

Pieter Omtzigt

Automatic identification of simultaneous equations models

2002/1



UNIVERSITÀ DELL'INSUBRIA
FACOLTÀ DI ECONOMIA

<http://eco.uninsubria.it>

In questi quaderni vengono pubblicati i lavori dei docenti della Facoltà di Economia dell'Università dell'Insubria. La pubblicazione di contributi di altri studiosi, che abbiano un rapporto didattico o scientifico stabile con la Facoltà, può essere proposta da un professore della Facoltà, dopo che il contributo sia stato discusso pubblicamente. Il nome del proponente è riportato in nota all'articolo. I punti di vista espressi nei quaderni della Facoltà di Economia riflettono unicamente le opinioni degli autori, e non rispecchiano necessariamente quelli della Facoltà di Economia dell'Università dell'Insubria.

These Working papers collect the work of the Faculty of Economics of the University of Insubria. The publication of work by other Authors can be proposed by a member of the Faculty, provided that the paper has been presented in public. The name of the proposer is reported in a footnote. The views expressed in the Working papers reflect the opinions of the Authors only, and not necessarily the ones of the Economics Faculty of the University of Insubria.

© Copyright Pieter Omtzigt
Printed in Italy in January 2002
Università degli Studi dell'Insubria
Via Ravasi 2, 21100 Varese, Italy

All rights reserved. No part of this paper may be reproduced in any form without permission of the Author.

Automatic identification of simultaneous equations models ^{*}

Pieter Omtzigt [†]

11 January 2001

Abstract

This paper provides an operational procedure for putting identifying restrictions on a simultaneous equations models. The algorithm works on the restrictions, not on the parameters, such that the identifying restrictions can be imposed before estimation.

Keywords: Simultaneous equations, identification, restriction, cointegration

1 Introduction

Consider the simultaneous equations model:

$$\begin{aligned}\beta' z_t &= A'y_t + B'x_t = u_t, t = 1, \dots, T \\ u_t &\sim iidN(0, \Omega)\end{aligned}$$

where y_t is a vector of length r with endogenous variables, x_t a vector of length q with predetermined variables and $\beta' = [A', B']'$ a $p \times r$ matrix of coefficients ($p = r + q$). Assume that A and therefore β is of full rank and that x_t and u_t are independent.

Likelihood inference on (β, Ω) is possible, but it is readily verified that a parameter point (β_1, Ω_1) is not uniquely identified (which means that there is at least one other parameter point (β_2, Ω_2) , with whom it shares the same probability measure). For any non-singular matrix C , $(\beta_1 C, C\Omega_1 C')$ has an identical probability measure. To uniquely identify a space we need to put restrictions on the parameter space. In this article we shall consider only within-equation restrictions on β of the form: (and thus not put any restrictions on Ω)

$$\beta = [H_1\varphi_1, \dots, H_r\varphi_r] \tag{1}$$

^{*} Antonietta Mira and Paolo Paruolo provided helpful comments on an earlier draft of this paper.

[†] Università dell'Insubria, Facoltà di Economia, Via Ravasi 2, 21100, Varese, Italy.

Email: pomtzig@eco.uninsubria.it

where H_i are $p \times s_i$ matrices of full column rank. Defining $R_i = (H_i)_\perp$ an equivalent expression of these restrictions is given by:

$$R_i' \beta_i = 0 \quad \text{for } i = 1, \dots, r \quad (2)$$

The first standard text book theorem (eg. Hsiao, 1983, corollary 3.3.2) states a necessary and sufficient condition for identification:

Theorem 1 *The parameter value (β, Ω) is uniquely identified if and only if for any $i = 1, \dots, r$*

$$\text{rank}(R_i' \beta_i) = r - 1$$

There are two problems in practise with this definition: it typically depends on the a-priory unknown parameters and it does not give an indication as how to identify a model if (2) fails. The first problem was tackled by Johansen (1995), who proved the following theorem:

Theorem 2 *If the only restrictions imposed on the parameters are (2) necessary and sufficient conditions for (β, Ω) to be generically identified are:*

$$\text{rank} (R_j' [H_{k_1}, \dots, H_{k_n}]) \geq n \quad (3)$$

for $n = 1, \dots, r - 1$

for all j

and for every set $\{k_1, \dots, k_n\}$ not containing j

This theorem gives conditions which only depend on the restrictions, not on the parameters. However if one of the rank conditions (3) fails, serious problems arise not just in the interpretation of potential estimates, but in the estimation and testing process itself: to my knowledge no analytical method exists to determine the number of restrictions imposed by (2) on the model.

In this paper we provide a simple algorithm to determine identifying restrictions, when (3) fails. Applications of this algorithm are:

1. A device for counting the number of restrictions in a particular model (if the restrictions are not identifying).
2. An instrument to be used for certain estimation algorithms, which require identification¹. For instance the algorithm of Johansen and Juselius (1994) requires identification. Without identification, it seems to work 95% of the time: for automatic model selection, this is however not sufficient.

¹If restrictions are placed on one vector only, that is $\beta_1 = H_1 \varphi_1$ then switching between β_1 and all the other vectors, which are restricted to lie in $\beta_{1\perp}$ and then normalizing them (most programs normalize estimates of eigenvectors by letting them sum to one) also ensures identification. This then ensures convergence in this particular case.

3. Only in identified models can (asymptotic) standard errors be given for all estimated parameters. Thus even if the algorithm is not used in 1. and 2. we can use the algorithm to find standard errors of the parameter. Now multiple identification schemes can work to our advantage as we can scan them all (usually there are only a few) and find those parameters, which we can restrict.
4. Davidson (1998) already provides an algorithm to find all possible restricted cointegration vectors (using Wald testing). He only considers one restricted vector at the time, but not a combination of them. Using the results in this paper and likelihood ratio tests, Omtzigt (2001) not only tests one restricted vector at the time, but also all possible combinations of them. The switching algorithm never breaks down here and the automatic selection procedure results in one preferred restricted model only (with possibly equivalent formulations).

For further discussions on the (restricted) simultaneous equations models, we refer to Koopmans et al. (1950), Fisher (1966), Hsiao (1983) and Sargan (1988) and references therein. For potential applications to the I(1) model we refer to Johansen (1995) and for the I(2) model to Johansen (2000).

The two main theorems and the algorithm that logically follow from them are given in the next paragraph. Then follows a number of comments on the proposed procedure itself. Paragraph four gives a detailed example of how the algorithm works in practice.

The appendices contain the proofs of all the theorems and a Matlab program, which implements the proposed procedure.

2 Results

In this paper we shall refer to (3) as rank conditions of order n . They can be given the following logical ordering:

$$\text{rank} (R'_j H_{k_1}) \geq 1, j \neq k_1 \quad (4)$$

$$\text{rank} (R'_j [H_{k_1}, H_{k_2}]) \geq 2, j \neq k_1 \neq k_2 \quad (5)$$

⋮

$$\text{rank} (R'_j [H_{k_1}, \dots, H_{k_{r-1}}]) \geq r - 1, j \neq k_1 \neq \dots \neq k_{r-1} \quad (6)$$

In case the rank conditions fail, many different ones may fail at the same time. We can find the first instance in the scheme above where a rank condition fails. Without loss of generality let this be a rank condition of order $m \geq 1$

$$\text{rank} (R'_j [H_{k_1}, \dots, H_{k_m}]) = m - 1, j \neq k_1 \neq \dots \neq k_m \quad (7)$$

The rank deficiency in (7) is necessarily exactly one as all the lower rank conditions hold:

$$\text{rank} (R'_j [H_{k_1}, \dots, H_{k_{m-1}}]) \geq m - 1, j \neq k_1 \neq \dots \neq k_{m-1} \quad (8)$$

$$\begin{aligned} & \vdots \\ \text{rank} (R_j' H_{k_1}) & \geq 1, j \neq k_1 \end{aligned} \quad (9)$$

The following theorem shows that we can always 'repair' this rank conditions by deleting one column from matrix H_j and adjusting R_j accordingly. Not any column can be deleted, but at least one of the columns repairs the rank condition.

Let the columns of H_j be h_{j1}, \dots, h_{js_j} and let $H_{j,-i} = [h_{11}, \dots, h_{1i-1}, h_{1i+1}, \dots, h_{1s}]$, that is H_j without column H_{ji} . Furthermore let $k_{ji} = [R_j, H_{j,-i}]_{\perp}$

Theorem 3 *For at least one of the columns h_{ji} of H_j ,*

$$\text{rank} ([R_j, k_{ji}]' [H_{k_1}, \dots, H_{k_m}]) = m \quad (10)$$

Without loss of generality, we shall assume that a condition involving R_1 is the first one to offend the rank condition and that h_{1d} is the added column in Theorem (3).

Let $H_1^* = [h_{11}, \dots, h_{1d-1}, h_{1d+1}, \dots, h_{1s}]$ be H_1 without h_{1d} . The next theorem shows that we can rotate the columns of any matrix β which is restricted as in (1) to find a matrix β^* which obeys all the previous restrictions implied by (1) and the new restriction, caused by shifting h'_{1d} from H_1 to R_1 .

Theorem 4 *For almost all $\beta = [H_1\varphi_1, \dots, H_r\varphi_r]$ there exists $\beta^* = [H_1^*\varphi_1^*, H_2\varphi_2, \dots, H_r\varphi_r]$ such that $sp(\beta) = sp(\beta^*)$*

The result has been split into two parts on purpose: theorem 3 only involves the restrictions, whereas theorem 4 shows that whatever the parameter value (or more precisely for all parameter values minus a subset of Lebesgue measure zero), the additional restriction can be satisfied. This means that we are only putting an extra identifying constraint on the model, which is exactly what we are looking for.

The idea of the proof is that if the rank condition of order m fails (and all the lower ones hold), then we can find exactly one linear combination of $(\beta_{k_1}, \dots, \beta_{k_m})$, say γ which lies in the space of β_j . Let $\beta_j = H_j\varphi_j$ and $\gamma = H_j\psi$. To distinguish β_j from γ we put one additional restriction on the β_j .

Together these last two theorems give rise to an operational algorithm to identify the space, given by any set of restrictions. Each time the rank condition is not satisfied by (H_1, \dots, H_r) we are able to take away a column of one of the H 's without imposing further restrictions. We repeat the operation until we have identifying restrictions (the algorithm is guaranteed to end as the number of columns of the matrices H is finite).

Formally we propose the following algorithm:

Algorithm 1

1. Check the rank conditions (4)-(6), for identification, starting with the lowest one, (4).
2. If all rank conditions are satisfied, go to 4.

3. When the first rank condition is broken, as in (7), find a column h_{ij} such that (10) is satisfied. Cancel this column from H_i and then go to 1.
4. The space is generically identified

3 An example

Consider the following matrix β with 5 rows and 3 columns, on which we impose within-equation restrictions (1) by means of the following matrices H_i : (Note that of each of the three matrices H_f the columns are orthogonal, such that $h_{fi} = k_{fi}$.)

$$\begin{aligned}
 H_1 &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, H_2 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, H_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \\
 H_1 &= [h_{11}, h_{12}, h_{13}], H_2 = [h_{21}, h_{22}, h_{23}], H_3 = [h_{31}, h_{32}, h_{33}]
 \end{aligned} \tag{11}$$

As bases of orthogonal complements to these matrices we choose:

$$\begin{aligned}
 R_1 &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ -1 & 0 \end{bmatrix}, R_2 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, R_3 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ -1 & 0 \end{bmatrix} \\
 R_1 &= [r_{11}, r_{12}], R_2 = [r_{21}, r_{22}], R_3 = [r_{31}, r_{32}]
 \end{aligned}$$

The algorithm now runs as follows:

3.1 First round

Check the first order rank conditions

$$\begin{aligned}
 \text{rank}(R_1' H_2) &= \text{rank} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} = 1 \\
 \text{rank}(R_1' H_3) &= \text{rank} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} = 1 \\
 \text{rank}(R_2' H_1) &= \text{rank} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} = 1 \\
 \text{rank}(R_2' H_3) &= \text{rank} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} = 1 \\
 \text{rank}(R_3' H_1) &= \text{rank} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} = 1 \\
 \text{rank}(R_3' H_2) &= \text{rank} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} = 1
 \end{aligned}$$

Check the second order rank conditions

As all first order rank conditions are satisfied, we check the second order rank conditions:

$$\text{rank}(R_1' [H_2, H_3]) = \text{rank} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} = 1$$

This rank condition fails, which means that we apply step 3 of the algorithm.

Find a column of H_1 which satisfies (10)

We add one of the columns of H_1 to R_1 and see whether this particular rank condition is repaired. Try $H_1^* = [h_{12}, h_{13}]$ and $R_1^* = [r_{11}, r_{12}, h_{11}]$. The rank condition becomes:

$$\text{rank}(R_1^{*'} [H_2, H_3]) = \text{rank} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \end{pmatrix} = 2$$

The rank condition is now satisfied and we take $H_1 = H_1^*$ and $R_1^* = R_1$ (leaving the other matrices as they were before) and start the algorithm at point 1:

3.2 Second round

Check the first order rank conditions

$$\text{rank}(R_1' H_2) = \text{rank} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = 1$$

$$\text{rank}(R_1' H_3) = \text{rank} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 0 \end{pmatrix} = 2$$

$$\text{rank}(R_2' H_1) = \text{rank} \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0$$

This rank condition fails.

Find a column of H_2 which satisfies (10)

When we move the first column of H_2 to R_2 we obtain the following candidates $H_2^* = [h_{22}, h_{23}]$ and $R_1^* = [r_{21}, r_{22}, h_{21}]$. The rank condition then reads:

$$\text{rank}(R_2^{*'} H_1) = \text{rank} \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} = 0$$

It is still not satisfied, so we try shifting the second column of H_2 : $H_2^* = [h_{21}, h_{23}]$ and $R_2^* = [r_{21}, r_{22}, h_{22}]$. This results in the following rank condition:

$$\text{rank}(R_2^{*'} H_1) = \text{rank} \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} = 1$$

The rank condition now holds and we take $H_2 = H_2^*$ and $R_2 = R_2^*$ to go back to step 1 of the algorithm:

3.3 Third round

Check the first order rank conditions

It is easily verified that of all the first order rank conditions, only the following one is not satisfied:

$$\text{rank}(R_3' H_2) = \text{rank} \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0$$

Find a column of H_3 which satisfies (10)

Shifting the first column of H_3 to R_3 would clearly not work, as that would imply $sp(H_2) = sp(H_3)$. (In this case even $H_2 = H_3$). We therefore shift the second column of H_3 : $H_3^* = [h_{31}, h_{33}]$ and $R_3^* = [r_{31}, r_{32}, h_{32}]$. The rank condition is now satisfied:

$$\text{rank}(R_3^{*'} H_2) = \text{rank} \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} = 1$$

For the next round take $H_3 = H_3^*$ and $R_3 = R_3^*$.

3.4 Fourth round

Check the first and second order rank conditions

All 6 first order and 3 second order rank conditions are satisfied, such that we conclude that the restrictions identify the model: The conditions of theorem 3 in Johansen (1995) now hold for this example

For completeness we shall also give the matrices S from theorem (4). If we have the matrices (11), then we can write the matrix β as:

$$\beta = \begin{bmatrix} \varphi_{11} & 0 & \varphi_{31} \\ 0 & \varphi_{21} & \varphi_{32} \\ \varphi_{12} & \varphi_{22} & 0 \\ \varphi_{13} & \varphi_{32} & \varphi_{33} \\ \varphi_{11} & 0 & \varphi_{31} \end{bmatrix}$$

The combination $\varphi_{32}\beta_2 - \varphi_{21}\beta_3 \equiv \gamma \in sp(H_1)$. When we post-multiply β by the full rank matrix

$$S_1 = \begin{bmatrix} 1 & 0 & 0 \\ \frac{\varphi_{32} \varphi_{11}}{\varphi_{22} \varphi_{31}} & 1 & 0 \\ -\frac{\varphi_{11}}{\varphi_{31}} & 0 & 1 \end{bmatrix}$$

we find

$$\beta^* = \begin{bmatrix} 0 & 0 & \varphi_{31} \\ 0 & \varphi_{21} & \varphi_{32} \\ \varphi_{12}^* & \varphi_{22} & 0 \\ \varphi_{13}^* & \varphi_{32} & \varphi_{33} \\ 0 & 0 & \varphi_{31} \end{bmatrix} \quad (12)$$

which satisfies the restrictions after the first round of the algorithm. Note that this transformation is not defined if $\varphi_{22} = 0$ or $\varphi_{31} = 0$.

Taking away the stars in the last expression, we can post-multiply again by

$$S_2 = \begin{bmatrix} 1 & \frac{\varphi_{22}}{\varphi_{12}} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

to obtain a matrix, satisfying the restrictions at the end of the second round. This step inserts a zero in place of φ_{22} in (12). In the last step, the matrix S_3 is given by:

$$S_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \frac{\varphi_{32}}{\varphi_{21}} \\ 0 & 0 & 1 \end{bmatrix}$$

Post-multiplication leads to the following general matrix:

$$\beta = \begin{bmatrix} 0 & 0 & \varphi_{31} \\ 0 & \varphi_{21} & 0 \\ \varphi_{12} & 0 & 0 \\ \varphi_{13} & \varphi_{32} & \varphi_{33} \\ 0 & 0 & \varphi_{31} \end{bmatrix} \quad (13)$$

which satisfies all the rank conditions and is therefore generically identified.

4 Comments and conclusions

Making a change in a broken rank condition can cause a previously satisfied rank condition to fail. In the example above, all rank conditions of first order are satisfied in the first round, but the change made causes first order rank conditions to fail subsequently. This demonstrates that in every round we have to start checking the lowest order rank conditions.

In the second round, we note that not any column can be eliminated from H , but we can still choose between deleting the second and the third column. This implies that the restrictions imposed by the algorithm are in general not unique. We thus find but only one of many ways to identify this space. It may be hard to attach an economic meaning to a particular identification in any one application. In some way this is the only weak point of the algorithm: in automatic search algorithms and other applications, the researcher may look for an different identification scheme to make economic sense of it. This however can easily be achieved by making available all equivalent identification schemes.

We have thus presented a way of identifying an under-identified parameter space in simultaneous equations models and hence rendered estimation possible. For the power of its application, we refer to Omtzigt (2001). The algorithms in MATLAB are available on request.

References

- [1] Davidson, J., (1998), Structural relations, cointegration and identification: some simple results and their application, *Journal of Econometrics* 87, 87-113
- [2] Fisher, F.M., (1966), *The identification problem in econometrics*, McGraw-Hill, New York
- [3] Hsiao, C. (1983), Identification, Chapter 4 in Z. Grilliches and M.D. Intriligator, *Handbook of Econometrics*, North-Holland, Amsterdam
- [4] Johansen, S., (1995), Identifying restrictions of linear equations: with applications to simultaneous equations and cointegration, *Journal of Econometrics* 69, 111-132
- [5] Johansen, S. (2000), Testing Hypotheses in the I(2) model, unpublished manuscript, European University Institute, Florence
- [6] Johansen, S. and K. Juselius (1994), Identification of the long-run and the short-run structure: an application to the IS-LM model, *Journal of Econometrics*, 63, 7-36
- [7] Koopmans, T.C., H. Rubin and R.B. Leipnik (1950), *Statistical Inference in Dynamic Economic Models*, Cowles Commission Monograph 10, J. Wiley, New York
- [8] Omtzigt (2001), Automatic model selection in vector autoregressive models, in P. Giudice (ed): *Metodi statistici per le applicazioni di data mining* (congress proceedings, university of Pavia, Pavia)
- [9] Sargan, J.D. (1988), *Lecture notes on advanced econometric theory*, Basil Blackwell, Oxford

5 Appendix: Proofs

The following theorem is needed for the proof of Theorem (3):

Theorem 5 *If space β is generically identified, then $\text{rank} [H_1, \dots, H_r] \geq r$*

Proof (by contradiction). Let β be of rank $w < r$. Then there exists a full rank matrix R such that $\lambda = \beta R$ and the first column of λ is a vector of zeros. Consequently λ (and thereby β) is not identified as H_1 is the empty space and thus

$rank(R'_2 H_1) = 0$.

When β is of full rank, it follows that $rank(H_1, \dots, H_r) \geq r$ ■

Proof of theorem 3. by (8) – (9) and Theorem 5 we know that

$$rank([H_{k_1}, \dots, H_{k_m}]) \geq m \quad (14)$$

$$rank([R_j, H_j]' [H_{k_1}, \dots, H_{k_m}]) \geq m$$

as (R_j, H_j) is a matrix of full rank. As H_j is of full column rank, $[k_{j1}, \dots, k_{ji}, R_j]$ is a square, full rank matrix, which together with (14) implies that

$$rank([k_{j1}, \dots, k_{ji}, R_j]' (H_{k_1}, \dots, H_{k_m})) \geq m$$

This combined with (7) means that

$$rank([R_j, k_{ji}]' [H_{k_1}, \dots, H_{k_m}]) = m$$

for at least one column of H_j . ■

We note that

$$rank(k'_{1d} [H_{k_1}, \dots, H_{k_m}]) = 1 \quad (15)$$

Proof of theorem 4. $rank(R'_1 [\beta_{k_1}, \dots, \beta_{k_m}]) = m - 1$, (7), implies that there exists an $s \times 1$ vector a_\perp , such that $R'_1 [\beta_{k_1}, \dots, \beta_{k_m}] a_\perp = 0$.

Therefore $[\beta_{k_1}, \dots, \beta_{k_m}] a_\perp \equiv \gamma \in sp(H_1)$
As $rank([R_1, h_{1d}]' [\beta_{k_1}, \dots, \beta_{k_m}]) = m$, $h'_{1d} \gamma \neq 0$ (with probability one). This implies that we can take $\beta^*_1 = \beta_1 - \gamma \left(\frac{h'_{1d} \beta_1}{h'_{1d} \gamma} \right) \in sp(H^*_1)$. This transformation is of the kind $\beta^* = \beta S$, where S is a matrix with ones on the diagonal and a number of non-zero elements in the first column. All other elements are zero. This matrix S is thus of full rank, which means that $sp(\beta) = sp(\beta^*)$ ■

6 Matlab program

```
function [Hblock,Rblock] = identify(Hblock)
    % For a given set of linear restrictions of the kind beta1 = H1*phi1

    % (without normalizations), this function provides an equivalent identifying
set
    % of restrictions
    r = size(Hblock,2);
    p =size(Hblock{1},1);
    % Get the orthogonal complements
    for f=1:r
        Rblock{f}= null(Hblock{f}');
    end
    % The main loop of the program
    identification = 0;
    % As long as there is no identification run the following loop
```

