Disproving the $P$-Matrix Property

Jiří Rohn

Technical report No. V-1111

21.01.2011
Disproving the \( P \)-Matrix Property

Jiří Rohn

Technical report No. V-1111

21.01.2011

Abstract:

We describe an algorithm which for each square matrix \( A \) satisfying \(|(A - I)^{-1}(A + I)x| \leq |x|\) for some \( x \neq 0 \) finds in polynomial time a nonpositive principal minor of \( A \), thus disproving its \( P \)-property. Such an \( x \) exists whenever \( A \) is not a \( P \)-matrix and \( A - I \) is nonsingular, but its construction in full generality is not given here; we only show that if \( |((A + I)^{-1}(A - I))_{jj}| \geq 1 \) for some \( j \) (a situation frequently encountered with randomly generated matrices), then \( x \) can be taken as \( x = ((A + I)^{-1}(A - I))_{*j} \).

Keywords:

Not-\( P \)-matrix, nonpositive principal minor, algorithm.

---

\(^1\)This work was supported by the Czech Republic Grant Agency under grant 201/09/1957 and by the Institutional Research Plan AV0Z10300504.
1 Introduction

Given an $n \times n$ matrix $A$ and a subset $J \subseteq \{1, \ldots, n\}$, denote by $A[J]$ the submatrix of $A$ consisting of rows and columns whose indices belong to $J$. In case of $J = \emptyset$ we define $A[\emptyset]$ to be the empty matrix, and we set $\det(A[\emptyset]) = 1$.

Submatrices of the form $A[J]$, $J \neq \emptyset$ are called principal submatrices, and $A$ is said to be a $P$-matrix (or, to possess the $P$-property) if determinants of all the principal submatrices (also called principal minors) are positive. Since there are $2^n - 1$ principal minors, the problem of verifying the $P$-property can be seen computationally difficult directly from the definition, and this intuitive view was confirmed in 1994 by Coxson’s result [2] saying that checking the $P$-property is a co-NP-complete problem.

In this paper we focus on the task of disproving the $P$-property, i.e., of finding a subset $J$ for which $\det(A[J]) \leq 0$. As far as we know, this problem has not been tackled in full generality as yet. Our main tool throughout the paper will be the function

$$f(t) = \det(A - I) \det(C - \text{diag}(t)), \quad t \in \mathbb{R}^n,$$

(1.1)

where

$$C = (A - I)^{-1}(A + I)$$

(thus assuming that $A - I$ is nonsingular), and diag($t$) is a diagonal matrix with diagonal vector $t$. We shall later essentially use the fact that $f$ is linear in each $t_i$ (because the variable $t_i$ appears in the matrix $C - \text{diag}(t)$ only once, namely in the $ii$th position). In [6] we proved that for each $y \in \{-1, 1\}^n$ there holds

$$f(y) = 2^n \det(A[J(y)]),$$

(1.2)

where

$$J(y) = \{j \mid y_j = -1\}.$$

Thus, the task of finding a nonpositive minor reduces to that of finding a $y \in \{-1, 1\}^n$ such that

$$f(y) \leq 0$$

holds. To do this, we proceed in two steps.

First, we show that if

$$|Cx| \leq |x|, \quad x \neq 0$$

(1.3)

holds for some $x$, then directly from $C$ and $x$ we can easily compute a vector $y \in [-1, 1]^n$ such that

$$f(y) = 0.$$

This already finds a nonpositive value of $f$, but still $y \in [-1, 1]^n$, whereas we need $y \in \{-1, 1\}^n$. Therefore in the second step, using the above-mentioned linearity of $f(t)$ in each $t_i$, we move the $y_i$’s towards the endpoints of the interval $[-1, 1]$ so that the nonpositiveness of $f$ keeps to be preserved. In this way, after a finite number of steps, we find a $y \in \{-1, 1\}^n$ for which $f(y) \leq 0$ so that for $J = \{j \mid y_j = -1\}$ we have $\det(A[J]) \leq 0$ and our problem is solved. This approach is formalized in the algorithm description in Fig. 2.1.

This brings us again to the beginning, namely to finding an $x$ satisfying (1.3). We do not solve the problem in full generality here, postponing its solution to a forthcoming paper. It is
sufficient to mention here that if $A - I$ is nonsingular, then $A$ is NOT a $P$-matrix if and only if such an $x$ exists (Rump [7], Rohn [3]), and it can be found by a not-a-priori exponential algorithm (Rohn [3]). In the present paper we shall only show a special case in which such an $x$ can be found easily, but this special case, as far as our numerical experiments show, occurs “almost always” for randomly generated examples: if
\[(C^{-1})_{jj} \geq 1\] (1.4)
for some $j$, then
\[x = (C^{-1})_{*j}\] (1.5)
is nontrivial and satisfies (1.3) (thus adding implicitly nonsingularity of $A + I$ to our assumptions). Indeed, we have $|Cx| = I_j \leq |x|$ because of (1.4).

2 The algorithm

Our algorithm is formulated as follows:

```
(01) function J = vec2min(A, x)
(02) % VECtor TO MINor.
(03) % Input: A, $x \neq 0$ with $|(A-I)^{-1}(A+I)x| \leq |x|$. 
(04) % Output: J with $\det(A[J]) \leq 0$.
(05) n = length(x); I = eye(n);
(06) C = (A - I)^{-1}(A + I);
(07) for i = 1 : n
(08) if x_i \neq 0, y_i = (Cx)_i/x_i; else y_i = 1; end
(09) end
(10) d = det(A - I);
(11) for i = 1 : n
(12) if y_i \neq -1 and y_i \neq 1
(13) y_i = 1;
(14) if d \cdot \det(C - \text{diag}(y)) > 0, y_i = -1; end
(15) end
(16) end
(17) J = \{ i | y_i = -1 \};
```

Figure 2.1: An algorithm for finding a nonpositive minor.

The algorithm is substantiated by the following theorem.
Theorem 1. For each square matrix $A$ and $x$ specified in line (03) the algorithm \texttt{vec2min} (Fig. 2.1) produces in polynomial time a subset $J$ for which $\det(A[J]) \leq 0$.

Proof. We shall use the function

$$f(t) = \det(A - I) \det(C - \text{diag}(t)), \quad t \in \mathbb{R}^n$$

introduced in (1.1). As explained in Section 1, $f$ is linear in each $t_i$. First we show that the vector $y$ computed in lines (07)-(09) satisfies $f(y) = 0$.

By assumption, $|Cx| \leq |x|$; in particular, for each $i$, $x_i = 0$ implies $(Cx)_i = 0$. Thus the vector $y$ constructed in lines (07)-(09) satisfies $|y_i| \leq 1$ and $(Cx)_i = y_i x_i$ for each $i$, hence $Cx = \text{diag}(y)x$, which gives that $(C - \text{diag}(y))x = 0$, where $x \neq 0$, so that $\det(C - \text{diag}(y)) = 0$, implying $f(y) = 0$.

Next we prove by induction on $i = 0, 1, \ldots, n$ that the vector $y$ obtained after completing line (15) satisfies

$$y_j = \pm 1 \quad (j = 1, \ldots, i)$$  \hspace{1cm} (2.1)

and

$$f(y) \leq 0.$$  \hspace{1cm} (2.2)

This is obviously so for $i = 0$. Thus assume that the induction hypothesis holds for some $i - 1 \geq 0$. At that moment,

$$f(y_1, \ldots, y_{i-1}, y_i, \ldots, y_n) \leq 0$$

for some $y_i \in [-1, 1]$. If $y_i = -1$ or $y_i = 1$, then we are done (line (12)). Thus assume that $y_i \in (-1, 1)$. If

$$f(y_1, \ldots, y_{i-1}, 1, \ldots, y_n) \leq 0,$$

then $y_i$ is set to 1 and (2.1), (2.2) are satisfied. If

$$f(y_1, \ldots, y_{i-1}, 1, \ldots, y_n) > 0,$$

then the function of one variable $t_i$

$$f(y_1, \ldots, y_{i-1}, t_i, \ldots, y_n)$$

is linear (as emphasized above), is positive at $t_i = 1$ and nonpositive at $t_i = y_i \in (-1, 1)$, hence it is increasing in $[-1, 1]$, which means that it is negative at $-1$. In this case $y_i$ is set to $-1$ (line (14)) and the induction hypothesis (2.1), (2.2) is proved.

In this way, we obtain that the vector $y$ constructed after completing the for-loop in lines (11)-(16) is a $\pm 1$-vector satisfying

$$f(y) = \det(A - I) \det(C - \text{diag}(y)) \leq 0.$$  \hspace{1cm} (2.3)

Now from (1.2) we have

$$\det(A[J]) = \frac{1}{2\pi} f(y) \leq 0,$$  \hspace{1cm} (2.3)
where
\[ J = \{ j \mid y_j = -1 \} \]
(see line (17)), which shows that the principal minor \( \det(A[J]) \) is nonpositive (output description in line (04)).

Polynomiality of the algorithm follows from the Bareiss’ result \[1\] proving existence of a polynomial-time algorithm for computing the determinant. This completes the proof. \( \square \)

3 Example

Using MATLAB, consider the 100 \( \times \) 100 randomly generated matrix

\[
\begin{bmatrix}
\text{rand('state',1); A=rand(100,100);}
\end{bmatrix}
\]

which can be reproduced because of the use of \texttt{rand('state',1)}. Here we have
\[
(C^{-1})_{35,35} = 12.6592 > 1,
\]
hence the vector \( x = (C^{-1})_{35} \) satisfies (1.3) (see (1.4), (1.5)). We give here for space reasons the vector \( x \) reshaped as a 20 \( \times \) 5 matrix to be read columnwise:

\[
\begin{bmatrix}
\text{reshape(x,20,5)}
\end{bmatrix}
\]

Now, applying the algorithm \texttt{vec2min}, we get a 30 \( \times \) 30 principal submatrix having a negative determinant:

\[
\text{vec2min}
\]
```matlab
>> tic, J=vec2min(A,x); J, det(A(J,J)), toc
J =
   Columns 1 through 10
       12    16    17    21    22    24    32    33    36    41
   Columns 11 through 20
       44    47    48    51    56    58    66    68    69    71
   Columns 21 through 30
       82    84    88    89    90    92    94    98    99   100
ans =
   -13.6141
Elapsed time is 0.517556 seconds.

Due to the well-known fact that determinants of large-size matrices computed in floating
point may be afflicted with big roundoff errors, the result may be considered hardly con-
vincing. To remove this doubt, we may compute a verified determinant of \( A[J] \) by means
of \[3]:

```matlab
>> format long, tic, dt=verdet(A(J,J)), toc
intval dt =
Elapsed time is 3.701657 seconds.

This shows that the principal minor is verified negative. Notice that the verification lasted
seven times longer than the computation of the main result itself; this is due to the verification
procedures involved (the verified determinant is computed as product of verified eigenvalues).
```
Bibliography


