A FORTRAN IV QUASI-DECISION ALGORITHM FOR THE $P$-EQUIVALENCE OF TWO MATRICES (*)

L. GOTTUSSO - A. TARSI SANTOLINI

Istituto di Matematica del Politecnico di Milano

SUMMARY - An iterative procedure is given which enables effecting a rather small number of permutations to decide, in a great number of cases, if two matrices can, or cannot, be reduced one to the other by means of permutations on the rows and columns. Some examples of application of this algorithm are given.

In the present paper we wish to take up again, essentially from the numerical point of view, a problem considered in a note presented by C. Böhm and A. Santolini [1][1], which concerns the study of the equivalence of two matrices; by saying that two matrices are equivalent we mean that they can be obtained one from the other by means of permutations on their rows and columns.

In the above mentioned paper a procedure of iterative type was given which can in many cases be used effectively to solve the problem. Moreover a program scheme was proposed, written in Algol, in order to better illustrate in detail the procedure. Owing to the essentially theoretical nature of the work, the practical efficiency of the method had not been established and it had been possible to say very little on those cases in which the procedure was not able to give an answer.

In view of the interest still presented by this question in various practical problems, we have thought it useful to take its study up again, trying, as far as possible, to establish, by means of results of numerical experiments, the actual efficiency of the procedure. We have in this way

(*) Lavoro eseguito nell'ambito del gruppo di ricerca n. 12 del C.N.R.

(1) We refer to this paper for the definitions and notations here adopted.
found that the number of cases in which the procedure was not able to
give an answer, could, bringing some modifications to the method, be greatly
reduced.

Not only, but in the numerical experiments made, we have also obtained
a precise answer.

§ 1. Description of the method.

We have compiled a program, written in Fortran, basing ourselves
essentially on the scheme of the Algol program. In view of our scope, that
is to obtain a functioning program, as flexible and fast as possible, we
naturally had to modify it in various places, but the modifications are not
generally related to the logical structure of the original program.

Let us briefly summarize the theoretical and the practical modifications
the knowledge of which may, we believe, help the reader to understand the
program given here below.

The basic idea of the iterative process consists in substituting to the
two matrices A and B examined, two other matrices C and D, equivalent
to them as regards the final result (that is \( A \sim B \iff C \sim D \)), but of inferior
order or with a larger number of different elements. Each element
\( c_{ik} (d_{ik}) \) contains informations both on the element \( a_{ik} (b_{ik}) \) and on the
relationship between the lines \( i \) and \( k \) of A (B). On the new matrices the non
equivalence control is simpler, the elements being more differentiated. In
the Algol program this was obtained by associating to each element \( a_{ik} \) an
appropriate matrix. Subsequently, the couple element-matrix was charactheri-
zed by a number that itself constituted the element \( c_{ik} \) of the transformed
matrix. In the present program the same result is more simply obtained by
associating to each element an appropriate vector, whose components are
the order couples of the corresponding elements of the lines \( i \) and \( k \).

Moreover, as may be seen from an accurate analysis of the program, we
have tried, for reasons of time, to reduce the number of vectors necessary
for constructing the matrices C and D to the minimum.

The theoretical modification consists essentially in having a further
number, always however very limited (in any case not greater than \( n^2 \)), of
permutations made on the rows and columns of the matrices, whenever the
iterative process has not led to a precise answer. For further details see the
subroutine ENDW of the program. We would like to point out that we have
obtained a precise answer also when examining matrices of small order, of
the type, that, in the above mentioned paper, were given as examples of
undecidable cases.