An algorithm for the calculation of the pseudo-inverse of a singular matrix

By David Mayne*

An algorithm for the calculation of the pseudo-inverse of a singular matrix is derived. The method is motivated by Wiener-Kalman filtering theory and uses successive "observations" to update the "estimation" of the pseudo-inverse. Illustrative and numerical examples are given so that the speed and accuracy of the method may be compared with ordinary inversion. Although the speed of the method may be made to approach that of ordinary inversion, to achieve reasonable accuracy double precision arithmetic must be employed with a consequent reduction in speed.

1. Introduction

The pseudo-inverse is often useful in minimization problems. Thus in some applications of control theory (or filtering theory) it is required to find the linear control (or filtering) law defined by:

$$u = \Delta x$$

which minimizes the quadratic cost function

$$V(u) = a + \tfrac{1}{2}u^T R u + u^T C x$$

where R is symmetric.

If R is invertible

$$\Delta = -R^{-1}C.$$

If R is singular, the optimum value of u is Δx, where now

$$\Delta = -R^{+}C$$

where R^+ denotes the pseudo-inverse of R.

Zadeh and Desoer (1963) define the pseudo-inverse as follows. If A is a $m \times n$ singular matrix of rank p, then the $n \times m$ matrix A^+ is the pseudo-inverse of A if:

(1) $A^+Ax = x$ for all $x \varepsilon \mathcal{N}(A)\perp = \mathcal{R}(A^T)$

(2) $A^+z = 0$ for all $z \varepsilon \mathcal{R}(A)\perp = \mathcal{N}(A^T)$

(3) $A^+(y+z) = A^+y + A^+z$ for all $y \varepsilon \mathcal{R}(A)$ and all $z \varepsilon \mathcal{R}(A)\perp$

where $\mathcal{R}(A), \mathcal{N}(A)$ denote, respectively, the range and null space of A, $\perp$ denotes orthogonal complement, and T denotes transpose.

In the derivation of the algorithm use will be made of filtering theory (Kalman, 1960) (or regression analysis) although a separate proof of the algorithm is given. Suppose at stage $r-1$, an estimate $\hat{\theta}_{r-1}$ of a set of m-parameters θ is known ($\hat{\theta}_{r-1}$ and θ are m-vectors) and the variance of the estimate is P_{r-1}. P_{r-1} is a $m \times m$ symmetric matrix defined by

$$P_{r-1} = E(\theta - \hat{\theta}_{r-1})(\theta - \hat{\theta}_{r-1})^T.$$

Then, the r^{th} scalar observation x_r, where

$$x_r = y_r^T\theta + v_r$$

(y_r is a known m-vector) can be used to provide an improved estimate $\hat{\theta}_r$ of variance P_r:

$$\hat{\theta}_r = \hat{\theta}_{r-1} + \frac{P_{r-1}y_r}{\sigma_{vv} + y_r^T P_{r-1} y_r}(x_r - y_r^T\hat{\theta}_{r-1})$$

$$P_r = P_{r-1} + \frac{P_{r-1}y_r y_r^T P_{r-1}}{\sigma_{vv} + y_r^T P_{r-1} y_r}$$

σ_{vv} is the variance of the measurement noise v_r which is assumed to have zero mean. This updating algorithm, assuming perfect measurements ($\sigma_{vv} = 0$), will be used in § 2.

2. Derivation of the algorithm

Let $x_1 \ldots x_p$ denote a set of p linearly independent vectors of dimension n chosen from $\mathcal{N}(A)\perp = \mathcal{R}(A^T)$. Since A has rank p the vectors $x_1 \ldots x_p$ constitute a basis for $\mathcal{N}(A)\perp$. The p m-dimensional vectors $y_1 \ldots y_p$ are calculated using equation (1)

$$y_r = Ax_r, \quad r = 1 \ldots p \tag{1}$$

Thus $y_r \varepsilon \mathcal{R}(A)$, $r = 1 \ldots p$. $y_1 \ldots y_p$ are linearly independent and constitute a basis for $\mathcal{R}(A)$. From the definition of the pseudo-inverse we have:

$$A^+Ax_r = x_r, \quad r = 1 \ldots p$$

i.e.

$$x_r = A^+y_r, \quad r = 1 \ldots p. \tag{2}$$

If we denote the j^{th} row of A^+ by the m-dimensional vector θ_j, then equation (2) may be written:

$$(x_r)_j = y_r^T\theta_j, \quad r = 1 \ldots p, \quad j = 1 \ldots n \tag{3}$$

where $(x_r)_j$ is the j^{th} component of x_r. $(x_r)_j$ may be regarded as an "observation" of the m unknown parameters θ_j, as in the estimation problem of §1; the noise v_r has zero variance. With this motivation the following algorithm for obtaining $\theta_j, j = 1 \ldots n$, is proposed:

$$\left.\begin{aligned} &\hat{\theta}_j(0) = 0, j = 1 \ldots n \\ &P_0 = I \end{aligned}\right\} \tag{4}$$

$$\hat{\theta}_j(r) = \hat{\theta}_j(r-1) + \left[\frac{P_{r-1}y_r}{y_r^T P_{r-1} y_r}\right][(x_r)_j - y_r^T\hat{\theta}_j(r-1)] \quad j = 1 \ldots n \tag{5}$$

*Dept. of Electrical Engineering, Imperial College, London, S.W.7.

$$P_r = P_{r-1} - \frac{P_{r-1}y_r \cdot y_r^T P_{r-1}}{y_r^T P_{r-1} y_r} \tag{6}$$

$$\hat{\theta}_j(p) = \theta_j, \quad j = 1 \ldots n \tag{7}$$

i.e. equations (5) and (6) are used iteratively, with the boundary conditions of equation (4), to obtain $\hat{\theta}_j(p)$, which will be shown to be equal to the j^{th} row of A^+.

It is necessary to establish some properties of P_r and $\hat{\theta}_j(r)$. From equation (6)

$$P_r y_r = 0, \quad r = 1 \ldots p. \tag{8}$$

Also:

$$P_r = \left[I - \frac{P_{r-1}y_r y_r^T}{y_r^T P_{r-1} y_r}\right] P_{r-1} = C_r P_{r-1}$$

$$\therefore P_r y_{r-1} = C_r P_{r-1} y_{r-1} = 0.$$

$$P_r y_{r-2} = C_r P_{r-1} y_{r-2} = C_r C_{r-1} P_{r-2} y_{r-2} = 0.$$

Thus: $P_r y_k = 0, \quad k \leqslant r, \quad r = 1 \ldots p$ (9)

where $y_k \varepsilon \mathscr{R}(A)$.

Consider now a vector z orthogonal to the space spanned by the linearly independent vectors $y_1 \ldots y_r$:

$$y_s^T z = 0, \quad s = 1 \ldots r. \tag{10}$$

Now:

$$P_0 = I \quad \therefore P_0 z = z$$

$$P_1 = P_0 - \frac{P_0 y_1 y_1^T P_0}{y_1^T P_0 y_1}, \quad \therefore P_1 z = z$$

(since $y_1^T P_0 z = y_1^T z = 0$)

$$P_2 = P_1 - \frac{P_1 y_2 y_2^T P_1}{y_2^T P_1 y_1}, \quad \therefore P_2 z = z \text{ etc.}$$

Thus: $P_r z = z$ (11)

for all z orthogonal to $y_1 \ldots y_r$.

y_{r+1} consists of two components, $\hat{y}_{r+1}$, the orthogonal projection of y_{r+1} into the space spanned by $y_1 \ldots y_r$, and $\tilde{y}_{r+1}$, that component of y_{r+1} orthogonal to $y_1 \ldots y_r$. From equations (9) and (11)

$$P_r y_{r+1} = P_r(\hat{y}_{r+1} + \tilde{y}_{r+1})$$
$$= P_r \tilde{y}_{r+1}$$
$$= \tilde{y}_{r+1}$$

(since $\hat{y}_{r+1} = \sum_{s=1}^{r} \eta_s y_s$).

Consider now the properties of $\hat{\theta}_j(r)$. From equation (5):

$$y_r^T \hat{\theta}_j(r) = (x_r)_j.$$

Also, from equations (5) and (9)

$$y_{r-1}^T \hat{\theta}_j(r) = y_{r-1}^T \hat{\theta}_j(r-1) = (x_{r-1})_j$$

(since $y_{r-1}^T P_{r-1} y_r = y_r^T P_{r-1} y_{r-1} = 0$).

Similarly $y_{r-2}^T \hat{\theta}_j(r) = y_{r-2}^T \hat{\theta}_j(r-1) = (x_{r-2})_j$ etc.

Thus $y_k^T \theta_j(r) = (x_k)_j \quad k \leqslant r, r = 1 \ldots p.$ (12)

If we let $\hat{\Theta}(r)$ denote the matrix whose j^{th} *row* is $\hat{\theta}_j(r)$, equation (12) becomes

$$\hat{\Theta}(r) y_k = x_k, \qquad k \leqslant r, r = 1 \ldots p. \tag{13}$$

In particular

$$\hat{\Theta}(p) y_k = x_k, \qquad k = 1 \ldots p. \tag{14}$$

Consider now a vector z orthogonal to the space spanned by $y_1 \ldots y_r$. From equation (5):

$$\hat{\theta}_j(1) = \frac{y_1 (x_1)_j}{y_1^T y_1}.$$

Therefore

$$z^T \hat{\theta}_j(1) = 0.$$

Also, from equation (5)

$$z^T \hat{\theta}_j(r) = z^T \hat{\theta}_j(r-1) + \left[\frac{z^T P_{r-1} y_r}{y_r^T P_{r-1} y_r}\right][(x_r)_j - y_r^T \hat{\theta}_j(r-1)].$$

But $z^T P_{r-1} y_r = (P_{r-1} z)^T y_r = z^T y_r = 0, r = 1 \ldots p.$

Therefore $z^T \hat{\theta}_j(r) = z^T \hat{\theta}_j(r-1).$

But $z^T \hat{\theta}_j(1) = 0.$

Therefore $z^T \hat{\theta}_j(r) = 0, \quad j = 1 \ldots n$

i.e. $\hat{\Theta}(r) z = 0$, for all z orthogonal to $y_1 \ldots y_r$.

In particular

$$\hat{\Theta}(p) z = 0, \quad \text{for all } z \varepsilon \mathscr{R}(A)\perp. \tag{15}$$

We can now use equations (14) and (15) to show that the matrix $\hat{\Theta}(p)$ constructed from $\hat{\theta}_j(p)$, $j = 1 \ldots n$, satisfies the definition of the pseudo-inverse. Let $x = \sum_{i=1}^{p} \xi_i x_i$; i.e. x is any vector in $\mathscr{N}(A)\perp$, depending on the choice of $\xi_1 \ldots \xi_p$, since $x_1 \ldots x_p$ constitute a basis for $\mathscr{N}(A)\perp$. Let $y = \sum_{i=1}^{p} \xi_i y_i = \sum_{i=1}^{p} \xi_i A x_i = Ax$. Then, from equation (14)

$$\hat{\Theta}(p) \sum_{i=1}^{p} \xi_i y_i = \sum_{i=1}^{p} \xi_i x_i = x \tag{16}$$

i.e. $\hat{\Theta}(p) Ax = x \qquad$ for all $x \varepsilon \mathscr{N}(A)\perp$.

Equation (16) is property (1) of the definition of the pseudo-inverse. Equation (15) is property (2). Property (3) is obviously satisfied. Therefore

$$\hat{\Theta}(p) = A^+ \tag{17}$$

i.e. the matrix whose rows are $\hat{\theta}_j(p), j = 1 \ldots n$, is the pseudo-inverse of A.

It only remains to show how a set of linearly independent vectors from $\mathscr{N}(A)\perp, x_1 \ldots x_p$, may be obtained. First we note that $\mathscr{R}(A^T) = \mathscr{N}(A)\perp$. Denote the *columns* of A^T, i.e. the *rows* of A, by the n-dimensional *column* vectors $\bar{x}_r, r = 1 \ldots m$. Obviously $\bar{x}_r \varepsilon \mathscr{N}(A)\perp$, $r = 1 \ldots m$, and we have to select p linearly independent vectors from the set $\bar{x}_1 \ldots \bar{x}_m$. Obtain the vectors $\bar{y}_1 \ldots \bar{y}_m$ where

$$\bar{y}_r = A\bar{x}_r, \quad r = 1 \ldots m$$

i.e. $\bar{y}_r$ is the r^{th} column of AA^T.

To obtain $\bar{y}_r$, $r = 1 \ldots m$, we have to evaluate all the inner products $\bar{x}_s^T\bar{x}_t$, $s = 1 \ldots m$, $t = 1 \ldots m$. The vectors x_r and y_r, required in equations (5) and (6), are selected from the set $\bar{x}_k$, $\bar{y}_k$, $k = 1 \ldots m$, as follows. Assume that $x_1 \ldots x_r$ and $y_1 \ldots y_r$ have already been selected and that $\hat{\theta}_j(r)$, $j = 1 \ldots n$, and P_r have been evaluated. We now select $\bar{y}_k$ from those vectors of the set $\bar{y}_1 \ldots \bar{y}_m$ which have not been previously used *or* discarded. The requirement for selection is that $\bar{y}_k$ is linearly independent of $y_1 \ldots y_r$. If it is *not*, then

$$P_r\bar{y}_k = 0 \tag{18}$$

since:
$$\bar{y}_k = \sum_{i=1}^{r} \xi_i y_i$$

and
$$P_r y_i = 0, \quad i = 1 \ldots r.$$

Therefore all $\bar{y}_k$ which satisfy equation (18) are discarded; to allow for numerical error the first $\bar{y}_k$ which satisfies the relation

$$|P_r\bar{y}_k| > \varepsilon \tag{19}$$

where $|\ .\ |$ denotes a suitable norm of a vector and ε is a suitably chosen small positive number, is called y_{r+1} and used to evaluate $\hat{\theta}_j(r+1)$, P_{r+1}. The selection procedure is then repeated to select $\bar{y}_{r+2}$.

3. Algorithm A

The algorithm for calculating the pseudo-inverse can now be summarized.

1. Denote the m rows of A by the n-dimensional vectors $\bar{x}_1 \ldots \bar{x}_m$.
2. Calculate the m m-dimensional vectors

   $\bar{y}_1 \ldots \bar{y}_m$ where

   $\bar{y}_r = A\bar{x}_r, \quad r = 1 \ldots m$

   i.e. $\bar{y}_r$ is the rth column of AA^T.
3. Set $\hat{\theta}_j(0) = 0, j = 1 \ldots n$, and $P_0 = I$.
4. At stage r select the first vector $\bar{y}_k$ (from those vectors of the set which have not been previously used or discarded) which satisfies equation (19). Call this vector y_{r+1}.
5. Calculate $\hat{\theta}_j(r+1)$, $j = 1 \ldots n$, and P_{r+1} using equations (5) and (6).
6. Repeat (4) and (5) until $\hat{\theta}_j(p)$, $j = 1 \ldots n$, is obtained. This occurs when all the vectors $\bar{y}_k$, $k = 1 \ldots m$, have either been used or discarded.
7. A^+ is the matrix whose jth row is $\hat{\theta}_j(p)$.

Approximately $\left[\frac{m^2n}{2} + \frac{3pm^2}{2} + 2pmn\right]$ multiplications are required for large m, n. If $m = n = p$ this becomes $4n^3$ multiplications.

3.1 Illustrative examples

Two simple examples are described to illustrate the method.

(i) $A = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \end{bmatrix}$

Therefore $x_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$, $x_2 = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$

$y_1 = \begin{bmatrix} 3 \\ 6 \end{bmatrix}$ $y_2 = \begin{bmatrix} 6 \\ 12 \end{bmatrix}$

$P_0 = I$; $P_0y_1 = \begin{bmatrix} 3 \\ 6 \end{bmatrix}$ $\therefore$ y_1 can be used

$\therefore$ $P_1 = I - \frac{1}{45}\begin{bmatrix} 3 \\ 6 \end{bmatrix}[3\ 6] = \begin{bmatrix} 0{\cdot}8 & -0{\cdot}4 \\ -0{\cdot}4 & 0{\cdot}2 \end{bmatrix}$

$P_1y_2 = 0$ $\therefore$ y_2 is discarded

$\hat{\theta}_1(0) = \hat{\theta}_2(0) = \hat{\theta}_3(0) = 0.$

Since $(x_1)_1 = (x_2)_1 = (x_3)_1 = 1$, then

$$\hat{\theta}_1(1) = \hat{\theta}_2(1) = \hat{\theta}_3(1) = 0 + \frac{1}{45}\begin{bmatrix} 3 \\ 6 \end{bmatrix} = \frac{1}{15}\begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$\therefore$ $A^+ = \frac{1}{15}\begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix}$

(ii) $A = \begin{bmatrix} 2 & 0 & 2 \\ 1 & 1 & 2 \end{bmatrix}$

$x_1 = \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix}$ $x_2 = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$

$y_1 = \begin{bmatrix} 8 \\ 6 \end{bmatrix}$ $y_2 = \begin{bmatrix} 6 \\ 6 \end{bmatrix}$

$P_0 = I$; $P_0y_1 = \begin{bmatrix} 8 \\ 6 \end{bmatrix}$ i.e. y_1 can be used

$P_1 = \frac{1}{100}\begin{bmatrix} 36 & -48 \\ -48 & 64 \end{bmatrix}$

$P_1y_2 = \frac{24}{100}\begin{bmatrix} -3 \\ 4 \end{bmatrix}$ i.e. y_2 can be used

$P_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

$\hat{\theta}_1(0) = \hat{\theta}_2(0) = \hat{\theta}_3(0) = 0$

$\hat{\theta}_1(1) = \frac{4}{100}\begin{bmatrix} 4 \\ 3 \end{bmatrix}$, $\hat{\theta}_2(1) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $\hat{\theta}_3(1) = \frac{4}{100}\begin{bmatrix} 4 \\ 3 \end{bmatrix}$

$\hat{\theta}_1(2) = \frac{1}{6}\begin{bmatrix} 3 \\ -2 \end{bmatrix}$, $\hat{\theta}_2(2) = \frac{1}{6}\begin{bmatrix} -3 \\ 4 \end{bmatrix}$, $\hat{\theta}_3(2) = \frac{1}{6}\begin{bmatrix} 0 \\ 2 \end{bmatrix}$

$$\therefore A^+ = \frac{1}{6}\begin{bmatrix} 3 & -2 \\ -3 & 4 \\ 0 & 2 \end{bmatrix}$$

4. Simplified algorithm

Let $\hat{y}_r$ denote the orthogonal projection of y_r on to the space spanned by $y_1 \ldots y_{r-1}$, and $\tilde{y}_r$ the component of y_r orthogonal to $y_1 \ldots y_{r-1}$. Then

$$y_r = \hat{y}_r + \tilde{y}_r.$$

From §2

$$\tilde{y}_r = P_{r-1} y_r.$$

Thus $$y_r^T P_{r-1} y_r = (\hat{y}_r + \tilde{y}_r)^T \tilde{y}_r = \tilde{y}_r^T \tilde{y}_r.$$

Define $\hat{x}_r$, $\tilde{x}_r$ as follows:

$$\hat{x}_r = \hat{\Theta}(r-1) y_r$$
$$\tilde{x}_r = x_r - \hat{x}_r.$$

Note that $\hat{x}_r$ and $\tilde{x}_r$ are not necessarily orthogonal. Equation (5) may now be written

$$\hat{\Theta}(r) = \hat{\Theta}(r-1) + \frac{\tilde{x}_r \tilde{y}_r^T}{\tilde{y}_r^T \tilde{y}_r}. \qquad (5A)$$

4.1 Algorithm B

If $\tilde{y}_r$ can be calculated without using P_{r-1}, the algorithm can be simplified.

$\hat{y}_r$ may be expressed as follows:

$$\hat{y}_r = \sum_{s=1}^{r-1} \eta_s y_s.$$

From §2

$$\hat{\Theta}(r-1) y_s = x_s, \quad s \leqslant r-1$$
$$\hat{\Theta}(r-1) \tilde{y}_r = 0$$

$$\therefore \quad \hat{x}_r = \hat{\Theta}(r-1) y_r = \hat{\Theta}(r-1)[\hat{y}_r + \tilde{y}_r] = \sum_{s=1}^{r-1} \eta_s x_s$$

and $$A\hat{x}_r = \sum_{s=1}^{r-1} \eta_s A x_s = \sum_{s=1}^{r-1} \eta_s y_s = \hat{y}_r$$

$$\therefore \quad \tilde{y}_r = y_r - \hat{y}_r$$
$$= Ax_r - A\hat{x}_r$$
$$= A\tilde{x}_r.$$

Therefore the simplified algorithm becomes

$$\hat{\Theta}(0) = 0$$

$$\hat{\Theta}(r) = \hat{\Theta}(r-1) + \frac{\tilde{x}_r (A\tilde{x}_r)^T}{(A\tilde{x}_r)^T (A\tilde{x}_r)}$$

where $$\tilde{x}_r = x_r - \hat{\Theta}(r-1) y_r$$
$$\hat{\Theta}(p) = A^+.$$

The criterion for selecting $\bar{y}_k$ to serve as y_r is

$$(A\tilde{x}_r)^T (A\tilde{x}_r) > \varepsilon^2.$$

4.2 Algorithm C

$\tilde{y}_r$ may be obtained using the Gram–Schmidt orthogonalization procedure. Assume that $\tilde{x}_s$, $\tilde{y}_s$, $s = 1 \ldots r-1$ have already been obtained. Then

$$\tilde{y}_r = y_r - \sum_{s=1}^{r-1} C_s \tilde{y}_s$$

where $$C_s = \frac{y_r^T \tilde{y}_s}{\tilde{y}_s^T \tilde{y}_s}.$$

It can be shown that

$$\tilde{x}_r = x_r - \sum_{s=1}^{r-1} C_s \tilde{x}_s.$$

Thus algorithm C is:

$$\hat{\Theta}(0) = 0$$

$$\hat{\Theta}(r) = \hat{\Theta}(r-1) + \frac{\tilde{x}_r \tilde{y}_r^T}{\tilde{y}_r^T \tilde{y}_r}$$

where $$\tilde{y}_r = y_r - \sum_{s=1}^{r-1} C_s \tilde{y}_s$$

$$\tilde{x}_r = x_r - \sum_{s=1}^{r-1} C_s \tilde{x}_s$$

and $$C_s = \frac{y_r^T \tilde{y}_s}{\tilde{y}_s^T \tilde{y}_s}.$$

The criterion for selecting $\bar{y}_k$ to serve as y_r is that $\tilde{y}_r^T \tilde{y}_r > \varepsilon^2$.

Because large errors may result in the calculation of A^+ if $\tilde{y}_r^T \tilde{y}_r$ is small even though greater than ε^2 (see §5) it is necessary to modify the algorithm slightly if reasonable accuracy is to be achieved under these conditions. Thus if $\varepsilon^2 < \tilde{y}_r^T \tilde{y}_r < \eta^2$, where η is a suitable norm, then a new x_r and y_r are chosen as follows:

$$x_r = \tilde{x}_r / \eta$$
$$y_r = Ax_r.$$

The orthogonalization procedure is then repeated to give a new $\tilde{x}_r$ and $\tilde{y}_r$. The algorithm using this feature will be referred to as the modified C algorithm.

5. The selection criterion

The rank p of the pseudo-inverse is determined by the number of independent vectors $y_1 \ldots y_p$ in the set $\bar{y}_1 \ldots \bar{y}_m$ which comprise the columns of A^T. A numerical criterion is used to judge independence. The selection or non-selection of a vector from the set $\bar{y}_1 \ldots \bar{y}_m$ can cause large variations in A^+. This is best illustrated by means of a simple example (using algorithm C).

$$A = \begin{bmatrix} 1 & 0 \\ 1 & \eta_1 \end{bmatrix}, \quad A^T = \begin{bmatrix} 1 & 1 \\ 0 & \eta_1 \end{bmatrix}, \quad \eta_1 \ll 1$$

$$AA^T = \begin{bmatrix} 1 & 1 \\ 1 & 1 + \eta_1^2 \end{bmatrix}$$

$$\therefore \quad y_1 = \tilde{y}_1 = \begin{bmatrix}1\\1\end{bmatrix}, \quad x_1 = \tilde{x}_1 = \begin{bmatrix}1\\0\end{bmatrix}$$

$$\therefore \quad \hat{\Theta}(1) = \frac{\begin{bmatrix}1\\0\end{bmatrix}[1\ \ 1]}{2} = \begin{bmatrix}\frac{1}{2} & \frac{1}{2}\\ 0 & 0\end{bmatrix}$$

$$\tilde{y}_2 = \begin{bmatrix}1\\1+\eta_1^2\end{bmatrix} - C_1\begin{bmatrix}1\\1\end{bmatrix}, \quad \tilde{x}_2 = \begin{bmatrix}1\\ \eta_1\end{bmatrix} - C_1\begin{bmatrix}1\\0\end{bmatrix}$$

where

$$C_1 = \frac{y_2\tilde{y}_1^T}{\tilde{y}_1^T\tilde{y}_1} = 1 + \eta_1^2/2$$

$$\therefore \ \tilde{y}_2 = \begin{bmatrix}-\eta_1^2/2\\ \eta_1^2/2\end{bmatrix} \qquad \tilde{x}_2 = \begin{bmatrix}-\eta_1^2/2\\ \eta_1\end{bmatrix}$$

The criterion for selecting y_2 is that

$$\tilde{y}_2^T\tilde{y}_2 = \eta_1^4/2 > \varepsilon^2.$$

If y_2 is discarded:

$$A^+ = \hat{\Theta}(1) = \begin{bmatrix}\frac{1}{2} & \frac{1}{2}\\ 0 & 0\end{bmatrix} \text{ and } p = 1.$$

If y_2 is selected:

$$A^+ = \hat{\Theta}(2) = \begin{bmatrix}\frac{1}{2} & \frac{1}{2}\\ 0 & 0\end{bmatrix} + \frac{\begin{bmatrix}-\eta_1^2/2\\ \eta_1\end{bmatrix}[-\eta_1^2/2\ \ \eta_1^2/2]}{\eta^4/2}$$

$$= \begin{bmatrix}\frac{1}{2} & \frac{1}{2}\\ 0 & 0\end{bmatrix} + \begin{bmatrix}\frac{1}{2} & -\frac{1}{2}\\ -\frac{1}{\eta_1} & \frac{1}{\eta_1}\end{bmatrix}$$

$$= \begin{bmatrix}1 & 0\\ -\frac{1}{\eta_1} & \frac{1}{\eta_1}\end{bmatrix} \text{ and } p = 2.$$

It can be seen that A^+ changes dramatically since $\frac{1}{\eta_1}$ is large. Since $\tilde{y}_2$ is obtained as a difference between large quantities, the percentage error in the elements of $\tilde{y}_2$ can be considerable if $\tilde{y}_2$ is small. The term

$$\frac{\tilde{x}^2\tilde{y}_2^T}{\tilde{y}_2^T\tilde{y}_2}$$

will then have large elements ($1/\eta_1$ in the example) with possibly large errors. Any method of finding the pseudo-inverse has to be able to reject non-independent vectors; the selection of a vector which has only a small orthogonal component will always lead to large terms in the pseudo-inverse. However, it is possible that alternative methods may be found which have better accuracy than the methods proposed in this paper.

6. Numerical examples

6.1 Inversion of a 6 × 6 non-singular matrix

To compare the algorithm with ordinary inversion the following matrix was selected:

$$A = \begin{bmatrix}1 & 2 & -1 & 2 & 3 & 7\\ 3 & 4 & 1 & -8 & 1 & 2\\ 9 & -2 & 1 & 4 & 6 & 8\\ 5 & 8 & -2 & 7 & 4 & -3\\ 8 & 1 & 6 & -3 & 4 & 3\\ 2 & 5 & 7 & 5 & 2 & -1\end{bmatrix}$$

To judge accuracy the matrix was inverted twice, and the average absolute error of each element of the resultant matrix calculated (note, $(A^+)^+ = A$).

(*a*) *Ordinary Inversion*
Time for single inversion: 4 units
Average absolute error after two inversions: $3{\cdot}5 \times 10^{-7}$.

(*b*) *Algorithm A* (*double precision*)
Time for single inversion: 26 units
Average absolute error after two inversions: 36×10^{-7}

(*c*) *Algorithm B* (*ordinary precision*)
Time for single inversion: 5 units

(*d*) *Algorithm B* (*double precision*)
Time for single inversion: 13 units
Average absolute error after two inversions: 121×10^{-7}.

(*e*) *Algorithm C* (*double precision*)
Time for single inversion: 14 units
Average absolute error after two inversions: 1×10^{-7}.

It can be seen that the accuracy of algorithm C is better than that of the ordinary inverse though the computing time is larger.

6.2 Inversion of a 6 × 6 singular matrix

(*a*) The pseudo-inverse of the matrix A was obtained using algorithm C where now:

$$A = \begin{bmatrix}1 & 2 & -1 & 2 & 3 & 7\\ 3 & 4 & 1 & -8 & 1 & 2\\ 9 & -2 & 1 & 4 & 6 & 8\\ 5 & 8 & -2 & 7 & 4 & -3\\ 8 & 1 & 6 & -3 & 4 & 3\\ 8 & 1 & 6 & -3 & 4 & 3\end{bmatrix}$$

Rows 5 and 6 are identical. The elements of the pseudo-inverse had values in the range $0{\cdot}3 \times 10^{-2}$ to $0{\cdot}15$. As before, to test the accuracy the pseudo-inverse of the pseudo-inverse was obtained and compared with A:

Time for single inversion: 12 units
Average absolute error after two inversions: $\frac{2}{3} \times 10^{-7}$.

(*b*) The calculation was repeated with the last element of row 6 (i.e. the number 3) replaced by 3·000001.

With the test number $\varepsilon = 10^{-7}$, the matrix was treated as singular (rank 5) and the pseudo-inverse almost identical to the pseudo-inverse of the singular matrix obtained:

Average absolute error after two inversions: 1×10^{-7}.

6.3 Inversion of a 6×6 near-singular matrix

To test the accuracy of the pseudo-inverse on ill-conditioned matrices the matrix A identical to that of §6.2 with the last element of row 6 (i.e. the number 3) replaced by 3·001 was used.

(*a*) Ordinary inversion:
Average absolute error after two inversions: $2{\cdot}3 \times 10^{-4}$.

(*b*) Pseudo-inverse (algorithm C, double precision):
Average absolute error after two inversions: 0·15.

It should be noted that the inverse of A had elements of order 10^4 in columns 5 and 6 and small percentage errors in these columns in the pseudo-inverse (errors of average absolute value 0·03) caused large percentage errors in the pseudo-inverse of the pseudo-inverse. The accuracy is clearly not sufficient for near singular matrices. The performance of the modified C algorithm (§4.2) was much better.

(*c*) Pseudo-inverse (modified C algorithm, double precision):
Time for single inversion: 16 units
Average absolute error after two inversions: $2{\cdot}5 \times 10^{-4}$.

The modified algorithm has an accuracy very close to that of ordinary inversion for the near-singular matrix. The value of η employed (see §4.2) was 0·03.

The author is very indebted to his colleague Mr. P. M. Newbold both for helpful discussions and valuable assistance in the numerical evaluation. The author also found the referee's comments most helpful; the development of algorithms B and C was motivated by these comments.

References

PENROSE, R. (1955). "A Generalised Inverse for Matrices", *Proc. Cambridge Phil. Soc.* Vol. 51, Part 3, pp. 406–413.
ZADEH, L., and DESOER, C. A. (1963). McGraw-Hill, New York, pp. 577–582.
KALMAN, R. E. (1960). "A New Approach to Linear Filtering and Prediction Problems", *Trans. A.S.M.E.* Vol. 82, p. 35.

Book Review

Error in Digital Computation, Volume II, edited by L. B. Rall, 1965; 228 pages. (London and New York: *John Wiley and Sons Ltd.*, 51s.)

Volume I, which was the proceedings of an advanced seminar conducted by the *Mathematics Research Centre* of the *United States Army* at the *University of Wisconsin* during October 1964, was reviewed here in January 1966. Volume II is not, as might be imagined, a continuation of the above proceedings; instead it is the proceedings of another seminar at the same place and on the same subjects, held during April 1965. The titles and authors of the eleven papers in the present volume follow:

(1) *Experimental Investigation of Unnormalized Arithmetic*, by R. L. Ashenhurst, 36 pages.
(2) *Error Bounds for Computations with Continued Fractions*, by Peter Henrici, 16 pages.
(3) *Error Bounds for Asymptotic Expansions of Special Functions in the Complex Plane*, by F. W. J. Olver, 22 pages.
(4) *Error Analysis of Transformations based on the Use of Matrices of the Form* $I-2ww^H$, by J. H. Wilkinson, 26 pages.
(5) *Automatic Local Coordinate Transformations to Reduce the Growth of Error Bounds in Interval Computation of Solutions of Ordinary Differential Equations*, by R. E. Moore, 38 pages.
(6) *Differential Inequalities and Error Bounds*, by Johann Schröder, 40 pages.
(7) *Discrete Representations of Partial Differential Operators*, by David M. Young and John H. Dauwalder, 38 pages.
(8) *Upper and Lower Bounds for Solutions of Integral Equations*, by R. W. Brown, 12 pages.
(9) *Convergence and Error Bounds for Approximate Solutions of Integral and Operator Equations*, by P. M. Anselone, 22 pages.
(10) *Applications of Functional Analysis to Error Estimation* by L. Collatz, 18 pages.
(11) *Error in the Solution of Linear Programming Problems*, by Philip Wolfe, 14 pages.

Papers 1 and 5 are developments of the work reported by the same authors in Volume I, and paper 4, by J. H. Wilkinson, is a further contribution to the work reported there by E. L. Albasiny. The remaining papers are on subjects not covered in Volume I.

The titles are self-explanatory. Some work is published here for the first time, *e.g.* in the first three papers. Papers 6, 8 and 9 describe recent work using the techniques of functional analysis, some of whose applications to error estimation are surveyed in paper 10. Paper 7 gives extensive tables of coefficients for 9-point representations of second-order partial differential operators in two variables. Paper 11 gives an assessment of the importance of error in linear programming, describes the Simplex method and measures used to combat error, but reports the practical non-existence of a theory of error for linear programming.

The book is a valuable collection of recent work on error in digital computation, and the papers in it are well supplied with references. It will be useful to readers wishing to acquaint themselves with the present state of the subject, or seeking new problems to investigate.

T. FULLER