Computers and Mathematics with Applications 71 (2016) 489–502

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

# Incremental numerical recipes for the high efficient inversion of the confluent Vandermonde matrices

Jerzy S. Respondek *

*Silesian University of Technology, Faculty of Automatic Control, Electronics and Computer Science, Institute of Computer Science, ul. Akademicka 16, 44-100 Gliwice, Poland*

ARTICLE INFO

*Article history:*
Received 27 May 2014
Received in revised form 4 September 2015
Accepted 12 December 2015
Available online 9 January 2016

*Keywords:*
Binary tree
Matrix inversion
Confluent Vandermonde matrices
Computational complexity

ABSTRACT

In the article, we propose an incremental algorithm for calculating the inversion of the confluent Vandermonde matrix and triangular factorization of this inversion. We implemented all the incremental operations, i.e. adding, deleting and changing the single matrix parameter to avoid repeating the same calculations again and again. Besides, contrary to other works in this field, the article derives an explicit analytic formula for the calculation of the triangular factorization of the confluent Vandermonde matrix inversion. Additionally, we propose a solution to these two problems with the use of a system of linear recursive equations. The results of this article do not require any symbolic calculations. Therefore they can be performed by a numerical algorithm implemented in any general-purpose programming language.



## 1. Introduction

This paper gives three results in the confluent Vandermonde special matrices topic:

- We present an explicit formula of the triangular factorization of the inversion.
- We derive a system of linear recursive equations for the inversion of the confluent Vandermonde matrix and its triangular factorization.
- We construct a dynamical data structure to implement incremental algorithms for adding, removing and changing the value and/or multiplicity of a selected matrix parameter.

The main results of this article are presented in Sections 2–4. These results do not require any symbolic calculations. Thus they can be performed by a numerical algorithm implemented in any general-purpose programming language.

The traditional data structures used by the numerical algorithms, including algorithms operating on matrices, are the 1D table and the matrix. This topic is presented for the arbitrary number of dimensions in [1]. The exception to that is the algorithm for the N-body system simulation, classical problem in physics, whose implementation without the use of dynamic data structures gives weak $O(n^2)$ time complexity. The famous article *An efficient program for many-body simulation* ([2], [3, ch. 5.1]) proposed the adaptically changing tree which stored in its nodes the whole groups of bodies. Such an implementation of the N-body simulation algorithm, with the use of tree data structure, improved the time complexity to $O(n \log n)$. Afterwards it was shown in [4] that the obtained efficiency is even better i.e. $O(n)$. This problem still remains a subject of research, e.g. [5].

* Tel.: +48 32 237 2151; fax: +48 32 237 2733.
*E-mail address:* jerzy.respondek@polsl.pl.

http://dx.doi.org/10.1016/j.camwa.2015.12.016
0898-1221/

The objective of Section 4 is to use a binary tree of the specially designed structure, together with a series of doubly-linked lists, to construct a set of incremental algorithms operating on the confluent Vandermonde matrices. The algorithms we designed are the matrix inversion and its triangular factorization with all the possible incremental operations, i.e. adding or removing a matrix parameter and changing its value and/or multiplicity. The implementation of all the incremental operations in a manner that avoids repeating all the calculations anew results in a better time complexity than the one obtained by the ordinary algorithms. That was possible thanks to the flexibility of the binary tree and doubly-linked lists, which are traditional in computer science but rarely used by the numerical algorithms specialists.

The paper is organized as follows: Section 1.1 justifies the importance of the confluent Vandermonde matrices, in Section 1.2, we compare the new results with others, Section 1.3 provides the necessary theoretical background, Section 2 provides the theorem for the explicit description of the entries of the triangular factorization of the inversion, Section 3—the linear recursive equation for calculating the desired inversion and its triangular factorization, in Section 4, we present the incremental algorithm operating on the confluent Vandermonde matrix together with a proper tree data structure and determine the computational complexities of the presented algorithm, in Section 5, we present the example of the proposed algorithm execution, in Section 6, we present the example of an application of the confluent Vandermonde matrices, in Section 7, we carry out performance tests, in Section 8, we perform accuracy tests and in Section 9 we give some conclusions.

### 1.1. Importance of the confluent Vandermonde matrices

The confluent Vandermonde matrices are formally defined in Section 1.3. They arise in a broad range of both theoretical and practical problems. Below we surveyed the problems which need to operate on the confluent Vandermonde matrices.

- Linear, ordinary differential equations: the Jordan transition matrix of the ODE, in the matrix differential equation form with the Frobenius companion matrix, is a confluent Vandermonde matrix ([6, pp. 86–95]).
- Control problems: investigating the so-called controllability [7,8] of the higher order systems with multiple characteristic polynomial zeros leads to the problem of inverting the confluent Vandermonde matrix [9]. As the examples of the higher order models of the physical objects one can mention Timoshenko's elastic beam equation [10] (4th order) and the Korteweg–de Vries equation of waves on shallow water surfaces [11] (3rd–7th order).
- Interpolation: apart from the ordinary polynomial interpolation with single nodes we consider the Hermite interpolation, allowing the multiple interpolation nodes. This augmented interpolation enables to predefine the derivatives of the interpolation polynomial in the data points. This way we gain better control over the interpolation curve, e.g. over its sloping and convexity in the data points. That problem leads to the system of linear equations, with the confluent Vandermonde matrix ([12, pp. 363–373]).
- The confluent Vandermonde matrix is used in coding information in the Hermitian code [13].
- Optimization of the non-homogeneous differential equations [14].
- Other types of special matrices are intensively investigated e.g. [15].
- The classic Vandermonde matrix remains a subject of research [16,17].

### 1.2. Comparison with other methods

In the literature the problem of inverting the confluent Vandermonde matrices and their inversion's triangular factorization has been tackled several times.

- The articles [18,19] present algorithms with $O(n^2)$ time complexity in a special case of all small matrix parameter multiplicities, but in a general case their complexity has a cubic order.
- The articles [20,21] propose a method which requires hand-held calculation of the fractional expansion of the polynomial inversion. The numerical part of the proposed methods is of the weak $O(n^4)$ class.
- The article [22] proposes algorithms of the $O(nr)$ order in general case, but it also requires the knowledge of all the matrix parameters at once.
- A step forward is made in the article [23], which enables to add new parameters, however it is still impossible to change or delete them.

To sum up, there are known algorithms operating on the confluent Vandermonde matrices. However, none of them enables to perform the incremental operations freely and efficiently. Moreover, some of the previous results cannot be even performed numerically as they require sophisticated symbolic calculations.

### 1.3. Theoretical background

Let us explain what the confluent Vandermonde matrix $V$ is. Let $\lambda_1, \lambda_2, \ldots, \lambda_r$ be the given real pair wise distinct zeros of the polynomial $p(s) = (s - \lambda_1)^{n_1} \ldots (s - \lambda_r)^{n_r}$ with multiplicities $n_1 + \cdots + n_r = n$. The confluent Vandermonde matrix