Logical Time Models to Study Cyber-Physical Systems

Hassan Khalil El Zein and Grygoriy Zholtkevych

School of Mathematics and Computer Science
V.N. Karazin Kharkiv National University
4, Svobody Sqr., Kharkiv, 61022, Ukraine
dr.hassanelzein@icloud.com; g.zholtkevych@karazin.ua

Abstract. The paper is devoted to problems caused by the nonlinearity of logical time in distributed, especially cyber-physical, systems. Two approaches to the modelling of such systems are considered in the paper. The operational approach is based on the traditional model that defines the admissible system behaviour as a set of acceptable schedules of the system. The paper argues in favour of restricting possible sets of schedules by that sets of schedules that satisfy certain safety properties. The denotational approach is stated in the language of category theory. This abstraction level clarifies concepts used in the models. In particular, it is explained the feature of linear models as terminal objects with respect to some natural class of morphisms. Further, the interrelation between these two approaches is represented as a formal relation and discuss some properties of the relation that need to be studied.

Keywords: cyber-physical system, logical time, clock, denotational semantic model, operational semantic model, schedule, safety property, clock structure, clock morphism

1 Introduction

The National Science Foundation of USA defines cyber-physical systems (CPS for short) as "engineered systems that are built from and depend upon the synergy of computational and physical components"[12, Synopsis of Program] and clarifies ibidem that "emerging CPS will be coordinated, distributed, and connected, and must be robust and responsive". The perspectives of the CPS-technology is estimated by this document as follows: "The CPS of tomorrow will far exceed the simple embedded systems of today in capability, adaptability, resiliency, safety, security, and usability. CPS technology will transform the way people interact with engineered systems, just as the Internet transformed the way people interact with information. New smart cyber-physical systems will drive innovation and competition in sectors such as the power grid, transportation, buildings, medicine, and manufacturing"[12, Synopsis of Program]. The last revision of the mentioned document [13] states the following: "CPS are engineered systems that are built from, and depend upon, the seamless integration of

computational algorithms and physical components. Advances in CPS will enable capability, adaptability, scalability, resiliency, safety, security, and usability that will far exceed the simple embedded systems of today. CPS technology will transform the way people interact with engineered systems – just as the Internet has transformed the way people interact with information. New smart CPS will drive innovation and competition in sectors such as agriculture, energy, transportation, building design and automation, healthcare, and manufacturing just as the Internet has transformed the way people interact with information. Indeed, it is also clear that CPS technologies are central to achieving the vision of Smart & Connected Communities (S&CC), including "Smart Cities", which spans these multiple sectors and includes the important attributes of efficiency, safety, and security" [13, Synopsis of Program]. Thus, comparing this texts we may say that the notion of CSP is a well-established concept and refer to a combined system of executive subsystems and network of controlling units (cyber components), which guarantee the wholeness of the system.

It should be emphasised that the development trend of modern technology is the integration of CPS-components through technology Internet of Things (IoT). Analysing trends of IoT- and CPS-technology Kate Carruthers notes [3]: "CPS include traditional embedded and control systems, and these will be transformed by new approaches from IoT. However, the challenge for IoT and CPS remains security and risk management. As less rigorously controlled systems are linked then risk becomes distributed and the provenance of software components becomes difficult to trace. This gives rise to questions around risk management and liability for breaches or damages".

Thus, we may state that any modern CPS should be considered as a safety-critical system. The necessity to use trustworthy strategies for development of systems of such a type is the first significant conclusion for the practice of system design.

The above reasons motivate our research, which is aimed to clarification of objective limits to the applicability of the clock model for the specification and computer-aided analysis of behavioural constraints for cyber components of CPS. The principal tool of our study is the clock model proposed by Leslie Lamport [8](see also [2, Chap. 2] and [6, Chap. 3]) for studying distributed computing. A survey of examples of applying this model to study CPS can be found in [11].

2 General Structure of a Cyber-Physical System

Remind that in the paper we use the term cyber-physical system to refer to a heterogeneous complex of natural objects and artificial subsystems. This complex

is managed by the system of interacting controllers (cyber components), which provides its operation as a whole entity.

This informal description can be refined by the following class diagram (Fig. 1) representing the abstract framework for CPS.

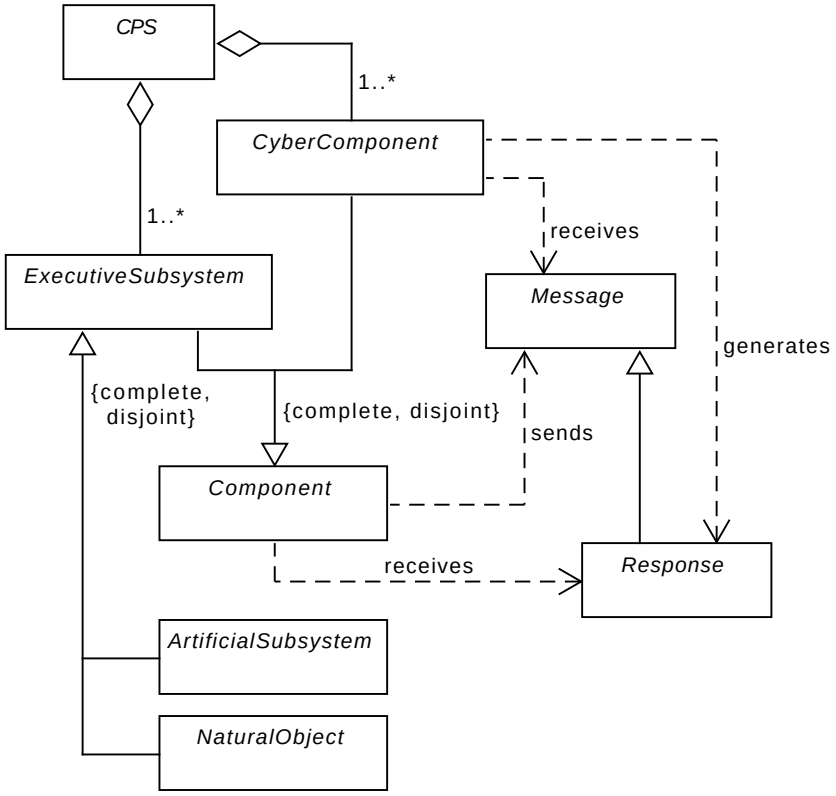

Fig. 1. CPS: Conceptual Framework

This framework establishes that

- any *CPS* consists of at least one instance of class *ExecutiveSubsystem* and at least one instance of class *CyberComponent*;
- each of these instances is an instance of class *Component*;
- an instance of class *ExecutiveSubsystem* is either an instance of class *NaturalObject* or of class *ArtificialSubsystem*.

This model fixes that the message interchange between components of CPS is the only way to provide interaction of these components:

- each instance of class *Component* sends an instance of class *Message*, which are received by some instances of class *CyberComponent*;
- each instance of class *CyberComponent* generates a special message, which is an instance of class *Response*, of course, a response depends on messages received by the cyber component that has generated this response;
- if the component receiving responses is an instance of class *ExecutiveSubsystem* then it executes the actions corresponding to the received response collection, otherwise, the behaviour is similar to the previous case.

The described architecture model establishes that the behaviour of each cyber component is determined only by a message stream, received by the component. In this case, the use of trace semantics to distinguish between the correct and incorrect behaviour of a cyber component is a reasonable solution.

The appropriate approach to the correctness of the logical time dependencies was apparently first used by Leslie Lamport [8].

The clock model is developed to specify and analyse the logical temporal relationships between the event occurrences (called below instants) inhabiting different event types. The first class citizens of the model are clocks, which are considered as sources of monotypic instants. The uniqueness of the source for each event type means that all instants of the same event type are linearly ordered in time, i.e. for any pair of such instants, we can exactly establish what instant from this pair has happened before.

We are interested in studying such relationships between instants which do not depend on fortuitous aspects of behaviours of the system being studied, but which fix regularities of behaviours of the system. In other words, our interests are focused on relations of causality.

There are two approaches to study these relationships, namely, the approach based on representing admissible system behaviours by using sequences of messages describing the sets of simultaneous event occurrences, and the approach based on representing admissible system behaviours by using objects of the special category, the category of clock structures. The first approach (see Sec 3) can be used to define the operational semantics of the specification languages, and the second approach (see Sec 4) can be used to define the denotational semantics for languages describing the temporal requirements limiting possible system behaviours.

Thus, understanding the interrelationship between these two approaches is an important both theoretical and applied problem for the theory of CPS. The first results concerning this problem were obtained in [15]. This paper develops the study presented in the mentioned paper.

3 Model of Acceptable Schedules

The operational approach goes back, apparently, to L. Lemport's long-standing paper [8]. This approach is based on the simple idea to distinguish correct and incorrect system behaviours by observing streams of system messages that carry information about occurred events. In the context the concept of safety property introduced by L. Lamport [7] is very important. This concept was formalised by B. Alpern and F. B. Schneider in [1]. They also established an interconnection between the concepts of safety and liveliness and the topological properties of the corresponding system traces.

3.1 Clocks, Messages, and Schedules

As mentioned above all monotypic instants have the same source. We call these sources by clocks and introduce some finite set C whose elements are used to refer to clocks.

Definition 1. *Any non-empty subset of C is called a message.*

The corresponding set of messages we denote below by $\mathbf{M}_C$.

Definition 2. *A schedule* (*or more precisely a C-schedule*) *is an infinite sequence of messages.*

As usual, the set of all C-schedule we denote by $\mathbf{M}_C^\omega$.
Any message $\mu \in \mathbf{M}_C$ is interpreted as the notification "at the moment, each clock belonging to μ and only they fired the event occurrence".

Definition 3. *Let $n \in \mathbb{N}$ and $\boldsymbol{u}$ be a non-empty finite sequence* (*a non-empty word*)[1] *of messages then the pair $(n, \boldsymbol{u})$ is called a local schedule that starts at time point n.*

To work with sequences and words, we use the notation like the Python notation for sequential data types.

3.2 Topology on $\mathbf{M}_C^\omega$

This subsection contains some facts of the general topology necessary to understand the topological nature of the notions of safety and liveliness. For more detailed acquaintance with the subject, you can refer to [1,14].

Proposition 1. *The family $\left\{Z_n(\boldsymbol{u}) \mid n \in \mathbb{N}_+, \boldsymbol{u} \in \mathbf{M}_C^+\right\}$ where $Z_n(\boldsymbol{u}) = \{\boldsymbol{\pi} \in \mathbf{M}_C^\omega \mid \boldsymbol{\pi}[n : n + len(\boldsymbol{u})] = \boldsymbol{u}\}$ forms the base of Tikhonov topology on $\mathbf{M}_C^\omega$.*

[1] The set of non-empty words we denote as usually by $\mathbf{M}_C^+$.

Proposition 2. *Let* $\{\boldsymbol{\pi}_n \mid n \in \mathbb{N}\}$ *be a sequence of C-schedules and* $\boldsymbol{\pi}$ *be a C-schedule then* $\boldsymbol{\pi}_n \xrightarrow[n\to\infty]{} \boldsymbol{\pi}$ *in Tikhonov topology iff for any* $M \in \mathbb{N}_+$ *there exists* $N \in \mathbb{N}_+$ *such that for any* $n > N$ *the equality* $\boldsymbol{\pi}_n[0:M] = \boldsymbol{\pi}[0:M]$ *holds.*

Definition 4. *Let P be a subset of* $\mathbf{M}_C^\omega$ *then P is called closed if for any schedule sequence* $\{\boldsymbol{\pi}_n \in P \mid n \in \mathbb{N}\}$ *such that there exists* $\boldsymbol{\pi} = \lim\limits_{n\to\infty} \boldsymbol{\pi}_n$ *the schedule* $\boldsymbol{\pi}$ *is also a member of P.*

3.3 Safety Properties

Speaking not formally, L. Lamport proposed to recognise the property of schedules (the set of schedules satisfying this property) a safety property if any violation of this properties can be detected by the way of system observing during a finite time interval [7]. The following definition describes formally a safety property.

Definition 5. *Let P be a property of C-schedules then P is a safety property iff for any* $\boldsymbol{\pi} \notin P$ *there exists* $n \in \mathbb{N}_+$ *such that* $\boldsymbol{\pi}' \notin P$ *for each* $\boldsymbol{\pi}' \in Z_0[\boldsymbol{\pi}[0:n]]$.

In other words, a property P is a safety property iff the set of schedules satisfying P is a closed set in Tikhonov topology.
One can find the detailed discussion of the formal definition of safety properties and their topological characteristics in [1].

We consider that any acceptable behaviour of CPS is being described by the corresponding safety property. Safety ensures that the corresponding property is physically correct because it can be checked using information obtained in the past and present. In other words, checking such a property does not require the presence of magical abilities like foresight ability.

4 Category of Clock Structures

We try to describe the denotational approach to modelling of CPS behaviour in this section. We emphasize that if the approach specified above gives acceptable schedules in physical time, then the denotational approach describes a pure logical picture of relations between event occurrences without any references to physical time.

4.1 Quasi-Ordered Sets

This subsection is given to introduce the mathematical basis for the denotational approach to semantic modelling of CPS behaviour. The principal source is [5]. It

is well known that the quasi-ordered set is a set equipped with a binary relation that is reflexive and transitive.

As usual, for a quasi-ordered set $(X, \preccurlyeq)$ we define the following derived binary relations on X (see Table 1).

Table 1. The Derived Relations on a Quasi-ordered Set

Name	Properties	Notation	Definition
coincidence	reflexive, antisymmetric, and transitive	$i \equiv j$	$i \preccurlyeq j \ \wedge \ j \preccurlyeq i$
precedence	irreflexive and transitive	$i \prec j$	$i \preccurlyeq j \wedge j \not\preccurlyeq i$
exclusion	symmetric	$i \# j$	$i \prec j \ \vee \ j \prec i$
independence	symmetric	$i \parallel j$	$i \not\preccurlyeq j \ \wedge \ j \not\preccurlyeq i$

Further, for a quasi-ordered set $(X, \preccurlyeq)$ and $i \in X$ the principal ideal generated by i is the subset $(\, i \,]$ of X defined as $(\, i \,] = \{j \in X \mid j \preccurlyeq i\}$.

4.2 Clock Structures

We start this section with the following formal definition.

Definition 6. *Let C be a finite set, each element of which is interpreted as a reference to the source* (*it is called a clock*) *of occurrences of the same event. Then a C-structure $\mathcal{S}$* [2] *is a triple $(\mathcal{I}, \gamma, \preccurlyeq)$ where*

- *$\mathcal{I}$ is the set of instants corresponding to the occurrences of events,*
- *$\preccurlyeq$ is a quasi-order on $\mathcal{I}$ that models the causality relation between instants, and, finally,*
- *$\gamma\colon \mathcal{I} \to C$ is a surjective mapping that associates each instant with the clock that is the source of this instant*

provided that the following axioms met:

***the axiom of unbounded liveness**:*
the set $\mathcal{I}$ is infinite; (1)

***the axiom of finite causality**:*
for any $i \in \mathcal{I}$ the corresponding principal ideal $(\, i \,]$ is finite; (2)

***the axiom of total ordering for clock timelines**:*
for each $c \in C$ the set $\mathcal{I}_c = \gamma^{-1}(c)$ is linearly ordered by the corresponding restriction of "$\preccurlyeq$". (3)

This definition is a repetition of the corresponding definition given in [10].

Some simple conclusions from this definition are gathered in the following proposition.

[2] Usually, one uses the term clock structure if C is uniquely determined by the context.

Proposition 3. *Let $\mathcal{S} = (\mathcal{I}, \gamma, \leqslant)$ be a C-structure then*

1. *width of the ordered set $(\mathcal{I}, \prec)$ is less than or equal to $|C|$;*
2. *for each $c \in C$ the set $\mathcal{I}_c$ is well-ordered;*
3. *for each $c \in C$ the ordinal type of $\mathcal{I}_c$ is less than or equal to ω;*
4. *there is at least one $c \in C$ such that its ordinal type equals ω;*
5. *the set $\mathcal{I}$ is countable;*
6. *if $i, j \in \mathcal{I}$ and $i \equiv j$ then either $i = j$ or $i \not\prec j$ and $j \not\prec i$, i.e. any equivalence class for the relation "$\equiv$" is an antichain for the strict order $\prec$;*
7. *if $i, j, i', j' \in \mathcal{I}$, $i \prec j$, $i \equiv i'$, and $j \equiv j'$ then $i' \prec j'$;*
8. *each instant $i \in \mathcal{I}$ is uniquely characterized by the pair $(\gamma(i), \mathrm{idx}(i))$ where $\mathrm{idx}\colon \mathcal{I} \to \mathbb{N}$ is defined as follows*

$$\mathrm{idx}(i) = \left|\{j \in \mathcal{I}_{\gamma(i)} \mid j \prec i\}\right|.$$

4.3 Morphisms of Clock Structures

As usual, we define morphisms of C-structures to describe the relationship between them.

Definition 7. *Let $\mathcal{S}'$ and $\mathcal{S}''$ be C-structures, $\mathcal{I}'$ and $\mathcal{I}''$ be the corresponding sets of instants then a mapping $f\colon \mathcal{I}' \to \mathcal{I}''$ is called a C-morphism from $\mathcal{S}'$ into $\mathcal{S}''$ if the following holds*

1. *$\gamma(i) = \gamma(f(i))$ for any $i \in \mathcal{I}'$;*
2. *$i \leqslant j$ implies $f(i) \leqslant f(j)$ for any $i, j \in \mathcal{I}'$;*
3. *$i \# j$ implies $f(i) \# f(j)$ for any $i, j \in \mathcal{I}'$.*

Note 1. Usually, we do not distinguish symbols used to denote the causality relations and the mappings associated instants with their sources for different clock systems.

Note 2. The fact that f is a C-morphism from $\mathcal{S}'$ into $\mathcal{S}''$ is as usually denoted by $f\colon \mathcal{S}' \to \mathcal{S}''$.

The following statement establishes an important property of C-morphisms.

Proposition 4. *Any C-morphism is an injective mapping.*

Proof. Indeed, let us suppose that $f\colon \mathcal{I}' \to \mathcal{I}''$ be a morphism of C-structures $(\mathcal{I}', \leqslant, \gamma)$ and $(\mathcal{I}'', \leqslant, \gamma)$, $i \neq j \in \mathcal{I}'$, and $f(i) = f(j)$. Then either $\gamma(i) \neq \gamma(j)$ or $\gamma(i) = \gamma(j)$ and $\mathrm{idx}(i) \neq \mathrm{idx}(j)$ (see Prop 3, item 8).
Firstly, let us assume that $\gamma(i) \neq \gamma(j)$ but then we get $\gamma(f(i)) = \gamma(i) \neq \gamma(j) = \gamma(f(j))$ and, therefore, $f(i) \neq f(j)$. This contradicts to the supposition, hence the case is impossible.

Secondly, let us assume that $\gamma(i) = \gamma(j) = c$ and $\mathrm{idx}(i) \neq \mathrm{idx}(j)$. Let for definiteness $\mathrm{idx}(i) < \mathrm{idx}(j)$ then $\gamma(i) = \gamma(j)$ ensures $i \prec j$. But this means that $i \prec j$, i.e. $i \# j$ and, therefore, $f(i) \# f(j)$, i.e. we have that $f(i) \equiv f(j)$ is false and, hence, $f(i) = f(j)$ is false also. Thus, in this case, we also obtain a contradiction with the supposition. The case $\mathrm{idx}(j) < \mathrm{idx}(i)$ is analysed by the similar way. □

The following statement is evident.

Proposition 5. *For any finite set C, the class of C-structures together with the class of C-morphisms form a small category* [3].

Corollary 1 (of Prop 4). *Any C-morphism is a monomorphism in the category of C-structures.*

The above results lead to the following classification of C-morphisms.

Definition 8. *A C-morphism $f\colon \mathcal{S}' \to \mathcal{S}''$ is called a covering C-morphism if the mapping $f\colon \mathcal{I}' \to \mathcal{I}''$ is surjective.*

The following evident proposition clarifies logical relations between different classes of C-morphisms.

Proposition 6. *Logical relations between the notions C-isomorphism, covering C-morphism, C-epimorphism, C-bimorphism, C-monomorphism, and C-morphism are shown in Fig. 2.*

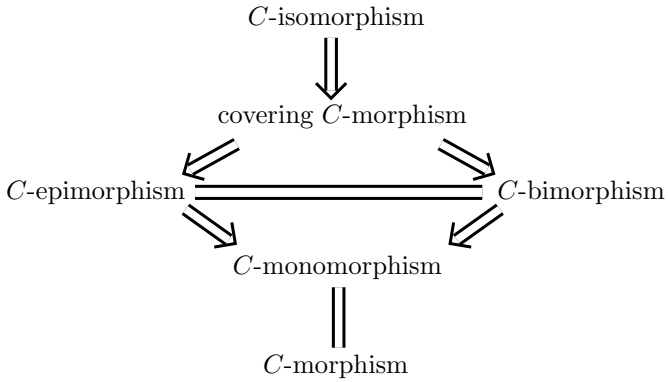

Fig. 2. Logical relations between different classes of C-morphisms

[3] The necessary definitions and results from the theory of categories can be found in [9].

5 Interrelation between Operational and Denotational Approaches

This section contains some results concerning mutual relationships between operational and denotational models. Below we construct a relation between clock structures and schedules. This relation describes the possible ways to dip a clock structure in physical time. Properties of schedules represent these ways.

5.1 Some Needed Technique

Below we need the following notion.

Definition 9. *Let $\mathcal{S} = (\mathcal{I}, \gamma, \preccurlyeq)$ be a C-structure, $A \subset \mathcal{I}$, and $i \in A$ then i is called a minimal instant in A if for any $j \in A$ the statement $j \prec i$ is false.*

The subset of minimal instants of A is below denoted by $\min A$.

We associate the sequence of slices $\mathcal{I}[0], \mathcal{I}[1], \ldots$ with any C-structure $\mathcal{S}$ in the following manner

$$\mathcal{I}[0] = \min \mathcal{I}$$
$$\mathcal{I}[n] = \min\left(\mathcal{I} \setminus \bigcup_{m=0}^{n-1} \mathcal{I}[m]\right) \quad \text{for } n > 0$$

where $\mathcal{I}$ is the instant set of $\mathcal{S}$.

Proposition 7. *The following properties hold:*

1. $|\mathcal{I}[n]| \leq |C|$ *for each* $n \in \mathbb{N}$;
2. *if* $i, j \in \mathcal{I}[n]$ *for some* $n \in \mathbb{N}$ *then either* $i \parallel j$ *or* $i \equiv j$;
3. *the sequence of slices is a covering of the set of instants;*
4. *if* $i \in \mathcal{I}[n]$ *for some* $n \in \mathbb{N}$, $j \in \mathcal{I}$, *and* $j \equiv i$ *then* $j \in \mathcal{I}[n]$;
5. *if* $i \in \mathcal{I}[n+1]$ *for some* $n \in \mathbb{N}$ *then there exists* $j \in \mathcal{I}[n]$ *such that* $j \prec i$.

Proof. 1) This property follows directly from Def 6 and item 6 of Prop 3.
2) Indeed, each of statements $i \prec j$ and $j \prec i$ contradicts to the statement that the both statements $i \in \mathcal{I}[n]$ and $j \in \mathcal{I}[n]$ are true.
3) Suppose that there exists some $i \in \mathcal{I}$ such that $i \notin \mathcal{I}[n]$ for any $n \in \mathbb{N}$ then using induction one can demonstrate that for each $n \in \mathbb{N}$ there exists $j_n \in \mathcal{I}[n]$ such that $j_n \prec i$. Taking into account that $\mathcal{I}[m] \bigcap \mathcal{I}[n] = \varnothing$ for $m \neq n$ we conclude that the set $\{j_n \mid n \in \mathbb{N}\}$ is infinite. But this set is a subset of $(\, i \,]$. This contradiction (see Def 6) demonstrates that the proposition is true.
4) If $j \notin \mathcal{I}[n]$ then item 3 ensures that $j \in \mathcal{I}[m]$ for some $m \neq n$.
If $m < n$ then $k \prec i$ for some $k \in \mathcal{I} \setminus \left(\bigcup_{s=0}^{n-1} \mathcal{I}[s]\right)$. The statement $i \equiv j$ ensures

that $k \prec j$ and, therefore, $j \notin \min\left(\mathcal{I} \setminus \bigcup\limits_{s=0}^{m-1} \mathcal{I}[s]\right)$. This contradiction shows that $m > n$. But similar reasoning demonstrates that the assumption $m > n$ leads also to a contradiction.
5) If $i \in \mathcal{I}[n+1]$ for some $i \in \mathcal{I}[n]$ then $j \not\prec i$ for any $j \in \mathcal{I} \setminus \left(\bigcup\limits_{0 \le k \le n} \mathcal{I}[k]\right)$. If $j \not\prec i$ for all $j \in \mathcal{I}[n]$ also then $j \not\prec i$ for any $j \in \mathcal{I} \setminus \left(\bigcup\limits_{0 \le k < n} \mathcal{I}[k]\right)$. This means that $i \in \bigcup\limits_{0 \le k < n} \mathcal{I}[k]$ and, therefore, $i \in \mathcal{I}[n+1]$ is false. This proves the item of the proposition. □

Corollary 2. *An instant i belongs to $\mathcal{I}[n]$ iff number of elements of each maximal chain in $(\, i \,]$ equals $n + 1$.*

Proof. It is being proved by induction with respect to n. □

Corollary 3. *Slice $\mathcal{I}[n]$ for each $n \in \mathbb{N}$ is a union of $\equiv$-equivalence classes of elements belonging the slice, moreover slice elements lying to different classes are independent.*

5.2 Linear Clock Structures

An important special class of clock structures is formed by so-called linear clock structures.

Definition 10. *A C-structure $\mathcal{L} = (\mathcal{I}, \gamma, \preccurlyeq)$ is called a linear C-structure if $i \parallel j$ is false for any $i, j \in \mathcal{I}$.*

Any linear clock structure holds the following property.

Proposition 8. *Let $\mathcal{L} = (\mathcal{I}, \gamma, \preccurlyeq)$ be a linear C-structure, $\mathcal{S}' = (\mathcal{I}', \gamma, \preccurlyeq)$ be a C-structure, and $f\colon \mathcal{I} \to \mathcal{I}'$ be a covering C-morphism then f is a C-isomorphism.*

Proof. Indeed, taking into account that f is a covering C-morphism one can conclude that f is a bijection, hence there exists the unique mapping $g\colon \mathcal{I}' \to \mathcal{I}$ such that $g(f(i)) = i$ for each $i \in \mathcal{I}$ and $f(g(i')) = i'$ for each $i' \in \mathcal{I}'$.
Now let us demonstrate that g is a C-morphism.
If $i', j' \in \mathcal{I}'$ and $i' \preccurlyeq j'$ then for $i = g(i')$ and $j = g(j')$ the linearity of $\mathcal{I}$ ensures either $j \prec i$ or $i \preccurlyeq j$.
The statement $j \prec i$ implies $j \# i$ and, therefore, $f(j) = j' \# f(i) = i'$, i.e. $j' \prec i' \vee i' \prec j'$. On the other hand, the statement $j \prec i$ implies $j \preccurlyeq i$ and, therefore, $j' \preccurlyeq i'$. But the assertions $i' \prec j'$ and $j' \preccurlyeq i'$ are evidently incompatible, hence $j' \prec i'$ is necessary true. Taking into account that $j' \prec i'$ and $i' \preccurlyeq j'$ are

incompatible one can conclude that $i \preccurlyeq j$ is the unique acceptable variant. Thus, we proved that $g(i') \preccurlyeq g(j')$.
Further, if $i', j' \in \mathcal{I}'$ and $i' \# j'$ then for $i = g(i')$ and $j = g(j')$ the linearity of $\mathcal{I}$ ensures $i \equiv j$ or $i \# j$. Taking into account that $i \equiv j$ ensures $i' \equiv j'$ we conclude that $i \equiv j$ contradicts to $i' \# j'$. Thus, we proved that $g(i') \# g(j')$.
Therefore, g is the inverse C-morphism for f. □

As it is below shown the proposition converse to Prop. 8 is also true.

5.3 Analogue of Szpilrajn Extension Theorem

The concept of extension is an important concept in ordered sets theory. Therefore we study its analogue for clock structures.

Definition 11. *Let $\mathcal{S} = (\mathcal{I}, \gamma, \preccurlyeq)$ be C-structures then a covering non-invertible C-morphism $e\colon \mathcal{I} \to \mathcal{I}'$ for some C-structure $\mathcal{S}' = (\mathcal{I}', \gamma, \preccurlyeq)$ is called an extension of $\mathcal{S}$. Whenever $\mathcal{S}'$ is a linear C-structure we say that e is a linear extension of $\mathcal{S}$.*

The following theorem refines Theorem 3 in [15].

Theorem (about Linear Extension). *Let $\mathcal{S} = (\mathcal{I}, \gamma, \preccurlyeq)$ be a C-structure and i_*, j_* be some pair of independent instants belonging to $\mathcal{I}$ then there exists a linear extension $e\colon \mathcal{S} \to \mathcal{L}$ such that the condition $e(i_*) \prec e(j_*)$ holds.*

To prove the theorem we need in the following lemmas.

Lemma 1. *Let $\mathcal{S} = (\mathcal{I}, \gamma, \preccurlyeq)$ be a C-structure and i_*, j_* be two independent instants in $\mathcal{I}$ then there exists an extension $e\colon \mathcal{S} \to \mathcal{S}^*$ such that $e(i_*) \prec e(j_*)$.*

Proof. Let $\mathbf{1} = \{*\}$ be some singleton. Then let us define

$$\begin{aligned}
&\mathcal{I}^* = \mathcal{I} \times \mathbf{1}\\
&\gamma(i, *) = \gamma(i) \quad \text{for any } i \in \mathcal{I}\\
&(i, *) \preccurlyeq (j, *) \text{ iff either } i \preccurlyeq j \text{ or } i \preccurlyeq i_* \text{ and } j_* \preccurlyeq j \quad \text{for any } i, j \in \mathcal{I}\\
&e(i) = (i, *) \quad \text{for any } i \in \mathcal{I}.
\end{aligned}$$

Let us check that "$\preccurlyeq$" defined on $\mathcal{I}^*$ is a quasi-order. Indeed, it is evident that this relation is reflexive. If now we have that $(i, *) \preccurlyeq (j, *)$ and $(j, *) \preccurlyeq (k, *)$ for some $i, j, k \in \mathcal{I}$ then the next variants are only possible:

1. $i \preccurlyeq j$ and $j \preccurlyeq k$; these conditions ensure $i \preccurlyeq k$ and, hence, $(i, *) \preccurlyeq (k, *)$;
2. $i \preccurlyeq i_*$, $j_* \preccurlyeq j$, and $j \preccurlyeq k$; these conditions ensure $i \preccurlyeq i_*$ and $j_* \preccurlyeq k$, but this means that $(i, *) \preccurlyeq (k, *)$;

3. $i \preccurlyeq j$, $j \preccurlyeq i_*$, and $j_* \preccurlyeq k$; these conditions ensure $(i,*) \preccurlyeq (k,*)$ that is checked similarly to above.

Thus, "$\preccurlyeq$" is a quasi-order on $\mathcal{I}^*$.
One can easily see that $(i_*,*) \preccurlyeq (j_*,*)$, but $(j_*,*) \not\preccurlyeq (j_*,*)$, i.e. $e(i_*) \prec e(j_*)$.
The construction ensures evidently that $i \# j$ implies $e(i) \# e(j)$.
And, finally, it is evident that $(\,(i,*)\,] \subset (\,i\,] \bigcup (\,i_*\,]$. Therefore, the set $(\,(i,*)\,]$ is finite for any $(i,*) \in \mathcal{I}^*$.
Thus, $\mathcal{S}^* = (\mathcal{I}^*, \gamma, \preccurlyeq)$ is a C-structure, $e\colon \mathcal{S} \to \mathcal{S}^*$ is a covering C-morphism, and $e(i_*) \prec e(j_*)$. □

Lemma 2. *Let $\mathcal{S} = (\mathcal{I}, \gamma, \preccurlyeq)$ then for $\mathcal{I}^* = \mathcal{I}\times\mathbf{1}$; $\gamma(i,*) = \gamma(i)$; and $(i,*) \preccurlyeq (j,*)$ meaning $i \in \mathcal{I}[m]$, $j \in \mathcal{I}[n]$, and $m \le n$ the triple $\mathcal{L} = (\mathcal{I}^*, \gamma, \preccurlyeq)$ is a linear C-structure. Moreover, $e\colon \mathcal{S} \to \mathcal{L}$ that is defined as $e(i) = (i,*)$ is a linear extension of $\mathcal{S}$.*

Proof. Taking into account that $\mathcal{I}[m] \bigcap \mathcal{I}[n] = \varnothing$ if $m \neq n$ and item 3 of Prop 7 one can conclude that $\preccurlyeq$ is a correctly defined quasi-order on $\mathcal{I}^*$, moreover items 4 and 5 of Prop 7 ensure that $i \preccurlyeq j$ implies $(i,*) \preccurlyeq (j,*)$ and $i \prec j$ implies $(i,*) \prec (j,*)$. Thus, $\mathcal{L}$ is a linear C-structure and e is a covering C-morphism. Therefore, e is a linear extension of $\mathcal{S}$. □

Proof (of Theorem about Linear Extension).
Let $e'\colon \mathcal{S} \to \mathcal{S}'$ be the extension of $\mathcal{S}$ constructed in accordance with Lemma 1 end e'' be the linear extension of $\mathcal{S}'$ constructed in accordance with Lemma 2 then $e = e'' \circ e'$ is a linear extension of $\mathcal{S}$.
Construction of e' ensures that $e'(i_*) \prec e'(j_*)$ and construction of e'' ensures that $i' \prec j'$ implies $e''(i') \prec e''(j')$. Thus, $e(i_*) \prec e(j_*)$. □

Corollary 4. *Each C-structure $\mathcal{S}$ is uniquely defined by the family $\{e_\alpha\}$ of all its linear extensions in the following sense*

$$\textit{if } i, j \in \mathcal{I} \textit{ then } i \preccurlyeq j \textit{ iff } e_\alpha(i) \preccurlyeq e_\alpha(j) \textit{ for all } e_\alpha.$$

Corollary 5. *If C-structure $\mathcal{S}$ satisfies the property "any covering C-morphism from $\mathcal{S}$ is a C-isomorphism" then this C-structure is linear.*

Note 3. Cor 5 is the claimed inversion of Prop 8.

Note 4. The obtained criterion for the linearity of a clock structure shows that all linear clock structures and only them are "weak terminal" with respect to covering clock morphisms.

5.4 Relation of Admissibility

Linear extensions of a clock structure can be considered as admissible realisations of the clock structure in physical time. In this subsection, the corresponding formal relation called admissibility is defined.

Let us assume that some finite set of clock are fixed. Then we associate the linear C-structure $\mathcal{L}^\pi = (\mathcal{I}^\pi, \preccurlyeq, \gamma)$ with any C-schedule π in the following manner:

1. $\mathcal{I}^\pi = \{(n, c) \in \mathbb{N} \times C \mid c \in \pi(n)\}$;
2. $\gamma^\pi(n, c) = c$ for $(n, c) \in \mathcal{I}^\pi$;
3. $(m, a) \preccurlyeq (n, b)$ means $m \leq n$ for any (m, a) and (n, b) belonging to $\mathcal{I}^\pi$.

Thus we can define the following relation.

Definition 12. *We say that a C-structure $\mathcal{S}$ admits a C-schedule π and denote this fact by $\mathcal{S} \models \pi$ if there exists an extension $e\colon \mathcal{S} \to \mathcal{L}^\pi$.*

Def 12 leads us to the notion property of clock structure.

Definition 13. *A set of C-schedules P is a property of C-structure $\mathcal{S}$ [4] if for any $\pi \in P$ the condition $\mathcal{S} \models \pi$ is fulfilled.*

Cor 4 of Theorem about Linear Extension shows that any clock structure can be uniquely specified by some property, which we call the characteristic property of the structure. Unfortunately, at this time we do not how can be characterised the class of properties that contains all characteristic properties of clock structures and only them. Such a hypothesis seems plausible.

Conjecture. *The characteristic property of any clock structure is a safety property.*

6 Conclusion

The paper discusses the problems of the interaction of components for an important class of complex systems, so-called cyber-physical systems. The clock model, first described by L. Lamport has been chosen as the principle tool for research. Such a choice is motivated by rich expressive means of the model language, which provide the specification process both synchronous and asynchronous methods of intercomponent interactions. The successful practice of using this model for specification temporal guarantees and constraints for system behaviour on the base of Clock Constraint Specification Language was also taking into account.

[4] Symbolically, $\mathcal{S} \models P$.

Two approaches to modelling logical time for cyber-physical system have been considered in the paper.

The first approach is based on the notion schedule, which is used to model an acceptable sequence of occurrences of system events. Each set of schedules can be considered as a specification of the required property of system behaviour. In the context, the key notion is safety property that ensures the guarantee to detect a violation of system behaviour during a finite time.

The second approach had been proposed to define the denotational semantics of Clock Constraint Specification Language. In the paper, our efforts have been focused on developing this approach on the base of the category-theoretic language. Sets of instants equipped by the causality relation and the mapping associating each instant with its source are objects of the corresponding category, a category of clock structures. Using the category-theoretic language has given a possibility to refine and make more rigorous this model due to the suited definition of morphisms. The theorem about linear extension under such an approach is becoming more expressive. This theorem, which has been proved in the paper, is a bridge between two approaches being analysed.

The manner to use Theorem about Linear Extension to establish interdependence between clock structures and properties of schedules has been described in the last part of the paper.

It has been shown that properties of clock structures are identified by sets of schedules. It is also clear that there exist sets that are not characteristic properties of the corresponding clock structure. But the problem to prove the Conjecture formulated in Subsection 5.4 is very important because if Conjecture is fulfilled then the role of safety properties obtains not only operational but denotational meaning.

Another very important and interest problem consists in studying the features of the class of properties that are characteristic properties of clock structures.

We hope that further study of this topic will lead to formulating the weakest necessary requirements for the languages of the behaviour constraint specification based on the clock model for cyber-physical systems.

References

1. Alpern, B., Schneider, F. B.: Defining Liveness. Information Processing Letters. 21, 181–185 (1985).
2. Cachin, C., Guerraoui, R., Rodrigues, L.: Introduction to Reliable and Secure Distributed Programming, 2nd edition. Springer-Verlag Berlin Heidelberg (2011).
3. Carruthers, K.: Internet of Things and Beyond: Cyber-Physical Systems. IEEE IoT Newsletter. May (2016), http://iot.ieee.org/newsletter/may-2016/internet-of-things-and-beyond-cyber-physical-systems.html

4. Glitia, C., Deantoni, J., Mallet, F.: Logical Time @ Work: Capturing Data Dependencies and Platform Constraints. In: Kaźmierski, T. J. J., Morawiec, A (Eds.). System Specification and Design Languages. LNEE, vol. 106, 223–238. Springer New York (2012).
5. Harzheim, E.: Ordered Sets. Springer Science & Business Media (2006).
6. Kshemkalyani, A. D., Singhal, M.: Distributed Computing: Principles, Algorithms, and Systems. Cambridge University Press (2008)
7. Lamport, L.: Proving the Correctness of Multiprocess Programs. IEEE Transactions on Software Engineering. 2, 125–143 (1977).
8. Lamport, L.: Time, clocks, and the ordering of events in a distributed system. CACM. 21(7), 558–565 (1978), http://lamport.azurewebsites.net/pubs/time-clocks.pdf.
9. Mac Lane, S.: Categories for the Working Mathematician, 2nd ed. Springer-Verlag New York Inc (1998).
10. Mallet, F.: Clock constraint specification language: specifying clock constraints with UML/MARTE. Innovations in Systems and Software Engineering. 4(3), 309–314 (2008).
11. Mallet, F.: MARTE/CCSL for Modeling Cyber-Physical Systems. In: Drechsler, R. and Kühne, U. (Eds.) Formal Modeling and Verification of Cyber-Physical Systems. Pp. 26–49. Springer Fachmedien Wiesbaden (2015).
12. The National Science Foundation. Cyber-Physical Systems (CPS). Program Solicitation NSF 12-520. Arlington, VA: NSF, 2012, https://www.nsf.gov/pubs/2012/nsf12520/nsf12520.htm#toc
13. The National Science Foundation. Cyber-Physical Systems (CPS). Program Solicitation NSF 17-529. Arlington, VA: NSF, 2017, https://www.nsf.gov/pubs/2017/nsf17529/nsf17529.htm#toc
14. Willard, S.: General Topology. Dover Publication Inc. Mineola, NY (1998)
15. Zholtkevych, G., Mallet, F., Zaretska, I., Zholtkevych, G.: Two Semantic Models for Clock Relations in the Clock Constraint Specification Language. In: Ermolayev, V. et al (Eds.) Information and Communication Technologies in Education, Research, and Industrial Applications. CCIS, vol. 412, 190–209. Springer International Publishing (2013).