Complexity of maximum fixed point problem in Boolean Networks

Florian Bridoux[1], Nicolas Durbec[1], Kevin Perrot[1], and Adrien Richard[2,3]

[1] Aix-Marseille Université, Université de Toulon, CNRS, LIS, Marseille, France
[2] Laboratoire I3S, CNRS, Université Côte d'Azur, France
[3] CMM, UMI CNRS 2807, Universidad de Chile, Chile

Abstract. A *Boolean network* (BN) with n components is a discrete dynamical system described by the successive iterations of a function $f : \{0,1\}^n \to \{0,1\}^n$. This model finds applications in biology, where fixed points play a central role. For example in genetic regulation they correspond to cell phenotypes. In this context, experiments reveal the existence of positive or negative influences among components: component i has a positive (resp. negative) influence on component j, meaning that j tends to mimic (resp. negate) i. The digraph of influences is called *signed interaction digraph* (SID), and one SID may correspond to multiple BNs. The present work opens a new perspective on the well-established study of fixed points in BNs. Biologists discover the SID of a BN they do not know, and may ask: given that SID, can it correspond to a BN having at least k fixed points? Depending on the input, this problem is in P or complete for NP, $\mathsf{NP}^{\#\mathsf{P}}$ or $\mathsf{NEXPTIME}$.

Keywords: Complexity · Boolean networks · Fixed points · Interaction graph

1 Introduction

A *Boolean network* (BN) with n components is a discrete dynamical system described by the successive iterations of a function

$$f : \{0,1\}^n \to \{0,1\}^n, \qquad x = (x_1, \dots, x_n) \mapsto f(x) = (f_1(x), \dots, f_n(x)).$$

The structure of the network is often described by a signed digraph D, called *signed interaction digraph* (SID) of f, catching effective positive and negative dependencies among components: the vertex set is $[n] := \{1, \dots, n\}$ and, for all $i, j \in [n]$, there is a positive (resp. negative) arc from i to j if $f_j(x) - f_j(y)$ is positive (resp. negative) for some $x, y \in \{0,1\}^n$ that only differ in $x_i > y_i$. The SID provides a very rough information about f. Hence, given a SID D, the set $F(D)$ of BNs f whose SID is D, is generally huge.

BNs have many applications. In particular, since the seminal papers of Kauffman [14, 15] and Thomas [30, 31], they are very classical models for the dynamics of gene networks. In this context, the first reliable experimental information often concern the SID of the network, while the actual dynamics are very difficult to observe [32, 18]. One is thus faced with the following question: *What can be said about the dynamics described by f according to D only?*

Among the many dynamical properties that can be studied, fixed points are of special interest, since they correspond to stable patterns of gene expression at the basis of particular cellular phenotypes [31, 3]. As such, they are arguably the property which has been the most thoroughly studied. The number of fixed points and its maximization in particular is the subject of a stream of work, *e.g.* in [26, 5, 24, 4, 12, 6, 11, 7].

From the complexity point of view, previous works essentially focused on decision problems of the following form: given f and a dynamical property P, what is the complexity of deciding if the dynamics described by f has the property P. For instance, it is well-known that deciding if f has a fixed point is NP-complete in general (see [17] and the references therein), and in P for some families of BNs, such as monotone or non-expansive BNs [13, 10]. However, as mentioned above, in practice, f is often unknown while its SID is well approximated. Hence, a much more natural question is: given a SID D and dynamical property P, what is the complexity of deciding if the dynamics described by some $f \in F(D)$ has the property P. Up to our knowledge, there is, perhaps surprisingly, no work concerning this kind of questions.

In this paper, we study this class of decision problems, focusing on the maximum number of fixed points. More precisely, given a SID D, we denote by $\phi(D)$ the maximum number of fixed points in a BN $f \in F(D)$, and we study the complexity of deciding if $\phi(D) \geq k$.

After the definitions in Section 2, we first study the problem when the positive integer k is fixed. We prove in Section 3 that, given a SID D, deciding if $\phi(D) \geq k$ is in P if $k = 1$. We also prove in Section 4 that the same problem is NP-complete if $k \geq 2$. Furthermore, these results remain true if the maximum in-degree $\Delta(D)$ is bounded by any constant $d \geq 2$. The case $k = 2$ is of particular interest since many works have been devoted to finding necessary conditions for the existence of multiple fixed points, both in the discrete and continuous settings, see [24, 25, 28, 16] and the references therein. Section 5 considers the case where k is part of the input. We prove that, given a SID D and a positive integer k, deciding if $\phi(D) \geq k$ is $\mathsf{NEXPTIME}$-complete, and becomes $\mathsf{NP}^{\#\mathsf{P}}$-complete if $\Delta(D)$ is bounded by a constant $d \geq 2$. Note that, from these results, we immediately obtain complexity results for the dual decision problem $\phi(D) < k$. A summary is given in Table 1.

In the case where k is fixed, while proving that the problem $\phi(D) \geq k$ belongs to NP, we study a decision problem of independent interest, called *extension* or *consistency* problem [9, 8, 2]. Here, the property P consists of a partial BN, that is, a function $h : X \to \{\mathtt{0}, \mathtt{1}\}^n$ where $X \subseteq \{\mathtt{0}, \mathtt{1}\}^n$. This partial BN may represent some experimental observations about the dynamics. Given a SID D, we prove that we can check in $\mathcal{O}(|X|^2 n^2)$ time if there is a BN $f \in F(D)$ which is consistent with h, that is, such that $f(x) = h(x)$ for all $x \in X$. Thus, the task consists in extending h to a global BN f under the constraint that the SID of f is D.

Table 1: Complexity results.

Problem	$\Delta(D) \leq d$	$k = 1$	$k \geq 2$	k given in input
$\phi(D) \geq k$	yes	P	NP-complete	$\mathsf{NP}^{\#\mathsf{P}}$-complete
	no			NEXPTIME-complete
$\phi(D) < k$	yes		coNP-complete	$\mathsf{coNP}^{\#\mathsf{P}}$-complete
	no			coNEXPTIME-complete

2 Definitions and Notations

Let V be a finite set. A *Boolean network* (BN) with component set V is defined as a function $f : \{0,1\}^V \to \{0,1\}^V$. A *configuration* $x \in \{0,1\}^V$ assigns a state $x_i \in \{0,1\}$ to each component $i \in V$. During an application of f, the state of component i evolves according to the *local* function $f_i : \{0,1\}^V \to \{0,1\}$, which is the coordinate i of f, *i.e.* $f_i(x) = f(x)_i$ for all $x \in \{0,1\}^V$. When $V = [n]$, we write $x = (x_1, \ldots, x_n)$ and $f(x) = (f_1(x), \ldots, f_n(x))$.

Given a configuration $x \in \{0,1\}^V$ and $I \subseteq V$, we denote by x_I the configuration $y \in \{0,1\}^I$ such that $y_i = x_i$ for all $i \in I$. Given $i \in V$, we denote the i-base vector e_i, that is, $(e_i)_i = 1$ and $(e_i)_j = 0$ for all $j \neq i$. If $x, y \in \{0,1\}^V$ then $x \oplus y$ is the configuration $z \in \{0,1\}^V$ such that $z_i = x_i \oplus y_i$ for all $i \in V$, where the addition is computed modulo two. Hence, $x \oplus e_i$ is the configuration obtained from x by flipping component i only.

A *signed digraph* $D = (V, A, \sigma)$ is a digraph (V, A) with an arc-labeling function σ from A to $\{-1, 0, 1\}$, that gives a sign (negative, null or positive) to each arc (i, j), denoted σ_{ij}. We say that D is *simple* if it has no null sign. Given a vertex i and $s \in \{-1, 0, 1\}$, we denote by $N^s_D(i)$ the set of in-neighbors j of i such that $\sigma_{ij} = s$, and we drop D in the notations when it is clear from the context. We call $N^1(i)$ (resp. $N^{-1}(i)$) the set of positive (resp. negative) in-neighbors of i. We also simply denote $N(i)$ the set of all in-neighbors of i. In the following, it is very convenient to set $\tilde{\sigma}_{ij} = 0$ if $\sigma_{ij} \geq 0$ and $\tilde{\sigma}_{ij} = 1$ otherwise.

The *signed interaction digraph* (SID) of a BN f with component set V is the signed digraph $D_f = (V, A, \sigma)$ defined as follows. First, given $i, j \in V$, there is an arc $(i, j) \in A$ if and only if there exists a configuration x such that $f_j(x \oplus e_i) \neq f_j(x)$ (*i.e.* the state of component i influences the state of component j). Second, the sign σ_{ij} of an arc $(i, j) \in A$ depends on whether the state of j tends to mimic or negate the state of i, and is defined as

$$\sigma_{ij} = \begin{cases} 1 & \text{if } f_j(x \oplus e_i) \geq f_j(x) \text{ for all } x \in \{0,1\}^n \text{ with } x_i = 0, \\ -1 & \text{if } f_j(x \oplus e_i) \leq f_j(x) \text{ for all } x \in \{0,1\}^n \text{ with } x_i = 0, \\ 0 & \text{otherwise.} \end{cases}$$

Given $j \in V$, we say that f_j is the AND (resp. OR) function if it is the ordinary logical and (resp. or) but inputs with a negative sign are flipped, *i.e*

$$f_j(x) = \bigwedge_{i \in N(j)} x_i \oplus \tilde{\sigma}_{ij} \qquad (\text{resp. } f_j(x) = \bigvee_{i \in N(j)} x_i \oplus \tilde{\sigma}_{ij}\ ).$$

Given a signed digraph D, we know that D is a SID (*i.e.* there exists a BN f with $D_f = D$), if and only if there is no vertex i such that $|N(i)| \leq 2$ and $|N^0(i)| = 1$ [23]. In particular, a simple signed digraph is always a SID.

A fundamental remark regarding the present work is that multiple BNs may have the same SID. Given a SID D with vertex set V, we denote by $F(D)$ the set of BNs admitting D as SID:

$$F(D) = \{f : \{\mathtt{0},\mathtt{1}\}^V \to \{\mathtt{0},\mathtt{1}\}^V \mid D_f = D\}.$$

The size of $F(D)$ is generally huge. If a component i has in-degree d in D, then the number of possible local functions f_i is doubly exponential according to d, thus it scales as the number of Boolean functions on d variables, 2^{2^d}. Hence, $|F(D)|$ is at least doubly exponential according to its maximum in-degree, denoted $\Delta(D)$. The precise value of $|F(D)|$ is not trivial, see `A006126` on the OEIS [1].

A *fixed point* of f is a configuration x such that $f(x) = x$, which is equivalent to $f_i(x) = x_i$ for all $i \in [n]$. We denote by $\Phi(f)$ the set of fixed points of f and $\phi(f) = |\Phi(f)|$. We are interested in a decision problem related to the maximum number of fixed points of BNs within $F(D)$, denoted

$$\phi(D) = \max\{\phi(f) \mid f \in F(D)\}.$$

More precisely, we will study the complexity of deciding if $\phi(D) \geq k$, where k is a positive integer, fixed or not. This gives the two following decision problems.

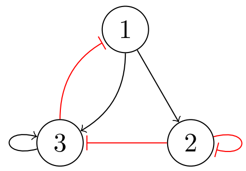
k-Maximum Fixed Point Problem (k-MFPP)
Input: a SID D.
Question: $\phi(D) \geq k$?

Maximum Fixed Point Problem (MFPP)
Input: a SID D and an integer $k \geq 1$.
Question: $\phi(D) \geq k$?

Cycles of interactions (in the SID) are known to play a fundamental role in the dynamical complexity of BN (the cycles we consider are always directed and without repeated vertices). Indeed, if D_f is acyclic then $\phi(f) = 1$ [26]. The *sign* of a cycle or a path in a signed digraph is the product of the signs of its arcs. It is well-known that if all the cycles of D_f are positive (resp. negative) then $\phi(f) \geq 1$ (resp. $\phi(f) \leq 1$), see [4, 25]. Hence, if all the cycles of a SID D are negative, then $\phi(D) \leq 1$. The previous notions are illustrated in Figure 1.

3 k-Maximum Fixed Point Problem for $k = 1$

A strongly connected component H in a signed digraph D is *trivial* if it has a unique vertex and no arc, and *initial* if D has no arc (i, j) where j is in H but not i. We first have a lemma to concentrate on simple signed digraphs.



$f_1(x) = \neg x_3$
$f_2(x) = x_1 \vee \neg x_2$
$f_3(x) = x_1 \vee (\neg x_2 \wedge x_3)$

$g_1(x) = \neg x_3$
$g_2(x) = x_1 \vee \neg x_2$
$g_3(x) = x_1 \wedge \neg x_2 \wedge x_3$

Fig. 1: Example of simple signed digraph D with two BNs $f, g \in F(D)$. BN f has no fixed point, and g has one fixed point ($\texttt{110}$), which is the maximum for BNs in $F(D)$, that is $\phi(D) = 1$. Note that D has two positive cycles and two negative cycles.

Lemma 1. *For any SID D, there is a simple SID D' such that $\phi(D) \geq 1 \iff \phi(D') \geq 1$, and D' is computable from D in constant parallel time.*

Proof. From D, the construction of D' is made component by component, independently, by removing incoming arcs. For $j \in [n]$,

– If $|N^0(j)| \geq 2$ then we delete all incoming arcs of j. If there exists $f \in F(D)$ and $y \in \Phi(f)$, then we can take $f' \in F(D')$ equal to f, except for $f'_j(x) = y_j$ (a constant). Conversely, if there exists $f' \in F(D')$ and $y \in \Phi(f')$, then we can take $f \in F(D)$ equal to f', except for

$$f_j(x) = (b_j \oplus \bigoplus_{i \in N^0(j)} x_i) \wedge \bigwedge_{i \in N(j) \setminus N^0(j)} (x_i \oplus \tilde{\sigma}_{ij})$$

with $b_j = \bigoplus_{i \in N^0(j)} y_i$, in the case $y_j = \texttt{0}$ (the case $y_j = \texttt{1}$ is symmetric, with OR instead of AND function). We have $f'_j(y) = f_j(y) = y_j$ hence $y \in \Phi(f)$.
– If $|N^0(j)| = 1$, then we delete this arc. One can check that, if $y \in \Phi(f)$ with $f \in F(D)$ (resp. $y \in \Phi(f')$ with $f' \in F(D')$), then there exists $i \in N(j) \setminus N^0(j)$ such that $y_i \oplus \tilde{\sigma}_{ij} = y_j$. Consequently, if there exists $f \in F(D)$ and $y \in \Phi(f)$ then we can take $f' \in F(D')$ equal to f, except that f'_j is the AND function if $y_j = 0$ and the OR function otherwise. Conversely, suppose there exists $f' \in F(D')$ and $y \in \Phi(f')$, and let $\{k\} = N^0(j)$. In the case $y_j = \texttt{0}$, we can construct a function $f \in F(D)$ equal to f', except for

$$f_j(x) = \big((x_i \oplus \tilde{\sigma}_{ij}) \vee (x_k \oplus y_k)\big) \wedge \bigwedge_{\ell \in N(j) \setminus \{i,k\}} \big((x_\ell \oplus \tilde{\sigma}_{\ell j}) \vee (x_k \oplus \neg y_k)\big).$$

We have $f_j(y) = \texttt{0} = y_j$ because the left hand side of the conjunction is false, thus $y \in \Phi(f)$ (the case $y_j = \texttt{1}$ is symmetric by switching OR and AND functions, and replacing y_k with $\neg y_k$). □

Lemma 2. *Let D be a simple SID. Then $\phi(D) \geq 1$ if and only if each non-trivial initial strongly connected component of D contains a positive cycle.*

Proof. The left to right implication has been proved by Aracena [4, Corollary 3]. For the converse, suppose that $D = (V, A, \sigma)$ has p initial strongly connected components $H_1, \ldots, H_p$. For all $k \in [p]$, if H_k is trivial then i_k denotes the unique vertex it contains, and otherwise we select a positive cycle C_k in H_k and an arc (j_k, i_k) inside. Then, D can be spanned by a forest of p vertex disjoint trees $T_1, \dots, T_p$ rooted in $i_1, \dots, i_p$ such that if H_k is not trivial then the path from i_k to j_k contained in T_k is the one contained in C_k. For all $k \in [p]$ and all vertices j in T_k, we denote by P_{kj} the path from i_k to j contained in T_k (if $j = i_k$ this path is of length zero and positive by convention).

Now, we define $f \in F(D)$ as follows. First, for all $k \in [p]$, if H_k is trivial then f_{i_k} is the constant $\mathtt{0}$ function, and otherwise f_{i_k} is the AND function. Second, for all $k \in [p]$ and all vertices $j \neq i_k$ in T_k, f_j is the AND function if P_{kj} is positive and the OR function otherwise. Next, we define $x \in \{\mathtt{0}, \mathtt{1}\}^V$ as follows: for all $j \in V$, $x_j = \mathtt{0}$ if and only if P_{kj} is positive (thus $x_{i_k} = \mathtt{0}$ for all $k \in [p]$).

We claim that $x \in \Phi(f)$. Indeed, given $k \in [p]$ and a vertex $j \neq i_k$ in T_k, it is easy to prove that $f_j(x) = x_j$ by induction on the length of P_{kj}. Next, if H_k is trivial then $f_{i_k}(x) = \mathtt{0}$. Otherwise, (j_k, i_k) is an arc of H_k. Let s be the sign of the path P_{kj_k}, which is in C_k by construction. Since C_k is positive, $s = \sigma_{j_k i_k}$. So if $\sigma_{j_k i_k} = 1$ then $x_{j_k} = \mathtt{0}$ and thus $f_{i_k}(x) = \mathtt{0}$, and if $\sigma_{j_k i_k} = -1$ then $x_{j_k} = \mathtt{1}$ and thus $f_{i_k}(x) = \mathtt{0}$. In all cases, $f_{i_k}(x) = \mathtt{0} = x_{i_k}$. We deduce that $x \in \Phi(f)$. □

Thus, to decide if $\phi(D) \geq 1$, it is sufficient to compute the non-trivial initial strongly connected components of D (this can be done in linear time [29]) and to check if they contain a positive cycle. As described below, this checking can be done in polynomial time using the following difficult theorem independently proved by Robertson, Seymour and Thomas [27] and McCuaig [20].

Theorem 1 ([20, 27]). *There exists a polynomial time algorithm for deciding if a given digraph contains a cycle of even length.*

Let D be a signed digraph with n vertices, and let $\tilde{D}$ be obtained from D by replacing each positive arc by a path of length two, with two negative arcs, where the internal vertex is new. Then $\tilde{D}$ has at most $n + n^2$ vertices, and it is easy to see that D has a positive cycle if and only if $\tilde{D}$ has a cycle of even length [21]. We then deduce the following theorem.

Theorem 2. 1-MFPP *is in* P.

4 k-Maximum Fixed Point Problem for $k \geq 2$

Theorem 3. *For any $k \geq 2$,* k-MFPP *is* NP*-complete, even with $\Delta(D) \leq 2$.*

Theorem 3 is obtained from Lemmas 9, 5 and 6.

Lemma 3. *For any $k \geq 2$,* k-MFPP *is in* NP*.*

Proof (sketch, see details in Appendix A). First, consider the case where $\Delta(D) \leq d$ for some constant d. Then a certificate of $\phi(D) \geq k$ could consist in a network $f \in F(D)$ and k distinct fixed points $x^{(1)}, \dots, x^{(k)}$. The fact that $f \in F(D)$, and $f(x^{(i)}) = x^{(i)}$ with distinct $x^{(i)}$ for all $i \in [k]$, is checked in polynomial time.

However, when $\Delta(D)$ is not bounded, $F(D)$ can be of doubly exponential size in n. Thus, some functions f require an exponential space to be encoded. Instead, one can give as a certificate a partial function $h : X \to \{0,1\}^n$ with $X \subseteq \{0,1\}^n$ such that $f(x) = h(x)$ for any $x \in X$. In the set X, we put k fixed points and configurations which assert the effectiveness of the arcs. To check the certificate it is sufficient to ensure that there are no inconsistencies (independently for each local function). As a result, the problem is in NP. □

A shorter certificate (only the k fixed points) is possible when D is simple (see appendix B). This result from the following theorem. Note that the extending partial Boolean functions is a well established topic [9, 8].

Theorem 4. *Let D be a simple SIG with vertex set V and consider a partial BN $h : X \to \{0,1\}^V$ with $X \subseteq \{0,1\}^V$. There is a $\mathcal{O}(|X|^2|V|^2)$-time algorithm to decide if there exists an extension of h in $F(D)$.*

We now prove that 2-MFPP is NP-hard. We will use observations from [4].

Lemma 4 ([4]). *Let $D = (V, A, \sigma)$ be a simple signed digraph, $f \in F(D)$ and x, y two distinct fixed points of f. Then there exists a positive cycle C in D such that, for any arc (i, j) in C, we have $x_i \oplus \tilde{\sigma}_{ij} = x_j \neq y_j = y_i \oplus \tilde{\sigma}_{ij}$.*

Remark 1. If the positive cycle C in Lemma 4 has only positive arcs, then either $x_i < y_i$ for all vertex i in C, or $x_i > y_i$ for all vertex i in C.

Remark 2. Given $f \in F(D)$ and x, y two distinct fixed points of f, for any feedback vertex set I of D we have $x_I \neq y_I$.

Lemma 5. *The problem* 2-MFPP *is NP-hard, even with $\Delta(D) \leq 2$.*

Proof. We reduce 3SAT to our problem. Let us consider a 3SAT instance ψ with n variables $\lambda_1, \dots, \lambda_n$ and m clauses $\mu_1, \dots, \mu_m$. We define the signed digraph $D_\psi = (V, A, \sigma)$, where $|V| = 4n + 2m + 1$, as follows (see Figure 2).

First, $V = R \cup P \cup L \cup \bar{L} \cup S \cup T$ with $R = \{r_i \mid i \in [n]\}$, $P = \{p_i \mid i \in [0, n]\}$, $L = \{\ell_i \mid i \in [n]\}$, $\bar{L} = \{\bar{\ell}_i \mid i \in [n]\}$, $S = \{s_i \mid i \in [m]\}$, and $T = \{t_i \mid i \in [m]\}$. To simplify the notation let $s_0 = p_0$ and $s_{m+1} = p_n$. Second,

$$\begin{aligned} A := &\bigcup\nolimits_{i \in [n]} \{(p_{i-1}, \ell_i), (p_{i-1}, \bar{\ell}_i), (\ell_i, p_i), (\bar{\ell}_i, p_i), (r_i, \ell_i), (r_i, \bar{\ell}_i)\} \\ &\cup \bigcup\nolimits_{j \in [m]} \{(t_i, s_i), (s_i, s_{i-1})\} \cup \{(p_n, s_m)\} \\ &\cup \{(\ell_i, t_j) \mid i \in [n],\ j \in [m] \text{ if } \lambda_i \text{ appears positively in } \mu_j\} \\ &\cup \{(\bar{\ell}_i, t_j) \mid i \in [n],\ j \in [m] \text{ if } \lambda_i \text{ appears negatively in } \mu_j\}. \end{aligned}$$

Arcs in $\{(s_i, t_i) \mid i \in [m]\} \cup \{(r_i, \ell_i) \mid i \in [n]\}$ are negative, all others are positive.

Let us first prove that if ψ is satisfiable then there exists a BN $f \in F(D_\psi)$ with has at least two fixed points. Consider a valid assignment $v : \{\lambda_1, \dots \lambda_n\} \to \{\perp, \top\}$. Let $I^\perp = \{i \in [n] \mid v(\lambda_i) = \perp\}$ and $I^\top = \{i \in [n] \mid v(\lambda_i) = \top\}$. We define $f \in F(D_\psi)$ as follows.

- For all $i \in I^\perp$ (resp. $I^\top$), f_{r_i} is the constant $\mathtt{0}$ (resp. $\mathtt{1}$) function.
- For all $i \in [n]$, f_{ℓ_i} and $f_{\bar{\ell}_i}$ are both AND functions.
- For all $i \in [0, n]$, f_{p_i} is the OR function.
- For all $i \in [m]$, f_{s_i} and f_{t_i} are the AND functions.

The two following configurations x and y are distinct fixed points of f, and therefore $\phi(D_\psi) \geq 2$: for all $j \in V$,

$$x_j = \begin{cases} \mathtt{1} & \text{if } j \in \{r_i \mid i \in I^\top\} \\ \mathtt{0} & \text{otherwise} \end{cases}$$

$$y_j = \begin{cases} \mathtt{1} & \text{if } j \in \{r_i \mid i \in I^\top\} \cup P \cup S \cup \{\ell_i \mid i \in I^\perp\} \cup \{\bar{\ell}_i \mid \in I^\top\} \\ \mathtt{0} & \text{otherwise.} \end{cases}$$

Now, we prove that if $\phi(D_\psi) \geq 2$ then ψ is satisfiable. Consider a BN $f \in F(D_\psi)$ with two distinct fixed points x and y. Remark that $\{p_0\}$ is a feedback vertex set of D_ψ. In other words, all cycles of D_ψ contain p_0. We deduce from Remark 2 that $x_{p_0} \neq y_{p_0}$ and that $\phi(D_\psi) \leq 2$. Without loss of generality, suppose that $x_{p_0} < y_{p_0}$. Remark also that any cycle containing one of the vertices $t_1, \ldots, t_m$ is negative, and that no positive cycle in D_ψ contains any negative arc. Thus, according to Remark 1, there exists a cycle C such that $x_j < y_j$ for every vertex j in C. In other words, $x_P < y_P$ and $x_S < y_S$ and for every $i \in [n]$ either C contains ℓ_i and we have $x_{\ell_i} < y_{\ell_i}$, or it contains $\bar{\ell}_i$ and we have $x_{\bar{\ell}_i} < y_{\bar{\ell}_i}$. We construct the following assignment v from C.

$$v(\lambda_i) = \begin{cases} \perp & \text{if } C \text{ contains } \ell_i, \\ \top & \text{if } C \text{ contains } \bar{\ell}_i. \end{cases}$$

For the sake of contradiction, suppose that v does not satisfy the formula. As a consequence, there is a clause μ_j which is false with assignment v. In other words, any variable which appears positively in the clause is assigned to false and any variable which appears negatively is assigned to true.

Let us prove that $x_{t_j} < y_{t_j}$. Since any incoming arc of t_j is positive, and since x and y are fixed points, it is sufficient to prove that, for every in-neighbor ℓ of t_j, we have $x_\ell < y_\ell$. By definition of D_ψ, any in-neighbor of t_j corresponds to a variable λ_i of the clause. If λ_i appears positively (resp. negatively) in clause μ_j then the in-neighbor of t_j corresponding to λ_i is ℓ_i (resp. $\bar{\ell}_i$). Since $v(\lambda_i) = \perp$ (resp. $\top$) because the clause is false then C contains ℓ_i (resp. $\bar{\ell}_i$) and we have $x_{\ell_i} < y_{\ell_i}$ (resp. $x_{\bar{\ell}_i} < y_{\bar{\ell}_i}$). As a result, $x_{t_j} < y_{t_j}$.

Now, the vertex s_j has two in-neighbors. One of them is s_{j+1} and we have $\sigma_{s_{j+1}s_j} = 1$ and $x_{s_{j+1}} < y_{s_{j+1}}$. The other is t_j with $\sigma_{t_j s_j} = -1$ and $x_{t_j} < y_{t_j}$. Hence, there are two possible local functions for f_{s_i}:

- $f_{s_j}(z) = z_{s_{j+1}} \vee \neg z_{t_j}$, and then $x_{s_j} = f_{s_j}(x) = x_{s_{j+1}} \vee \neg x_{t_j} = \mathtt{0} \vee \neg \mathtt{0} = \mathtt{1}$.
- $f_{s_i}(z) = z_{s_{i+1}} \wedge \neg z_{t_j}$, and then $y_{s_j} = f_{s_j}(y) = y_{s_{j+1}} \wedge \neg y_{t_j} = \mathtt{1} \wedge \neg \mathtt{1} = \mathtt{0}$.

In both cases, we do not have $x_{s_j} < y_{s_j}$, which is a contradiction since s_j is in C. As a result, the 3SAT instance ψ is satisfiable. Additionally, remark that

$\phi(D_\psi) \geq 1$ because, with the constant $\mathtt{1}$ function for the vertices in R, and the OR local function everywhere else, the configuration $z_i = \mathtt{1}$ for all i is a fixed point. We can conclude that $\phi(D_\psi) = 1$ when ψ is unsatisfiable.

To get a bounded degree $\Delta(D_\psi) \leq 2$, notice that only vertices in T have in-degree three, which can be decreased by adding an intermediate vertex (see the right picture in Figure 2) while preserving the correctness of the reduction. □

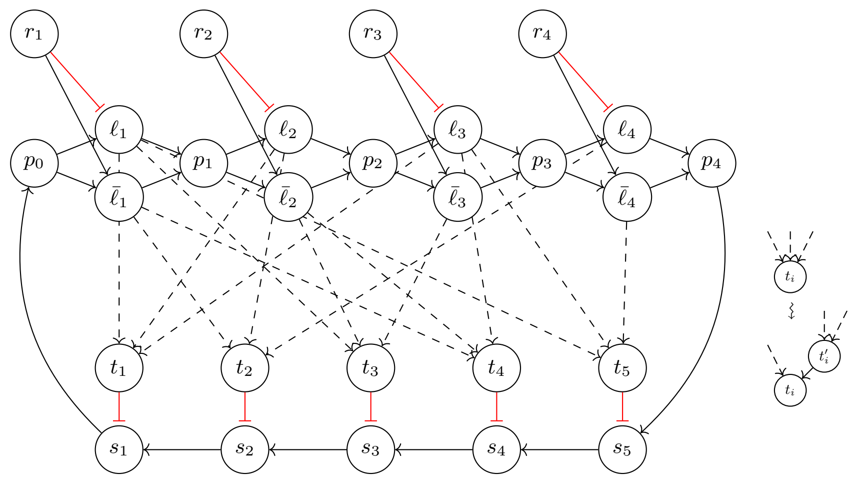


Fig. 2: Example of construction in the reduction from 3SAT to k-MFPP (Lemma 5). This signed digraph D_ψ implements the following 3SAT instance ψ:
$(\lambda_1 \vee \lambda_2 \vee \lambda_3) \wedge (\neg\lambda_1 \vee \lambda_2 \vee \lambda_4) \wedge (\lambda_1 \vee \neg\lambda_2 \vee \neg\lambda_3) \wedge (\neg\lambda_1 \vee \neg\lambda_2 \vee \lambda_3) \wedge (\lambda_1 \vee \lambda_3 \vee \neg\lambda_4)$
which is satisfiable if and only if $\phi(D_\psi) \geq 2$, otherwise $\phi(D_\psi) = 1$.

We can extend the NP-hardness reduction to any $k \geq 2$.

Lemma 6. *For any $k \geq 2$, k-MFPP is NP-hard, even with $\Delta(D) \leq 2$.*

Proof. Let $\ell = \lfloor \log_2(k-1) \rfloor$, *i.e.* $2^\ell < k \leq 2^{\ell+1}$. Given a formula, consider the digraph D from Lemma 5, and add ℓ new isolated vertices with positive loops. Then 1 or 2 fixed points on D_ψ become respectively 2^ℓ or $2^{\ell+1}$ fixed points. □

Remark 3. For $\Delta(D) \leq 1$, $|F(D)| = 1$ since each local function is the identity or the negation, and computing $\phi(D)$ is in $\mathcal{O}(|D|)$, hence k-MFPP $\in$ P.

5 Maximum Fixed Point Problem

Theorem 5. *When $\Delta(D) \leq d$, MFPP is $NP^{\#P}$-complete.*

In this first part of the section, we prove Theorem 5, from Lemmas 7 and 8.

Lemma 7. *When $\Delta(D) \leq d$,* MFPP*is in* $\mathsf{NP}^{\#\mathsf{P}}$.

Proof. An algorithm in $\mathsf{NP}^{\#\mathsf{P}}$ to solve MFPP is, on input D, k:

1. guess local functions f_i for $i \in [n]$ (polynomial from $\Delta(D) \leq d$),
2. construct $\psi = (f_1(x) = x_1) \wedge \cdots \wedge (f_n(x) = x_n)$ on variables $x_1, \ldots, x_n$,
3. compute the number of solutions of ψ with the $\#\mathsf{P}$ oracle, that is $\phi(f)$,
4. accept if and only if $\phi(f) \geq k$.

A non-deterministic branch accepts if and only if $\phi(D) \geq k$. □

Lemma 8. *When $\Delta(D) \leq d$,* MFPP*is* $\mathsf{NP}^{\#\mathsf{P}}$*-hard.*

Proof (sketch, see details in Appendix C). We consider the following problem.

Existential-Majority-3SAT (E-Maj3SAT) *Input:* A 3SAT formula ψ on $\{\lambda_1, \ldots, \lambda_n\}$ and $s \in [n]$ *Question:* Is there an assignment v of $\lambda_1, \ldots, \lambda_s$ such that the majority of assignments of $\lambda_{s+1}, \ldots, \lambda_n$ satisfy ψ?

We know that E-Maj3SAT is $\mathsf{NP}^{\mathsf{PP}}$-complete [19] and that $\mathsf{NP}^{\#\mathsf{P}} = \mathsf{NP}^{\mathsf{PP}}$ (direct extension of $\mathsf{P}^{\#\mathsf{P}} = \mathsf{P}^{\mathsf{PP}}$ [22]). Consequently, it is sufficient to prove that we can reduce E-Maj3SAT to MFPP. To represent an instance (ψ, s) of E-Maj3SAT, we construct a digraph $D_{\psi,s}$ similar to the digraph D_ψ constructed in Lemma 5 except that we add a positive loop to the $q = n-s$ vertices $r_{s+1}, \ldots, r_n$. We claim that $\phi(D_{\psi,s}) = \alpha + 2^q$, with

$$\alpha = \max_{v:\{\lambda_1,\ldots,\lambda_s\}\to\{\perp,\top\}} |\{u : \{\lambda_{s+1}, \ldots, \lambda_n\} \to \{\perp, \top\} \mid v \cup u \text{ satisfies } \psi\}|.$$

Indeed, consider $f \in F(D_{\psi,s})$ with $\phi(f) = \phi(D_{\psi,s})$. As in Lemma 5, the functions f_i for $i \in \{\ell_1, \bar{\ell}_1, \ldots, \ell_s, \bar{\ell}_s\}$ correspond to an assignment v of $\lambda_1, \ldots, \lambda_s$. Moreover, each valuation u of $\lambda_{s+1}, \ldots, \lambda_n$ corresponds to one (resp. two) fixed points if the assignment $v \cup u$ makes ψ false (resp. true). As a consequence, the reduction is correct by setting $k = \frac{3}{2}2^q$. □

In this second part, we study MFPP with unbounded maximum degree.

Theorem 6. *When $\Delta(D)$ is unbounded,* MFPP *is* $\mathsf{NEXPTIME}$*-complete.*

Proof (sketch, see details in Appendix D). It is easy to see that the problem MFPP with unbounded degree is in $\mathsf{NEXPTIME}$. Indeed, to know if $\phi(D) \geq k$ it is sufficient to guess a function $f \in F(D)$ (encoded in exponential space), to compute $\phi(f)$ (in exponential time) and then accept if $\phi(f) \geq k$, reject otherwise. A non-deterministic branch accepts if and only if $\phi(D) \geq k$.

For the hardness, we reduce from Succint-3SAT [22], which is 3SAT where ψ has $n = 2^{\tilde{n}}$ variables, $m = 2^{\tilde{m}}$ clauses, and is given by a circuit C with:

- $\tilde{m}$ input bits for the clauses, and 2 for the three literal positions,
- $\tilde{n}$ output bits to give the corresponding variable, and 1 for its polarity.

D is acyclic, has in-degree at most 2, and has simple OR, AND, NOT, identity or constant functions. The idea is to generalize the construction from the proof of Theorem 3, with one literal for each node of the circuit C (top part), and additional clauses implementing the circuit (bottom part). With non-trivial additional elements, choosing local functions correspond to choosing an assignment. There will be a maximum of one (resp. two) fixed point for each non-satisfied (resp. satisfied) clause. As a result, ψ is satisfiable if and only if $\phi(D) \geq 2m$. □

6 Conclusion

This first work raises many open questions. First, is the problem 1-MFPP P-complete? We proved that it is equivalent to the problem of finding an even cycle in a digraph, for which the P versus NP-complete status remained open until [20, 27]. Now we know that the problem is in P, but is it a tight bound?

Several natural extensions of the present results may be addressed. What happens to the complexity when we study the minimum number of fixed points instead of the maximum? And for digraphs with only positive arcs? What about limit cycles of period greater than one instead of fixed points? Understanding the complexity of computing bounds on dynamical properties of BNs respecting a given interaction digraph is a new and promising approach, both on the theoretical and practical points of view.

7 Acknowledgments

The authors would like to thank for their support the *Young Researcher* project ANR-18-CE40-0002-01 “FANs”, project ECOS-CONICYT C16E01, and project STIC AmSud CoDANet 19-STIC-03 (Campus France 43478PD).

References

1. The Online Encyclopedia of Integer Sequences, founded in 1964 by N. J. A. Sloane. Sequence `A006126`, https://oeis.org/A006126
2. Akutsu, T., Miyano, S., Kuhara, S.: Identification of genetic networks from a small number of gene expression patterns under the boolean network model. In: Biocomputing'99, pp. 17–28. World Scientific (1999)
3. Albert, R.: Boolean modeling of genetic regulatory networks. In: Complex Networks, Lect. Notes in Ph., vol. 650, pp. 459–481. Springer Berlin Heidelberg (2004)
4. Aracena, J.: Maximum number of fixed points in regulatory Boolean networks. Bulletin of Mathematical Biology **70**(5), 1398–1409 (2008)
5. Aracena, J., Demongeot, J., Goles, E.: Fixed points and maximal independent sets in and-or networks. Discrete Applied Math. **138**(3), 277–288 (2004)
6. Aracena, J., Richard, A., Salinas, L.: Maximum number of fixed points in AND-OR-NOT networks. J. of Comput. and System Sciences **80**(7), 1175–1190 (2014)
7. Aracena, J., Richard, A., Salinas, L.: Number of fixed points and disjoint cycles in monotone boolean networks. SIAM J. on Discrete Math. **31**(3), 1702–1725 (2017)

8. Boros, E., Ibaraki, T., Makino, K.: Error-free and best-fit extensions of partially defined boolean functions. Information and Computation **140**(2), 254–283 (1998)
9. Crama, Y., Hammer, P.L.: Boolean Functions: Theory, Algorithms, and Applications. Cambridge University Press (2011)
10. Feder, T.: A new fixed point approach for stable networks and stable marriages. J. of Comput. and System Sciences **45**(2), 233–284 (1992)
11. Gadouleau, M., Richard, A., Riis, S.: Fixed points of Boolean networks, guessing graphs, and coding theory. SIAM J. on Discrete Math. **29**(4), 2312–2335 (2015)
12. Gadouleau, M., Riis, S.: Graph-theoretical constructions for graph entropy and network coding based communications. IEEE Trans. on Inf. Th. **57**(10), 6703–6717 (2011)
13. Goles, E., Salinas, L.: Sequential operator for filtering cycles in Boolean networks. Advances in Applied Math. **45**(3), 346–358 (2010)
14. Kauffman, S.A.: Metabolic stability and epigenesis in randomly connected nets. J. of Theoretical Biology **22**, 437–467 (1969)
15. Kauffman, S.A.: Origins of Order Self-Organization and Selection in Evolution. Oxford University Press (1993)
16. Kaufman, M., Soulé, C., Thomas, R.: A new necessary condition on interaction graphs for multistationarity. J. of Theoretical Biology **248**(4), 675–685 (2007)
17. Kosub, S.: Dichotomy results for fixed-point existence problems for boolean dynamical systems. Math. in Comput. Sci. **1**(3), 487–505 (2008)
18. Le Novère, N.: Quantitative and logic modelling of molecular and gene networks. Nature Reviews Genetics **16**, 146–158 (2015)
19. Littman, M.L., Goldsmith, J., Mundhenk, M.: The computational complexity of probabilistic planning. J. Artif. Intell. Res. **9**, 1–36 (1998)
20. McCuaig, W.: Pólya's permanent problem. the electronic journal of combinatorics **11**(1), 79 (2004)
21. Montalva, M., Aracena, J., Gajardo, A.: On the complexity of feedback set problems in signed digraphs. Electronic Notes in Discrete Math. **30**, 249–254 (2008)
22. Papadimitriou, C.H.: Computational complexity. Addison-Wesley (1994)
23. Paulevé, L., Richard, A.: Topological Fixed Points in Boolean Networks. Comptes Rendus de l'Académie des Sciences-Series I-Math. **348**(15-16), 825–828 (2010)
24. Remy, E., Ruet, P., Thieffry, D.: Graphic requirements for multistability and attractive cycles in a Boolean dynamical framework. Advances in Applied Math. **41**(3), 335–350 (2008)
25. Richard, A.: Positive and negative cycles in boolean networks. J. of theoretical biology **463**, 67–76 (2019)
26. Robert, F.: Discrete iterations: a metric study, Series in Computational Math., vol. 6. Springer (1986)
27. Robertson, N., Seymour, P., Thomas, R.: Permanents, pfaffian orientations, and even directed circuits. Annals of Math. **150**(3), 929–975 (1999)
28. Soulé, C.: Mathematical approaches to differentiation and gene regulation. C.R. Paris Biologies **329**, 13–20 (2006)
29. Tarjan, R.: Depth-first search and linear graph algorithms. SIAM journal on computing **1**(2), 146–160 (1972)
30. Thomas, R.: Boolean formalization of genetic control circuits. J. of Theoretical Biology **42**(3), 563–585 (1973). https://doi.org/10.1016/0022-5193(73)90247-6
31. Thomas, R., d'Ari, R.: Biological Feedback. CRC Press (1990)
32. Thomas, R., Kaufman, M.: Multistationarity, the basis of cell differentiation and memory. II. Logical analysis of regulatory networks in terms of feedback circuits. Chaos: An Interdisciplinary J. of Nonlinear Sci. **11**(1), 180–195 (2001)

A k-MFPP $\in$ NP for $k \geq 2$, proof of Lemma 3

Lemma 9. *For any $k \geq 2$, k-MFPP is in NP.*

Proof. Consider a SID $D = (V, A, \sigma)$ where $V = [n]$. Suppose that $k \leq 2^n$, otherwise the answer is yes. Given $j \in [n]$ and $s, s' \in \{-1, 0, 1\}$, we set $N^{\{s,s'\}}(j) = N^s(j) \cup N^{s'}(j)$. For any $j \in [n]$, let $I(j) = N^{\{-1,1\}}(j)$ and $b^{(j)} \in \{0, 1\}^n$ such that for any $i \in I(j)$, $b_i^{(j)} = \tilde{\sigma}_{ij}$. To check that $\phi(D) \geq k$ one can use the following algorithm (choose_in is non-deterministic).

Algorithm 1 $\phi(D) \geq k$

Require: A signed interaction digraph D.

for $j \in [n]$ **do**
    $X_j \leftarrow \emptyset$
end for
FIXED_POINTS $\leftarrow \emptyset$
for $p \in [k]$ **do**
    $x \leftarrow$ choose_in$\big(\{0,1\}^n \setminus$ FIXED_POINTS$\big)$.
    FIXED_POINTS $\leftarrow$ FIXED_POINTS $\cup \{x\}$
    for $j \in [n]$ **do**
        $X_j \leftarrow X_j \cup \{(x, x_j)\}$
    end for
end for
for $j \in [n]$ **do**
    for $i \in N(j)$ **do**
        if $i \in N^{\{-1,0\}}(j)$ **then**
            $x \leftarrow$ choose_in$\big(\{x \in \{0,1\}^n \mid x_i = 0\}\big)$
            $X_j \leftarrow X_j \cup \{(x, 1)\}$
            $X_j \leftarrow X_j \cup \{(x + e_i, 0)\}$
        end if
        if $i \in N^{\{0,1\}}(j)$ **then**
            $x \leftarrow$ choose_in$\big(\{x \in \{0,1\}^n \mid x_i = 0\}\big)$
            $X_j \leftarrow X_j \cup \{(x, 0)\}$
            $X_j \leftarrow X_j \cup \{(x + e_i, 1)\}$
        end if
    end for
end for
for $j \in [n]$ **do**
    for $(x, y) \in X_j$ **do**
        for $(x', y') \in X_j$ **do**
            if $x_{N^0(j)} = x'_{N^0(j)}$ and $(x' + b^{(j)})_{I(j)} \leq (x + b^{(j)})_{I(j)}$ and $y' > y$ **then**
                return False
            end if
        end for
    end for
end for
return True

We can see that executed on a non-deterministic machine, this algorithm stops in a polynomial time, $\mathcal{O}(n\Delta(D)^2)$ where n is the size of V and $\Delta(D)$ is the maximum incoming degree of D. It stops as well in $\mathcal{O}(m^2)$ steps where m is the size of A.

Now, let us see why this algorithm works. In other words, why a non-deterministic branch of this algorithm returns true if and only if the answer of the problem is yes.

First, suppose there exists $f \in F(D)$ with k fixed points. One can consider the following execution. First, the k configurations chosen in line 6 are k fixed points of f. Second, the configuration x chosen in line 13 is a configuration such that $\mathtt{0} = f_j(x) < f_j(x+e_i) = \mathtt{1}$. This configuration exists since $i \in N^{\{-1,0\}}(j)$ and $f \in F(D)$. Similarly, the configuration chosen line in 18 is a configuration x such that $f_j(x) > f_j(x+e_i)$. Because $f \in F(D)$, the condition on line 27 is never satisfied. This execution of the algorithm then returns "True".

Second, suppose that the algorithm returns "True". Consider an accepting branch of the execution. Let us define $f \in F(n)$ and prove that $f \in F(D)$. For each $j \in [n]$ and $x \in \{\mathtt{0},\mathtt{1}\}^n$, we define $f_j(x)$ as follows: $f_j(x) = \mathtt{1}$ if and only if there exist $(x', \mathtt{1}) \in X_j$ such that

$$x_{N^0(j)} = {x'}_{N^0(j)} \text{ and } (x' + b^{(j)})_{I(j)} \leq (x + b^{(j)})_{I(j)}.$$

Consider the SID $D' = (V, A', \sigma')$ of f and let us show that $D = D'$. Fix $j \in [n]$ and let us see that for any $s \in \{-1, 0, 1\}$, we have $N^s_{D'}(j) = N^s_D(j)$. First, prove that $N_{D'}(j) \subseteq N_D(j)$. Take $i \in N_{D'}(j)$. There exists $x \in \{\mathtt{0},\mathtt{1}\}^n$ such that $f_j(x) = \mathtt{1} \neq \mathtt{0} = f_j(x+e_i)$. By definition of f_j, there exists $(x', \mathtt{1}) \in X_j$ such that

$$x_{N^0_D(j)} = {x'}_{N^0_D(j)} \text{ and } (x' + b^{(j)})_{I(j)} \leq (x + b^{(j)})_{I(j)}.$$

Now, if $i \notin N_D(j)$, we also have

$$(x+e_i)_{N^0_D(j)} = {x'}_{N^0_D(j)} \text{ and } (x' + b^{(j)})_{I(j)} \leq (x + e_i + b^{(j)})_{I(j)}.$$

Thus, $f_i(x + e_i) = \mathtt{1}$, a contradiction. Hence, $i \in N(j)$ and $N_{D'}(j) \subseteq N_D(j)$.

The instruction lines $13, 14, 15$ and $18, 19, 20$, ensure the following: if $i \in N^{\{-1,0\}}_D(j)$ then $i \in N^{\{-1,0\}}_{D'}(j)$, and if $i \in N^{\{0,1\}}_D(j)$ then $i \in N^{\{0,1\}}_{D'}(j)$. As a result, $N^0_D(j) \subseteq N^0_{D'}(j)$. It is now sufficient to prove that $N^{-1}_D(j) \cap N^0_{D'}(j) = \emptyset$ and that $N^1_D(j) \cap \in N^0_{D'}(j) = \emptyset$. Consider $i \in N^{-1}_D(j)$ and suppose for the sake of contradiction that $i \in N^0_{D'}(j)$. It means that there exists $x \in \{\mathtt{0},\mathtt{1}\}^n$ with $x_i = \mathtt{0}$ such that $\mathtt{0} = f_j(x) < f_j(x+e_i) = \mathtt{1}$. Thus, there exists $(x', \mathtt{1})$ in X_j such that

$$(x+e_i)_{N^0_D(j)} = {x'}_{N^0_D(j)} \text{ and } (x' + b^{(j)})_{I(j)} \leq (x + e_i + b^{(j)})_{I(j)}.$$

Since $i \in N^{-1}_D(j)$, we have $b^{(j)}_\ell = \tilde{\sigma}_{ij} = 1$. Then,

$$(x' + b^{(j)})_{I(j)} \leq (x + e_i + b^{(j)})_{I(j)} \leq (x + b^{(j)})_{I(j)}.$$

Hence, $f_j(x) = \mathtt{1}$, a contradiction. Thus, $i \in N^{-1}_D(j)$, $N^{-1}_D(j) \cap N^0_{D'}(j) = \emptyset$ and $N^{-1}_{D'}(j) \subseteq N^{-1}_D(j)$.

The proof of $N^1_{D'}(j) \subseteq N^1_D(j)$ is similar. □

B The extension problem, proof of Theorem 7

Let D be a simple signed digraph with vertex set V, and let h be a *partial* BNs on V, that is, a function $h : X \to \{\mathtt{0},\mathtt{1}\}^V$ where $X \subseteq \{\mathtt{0},\mathtt{1}\}^V$. A *$D$-extension* of h is a BN $f \in F(D)$ which is *consistent* with h, that is, such that $f_i(x) = h_i(x)$ for all $x \in X$ and $i \in V$ with $h_i(x) \in \{0, 1\}$.

Theorem 7. *Let D be a simple signed digraph with vertex set V and consider a partial BN $h : X \to \{\mathtt{0},\mathtt{1}\}^V$ with $X \subseteq \{\mathtt{0},\mathtt{1}\}^V$. There is a $\mathcal{O}(|X|^2|V|^2)$-time algorithm to decide if there exists a D-extension of h.*

The proof involves three lemmas and some additional definitions. Let $\leq$ be the partial order on $\{\mathtt{0},\mathtt{1}\}^n$ defined by $x \leq y$ if and only if $x_i \leq y_i$ for all $i \in [n]$. An *antichain* is a subset $\{\mathtt{0},\mathtt{1}\}^n$ that does not contain distinct comparable elements. Given $A \subseteq \{\mathtt{0},\mathtt{1}\}^n$, we denote by A^- the set of $x \in \{\mathtt{0},\mathtt{1}\}^n$ such that $x \leq a$ for some $a \in A$, and we denote by A^+ the set of $x \in \{\mathtt{0},\mathtt{1}\}^n$ such that $a \leq x$ for some $a \in A$. Let $f : \{\mathtt{0},\mathtt{1}\}^n \to \{\mathtt{0},\mathtt{1}\}$ be a Boolean function (BF). f is *monotone* if $x \leq y$ implies $f(x) \leq f(y)$. f *depends on input* $i \in [n]$ if $f(x) \neq f(x + e_i)$ for some $x \in \{\mathtt{0},\mathtt{1}\}^n$, and f is *total* if it depends on its n inputs. A *partial* BF is a function $h : X \to \{\mathtt{0},\mathtt{1}\}$, where $X \subseteq \{\mathtt{0},\mathtt{1}\}^n$. A *monotone extension* of h is a total monotone BF f which is *consistent* with h, *i.e.* $f(x) = h(x)$ for all $x \in X$ with $h(x) \in \{\mathtt{0},\mathtt{1}\}$.

The first lemma is a simple characterization of the partial BFs that admit a monotone extension.

Lemma 10. *Let $h : X \to \{\mathtt{0},\mathtt{1}\}$ be a partial BF. Let A be the set of maximal elements of $h^{-1}(\mathtt{0})$ and let B be the set of minimal elements of $h^{-1}(\mathtt{1})$. Then h has a monotone extension if and only if, among the three conditions below, (1) is true and at least one of (2) and (3) is true:*

(1) There is no $a \in A$ and $b \in B$ with $b \leq a$.
(2) For all $i \in [n]$, there exists $a \in A$ with $a_i = \mathtt{0}$ or $b \in B$ with $b_i = \mathtt{1}$.
(3) $A^- \cup B^+ \neq \{\mathtt{0},\mathtt{1}\}^n$.

Proof. Suppose first that h has a monotone extension f. Then it is clear *(1)* is true. Suppose that the condition *(2)* is false, and let us prove that this forces the condition *(3)* to be true. Since *(2)* is false, there exists $i \in [n]$ such that $a_i = \mathtt{1}$ for all $a \in A$ and $b_i = \mathtt{0}$ for all $b \in B$. Since f depends on input i, there exists $x \in \{\mathtt{0},\mathtt{1}\}^n$ such that $f(x) \neq f(x+e_i)$. Suppose, without loss, that $x_i = \mathtt{0}$. Then $x \leq x + e_i$ thus $f(x) = \mathtt{0}$ and $f(x + e_i) = \mathtt{1}$. If $x \leq a$ for some $a \in A$, then $x + e_i \leq a$ (since $a_i = \mathtt{1}$) but then $f(x + e_i) \leq f(a) = h(x) = \mathtt{0}$, a contradiction. Thus, $x \notin A^-$. Furthermore, if $b \leq x$ for some $b \in B$, then $\mathtt{1} = f(b) \leq f(x)$, a contradiction. Thus, $x \notin B^+$. Hence, $x \notin A^- \cup B^+$ thus *(3)* is true.

Suppose now that *(1)* is true and that at least one of *(2)* and *(3)* is true. Let I be the set of $i \in [n]$ such that $a_i = \mathtt{1}$ for all $a \in A$ and $b_i = \mathtt{0}$ for all $b \in B$. Thus, $I = \emptyset$ is equivalent to *(2)*. Let $Z := \{\mathtt{0},\mathtt{1}\}^n \setminus (A^- \cup B^+)$. We have

$$\forall i \in I, \qquad z \in Z \iff z + e_i \in Z.$$

Indeed, suppose that $z \in Z$. If $z + e_i \leq a$ for some $a \in A$, then $z \leq a$ since $a_i = \mathtt{1}$, a contradiction. If $b \leq z + e_i$ for some $b \in B$, then $b \leq z$ since $b_i = \mathtt{0}$, a contradiction. Thus, $z + e_i \in Z$ and this proves the equivalence. Now we fix an element z of $\{\mathtt{0},\mathtt{1}\}^n$ as follows:

1. If $I = \emptyset$, which is equivalent to *(2)*, then z is any member of A,
2. If $I \neq \emptyset$, then *(3)* is true, thus $Z \neq \emptyset$, and z is any minimal element of Z.

By the equivalence above and the choice of z, we have

$$I \neq \emptyset \;\Rightarrow\; z_i = \mathtt{0} \;\forall i \in I.$$

Suppose that $I \neq \emptyset$ and $a \leq z$ for some $a \in A$. For all $i \in I$ we then have $a_i = \mathtt{1}$ and thus $z_i = \mathtt{1}$, which contradicts the above implication. Thus,

$$I \neq \emptyset \;\Rightarrow\; z \notin A^+.$$

Let us now define $f : \{\mathtt{0},\mathtt{1}\}^n \to \{\mathtt{0},\mathtt{1}\}$ as follows:

$$f(x) := \begin{cases} \mathtt{0} \text{ if } x \in A^- \text{ or } x \leq z \text{ or } x \in B^- \setminus (B \cup A^+), \\ \mathtt{1} \text{ otherwise.} \end{cases}$$

Let us prove that f is consistent with h. Indeed, if $x \in h^{-1}(\mathtt{0})$ then $x \in A^-$ and thus $f(x) = \mathtt{0} = h(x)$. If $x \in h^{-1}(\mathtt{1})$ then $x \in B^+$ thus $x \notin A^-$ and $x \notin B^- \setminus B$. Furthermore, since z belongs to $A \cup Z$, which is disjoint from B^+, we cannot have $x \leq z$. Thus, $f(x) = \mathtt{1} = h(x)$.

We now prove that f is monotone. Suppose that $x \leq y$ and $f(x) = \mathtt{1}$. If $y \in A^-$ then $x \in A^-$ and $f(x) = \mathtt{0}$, a contradiction. If $y \leq z$ then $x \leq z$ and $f(x) = \mathtt{0}$, a contradiction. If $y \in B^- \setminus (B \cup A^+)$ then $x \in B^- \setminus B$, and since $f(x) = \mathtt{1}$ we deduce that $x \in A^+$, but then $y \in A^+$, a contradiction. Thus, $f(y) = \mathtt{1}$. This proves that f is monotone.

It remains to prove that f depends on its n inputs. Let $i \in [n]$. We consider three cases.

1. Suppose that there exists $a \in A$ with $a_i = \mathtt{0}$. We have $f(a) = \mathtt{0}$. Suppose, for a contradiction, that $f(a + e_i) = \mathtt{0}$. Since it is clear that $a + e_i \in A^+ \setminus A^-$, we deduce that $a \leq a + e_i \leq z$, and thus $z \in A^+$. By the implication above, $I = \emptyset$. So, by the choice of z, we have $z \in A$, hence $a + e_i \in A^-$, a contradiction. Thus, $f(a + e_i) = \mathtt{1}$ and f depends on input i.
2. Suppose that there exists $b \in B$ with $b_i = \mathtt{1}$. As proved above, we have $f(b) = \mathtt{1}$. Since $b + e_i \in B^- \setminus B$, if $b + e_i \notin A^+$, then $f(b + e_i) = \mathtt{0}$ thus f depends on input i. If $b + e_i \in A^+$ then there exists $a \in A$ with $a \leq b + e_i$. Thus, $a_i = \mathtt{0}$ and, by the first case, f depends again on input i.
3. Suppose that $i \in I$. We have $f(z) = \mathtt{0}$ and, by the implication above, $z_i = \mathtt{0}$, thus $z + e_i \not\leq z$. By the equivalence above, $z + e_i \in Z$ and thus $z + e_i \notin A^-$. Since $i \in I$ and $z_i = \mathtt{0}$, we have $z + e_i \notin B^-$. We deduce that $f(b + e_i) = \mathtt{1}$, and thus f depends on input i.

$\square$

The second lemma shows that the condition *(3)* in the previous lemma is easy to test.

Lemma 11. *Let A and B be two antichains of $\{0,1\}^n$ such that there is no $a \in A$ and $b \in B$ with $b \leq a$. The following two conditions are equivalent:*

(1) $A^- \cup B^+ = \{0,1\}^n$.
(2) $A \cup B \neq \emptyset$ *and* $x + e_i \in A^- \cup B^+$ *for all* $x \in A \cup B$ *and* $i \in [n]$.

Proof. If *(1)* is true then we trivially have *(2)*. For the converse, suppose that *(2)* is true. We prove that *(1)* is true by induction on n. The case $n = 1$ is obvious, so suppose that $n \geq 2$. Suppose first that $B = \emptyset$, and let $a \in A$. If $a_i = 0$ for some $i \in [n]$ then $a \leq a + e_i$ and thus $a + e_i \notin A^-$, since A is an antichain, and this contradicts *(2)*. We deduce that $a_i = 1$ for all $i \in [n]$, and thus $A^- = \{0,1\}^n$. We prove similarly that $B^+ = \{0,1\}^n$ if $A = \emptyset$. So suppose that $A \neq \emptyset$ and $B \neq \emptyset$. Let A_0 be the set of $a \in A$ with $a_n = 0$ and let $A_1 := A \setminus A_0$. Let B_0 and B_1 be defined similarly. Suppose that $A_1 \cup B_1 = \emptyset$ and let $a \in A_0$. Then $a + e_n \notin A^-$ since $a \leq a + e_n$ and A is an antichain. Thus, $a + e_n \in B^+$, that is, there exists $b \in B$ with $b \leq a + e_n$. Since B_1 is empty, $b_n = 0$ thus $b \leq a$ and this contradicts our assumptions on A and B. Thus, $A_1 \cup B_1 \neq \emptyset$. Since there is no $a \in A_1$ and $b \in B_1$ with $b \leq a$, and since $x + e_i \in A_1^- \cup B_1^+$ for all $x \in A_1 \cup B_1$ and $i \in [n-1]$, by induction hypothesis, $A_1^- \cup B_1^+$ is the set of $x \in \{0,1\}^n$ with $x_n = 1$. We prove similarly that $A_0^- \cup B_0^+$ is the set of $x \in \{0,1\}^n$ with $x_n = 0$. We conclude that $A^- \cup B^+ = \{0,1\}^n$. □

The third lemma puts together the two previous ones and shows that there is fast algorithm for the monotone extension problem.

Lemma 12. *There is a $\mathcal{O}(|X|^2 n)$-time algorithm that takes as input a partial BF $h : X \to \{0,1\}$ with $X \subseteq \{0,1\}^n$, and decides if a monotone extension of h exists.*

Proof. The algorithm is as follows. We compute the set A of maximal elements of $X_0 = h^{-1}(0)$ and the set B of minimal elements of $X_1 = h^{-1}(1)$. This can be done in $\mathcal{O}(|X|^2 n)$. Then, we test the conditions *(1)*, *(2)* and *(3)* of Lemma 10. The first can be tested in $\mathcal{O}(|X|^2 n)$ and the second in $\mathcal{O}(|X| n)$. The third can be tested in $\mathcal{O}(|X|^2 n)$ using the equivalence of Lemma 11. By Lemma 10, h has a monotone extension if and only if *(1)* is true and *(2)* or *(3)* is true. □

We are now in position to prove Theorem 7.

Proof (of Theorem 7). Let D be a simple signed digraph with vertex set V and let $h : X \to \{0,1\}$ a partial BN on V. Let $i \in V$ and let $F_i(D) = \{f_i \mid f \in F(D)\}$ be the set of possible local functions for the component i. Let I be the set of in-neighbors of i in D. Since local functions in $F_i(D)$ only depend on inputs in I, we can regard them as functions from $\{0,1\}^I$ to $\{0,1\}$. Then, we say that h_i has a *D-extension* if there exists $f_i \in F_i(D)$ consistent with h_i, *i.e.* $g(x_I) = h_i(x)$ for all $x \in X$ with $h_i(x) \in \{0,1\}$. Clearly, to prove the theorem, it is sufficient to prove that we can decide in $\mathcal{O}(|X|^2|V|)$ if h_i has a D-extension. Note that if $I=\emptyset$, then h_i has a D-extension if and only if there is no $x,y\in X$ with $h_i(x)=\mathtt{0}$ and $h_i(y)=\mathtt{1}$. So we assume in the following that $I\neq\emptyset$.

Let J be the set of negative in-neighbors of i in D, and let e_J be the configuration in $\{\mathtt{0},\mathtt{1}\}^I$ defined by $(e_J)_j=\mathtt{1}$ if and only if $j\in J$. For each BF $f_i:\{\mathtt{0},\mathtt{1}\}^I\to\{\mathtt{0},\mathtt{1}\}$ we define the BF $\tilde{f}_i:\{\mathtt{0},\mathtt{1}\}^I\to\{\mathtt{0},\mathtt{1}\}$ as follows:

$$\forall x\in\{\mathtt{0},\mathtt{1}\}^I,\qquad \tilde{f}_i(x+e_J)=f_i(x).$$

Note that $\tilde{\tilde{f}}_i=f_i$. Furthermore, it is an easy exercise to prove that

$$f_i\in F_i(D)\iff \tilde{f}_i \text{ is total and monotone.}$$

If there is $x,y\in X$ with $x_I=y_I$ and $h_i(x)=0$ and $h_i(x)=1$, then h_i has clearly no D-extension. So assume that there is such x,y. This allows us to define, without ambiguity, the partial BF $\tilde{h}_i$ on $\{x_I+e_J:x\in X\}$ by:

$$\forall x\in X,\qquad \tilde{h}_i(x_I+e_J)=h_i(x).$$

It is easy to see that

$$f_i \text{ is a } D\text{-extension of } h_i \iff \tilde{f}_i \text{ is a monotone extension of } \tilde{h}_i$$

Now, by Lemma 12, there is a $\mathcal{O}(|X|^2|I|)$-time algorithm to decide if $\tilde{h}_i$ has a monotone extension, and by the equivalence above, we can decide with the same complexity if h_i has a D-extension. □

C Maximum Fixed Point Problem, proof of Lemma 8

Lemma 8 states that when $\Delta(D)\leq d$, the problem MFPP is $\mathsf{NP}^{\#\mathsf{P}}$-hard.

Proof (Lemma 8). We consider the following problem.

Existential-Majority-3SAT (E-Maj3SAT) *Input:* ψ a formula on $\{\lambda_1,\dots,\lambda_n\}$ and $s\in[n]$ *Question:* There exists an assignment v of $\lambda_1,\dots,\lambda_s$ such that the majority of assignments of $\lambda_{s+1},\dots,\lambda_n$ satisfy ψ?

We know that E-Maj3SAT is $\mathsf{NP}^{\mathsf{PP}}$-complete [19] and that $\mathsf{NP}^{\#\mathsf{P}}=\mathsf{NP}^{\mathsf{PP}}$ (direct extension of $\mathsf{P}^{\#\mathsf{P}}=\mathsf{P}^{\mathsf{PP}}$ [22]). Consequently, it is sufficient to prove that we can reduce E-Maj3SAT to MFPP. To represent an instance (ψ,s) of E-Maj3SAT, we construct a digraph $D_{\psi,s}=(V,A,\sigma)$ (see Figure 3) similar to the digraph D_ψ constructed in Lemma 5 except that we add a positive loop to the $q=n-s$ vertices $r_{s+1},\dots,r_n$. We claim that $\phi(D_{\psi,s})=\alpha+2^q$, with

$$\alpha=\max_{v:\{\lambda_1,\dots,\lambda_s\}\to\{\bot,\top\}}|\{u:\{\lambda_{s+1},\dots,\lambda_n\}\to\{\bot,\top\}\mid v\cup u \text{ satisfies } \psi\}|,$$

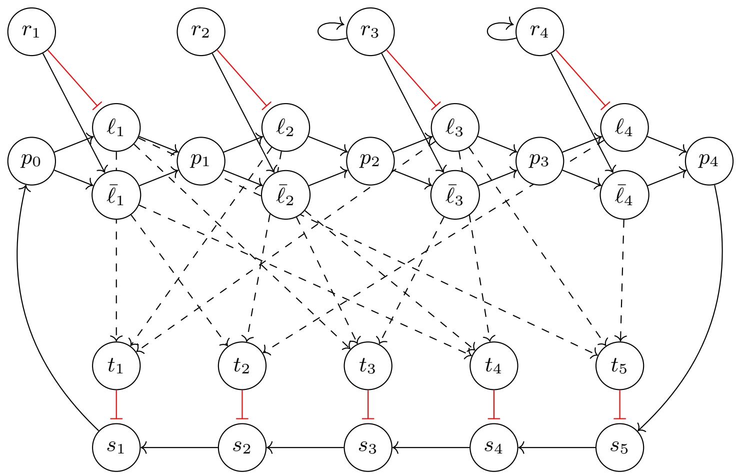


Fig. 3: Example of reduction from E-MAJ3SAT to MFPP restricted to D with a bounded degree. Here is represented $D_{\psi,2}$ for the formula ψ from Figure 2, *i.e.* the E-MAJ3SAT problem asks for the existence of $v : \{\lambda_1, \lambda_2\} \to \{\bot, \top\}$ such that the majority of $v' : \{\lambda_3, \lambda_4\} \to \{\bot, \top\}$ satisfy ψ. This problem is satisfiable if and only if $\phi(D_{\psi,2}) \geq \frac{3}{2}2^2$.

where $v \cup u$ is the assignment of $\lambda_1, \ldots, \lambda_n$ resulting from v and u.

In one direction, consider an assignment v of $\lambda_1, \ldots, \lambda_s$ such that there are α assignments u of $\lambda_{s+1}, \ldots, \lambda_n$ which make ψ true. Let us prove that there exists f such that $\phi(f) \geq \alpha + 2^q$. We define f as in Lemma 5, except for $j \in \{r_{s+1}, \ldots, r_n\}$ where f_j is the function $f_j(x) = x_j$ instead of a constant function. Now, for any assignment u of $\lambda_{s+1}, \ldots, \lambda_n$, let $I_u^\top = \{i \mid v \cup u(\lambda_i) = \top\}$ and $I_u^\bot = \{i \mid v \cup u(\lambda_i) = \bot\}$. Let us define x^u and y^u as

$$\forall j \in [n],\ x_j^u = \mathtt{1} \text{ if } j \in \{r_i \mid i \in I_u^\top\} \text{ and } \mathtt{0} \text{ otherwise, and}$$

$$\forall j \in [n],\ y_j^u = \mathtt{1} \text{ if } j \in \bigcup_{i \in I_u^\top} \{r_i, \ell_i\} \cup \bigcup_{i \in I_u^\bot} \{\bar{\ell}_i\} \cup P \cup S \text{ and } \mathtt{0} \text{ otherwise.}$$

One easily checks that all the 2^q configurations x^u are distinct fixed points of f and that a configuration y^u is another fixed point when $v \cup u$ is a valid assignment of ψ. As a result, $\phi(f) \geq \alpha + 2^q$. Thus, $\phi(D_{\psi,s}) \geq \alpha + 2^q$.

In the other direction, consider a function $f \in F(D_{\psi,s})$ which maximizes $\phi(f)$. Let $\beta + 2^q$ be the number of fixed point of $f \in F(D_{\psi,s})$. We know that $\beta \geq \alpha$, and want to prove that $\beta \leq \alpha$. Let us prove that we can construct an assignment v of $\lambda_1, \ldots, \lambda_s$, such that there are at least β assignments u of $\lambda_{s+1}, \ldots, \lambda_n$ such that $v \cup u$ satisfy ψ.

Let us consider the set A of configuration $a \in \{\mathtt{0},\mathtt{1}\}^q$ such that there exist two distinct fixed points $x^a, y^a \in \Phi(f)$ such that

$$x^a_{\{r_{s+1},\dots,r_n\}} = y^a_{\{r_{s+1},\dots,r_n\}} = a.$$

If $A = \emptyset$ then two distinct fixed points have different values on $\{r_{s+1},\dots,r_n\}$ therefore $\phi(f) \leq 2^q$, and from what precedes we have $\phi(f) = \phi(D_{\psi,s}) = 2^q$ *i.e.* $\beta = \alpha = 0$. Therefore, we can consider that $A \neq \emptyset$.

Note that $\{r_{s+1},\dots,r_n\} \cup \{p_0\}$ is a feedback vertex set of $D_{\psi,s}$. Thus, by Remark 2, for any $a \in A$, $x^a_{p_0} \neq y^a_{p_0}$. Without loss of generality, we consider $x^a_{p_0} < y^a_{p_0}$. Furthermore, by Remark 1, there exists a positive cycle C^a such that:

- C^a contains $S \cup P$ and either ℓ_i or $\bar{\ell}_i$ for any $i \in [n]$, and
- for any $j \in C^a$, $x^a_j < y^a_j$ (because all arcs of C_a are positive).

Let us define the assignment v of $\lambda_1,\dots,\lambda_s$ as follows, for any $i \in [s]$.

- If f_{ℓ_i} is the AND (resp. OR) function and f_{r_i} is the constant $\mathtt{1}$ (resp. $\mathtt{0}$) function, then we set $v(\lambda_i) = \top$. We can see that in this case, for any $a \in A$, we have $x^a_{\ell_i} = y^a_{\ell_i} = \mathtt{0}$ (resp. $\mathtt{1}$). As a result, any C^a contains $\bar{\ell}_i$.
- Otherwise, we set $v(\lambda_i) = \bot$. We can see that in this case, for any $a \in A$, $x^a_{\ell_i} \neq y^a_{\ell_i}$ and we can consider that C^a contains ℓ_i.

Let us now define an assignment u^a of $\lambda_{s+1},\dots,\lambda_n$ for each $a \in A$, as follows. For any $i \in [q]$:

- if C^a contains ℓ_{s+i} then $u^a(\lambda_{s+i}) = \bot$, and
- if C^a contains $\bar{\ell}_{s+i}$ then $u^a(\lambda_{s+i}) = \top$.

With the same reasoning as in Lemma 5, we know that for any $a \in A$, $v \cup u^a$ is a valid assignment of ψ. It is now sufficient to prove that for any distinct $a, a' \in A$, $u^a \neq u^{a'}$. Let i be such that $a_i \neq a'_i$. Let us prove that $u^a(\lambda_{s+i}) \neq u^{a'}(\lambda_{s+i})$. Without loss of generality let $a_i = \mathtt{0}$ and $a'_i = \mathtt{1}$.

- If $u^a(\lambda_{s+i}) = \bot$ then C_a contains ℓ_{s+i}, and $f_{\ell_{s+i}}$ is the AND function. Thus, $f_{\ell_{s+i}}(x^{a'}) = f_{\ell_{s+i}}(y^{a'}) = \neg\mathtt{1} \wedge \cdots = \mathtt{0}$. As a consequence $C^{a'}$ does not contain ℓ_{s+i} and thus $u^a(\lambda_{s+i}) \neq u^{a'}(\lambda_{s+i})$.
- If $u^a(\lambda_{s+i}) = \top$ then C_a contains $\bar{\ell}_{s+i}$, and $f_{\bar{\ell}_{s+i}}$ is the OR function. Thus, $f_{\bar{\ell}_{s+i}}(x^{a'}) = f_{\bar{\ell}_{s+i}}(y^{a'}) = \mathtt{1} \vee \cdots = \mathtt{1}$. As a consequence $C^{a'}$ does not contain $\bar{\ell}_{s+i}$ and thus $u^a(\lambda_{s+i}) \neq u^{a'}(\lambda_{s+i})$.

As a result, for each $a \in A$ the two distinct fixed points x^a, y^a correspond to an assignment u^a such that $v \cup u^a$ satisfies ψ. Therefore, we have $\beta \leq \alpha$, hence $\beta = \alpha$ and $\phi(D_{\psi,s}) = \alpha + 2^q$. This means that $\alpha \geq 2^{q-1}$ if and only if $\phi(D_{\psi,s}) \geq \frac{3}{2}2^q$, which concludes the reduction from E-MAJ3SAT to MFPP.

Remark that $D_{\psi,s}$ is a digraph of maximum in-degree 3. More precisely, only vertices of $T = \{t_i \mid i \in [m]\}$ can have an in-degree superior to 2. Furthermore, using the same trick as in Lemma 5, we can add intermediate components to limit the maximum in-degree to 2. $\square$

D Maximum Fixed Point Problem, proof of Theorem 6

Proof (Theorem 6). Below we prove only the NEXPTIME-hardness of MFPP with unbounded degree.

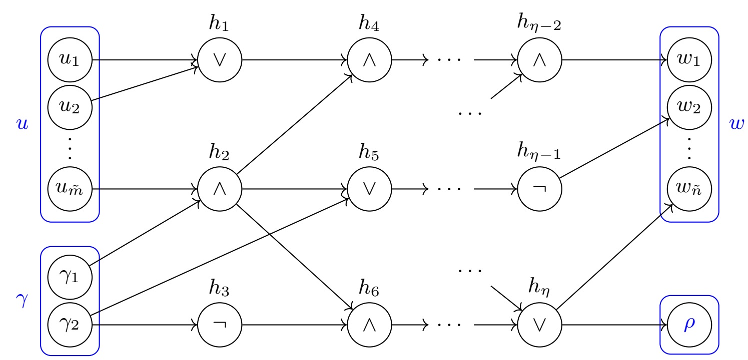


Fig. 4: Circuit encoding a 3SAT formula ψ with $m = 2^{\tilde{m}}$ clauses and $n = 2^{\tilde{n}}$ variables.

To prove that the problem is NEXPTIME-hard, we reduce from Succint-3SAT which is NEXPTIME-hard [22]. Let D be an instance of circuit which encodes a Succint-3SAT problem with $n = 2^{\tilde{n}}$ variables $\lambda_1, \ldots, \lambda_n$ and $m = 2^{\tilde{m}}$ clauses $\mu_1, \ldots, \mu_m$. To lighten the notations, let us denote as well the variables by λ_w for $w \in \{\mathtt{0}, \mathtt{1}\}^{\tilde{n}}$ and the clauses by μ_u for $u \in \{\mathtt{0}, \mathtt{1}\}^{\tilde{m}}$. The circuit C encodes a formula ψ as follows (and as represented in Figure 4). Given the input:

– in $\{u_1, \ldots, u_{\tilde{m}}\}$, the index $u \in [m]$ of a clause, and
– in $\{\gamma_1,\ \gamma_2\}$, the position $\gamma \in \{1, 2, 3\}$ of a literal in the clause μ_u,

the circuit C outputs:

– in $\{w_1, \ldots, w_{\tilde{n}}\}$, the index $w \in [n]$ of the γ^{th} variable in the clause μ_u, and
– in $\{\rho\}$, the polarity of this literal ($\mathtt{0}$ for $\neg\lambda_w$, and $\mathtt{1}$ for λ_w).

We denote $\{h_1, \ldots, h_\eta\}$ the set of vertices between the inputs and outputs. The circuit C has a total of $\tilde{m} + \tilde{n} + \eta + 3$ vertices, is acyclic, and apart from the inputs each vertex of D computes one of the following function:

– a constant function ($\mathtt{0}$ or $\mathtt{1}$) from 0 in-neighbor, or
– a NOT function from 1 in-neighbor, or
– an AND or OR function from 2 in-neighbors, or
– only for the outputs: an identity function from 1 in-neighbor.

From D, the reduction consists in constructing the digraph D_C represented in Figure 5. The idea is to encode the circuit C (itself encoding the formula ψ) within simple constraints in disjunctive normal form, and use again the construction from Lemma 5. Furthermore, additional elements (positive loops and null arcs) will enforce the BN to correspond to a valuation on $\lambda_1, \ldots, \lambda_n$, and have two fixed points for each satisfied clause of the formula ψ. As a consequence we will have $\phi(D_C) \geq 2m$ if and only if ψ is satisfiable.

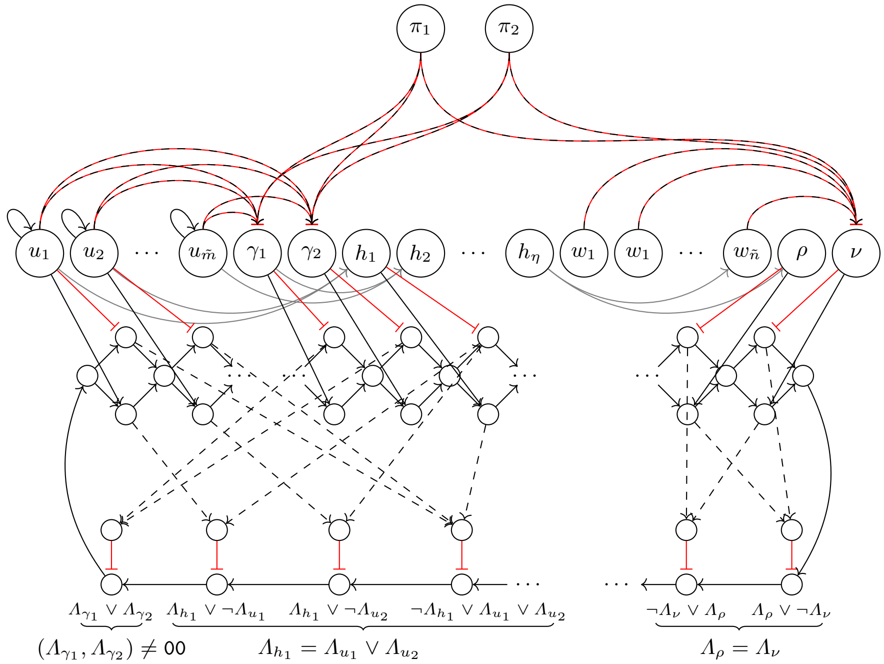


Fig. 5: Digraph D_C encoding the Succint-3SAT instance C, which is a circuit itself encoding a formula ψ (see Figure 4). Null arcs on the top are dashed in black and red. We have $\phi(D) \geq 2m = 2^{\tilde{m}+1}$ if and only if ψ is satisfiable.

Let us define the set of meta-variables (to distinguish them from the variables of ψ)

$$\Lambda = \{\Lambda_i \mid i \in V'\} \text{ with } V' = U \cup \{\gamma_1, \gamma_2\} \cup H \cup \{\rho, \nu\},$$

$U = \{u_1, \ldots, u_{\tilde{m}}\}$, and $H = \{h_1, \ldots, h_\eta\}$. On these meta-variables, we define the 3SAT meta-formula Ψ as follows.

- Add the clause $\Lambda_{\gamma_1} \vee \Lambda_{\gamma_2}$,
- For each $h_i \in H$,
  - if h_i is a constant 0 in circuit C then add the clause $\neg\Lambda_{h_i}$,
  - if h_i is a constant 1 in circuit C then add the clause Λ_{h_i},
  - if h_i is a NOT of h_j in circuit C then add the clauses

$$(\Lambda_{h_i} \vee \Lambda_{h_j}) \wedge (\neg\Lambda_{h_i} \vee \neg\Lambda_{h_j}),$$

- if h_i is an OR of h_j and h_k in circuit C then add the clauses

$$(\Lambda_{h_i} \vee \neg\Lambda_{h_j}) \wedge (\Lambda_{h_i} \vee \neg\Lambda_{h_k}) \wedge (\neg\Lambda_{h_i} \vee \Lambda_{h_j} \vee \Lambda_{h_k}),$$

- if h_i is an AND of h_j and h_k in circuit C then add the clauses

$$(\neg\Lambda_{h_i} \vee \Lambda_{h_j}) \wedge (\neg\Lambda_{h_i} \vee \Lambda_{h_k}) \wedge (\Lambda_{h_i} \vee \neg\Lambda_{h_j} \vee \neg\Lambda_{h_k}).$$

– For each $i \in W \cup \{\rho\}$ of in-neighbor h_j in circuit C, add the clauses

$$(\Lambda_i \vee \neg\Lambda_{h_j}) \wedge (\neg\Lambda_i \vee \Lambda_{h_j}).$$

– Add the clause $(\Lambda_\nu \vee \neg\Lambda_\rho) \wedge (\neg\Lambda_\nu \vee \Lambda_\rho)$.

Now, we construct the digraph D_Ψ corresponding to the meta-formula Ψ on the meta-variable set Λ, as in the proof of Lemma 5. For each $i \in \Lambda$ we simply denote i instead of r_i (the top component in Figure 2).

To get D_C from D_Ψ, we add a positive loop on components of the set U, plus two vertices $\{\pi_1, \pi_2\}$ and the following arcs:

– for $i \in \{1, 2\}$ and $j \in \{\gamma_1, \gamma_2, \nu\}$, an arc from π_i to j labeled 0,
– for $i \in [\tilde{m}]$ and $j \in \{\gamma_1, \gamma_2\}$, an arc from u_i to j labeled 0,
– for $i \in [\tilde{n}]$, an arc from w_i to ν labeled 0.
– for $j \in H \cup W \cup \{\rho\}$, an arc from the in-neighbors of j in D and labeled
  - 1 if j computes the identity, AND or OR function in D, and
  - -1 if j computes the NOT function in D.

The construction of D_C is finished. The purpose of this last part is to enforce that for any $f \in F(D_C)$, there are two fixed points sharing the same value u on components $\{u_1, \ldots, u_{\tilde{m}}\}$ if and only if the corresponding clause μ_u is satisfied by the assignment v given in f_ν (more precisely by the literal given in $f_{\gamma_1}, f_{\gamma_2}$).

We claim that $\phi(D_C) = \alpha + m$, with

$$\alpha = \max_{v:\{\lambda_1,\ldots,\lambda_n\}\to\{\perp,\top\}} |\{\mu_i \mid i \in [m] \text{ such that } v \text{ satisfies } \mu_i\}|.$$

Proving the claim finishes the proof with $k = 2m$, since $\phi(D) \geq 2m$ if and only if ψ is satisfiable.

In one direction, to prove $\phi(D_C) \geq \alpha + m$, let us consider an assignment $v : \{\lambda_1, \ldots, \lambda_n\} \to \{\perp, \top\}$ which satisfies α clauses of ψ. Let us prove the existence of $f \in F(D_C)$ such that $\phi(f) \geq \alpha + m$. The BN f is constructed from v as follows. We set $f_{\pi_1} : x \mapsto \mathtt{0}$ and $f_{\pi_2} : x \mapsto \mathtt{0}$. Note that as a result, each fixed point $x \in \Phi(f)$ respects $x_{\{\pi_1,\pi_2\}} = \mathtt{00}$. Regarding f_ν, we set

$$f_\nu(x) = \begin{cases} x_{\pi_1} \oplus x_{\pi_2} \oplus \bigoplus_{i \in W} x_i & \text{if } x_{\pi_1} = \mathtt{1} \text{ or } x_{\pi_2} = \mathtt{1}, \\ \mathtt{1} & \text{if } x_{\{\pi_1,\pi_2\}} = \mathtt{00} \text{ and } v(\lambda_i) = \top \text{ with } i = x_W, \\ \mathtt{0} & \text{otherwise.} \end{cases}$$

The first part of the definition of f_ν ensures that the arcs between the in-neighbors of ν and ν are all effective. The two last parts enforce that f_ν encodes v. In other words, for any $x \in \Phi(x)$, $f_\nu(x)$ equals $\mathtt{1}$ if and only if $v(\lambda_w) = \top$ with $w = x_W$. We set similarly f_{γ_1} and f_{γ_2}, so that we have the following. For any $x \in \Phi(f)$, $f_{\{\gamma_1,\gamma_2\}}(x) = \gamma$ with γ the position of a first literal satisfying the clause μ_{x_U} with the valuation v (and anything if such a γ does not exist). Furthermore, we set $f_{u_i} : x \mapsto x_{u_i}$ for $i \in [\tilde{m}]$ and each function f_j for $j \in H \cup W \cup \{\rho\}$ computes the same function as in the circuit C. The other local functions f_i for $i \in L \cup \bar{L} \cup P \cup S \cup T$ are identical to those of Lemma 5.

One can check that if two fixed point are identical on the components of U, then they are identical in all V'. Furthermore, the choice of local function f_i we made is similar to Lemma 8. Consequently, we have the following properties.

- Each clause μ_u corresponds to at least a fixed point x^u with $x^u_U = u$.
- Consider the following meta-assignment $v^u : \{\Lambda_1, \ldots, \Lambda_n\} \to \{\bot, \top\}$. We set $v^u(\Lambda_i) = \top$ if and only if $x^u_i = \mathtt{1}$ for any $i \in V'$. The clause μ_u corresponds to two distinct fixed points x^u and y^u if and only if v^u is a valid assignment of Ψ.

Furthermore, for any clause μ_u with $u \in \{\mathtt{0},\mathtt{1}\}^{\tilde{m}}$ such that v satisfies μ_u we have the following properties.

- v^u satisfies the clauses of Λ which prevent Λ_{γ_1} and Λ_{γ_2} to be assigned to $\bot$ together. Indeed, $x^u_{\{\gamma_1,\gamma_2\}}$ encodes the position γ of the first literal in μ_u which satisfies μ_u.
- v^u satisfies all clauses of Ψ which assert that the circuit is correctly computed with u and γ in input.
- v^u satisfies $v^u(\Lambda_\rho) = v^u(\Lambda_v)$. Indeed, the assignment v satisfies the clause μ_u through the variable λ_w (with $w = x^u_W$) of polarity x_ρ. Thus, we have $x^u_\rho = \mathtt{0}$ (resp. $\mathtt{1}$) and $v(\lambda_w) = \bot$ (resp. $\top$) and then $x^u_\nu = \mathtt{0}$ (resp. $\mathtt{1}$). As a result, $v^u(\Lambda_\rho) = v^u(\Lambda_v) = \bot$ (resp. $\top$).

We can conclude that f has two (resp. one) fixed points for each $u \in \{\mathtt{0},\mathtt{1}\}^{\tilde{m}}$ encoding a clause μ_u satisfied (resp. not satisfied) by v, hence a total of $m + \alpha$.

In the other direction, to prove $\phi(D_C) \leq \alpha + m$, let us consider $f \in F(D_C)$ which has $m + \beta$ fixed points. Let us prove the existence of $v : \{\lambda_1, \ldots, \lambda_n\} \to \{\bot, \top\}$ which satisfies at least β clauses of ψ.

The valuation v is given by f_ν as follows, for any $w \in \{\mathtt{0},\mathtt{1}\}^{\tilde{n}}$ encoding a variable λ_w,

$$v(\lambda_i) = \begin{cases} \top & \text{if } r_\nu(f_\nu(x)) = \mathtt{1} \text{ with } x_{\pi_1} = x_{\pi_2} = \mathtt{0} \text{ and } x_W = r_W(w) \\ \bot & \text{otherwise} \end{cases}$$

where we have the following technical adjustment:

$$r_i(x) = \begin{cases} x_i & \text{if } f_{\ell_i} \text{ is the AND function} \\ \neg x_i & \text{otherwise.} \end{cases}$$

As in the proof of Lemma 8, we consider the set A of configurations $a \in \{\mathtt{0},\mathtt{1}\}^{\tilde{m}}$ such that there exist two distinct fixed points $x^a, y^a \in \Phi(f)$ such that $x^a_U = y^a_U = a$. We know that $U \cup \{p_0\}$ is a feedback vertex set of D_C. Therefore, for each $a \in A$ there are at most two distinct fixed points x, y with $x_U = y_U = a$ (one with $x_{p_0} = \mathtt{0}$ and the other with $y_{p_0} = \mathtt{1}$), and as a consequence $|A| = \beta$. Let us see that for each $a \in A$ the fixed points x^a and y^a correspond to a clause μ_a of ψ satisfied by v.

We see that for any $a \in A$ and the two fixed points x^a,y^a, if f_{ℓ_i} is the AND (resp. OR) function, then $y^a_i = \mathtt{0}$ (resp. $\mathtt{1}$) if and only if $\Lambda_i = \bot$ (resp. $\top$). In particular, if we consider that the local function at f_{ℓ_i} is always the AND function, then the r_ν and r_W adjustments disappear from the definition of v. Without loss of generality, we can consider that this is the case.

From the construction of G_Φ as in the proof of Lemmas 5 and 8, two fixed points x^a, y^a with $x^a = y^a = a$ correspond to satisfying Ψ (with truth value of variable Λ_i encoded in the difference of $x^a_{\ell_{\Lambda_i}}$ and $x^a_{\bar{\ell}_{\Lambda_i}}$ along a positive cycle). From the clause $\Lambda_{\gamma_1} \vee \Lambda_{\gamma_2}$ in Ψ we cannot have $x^a_{\gamma_1} = x^a_{\gamma_2} = \mathtt{0}$, hence the components γ_1, γ_2 encode a literal position in $\{1, 2, 3\}$. The rest of the reasoning is symmetric to the other direction: from the constraints of circuit C implemented in the meta-formula Ψ and the equality of components ρ and ν, the pair of fixed points is such that v satisfies clause μ_a with the literal at position encoded in components γ_1, γ_2 (this literal is the variable encoded on W with the polarity encoded in ρ, which is positive if and only if the variable is set to $\top$ in v). As a consequence, v satisfies β clauses of ψ. $\square$