State Complexity of Unary SV-XNFA with Different Acceptance Conditions

Laurette Marais, Lynette Van Zijl

State Complexity of Unary SV-XNFA with Different Acceptance Conditions

Laurette Marais[1,2] and Lynette van Zijl[1]

[1] Department of Computer Science, Stellenbosch University, South Africa
[2] Meraka Institute, CSIR, South Africa

Abstract. Unary self-verifying symmetric difference automata were introduced in [1], with an upper bound of $O(2^n)$ and lower bound of $2^{n-1}-1$ for state complexity. Implicit in the interpretation of self-verifying acceptance for the symmetric difference case was the assumption that no state could be both an accept state and a reject state. We present another interpretation of acceptance more aligned to the equivalence of symmetric difference automata to weighted automata over GF(2), where states that both accept and reject are allowed, and we give a tight bound of $2^{n-1}-1$ for state complexity for both interpretations of acceptance.

1 Introduction

In [1] we showed how the concepts of symmetric difference finite state automata (XNFA) and self-verifying acceptance (SV) could be combined, resulting in self-verifying symmetric difference finite automata (SV-XNFA). We also provided an upper bound of $O(2^n)$ on state complexity for n-state SV-XNFA in the unary case, as well as a lower bound of $2^{n-1}-1$. XNFA are useful in practice, with applications in, for example, cryptography [2], and succinctly recognize groups of languages that cannot be recognized succinctly by NFAs [3]. SV-NFAs are interesting *per se* [4], and so we present a comparison between SV-NFAs and SV-XNFAs.

It is customary for XNFA states to reflect the parity of the symmetric difference operation with the requirement that any state in the equivalent deterministic automaton (XDFA) contain an odd number of final XNFA states [5]. For SV-XNFA, we extended this to both the accepting state set F^a and the rejecting state set F^r, requiring that an SV-XDFA state contain an odd number of either of the two final state sets, but not both. The implicit assumption was that an SV-XNFA state must itself either accept or reject or do neither, which is consistent with self-verification for union automata [4] and automata theory in general, where any particular state usually cannot both accept and reject.

In this paper we examine this implicit assumption more closely. We call the interpretation of SV-XNFA acceptance where it is required that $F^a \cap F^r = \emptyset$ disjunctive acceptance and we define so-called GF(2)-acceptance, where we allow $F^a \cap F^r$ to be non-empty. The result is that a final state may be an accept

state, a reject state, or it may be both, and we show the implications of this interpretation in Section 3. We present various results for SV-XNFA for each of these forms of acceptance, finally showing that $2^{n-1}-1$ is indeed a tight bound for the state complexity of both forms of acceptance.

2 Preliminaries

Definition 1. *An SV-XNFA with disjunctive acceptance is an SV-XNFA as defined in [1], i.e. a 6-tuple $N = (Q, \Sigma, \delta, Q_0, F^a, F^r)$, where Q, Σ, δ and Q_0 are defined as for XNFA, and F^a and F^r are the accept states and reject states, respectively, with the following requirement: for each input string w in Σ^*, there exist an odd number of paths ending in accept states, and zero or an even number of the paths ending in reject states, or vice versa. This is consistent with the parity acceptance typically applied to XNFA. Furthermore, $F^a \cap F^r = \emptyset$.*

The transition function $\delta : Q \times \Sigma \to 2^Q$ (where 2^Q represents the power set over Q) can be extended to strings in the Kleene closure Σ^* of the alphabet:

$$\delta^*(q, w_0 w_1 \dots w_k) = \delta(\delta(\dots \delta(q, w_0), w_1), \dots, w_k).$$

For convenience, we write $\delta(q, w)$ to mean $\delta^*(q, w)$.

The choice of F^a and F^r for a given SV-XNFA N is called an *SV-assignment* of N. An SV-assignment where either F^a or F^r is empty, is called a *trivial SV-assignment*. Otherwise, if both F^a and F^r are nonempty, the SV-assignment is *non-trivial*. An SV-assignment that results in a language that is not the empty language or the universal language is called an *interesting SV-assignment*. For a detailed introduction to unary SV-XNFA, the interested reader is referred to [1].

XNFA have been shown to be equivalent to weighted automata over the finite field (Galois field) of two elements, or GF(2) [5,6]. Let $N = (Q, \Sigma, \delta, Q_0, F)$ be a unary XNFA with n states and $\Sigma = \{a\}$. We can represent the transition function $\delta : Q \times \Sigma \to 2^Q$ as an $n \times n$ matrix M over GF(2) whose (p, q)-th entry represents the weight (1 or 0) of the transition from p to q. Such a matrix has a characteristic polynomial $c(X) = \det(XI - M)$, where I is the identity matrix. Note that in [1] we used column vectors to represent the transitions from one state to another. In this paper we use row vectors as described, because it allows for a more intuitive presentation of the matrix and vector multiplication that follows. However, the results are identical, since any matrix and its transpose have the same characteristic polynomial.

We encode the initial states Q_0 as a vector of length n of elements in GF(2), namely $v(Q_0) = [q_{0_0}\ q_{0_1}\ \dots\ q_{0_{n-1}}]$, where $q_{0_i} = 1$ if $q_i \in Q_0$ and $q_{0_i} = 0$ otherwise. Similarly, we encode the final states as an n-length vector, $v(F) = [q_{F_0}\ q_{F_1}\ \dots\ q_{F_{n-1}}]$. We abuse notation by letting $\delta : Q \times \Sigma \to 2^Q$ (a function to sets of states) and $\delta : Q \times \Sigma \to \mathbb{Z}_2^n$ (a function to vectors of length n over GF(2)) depending on the context. Then the weight of a word w_k of length k is given by

$$\Delta(w_k) = v(Q_0) M^k v(F)^T.$$

In fact, $v(Q_0)M^k$ is a vector that encodes the XNFA states reachable from the initial states after reading k letters, or equivalently, it encodes the XDFA state that is reached from the initial state after reading k letters. That is, $\Delta(w) = \delta(w)v(F)^T$.

The important advantage of this interpretation is the fact that one can perform a change of basis on the transition matrix and initial and final state vectors of an XNFA to produce an equivalent XNFA. This ability is essential in, for example, minimisation algorithms for XNFA [6].

Let $N' = (Q, \Sigma, \delta', Q'_0, F')$ be an XNFA, with transition matrix $M' = A^{-1}MA$ for some non-singular $n \times n$ matrix A, and let Q'_0 and F' be such that $v(Q'_0) = v(Q_0)A$ and $v(F')^T = A^{-1}v(F)^T$. Then

$$\begin{aligned}\Delta'(w_k) &= v(Q'_0)(M'^k)v(F')^T \\ &= v(Q_0)A(A^{-1}MA)^kA^{-1}v(F)^T \\ &= v(Q_0)(M^k)v(F)^T \\ &= \Delta(w_k).\end{aligned}$$

Now, if we require for SV-XNFA that F^a and F^r be disjunct, a similar change of basis where $v(F'^a)^T = A^{-1}v(F^a)^T$ and $v(F'^r)^T = A^{-1}v(F^r)^T$ would not necessarily result in an equivalent SV-XNFA, since the resulting F'^a and F'^r might not be disjunct. Given M' and Q'_0, it might be possible to choose another F'^a and F'^r that are disjunct so that the result is an SV-XNFA, but it is not immediately clear that such a choice would always be possible nor that the language would be preserved [7].

This brings us to the interpretation of SV-XNFA acceptance that excludes the requirement that F^a and F^r be disjunct. The result is that any SV-XNFA state is allowed to be both an accept and a reject state, or one or neither, as long as the SV-condition is met, i.e. that every word is either explicitly accepted by the automaton or explicitly rejected, but not both. We call this GF(2)-acceptance, since it is consistent with the interpretation of XNFA as weighted automata over GF(2) because a change of basis results in an equivalent XNFA (see Section 3).

Definition 2. *An SV-XNFA with GF(2)-acceptance is an SV-XNFA as defined in [1], i.e. a 6-tuple $N = (Q, \Sigma, \delta, Q_0, F^a, F^r)$, where Q, Σ, δ and Q_0 are defined as for XNFA , and F^a and F^r are defined as in the disjunctive acceptance case, but without the requirement that $F^a \cap F^r = \emptyset$.*

Note that an SV-assignment for disjunctive acceptance is also an SV-assignment for GF(2)-acceptance, but the reverse is not necessarily true, since the latter may involve assigning some states to both F^a and F^r.

2.1 Unary XNFA: matrices and polynomials over GF(2)

Unary XNFA have been shown to be equivalent to linear feedback shift registers (LFSRs) [3]. We now give some relevant results from [2] relating LFSRs, and hence unary XNFA, to matrices and polynomials over GF(2).

Any $n \times n$ matrix M over GF(2) has a characteristic polynomial $c(X) = \det(XI - M)$. On the other hand, every polynomial $c(X)$ over GF(2) is the characteristic polynomial of some matrix M of the form shown in Fig. 1. M is

$$M = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & & \dots & 1 \\ c_0 & c_1 & \dots & c_{n-2} & c_{n-1} \end{bmatrix}$$

Fig. 1. Companion matrix of $c(X)$

$$M' = \left[\begin{array}{c|c|c|c} A_1 & 0 & \dots & 0 \\ \hline 0 & A_2 & \dots & 0 \\ \hline \vdots & \vdots & \vdots & \vdots \\ \hline 0 & 0 & \dots & A_m \end{array}\right]$$

Fig. 2. Block diagonal matrix of companion matrices

said to be the companion matrix of $c(X)$. The following theorem further relates matrices and polynomials over GF(2).

Theorem 1. *[2] Every matrix M over GF(2) is similar to a matrix M' of the form shown in Figure 2, where each of the submatrices A_i is a companion matrix of a polynomial that is irreducible over GF(2) or of a power of a polynomial that is irreducible over GF(2), and the 0's are 0 submatrices of appropriate sizes.*

Each $c(X)$ over GF(2) is associated with a certain cycle structure. Specifically, the properties of the characteristic polynomial of a unary XNFA N allow conclusions about the possible length of the cycle of states of its equivalent XDFA N_D (see [1] in particular, as well as for example [2, 3, 8]). The choice of initial states for an XNFA determines which cycle in its polynomial cycle structure is the equivalent XDFA.

We say that a matrix M has an SV-assignment if some XNFA with M as its transition matrix has an SV-assignment.

In the rest of this paper, we consider only unary SV-XNFA with non-singular matrices, whose cycle structures do not include transient heads, i.e. states that are only reached once before a cycle is reached. By Lemma 1 of [1], this means that we only consider matrices with a characteristic polynomial $c(X) = X^n + c_{n-1}X^{n-1} + ... + c_1X + c_0$ that does not have X as a factor, and hence $c_0 = 1$.

2.2 Unary XNFA: linear recurrences over GF(2)

Since the structure of an XDFA is cyclic, for any state d_k of the XDFA that is reached after k letters have been read, there is some integer l so that, if $v(d_k) = v(Q_0)M^k$ for some k, then $v(d_k) = v(Q_0)M^{l+k}$. That is, l is the length of the cycle to which d_k belongs. This means that given any $v(d_k) = v(Q_0)M^k$ for some k, $v(d_{k-i}) = v(Q_0)M^{k-i}$ is well-defined.

We introduce the notion of linear recurrences with respect to XNFA to provide more information about how XDFA states occur together in a cycle. A linear

recurrence over a finite field has a characteristic polynomial [9]. Specifically, the polynomial $c(X) = X^n + c_{n-1}X^{n-1} + ... + c_0$ characterises the linear recurrence $s_t = c_{n-1}s_{t-1} + c_{n-2}s_{t-2} + ... + c_0 s_{t-n}$. Let $c(X)$ be the characteristic polynomial of

1. a transition matrix M for an n-state XNFA N,
2. a linear recurrence over GF(2), namely $s_t = \Sigma_{i=1}^{n} c_{n-i}s_{t-i}$.

Let $\bar{s}_t = [s_{t_0}\ s_{t_1}\ ...\ s_{t_{n-1}}]$ be a vector of length n of elements in GF(2). Then,

$$\begin{aligned}[s_{t_0}\ s_{t_1}\ ...\ s_{t_{n-1}}] = c_{n-1}[s_{t_0-1}\ s_{t_1-1}\ ...\ s_{t_{n-1}-1}] + \\ c_{n-2}[s_{t_0-2}\ s_{t_1-2}\ ...\ s_{t_{n-1}-2}] + ... + \\ c_0[s_{t_0-n}\ s_{t_1-n}\ ...\ s_{t_{n-1}-n}].\end{aligned} \tag{1}$$

That is, $\bar{s}_t = c_{n-1}\bar{s}_{t-1} + c_{n-2}\bar{s}_{t-2} + ... + c_0\bar{s}_{t-n}$.

Let $\bar{s}_0 = v(Q_0)$. The linear recurrence and the behaviour of the XNFA are both characterised by $c(X)$, so $\bar{s}_1 = v(Q_0)M$. In general $\bar{s}_k = v(Q_0)M^k$. We therefore have

$$\begin{aligned}v(d_t) &= v(Q_0)M^t \\ &= \bar{s}_t \\ &= c_{n-1}\bar{s}_{t-1} + c_{n-2}\bar{s}_{t-2} + ... + c_0\bar{s}_{t-n} \\ &= c_{n-1}v(Q_0)M^{t-1} + c_{n-2}v(Q_0)M^{t-2} + ... + c_0 v(Q_0)M^{t-n} \\ &= c_{n-1}v(d_{t-1}) + c_{n-2}v(d_{t-2}) + ... + c_0 v(d_{t-n}).\end{aligned} \tag{2}$$

Therefore, $d_t = \bigoplus_{i=1}^{n} c_{n-i}d_{t-i}$.

Notation

In this paper we let $\bar{s}_i$ refer to either the vector representing some set of states, or the set of states themselves, depending on the context. We use the symbol $\oplus$ and its sigma notation equivalent $\bigoplus$ to denote the boolean XOR operation when applied to boolean ones and zeroes, and the symmetric difference set operation when applied to sets, and specifically sets of states.

3 Main Results

This section presents results on SV-XNFA with both disjunctive acceptance and GF(2)-acceptance. We start by giving an example, to which we will refer back in the rest of the section, as various notions are discussed.

Example 1. Let N be an SV-XNFA with $Q_0 = \{q_0\}$, $F^a = \{q_0, q_1\}$ and $F^r = \{q_2, q_3\}$ and with its transition matrix being the companion matrix M for the polynomial $c(X) = X^4 + X^3 + X^2 + 1$ given in Fig. 1. Let the matrices A and M' (also shown in Fig. 1) be related to M in the sense that $M' = A^{-1}MA$. For

$$M = \begin{bmatrix} 0\,1\,0\,0 \\ 0\,0\,1\,0 \\ 0\,0\,0\,1 \\ 1\,0\,1\,1 \end{bmatrix} \qquad A = \begin{bmatrix} 1\,0\,0\,1 \\ 0\,1\,1\,0 \\ 0\,1\,1\,1 \\ 0\,1\,0\,1 \end{bmatrix} \qquad M' = \begin{bmatrix} 0\,1\,0\,0 \\ 1\,0\,0\,1 \\ 1\,1\,1\,0 \\ 0\,0\,1\,0 \end{bmatrix}$$

Fig. 3. Example 1: matrices M, A and M'

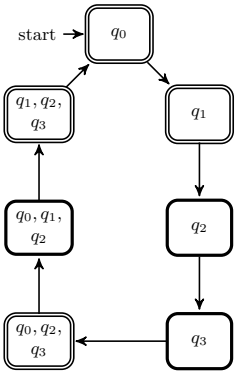

Fig. 4. N_D

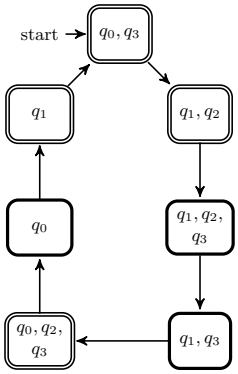

Fig. 5. N'_D

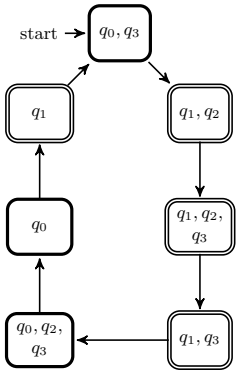

Fig. 6. N''_D

now we only say that N' and N'' are SV-XNFA derived from N (both have M' as their transition matrix), and their equivalent XDFA's are given in Fig. 5 and Fig. 6, respectively, with a double edge indicating an accept state and a thick edge indicating a reject state.

Our first lemma provides a way to determine, given any cycle, whether an SV-assignment is possible.

Lemma 1. *Let $(d_1, d_2, ..., d_m)$ be a cycle representing an XDFA where $d_i \subseteq Q$ for $1 \leq i \leq m$, and Q is the set of states of the equivalent XNFA. Given either disjunctive acceptance or GF(2)-acceptance, the cycle has an SV-assignment if and only if for some some choice of $Q^F \subseteq Q$, where $p_j = 1$ for all $q_j \in Q^F$ and $p_j = 0$ otherwise, then*

$$\bigwedge_{i=1}^{m} \bigoplus_{q_j \in d_i} p_j = 1 \tag{3}$$

Proof. The expression in Equation 3 can only evaluate to 1 if every XDFA state d_i contains an odd number of XNFA states that result in a value of 1. This means that for some choice of Q^F, an odd number of its elements must be present in every XDFA state. For disjunctive acceptance, Q^F represents those XNFA states that must be assigned to either F^a or F^r for the cycle to have an SV-assignment.

For GF(2)-acceptance, Q^F represents those XNFA states that must be assigned to either F^a or F^r but not both for the cycle to have an SV-assignment.

That is, every XDFA state must contain an odd number of states that contribute to the count of either F^a or F^r but not both, so that one but not both of the counts sum to an odd number. □

Since $p \wedge p = p$, we also have the following corollary.

Corollary 1. *For* $0 \leq k \leq m$,

$$\bigwedge_{i=1}^{m} \bigoplus_{q_j \in d_i} p_j = \bigwedge_{i=1}^{m} \bigoplus_{q_j \in d_i} p_j \ \wedge \bigoplus_{q_j \in d_k} p_j. \tag{4}$$

We assign some index $l > m$ *to a repeated state and generalise in the following way for any* $L \subseteq \{m+1, m+2, ...\}$*:*

$$\bigwedge_{i=1}^{m} \bigoplus_{q_j \in d_i} p_j = \bigwedge_{i=1}^{m} \bigoplus_{q_j \in d_i} p_j \ \wedge \bigwedge_{l \in L} \bigoplus_{q_j \in d_l} p_j. \tag{5}$$

Example 2. Consider an XNFA with $Q_0 = \{q_0, q_3\}$ and the transition matrix given in Fig. 1. Then the states of the equivalent XDFA are those shown in the cycles of Fig. 5 and Fig. 6, which leads to the following expression:

$$(p_0 \oplus p_3) \wedge (p_1 \oplus p_2) \wedge (p_1 \oplus p_2 \oplus p_3) \wedge (p_1 \oplus p_3) \wedge (p_0 \oplus p_2 \oplus p_3) \wedge p_0 \wedge p_1. \tag{6}$$

If we choose choose $Q^F = \{q_0, q_1\}$, the expression becomes the following:

$$(1 \oplus 0) \wedge (1 \oplus 0) \wedge (1 \oplus 0 \oplus 0) \wedge (1 \oplus 0) \wedge (1 \oplus 0 \oplus 0) \wedge 1 \wedge 1 = 1. \tag{7}$$

We can assign $F^a = \{q_1\}$ and $F^r = \{q_0\}$, and for now we only note that it is an SV-assignment given either disjunctive acceptance or GF(2)-acceptance. Fig. 6 corresponds to this choice of final states. □

Example 3. Consider again the XNFA and equivalent XDFA in Example 2. The characteristic polynomial of M is $c(X) = X^4 + X^3 + X^2 + 1$, so state transition behaviour is characterised by $s_t = s_{t-1} + s_{t-2} + s_{t-4}$. Let $\bar{s}_t = \{q_1, q_2\}$, then

$$\begin{aligned} \bar{s}_{t-1} + \bar{s}_{t-2} + \bar{s}_{t-4} &= \{q_0, q_3\} \oplus \{q_1\} \oplus \{q_0, q_2, q_3\} \\ &= \{q_1, q_2\} \\ &= \bar{s}_t. \end{aligned}$$

□

The following two lemmas shed more light on linear recurrences in XDFA cycles.

Lemma 2. *The RHS (right hand side) of the linear recurrence* $s_t = c_{n-1}s_{t-1} + c_{n-2}s_{t-2} + ... + c_0 s_{t-n}$ *of a polynomial* $c(X) = X^n + c_{n-1}x^{n-1} + ... + c_1 x + c_0$ *has an odd number of terms if* $X + 1$ *is a factor of* $c(X)$ *and an even number otherwise.*

Lemma 3. *Let d_1 be any state in an XDFA cycle of an equivalent XNFA with state set Q and let the cycle be characterised by the linear recurrence $s_t = \sum_{i=1}^{n} c_{n-i} s_{t-i}$. Let $\sigma_1 = \bigoplus_{q_j \in d_1} p_j$ for some choice of $Q^F \subseteq Q$ so that $p_j = 1$ if $q_j \in Q^F$ and $p_j = 0$ otherwise. Furthermore, let $T \subseteq \{2, ..., n\}$ be the set of indices such that $d_1 = \bigoplus_{k \in T} d_k$. Then*

$$\sigma_1 = \bigoplus_{k \in T} \sigma_k. \tag{8}$$

In the case where the cycle length $m \leq n$, it is possible that $d_{1-i} = d_{1-j}$ for some i, j. We assign to the l-th duplicate of a state d_k (including any occurrences of d_1 itself) the index $lm + k$, referring to it as d_{lm+k}.

Theorem 2. *An XNFA N with characteristic polynomial $c(X) = X^n + c_{n-1}x^{n-1} + ... + c_1 x + c_0$ has no SV-assignment, given either disjunctive acceptance or GF(2)-acceptance, if $X + 1$ is not a factor of $c(X)$.*

Proof. From the discussion in Section 2.2, we know that the state transition behaviour of N is described by $s_t = c_{n-1}s_{t-1} + c_{n-2}s_{t-2} + ... + c_0 s_{t-n}$.

That is, in its equivalent XDFA N_D, each state is the $\oplus$-sum of some number of states in its cycle. Consider any cycle of N_D and let d_1 be any state in the cycle. Let $\mathbb{T} = \{2, ..., n\}$ and let $T_1 \subseteq \mathbb{T}$ be the set of indices so that

$$d_1 = \bigoplus d_i, i \in T_1.$$

If $m > n$, we use Equation 3 from Lemma 1 as well as Lemma 3 to determine if the cycle has an SV-assignment. Since Lemma 1 applies to both disjunctive and GF(2)-acceptance, the rest of the proof applies similarly.

$$\begin{aligned}
\bigwedge_{i=1}^{m} \bigoplus_{q_j \in d_i} p_j &= \bigwedge_{i=1}^{m} \sigma_i \\
&= \sigma_1 \wedge \bigwedge_{i \in T_1} \sigma_i \wedge \bigwedge_{i \in \mathbb{T} \setminus T_1} \sigma_i \\
&= \bigoplus_{i \in T_1} \sigma_i \wedge \bigwedge_{i \in T_1} \sigma_i \wedge \bigwedge_{i \in \mathbb{T} \setminus T_1} \sigma_i.
\end{aligned}$$

If $m \leq n$, we let $K = \{i \in T_1 |\, i > m\}$ and use Equation 5 from Corollary 1 and Lemma 3 in the following way:

$$\begin{aligned}
\bigwedge_{i=1}^{m} \bigoplus_{q_j \in d_i} p_j &= \bigwedge_{i=1}^{m} \bigoplus_{q_j \in d_i} p_j \ \wedge \bigwedge_{i \in K} \bigoplus_{q_j \in d_i} p_j \\
&= \bigwedge_{i=1}^{m} \sigma_i \wedge \bigwedge_{i \in K} \sigma_i \\
&= \sigma_1 \wedge \bigwedge_{i \in T_1} \sigma_i \ \wedge \bigwedge_{i \in \mathbb{T} \setminus T_1} \sigma_i \\
&= \bigoplus_{i \in T_1} \sigma_i \ \wedge \bigwedge_{i \in T_1} \sigma_i \ \wedge \bigwedge_{i \in \mathbb{T} \setminus T_1} \sigma_i.
\end{aligned}$$

In both cases, if $c(X)$ does not have $X + 1$ as a factor, then by Lemma 2, $|T_1|$ is even. Therefore, $\bigoplus_{i \in T_1} \sigma_i \wedge \bigwedge_{i \in T_1} \sigma_i = 0$, and so the cycle does not have an SV-assignment. □

Having shown that a characteristic polynomial with $X + 1$ is a necessary condition for a matrix to have an SV-assignment, we now prepare the ground for showing in Theorem 3 that it is also a sufficient condition. We first determine that performing a change of basis on an SV-XNFA always results in another SV-XNFA, albeit in different ways for disjunctive acceptance and GF(2)-acceptance.

Lemma 4. *Given GF(2)-acceptance, for any n-state XNFA N with transition matrix M, if N has an SV-assignment, then there is an N' with transition matrix M' that is similar to M, so that N' has an SV-assignment and N and N' accept the same language. Hence, if N has an (interesting) SV-assignment, then so does N'.*

Proof. If M' is similar to M, then $M' = A^{-1}MA$ for some non-singular $n \times n$ matrix A. We encode the initial states as the vector $v(Q_0) = [q_{0_0}\ q_{0_1}\ \ldots\ q_{0_{n-1}}]$, where $q_{0_i} = 1$ if $q_{0_i} \in Q_0$ and $q_{0_i} = 0$ otherwise. Similarly, we let $v(F^a) = [q_{a_0}\ q_{a_1}\ \ldots\ q_{a_{n-1}}]$ and $v(F^r) = [q_{r_0}\ q_{r_1}\ \ldots\ q_{r_{n-1}}]$, where q_{a_i} and q_{r_i} indicate membership to F^a and F^r respectively.

We define the following functions, where the SV-constraint is choosing F^a and F^r in such a way that $accept(N, a^k) \neq reject(N, a^k)$ for any k.

$$\begin{aligned}
accept(N, a^k) &= v(Q_0)(M^k)v(F^a)^T \\
reject(N, a^k) &= v(Q_0)(M^k)v(F^r)^T
\end{aligned}$$

Now, we choose the initial states Q'_0, and final states F'^a and F'^r so that $v(Q'_0) = v(Q_0)A$, $v(F'^a)^T = A^{-1}v(F^a)^T$ and $v(F'^r)^T = A^{-1}v(F^r)^T$. Then

$$\begin{aligned}
accept(N', a^k) &= v(Q'_0)(M'^k)v(F'^a)^T \\
&= v(Q_0)A(A^{-1}MA)^k A^{-1}v(F^a)^T \\
&= v(Q_0)(M^k)v(F^a)^T \\
&= accept(N, a^k)
\end{aligned}$$

and

$$\begin{aligned} reject(N', a^k) &= v(Q'_0)(M'^k)v(F'^r)^T \\ &= v(Q_0)A(A^{-1}MA)^k A^{-1}v(F^r)^T \\ &= v(Q_0)(M^k)v(F^r)^T \\ &= reject(N, a^k). \end{aligned}$$

By assumption, F^a and F^r are an (interesting) SV-assignment for N, and so F'^a and F'^r are an (interesting) SV-assignment for N'. Furthermore, N and N' accept the same language. □

Lemma 5. *Given disjunctive acceptance, for any n-state XNFA N with transition matrix M, if N has an SV-assignment, then there is an N'' with transition matrix M' that is similar to M, so that N'' has an SV-assignment, but N and N'' do not necessarily accept the same language.*

Proof. We construct N'' so that $Q''_0 = Q'_0$ as in Lemma 4. However, we let $F''^a = F'^a \setminus F'^r$ and $F''^r = F'^r \setminus F'^a$. That is, F''^a is the set of states that occur in F'^a but not in F'^r and vice versa for F''^r, so that $F''^a \cap F''^r = \emptyset$. Recall from Lemma 1 that for F'^a and F'^r to be an SV-assignment for GF(2)-acceptance, there must be some Q^F so that an odd number of XNFA states in each XDFA state are either accept or reject states but not both. F''^a and F''^r are precisely those states, and so are an SV-assignment for disjunctive acceptance.

However, it is possible that $F'^a \subset F'^r$ or vice versa, and so it is possible that F''^a or F''^r is empty even if F'^a and F'^r are non-empty. So although F''^a and F''^r are an SV-assignment, clearly N'' does not necessarily accept the same language as N. □

Example 4. Let N be the SV-XNFA with matrix given in Fig. 1. Then the equivalent XDFA N_D is the cycle as shown in Fig. 4. Note that in this cycle, both disjunctive acceptance and GF(2)-acceptance place the same constraints on possible SV-assignments, since the XNFA states each appear alone in XDFA states and therefore must accept or reject but cannot do both.

We use non-singular matrix A as shown in Fig. 1, and we perform two changes of basis: as described in Lemma 4 to get an XNFA N', and as described in Lemma 5 to get an XNFA N''_D. Both N' and N'' have transition matrix M' (Fig. 1). The equivalent XDFA N'_D with GF(2)-acceptance is the cycle as shown in Fig. 5, with $F'^a = \{q_1, q_3\}$ and $F'^r = \{q_0, q_3\}$. Note, for example, that the state $\{q_0, q_2, q_3\}$ accepts, because it contains an odd number of accept states, i.e. q_3, and an even number of reject states, i.e. q_0 and q_3. The XDFA N''_D with disjunctive acceptance is shown in Fig. 6, with $F''^a = \{q_1\}$ and $F''^r = \{q_0\}$.

The following lemma asserts the existence of SV-assignments for certain matrices.

Lemma 6. *Any matrix M that is a block diagonal matrix of companion matrices, with characteristic polynomial $c(X) = (X+1)\phi(X)$, has an SV-assignment, given either disjunctive or GF(2)-acceptance.*

Theorem 3. *Given either disjunctive acceptance or GF(2)-acceptance, any matrix M with characteristic polynomial $c(X) = (X+1)\phi(X)$ has an SV-assignment.*

Proof. By Theorem 1, M is similar to some block diagonal matrix M' with the companion matrices of factors of $c(X)$ on the diagonal. By Lemma 6, M' has an SV-assignment given either disjunctive acceptance or GF(2)-acceptance. Therefore, by Lemma 4 M has an SV-assignment given GF(2)-acceptance, and by Lemma 5 M has an SV-assignment given disjunctive acceptance. □

The following theorem follows directly from Theorems 2 and 3.

Theorem 4. *Any matrix M has an SV-assignment given either disjunctive acceptance or GF(2)-acceptance, if and only if its characteristic polynomial has $X + 1$ as a factor.*

Along with Theorem 4, Lemma 7 and Theorem 5 that follow provide the grounds for concluding that $2^{n-1} - 1$ is a tight bound on the state complexity of unary SV-XNFA for both disjunctive acceptance and GF(2)-acceptance.

Lemma 7. *For an XNFA with a characteristic polynomial $c(X)$ with degree n that has $X + 1$ as a factor, the longest possible cycle has length $2^{n-1} - 1$.*

Proof. Proof. Suppose $c(X)$ has two irreducible factors, $\phi_1 = X + 1$ and ϕ_2, where ϕ_2 is an irreducible polynomial with degree $n - 1$. By Theorem 1 of [1], if ϕ_2 is primitive it has a single cycle of length $2^{n-1} - 1$, and together with $X + 1$ induces a cycle for $c(X)$ of the same length. If it is non-primitive it has cycles of length b where b is a factor of $2^{n-1} - 1$, inducing cycles of length b for $c(X)$ together with $X + 1$. Hence, the maximum cycle length is $2^{n-1} - 1$.

Now suppose that $c(X)$ has three irreducible factors, $\phi_1 = X+1$, ϕ_2 of degree $k \leq n-2$ and ϕ_3 of degree $n-k-1$, with $k > n-k-1$. Cycles of $c(X)$ induced together with $X+1$ can only produce at most cycles of length $2^k - 1 < 2^{n-1} - 1$. Consider the cycle induced by ϕ_2 and ϕ_3. Since it will have greatest possible length if 2^k and 2^{n-k-1} are relatively prime, we assume this to be the case. The cycle induced has length $lcm(2^k - 1, 2^{n-k-1} - 1) = (2^k - 1) * (2^{n-k-1} - 1)$. That is,

$$\begin{aligned}(2^k - 1) * (2^{n-k-1} - 1) &= 2^{n-1} - 2^k - 2^{n-k-1} - 1 \\ &< 2^{n-1} - 1.\end{aligned}$$

Cycles of $c(X)$ are induced by pairs of factors of $c(X)$, and so if $c(X)$ had more irreducible factors, they would have smaller degree and so would induce even shorter cycles. Therefore, $2^{n-1} - 1$ is the longest possible cycle for a polynomial $c(X)$ of degree n that has $X + 1$ as a factor. □

Theorem 5. *Given either disjunctive acceptance or GF(2)-acceptance, for any $n \geq 2$, there is a language $\mathcal{L}_n$ so that some n-state SV-XNFA accepts $\mathcal{L}_n$ and the minimal SV-XDFA that accepts $\mathcal{L}_n$ has $2^{n-1} - 1$ states.*

Proof. Theorem 7 of [1] gives a proof of the statement with regards to disjunctive acceptance. Since any SV-assignment for disjunctive acceptance is also an SV-assignment for GF(2)-acceptance, it is also a proof for the statement with regards to GF(2)-acceptance. □

4 Conclusion

We have shown a close similarity between SV-XNFA with two different acceptance conditions, namely disjunctive acceptance and GF(2)-acceptance. In particular, they have the same state complexity bound of $2^{n-1} - 1$. Disjunctive acceptance shares a typical requirement of most other finite state automata, i.e. that a state cannot both accept and reject. However, for self-verification in unary XNFA, this removes the equivalence known between XNFA and weighted automata over GF(2), since a so-called change of basis does not preserve the language. This has implications for operations such as minimisation, which depend upon it [6]. GF(2)-acceptance does preserve the equivalence, but results in the need for SV-XNFA states that both accept and reject. Whereas for disjunctive acceptance, neutral states are non-final, since they neither accept nor reject, GF(2)-acceptance introduces the notion of neutral final states that both accept and reject. While this is perhaps counter-intuitive, it allows for SV-XNFA that behave more predictably.

References

1. Marais, L., van Zijl, L.: Unary Self-verifying Symmetric Difference Automata. In: Descriptional Complexity of Formal Systems: 18th IFIP WG 1.2 International Conference, DCFS 2016, Bucharest, Romania, July 5-8, 2016. Proceedings. Springer International Publishing (2016) 180–191
2. Stone, H.S.: Discrete Mathematical Structures and their Applications. Science Research Associates Chicago (1973)
3. Van Zijl, L.: Nondeterminism and Succinctly Representable Regular Languages. In: Proceedings of the 2002 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists. SAICSIT '02, Republic of South Africa, South African Institute for Computer Scientists and Information Technologists (2002) 212–223
4. Jirásková, G., Pighizzini, G.: Optimal Simulation of Self-verifying Automata by Deterministic Automata. Information and Computation **209**(3) (2011) 528 – 535 Special Issue: 3rd International Conference on Language and Automata Theory and Applications (LATA 2009).
5. Vuillemin, J., Gama, N.: Compact Normal Form for Regular Languages as Xor Automata. In: Implementation and Application of Automata: 14th International Conference, CIAA 2009, Sydney, Australia, July 14-17, 2009. Proceedings. Springer Berlin Heidelberg, Berlin, Heidelberg (2009) 24–33
6. Van der Merwe, B., Tamm, H., Van Zijl, L.: Minimal DFA for Symmetric Difference NFA. In: Descriptional Complexity of Formal Systems: 14th International Workshop, DCFS 2012, Braga, Portugal, July 23-25, 2012. Proceedings. Springer Berlin Heidelberg, Berlin, Heidelberg (2012) 307–318
7. Van der Merwe, B. private communication (2017)
8. Dornhoff, L.L., Hohn, F.E.: Applied Modern Algebra. Macmillan Publishing Co., Inc. (1978)
9. McEliece, R.J.: Finite Fields for Computer Scientists and Engineers. Volume 23. Springer Science & Business Media (1987)