

# Detecting Emergent Phenomena in Cellular Automata Using Temporal Description Logics<sup>\*</sup>

Stathis Delivorias<sup>1</sup>, Haralampos Hatzikirou<sup>2</sup>,  
Rafael Peñaloza<sup>1,2</sup>, and Dirk Walther<sup>1,2</sup>

<sup>1</sup> TU Dresden, Theoretical Computer Science, Germany

<sup>2</sup> Center for Advancing Electronics Dresden, Germany

{Stathis.Delivorias, Haralampos.Hatzikirou}@tu-dresden.de

{Rafael.Penalaza, Dirk.Walther}@tu-dresden.de

**Abstract.** Cellular automata are discrete mathematical models that have been proven useful as representations of a wide variety of systems exhibiting emergent behavior. Detection of emergent behavior is typically computationally expensive as it relies on computer simulations. We propose to specify cellular automata using a suitable Temporal Description Logic and we show that we can formulate queries about the evolution of a cellular automaton as reasoning tasks in this logic.

## 1 Introduction

Cellular automata are discrete mathematical models that have been proven useful as representations of a wide variety of systems that feature non-linear dynamics. Initially cellular automata were introduced by John von Neumann as a formal model for cellular growth and replication. These automata have been successfully applied to model complex systems in physics, biology and many other scientific disciplines. Applications can be found in fluid dynamics, biological pattern formation, neural networks, cooperative systems, etc. Cellular automata have also been intensively studied in dynamical systems theory and computation theory, and they have been considered as discrete dynamical systems as well as a model for parallel computation [9, 4, 7].

We focus on complex systems exhibiting emergent behavior, where the properties at the larger scale are qualitatively different from those at the smaller scale. New functionalities may emerge when small components (like cells) are aggregated into larger entities (tissues and organs). We understand emergence as a property of a composite system (i.e., not just a component of the system) that arises through the interaction of the components and that persists over a period of time. Examples of emergence are plentiful and varied, and they typically involve some form of oscillation and pattern formation. Detection of emergent phenomena is typically computationally expensive as it involves the exploration of possible parameter settings and the corresponding phase space of the system.

---

<sup>\*</sup> This work was partially supported by the German Research Foundation (DFG) within the Cluster of Excellence ‘Center for Advancing Electronics Dresden’.

The challenge we encounter is the description of interesting temporal system properties and their verification in an automated way. The general aim of this work is to develop a logic-based modeling language that is suited for describing such phenomena but that still allows effective reasoning. Our method of specifying properties of the evolution of cellular automata is using a suitable Temporal Description Logic. Queries about the evolution of a cellular automaton can then be formulated as reasoning tasks in this logic.

Description Logics (DLs) are a well-known family of logic-based knowledge representation formalisms [2]. They are designed to describe and reason about static aspects of an application domain. DLs form the logical basis of the web ontology language OWL 2, which is the current recommendation of the World Wide Web Consortium (W3C) [8]. Ontology languages based on DLs have been widely adopted for modeling a large class of domains, and an increasing number of ontologies is made available in dedicated repositories. To represent dynamical patterns, combinations of DLs and temporal logics have been proposed resulting in Temporal Description Logics [1, 5]. In this paper, we describe cellular automata using the Temporal DL  $LTL_{\mathcal{ALCO}}$ , which is a combination of Linear-time Temporal Logic (LTL) [6] and the basic DL  $\mathcal{ALC}$  extended with nominals [2]. The idea is to use the standard reasoning tasks in this logic such as subsumption and satisfiability checking w.r.t. a knowledge base to derive properties of the evolution of the cellular automaton. In particular, we aim at describing and verifying emergent properties such as oscillation and drift.

Expressive Temporal DLs tend to exhibit high worst-case reasoning complexity [5]. However, it remains to be seen whether reasoning in practice is actually unfeasible. An alternative to using expressive logics is to design a Temporal DL that is expressive enough to enable us to specify interesting queries on cellular automata while, at the same time, allowing for efficient reasoning. Finding the right balance between expressivity and computational complexity is at the core of research in DL.

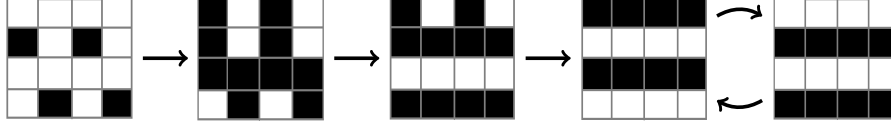
## 2 Cellular Automata

We consider two-dimensional synchronous cellular automata that are based on a regular rectangular finite lattice.

**Definition 1.** A 2-D finite regular cellular automaton is a tuple  $A = \langle S, n, f \rangle$  where  $S$  is a finite set of states,  $n \in \mathbb{N}$  is the diameter of the automaton, and  $f : S^{k \times k} \rightarrow S$  is the local transition function, where  $k \leq n$  is an odd number.  $\dashv$

We refer to 2-D finite regular cellular automata simply as CA, and we refer to matrix coordinates as *cells*.

**Definition 2.** A configuration of the CA  $A$  is a matrix  $C \in S^{n \times n}$ . Let  $m = \frac{k-1}{2}$  and  $\ell = n + 2m$ . A boundary condition for  $A$  is an operator  $*$ :  $S^{n \times n} \rightarrow S^{\ell \times \ell}$  such that for all  $C \in S^{n \times n}$  and all  $(i, j) \in \{1, \dots, n\}^2$ ,  $C^*[i + m, j + m] = C[i, j]$ .



**Fig. 1.** The temporal realization for the execution triplet  $(A_1, *, C_0)$ .

Given a boundary condition  $*$ , a configuration  $C$  and a cell  $(x, y) \in \{1, \dots, n\}^2$ , the neighborhood of  $(x, y)$  w.r.t.  $(A, C^*)$  is  $\mathbf{N}_{(x,y),C^*} \in S^{k \times k}$  with

$$\mathbf{N}_{(x,y),C^*}[i, j] = C^*[i + (x - 1), j + (y - 1)].$$

A function  $F : S^{n \times n} \rightarrow S^{n \times n}$  is the global transition function of  $A$  w.r.t.  $*$  if for all  $C \in S^{n \times n}$  it holds that  $F(C)[i, j] = f(\mathbf{N}_{(i,j),C^*})$ . The infinite sequence  $\mathbf{C} = (C_0, C_1, C_2, \dots)$  where  $C_i \in S^{n \times n}$  and  $C_i = F(C_{i-1})$  for all  $i \in \mathbb{N}$  is the temporal realization of the automaton, with  $C_0$  as its initial configuration.  $\dashv$

Since a temporal realization is uniquely determined by a cellular automaton  $A$ , a boundary condition  $*$  and an initial configuration  $C_0$ , we denote it with  $\mathbf{C}(A, *, C_0)$ . Moreover, we refer to  $(A, *, C_0)$  as *execution triplet*.

*Example 3.* Let  $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2$ , where

$$\mathcal{R}_1 = \left\{ \begin{pmatrix} x_1 & x_2 & x_3 \\ 0 & 1 & 0 \\ x_4 & x_5 & x_6 \end{pmatrix} \middle| x_i \in \mathbb{Z}_2 \text{ for } i \in \{1, \dots, 6\} \right\}, \text{ and}$$

$$\mathcal{R}_2 = \left\{ \begin{pmatrix} x_1 & x_2 & x_3 \\ x_4 & 0 & x_5 \\ x_6 & x_7 & x_8 \end{pmatrix} \middle| x_i \in \mathbb{Z}_2 \text{ for } i \in \{1, \dots, 8\}, x_2 = 1 \text{ or } x_7 = 1 \right\}.$$

Consider  $A_1 = \langle \mathbb{Z}_2, 4, f \rangle$  with  $f : \mathbb{Z}_2^{3 \times 3} \rightarrow \mathbb{Z}_2$  given by

$$f(N) = \begin{cases} 1 & \text{if } N \in \mathcal{R} \\ 0 & \text{if } N \notin \mathcal{R}, \end{cases}$$

the boundary condition  $*$ :  $\mathbb{Z}_2^{4 \times 4} \rightarrow \mathbb{Z}_2^{6 \times 6}$  where  $C^*[i, j] = 0$  holds for every  $(i, j) \in \{1, \dots, 6\}^2 \setminus \{2, 3, 4, 5\}^2$ , and the initial configuration

$$C_0 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}.$$

Figure 1 provides a graphical representation of the temporal realization of this CA. After the fourth step, the configuration cycles back to the third configuration, i.e.  $C_4 = C_3$ . This means that the remainder of the temporal realization cycles between two global states. This property is called a *global oscillation* (of period 2).  $\triangleleft$

We formally define some characteristic emergent properties of cellular automata.

**Definition 4.** *Let  $\Lambda$  be a CA. The temporal realization  $\mathbf{C}(\Lambda, *, C_0)$  results in a global oscillation of period  $\tau > 0$  if there exists a  $t_0 \in \mathbb{N}$  such that for every  $t_1 \geq t_0$ ,  $C_{t_1} = C_{t_1+\tau}$ . We call the smallest such  $t_0$  the starting moment of the global oscillation.*  $\dashv$

Every temporal realization of a CA  $\Lambda$  results in a global oscillation of some period. The total number of configurations of  $\Lambda = \langle S, n, f \rangle$  is  $|S|^{n^2}$ . Since every configuration of  $\Lambda$  has a unique successor, the starting moment  $t_0$  of the global oscillation of a temporal realization satisfies  $t_0 < |S|^{n^2}$ .

**Definition 5.** *Let  $\Lambda$  be a CA. The temporal realization  $\mathbf{C}(\Lambda, *, C_0)$  leads to a fixed point if it results in a global oscillation of period 1.  $\mathbf{C}(\Lambda, *, C_0)$  yields a phase transition if it leads to a fixed point with starting moment  $t_0$ , where  $C_{t_0}[i, j] = C_{t_0}[i', j']$  for all  $(i, j), (i', j') \in \{1, \dots, n\}^2$ .*  $\dashv$

We now discuss oscillating patterns that appear only at a part of the grid. We use the notion of a *block element* of a matrix, which is a connected submatrix. Let  $M$  be an  $n \times n$  matrix. An  $s_1 \times s_2$  matrix  $M_0$ , for  $s_1, s_2 \leq n$ , is a *block element* of  $M$  if there is a tuple  $(x, y, s_1, s_2) \in \{1, \dots, n\}^4$  such that  $M_0[i, j] = M[x-1+i, y-1+j]$  for all  $(i, j) \in \{1, \dots, s_1\} \times \{1, \dots, s_2\}$ . The tuple  $(x, y, s_1, s_2)$  denotes the *block coordinates* of  $M_0$  in  $M$ , and  $C(x, y, s_1, s_2)$  denotes the block element of the matrix  $C$  with block coordinates  $(x, y, s_1, s_2)$ . Note that  $x$  and  $y$  describe the position of the block element, while  $s_1$  and  $s_2$  refer to the size of the block element. We also represent the tuple  $(x, y, s_1, s_2)$  with two vectors  $\mathbf{x} = \langle x, y \rangle$ ,  $\mathbf{s} = \langle s_1, s_2 \rangle$ , and we write  $C(\mathbf{x}, \mathbf{s})$ .

**Definition 6.** *The temporal realization  $\mathbf{C}(\Lambda, *, C_0)$  results in a localized oscillation of period  $\tau$  if there exist a  $t_0 \in \mathbb{N}$  and block coordinates  $(x, y, s_1, s_2)$  such that for every  $t_1 \geq t_0$ , it holds that  $C_{t_1}(x, y, s_1, s_2) = C_{t_1+\tau}(x, y, s_1, s_2)$ .*  $\dashv$

We also consider the drifting of a *particle* during a temporal realization of a CA, where by particle we mean a structure of oscillating size and shape. To this end, we require the additional notions of velocity, trajectory and size oscillation. A *velocity function* with period  $\tau$  is a function  $\mathbf{v} : \mathbb{N} \rightarrow \mathbb{Z}^2$  such that  $\mathbf{v}(t+\tau) = \mathbf{v}(t)$  for all  $t \in \mathbb{N}$ . If  $\mathbf{v}$  is a velocity function of period  $\tau$ , given an *initial position*  $\mathbf{x}_0 \in \mathbb{N}^2$ , a function  $\mathbf{g} : \mathbb{N} \rightarrow \mathbb{Z}^2$  with  $\mathbf{g}(0) = \mathbf{x}_0$  and  $\mathbf{g}(t) = \mathbf{x}_0 + \sum_{\kappa=0}^{t-1} \mathbf{v}(\kappa)$  for  $t > 0$ , is called a *drifting trajectory*. A *size oscillation* of period  $\tau$  is a function  $\mathbf{s} : \mathbb{N} \rightarrow \mathbb{N}^2$  such that  $\mathbf{s}(t+\tau) = \mathbf{s}(t)$  for all  $t \in \mathbb{N}$ .

**Definition 7.** *Let  $\Lambda$  be a CA. The temporal realization  $\mathbf{C}(\Lambda, *, C_0)$  exhibits a drift of period  $\tau$  for  $t \geq \tau$  time steps if there exist a  $t_0 \in \mathbb{N}$ , a drifting trajectory  $\mathbf{g}$  and a size oscillation  $\mathbf{s}$  such that for every  $t_1 \in \mathbb{N}$  with  $t_0 \leq t_1 < t_0 + t - \tau$ , it holds that  $C_{t_1}^*(\mathbf{g}(t_1 - t_0), \mathbf{s}(t_1 - t_0)) = C_{t_1+\tau}^*(\mathbf{g}(t_1 - t_0 + \tau), \mathbf{s}(t_1 - t_0 + \tau))$ .*  $\dashv$

### 3 Temporal Description Logic

We use  $LTL_{\mathcal{ALCCO}}$  to describe CA. As we will see, there is a clear and strong correlation between a certain class of knowledge bases defined in this language, and the cellular automata described in the previous section.

Let  $N_C$ ,  $N_R$  and  $N_I$  be mutually disjoint sets of *concept*-, *role*- and *individual names*, respectively. Concepts  $C$  in  $LTL_{\mathcal{ALCCO}}$  are built using the grammar rule:

$$C ::= \top \mid A \mid \{a\} \mid \neg C \mid C \sqcap C \mid \exists r.C \mid \bigcirc C \mid CU C$$

where  $A \in N_C$ ,  $r \in N_R$ , and  $a \in N_I$ . The symbols  $\top$  (“truth”),  $\{a\}$  (“nominal”),  $\neg$  (“negation”),  $\sqcap$  (“conjunction”) and  $\exists$  (“existential restriction”) are logical operators of the Description Logic  $\mathcal{ALCCO}$ , whereas  $\bigcirc$  (“next”) and  $\mathcal{U}$  (“until”) are temporal operators of the Temporal Logic LTL. We use the usual abbreviations  $\perp = \neg\top$  (“falseness”),  $C \sqcup D = \neg(\neg C \sqcap \neg D)$  (“disjunction”),  $\forall r.C = \neg\exists r.\neg C$  (“universal restriction”),  $\diamond C = \top \mathcal{U} C$  (“eventually”), and  $\square C = \neg\diamond\neg C$  (“always”).

An expression of the form  $C \sqsubseteq D$ , where  $C$  and  $D$  are concepts, is called a *concept inclusion*. A finite set of concept inclusions is a *TBox*. An expression of the form  $C(a)$ , where  $C$  is a concept and  $a \in N_I$ , is called a *concept assertion*. For  $r \in N_R$  and  $a, b \in N_I$ , an expression of the form  $r(a, b)$  is called a *role assertion*, and  $\square r(a, b)$  is a *rigid role assertion*. A finite set of concept and role assertions is called an *ABox*. A pair  $\mathcal{K} = (\mathcal{T}, \mathcal{A})$  consisting of an TBox  $\mathcal{T}$  and an ABox  $\mathcal{A}$  is called a *knowledge base*.

We are interested in describing the evolution of a CA. In particular, we want to describe how the states of its cells evolve over time.  $LTL_{\mathcal{ALCCO}}$  can naturally express relevant properties of CAs, as it is illustrated by the following example.

*Example 8.* The fact that every cell eventually reaches a fixed point can be expressed in  $LTL_{\mathcal{ALCCO}}$  using the concept inclusion

$$\top \sqsubseteq \diamond \square B \sqcup \diamond \square \neg B,$$

stating that every cell ( $\top$ ) is either ( $\sqcup$ ) at some point in the future ( $\diamond$ ) going to stay forever ( $\square$ ) black, or it will eventually remain forever white ( $\neg B$ ). On the other hand, the property that no cell reaches a fixed point can be expressed as

$$\top \sqsubseteq \square \diamond B \sqcap \square \diamond \neg B.$$

To express that after  $t$  transitions, every cell has at least one neighbor that is black we use

$$\top \sqsubseteq \bigcirc^t \exists r_n.B,$$

where  $\bigcirc^t$  indicates  $t$  steps and  $\exists r_n.B$  the existence of a black neighbor.  $\triangleleft$

We want to use individual names to refer to the cells of a CA. As the set of cells does not change over time, we use a fixed-domain semantics for  $LTL_{\mathcal{ALCCO}}$ . A *temporal interpretation* is a pair  $\mathcal{I} = (\Delta, \cdot^{\mathcal{I}})$ , where  $\Delta$  is a non-empty *domain*

and  $\cdot^{\mathcal{I}}$  is a function that maps every  $a \in N_I$  to  $a^{\mathcal{I}} \in \Delta$ , under the unique name assumption (i.e., no two individual names are mapped to the same domain element); every  $A \in N_C$  to  $A^{\mathcal{I}} \subseteq \mathbb{N} \times \Delta$ , and every  $r \in N_R$  to  $r^{\mathcal{I}} \subseteq \mathbb{N} \times \Delta \times \Delta$ . Then  $\cdot^{\mathcal{I}}$  is extended to concepts by setting  $\top^{\mathcal{I}} = \mathbb{N} \times \Delta$ ,  $(\neg C)^{\mathcal{I}} = (\mathbb{N} \times \Delta) \setminus C^{\mathcal{I}}$ ,  $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$ ,  $\{a\}^{\mathcal{I}} = \mathbb{N} \times \{a^{\mathcal{I}}\}$ ,

$$\begin{aligned} (\exists r.C)^{\mathcal{I}} &= \{(t, x) \in \mathbb{N} \times \Delta \mid \exists y \in \Delta. (t, x, y) \in r^{\mathcal{I}} \wedge (t, y) \in C^{\mathcal{I}}\}, \\ (\bigcirc C)^{\mathcal{I}} &= \{(t, x) \in \mathbb{N} \times \Delta \mid (t+1, x) \in C^{\mathcal{I}}\}, \text{ and} \\ (CUD)^{\mathcal{I}} &= \{(t, x) \mid \exists t_0 > t. (t_0, x) \in D^{\mathcal{I}} \wedge \forall t' : t \leq t' < t_0. (t', x) \in C^{\mathcal{I}}\}. \end{aligned}$$

A temporal interpretation  $\mathcal{I}$  *satisfies* a concept  $C$  if  $C^{\mathcal{I}} \neq \emptyset$ ;  $\mathcal{I}$  *satisfies* a concept inclusion  $C \sqsubseteq D$  if  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ ;  $\mathcal{I}$  *satisfies* a concept assertion  $C(a)$  if  $(0, a^{\mathcal{I}}) \in C^{\mathcal{I}}$ , a role assertion  $r(a, b)$  if  $(0, a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$  and a rigid role assertion  $\Box r(a, b)$  if  $(t, a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ , for every  $t \in \mathbb{N}$ . We say that  $\mathcal{I}$  is a *model* of a TBox  $\mathcal{T}$  or an ABox  $\mathcal{A}$  if it satisfies every concept inclusion in  $\mathcal{T}$  or every assertion in  $\mathcal{A}$ , respectively; and  $\mathcal{I}$  is a model of a KB  $\mathcal{K} = (\mathcal{T}, \mathcal{A})$  if  $\mathcal{I}$  is a model of both,  $\mathcal{T}$  and  $\mathcal{A}$ . If there is a model of a KB  $\mathcal{K}$  that satisfies  $C$ , then  $C$  is *satisfiable* w.r.t.  $\mathcal{K}$ . If every model of  $\mathcal{K}$  satisfies  $C \sqsubseteq D$ , then  $C$  is *subsumed* by  $D$  w.r.t.  $\mathcal{K}$ .

In the next section, we show a correspondence between CA as defined in Section 2 and  $LTL_{ACC\mathcal{O}}$  KBs describing them. To this end, we require the following notion of isomorphism.

**Definition 9.** *Two temporal interpretations  $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ ,  $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$  are isomorphic if there is a bijection  $h : \Delta^{\mathcal{I}} \rightarrow \Delta^{\mathcal{J}}$ , called isomorphism, such that  $h(a^{\mathcal{I}}) = a^{\mathcal{J}}$  for every  $a \in N_I$ , and for all  $x, y \in \Delta^{\mathcal{I}}$ , and  $t \in \mathbb{N}$ , it holds that:*

- (a)  $(t, x) \in A^{\mathcal{I}}$  iff  $(t, h(x)) \in A^{\mathcal{J}}$ , for every  $A \in N_C$ ; and
- (b)  $(t, x, y) \in r^{\mathcal{I}}$  iff  $(t, h(x), h(y)) \in r^{\mathcal{J}}$ , for every  $r \in N_R$ . ↯

**Lemma 10.** *If  $\mathcal{I}$  and  $\mathcal{J}$  are two isomorphic temporal interpretations, then:*

- (i) *A concept  $C$  is satisfied by  $\mathcal{I}$  iff  $C$  is satisfied by  $\mathcal{J}$ ;*
- (ii)  *$\mathcal{I}$  satisfies  $C \sqsubseteq D$  iff  $\mathcal{J}$  satisfies  $C \sqsubseteq D$ ; and*
- (iii)  *$\mathcal{I}$  is a model for a TBox  $\mathcal{T}$ , an ABox  $\mathcal{A}$  or a KB  $\mathcal{K}$  iff  $\mathcal{J}$  is a model for  $\mathcal{T}$ ,  $\mathcal{A}$  or  $\mathcal{K}$ , respectively. ↯*

## 4 Reasoning about CA with TDL

In the following, we focus on a specific class of CA. We use  $\mathbb{Z}_2$  as the set of states and a neighborhood of size  $3 \times 3$ . The boundary condition  $*$  is such that for all configurations  $C$ ,  $C^*[i, j] = 0$  for every  $(i, j) \in \{1, \dots, n+2\}^2 \setminus \{2, \dots, n+1\}^2$ , where  $n$  is the diameter of the automaton. Hence, from now on, when referring to a CA  $\mathcal{A}$  we mean a tuple of the form  $\langle \mathbb{Z}_2, n, f \rangle$ , where  $f : \mathbb{Z}_2^{3 \times 3} \rightarrow \mathbb{Z}_2$ . We use the following signature to describe such a CA  $\mathcal{A}$ :

$$N_I = \{a_{xy} \mid x, y \in \{1, \dots, \ell\}\},$$

$$N_C = \{B\},$$

$$N_R = \{r_r, r_l, r_d, r_u\},$$

where  $\ell = n+2$  is a parameter that depends on the diameter  $n$  of  $A$ . The intended meaning of these symbols is as follows. The individual names in  $N_I$  represent the cells of the automaton, the concept name  $B$  states that a cell is black (formally, its state is  $1 \in \mathbb{Z}_2$ ) and the four role names in  $N_R$  represent the right, left, up and down neighbors, respectively, as they are found in the neighborhood matrix. We use a symbol for the bijection between  $\mathbb{Z}_2$  and  $\{B, \neg B\}$ . Let  $\chi: \{B, \neg B\} \rightarrow \mathbb{Z}_2$  with  $\chi(B) = 1$  and  $\chi(\neg B) = 0$ .

A *neighborhood concept* is a complex concept of the form, where  $X_i$ , for  $1 \leq i \leq 9$ , ranges over  $\{B, \neg B\}$ :

$$\begin{aligned} & \exists r_l. \exists r_u. X_1 \sqcap \exists r_u. X_2 \sqcap \exists r_r. \exists r_u. X_3 \sqcap \exists r_l. X_4 \sqcap X_5 \sqcap \\ & \sqcap \exists r_r. X_6 \sqcap \exists r_l. \exists r_d. X_7 \sqcap \exists r_d. X_8 \sqcap \exists r_r. \exists r_d. X_9. \end{aligned}$$

We denote such concepts as  $N(M)$  where  $M = \begin{pmatrix} \chi(X_1) & \chi(X_2) & \chi(X_3) \\ \chi(X_4) & \chi(X_5) & \chi(X_6) \\ \chi(X_7) & \chi(X_8) & \chi(X_9) \end{pmatrix}$ .

**Definition 11.** Let  $(A, *, C_0)$  be an execution triplet. The knowledge base for  $(A, *, C_0)$  is defined as  $\mathcal{K}_{(A, *, C_0)} = (\mathcal{T}_{(A, *, C_0)}, \mathcal{A}_{(A, *, C_0)})$ , where

$$\begin{aligned} \mathcal{T}_{(A, *, C_0)} &= \{N(M) \sqsubseteq \bigcirc X \mid M \in \mathbb{Z}_2^{3 \times 3}, X = \chi^{-1}(f(M))\} \cup \\ & \quad \{\top \sqsubseteq \{a_{ij} \mid i, j \in \{1, \dots, \ell\}\}\}, \\ \mathcal{A}_{(A, *, C_0)} &= \{\Box r_r(a_{xy}, a_{zw}) \mid x, y, z, w \in \{1, \dots, \ell\}, x = z, w = y + 1\} \cup \\ & \quad \{\Box r_l(a_{xy}, a_{zw}) \mid x, y, z, w \in \{1, \dots, \ell\}, x = z, w = y - 1\} \cup \\ & \quad \{\Box r_d(a_{xy}, a_{zw}) \mid x, y, z, w \in \{1, \dots, \ell\}, y = w, z = x + 1\} \cup \\ & \quad \{\Box r_u(a_{xy}, a_{zw}) \mid x, y, z, w \in \{1, \dots, \ell\}, y = w, z = x - 1\} \cup \\ & \quad \{(\Box \forall r. \perp)(a) \mid r \in N_R, \Box r(a, b) \notin \mathcal{A}_1, a, b \in N_I\} \cup \\ & \quad \{X(a_{ij}) \mid i, j \in \{2, \dots, \ell - 1\}, X = \chi^{-1}(C_0[i - 1, j - 1])\} \cup \\ & \quad \{\Box \neg B(a_{ij}) \mid (i, j) \in \{1, \dots, \ell\}^2 \setminus \{2, \dots, \ell - 1\}^2\}. \quad \dashv \end{aligned}$$

We show below that there is a one-to-one correspondence between the knowledge base for  $(A, *, C_0)$  and the temporal realization of this execution triplet. To this end, we employ the notion of a canonical interpretation defined as follows.

**Definition 12.** The canonical interpretation for the execution triplet  $(A, *, C_0)$  is defined as  $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ , where  $\Delta^{\mathcal{J}} = \{1, \dots, \ell\}^2$ ,  $B^{\mathcal{J}} = \{(t, (i, j)) \mid C_t^*[i, j] = 1\}$ ,  $a_{ij}^{\mathcal{J}} = (i, j)$  for every  $a_{ij} \in N_I$ , and

$$\begin{aligned} r_r^{\mathcal{J}} &= \{(t, (x, y), (z, w)) \mid t \in \mathbb{N}, x = z, w = y + 1\}, \\ r_l^{\mathcal{J}} &= \{(t, (x, y), (z, w)) \mid t \in \mathbb{N}, x = z, w = y - 1\}, \\ r_d^{\mathcal{J}} &= \{(t, (x, y), (z, w)) \mid t \in \mathbb{N}, y = w, z = x + 1\}, \\ r_u^{\mathcal{J}} &= \{(t, (x, y), (z, w)) \mid t \in \mathbb{N}, y = w, z = x - 1\}. \quad \dashv \end{aligned}$$

First we show that this canonical interpretation is the unique model of  $\mathcal{K}_{(A, *, C_0)}$ , up to isomorphism.

**Lemma 13.** *Let  $(\Lambda, *, C_0)$  be an execution triplet. The canonical interpretation  $\mathcal{J}$  is a model of  $\mathcal{K}_{(\Lambda, *, C_0)}$ , and every model of  $\mathcal{K}_{(\Lambda, *, C_0)}$  is isomorphic to  $\mathcal{J}$ .  $\dashv$*

We now show how to decide the existence of emergent properties in CA through standard reasoning in  $\text{LTL}_{\mathcal{ALCCO}}$ . We use the following notation:  $\bigcirc^1 := \bigcirc$  and  $\bigcirc^{n+1} := \bigcirc \bigcirc^n$ ,  $(\exists r)^1 := \exists r$  and  $(\exists r)^{n+1} := \exists r.(\exists r)^n$ , for all  $n \in \mathbb{N}$ . In addition, for any concept  $C$ , we set  $(\exists r)^0.C := C$  and  $\bigcirc^0 C := C$ .

**Proposition 14.** *Let  $\Lambda$  be a CA. The temporal realization  $\mathbf{C}(\Lambda, *, C_0)$  results in a global oscillation of period  $\tau$  iff  $\top$  is subsumed by  $\diamond \square ((B \sqcap \bigcirc^\tau B) \sqcup (\neg B \sqcap \bigcirc^\tau \neg B))$  w.r.t.  $\mathcal{K}_{(\Lambda, *, C_0)}$ .  $\dashv$*

**Corollary 15.** *The temporal realization  $\mathbf{C}(\Lambda, *, C_0)$  yields a fixed point iff  $\top$  is subsumed by  $\diamond \square ((B \sqcap \bigcirc B) \sqcup (\neg B \sqcap \bigcirc \neg B))$  w.r.t.  $\mathcal{K}_{(\Lambda, *, C_0)}$ .  $\mathbf{C}(\Lambda, *, C_0)$  yields a phase transition iff  $\top$  is subsumed by either  $\diamond \square (B \sqcap \bigcirc B)$  or by  $\diamond \square (\neg B \sqcap \bigcirc \neg B)$  w.r.t.  $\mathcal{K}_{(\Lambda, *, C_0)}$ .  $\dashv$*

The complex concept  $D_\tau = ((B \sqcap \bigcirc^\tau B) \sqcup (\neg B \sqcap \bigcirc^\tau \neg B))$  for arbitrary  $\tau \in \mathbb{N}$  will be used to model local oscillation. We call it *individual  $\tau$ -repetition*. We can then define the concept of  *$(i, j)$ -sized  $\tau$ -repetition* inductively:  $D_\tau^{1 \times 1} = D_\tau$ ,  $D_\tau^{(i+1) \times 1} = D_\tau^{i \times 1} \sqcap (\exists r_d)^i . D_\tau$  and  $D_\tau^{i \times (j+1)} = D_\tau^{i \times j} \sqcap (\exists r_r)^j . D_\tau^{i \times 1}$ .

**Proposition 16.** *Let  $\Lambda$  be a CA and  $\mathbf{C}(\Lambda, *, C_0)$  be a temporal realization.  $\mathbf{C}(\Lambda, *, C_0)$  results in local oscillation of period  $\tau$  iff  $\mathcal{K}_{(\Lambda, *, C_0)}$  satisfies*

$$\diamond \bigsqcup_{(i,j) \in \{1, \dots, n\}^2} \square D_\tau^{i \times j} \quad \dashv$$

Let  $\mathbf{r} = (r_1, r_2, \dots, r_n)$  be a finite sequence of role symbols. We call  $\mathbf{r}$  a *composite role* and we abbreviate with  $\exists \mathbf{r}$  the syntactic entity of the form  $\exists r_1. \exists r_2 \dots \exists r_n$ . Let  $n = |\mathbf{r}|$  be the *length* of  $\mathbf{r}$ . We consider the empty sequence as a composite role of length 0. To model the drift property we use the complex concept  $D_{\tau, \mathbf{r}} = ((B \sqcap \bigcirc^\tau \exists \mathbf{r}. B) \sqcup (\neg B \sqcap \bigcirc^\tau \exists \mathbf{r}. \neg B))$  where  $\mathbf{r}$  is a composite role and  $\tau \in \mathbb{N}$ . We call it *individual  $\mathbf{r}$ -drifting  $\tau$ -repetition*. The concept of  *$(i, j)$ -sized  $\mathbf{r}$ -drifting  $\tau$ -repetition* is then defined inductively:  $D_{\tau, \mathbf{r}}^{1 \times 1} = D_{\tau, \mathbf{r}}$ ,  $D_{\tau, \mathbf{r}}^{(i+1) \times 1} = D_{\tau, \mathbf{r}}^{i \times 1} \sqcap (\exists r_d)^i . D_{\tau, \mathbf{r}}^{1 \times 1}$  and  $D_{\tau, \mathbf{r}}^{i \times (j+1)} = D_{\tau, \mathbf{r}}^{i \times j} \sqcap (\exists r_r)^j . D_{\tau, \mathbf{r}}^{i \times 1}$ .

**Proposition 17.** *Let  $\Lambda$  be a CA. The temporal realization  $\mathbf{C}(\Lambda, *, C_0)$  exhibits a drift of period  $\tau$  for  $t$  time steps iff  $\mathcal{K}_{(\Lambda, *, C_0)}$  satisfies*

$$\diamond \bigsqcup_{|\mathbf{v}| < n^2} \prod_{t_1=0}^{\tau-1} \bigsqcup_{|\mathbf{p}| < n^2} \left( \bigsqcup_{(i,j) \in \{1, \dots, n\}^2} \left( \prod_{t_2=0}^{\mu} \bigcirc^{t_1+t_2\tau} (\exists \mathbf{v})^{t_2} . \exists \mathbf{p} . D_{\tau, \mathbf{r}}^{i \times j} \right) \right)$$

where  $\mathbf{p}$  and  $\mathbf{r}$  are composite roles and  $\mu = \lfloor \frac{t-t_1}{\tau} \rfloor$ .  $\dashv$



The study of combinations of Temporal Logics and Description Logics has a long history. In particular, the combination of the basic DL  $\mathcal{ALC}$  with LTL, and some of their extensions, has been studied in [3, 10]. For these logics, it has been shown that all the standard reasoning problems, such as satisfiability and subsumption between concepts, are EXPSPACE-complete. For our reductions, we have included two kinds of additional expressivity into our temporal DL: nominals and rigid role assertions.

At first sight, the use of rigid role assertions may seem problematic. Indeed, it is known that reasoning in  $LTL_{\mathcal{ALC}}$  w.r.t. TBoxes is undecidable whenever rigid roles are allowed; even if no TBox is used, the problem is still non-elementary [3]. However, in our case, the *rigidness* of a role is limited to the few named individuals that describe the cells of the automaton. Indeed, the rigid role assertion  $\Box r(a, b)$  is merely a syntactic variant of the concept inclusion  $\{a\} \sqsubseteq \exists r.\{b\}$ . The only remaining question is the cost of including nominals into our formalism.

To the best of our knowledge, the precise complexity of reasoning in  $LTL_{\mathcal{ALCO}}$  has never been settled. While we provide no formal proof for this, we conjecture that reasoning in  $LTL_{\mathcal{ALCO}}$  should not be harder than for  $LTL_{\mathcal{ALC}}$ , i.e., it should remain EXPSPACE-complete. This conjecture is supported by the fact that typically nominals do not increase the complexity of reasoning in description logics. Moreover, at this high complexity, it is possible to guess exponentially large structures without leaving the complexity class. Unfortunately, for practical matters, the efficiency of reasoning algorithms is typically negatively affected by the amount of nominals used. Whether this is indeed an issue in our case and whether it can be improved upon remains a topic for future work.

## 5 Conclusions

We have proposed to describe cellular automata using a Temporal Description Logic. Queries about the temporal evolution of a cellular automaton can then be formulated as reasoning tasks in this logic. In particular, we have described the emergent properties of global oscillation, local oscillation, and drift in the Temporal DL  $LTL_{\mathcal{ALCO}}$ . We have shown that verifying these properties w.r.t. a cellular automaton can be formulated as subsumption and satisfiability checking problems in that logic (cf. Propositions 14, 16 and 17).

There are also other relevant properties of a cellular automaton that would be interesting to describe and verify, which require to define different reasoning tasks for the Temporal DL. For instance, properties of the local transition rules that hold for all initial conditions, or derive the set of initial conditions that lead to a given configuration.

The study of computational complexity of the reasoning tasks can possibly help to further classify cellular automata in terms of the types of queries that can be answered within certain resource bounds. For future work, it would be interesting to analyze in detail the computational complexity of  $LTL_{\mathcal{ALCO}}$ , and also to find other logics that possibly offer a better compromise between expressivity and computational complexity. It is also important to check whether the

full expressivity of  $LTL_{ALCO}$  is needed for expressing the class of CA that we consider, and whether the same results can be achieved using fewer or different logical operators.

An interesting potential application could be the prediction of catastrophic events, here regarded as emergent behavior, in multi-component dynamic systems. To illustrate our perspective, electronic malfunctions can emerge from micro-circuit interactions. Typically, CA can be used to model such systems. Monitoring is the golden standard for identifying potential malfunctions. Our envisaged logic-based language, combined with system monitoring, could identify signatures in the CA rules and improve the predictability of such catastrophic events.

Another prospective application motivated by Temporal DL reasoning on cellular automata could be in clinical praxis. In particular, cardiograms or encephalograms provide a short-time observation of hearts or brains, respectively. Extensions of our methodology could potentially extract essential properties of that system and the reasoner could provide a prediction of dangerous emergent behaviour such as heart attacks or epileptic seizures.

## References

1. A. Artale, R. Kontchakov, C. Lutz, F. Wolter, and M. Zakharyashev. Temporalising tractable description logics. In *Proc. of TIME'07: the 14th International Symposium on Temporal Representation and Reasoning*, pages 11–22. IEEE Computer Society Press, 2007.
2. F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The description logic handbook: theory, implementation, and applications*. Cambridge University Press, 2007.
3. D. Gabbay, A. Kurusz, F. Wolter, and M. Zakharyashev. *Many-Dimensional Modal Logics: Theory and Applications*. Elsevier, 2003.
4. P. D. Lena and L. Margara. Computational complexity of dynamical systems: The case of cellular automata. *Information and Computation*, 206(9-10):1104–1116, 2008.
5. C. Lutz, F. Wolter, and M. Zakharyashev. Temporal description logics: A survey. In *Proc. of TIME'08: the 15th International Symposium on Temporal Representation and Reasoning*, pages 3–14. IEEE Computer Society Press, 2008.
6. A. Pnueli. The temporal logic of programs. In *Proc. of SFCS'77: the 18th Annual Symposium on Foundations of Computer Science*, pages 46–57. IEEE Computer Society Press, 1977.
7. K. Sutner. Computational classification of cellular automata. *Int. J. General Systems*, 41(6):595–607, 2012.
8. W3C OWL Working Group. *OWL 2 Web Ontology Language: Document Overview*, 2009. <http://www.w3.org/TR/owl2-overview/>.
9. S. Wolfram. *A New Kind of Science*. Wolfram Media, 2002.
10. F. Wolter and M. Zakharyashev. Temporalizing description logics. In *Proc. of FroCoS'98: the 2nd International Symposium on Frontiers of Combining Systems*, pages 379–402. Kluwer Academic Publishers, 2000.