Applied Mathematics and Computation 217 (2011) 7588–7598

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Effective partitioning method for computing generalized inverses and their gradients

Marko D. Petković *, Milan B. Tasić, Predrag S. Stanimirović

University of Niš, Faculty of Science and Mathematics, Višegradska 33, 18000 Niš, Serbia

ARTICLE INFO

Keywords:
Generalized inverses
Differentiation
Rational matrices
Polynomial matrices
Sparse matrices
Symbolic computation

ABSTRACT

We extend the algorithm for computing {1}, {1,3}, {1,4} inverses and their gradients from [11] to the set of multiple-variable rational and polynomial matrices. An improvement of this extension, appropriate to sparse polynomial matrices with relatively small number of nonzero coefficient matrices as well as in the case when the nonzero coefficient matrices are sparse, is introduced. For that purpose, we exploit two effective structures form [6], which make use of only nonzero addends in polynomial matrices, and define their partial derivatives. Symbolic computational package MATHEMATICA is used in the implementation. Several randomly generated test matrices are tested and the CPU times required by two used effective structures are compared and discussed.

1. Introduction

The following Penrose equations are crucial in pseudoinverses definition:

$$(1)\quad AXA = A,\quad (2)\quad XAX = X,\quad (3)\quad (AX)^T = AX,\quad (4)\quad (XA)^T = XA.$$

For a subset ‡ of the elements from the set {1,2,3,4}, the set of matrices obeying the equations determined by the set ‡ is denoted by $A\{‡\}$. A matrix from $A\{‡\}$ is called an ‡-inverse of A and it is denoted by $A^{‡}$. The Moore–Penrose inverse is the unique matrix satisfying all the Eqs. (1)–(4), and it is denoted by $A^{\dagger}$.

A lot of methods were proposed to compute various generalized inverses of a matrix. These include methods arising from the Cayley–Hamilton theorem, the full-rank factorization and the singular value decomposition (see for instance, [1,15]). Greville in [2] proposed a finite recursive algorithm for determining the Moore–Penrose inverse. Due to its ability to undertake sequential computing, this method has been extensively applied in statistical inference, filtering theory, linear estimation theory, optimization and also in analytical dynamics [4]. About a decade ago, Udwadia and Kalaba gave an alternative and a simple constructive proof of Greville's formula [12]. A generalization of Greville's method to the weighted Moore–Penrose inverse is introduced in [14]. The results in [14] are established by using a new technique.

Symbolic computation of generalized inverses and their gradients is one of the most interesting areas of computer algebra. Matrix differentiation is of considerable importance in statistics. It is especially useful in connection with the maximum likelihood estimation of the parameters in a statistical model. The maximum likelihood estimates of the model's parameters satisfy the equations (known as the likelihood equations) obtained by equating to zero the first-order partial derivatives (with respect to model's parameters) of the logarithm of the so-called likelihood function [3]. In many important cases, the likelihood function involves the determinant and/or inverse of a matrix. The gradient of the pseudo-inverse may be needed for sensitivity analysis, optimizations or in the nonlinear least squares problems.

* Corresponding author.
E-mail addresses: dexterofnis@gmail.com (M.D. Petković), milan12t@ptt.rs (M.B. Tasić), pecko@pmf.ni.ac.rs (P.S. Stanimirović).

0096-3003/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.amc.2011.02.051

There is a lot of extensions of the partitioning method to sets of rational and polynomial matrices. The algorithm for the computation of the Moore–Penrose inverse of the one-variable polynomial and/or rational matrix, based on the Greville's partitioning algorithm, was introduced in [8]. The extension of results from [8] to the set of the two-variable rational and polynomial matrices is introduced in the paper [7]. The Wang's partitioning method from [14], aimed in the computation of the weighted Moore–Penrose inverse, is extended to sets of one-variable rational and polynomial matrices in the paper [9]. Also the efficient algorithm for computing the weighted Moore–Penrose inverse, appropriate for the polynomial matrices where only a few polynomial coefficients are nonzero, is established in [6]. Udwadia and Kalaba derived in [13] a constructive procedure for determining different types of generalized inverses for constant matrices. These results are extended to one-variable rational and polynomial matrices in [10].

An efficient method for direct simultaneous computation of the Moore–Penrose inverse in conjunction with its gradient is derived in [5]. Layton in [5] used the approach to simply differentiate terms arising from the Grevile's partitioning method. The resulting algorithm in [5] is usable and efficient because of its unique property that it requires only elementary matrix operations, such as addition, subtraction and multiplication. In the paper [11] the Layton's method is combined with the representation of the Moore–Penrose inverse of one-variable polynomial matrix from [8]. In consequence, the algorithm for computing the gradient of the Moore–Penrose inverse for one-variable polynomial matrix is developed. Moreover, using the representation of various types of pseudo-inverses from [10], more general algorithms for computing partial derivatives of {1}, {1,3} and {1,4} inverses of one-variable rational and polynomial matrices are derived in [11].

As usual, let $\mathbb{R}$ be the set of real numbers, $\mathbb{R}^{m\times n}$ be the set of complex $m \times n$ matrices, and $\mathbb{R}_r^{m\times n} = \{X \in \mathbb{R}^{m\times n} | \text{rank}(X) = r\}$. Let $A \in \mathbb{R}(s_1, s_2, \ldots, s_p)^{m\times n}$ be arbitrary multi-variable rational or polynomial matrix. In order to make a notation shorter we denote $S = (s_1, s_2, \ldots, s_p)$ and write $A(S)$ instead of $A(s_1, s_2, \ldots, s_p)$. Let $A_i(S)$ be the submatrix of $A(S)$ consisting of first i columns of $A(S)$. If ith column of $A(S)$ is denoted by $a_i(S)$, then it is obvious that $A_i(S)$ is partitioned as $A_i(S) = [A_{i-1}(S)|a_i(S)]$, $i = 2, \ldots, n$, assuming $A_1(S) = [a_1(S)]$. The set of polynomials (resp. rational functions) with complex coefficients in the variables S are denoted by $\mathbb{R}[S]$ (resp. $\mathbb{R}(S)$). The set of $m \times n$ matrices with elements in $\mathbb{R}[S]$ (resp. $\mathbb{R}(S)$) are denoted by $\mathbb{R}[S]^{m\times n}$ (resp $\mathbb{R}(S)^{m\times n}$). Also by $\mathbf{O}$, we denote an appropriate zero matrix and by $\mathbf{0}$, appropriate zero vector.

Main results of the present article are summarized in the following.

– Algorithms from [10], which give recursive rules for computation of $A_i^{\ddagger}(s)$ in terms of $A_{i-1}^{\ddagger}(s)$, $s \in \mathbb{R}$, are extended from the single-variable polynomial matrix case to the multi-variable polynomial matrix case.
– Algorithm which gives recurrent relations between the partial derivatives $\frac{\partial A_i^{\ddagger}(S)}{\partial s_k}$ and $\frac{\partial A_{i-1}^{\ddagger}(S)}{\partial s_k}$, established in [11], is improved in the case when a great number of coefficient matrices vanishes to zero matrix as well as in the case when the nonzero coefficient matrices are sparse. In order to define effective algorithm for computing partial derivatives of generalized inverses, it is necessary to define partial derivatives of effective structures defined in the papers [6,7].

Generally, the present paper is a continuation of the papers [6,7,9–11] on multi-variable polynomial matrices and sparse matrices.

Extension of results from [10,11] to multi-variable polynomial matrices is presented in the second section. In this way, a finite recursive algorithm for symbolic computation of generalized inverses $A_i^{\ddagger}(S)$ and their gradients is derived. Algorithms effectively applicable to sparse polynomial matrices are developed in the third section. Implementation, evaluated in the package MATHEMATICA, is exploited in development of several illustrative examples in the last section. A comparison of two used effective structures is given.

2. Multi-variable polynomial matrix case

Udwadia and Kalaba [13] proved the following theorem which gives the expressions for compute generalized inverses of partitioned matrix $A_k = [A_{k-1}|a_k]$ where $A_{k-1} \in \mathbb{R}^{m\times k}$, $a_k \in \mathbb{R}^{m\times 1}$ and $k \in \{2, 3, \ldots, n\}$.

Theorem 2.1 [13]. *Denote by ‡ any of following generalized inverses: {1}, {1,3}, {1,4} or {1,2,3,4}. Also let* $c_k = (I - A_k A_k^{\ddagger}) a_k$ *and* $d_k = A_k^{\ddagger} a_k$. *Then*

$$A_k^{\ddagger} = \begin{bmatrix} A_{k-1}^{\ddagger} - d_k b_k^T \\ b_k^T \end{bmatrix}$$

where b_k *is given by:*

1. *If* $c_k = \mathbf{0}$ *then*

$$b_k = \frac{(A_{k-1}^{\ddagger})^T d_k}{1 + d_k^T d_k}, \quad \ddagger = \{1,2,3,4\} \text{ or } \ddagger = \{1,4\}$$

and b_k *is an arbitrary vector from* $\mathbb{R}^{m\times 1}$ *if* ‡ = {1,3} *or* {1}.

2. *Otherwise, if* $c_k \neq \mathbf{0}$ *then*

$$b_k = \frac{c_k^T (I - A_{k-1} A_{k-1}^{\ddagger})}{c_k^T c_k}, \quad \ddagger = \{1,4\} \text{ or } \ddagger = \{1\}$$