

# Numerical Differentiation and the Solution of Multidimensional Vandermonde Systems

By G. Galimberti and V. Pereyra

**Abstract.** We define multidimensional Vandermonde matrices (MV) to be certain submatrices of Kronecker products of standard Vandermonde matrices. These MV matrices appear naturally in multidimensional problems of polynomial interpolation. An explicit algorithm is produced to solve systems of linear equations with MV matrices of coefficients. This is an extension of work of Stenger for the two-dimensional case. Numerical results for three-dimensional numerical differentiation are given.

**1. Introduction.** Multidimensional Vandermonde matrices and their associated systems of linear equations will be introduced in Section 2. They are obtained as some special submatrices of Kronecker products of standard Vandermonde matrices. Systems of this type appear naturally in multidimensional problems of numerical differentiation, interpolation, and so on.

Stenger [2] has studied the solvability of a more general class of submatrices of multidimensional Kronecker products and has given an explicit algorithm for its solution, but only in two dimensions. In fact, our work consists of an extension of Stenger's results (in a restricted case) to dimensions  $n > 2$ . We also give explicit computational algorithms for  $n = 2, 3$ .

The problem that motivated this research is the following:

Let  $f(\mathbf{x}) = f(x_1, \dots, x_n)$  be an  $m$ -times continuously differentiable function on a domain  $E$  of the  $n$ -dimensional space  $R^n$ . Let  $L$  be a linear differential operator of order  $k < m$ , of the form:

$$(1.1) \quad L(f(\mathbf{x})) = \sum_{i=0}^k \sum_{\mu_1 + \dots + \mu_n = i} \frac{1}{\mu_1! \dots \mu_n!} a_{\mu}(\mathbf{x}) D_{x_1}^{\mu_1} D_{x_2}^{\mu_2} \dots D_{x_n}^{\mu_n} f(\mathbf{x}),$$

where the coefficients  $a_{\mu_1 \dots \mu_n}$  are given functions.

Let  $\{\Delta_r\}$  be a set of "increments" of  $\hat{\mathbf{x}}$ , i.e.:

$$\hat{\mathbf{x}} + \Delta_r \in E \subset R^n, \quad \text{and} \quad \|\Delta_r\|_{\infty} \leq h.$$

We are interested in producing numerical differentiation formulae in terms of ordinates, which approximate  $L(f(\mathbf{x}))$  at a given point  $\mathbf{x} = \hat{\mathbf{x}}$ :

$$(1.2) \quad L_{\Delta}(f(\hat{\mathbf{x}})) \equiv \sum_r \omega_r f(\hat{\mathbf{x}} + \Delta_r) = L(f(\hat{\mathbf{x}})) + O(h^{p+1}),$$

where  $k \leq p < m$ .

If we expand the left-hand side of this last equation in a multiple Taylor formulae at  $\hat{\mathbf{x}}$  and equate coefficients corresponding to the same derivatives in  $L(f(\hat{\mathbf{x}}))$ , then the following system of linear equations for the  $\omega_r$  is obtained:

Received July 14, 1969.

AMS Subject Classifications. Primary 6520, 6555.

Key Words and Phrases. Multidimensional Vandermonde matrices, numerical differentiation, Kronecker products, Vandermonde systems of equations.

$$(1.3) \quad \sum_r \delta_{1r}^{\mu_1} \cdots \delta_{nr}^{\mu_n} \omega_r = a_{\mu_1, \dots, \mu_n}(\hat{\mathbf{x}}), \quad 0 \leq \sum_{i=1}^n \mu_i \leq p,$$

where  $\Delta_r = (\delta_{1r}, \dots, \delta_{nr})$ .

We show in Section 2 how to determine the set of increments  $\{\Delta_r\}$ , in order that system (1.3) be solvable, and, furthermore, we develop an algorithm to recursively obtain the  $\omega_r$ .

In Section 3, we make this algorithm explicit for the cases  $n = 2, 3$ , and in Section 4 we present some numerical results, obtained with a computer implementation. Of course, since systems like (1.3) appear in many other problems, the algorithm has a much wider field of applicability.

It is appropriate to remark that the operation count for our algorithm, corresponding to the solution of an  $n$ -dimensional Vandermonde system of order  $p$  (see Section 2), is proportional to  $p^{n+1}$ . For  $n = 3$ , such a system has size  $\frac{1}{6}p(p+1)(p+2)$ , and assuming that Gaussian elimination were applicable (which in general will not be the case), the operation count would be proportional to  $p^9$ , as compared with  $p^4$  for the algorithm developed here. Another important point is that the system matrix need not be explicitly generated at any time.

Stoyakovich [3] considers the problem of polynomial interpolation in  $n$  dimensions. His approach includes also the solution of some systems of linear equations, but differs from ours in several respects.

**2. Multidimensional Vandermonde Systems.** The Kronecker product of two matrices  $A, B$ , is a new matrix of the form:

$$(2.1) \quad A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}.$$

We shall use, in what follows, the usual conventions for multi-subscripts:  $\mathbf{s} = (s_1, \dots, s_n)$ ,  $s_i$  nonnegative integers;  $|\mathbf{s}| = \sum_{i=1}^n s_i$ . If  $\mathbf{y} = (y_1, \dots, y_n)$ , then  $\mathbf{y}^{\mathbf{s}} \equiv y_1^{s_1} \cdots y_n^{s_n}$ .

Let  $n, p, q$  be given integers,  $n \geq 2, q \geq p \geq 2$ . Let  $I = \{x_{js} \mid (j, s) \in \{1, \dots, n\} \times \{1, \dots, q\}\}$  be a set of real numbers satisfying  $x_{js_1} \neq x_{js_2}$  if  $s_1 \neq s_2$ . Let us form with these numbers the Vandermonde matrices  $V_j = V_j(x_{j1}, \dots, x_{jq}), j = 1, \dots, n$ .

Let  $K$  be their Kronecker product  $K = V_1 \otimes V_2 \otimes \cdots \otimes V_n$ .

The elements of  $K$  are of the form

$$(2.2) \quad K_{\mathbf{s}}^{\boldsymbol{\mu}} \equiv K_{s_1, \dots, s_n}^{\mu_1, \dots, \mu_n}(x_{1s_1}^{\mu_1}, \dots, x_{ns_n}^{\mu_n}) = (\mathbf{x}_{\mathbf{s}}^{\boldsymbol{\mu}}).$$

In order to introduce an adequate extension to  $n$  dimensions of Vandermonde matrices, we shall select some rows and columns out of  $K$ . First of all, we choose those rows for which  $|\boldsymbol{\mu}| \leq p$ . This can be better described by asking the components of  $\boldsymbol{\mu}$  to satisfy:

$$(2.3) \quad \begin{aligned} \mu_1 &\leq p, \\ \mu_2 &\leq p - \mu_1, \\ &\vdots \\ \mu_n &\leq p - \sum_{i=1}^{n-1} \mu_i. \end{aligned}$$

The set of all  $n$ -tuples defined by (2.3) will be denoted by  $M(n, p)$ . The number of elements of  $M(n, p)$  will be denoted by  $m(n, p)$ . Observe that this set satisfies:

$$(2.4) \quad M(n, p) = \sum_{\mu_1=0}^p \{(\mu_1, \boldsymbol{\mu}'): \boldsymbol{\mu}' \in M(n-1, p-\mu_1)\},$$

where the sum indicates reunion of disjoint sets.

The set of all  $n$ -tuples  $(x_{1s_1}, \dots, x_{ns_n})$  with  $x_{is_i} \in I$ , will be called  $X$ .

Now we shall select an equal number of columns of  $K$  by choosing  $m(n, p)$  different elements of  $X$  and taking the columns associated with them.

The submatrix so defined will be called an  $n$ -dimensional Vandermonde matrix of order  $p$  and be denoted by  $\Omega$ . It is clear that 1-dimensional matrices coincide with the usual Vandermonde matrices.

Given  $\boldsymbol{\omega}$  and  $\mathbf{d}$ ,  $m(n, p)$ -vectors, an  $n$ -dimensional Vandermonde system of order  $p$  will be defined by

$$(2.5) \quad \Omega \boldsymbol{\omega} = \mathbf{d}.$$

Observe that for  $n > 1$ , the matrix  $\Omega$  may be singular.

*Definition.* A set of  $m(n, p)$  points  $\Gamma \subset X$  shall be called  $(p-1)$ -independent, iff the corresponding  $n$ -dimensional Vandermonde matrix of order  $p$  is nonsingular.

We describe now a procedure for choosing  $m(n, p)$  points (in many different ways), which we shall eventually prove to produce a  $(p-1)$ -independent set. In the course of the proof, we shall elaborate a method to solve the system of linear equations (2.5) in this nonsingular case.

We shall choose first a set of subscripts in an appropriate way.

For  $j_n = 0, \dots, p$ , choose arbitrary (distinct)

$$1 \leq s(j_n) \leq q.$$

For  $j_n = 0, \dots, p; j_{n-1} = 0, \dots, p - j_n$ , choose

$$1 \leq s(j_{n-1}, j_n) \leq q, \quad s(j_{n-1}, j_n) \neq s(j'_{n-1}, j_n) \quad \text{if } j_{n-1} \neq j'_{n-1}.$$

For  $k = n-2, n-3, \dots, 1$  and for  $j_n = 0, \dots, p; j_{n-1} = 0, \dots, p - j_n; \dots; j_k = 0, \dots, p - \sum_{i=k+1}^n j_i$ , choose  $1 \leq s(j_k, \dots, j_n) \leq q$ , satisfying

$$s(j_k, j_{k+1}, \dots, j_n) \neq s(j'_k, j_{k+1}, \dots, j_n) \quad \text{if } j_k \neq j'_k.$$

To shorten somewhat the notation, we introduce the partial vector subscripts  $J_k = (j_k, \dots, j_n)$ .

Finally, the chosen  $n$ -tuples are:

$$(x_{1,s(J_1)}, x_{2,s(J_2)}, \dots, x_{n,s(J_n)}).$$

The set of all  $n$ -subscripts  $J_1 = (j_1, \dots, j_n)$  satisfying the above conditions will be denoted by  $M^*(n, p)$ . Obviously, the number of elements of  $M^*(n, p)$  is  $m(n, p)$ . A relation similar to (2.4) is satisfied by  $M^*(n, p)$ :

$$M^*(n, p) = \sum_{|J_2|=0}^p \{(j_1, J_2): J_2 \in M^*(n-1, p), j_1 \leq p - |J_2|\}.$$

Once the correspondence between  $n$ -tuples  $J_1$  and  $n$ -tuples  $(s(J_1), \dots, s(J_n))$  is

established, we can use, without ambiguity, the notation  $\mathbf{x}_{J_1}$  for  $(x_{1,s(J_1)}, \dots, x_{n,s(J_n)})$ . We shall do so.

In subscript notation, (2.5) (for the above choice of basis points) is:

$$(2.6) \quad \sum_{J_1 \in M^*(n,p)} \mathbf{x}_{J_1}^{\mu} \omega_{J_1} = d_{\mu}, \quad \mu \in M(n, p).$$

We are interested in solving systems of this form. As we said in the Introduction, they are associated with many problems of approximation in  $n$ -dimensional spaces.

We shall show first how to reduce the solution of system (2.6) to the solution of some  $(n - 1)$ -dimensional Vandermonde systems of order  $\leq p$ . Proceeding with this reduction, we shall finally arrive at nonsingular one-dimensional Vandermonde systems, which we certainly know how to solve [1]. This constructive proof will give us an algorithm for solving the system and, simultaneously, a proof of the independence of the basis points. This is an extension of a result of Stenger [2], who proves the same for  $n = 2$ .

An important case is  $n = 3$ , for which we shall give a complete detailed algorithm in Section 3.

We then assume that  $(n - 1)$ -dimensional Vandermonde systems of order  $\leq p$ , with basis points chosen as explained above, can be solved.

The first step is to obtain an adequate reformulation of the system in terms of certain parameters:

$$(A) \quad c_{J_2}^{(\mu_1)} = \sum_{j_1=0}^{p-|J_2|} x_{J_1}^{\mu_1} \omega_{J_1}, \quad J_2 \in M^*(n-1, p), \quad \mu_1 = 0, \dots, p;$$

$$(B) \quad \sum_{J_2 \in M^*(n-1,p)} \mathbf{x}_{J_2}^{\mu'} c_{J_2}^{(\mu_1)} = d_{\mu}, \quad \mu = (\mu_1, \mu') \in M(n, p),$$

where

$$\mathbf{x}_{J_2}^{\mu'} = x_{J_2}^{\mu'_2} \cdot x_{J_2}^{\mu'_3} \cdots x_{J_2}^{\mu'_n}.$$

If we set now  $\mu_1 = 0$  in (B), a square system for the unknown parameters  $c_{J_2}^{(0)}$  is obtained:

$$(2.7) \quad \sum_{J_2 \in M^*(n-1,p)} \mathbf{x}_{J_2}^{\mu'} c_{J_2}^{(0)} = d_{0,\mu'}, \quad \mu' \in M(n-1, p).$$

This is an  $(n - 1)$ -dimensional Vandermonde system of order  $p$  and, according to our inductive hypothesis, it can be solved.

(i) Solving (2.7), we obtain  $c_{J_2}^{(0)}$ ,  $J_2 \in M^*(n - 1, p)$ .

With these values we can enter into those equations (A) which have  $|J_2| = p$ , in order to obtain the corresponding  $\omega_{J_1}$ :

(ii)  $\omega_{J_1} = c_{J_2}^{(0)}$ ,  $j_1 = 0, |J_2| = p$ .

Using these values in equations (A) with  $\mu_1 = 1, 2, \dots, p$ , and the same  $J_2$  as in (ii), we get the remaining parameters  $c_{J_2}^{(\mu_1)}$ :

(iii)  $c_{J_2}^{(\mu_1)} = x_{J_1}^{\mu_1} \omega_{J_1}$ ,  $|J_2| = p, \mu_1 \leq p$ .

This ends the first step of the reduction. Now we shall make an induction on  $\mu_1$ . Let us assume that for  $\mu_1 = 0, \dots, r - 1$  we know:

$$(I_{r-1}) \begin{cases} \omega_{J_1}, & |J_2| \geq p - r + 1; \\ c_{J_2}^{(\mu_1)}, & |J_2| \geq p - r + 1, \quad \mu_1 = 0, \dots, p; \\ c_{J_2}^{(\mu_1)}, & |J_2| \leq p - r, \quad \mu_1 = 0, \dots, r - 1. \end{cases}$$

This is certainly true for  $r = 1$ .

We shall show that from these known quantities we can compute the corresponding ones for  $\mu_1 = r$ .

First of all, we partition the set  $M^*(n - 1, p)$  into two disjoint sets:

$$M(n - 1, p) = \widehat{M}(n - 1, p - r) + M^*(n - 1, p - r),$$

where  $\widehat{M}(n - 1, p - r) = \{J_2 \in M^*(n - 1, p) : |J_2| \geq p - r + 1\}$ .

We can now write (B) in the form:

$$(2.8) \quad \sum_{J_2 \in M^*(n-1, p-r)} \mathbf{x}_{J_2}^{\mu'} c_{J_2}^{(r)} = d_{r, \mu'} - \sum_{J_2 \in \widehat{M}(n-1, p-r)} \mathbf{x}_{J_2}^{\mu'} c_{J_2}^{(r)}, \quad \mu' \in M(n - 1, p - r).$$

According to  $(I_{r-1})$ , the right-hand side is known. Thus, (2.8) is an  $(n - 1)$ -dimensional Vandermonde system of order  $(p - r)$ , which can be solved in order to provide the unknown parameters  $c_{J_2}^{(r)}$  for  $J_2 \in M^*(n - 1, p - r)$ .

Considering now (A) for all  $J_2 \in M^*(n - 1, p)$  satisfying  $|J_2| = p - r$ , we get:

$$(2.9) \quad c_{J_2}^{(\mu_1)} = \sum_{j_1=0}^r x_{j_1}^{\mu_1} \omega_{J_1}, \quad J_2 \in M^*(n - 1, p), \quad |J_2| = p - r, \quad \mu_1 = 0, \dots, r.$$

The  $c_{J_2}^{(\mu_1)}$  are known from  $(I_{r-1})$  for  $\mu_1 = 0, \dots, r - 1$ ,  $|J_2| = p - r$ , and the  $c_{J_2}^{(r)}$  have just been calculated. Therefore, (2.9) is a one-dimensional, nonsingular Vandermonde system of order  $r$ , which we can solve for the unknown quantities  $\omega_{J_1}$ ,  $j_1 = 0, \dots, r$ ,  $|J_2| = p - r$ .

Once these are obtained, we can feed them back into (A) to compute the remaining  $c_{J_2}^{(\mu_1)}$ :

$$c_{J_2}^{(\mu_1)} = \sum_{j_1=0}^r x_{j_1}^{\mu_1} \omega_{J_1}, \quad \mu_1 = r, \dots, p, \quad |J_2| = p - r.$$

This completes the induction step  $r$ , since all the necessary quantities have been obtained.

After the  $r = p$  step we would have computed all the  $\omega_{J_1}$ , and the system (2.6) is solved.

**3. Explicit Algorithms for  $n = 2, 3$ .** We explicit now the general algorithm, described in Section 2 for the cases  $n = 2, 3$ .

In order to avoid the confusing shifting of subscripts that is necessary in some computer languages, we shall restate our problem in a slightly different way. For  $n = 2$ , given  $p \geq 2$ ,  $\beta_j$  ( $j = 1, \dots, p$ ),  $\alpha_{ij}$  ( $j = 1, \dots, p; i = 1, \dots, p + 1 - j$ ),  $d_{v\mu}$  ( $v = 1, \dots, p; \mu = 1, \dots, p + 1 - v$ ), where the  $\beta$ 's and  $\alpha$ 's satisfy the conditions of Section 2:  $\beta_j \neq \beta_k$  if  $j \neq k$ ,  $\alpha_{ij} \neq \alpha_{kj}$  if  $i \neq k$ , we would like to solve the system of linear equations:

$$(3.1) \quad \sum_{j=1}^p \sum_{i=1}^{p+1-j} \alpha_{ij}^{v-1} \beta_j^{\mu-1} \omega_{ij} = d_{v\mu}$$

for the unknowns  $\omega_{ij}$ .

The algorithm is developed in  $p$  steps.

*Algorithm BIDIM.* First we define  $d_{v\mu}^{(1)} = d_{v\mu}$ .

*Step  $k$*  ( $k = 1, \dots, p$ ):

(i) Obtain  $c_{kj}$  ( $j = 1, \dots, p + 1 - k$ ) by solving the one-dimensional Vandermonde system [1]:

$$\sum_{j=1}^{p+1-k} \beta_j^{\mu-1} c_{kj} = d_{k\mu}^{(k)}, \quad \mu = 1, \dots, p+1-k.$$

(If  $k = p$  then  $c_{p1} = d_{p1}^{(p)}$ .)

(ii) Obtain  $\omega_{i,p+1-k}$  ( $i = 1, \dots, k$ ) by solving the Vandermonde system:

$$\sum_{i=1}^k \alpha_{i,p+1-k}^{\nu-1} \omega_{i,p+1-k} = c_{\nu,p+1-k}, \quad \nu = 1, \dots, k,$$

(if  $k = 1$ ,  $\omega_{1p} = c_{1p}$ ; if  $k = p$  the process ends).

(iii) Compute  $c_{\nu,p+1-k}$  ( $\nu = k+1, \dots, p$ ) by means of:

$$c_{\nu,p+1-k} = \sum_{i=1}^k \alpha_{i,p+1-k}^{\nu-1} \omega_{i,p+1-k}.$$

(iv) Compute

$$d_{\nu\mu}^{(k+1)} = d_{\nu\mu}^{(k)} - \beta_{p+1-k}^{\mu-1} c_{\nu,p+1-k}, \quad (\nu = k+1, \dots, p; \mu = 1, \dots, p+1-\nu).$$

For  $n = 3$  we have to solve the system:

$$(3.2) \quad \sum_{k=1}^p \sum_{j=1}^{p+1-k} \sum_{i=1}^{p+2-j-k} \alpha_{ijk}^{\nu-1} \beta_{jk}^{\mu-1} \gamma_k^{\lambda-1} \omega_{ijk} = d_{\nu\mu\lambda},$$

$$\nu = 1, \dots, p; \quad \mu = 1, \dots, p+1-\nu; \quad \lambda = 1, \dots, p+2-\mu-\nu.$$

(Now the  $\gamma$ 's,  $\beta$ 's and  $\alpha$ 's satisfy:  $\gamma_j \neq \gamma_k$  if  $j \neq k$ ;  $\beta_{jk} \neq \beta_{ik}$  if  $j \neq i$ ;  $\alpha_{ijk} \neq \alpha_{sjk}$  if  $i \neq s$ .) The algorithm takes again  $p$  steps.

*Algorithm TRIDIM.* First we define  $d_{\nu\mu\lambda}^{(1)} = d_{\nu\mu\lambda}$ .

*Step N* ( $N = 1, \dots, p$ ):

(i) Obtain  $c_{Njk}$  ( $k = 1, \dots, p+1-N$ ;  $j = 1, \dots, p+2-N-k$ ) by solving the bidimensional Vandermonde system (using BIDIM):

$$\sum_{k=1}^{p+1-N} \sum_{j=1}^{p+2-N-k} \beta_{jk}^{\nu-1} \gamma_k^{\lambda-1} c_{Njk} = d_{N\mu\lambda}^{(N)},$$

$$\mu = 1, \dots, p+1-N; \quad \lambda = 1, \dots, p+2-N-\mu.$$

(If  $N = p$ ,  $c_{p11} = d_{p11}^{(p)}$ .)

(ii) For  $k = 1, \dots, p+1-N$ ;  $j = p+2-N-k$  obtain  $\omega_{ijk}$  ( $i = 1, \dots, N$ ) by solving the Vandermonde systems:

$$\sum_{i=1}^N \alpha_{ijk}^{\nu-1} \omega_{ijk} = c_{\nu jk}, \quad \nu = 1, \dots, N.$$

(If  $N = 1$ ,  $\omega_{1jk} = c_{1jk}$  ( $j+k = p+1$ ); if  $N = p$  the process ends.)

(iii) For  $k = 1, \dots, p+1-N$ ;  $j = p+2-N-k$  compute

$$c_{\nu jk} = \sum_{i=1}^N \alpha_{ijk}^{\nu-1} \omega_{ijk} \quad \nu = N+1, \dots, p.$$

(iv) Compute

$$d_{\nu\mu\lambda}^{(N+1)} = d_{\nu\mu\lambda}^{(N)} - \sum_{k+j=p+2-N} \beta_{jk}^{\mu-1} \gamma_k^{\lambda-1} c_{\nu jk},$$

$$\nu = N+1, \dots, p; \quad \mu = 1, \dots, p+1-\nu; \quad \lambda = 1, \dots, p+2-\mu-\nu.$$

**4. Numerical Results.** Algorithm TRIDIM was implemented as a FORTRAN IV program and tested on an IBM/360-40 computer. A listing of this program can be found in Pub. 69-07, Departamento de Computación, Facultad de Ciencias, Universidad Central de Venezuela.

The Vandermonde systems were those obtained in the process of generating numerical differentiation formulae as explained in the Introduction. An evenly spaced mesh of width  $h$  was used throughout, and the nodal points considered were those contained on the tetrahedron  $x \geq 0, y \geq 0, z \geq 0, x + y + z \leq p \cdot h$ , translated so its center of mass coincided with the point  $(x_0, y_0, z_0)$  at which  $L(f)$  is desired.

We give here results for two functions:  $f_1(x, y, z) = \sin(x + y + z)$  and  $f_2(x, y, z) = e^{-(x+y+z)}$ , several orders  $p$  and mesh sizes  $h$  and two differential operators:

$$L_1(f) = h(D_x + D_y + D_z)f;$$

$$L_2(f) = \sum_{k=2}^4 \sum_{j=1}^{k-1} h^k a_{kj} [D_x^j D_y^{k-j} + D_y^j D_z^{k-j} + D_z^j D_x^{k-j}] f$$

with  $a_{21} = 1, a_{31} = a_{32} = 0.5, a_{41} = a_{43} = 0.15, a_{42} = 0.25$ .

The Tables below show  $\varepsilon$ : the relative errors between the exact  $L_i(f_j)$  and the approximated values obtained with  $L_{ih}(f_j)$  (see (1.2)). The weights  $\omega_r$  for  $L_{ih}(f_j)$  have been computed with TRIDIM (see (1.3)), the whole process being carried on double precision ( $\sim 16.8$  decimal digits). As usual:  $m, n \equiv m \times 10^n$ .

		$\varepsilon$ for $L_1(f_1)$		
$h \backslash p$	3	6	10	
1/4	4, -2	1, -5	1, -9	
1/8	1, -2	7, -7	9, -12	
1/16	2, -3	2, -8	2, -14	

$(x_0 = y_0 = z_0 = 0.25)$

		$\varepsilon$ for $L_2(f_1)$		
$h \backslash p$	5	8	10	
1/4	2, -1	1, -4	4, -6	
1/8	4, -2	2, -6	2, -8	
1/16	1, -2	3, -8	6, -11	

$(x_0 = y_0 = z_0 = 0)$

		$\varepsilon$ for $L_1(f_2)$		
$h \backslash p$	3	6	10	
1/4	5, -2	5, -5	2, -8	
1/8	1, -2	1, -6	2, -11	
1/16	2, -3	3, -8	3, -14	

$(x_0 = y_0 = z_0 = 2.5)$

		$\varepsilon$ for $L_2(f_2)$		
$h \backslash p$	5	8	10	
1/4	7, -2	1, -4	4, -6	
1/8	7, -3	1, -6	1, -8	
1/16	8, -4	2, -8	3, -11	

$(x_0 = y_0 = z_0 = 2.5)$

Departamento de Computación  
Facultad de Ciencias  
Universidad Central de Venezuela.  
Caracas

1. Å. BJÖRCK & V. PEREYRA, "Solution of Vandermonde systems of equations," *Publ. 70-02, Dept. Comp., Fac. Ci. Univ. Cent. Venezuela.*
2. F. STENGER, "Kronecker product extensions of linear operators," *SIAM J. Numer. Anal.*, v. 5, 1968, pp. 422-435. MR 38 #4014.
3. M. STOYAKOVICH, "Inversion of the matrices encountered in relay switching circuit synthesis theory," *Ž. Vychisl. Mat. i Mat. Fiz.*, v. 6, 1966, pp. 158-161.