Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 134604, 10 pages
http://dx.doi.org/10.1155/2014/134604

Research Article

Computation of Gram Matrix and Its Partial Derivative Using Precise Integration Method for Linear Time-Invariant Systems

Sulan Li,[1] Yuanhao Ren,[1] Hong Bao,[1] and Wei Zhang[2]

[1] *Key Laboratory of Electronic Equipment Structure Design of Ministry of Education, Xidian University, 2 Taibai Road, Xi'an 710071, China*
[2] *Wuhan Second Ship Design and Research Institute, 450 Zhongshan Road, Wuhan 430064, China*

Correspondence should be addressed to Sulan Li; slli@xidian.edu.cn

Received 14 October 2013; Revised 9 March 2014; Accepted 9 March 2014; Published 22 April 2014

Academic Editor: Alexander Timokha

Gram matrix is an important tool in system analysis and design as it provides a description of the input-output behavior for system; its partial derivative matrix is often required in some numerical algorithms. It is essential to study computation of these matrices. Analytical methods only work in some special circumstances; for example, the system matrix is diagonal matrix or Jordan matrix. In most cases, numerical integration method is needed, but there are two problems when compute using traditional numerical integration method. One is low accuracy: as high accuracy requires extremely small integration step, it will result in large amount of computation; and another is stability and stiffness issues caused by the dependence on the property of system matrix. In order to overcome these problems, this paper proposes an efficient numerical method based on the key idea of precise integration method (PIM) for the Gram matrix and its partial derivative of linear time-invariant systems. Since matrix inverse operation is not required in this method, it can be used with high precision no matter the system is normal or singular. The specific calculation algorithm and block diagram are also given. Finally, numerical examples are given to demonstrate the correctness and validity of this method.

1. Introduction

Gram matrix contains elements which are scalar products of repeated integrals of impulse response of the system. It provides a description of the input-output behavior of the system and is therefore a potential tool in system analysis and design. Not only it determines controllability and observability of system [1], but also in several applications [2–10], it is essential. Such as in [2], Gram matrix was used in identification of linear system, in [3, 4], methods based on Gram matrix for observability restoration were presented, in [5, 6], Gram matrix was applied in state estimation and observability analysis, and in [4, 7–10] problems of state estimation using Gram matrix associated with Jacobian matrix for power system were studied. Great attention has been gained in the theoretical study and practical engineering, so it is very necessary to study its calculation.

If the system matrix is in a special form, for example, is a diagonal matrix or a Jordan matrix, or when the order of the system under consideration is not large, analytical solution of Gram matrix will be obtained [1]. But in practice, most of the system matrices are not in conformity with the forms above; the order is large or even system matrix is singular in some cases; analytical method is essentially useless; some other methods can be used to solve it. A variety of methods permit a both fast and accurate computation of the Gram matrix of system [11–16]. In [11], a method for evaluation of the Gram matrix from the coefficients of the system transfer function is proposed, but it is tedious and cumbersome; the methods proposed in [12–14] computed Gram matrix using its properties in time-domain, but as the authors stated, the technique though very elegant in principle it gives poor results due to two reasons. Firstly, it is difficult to obtain accurate measurements of impulse response date since the data obtained is usually contaminated with noise. And secondly, numerical integration techniques again introduce errors due to truncation and round off. Reference [15] presents an efficient algorithm for orthogonal decomposition of derivatives and antiderivatives of functions with rational Laplace transforms. A method for computing extended Gram matrix was proposed based on the new theorem related to Rough $\alpha-\beta$ expansion. But it is relatively applied in a narrow scope and not suited for irrational resonant system. In [16], a method for evaluation of the Gram matrix via a Kautz model is proposed. However, the efficiency or accuracy of the computation heavily depends on the choice of Kautz parameters, and it is also only suitable for stable systems. And traditional numerical integration methods [17], such as trapezoidal integration, Simpson integration, and Gaussian integral, also can be used to compute Gram matrix. Unfortunately, they are always with low accuracy; to improve accuracy, integration step needs to be divided as small as possible; thus, computational effort will greatly increase and the solution also heavily depends on the behavior of system matrix, which will result in some troubles, such as stability and stiffness issues.

To overcome the above shortcomings and improve the accuracy of time step integration, Zhong and Williams proposed the precise integration method originally in [18]. It has been widely used in solving matrix integrals because of its high precision, stability, and fast calculation. On application to linear time-invariant dynamic homogeneous systems [19], the method can give precise numerical results approaching the exact solution at the integration points. In [20–23], it is applied to non-homogeneous dynamic systems. And Duhamel integral term is approximated using polynomial functions, sin/cos functions, exponential functions, and their combinations firstly. The corresponding recursive format is derived [20]. The accuracy of the algorithm is restricted by the inherent errors in matrix inversion and simulation of the applied loading. References [21, 22] deal with the Duhamel term using other methods, but it is still restricted by the inherent errors in matrix inversion. Reference [23] takes the idea of precise integration of exponential matrix into Duhamel integration to avoid the matrix inversion. Reference [24] reviews the precise integration method; it is not only used to solve time-invariant system, but also for time-variant, nonlinear system, and two-point boundary value problems. References [25–27] discuss the numerical stability and accuracy of precise time integration method in detail.

Unfortunately, in the above research, precise time integration method is only used to calculate the dynamic response of system or further to calculate two-point boundary value problems. It is necessary to notice that there is only one term of matrix exponent function in the above integrands [18–24]; the way how to deal with the part of matrix exponent function is the difficulty of matrix integration methods. But for Gram matrix and its partial derivative matrix, there are two terms of matrix exponent functions multiplied and separated by a matrix. Considering that general matrix operations cannot be directly combined as they are not commutative, so existing precise integration method cannot be applied to solve Gram matrix and its partial derivative matrix directly. Therefore, based on the key ideas of precise integration, this paper gives the specific computing method and derives the incremental form of the additional theorem for the computation of Gram matrix and its partial derivative matrix.

The rest of this paper is organized as follows: In Section 2, the formulation of Gram matrix and its partial derivative matrix are stated, and then the precise integration method to compute these matrices is described in detail. In Section 3, some representative numerical examples are presented and simulation results are discussed. Finally, conclusions are presented in Section 4.

2. Materials and Methods

2.1. Gram Matrix. A linear time-invariant system can be given in matrix/vector form as

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{ax} + \mathbf{bu},\\ \mathbf{y} &= \mathbf{cx},\end{aligned} \tag{1}$$

in which a dot above $\dot{()}$ means the differentiation with respect to time t, x is an n-dimensional state vector, u is a p-dimensional input vector, and y is a q-dimensional output vector. **a**, **b,** and **c** are $n \times n$, $n \times p$ and $q \times n$ dimensional system matrix, input matrix, and output matrix, respectively. For any time $t > 0$, the corresponding Gram matrix contains controllability matrix $\mathbf{W}_C^t$ and observability matrix $\mathbf{W}_O^t$; they are defined as

$$\mathbf{W}_C^t = \int_0^t e^{\mathbf{a}\tau}\mathbf{bb}^T e^{\mathbf{a}^T\tau} d\tau, \tag{2}$$

$$\mathbf{W}_O^t = \int_0^t e^{\mathbf{a}^T\tau}\mathbf{c}^T \mathbf{c} e^{\mathbf{a}\tau} d\tau. \tag{3}$$

If $\mathbf{W}_C^t$ and $\mathbf{W}_O^t$ are nonsingular separately, the relevant system is completely controllable and observable. Considering that $\mathbf{W}_C^t$ and $\mathbf{W}_O^t$ have the similar form, therefore take $\mathbf{W}_C^t$ as example to illustrate the computation of Gram matrix using precise integration method. To facilitate the derivation, supposing that $\mathbf{A}_1 = \mathbf{a}$, $\mathbf{A}_2 = \mathbf{a}^T$, $\mathbf{B} = \mathbf{bb}^T$, then Gram matrix corresponding to (2) is equivalent to the following forms:

$$\mathbf{W}_C^t = \int_0^t e^{\mathbf{A}_1\tau}\mathbf{B}e^{\mathbf{A}_2\tau} d\tau. \tag{4}$$

In case the system corresponding to (1) is stable, then with the increasing of time t, $\mathbf{W}_C^t$ will converge to a constant value, that is to say $\lim_{t\to\infty}\int_0^t e^{\mathbf{A}_1\tau}\mathbf{B}e^{\mathbf{A}_2\tau} d\tau$ = constant; the built-in function gram () in MATLAB can be used to get this constant.

2.2. Partial Derivative of Gram Matrix. If the elements of system matrix **a** are functions of a variable, the resulting Gram matrix $\mathbf{W}_C^t$ is also a function of this variable; then there is a partial derivative matrix to this variable, which is often required in some numerical optimization algorithm. It is also necessary to study the computation of this partial derivative matrix. Assume that the system matrix is a function of w (w and t are independent); that is to say

$$\mathbf{W}_C^t = \int_0^t e^{\mathbf{A}_1(w)\tau}\mathbf{B}e^{\mathbf{A}_2(w)\tau} d\tau. \tag{5}$$

For a specific system, expressions of Gram matrix are generally complicated enough, so it is rather cumbersome to solve the partial derivatives matrix directly. Consider

$$\frac{\partial \mathbf{W}_C^t}{\partial w} = \frac{\partial \left(\int_0^t e^{\mathbf{A}_1(w)\tau} \mathbf{B} e^{\mathbf{A}_2(w)\tau} d\tau \right)}{\partial w}. \tag{6}$$

Therefore, make the following processing:

$$\begin{aligned} \frac{\partial \mathbf{W}_C^t}{\partial w} &= \int_0^t \frac{\partial \left(e^{\mathbf{A}_1(w)\tau} \mathbf{B} e^{\mathbf{A}_2(w)\tau} \right)}{\partial w} d\tau \\ &= \int_0^t \left(\frac{\partial \left(e^{\mathbf{A}_1 \tau} \right)}{\partial w} \mathbf{B} e^{\mathbf{A}_2 \tau} + e^{\mathbf{A}_1 \tau} \mathbf{B} \frac{\partial \left(e^{\mathbf{A}_2 \tau} \right)}{\partial w} \right) d\tau. \end{aligned} \tag{7}$$

According to the following relationship:

$$\frac{\partial \left(e^{\mathbf{A}t} \right)}{\partial w} = t e^{\mathbf{A}t} \frac{\partial (\mathbf{A})}{\partial w} \tag{8}$$

(6) is equivalent to

$$\begin{aligned} \frac{\partial \mathbf{W}_C^t}{\partial w} &= \int_0^t \left(\tau e^{\mathbf{A}_1 \tau} \frac{\partial (\mathbf{A}_1)}{\partial w} \mathbf{B} e^{\mathbf{A}_2 \tau} + e^{\mathbf{A}_1 \tau} \mathbf{B} \tau e^{\mathbf{A}_2 \tau} \frac{\partial (\mathbf{A}_2)}{\partial w} \right) d\tau \\ &= \int_0^t \tau e^{\mathbf{A}_1 \tau} \left(\frac{\partial (\mathbf{A}_1)}{\partial w} \mathbf{B} \right) e^{\mathbf{A}_2 \tau} d\tau \\ &\quad + \int_0^{\rho} \tau e^{\mathbf{A}_1 \tau} \mathbf{B} e^{\mathbf{A}_2 \tau} d\tau \cdot \frac{\partial (\mathbf{A}_2)}{\partial w}. \end{aligned} \tag{9}$$

Two different integration terms are contained in (9), and these two terms can be summarized to the computation of integration as follows:

$$\mathbf{W}_{C1}^t = \int_0^t \tau e^{\mathbf{A}_1 \tau} \mathbf{P} e^{\mathbf{A}_2 \tau} d\tau, \tag{10}$$

in which the dimensions of $\mathbf{P}$ and $\mathbf{B}$ are equivalent. As long as (10) is solved, the partial derivative of Gram matrix corresponding to (9) will be solved. Therefore, take (10) as example to illustrate the computation of partial derivative matrix in the following.

2.3. Precise Integration Method for Computation of Gram Matrix. Now the problem is to compute (4) using precise integration method. A basic time step, denoted as η, is necessary to be given for the numerical integration. The following relationship is satisfied:

$$t_0 = 0, \quad t_1 = \eta, \ldots, t_k = k\eta, \quad t_{k+1} = (k+1)\eta, \ldots. \tag{11}$$

According to (4), it can be obtained that between two Gram matrices, at any time $t_k = k \cdot \eta$ $(k = 0, 1, 2, \ldots)$ and the next interval $t_{k+1} = t_k + \eta$, the following relationship is satisfied:

$$\begin{aligned} \mathbf{W}_C^{k+1} &= \mathbf{W}_C^k + e^{\mathbf{A}_1 t_k} \int_0^{\eta} e^{\mathbf{A}_1 \tau} \mathbf{B} e^{\mathbf{A}_2 \tau} d\tau \cdot e^{\mathbf{A}_2 t_k} \\ &= \mathbf{W}_C^k + e^{\mathbf{A}_1 t_k} \cdot \mathbf{W}_C^{\eta} \cdot e^{\mathbf{A}_2 t_k}. \end{aligned} \tag{12}$$

Therefore, the problem is transformed to compute $\mathbf{W}_C^{\eta}$. Next, the precise integration method for $\mathbf{W}_C^{\eta}$ will be given. According to the key point of precise integration method, the corresponding additional theorem is derived:

$$\mathbf{W}_C^{\eta} = \int_0^{\eta} e^{\mathbf{A}_1 \tau} \mathbf{B} e^{\mathbf{A}_2 \tau} d\tau, \tag{13}$$

$$\begin{aligned} \mathbf{W}_C^{2\eta} &= \int_0^{2\eta} e^{\mathbf{A}_1 \tau} \mathbf{B} e^{\mathbf{A}_2 \tau} d\tau \\ &= \int_0^{\eta} e^{\mathbf{A}_1 \tau} \mathbf{B} e^{\mathbf{A}_2 \tau} d\tau + \int_{\eta}^{2\eta} e^{\mathbf{A}_1 \tau} \mathbf{B} e^{\mathbf{A}_2 \tau} d\tau \\ &= \mathbf{W}_C^{\eta} + \int_0^{\eta} e^{\mathbf{A}_1 \eta} e^{\mathbf{A}_1 \tau'} \mathbf{B} e^{\mathbf{A}_2 \tau'} e^{\mathbf{A}_2 \eta} d\tau' \\ &= \mathbf{W}_C^{\eta} + \boldsymbol{\varphi}_1(\eta) \int_0^{\eta} e^{\mathbf{A}_1 \tau'} \mathbf{B} e^{\mathbf{A}_2 \tau'} d\tau' \cdot \boldsymbol{\varphi}_2(\eta) \\ &= \mathbf{W}_C^{\eta} + \boldsymbol{\varphi}_1(\eta) \mathbf{W}_C^{\eta} \boldsymbol{\varphi}_2(\eta). \end{aligned} \tag{14}$$

Equation (14) is the main key point of precise integration method. According to this equation, take ρ as precise interval to divide the basic interval η:

$$\rho = \frac{\eta}{2^N}. \tag{15}$$

N is the weighting coefficient of matrix $\mathbf{A}$; generally the computational results are accurate enough when $N = 20$ [18, 29]; that is to say $\rho = \eta/1048576$. η is a small time interval, and ρ is an extremely small time interval. Hence for the interval, the truncated Taylor expansion is applied with high precision:

$$\begin{aligned} \mathbf{W}_C^{\rho} &= \int_0^{\rho} e^{\mathbf{A}_1 \tau} \mathbf{B} e^{\mathbf{A}_2 \tau} d\tau \\ &= \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} \int_0^{\rho} \frac{\mathbf{A}_1^k \tau^k \mathbf{B} \mathbf{A}_2^m \tau^m}{k! m!} d\tau \\ &= \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} \frac{\mathbf{A}_1^k \mathbf{B} \mathbf{A}_2^m \rho^{k+m+1}}{k! m! (k+m+1)} \\ &\approx \mathbf{B}\rho + \rho^2 \left(\frac{\mathbf{B}\mathbf{A}_2}{2} + \frac{\mathbf{A}_1 \mathbf{B}}{2} \right) \\ &\quad + \rho^3 \left(\frac{\mathbf{B}\mathbf{A}_2^2}{6} + \frac{\mathbf{A}_1 \mathbf{B} \mathbf{A}_2}{3} + \frac{\mathbf{A}_1^2 \mathbf{B}}{6} \right) \\ &\quad + \rho^4 \left(\frac{\mathbf{B}\mathbf{A}_2^3}{24} + \frac{\mathbf{A}_1 \mathbf{B} \mathbf{A}_2^2}{8} + \frac{\mathbf{A}_1^2 \mathbf{B} \mathbf{A}_2}{8} + \frac{\mathbf{A}_1^3 \mathbf{B}}{24} \right) \end{aligned} \tag{16}$$

$$\begin{aligned}\boldsymbol{\varphi}_{1(2)}(\rho) &= \exp\left(\mathbf{A}_{1(2)}\rho\right) = \sum_{k=0}^{\infty}\frac{\mathbf{A}_{1(2)}^{k}\rho^{k}}{k!}\\ &\approx \mathbf{I} + \left(\mathbf{A}_{1(2)}\rho\right) + \frac{\left(\mathbf{A}_{1(2)}\rho\right)^{2}}{2} + \frac{\left(\mathbf{A}_{1(2)}\rho\right)^{3}}{6}\\ &\quad + \frac{\left(\mathbf{A}_{1(2)}\rho\right)^{4}}{24} = \mathbf{I} + \boldsymbol{\varphi}'_{1(2)}(\rho).\end{aligned} \qquad (17)$$

Considering that ρ is very small, the first five term series expansion should be enough. Theoretically, the solution at basic interval will be obtained through merge operations N times as described by (14). But it can be seen from (17) that $\boldsymbol{\varphi}_1(\rho)$ and $\boldsymbol{\varphi}_2(\rho)$ both depart from the unit matrix $\mathbf{I}$ to a very small extent. And $\boldsymbol{\varphi}'_{1(2)}(\rho)$ is extremely small; if it is added to the unit matrix $\mathbf{I}$, it will become an appended part and its precision will be seriously dropped in the round-off operation in computer arithmetic. In order to avoid the resulting loss of accuracy, the increment $\boldsymbol{\varphi}'_{1(2)}(\rho)$ and the unit matrix $\mathbf{I}$ should be separated, and $\boldsymbol{\varphi}'_{1(2)}(\rho)$ is kept in the memory rather than the matrix $\boldsymbol{\varphi}_{1(2)}(\rho)$. So, (14) is rewritten into the following form of incremental part $\boldsymbol{\varphi}'_{1(2)}(\rho)$:

$$\mathbf{W}_{C}^{2\rho} = \mathbf{W}_{C}^{\rho} + \left(\mathbf{I} + \boldsymbol{\varphi}'_1(\rho)\right)\mathbf{W}_{C}^{\rho}\left(\mathbf{I} + \boldsymbol{\varphi}'_2(\rho)\right). \qquad (18)$$

So at each step, attention will be focused on the incremental, rather than the whole amount; the precision loss caused by computer rounding operation will be avoided. And this is the second key point of precise integration method.

2.4. Precise Integration Method for Computation of Partial Derivative Matrix. The strategy of precise integration method will be used to deal with $\mathbf{W}_{C1}^{t}$. The corresponding additional theorem is derived firstly:

$$\begin{aligned}\mathbf{W}_{C1}^{2\rho} &= \int_0^{2\rho}\tau e^{\mathbf{A}_1\tau}\mathbf{P}e^{\mathbf{A}_2\tau}d\tau\\ &= \int_0^{\rho}\tau e^{\mathbf{A}_1\tau}\mathbf{P}e^{\mathbf{A}_2\tau}d\tau + \int_{\rho}^{2\rho}\tau e^{\mathbf{A}_1\tau}\mathbf{P}e^{\mathbf{A}_2\tau}d\tau\\ &= \mathbf{W}_{C1}^{\rho} + \int_{\rho}^{2\rho}\tau e^{\mathbf{A}_1\tau}\mathbf{P}e^{\mathbf{A}_2\tau}d\tau\\ &= \mathbf{W}_{C1}^{\rho} + \int_0^{\rho}\tau e^{\mathbf{A}_1\rho}e^{\mathbf{A}_1\tau}\mathbf{P}e^{\mathbf{A}_2\tau}e^{\mathbf{A}_2\rho}d\tau\\ &\quad + \int_0^{\rho}\rho e^{\mathbf{A}_1\rho}e^{\mathbf{A}_1\tau}\mathbf{P}e^{\mathbf{A}_2\tau}e^{\mathbf{A}_2\rho}d\tau\\ &= \mathbf{W}_{C1}^{\rho} + \boldsymbol{\varphi}_1(\rho)\mathbf{W}_{C1}^{\rho}\boldsymbol{\varphi}_2(\rho) + \rho\boldsymbol{\varphi}_1(\rho)\mathbf{W}_{C}^{\rho}\boldsymbol{\varphi}_2(\rho).\end{aligned} \qquad (19)$$

In the above expression, $\boldsymbol{\varphi}_1(\rho)$, $\mathbf{W}_C(\rho)$, and $\boldsymbol{\varphi}_2(\rho)$ can refer to the preview results. For $\mathbf{W}_{C1}(\rho)$, finite Taylor expansion can be used to approximate

$$\begin{aligned}\mathbf{W}_{C1}^{\rho} &= \int_0^{\rho}\tau e^{\mathbf{A}_1\tau}\mathbf{P}e^{\mathbf{A}_2\tau}d\tau = \sum_{k=0}^{\infty}\sum_{m=0}^{\infty}\int_0^{\rho}\frac{\tau\mathbf{A}_1^{k}\tau^{k}\mathbf{P}\mathbf{A}_2^{m}\tau^{m}}{k!m!}d\tau\\ &= \sum_{k=0}^{\infty}\sum_{m=0}^{\infty}\frac{\mathbf{A}_1^{k}\mathbf{P}\mathbf{A}_2^{m}\rho^{k+m+2}}{k!m!\,(k+m+2)}\\ &\approx \rho^2\frac{\mathbf{P}}{2} + \rho^3\left(\frac{\mathbf{P}\mathbf{A}_2}{3} + \frac{\mathbf{A}_1\mathbf{P}}{3}\right)\\ &\quad + \rho^4\left(\frac{\mathbf{P}\mathbf{A}_2^2}{8} + \frac{\mathbf{A}_1\mathbf{P}\mathbf{A}_2}{4} + \frac{\mathbf{A}_1^2\mathbf{P}}{8}\right)\\ &\quad + \rho^5\left(\frac{\mathbf{P}\mathbf{A}_2^3}{30} + \frac{\mathbf{A}_1\mathbf{P}\mathbf{A}_2^2}{10} + \frac{\mathbf{A}_1^2\mathbf{P}\mathbf{A}_2}{10} + \frac{\mathbf{A}_1^3\mathbf{P}}{30}\right).\end{aligned} \qquad (20)$$

The incremental form of the additional theorem is

$$\begin{aligned}\mathbf{W}_{C1}^{2\rho} &= \mathbf{W}_{C1}^{\rho} + \left(\mathbf{I} + \boldsymbol{\varphi}'_1(\rho)\right)\mathbf{W}_{C1}^{\rho}\left(\mathbf{I} + \boldsymbol{\varphi}'_2(\rho)\right)\\ &\quad + \rho\left(\mathbf{I} + \boldsymbol{\varphi}'_1(\rho)\right)\mathbf{W}_{C}^{\rho}\left(\mathbf{I} + \boldsymbol{\varphi}'_2(\rho)\right).\end{aligned} \qquad (21)$$

Now, the numerical solution of $\mathbf{W}_{C1}^{t}$ is obtained using precise integration method according to (21); therefore, the exact numerical solution of partial derivatives of Gram matrix can be obtained . Since the dimensions of $\mathbf{P}$ and $\mathbf{B}$ are equivalent, their values do not affect the specific verification. Therefore, in the numerical examples of the next section, calculations are based on $\mathbf{P} = \mathbf{B}$.

2.5. Programming Flow Chart of the Method. Take the computation of Gram matrix at basic interval η as example, the programming flow chart is given in Figure 1.

The amount of computation mainly focuses on matrix multiplication [29] and contains two parts: one is the computation of initial basic interval, and another is N times merging during the addition theorem. The amount of computation is about $O[(N+q-1)n^3 + (N+q-1)mn^2]$, in which N is the weighting coefficient of matrix $\mathbf{A}$ (see (15)), q represents the order of Taylor expansion series (see (16), (17)), m is number of columns of matrix $\mathbf{B}$, and n is order of matrix $\mathbf{A}$.

3. Numerical Examples

In this section, we verify the method presented in the preceding section with three numerical examples.

Example 1. The system matrix is $\mathbf{a} = [-1]$, $\mathbf{b} = [1]$; the integration time gradually increases with increments of 0.2 s; compute Gram matrices $\mathbf{W}_{C}^{t}$ and $\mathbf{W}_{C1}^{t}$.

According to

$$\mathbf{W}_{C}^{t} = \int_0^{t} e^{\mathbf{A}_1\tau}\mathbf{B}e^{\mathbf{A}_2\tau}d\tau, \qquad \mathbf{W}_{C1}^{t} = \int_0^{t}\tau e^{\mathbf{A}_1\tau}\mathbf{B}e^{\mathbf{A}_2\tau}d\tau, \quad (22)$$

Table 1: Analytical solution and numerical solution using precise integration method of $\mathbf{W}_C^t$.

Time/s	$t = 0.2$	$t = 0.4$	$t = 0.6$	$t = 0.8$	$t = 1.0$	$t = 10$
Analytical solution	$\frac{1 - e^{-0.4}}{2}$	$\frac{1 - e^{-0.8}}{2}$	$\frac{1 - e^{-1.2}}{2}$	$\frac{1 - e^{-1.6}}{2}$	$\frac{1 - e^{-2.0}}{2}$	$\frac{1 - e^{-20}}{2}$
Numerical solution	0.1648399769821804	0.2753355179413893	0.3494028940438990	0.3990517410026724	0.4323323583816937	0.4999999989694231
Absolute error ($\times 10^{-17}$)	4.211244565981330	6.964767980438173	6.327555838522091	8.193205378981371	0.5211907116443345	9.037574721516070

Table 2: Analytical solution and numerical solution using precise integration method of $\mathbf{W}_{C1}^{t}$.

Time/s	$t = 0.2$	$t = 0.4$	$t = 0.6$	$t = 0.8$	$t = 1.0$
Analytical solution	$\frac{1}{4} - \frac{7}{20e^{0.4}}$	$\frac{1}{4} - \frac{9}{20e^{0.8}}$	$\frac{1}{4} - \frac{11}{20e^{1.2}}$	$\frac{1}{4} - \frac{13}{20e^{1.6}}$	$\frac{1}{4} - \frac{15}{20e^{2}}$
Numerical solution	0.01538798388752625	0.04780196614725030	0.08434318344828887	0.1187672633034740	0.1484985375725405
Absolute error ($\times 10^{-17}$)	0.6580362078975456	1.966176961971875	1.964307811561096	2.324494307987107	1.99377149409639

Table 3: Analytical solution and numerical solution using precise integration method of $\mathbf{W}_C^t$.

Time/s	$t = 0.2$	$t = 0.4$
Analytical solution	$\begin{bmatrix} 0.2000000000000000 & 0.2214027581601699 \\ 0.2214027581601699 & 0.2459123488206352 \end{bmatrix}$	$\begin{bmatrix} 0.4000000000000000 & 0.4918246976412704 \\ 0.4918246976412704 & 0.6127704642462339 \end{bmatrix}$
Numerical solution	$\begin{bmatrix} 0.2000000000000000 & 0.2214027581601699 \\ 0.2214027581601699 & 0.2459123488206353 \end{bmatrix}$	$\begin{bmatrix} 0.4000000000000000 & 0.4918246976412705 \\ 0.4918246976412705 & 0.6127704642462341 \end{bmatrix}$
Absolute error ($\times 10^{-16}$)	$\begin{bmatrix} 0 & -0.2775557561562891 \\ -0.2775557561562891 & -1.387778780781446 \end{bmatrix}$	$\begin{bmatrix} 0 & -1.110223024625157 \\ -1.110223024625157 & -1.110223024625157 \end{bmatrix}$

Table 4: Analytical solution and numerical solution using precise integration method of $\mathbf{W}_{C1}^t$.

Time/s	$t = 0.2$	$t = 0.4$
Analytical solution	$\begin{bmatrix} 0.02000000000000000 & 0.02287779347186403 \\ 0.02287779347186403 & 0.02622629535380947 \end{bmatrix}$	$\begin{bmatrix} 0.8000000000000000 & 0.1049051814152379 \\ 0.1049051814152379 & 0.1387229535753766 \end{bmatrix}$
Numerical solution	$\begin{bmatrix} 0.02000000000000000 & 0.02287779347186414 \\ 0.02287779347186414 & 0.02622629535380947 \end{bmatrix}$	$\begin{bmatrix} 0.08000000000000000 & 0.1049051814152378 \\ 0.1049051814152378 & 0.1387229535753767 \end{bmatrix}$
Absolute error ($\times 10^{-16}$)	$\begin{bmatrix} 0 & -1.110223024625157 \\ -1.110223024625157 & -0.06938893903907228 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.2775557561562891 \\ 0.2775557561562891 & -0.8326672684688674 \end{bmatrix}$

$\mathbf{A}_1 = [-1]$, $\mathbf{A}_2 = [-1]$, $\mathbf{B} = [1]$ as these matrices are all of one-dimensional; the following results can be obtained using the analytical method:

$$\mathbf{W}_C^t = \frac{1 - e^{-2t}}{2}, \qquad \mathbf{W}_{C1}^t = \frac{1}{4} - \frac{2t + 1}{4e^{2t}}. \tag{23}$$

Using the precise integration method above, taking $N = 20$, numerical solutions of $\mathbf{W}_C^t$ and $\mathbf{W}_{C1}^t$ can be obtained. Compared with the analytical solution and calculating the errors, the results are shown in Tables 1 and 2.

From the data of tables, it can be seen that the difference between the result using method described in this paper and using analytical method is very small, so the numerical solution can be seen as the exact solution on the computer. Meanwhile, as the system is stable (poles are in the left half-plane), the steady-state value of Gram matrix can be calculated by using built-in functions gram () in MATLAB. In this example, this steady state value is 0.5. From Table 1, it can be seen that $\mathbf{W}_C(t)$ is increasing with the increase of time t and gradually approaching to the steady state value. When the time t iterates to $t = 10$ s, $\mathbf{W}_C(t)$ almost equals to the steady-state value. It also verifies the validity and effectiveness of the proposed method in this paper from another way.

Example 2. The system matrix is = $\left[\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}\right]$, $\mathbf{b} = \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]$; the integration time t is taken as 0.2 s and 0.4 s, respectively, compute Gram matrix $\mathbf{W}_C^t$ and $\mathbf{W}_{C1}^t$.

According to

$$\mathbf{W}_C^t = \int_0^t e^{\mathbf{A}_1\tau}\mathbf{B}e^{\mathbf{A}_2\tau}d\tau, \qquad \mathbf{W}_{C1}^t = \int_0^t \tau e^{\mathbf{A}_1\tau}\,\mathbf{B}e^{\mathbf{A}_2\tau}d\tau, \tag{24}$$

$\mathbf{A}_1 = \left[\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}\right]$, $\mathbf{A}_2 = \left[\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}\right]$, $\mathbf{B} = \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]$; these matrix are of two-dimensional, and system matrix is singular that is different from the previous example. The following results can be obtained using analytical method:

$$\mathbf{W}_C^t = \begin{bmatrix} t & e^t - 1 \\ e^t - 1 & \dfrac{e^{2t} - 1}{2} \end{bmatrix},$$
$$\mathbf{W}_{C1}^t = \begin{bmatrix} \dfrac{t^2}{2} & e^t\,(t-1) + 1 \\ e^t\,(t-1) + 1 & \dfrac{e^{2t}\,(2t-1) - 1}{4} \end{bmatrix}. \tag{25}$$

With the precise integration method, taking $N = 20$, numerical solutions of $\mathbf{W}_C^t$ and $\mathbf{W}_{C1}^t$ can be obtained. Compared with the analytical solution, the results are shown in Tables 3 and 4.

System matrix $\mathbf{a}$ is singular and the corresponding system is also unstable (the corresponding poles are $p_1 = 0$ and $p_2 = 1$, resp., p_2 is in the right half-plane); built-in functions gram () in MATLAB cannot be used to calculate the steady value. However, from the data in Tables 3 and 4, it can be seen that in this case methods described herein can also obtain very favorable effect, compared with the analytical method, up to ten or more significant digits. Therefore, in many cases, when other methods are unable to solve, the advantages of the method proposed in this paper are very obvious.

Example 3. In this example, the method is applied to a positioning gantry system, as shown in Figure 2. The model specification and parameters described in the following refer to paper [28]. In this model, a mass M is connected to a fixed surface by a linear spring with constant K_s. A flexible inelastic belt is connected to the mass and wraps around a pulley with radius r, which is mounted on a DC motor with armature resistance R_a and motor constant K_t. It is assumed that the rotor inertia of the motor and the inertia of the pulley are negligible. The motor will be actuated by a voltage signal. The

Table 5: Comparison of the results.

	$t = 0.5$
Reference [28]	$\begin{bmatrix} 6.541649458854454 \times 10^{-8} & 2.434956479172767 \times 10^{-11} \\ 2.434956479172767 \times 10^{-11} & 2.400278326897477 \times 10^{-5} \end{bmatrix}$
Method in this paper	$\begin{bmatrix} 6.542020950535066 \times 10^{-8} & 2.434956479158631 \times 10^{-11} \\ 2.434956479158631 \times 10^{-11} & 2.455144469453847 \times 10^{-5} \end{bmatrix}$
Absolute error	$\begin{bmatrix} 3.714916806121008 \times 10^{-12} & -1.413541806928020 \times 10^{-22} \\ -1.413541806928020 \times 10^{-22} & 5.486614255637052 \times 10^{-7} \end{bmatrix}$

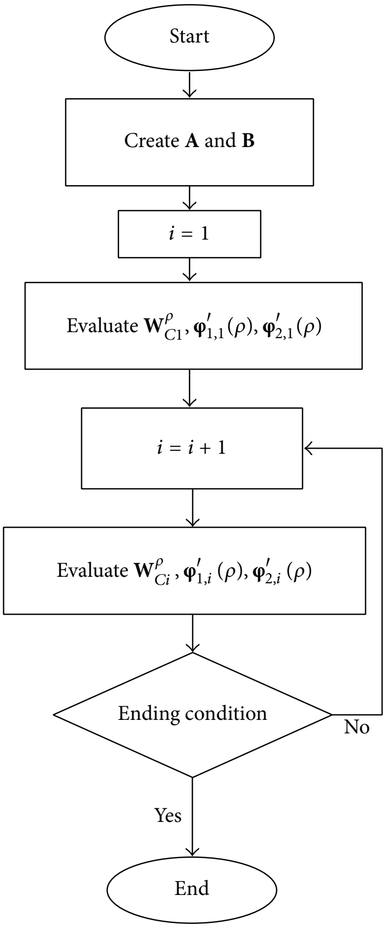
Start
Create **A** and **B**
$i = 1$
Evaluate $\mathbf{W}_{C1}^{\rho}, \boldsymbol{\varphi}_{1,1}'(\rho), \boldsymbol{\varphi}_{2,1}'(\rho)$
$i = i + 1$
Evaluate $\mathbf{W}_{Ci}^{\rho}, \boldsymbol{\varphi}_{1,i}'(\rho), \boldsymbol{\varphi}_{2,i}'(\rho)$
Ending condition
No
Yes
End

Figure 1: The block diagram of the method.

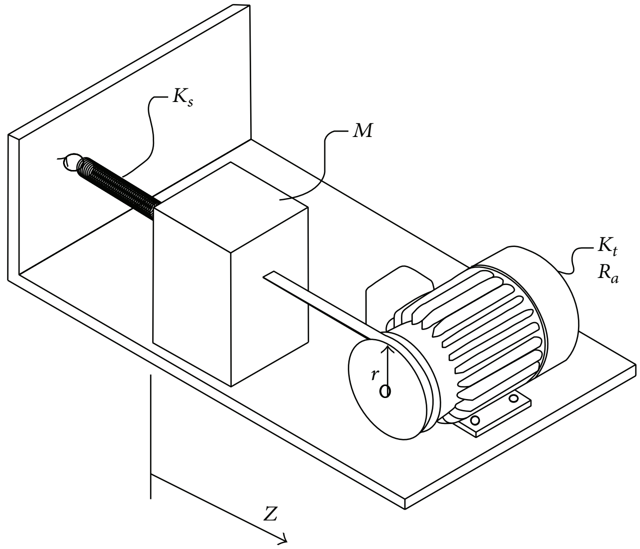

Figure 2: Configuration of positioning gantry [28].

displacement of the mass from its original position is Z. The system can be modeled by the following equations [28]:

$$\dot{\mathbf{x}} = \mathbf{a}\mathbf{x} + \mathbf{b}u,$$
$$\mathbf{a} = \begin{bmatrix} 0 & 1 \\ -\dfrac{k}{m} & -\dfrac{b}{m} \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 0 \\ \dfrac{1}{m} \end{bmatrix}, \tag{26}$$

where the terms m, b, and k are given by [28]

$$m = \frac{MrR_a}{K_t}, \qquad b = \frac{K_t}{r}, \qquad k = \frac{K_s r R_a}{K_t}. \tag{27}$$

According to (4), using the values shown in Table 2.1 in page 29 of [28], Gram matrix $\mathbf{W}_C^t$ can be calculated with the method proposed previews, and its results were also given by (3.21)–(3.25) in page 47-48 of [28]. Both of them are listed in Table 5.

From the data of Table 5, it can be seen that for positioning gantry system, the difference between the results is very small. This indicates that this method is effective.

4. Conclusions

This paper proposes a method using the key idea of precise integration for linear time-invariant systems to deal with the Gram matrix and its partial derivative matrix; accuracy can be compared with the exact solution in computer, the high efficiency due to making full use of the homogeneity of linear system to time, namely time-invariant properties of dynamic equation to the translation of time coordinate, and suitable for 2^N algorithm. To achieve high accuracy of results, the time is divided as small as possible. And since the method focuses on calculating the part of incremental, avoid numerical morbidity caused by large number eating decimal value. Besides, this method does not require matrix inversion, so no matter the system is singular or unstable, it can also compute correctly and effectively.

The approach is suitable for linear time-invariant system, but nonlinear and time-varying systems are much more in real life. The computation of Gram matrix and its partial derivative cannot be neglected in these cases. Methods described above bring a basis for these difficult tasks. Some kind of approximation needs to be introduced in precise integration method, such as perturbation method or making use of analytical functions to approximate. As some technique problems still remain, further research and discussions are needed on this issue.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This project is supported by National Natural Science Foundation of China (nos. 51305321 and 51035006) and the Fundamental Research Funds for the Central Universities (nos. K5051304021 and K5051304026). These supports are gratefully acknowledged. Many thanks are due to the editor, associate editor, and reviewers for their informative, valuable, and constructive comments.

References

[1] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, John Wiley & Sons, New York, NY, USA, 2005.

[2] V. K. Jain and R. D. Gupta, "Identification of linear systems through a Grammian technique," *International Journal of Control*, vol. 12, pp. 421–431, 1970.

[3] G. N. Korres, "A gram matrix-based method for observability restoration," *IEEE Transactions on Power Systems*, vol. 26, no. 4, pp. 2569–2571, 2011.

[4] M. C. de Almeida, A. V. Garcia, and E. N. Asada, "Active and reactive observability restoration subproblems in generalized power system state estimation," in *Proceedings of the 6th Power Systems Computation Conference (PSCC '08)*, pp. 1–8, Glasgow, UK, July 2008.

[5] G. N. Korres, "A numerical algorithm for observability analysis based on Gram matrix," in *Proceedings of the 7th Mediterranean Conference and Exhibition on Power Generation, Transmission, Distribution and Energy Conversion*, pp. 133–138, Ayia Napa, Cyprus, November 2010.

[6] G. N. Korres and N. M. Manousakis, "State estimation and observability analysis for phasor measurement unit measured systems," *IET Generation, Transmission, and Distribution*, vol. 6, no. 9, pp. 902–913, 2011.

[7] M. C. de Almeida, E. N. Asada, and A. V. Garcia, "A numerical method for finding spanning trees in power system state estimation," in *Proceedings of the International Conference on Power System Technology (PowerCon '06)*, pp. 1–6, Chongqing, China, October 2006.

[8] M. C. de Almeida, E. N. Asada, and A. V. Garcia, "On the use of Gram matrix in observability analysis," *IEEE Transactions on Power Systems*, vol. 23, no. 1, pp. 249–251, 2008.

[9] M. C. de Almeida, E. N. Asada, and A. V. Garcia, "Power system observability analysis based on gram matrix and minimum norm solution," *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1611–1618, 2008.

[10] M. C. de Almeida, E. N. Asada, and A. V. Garcia, "Identifying critical sets in state estimation using Gram matrix," in *Proceedings of the IEEE Bucharest PowerTech*, pp. 1–5, Bucharest, Romania, July 2009.

[11] V. Sreeram and K. S. Yong, "Evaluation of Gram matrix diagonal elements using inners technique," *Electronics Letters*, vol. 27, no. 12, pp. 1069–1071, 1991.

[12] V. Sreeram and P. Agathoklis, "On the computation of the Gram matrix in time domain and its application," *IEEE Transactions on Automatic Control*, vol. 38, no. 10, pp. 1516–1520, 1993.

[13] V. Sreeram and P. Agathoklis, "The computation of Gram matrix via impulse-response Grammians," in *Proceedings of the American Control Conference*, pp. 727–728, Boston, Mass, USA, June 1991.

[14] V. Sreeram and P. Agathoklis, "The computation of Gram matrix via the solution of Lyapunov equation," in *Proceedings of the 30th IEEE Conference on Decision and Control*, pp. 2092–2093, Brighten, UK, December 1991.

[15] M. Telescu, N. Tanguy, P. Bréhonnet, and P. Vilbé, "Efficient Gram-matrix computation for irrational resonant systems using Kautz models," *Signal Processing*, vol. 87, no. 12, pp. 3234–3240, 2007.

[16] M. Telescu, P. Bréhonnet, N. Tanguy, P. Vilbé, and L. C. Calvez, "Orthogonal decomposition of derivatives and antiderivatives for easy evaluation of extended Gram matrix," *Signal Processing*, vol. 86, no. 11, pp. 3486–3489, 2006.

[17] J. Pang, Y. Du, P. Hu, W. Li, and Z. D. Ma, "An energy conservation algorithm for nonlinear dynamic equation," *Journal of Applied Mathematics*, vol. 2012, Article ID 453230, 18 pages, 2012.

[18] W. X. Zhong and F. W. Williams, "Precise time step integration method," *Proceedings of the Institution of Mechanical Engineers C: Journal of Mechanical Engineering Science*, vol. 208, no. 6, pp. 427–430, 1994.

[19] Z. Wanxie, "Precise computation for transient analysis," *Computational Structural Mechanics and Applications*, vol. 12, no. 1, pp. 1–6, 1995.

[20] J. H. Lin, W. P. Shen, and F. W. Williams, "Accurate high-speed computation of non-stationary random structural response," *Engineering Structures*, vol. 19, no. 7, pp. 586–593, 1997.

[21] J. H. Lin, W. P. Shen, and F. W. Williams, "A high precision direct integration scheme for structures subjected to transient dynamic loading," *Computers & Structures*, vol. 56, no. 1, pp. 113–120, 1995.

[22] Y. X. Gu, B. S. Chen, and H. W. Zhang, "Precise time integration with dimension expanding method," *Acta Mechanica Sinica*, vol. 32, no. 4, pp. 447–456, 2000.

[23] T. Shujun and Z. Wanxie, "Precise integration method for Duhamel terms arising from non-homogenous dynamic systems," *Chinese Journal of Theoretical and Applied Mechanics*, vol. 39, no. 3, pp. 374–381, 2007.

[24] Z. Wan-Xie, "On precise integration method," *Journal of Computational and Applied Mathematics*, vol. 163, no. 1, pp. 59–78, 2004.

[25] M. Wang and F. T. K. Au, "Assessment and improvement of precise time step integration method," *Computers & Structures*, vol. 84, no. 12, pp. 779–786, 2006.

[26] M. F. Wang and F. T. K. Au, "On the precise integration methods based on Padé approximations," *Computers and Structures*, vol. 87, no. 5-6, pp. 380–390, 2009.

[27] Q. Liu, J. Zhang, and L. Yan, "A numerical method of calculating first and second derivatives of dynamic response based on Gauss precise time step integration method," *European Journal of Mechanics A: Solids*, vol. 29, no. 3, pp. 370–377, 2010.

[28] D. L. Peters, *Coupling and controllability in optimal design and control [Ph.D. thesis]*, The University of Michigan, 2010.

[29] T. Shuju, *Improvement of precise integration method and its application in dynamics and control [Ph.D. thesis]*, Dalian University of Technology, 2009.