UNIVERSITY OF
NEWCASTLE

University of Newcastle upon Tyne

COMPUTING SCIENCE

Transition Systems of Elementary Net Systems with Localities

M. Koutny and M. Pietkiewicz-Koutny.

TECHNICAL REPORT SERIES

No. CS-TR-948 **Febuary, 2006**

TECHNICAL REPORT SERIES

No. CS-TR-948 Febuary, 2006

Transition Systems of Elementary Net Systems with Localities

Maciej Koutny, Marta Pietkiewicz-Koutny

Abstract

In this paper, we investigate transition systems of a class of Petri nets suitable for the modelling and behavioural analysis of globally asynchronous locally synchronous (GALS) systems. The considered model of Elementary Net Systems with Localities (ENL-systems) is basically that of Elementary Net Systems (EN-systems) equipped with an explicit notion of locality. Each locality identifies a distinct set of events which may only be executed synchronously, i.e., in a maximally concurrent manner. For this reason, the overall behaviour of an ENL-system cannot be represented by an interleaved transition system, with arcs being labelled by single events, but rather by a suitable notion of a step transition system, with arcs being labelled by sets of events executed concurrently.

We completely characterise transition systems which can be generated by Elementary Net Systems with Localities under their intended concurrency semantics. In developing a suitable characterisation, we follow the standard approach in which key relationships between a Petri net and its transition system are established via the the regions of the latter defined as specific sets of states of the transition system. We argue that this definition is insufficient for the class of transition systems of ENL-systems, and then augment the standard notion of a region with some additional information, leading to the notion of a region with explicit input and output events (or io-region).

We define, and show consistency of, two behaviour preserving translations between ENL-systems and their transition systems. As a result, we provide a solution to the synthesis problem of Elementary Net Systems with Localities, which consists in constructing an ENL-system for a given transition system in such a way that the transition system of the former is isomorphic to the latter.

Printed and published by the University of Newcastle upon Tyne,
Computing Science, Claremont Tower, Claremont Road,
Newcastle upon Tyne, NE1 7RU, England.

Bibliographical details

KOUTNY, M., PIETKIEWICZ-KOUTNY, M..

Transition Systems of Elementary Net Systems with Localities
[By] M. Koutny and M. Pietkiewicz-Koutny.

Newcastle upon Tyne: University of Newcastle upon Tyne: Computing Science, 2006.

(University of Newcastle upon Tyne, Computing Science, Technical Report Series, No. CS-TR-948)

Added entries

UNIVERSITY OF NEWCASTLE UPON TYNE
Computing Science. Technical Report Series. CS-TR-948

Abstract

In this paper, we investigate transition systems of a class of Petri nets suitable for the modelling and behavioural analysis of globally asynchronous locally synchronous (GALS) systems. The considered model of Elementary Net Systems with Localities (ENL-systems) is basically that of Elementary Net Systems (EN-systems) equipped with an explicit notion of locality. Each locality identifies a distinct set of events which may only be executed synchronously, i.e., in a maximally concurrent manner. For this reason, the overall behaviour of an ENL-system cannot be represented by an interleaved transition system, with arcs being labelled by single events, but rather by a suitable notion of a step transition system, with arcs being labelled by sets of events executed concurrently.

We completely characterise transition systems which can be generated by Elementary Net Systems with Localities under their intended concurrency semantics. In developing a suitable characterisation, we follow the standard approach in which key relationships between a Petri net and its transition system are established via the the regions of the latter defined as specific sets of states of the transition system. We argue that this definition is insufficient for the class of transition systems of ENL-systems, and then augment the standard notion of a region with some additional information, leading to the notion of a region with explicit input and output events (or io-region).

We define, and show consistency of, two behaviour preserving translations between ENL-systems and their transition systems. As a result, we provide a solution to the synthesis problem of Elementary Net Systems with Localities, which consists in constructing an ENL-system for a given transition system in such a way that the transition system of the former is isomorphic to the latter.

About the author

Maciej Koutny obtained his MSc (1982) and PhD (1984) from the Warsaw University of Technology. In 1985 he joined the then Computing Laboratory of the University of Newcastle upon Tyne to work as a Research Associate. In 1986 he became a Lecturer in Computing Science at Newcastle, and in 1994 was promoted to an established Readership at Newcastle. In 2000 he became a Professor of Computing Science.

Marta Pietkiewicz-Koutny received her M.Sc. in Applied Mathematics from the Warsaw University of Technology in 1982. In 1984 she joined the Department of Operational Research, Institute of Econometrics in the Warsaw University of Economics where she worked as a junior lecturer until 1986. In 1987 she joined the Computing Laboratory of the University of Newcastle upon Tyne first as a research associate and then as a demonstrator (1988-1997). In the period 1997-2000 she was a Ph.D. student at the Department of Computing Science of the University of Newcastle upon Tyne, and in December 2000 she was awarded her Ph.D. degree.

Suggested keywords

THEORY OF CONCURRENCY,
PETRI NETS,
ELEMENTARY NET SYSTEMS,
GALS SYSTEMS,
LOCALITIES,
ANALYSIS AND SYNTHESIS,
STEP SEQUENCE SEMANTICS,
STRUCTURE AND BEHAVIOUR OF NETS,
THEORY OF REGIONS,
TRANSITION SYSTEMS

Transition Systems of Elementary Net Systems with Localities

Maciej Koutny and Marta Pietkiewicz-Koutny

School of Computing Science, University of Newcastle
Newcastle upon Tyne, NE1 7RU, United Kingdom
{maciej.koutny,marta.koutny}@newcastle.ac.uk

Abstract. In this paper, we investigate transition systems of a class of Petri nets suitable for the modelling and behavioural analysis of globally asynchronous locally synchronous (GALS) systems. The considered model of Elementary Net Systems with Localities (ENL-systems) is basically that of Elementary Net Systems (EN-systems) equipped with an explicit notion of locality. Each locality identifies a distinct set of events which may only be executed synchronously, i.e., in a maximally concurrent manner. For this reason, the overall behaviour of an ENL-system cannot be represented by an interleaved transition system, with arcs being labelled by single events, but rather by a suitable notion of a step transition system, with arcs being labelled by sets of events executed concurrently.

We completely characterise transition systems which can be generated by Elementary Net Systems with Localities under their intended concurrency semantics. In developing a suitable characterisation, we follow the standard approach in which key relationships between a Petri net and its transition system are established via the the regions of the latter defined as specific sets of states of the transition system. We argue that this definition is insufficient for the class of transition systems of ENL-systems, and then augment the standard notion of a region with some additional information, leading to the notion of a region with explicit input and output events (or io-region).

We define, and show consistency of, two behaviour preserving translations between ENL-systems and their transition systems. As a result, we provide a solution to the synthesis problem of Elementary Net Systems with Localities, which consists in constructing an ENL-system for a given transition system in such a way that the transition system of the former is isomorphic to the latter.
Keywords: theory of concurrency, Petri nets, elementary net systems, GALS systems, localities, analysis and synthesis, step sequence semantics, structure and behaviour of nets, theory of regions, transition systems.

1 Introduction

Several real-life computational systems exhibit dynamic behaviour which could best be described as following the 'globally asynchronous locally synchronous' paradigm (or GALS). Prominent examples of such systems can be found in hardware design, where a VLSI chip may contain multiple clocks responsible for synchronising different subsets of gates, and in biologically motivated computing, where a membrane system models a cell with compartments, inside which reactions are carried out in co-ordinated pulses. In these cases, the activities in different localities can proceed independently, subject to communication and/or synchronisation constraints. To express such systems in a formal manner, [8] introduced *Place/Transition-nets with localities* (PTL-nets), which are basically PT-nets equipped with the notion a locality. Each locality identifies a distinct set of transitions which may only be executed synchronously, i.e., in a maximally concurrent manner. The aim of [8] was then to look at the way in which the standard concurrency techniques of Petri nets could be used to provide a similar treatment for the new model. In this paper, we adapt the model of [8] to the case of Elementary Net Systems (EN-systems), which are a fundamental class of safe Petri nets, and set ourselves the task of finding a characterisation of all transition systems generated by such nets.

ENL-systems

To explain the basic concepts relating to ENL-systems, we consider the net shown in figure 1(a), which models a concurrent system consisting of one producer (the left triangle-like subnet), and one consumer process (the right square-like subnet). The two subsystems are connected by a buffer-like condition b_0 which holds items produced by the producer using the event `p2`, and consumed by the consumer using the event `c1`. The net would be a standard EN-system if we ignored the integer labels, 1 and 2, shown in the middle of the events. These labels represent *localities* to which the various events belong. We can then observe that events `p1` and `p2` belong to the same locality, while the remaining events to a different one.

In general, the way events are assigned to different localities will have a strong impact on the step sequences generated by an ENL-system, as it is required that within each locality events are executed in a maximally concurrent way. For the net in figure 1(a), this does not have any apparent effect since the subnets corresponding to the two localities are strictly sequential. This changes radically for the slightly modified example shown in figure 1(b), which models a system consisting of one producer and two co-located consumers (indicated by the two tokens in the right subnet).

For example, though under the standard EN-systems' semantics this net generates the step sequence $\{\texttt{p2}\}\{\texttt{c1}\}$, the execution model of ENL-systems will reject it for the following reason: After executing the initial step $\{\texttt{p2}\}$, the net can execute the step $\{\texttt{c1},\texttt{c4}\}$ consisting of two co-located events, and so executing `c1` alone violates the maximal concurrency execution rule within locality 2. A possible way of executing a valid step could then be to add the 'missing' event `c4`, resulting in the legal step sequence $\{\texttt{p2}\}\{\texttt{c1},\texttt{c4}\}$. Another legal step sequence, according to the intended semantics, could be $\{\texttt{p2}\}\{\texttt{c1},\texttt{c4},\texttt{p1}\}$. Note that in the latter case the second step {c1, c4, p1} is maximally concurrent in a global sense, as it cannot be extended any further.

If all the events of an ENL-system belong to the same locality (and so no extra labelling is really needed), then the notion of an ENL-system reduces to that of an Elementary Net System with Maximal Concurrency. In a nutshell, such a system is an EN-system executed under the maximal concurrency rule, i.e., in such a way that an executed step cannot be extended any further without violating the basic constraint embodied in the structure of the net.

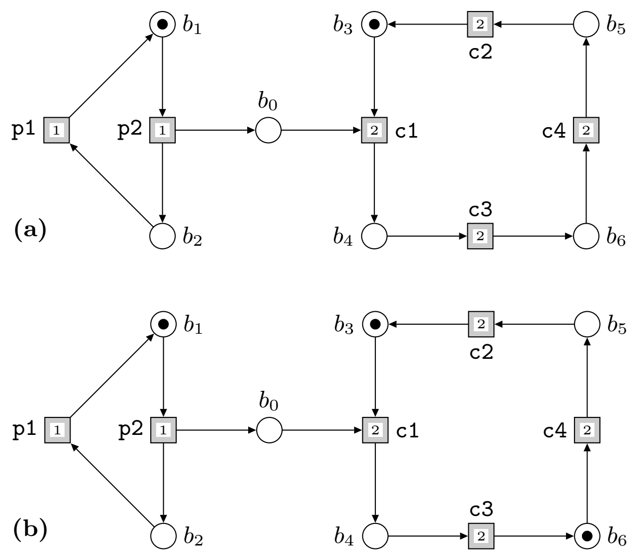

Fig. 1. ENL-system for the one-producer/one-consumer scenario **(a)**; and for the one-producer/two-consumers scenario **(b)**.

ENL-systems and the synthesis problem

Let us consider the EN-system together with its interleaving transition system shown in figure 2(a,b), and the ENL-system together with its step transition system shown in figure 2(c,d). Suppose that we are to solve the synthesis problem for the ENL-systems in the case of the example shown in figure 2. It can be formulated as a task of finding a method which, given a transition system, constructs a net in such a way that its behaviour (expressed as a transition system) is isomorphic to the given transition system.

This problem was solved for the class of EN-systems in [6], using the notion of a region which links nodes of transition systems (global states) with conditions in the corresponding nets (local states). The solution was later extended to the pure bounded PT-nets [4], general Petri nets [10], safe nets [14] and EN-systems with inhibitor arcs [5, 9, 12], by adopting the definition of a region or using some extended notion of a generalised region [3].

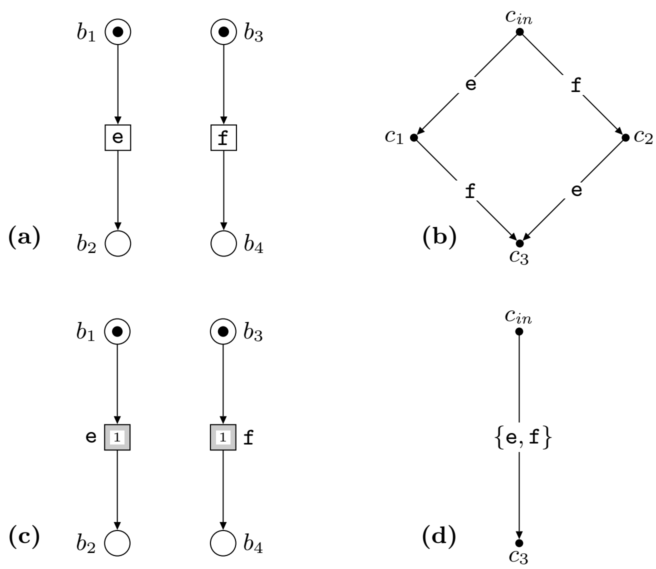

Fig. 2. An EN-system **(a)**; its interleaving transition system **(b)**; an ENL-system **(c)**; and step transition system **(d)**. Note that $c_{in} = \{b_1, b_3\}$, $c_1 = \{b_2, b_3\}$, $c_2 = \{b_1, b_4\}$ and $c_3 = \{b_2, b_4\}$.

The way the standard region construction works can be explained in the following way. Let us try to retrieve the original EN-system from the interleaving transition system in figure 2(b). A region of a transition system is meant to encompass precisely those states where a given condition of the EN-system holds. So, for example, the region corresponding to the condition b_1 is the set comprising of two states, $r_1 = \{c_{in}, c_2\}$. Its defining characteristics is that is has the same 'crossing relationship' with both events `e` and `f`. Indeed, both `e`-arcs leave r_1, and both `f`-arcs do not cross r_1's boundary. There are three further regions with similar 'stable' crossing characteristics: $r_2 = \{c_1, c_3\}$ (`e`-arcs enter it and `f`-arcs do not cross the boundary), $r_3 = \{c_{in}, c_1\}$ (`f`-arcs leave it and `e`-arcs do not cross the boundary) and $r_4 = \{c_2, c_3\}$ (`f`-arcs enter it and `e`-arcs do not cross the boundary).[1] The synthesis procedure then derives an EN-system in the following way: for each region r_i a fresh condition b'_i is constructed and its input and output events are determined by the crossing relationships mentioned above, e.g., for r_1 `e` is an output event, and `f` is not joined to it by any arc. In this particular case, the resulting EN-system is actually isomorphic to the original

[1] Note that there are two more 'trivial' regions, $r = \{c_{in}, c_1, c_2, c_3\}$ and $r' = \varnothing$, which are ignored by the synthesis procedure.

one, though in the general case this cannot be guaranteed. But what can be guaranteed is that the transition systems of the two EN-systems are isomorphic.

One might attempt to apply the same procedure also in the case of the step transition system in figure 2(d), under the assumption that $\mathtt{e}$ and $\mathtt{f}$ are co-located events. However, this is not going to get us back to a desired ENL-system since there are at most 2 non-trivial regions there, $r_1 = \{c_{in}\}$ and $r_2 = \{c_3\}$, and so the construction can at best generate 2 conditions. However, we need at least 4 conditions in the resulting ENL-system, to be able to support a pair of concurrent events which were executed at the initial state. In other words, we have too few standard regions to construct enough conditions if the target is an appropriate ENL-system.

About this paper

An intuitive reason why the standard construction failed to work for the step transition system in figure 2(d) was that the set-of-states notion of region is not rich enough for the purposes of synthesising ENL-systems.

The modification to the original notion we will propose is based on the explicit *input and output events* of a set of states, which is consistent with the underlying idea of static crossing relationship between events and regions. More precisely, we will work with *io-regions*, each such region being a triple: $\mathfrak{r} = (in, r, out)$, where r is a set of states, in is a set of events which are responsible for entering r, and out is a set of events which are responsible for leaving r. Intuitively, we will require that in each step leaving (or entering) r there is a unique event belonging to out (resp. to in) responsible for it, and, conversely, executing an event from out or from in always results in crossing of the boundary of r in an appropriate way. In the case of the example transition system, we will identify four io-regions: $\mathfrak{r}_1 = (\varnothing, \{c_{in}\}, \{\mathtt{e}\})$ $\mathfrak{r}_2 = (\{\mathtt{e}\}, \{c_3\}, \varnothing\})$, $\mathfrak{r}_3 = (\varnothing, \{c_{in}\}, \{\mathtt{f}\})$ and $\mathfrak{r}_4 = (\{\mathtt{f}\}, \{c_3\}, \varnothing)$. Now we have enough regions to re-constitute the conditions of the original ENL-system, namely each $\mathfrak{r}_i$ corresponds to b_i. The rest of the construction is basically the same as in the standard approach.

The paper is organised as follows. In the next section, we introduce step transition systems and their io-regions. After that we define ENL-transition systems. In section 3, we introduce formally ENL-systems and show that their transition systems are ENL-transition systems. We also demonstrate how to construct a corresponding ENL-system for a given ENL-transition system. In the concluding section, we compare our approach with other works, and outline how the modified notion of region could be used to solve the synthesis problem for other semantics of EN-systems.

2 Step transition systems and io-regions

In this section, we set the scene by introducing general step transition systems, which after further restrictions will be used to provide a behavioural model for ENL-systems, and the key notion of an io-region of a step transition system.

Let $\mathcal{E}$ be a non-empty set of *events* fixed throughout this paper. We also assume that there is a *locality mapping*, $\mathfrak{L} : \mathcal{E} \to \mathbb{N}$, associating to each event $e \in \mathcal{E}$ its locality $\mathfrak{L}(e)$; implicitly, each non-empty inverse image $\mathfrak{L}^{-1}(n)$ determines a set of co-located events.

A *step transition system* [1, 7] is a triple $\mathfrak{ts} \stackrel{\text{df}}{=} (S, T, s_{in})$ where:

TS1 S is a non-empty finite set of *states*.
TS2 $T \subseteq S \times (2^{\mathcal{E}} \setminus \{\varnothing\}) \times S$ is a finite set of *transitions*.
TS3 $s_{in} \in S$ is the *initial state*.

Throughout the rest of this section, the step transition system $\mathfrak{ts}$ will be fixed. We will denote by $\mathcal{E}_{\mathfrak{ts}}$ the set of all the events appearing in steps labelling the transitions of $\mathfrak{ts}$, i.e.,

$$\mathcal{E}_{\mathfrak{ts}} \stackrel{\text{df}}{=} \bigcup_{(s,u,s') \in T} u \, .$$

We will use $s \xrightarrow{u} s'$ whenever $(s, u, s') \in T$, and respectively call s the *source* and s' the *target* of this transition. We will also say that the step u is *enabled* at s, and denote this by $s \xrightarrow{u}$. Moreover, we will denote $s \longrightarrow s'$ if $s \xrightarrow{u} s'$, for some u.

We now introduce a central notion whose aim is to link the nodes of a transition system (global states) with the conditions in the corresponding net (local states).

Definition 1. *A region with explicit input and output events (or* io-region*) is a triple* $\mathfrak{r} \stackrel{\text{df}}{=} (in, r, out) \in 2^{\mathcal{E}_{\mathfrak{ts}}} \times 2^S \times 2^{\mathcal{E}_{\mathfrak{ts}}}$ *such that the following four conditions are satisfied, for every transition* $s \xrightarrow{u} s'$ *of the step transition system* $\mathfrak{ts}$*:*

1. *If* $s \in r$ *and* $s' \notin r$ *then* $|u \cap in| = 0$ *and* $|u \cap out| = 1$.
2. *If* $s \notin r$ *and* $s' \in r$ *then* $|u \cap in| = 1$ *and* $|u \cap out| = 0$.
3. *If* $u \cap out \neq \varnothing$ *then* $s \in r$ *and* $s' \notin r$.
4. *If* $u \cap in \neq \varnothing$ *then* $s \notin r$ *and* $s' \in r$.

We denote $\|\mathfrak{r}\| \stackrel{\text{df}}{=} r$, ${}^\bullet\mathfrak{r} \stackrel{\text{df}}{=} in$ *and* $\mathfrak{r}^\bullet \stackrel{\text{df}}{=} out$.

An io-region $\mathfrak{r}$ is *trivial* if $\|\mathfrak{r}\| = \varnothing$ or $\|\mathfrak{r}\| = S$; otherwise it is *non-trivial.*

Proposition 1. *There are exactly two trivial io-regions:* $(\varnothing, \varnothing, \varnothing)$ *and* $(\varnothing, S, \varnothing)$.

Proof. From definition 1(3,4) and TS2, if $\mathfrak{r}$ is trivial then it must be the case that ${}^\bullet\mathfrak{r} = \mathfrak{r}^\bullet = \varnothing$. □

Proposition 2. $\mathfrak{r} = (in, r, out)$ *is an io-region if and only if* $\overline{\mathfrak{r}} \stackrel{\text{df}}{=} (out, S \setminus r, in)$ *is an io-region.*

Proof. Follows directly from definition 1. □

In general, an io-region $\mathfrak{r}$ cannot be identified only by its set of states $\|\mathfrak{r}\|$; in other words, ${}^\bullet\mathfrak{r}$ and $\mathfrak{r}^\bullet$ may not be recoverable from $\|\mathfrak{r}\|$. However, if the transition system is *thin*, i.e., for every event $e \in \mathcal{E}_{\mathfrak{ts}}$ there is a transition $s \xrightarrow{\{e\}} s'$ of $\mathfrak{ts}$, then different io-regions are based on different sets of states.[2]

Proposition 3. *If* $\mathfrak{ts}$ *is thin and* $\mathfrak{r} \neq \mathfrak{r}'$ *are io-regions, then* $\|\mathfrak{r}\| \neq \|\mathfrak{r}'\|$.

Proof. Suppose that $\|\mathfrak{r}\| = \|\mathfrak{r}'\|$. Since $\mathfrak{r} \neq \mathfrak{r}'$, we have that ${}^\bullet\mathfrak{r} \neq {}^\bullet\mathfrak{r}'$ or $\mathfrak{r}^\bullet \neq \mathfrak{r}'^\bullet$. Assume, without loss of generality, that ${}^\bullet\mathfrak{r} \neq {}^\bullet\mathfrak{r}'$. Then, again without loss of generality, we have that ${}^\bullet\mathfrak{r} \neq \varnothing$.

Let us take any $e \in {}^\bullet\mathfrak{r}$. Since $\mathfrak{ts}$ is thin, there is a transition $s \xrightarrow{\{e\}} s'$ of $\mathfrak{ts}$. By definition 1(4), we have $s \notin \|\mathfrak{r}\|$ and $s' \in \|\mathfrak{r}\|$. Hence, by $\|\mathfrak{r}\| = \|\mathfrak{r}'\|$, we also have that $s \notin \|\mathfrak{r}'\|$ and $s' \in \|\mathfrak{r}'\|$. Thus, by $s \xrightarrow{\{e\}} s'$ and definition 1(2), $e \in {}^\bullet\mathfrak{r}'$. As a result, ${}^\bullet\mathfrak{r} \subseteq {}^\bullet\mathfrak{r}'$. By proceeding in a similar way, we may then show that ${}^\bullet\mathfrak{r} = {}^\bullet\mathfrak{r}'$ and $\mathfrak{r}^\bullet = \mathfrak{r}'^\bullet$. This, together with $\|\mathfrak{r}\| = \|\mathfrak{r}'\|$, produces a contradiction with $\mathfrak{r} \neq \mathfrak{r}'$. □

The set of all *non-trivial* io-regions will be denoted by $\mathfrak{R}_{\mathfrak{ts}}$ and, for every state $s \in S$, we will denote by $\mathfrak{R}_s$ the set of non-trivial io-regions containing s,

$$\mathfrak{R}_s \stackrel{\text{df}}{=} \{\mathfrak{r} \in \mathfrak{R}_{\mathfrak{ts}} \mid s \in \|\mathfrak{r}\|\} .$$

The sets of *pre-io-regions*, ${}^\circ e$, and *post-io-regions*, e°, of an event $e \in \mathcal{E}_{\mathfrak{ts}}$ are then defined as:

$${}^\circ e \stackrel{\text{df}}{=} \{\mathfrak{r} \in \mathfrak{R}_{\mathfrak{ts}} \mid e \in \mathfrak{r}^\bullet\} \quad \text{and} \quad e^\circ \stackrel{\text{df}}{=} \{\mathfrak{r} \in \mathfrak{R}_{\mathfrak{ts}} \mid e \in {}^\bullet\mathfrak{r}\} .$$

Moreover, the sets of pre-io-regions and post-io-regions of a set of events $u \subseteq \mathcal{E}_{\mathfrak{ts}}$ are respectively given by:

$${}^\circ u \stackrel{\text{df}}{=} \bigcup_{e \in u} {}^\circ e \quad \text{and} \quad u^\circ \stackrel{\text{df}}{=} \bigcup_{e \in u} e^\circ .$$

Proposition 4. *If* $s \xrightarrow{u} s'$ *is a transition of* $\mathfrak{ts}$*, then*

1. $\mathfrak{r} \in {}^\circ u$ *implies* $s \in \|\mathfrak{r}\|$ *and* $s' \notin \|\mathfrak{r}\|$.
2. $\mathfrak{r} \in u^\circ$ *implies* $s \notin \|\mathfrak{r}\|$ *and* $s' \in \|\mathfrak{r}\|$.

Proof. Follows directly from the definitions of ${}^\circ u$ and u°, as well as definition 1(3,4). □

The sets of pre- and post-io-regions of a step involved in a transition of $\mathfrak{ts}$ are, in fact, disjoint unions of sets of respectively pre- and post-io-regions of events it comprises.

[2] As we have already seen being thin is not, in general, a property of step transition systems generated by ENL-systems. However, transition systems generated by, e.g., EN-systems or EN-systems with inhibitor arcs, are thin and then the standard definition of a region as a set of states is sufficient.

Proposition 5. *If u is a step appearing in one of the transitions of $\mathfrak{ts}$, then*

$$^{\circ}u = \biguplus_{e\in u} {^{\circ}e} \quad \textit{and} \quad u^{\circ} = \biguplus_{e\in u} e^{\circ} \ .$$

Proof. Let $s \xrightarrow{u} s'$ and $e, f \in u$ be such that $e \neq f$. Suppose that $\mathfrak{r} \in {^{\circ}e} \cap {^{\circ}f}$ which means that $e, f \in \mathfrak{r}^{\bullet}$. This means, by definition 1(3), that $s \in \|\mathfrak{r}\|$ and $s' \notin \|\mathfrak{r}\|$. Thus, by definition 1(1), $|u \cap \mathfrak{r}^{\bullet}| = 1$, a contradiction with $e, f \in u \cap \mathfrak{r}^{\bullet}$. Hence the first part of the result holds. The second one can be shown in a similar way. □

Proposition 6. *If u is a step appearing in one of the transitions of $\mathfrak{ts}$, then $^{\circ}u \cap u^{\circ} = \varnothing$.*

Proof. Suppose that $s \xrightarrow{u} s'$ and $\mathfrak{r} \in {^{\circ}u} \cap u^{\circ}$. Then, by proposition 4, $s \notin \|\mathfrak{r}\|$ and $s \in \|\mathfrak{r}\|$. We thus obtained a contradiction. □

Proposition 7. *If $s \xrightarrow{u} s'$ then $\mathfrak{R}_s \setminus \mathfrak{R}_{s'} = {^{\circ}u}$ and $\mathfrak{R}_{s'} \setminus \mathfrak{R}_s = u^{\circ}$.*

Proof. We show that $\mathfrak{R}_s \setminus \mathfrak{R}_{s'} = {^{\circ}u}$, as the second part can be shown in a similar way. By proposition 4, $^{\circ}u \subseteq \mathfrak{R}_s$ and $^{\circ}u \cap \mathfrak{R}_{s'} = \varnothing$. Hence $^{\circ}u \subseteq \mathfrak{R}_s \setminus \mathfrak{R}_{s'}$. Suppose that $\mathfrak{r} \in \mathfrak{R}_s \setminus \mathfrak{R}_{s'}$, which implies that $s \in \|\mathfrak{r}\|$ and $s' \notin \|\mathfrak{r}\|$. Hence, by definition 1(1) and $s \xrightarrow{u} s'$, $u \cap \mathfrak{r}^{\bullet} \neq \varnothing$. Hence $\mathfrak{r} \in {^{\circ}u}$ and so $\mathfrak{R}_s \setminus \mathfrak{R}_{s'} \subseteq {^{\circ}u}$. Consequently, $\mathfrak{R}_s \setminus \mathfrak{R}_{s'} = {^{\circ}u}$. □

To characterise fully transition systems generated by ENL-systems, we will need the notion of a potential step. The set of all *potential steps* $\mathbb{U}_{\mathfrak{ts}}$ of $\mathfrak{ts}$ is defined as follows:

$$\mathbb{U}_{\mathfrak{ts}} \stackrel{\text{df}}{=} \{u \subseteq \mathcal{E}_{\mathfrak{ts}} \mid u \neq \varnothing \ \wedge \ \forall e, f \in u : (e \neq f \Rightarrow ({^{\circ}e} \cup e^{\circ}) \cap ({^{\circ}f} \cup f^{\circ}) = \varnothing)\} \ .$$

Proposition 8. *If $s \xrightarrow{u} s'$ then $u \in \mathbb{U}_{\mathfrak{ts}}$.*

Proof. Follows from TS2 and propositions 5 and 6. □

2.1 ENL-transition systems

A step transition system $\mathfrak{ts} = (S, T, s_{in})$ is an *ENL-transition system* if it satisfies the following axioms:

- **A1** For every $s \in S \setminus \{s_{in}\}$, there are $(s_0, u_0, s_1), \ldots, (s_{n-1}, u_{n-1}, s_n) \in T$ such that $s_0 = s_{in}$ and $s_n = s$.
- **A2** For every event $e \in \mathcal{E}_{\mathfrak{ts}}$, both $^{\circ}e$ and e° are non-empty.
- **A3** For all states $s, s' \in S$, if $\mathfrak{R}_s = \mathfrak{R}_{s'}$ then $s = s'$.
- **A4** Let $s \in S$ and $u \in \mathbb{U}_{\mathfrak{ts}}$ be such that $^{\circ}u \subseteq \mathfrak{R}_s$ and $u^{\circ} \cap \mathfrak{R}_s = \varnothing$, and there is no $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{ts}}$ satisfying $\mathfrak{L}(e) \in \mathfrak{L}(u)$ and $^{\circ}e \subseteq \mathfrak{R}_s$ and $e^{\circ} \cap \mathfrak{R}_s = \varnothing$. Then $s \xrightarrow{u}$.
- **A5** If $s \xrightarrow{u}$ then there is no $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{ts}}$ satisfying $\mathfrak{L}(e) \in \mathfrak{L}(u)$ and $^{\circ}e \subseteq \mathfrak{R}_s$ and $e^{\circ} \cap \mathfrak{R}_s = \varnothing$.

The (A1) axiom implies that all the states in $\mathfrak{ts}$ are reachable from the initial state. (A2) will ensure that every event in a synthesised ENL-system will have at least one input condition and at least one output condition. (A3) was used for other transition systems as well, and is usually called the *state separation property* [3, 11], and it guarantees that $\mathfrak{ts}$ is deterministic. (A4) is a variation of the *forward closure property* [11] or the *event/state separation property* [3]. (A5) ensures that every step in a transition system is indeed a maximal step w.r.t. localities of the events it comprises.

Proposition 9. *If $s \xrightarrow{u} s'$ and $s \xrightarrow{u} s''$ then $s' = s''$.*

Proof. Follows from proposition 7 and (A3). □

3 ENL-systems

A *net* is a tuple $\mathfrak{net} \stackrel{\text{df}}{=} (B, E, F)$ such that B and $E \subseteq \mathcal{E}$ are finite disjoint sets, and $F \subseteq (B \times E) \cup (E \times B)$. The meaning and graphical representation of B (conditions), E (events) and F (flow relation) is the same as in the standard net theory. Moreover, in diagrams, boxes representing events with localities are shaded with the actual locality being shown in the middle (see figure 1). We denote, for every $x \in B \cup E$,

$$^\bullet x \stackrel{\text{df}}{=} \{y \mid (y, x) \in F\} \quad \text{and} \quad x^\bullet \stackrel{\text{df}}{=} \{y \mid (x, y) \in F\} \ ,$$

and we call them the *pre-elements* and *post-elements*, respectively. The dot-notation extends in the usual way to sets:

$$^\bullet X \stackrel{\text{df}}{=} \bigcup_{x \in X} {^\bullet x} \ \text{ and } \ X^\bullet \stackrel{\text{df}}{=} \bigcup_{x \in X} x^\bullet \ .$$

It is assumed that for every $e \in E$, the sets $^\bullet e$ and $e^\bullet$ are non-empty and disjoint.

An *elementary net system with localities* (ENL-system) is a tuple

$$\mathfrak{enl} \stackrel{\text{df}}{=} (B, E, F, c_{in})$$

such that $\mathfrak{net}_{\mathfrak{enl}} \stackrel{\text{df}}{=} (B, E, F)$ is the (underlying) net and $c_{in} \subseteq B$ is the *initial case* (in general, any subset of B is a *case*). We will assume that $\mathfrak{enl}$ is fixed until the end of this section.

The concurrency semantics of ENL-systems will be based on steps of simultaneously executed events. We first define the set of *valid steps* of the ENL-system:

$$\mathbb{U}_{\mathfrak{enl}} \stackrel{\text{df}}{=} \{u \subseteq E \mid u \neq \varnothing \ \wedge \ \forall e, f \in u : \ (e \neq f \Rightarrow (^\bullet e \cup e^\bullet) \cap (^\bullet f \cup f^\bullet) = \varnothing)\} \ .$$

A step $u \in \mathbb{U}_{\mathfrak{enl}}$ is *enabled* at a case $c \subseteq B$ if $^\bullet u \subseteq c$ and $u^\bullet \cap c = \varnothing$, and there is no step $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{enl}}$ satisfying $\mathfrak{L}(e) \in \mathfrak{L}(u)$ and $^\bullet e \subseteq c$ and $e^\bullet \cap c = \varnothing$.

The transition relation of $\mathfrak{net}_{\mathfrak{enl}}$, denoted by $\rightarrow_{\mathfrak{net}_{\mathfrak{enl}}}$, is then given as the set of all triples

$$(c, u, c') \in 2^B \times \mathbb{U}_{\mathfrak{enl}} \times 2^B$$

such that u is enabled at c and $c' = (c \setminus {}^\bullet u) \cup u^\bullet$.

The *state space* of $\mathfrak{enl}$, denoted by $C_{\mathfrak{enl}}$, is the least subset of 2^B containing c_{in} such that if $c \in C_{\mathfrak{enl}}$ and $(c, u, c') \in \rightarrow_{\mathfrak{net}_{\mathfrak{enl}}}$ then $c' \in C_{\mathfrak{enl}}$. The *transition relation* of $\mathfrak{enl}$, denoted by $\rightarrow_{\mathfrak{enl}}$, is then defined as $\rightarrow_{\mathfrak{net}_{\mathfrak{enl}}}$ restricted to $C_{\mathfrak{enl}} \times \mathbb{U}_{\mathfrak{enl}} \times C_{\mathfrak{enl}}$. We will use $c \xrightarrow{u}_{\mathfrak{enl}} c'$ to denote that $(c, u, c') \in \rightarrow_{\mathfrak{enl}}$. Also, $c \xrightarrow{u}_{\mathfrak{enl}}$ if $(c, u, c') \in \rightarrow_{\mathfrak{enl}}$, for some c'.

Proposition 10. *If $c \xrightarrow{u}_{\mathfrak{enl}} c'$ then $c \setminus c' = {}^\bullet u$ and $c' \setminus c = u^\bullet$.*

Proof. From $c \xrightarrow{u}_{\mathfrak{enl}} c'$ we have that u is enabled at c (which implies ${}^\bullet u \subseteq c$ and $u^\bullet \cap c = \varnothing$) and $c' = (c \setminus {}^\bullet u) \cup u^\bullet$. One can easily check that these imply $c \setminus c' = {}^\bullet u$ and $c' \setminus c = u^\bullet$. ☐

3.1 Transition systems generated by ENL-systems

The construction of a step transition system for a given ENL-system is straightforward.

Let $\mathfrak{enl} = (B, E, F, c_{in})$ be an ENL-system. Then

$$\mathfrak{ts}_{\mathfrak{enl}} \stackrel{\text{df}}{=} (C_{\mathfrak{enl}}, \rightarrow_{\mathfrak{enl}}, c_{in})$$

is the *transition system generated* by $\mathfrak{enl}$.

Theorem 1. *$\mathfrak{ts}_{\mathfrak{enl}}$ is an ENL-transition system.*

Proof. Clearly, $\mathfrak{ts}_{\mathfrak{enl}}$ is a step transition system. We need to prove that it satisfies the five axioms defining ENL-transition systems. Before doing this, we will show that, for every $b \in B$,

$$\mathfrak{r}_b \stackrel{\text{df}}{=} ({}^\bullet b, \{c \in C_{\mathfrak{enl}} \mid b \in c\}, b^\bullet)$$

is a (possibly trivial) io-region of $\mathfrak{ts}_{\mathfrak{enl}}$. Moreover, if $\varnothing \neq \|\mathfrak{r}_b\| \neq C_{\mathfrak{enl}}$ then $\mathfrak{r}_b$ is non-trivial.

To show that definition 1 holds for $\mathfrak{r}_b$, we assume that $c \xrightarrow{u}_{\mathfrak{enl}} c'$ in $\mathfrak{ts}_{\mathfrak{enl}}$, and proceed as follows:

Proof of definition 1(1) for $\mathfrak{r}_b$. We need to show that $c \in \|\mathfrak{r}_b\|$ and $c' \notin \|\mathfrak{r}_b\|$ implies $|u \cap {}^\bullet b| = 0$ and $|u \cap b^\bullet| = 1$.
From $c \in \|\mathfrak{r}_b\|$ ($c' \notin \|\mathfrak{r}_b\|$) it follows that $b \in c$ (resp. $b \notin c'$). Hence $b \in c \setminus c'$. From proposition 10 we have $c \setminus c' = {}^\bullet u$ and $c' \setminus c = u^\bullet$. Hence $b \in {}^\bullet u$ and, as a consequence, there exists $e \in u$ such that $b \in {}^\bullet e$, and so $e \in b^\bullet$. We therefore have $e \in u \cap b^\bullet$. Hence $u \cap b^\bullet \neq \varnothing$. Suppose that there is $f \neq e$ such that $f \in u \cap b^\bullet$. Then we have $f \in u$ and $b \in {}^\bullet f$ which implies $b \in {}^\bullet f \cap {}^\bullet e$. We obtained a contradiction with $e, f \in u \in \mathbb{U}_{\mathfrak{enl}}$. Hence $|u \cap b^\bullet| = 1$.
From $b \notin c'$ and $c' \setminus c = u^\bullet$, we have $b \notin u^\bullet$. Let $e \in u$ ($u \neq \varnothing$ by definition). Then $b \notin e^\bullet$, and therefore $e \notin {}^\bullet b$. Hence $|u \cap {}^\bullet b| = 0$.

Proof of definition 1(2) for $\mathfrak{r}_b$. Can be proved similarly as definition 1(1).

Proof of definition 1(3) for $\mathfrak{r}_b$. We need to show that $u \cap b^\bullet \neq \varnothing$ implies $c \in \|\mathfrak{r}_b\|$ and $c' \notin \|\mathfrak{r}_b\|$.
From proposition 10, we have $c \setminus c' = {}^\bullet u$ and $c' \setminus c = u^\bullet$. From $u \cap b^\bullet \neq \varnothing$, we have that there is $e \in u$ such that $e \in b^\bullet$, and so $b \in {}^\bullet e$. Consequently, $b \in {}^\bullet u = c \setminus c'$, and so $b \in c$ and $b \notin c'$. We therefore obtained that $c \in \|\mathfrak{r}_b\|$ and $c' \notin \|\mathfrak{r}_b\|$.

Proof of definition 1(4) for $\mathfrak{r}_b$. Can be proved similarly as definition 1(3).
Clearly, if $\varnothing \neq \|\mathfrak{r}_b\| \neq C_{\mathfrak{enl}}$ then $\mathfrak{r}_b$ is a non-trivial io-region.

We may now proceed with the proof proper.

Proof of (A1). Follows directly from the definition of $C_{\mathfrak{enl}}$.

Proof of (A2). We observe that if $e \in \mathcal{E}_{\mathfrak{ts}_{\mathfrak{enl}}}$ then $\{\mathfrak{r}_b \mid b \in {}^\bullet e\} \subseteq {}^\circ e$ and $\{\mathfrak{r}_b \mid b \in e^\bullet\} \subseteq e^\circ$ (follows from the definitions of ${}^\circ e$, e° and $\mathfrak{r}_b$). This and ${}^\bullet e \neq \varnothing \neq e^\bullet$ yields ${}^\circ e \neq \varnothing \neq e^\circ$.

Proof of (A3). Suppose that $c \neq c'$ are two cases in $C_{\mathfrak{enl}}$. Without loss of generality, we may assume that there is $b \in c \setminus c'$. Hence $c \in \|\mathfrak{r}_b\|$ and $c' \notin \|\mathfrak{r}_b\|$. Thus, by the fact that $\mathfrak{r}_b$ is not trivial ($\varnothing \neq \|\mathfrak{r}_b\| \neq C_{\mathfrak{enl}}$) and $\mathfrak{r}_b \in \mathfrak{R}_c \setminus \mathfrak{R}_{c'}$, (A3) holds.

Proof of (A4). Suppose that $c \in C_{\mathfrak{enl}}$ and $u \in \mathbb{U}_{\mathfrak{ts}_{\mathfrak{enl}}}$ are such that ${}^\circ u \subseteq \mathfrak{R}_c$ and $u^\circ \cap \mathfrak{R}_c = \varnothing$ and there is no $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{ts}_{\mathfrak{enl}}}$ satisfying: $\mathfrak{L}(e) \in \mathfrak{L}(u)$ and ${}^\circ e \subseteq \mathfrak{R}_c$ and $e^\circ \cap \mathfrak{R}_c = \varnothing$. We need to show that $c \xrightarrow{u}_{\mathfrak{enl}}$.
First we show ${}^\bullet u \subseteq c$. Let $e \in u$. Consider $b \in {}^\bullet e$. We have already shown that this implies $\mathfrak{r}_b \in {}^\circ e$. From ${}^\circ u \subseteq \mathfrak{R}_c$, we have that $\mathfrak{r}_b \in \mathfrak{R}_c$, and so $c \in \|\mathfrak{r}_b\|$. Consequently, $b \in c$. Hence, for all $e \in u$ we have ${}^\bullet e \subseteq c$, and so ${}^\bullet u \subseteq c$.
Now we show that ${}^\bullet u \cap c = \varnothing$. Let $e \in u$. Consider $b \in e^\bullet$. We have already shown that this implies $\mathfrak{r}_b \in e^\circ$. From $u^\circ \cap \mathfrak{R}_c = \varnothing$, we have that $\mathfrak{r}_b \notin \mathfrak{R}_c$, and so $c \notin \|\mathfrak{r}_b\|$. Consequently, $b \notin c$. Hence, for all $e \in u$ we have $e^\bullet \cap c = \varnothing$, and so $u^\bullet \cap c = \varnothing$.
Now we need to prove that there is no step $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{enl}}$ satisfying: $\mathfrak{L}(e) \in \mathfrak{L}(u)$ and ${}^\bullet e \subseteq c$ and $e^\bullet \cap c = \varnothing$.
Suppose that this is not the case. Let $u \uplus \{e_1\} \in \mathbb{U}_{\mathfrak{enl}}$ be a step satisfying these conditions. Now we have two cases.
Case 1: There is no $u \uplus \{e_1\} \uplus \{f\} \in \mathbb{U}_{\mathfrak{enl}}$ such that $\mathfrak{L}(f) \in \mathfrak{L}(u \uplus \{e_1\})$ and ${}^\bullet f \subseteq c$ and $f^\bullet \cap c = \varnothing$. This implies $c \xrightarrow{u \uplus \{e_1\}}_{\mathfrak{enl}}$. By proposition 8, we have that $u \uplus \{e_1\} \in \mathbb{U}_{\mathfrak{ts}_{\mathfrak{enl}}}$. Moreover, $\mathfrak{L}(e_1) \in \mathfrak{L}(u)$ and, by proposition 7, we have ${}^\circ(u \uplus \{e_1\}) \subseteq \mathfrak{R}_c$ and $(u \uplus \{e_1\})^\circ \cap \mathfrak{R}_c = \varnothing$. We therefore obtained a contradiction with our assumptions.
Case 2: We can find $u \uplus \{e_1\} \uplus \{e_2\} \in \mathbb{U}_{\mathfrak{enl}}$ such that $\mathfrak{L}(e_2) \in \mathfrak{L}(u \uplus \{e_1\})$ and ${}^\bullet e_2 \subseteq c$ and $e_2^\bullet \cap c = \varnothing$. Then we consider Cases 1 and 2 again, taking $u \uplus \{e_1\} \uplus \{e_2\}$ instead of $u \uplus \{e_1\}$. Since the number of events in E is finite, we will eventually end up in Case 1. This means that, eventually, we will obtain a contradiction.

Proof of (A5). We need to show that, if $c \xrightarrow{u}_{\mathfrak{enl}}$ then there is no $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{ts}_{\mathfrak{enl}}}$ satisfying $\mathfrak{L}(e) \in \mathfrak{L}(u)$ and $^\circ e \subseteq \mathfrak{R}_c$ and $e^\circ \cap \mathfrak{R}_c = \varnothing$.
Suppose that there *is* $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{ts}_{\mathfrak{enl}}}$ satisfying $\mathfrak{L}(e) \in \mathfrak{L}(u)$ and $^\circ e \subseteq \mathfrak{R}_c$ and $e^\circ \cap \mathfrak{R}_c = \varnothing$ (†).

We have already shown that for $e \in \mathcal{E}_{\mathfrak{ts}_{\mathfrak{enl}}}$, $b \in {}^\bullet e$ implies $\mathfrak{r}_b \in {}^\circ e$, and $b \in e^\bullet$ implies $\mathfrak{r}_b \in e^\circ$. From this and $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{ts}_{\mathfrak{enl}}}$ we have $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{enl}}$.
We will show that ${}^\bullet e \subseteq c$. We have $b \in {}^\bullet e$ implies $\mathfrak{r}_b \in {}^\circ e$. But $^\circ e \subseteq \mathfrak{R}_c$, so $\mathfrak{r}_b \in \mathfrak{R}_c$. This means $c \in \|\mathfrak{r}_b\|$, and consequently, $b \in c$. Hence ${}^\bullet e \subseteq c$
We will show that $e^\bullet \cap c = \varnothing$. We have $b \in {}^\bullet e$ implies $\mathfrak{r}_b \in e^\circ$. But $e^\circ \cap \mathfrak{R}_c = \varnothing$, so $\mathfrak{r}_b \notin \mathfrak{R}_c$. This means $c \notin \|\mathfrak{r}_b\|$, and consequently, $b \notin c$. Hence $e^\bullet \cap c = \varnothing$.
As a result, assuming (†) leads to a contradiction with $c \xrightarrow{u}_{\mathfrak{enl}}$. □

3.2 ENL-systems generated by ENL-transition systems

The reverse translation, from an ENL-transition systems to ENL-systems, is based on the pre- and post-io-regions of events appearing in a transition system.

Let $\mathfrak{ts} = (S, T, s_{in})$ be an ENL-transition system. The net system *associated* with $\mathfrak{ts}$ is defined as

$$\mathfrak{enl}_{\mathfrak{ts}} \stackrel{\text{df}}{=} (\mathfrak{R}_{\mathfrak{ts}}, \mathcal{E}_{\mathfrak{ts}}, F_{\mathfrak{ts}}, \mathfrak{R}_{s_{in}})$$

where $F_{\mathfrak{ts}}$ is defined thus:

$$F_{\mathfrak{ts}} \stackrel{\text{df}}{=} \{(\mathfrak{r}, e) \in \mathfrak{R}_{\mathfrak{ts}} \times \mathcal{E}_{\mathfrak{ts}} \mid \mathfrak{r} \in {}^\circ e\} \;\cup\; \{(e, \mathfrak{r}) \in \mathcal{E}_{\mathfrak{ts}} \times \mathfrak{R}_{\mathfrak{ts}} \mid \mathfrak{r} \in e^\circ\}\,. \tag{1}$$

Proposition 11. *For every* $e \in \mathcal{E}_{\mathfrak{ts}}$*,* $^\circ e = {}^\bullet e$ *and* $e^\circ = e^\bullet$.

Proof. Follows directly from the definition of $\mathfrak{enl}_{\mathfrak{ts}}$. □

Note that the above construction produces a net which is saturated with conditions.

Theorem 2. $\mathfrak{enl}_{\mathfrak{ts}}$ *is an ENL-system.*

Proof. The only thing we need to observe is that, for every event e of $\mathcal{E}_{\mathfrak{ts}}$, it is the case that: ${}^\bullet e \neq \varnothing \neq e^\bullet$, which follows from (A2) and proposition 11; and ${}^\bullet e \cap e^\bullet = \varnothing$, which follows from propositions 6 and 11. □

We will now show that the ENL-system associated with an ENL-transition system $\mathfrak{ts}$ generates a transition system which is isomorphic to $\mathfrak{ts}$.

Proposition 12. *Let* $\mathfrak{ts} = (S, T, s_{in})$ *be an ENL-transition system and*

$$\mathfrak{enl} = \mathfrak{enl}_{\mathfrak{ts}} = (\mathfrak{R}_{\mathfrak{ts}}, \mathcal{E}_{\mathfrak{ts}}, F_{\mathfrak{ts}}, \mathfrak{R}_{s_{in}}) = (B, E, F, c_{in})$$

be the ENL-system associated with it.

1. $C_{\mathfrak{enl}} = \{\mathfrak{R}_s \mid s \in S\}$.
2. $\rightarrow_{\mathfrak{enl}} = \{(\mathfrak{R}_s, u, \mathfrak{R}_{s'}) \mid (s, u, s') \in T\}$.

Proof. Note that from the definition of $C_{\mathfrak{enl}}$, every $c \in C_{\mathfrak{enl}}$ is a case reachable from c_{in} in $\mathfrak{enl}$; and that from axiom (A1), every $s \in S$ is a state reachable from s_{in} in $\mathfrak{ts}$.

We first show that if $c \xrightarrow{u}_{\mathfrak{enl}} c'$ and $c = \mathfrak{R}_s$, for some $s \in S$, then there is $s' \in S$ such that $s \xrightarrow{u} s'$ and $c' = \mathfrak{R}_{s'}$. By $c \xrightarrow{u}_{\mathfrak{enl}} c'$, $u \in \mathbb{U}_{\mathfrak{enl}}$ is a step such that ${}^\bullet u \subseteq c$ and $u^\bullet \cap c = \varnothing$, and there is no step $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{enl}}$ satisfying $\mathfrak{L}(e) \in \mathfrak{L}(u)$ and ${}^\bullet e \subseteq c$ and $e^\bullet \cap c = \varnothing$. Moreover, $c' = (c \setminus {}^\bullet u) \cup u^\bullet$.
Hence, by proposition 11 and (A4), $u \in \mathbb{U}_{\mathfrak{ts}}$ and $s \xrightarrow{u} s'$, for some $s' \in S$. Then, by proposition 7, $\mathfrak{R}_{s'} = (\mathfrak{R}_s \setminus {}^\circ u) \cup u^\circ$. At the same time, we have $c' = (c \setminus {}^\bullet u) \cup u^\bullet$. Hence, by proposition 11 and $c = \mathfrak{R}_s$, we have that $c' = \mathfrak{R}_{s'}$.
As a result, we have shown (note that $c_{in} = \mathfrak{R}_{s_{in}} \in \{\mathfrak{R}_s \mid s \in S\}$) that

$$\begin{array}{rcl} C_{\mathfrak{enl}} & \subseteq & \{\mathfrak{R}_s \mid s \in S\} \\ \rightarrow_{\mathfrak{enl}} & \subseteq & \{(\mathfrak{R}_s, u, \mathfrak{R}_{s'}) \mid (s,u,s') \in T\} \,. \end{array}$$

We now prove the reverse inclusions. By definition, $\mathfrak{R}_{s_{in}} \in C_{\mathfrak{enl}}$. It is enough to show that if $s \xrightarrow{u} s'$ and $\mathfrak{R}_s \in C_{\mathfrak{enl}}$, then $\mathfrak{R}_{s'} \in C_{\mathfrak{enl}}$ and $\mathfrak{R}_s \xrightarrow{u}_{\mathfrak{enl}} \mathfrak{R}_{s'}$. By (A5) and propositions 7, 8 and 11, u is a valid step in $\mathfrak{enl}$ which is enabled at the case $\mathfrak{R}_s$. So, there is a case c' such that $\mathfrak{R}_s \xrightarrow{u}_{\mathfrak{enl}} c'$ and $c' = (\mathfrak{R}_s \setminus {}^\bullet u) \cup u^\bullet$. From propositions 7 and 11 we have that $c' = \mathfrak{R}_{s'}$. Hence we obtain that $\mathfrak{R}_s \xrightarrow{u}_{\mathfrak{enl}} \mathfrak{R}_{s'}$ and so also $\mathfrak{R}_{s'} \in C_{\mathfrak{enl}}$. □

Theorem 3. *Let $\mathfrak{ts} = (S, T, s_{in})$ be an ENL-transition system and $\mathfrak{enl} = \mathfrak{enl}_{\mathfrak{ts}}$ be the ENL-system associated with it. Then $\mathfrak{ts}_{\mathfrak{enl}}$ is isomorphic to $\mathfrak{ts}$.*

Proof. Let $\psi : S \rightarrow C_{\mathfrak{enl}}$ be a mapping given by $\psi(s) = \mathfrak{R}_s$, for all $s \in S$ (note that, by proposition 12(1), ψ is well-defined). We will show that ψ is an isomorphism for $\mathfrak{ts}$ and $\mathfrak{ts}_{\mathfrak{enl}}$.
Note that $\psi(s_{in}) = \mathfrak{R}_{s_{in}}$. By proposition 12(1), ψ is onto. Moreover, by (A3), it is injective. Hence ψ is a bijection. We then observe that, by proposition 12(2), we have $(s, u, s') \in T$ if and only if $(\psi(s), u, \psi(s')) \in \longrightarrow_{\mathfrak{enl}}$. Hence ψ is an isomorphism for $\mathfrak{ts}$ and $\mathfrak{ts}_{\mathfrak{enl}}$. □

4 Concluding Remarks

In this paper, we have completely characterise transition systems which can be generated by the elementary net systems with localities. In doing so, we followed the standard approach in which key relationships between a Petri net and its transition system are established via the notion of a region. The standard definition of regions is insufficient for the class of transition systems of ENL-systems, and we augmented it with some additional information, leading to the notion of an io-region.

We defined, and showed consistency of, two behaviour preserving translations between ENL-systems and their transition systems. As a result, we provided a solution to the synthesis problem of ENL-systems, which consists in constructing an ENL-system for a given transition system in such a way that the transition system of the former is isomorphic to the latter.

In this paper, we have completely characterised transition systems which can be generated by the elementary net systems with localities. In doing so, we followed the standard approach in which key relationships between a Petri net and its transition system are established via the notion of a region. The standard definition of regions is insufficient for the class of transition systems of ENL-systems, and we augmented it with some additional information, leading to the notion of an io-region.

We defined, and showed consistency of, two behaviour preserving translations between ENL-systems and their transition systems. As a result, we provided a solution to the synthesis problem of ENL-systems, which consists in constructing an ENL-system for a given transition system in such a way that the transition system of the former is isomorphic to the latter.

The previous work which appears to be closest to what has been proposed in this paper is due to Badouel and Darondeau [3]. In much more general a framework than the basic Elementary Net Systems it discusses the notion of a step transition system, generalising that introduced by Mukund [10]; in particular, by dropping the assumption that a transition system should exhibit the so-called *intermediate state property*:

$$s \xrightarrow{\alpha+\beta} s' \;\Rightarrow\; \exists s'' : s \xrightarrow{\alpha} s'' \xrightarrow{\beta} s' \;.$$

This clearly is a characteristic shared by the class of the ENL-transition systems. But the step transition systems of [3] still exhibit what one might call a *weak intermediate state property* (or *subset property*):

$$s \xrightarrow{\alpha+\beta} s' \;\Rightarrow\; \exists s'' : s \xrightarrow{\alpha} s'' \;.$$

However, this is a key property which is *not satisfied* by the ENL-transition systems. We feel that it is an important question to find out whether or to what extent the theory of Badouel and Darondeau [3] could be adopted to work for the ENL-transition systems and their extensions.

4.1 Future work

We believe that the notion of an io-region may be used to characterise transition systems of other extensions of EN-systems, as well as non-safe Petri nets (after suitable adaptations, of course). We now briefly outline some initial thoughts, which all boil down to suitable modifications of the last two axioms, (A4) and (A5).

Let us consider EN-systems with maximal concurrency semantics. In this case we do not consider localities, but only assume that all enabled steps are chosen according to the maximal concurrency paradigm.

A4a Let $s \in S$ and $u \in \mathbb{U}_{\mathfrak{ts}}$ be such that $^\circ u \subseteq \mathfrak{R}_s$ and $u^\circ \cap \mathfrak{R}_s = \varnothing$, and there is no $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{ts}}$ satisfying $^\circ e \subseteq \mathfrak{R}_s$ and $e^\circ \cap \mathfrak{R}_s = \varnothing$. Then $s \xrightarrow{u}$.

A5a If $s \xrightarrow{u}$ then there is no $u \uplus \{e\} \in \mathbb{U}_{\mathfrak{ts}}$ satisfying $^{\circ}e \subseteq \mathfrak{R}_s$ and $e^{\circ} \cap \mathfrak{R}_s = \varnothing$.

As a second example, we consider EN-systems with constrained parallelism. Again, in this case we do not consider localities, but rather assume that no enabled step can comprise less than m events and more than $n \in \mathbb{N} \cup \{\infty\}$ events, where $0 < m < n \leq \infty$.

A4b Let $s \in S$ and $u \in \mathbb{U}_{\mathfrak{ts}}$ be such that $^{\circ}u \subseteq \mathfrak{R}_s$ and $u^{\circ} \cap \mathfrak{R}_s = \varnothing$ and $m \leq |u| \leq n$. Then $s \xrightarrow{u}$.

A5b If $s \xrightarrow{u}$ then $m \leq |u| \leq n$.

As a third example, we consider EN-systems where there are two kinds of events, $\mathcal{E}_s$ and $\mathcal{E}_h$, modelling respectively software and hardware actions. It is also assumed that the occurrence of each software event $e \in \mathcal{E}_s$ is supported by one of the hardware events of a pre-defined set $supp_e \subseteq \mathcal{E}_h$.

A4c Let $s \in S$ and $u \in \mathbb{U}_{\mathfrak{ts}}$ be such that $^{\circ}u \subseteq \mathfrak{R}_s$ and $u^{\circ} \cap \mathfrak{R}_s = \varnothing$ and, for every $e \in u \cap \mathcal{E}_s$, it is the case that $u \cap supp_e \neq \varnothing$. Then $s \xrightarrow{u}$.

A5c If $s \xrightarrow{u}$ then, for every $e \in u \cap \mathcal{E}_s$, it is the case that $u \cap supp_e \neq \varnothing$.

Finally, without going into technical details, we feel that if the enabling relation for a class of EN-systems' extensions can be expressed by a formula which refers to pre-sets and post-set of steps, possibly using quantifiers without referring to specific conditions, then one can derive a suitable modification of the axioms (A4) and (A5) by suitably replacing references to pre- and post-sets by the corresponding references to pre- and post-io-regions.

References

1. Arnold A.: *Finite transition systems.* Prentice Hall International (1994).
2. Badouel E., Bernardinello L., Darondeau Ph.: *The synthesis problem for elementary net systems is NP-complete.* Theoretical Computer Science 186 (1997), 107-134.
3. Badouel E., Darondeau Ph.: *Theory of regions.* Lectures on Petri Nets I: Basic Models, Advances in Petri Nets, W. Reisig and G. Rozenberg (Eds.), Springer-Verlag, LNCS 1491 (1998), 529-586.
4. Bernardinello L., De Michelis G., Petruni K., Vigna S.: *On the synchronic structure of transition systems.* In: J.Desel (Ed.) Structures in Concurrency Theory, Berlin 1995, Workshops in Computing, Springer-Verlag (1995), 69-84.
5. Busi N., Pinna G.M.: *Synthesis of nets with inhibitor arcs.* Proc. of CONCUR'97, A. Mazurkiewicz and J. Winkowski (Eds.), Springer-Verlag, LNCS 1243 (1997), 151-165.
6. Ehrenfeucht A., Rozenberg G.: *Partial 2-structures; Part I: Basic notions and the representation problem, and Part II: State spaces of concurrent systems.* Acta Informatica 27 (1990), 315-368.
7. Keller R.M.: *Formal verification of parallel programs.* CACM 19 (1976), 371-389.
8. Kleijn H.C.M., Koutny M. and Rozenberg G.: *Towards a Petri net semantics for membrane systems* Proc. of WMC'05, R.Freund, G.Paun, G.Rozenberg and A.Salomaa (Eds.), Springer-Verlag, LNCS 3850 (2006), 292-309.

9. Montanari U., Rossi F.: *Contextual nets.* Acta Informatica 32 (1995), 545-596.
10. Mukund M.: *Petri nets and step transition systems.* International Journal of Foundations of Computer Science 3 (1992), 443-478.
11. Nielsen M., Rozenberg G., Thiagarajan P.S.: *Elementary transition systems.* Theoretical Computer Science 96 (1992), 3-33.
12. Pietkiewicz-Koutny M.: *The synthesis problem for elementary net systems with inhibitor arcs.* Fundamenta Informaticae 40 (1999), 251-283.
13. Pietkiewicz-Koutny M.: *Synthesising elementary net systems with inhibitor arcs from step transition systems.* Fundamenta Informaticae 50 (2002), 175-203.
14. Winskel G., Nielsen M.: *Models for concurrency.* In: S.Abramsky, Dov M.Gabbay and T.S.E.Maibaum (Eds.), Handbook of Logic in Computer Science 4 (1995), 1-148.