# Reasoning about transfinite sequences (extended abtract)*

Stéphane Demri[1] and David Nowak[2]

[1] LSV/CNRS UMR 8643 & INRIA Futurs projet SECSI & ENS Cachan, France
[2] Deparment of Information Science, The University of Tokyo, Japan

**Abstract.** We introduce a family of temporal logics to specify the behavior of systems with Zeno behaviors. We extend linear-time temporal logic LTL to authorize models admitting Zeno sequences of actions and quantitative temporal operators indexed by ordinals replace the standard next-time and until future-time operators. Our aim is to control such systems by designing controllers that safely work on $\omega$-sequences but interact synchronously with the system in order to restrict their behaviors. We show that the satisfiability problem for the logics working on $\omega^k$-sequences is EXPSPACE-complete when the integers are represented in binary, and PSPACE-complete with a unary representation. To do so, we substantially extend standard results about LTL by introducing a new class of succinct ordinal automata that can encode the interaction between the different quantitative temporal operators.

## 1 Introduction

*Control of physical systems.* Modelling interaction between a computer system and a physical system has to overcome the difficulty of the different time scales. For example, reasoning about the connection between the physical description of an electric circuit and its logical description in VHDL (standard language designed and optimized for describing the behavior of digital systems) needs to take into account that the two descriptions are dealing with objects running at distinct speeds. The speeds can be so different that some abstraction consists in assuming one system evolves infinitely quicker than the other one. Another kind of interaction consists of controlling a physical system by a computer system. Usually, a physical system is modelled by differential equations. Solving those equations can then involve computations of limits. For instance, in the bouncing ball example [14], in a finite amount of time an infinite number of actions can be performed. It is a Zeno sequence of actions. However, Zeno behaviors are usually excluded from the modelling of real-time controllers, which is a quite reasonable requirement (see e.g. [7]), but also from the modelling of the physical systems, see some exception in [5]. This is a quite drastic limitation, since Zeno sequences are often acceptable behaviors for physical systems.

* The first author acknowledges partial support by the ACI "Sécurité et Informatique" CORTOS. The second author acknowledges partial support by the e-Society project of MEXT. Part of this work was done while the second author was affiliated to LSV, CNRS & ENS de Cachan.

*Beyond ω-sequences.* Our main motivation in this paper is to model Zeno behaviors and ultimately to control physical systems admitting such behaviors. To do so, we introduce a specification logical language that is interpreted on well-ordered linear orderings. Reasoning problems based on this logical language should admit efficient algorithms, as good as those for standard specification languages as linear-time temporal logic LTL. The $\omega$-sequences are already familiar objects in model-checking, see e.g. [28], even though such infinite objects are never manipulated when model-checking finite-state programs. Indeed, most problems on Büchi automata reduce to standard reachability questions on finite graphs. In a similar fashion, the behaviors of physical system are modeled in the paper by sequences indexed by countable ordinals, i.e. equivalence classes of well-ordered linear orderings, even though as we will show most problems will also reduce to questions on finite graphs. For instance, the law of movement of the bouncing ball is modelled by a set of sequences of length $\omega^2$. The specification of the ball, i.e. the set of acceptable behaviors, is also characterized as a set of sequences of the same length $\omega^2$. On the other hand, the controller is a computer system whose complete executions are $\omega$-sequences. In this paper, we allow Zeno behaviors of physical systems and we will present a specification language working on sequences indexed by ordinals greater than the usual first infinite ordinal $\omega$.

*Our contribution.* We introduce a class of logics LTL($\alpha$) indexed by a countable ordinal $\alpha$ closed under addition whose models are sequences of length $\alpha$. Quantitative extensions of the standard next-time $\mathtt{X}$ and until $\mathtt{U}$ operators are considered by allowing operators of the form $\mathtt{X}^\beta$ and $\mathtt{U}^\beta$ with $\beta$ smaller than $\alpha$. As shown in the paper, for every $\alpha \leq \omega^\omega$, LTL($\alpha$) can be viewed as a fragment of the monadic second-order theory $\langle \omega^\omega, < \rangle$ known to be decidable, see e.g. [10]. For every $k \geq 1$, we show that LTL($\omega^k$) satisfiability is PSPACE-complete with an unary encoding of integers and EXPSPACE-complete with a binary encoding. This generalizes non-trivially what is known about LTL. We reduce the satisfiability problem to the emptiness problem of ordinal automata recognizing transfinite words [9, 13, 29, 19, 8]. The reduction entails that the satisfiability problem has an elementary complexity (by using [11]) but does not guarantee the optimal upper bound. To do so, we introduce a class of succinct ordinal automata of level $k$, $k \geq 1$ in which the LTL($\omega^k$) formulae can be translated into and we prove that the emptiness problem is in NLOGSPACE. Succinctness allows us to reduce by one exponential the size of the automata obtained by translation which provides us the optimal upper bound. Finally, we introduce and motivate a control problem with inputs a physical system $\mathcal{S}$ modelled by an ordinal automaton working on $\omega^k$-sequences, and an LTL($\omega^k$) formula $\phi$ describing the desirable behaviors of the system. The problem we introduce is the existence of a controller $\mathcal{C}$ working on $\omega$-sequences such that the system $\mathcal{S} \times_k \mathcal{C}$ satisfies $\phi$. The synchronization operation $\times_k$ takes into account the different time scales between $\mathcal{S}$ and $\mathcal{C}$. As a by-product of our results, checking whether a controller satisfies the above conditions can be done effectively but we leave the question of the synthesis of such controllers for future work.

*Related work.* Our original motivation in this work is the control of systems with legal Zeno behaviors by systems whose complete executions are $\omega$-sequences. The theory of control of discrete event systems was introduced in [25]. In this theory, a process is a deterministic non-complete finite automaton over an alphabet of events. The control problem consists in, given a process $P$ and a set $S$ of admissible behaviors, finding a process $Q$ such that the behaviors of $P \times Q$ are in $S$ and such that $Q$ reacts to all uncontrollable events and cannot detect unobservable events. Extension to specifications from the modal $\mu$-calculus can be found in [2] whereas the control of timed systems (without Zeno behaviors) is for instance studied in [3, 7]. It is plausible that the techniques from the above-mentioned works (see also [24]) can be adapted to the control problem we have introduced but the technical contribution of this paper is mainly oriented towards satisfiability and model-checking issues.

The logics we have introduced belong to the long tradition of quantitative versions of LTL. LTL-like logics having models non isomorphic to $\omega$ can be found in [1, 27, 26, 20, 22]. Temporal operators in the real-time logics from [1, 20, 22] are indexed by intervals as our logics LTL($\alpha$). However, among the above-mentioned works, only Rohde's thesis [27] contains a LTL-like logic interpreted over $\alpha$-sequences with ordinal $\alpha$ but the temporal operators are simply the standard next-time and until operators without any decoration. It is shown in [27] that the satisfiability problem for such a logic can be decided in exponential-time when the inputs are the formula to be tested and the countable ordinal from which the model is built.

In the paper, we follow the automata-based approach for temporal logics from [28] but we are dealing with ordinal automata recognizing words of length $\alpha$ for some countable ordinal $\alpha$. So, we extend the reduction from LTL into generalized Büchi automata to the reduction from LTL($\omega^k$) into ordinal automata recognizing words of length $\omega^k$. Many classes of ordinal automata have been introduced in the literature. We recall below some of them. In [9, 13] automata recognizing $\omega^k$-sequences for some $k \geq 1$ are introduced making essential the concept of layer. In [10, 29, 19], such automata are generalized to recognize $\alpha$-sequences for $\alpha$ countable. Correspondences between these different classes can be found in [4]. In the paper, we mainly adopt the definitions from [19]. An elegant and powerful extension to automata recognizing words indexed elements from a linear ordering can be found in [8]. As far as we know, automata recognizing sequences of length greater than $\omega$ designed to solve verification problems have been first used in [18] to model concurrency by limiting the state explosion problem. Similarly, timed automata accepting Zeno words are introduced in [5] in order to model physical phenomena with convergent execution. The emptiness problem for such automata is shown to be decidable [5].

As LTL can be viewed as the first-order fragment of monadic second order theory over $\langle \mathbb{N}, < \rangle$, theories over $\langle \alpha, < \rangle$ for some countable ordinal $\alpha$ have been also studied by Büchi [9], see also [10, 4]. For instance, decidability of monadic second order theories over $\langle \alpha, < \rangle$ for some countable ordinal $\alpha$ is shown in [10].

Decidability status of elementary theories over countable ordinals have been established in [6, 12] whereas relationships with other theories are shown in [23].

Because of lack of space, the proofs can be found in [15].

## 2 Temporal Logics on Transfinite Sequences

### 2.1 Ordinals

We recall basic definitions and properties about ordinals. An *ordinal* is a totally ordered set which is *well ordered*, i.e. all its non-empty subset have a least element. Order-isomorphic ordinals are considered equals. An ordinal $\alpha$ is a *successor* ordinal iff there exists an ordinal $\beta$ such that $\alpha = \beta + 1$. An ordinal which is not 0 or a successor ordinal, is a *limit* ordinal. The first limit ordinal is written $\omega$. Addition, multiplication and exponentiation can be defined on ordinals inductively: $\alpha + 0 = \alpha$, $\alpha + (\beta + 1) = (\alpha + \beta) + 1$ and $\alpha + \beta = sup\{\alpha + \gamma : \gamma < \beta\}$ where $\beta$ is a limit ordinal. Multiplication and exponentiation are defined similarly. $\epsilon_0$ is the closure of $\omega \cup \{\omega\}$ under ordinal addition, multiplication and exponentiation. By the Cantor Normal Form theorem, for any ordinal $\alpha < \epsilon_0$, there are unique ordinals $\beta_1, \ldots, \beta_p$, and unique integers $n_1, \ldots, n_p$ such that $\alpha > \beta_1 > \cdots > \beta_p$ and $\alpha = \omega^{\beta_1}.n_1 + \cdots + \omega^{\beta_p}.n_p$. If $\beta < \omega^\omega$, then the $\beta_i$'s are integers. Whenever $\alpha \leq \beta$, there is a unique ordinal $\gamma$ such that $\alpha + \gamma = \beta$. We write $\beta - \alpha$ to denote $\gamma$. For instance, $\omega^2 - \omega = \omega^2$, $\omega \times 3 - \omega = \omega \times 2$ and $\omega^2 - \omega^3$ is not defined since $\omega^3 > \omega^2$.

An ordinal $\alpha$ is said to be closed under addition whenever $\beta, \beta' < \alpha$ implies $\beta + \beta' < \alpha$. For instance, 0, 1, $\omega$, $\omega^2$, $\omega^3$, and $\omega^\omega$ are closed under addition. In the sequel, we shall consider logics whose models are $\alpha$-sequences, i.e. mappings of the form $\alpha \to \Sigma$ for some finite alphabet $\Sigma$ and ordinal $\alpha$ closed under addition.

### 2.2 Quantitative Extensions of LTL

For every ordinal $\alpha$ closed under addition, we introduce the logic LTL($\alpha$) whose models are precisely sequences of the form $\sigma : \alpha \to 2^{\mathrm{AP}}$ for some countably infinite set AP of atomic propositions. The formulae of LTL($\alpha$) are defined as follows: $\phi ::= p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \mathtt{X}^\beta\phi \mid \phi_1 \mathtt{U}^{\beta'}\phi_2$, where $p \in \mathrm{AP}$, $\beta < \alpha$ and $\beta' \leq \alpha$. The satisfaction relation is inductively defined below where $\sigma$ is a model for LTL($\alpha$) and $\beta < \alpha$:

- $\sigma, \beta \models p$ iff $p \in \sigma(\beta)$,
- $\sigma, \beta \models \phi_1 \wedge \phi_2$ iff $\sigma, \beta \models \phi_1$ and $\sigma, \beta \models \phi_2$, $\sigma, \beta \models \neg\phi$ iff not $\sigma, \beta \models \phi$,
- $\sigma, \beta \models \mathtt{X}^{\beta'}\phi$ iff $\sigma, \beta + \beta' \models \phi$,
- $\sigma, \beta \models \phi_1 \mathtt{U}^{\beta'}\phi_2$ iff there is $\gamma < \beta'$ such that $\sigma, \beta + \gamma \models \phi_2$ and for every $\gamma' < \gamma$, $\sigma, \beta + \gamma' \models \phi_1$.

Actually in order to study the decidability/complexity of LTL($\alpha$), we restrict ourselves to countable limit ordinals $\alpha$ so that the set of formulae is itself countable. Furthermore, for studying complexity issues, it is necessary to specify the encoding of the ordinals $\beta \leq \alpha$ occurring in LTL($\alpha$) formulae. In the sequel, we use Cantor normal form to encode ordinals $1 \leq \beta \leq \omega^\omega$, and the natural numbers occurring in such normal forms are represented in binary.

**Proposition 1.** *Satisfiability for* LTL($\omega^\alpha$), $0 \leq \alpha \leq \omega$, *is decidable.*

The model-checking for LTL($\alpha$) takes as inputs an ordinal automaton $\mathcal{A}$ with alphabet AP (see Def. 1) and an LTL($\alpha$) formula $\phi$ and checks whether there is an $\alpha$-sequence $\sigma$ accepted by $\mathcal{A}$ such that $\sigma, 0 \models \phi$.

## 3 Automata-based Approach

In this section, we show how to construct an ordinal automaton $\mathcal{A}_\phi$ such that its set of accepted words is precisely the models of $\phi$, extending the approach for LTL from [28]. In the rest of this section, $\phi \in$ LTL($\omega^k$) for some $k \geq 1$.

### 3.1 Ordinal Automata

We define ordinal automata as a generalization of Muller automata.

**Definition 1 (Ordinal Automaton).** *An ordinal automaton is a tuple* $(Q, \Sigma, \delta, E, I, F)$ *where:*

- $Q$ *is a finite set of states,* $\Sigma$ *is a finite alphabet,*
- $\delta \subseteq Q \times \Sigma \times Q$ *is a one-step transition relation,*
- $E \subseteq 2^Q \times Q$ *is a limit transition relation,*
- $I \subseteq Q$ *[resp.* $F \subseteq Q$*] is a finite set of initial [resp. final] states.*

We write $q \xrightarrow{a} q'$ whenever $\langle q, a, q' \rangle \in \delta$ and $q \rightarrow q'$ iff $q \xrightarrow{a} q'$ for some $a \in \Sigma$. A path of length $\alpha+1$ is a map $r : \alpha+1 \rightarrow Q$ such that for every $\beta \in \alpha$, $r(\beta) \rightarrow r(\beta{+}1)$ and for every limit ordinal $\beta \in \alpha$, there is $P \rightarrow r(\beta) \in E$ s.t. $P = inf(\beta, r)$ with $inf(\beta, r) \stackrel{\text{def}}{=} \{q \in Q :$ for every $\gamma \in \beta,$ there is $\gamma'$ such that $\gamma < \gamma' < \beta$ and $r(\gamma') = q\}$.

A run of length $\alpha+1$ is a path of length $\alpha+1$ such that $r(0) \in I$. If $r(\alpha) \in F$ then $r$ is said to be accepting. The set of sequences recognized by the automaton $\mathcal{A}$, denoted by L($\mathcal{A}$), is the set of $\alpha$-sequences $\sigma : \alpha \rightarrow \Sigma$ for which there is an accepting run $r$ of length $\alpha+1$ verifying for every $\beta \in \alpha$, $r(\beta) \xrightarrow{\sigma(\beta)} r(\beta+1)$.

Ordinal automata from Definition 1 are those defined in [19].

### 3.2 Hintikka Sequences

We define below a notion of closure which generalizes the Fisher-Ladner closure [16].

**Definition 2 (Closure).** *The closure of* $\phi$*, denoted by* $cl(\phi)$*, is the smallest set of* LTL($\omega^k$) *formulae such that*

- $\bot, \phi \in cl(\phi)$, *and* $\neg\psi \in cl(\phi)$ *implies* $\psi \in cl(\phi)$,
- $\psi \in cl(\phi)$ *implies* $\neg\psi \in cl(\phi)$ *(we identify* $\neg\neg\psi$ *with* $\psi$*),*
- $\psi_1 \wedge \psi_2 \in cl(\phi)$ *implies* $\psi_1, \psi_2 \in cl(\phi)$,
- $\mathtt{X}^{\beta}\psi \in cl(\phi)$ *and* $\beta \geq \omega^n$ $(0 \leq n < k)$ *imply* $\mathtt{X}^{\beta-\omega^n}\psi \in cl(\phi)$,
- $\psi_1\mathtt{U}^{\beta}\psi_2 \in cl(\phi)$ *and* $\beta \geq \omega^n$ $(0 \leq n \leq k)$ *imply the formulae below belong to* $cl(\phi)$*:* $\psi_1$, $\psi_2$, $\mathtt{X}^{\omega^n}(\psi_1\mathtt{U}^{\beta-\omega^n}\psi_2)$, $\top\mathtt{U}^{\omega^n}\neg\psi_1$, $\psi_1\mathtt{U}^{\omega^n}\psi_2$.

It is not difficult to show that the notion of closure introduced above generalizes what is done for LTL. From a formula $\phi$, we build an ordinal automata $\mathcal{A}_\phi$ such that $\mathrm{L}(\mathcal{A}_\phi)$ is precisely the set of LTL($\omega^k$) models satisfying $\phi$. Following [28], the states of $\mathcal{A}_\phi$ are subsets of $cl(\phi)$ containing formulae to be satisfied in the future, including the current position. Hence, $cl(\phi)$ is built in such a way that if either $q' \to q$ or $P \to q$ are transitions in $\mathcal{A}_\phi$, then all the formulae to be satisfied in $q$ depending on $q'$ and $P$ are part of $cl(\phi)$.

**Definition 3.** *A set* $X \subseteq cl(\phi)$ *is said to be locally maximally consistent with respect to* $\phi$ *iff it satisfies the conditions below:*

**(mc1)** $\bot \notin X$,
**(mc2)** *for every* $\psi \in cl(\phi)$, $\psi \in X$ *iff* $\neg\psi \notin X$,
**(mc3)** *for every* $\psi_1 \wedge \psi_2 \in cl(\phi)$, $\psi_1 \wedge \psi_2 \in X$ *iff* $\psi_1, \psi_2 \in X$,
**(mc4)** *for every* $\mathtt{X}^0\psi \in cl(\phi)$, $\mathtt{X}^0\psi \in X$ *iff* $\psi \in X$,
**(mc5)** *for every* $\psi_1\mathtt{U}^0\psi_2 \in cl(\phi)$, $\psi_1\mathtt{U}^0\psi_2 \notin X$,
**(mc6)** *for all* $\psi_1\mathtt{U}^{\beta}\psi_2 \in cl(\phi)$ *and* $\beta \geq \omega^n \geq 1$, $\psi_1\mathtt{U}^{\beta}\psi_2 \in X$ *iff either* $\psi_1\mathtt{U}^{\omega^n}\psi_2 \in X$ *or* $\neg(\top\mathtt{U}^{\omega^n}\neg\psi_1), \mathtt{X}^{\omega^n}(\psi_1\mathtt{U}^{\beta-\omega^n}\psi_2) \in X$,
**(mc7)** *for all* $\psi_1\mathtt{U}^{\beta}\psi_2, \psi_1\mathtt{U}^{\beta'}\psi_2 \in cl(\phi)$ *with* $\beta \leq \beta'$, $\psi_1\mathtt{U}^{\beta}\psi_2 \in X$ *implies* $\psi_1\mathtt{U}^{\beta'}\psi_2 \in X$,
**(mc8)** *for every* $\psi_1\mathtt{U}^1\psi_2 \in cl(\phi)$, $\psi_1\mathtt{U}^1\psi_2 \in X$ *iff* $\psi_2 \in X$.

We denote by $maxcons(\phi)$ the set of locally maximally consistent subsets of $cl(\phi)$.

For standard LTL, an Hintikka sequence $\rho$ for a formula $\phi$ is an $\omega$-sequence of sets of subformulae of $\phi$ such that $\phi$ is satisfiable iff $\phi$ has an Hintikka sequence. Local conditions in $\rho$ between two successive elements of the sequence are easy to handle in Büchi automata with the transition relation. The only global condition, stating that if $\psi_1\mathtt{U}\psi_2$ occurs in the sequence, then some future element in the sequence contains $\psi_2$, is handled by the Büchi acceptance condition. Sometimes the non-uniform treatment between local conditions and the global condition is the source of confusion. The Hintikka sequences defined below are based on a similar principle except that we can extend advantageously the notion of locality. The Hintikka sequences $\rho$ are of the form $\rho : \omega^k \to 2^{cl(\phi)}$. Encoding conditions between $\rho(\beta)$ and $\rho(\beta+1)$ can be performed by one-step transitions in ordinal automata. However, the presence of limit transitions allows us also to admit conditions between $\rho(\beta)$ and $\rho(\beta+\omega^{n'})$ with $0 \leq n' < k$. Hence, the global condition in Hintikka sequences of LTL formulae is replaced by a condition between $\rho(\beta)$ and $\rho(\beta+\omega)$. For transfinite sequences, the local and global conditions can be treated uniformly.

**Definition 4 (Hintikka Sequence).** *An Hintikka sequence for* $\phi$ *is a sequence* $\rho : \omega^k \rightarrow 2^{cl(\phi)}$ *such that*

**(hin1)** $\phi \in \rho(0)$,
**(hin2)** *for every* $\beta < \omega^k$, $\rho(\beta) \in maxcons(\phi)$,
**(hin3)** *for all* $\beta < \omega^k$, $\mathtt{X}^{\beta'}\psi \in cl(\phi)$ *and* $0 \leq n' < k$ *such that* $\beta' \geq \omega^{n'}$, $\mathtt{X}^{\beta'}\psi \in \rho(\beta)$ *iff* $\mathtt{X}^{\beta'-\omega^{n'}}\psi \in \rho(\beta + \omega^{n'})$,
**(hin4)** *for all* $\beta < \omega^k$ *and* $\psi_1 \mathtt{U}^{\beta'} \psi_2 \in cl(\phi)$, *(A)* $\psi_1 \mathtt{U}^{\beta'} \psi_2 \in \rho(\beta)$ *iff (B) there is* $\beta \leq \beta'' < \beta + \beta'$ *such that* $\psi_2 \in \rho(\beta'')$ *and for every* $\beta \leq \gamma < \beta''$, $\psi_1 \in \rho(\gamma)$.

**Proposition 2.** $\phi$ *is* $\mathrm{LTL}(\omega^k)$ *satisfiable iff* $\phi$ *has an Hintikka sequence.*

### 3.3 Automaton Construction

We build an ordinal automaton $\mathcal{A}_\phi$ that recognizes only words of length $\omega^k$ over the alphabet $2^{\mathrm{AP}}$ (assuming that AP is the finite set of atomic propositions occurring in $\phi$). The automaton $\mathcal{A}_\phi = \langle Q, \Sigma, \delta, E, I, F \rangle$ is defined as follows:

- $\Sigma = 2^{\mathrm{AP}}$, $Q = maxcons(\phi) \times \{0, \ldots, k\}$,
- $I = \{\langle X, 0 \rangle \in Q : \phi \in X\}$, $F = \{\langle X, n \rangle \in Q : n = k\}$,
- $\langle X, n \rangle \xrightarrow{a} \langle X', n' \rangle \in \delta$ iff (one-step transition)
  **(A1)** $n < k$ and $n' = 0$,
  **(A2)** $X \cap \mathrm{AP} = a$,
  **(A3)** for every $\mathtt{X}^\beta\psi \in cl(\phi)$ such that $\beta \geq 1$, $\mathtt{X}^\beta\psi \in X$ iff $\mathtt{X}^{\beta-1}\psi \in X'$.
- In order to define $E$, we introduce preliminary definitions. For every $\psi_1 \mathtt{U}^\alpha \psi_2 \in cl(\phi)$, we write $P_{\psi_1 \mathtt{U}^\alpha \psi_2}$ to denote the set $\{\langle X, n \rangle :$ either $\psi_2 \in X$ or $\neg(\psi_1 \mathtt{U}^\alpha \psi_2) \in X\}$. For every $\langle X, n \rangle \in Q$ we write $Q_{\langle X, n \rangle}$ to denote the subset of $Q$ such that for every $\langle X', n' \rangle \in Q$, $\langle X', n' \rangle \in Q_{\langle X, n \rangle} \stackrel{\text{def}}{\Leftrightarrow}$
  **(A4)** $n' < n$,
  **(A5)** for every $\mathtt{X}^\alpha\psi \in cl(\phi)$ with $\alpha \geq \omega^n$, $\mathtt{X}^\alpha\psi \in X'$ iff $\mathtt{X}^{\alpha-\omega^n}\psi \in X$.
  For every $\langle X, n \rangle \in Q$, $Z \rightarrow \langle X, n \rangle \in E$ iff
  **(A6)** $n \geq 1$,
  **(A7)** $Z \subseteq Q_{\langle X, n \rangle}$,
  **(A8)** $Z$ contains a state of the form $\langle Y, n-1 \rangle$,
  **(A9)** for all $\psi_1 \mathtt{U}^\beta \psi_2 \in cl(\phi)$ and $\beta \geq \omega^n$ such that $\neg(\psi_1 \mathtt{U}^{\beta-\omega^n} \psi_2) \in X$, $P_{\psi_1 \mathtt{U}^\beta \psi_2} \cap Z \neq \emptyset$.

Observe the similarities between (A3) and (A5) and between (A9) and (mc6). For $\mathrm{LTL}(\omega)$, the above construction roughly corresponds to the Muller automaton obtained from the generalized Büchi automaton for the LTL formula $\phi$.

The automaton $\mathcal{A}_\phi$ has $2^{2^{\mathcal{O}(|\phi|)}}$ states and $2^{2^{2^{\mathcal{O}(|\phi|)}}}$ transitions. By [11, Proposition 6], the emptiness problem for ordinal automata is in P. So checking whether $\mathcal{A}_\phi$ accepts at least one word can be done in triple exponential time, which provides an elementary bound but not optimal as shown in the sequel.

**Proposition 3.** $\mathrm{L}(\mathcal{A}_\phi) = \mathrm{Mod}(\phi)$.

We invite the reader to consult the tedious proof of Proposition 3 in [15] to understand the relationships between the conditions (mc$\star$), (hin$\star$) and (A$\star$).

## 4 Computational Complexity

In this section, we show complexity results about satisfiability of LTL($\omega^k$) with $1 \leq k < \omega$.

**Theorem 1.** *For every ordinal $\alpha \geq 1$, satisfiability for* LTL($\omega^\alpha$) *is* EXPSPACE-*hard.*

### 4.1 Succinct Ordinal Automata of Level $k$

In order to refine the complexity result from Sect. 3, we define below specialized ordinal automata that recognize $\omega^k$-sequences. Similar automata can be found in the literature, see e.g. [13, 19, 4].

**Definition 5 (Ordinal Automaton of Level $k$).** *An ordinal automaton $\mathcal{A} = \langle Q, \Sigma, \delta, E, I, F \rangle$ is said to be of level $k \geq 1$ iff there is a map $l : Q \to \{0, \ldots, k\}$ such that*

- *for every $q \in F$, $l(q) = k$;*
- *$q \xrightarrow{a} q' \in \delta$ implies $l(q') = 0$ and $l(q) < k$;*
- *$P \to q \in E$ implies*
  1. *$l(q) \geq 1$,*
  2. *for every $q' \in P$, $l(q') < l(q)$,*
  3. *there is $q' \in P$ such that $l(q') = l(q) - 1$.*

The automaton built in Section 3 is of level $k$ when the input formula is in LTL($\omega^k$). However, $\mathcal{A}_\phi$ is of triple [resp. double] exponential size in $|\phi|$ when integer are encoded in binary [resp. unary] which is still too much to characterize accurately the complexity of LTL($\omega^k$) satisfiability. That is why we introduce below a special class of ordinal automata which can represent succinctly an exponential amount of limit transitions as the generalized Büchi automata can be viewed as a succinct representation of Muller automata. Hence, we shall construct $\mathcal{A}'_\phi$ such that L($\mathcal{A}'_\phi$) = L($\mathcal{A}_\phi$), and $\mathcal{A}'_\phi$ is "only" of double [resp. simple] exponential size in $|\phi|$ when integers are encoded in binary [resp. unary].

**Definition 6 ($p(\cdot)$-Succinct Ordinal Automaton of Level $k$).** *Given a polynom $p(\cdot)$, a $p(\cdot)$-succinct ordinal automaton of level $k$ is a structure $\mathcal{A} = \langle Q, \Sigma, \delta, E, I, F, l \rangle$ defined as an ordinal automata of level $k$ except that $E$ is a set of tuples of the form $\langle P_0, P_1, \ldots, P_n, q \rangle$ with $n \geq 0$, $q \in Q$ and $P_0, \ldots, P_n \subseteq Q$ such that*

- *$\langle P_0, P_1, \ldots, P_n, q \rangle \in E$ implies*
  1. *$1 \leq l(q) \leq k$,*
  2. *each state in $P_0$ is of level $l(q) - 1$,*
  3. *each state in $P_1 \cup \cdots \cup P_n$ is of level less than $l(q) - 1$,*
  4. *$n \leq p(|Q|)$,*
- *for every state $q$ of level strictly more than 0, there is at most one tuple in $E$ of the form $\langle P_0, P_1, \ldots, P_n, q \rangle$.*

Each tuple $\langle P_0, P_1, \ldots, P_n, q\rangle$ encodes succinctly the set of limit transitions

$$trans(\langle P_0, P_1, \ldots, P_n, q\rangle) \stackrel{\text{def}}{=}$$

$$\{P \to q : \ P \subseteq Q, \ \forall \ i \ P_i \cap P \neq \emptyset \text{ and } \forall q' \in P, \ l(q') < l(q)\}.$$

In the sequel, given a $p(\cdot)$-succinct ordinal automaton $\mathcal{A}$ of level $k$, we write $\mathcal{A}^o = \langle Q, \Sigma, \delta, E', I, F, l\rangle$ to denote the ordinal automaton of level $k$ with $E' = \bigcup_{t \in E} trans(t)$. The language recognized by $\mathcal{A}$ is defined as the language recognized by $\mathcal{A}^o$. In that way, a $p(\cdot)$-succinct ordinal automaton of level $k$ is simply a succinct encoding of some ordinal automaton of level $k$. An important property of such automata rests on the fact that the size of $E$ is in $\mathcal{O}(|Q|^2 \times p(|Q|))$. By contrast, in an ordinary ordinal automaton of level $k$, the cardinality of the set of limit transitions can be in the worst case exponential in $|Q|$.

The automaton $\mathcal{A}_\phi$ from Sect. 3.3 can be viewed as a $p_0(\cdot)$-succinct ordinal automaton of level $k$ with $p_0(x) = x$.

Lemma 1 below is the key property to obtain the NLOGSPACE upper bound for the emptiness problem of ordinal automata of level $k$, even in the succinct version. It generalizes substantially the property that entails that the graph accessibility problem and the emptiness problem for generalized Büchi automata can be solved in non-deterministic logarithmic space.

**Lemma 1.** *Let $\mathcal{A}$ be an automaton of level $k$ and $r$ be a run of length $\omega^{k'} + 1$ for some $1 \leq k' \leq k$. Then, there is a path $r'$ of length $\omega^{k'} + 1$ such that*

- $r'(0) = r(0)$ *and* $r'(\omega^{k'}) = r(\omega^{k'})$,
- *there are* $K \leq |Q|$ *and* $K' \leq |Q|^2$ *such that for every* $\alpha \geq \omega^{k'-1} \times K$ *such that the normal form of* $\alpha$ *is* $\omega^{k'-1} \times n + \beta$, $r'(\alpha) = r'(\omega^{k'-1} \times (n + K') + \beta)$.

### 4.2 An Optimal Algorithm to Test Emptiness

In order to test emptiness of the language recognized by an automaton of level $k$, we introduce a function $acc(q, q')$ (see Fig. 1) that returns $\top$ iff there is a path $r$ of length $\omega^{l(q')}$ such that $r(0) = q$ and $r(\omega^{l(q')}) = q'$. We design the following non-deterministic algorithm:

```
Empty?(A)
Guess q_0 ∈ I and q_f ∈ F;
InLoop := false;
acc(q_0, q_f).
```

Nondeterminism is also highly present in the definition of $acc(q_0, q_f)$. A few global variables are used. The variable InLoop is a Boolean equals to `true` iff $q'$ in a call $acc(q, q')$ belongs in the periodic part of the run. Moreover, for every $i \in \{1, \ldots, k\}$, the variable $\uparrow_i$ contains the address of the occurrence of a state in the leftmost part of a rule $P \to q''$ with $l(q'') = i$: $\mathcal{O}(k \times log|\mathcal{A}|)$ bits are needed in total. Remember that $\mathcal{A}$ is encoded as a string and the address of the occurrence of a state is simply a position in that string, which requires only

$acc(q, q')$ $(l(q') \leq k,\ l(q) = 0)$

$k' := l(q') - 1$;
If $k' \geq 0$ then
  Guess a rule $P \rightarrow q'$;
  $\uparrow_{k'+1}$ takes the value of the address of the first state in $P$;
  Guess $K \leq |Q|$ and $K' \leq |Q|^2$;
  Guess $\mathrm{q}_{k'}^{\mathrm{repeat}} \in P$ such that $l(\mathrm{q}_{k'}^{\mathrm{repeat}}) = k'$ (repeating state);
  $\mathrm{q}_0 := q$;
  For $i = 1$ to $K$ do
    Guess $\mathrm{q}_{k'} \in P$ of level $k'$;
    If $acc(\mathrm{q}_0, \mathrm{q}_{k'})$ then guess $\mathrm{q}_0$ such that $l(\mathrm{q}_0) = 0$ and $\mathrm{q}_{k'} \rightarrow \mathrm{q}_0$;
  If $\mathrm{q}_{k'} \neq \mathrm{q}_{k'}^{\mathrm{repeat}}$ then **abort**;
  If $k' + 1 = k$ then InLoop = `true`;
  Guess $\mathrm{q}_{k'} \in P$ of level $k'$;
  If InLoop = `true` then (Check&Update($\mathrm{q}_0$);Check&Update($\mathrm{q}_{k'}$));
  For $i = 1$ to $K'$ do
    If $acc(\mathrm{q}_0, \mathrm{q}_{k'})$ then
      Guess $\mathrm{q}_0$ such that $l(\mathrm{q}_0) = 0$ and $\mathrm{q}_{k'} \rightarrow \mathrm{q}_0$;
      $\mathrm{q}_{k'}^{\mathrm{aux}} := \mathrm{q}_{k'}$;
      Guess $\mathrm{q}_{k'} \in P$ of level $k'$;
      If $i \neq K'$ then (Check&Update($\mathrm{q}_0$);Check&Update($\mathrm{q}_{k'}$));
    otherwise **abort**;
  If one of the conditions below fails then **abort** otherwise **accept**
   1. $\uparrow_{k'+1} \neq nil$ (some state in $P$ has not been visited infinitely often),
   2. $\mathrm{q}_{k'}^{\mathrm{aux}} \neq \mathrm{q}_{k'}^{\mathrm{repeat}}$ (wrong choice of the repeating state of level $k'$)
otherwise if $q \rightarrow q'$ then **accept** otherwise **abort**.

**Fig. 1.** Accessibility function

$\mathcal{O}(log|\mathcal{A}|)$ bits. The variable $\uparrow_i$ is updated when the state whose address is $\uparrow_i$ is detected in the periodic part of the run.

In the definition of $acc(q, q')$, in order to test whether there is a path $r$ of length $\omega^{l(q')}$ such that $l(q') \geq 1$, $r(0) = q$ and $r(\omega^{l(q')}) = q'$, Lemma 1 guarantees that the periodic part of $r$ is of length at most $\omega^{l(q')-1} \times |Q|^2$ and the prefix is of length at most $\omega^{l(q')-1} \times |Q|$. This explains the two main loops of $acc(q, q')$. When a state $t$ is guessed in the periodic part of the run, one has to check that $t$ indeed belongs to rules of the form $P \rightarrow q''$ with $l(q'') > l(q_t)$ and one updates the variables $\uparrow_i$ since $t$ has been detected (see Fig. 2).

**Theorem 2.** *For every $k \geq 0$, the emptiness problem for ordinal automata of level $k$ is* NLOGSPACE*-complete.*

**Corollary 1.** *The emptiness problem for Muller automata is* NLOGSPACE*-complete.*

The discipline on memory space done in the algorithm in Fig. 1 can be adapted to succinct ordinal automata.

Check&Update($q$)
For $1 \leq i \leq k$ do
If $\uparrow_i$ contains the address of an occurrence of $q$ in the leftmost part of a rule then $\uparrow_i$ takes the value of the next state in the rule (possibly the rightmost state in the rule);
If $l(q) \leq i-1$ and $q$ does not occur in the leftmost part of the rule that is currently pointed by $\uparrow_i$ then **abort**. (one needs another variable to visit the states in the leftmost part of that rule)
**accept**.

**Fig. 2.** Update of the variables $\uparrow_i$s

**Corollary 2.** *For all $k \geq 0$ and polynom $p(\cdot)$, the emptiness problem for $p(\cdot)$-succinct ordinal automata of level $k$ is* NLOGSPACE*-complete.*

### 4.3 Optimal Complexity Upper Bounds

**Theorem 3.** *For every $k \geq 1$, the satisfiability problem for* LTL($\omega^k$) *is* PSPACE*-complete when the integers are encoded in unary and the problem is in* EXPSPACE*-complete when the integers are encoded in binary.*

**Corollary 3.** *For every $k \geq 1$, the model-checking problem for* LTL($\omega^k$) *is decidable.*

Since the complexity of the emptiness problem for ordinal automata is not completely characterized (we know it is in P by [11] but P-hardness is open), our decidability proof does not provide a full characterization of the complexity of the model-checking problem for LTL($\omega^k$). However, with space ressources, it is at most two exponential higher than the satisfiability problem.

Since the languages recognized by $x$-succinct ordinal automata of level $k$ can be shown to be closed under intersection, we have the following result.

**Theorem 4.** *For every $k \geq 1$, the model-checking problem for* LTL($\omega^k$) *restricted to $x$-succinct ordinal automata of level $k$ is* PSPACE*-complete when the integers are encoded in unary and the problem is* EXPSPACE*-complete when the integers are encoded in binary.*

## 5 Application: Control of Physical Systems

In this section, we formalize the control problem of a physical system by a computer system by using ordinal automata and the logics LTL($\omega^k$). Even though it is the original motivation of our investigations on the logics LTL($\alpha$), at this point of the paper we have all the necessary definitions and results to state concisely the problem. We model a system by an ordinal automaton recognizing $\omega^k$-sequences. For instance, the law of movement of the bouncing ball corresponds to $\omega^2$-sequences and the set of acceptable behaviors of the ball is modelled by a set of sequences of the same length $\omega^2$. On the other hand, the controller is an operational model working on $\omega$-sequences.

Before stating the control problem, we need to give definitions about the synchronous product between ordinal automata and about the way to transform an ordinal automaton of level 1 into an ordinal automaton of level $k \geq 2$ that has relevant actions only on states in positions of the form $\omega^{k-1} \times n$ (*lifting*). As usual, LTL($\omega^k$) formulae can be viewed equivalently as ordinal automata of level $k$ and we shall use these different representations depending on the context (see [2] for a similar standard treatment between formulae and automata).

*Synchronous product.* We define below the synchronous product of two ordinal automata such that if they have the same alphabet then the language recognized by the product is the intersection language. Otherwise, a letter that is present in a single automaton can only affect the state component in the product related to this automaton. This is useful to deal with unobservable actions (see below). Given two ordinal automata $\mathcal{A}_i = \langle Q_i, \Sigma_i, \delta_i, E_i, I_i, F_i \rangle$, for $i = 1, 2$, their synchronous product is defined as $\mathcal{A}_1 \times \mathcal{A}_2 = \langle Q, \Sigma, \delta, E, I, F \rangle$ where:

- $Q = Q_1 \times Q_2$, $\Sigma = \Sigma_1 \cup \Sigma_2$.
- $\langle q_1, q_2 \rangle \xrightarrow{a} \langle q_1', q_2' \rangle \in \delta$ iff either:
  - $a \in \Sigma_1 \cap \Sigma_2$, $q_1 \xrightarrow{a} q_1' \in \delta_1$, and $q_2 \xrightarrow{a} q_2' \in \delta_2$; or
  - $a \in \Sigma_1 \backslash \Sigma_2$, $q_1 \xrightarrow{a} q_1' \in \delta_1$, and $q_2 = q_2'$; or
  - $a \in \Sigma_2 \backslash \Sigma_1$, $q_2 \xrightarrow{a} q_2' \in \delta_2$, and $q_1 = q_1'$.
- $P \to \langle q_1, q_2 \rangle \in E$ iff there exist $P_1 \to q_1 \in E_1$ and $P_2 \to q_2 \in E_2$ such that $\{q : \langle q, q' \rangle \in P\} = P_1$ and $\{q' : \langle q, q' \rangle \in P\} = P_2$.
- $I = I_1 \times I_2$, $F = F_1 \times F_2$.

We write $w/\Sigma$ for the subword of $w$ consisting only of the letters from $\Sigma$.

**Proposition 4.** $w \in \mathrm{L}(\mathcal{A}_1 \times \mathcal{A}_2) \;\Leftrightarrow\; w/\Sigma_1 \in \mathrm{L}(\mathcal{A}_1)$ *and* $w/\Sigma_2 \in \mathrm{L}(\mathcal{A}_2)$.

*Lifting.* In order to synchronize the system with a controller working on $\omega$-sequences, we need to transform the controller so that its product with $\mathcal{S}$ only constraints states on positions $\omega^{k-1} \times n$, $n \in \mathbb{N}$. The other positions are not constrained.

Let $\mathcal{A} = \langle Q, \Sigma, \delta, E, I, F, l \rangle$ be an automaton of level 1. We define its lifting $lift_k(\mathcal{A})$ at level $k \geq 2$ to be the automaton $\langle Q', \Sigma, \delta', E', I', F, l' \rangle$ by:

- $Q' = (\{0, \ldots, k-1\} \times (Q \setminus F)) \cup F$, $I' = \{k-1\} \times I$,
- $l'(q) = k$ for $q \in F$ and $l'(\langle i, q' \rangle) = i$,
- $\delta' = \{\langle k-1, q \rangle \xrightarrow{a} \langle 0, q' \rangle \;:\; q \xrightarrow{a} q' \in \delta\} \cup$ ,
  $\{\langle i, q \rangle \xrightarrow{a} \langle 0, q \rangle \;:\; 0 \leq i < k,\; a \in \Sigma,\; q \notin F\}$

– $E' = \{\{\langle 0, q\rangle, \ldots, \langle i{-}1, q\rangle\} \rightarrow \langle i, q\rangle \ : \ 1 \leq i < k, \ q \in Q\} \cup \{\{\langle 0, q_1\rangle, \ldots, \langle k-1, q_1\rangle, \ldots, \langle 0, q_n\rangle, \ldots, \langle k-1, q_n\rangle\} \rightarrow q \ \mid \ \{q_1, \ldots q_n\} \rightarrow q \in E\}$.

**Proposition 5.** *For all* $w \in \Sigma^{\omega^k}$, $w \in \mathrm{L}(lift_k(\mathcal{A}))$ *iff the word* $w' \in \Sigma^{\omega}$*, defined by* $w'(i) = w(\omega^{k-1} \times i)$*, is in* $\mathrm{L}(\mathcal{A})$.

*The control problem.* A physical system $\mathcal{S}$ is modelled as a structure

$$\langle \mathcal{A}, Act_C, Act_O, Act\rangle$$

where $\mathcal{A}$ is an ordinal automaton of level $k$ with alphabet $2^{Act}$ where $Act$ is a finite set of actions, $Act_O \subseteq Act$ is the set of observable actions and $Act_C \subseteq Act_O$ is the set of controllable actions. The set of uncontrollable actions is denoted by $Act_{NC}$. A specification of the system $\mathcal{S}$ is naturally an LTL($\omega^k$) formula $\psi$. A controller $\mathcal{C}$ for the pair $\langle \mathcal{S}, \psi\rangle$ is a system whose complete executions are $\omega$-sequences (typically ordinal automata of level 1) verifying the properties below.

– Only observable actions are present in the controller. Hence, thanks to the synchronization mode, in the product system between $\mathcal{S}$ and $\mathcal{C}$, unobservable actions do not change the $\mathcal{C}$-component of the current state. So the alphabet of $\mathcal{C}$ is $2^{Act_O}$.
– From any state of $\mathcal{C}$, uncontrollable actions can always be executed: $\forall q \cdot \forall a \subseteq Act_O \setminus Act_C$, there is a transition $q \xrightarrow{b} q'$ in $\mathcal{C}$ such that $b \cap Act_{NC} = a$.
– Finally, the system $\mathcal{S}$ controlled by $\mathcal{C}$ satisfies $\psi$. Because $\mathcal{S}$ and $\mathcal{C}$ work on sequences of different length, the controlled system is in fact equal to $lift_k(\mathcal{C}) \ \times \ \mathcal{S}$. So $lift_k(\mathcal{C}) \ \times \ \mathcal{S} \models \psi$ should hold. This is equivalent to the emptiness of the language of the product automaton $lift_k(\mathcal{C}) \ \times \ \mathcal{S} \ \times \ \mathcal{A}_{\neg\psi}$.

As a consequence of Corollary 3 we obtain the following result.

**Proposition 6.** *The problem of checking whether* $lift_k(\mathcal{C}) \ \times \ \mathcal{S} \ \times \ \mathcal{A}_{\neg\psi}$ *given a physical system* $\mathcal{S}$*, a controller* $\mathcal{C}$ *and a specification* $\psi$ *is decidable.*

We explained how to check that a controller is correct with respect to a specification, but we do not address here the controller synthesis issue. Moreover, by assuming that $\mathcal{S}$ and $\mathcal{C}$ are succinct ordinal automata, we can improve considerably the complexity of the above problem (see e.g., Theorem 4).

*Example.* Consider the system is a bouncing ball [14] with three actions *lift-up*, *bounce* and *stop*, where only *lift-up* is controllable, and only *stop* and *lift-up* are observable. The law of the ball is described by the following LTL($\omega^2$) formula:

$$\phi = \mathtt{G}^{\omega^2}(\textit{lift-up} \ \Rightarrow \ \mathtt{X}^1(\mathtt{G}^{\omega}\,\textit{bounce} \ \wedge \ \mathtt{X}^{\omega}\,\textit{stop}))$$

$\mathtt{G}^{\alpha}\varphi$ is an abbreviation for $\neg(\top \mathtt{U}^{\alpha} \neg\varphi)$. Informally, $\phi$ states that when the ball is lifted-up, then it bounces an infinite number of times in a finite time and then stops. An equivalent ordinal automaton $\mathcal{A}_{\phi}$ working on $\omega^2$-sequences can be easily defined. The specification is given by the LTL($\omega^2$) formula: $\psi = \mathtt{G}^{\omega^2}\mathtt{X}^1\,\textit{bounce}$.

Informally, $\psi$ states that the ball should almost always be bouncing. A possible controller for this system is described by the following LTL formula:

$$\varphi = \mathit{lift\text{-}up} \wedge \mathtt{G}^{\omega}(\mathit{stop} \Rightarrow \mathit{lift\text{-}up})$$

Informally, $\varphi$ states that the controller should lift-up the ball at the beginning and then lift-up it again each time it stops. Similarly, an equivalent ordinal automaton $\mathcal{A}_{\varphi}$ working on $\omega$-sequences can be easily defined.

## 6 Concluding Remarks

We have introduced a family of temporal logics to specify the behavior of systems by assuming that the sequence of actions is isomorphic to some well-ordered linear ordering (see the bouncing ball example in Sect. 5). Our aim is to control such physical systems by designing controllers that safely work on $\omega$-sequences but interact synchronously with the physical system in order to restrict their behaviors. We have extended linear-time temporal logic LTL to $\alpha$-sequences for any countable ordinal $\alpha$ closed under addition, by considering quantitative operators indexed by ordinals smaller than $\alpha$. This is a new class of linear-time temporal logics for which we have shown that LTL($\omega^{\omega}$) is decidable by reduction to the monadic second-order theory $\langle \omega^{\omega}, < \rangle$ and for every $k \geq 1$, LTL($\omega^{k}$) satisfiability problem is PSPACE-complete [resp. EXPSPACE-complete] when the integers are encoded in unary [resp. in binary] generalizing what is known about LTL. Our proof technique is inspired from [28] with significant extensions in order to deal with the interaction between arithmetics on ordinals and temporal operators. Moreover, we have introduced a new class of succinct ordinal automata in order to fully characterize the complexity of the logics. The treatment of these aspects leads to the most difficult technical parts of the paper.

A lot of work remains to be done even though our logics have been shown to admit reasoning tasks of complexity similar to that of LTL. Synthesis of controllers working on $\omega$-sequences on the line of Sect. 5 is on the top of our priority list. Moreover, LTL is known to be initially equivalent to the first-order theory of $\langle \omega, < \rangle$ by Kamp's theorem [21] and by the separation theorem [17]. Is LTL($\omega^{k}$) also initially equivalent to the first-order theory of $\langle \omega^{k}, < \rangle$?

## References


1. R. Alur, T. Feder, and T. Henzinger. The benefits of relaxing punctuality. *Journal of the ACM*, 43:116–146, 1996.
2. A. Arnold, A. Vincent, and I. Walukiewicz. Games for synthesis of controllers with partial observation. *TCS*, 303(1):7–34, 2003.
3. E. Asarin, O. Maler, and A. Pnueli. Symbolic controller synthesis for discrete and timed systems. In *Hybrid systems II*, volume 999 of *LNCS*, pages 1–20. Springer, 1995.
4. N. Bedon. *Langages reconnaissables de mots indexés par des ordinaux.* PhD thesis, Université Marne-la-Vallée, 1998.

5. B. Bérard and C. Picaronny. Accepting Zeno words: a way toward timed refinements. In *MFCS'97*, volume 1295 of *LNCS*, pages 149–158. Springer, 1997.
6. A. Bès. Decidability and definability results related to the elementary theory of ordinal multiplication. *Fundamenta Mathematicae*, 171:197–211, 2002.
7. P. Bouyer, D. D'Souza, P. Madhusudan, and A. Petit. Timed control with partial observability. In *CAV'03*, volume 2725 of *LNCS*, pages 180–192. Springer, 2003.
8. V. Bruyère and O. Carton. Automata on linear orderings. In *MFCS 2001*, volume 2136 of *LNCS*, pages 236–247. Springer-Verlag, 2001.
9. J. Büchi. Transfinite automata recursions and weak second order theory of ordinals. In *Int. Cong. Logic, Methodology and Philosophy of Science, Jerusalem*, pages 3–23, 1964.
10. J. Büchi and D. Siefkes. *The monadic second order theory of all countable ordinals*, volume 328 of *Lecture Notes in Mathematics*. Springer, 1973.
11. O. Carton. Accessibility in automata on scattered linear orderings. In *MFCS 2002*, volume 2420 of *LNCS*, pages 155–164. Springer, 2002.
12. C. Choffrut. Elementary theory of ordinals with addition and left translation by $\omega$. In *DLT'01*, volume 2295 of *LNCS*, pages 15–20. Springer, 2002.
13. Y. Choueka. Finite automata, definable sets, and regular expressions over $\omega^n$-tapes. *JCSS*, 17:81–97, 1978.
14. P. Cuijpers, M. Reniers, and A. Engels. Beyond Zeno-behaviour. Technical report, TU of Eindhoven, 2001.
15. S. Demri and D. Nowak. Reasoning about transfinite sequences. arXiv:cs.LO/0505073, May 2005.
16. M. Fischer and R. Ladner. Propositional dynamic logic of regular programs. *JCSS*, 18:194–211, 1979.
17. D. Gabbay, A. Pnueli, S. Shelah, and J. Stavi. On the temporal analysis of fairness. In *POPL'80*. ACM Press, 1980.
18. P. Godefroid and P. Wolper. A partial approach to model checking. *I&C*, 110(2):305–326, 1994.
19. J. Hemmer and P. Wolper. Ordinal finite automata and languages (extended abstract). Technical report, Université de Liège, 1991.
20. Y. Hirshfeld and A. Rabinovich. Logics for real time: decidability and complexity. *Fundamenta Informaticae*, 62:1–28, 2004.
21. J. Kamp. *Tense Logic and the theory of linear order*. PhD thesis, UCLA, USA, 1968.
22. C. Lutz, D. Walther, and F. Wolter. Quantitative temporal logics: PSPACE and below. In *TIME'05*, 2005. To appear.
23. F. Maurin. The theory of integer multiplication with order restricted to primes is decidable. *The Journal of Symbolic Logic*, 62(1):123–130, 1997.
24. A. Pnueli and R. Rosner. On the synthesis of a reactive module. In *16th ACM POPL, Austin, Texas*, pages 179–190, 1989.
25. P. J. G. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77:81–98, 1989.
26. M. Reynolds. The complexity of the temporal logic with until over general linear time. *JCSS*, 66(2):393–426, 2003.
27. S. Rohde. *Alternating Automata and The Temporal Logic of Ordinals*. PhD thesis, University of Illinois, 1997.
28. M. Vardi and P. Wolper. Reasoning about infinite computations. *I&C*, 115:1–37, 1994.
29. J. Wojciechowski. Classes of transfinite sequences accepted by nondeterministic finite automata. *Annales Societatid Mathematicae Polonae*, pages 191–223, 1984.