Institute of Computer Science
Academy of Sciences of the Czech Republic

Disproving the P-Matrix Property

Jiří Rohn

Technical report No. V-1111

21.01.2011

Pod Vodárenskou věží 2, 182 07 Prague 8, phone: +420 266 051 111, fax: +420 286 585 789,
e-mail:rohn@cs.cas.cz

Institute of Computer Science
Academy of Sciences of the Czech Republic

Disproving the P-Matrix Property

Jiří Rohn[1]

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 $x = ((A+I)^{-1}(A-I))_{\bullet j}$.

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 $J \subseteq \{1, \dots, n\}$, 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

$$f(t) = \det(A - I)\det(C - \operatorname{diag}(t)), \qquad t \in \mathbb{R}^n, \tag{1.1}$$

where

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

(thus assuming that $A - I$ is nonsingular), and $\operatorname{diag}(t)$ 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 $C - \operatorname{diag}(t)$ only once, namely in the iith position). In [6] we proved that for each $y \in \{-1, 1\}^n$ there holds

$$f(y) = 2^n \det(A[J(y)]), \tag{1.2}$$

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

$$|Cx| \leq |x|, \qquad x \neq 0 \tag{1.3}$$

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 [4]), and it can be found by a not-a-priori exponential algorithm (Rohn [5])*. 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*

$$|(C^{-1})_{jj}| \geq 1 \tag{1.4}$$

for some j, then

$$x = (C^{-1})_{\bullet j} \tag{1.5}$$

is nontrivial and satisfies (1.3) (thus adding implicitly nonsingularity of $A+I$ to our assumptions). Indeed, we have $|Cx| = I_{\bullet j} \leq |x|$ because of (1.4).

2 The algorithm

Our algorithm is formulated as follows:

```
function J = vec2min(A, x)
% VECtor TO MINor.
% Input: A, x ≠ 0 with |(A − I)^{-1}(A + I)x| ≤ |x|.
% Output: J with det(A[J]) ≤ 0.
n = length(x); I = eye(n);
C = (A − I)^{-1}(A + I);
for i = 1 : n
   if x_i ≠ 0, y_i = (Cx)_i/x_i; else y_i = 1; end
end
d = det(A − I);
for i = 1 : n
   if y_i ≠ −1 and y_i ≠ 1
      y_i = 1;
      if d · det(C − diag(y)) > 0, y_i = −1; end
   end
end
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* **vec2min** *(Fig. 2.1) produces in polynomial time a subset J for which* $\det(A[J]) \le 0$*.*

Proof. We shall use the function

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

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

$$y_j = \pm 1 \qquad (j = 1, \ldots, i) \tag{2.1}$$

and

$$f(y) \le 0. \tag{2.2}$$

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

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

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

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

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 ± 1-vector satisfying

$$f(y) = \det(A - I)\det(C - \operatorname{diag}(y)) \le 0.$$

Now from (1.2) we have

$$\det(A[J]) = \tfrac{1}{2^n} f(y) \le 0, \tag{2.3}$$

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. □

3 Example

Using MATLAB, consider the 100×100 randomly generated matrix

```
>> rand('state',1); A=rand(100,100);
```

which can be reproduced because of the use of `rand('state',1)`. Here we have

$$(C^{-1})_{35,35} = 12.6592 > 1,$$

hence the vector $x = (C^{-1})_{\bullet 35}$ satisfies (1.3) (see (1.4), (1.5)). We give here for space reasons the vector x reshaped as a 20×5 matrix to be read columnwise:

```
>> reshape(x,20,5)
ans =
    4.4690    8.9095   -3.8182   -5.0737   -4.4761
    1.6366   -8.4748   -7.3249    2.9782   -3.3127
   -2.3315    9.0209    1.2753   -2.8727   -0.4856
   -7.6401   -3.5210   -4.0098    9.4804   -2.7190
    3.9239   -3.0288    4.6342   11.7851   -0.1305
    2.1388   -2.5015    3.5267   -8.6483   -0.7691
    8.3985    0.9246    2.7164    0.5026   -7.0660
   12.2784   -7.2008    2.3232    4.2166   -1.7433
    4.0095    5.0032   -0.3533    2.5050    2.0941
   -0.4785   -2.9658   10.6911   -5.8337   -1.5813
   -8.3985   -4.6726   -1.8939   -8.0005  -10.7130
    1.6613    3.4313   -4.2710    5.4276    3.7095
   -1.7192    9.5956   -4.8951  -12.5372   -0.4862
    2.4016    7.8701   -0.7614   -0.2599  -12.0846
   15.1720   12.6592    5.4886   -2.2354   -2.5968
   10.9240  -10.0155    3.4556   -0.1458    7.0576
   -2.2835    2.3146   -5.8519  -11.0918    2.7710
   14.4889  -10.6511    6.5952    4.3883  -11.4782
  -10.7842   -7.1126   -7.5142   -1.4689    7.1920
    3.5978   -2.2741   -2.2772   -3.5726    8.7554
```

Now, applying the algorithm **vec2min**, we get a 30×30 principal submatrix having a negative determinant:

```
>> 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 convincing. To remove this doubt, we may compute a *verified* determinant of $A[J]$ by means of [3]:

```
>> format long, tic, dt=verdet(A(J,J)), toc
intval dt =
[ -13.61413607696574, -13.61413607695881]
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

[1] E. F. Bareiss, *Sylvester's identity and multistep integer-preserving Gaussian elimination*, Mathematics of Computation, 103 (1968), pp. 565–578. 5

[2] G. E. Coxson, *The P-matrix problem is co-NP-complete*, Mathematical Programming, 64 (1994), pp. 173–178. 2

[3] J. Rohn, *VERDET: Verified determinant of a complex (or real) matrix*, 2008. `http://uivtx.cs.cas.cz/~rohn/matlab/verdet.html`. 6

[4] J. Rohn, *An algorithm for solving the absolute value equation*, Electronic Journal of Linear Algebra, 18 (2009), pp. 589–599. `http://www.math.technion.ac.il/iic/ela/ela-articles/articles/vol18_pp589-599.pdf`. 3

[5] J. Rohn, *An algorithm for finding a singular matrix in an interval matrix*, Technical Report 1087, Institute of Computer Science, Academy of Sciences of the Czech Republic, Prague, November 2010. `http://uivtx.cs.cas.cz/~rohn/publist/singreg.pdf`. 3

[6] J. Rohn, *On Rump's characterization of P-matrices*, Technical Report 1093, Institute of Computer Science, Academy of Sciences of the Czech Republic, Prague, November 2010. `http://uivtx.cs.cas.cz/~rohn/publist/rumppmat.pdf`. 2

[7] S. M. Rump, *On P-matrices*, Linear Algebra and Its Applications, 363 (2003), pp. 237–250. 3