An algorithmic characterization of P-matricity

Ibtihel BEN GHARBIA, J. Charles GILBERT

An algorithmic characterization of
P-matricity

Ibtihel Ben Gharbia[†], J. Charles Gilbert[†]

Project-Team Pomdapi

Abstract: It is shown that a nondegenerate square real matrix M is a **P**-matrix if and only if, whatever is the real vector q, the Newton-min algorithm does not cycle between two points when it is used to solve the linear complementarity problem $0 \leqslant x \perp (Mx + q) \geqslant 0$.

Key-words: Linear complementarity problem, semismooth Newton method, Newton-min algorithm, **NM**-matrix, **P**-matricity characterization, **P**-matrix.

[†] INRIA Paris-Rocquencourt, project-team Pomdapi, BP 105, F-78153 Le Chesnay (France); e-mails : `Ibtihel.Ben_Gharbia@inria.fr`, `Jean-Charles.Gilbert@inria.fr`.

Une caractérisation algorithmique de la P-matricité

Résumé :   Nous montrons qu'une matrice réelle carrée non dégénérée M est une **P**-matrice si, et seulement si, quel que soit le vecteur réel q, l'algorithme de Newton-min ne fait pas de cycle de deux points lorsqu'il est utilisé pour résoudre le problème de complémentarité linéaire $0 \leqslant x \perp (Mx + q) \geqslant 0$.

Mots-clés :   Algorithme de Newton-min, caractérisation de la **P**-matricité, méthode de Newton semi-lisse, **NM**-matrice, **P**-matrice, problème de complémentarité linéaire.

<div align="center">**Table of contents**</div>

1 Introduction

Being given a positive integer n, a matrix $M \in \mathbb{R}^{n \times n}$, and a vector $q \in \mathbb{R}^n$, the *linear complementarity problem* consists in determining a vector $x \in \mathbb{R}^n$ such that

$$x \geqslant 0, \qquad Mx + q \geqslant 0, \qquad \text{and} \qquad x^\mathsf{T}(Mx + q) = 0.$$

Inequalities have to be understood componentwise; for example $x \geqslant 0$ means $x_i \geqslant 0$ for all indices $i \in [\![1, n]\!] := \{1, \ldots, n\}$. The Euclidean scalar product of two vectors u and v is denoted by $u^\mathsf{T}v = \sum_i u_i v_i$. This problem is sometimes written in compact form as follows

$$\text{LCP}(M, q): \qquad 0 \leqslant x \perp (Mx + q) \geqslant 0,$$

where the sign $\perp$ is used to denote the perpendicularity with respect to the Euclidean scalar product.

Let M_{IJ} denote the submatrix of a matrix M formed of its rows with indices in I and its columns with indices in J. An $n \times n$ real matrix M is a **P**-*matrix* if its principal minors are positive: for all $I \subseteq [\![1, n]\!]$, $\det M_{II} > 0$ (by convention $\det M_{\varnothing\varnothing} = 1$). The class of **P**-matrices is denoted by **P** (the order n of the matrices is implicit and assumed fixed in that notation). These matrices have an eminent role in linear complementarity problems since it can be shown that $M \in \mathbf{P}$ if and only if $\text{LCP}(M, q)$ has one and only one solution, whatever is q [25, 7; 1958]. Another characterization of **P**-matricity, which will be useful below, is the following [9, 7; 1962]:

$$M \in \mathbf{P} \qquad \Longleftrightarrow \qquad \text{any } x \text{ verifying } x \cdot (Mx) \leqslant 0 \text{ vanishes}, \tag{1.1}$$

where we have denoted by $u \cdot v$ the *Hadamard product* of the vectors u and v, which is a vector whose ith component is $u_i v_i$.

There are many other equivalent conditions for a matrix to be in **P**, than the three given above [1, 14, 24, 7, 23]. This paper gives still another characterization of **P**-matricity, expressed in terms of a property of the algorithm for solving $\text{LCP}(M, q)$ that is described in section 2, the Newton-min algorithm. It is shown that $M \in \mathbf{P}$ if and only if the Newton-min algorithm does not cycle between two distinct points, whatever is q, when it is used to solve $\text{LCP}(M, q)$.

2 The Newton-min algorithm

Let I be a subset of $[\![1, n]\!]$. We denote by $I^c := [\![1, n]\!] \setminus I$ the complementary set of I in $[\![1, n]\!]$ and by $|I|$ the cardinality of I. For a vector $v \in \mathbb{R}^n$, v_I is the vector in $\mathbb{R}^{|I|}$ whose components are the components v_i's of v with index $i \in I$.

The *Newton-min algorithm* is a short name for the semismooth Newton method [21, 22, 15, 27; 1977-2011] for solving the nonsmooth piecewise linear equation

$$\min(x, Mx + q) = 0 \in \mathbb{R}^n,$$

which is equivalent to LCP(M, q) [18, 20; 1976-1977]. Above, the "min" operator acts componentwise. More specifically, at the current iterate $x \in \mathbb{R}^n$, the algorithm determines a set of indices

$$I \equiv I(x) := \{i \in [\![1, n]\!] : x_i > (Mx + q)_i\} \tag{2.1}$$

and computes the next iterate x^+ as the unique solution to the system

$$x^+_{I^c} = 0 \qquad \text{and} \qquad (Mx^+ + q)_I = 0. \tag{2.2}$$

The uniqueness of the solution to the system (2.2) is certainly ensured if M is *nondegenerate*, meaning that the principal minors of M do not vanish (we denote by $\mathbf{D^c}$ the set of nondegenerate matrices), so that this well-posedness condition of the algorithm will always be assumed. In that case,

$$x^+_I = -M^{-1}_{II} q_I,$$

where M^{-1}_{II} is a compact notation for $(M_{II})^{-1}$. The Newton-min algorithm can be traced back at least to Aganagić in [1; 1984]; see paragraph 7 of the introduction of [3] for more details on its origin and a discussion on the contributions from [6, 17, 10, 5, 4, 13, 16]; see also [12].

A more general form of the algorithm allows the iteration to put in I any of the indices in

$$E \equiv E(x) := \{i \in [\![1, n]\!] : x_i = (Mx + q)_i\}$$

(see section 2 in [3] for instance). We do not consider such a general version of the algorithm below. The choice of the present version of the Newton-min algorithm, which imposes $I \cap E = \varnothing$, is motivated by the following considerations. First, it would certainly be more difficult to analyze the cycles of the Newton-min algorithm, which is what we do below, if this one was not a *Markov process* in x, i.e., if the next iterate x^+ would not only depend on the current iterate x, as this would be the case if the indices in E going in I were not determined by a rule depending only on x. Next, with the rule (2.1), the linear system to solve at each iteration has a smaller size $|I|$, which makes this rule numerically natural.

By definition (see (2.2)), except possibly for the initial iterate, the Newton-min algorithm only visits points x satisfying $x \cdot (Mx + q) = 0$ or equivalently

$$x_{I^c} = 0 \qquad \text{and} \qquad (Mx + q)_I = 0, \tag{2.3}$$

for some (possibly empty) index set $I \subseteq [\![1, n]\!]$. The unique point x, related to some index set I, satisfying (2.3) is denoted by

$$x^{(I)}$$

and is called a *node*. Clearly, there holds $x^{(\varnothing)} = 0$. Since there are 2^n different index sets I, there are *at most* 2^n nodes (two different index sets may yield the same node; for example, there is a single node, zero, if and only if $q = 0$).

Since the Newton-min algorithm (2.1)-(2.2) is a Markov process in x and only visits nodes and since the number of nodes is finite, either the algorithm converges or it cycles by visiting a finite number of distinct nodes repetitively. The identification of the conditions of convergence of the Newton-min algorithm may, therefore, go through the analysis of the conditions that prevent cycles from occurring. The case of the 2-cycles is considered in the next section (for $k \geqslant 2$, a *k-cycle* is a cycle made of k distinct nodes). Let us formalize this a bit more.

For $k \geqslant 2$, we denote by $\mathbf{NM}_k$ the set of nondegenerate matrices $M \in \mathbb{R}^{n \times n}$ such that the Newton-min algorithm does not produce k-cycles when it is used to solve $\mathrm{LCP}(M, q)$, whatever is q. Therefore, for the reasons given at the beginning of the previous paragraph,

$$\mathbf{NM} := \bigcap_{k \geqslant 2} \mathbf{NM}_k \tag{2.4}$$

is the class of nondegenerate matrices $M \in \mathbb{R}^{n \times n}$ such that the Newton-min algorithm converges, whatever are q and the initial point. In this paper, we prove that

$$\mathbf{NM}_2 = \mathbf{P}.$$

Since $\mathbf{NM}$ is included in $\mathbf{NM}_2$, this identity implies in particular that

$$\mathbf{NM} \subseteq \mathbf{P}, \tag{2.5}$$

i.e., the set of matrices ensuring the convergence of the Newton-min algorithm, whatever are q and the initial point, is contained in $\mathbf{P}$. It has been shown in [2, 3; 2009] that $\mathbf{P} \subseteq \mathbf{NM}$ if $n = 1$ or $n = 2$ (hence $\mathbf{P} = \mathbf{NM}$ in that case) and that $\mathbf{P} \nsubseteq \mathbf{NM}$ if $n \geqslant 3$ (hence the inclusion (2.5) is strict in that case).

We recall that an $\mathbf{M}$-*matrix* is a $\mathbf{P}$-matrix with nonpositive off-diagonal elements ($M_{ij} \leqslant 0$ when $i \neq j$). According to [1; 1984, theorem 6.2],

$$\mathbf{M} \subseteq \mathbf{NM}.$$

It is also known that $M \in \mathbf{NM}$ if M is sufficiently close to an $\mathbf{M}$-matrix [13; 2003].

3   A dual characterization of the absence of 2-cycle

The Newton-min algorithm makes a 2-*cycle* if it generates successively the iterates

$$x^{(I)} \to x^{(J)} \to x^{(I)} \to x^{(J)} \to \cdots,$$

for distinct nodes $x^{(I)} \neq x^{(J)}$ associated with index sets I and $J \subseteq [\![1, n]\!]$. The dual characterization of the absence of 2-cycle of the Newton-min algorithm given in this section is a first step in the derivation of our main result (theorem 4.2).

We start by recalling Motzkin's theorem of the alternative (see [11; 2010, theorem 3.15 or 7.17] for instance), on which the dual characterization of proposition 3.2 rests.

> **Lemma 3.1 (Motzkin)** *Let $A \in \mathbb{R}^{m_A \times n}$ and $B \in \mathbb{R}^{m_B \times n}$ be two matrices with the same number of columns. Then, there is a vector $x \in \mathbb{R}^n$ satisfying*
>
> $$Ax < 0 \qquad and \qquad Bx \leqslant 0 \tag{3.1}$$
>
> *if and only if there is no vector $\alpha \in \mathbb{R}_+^{m_A} \setminus \{0\}$ and $\beta \in \mathbb{R}_+^{m_B}$ such that*
>
> $$A^{\mathsf{T}} \alpha + B^{\mathsf{T}} \beta = 0. \tag{3.2}$$

Strict inequalities on vectors must also be understood componentwise; hence $Ax < 0$ in (3.1) means $(Ax)_i < 0$ for all $i \in [\![1, m_A]\!]$. The (components of the) vectors α and β in the lemma will be called *Motzkin multipliers* below. There is one such scalar multiplier for each inequality in (3.1).

   The next proposition gives a dual condition on the matrix M such that the Newton-min algorithm does not cycle between two given nodes when it is used to solve $\mathrm{LCP}(M, q)$, whatever is the vector q. The dual aspect of condition (3.3) comes from lemma 3.1 and therefore involves, like in (3.2), the transpose of (modified) submatrices of M. The *symmetric difference* of two index sets I and $J \subseteq [\![1, n]\!]$ is denoted by

$$I \triangle J := (I \cap J^c) \cup (I^c \cap J) = (I \cup J) \setminus (I \cap J).$$

For a subset $K \subseteq [\![1, n]\!]$, we also note $M_{KK}^{-\mathsf{T}} := ((M_{KK})^{-1})^{\mathsf{T}}$.

Proposition 3.2 (no cycle $x^{(I)} \to x^{(J)} \to x^{(I)}$) *Suppose that $M \in \mathbf{D^c}$ and let be given two different subsets I and $J \subseteq [\![1, n]\!]$. Then the following conditions are equivalent:*

(i) *there is an $\alpha \in \mathbb{R}_+^{|I \triangle J|} \setminus \{0\}$ such that*

$$\begin{pmatrix} M_{(I \cap J^c)(I \cap J^c)} & -M_{(I \cap J^c)(I^c \cap J)} \\ -M_{(I^c \cap J)(I \cap J^c)} & M_{(I^c \cap J)(I^c \cap J)} \end{pmatrix}^{\mathsf{T}} \alpha$$

$$\geqslant \begin{pmatrix} -M_{(I \cap J)(I \cap J^c)} & M_{(I \cap J)(I^c \cap J)} \end{pmatrix}^{\mathsf{T}} M_{(I \cap J)(I \cap J)}^{-\mathsf{T}} \begin{pmatrix} -M_{(I \cap J^c)(I \cap J)} \\ M_{(I^c \cap J)(I \cap J)} \end{pmatrix}^{\mathsf{T}} \alpha, \qquad (3.3)$$

where the right hand side is zero when $I \cap J = \varnothing$,

(ii) *whatever is q, the Newton-min algorithm does not make the cycle $x^{(I)} \to x^{(J)} \to x^{(I)}$ when it is used to solve $\mathrm{LCP}(M, q)$.*

PROOF. 1) *Preliminaries.* Let us first specify the conditions on q such that the Newton-min algorithm makes the cycle $x^{(I)} \to x^{(J)} \to x^{(I)}$ when it is used to solve $\mathrm{LCP}(M, q)$. Let $x^1 = x^{(I)}$, so that by (2.3),

$$\begin{cases} x_I^1 = -M_{II}^{-1} q_I \\ x_{I^c}^1 = 0 \end{cases} \qquad \text{and} \qquad \begin{cases} (Mx^1 + q)_I = 0 \\ (Mx^1 + q)_{I^c} = q_{I^c} - M_{I^c I} M_{II}^{-1} q_I. \end{cases}$$

We have used the nonsingularity of M_{II}, which comes from the nondegeneracy of M. Now, by definition of the Newton-min algorithm (2.1)-(2.2), the next iterate x^2 is $x^{(J)}$ if and only if

$$\underbrace{-(M_{II}^{-1} q_I)_{J^c} \leqslant 0}_{\beta_{I \cap J^c}}, \qquad \underbrace{(M_{II}^{-1} q_I)_J < 0}_{\alpha'_{I \cap J}},$$

$$\underbrace{q_{I^c \cap J} < M_{(I^c \cap J)I} M_{II}^{-1} q_I}_{\alpha_{I^c \cap J}}, \qquad \text{and} \qquad M_{(I \cup J)^c I} M_{II}^{-1} q_I \leqslant q_{(I \cup J)^c}. \qquad (3.4)$$

The vectors under the braces are Motzkin multipliers, which will be used below. In that case

$$\begin{cases} x_J^2 = -M_{JJ}^{-1} q_J \\ x_{J^c}^2 = 0 \end{cases} \qquad \text{and} \qquad \begin{cases} (Mx^2 + q)_J = 0 \\ (Mx^2 + q)_{J^c} = q_{J^c} - M_{J^c J} M_{JJ}^{-1} q_J. \end{cases}$$

We have used the nonsingularity of M_{JJ}, which comes from the nondegeneracy of M. Now, the next iterate is $x^{(I)}$ if and only if

$$\underbrace{-(M_{JJ}^{-1} q_J)_{I^c} \leqslant 0}_{\beta_{I^c \cap J}}, \qquad \underbrace{(M_{JJ}^{-1} q_J)_I < 0}_{\alpha''_{I \cap J}},$$

$$\underbrace{q_{I \cap J^c} < M_{(I \cap J^c)J} M_{JJ}^{-1} q_J}_{\alpha_{I \cap J^c}}, \qquad \text{and} \qquad M_{(I \cup J)^c J} M_{JJ}^{-1} q_J \leqslant q_{(I \cup J)^c}. \qquad (3.5)$$

The vectors under the braces are Motzkin multipliers, which will be used below. We have shown that the Newton-min algorithm makes the cycle $x^{(I)} \to x^{(J)} \to x^{(I)}$ if and only if q satisfies the linear inequalities in (3.4)-(3.5).

Observe now that the components of q with indices in $(I \cup J)^c$ intervene only in the last inequalities in (3.4)-(3.5) and that these inequalities can be satisfied by taking these components of q sufficiently large. Therefore, below, we do not have to consider the satisfiability of these last inequalities. This is the reason why we have not assigned Motzkin multipliers to these inequalities.

By Motzkin's theorem of the alternative (lemma 3.1), there is a q satisfying the linear inequalities in (3.4)-(3.5) if and only if one cannot find

$$(\alpha, \alpha', \alpha'', \beta) \in \mathbb{R}_+^{|I \triangle J|} \times \mathbb{R}_+^{|I \cap J|} \times \mathbb{R}_+^{|I \cap J|} \times \mathbb{R}_+^{|I \triangle J|}$$

(these are the vectors under the braces in (3.4)-(3.5)) such that $(\alpha, \alpha', \alpha'') \neq 0$ and

- $(M_{II}^{-\mathsf{T}})_{I \cap J^c} \begin{pmatrix} -\beta_{I \cap J^c} \\ \alpha'_{I \cap J} \end{pmatrix} - (M_{II}^{-\mathsf{T}})_{I \cap J^c} M_{(I^c \cap J)I}^{\mathsf{T}} \alpha_{I^c \cap J} + \alpha_{I \cap J^c} = 0,$ \hfill (3.6a)

- $(M_{II}^{-\mathsf{T}})_{I \cap J} \begin{pmatrix} -\beta_{I \cap J^c} \\ \alpha'_{I \cap J} \end{pmatrix} - (M_{II}^{-\mathsf{T}})_{I \cap J} M_{(I^c \cap J)I}^{\mathsf{T}} \alpha_{I^c \cap J} + (M_{JJ}^{-\mathsf{T}})_{I \cap J} \begin{pmatrix} \alpha''_{I \cap J} \\ -\beta_{I^c \cap J} \end{pmatrix}$
  $- (M_{JJ}^{-\mathsf{T}})_{I \cap J} M_{(I \cap J^c)J}^{\mathsf{T}} \alpha_{I \cap J^c} = 0,$ \hfill (3.6b)

- $\alpha_{I^c \cap J} + (M_{JJ}^{-\mathsf{T}})_{I^c \cap J} \begin{pmatrix} \alpha''_{I \cap J} \\ -\beta_{I^c \cap J} \end{pmatrix} - (M_{JJ}^{-\mathsf{T}})_{I^c \cap J} M_{(I \cap J^c)J}^{\mathsf{T}} \alpha_{I \cap J^c} = 0,$ \hfill (3.6c)

where A_R denotes the submatrix of a matrix A formed of its rows with indices in R.

2) *The case $I \cap J = \varnothing$.* Then equation (3.6b) is not present, there are no Motzkin multipliers α' and α'', $I \cap J^c = I$, $I^c \cap J = J$, $I \triangle J = I \cup J$, and the above claim simplifies as follows: there is a q satisfying the linear inequalities (3.4)-(3.5) if and only if one cannot find

$$(\alpha, \beta) \in \mathbb{R}_+^{|I \cup J|} \times \mathbb{R}_+^{|I \cup J|},$$

such that $\alpha \neq 0$ and

$$\begin{pmatrix} M_{II} & -M_{IJ} \\ -M_{JI} & M_{JJ} \end{pmatrix}^{\mathsf{T}} \begin{pmatrix} \alpha_I \\ \alpha_J \end{pmatrix} = \begin{pmatrix} \beta_I \\ \beta_J \end{pmatrix} \geqslant 0.$$

After discarding β, the contrapositive of the claim becomes the equivalence $(i) \Leftrightarrow (ii)$ with zero in the right hand side of (3.3).

3) *The general case.* With the notation

$$u := \begin{pmatrix} -\beta_{I \cap J^c} \\ \alpha'_{I \cap J} \end{pmatrix} - M_{(I^c \cap J)I}^{\mathsf{T}} \alpha_{I^c \cap J} \quad \text{and} \quad v := \begin{pmatrix} \alpha''_{I \cap J} \\ -\beta_{I^c \cap J} \end{pmatrix} - M_{(I \cap J^c)J}^{\mathsf{T}} \alpha_{I \cap J^c} \tag{3.7}$$

the system (3.6) reads equivalently

$$\begin{pmatrix} (M_{II}^{-\mathsf{T}})_{I \cap J^c} u \\ (M_{II}^{-\mathsf{T}})_{I \cap J} u \\ 0 \end{pmatrix} + \begin{pmatrix} \alpha_{I \cap J^c} \\ 0 \\ \alpha_{I^c \cap J} \end{pmatrix} + \begin{pmatrix} 0 \\ (M_{JJ}^{-\mathsf{T}})_{I \cap J} v \\ (M_{JJ}^{-\mathsf{T}})_{I^c \cap J} v \end{pmatrix} = 0. \tag{3.8}$$

If we multiply the first two equations to the left by M_{II}^{T}, if we multiply the last two equations to the left by M_{JJ}^{T}, and if we use the second equation, we obtain the equivalent system

$$u + M_{(I \cap J^c)I}^{\mathsf{T}} \alpha_{I \cap J^c} - M_{(I \cap J)I}^{\mathsf{T}} (M_{II}^{-\mathsf{T}})_{I \cap J} u = 0, \tag{3.9a}$$

$$v + M_{(I^c \cap J)J}^{\mathsf{T}} \alpha_{I^c \cap J} + M_{(I \cap J)J}^{\mathsf{T}} (M_{II}^{-\mathsf{T}})_{I \cap J} u = 0. \tag{3.9b}$$

Now the rows with indices in $I \cap J$ of these last two equations read

$$\alpha'_{I\cap J} - M^{\mathsf{T}}_{(I^c\cap J)(I\cap J)}\alpha_{I^c\cap J} + M^{\mathsf{T}}_{(I\cap J^c)(I\cap J)}\alpha_{I\cap J^c} - M^{\mathsf{T}}_{(I\cap J)(I\cap J)}(M^{-\mathsf{T}}_{II})_{I\cap J}u = 0,$$
$$\alpha''_{I\cap J} - M^{\mathsf{T}}_{(I\cap J^c)(I\cap J)}\alpha_{I\cap J^c} + M^{\mathsf{T}}_{(I^c\cap J)(I\cap J)}\alpha_{I^c\cap J} + M^{\mathsf{T}}_{(I\cap J)(I\cap J)}(M^{-\mathsf{T}}_{II})_{I\cap J}u = 0.$$

By adding these equations, we obtain $\alpha'_{I\cap J} + \alpha''_{I\cap J} = 0$, which implies that $\alpha'_{I\cap J} = \alpha''_{I\cap J} = 0$ since the components of these two vectors are nonnegative. One deduces then from any of these last equations that

$$(M^{-\mathsf{T}}_{II})_{I\cap J}u = M^{-\mathsf{T}}_{(I\cap J)(I\cap J)}\left(M^{\mathsf{T}}_{(I\cap J^c)(I\cap J)}\alpha_{I\cap J^c} - M^{\mathsf{T}}_{(I^c\cap J)(I\cap J)}\alpha_{I^c\cap J}\right). \tag{3.10}$$

Now the $I \cap J^c$ component of equation (3.9a) and the $I^c \cap J$ component of equation (3.9b) read

$$-M^{\mathsf{T}}_{(I^c\cap J)(I\cap J^c)}\alpha_{I^c\cap J} + M^{\mathsf{T}}_{(I\cap J^c)(I\cap J^c)}\alpha_{I\cap J^c}$$
$$\quad - M^{\mathsf{T}}_{(I\cap J)(I\cap J^c)}M^{-\mathsf{T}}_{(I\cap J)(I\cap J)}\left(M^{\mathsf{T}}_{(I\cap J^c)(I\cap J)}\alpha_{I\cap J^c} - M^{\mathsf{T}}_{(I^c\cap J)(I\cap J)}\alpha_{I^c\cap J}\right)$$
$$\quad = \beta_{I\cap J^c}, \tag{3.11a}$$
$$-M^{\mathsf{T}}_{(I\cap J^c)(I^c\cap J)}\alpha_{I\cap J^c} + M^{\mathsf{T}}_{(I^c\cap J)(I^c\cap J)}\alpha_{I^c\cap J}$$
$$\quad + M^{\mathsf{T}}_{(I\cap J)(I^c\cap J)}M^{-\mathsf{T}}_{(I\cap J)(I\cap J)}\left(M^{\mathsf{T}}_{(I\cap J^c)(I\cap J)}\alpha_{I\cap J^c} - M^{\mathsf{T}}_{(I^c\cap J)(I\cap J)}\alpha_{I^c\cap J}\right)$$
$$\quad = \beta_{I^c\cap J}. \tag{3.11b}$$

We have deduced the system (3.11) and $\alpha'_{I\cap J} = \alpha''_{I\cap J} = 0$ from (3.9). Reciprocally, to show that there has been no loss of information in that operation, let us show that one can recover the system (3.9) from (3.11) and $\alpha'_{I\cap J} = \alpha''_{I\cap J} = 0$, provided we define u and v by (3.7). Indeed, let us denote by w the right hand side of (3.10) and let us first show that the identity (3.10) is verified:

$$(M^{-\mathsf{T}}_{II})_{I\cap J}u = (M^{-\mathsf{T}}_{II})_{I\cap J}\begin{pmatrix} -\beta_{I\cap J^c} - M^{\mathsf{T}}_{(I^c\cap J)(I\cap J^c)}\alpha_{I^c\cap J} \\ -M^{\mathsf{T}}_{(I^c\cap J)(I\cap J)}\alpha_{I^c\cap J} \end{pmatrix} \quad [(3.7), \alpha'_{I\cap J} = 0]$$
$$= (M^{-\mathsf{T}}_{II})_{I\cap J}\begin{pmatrix} -M^{\mathsf{T}}_{(I\cap J^c)(I\cap J^c)}\alpha_{I\cap J^c} + M^{\mathsf{T}}_{(I\cap J)(I\cap J^c)}w \\ -M^{\mathsf{T}}_{(I\cap J^c)(I\cap J)}\alpha_{I\cap J^c} + M^{\mathsf{T}}_{(I\cap J)(I\cap J)}w \end{pmatrix} \quad \begin{bmatrix} (3.11a) \text{ and} \\ \text{definition of } w \end{bmatrix}$$
$$= (M^{-\mathsf{T}}_{II})_{I\cap J}M^{\mathsf{T}}_{II}\begin{pmatrix} -\alpha_{I\cap J^c} \\ w \end{pmatrix}$$
$$= w.$$

Now the $I \cap J^c$ component of (3.9a) is a consequence of (3.11a); the $I \cap J$ components of (3.9a) and (3.9b) are consequences of $\alpha'_{I\cap J} = \alpha''_{I\cap J} = 0$ and (3.10); and the $I^c \cap J$ component of (3.9b) is a consequence of (3.11b).

Therefore, getting rid of the vector $\beta \in \mathbb{R}^{|I \triangle J|}_+$ in (3.11), we see that there is a q such that the Newton-min algorithm makes the cycle $x^{(I)} \to x^{(J)} \to x^{(I)}$ when it is used to solve LCP(M, q) if and only if one cannot find an $\alpha \in \mathbb{R}^{|I \triangle J|}_+ \setminus \{0\}$ such that

$$\begin{pmatrix} M_{(I\cap J^c)(I\cap J^c)} & -M_{(I\cap J^c)(I^c\cap J)} \\ -M_{(I^c\cap J)(I\cap J^c)} & M_{(I^c\cap J)(I^c\cap J)} \end{pmatrix}^{\mathsf{T}}\alpha$$
$$\geqslant \begin{pmatrix} -M_{(I\cap J)(I\cap J^c)} & M_{(I\cap J)(I^c\cap J)} \end{pmatrix}^{\mathsf{T}} M^{-\mathsf{T}}_{(I\cap J)(I\cap J)}\begin{pmatrix} -M_{(I\cap J^c)(I\cap J)} \\ M_{(I^c\cap J)(I\cap J)} \end{pmatrix}^{\mathsf{T}}\alpha.$$

The equivalence $(i) \Leftrightarrow (ii)$ follows. $\qquad\square$

Observe that, as expected, condition (3.3) is symmetric in I and J, in the sense that the permutation $I \leftrightarrow J$ does not modify the condition.

Proposition 3.2 is apparently difficult to use if the goal is to characterize the class $\mathbf{NM_2}$ of matrices. It requires indeed to consider all the possible pairs of distinct subsets I and $J \subseteq [\![1, n]\!]$. In addition, for each of these pairs, each choice of $\alpha \in \mathbb{R}_+^{|I \triangle J|} \setminus \{0\}$ in (3.3) may yield different conditions on M that make the Newton-min algorithm avoid the 2-cycle $x^{(I)} \to x^{(J)} \to x^{(I)}$. This looks like a long-term and hazardous task. We show in theorem 4.2 below, however, that it is equivalent to avoid all the 2-cycles or to avoid the small subset of 2-cycles $x^{(I)} \to x^{(J)} \to x^{(I)}$, in which $I = J \cup \{i\}$ with $i \notin J$. In other words, this small subset of 2-cycles contains all the information on M that is needed to prevent the Newton-min algorithm from making 2-cycles. A precious interest of the choice $I = J \cup \{i\}$ is that α is then a positive *scalar*, which can be eliminated from the inequality (3.3). The task of characterizing the matrices M in $\mathbf{NM_2}$ is then much easier. It is shown in the next section that these matrices are the **P**-matrices.

4   A characterization of P-matricity

Let us start by an elementary lemma.

> **Lemma 4.1** *Suppose that I and J are two index sets included in $[\![1, n]\!]$ such that $J \setminus I \neq \varnothing$ and that $x^{(I)} = x^{(J)}$ for some matrix $M \in \mathbf{D^c}$ and some vector q. Then the Newton-min algorithm (2.1)-(2.2) does not make the null displacement $x^{(I)} \to x^{(J)}$ when it is used to solve* LCP(M, q).

PROOF. We argue by contradiction, assuming that the Newton-min algorithm goes from $x^{(I)}$ to $x^{(J)}$. Then, $(Mx^{(I)} + q)_{I^c \cap J} < x_{I^c \cap J}^{(I)}$ [by the algorithm rule (2.1)] and $x_{I^c \cap J}^{(I)} = 0$ [by the definition of $x^{(I)}$, see (2.3)$_1$], so that

$$(Mx^{(I)} + q)_{I^c \cap J} < 0.$$

On the other hand,

$$(Mx^{(J)} + q)_{I^c \cap J} = 0,$$

by the definition of $x^{(J)}$, see (2.3)$_2$. Therefore $(Mx^{(I)} + q)_{I^c \cap J} \neq (Mx^{(J)} + q)_{I^c \cap J}$, contradicting the fact that $x^{(I)} = x^{(J)}$ for the given q. $\qquad\square$

Here is a comment on this lemma. It is known that if $x^{(I)}$ is a solution to LCP(M, q) and if M is nondegenerate, then $x^{(J)} = x^{(I)}$, where $x^{(J)}$ is the iterate following $x^{(I)}$ (see lemma 4.1 in [2] and the references thereof). It is not difficult to see, however, that then J is included in I, so that $J \setminus I = \varnothing$ and lemma 4.1 does not apply.

We denote by $\text{cof}(M)$ the *cofactor matrix* of a matrix $M \in \mathbb{R}^{n \times n}$, whose element $[\text{cof}(M)]_{ij}$ is the *cofactor* $\text{cof}(M_{ij})$ of the element M_{ij} of M, that is

$$\text{cof}(M_{ij}) := (-1)^{i+j} \det M_{([\![1, n]\!] \setminus \{i\})([\![1, n]\!] \setminus \{j\})}. \tag{4.1}$$

We use the notation $\text{cof}_{II}(M_{ij})$ for the cofactor of the element M_{ij} in M_{II}. Recall [19; 1987, chapter VI] that for any index i and j:

$$\det M = \sum_{i'} M_{i'j} \text{cof}(M_{i'j}) = \sum_{j'} M_{ij'} \text{cof}(M_{ij'}) \tag{4.2}$$

and that

$$M^{-1} = (\det M)^{-1} \text{cof}(M^{\mathsf{T}}). \tag{4.3}$$

Our main result is given in theorem 4.2 below. The implication $(i) \Rightarrow (ii)$ of the theorem was already proven in [2; 2009, lemma 4.3], but that part of the paper was not selected by the refereeing process for appearing in the published version of the paper [3; 2012].

Theorem 4.2 (a characterization of P-matricity) *Suppose that $M \in \mathbf{D^c}$. Then the following conditions are equivalent:*

  (i) $M \in \mathbf{P}$,
 (ii) *for any q, the Newton-min algorithm does not cycle between two different nodes when it is used to solve* $\mathrm{LCP}(M, q)$,
(iii) *for any q, for any subset $J \subseteq [\![1, n]\!]$, and for any index $i \in [\![1, n]\!] \setminus J$, the Newton-min algorithm does not cycle between the nodes $x^{(J)}$ and $x^{(J \cup \{i\})}$ when it is used to solve* $\mathrm{LCP}(M, q)$,
 (iv) *for any subset $J \subseteq [\![1, n]\!]$ and any index $i \in [\![1, n]\!] \setminus J$, there holds*

$$M_{ii} \geqslant M_{\{i\}J} M_{JJ}^{-1} M_{J\{i\}}, \tag{4.4}$$

where the right hand side is zero when $J = \varnothing$.

PROOF. $[(i) \Rightarrow (ii)]$ We prove the contrapositive, assuming that the algorithm visits in order the following nodes $x^{(I)} \to x^{(J)} \to x^{(I)}$, for some I and $J \subseteq [\![1, n]\!]$ and some $q \in \mathbb{R}^n$ such that $x^{(I)} \neq x^{(J)}$. We simplify the notation by setting $x^1 := x^{(I)}$ and $x^2 := x^{(J)}$. Since the Newton-min algorithm goes from x^1 to x^2 and from x^2 to x^1, the very definition (2.1)-(2.2) of the algorithm implies that

$$x^1_{J^c} \leqslant (Mx^1 + q)_{J^c} \qquad \text{and} \qquad x^1_J > (Mx^1 + q)_J, \tag{4.5}$$
$$x^2_{I^c} \leqslant (Mx^2 + q)_{I^c} \qquad \text{and} \qquad x^2_I > (Mx^2 + q)_I. \tag{4.6}$$

After a possible rearrangement of the component order, we get

$$x^2 - x^1 = \begin{pmatrix} 0_{I \cap J^c} \\ x^2_{I \cap J} \\ x^2_{I^c \cap J} \\ 0_{I^c \cap J^c} \end{pmatrix} - \begin{pmatrix} x^1_{I \cap J^c} \\ x^1_{I \cap J} \\ 0_{I^c \cap J} \\ 0_{I^c \cap J^c} \end{pmatrix} = \begin{pmatrix} -x^1_{I \cap J^c} \\ (x^2 - x^1)_{I \cap J} \\ x^2_{I^c \cap J} \\ 0_{I^c \cap J^c} \end{pmatrix} . \quad \begin{matrix} [+] \\ [?] \\ [-] \\ [0] \end{matrix}$$

The extra column on the right gives the sign of each component, when appropriate: the components of $x^2 - x^1$ with indices in $I \cap J^c$ are nonnegative since $-x^1_{I \cap J^c} \geqslant -(Mx^1 + q)_{I \cap J^c}$ [by $(4.5)_1$] $= 0$ [by $(2.2)_2$] and the components of $x^2 - x^1$ with indices in $I^c \cap J$ are nonpositive since $x^2_{I^c \cap J} \leqslant (Mx^2 + q)_{I^c \cap J}$ [by $(4.6)_1$] $= 0$ [by $(2.2)_2$]. On the other hand, by $(2.2)_2$, there holds

$$M(x^2 - x^1) = \begin{pmatrix} (Mx^2)_{I \cap J^c} \\ -q_{I \cap J} \\ -q_{I^c \cap J} \\ (Mx^2)_{I^c \cap J^c} \end{pmatrix} - \begin{pmatrix} -q_{I \cap J^c} \\ -q_{I \cap J} \\ (Mx^1)_{I^c \cap J} \\ (Mx^1)_{I^c \cap J^c} \end{pmatrix} = \begin{pmatrix} (Mx^2 + q)_{I \cap J^c} \\ 0 \\ -(Mx^1 + q)_{I^c \cap J} \\ (M(x^2 - x^1))_{I^c \cap J^c} \end{pmatrix} . \quad \begin{matrix} [-] \\ [0] \\ [+] \\ [?] \end{matrix}$$

The extra column on the right gives the sign of each component, when appropriate: the components of $M(x^2 - x^1)$ with indices in $I \cap J^c$ are nonpositive since $(Mx^2 + q)_{I \cap J^c} \leqslant x^2_{I \cap J^c}$ [by $(4.6)_2$] $= 0$ [by $(2.2)_1$] and the components of $M(x^2 - x^1)$ with indices in $I^c \cap J$ are nonnegative since $-(Mx^1 + q)_{I^c \cap J} \geqslant -x^1_{I^c \cap J}$ [by $(4.5)_2$] $= 0$ [by $(2.2)_1$]. Therefore

$$(x^2 - x^1) \cdot M(x^2 - x^1) \leqslant 0.$$

Since $x^1 \neq x^2$, M cannot be a **P**-matrix (see (1.1)).

$[(ii) \Rightarrow (iii)]$ Let $q \in \mathbb{R}^n$, $J \subseteq [\![1,n]\!]$, and $i \in [\![1,n]\!] \setminus J$. Set $I := J \cup \{i\}$. If $x^{(I)} \neq x^{(J)}$, then (iii) is a clear consequence of (ii). If $x^{(I)} = x^{(J)}$, then (iii) is a consequence of lemma 4.1, according to which the Newton min does not go from $x^{(J)}$ to $x^{(I)}$ because $I \setminus J \neq \varnothing$.

$[(iii) \Rightarrow (iv)]$ Let J and i be like in (iv), and set $I = J \cup \{i\}$. By (iii), whatever is q, the Newton-min algorithm does not cycle between the nodes $x^{(I)}$ and $x^{(J)}$ when it is used to solve LCP(M,q). Then, the implication $(ii) \Rightarrow (i)$ of proposition 3.2 shows that there is a scalar $\alpha > 0$ such that (3.3) holds. Since $I \cap J^c = \{i\}$, $I^c \cap J = \varnothing$, $I \cap J = J$, $I \triangle J = \{i\}$, and α is a positive scalar that can be eliminated from (3.3), this inequality simplifies in (4.4) (use also the fact that $M_{\{i\}J} M_{JJ}^{-1} M_{J\{i\}}$ is a scalar, hence equal to its transpose). In case $J = \varnothing$, the condition (i) of proposition 3.2 indicates that the inequality (4.4) becomes $M_{ii} \geqslant 0$.

$[(iv) \Rightarrow (i)]$ We prove by induction that $\det M_{II} > 0$ for any $I \subseteq [\![1,n]\!]$, which is equivalent to $M \in \mathbf{P}$. By applying (iv) with $J = \varnothing$, we obtain $M_{ii} > 0$ for a nondegenerate matrix, so that $\det M_{II} > 0$ when $|I| = 1$. Now, assume that J and i are chosen like in (iv), that $I = J \cup \{i\}$, that $\det M_{JJ} > 0$ (induction assumption), and let us show that $\det M_{II} > 0$, which will conclude the proof of $(iv) \Rightarrow (i)$.

Let us denote the indices in J by j_k, $k \in [\![1,|J|]\!]$. Using the cofactor matrix of M_{JJ} in (4.4) and the induction assumption $\det M_{JJ} > 0$, one gets

$$
\begin{aligned}
0 \;\leqslant\;& M_{ii} \det M_{JJ} - M_{\{i\}J} \operatorname{cof}(M_{JJ}^{\mathsf{T}}) M_{J\{i\}} \qquad [(4.4),\ (4.3),\ \det M_{JJ} > 0] \\
=\;& M_{ii} \det M_{JJ} - \sum_{k=1}^{|J|} \sum_{l=1}^{|J|} M_{ij_k} \operatorname{cof}_{JJ}([M_{JJ}]_{lk}) M_{j_l i} \\
=\;& M_{ii} \det M_{JJ} - \sum_{k=1}^{|J|} \sum_{l=1}^{|J|} M_{ij_k} (-1)^{l+k} \det M_{(J\setminus\{j_l\})(J\setminus\{j_k\})} M_{j_l i} \qquad [(4.1)] \\
=\;& M_{ii} \det M_{JJ} + \sum_{k=1}^{|J|} (-1)^{k+|J|+1} M_{ij_k} \sum_{l=1}^{|J|} M_{j_l i} (-1)^{l+|J|} \det M_{(J\setminus\{j_l\})(J\setminus\{j_k\})} \\
=\;& M_{ii} \det M_{JJ} + \sum_{k=1}^{|J|} (-1)^{k+|J|+1} M_{ij_k} \det \big(M_{J(J\setminus\{j_k\})},\, M_{J\{i\}} \big) \qquad [(4.2)] \\
=\;& \det M_{II} \qquad [(4.2)].
\end{aligned}
$$

Therefore $\det M_{II} > 0$ by the nondegeneracy of M. □

Even though the following consequence of theorem 4.2 is clear and was already summarized by formula (2.5) in the introduction, we quote it in a corollary to make easier a possible future citation.

It is clear that **NM** is included in the set $\mathbf{D^c} \cap \mathbf{Q}$ of nondegenerate matrices ensuring that LCP(M,q) has a solution, whatever is q; indeed, if $M \in \mathbf{D^c} \setminus \mathbf{Q}$, one can find a vector q such that LCP(M,q) has no solution, in which case, the Newton-min algorithm has no other choice than cycling (we recall from lemma 4.1 in [2] that, when $M \in \mathbf{D^c}$, the sequence generated by the Newton-min algorithm can be stationnary only at a solution). The stronger inclusion (4.7), however, was not clear to us, before theorem 4.2 was establised.

Corollary 4.3 (NM is included in P) *The set of matrices $M \in \mathbf{D^c}$ ensuring the convergence of the Newton-min algorithm when it is used to solve* LCP(M,q)*, whatever are the*

vector q and the initial point, is included in **P***. More compactly*

$$\mathbf{NM} \subseteq \mathbf{P}. \tag{4.7}$$

PROOF. Observe first that **NM** is indeed the set of matrices $M \in \mathbf{D^c}$ ensuring the convergence of the Newton-min algorithm when it is used to solve $\text{LCP}(M, q)$, whatever is the vector q (see the discussion before formula (2.4)). Next, $\mathbf{NM} \subseteq \mathbf{NM_2}$ by the definition (2.4) of **NM** and $\mathbf{NM_2} = \mathbf{P}$ by theorem 4.2. □

The implication $(ii) \Rightarrow (i)$ of theorem 4.2, according to which only the **P**-matrices in $\mathbf{D^c}$ prevent the Newton-min algorithm from cycling between two nodes, is ultimately based on Motzkin's theorem of the alternative, which supports the implication $(iii) \Rightarrow (iv)$ of theorem 4.2, while the implication $(ii) \Rightarrow (iii)$ is straightforward and the implication $(iv) \Rightarrow (i)$ has an algebraic nature. Motzkin's theorem is not constructive whereas, for $M \in \mathbf{D^c} \setminus \mathbf{P}$, it would certainly be interesting (and reassuring) to be able to construct a q and to select two nodes such that the Newton-min algorithm cycles between these nodes when it is used to solve $\text{LCP}(M, q)$. This is the goal of the next proposition, which takes inspiration from the contrapositive of the implications $(iii) \Rightarrow (iv) \Rightarrow (i)$ of theorem 4.2: if $M \in \mathbf{D^c} \setminus \mathbf{P}$, there are a vector $q \in \mathbb{R}^n$, a subset $J \subseteq [\![1, n]\!]$, and an index $i \in [\![1, n]\!] \setminus J$ such that the Newton-min algorithm cycles between the nodes $x^{(J)}$ and $x^{(J \cup \{i\})}$. According to the contrapositive of the implication $(iv) \Rightarrow (i)$, when $M \in \mathbf{D^c} \setminus \mathbf{P}$, there are index sets J and $I := J \cup \{i\}$ such that $\det M_{JJ} > 0$ and $\det M_{II} < 0$; this provides a means to select the index sets. One still has to find the vector q and this is precisely what the next proposition brings by exhibiting a whole family of vectors q such that the cycle $x^{(J)} \to x^{(I)} \to x^{(J)}$, with the above J and I, occurs.

We adopt the notation $t^+ := \max(0, t)$ for $t \in \mathbb{R}$. The "max" operator is supposed to act componentwise on vectors.

Proposition 4.4 (2-cycle for $M \notin \mathbf{P}$) *Suppose that $M \in \mathbf{D^c} \setminus \mathbf{P}$. Then*

1) *there are two index sets I and $J \subseteq [\![1, n]\!]$ and an index $i \in [\![1, n]\!]$ such that $I = J \cup \{i\}$, $\det M_{II} < 0$, and $\det M_{JJ} > 0$,*

2) *for any two index sets I and $J \subseteq [\![1, n]\!]$ and an index $i \in [\![1, n]\!]$ having the properties given in point 1, the Newton-min algorithm cycles between $x^{(I)}$ and $x^{(J)}$ when the components of q are determined in order as follows*

$$q_J = -M_{JJ}e^J, \tag{4.8}$$

$$q_i = -M_{iJ}e^J - \varepsilon, \quad \text{with } 0 < \varepsilon < \frac{|\det M_{II}|}{\max_{j \in J}[\text{cof}_{II}(M_{ij})]^+}, \tag{4.9}$$

$$q_{I^c} \geqslant \max \left(M_{I^c J} M_{JJ}^{-1} q_J, M_{I^c I} M_{II}^{-1} q_I \right), \tag{4.10}$$

where e^J is the vector of all ones in $\mathbb{R}^{|J|}$.

PROOF. 1) Since $M \notin \mathbf{P}$, some principal minor of M is negative. Then one can choose an index set I with the smallest cardinal number $|I|$ such that $\det M_{II} < 0$. Since $|I| \geqslant 1$, one can choose an index $i \in I$ and set $J := I \setminus \{i\}$ (J may be empty). The properties in point 1 are verified for the selected sets I and J, and the index i (recall that $\det M_{\varnothing\varnothing} = 1$ by convention).

2) Let the index sets I and J and the index i be such that $I = J \cup \{i\}$, $\det M_{II} < 0$, and

$\det M_{JJ} > 0$. Suppose that q satisfies (4.8)-(4.10). Let $x^1 := x^{(J)}$, so that

$$\begin{cases} x_J^1 = -M_{JJ}^{-1} q_J \\ x_{J^c}^1 = 0 \end{cases} \qquad \text{and} \qquad \begin{cases} (Mx^1 + q)_J = 0 \\ (Mx^1 + q)_{J^c} = q_{J^c} - M_{J^c J} M_{JJ}^{-1} q_J. \end{cases}$$

For a q satisfying the assumption, there hold

$$-M_{JJ}^{-1} q_J = e^J > 0 \qquad [(4.8)],$$
$$q_i - M_{iJ} M_{JJ}^{-1} q_J = -\varepsilon < 0 \qquad [(4.8) \text{ and } (4.9)],$$
$$q_{I^c} - M_{I^c J} M_{JJ}^{-1} q_J \geqslant 0 \qquad [(4.10)].$$

These inequalities imply that the next iterate visited by the Newton-min algorithm (2.1)-(2.2) is $x^2 := x^{(I)}$, so that

$$\begin{cases} x_I^2 = -M_{II}^{-1} q_I \\ x_{I^c}^2 = 0 \end{cases} \qquad \text{and} \qquad \begin{cases} (Mx^2 + q)_I = 0 \\ (Mx^2 + q)_{I^c} = q_{I^c} - M_{I^c I} M_{II}^{-1} q_I. \end{cases}$$

We now want to show that appropriate inequalities on q are verified that ensure that the iterate following x^2 is x^1. Observe that by (4.8), (4.9), and (4.3)

$$M_{II}^{-1} q_I = -M_{II}^{-1} M_{IJ} e^J - \varepsilon\, M_{II}^{-1} e^{I,i} = -(e^I - e^{I,i}) - \varepsilon\, (\det M_{II})^{-1} \operatorname{cof}_{II}(M_{iI})^\mathsf{T},$$

where $e^{I,i} \in \mathbb{R}^{|I|}$ is a vector whose components are all zero, except the one at the position of i in I whose value is 1. Therefore

$$\forall\, j \in J: \qquad (-M_{II}^{-1} q_I)_j = 1 + \varepsilon\, (\det M_{II})^{-1} \operatorname{cof}_{II}(M_{ij}) > 0, \tag{4.11}$$

since $-\varepsilon\, (\det M_{II})^{-1} \operatorname{cof}_{II}(M_{ij}) \leqslant \varepsilon\, |\det M_{II}|^{-1} \max_{j \in J} [\operatorname{cof}_{II}(M_{ij})]^+ < 1$, by the choice of ε in (4.9). On the other hand,

$$(-M_{II}^{-1} q_I)_i = \varepsilon\, (\det M_{II})^{-1} \operatorname{cof}_{II}(M_{ii}) = \varepsilon\, (\det M_{II})^{-1} (\det M_{JJ}) < 0, \tag{4.12}$$

since $(\det M_{II}) < 0$ and $(\det M_{JJ}) > 0$ by assumption. Finally, by (4.10), there holds

$$q_{I^c} - M_{I^c I} M_{II}^{-1} q_I \geqslant 0. \tag{4.13}$$

The inequalities (4.11), (4.12), and (4.13) imply that the iterate following x^2 is indeed x^1. Hence the Newton-min algorithm cycles between the nodes $x^{(J)}$ and $x^{(I)}$. $\qquad\square$

Thanks to proposition 4.4, the equivalence $(i) \Leftrightarrow (ii)$ of proposition 4.2 can now be proven without proposition 3.2 and Motzkin's theorem of the alternative, so that one can wonder whether section 3 is still useful. We have maintained it in the paper for two reasons. First, proposition 3.2 has brought a clear indication on the way of choosing the index sets I and J in proposition 4.4, whose origin could be obscure otherwise. Next, the dual characterization of the absence of 2-cycle that it provides may be useful in the extension of this work.

5   Discussion and perspectives

A natural extension of the present work would consist in looking at the possibility to give a *simple* algebraic description of the classes $\mathbf{NM}_k$, for $k \geqslant 3$, and $\mathbf{NM}$. A puzzling feature of the approach followed in this paper to characterize $\mathbf{NM}_2$ is its intrinsic conjunctive-disjunctive nature. It is conjunctive in the sense that it provides conditions to satisfy for each possible 2-cycle that the matrix must prevent. Its disjunctive aspect comes from the use of Motzkin's

theorem of the alternative (lemma 3.1), which provides a possibility to avoid a given cycle for each acceptable choice of multipliers α in proposition 3.2, and there may be many. For example, when $M \in \mathbf{P}$, the same approach shows that the Newton-min algorithm does not make the 3-cycle $x^{(\{i\})} \to x^{(\{j\})} \to x^{(\{k\})} \to x^{(\{i\})}$, for three different indices i, j, and $k \in [\![1, n]\!]$, when it is used to solve $\mathrm{LCP}(M, q)$, whatever is $q \in \mathbb{R}^n$, if and only if *one* of the following eight conditions holds

$$
\begin{aligned}
& M_{ik}M_{jj} \leqslant M_{ij}^+ M_{jk}, \quad M_{ji}M_{kk} \leqslant M_{jk}^+ M_{ki}, \quad M_{kj}M_{ii} \leqslant M_{ki}^+ M_{ij}, \\
& M_{jj}^+ M_{ik} \leqslant M_{jk}M_{ii}, \quad M_{kj}^+ M_{ji} \leqslant M_{ki}M_{jj}, \quad M_{ik}^+ M_{kj} \leqslant M_{ij}M_{kk}, \\
& M_{ik}^+ M_{kj}^+ M_{ji}^+ \leqslant M_{ii}M_{jj}M_{kk}, \quad \text{or} \quad M_{ij}^+ M_{jk}^+ M_{ki}^+ > M_{ii}M_{jj}M_{kk}.
\end{aligned}
$$

Note that conditions 1, 2, 3, and 7 are satisfied by an **M**-matrix. We don't know whether this disjunctive form of the conditions disappears for the whole classes $\mathbf{NM}_k$ or $\mathbf{NM}$, i.e., when all the possible cycles must be avoided, as it does for $\mathbf{NM}_2 = \mathbf{P}$.

Another natural question is whether the membership to $\mathbf{NM}$ can be determined in polynomial time; we recall that the membership to $\mathbf{P}$ is a co-NP-complete problem [8, 26; 1994-2000]. It would also be important to know whether the Newton-min algorithm solves the linear complementarity problem $\mathrm{LCP}(M, q)$ in polynomial time when $M \in \mathbf{NM}$; recall that it does when $M \in \mathbf{M}$ [16; 2004].

Acknowledgments

References

[1] M. Aganagić (1984). Newton's method for linear complementarity problems. *Mathematical Programming*, 28, 349–362. [doi]. 3, 4, 5

[2] I. Ben Gharbia, J.Ch. Gilbert (2009). Nonconvergence of the plain Newton-min algorithm for linear complementarity problems with a P-matrix – The full report. Rapport de Recherche 7160, INRIA, BP 105, 78153 Le Chesnay, France. [preprint]. 5, 9, 10, 11

[3] I. Ben Gharbia, J.Ch. Gilbert (2012). Nonconvergence of the plain Newton-min algorithm for linear complementarity problems with a P-matrix. *Mathematical Programming*, 134, 349–364. [doi]. 4, 5, 10

[4] M. Bergounioux, M. Haddou, M. Hintermüller, K. Kunisch (2000). A comparison of a Moreau-Yosida-based active set strategy and interior point methods for constrained optimal control problems. *SIAM Journal on Optimization*, 11, 495–521. [doi]. 4

[5] M. Bergounioux, K. Ito, K. Kunisch (1999). Primal-dual strategy for constrained optimal control problems. *SIAM Journal on Control and Optimization*, 37, 1176–1194. [doi]. 4

[6] R. Chandrasekaran (1970). A special case of the complementary pivot problem. *Opsearch*, 7, 263–268. 4

[7] R.W. Cottle, J.-S. Pang, R.E. Stone (2009). *The Linear Complementarity Problem*. Classics in Applied Mathematics 60. SIAM, Philadelphia, PA, USA. 3

[8] G.E. Coxson (1994). The P-matrix problem is co-NP-complete. *Mathematical Programming*, 64, 173–178. [doi]. 14

[9] M. Fiedler, V. Pták (1962). On matrices with nonpositive off-diagonal elements and principal minors. *Czechoslovak Mathematics Journal*, 12, 382–400. 3

[10] A. Fischer, C. Kanzow (1996). On finite termination of an iterative method for linear complementarity problems. *Mathematical Programming*, 74, 279–292. [doi]. 4

[11] O. Güler (2010). *Foundations of Optimization*. Graduate Texts in Mathematics 258. Springer. [doi]. 5

[12] P.T. Harker, J.-S. Pang (1990). A damped-Newton method for the linear complementarity problem. In E.L. Allgower, K. Georg (editors), *Computational Solution of Nonlinear Systems of Equations*, Lecture in Applied Mathematics 26. AMS, Providence, RI. 4

[13] M. Hintermüller, K. Ito, K. Kunisch (2003). The primal-dual active set strategy as a semismooth Newton method. *SIAM Journal on Optimization*, 13, 865–888. [doi]. 4, 5

[14] R.A. Horn, Ch.R. Jonhson (1991). *Topics in Matrix Analysis*. Cambridge University Press, New York, NY, USA. 3

[15] K. Ito, K. Kunisch (2008). *Lagrange Multiplier Approach to Variational Problems and Applications*. Advances in Design and Control. SIAM Publication, Philadelphia. [doi]. 4

[16] Ch. Kanzow (2004). Inexact semismooth Newton methods for large-scale complementarity problems. *Optimization Methods and Software*, 19, 309–325. [doi]. 4, 14

[17] M. Kojima, S. Shindo (1986). Extension of Newton and quasi-Newton methods to systems of PC^1 equations. *Journal of Operations Research Society of Japan*, 29, 352–375. 4

[18] M.M. Kostreva (1976). *Direct algorithms for complementarity problems*. PhD Thesis, Rensselaer Polytechnic Institute, Troy, New York. 4

[19] S. Lang (1987). *Linear algebra*. Undergraduate Texts in Mathematics. Springer. (third edition). 9

[20] O.L. Mangasarian (1977). Solution of symmetric linear complementarity problems by iterative methods. *Journal of Optimization Theory and Applications*, 22, 465–485. [doi]. 4

[21] R. Mifflin (1977). Semismooth and semiconvex functions in constrained optimization. *SIAM Journal on Control and Optimization*, 15, 959–972. [doi]. 4

[22] L. Qi, J. Sun (1993). A nonsmooth version of Newton's method. *Mathematical Programming*, 58, 353–367. [doi]. 4

[23] J. Rohn (2012). On Rump's characterization of *P*-matrices. *Optimization Letters*, 6, 1017–1020. [doi]. 3

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

[25] H. Samelson, R.M. Thrall, O. Wesler (1958). A partition theorem for the Euclidean *n*-space. *Proceedings of the American Mathematical Society*, 9, 805–807. [editor]. 3

[26] P. Tseng (2000). Co-NP-completeness of some matrix classification problems. *Mathematical Programming*, 88, 183–192. 14

[27] M. Ulbrich (2011). *Semismooth Newton Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces*. MPS-SIAM Series on Optimization 11. SIAM Publications, Philadelphia, PA, USA. [doi]. 4

Index