UDC 004.62 https://doi.org/10.15407/pp2022.03-04.216

TABLE INTERPRETATION OF THE TEMPORAL DESCRIPTION LOGIC LTL_{ALC}

Valerii Reznichenko, Inna Chystiakova

Дескриптивні логіки широко застосовуються для опису та представлення знань у семантичному веб. Це сучасний та потужний механізм, що надає можливість виводу знань із уже існуючих. Завдяки цьому, концептуальне моделювання предметних областей, з урахуванням використання механізмів виводу, стало однією з галузей застосування дескриптивних логік. Концептуальне моделювання використовується для створення баз даних та знань. Ключовим питанням при моделюванні предметної області постає можливість відслідковувати динаміку зміни стану предметної області у часі. Тобто, потрібно описати не лише поточний актуальний стан бази даних (знань), але і передісторію також. Для вирішення цієї задачі використовуються темпоральні дескриптивні логіки. Вони мають такий самий набір алгоритмічних проблем, який присутній у звичайних дескриптивних логіках, але до них додаються питання, пов'язані з описом знань у часі. Це стосується форми часу (неперервний чи дискретний), структури (моменти часу, інтервали, ланцюжки інтервалів), лінійності (лінійний чи розгалужений), область (теперішній, минулий, майбутній), поняття «зараз», спосіб вимірювання тощо. Актуальною задачею на сьогоднішній день є створення алгоритму темпоральної інтерпретації звичайних дескриптивних логік. Тобто, показати спосіб, яким чином темпоральну дескриптивну логіку можна застосувати до звичайної дескриптивної логіки. У роботі представлено алгоритм інтерпретації темпоральної LTL у ALC. Для опису обрано лінійний, не розгалужений час, що представлений у вигляді цілої тимчасової осі з заданою на ній лінійною впорядкованістю. Розглядається лише майбутній час. Алгоритм містить графічні нотації застосування LTL у ALC: концептів, конструкторів концептів, ролей, конструкторів ролей, TBox та ABox. Для наочності застосування алгоритму використовуються багаточисельні приклади.

Ключові слова: LTL, DL, ALC, LTL_{ALC}, дескриптивна логіка, темпоральна дескриптивна логіка.

Description logics are widely used to describe and represent knowledge in the Semantic Web. This is a modern and powerful mechanism that provides the possibility of extracting knowledge from already existing ones. Thanks to this, conceptual of subject areas modeling has become one of the fields of application of descriptive logics, taking into account the use of inference mechanisms. Conceptual modeling is used to create databases and knowledge bases. A key issue of the subject area modeling is the ability to monitor the dynamics of changes in the state of the subject area over time. It is necessary to describe not only the current actual state of the database (knowledge bases), but also the background. Temporal descriptive logics are used to solve this problem. They have the same set of algorithmic problems that are presented in conventional descriptive logics, but to them are added questions related to the description of knowledge in time. This refers to the form of time (continuous or discrete), time structure (moments of time, intervals, chains of intervals), time linearity (linear or branched), domain (present, past, future), the concept of "now", the method of measurement, etc. An urgent task today is to create an algorithm for the temporal interpretation of conventional descriptive logics. That is, to show a way in which temporal descriptive logic can be applied to ordinary descriptive logic. The paper presents an algorithm for temporal interpretation of LTL into ALC. Linear, unbranched time is chosen for the description goal. It is presented in the form of a whole temporal axis with a given linear order on it. Only the future tense is considered. The algorithm contains graphic notations of LTL application in ALC: concepts, concept constructors, roles, role constructors, TBox and ABox. Numerous examples are used to illustrate the application of the algorithm.

Keywords: description logic, temporal description logic, DL, LTL, ALC, LTL_{ALC}.

Introduction

Description logics (DL) are a family of knowledge representation formalisms. They have different application areas, including semantic web ontologies. Particularly, OWL and OWL 2 of the W3C are grounded on the corresponding DL. Conceptual domain modeling problem that considers the use of withdrawal mechanisms became one of the DL application areas. Specially, it refers to the database and data knowledge issues. For example, extended ER-modeling language and extended UML class diagram were described with the help of DL. This allowed to use DL reasoners to check the software descriptions consistency (integrity) and perform influence of new implicitly specified knowledge.

Tracking the dynamics of software state changes over time is the key question during the software conceptual modeling. That is the situation, when software conceptual informational model and the correspondence database track both the current state and all the prehistory. Such a temporal database should both allow an ability to fix a lot of states in the past (and probably in the future) and give an opportunity to operate these sets of states with their temporal integrity constraints descriptions. To satisfy these needs, a big number of publications on the temporal DL (TDL) theme have been appeared the last several years.

For example, such TDLs were proposed and discovered in modern literature: ITL Interval temporal logic) [1], LTL (Linear temporal logic) [2], STL (Signal temporal logic) [3], TTL (Timestamp temporal logic) [4], PSL (Property specification language) [5], CTL* which generalizes LTL and CTL [6], HML (Hennessy–Milner logic) [7], MTL (Metric temporal logic) [8], MITL (Metric interval temporal logic) [8], TPTL (Timed propositional temporal logic) [9], TLTL (Truncated Linear Temporal Logic) [10].

This paper is dedicated to the table interpretation of the temporal description logic LTL_{ALC}. A survey of the LTL temporal description logic is also given. Publication includes the designation of the database that supports the interpretation of LTL and a set of temporal operations in such a database. The results are presented in a table form.

To define temporal interpretation of any DL there are three main questions:

1) What DL will be used?

2) What TDL will be used?

3) What interpretation of time is chosen?

According to the first question, there is a whole DL family characterized by the composition of elements (individuals, concepts, roles) and a set of constructors and axioms defined on these elements. About the second question, a chose of the TDL depends on the goals of the research. A rather incomplete list of temporal logics that is presented above. Third question can be answered according to the set of aspects:

- continuous and discrete time
- linear and branching time
- present, past, future
- definition of "now"
- time structure (points of time, intervals, chains of intervals)
- ways to measure time.

As for this research, the following variants were chosen:

Linear discrete non-branching time, represented as an integer time axis with a linear ordering specified on it. (N<). Only the future tense is considered.

LTL (Linear time temporal logic).

ALC (Attributive Concept Language with Complements).

In this research the focus is made on the issues that related to informational models. The list of issues looks like below: syntax and semantics of the language, data model (data structure, operations, integrity constraints) and query language.

The aim of the present work is to show the algorithm for application the temporal logic LTL to the DL ALC.

Section 2 is dedicated to some preliminaries that touch a brief survey of DL ALC and LTL basics. The rules of the DL temporal interpretation presented in Section 3. The temporal DL interpretation can be found in Section 4. Section 5 is dedicated to the temporal interpretation of roles. The temporal TBox description is presented in Section 6. The temporal ABox description can be found in Section 7. Section 8 is dedicated to the main conclusions and future research.

Preliminaries

LTL (Linear time temporal logic) is the logic that considers causal relationships in terms of time. The concepts of LTL are formed using the concept constructors of DL ALC enriched with the temporal constructors. This logic is used to describe the sequence of events and their relationship over time.

There are such temporal operators in the LTL logic:

- $\Diamond$ ($\Diamond\varphi$) – existence in some moment of time in future
- $\Box$ ($\Box\varphi$) – **global** existence in some moment of time (forever)
- $\bigcirc$ ($\bigcirc\varphi$) – existence in the **next** moment of time in future
- $\mathcal{U}$ ($\varphi\ \mathcal{U}\ \psi$) – until. φ exists until ψ in the current moment of time or in the future
- $\mathcal{R}$ ($\varphi\ \mathcal{R}\ \psi$) – release. φ exists until **the first come** of ψ including current moment of time. If ψ will not come, φ will exist forever.

Every temporal operator can be written with the help of operators "Until" ($\mathcal{U}$) and "Next" ($\bigcirc$). For example, existence and global existence operators can be written like below:

$\Diamond\varphi \equiv \top\ \mathcal{U}\varphi$

$\Box\varphi \equiv \neg\Diamond\neg\varphi \ \equiv \bot\mathcal{R}\varphi$

$\varphi\ \mathcal{R}\ \psi \equiv \neg\ (\neg\varphi\ \mathcal{U}\ \neg\psi)$

Equivalent transformation rules

There are the following rules for equivalent transformations of LTL formulas.

1. Duality (moving negation).

a) Operator $\bigcirc$ is self-dual:

$\neg\bigcirc\varphi \equiv \bigcirc\neg\varphi$

b) Operators $\Box$ and $\Diamond$ are dual:

$\neg\Box\varphi \equiv \Diamond\neg\varphi$

$\Box\varphi \equiv \neg\Diamond\neg\varphi$

$\neg\Diamond\varphi \equiv \Box\neg\varphi$

$\Diamond\varphi \equiv \neg\Box\neg\varphi$

c) Operators $\mathcal{U}$ and $\mathcal{R}$ are dual:

$\neg(\varphi\ \mathcal{U}\ \psi) \equiv (\neg\varphi\ \mathcal{R}\ \neg\psi)$

$\neg(\varphi\ \mathcal{R}\ \psi) \equiv (\neg\varphi\ \mathcal{U}\ \neg\psi)$

$(\varphi\ \mathcal{U}\ \psi) \equiv \neg(\neg\varphi\ \mathcal{R}\ \neg\psi)$

$(\varphi\ \mathcal{R}\ \psi) \equiv \neg(\neg\varphi\ \mathcal{U}\ \neg\psi)$

2. Distributivity rules.

a) Distributivity of $\bigcirc$ with respect to $\vee$, $\wedge$ and $\mathcal{U}$:

$\bigcirc(\phi \vee \psi) \equiv \bigcirc\phi \vee \bigcirc\psi$

$\bigcirc(\phi \wedge \psi) \equiv \bigcirc\phi \wedge \bigcirc\psi$

$\bigcirc(\phi\ \mathcal{U}\ \psi) \equiv \bigcirc\phi\ \mathcal{U}\ \bigcirc\psi$

b) Distributivity of $\Diamond$ with respect to $\vee$ and $\Box$ with respect to $\wedge$:

$\Diamond(\phi \vee \psi) \equiv \Diamond\phi \vee \Diamond\psi$

$\Box(\phi \wedge \psi) \equiv \Box\phi \wedge \Box\psi$

3. Special properties.

$\Diamond\phi \equiv \Diamond\Diamond\phi$

$\Box\phi \equiv \Box\Box\phi$

Description logic ALC.

Traditionally, DL is aimed at presenting terminological knowledge about software. Individuals, concepts, and roles are the main objects of such presentation. There are the following three types of sets in DL:

- $N_I = \{a_1, a_2,..., a_m\}$ – set of individuals names
- $N_C = \{A_1, A_2,..., A_n\}$ – set of atomic concepts names
- $N_{R} = \{R_1, R_2,..., R_k\}$ – set of atomic roles names

In relation to concepts and roles, constructors are defined that allow to build new concepts / roles from already defined individuals, concepts, and roles. Description logic in a DL family is specified by a set of valid concept and role constructors.

ALC (Atributive Language with Complement) is the simplest option in the DL family. It contains the minimum set of constructors, that is mandatory for all other DL in the family.

ALC syntax. Syntax ALC (a set of valid concepts is given by the following inductive definition.

- $\top$ - Thing (universal concept)
- A – atomic concept (every atomic concept is concept)
- C – arbitrary concept, $\neg C$ – negotiation of C is also a concept
- $C \sqcap D$ – concept intersection is also a concept, if C and D are concepts
- $\exists R.C$ – existential restriction is also a concept, if C is a concept and R – atomic role

Usually, ALC also include the following additional concepts and constructors, which are defines with the help of concepts and constructors defined above:

- $\bot \equiv \neg\top$ – Nothing (empty concept)
- $C \sqcup D \equiv \neg(\neg C \sqcap \neg D)$ – concept union is also a concept, if C and D are concepts
- $\forall R.C \equiv \neg\exists R.\neg C$ – universal restriction is also a concept, if C is a concept and R – atomic role

Usually, to declare ALC syntax an extended version is used. It defines as follows:

$\top \mid \bot \mid A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C$

ALC semantics. ALC semantics (just like any other DL) is based on the interpretation notion. It looks like this: $I = (\Delta, \cdot^I)$, where Δ is a non-empty set, called the domain of interpretation (domain) and a^I is an interpretation function. The interpretation function puts in line:

- each individual name $a \in N_I$ to element $a^I \in \Delta$
- each atomic concept name $A \in N_C$ to arbitrary subset $A^I \subseteq \Delta$
- each atomic role name $R \in N_R$ to arbitrary subset $R^I \subseteq \Delta \times \Delta$.

The interpretation function $\cdot^I$ extends to ALC compound concepts as follows:

- $\top^I = \Delta$, $\bot^I = \emptyset$
- $(\neg C)^I = \Delta \setminus C^I$;
- $(C \sqcap D)^I = C^I \cap D^I$, $(C \sqcup D)^I = C^I \cup D^I$
- $(\exists R.C)^I = \{e \in \Delta \mid$ exist $d \in \Delta$ such that $(e, d) \in R^I$ и $d \in C^I\}$
- $(\forall R.C)^I = \{ e \in \Delta \mid$ for all $d \in \Delta$ such, that $(e, d) \in R^I$, occurs $d \in C^I\}$

Terminologies and statements (TBox and ABox).

Concepts describe facts that exist in the subject area, and their constructors allow to perform operations on them. They work as the basis to describe subject area knowledge. There are two types of knowledge: *intensional knowledge* (general knowledge about concepts) and *extensional knowledge* (knowledge about individual objects). Intensional knowledge is more stable and permanent. Extensional knowledge is more exposed modifications. According to this division, knowledge that are recorded using DL can be subdivided into:

- a set of terminological axioms (TBox)
- a set of statements (facts) about individuals (ABox)

Both TBox and ABox form a knowledge base.

TBox. This is a finite set of terminological axioms (terminologies) of the form:

- $C \sqsubseteq D$ – concept inclusion axioms
- $C \equiv D$ – axioms of concept identity

Obviously, these axioms are represented in the following way through the axioms of inclusion:

$C \sqsubseteq D \wedge D \sqsubseteq C$

ABox. This is a finite set of assertion axioms (statements) of the form:

C(a) – individual a is an instance of concept C

R(a, b) – individuals a and b are connected with the role R

Model and feasibility. Concept C is feasible if interpretation I exist, which is $C^I \neq \emptyset$. Such interpretation is called ***concept model***.

Interpretation I is the model of TBox T, if $C^I \subseteq D^I$ for each $C \sqsubseteq D$ from T.

Interpretation I is the model ABox A, if $a^I \in C^I$ for each $C(a) \in A$ and $(a^I, b^I) \in R^I$ for each $R(a, b) \in A$.
Concept C is feasible to the towards to TBox T if a common model for C and T exists.

DL temporal interpretation

There are many ways to use standard temporal logic in DL. The following way is chosen in this paper. Temporal operators are used as addition al concept constructors that allows to describe temporal behavior of individuals which belong to concepts. In addition, temporal operators can be applied to the roles, TBoxes, and ABoxes. All these options are united by a single notion of temporal DL interpretation.

DL temporal interpretation $I = (\Delta, \cdot^I)$ consists of non-empty domain Δ and $\cdot^I$ – interpretation function, which maps:

– each individual name $a \in N_I$ into element $a^I \in \Delta$;
– each concept name $A \in N_C$ into subset $A^I \subseteq \mathbb{N} \times \Delta$;
– each role name $R \in N_R$ into subset $R^I \subseteq \mathbb{N} \times \Delta \times \Delta$.

Here time is presented with natural numbers N. On these numbers the order is given < $(\mathbb{N}, <)$. For example, $(n, d) \in A^I$ means that individual d in the interpretation I is an element A in the n moment of time. Same for roles. Thus, concepts/roles (composition of concepts/roles) change over time, and this is the essence of their temporal interpretation. In turn, in the DL temporal interpretation defined above, the names of individuals do NOT change in time. That is, they are interpreted in the same way at all points in time. In this sense they are said to be ***rigid***.

Note, the DL temporal interpretation defined above is a special case of a first-order temporal structure without function symbols and an equality predicate and provided there are predicates no higher than binary (two-place).

There is an equivalent representation of temporal interpretation I in the case of endless sequence I(0), I(1),... of non-temporal interpretations, that are defined on the general domain Δ and with a fixed interpretation of individual names.

Standard domain assumption is a restriction on domain, which doesn`t allow domain to change in time. It means that a set of admissible individuals can`t change in time. There are alternative variants of the temporal interpretation, that includes a domain extension $\Delta^{I(0)} \subseteq \Delta^{I(1)} \subseteq \ldots$, domain narrowing $\Delta^{I(0)} \supseteq \Delta^{I(1)} \supseteq \ldots$ and just domain modifying.

Concept constructors in the DL temporal interpretation. Concept constructors have traditional interpretation in the DL temporal interpretation. They are interpreted with the help of the DL standard way in each moment of time and regardless of the other moments of time. Here are examples with the following preliminaries:

– X-axis means the time
– Y-axis means concept individuals
– table is the temporal concept interpretation
– red table cell means that individual belongs to the concept in the current moment of time

Temporal concept "Thing" ⊤. Graphical meaning of the temporal concept Thing is shown on figure 1.

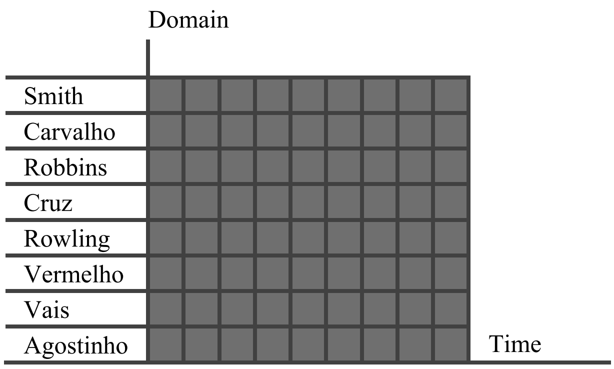

Figure 1. Temporal concept "Thing" ⊤

Temporal concept "Nothing" ⊥. Graphical meaning of the temporal concept Nothing is shown on figure 2.

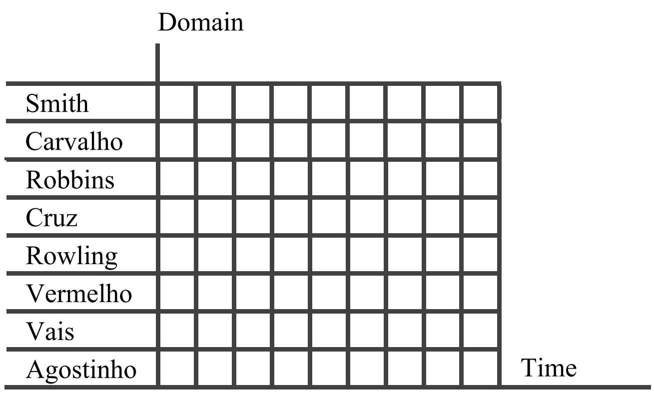

Figure 2. Temporal concept "Nothing" ⊥

Temporal concept "Department staff" (Concept C). Graphical meaning of the temporal concept "Department staff" (Concept C) is shown on figure 3.

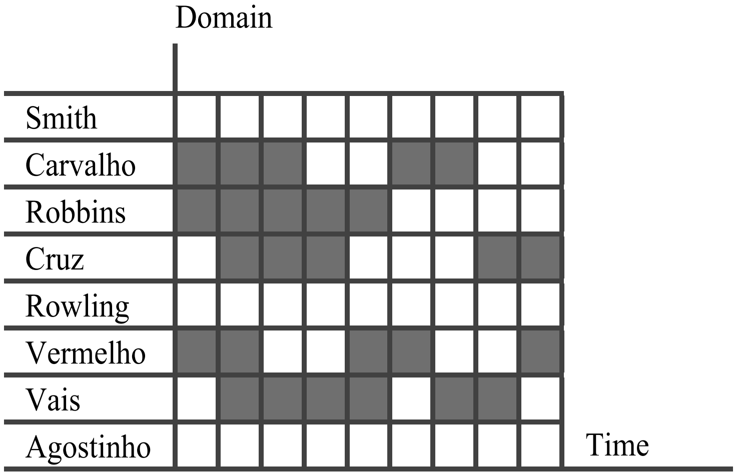

Figure 3. Temporal concept "Department staff" (Concept C)

Temporal concept "Articles publication" (Concept D). Graphical meaning of the temporal concept "Articles publication" (Concept D) is shown on figure 4.

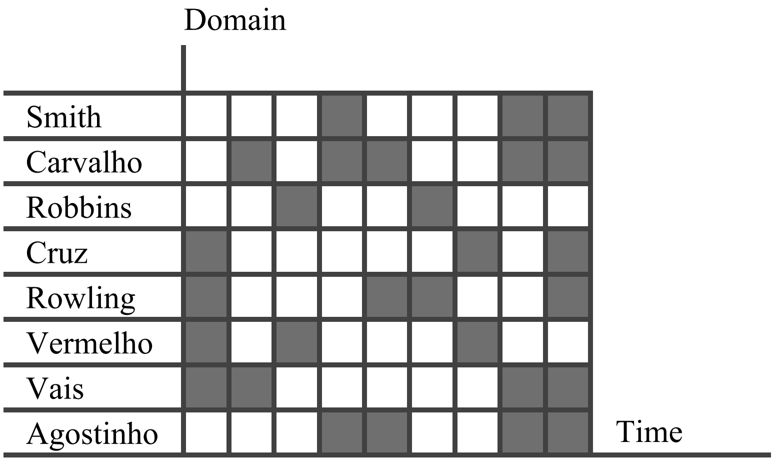

Figure 4. Temporal concept "Articles publication" (Concept D)

Temporal concept "Articles publication by department staff" (Concept C ⊓ D). Graphical meaning of the temporal concept "Articles publication by department staff" (Concept C ⊓ D) is shown on figure 5.

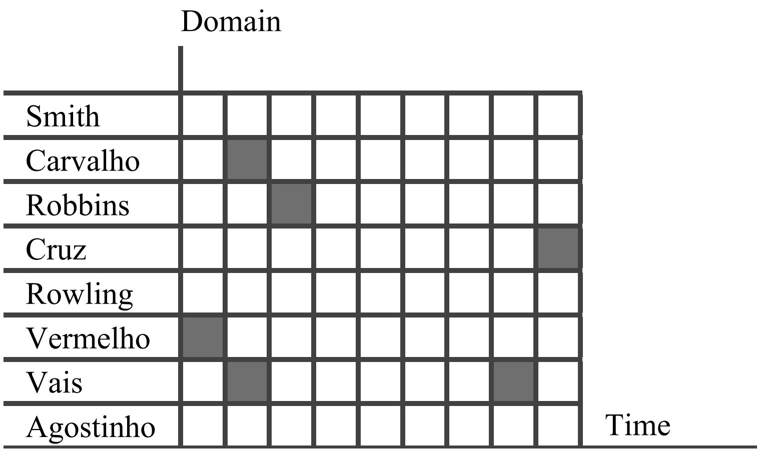

Figure 5. "Articles publication by department staff" (Concept C ⊓ D)

Temporal concept NOT "Department staff" (¬C). Graphical meaning of the temporal concept "Department staff" (Concept ¬C) is shown on figure 6.

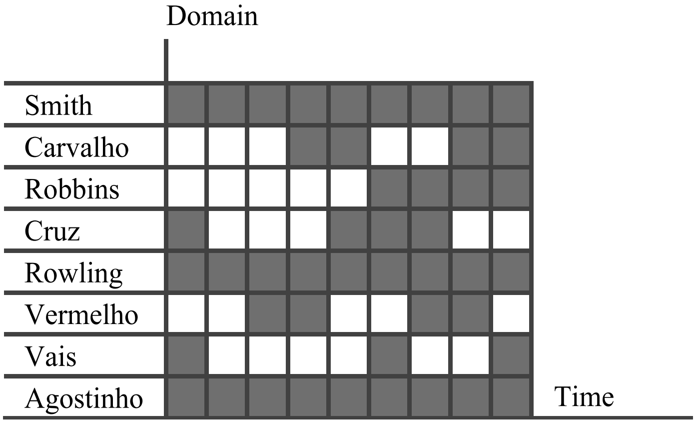

Figure 6. Temporal concept NOT "Department staff" (¬C)

Note. The time axis is infinite in LTL. We make the nest assumption to have a deal with the end axis in this application. On the time axis, there is always a point in time that has the following property. If the cell is red in the moment of time, then all remaining cells to the right up to infinity are red for every table line. If the cell is white in the moment of time, then all remaining are also white.

Temporal DL LTL_{ALC}

In previous section we described the meaning of the temporal DL interpretation and showed how the concept constructors are used. Concept constructors are operations over the columns of the tables (in examples, given in this paper). In turn, to use of LTL requires to the include the DL temporal operators (constructors). They are operations over the rows of the table.

The syntax for including temporal statements in DL is very simple. If concepts C and D (atomic or arbitrary) then ○C, ◊C, □C, C $\mathcal{U}$ D и C $\mathcal{R}$ D are also concepts. These operators have the following semantics:

$(\bigcirc C)^I = \{(n, d) \mid (n+1, d) \in C^I\}$ – next

$(\Diamond C)^I = \{(n, d) \mid \exists m \geq n\ (m, d) \in C^I\}$ – exist

$(\Box C)^I = \{(n, d) \mid \forall m \geq n\ (m, d) \in C^I\}$ – always

$(C\ \mathcal{U}\ D)^I = \{(n, d) \mid \exists m \geq n\ ((m, d) \in D^{\mathcal{I}} \wedge (k, d) \in C^{\mathcal{I}}$ для $n \leq k < m)\}$ – Until

$(C\ \mathcal{R}\ D)^I = (\neg(\neg C\ \mathcal{U}\ \neg D))^I$ – Release

Here are examples.

$(\bigcirc C)^I = \{(n, d) \mid (n+1, d) \in C^I\}$

○C means that "in the next moment of time C will take place". The following figure 7 shows five examples where a pair of rows means the separate example. Blue row is the meaning of C. Yellow row is the meaning of ○C. Red cell is the target. The first line is the current moment of time. The second line is the next moment of time. As can be seen from the examples, the operator ○ means shifting the red cells C to the left.

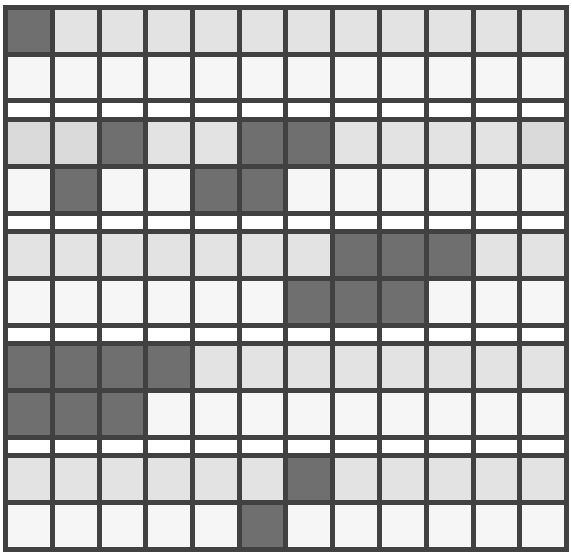

Figure 7. Examples of the operator ○C usage

The following two figures show the temporal meanings of the concepts. On the figure 8 is concept C, on the figure 9 is ◊C. The action of the operator ◊ can be described by the following. By each individual (row) the first red cell from the right side is found. All subsequent cells, up to the first, are converted to red. If there is no such cell, then the entire row remains white.

C^I

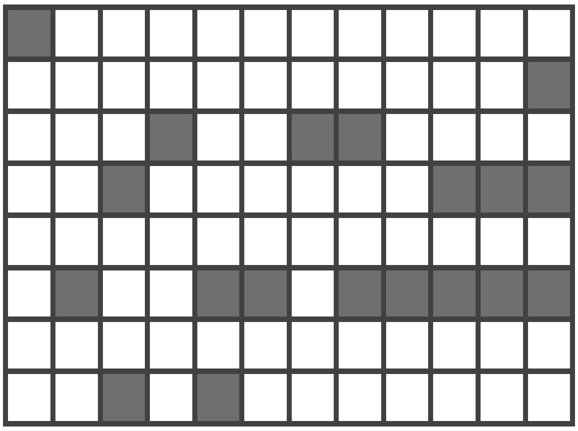

Figure 8. Temporal interpretation of concept C^I

$(\Diamond C)^I = \{(n, d) \mid \exists m \geq n\ (m, d) \in C^I\}$

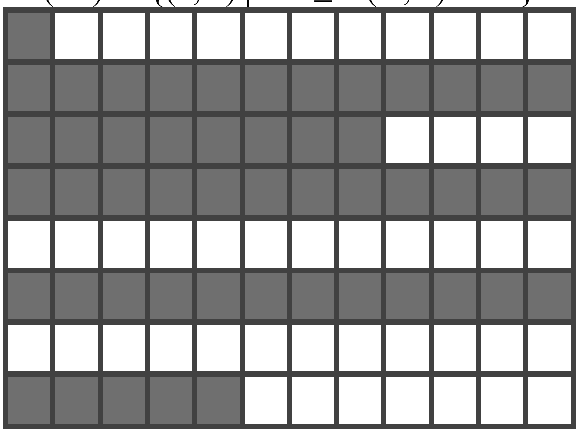

Figure 9. Temporal interpretation of concept $(\Diamond C)^I$

The following two figures show the temporal meanings of the concepts. On the figure 10 is concept C, on the figure 11 is □C. The action of the operator □ can be described by the following. By each individual (row) the first white cell from the right side is found. All subsequent cells, up to the first, are converted to white. If there is no such cell, then the entire row remains red.

C^I

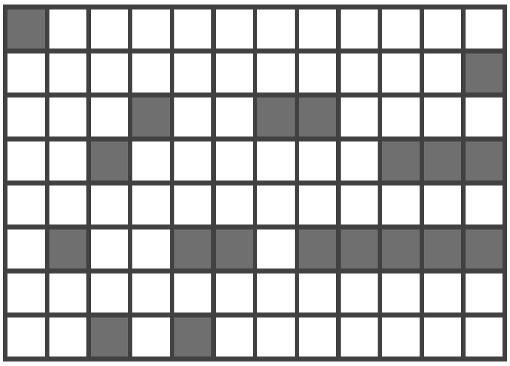

Figure 10. Temporal interpretation of concept C^I

$(\Box C)^I = \{(n, d) \mid \forall m \geq n\ (m, d) \in C^I\}$

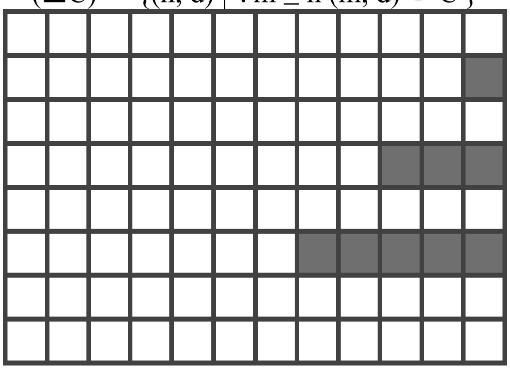

Figure 11. Temporal interpretation of concept $(\Box C)^I$

As can be seen from these examples, the operators □ and ◊ are dual, that is:

$\neg\Box C \equiv \Diamond\neg C$ $\Box C \equiv \neg\Diamond\neg C$

$\neg\Diamond C \equiv \Box\neg C$ $\Diamond C \equiv \neg\Box\neg C$

The following figure 12 shows an example of the $\mathcal{U}$ operator. The result of the operator's action is those red cells C that are obtained by executing the following procedure.

Each continuous range of red cells of C is taken. According to the range, the rightmost red cell of D is found. The chosen range of C is cut from the right side by the the rightmost red cell of D. If there is no red cell of D within the range of red cells of C, then this range is not included in the result $\mathcal{U}$.

$(C\ \mathcal{U}\ D)^{I =}\ \{(n, d) \mid \exists m \geq n\ ((m, d) \in D^{\mathcal{I}} \wedge (k, d) \in C^I$ для $n \leq k < m)\}$ – while

C

D

$C\ \mathcal{U}\ D$

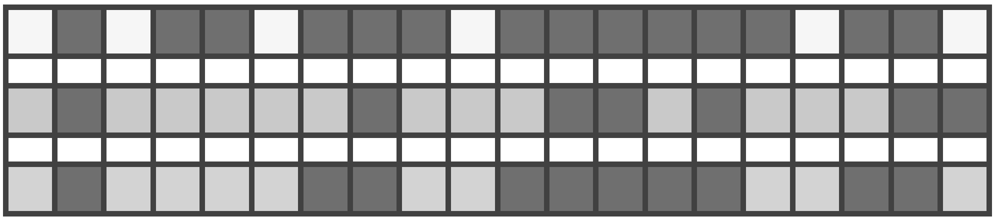

Figure 12. Temporal interpretation of concept $C\ \mathcal{U}\ D$

The following figure 13 shows an example of the $\mathcal{R}$ operator. The result of the operator's action includes the following cells: all the red cells of C and all that red cells of D which are adjacent to the left of the red ranges of C or overlap them.

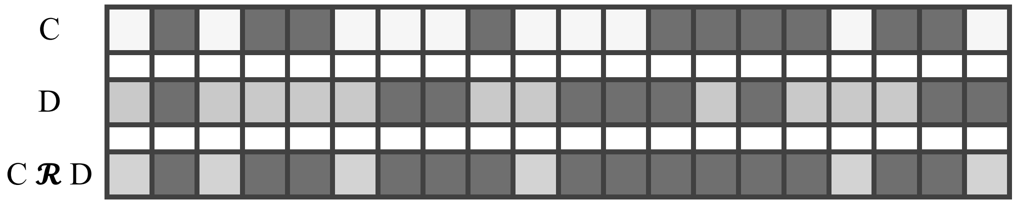

Figure 13. Temporal interpretation of concept C $\mathcal{R}$ D

Temporal constructor ∃R.C (∀R.C)

Role R is a binary relation. The first members (from the left side) of the role will be called "predecessors" (R-predecessors) in this paper. The second members (from the right side) of the role will be called "followers" (R-followers). In the general case, a role is a binary relation of the m:n type. That means many R-successors can correspond to each R-predecessor and vice versa. Concept constructor ∃R.C can build a concept from the role (R) and concept (C) in the following way. The result of the operation is such a set of R-predecessors for which ***at least*** one R-follower belongs to the concept C. In turn, the constructor ∀R.C defines such a set of R-predecessors, in which ***all*** R-successors belong to the concept C.

These constructors are dual. That means:

$\forall R.C \equiv \neg\exists R.\neg C$, $\exists R.C \equiv \neg\forall R.\neg C$

Figure 14 illustrates the essence of these constructors. The following image illustrates the meaning of the temporal role R. X-axis means role followers; Y-axis means role predecessors. Red cells of the table mean which R-predecessor and which R-follower are in the R relation.

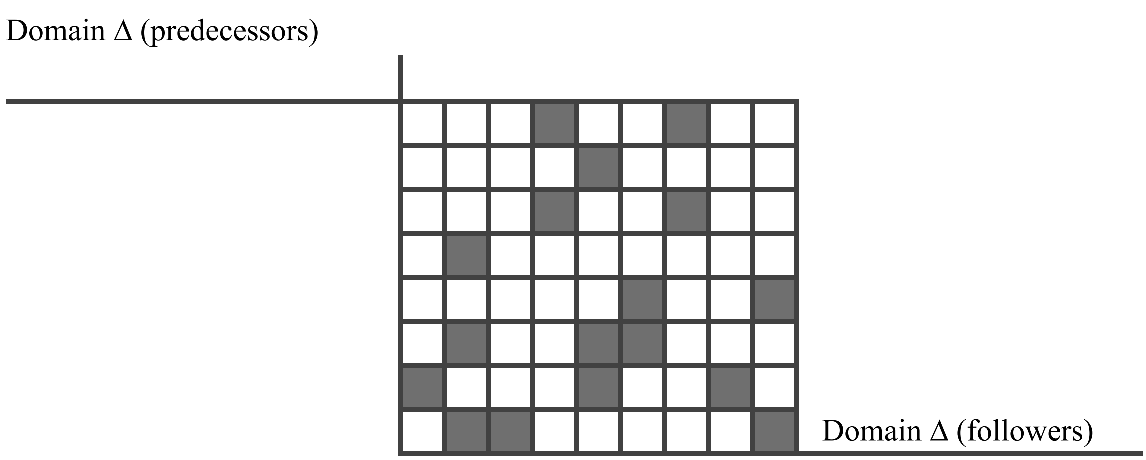

Figure 14. Temporal role R

Figure 15 illustrates the meaning of the ∀R.C and ∃R.C. The subject area is participation of department staff in projects. X-axis shows projects, Y-axis shows department staff. Yellow cells correspond to the NASU projects (concept C). Result of concept ∀R.C contains such rows, which contain all the red cells inside the yellow range. Result of concept ∃R.C contains such rows, which contain at least one red cell is in the yellow range.

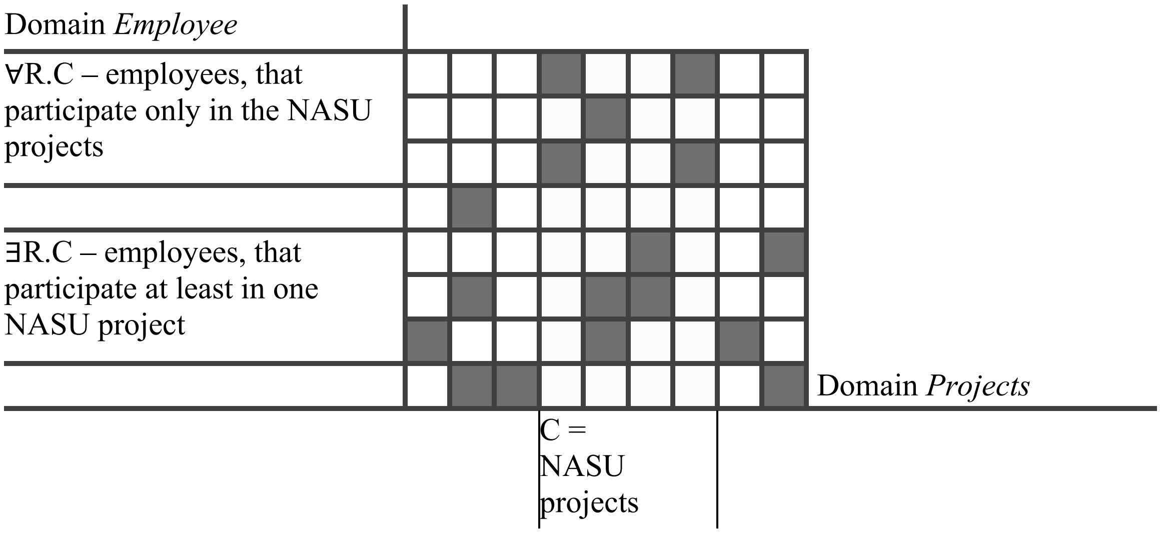

Figure 15. Temporal concepts ∀R.C and ∃R.C

The temporal interpretation of the constructor ∃R.C is as follows:

$(\exists R.C)^I = \{(n, d) \in \Delta \mid \exists(n, c) \in C^I \wedge (n, d, c) \in R^I\}$

Temporal roles

In terms of temporal interpretation and application of temporal operators role and concept have similar meaning. Temporal interpretation for concepts defines with the help of individuals. Temporal interpretation for roles defines with the help of pair of individuals. Formal interpretation of the role temporal interpretation:

$R^I \subseteq \mathbb{N} \times \Delta \times \Delta$

Here is an example. The subject area is the same (participation of department staff in projects). Role is participation of employees in projects. Domain of the role is the following set of pairs:

Smith, project1	Smith, project2
Cruz, project1	Cruz, project2
Robbins, project1	Robbins, project2

The temporal interpretation of the role looks like this (fig.16):

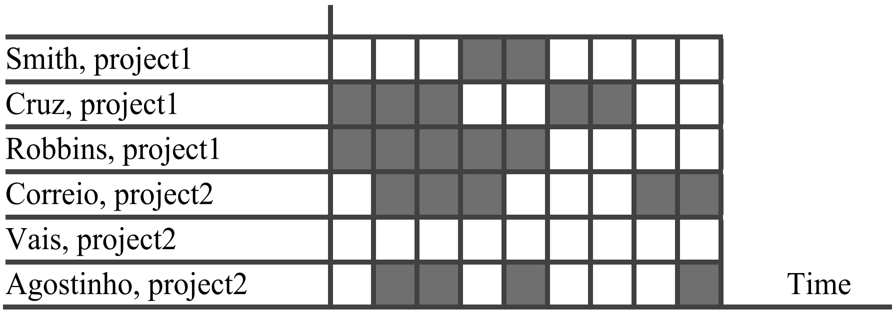

Figure 16. Temporal interpretation of role "Participation of the employee in projects" (domain Employee x Project)

Role constructors are absent in the DL ALC. Traditional temporal operators can be applied to the roles in TLT_{ALC}. Semantics is the following:

$(\bigcirc R)^I = \{(n, d, d') \mid (n+1, d, d') \in R^I\}$

$(\Diamond R)^I = \{(n, d, d') \mid \exists m \geq n\ (m, d, d') \in R^I\}$

$(\Box R)^I = \{(n, d) \mid \forall m \geq n\ (m, d, d') \in R^I\}$

$(R\ \mathcal{U}\ S)^I = \{(n, d, d') \mid \exists m \geq n\ ((m, d, d') \in S^I \wedge (k, d, d') \in R^I \text{ для } n \leq k < m)\}$

Figure 17–20 shows how to use these operators over the roles.

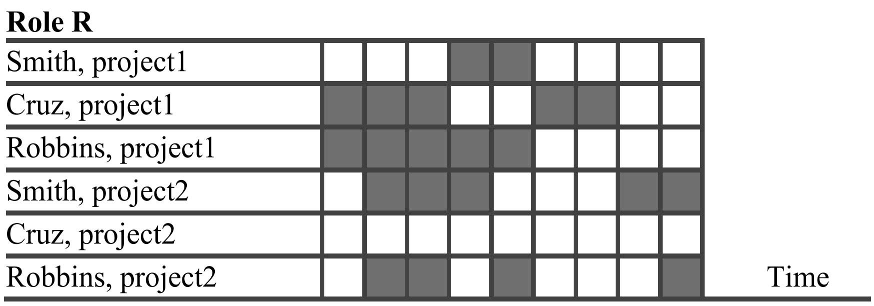

Figure 17. Temporal role R

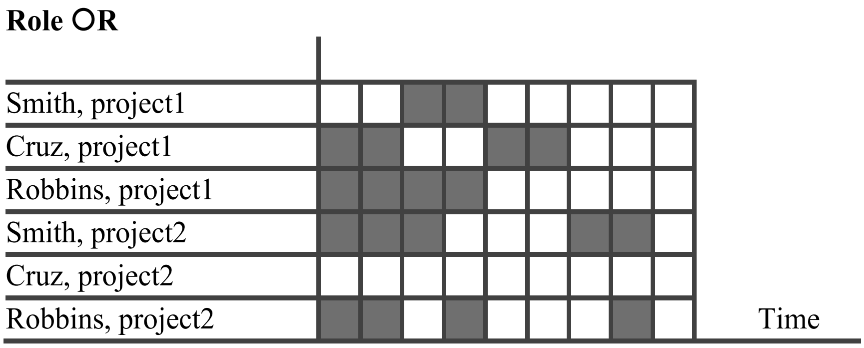

Figure 18. Temporal concept ○R

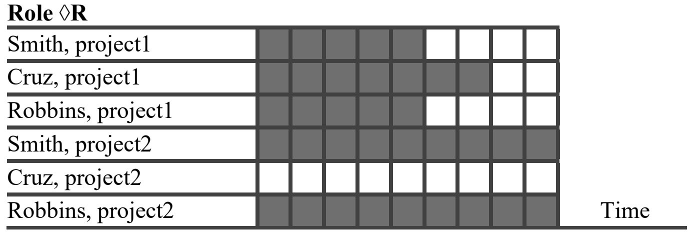

Figure 19. Temporal concept ◊R

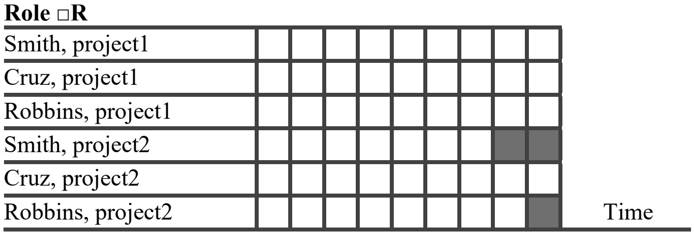

Figure 20. Temporal concept □R

Temporal TBox

Development over time of concepts and roles in TDL was shown in previous sections. But there are such temporal statements that can't be described with the help of temporal concepts and roles. They require terminological axiom (concept inclusion) usage. For example, *fixed* concept means, that concept doesn't change in time. It keeps the set of its individuals regardless of time. In other words, the fixed concept will be interpreted as a row with all the red cells or a row with all the white cells. Fixed concept can be shown with the following concept inclusion:

$C \sqsubseteq \Box C, \quad \neg C \sqsubseteq \Box \neg C$

TTBox definition requires to define TTBox formulas, which involve the use of temporal operators. There are two situations:

– temporal operators are applied only to the terminological axiom in general, but aren`t applied to their concepts. This situation is called temporal ALC TBox (ALC TBox).

– temporal operators are applied both to the terminological axiom and to their concepts (except roles). This situation is called temporal LTL_{ALC} TBox ($LTL_{\mathcal{ALC}}$ TBox).

Temporal ALC TBox.

Syntactic rules for constructing formulas ALC TBox:

$C \sqsubseteq D, \quad \neg\varphi, \qquad \varphi \wedge \psi, \quad \bigcirc\varphi, \qquad \varphi\,\mathcal{U}\,\psi$

where:

– C and D – atomic concepts with temporal interpretation

– φ and ψ – temporal TBox

Semantics of these formulas defines as follows:

$I, n \vDash C \sqsubseteq D \Leftrightarrow \{d \mid (n, d) \in C^I\} \subseteq \{d \mid (n,d) \in D^I\}$

$I, n \vDash \neg\varphi \qquad \Leftrightarrow I, n \nvDash \varphi$

$I, n \vDash \varphi \wedge \psi \Leftrightarrow I, n \vDash \varphi \text{ and } I, n \vDash \psi$

$I, n \vDash \bigcirc\varphi \Leftrightarrow I, n+1 \vDash \varphi$

$I, n \vDash \varphi\,\mathcal{U}\,\psi \Leftrightarrow \exists m \geq n \{I, m \vDash \psi \text{ and } \forall n \leq k < m\ I, k \vDash \varphi\}$

This record $I, n \vDash \varphi$ means: φ is true in the n moment of time in interpretation I. Definition if the terminological axiom is true or false depends on each moment of time (locally). It is not a general definition (globally). Here are graphical examples of temporal ALC TBox and the result of temporal axioms. Figures 21–23 demonstrate temporal concepts "Institute staff" (C), "Department staff" (D) and "Project staff" (E).

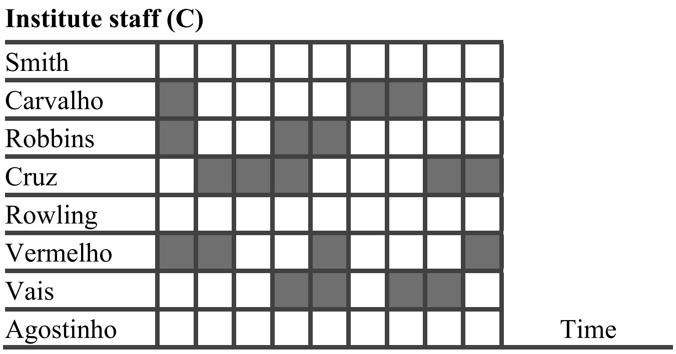

Figure 21. Temporal concept "Institute staff" (C)

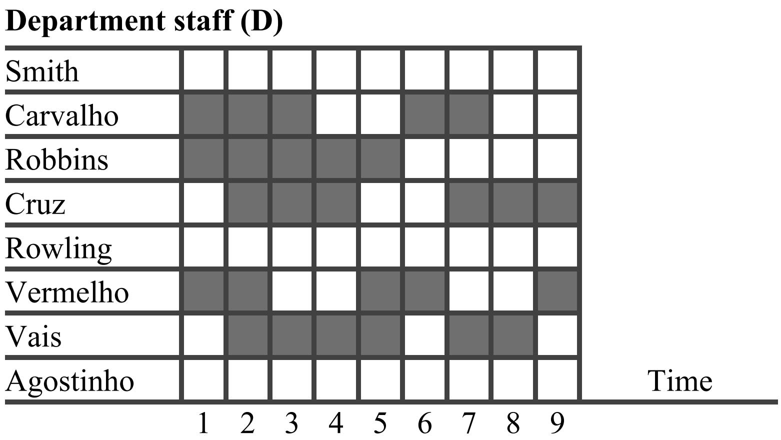

Figure 22. Temporal concept “Department staff” (D)

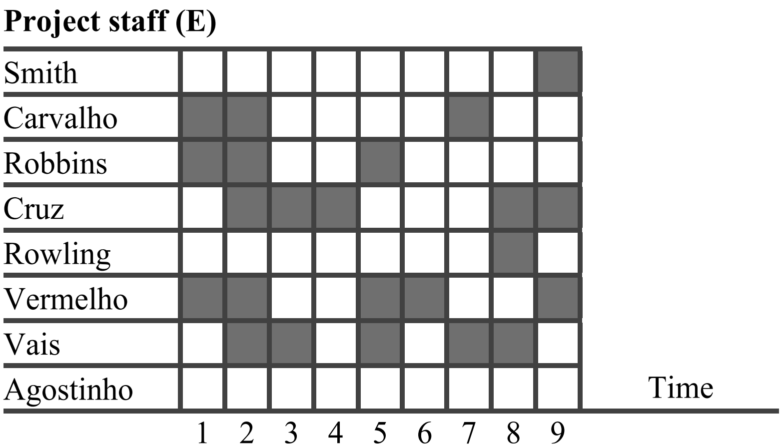

Figure 23. Temporal concept “Project staff” (E)

Here are examples of the result of the different terminological axioms. Result contains its truth values at the corresponding time. In these examples result is shown as a concept, that contains single individual, that belongs to this concept in the corresponding moment of time, when the terminological axiom is true.

Axiom C ⊑ D is true for all the moments of time. It means that all the department staff employees are institute employees.

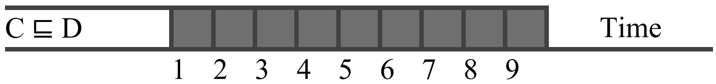

Figure 24. Temporal axiom C ⊑ D

Axiom D ⊑ C is true for 1,5,8,9 moments of time. So, department will include all the institute staff on that moment of time.

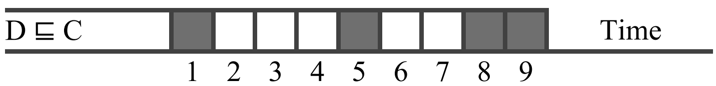

Figure 25. Temporal axiom D ⊑ C

Axiom E ⊑ D is true for 1–7 moments of time. So, only institute staff will work on a project on that moment of time.

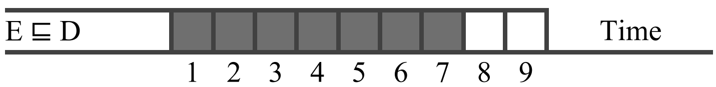

Figure 26. Temporal axiom E ⊑ D

Axiom D ⊑ E is true for 1, 8, 9 moments of time. In the moment 1 axiom E ⊑ D is true. So, at this moment only institute employees will work on project and no one else. As for 8 and 9 moments of time, all the institute staff and someone else will work on project.

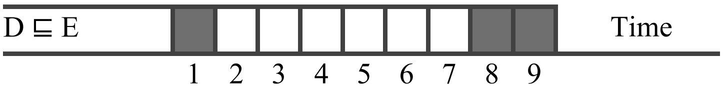

Figure 27. Temporal axiom $D \sqsubseteq E$

Axiom $C \sqsubseteq E$ is true for 1, 2, 3, 5, 7, 8, 9 moments of time. So, only department staff will work on project at these moments of time.

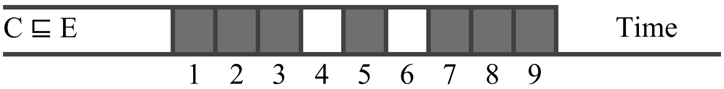

Figure 28. Temporal axiom $C \sqsubseteq E$

Axiom $E \sqsubseteq C$ is true for 1, 4, 5, 7 moments of time. In the moments of time 1, 5, 7 $C \sqsubseteq E$ is true. So, at these moments of time only department staff will work on project and no one else. At the 4th moment of time only department employees, but not all of them will work on the project.

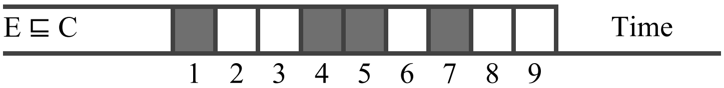

Figure 29. Temporal axiom $E \sqsubseteq C$

Axiom $E \sqsubseteq D \wedge D \sqsubseteq E$ is true for 1 moment of time. In this moment of time all the institute, but no one else will work on project.

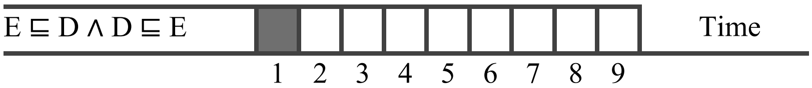

Figure 30. Temporal axiom $E \sqsubseteq D \wedge D \sqsubseteq E$

Axiom $C \sqsubseteq E \wedge E \sqsubseteq D$ is true for the 1–7 moments of time. So, at these moments of time only employees that are both department employees and institute employees will work on project.

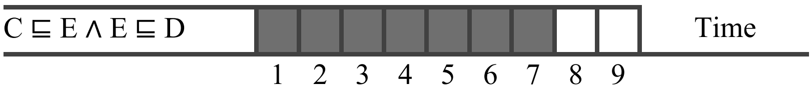

Figure 31. Temporal axiom $C \sqsubseteq E \wedge E \sqsubseteq D$

Axiom $\Diamond(E \sqsubseteq C)$ is true for the 1–7 moments of time (as terminological axiom $E \sqsubseteq C$ is true in the 1, 4, 5, 7 moments of time).

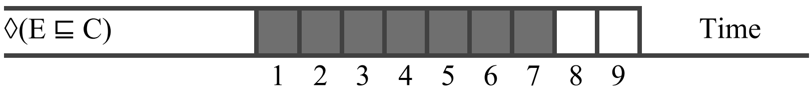

Figure 32. Temporal axiom $\Diamond(E \sqsubseteq C)$

Axiom $\Box(C \sqsubseteq E)$ is true for the 7, 8, 9 moments of time (because $C \sqsubseteq E$ is true for the 1, 2, 3, 5, 7, 8, 9 moments of time).

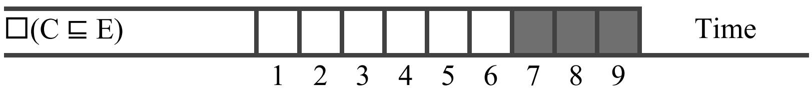

Figure 33. Temporal axiom $\Box(C \sqsubseteq E)$

Axiom $\Diamond\Box(C \sqsubseteq E)$ is true for the 1–9 moments of time.

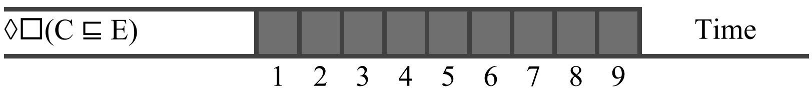

Figure 34. Temporal axiom $\Diamond\Box(C \sqsubseteq E)$

Axiom $\bigcirc\Box(C \sqsubseteq E)$ is true for the 6, 7, 8 moments of time.

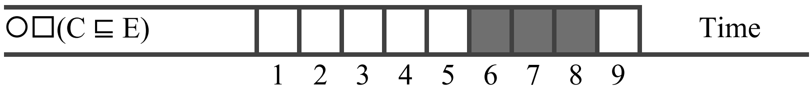

Figure 35. Temporal axiom $\bigcirc\Box(C \sqsubseteq E)$

Temporal LTL_{ALC} TBox.

Expressive possibilities obtained when the temporal operators are applied to concepts are incomparable (mismatched) in comparison when temporal operators are applied to the terminological axioms. Here is an example in non-temporal LTL_{ALC} TBox:

independent_country $\sqsubseteq$ $\Box$independent_country

In this example a statement is approved. that the extension (i.e. temporal scope) of the concept of "independent_country" cannot decrease, but this is not expressible in the temporal ALC TBox. From the other side the statement "eventually all European countries will become members of the EU forever" is expressed with the following temporal ALC TBox:

$\Diamond$(european_country $\sqsubseteq$ member_EU)

But this example is not expressible in the non-temporal LTL_{ALC} TBox.

Temporal operators are applied only to the concepts and terminological axioms. Interaction between TL and DL components in TDL is severely limited. In its turn it can be expressed quite developed types of relationships in the temporal LTL_{ALC} TBox. For example:

$\bigcirc\neg(\top \sqsubseteq A) \Leftrightarrow \neg(\top \sqsubseteq \bigcirc A)$

Temporal ABox

To define temporal ABox two options should be considered:

– ABox that has temporal interpretation. It is called non-temporal ABox.

– ABox that has temporal interpretation and temporal operators. It is called temporal ABox.

Usual ABox contains the following types of predicates: C(a) and R(A, b). There are no constructors for them as for concepts. What about temporal interpretation, its semantics defines as follows:

$I, n \vDash C(a) \Leftrightarrow (n, a^I) \in C^I$

$I, n \vDash R(a, b) \Leftrightarrow (n, a^I, b^I) \in R^I$

To define if ABox is true determines for each moment of time (locally), but not in general (globally).

In the case of temporal ABox it is supposed to use such temporal operators as in the TBox:

$C(a), R(a,b), \neg\varphi, \varphi \wedge \psi, \bigcirc\varphi, \varphi\mathcal{U}\psi$ where:

– C(A) and R(a, b) – atomic statements ABox with temporal interpretation

– φ and ψ – temporal ABox

Semantics of these formulas defines as follows:

$I, n \vDash \neg\varphi \Leftrightarrow I,n \nvDash \varphi$

$I, n \vDash \varphi \wedge \psi \Leftrightarrow I,n \vDash \varphi$ and $I,n \vDash \psi$

$I, n \vDash \bigcirc\varphi \Leftrightarrow I, n+1 \vDash \varphi$

$I, n \vDash \varphi\mathcal{U}\psi \Leftrightarrow \exists m \geq n \{I,m \vDash \psi$ and $\forall n \leq k < m\ I,k \vDash \varphi\}$

Here are several examples:

– C(Smith) – becomes true in the moments 1, 7, 8.

– C(Cruz) – becomes true in the moments 1, 4, 5.

– E(Smith) – becomes true in the moments 1, 2, 7.

– E(Robbins) – becomes true in the moments 2, 3, 4, 8, 9.

– C(Smith) $\wedge$ C(Cruz) – becomes true in the moment 1. This is the moment when these two employees work at the same time.

– C(Smith) $\wedge$ E(Cruz) – becomes true in the moment 8. This is the moment when Smith was a department employee and Cruz worked on the project.

– C(Smith) $\wedge$ E(Smith) – becomes true in the moment 1 and 7. These are moments, when Smith was a department employee and worked on project.

– C(Smith) $\wedge$ ¬E(Smith) – becomes true in the moment 8. These are moments when Smith was a department employee and didn't work on project.

– $\Diamond$C(Smith) – becomes true in the moment 1–5.

– E(Robbins) – becomes true in the moments 8, 9.

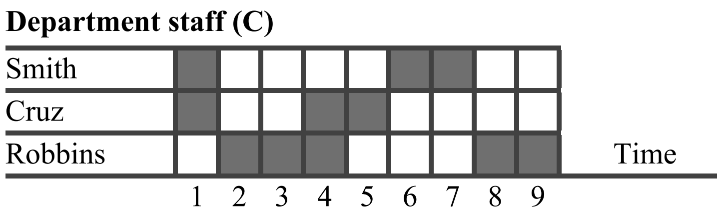

Figure 36. Temporal concept "Department staff" (C)

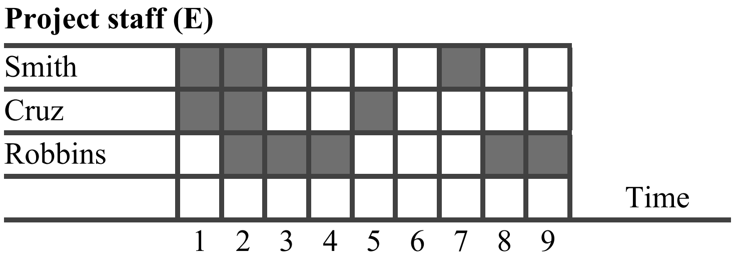

Figure 37. temporal concept "Project staff" (E)

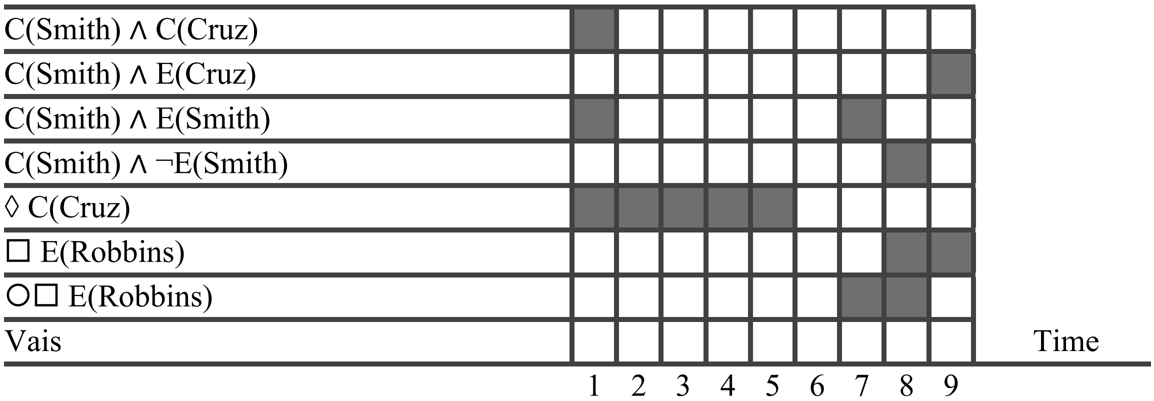

Figure 38. Examples of temporal ABox

Conclusions

The algorithm for applying LTL logic to the DL ALC briefly has shown in the paper with the help of numerous examples. There are following questions for the future research:

- algorithm for applying DL to the temporal logic CTL.
- DL family logic extension which can be used for temporality applying.
- temporal query languages syntax and semantics.

References

1. CAU A., MOSZKOWSKI B. (2021) *Interval Temporal Logic* [Online] Available from: http://antonio-cau.co.uk/ITL/ [Accessed: 21 May 2022]
2. LUTZ C., WOLTER F., AND ZAKHARYASCHEV M. (2008) Temporal description logics: A survey. *In 15th International Symposium on Temporal Representation and Reasoning* IEEE Computer Society Press, pages 3–14
3. DONZÉ, A. (2013). On Signal Temporal Logic. RV 2013. [Lecture notes] Legay, A., Bensalem, S. (eds) Runtime Verification. Vol 8174. Springer, Berlin, Heidelberg.
4. MEHRABIAN M., KHAYATIAN M., SHRIVASTAVA A., EIDSON J.C., DERLER P., ANDRADE H.A., LI-BABOUD Y.-S., GRIFFOR E., WEISS M., STANTON K. (2017) Timestamp Temporal Logic (TTL) for Testing the Timing of Cyber-Physical Systems. *ACM Transactions on Embedded Computer Syst*ems 16 p. 169:1 – 169:20
5. EISNER C., FISMAN D. (2006) *A Practical Introduction to PSL* [Online] Available from: https://link.springer.com/book/10.1007/978-0-387-36123-9 [Accessed: 21 May 2022]
6. WANWEI LIU (2017) *Basic Temporal Logic – LTL, CTL, CTL** [Online] Available from: https://data.educoder.net/api/attachments/667?disposition=inline [Accessed: 1 June 2022]
7. FERLEZ J., CLEAVELAND R., MARCUS S. (2017) *Bisimulation and Hennessy-Milner Logic for Generalized Synchronization Trees∗* EPTCS 255 pp. 35-50
8. CS INTRANET (2017) *Metric Temporal Logic: Tools and Experiments* [Online] Available from: https://cgi.csc.liv.ac.uk/~ullrich/MTL/ [Accessed: 15 June 2022]
9. BOUYER P., LAROUSSINIE F., MARKEY N., OUAKNINE J., WORREL J. (2017) *Timed temporal logics* [Online] Available from: https://hal.archives-ouvertes.fr/hal-01566439/document [Accessed: 1 April 2022]
10. EISNER C., FISMAN D., HAVLICEK J., LUSTIG Y., MCISAAC A., CAMPENHOUT D.V. (2017) *Reasoning with Temporal Logic on Truncated Paths* [Online] Available from: https://www.cs.bgu.ac.il/~dana/documents/EFHLMV_CAV03_full.pdf [Accessed: 20 May 2022]

Література

1. CAU A., MOSZKOWSKI B. (2021) Interval Temporal Logic [Online] Available from: http://antonio-cau.co.uk/ITL/ [Accessed: 21 May 2022]
2. LUTZ C., WOLTER F., AND ZAKHARYASCHEV M. (2008) Temporal description logics: A survey. In 15th International Symposium on Temporal Representation and Reasoning IEEE Computer Society Press, pages 3–14
3. DONZÉ, A. (2013). On Signal Temporal Logic. RV 2013. [Lecture notes] Legay, A., Bensalem, S. (eds) Runtime Verification. Vol 8174. Springer, Berlin, Heidelberg.
4. MEHRABIAN M., KHAYATIAN M., SHRIVASTAVA A., EIDSON J.C., DERLER P., ANDRADE H.A., LI-BABOUD Y.-S., GRIFFOR E., WEISS M., STANTON K. (2017) Timestamp Temporal Logic (TTL) for Testing the Timing of Cyber-Physical Systems. ACM Transactions on Embedded Computer Systems 16 p. 169:1 – 169:20
5. EISNER C., FISMAN D. (2006) A Practical Introduction to PSL [Online] Available from: https://link.springer.com/book/10.1007/978-0-387-36123-9 [Accessed: 21 May 2022]

6. WANWEI LIU (2017) Basic Temporal Logic – LTL, CTL, CTL* [Online] Available from: https://data.educoder.net/api/attachments/667?disposition=inline [Accessed: 1 June 2022]
7. FERLEZ J., CLEAVELAND R., MARCUS S. (2017) Bisimulation and Hennessy-Milner Logic for Generalized Synchronization Trees∗ EPTCS 255 pp. 35-50
8. CS INTRANET (2017) Metric Temporal Logic: Tools and Experiments [Online] Available from: https://cgi.csc.liv.ac.uk/~ullrich/MTL/ [Accessed: 15 June 2022]
9. BOUYER P., LAROUSSINIE F., MARKEY N., OUAKNINE J., WORREL J. (2017) Timed temporal logics [Online] Available from: https://hal.archives-ouvertes.fr/hal-01566439/document [Accessed: 1 April 2022]
10. EISNER C., FISMAN D., HAVLICEK J., LUSTIG Y., MCISAAC A., CAMPENHOUT D.V. (2017) Reasoning with Temporal Logic on Truncated Paths [Online] Available from: https://www.cs.bgu.ac.il/~dana/documents/EFHLMV_CAV03_full.pdf [Accessed: 20 May 2022]

Received 22.08.2022

About authors:

Reznichenko Valerii Anatoliiovych,
PhD physics and mathematics,
Leading researcher at the Institute of software systems of NASU.
The number of publications in Ukrainian journals is 62.
The number of publications in foreign journals is 3.
Hirsh index is 12. ORCID: https://orcid.org/0000-0002-4451-8931.

Chystiakova Inna Serhiivna,
PhD technical science,
Researcher at the Institute of software systems of NASU.
The number of publications in Ukrainian journals is 14.
The number of publications in foreign journals is 1.
Hirsh index is 6. ORCID: https://orcid.org/0000-0001-7946-3611.

Place of work:

Institute of Software Systems NAS of Ukraine,
03187, Kyiv-187,
Academician Glushkov Avenue, 40, build 5.
Phone: +38(050) 368 49 27.

Прізвища та ініціали авторів і назва доповіді українською мовою:
Резніченко В.А., Чистякова І.С.
Таблична інтерпретація темпоральної дескриптивної логіки LTL_{ALC}

Прізвища та ініціали авторів і назва доповіді англійською мовою:
Reznichenko V.A., Chystiakova I.S.
Table interpretation of the temporal description logic LTL_{ALC}