Technical Notes and Correspondence

The Recursive Reduced-Order Numerical Solution of the Singularly Perturbed Matrix Differential Riccati Equation

T. GRODT AND Z. GAJIC

Abstract—**Under stabilizability-observability conditions imposed on a singularly perturbed system, an efficient numerical method for solving the corresponding matrix differential Riccati equation is obtained in terms of the reduced-order problems. The order reduction is achieved via the use of the Chang transformation applied to the Hamiltonian matrix of a singularly perturbed linear-quadratic control problem. In addition, an efficient numerical recursive algorithm with the quadratic rate of convergence is developed for solving algebraic equations comprising the Chang transformation.**

I. Introduction

A differential Riccati equation of a singularly perturbed system [1] is given by

$$-\dot{P}(t)=P(t)A+A^TP(t)+Q-P(t)SP(t),\ P(T)=F \tag{1}$$

where

$$A=\begin{bmatrix} A_1 & A_2 \\ \frac{A_3}{\epsilon} & \frac{A_4}{\epsilon} \end{bmatrix},\ Q=\begin{bmatrix} Q_1 & Q_2 \\ Q_2^T & Q_3 \end{bmatrix},\ Q\geq 0$$

$$S=BR^{-1}B^T=\begin{bmatrix} S_1 & \frac{Z}{\epsilon} \\ \frac{Z^T}{\epsilon} & \frac{S_2}{\epsilon^2} \end{bmatrix},\ F=\begin{bmatrix} F_1 & \epsilon F_2 \\ \epsilon F_2^T & \epsilon F_3 \end{bmatrix},\ R>0$$

are $n \times n$ constant matrices and ϵ is a small positive parameter. The presence of a small parameter ϵ makes this problem numerically ill-defined, producing a so-called stiff numerical problem (huge slope at terminal time) [2]. In order to overcome this difficulty a Taylor series expansion approach, with respect to a small parameter ϵ, has been taken in [3] leading to a family of well-defined reduced-order problems. However, the Taylor series expansion method is not recursive in its application. When one is interested in a high degree of accuracy, or when ϵ is not very small, the size of computations required can be considerable. In such cases, the advantage of using the series expansion method (the important theoretical tool) is questionable from the numerical point of view, and sometimes (see Example 2) that method is almost not applicable.

In this note we will exploit the known Hamiltonian form of the solution of the Riccati equation [4] and a nonsingular transformation due to Chang [5] in order to obtain an efficient recursive numerical method for solving (1). The Chang transformation is used to block diagonalize the Hamiltonian, so that the required solution is obtained in terms of reduced-order problems. In addition, an efficient Newton type algorithm (with the quadratic rate of convergence, that is, $0(\epsilon^{2^k})$—where k is a number of iterations) is developed for solving algebraic equations comprising the Chang transformation.

Manuscript received April 25, 1987; revised September 25, 1987. Paper recommended by Associate Editor, J. H. Chow.

The authors are with the Department of Electrical and Computer Engineering, Rutgers University, Piscataway, NJ 08855-0909.

IEEE Log Number 8821014.

II. Hamiltonian Method for Solving the Singularly Perturbed Matrix Riccati Differential Equation

The solution of (1) can be sought in the form

$$P(t)=M(t)N^{-1}(t) \tag{2}$$

where matrices $M(t)$ and $N(t)$ satisfy a system of linear equations [4]

$$\dot{M}(t)=-A^TM(t)-QN(t),\qquad M(T)=F \tag{3}$$

$$\dot{N}(t)=-SM(t)+AN(t),\qquad N(T)=I \tag{4}$$

and $N(t)$ is assumed to be nonsingular for $\forall t,\ t < T$. This approach is considered as the most efficient numerical method for the solution of the differential Riccati equation [6], where the invertibility problem of $N(t)$ is solved by performing a reinitialization along the path $t_o < t < T$ whenever $N(t)$ is close to being singular.

Knowing the nature of the solution of (1), which is properly scaled as [1], [3]

$$P(t)=\begin{bmatrix} P_1(t) & \epsilon P_2(t) \\ \epsilon P_2^T(t) & \epsilon P_3(t) \end{bmatrix},\ P(T)=F=\begin{bmatrix} F_1 & \epsilon F_2 \\ \epsilon F_2^T & \epsilon F_3 \end{bmatrix} \tag{5}$$

where dim $P_1 = n_1 \times n_1$, dim $P_3 = n_2 \times n_2$, $n_1 + n_2 = n$ (n_1-slow variables, n_2-fast variables, [1]) we introduce compatible partitions of $M(t)$ and $N(t)$ matrices

$$M(t)=\begin{bmatrix} M_1(t) & M_2(t) \\ M_3(t) & M_4(t) \end{bmatrix},\ N(t)=\begin{bmatrix} N_1(t) & N_2(t) \\ N_3(t) & N_4(t) \end{bmatrix}. \tag{6}$$

The invertibility of $N(t)$ for every t, $t_o \leq t < T$, plays an important role in the proposed method. The condition under which $N(t)$ is an invertible matrix is stated in the following lemma.

Lemma: If the triple $(A, B, \sqrt{Q})$ is stabilizable-observable, then the matrix $N(t)$, with $N(T) = I$ is invertible for any $t \in (t_o, T)$.

Proof: By using the dichotomy transformation introduced in [7],

$$\begin{bmatrix} M \\ N \end{bmatrix}=\begin{bmatrix} K & P \\ I & I \end{bmatrix}\begin{bmatrix} \hat{M} \\ \hat{N} \end{bmatrix} \tag{7}$$

$$\begin{bmatrix} \hat{M} \\ \hat{N} \end{bmatrix}=\begin{bmatrix} (K-P)^{-1} & -(K-P)^{-1}P \\ -(K-P)^{-1} & I+(K-P)^{-1}P \end{bmatrix}\begin{bmatrix} M \\ N \end{bmatrix} \tag{8}$$

where P and K are unique positive definite and negative definite solutions of the algebraic Riccati equation corresponding to (2), the system (3), (4) can be transformed in

$$\begin{bmatrix} \dot{\hat{M}} \\ \dot{\hat{N}} \end{bmatrix}=\begin{bmatrix} A-SK & 0 \\ 0 & A-SP \end{bmatrix}\begin{bmatrix} \hat{M} \\ \hat{N} \end{bmatrix} \tag{9}$$

with terminal conditions

$$\hat{M}(T)=(K-P)^{-1}(F-P)$$

$$\hat{N}(T)=I+(K-P)^{-1}(F-P)=I+\hat{M}(T).$$

It is known that $(A - SK)$ is an unstable matrix and that matrix $(A - SP)$ is stable [7]. The solution of (9) is given by

$$\hat{M}(t)=e^{(A-SK)(t-T)}\hat{M}(T)$$

$$\hat{N}(t)=e^{(A-SP)(t-T)}\hat{N}(T). \tag{10}$$

0018-9286/88/0800-0751$01.00

Using (7)–(10) it can be easily shown that

$$N(t)=e^{(A-SP)(t-T)}[I+(I-e^{S(P-K)(t-T)}(K-P)^{-1}(P-F)]N(T),$$

that is

$$N(t)=\phi(t-T)N(T) \tag{11}$$

with obvious definition of $\phi(t-T)$. Since $\phi(t-T)$ plays the role of the transition matrix of $N(t)$, and by very well-known facts is nonsingular, the regularity of $N(t)$ is determined by $N(T)$ only. Thus, having chosen $N(T)$ as an identity will assure the nonsingularity of $N(t)$ for any $t<T$, and prove the given lemma.

Partitioning (3) and (4), according to (6), will reveal a decoupled structure, that is, equations for M_1, M_3, N_1, and N_3 are independent of equations for M_2, M_4, N_2, and N_4 and vice versa. Introducing a notation

$$U=\begin{bmatrix} M_1 \\ N_1 \end{bmatrix}, \quad \epsilon V=\begin{bmatrix} M_3 \\ \epsilon N_3 \end{bmatrix}, \quad X=\begin{bmatrix} M_2 \\ N_2 \end{bmatrix}, \quad \epsilon Y=\begin{bmatrix} M_4 \\ \epsilon N_4 \end{bmatrix} \tag{12}$$

$$T_1=\begin{bmatrix} -A_1^T & -Q_1 \\ -S_1 & A_1 \end{bmatrix}, \quad T_2=\begin{bmatrix} -A_3^T & -Q_2 \\ -Z & A_2 \end{bmatrix}$$

$$T_3=\begin{bmatrix} -A_2^T & -Q_2^T \\ -Z^T & A_3 \end{bmatrix}, \quad T_4=\begin{bmatrix} -A_4^T & -Q_3 \\ -S_2 & A_4 \end{bmatrix} \tag{13}$$

and after doing some algebra, we get two systems of singularly perturbed matrix equations

$$\dot{U}=T_1U+T_2V$$

$$U(T)=\begin{bmatrix} F_1 \\ I \end{bmatrix}, \quad V(T)=\begin{bmatrix} F_2^T \\ 0 \end{bmatrix} \tag{14}$$

$$\epsilon\dot{V}=T_3U+T_4V$$

$$\dot{X}=T_1X+T_2Y$$

$$X(T)=\begin{bmatrix} \epsilon F_2 \\ 0 \end{bmatrix}, \quad Y(T)=\begin{bmatrix} F_3 \\ I \end{bmatrix} \tag{15}$$

$$\epsilon\dot{Y}=T_3X+T_4Y.$$

Note that these two systems have exactly the same form and they differ in terminal conditions only. From this point we will proceed by applying the Chang transform to (14) and (15). This transformation is defined by [5]

$$J=\begin{bmatrix} I-\epsilon HL & -\epsilon H \\ L & I \end{bmatrix} \tag{16}$$

and

$$J^{-1}=\begin{bmatrix} I & \epsilon H \\ -L & I-\epsilon LH \end{bmatrix} \tag{17}$$

where L and H satisfy

$$T_4L-T_3-\epsilon L(T_1-T_2L)=0 \tag{18}$$

$$-H(T_4+\epsilon LT_2)+T_2+\epsilon(T_1-T_2L)H=0 \tag{19}$$

applied to (15) and (16) will produce

$$\dot{\hat{U}}=(T_1-T_2L)\hat{U}, \quad \hat{U}(T)=(I-\epsilon HL)U(T)-\epsilon HV(T) \tag{20}$$

$$\epsilon\dot{\hat{V}}=(T_4+\epsilon LT_2)\hat{V}, \quad \hat{V}(T)=LU(T)+V(T) \tag{21}$$

$$\dot{\hat{X}}=(T_1-T_2L)\hat{X}, \quad \hat{X}(T)=(I-\epsilon HL)X(T)-\epsilon HY(T) \tag{22}$$

$$\epsilon\dot{\hat{Y}}=(T_4+\epsilon LT_2)\hat{Y}, \quad \hat{Y}(T)=LX(T)+Y(T). \tag{23}$$

Solutions of (20)–(23) are given by

$$\hat{U}(t)=e^{(T_1-T_2L)(t-T)}\hat{U}(T) \tag{24}$$

$$\hat{V}(t)=e^{1/\epsilon(T_4+\epsilon LT_2)(t-T)}\hat{V}(T) \tag{25}$$

$$\hat{X}(t)=e^{(T_1-T_2L)(t-T)}\hat{X}(T) \tag{26}$$

$$\hat{Y}(t)=e^{1/\epsilon(T_4+\epsilon LT_2)(t-T)}\hat{Y}(T) \tag{27}$$

so that in the original coordinates we have

$$U(t)=e^{(T_1-T_2L)(t-T)}\hat{U}(T)+\epsilon He^{1/\epsilon(T_4+\epsilon LT_2)(t-T)}\hat{V}(T) \tag{28}$$

$$V(t)=-Le^{(T_1-T_2L)(t-T)}\hat{U}(T)+(I-\epsilon LH)e^{1/\epsilon(T_4+\epsilon LT_2)(t-T)}\hat{V}(T) \tag{29}$$

$$X(t)=e^{(T_1-T_2L)(t-T)}\hat{X}(T)+\epsilon He^{1/\epsilon(T_4+\epsilon LT_2)(t-T)}\hat{Y}(T) \tag{30}$$

$$Y(t)=-Le^{(T_1-T_2L)(t-T)}\hat{X}(T)+(I-\epsilon LH)e^{1/\epsilon(T_4+\epsilon LT_2)(t-T)}\hat{Y}(T). \tag{31}$$

Partitioning (28)–(31) according to (12) will produce all components of matrices $M(t)$ and $N(t)$, that is

$$\begin{bmatrix} M_1(t) \\ N_1(t) \end{bmatrix}=\begin{bmatrix} U_1(t) \\ U_2(t) \end{bmatrix}=U(t), \quad \begin{bmatrix} M_2(t) \\ N_2(t) \end{bmatrix}=\begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix}=X(t)$$

$$\begin{bmatrix} \frac{1}{\epsilon}M_3(t) \\ N_3(t) \end{bmatrix}=\begin{bmatrix} V_1(t) \\ V_2(t) \end{bmatrix}=V(t), \quad \begin{bmatrix} \frac{1}{\epsilon}M_4(t) \\ N_4(t) \end{bmatrix}=\begin{bmatrix} Y_1(t) \\ Y_2(t) \end{bmatrix}=Y(t)$$

so that the required solution of (1) is given by

$$P(t)=\begin{bmatrix} U_1(t) & X_1(t) \\ \epsilon V_1(t) & \epsilon Y_1(t) \end{bmatrix}\begin{bmatrix} U_2(t) & X_2(t) \\ V_2(t) & Y_2(t) \end{bmatrix}^{-1}. \tag{32}$$

Thus, in order to get the numerical solution of (1), that is $P(t)$, which has dimensions $n\times n=(n_1+n_2)\times(n_1\times n_2)$, we have to solve two simple algebraic equations (18) and (19) of dimensions of $(2n_2\times 2n_1)$ and $(2n_1\times 2n_2)$, respectively. The existing numerical algorithms for solving (18) and (19) can be found in [8] and [9]. Then, two exponential forms $\exp[(T_1-T_2L)(t-T)]$ and $\exp[1/\epsilon(T_4+\epsilon LT_2)(t-T)]$, have to be transformed in the matrix forms by using some of the well-known approaches [10]. Finally, the inversion of the matrix $N(t)$ has to be performed.

III. Newton Method for Solving Algebraic Equations Composing the Chang Transformation

The algebraic equations (18), which are weakly nonlinear equations and (19), a linear Lyapunov type equation, play the crucial role in the developed method and a very important role in the linear theory of singular perturbations [1]. The existing methods for solving (18) and (19) are recursive type algorithms with a rate of convergence of $0(\epsilon^k)$, where k represents the number of iterations [8], [9]. In this section a new method for solving (18) and (19) with a quadratic rate of convergence, that is $0(\epsilon^{2^k})$, will be developed. This method is based on the Newton type recursive scheme. It is a very well-known fact that the Newton method converges quadratically in the neighborhood of the sought solution and that its main problem is in the choice of the initial guess. For the algebraic equation (18) the initial guess is easily obtained with the accuracy of $0(\epsilon)$, by setting $\epsilon=0$ in that equation, that is

$$L^{(0)}=T_4^{-1}T_3=L+0(\epsilon). \tag{33}$$

Thus, the Newton sequence will be $0(\epsilon^2)$, $0(\epsilon^4)$, $0(\epsilon^8)$, $\cdots$, $0(\epsilon^{2^k})$ close to the exact solution, respectively, in each iteration.

The Newton type algorithm of (18), can be constructed by setting $L^{(i+1)}$

$= L^{(i)} + \Delta L^{(i)}$ and neglecting $0(\Delta L)^2$ terms. This will produce a Lyapunov type equation of the form

$$D_1^{(i)} L^{(i+1)} + L^{(i+1)} D_2^{(i)} = Q^{(i)} \tag{34}$$

where

$$D_1^{(i)} = T_4 + \epsilon L^{(i)} T_2,\ D_2^{(i)} = -\epsilon (T_1 - T_2 L^{(i)})$$

$$Q^{(i)} = T_3 + \epsilon L^{(i)} T_2 L^{(i)}, \qquad i = 0, 1, 2, \cdots$$

with the initial condition given by (33).

Having found the solution of (18), up to the required degree of accuracy, one can get the solution of (19) by solving directly a Lyapunov equation of the form

$$H^{(i)} D_1^{(i)} + D_2^{(i)} H^{(i)} = T_2 \tag{35}$$

which implies $H^{(i)} = H + 0(\epsilon^{2^i})$.

Note that the existence of the solutions of (18) and (19) are guaranteed by the nonsingularity of T_4. The sufficient condition for the convergence of the algorithm (34) is given by [11]

$$\| \Delta L^{(i)} \| \le \| Q^{(i)} \| = \| T_3 + \epsilon L^{(i)} T_2 L^{(i)} \| \tag{36}$$

which is almost always satisfied, except for some special cases, for example, $T_3 \simeq 0$ and $T_2 \simeq 0$, which correspond to a system already in a block diagonal form.

One has to point out, that contrary to previously used algorithms for solving (18), (19), [8], [9], which require recursive solution of linear equations, in the proposed method one is faced with the recursive solution of Lyapunov equations. Thus, for the price of speeding up the convergence from $0(\epsilon^k)$ to $0(\epsilon^{2^i})$ slightly more computations have to be performed per iteration. However, the size of computations required is of the same order, that is of $0(n^3)$ for both the solution of the Lyapunov and solution of linear equations, so that the comparison of the rate of convergence of these two algorithms plays the dominant role. In order to demonstrate the efficiency of the proposed algorithm, we have run a fifth-order example. Matrices T_1, T_2, T_3, and T_4 are chosen randomly (standard deviation equal to 1, and mean value equal to zero) such that T_4 is the invertible matrix. The simulation results for different values of a small parameter are given in Table I. It can be seen that the Newton method is much more powerful than the successive approximation recursive scheme [8], [9]. In Table II we have shown the propagation of the error per iteration when $\epsilon = 0.2$ for the Newton method.

Example 1:

T_1

−2.014	−0.058	0.499	0.585	1.372
1.366	−0.805	0.320	0.548	0.950
−0.952	0.747	0.984	−1.816	−1.563
−1.241	0.758	−1.126	0.497	−0.131
0.663	−0.021	−0.640	−0.296	1.375

T_3

−1.496	−0.666	0.699	1.262	−0.731
1.343	0.563	0.812	−1.300	−0.616
−0.521	−0.962	−0.141	−1.159	0.939
1.071	−0.943	0.017	0.696	1.295
1.397	−1.436	0.843	−1.488	0.524

IV. Numerical Solution of the Singularly Perturbed Matrix Differential Riccati Equation

The Hamiltonian method developed in Section II will be used for the numerical solution of the singularly perturbed matrix differential Riccati equation. Since the matrices $M(t)$ and $N(t)$ contain unstable modes of the Hamiltonian also [4], then even though a product $M(t)$ and $N^{-1}(t)$ tends to a constant as $t \to \infty$, the inversion of the nonsingular matrix $N(t)$, which contains huge elements, will hurt the accuracy.

TABLE I
DEPENDENCE OF THE NUMBER OF ITERATIONS ON ϵ

ϵ	number of required iterations such that $\|\|L^{(i+1)} - L^{(i)}\|\|_\infty < 10^{-7}$	
	Newton method	Successive approximations
0.3	6	*
0.2	5	*
0.1	4	*
0.04	4	19
0.02	4	11
0.01	3	7
0.001	2	4

* does not converge

TABLE II
PROPAGATION OF THE ERROR PER ITERATION FOR A CONSTANT VALUE OF ϵ FOR THE NEWTON METHOD

ϵ=0.2 i	$\|\|L^{(i+1)} - L^{(i)}\|\|_\infty$
1	2.40745×10^{0}
2	7.80653×10^{-1}
3	4.21800×10^{-2}
4	0.88748×10^{-4}
5	0.17808×10^{-8}

The reinitialization version of the Hamiltonian approach, which leads to the known Kalman–Englar method [4], is considered as the most efficient numerical method for the solution of the general matrix differential Riccati equation [6]. The reinitialization technique applied to the results of Section II will modify formulas (3), (14), and (15), respectively, in

$$M(k\Delta t) = P(k\Delta t) \tag{37}$$

$$U(k\Delta t) = \begin{bmatrix} P_1(k\Delta t) \\ I \end{bmatrix}, \quad V(k\Delta t) = \begin{bmatrix} P_2^T(k\Delta t) \\ 0 \end{bmatrix} \tag{38}$$

$$X(k\Delta t) = \begin{bmatrix} \epsilon P_2(k\Delta t) \\ 0 \end{bmatrix}, \quad Y(k\Delta t) = \begin{bmatrix} P_3(k\Delta t) \\ I \end{bmatrix} \tag{39}$$

T_2

−1.796	−0.009	−0.840	1.819	0.794
0.158	0.467	1.324	−0.123	0.629
−0.433	0.248	−1.181	−1.426	0.297
−1.599	0.269	−0.133	−0.845	−0.769
1.967	−0.565	0.776	1.419	−0.450

T_4

−1.367	−0.885	−0.506	−1.174	1.435
0.133	1.319	1.244	0.892	−1.221
−0.296	1.333	1.002	−0.927	−0.794
0.780	1.358	0.607	−0.511	0.671
−0.999	0.914	−1.320	−0.556	−1.135

where k represents the number of steps and Δt is an integration step. This will introduce slight modifications in formulas (20)–(31), namely, instead of the final time T, a discrete time $k\Delta t$ has to be used. These changes can be implemented very easily from the programming point of view.

Example 2: The recursive solution of the differential matrix Riccati equation of singularly perturbed systems is demonstrated on the seventh-order model of the synchronous machine connected to an infinite bus [8].

TABLE III
SIMULATION RESULTS FOR THE ELEMENT $P_{11}(t)$

time = t	0.1	0.5	1.0
$P_{11}^{(exact)}$	1.9699	6.6483	9.6600
$P_{11}^{(12)}$	"	"	"
$P_{11}^{(11)}$	"	"	9.6599
$P_{11}^{(10)}$	1.9698	"	9.6601
$P_{11}^{(9)}$	1.9700	6.6484	9.6598
$P_{11}^{(8)}$	1.9696	6.6482	9.6602
$P_{11}^{(7)}$	1.9703	6.6487	9.6603
$P_{11}^{(6)}$	1.9694	6.6471	9.6572
$P_{11}^{(5)}$	1.9703	6.6500	9.6671
$P_{11}^{(4)}$	1.9720	6.6496	9.6477
$P_{11}^{(3)}$	1.9537	6.6488	9.6991
$P_{11}^{(2)}$	2.0603	6.6520	9.5417
$P_{11}^{(1)}$	1.9847	6.7926	9.8624
$P_{11}^{(0)}$	1.9742	7.0256	10.4610

The system matrix A is given by

$$A=\begin{bmatrix} -0.58 & 0 & 0 & -0.27 & 0 & 0.2 & 0 \\ 0 & -1. & 0 & 0 & 0 & 1. & 0 \\ 0 & 0 & -5. & 2.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 337. & 0 & 0 \\ -0.14 & 0 & 0.14 & -0.2 & -0.28 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.08 & 2. \\ -173. & 66.7 & -116. & 40.9 & 0 & -66.7 & -16.7 \end{bmatrix}.$$

Remaining matrices are chosen as $Q = I$, $F = 0$, S_1, S_2, and Z have all entries equal to 1. The eigenvalues of A are $-8.53 \pm j8.22$, -3.93, $-0.362 \pm j0.56$, $-0.86 \pm j8.37$. Two fast and five slow variables are separated by the choice of the small singular perturbation parameter $\epsilon = 0.4$ (roughly the ratio of 3.93 and 8.53). Simulation results for the element $P_{11}(t)$ are given in Table III. It can be seen that in order to get the accuracy of four decimal digits it takes 12 iterations (the successive approximation method was used for solving algebraic equations composing the Chang transformation—in order to be able to compare the proposed recursive scheme to the power series expansion method, since both methods are producing the same order of accuracy). This result is expected since $0(0.4^{12}) \simeq 10^{-5}$. That means if the power series expansion method had been used, in order to get the same accuracy, it would have required 12 terms, that is [3]

$$P(t,\epsilon)=\sum_{m=0}^{11}\frac{\epsilon^m}{m!}\{P_s^{(m)}(t)+P_f^{(m)}(\tau)\}+0(\epsilon^{12}),\quad \tau=\frac{t-T}{\epsilon}$$

where

$$P_s^{(m)}(t)=\begin{bmatrix} P_{1s}^{(m)}(t) & \epsilon P_{2s}^{(m)}(t) \\ \epsilon P_{2s}^{(m)T}(t) & \epsilon P_{3s}^{(m)}(t) \end{bmatrix},$$

$$P_f^{(m)}(\tau)=\begin{bmatrix} P_{1f}^{(m)}(\tau) & \epsilon P_{2f}^{(m)}(\tau) \\ \epsilon P_{2f}^{(m)T}(\tau) & \epsilon P_{3f}^{(m)}(\tau) \end{bmatrix}.$$

It is shown in [3, pp. 21, formula 32] that the right-hand sides of differential equations for $P_{1f}^{(1)}(\tau)$, $P_{2f}^{(1)}(\tau)$, and $P_{3f}^{(1)}(\tau)$ contain, respectively, 7, 23, and 22 terms, each consisting of a product of two or three matrices. Thus, the size of computations required for only an $0(\epsilon^2)$ accuracy is already enormous. The complexity of the right-hand side of differential equations for $P_f^{(m)}(\tau)$ grows extremely quickly with the increase of m so that this nice theoretical method is not convenient for the practical computations. For an $0(\epsilon^{12})$ accuracy, the right-hand sides of the differential equations for the power series expansion method will contain hundreds or even thousands of terms, and Example 2 cannot be efficiently solved by using that method.

All simulation results of this note are obtained by using the software package L-A-S for the computer aided control system design [12].

V. CONCLUSIONS

The recursive method for the numerical solution of the singularly perturbed Riccati differential equation proposed in this note is very important in two cases: a) ϵ is not very small; b) high order of accuracy is required. The first case represents one of the main problems in the modern numerical analysis of the singularly perturbed problems. It was pointed by P. Hemker [13] that "numerical analysis of singular perturbation problems mainly concentrates on the following question: how to find a numerical approximation to the solution for small as well as intermediate values of ϵ, where no short asymptotic expansion is available. Or, more general, how to construct a single numerical method that can be applied both in the case of extremely small ϵ and for larger values of ϵ, when one wouldn't consider the problem as singularly perturbed any longer." Results reported in this note resolve that problem in the case of the singularly perturbed Riccati differential equation.

REFERENCES

[1] P. Kokotovic and H. Khalil, *Singular Perturbations in Systems and Control.* New York: IEEE Press, 1986.

[2] W. Miranker, *Numerical Methods for Stiff Equations.* Amsterdam, The Netherlands: Reidel, 1981.

[3] R. Yackel and P. Kokotovic, "A boundary layer method for the matrix Riccati equation," *IEEE Trans. Automat. Contr.*, vol. AC-17, pp. 17-24, 1973.

[4] H. Kwakernaak and R. Sivan, *Linear Optimal Control Systems.* New York: Wiley, 1972.

[5] K. Chang, "Singular perturbations of a general boundary value problem," *SIAM J. Math. Anal.*, vol. 3, pp. 520-526, 1972.

[6] C. Kenney and R. Leipnik, "Numerical integration of the differential matrix Riccati equation," *IEEE Trans. Automat. Contr.*, vol. AC-30, pp. 962-970, 1985.

[7] R. Wilde and P. Kokotovic, "A dichotomy in linear control theory," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 382-383, 1972.

[8] P. Kokotovic, J. Allemong, J. Winkelman, and J. Chow, "Singular perturbations and iterative separation of time scales," *Automatica,* vol. 16, pp. 23-33, 1980.

[9] Z. Gajic, "Numerical fixed-point solution for near-optimum regulators of linear-quadratic Gaussian control problem for singularly perturbed systems," *Int. J. Contr.*, vol. 43, pp. 373-387, 1986.

[10] C. Molen and C. van Loan, "Nineteen dubious ways to compute the exponential of a matrix," *SIAM Rev.*, vol. 20, pp. 801-836, 1978.

[11] P. Belanger and T. McGillivray, "Computational experience with the solution of the matrix Lyapunov equation," *IEEE Trans. Automat. Contr.*, vol. AC-20, pp. 799-800, 1976.

[12] P. West, S. Bingulac, and W. Perkins, "L-A-S: A computer-aided control system design language," in *Computer Aided Control System Engineering,* M. Jamshidi and C. Herget, Eds. Amsterdam, The Netherlands: North-Holland, 1985.

[13] P. Hemker, "Numerical aspects of singular perturbation problems," in *Asymptotic Analysis II, Lecture Notes in Mathematics,* vol. 985. New York: Springer-Verlag, 1983, pp. 267-287.