Embeddings of local automata

Marie-Pierre Béal, Sylvain Lombardy, Dominique Perrin

Embeddings of local automata

Marie-Pierre Béal
Université Paris-Est
Institut Gaspard-Monge
77454 Marne-la-Vallée, France
beal@univ-mlv.fr

Sylvain Lombardy
Université Paris-Est
Institut Gaspard-Monge
77454 Marne-la-Vallée, France
lombardy@univ-mlv.fr

Dominique Perrin
Université Paris-Est
Institut Gaspard-Monge
77454 Marne-la-Vallée, France
perrin@univ-mlv.fr

***Abstract*— A local automaton is by definition such that a bounded information about the past and the future is enough to determine the present state. Due to this synchronization property, these automata play an important role for coding purposes. We prove that any irreducible local automaton is contained in a complete one. The proof uses a result from symbolic dynamics due to M. Nasu called the masking lemma. A consequence of this result in the theory of variable length codes is that any locally parsable regular code is included in a maximal one with the same synchronisation delay.**

I. Introduction

The problem of embedding a finite automaton into one having specified properties is an old one in automata theory (see [10] for example). We will consider here the embedding of a local automaton into a complete one.

The embedding of an unambiguous irreducible automaton into a complete one has been solved in [1] using a construction due to Ehrenfeucht and Rosenberg [4]. This result has a formulation in terms of codes. Indeed, the stabilizer of a state in an unambiguous automaton is a free submonoid. Thus the above embedding implies that any regular code is contained in a maximal one.

A local automaton is by definition such that a bounded information about the past and the future is enough to determine the present state. Formally there are integers λ, ρ such that for any pair of paths $p \xrightarrow{u} q \xrightarrow{v} r$ and $p' \xrightarrow{u} q' \xrightarrow{v} r'$ having the same label with $|u| = \lambda$ and $|v| = \rho$ one has $q = q'$. A subautomaton of a local automaton is obviously still local but conversely it is not clear how one can add transitions or states to a local automaton until it is complete, in the sense that any word on the underlying alphabet is the label of a path.

The problem of completing local automata has already been considered in [7], where a method is given which allows one to complete a local deterministic automaton when it is possible. We prove here that any local automaton is always contained in a complete one. Our proof relies on a result known as Nasu's masking lemma [9].

The stabilizer of a state in a local automaton is the star of a *locally parsable code* (also called codes with *finite synchronization delay*). The previous result gives an alternative proof of a result of Bruyère [3] according to which any locally parsable regular code is included in a maximal one.

In Section II, we introduce some definitions on automata equivalences. We prove our main result in Section III. We conclude with the application to locally parsable codes in Section IV.

II. Equivalences of automata

We denote by $\mathcal{A} = (Q, E)$ a finite automaton on the alphabet A with Q as set of states and $E \subset Q \times A \times Q$ as set of edges. We consider all states as both initial and terminal. The automaton is said to be *complete* (with respect to the alphabet A) if for any word $w \in A^*$ there exists a path labelled w. It is said to be *unambiguous* if for any $p, q \in Q$ and w in A^* there is at most one path from p to q labelled w. It is said to be *irreducible* if for any $p, q \in Q$ there is at least one word w such that there is a path from p to q labelled w.

An automaton is said to be *essential* if any state has at least one incoming edge and at least one outgoing edge. Clearly an irreducible automaton is essential.

An automaton $\mathcal{A} = (P, E)$ on the alphabet A is a *subautomaton* of an automaton $\mathcal{B} = (Q, F)$ on the alphabet A, if $P \subset Q$ and $E \subseteq F \cap (P \times A \times P)$.

An automaton is *deterministic* if for each state p and each letter a there is at most one edge labelled a going out of p.

The *transition matrix* of an automaton $\mathcal{A} = (Q, E)$ is the $Q \times Q$ matrix with elements in the set $\mathfrak{P}(A)$ of subsets of A defined for $p, q \in Q$ by

$$M_{p,q} = \{a \in A \mid \text{ there is an edge } (p, a, q) \text{ in } E\}.$$

The elements of M can be considered as elements of the semiring $\mathfrak{P}(A^*)$ of subsets of A^*, where 0 is the empty set and 1 is the set containing the empty word. Such matrices can therefore be multiplied.

Let $\mathcal{A} = (P, E)$ and $\mathcal{B} = (Q, F)$ be two automata on the alphabet A. Let M be the transition matrix of $\mathcal{A}$ and let N be the transition matrix of $\mathcal{B}$. We say that $\mathcal{A}$ and $\mathcal{B}$ are *elementary equivalent* if there exist a $(P \times Q)$-matrix R and a $(Q \times P)$-matrix S both with elements in $\mathfrak{P}(A) \cup \{1\}$, where 1 is the empty word, such that $M = RS$ and $N = SR$. We also say that $\mathcal{A}$ is (R, S)-elementary equivalent to $\mathcal{B}$.

This notion is classical in symbolic dynamics. It is usually formulated for subshifts of finite type (see [6] p. 225). Our definition is a particular case of the notion of symbolic elementary equivalence for sofic shifts introduced in [8].

For a $P \times Q$-matrix R with elements in $\mathfrak{P}(A) \cup \{1\}$, we say that a triple $(p, a, q) \in P \times (A \cup \{1\}) \times Q$ is an R-edge when $a \in R_{p,q}$. Thus when $\mathcal{A}$ is (R, S)-elementary equivalent to $\mathcal{B}$, each edge of $\mathcal{A}$ is the sequence of an R-edge and a consecutive S-edge, and each edge of $\mathcal{B}$ is a sequence of an S-edge and a consecutive R-edge. The R-edges and S-edges can be considered as edges of a bipartite graph called the *auxiliary graph* of the equivalence (it is called a *bipartite code* in [8] and [6, p. 355]).

Example 1: Let $\mathcal{A}$ and $\mathcal{B}$ be the automata represented on Figure 1.

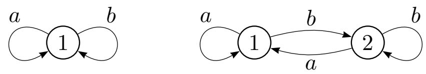

Fig. 1. Two elementary equivalent automata $\mathcal{A}$ (on the left) and $\mathcal{B}$ (on the right).

The transition matrices M and N of $\mathcal{A}$ and $\mathcal{B}$ and matrices R, S such that $M = RS$ and $N = SR$ are given by

$$M = \begin{bmatrix} a+b \end{bmatrix}, \quad N = \begin{bmatrix} a & b \\ a & b \end{bmatrix}, \quad R = \begin{bmatrix} a & b \end{bmatrix}, \quad S = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Thus $\mathcal{A}$ and $\mathcal{B}$ are elementary equivalent. The auxiliary graph is shown in Figure 2 (with continuous lines).

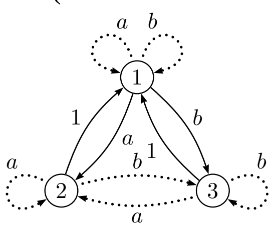

Fig. 2. The auxiliary graph (with continuous edges).

Let $\mathcal{A} = (Q, E)$ be an automaton. The *input* of a state $q \in Q$ is the set of pairs (p, a) such that one has $p \xrightarrow{a} q$. Its *output* is the set of pairs (a, r) such that one has $q \xrightarrow{a} r$.

An *input merge equivalence* is an equivalence on the set Q of states of $\mathcal{A}$ such that for any pair p, p' of equivalent states, any letter a and any state q, one has

(i) $p \xrightarrow{a} q$ if and only if $p' \xrightarrow{a} q$,
(ii) $q \xrightarrow{a} p$, $q \xrightarrow{a} p'$ implies $p = p'$.

Thus, in such an equivalence, two equivalent and distinct states have the same output and disjoint inputs.

The quotient of $\mathcal{A} = (Q, E)$ by such an equivalence is the automaton $\overline{\mathcal{A}} = (\overline{Q}, \overline{E})$ with states the set $\overline{Q}$ of equivalence classes, and edges the induced edges on the classes, *i.e.* $(p, a, q) \in E$ for some $p, q \in Q$ implies $(\overline{p}, a, \overline{q})$ is in $\overline{E}$.

Thus, in the quotient the output of a class of states is the common output of its elements and its input is the union, or merge, of the inputs of its elements (whence the name of an input merge equivalence). We say that $\overline{\mathcal{A}}$ is obtained from $\mathcal{A}$ by an input merge. We also say that $\mathcal{A}$ is obtained from $\overline{\mathcal{A}}$ by an *input split* .

Note that if $\mathcal{A}$ is an unambiguous essential automaton, there is a largest input merge equivalence. It is defined by $q \equiv q'$ if q and q' have the same output. Indeed, since $\mathcal{A}$ is essential, if $q \equiv q'$, they have the same non-empty output. Since it is unambiguous, they have disjoint inputs.

Example 2: Let $\mathcal{A}$, $\mathcal{B}$ be the automata of Example 1. The automaton $\mathcal{A}$ is obtained from $\mathcal{B}$ by an input merge.

An *output merge equivalence* on the set of states of an automaton $\mathcal{A}$ is defined symmetrically. It is an equivalence on the set Q of states such that equivalent and distinct states have the same input and disjoint outputs. The quotient $\overline{\mathcal{A}}$ is defined in the same way by merging the outputs of the elements of a class. We say that $\overline{\mathcal{A}}$ is obtained from $\mathcal{A}$ by an output merge. We also say that $\mathcal{A}$ is obtained from $\overline{\mathcal{A}}$ by an *output split* .

Observe that the quotient of an automaton $\mathcal{A}$ by an input or an output merge equivalence is unambiguous if and only if $\mathcal{A}$ is unambiguous.

The notions of input and output merge equivalence are classical in symbolic dynamics. They are usually formulated for subshifts of finite type (see [6, p. 225]). The extension of the definitions to sofic shifts is due to Nasu [8]; see also [5].

The following result is an element of William's Classification Theorem (see [6]).

Proposition 3: If the automaton $\mathcal{A}$ is obtained from the automaton $\mathcal{B}$ by an input (or output) merge, then $\mathcal{A}$ and $\mathcal{B}$ are elementary equivalent.

Proof: We treat the case of an input merge. Let $\mathcal{B} = (Q, E)$ and $\mathcal{A} = (\overline{Q}, \overline{E})$. Let M and N be the transition matrices of $\mathcal{A}$ and $\mathcal{B}$. Let R be the $\overline{Q} \times Q$-matrix defined for $p, q \in Q$ by

$$R_{\bar{p},q} = \{a \in A \mid \text{ there is an edge } (p, a, q) \text{ in } E\}$$

(note that R is well-defined because of the definition of an input merge). Let S be the $Q \times \overline{Q}$-matrix defined by

$$S_{p,\bar{q}} = \begin{cases} 1 & \text{if } \bar{p} = \bar{q} \\ 0 & \text{otherwise} \end{cases}$$

Then $M = RS$ and $N = SR$. Thus $\mathcal{A}$ and $\mathcal{B}$ are elementary equivalent. ■

Proposition 4: Let $\mathcal{A}$ and $\mathcal{B}$ be two irreducible (R, S)-elementary equivalent automata. Then, either R has elements in $\mathfrak{P}(A)$ and S in $\{0, 1\}$ or conversely.

Proof: Set $\mathcal{A} = (P, E)$ and $\mathcal{B} = (Q, F)$. Suppose that there are $p \in P$, $a \in A$, and $q \in Q$ such that $a \in R_{pq}$. Let $p' \in P$ and $q' \in Q$. We denote $s = S_{q'p'}$. Let us show that, if $s \neq 0$ then $s = 1$. Since $\mathcal{B}$ is irreducible, there is path $q \xrightarrow{u} q'$ labelled by $u = a_1 \ldots a_n$. We have $a_i = s_i r_i$, for $1 \leq i \leq n$, with $r_i \in R_{p_i q_i}$, and $s_i \in S_{q_{i-1} p_i}$, and thus the path

$$p \xrightarrow{a} q_0 \xrightarrow{s_1} p_1 \xrightarrow{r_1} q_1 \xrightarrow{s_2} \cdots \xrightarrow{s_n} p_n \xrightarrow{r_n} q_n \xrightarrow{s} p',$$

with $q_0 = q$ and $q_n = q'$. Since as_1 is the label of an edge in E, $s_1 = 1$. Since $s_1 r_1$ is the label of an edge in F, $r_1 \in A$ and so on. Finally $r_n \in A$ and thus $s = 1$.

This shows that if one entry of R contains an element of A then all the entries of S are 0 or 1. Furthermore, if all the entries of S are 0 or 1, then all non-null entries of R are in $\mathfrak{P}(A)$. The proof of the symmetrical case is similar. ■

We denote by $L(\mathcal{A})$ the set of words labelling a finite path in $\mathcal{A}$. Two automata $\mathcal{A}$ and $\mathcal{B}$ are said to be *equivalent* if $L(\mathcal{A}) = L(\mathcal{B})$.

We say that two automata $\mathcal{A}$ and $\mathcal{B}$ are *strongly equivalent* if there is a sequence $\mathcal{A}_0, \mathcal{A}_1, \ldots, \mathcal{A}_n$ of automata such that $\mathcal{A}_0 = \mathcal{A}$, $\mathcal{A}_n = \mathcal{B}$ and $\mathcal{A}_i$ is elementary equivalent to $\mathcal{A}_{i+1}$ for $1 \leq i \leq n-1$.

We state without proof the following Proposition.

Proposition 5: Two strongly equivalent essential automata are equivalent.

We will now describe a construction due to Nasu (see [6, p. 354], Lemma 10.2.3). It associates to automata $\mathcal{A}_1, \mathcal{A}_2, \mathcal{B}_1$ such that $\mathcal{A}_1$ is elementary equivalent to $\mathcal{A}_2$ and $\mathcal{A}_1$ is a subautomaton of $\mathcal{B}_1$, an automaton $\mathcal{B}_2$ elementary equivalent to $\mathcal{B}_1$ and such that $\mathcal{A}_2$ is a subautomaton of $\mathcal{B}_2$. The automaton $\mathcal{B}_2$ is called the *Nasu embedding* of $\mathcal{A}_2$ with respect to $\mathcal{A}_1, \mathcal{B}_1$. The intuitive idea behind the construction is to cut the edges of $\mathcal{B}_1$ in to halves and to create a new state in the middle which is a state of $\mathcal{B}_2$.

Set $\mathcal{A}_1 = (P_1, E_1)$, $\mathcal{A}_2 = (P_2, E_2)$ and $\mathcal{B}_1 = (Q_1, F_1)$. Let R, S be matrices such that RS and SR are respectively the transition matrices of $\mathcal{A}_1$ and $\mathcal{A}_2$. Let $Q_2 = P_2 \cup (F_1 \setminus E_1)$. We define a $Q_1 \times Q_2$-matrix $\widehat{R}$ and a $Q_2 \times Q_1$-matrix $\widehat{S}$ as follows.

Suppose first that R has $0, 1$ entries and S entries in $\mathfrak{P}(A)$. Then, for $q_1 \in Q_1$ and $q_2 \in Q_2$, we define

$$\widehat{R}_{q_1,q_2} = \begin{cases} R_{q_1,q_2} & \text{if } q_1 \in P_1, q_2 \in P_2 \\ 1 & \text{if } q_2 \in F_1 \setminus E_1 \text{ and } o(q_2) = q_1 \\ 0 & \text{otherwise} \end{cases}$$

and

$$\widehat{S}_{q_2,q_1} = S_{q_2,q_1} \cup \{l(q_2) \mid q_2 \in F_1 \setminus E_1,\ e(q_2) = q_1\},$$

where $o(q_2)$ denotes the origin of q_2, $l(q_2)$ its label, and $e(q_2)$ its end.

An illustration of this construction is given in Example 7.

In the case where R has entries in $\mathfrak{P}(A)$ and S has $0, 1$ entries, we define

$$\widehat{R}_{q_1,q_2} = R_{q_1,q_2} \cup \{a \in A \mid q_2 \in F_1 \setminus E_1,\ o(q_2) = q_1,\ l(q_2) = a\}$$

and

$$\widehat{S}_{q_2,q_1} = \begin{cases} S_{q_2,q_1} & \text{if } q_1 \in P_1, q_2 \in P_2 \\ 1 & \text{if } q_2 \in F_1 \setminus E_1 \text{ and } e(q_2) = q_1 \\ 0 & \text{otherwise .} \end{cases}$$

It is clear that in both cases, $\widehat{R}\widehat{S}$ is the transition matrix of $\mathcal{B}_1$. The automaton $\mathcal{B}_2 = (Q_2, F_2)$ is defined by its transition matrix which is $\widehat{S}\widehat{R}$. To sum up, we have the following statement.

Proposition 6: Let $\mathcal{A}_1, \mathcal{A}_2, \mathcal{B}_1$ be automata such that $\mathcal{A}_1$ is a subautomaton of $\mathcal{B}_1$ and $\mathcal{A}_1$ is elementary equivalent to $\mathcal{A}_2$. Let $\mathcal{B}_2$ be the Nasu embedding of $\mathcal{A}_2$ with respect to $\mathcal{A}_1, \mathcal{B}_1$. Then $\mathcal{A}_2$ is a subautomaton of $\mathcal{B}_2$ and $\mathcal{B}_1$ is elementary equivalent to $\mathcal{B}_2$. Moreover, if $\mathcal{A}_2$ and $\mathcal{B}_1$ are irreducible (resp. essential), then $\mathcal{B}_2$ is irreducible (resp. essential).

Example 7: Figure 3 represents the Nasu embedding $\mathcal{B}_2$ of $\mathcal{A}_2$ with respect to $\mathcal{A}_1, \mathcal{B}_1$. The automata $\mathcal{A}_1$ and $\mathcal{A}_2$ are the elementary equivalent automata of Example 1. If A_i (resp. B_i) denotes the transition matrix of the automaton $\mathcal{A}_i$ (resp. $\mathcal{B}_i$) of Figure 3, we have

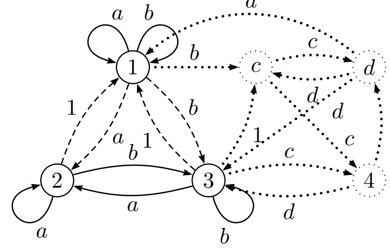

Fig. 3. The Nasu embedding $\mathcal{B}_2$ of $\mathcal{A}_2$ with respect to $\mathcal{A}_1, \mathcal{B}_1$. The automata $\mathcal{A}_1 = (P_1, E_1)$ with $P_1 = \{2, 3\}$ and $\mathcal{A}_2 = (P_2, E_2)$ with $P_2 = \{1\}$ are represented with continuous edges. We have $\mathcal{B}_1 = (Q_1, F_1)$ with $Q_1 = \{2, 3, 4\}$ and $\mathcal{B}_2 = (Q_2, F_2)$ with $Q_2 = \{1, c, d\}$. The automaton $\mathcal{B}_2$ has two additional states c, d corresponding to the additional edges $(3, c, 4)$ and $(4, d, 3)$ of $\mathcal{B}_1$. The transitions defined by R and S are represented with dashed edges. Additional states and edges are represented with dots.

$$A_1 = \begin{bmatrix} a & b \\ a & b \end{bmatrix} = RS = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} a & b \end{bmatrix},\ A_2 = \begin{bmatrix} a+b \end{bmatrix} = SR.$$

$$B_1 = \begin{bmatrix} a & b & 0 \\ a & b & c \\ 0 & d & 0 \end{bmatrix} = \widehat{R}\widehat{S} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a & b & 0 \\ 0 & 0 & c \\ 0 & d & 0 \end{bmatrix},$$

$$B_2 = \begin{bmatrix} a+b & b & 0 \\ 0 & 0 & c \\ d & d & 0 \end{bmatrix} = \widehat{S}\widehat{R} = \begin{bmatrix} a & b & 0 \\ 0 & 0 & c \\ 0 & d & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

III. Local automata

An automaton is said to be (λ, ρ)-*local* if for all pairs of paths $p \xrightarrow{u} q \xrightarrow{v} r$ and $p' \xrightarrow{u} q' \xrightarrow{v} r'$ with $|u| = \lambda$ and $|v| = \rho$, one has $q = q'$. The automaton is said to be *local* if it is (λ, ρ)-local for some $\lambda, \rho \geq 0$. An automaton which has a single state is $(0, 0)$-local.

Thus in a local automaton, one can recover a biinfinite path from its label using a sliding window of fixed size.

A word w is said to be a *constant* for an automaton $\mathcal{A}$ if for two paths $p \xrightarrow{w} q$ and $p' \xrightarrow{w} q'$ with label w, one has also $p \xrightarrow{w} q'$ and $p' \xrightarrow{w} q$. The empty word is a constant only when the automaton has a single state.

Note that if $\mathcal{A}$ is a (λ, ρ)-local automaton, then an input split of $\mathcal{A}$ is a $(\lambda+1, \rho)$-local automaton, and an output split of $\mathcal{A}$ is a $(\lambda, \rho+1)$-local automaton.

An automaton is said to have *order* n if any word of length n is a constant. Note that a (λ, ρ)-local automaton has order $\lambda + \rho$. The minimal order can however be strictly less than $\lambda+\rho$. An unambiguous automaton with order n is (n, n)-local.

Let $\lambda, \rho \geq 0$. The *free* (λ, ρ)-*local* automaton is the automaton with set of states consisting of pairs (u, v) of words, with u of length λ and v of length ρ, and edges the triples $((u, v), a, (u', v'))$ such that there are letters b, c with $uvc = bu'v'$ and a is the first letter of vc. It is clear that this automaton is (λ, ρ)-local and complete.

The free $(n, 0)$-local automaton is usually known as the *de Bruijn automaton* of order n.

Lemma 8: The free (λ, ρ)-local automaton has order $\max(\lambda, \rho)$.

Proof: Set $n = \max(\lambda, \rho)$. Let $\mathcal{F}_{(\lambda,\rho)}$ be the free (λ, ρ)-local automaton. Let $(u, v) \xrightarrow{w} (x, y)$ and $(u', v') \xrightarrow{w} (x', y')$ be two paths labelled w of length n in $\mathcal{F}_{(\lambda,\rho)}$, with u, u', x, x' of length λ and v, v', y, y' of length ρ. Since $\lambda, \rho \le n$, v, v' are prefixes of w and x, x' are suffixes of w. Hence $v = v'$ and $x = x'$ and there are paths $(u, v) \xrightarrow{w} (x, y')$ and $(u', v) \xrightarrow{w} (x, y)$ in $\mathcal{F}_{(\lambda,\rho)}$. This shows that $\mathcal{F}_{(\lambda,\rho)}$ has order n. ■

Example 9: The free $(1, 1)$-local automaton on the alphabet $\{a, b\}$ is represented on Figure 4. The label of an edge going out of a state is its second letter.

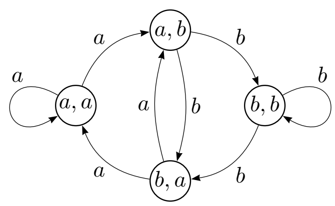

Fig. 4. The free $(1, 1)$-local automaton. It has order 1.

The following result shows that any (λ, ρ)-local automaton is strongly equivalent to a subautomaton of the free (λ, ρ)-local automaton.

Proposition 10: If $\mathcal{A}$ is (λ, ρ)-local automaton, there is a sequence $\mathcal{A}_0, \mathcal{A}_1, \ldots, \mathcal{A}_{\lambda+\rho}$ of (λ, ρ)-local automata such that

(i) $\mathcal{A}_0 = \mathcal{A}$.
(ii) For $i = 0, \ldots, \lambda - 1$, $\mathcal{A}_i$ is obtained from $\mathcal{A}_{i+1}$ by an input merge.
(iii) For $i = \lambda, \ldots, \lambda + \rho - 1$, $\mathcal{A}_i$ is obtained from $\mathcal{A}_{i+1}$ by an output merge.
(iv) $\mathcal{A}_{\lambda+\rho}$ is a subautomaton of the free (λ, ρ)-local automaton.

Proof: Let $\mathcal{A} = (P, E)$. We define for $1 \le i \le \lambda$, $\mathcal{A}_i = (P_i, E_i)$ where P_i is the set of pairs (u, q) with $u \in A^i$ and $q \in P$ such that there is a path labelled u leading to q in $\mathcal{A}$. There is an edge labelled a from (u, p) to (v, q) in $\mathcal{A}_i$ if $u = bu'$, $v = u'a$ for $b \in A$ and $(p, a, q) \in E$. For $j = 1, \ldots, \rho$, let $\mathcal{A}_{\lambda+j} = (P_{\lambda+j}, E_{\lambda+j})$ where $P_{\lambda+j}$ is the set of triples $(u, p, v) \in A^\lambda \times P \times A^j$ such that there is a path labelled u leading to p and a path labelled v leaving p in $\mathcal{A}$. There is an edge labelled a from (u, p, v) to (w, q, t) if and only if $u = bu'$, $w = u'a$, $v = av'$, $t = v'c$ and $(p, a, q) \in E$.

For $0 \le i \le \lambda - 1$, the equivalence θ_{i+1} on P_{i+1} defined by $(u, p) \equiv (u', p')$ if $p = p'$ and u, u' differ at most by the first letter is an input merge. Similarly, for $\lambda \le i \le \lambda + \rho - 1$, the equivalence θ_{i+1} on P_{i+1} defined by $(u, p, v) \equiv (u', p', v')$ if $u = u'$, $p = p'$ and v, v' differ at most by their last letter, is an output merge. This shows that conditions (ii) and (iii) are satisfied.

Finally, since $\mathcal{A}$ is (λ, ρ)-local, in a state (u, p, v) of $A_{\lambda+\rho}$, the state p is determined by (u, v). Thus condition (iv) is also satisfied.

■

Observe that if $\mathcal{A}$ is moreover supposed to be irreducible, then all the automata $\mathcal{A}_i$ constructed as above are also irreducible.

We will prove the following result.

Theorem 11: Any irreducible (λ, ρ)-local automaton is a subautomaton of a complete irreducible local automaton of order $\max(\lambda, \rho)$.

Note that this statement implies that any irreducible local automaton of order n is a subautomaton of a complete irreducible local automaton of order n.

The proof uses Nasu embeddings.

Proposition 12: Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be automata such that $\mathcal{A}_2$ is obtained from $\mathcal{A}_1$ by input (or output) merge. If $\mathcal{A}_1$ is a subautomaton of a local automaton $\mathcal{B}_1$ of order n, then the Nasu embedding of $\mathcal{A}_2$ with respect to $\mathcal{A}_1$ and $\mathcal{B}_1$, is a local automaton of order n.

Proof: We treat the case of an input merge. The case of an output merge is symmetrical. Let $\mathcal{B}_1 = (Q_1, F_1)$ be a local automaton containing $\mathcal{A}_1 = (P_1, E_1)$ as a subautomaton. Assume that $\mathcal{A}_2 = (P_2, E_2)$ is obtained from $\mathcal{A}_1$ by input merge. Let $F = F_1 \setminus E_1$, $Q_2 = P_2 \cup F$ and let $\mathcal{B}_2 = (Q_2, F_2)$ be the Nasu embedding of $\mathcal{A}_2$ with respect to $\mathcal{A}_1, \mathcal{B}_1$.

Let R, S be the matrices defined as in the proof of Proposition 3. The transition matrix of $\mathcal{A}_1$ is equal to RS while the transition matrix of $\mathcal{A}_2$ is equal to SR. We have $a \in S_{\bar{p},q}$ if $(p, a, q) \in E_1$ and $R_{p,\bar{p}} = 1$. The additional $\widehat{S}$-edges are the triples (f, a, q) such that $f = (p, a, q)$ is in F. The additional $\widehat{R}$-edges are the triples $(p, 1, f)$ such that f begins with p. An example of this construction is described in Figure 3.

Let π be the map defined on the set $\mathcal{P}$ of paths of $\mathcal{B}_1$ of length at least 2 onto the set of nonempty paths of $\mathcal{B}_2$ as follows. If $(e_0, e_1, \ldots, e_n) \in \mathcal{P}$ with $e_i = (q_i, a_i, q_{i+1})$, we define $\pi(e_0, e_1, \ldots, e_n) = (g_0, g_1, \ldots, g_{n-1})$ with $g_i = (p_i, a_i, p_{i+1})$ defined by $p_i = \bar{q_i}$ if $e_i \in E_1$ and $p_i = e_i$ if $e_i \in F$.

We claim that π is a surjective map from the set $\mathcal{P}$ onto the set of nonempty paths of $\mathcal{B}_2$. We have to verify that each g_i is an edge of $\mathcal{B}_2$. We distinguish four cases.

(i) If $p_i, p_{i+1} \in P_2$. Then g_i is an edge of $\mathcal{A}_2$ and thus of $\mathcal{B}_2$. It is the concatenation of an S-edge and an R-edge.
(ii) If $p_i \in P_2$ and $p_{i+1} \in F$. Then $p_i = \bar{q_i}$ and $p_{i+1} = e_{i+1}$. Thus g_i is the concatenation of an S-edge and an $(\widehat{R} - R)$-edge.
(iii) If p_i, p_{i+1} are in F, then $p_i = e_i$ and $p_{i+1} = e_{i+1}$. Thus $g_i = (p_i, a_i, p_{i+1})$ is the concatenation of an $(\widehat{S} - S)$-edge and an $(\widehat{R} - R)$-edge.
(iv) If $p_i \in F$ and $p_{i+1} \in P_2$ then $p_i = e_i$. Thus g_i is the concatenation of an $(\widehat{S} - S)$-edge and an R-edge.

Thus π is well defined. Moreover, one can verify that for any nonempty path c in $\mathcal{B}_2$ there exists $c' \in \mathcal{P}$ such that $\pi(c') = c$. Thus π is surjective.

Let n be the order of $\mathcal{B}_1$. Let $c = p \xrightarrow{w} q$ and $c' = p' \xrightarrow{w} q'$ be two paths in $\mathcal{B}_2$ with $|w| = n$. Then there exist paths $d = r \xrightarrow{w} s \xrightarrow{a} t$ and $d' = r' \xrightarrow{w} s' \xrightarrow{a'} t'$ in $\mathcal{B}_1$, with $a, a' \in A$, such that $\pi(d) = c$ and $\pi(d') = c'$. Since $\mathcal{B}_1$ has order n, w is a constant for $\mathcal{B}_1$. Thus we have also paths $e = r \xrightarrow{w} s' \xrightarrow{a'} t'$ and $e' = r' \xrightarrow{w} s \xrightarrow{a} t$ in $\mathcal{B}_1$. It is easy to verify that $\pi(e)$ is a path from p to q' and $\pi(e')$ a path from p' to q. Thus w is a constant for $\mathcal{B}_2$. ■

We now prove Theorem 11. *Proof:* Let $\mathcal{A}$ be an irreducible (λ, ρ)-local automaton. By Proposition 10 there is a sequence $\mathcal{A}_0, \mathcal{A}_1, \ldots, \mathcal{A}_{\lambda+\rho}$ of automata such that $\mathcal{A} = \mathcal{A}_0$, $\mathcal{A}_{\lambda+\rho}$ is a subautomaton of the free (λ, ρ)-local automaton and each $\mathcal{A}_i$ is a merge of A_{i+1}. By Lemma 8, the free (λ, ρ)-local automaton has order $n = \max(\lambda, \rho)$. Since $\mathcal{A}_{\lambda+\rho}$ is included in the free (λ, ρ)-local automaton, we may build using repeatedly Proposition 12 a sequence $\mathcal{B}_{\lambda+\rho}, \ldots, \mathcal{B}_1, \mathcal{B}_0$ of complete local irreducible automata with order n such that $\mathcal{A}_i$ is contained in $\mathcal{B}_i$ for $i = \lambda + \rho, \ldots, 0$. Thus $\mathcal{B} = \mathcal{B}_0$ is a complete local automaton with order n containing $\mathcal{A}$. Since $\mathcal{A}$ is irreducible, all $\mathcal{A}_i$ and $\mathcal{B}_i$ are irreducible. ■

Example 13: Let $\mathcal{A} = \mathcal{A}_0$ be the deterministic automaton represented on the left of Figure 5. It is $(3, 0)$-local and has order 3.

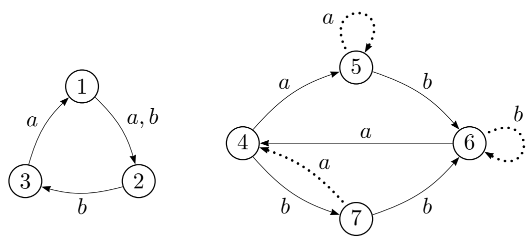

Fig. 5. A local automaton $\mathcal{A}$ (on the left) and a split $\mathcal{A}_1$ of $\mathcal{A}$ with its completion $\mathcal{B}_1$ with the dotted edges (on the right)

The automaton $\mathcal{A}$ cannot be completed in a local deterministic automaton (see [7]). We have represented on the right of Figure 5 a split $\mathcal{A}_1$ of $\mathcal{A}$, obtained by an input split of state 2 in two states 5 and 7. This automaton can easily be completed as indicated on the right of Figure 5. The auxiliary graph is shown on the left of Figure 6. The final result is shown on the right of Figure 6. It is a complete local automaton containing $\mathcal{A}$.

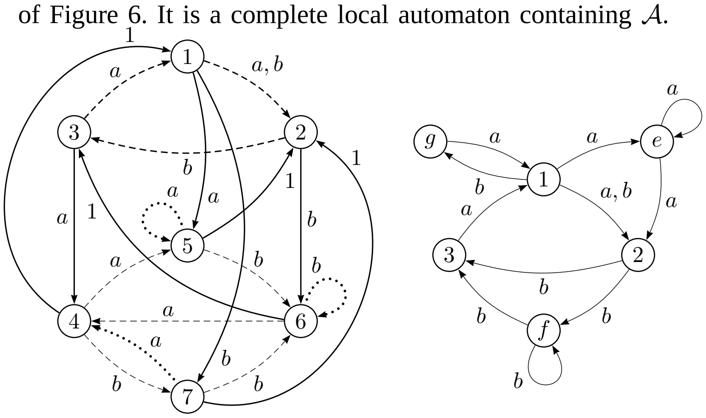

Fig. 6. The auxiliary graph (with continuous edges) on the left. The automata $\mathcal{A}$ and $\mathcal{A}_1$ on the left are represented with dashed edges and the additional edges with dotted lines. The result on the right is the embedding of $\mathcal{A}$ with respect to $\mathcal{A}_1$ and $\mathcal{B}_1$. This automaton has order 3 although it is $(3, 1)$-local.

IV. Application to locally parsable codes

In this section, we briefly show how our result applies to the theory of variable length codes.

A *variable length code* (or *code*) is a set X of finite words which is uniquely decipherable: $x_1 x_2 \ldots x_n = y_1 y_2 \ldots y_m$, with $x_i, y_j \in X$, implies $n = m$ and $x_i = y_i$ for all i.

Let X be a code. A word w is a *constant* for X^* if $uwv, u'wv' \in X^*$ implies $uwv', u'wv \in X^*$. A code X has *literal synchronization delay* s if any word of A^s is a constant for X^*. A code is *locally parsable* (or has *finite synchronization delay*) if there is an integer s such that it has literal synchronization delay s.

The stabilizer of a state in a local automaton with order s is X^*, where X is a locally parsable code with synchronization delay s. Conversely, if X is a locally parsable regular code with synchronization delay s, one can build an irreducible (s, s)-local automaton such that X^* is the stabilizer of some state (see [2].)

Since our result allows to complete this automaton in a complete one of order s, it gives a maximal regular locally parsable code Y containing X which has the same synchronization delay as X.

For instance, the code $X = \{aba, baba\}$, which has literal synchronization delay 3 can be completed into the regular code $Y = ab \cup (a^+ \cup (a^+ \cup b)b^+ a$ which has literal synchronization delay 3. An automaton recognizing Y^* is given on the right of Figure 6, with state 1 as initial and final state.

This gives an alternative proof of a result of Bruyère according to which any locally parsable regular code is included in a maximal one [3] where the conservation of the delay was not guaranteed.

Acknowledgements: We would like to thank Brian Marcus who pointed out to us the possibility of using Nasu's masking lemma in this problem. We also thank the referees who helped us to improve the presentation.

References

[1] Jonathan Ashley, Brian Marcus, Dominique Perrin, and Selim Tuncel. Surjective extensions of sliding block codes. *SIAM J. Discrete Math.*, 6:582–611, 1993.

[2] Jean Berstel, Dominique Perrin, and Christophe Reutenauer. *Codes and Automata*. Cambridge University Press, 2007. To appear, see `http://www-igm.univ-mlv.fr/~berstel/LivreCodes/newlivreCodes.pdf`.

[3] Véronique Bruyère. On maximal codes with bounded synchronization delay. *Theor. Comput. Sci.*, 204:11–28, 1988.

[4] Andrei Ehrenfeucht and Gregorz Rozenberg. Each regular code is included in a regular maximal code. *RAIRO Informat. Theor.*, 20:89–96, 1983.

[5] Wolfgang Krieger. On sofic systems. I. *Israel J. Math.*, 48(4):305–330, 1984.

[6] Douglas A. Lind and Brian H. Marcus. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, 1995.

[7] Rosalba Montalbano. Local automata and completion. In *STACS 93*, volume 665 of *Lecture Notes in Comput. Sci.*, pages 333–342. Springer verlag, 1993.

[8] Masakazu Nasu. Topological conjugacy for sofic systems. *Ergodic Theory Dynam. Systems*, 6(2):265–280, 1986.

[9] Masakazu Nasu. Topological conjugacy for sofic systems and extensions of automorphisms of finite subsystems of topological markov shifts. In *Proceedings of Maryland special year in Dynamics 1986–87*, volume 1342 of *Lecture Notes in Mathematics*, pages 564–607. Springer verlag, 1988.

[10] Marcel-Paul Schützenberger. A remark on incompletely specified automata. *Information and Control*, 8:373–376, 1965.