, W.W. Lin. An iterative algorithm for the solution of the discrete-time algebraic Riccati equation. Linear Algebra and Its Applications, Vol. 188-189(1993), p. 465-488.

[5] C.H. Guo, P. Lancaster. Analysis and modification of Newton's method for algebraic Riccati equations. Mathematics of computation, Vol. 67(1998), p.1089~1105.

[6] M.L. Ni. A note on the maximum solutions of Riccati equations. Automatica. Vol. 27(1991), p. 1059-1060.

[7] P. Benner, A.J. Laub, V. Mehrmann. Benchmarks for the numerical solution of algebraic Riccati equations. IEEE Control Systems, 1997:18-28

[8] W.X. Zhong. The Analytical solutions of Matrix Riccati differential equations, Chinese quarterly mechanics. Vol. 21(2000), p. 2-7

[9] M.L. Ni, H.X. Wu. A Riccati equation approach to the design of linear robust controllers. Automatica, Vol. 29(1993), p.1603-1605.

[10] A. Bogdanov, E. Wan. State-dependent Riccati equation control for small autonomous helicopters. Journal of Guidance, Control and Dynamics, Vol. 30(2007), p. 47-60.

[11] X.L. Hou, S.Y. Yan, J. Liu, T.G. Wang. Dynamic design variables optimization method and its application in engineer static mechanics, Chinese Journal of Computational Mechanics, Vol. 20(2003), p. 296~301

[12] X.L. Hou, L.Z. Gu, X.H. Xu. Swing up control of circular rail single inverted pendulum. Control and Decision, Vol. 18 (2003), p. 483-486
[13] X.L. Hou, F.M. Li. Method and application of optimization program about complex electric circuit. Control Engineering of China. Vol. 9 （2002）, p. 85-87,96
[14] X.L. Hou, N. Zhang, H.S. Xu. Algorithm and application about algebraic Riccati equation based on dynamic design variables optimization method , Jounal of Shenyang Jianzhu University (Natural Science) , Vol. 26 (2010), p. 609-612