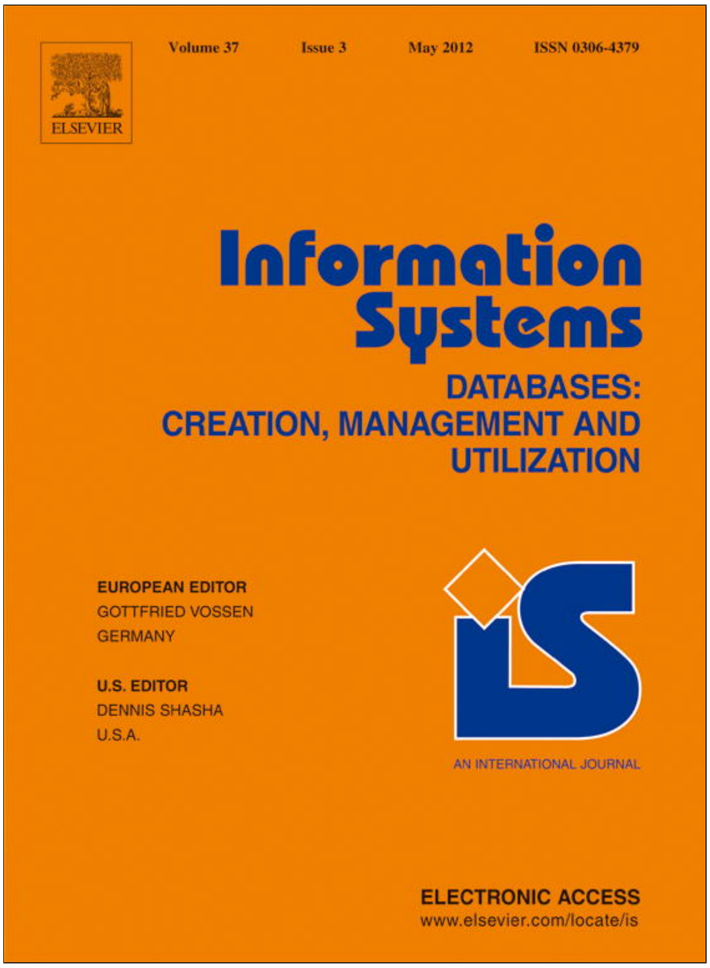
Volume 37
Issue 3
May 2012
ISSN 0306-4379
ELSEVIER
Information Systems
DATABASES: CREATION, MANAGEMENT AND UTILIZATION
EUROPEAN EDITOR
GOTTFRIED VOSSEN
GERMANY
U.S. EDITOR
DENNIS SHASHA
U.S.A.
AN INTERNATIONAL JOURNAL
ELECTRONIC ACCESS
www.elsevier.com/locate/is

Information Systems 37 (2012) 238–256

Contents lists available at SciVerse ScienceDirect

Information Systems

journal homepage: www.elsevier.com/locate/infosys

A model of fuzzy spatio-temporal knowledge representation and reasoning based on high-level Petri nets

Slobodan Ribaric, Tomislav Hrkac *

University of Zagreb, Faculty of Electrical Engineering and Computing, Croatia

ARTICLE INFO

Article history:
Received 11 May 2011
Accepted 20 September 2011
Recommended by: D. Shasha
Available online 17 November 2011

Keywords:
Knowledge representation
Fuzziness
Spatio-temporal reasoning
Multi-agent systems
High-level Petri nets

ABSTRACT

In many application areas there is a need to represent human-like knowledge related to spatio-temporal relations among multiple moving objects. This type of knowledge is usually imprecise, vague and fuzzy, while the reasoning about spatio-temporal relations is intuitive. In this paper we present a model of fuzzy spatio-temporal knowledge representation and reasoning based on high-level Petri nets. The model should be suitable for the design of a knowledge base for real-time, multi-agent-based intelligent systems that include expert or user human-like knowledge. The central part of the model is the knowledge representation scheme called FuSpaT, which supports the representation and reasoning for domains that include imprecise and fuzzy spatial, temporal and spatio-temporal relationships. The scheme is based on the high-level Petri nets called Petri nets with fuzzy spatio-temporal tokens (PeNeFuST). The FuSpaT scheme integrates the theory of the PeNeFuST and 117 spatio-temporal relations.

The reasoning in the proposed model is a spatio-temporal data-driven process based on the dynamical properties of the scheme, i.e., the execution of the Petri nets with fuzzy spatio-temporal tokens. An illustrative example of the spatio-temporal reasoning for two agents in a simplified robot-soccer scene is given.

1. Introduction

In the past 10 years, one of the central problems in intelligent system design has been the development of appropriate knowledge-representation schemes that support spatio-temporal representation and reasoning [1–5]. These schemes have been used in different application areas such as computer vision and robot navigation [6–14], multimedia [15–18], geographical information systems (GIS) [2,19–23], natural language processing and engineering design [24], etc.

Recently, in many application domains there is a need for the development of knowledge-representation schemes that support the human-like knowledge representation of spatial, temporal and spatio-temporal information and human ways of reasoning. Most human knowledge, however, is typically expressed in vague and imprecisely defined concepts and the inference is mostly supported by common-sense and intuitive reasoning. One of the approaches to enable the representation and handling of such a type of knowledge is to introduce the concept of fuzziness [25]. Although some successful formalisms have been proposed for the separate representation of fuzzy temporal [26] or fuzzy spatial [27] data, relatively little work has been done in the field of integrated fuzzy spatio-temporal knowledge representation and reasoning.

The motivation for our research was the development of a model that allows the human-like representation of spatial, temporal and spatio-temporal information and reasoning that is suitable for the knowledge-base design used in computer-vision and robot-navigation intelligent systems based on the concepts of a multi-agent system (MAS) [28–30].

* Corresponding author. Tel.: +385 16129926.
E-mail addresses: slobodan.ribaric@fer.hr (S. Ribaric), tomislav.hrkac@fer.hr (T. Hrkac).

0306-4379/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.is.2011.09.010

The main goals that have to be achieved are as follows:

(i) The model has to support the design of a knowledge base that includes fuzzy and imprecise temporal, spatial and spatio-temporal relations among moving agents and/or objects. The fuzziness and imprecision related to temporal, spatial and spatio-temporal relationships have to be expressed in a form that is appropriate for human experts and users.
(ii) The model has to be appropriate for multi-agent-based systems in such a way that it enables the independent design of a knowledge base for each of the agents, and the modeling of the interactions among them.
(iii) The model has to allow a hierarchical representation and modeling of the system at different abstraction levels.
(iv) The model should be based on a well-defined formalism that allows a formal analysis of different spatial, temporal and spatio-temporal relationships among the objects (according to Ferber [28], the agents are specific objects, representing the active entities in the system), by changing the initial conditions, temporal, spatial, spatio-temporal relationships or the final goals of a modeled system.
(v) The aim of the model is the development of a knowledge base for real-time applications, meaning that the model has to support an efficacious short-time-consuming reasoning process.
(vi) The proposed model has to be suitable for the design of a program simulator based on an object-oriented programming environment.
(vii) For small or moderate-sized modeled systems the model should offer a graphical representation of the knowledge base for each of the agents.

In this paper we propose a model, the main component of which is an original fuzzy spatio-temporal knowledge representation and reasoning scheme called FuSpaT, which fulfils the above-mentioned main goals. The proposed scheme is based on an original high-level Petri net, called the Petri net with fuzzy spatio-temporal tokens (PeNeFuST). The model also includes an object-oriented simulator that contains tools for the analysis of a modeled system.

The rest of the paper is organized as follows. The related work concerning the proposed scheme is introduced in Section 2. In Section 3, we briefly describe the theoretical basis of the high-level Petri net with fuzzy spatio-temporal tokens. Section 4 presents the fuzzy spatio-temporal knowledge representation scheme called FuSpaT. In Section 5, an example of using the FuSpaT for modeling the details of a robot-soccer scene is given. The conclusion and future work are discussed in Section 6.

2. Related work

Spatio-temporal formalisms have been previously discussed in the literature. The related works concerning the proposed spatio-temporal knowledge-representation scheme can be in general divided as follows:

(i) crisp-like spatio-temporal representation models:
  - approaches that temporalize the models that are based on a spatial formalism [31,32], or vice versa [1];
  - approaches based on a description of possible changes of positions and relative orientations of the objects [4,5,24,33,34];
  - Petri net-based models [17,35–40];
  - a hybrid approach that takes into account elements from temporal logic and elements from point-set theory and point-set topology [3];

(ii) fuzzy spatio-temporal models based on the following approaches:
  - fuzzy set theory and fuzzy spatio-temporal relationships [41];
  - linguistic descriptions of the moving objects [42];
  - fuzzy-rule-based reasoning [43–45].

The brief descriptions of the related works follow. Bennett et al. [31] introduced a temporalization of the topological RCC8 calculus. Ragni and Woelfl [32] investigated a temporalization of the cardinal directions [46] in order to define a method for encoding temporized spatial constraint satisfaction problems as deterministic planning problems.

Hornsby and Egenhofer presented an approach to spatio-temporal knowledge representation based on a description of possible changes of real-world phenomena, called identifiable objects, modeled at a high level of abstraction [2]. The foundation of the model is a set of primitives and the operations that can be performed on them. These primitives are the *identity states of objects* and *transitions*. The term *object* refers to the representation of a real-world phenomenon in an information system. *Identity states* are associated with objects, capturing the notion that although an object's identity is enduring, the state of the identity may change, e.g., from existing to non-existing. The objects and their associated identities are linked through another primitive, the *transition*. The progression of an object from one state of identity to another is modeled by the transitions. The authors proposed an iconic visual language called *change description language* (CDL) to describe the changes to the identity states of objects. Although no explicitly spatial information has been incorporated into this model of change, it has been shown that tracking the changes to an object's identity over periods of existence and non-existence gives useful insights into the behavior of an object over time that are relevant to many cases of spatio-temporal change. The proposed model was used in GIS applications.

Erwig and Schneider [3] described a more explicit framework for the representation of spatio-temporal data by means of so-called *spatio-temporal predicates*. This framework is based on a hybrid approach that takes into account elements from temporal logic and elements from point-set theory and point-set topology. Presupposing a continuous model of time, they employ *temporal functions* as the basis of an algebraic model for three basic *spatio-temporal data types*: *moving point*, *evolving line* and *evolving region*. Then, starting with eight basic topological predicates of the so-called nine-intersection model (which has been shown to be equivalent to the RCC-8 variant of the region connection calculus) and by using the concepts of *temporal lifting* and *temporal aggregation*, they define eight basic spatio-temporal predicates as temporally lifted spatial (topological) predicates with a certain preferred or default temporal aggregation. The relationships between two spatio-temporal objects can then be appropriately modeled by scenarios described as sequences of spatial and basic spatio-temporal predicates. The authors also define an algebra of spatio-temporal predicates and several inference rules.

Wolter and Zakharyaschev [1] proposed a family of decidable spatio-temporal formalisms, based on a combination of the region connection calculus (RCC) as a topological model of space with several temporal formalisms, including a linear temporal model based on time points, a branching model of time and Allen's temporal interval logic [47].

All of the above-described works are based on a purely topological view of space and do not take into account the relative positions of the objects with respect to each other, which can be a drawback when representing some common scenarios. In a series of papers [4,5,33,34], Van de Weghe et al. proposed a *qualitative trajectory calculus* (*QTC*) as a language for representation and reasoning about the movements of point objects in a qualitative framework, able to differentiate between groups of disconnected objects. Several variations of QTC were developed. The simpler QTC-basic (QTC_B) considers only the change in the distance between the objects, while the more general QTC-double-cross (QTC_C) also takes into account the relative orientation of the object movements in two dimensions. The relative movement of the two objects can then be represented by a four-component label, where the first two components describe the tendency of the change of the distance of an object with respect to the current position of another object, while the other two components describe the relative orientation of the object movements with respect to the reference line that connects them. Each of the components can assume one of the three possible qualitative values ($+,-$, or 0), resulting in $3^4=81$ possible QTC_C relationships. More complex scenarios can be modeled by means of the so-called *conceptual animations*, i.e., sequences of the basic QTC_C relationships associated with the corresponding time points or time intervals. Two reasoning formalisms are proposed: a formalism based on composition tables and a formalism based on conceptual neighborhood diagrams.

Several Petri net-based spatio-temporal representation formalisms have also been proposed. A *spatial and temporal relationship Petri net* (*STRPN*) [17] is a Petri net-based knowledge representation scheme able to describe the spatio-temporal relationships of moving multimedia objects that may refer to each other for synchronization and computing spatial display addresses. A STRPN is, in fact, an extension of the object composition Petri net (OCPN), which is itself an extension of the Petri net [39] to specify the temporal relationships of multimedia objects. To the temporal representation capabilities of an OCPN, a STRPN adds the capability to describe the spatial relationships between objects. The objects are represented by their minimum bounding boxes, and their relative spatial relationships are represented by 4-tuples, that can express 169 different spatial relationships. A STRPN extends the basic Petri net model with three different types of places and three different types of transitions (with different firing rules). *Media places* (MPs) hold the playing information of the multimedia objects. *Address places* (APs) buffer the spatial information of the referential objects. This information is forwarded to targeted media places when the targeted media places have tokens. *Delay places* (DPs) delay the play of the multimedia objects to coordinate the temporal sequences of the presentations. As in an OCPN, all the places in the STRPN have playing durations within which the tokens are locked. The tokens are then unlocked after the plays are finished. Three types of transitions are distinguished according to the firing rule. *UnLock then fire* (ULF) transitions fire only when all the input places are unlocked. An *enabled then firing continuously* (EFC) transition fires when a new token arrives in its input place and keeps firing until its input token ceases to exist. An *enabled then firing once* (EFO) transition fires immediately when a new token arrives but does not fire again until the next token arrives. Although a STRPN can describe a set of spatio-temporal relationships between objects, its expressiveness is still limited.

A similar, but more complex Petri net-based scheme, called the *multimedia color time Petri net* (MMCTPN) has been proposed by Gomaa et al. [35]. This scheme supports all the capabilities of the STRPN, but adds a user-interaction modeling capability. It is based on color Petri nets [36–38] and specifies four different types of places and four different types of transitions as well as two different types of tokens.

Ribarić and Hrkać [40] proposed a crisp spatio-temporal representation model based on the high-level Petri net called *Petri net with spatio-temporal tokens (PNSTT)*, which is used as the main building block of a knowledge-representation scheme called SpaTem. The SpaTem scheme integrates the theory of the PNSTT and 117 spatio-temporal relationships.

Special efforts in knowledge-base development are made to imitate human-like expert-knowledge representation and human ways of reasoning. Köprülü et al. [41] proposed a model for representing and querying the spatio-temporal properties of the objects in video data. They introduced a set of so-called *fuzzy spatio-temporal relationships* between objects. In reality, however, these relationships are purely *spatial* relationships that become non-exact because of the objects' movement during a certain time interval.

De Runz et al. [48] proposed the use of a fuzzy set theory to represent imprecise multi-modal archaeological data, such as the localization and orientation of antic streets and the estimated time periods of their existence. The described method, however, does not provide an explicit definition of a set of spatio-temporal relationships and it does not support any mechanism of spatio-temporal inference.

Sjahputera et al. [42] described a system capable of linguistically describing an object in motion. The system tracks a single object moving in a straight path at a constant velocity and generates a so-called *dynamic linguistic description* that classifies the direction of the object's movement into one of four possible categories that can be further modified by linguistic *hedges*, such as "*mostly*" or "*a little*". Although the described approach uses linguistic expressions to model the spatio-temporal knowledge, it is not based on fuzzy set theory, because no numerical membership degrees are assigned to the data. In addition, this system does not provide any reasoning mechanisms.

In a number of papers [43–45] some problem-specific systems are described, based on the use of fuzzy rule-based reasoning on a spatio-temporal data. Su [44] described a fuzzy rule-based approach to spatio-temporal hand-gesture recognition. Zaboli et al. [45] described a fuzzy rule-based system for knowledge discovery in analyzed traffic images. Shultz et al. [43] presented a spatio-temporal method of forestry evolution for a sequence of satellite images through the use of a fuzzy rule-based inference system.

3. Petri net with fuzzy spatio-temporal tokens

In this section we describe a new high-level model of the Petri net, called the Petri net with fuzzy spatio-temporal tokens (PeNeFuST). The PeNeFuST is based on a *p*-space-timed net model (which associates information about the time duration of an action or state and the corresponding change of an object's spatial position to each place) and the concept of fuzzy spatio-temporal tokens. A fuzzy spatio-temporal token in the proposed model has a double role: it denotes the state of the modeled system and it carries the spatio-temporal information.

3.1. Formal definition of a Petri net with fuzzy spatio-temporal tokens

The Petri net with fuzzy spatio-temporal tokens (PeNeFuST) is a high-level Petri net defined as the following *n*-tuple:

$$PeNeFuST = (P,T,I,O,\Psi,M,\Omega,\nu,\lambda_0,\delta_0,\kappa,\vartheta),$$

where P, T, I and O are the components of a generalized Petri net (PN), defined as follows [39]: P is a finite set of places $P=\{p_1,p_2,\ldots,p_n\}$, $n\geq 0$, T is a finite set of transitions $T=\{t_1,t_2,\ldots,t_q\}$, $q\geq 0$, I is an input function $I:T\rightarrow P^\infty$, a mapping from transitions into bags of places, and O is an output function, $O:T\rightarrow P^\infty$, a mapping from transitions into bags of places, $P\cap T=\emptyset$.

A function $\Psi:P\rightarrow \Sigma\cup T\cup(\Sigma\times T)$ associates the fuzzy spatial, temporal or spatio-temporal information to the each place. Ψ is a mapping from a set of places to a set of either fuzzy d-dimensional spatial ($\Sigma=(\Sigma_1\times\Sigma_2\times\ldots\times\Sigma_d)$), fuzzy temporal ($T$) or fuzzy spatio-temporal ($\Sigma\times T$) values, where each component of the value is a fuzzy number. In general, the above components of fuzzy values can be represented by any convex and normal fuzzy subset of R [49]. In our model the triangular and/or trapezoidal fuzzy numbers are used. In many applications, the function Ψ can be reduced to the form $\Psi:P\rightarrow\Sigma\times T$.

A set $M=\{m_1^0,m_1^1,\ldots,m_1^j,m_2^0,\ldots,m_2^s,\ldots,m_r^0,\ldots,m_r^p\}$, where $1\leq r<\infty$, is a set of fuzzy spatio-temporal (S-T) tokens. A fuzzy S-T token, like a token in the colored Petri nets [36] has individuality, i.e., it carries inherent information about the visited places and the corresponding spatial changes and the temporal durations of the corresponding activities. A fuzzy S-T token m_i^{k+1} is the successor of the fuzzy S-T token m_i^k, $k=0,1,2,\ldots$, meaning that m_i^{k+1} is generated in the output place of a fired transition, after m_i^k is removed from its input place.

An injective function $\Omega:P\rightarrow\wp(M)$ is called the marking of the PeNeFuST. A $\wp(M)$ denotes the power set of M. With Ω_0 we denote the initial marking, i.e., the initial distribution of fuzzy S-T tokens at the places of the PeNeFuST.

The function ν is a mapping called a spatio-temporal track, and it is defined as follows: $\nu:M\rightarrow \&_{i=1}^{k}\langle p_i,\Psi(p_i)\rangle$, where & denotes concatenation, and k is a number of the visiting places for a token $m_j^k\in M$. It assigns a history of visited places and the corresponding spatial changes and the temporal durations of the corresponding activities to the fuzzy spatio-temporal tokens.

$\lambda_0:M\rightarrow\tilde{R}^d$ is a mapping that associates the initial fuzzy spatial position of the object located in a d-dimensional world to the fuzzy S-T token, where $\tilde{R}^d$ denotes a set of fuzzy numbers defined in R^d.

$\delta_0:M\rightarrow\tilde{R}$, where $\tilde{R}$ denotes a set of fuzzy numbers defined in R, is a mapping from a set of fuzzy S-T tokens to a set of fuzzy numbers, and it specifies the time for which the activity of the object is postponed.

$\kappa:M\rightarrow[0,1]$ is a mapping that associates a degree of confidence about the information carried by each fuzzy spatio-temporal token.

The function $\vartheta:T\rightarrow[0,1]$ is a mapping that assigns a so-called firing threshold to each transition t_j. In order for the transition t_j to fire, there has to be enough tokens in its input places $I(t_j)$, and each of the tokens has to have a degree of confidence κ greater than or equal to the firing threshold $\vartheta(t_j)$.

The ν, λ_0 and δ_0 determine the structure of the fuzzy S-T tokens. Additional information can be extracted from a fuzzy S-T token: the total accumulated time δ_{ac}, i.e., the time durations of all the activities related to the object; the current position of the object λ_c and the degree of confidence about the spatial, temporal or spatio-temporal information κ associated with the S-T token. A fuzzy S-T token carries with it its entire history of visited places, the spatial changes and the time durations of the object's activities. The complete structure of an S-T token is as follows:

$$(\lambda_0,\delta_0,\langle p_i,\Psi(p_i)\rangle,\langle p_j,\Psi(p_j)\rangle,\ldots,\langle p_s,\Psi(p_s)\rangle,\lambda_c,\delta_{ac},\kappa).$$

During the initial marking, the S-T tokens with the following structure $(\lambda_{0i},\delta_{0i},\langle p_i,\Psi(p_i)\rangle,\lambda_{ci},\delta_{aci},\kappa)$ are put into the place $p_i\in P$, $1\leq i\leq n$.

3.2. Graph of the PeNeFuST

The PeNeFuST can be represented by a bipartite directed multigraph. The circles represent the places, while the bars represent the transitions. The directed arcs connecting the places and the transitions are defined by means of an input function I, while the arcs directed from the transitions to the places are defined by an output function O. Multiple input places and multiple output places are represented by multiple arcs. The fuzzy spatio-temporal tokens are represented by dots ($\bullet$) in the places. Due to the individuality of the tokens, every dot is labeled with $m_i^k \in M$; $i=1,2,\ldots,r$; $k=0,1,2,\ldots$.

3.3. Execution of the PeNeFuST

In general, tokens give dynamical properties to marked Petri nets (PNs) and they are used to define the execution of the marked PNs [39]. In the PeNeFuST, the general rule of execution is slightly modified in the following manner: in the PeNeFuST a transition t_j is enabled if each of its input places has at least as many fuzzy S-T tokens, having a degree of confidence κ greater than or equal to the firing threshold $\vartheta(t_j)$, in it as arcs from the place to the transition *and* if the time of duration of the object's activity attached to the place has elapsed. Such S-T tokens are called *movable* S-T tokens. The firing of an enabled transition in the PeNeFuST is performed *automatically* and *immediately* after the transition is enabled. The number of fuzzy S-T tokens at the input and output places of the fired transition is changed in accordance with the basic definition for the original marked PN [39]. The firing of the enabled transition in the PeNeFuST removes the S-T tokens (ancestors) from its input places and simultaneously generates S-T tokens (successors) in its output places. At this moment the structure of a new S-T token is updated by information corresponding to the place p_i according to $\Psi(p_i)$.

If the number of movable S-T tokens at the input place p_i is larger than $\#(p_i,I(t_j))$ for the enabled transition t_j (where # denotes the number of appearances of p_i in the bag $I(t_j)$), then different selection of strategies for choosing the S-T tokens—ancestors can be used. For example: (i) the random selection of the S-T tokens; (ii) the LIFO (last-in–first-out) strategy, based on the order of the arrival of the tokens into the place; (iii) the FIFO (first-in–first-out) strategy; or (iv) the selection of the S-T tokens with the stormiest history, i.e., the S-T tokens with the most complex structure. In our knowledge-representation scheme, we use the strategy (iv) to specify the S-T tokens—ancestors, because these S-T tokens contain the richest information needed for the fuzzy spatio-temporal reasoning and they are used for the generation of S-T tokens—successors.

Example 1. Let us suppose that an agent is initially situated in a 2D world of a size 10×10 spatial units, at the position A with the approximate coordinates (3,5), as depicted in Fig. 1(a). In Fig. 1(a), the intensity of the gray level represents the values of a fuzzy variable corresponding to the initial (A) and final (B) positions of the object. The degree of confidence that the agent is at the position A is 1.0. The agent is unmovable for approximately 10 time units, and after that it moves with a constant velocity for approximately 7 time units, traversing approximately 4 spatial units in the direction of the x-axis and 2 spatial units in the direction of the y-axis, reaching the final position B (Fig. 1(a)). After achieving the final position, the agent stays there forever.

The described simple scenario can be represented by the generic form of the PeNeFuST. The model consists of two places (p_1 and p_2) and one transition (t_1) (Fig. 1(b)) and can be formally represented as follows: $P=\{p_1,p_2\}$; $T=\{t_1\}$; $I(t_1)=\{p_1\}$; $O(t_1)=\{p_2\}$; $M=\{m_1^0,m_1^1\}$.

The movement of the agent is modeled by the place p_1, while the place p_2 corresponds to the final state of the agent at its final position. The imprecise spatial and temporal information associated with the agent's activities can be modeled by triangular fuzzy numbers. These

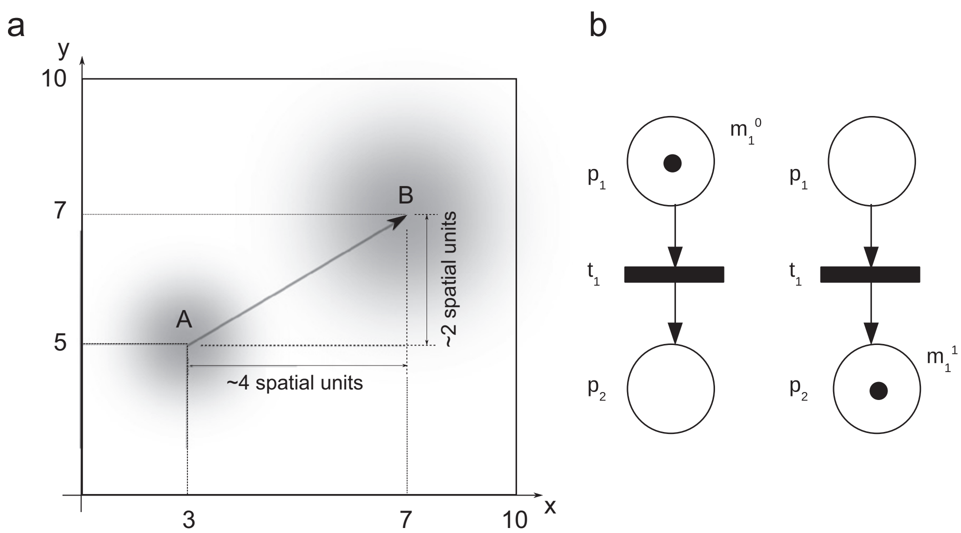

Fig. 1. Representation of a simple, imprecisely known agent movement, with a generic PeNeFuST structure. (a) A movement of an agent. (b) Generic PeNeFuST model.

fuzzy values are associated with the corresponding places of the PeNeFuST by means of the function Ψ:

$\Psi(p_1) = (\langle(3,4,5),(1,2,3)\rangle,(5,7,9))$—the agent moves with a constant velocity for approximately 7 time units, traversing approximately 4 spatial units in the direction of the x-axis and 2 spatial units in the direction of the y-axis.

$\Psi(p_2) = (\langle(0,0,0),(0,0,0)\rangle,(\infty,\infty,\infty))$—the agent stays at the final position forever.

The information about the initial position of the agent λ_0, the initial time of detainment δ_0, as well as the degree of confidence κ about the above information is specified in the initial structure of the token m_1^0:

$m_1^0 = (\langle(2,3,4),(4,5,6)\rangle,(7,10,13),\langle p_1,(\langle(3,4,5),(1,2,3)\rangle,(5,7,9))\rangle,\lambda_c,\delta_{ac},\kappa)$, where :
$\lambda_0 = \langle(2,3,4),(4,5,6)\rangle$—the agent is at the position A with the approximate coordinates (3,5).
$\delta_0 = (7,10,13)$—the agent is unmovable for about 10 time units.
$\langle p_1,(\langle(3,4,5),(1,2,3)\rangle,(5,7,9))\rangle$—the agent is moving approximately 4 spatial units in the direction of the x-axis and 2 spatial units in the direction of the y-axis; the time duration of its moving is about 7 time units.
$\lambda_c = \lambda_0$ and $\delta_{ac} = \delta_0$.
$\kappa = 1.0$.

Note that all the above 3-tuples represent triangular fuzzy numbers, as shown in Fig. 2.

After approximately 7 time units the token m_1^0 becomes movable and the transition t_1 is enabled and automatically fired, resulting in the removal of the token m_1^0 from the place p_1 and the simultaneous placing of its successor m_1^1 at the place p_2.

The structure of the token m_1^1 is

$$m_1^1 = (\lambda_0,\delta_0,\langle p_1,\Psi(p_1)\rangle,\langle p_2,\Psi(p_2)\rangle,\lambda_c,\delta_{ac},\kappa),$$

where

$$\langle p_2,\Psi(p_2)\rangle = \langle p_2,(\langle(0,0,0),(0,0,0)\rangle,(\infty,\infty,\infty))\rangle$$

indicating that there is no additional moving of the agent in the direction of the x-axis and y-axis and (∞,∞,∞) indicates that the agent stays at the final position B forever.

The current position of the agent (position B) is $\lambda_c = \langle(6,7,8),(6,7,8)\rangle$, the total accumulated time is $\delta_{ac} = (12,17,22)$, and $\kappa = 1.0$.

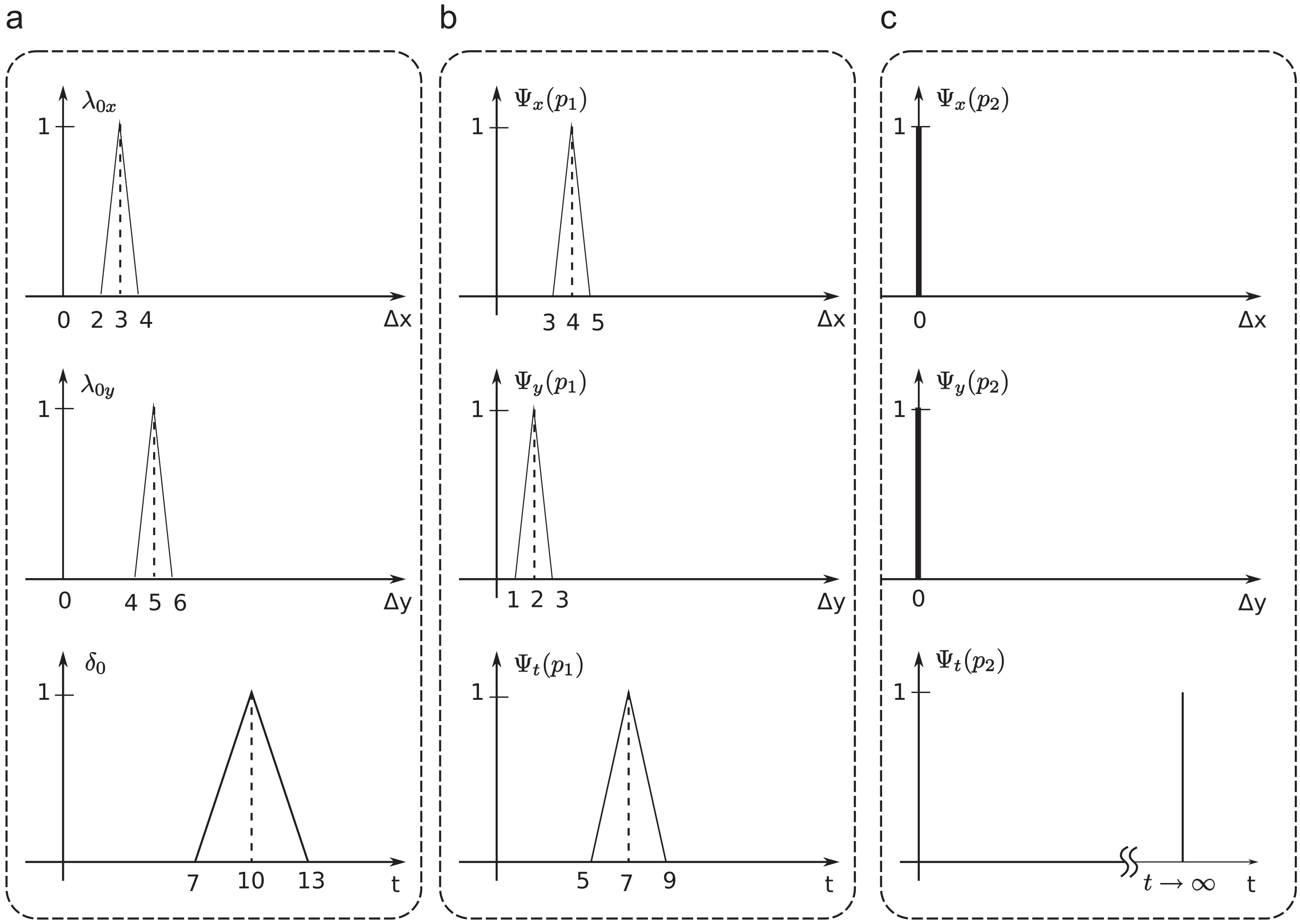

Fig. 2. Fuzzy values for Example 1: (a) fuzzy values corresponding to the initial position λ_0 of the agent (the top and middle rows show x and y coordinates, respectively) and the initial temporal delay δ_0 (bottom row); (b) fuzzy values corresponding to the $\Psi(p_1)$ (top and middle rows correspond to the changes in the x and y coordinates, respectively, while the bottom row corresponds to the temporal duration of the activity); (c) fuzzy values corresponding to the $\Psi(p_2)$ (top and middle rows correspond to the changes in the x and y coordinates, respectively, while the bottom row corresponds to the temporal duration of the activity—the object stays forever).

4. FuSpaT knowledge-representation scheme

4.1. Formal definition

The FuSpaT is defined as the following 8-tuples:

$$FuSpaT = (PeNeFuST, TLM, SLM, STLM, \alpha, \beta, \mathcal{L}, F).$$

The components of the above definition can be described as follows:

PeNeFuST is a Petri net with fuzzy spatio-temporal tokens.

TLM is a temporal logical module that supports temporal inference. A detailed description of the *TLM* will be given in Section 4.1.1.

SLM is a fuzzy spatial logical module that is capable of inferring about the fuzzy spatial relationships between objects. A detailed description of the *SLM* will be given in Section 4.1.2.

STLM is a fuzzy spatio-temporal module that integrates the spatial information obtained from the *SLM* and the corresponding temporal information from the *TLM* into spatio-temporal information. It is described in more detail in Section 4.1.3.

A function $\alpha: P \rightarrow (Ac \cup Cs)$ is a bijective function from a set of places P to a union of a set of activities and/or states Ac and a set of control states Cs.

A surjective function $\beta: T \rightarrow (Ev \cup Ce)$ is a mapping from a set of transitions T to a union of a set of events Ev and a set of control events Ce.

The functions α and β give to the FuSpaT a semantical interpretation of the model.

$\mathcal{L}$ is a linguistic variable used to express the degree of confidence (related to the temporal, spatial or spatio-temporal relationships) in a user-friendly form. The values of the linguistic variable $\mathcal{L}$ are from the following set: {*not true, minimally true, minorly true, more-or-less true, moderately true, considerably true, very true, extremely true, always true*}. The values of the linguistic variable $\mathcal{L}$ are transformed to the intervals according to Table 1 [50].

Note that the initial user's specification of the degree of confidence expressed by the linguistic variable is mapped into the middle point of the corresponding interval.

An F is a set of flags. In general, a flag $f_i \in F$ has the following structure:

$$f_i = (p_i, p_j, \langle tr \text{ or } sr \text{ or } str \rangle, p_k, p_l, \ldots, p_v),$$

where $p_i, p_j \in P$ are the places that potentially have S-T tokens which have to be tested by the *TLM*, *SLM* or *STLM*, according to the specification contained in *tr* or *sr* or *str*, where:

tr is the temporal relationship based on 13 possible Allen's time-interval relationships and on an extension of five time-point–time-interval relationships and three time-point relationships.

sr is the spatial relationship taken, in general, from the set of nine possible relationships.

str is the spatio-temporal relationship taken from the set of 117 possible relationships.

Table 1
The possible values of the linguistic variable [50].

Linguistic variable	Numerical interval
Always true	[1.00, 1.00]
Extremely true	[0.95, 0.99]
Very true	[0.80, 0.94]
Considerably true	[0.65, 0.79]
Moderately true	[0.45, 0.64]
More or less true	[0.30, 0.44]
Minorly true	[0.10, 0.29]
Minimally true	[0.01, 0.09]
Not true	[0.00, 0.00]

The evaluation of the *tr*, *sr* and *str* is based on information that is carried by the fuzzy S-T tokens from places p_i and p_j. This information is transferred to the *TLM* or *SLM* or *STLM* by sending copies of the S-T tokens to it. The destination (*TLM*, *SLM* or *STLM*) of the S-T token copies depends on the type of specified relationship.

The $p_k, p_l, \ldots, p_v$, where $\alpha(p_k) \in Cs, \alpha(p_l) \in Cs, \ldots \alpha(p_v) \in Cs$, in the flag f_i specify the places in which the *TLM*, *SLM* or *STLM* puts the tokens, depending on the result of the evaluation specified in $\langle tr \text{ or } sr \text{ or } str \rangle$. Such places are called *control places*. A transition having one or more control places as an input place is called a *control transition*. The tokens, called *control tokens*, are treated as S-T tokens without temporal and spatial histories.

A degenerative type of flag $f_{Gi} = (p_g, -, -, -)$; $i = 1, 2, \ldots$, and $1 \le g \le n$, where n is the cardinality of a set P, is used to denote the goal state of the system, where p_g denotes the place that corresponds to one of the goals of the modeled system.

4.1.1. Temporal logical module (TLM)

A *TLM* is a temporal logical module that supports, in general, the following temporal relationships: 13 Allen's time-interval relationships [47], five relationships between the time point and the time interval, and three temporal relationships between the time points. The relationships time-point–time-interval and time-point–time-point are obtained by letting one or both of the time intervals degenerate to a time point(s) [51].

The inputs to the *TLM* are copies of two fuzzy S-T tokens and a specification of the temporal relationship that has to be evaluated. An output of the *TLM* is a control token with a value κ equal to the degree of confidence that the relationship is satisfied. The degree of confidence is internally expressed by a value from the interval $[0, 1]$. It can be represented to user or expert by means of a value of the linguistic variable $\mathcal{L}$. The additional output of the *TLM* is a temporal relation with its degree of confidence, which is directed to the spatio-temporal logical module *STLM*. The temporal relationships are generalized to the fuzzy case in the following manner. First, the 13 temporal relationships have to be expressed by means of the relationships between the starting (A^- or B^-) and ending (A^+ or B^+) time points of the intervals A and B, possibly connected by means of logical operators (Table 2) [52]. There are three possible relationships between the starting and ending time points of the intervals: before ($<$), after ($>$) and equal ($=$). For example,

Table 2
Definition of 13 temporal relationships between the intervals A and B. A^- and B^- denote the starting points of the intervals, while A^+ and B^+ denote their ending points [52].

$A<B$: A before B	$(A^+ < B^-)$
$A>B$: A after B	$(B^+ < A^-)$
A o B: A overlaps B	$(A^- < B^-) \wedge (A^+ > B^-) \wedge (A^+ < B^+)$
A oi B: A overlaps inv. B	$(B^- < A^-) \wedge (B^+ > A^-) \wedge (B^+ < A^+)$
A d B: A during B	$(A^- > B^-) \wedge (A^+ < B^+)$
A di B: A during inv. B	$(B^- > A^-) \wedge (B^+ < A^+)$
A m B: A meets B	$(A^+ = B^-)$
A mi B: A meets inv. B	$(B^+ = A^-)$
A s B: A starts B	$(A^- = B^-) \wedge (A^+ < B^+)$
A si B: A starts inv. B	$(A^- = B^-) \wedge (B^+ < A^+)$
A f B: A finishes B	$(B^- < A^-) \wedge (A^+ = B^+)$
A fi B: A finishes inv. B	$(A^- < B^-) \wedge (A^+ = B^+)$
$A{=}B$: A equals B	$(B^- = A^-) \wedge (A^+ = B^+)$

the crisp interval relationship $A\ d\ B$ (interval A during the interval B) can be expressed as $(A^- > B^-) \wedge (A^+ < B^+)$. All the values A^-, B^-, A^+ and B^+ can be generalized to fuzzy values and represented by fuzzy triangular numbers. In order to determine the degree of confidence for the relationships $<$, $>$ and $=$, two fuzzy numbers have to be compared.

The crisp comparison operators $<$, $>$ and $=$ are generalized to fuzzy numbers and expressed by means of the degree of confidence, using the well-known extension principle [49,53], so that the result of comparing two fuzzy numbers is a fuzzy value. The generalized comparison operators are denoted as $<_f$, $>_f$ and $=_f$. In general, the extension principle is defined by the equation:

$$\mu_{A|*|B}(z) = \sup_{z=x*y} \{\min(\mu_A(x), \mu_B(y))\},$$

where $\forall x,y \in E$, $\mu_A(x) \in [0,1]$ and $\mu_B(xy) \in [0,1]$ are membership functions that define the measures of belonging of the elements of E to the fuzzy subsets A and B, respectively. The symbol $*$ denotes any crisp operator, and $|*|$ is its extension to fuzzy numbers. The extension principle is used to evaluate fuzzy temporal (and spatial) relations. The measure of satisfaction of the relation is expressed by the degree of confidence $\kappa = \mu_{A|*|B}$, where A and B are fuzzy numbers representing the temporal (or spatial or spatio-temporal) information and $|*|$ denotes a fuzzy temporal (or fuzzy spatial or spatio-temporal) relationship.

Based on the above extension principle, the confidence degrees of the fuzzy relationships $\le_f$ and $\ge_f$ are first defined as a basis for the relationships $<_f, >_f$ and $=_f$. For two triangular fuzzy numbers $A=(a_1,a_2,a_3)$ and $B=(b_1,b_2,b_3)$ the fuzzy relationships $A \le_f B$ and $A \ge_f B$ are defined as

$$\mu_{A \le_f B} = \begin{cases} 0 & \text{for } a_1 > b_3, \\ \dfrac{b_3 - a_1}{b_3 - a_1 + a_2 - b_2} & \text{for } a_1 \le b_3 \ \&\ b_2 < a_2, \\ 1 & \text{for } a_2 \le b_2, \end{cases}$$

$$\mu_{A \ge_f B} = \begin{cases} 0 & \text{for } b_1 > a_3, \\ \dfrac{a_3 - b_1}{a_3 - b_1 + b_2 - a_2} & \text{for } b_1 \le a_3 \ \&\ a_2 < b_2, \\ 1 & \text{for } b_2 \le a_2. \end{cases}$$

(The above expressions for triangular fuzzy numbers are obtained based on well-known extension principle [53] and by simplification of the expressions for more general trapezoid fuzzy numbers.)

The logical operators $\wedge$, $\vee$ and $\neg$ are generalized to the fuzzy case using well-known equations [25,49]:

$$\mu_{A \wedge B} = \min(\mu_A, \mu_B),$$

$$\mu_{A \vee B} = \max(\mu_A, \mu_B),$$

$$\mu_{\neg A} = 1 - \mu_A.$$

The fuzzy relationships $=_f$, $<_f$ and $>_f$ can then be expressed as

$$A =_f B \Leftrightarrow (A \le_f B) \wedge (A \ge_f B),$$

$$A <_f B \Leftrightarrow (A \le_f B) \wedge \neg(A =_f B),$$

$$A >_f B \Leftrightarrow (A \ge_f B) \wedge \neg(A =_f B),$$

which gives

$$\mu_{A =_f B} = \min(\mu_{A \le_f B}, \mu_{A \ge_f B}),$$

$$\mu_{A <_f B} = \min(\mu_{A \le_f B}, 1 - \mu_{A =_f B}),$$

$$\mu_{A >_f B} = \min(\mu_{A \ge_f B}, 1 - \mu_{A =_f B}).$$

Example 2. Two fuzzy intervals $A=(A^-,A^+)$ and $B=(B^-,B^+)$ are specified by their starting and ending points that are defined as triangular fuzzy numbers: $A^- = (1,2,3)$; $A^+ = (5,6,7)$; $B^- = (1,3,5)$; $B^+ = (7,9,10)$, as shown in Fig. 3.

The fuzzy relationship "*during*" ($A\ d_f\ B$) can be evaluated as follows. The relationship "*during*" is defined as: $A d_f B \equiv (A^- >_f B^-) \wedge (A^+ <_f B^+)$ (Table 2). Therefore, the degree of confidence for the relationship "*during*" can be expressed as

$$\begin{aligned} \mu_{A d_f B} &= \min(\mu_{A^- >_f B^-}, \mu_{A^+ <_f B^+}) \\ &= \min(\min(\mu_{A^- \ge_f B^-}, 1 - \mu_{A^- =_f B^-}), \\ &\quad \min(\mu_{A^+ \le_f B^+}, 1 - \mu_{(A^+ =_f B^+)})) \\ &= \min(\min(\mu_{A^- \ge_f B^-}, 1 - \min(\mu_{A^- \le_f B^-}, \mu_{A^- \ge_f B^-})), \\ &\quad \min(\mu_{A^+ \ge_f B^+}, 1 - \min(\mu_{A^+ \le_f B^+}, \mu_{A^+ \ge_f B^+}))) \end{aligned}$$

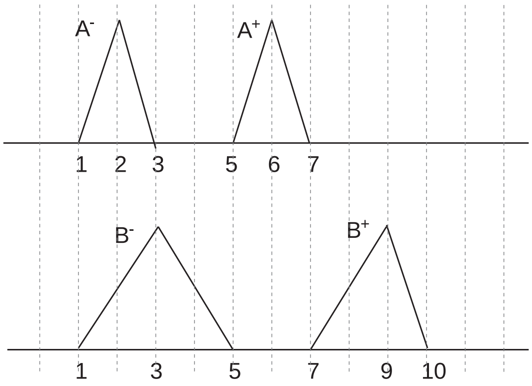

Fig. 3. Two fuzzy time intervals A and B specified by their starting points A^- and B^- and ending points A^+ and B^+ (Example 2).

By using the equations for the fuzzy relationships $\geq_f$ and $\leq_f$, the following values can be calculated:

$$\mu_{A^- \geq_f B^-} = \frac{3-1}{3-1+3-2} = \frac{2}{3},$$

$$\mu_{A^- \leq_f B^-} = 1,$$

$$\mu_{A^+ \geq_f B^+} = \frac{7-7}{7-7+9-6} = 0,$$

$$\mu_{A^+ \leq_f B^+} = 1.$$

Substitution of the calculated values into the equation for μ_{Ad_fB} gives

$$\begin{aligned}\mu_{Ad_fB} &= \min(\min(\tfrac{2}{3},1-\min(1,\tfrac{2}{3})),\min(1,1-\min(1,0)))\\ &= \min(\min(\tfrac{2}{3},1-\tfrac{2}{3}),\min(1,1-0))\\ &= \min(\min(\tfrac{2}{3},\tfrac{1}{3}),1) = \min(\tfrac{1}{3},1) = \tfrac{1}{3}.\end{aligned}$$

Therefore, the fuzzy relationship “during” between the fuzzy intervals A and B is satisfied with the degree of confidence κ equal to $\frac{1}{3}$. In the user friendly form, the degree of confidence $\kappa = \frac{1}{3}$ is represented by the value of the linguistic variable “*more or less true*” (Table 1).

4.1.2. Spatial logical module (SLM)

A *SLM* is a fuzzy spatial logical module that is capable of inferring about the fuzzy spatial relationships between objects. In general, the combinations of possible 2D spatial relationships between two non-concave objects are represented by 169 relationships [54]. These relationships are based on an analogy with the 13 well-known Allen’s time-interval relationships [47], extended to 2D space. These 169 relationships can also be considered as an extension of the RCC8 [55] where regions are represented by the minimum bounding boxes, in the sense that they give more precise information about 2D spatial relationships.

In many applications, however, the positions of the physical objects or agents can be represented by points (for example, in robot-vision systems the position of an object is often represented by a point corresponding to the centroid); therefore, in the paper we consider only 2D relationships between objects that are reduced to points.

The spatial relationships of the objects that are represented by points have an analogy to Allen’s time-interval relationships extended to 2D, where the time intervals degenerate to the time points. In this case, there are only three relationships in the time domain [51], and consequently only nine crisp relationships in the 2D spatial domain (Fig. 4). These nine spatial relationships between the objects A and B can be denoted as: A lb B (A is to the left and below B), A b B (A is below B), A rb B (A is to the right and below B), A l B (A is to the left of B), A m B (A meets B), A r B (A is to the right of B), A la B (A is to the left and above B), A a B (A is above B), and finally A ra B (A is to the right and above B).

The inputs to the *SLM* are copies of two fuzzy S-T tokens and a specification of the spatial relationship that has to be evaluated. An output of the *SLM* is a control token having the degree of confidence κ corresponding to the degree of confidence that the relationship is satisfied (in the interval $[0,1]$). The additional output of the *SLM* is a spatial relation with its degree of confidence, which is directed to the spatio-temporal logical module *STLM*. The evaluation of the degree of confidence of the spatial relation is also based on the extension principle. In order to calculate a degree of confidence for the specified relationship between the two fuzzy values, nine relationships have to be expressed by means of the relationships between the x and y coordinates of the objects, connected with logical operators (Table 3). The crisp relationships between the coordinates of the objects are: less ($<$), greater ($>$) and equal ($=$). For example, the relationship A lb B (object A is left and below the object B) can be expressed as $(x_A < x_B) \wedge (y_A > y_B)$, where (x_A, y_A) and (x_B, y_B) are the coordinates of the objects A and B, respectively. All the values x_A, x_B, y_A and y_B can be generalized to the fuzzy values and represented by fuzzy triangular numbers. The relationships $<$, $>$ and $=$ are generalized to the fuzzy case in an identical way as in Section 4.1.1.

4.1.3. Spatio-temporal logical module (STLM)

A *STLM* integrates the spatial information obtained from the *SLM* and the corresponding temporal

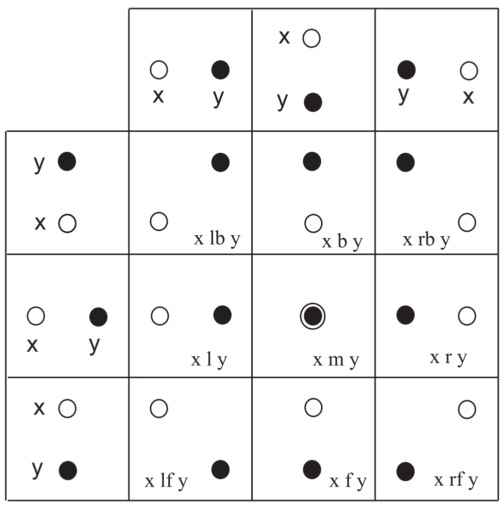

Fig. 4. Nine 2D spatial relationships between two objects represented as points (○ represents object x and • represents object y).

Table 3
Definition of nine spatial relationships between the objects A and B. x_A and y_A denote the coordinates of the object A, while x_B and y_B denote the coordinates of the object B.

A lb B: (A left and below B)	$(x_A < x_B) \wedge (y_A > y_B)$
A b B: (A below B)	$(x_A = x_B) \wedge (y_A > y_B)$
A rb B: (A right and below B)	$(x_A > x_B) \wedge (y_A > y_B)$
A l B: (A left of B)	$(x_A < x_B) \wedge (y_A = y_B)$
A m B: (A meets B)	$(x_A = x_B) \wedge (y_A = y_B)$
A r B: (A right of B)	$(x_A > x_B) \wedge (y_A = y_B)$
A la B: (A left and above B)	$(x_A < x_B) \wedge (y_A < y_B)$
A a B: (A above B)	$(x_A = x_B) \wedge (y_A < y_B)$
A ra B: (A right and above B)	$(x_A > x_B) \wedge (y_A < y_B)$

information from the *TLM* into spatio-temporal information. Based on the combination of 13 temporal relationships and nine spatial relationships (Fig. 4), a total of 117 spatio-temporal relationships are supported by the *STLM* (Table 4).

A *STLM* is implemented as a look-up table with three inputs: (i) the spatial relationship obtained from the *SLM* accompanied by its degree of confidence; (ii) the temporal relationship obtained from the *TLM* accompanied by its degree of confidence; and (iii) a set of flags *F*. The output is coded and represented in the form of a semantical interpretation of the spatio-temporal relationship (Table 4). For example, if the inputs are "X lb Y" (spatial relationship) and "X=Y" (temporal relationship), the corresponding entry from Table 4 "X lb=Y" is interpreted as: "The agent X is to the left and below the agent Y and their activities are simultaneous". The result of the evaluation of the spatio-temporal relationship by the *STLM* is a control token containing a degree of confidence that the spatio-temporal relationship is satisfied (in the interval $[0,1]$), obtained as a fuzzy logical *and* ($\wedge$) between the degrees of confidence for the corresponding spatial and temporal relationships. The value of the linguistic variable $\mathcal{L}$ is used for a user-friendly interpretation of the degree of confidence that the spatio-temporal relationship is satisfied.

4.2. Reasoning

The spatial and temporal information is contained in the FuSpaT model of the world. The reasoning process in the proposed knowledge-representation scheme is defined as a *spatio-temporal data-driven process* as follows: The input in the reasoning process is the initial distribution of S-T tokens in the FuSpaT model. This distribution determines the current positions, the activities and the states and spatio-temporal relationships between the activities of the objects or the agents. As time proceeds, the enabled transitions are automatically fired and the S-T tokens are distributed through the graph of the PeNeFuST. This corresponds to changes of the objects' or agents' positions and their activities. Depending on their path through the graph, the time duration of the activities (states), and the corresponding spatial changes of the objects, each S-T token carries a history of the execution of the PeNeFuST. The firing sequences are additionally controlled by the *TLM*, *SLM* or *STLM*.

The main step of reasoning in the proposed model can be described as follows (Fig. 5). When an S-T token arrives at the place denoted by a flag, its copies are sent to the *TML* and the *SLM* (in general, if the relationship specified by the flag is *str*—spatio-temporal relationship). At the moment when another S-T token arrives at another place denoted by the same flag, its copies are also sent to the *TLM* and the *SLM*. Note that the additional input for the *TLM* and the *SLM* is a set of flags *F*. When the *TLM* and the *SLM* have received both copies of the S-T tokens, the following simultaneous activities are performed:

(i) The *TLM* evaluates the temporal relationship specified by the flag (extracted from *str*) as it was described in Section 4.1.1. The output of the *TLM* is a control token with a degree of confidence of the specified temporal relationship which is sent to the knowledge base (the case when the flag specifies only the temporal relationship). The additional output of the TLM is the temporal relationship and its degree of confidence, directed to the *STLM* (in the case when the spatio-temporal relationship is specified). The optional output of the *TLM* is the user-friendly interpretation of the degree of confidence of the specified temporal relationship by means of a value of the linguistic variable $\mathcal{L}$.

(ii) The *SLM* evaluates the spatial relationship specified by the flag (extracted from *str*) (see Section 4.1.2). The output of the *SLM* is a control token with a degree of confidence of the specified spatial relationship which is sent back to the knowledge base (the case when the flag specifies only the spatial relationship). Additionally, the output of the *SLM* can be the spatial relationship and its degree of confidence, directed to the *STLM* (in the case when the spatio-temporal relationship is specified). Also, as an optional output of the *SLM* there is the value of the linguistic variable $\mathcal{L}$—the user-friendly interpretation of the degree of confidence of the specified spatial relationship.

The outputs of the *TLM* and the *SLM* (the temporal relationship and its degree of confidence, and the spatial relationship and its degree of confidence, respectively) are sent to the *STLM*. The *STLM* is realized as look-up table where the temporal relationships correspond to the rows of the look-up table, while the spatial relationships correspond to the columns. The spatio-temporal relationship for specific inputs of the *STLM* is obtained as the content of the table-entry on crossing of the corresponding row and column of the look-up table. The degree of

Table 4
Representation of the 117 spatio-temporal relationships supported by the *STLM*; ○ and • represent the positions of the objects X and Y, respectively; xxx and yyy denote the corresponding time intervals.

	X lb Y • ○	X b Y • ○	X rb Y • ○	X l Y ○ •	X m Y ⊙	X r Y • ○	X la Y ○ •	X a Y ○ •	X ra Y ○ •
$X<Y$ xx yy	X lb < Y	X b < Y	X rb < Y	X l < Y	X m < Y	X r < Y	X la < Y	X a < Y	X ra < Y
X = Y xxx yyy	X lb = Y	X b = Y	X rb = Y	X l = Y	X m = Y	X r = Y	X la = Y	X a = Y	X ra = Y
X *m* Y xx yy	X lbm Y	X bm Y	X rbm Y	X lm Y	X mm Y	X rm Y	X lam Y	X am Y	X ram Y
X *o* Y xxx yyy	X lbo Y	X bo Y	X rbo Y	X lo Y	X mo Y	X ro Y	X lao Y	X ao Y	X rao Y
X *d* Y xx yyyy	X lbd Y	X bd Y	X rbd Y	X ld Y	X md Y	X rd Y	X lad Y	X ad Y	X rad Y
X *s* Y xx yyyy	X lbs Y	X bs Y	X rbs Y	X ls Y	X ms Y	X rs Y	X las Y	X as Y	X ras Y
X *f* Y xx yyyy	X lbf Y	X bf Y	X rbf Y	X lf Y	X mf Y	X rf Y	X laf Y	X af Y	X raf Y
X > Y xx yy	X lb > Y	X b > Y	X rb > Y	X l > Y	X m > Y	X r > Y	X la > Y	X a > Y	X ra > Y
X *mi* Y xx yy	X lbmi Y	X bmi Y	X rbmi Y	X lmi Y	X mmi Y	X rmi Y	X lami Y	X ami Y	X rami Y
X *oi* Y xxx yyy	X lboi Y	X boi Y	X rboi Y	X loi Y	X moi Y	X roi Y	X laoi Y	X aoi Y	X raoi Y
X *di* Y xxxx yy	X lbdi Y	X bdi Y	X rbdi Y	X ldi Y	X mdi Y	X rdi Y	X ladi Y	X adi Y	X radi Y
X *si* Y xxxx yy	X lbsi Y	X bsi Y	X rbsi Y	X lsi Y	X msi Y	X rsi Y	X lasi Y	X asi Y	X rasi Y
X *fi* Y xxxx yy	X lbfi Y	X bfi Y	X rbfi Y	X lfi Y	X mfi Y	X rfi Y	X lafi Y	X afi Y	X rafi Y

confidence of the spatio-temporal relationship is determined as a minimum of the degrees of confidence for temporal and spatial relationships, i.e., as a fuzzy logical *AND* operation.

Based on the flag, the *STLM* sends control token(s) with the obtained degree of confidence to the control place(s) of the modeled system. Optionally, the additional user-friendly output of the *STML* is the spatio–temporal relationship with the degree of confidence expressed by the value of the linguistic variable $\mathcal{L}$. The value of the linguistic variable $\mathcal{L}$ is obtained by mapping the degree of confidence κ to the set of possible values of $\mathcal{L}$ (see Table 1).

Each control token carries an appropriate degree of confidence $\kappa \in [0,1]$, corresponding to the degree of confidence that the spatial, temporal or spatio-temporal relationship is satisfied. These control tokens have an influence on the result of the firing sequence by means of enabling transitions (the transition t_j becomes enabled if there are enough tokens with $\kappa \geq \vartheta(t_j)$ in its input places). The combination of the S-T tokens associated with activities and the control tokens, both present at the same time, can be interpreted as a fuzzy spatio-temporally dependent *if–then* rule implementation: If there are enough S-T and control tokens at the corresponding input places with fuzzy confidence degrees κ greater than or equal to the transition firing threshold ϑ, the transition is automatically fired, i.e., the spatio-temporally dependent rule is activated and the action (or conclusion) is

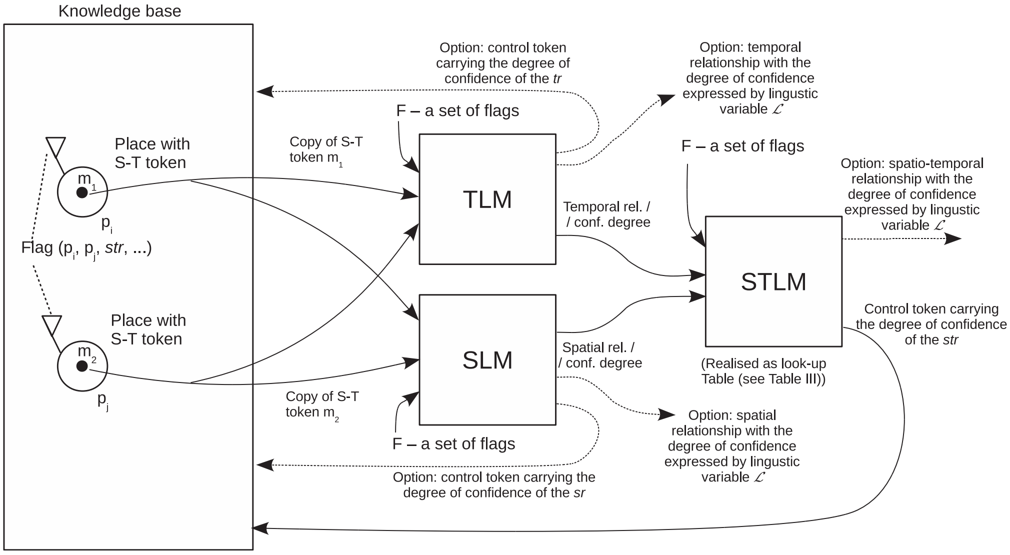

Fig. 5. The process of reasoning in the PeNeFuST model.

generated. It is obvious that the above-described process is driven by spatio-temporal events. If some places in the graph are denoted as goal states, the scheme can conclude if these spatio-temporally dependent goals may be achieved. Based on information in the S-T token that has achieved the goal state, the sequences of activities (states) that lead to the goal can be registered. By varying the initial marking of the scheme it can be used for planning in spatio-temporally rich domains.

4.3. Formal analysis of the FuSpaT model

In general, analysis problems for Petri net-based models, such as safeness, boundedness, conservation, etc., are based on the reachability tree [39,56], which is a finite representation of all the markings of the Petri net that can be reached from the initial marking. The nodes of the reachability tree represent the markings of the Petri net and its arcs represent possible changes in state resulting from the firing of transitions.

In our proposed model, however, a direct application of the reachability tree is not possible, because the original reachability tree construction algorithm presupposes that new markings can result only by firing enabled transitions in the current marking. In the FuSpaT model, however, this is not the case, because new markings can be achieved by placing control tokens to control places. These control tokens can then result in enabling additional transitions that would not be enabled otherwise.

In order to overcome the above-described problem and to enable a formal analysis of the FuSpaT model, we propose the construction of a modified reachability tree called the *conditional reachability tree*. The construction of the conditional reachability tree is based on the distinction between the two possible cases of reaching the new marking. The *unconditional immediate reachability* is defined in the same way as the immediate reachability in the generalized Petri nets: the marking μ' is unconditionally immediately reachable from the marking μ if there exists a transition t_j such that t_j is enabled in μ and its firing results in μ':

$$\exists t_j : \delta'(\mu,t_j) = \mu',$$

where δ' is the next-state function [39], modified according to the execution of the PeNeFuST.

The *conditional immediate reachability* is defined to account for the markings that can be reached only if control tokens are placed to control places. The marking μ' is conditionally immediately reachable from the marking μ if there exists another marking μ'' that can be reached from μ by placing certain control tokens to control places and μ' is unconditionally immediately reachable from μ'':

$$\exists \mu' \exists \mu_{ctrl} \exists t_j : (\delta'(\mu',t_j) = \mu'') \wedge (\mu' = \mu + \mu_{ctrl}),$$

where μ_{ctrl} denotes the marking vector having non-zero values only at the positions corresponding to the control places, and $+$ denotes the vector addition.

Based on the above definitions, the algorithm of the conditional reachability tree construction can be developed. The algorithm is based on the original reachability tree algorithm [39], with appropriate modifications related to the conditional reachability.

The conditional reachability tree can be graphically represented in a similar manner to the reachability tree of the generalized Petri net, except that the arcs corresponding to the conditional reachability are represented by dashed lines, with the specification of the conditions that have to be satisfied specified near the arc.

Based on the proposed model FuSpaT, an object-oriented simulator was developed. Some components of the program simulator are illustrated in the example that follows.

5. An example

In this section we give an example of using the FuSpaT model for planning in a simple dynamical scene. The scene represents a detail of a simplified scenario from the world of robot soccer. Robot soccer is selected because it provides a test bed where different models and algorithms can be tested and where many real-world characteristics are present. Two robots are situated on the pitch with a size of 60×80 spatial units (Fig. 6).

Robot A (Team 1) is initially situated at the crisp position (30,60) and it initially possesses the ball. *Robot B* (Team 2) is situated at the crisp position (20,20). There are three possible strategies available to *Robot A*:

(a) to shoot the ball directly towards the goal;
(b) to try to bypass *Robot B* and then shoot the ball towards the goal;
(c) to shoot the ball towards the perimeter fence of the pitch in such a way that the ball rebounds in the direction of the goal.

Robot B chooses its activity based on the activity of *Robot A*. If *Robot A* chooses the strategy (a), *Robot B* moves towards the center of the pitch and tries to intercept the ball. If *Robot A* chooses strategy (b) or (c), *Robot B* moves towards the right and tries to intercept the ball.

Let us suppose that we want to find an answer to the following question: can *Robot A* achieve the goal state "the ball is in the goal", and, if it can, which strategy should be selected to achieve the goal state?

The described scenario can be modeled by the FuSpaT model depicted in Fig. 7.

The sets of places and transitions, the input and output functions, as well as the semantic meanings of the places and transitions specified by means of the functions α and β are denoted in Fig 7.

A firing threshold $\vartheta(t_i) = 0.4; i = 4, 5, \ldots, 11$ is experimentally assigned to the control transitions $t_4, t_5, t_6, t_7, t_8, t_9, t_{10}$ and t_{11}. All the other transitions have a firing threshold equal to 0:

$$\vartheta(t_1) = \vartheta(t_2) = \vartheta(t_3) = 0,$$

$$\vartheta(t_4) = \vartheta(t_5) = \vartheta(t_6) = \vartheta(t_7) = \vartheta(t_8) = \vartheta(t_9) = \vartheta(t_{10}) = \vartheta(t_{11}) = 0.4.$$

Let us further suppose that the temporal and spatial information about the activities is available based on prior knowledge about the characteristics of the robots and the speed of the ball movement, and that it can be modeled by means of the function Ψ as follows.

For *Robot A*:

$\Psi(p_1) = (\langle (0,0,0),(0,0,0) \rangle,(0,0,0))$; $\Psi(p_2) = (\langle (-5,0,5),(30,30,30) \rangle,(3,3,3))$; $\Psi(p_3) = (\langle (15,20,25),(-35,-30,-25) \rangle,(4,5,6))$; $\Psi(p_4) = (\langle (25,30,35),(-35,-30,-25) \rangle,(1,2,3))$; $\Psi(p_5) = (\langle (0,0,0),(0,0,0) \rangle,(0,0,0))$; $\Psi(p_6) = (\langle (0,0,0),(0,0,0) \rangle,(0,0,0))$; $\Psi(p_7) = (\langle (0,0,0),(0,0,0) \rangle, (0,0,0))$; $\Psi(p_8) = (\langle (-25, -20, -15), (-35, -30, -25) \rangle,(1, 2, 3))$;

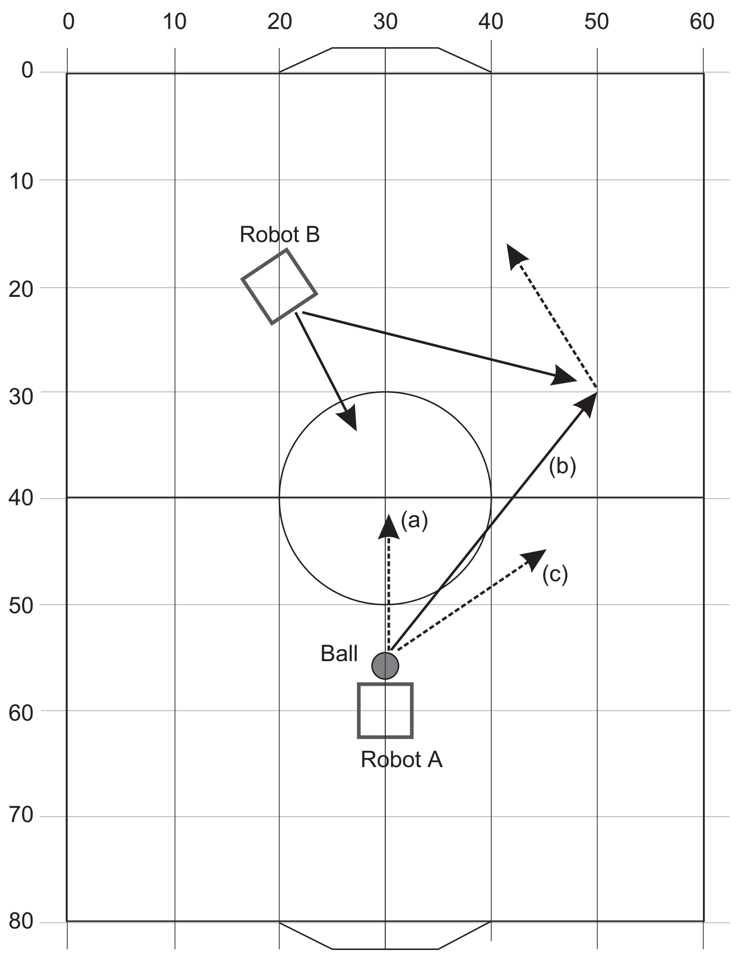

Fig. 6. A simplified scenario from the robot-soccer world.

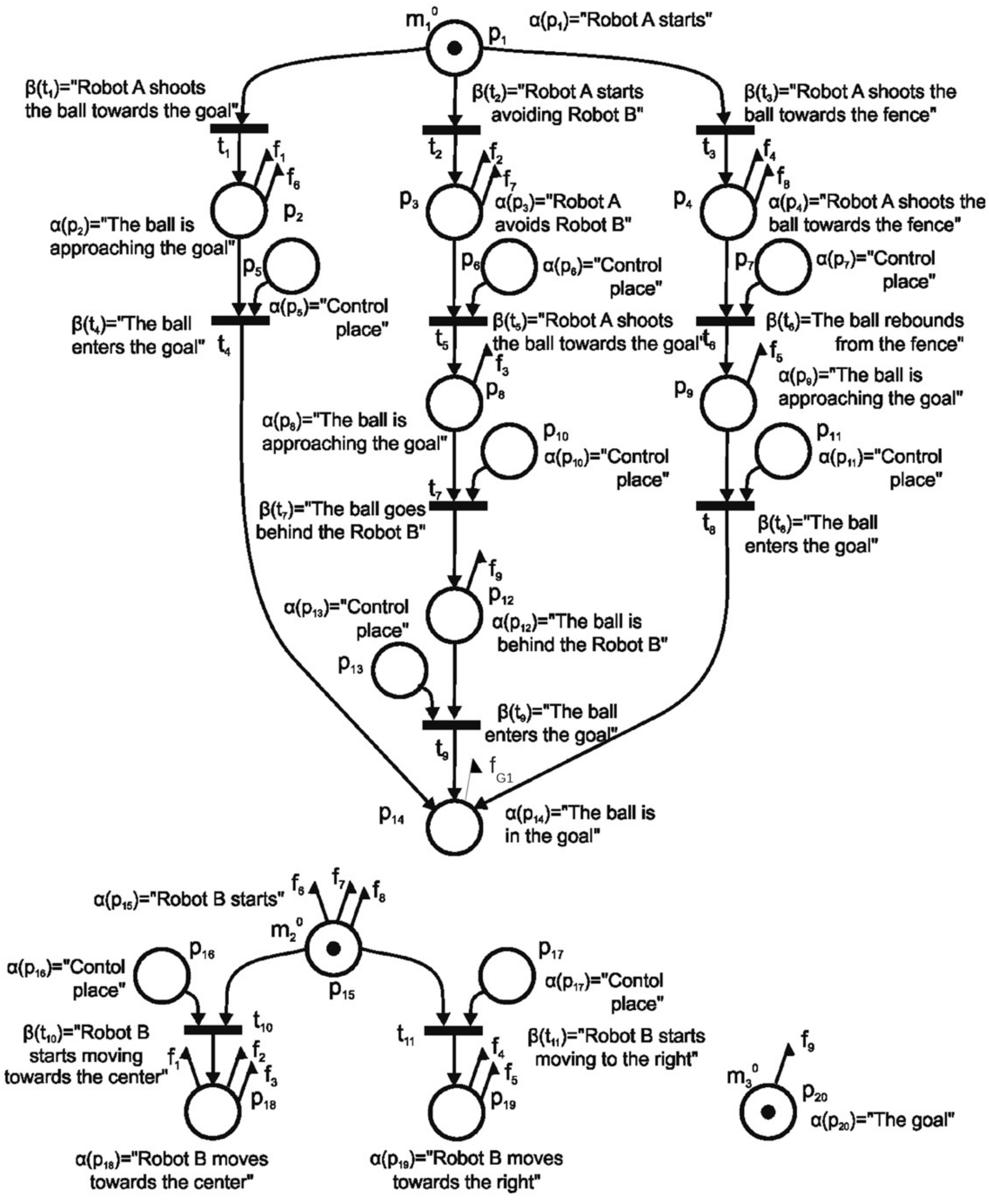

Fig. 7. FuSpaT model of the simplified scenario from the robot-soccer world.

$\Psi(p_9) = (\langle(-25, -20, -15), (-35, -30, -25)\rangle, (1,2,3))$; $\Psi(p_{10}) = (\langle(0,0,0),(0,0,0)\rangle,(0,0,0))$; $\Psi(p_{11}) = (\langle(0,0,0),(0,0,0)\rangle,(0,0,0))$; $\Psi(p_{12}) = (\langle(0,0,0),(0,0,0)\rangle,(0,0,0))$; $\Psi(p_{13}) = (\langle(0,0,0),(0,0,0)\rangle,(0,0,0))$; $\Psi(p_{14}) = (\langle(0,0,0),(0,0,0)\rangle,(\infty,\infty,\infty))$.

For example, $\Psi(p_9)=(\langle(-25,-20,-15),(-35,-30,-25)\rangle,(1,2,3))$ specifies as follows: a pair $\langle(-25,-20,-15),(-35,-30,-25)\rangle$ denotes the fuzzy change of the position of the ball in the x and y directions, respectively, while the triplet (1,2,3) represents the temporal duration of this activity (about two time units).

For *Robot B*:

$\Psi(p_{15}) = (\langle(0,0,0),(0,0,0)\rangle,(1,2,3))$; $\Psi(p_{16}) = (\langle(0,0,0),(0,0,0)\rangle,(0,0,0))$; $\Psi(p_{17}) = (\langle(0,0,0),(0,0,0)\rangle,(0,0,0))$; $\Psi(p_{18}) = (\langle(8,10,12),(8,10,12)\rangle,(1,2,3))$; $\Psi(p_{19}) = (\langle(25,30,35),(8,10,12)\rangle,(4,5,6))$; $\Psi(p_{20}) = (\langle(0,0,0),(0,0,0)\rangle,(\infty,\infty,\infty))$.

The interaction between the robots in the scene is modeled by a set of flags

$$F=\{f_1,f_2,f_3,f_4,f_5,f_6,f_7,f_8,f_9,f_{G1}\},$$

where $f_1=(p_2,p_{18},str_1,p_5)$; $f_2=(p_3,p_{18},str_2,p_6)$; $f_3=(p_8,p_{18},str_3,p_{10})$; $f_4=(p_4,p_{19},str_4,p_7)$; $f_5=(p_9,p_{19},str_5,p_{11})$; $f_6=(p_2,p_{15},str_6,p_{16})$; $f_7=(p_3,p_{15},str_7,p_{17})$; $f_8=(p_4,p_{15},str_8,p_{17})$; $f_9=(p_{12},p_{20},sr_9,p_{13})$ and $f_{G1}=(p_{14},-,-,-)$.

For example, the flag $f_1=(p_2,p_{18},str_1,p_5)$ specifies the evaluation of the spatio-temporal relationship str_1 based on the information obtained from the S-T tokens at the places p_2 and p_{18}, where the semantic interpretations of p_2 and p_{18} are "the ball is approaching the goal" and "*Robot B* is moving to the center of the pitch", respectively. The S-T relationship str_1 is defined as $str_1 = U\backslash\{m = mm, mo, md, ms, mf, mmi, moi, mdi, msi, mfi\}$ (see Table 4), meaning that str_1 includes all the possible S-T relationships (connected with logical *or*), except the relationships that represent the spatial meeting of the ball and the *Robot B* in the overlapping temporal periods. If the relationship str_1 is satisfied, the control token is placed at the control place p_5. If the token has a degree of confidence greater than or equal to 0.4, the control transition t_4 will be enabled (see Fig. 7).

The meaning of the other flags can be described in an analogous way. In our example, the spatio-temporal relationships are $str_2=str_3=str_4=str_5=str_1$.

The flags f_6, f_7 and f_8 enable the selection of the *Robot B*'s activity. Since the selection of the *Robot B*'s activity depends only on the activity of *Robot A*, $str_6=str_7=str_8=U$, where U represents a set of all possible S-T relationships.

Flag f_9 enables the entrance of the ball into the goal if the ball hits the spatial area of the goal, i.e., if the spatial relationship $sr_9=m$ is satisfied.

The degenerative flag f_{G1} denotes the goal state of the system, "The ball is in the goal".

The initial marking of the PeNeFuST is

$$\Omega_0 = (\{m_1^0\},\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\emptyset,\{m_2^0\},\emptyset,\emptyset,\emptyset,\emptyset,\{m_3^0\}),$$

where $m_1^0=(\lambda_0=\langle 30,60\rangle,\delta_0=0,\langle p_1,(\langle 0,0\rangle,0)\rangle,\lambda_c=\langle 30,60\rangle,\delta_{ac}=0,\kappa=1.0)$, where $\langle 30,60\rangle$ denotes an initial crisp position of the *Robot A*. $m_2^0=(\lambda_0=\langle 20,20\rangle,\delta_0=0,\langle p_{15},(\langle 0,0\rangle,(0,1,2)\rangle,\lambda_c=\langle 20,20\rangle,\delta_{ac}=(0,1,2),\kappa=1.0)$, where $\langle 20,20\rangle$ denotes an initial crisp position of the *Robot B* and the triplet $\langle 0,1,2\rangle$ represents the time of detainment (around one time unit) of its activity. $m_3^0=(\lambda_0=\langle(20,20,40,40),0\rangle,\delta_0=0,\langle p_{20},(\langle 0,0\rangle,\infty)\rangle,\lambda_c=\langle(20,20,40,40),0\rangle,\delta_{ac}=\infty,\kappa=1.0)$, where the 4-tuple $\langle 20,20,40,40\rangle$ represents a degenerative trapezoid number that denotes the position of the goal on the pitch.

An execution of the described model can be simulated using the developed program simulator. The simulator enables an execution of the model, either step by step or all at once, an observation of the current marking of the PeNeFuST at each moment (Fig. 8), an inspection of the structure of all the tokens and a graphical representation of the spatial and temporal relationships among the events contained in the tokens, an inspection of the flags and the construction of the conditional reachability tree.

In the first step of the simulation, the transitions t_1, t_2 and t_3 are enabled, but only one of them can be fired, corresponding to the initial selection of the possible strategies of *Robot A*. This is a non-deterministic situation and the simulator asks the user which transition to fire. Let us suppose that the user chooses the transition t_3 to be fired, in an attempt to find out whether the goal can be achieved by means of strategy (c). By firing transition t_3, token m_1^0 is removed from the place p_1 and its successor m_1^1 is put into the place p_4 (see Fig. 7 or Fig. 8). At this moment, there are S-T tokens present in both places specified by the flag f_8 (i.e., the places p_4 and p_{15}), so that the data-driven spatio-temporal logical module *STLM* is activated in order to evaluate the spatio-temporal relationship str_8 specified by the flag f_8. Since $str_8=U$, it is trivially satisfied with the confidence $\kappa=1.0$ and the control token with $\kappa=1.0$ is placed into the control place p_{17}, thus enabling the transition t_{11}. By firing transition t_{11}, token m_2^0 is removed from the place p_{15} and its successor m_2^1 is put into the place p_{19}. The presence of the S-T tokens in places p_4 and p_{19} activates the evaluation of the relationship str_4 specified by the flag f_4.

At each step of the execution, owing to the program simulator, the user can monitor the current structure of the tokens in a separate *Token-view* window. The *Token-view* window for the situation after the firing of the transition t_{11} is shown in Fig. 9. The window is composed of three parts. In the uppermost part, numerical information about the current structure of all the tokens is shown. In the middle part, the spatial relationships among the object are graphically represented. The initial positions of the objects are represented as small black circles. The lines, denoted by the labels of the places, represent the movements of the objects during the activities associated with the places. The fuzzy intermediate positions of the objects are represented by the rectangles. The rectangles correspond to the bases of the 2D triangular fuzzy numbers, and the position having the peak fuzzy value (which is always 1) is indicated by the intersection of the dashed lines inside the rectangle. For example, the rectangle at position (60,30) represents the fuzzy position of the ball at fuzzy time about 3 time units, while the rectangle at position (30,50) represents the fuzzy position of the *Robot B* after the activity corresponding to the place p_{19}, i.e., after approximately 6 time units (Fig. 9). The bottom part of the window provides a graphical representation of the fuzzy time intervals. The fuzzy time points corresponding to the beginnings and ends of the intervals are graphically represented as triangular fuzzy numbers, and the intervals between them are denoted by the labels of the corresponding places. The first line corresponds to the S-T token m_1^1, where the bar at the time point 0 represents the duration of the activity at the place p_1 as well as the beginning of the interval corresponding to the place p_4, while the triangular fuzzy number $(1,2,3)$ represent the end of the interval associated to the place p_4. The second line represents the time information related to S-T token m_2^1.

Let us go back to the execution process. Based on the flag f_4, by evaluating the S-T tokens m_1^1 and m_2^1 shown in Fig. 9, the *STLM* finds that the spatio-temporal relationship str_4 is satisfied with a degree of confidence 0.5, and places the control token with $\kappa=0.5$ into the control place p_7. The degree of confidence 0.5 is obtained based on evaluating 106 S-T relationships (from the complete set of 117 relationships) which are included in the relationship str_4, where $str_4=U\backslash\{m=,mm,mo,md,ms,mf,mmi,moi,mdi,msi,mfi\}$; note that 11 relationships are

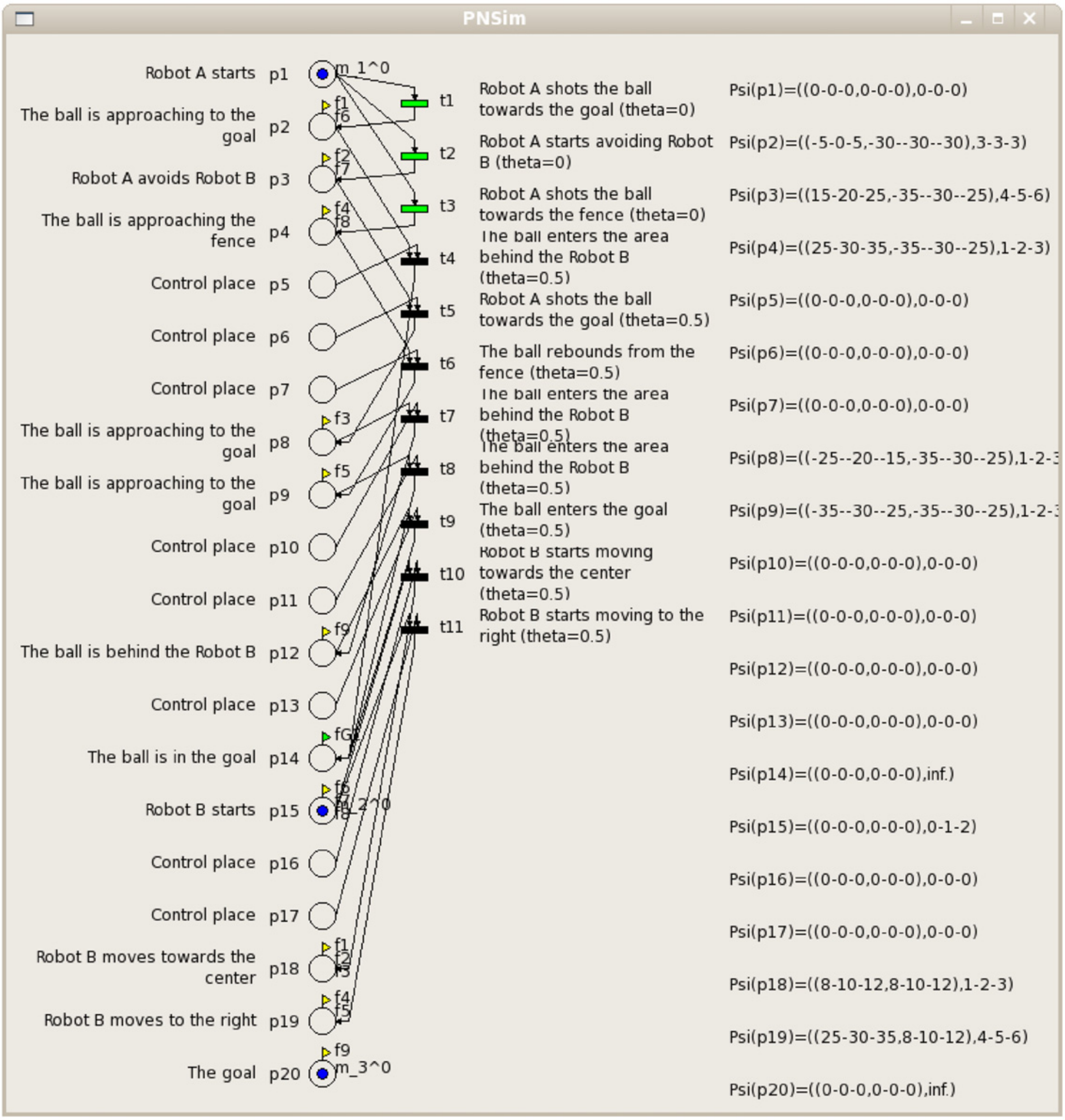

Fig. 8. Graphical representation of the FuSpaT model during the simulation.

excluded from U. The transition t_6 becomes enabled because the degree of confidence of control token is greater than 0.4, and it automatically fires. In the subsequent steps, the relationship str_5 (specified by the flag f_5) is evaluated by the *STLM*, the control token having $\kappa = 0.5$ is put into the control place p_{11} and the fuzzy S-T token m_1^3 finally reaches the place p_{14}, denoted as a goal place. Therefore, it can be inferenced that the strategy (c) leads to the goal. By resetting the simulation and trying other strategies, the user can find that other strategies do not lead to the goal for the same initial conditions.

6. Conclusion

The original high-level Petri nets, called Petri nets with fuzzy spatio-temporal tokens (PeNeFuST) are used for the modeling, planning and analyzing activities in spatio-temporal domains. Based on the PeNeFuST and the knowledge-representation scheme called FuSpaT, the model of the fuzzy spatio–temporal knowledge representation and reasoning is proposed. This model is used to build a knowledge base and support fuzzy spatio-temporal reasoning for problems that require the integration of both spatial and temporal information.

The proposed FuSpaT scheme uses the spatio-temporal logical module (STLM), which is implemented as a look-up table that integrates the spatial and temporal information obtained from the spatial (SLM) and temporal (TLM) logical modules. In our model, the STLM supports 117 spatio-temporal relationships (9 spatial relationships $\times$ 13 temporal relationships). The proposed model supports a spatio-temporal data-driven reasoning process, and unified representations of different temporal, as well as spatial and spatio-temporal information, the ability to use

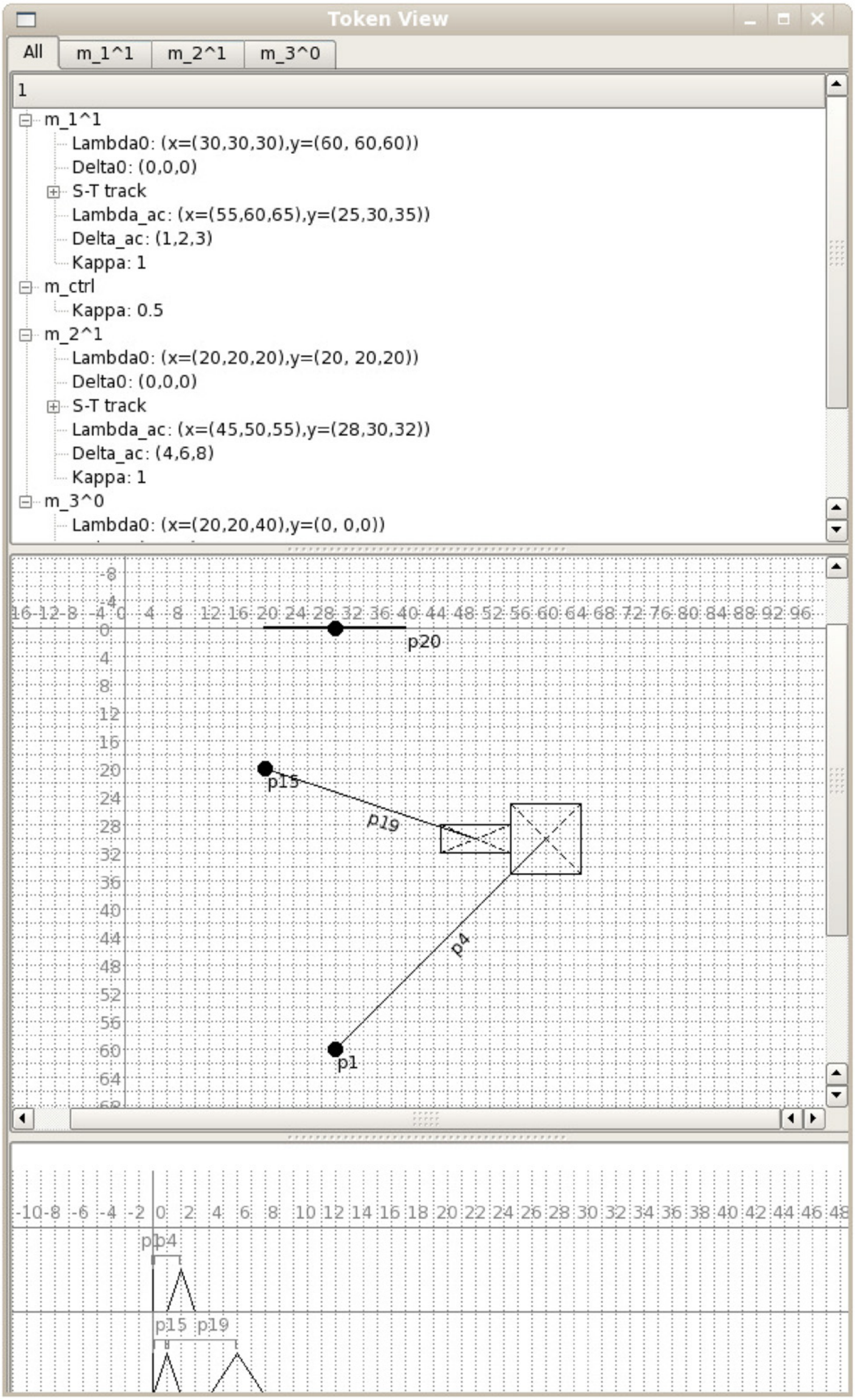

Fig. 9. A *Token-view* window of the simulator for the situation after the firing of the transition t_{11}.

linguistic variables to represent user or expert belief in the truth of the temporal, spatial and/or spatio-temporal relationships, the ability of independent modeling of activities for each of the agents, and the specifying of their interactions by means of the flags. One of the advantages is the existence of well-defined methods (based on the modified PN theory) for the analysis of the different spatial and temporal relationships among the agents or objects by changing the initial marking and spatial and time values assigned to the S-T tokens and places. The proposed model allows a hierarchical representation of the scenes on the different abstraction levels (based on well-known concepts of the PN called refinement and abstraction). It is domain independent, but suitable for the independent modeling of multiple agents in multi-agent systems. Based on the proposed theory and model of fuzzy spatio-temporal knowledge representation and inference, the program simulator and tools for the analysis have been developed in the C++ environment, enabling the use of the model in different application areas. Future work will consist of a further experimental validation of the proposed model.

References

[1] F. Wolter, M. Zakharyaschev, Qualitative Spatiotemporal Representation and Reasoning: A Computational Perspective, Morgan Kaufmann, Amsterdam, 2003, pp. 175–215.

[2] K. Hornsby, M. Egenhofer, Identity-based change: a foundation for spatio-temporal knowledge representation, International Journal of Geographical Information Science 14 (3) (2000) 207–224.

[3] M. Erwig, M. Schneider, Spatio-temporal predicates, IEEE Transactions on Knowledge and Data Engineering 14 (4) (2002) 881–901.

[4] N. Van de Weghe, A.G. Cohn, P. de Maeyer, F. Witlox, Representing moving objects in computer-based expert systems: the overtake event example, Expert Systems with Applications 29 (2005) 977–983.

[5] N. Van de Weghe, A.G. Cohn, G. de Tre, P. de Maeyer, A qualitative trajectory calculus as a basis for representing moving objects in geographical information systems, Control and Cybernetics 35 (1) (2006) 97–120.

[6] B. Bennett, D.R. Magee, A.G. Cohn, D.C. Hogg, Using spatio-temporal continuity constraints to enhance visual tracking of moving objects, in: L.S.R.L. de Mántaras (Ed.), 16th European Conference on Artificial Intelligence (ECAI 2004), 2004, pp. 922–926.

[7] B. Bennett, D. Magee, A. Cohn, D. Hogg, Enhanced tracking and recognition of moving objects by reasoning about spatio-temporal continuity, Image and Vision Computing 26 (1) (2008) 67–81.

[8] A. Del Bimbo, E. Vicario, D. Zingoni, Symbolic description and visual querying of image sequences using spatio-temporal logic, IEEE Transactions on Knowledge and Data Engineering 7 (4) (1995) 609–621.

[9] J. Fernyhough, A.G. Cohn, D.C. Hogg, Constructing qualitative event models automatically from video input, Image and Vision Computing 18 (2) (2000) 81–103.

[10] G. Jeon, J. Jeong, J. Lee, J. You, C. Wu, A spatio-temporal fuzzy interpolation algorithm for video deinterlacing, in: IEEE International Conference on Image Processing, 2006, pp. 1789–1792.

[11] X. Lan, D.P. Huttenlocher, A unified spatio-temporal articulated model for tracking, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04), 2004, pp. 722–729.

[12] P. Matsakis, J. Keller, L. Wendling, J. Marjamaa, O. Sjahputera, Linguistic description of relative positions in images, IEEE Transactions on Systems, Man and Cybernetics, Part B 31 (4) (2001) 573–588.

[13] C.-B. Shim, Y.-W. Shin, Spatio-temporal modeling of moving objects for content- and semantic-based retrieval in video data, in: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), 9th International Conference on Knowledge-Based Intelligent Information & Engineering Systems, Lecture Notes in Artificial Intelligence, vol. 3684, Springer, 2005, pp. 343–351.

[14] K. Walker, A.C. Esterline, Fuzzy motion planning using the Takagi–Sugeno method, in: Proceedings of the IEEE Southeastcon 2000, 2000, pp. 56–59.

[15] W. Al-Khatib, Y.F. Day, A. Ghafoor, P.B. Berra, Semantic modeling and knowledge representation in multimedia databases, IEEE Transactions on Knowledge and Data Engineering 11 (1) (1999) 64–80.

[16] S.-C. Chen, R. Kashyap, A spatio-temporal semantic model for multimedia database systems and multimedia information systems, IEEE Transactions on Knowledge and Data Engineering 13 (4) (2001) 607–622.

[17] P.-Y. Hsu, Y.-B. Chang, Y.-L. Chen, Strpn: a Petri-net approach for modeling spatial-temporal relations between moving multimedia objects, IEEE Transactions on Software Engineering 29 (1) (2003) 63–76.

[18] M. Vazirgiannis, Y. Theodoridis, T. Sellis, Spatio-temporal composition and indexing for large multimedia applications, Multimedia Systems 6 (1998) 284–298.

[19] S. Dragicevic, D.J. Marceau, An application of fuzzy logic reasoning for GIS temporal modeling of dynamic processes, Fuzzy Sets and Systems 113 (2000) 69–80.

[20] Z. Yin, A multi-scale GIS database model based on Petri net, in: J. Jiang (Ed.), ISPRS Workshop on Service and Application of Spatial Data Infrastructure, 2005, pp. 271–275.

[21] P. Fisher, T. Cheng, J. Wood, Higher order vagueness in geographical information: empirical geographical population of type n fuzzy sets, Geoinformatica 11 (2007) 311–330.

[22] E. Guilbert, H. Lin, A new model for cloud tracking and analysis on satellite images, Geoinformatica 11 (2007) 287–309.

[23] S. Schockaert, M. De Cock, Reasoning about vague topological information, in: 16th ACM Conference on Information and Knowledge Management, 2007, pp. 593–602.

[24] A.G. Cohn, S.M. Hazarika, Qualitative spatial representation and reasoning: an overview, Fundamenta Informaticae 43 (2001) 2–32.

[25] L.A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.

[26] D. Dubois, H. Prade, Processing fuzzy temporal knowledge, IEEE Transactions on Systems, Man and Cybernetics 19 (4) (1989) 729–744.

[27] A.G. Cohn, N.M. Gotts, The 'egg-yolk' representation of regions with indeterminate boundaries, in: P. Borrough, A.M. Franl (Eds.), GISDATA Specialist Meeting on Geographical Entities with Undetermined Boundaries, 1996, pp. 171–187.

[28] J. Ferber, Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence, Addison Wesley, Reading, Massatchusetts, 1999.

[29] G. Weiss, Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, The MIT Press, Cambridge, Massatchusetts, 2000.

[30] V. Subrahmanian, P. Bonatti, J. Dix, T. Eiter, S. Kraus, F. Ozcan, R. Ross, Heterogeneous Agent Systems, The MIT Press, Cambridge, Massatchusetts, 2000.

[31] B. Bennett, A. Cohn, F. Wolter, M. Zakharyaschev, Multi-dimensional modal logic as a framework for spatio-temporal reasoning, Applied Intelligence 17 (3) (2002) 239–251.

[32] M. Ragni, S. Wölfl, Temporalizing cardinal directions: from constraint satisfaction to planning, in: J.M.P. Doherty (Ed.), Knowledge Representation Conference (KR 2006), 2006, pp. 472–480.

[33] N. Van de Weghe, B. Kuijpers, P. Bogaert, P.D. Maeyer, A qualitative trajectory calculus and the composition of its relations, in: A. Rodriguez, I. Cruz, M. Egenhofer, S. Levashkin (Eds.), 1st International Conference on Geospatial Semantics (GeoS), Lecture Notes in Computer Science, vol. 3799, Springer, 2005, pp. 60–76.

[34] N. Van de Weghe, G.D. Tre, B. Kuijpers, P.D. Maeyer, The double-cross and the generalization concept as a basis for representing and comparing shapes of polylines, in: E. Zimanyi, E. Stefanakis (Eds.), 1st International Workshop on Semantic-Based Geographical Information Systems (SeBGIS), Lecture Notes in Computer Science, vol. 3770, Springer, 2005, pp. 1087–1096.

[35] A. Gomaa, N. Adam, V. Atluri, Color time Petri net for interactive adaptive multimedia objects, in: Y.-P.P. Chen (Ed.), 11th International Multimedia Modelling Conference (MMM'05), IEEE Press, 2005, pp. 147–157.

[36] K. Jensen, An introduction to the theoretical aspects of coloured Petri nets, Lecture Notes in Computer Science, vol. 803, Springer, 1994, pp. 230–272.

[37] K. Jensen, A brief introduction to coloured Petri nets, in: E. Brinksma (Ed.), TACA'97 Workshop: Tools and Algorithms for the Construction and Analysis of Systems, Lecture Notes in Computer Science, vol. 1217, Springer, 1997, pp. 203–208.

[38] K. Jensen, An introduction to the practical use of coloured Petri nets, Lecture Notes in Computer Science, vol. 1492, Springer, 1998, pp. 237–292.

[39] J.L. Peterson, Petri Net Theory and Modeling of Systems, Prentice Hall, Englewood Cliffs, 1981.

[40] S. Ribarić, T. Hrkać, A knowledge representation and reasoning based on Petri nets with spatio-temporal tokens, in: International Conference on Computer As a Tool, EUROCON 2007, 2007, pp. 793–800.

[41] M. Koprulu, N.K. Cicekli, A. Yazici, Spatio-temporal querying in video databases, Information Science 160 (1–4) (2004) 131–152.
[42] O. Sjahputera, P. Matsakis, J. Keller, R. Bondugula, Linguistic descriptions of an object in motion, in: 2002 Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS 2002), 2002, pp. 243–248.
[43] R.E.O. Schultz, T.M. Centeno, M.R. Delgado, Spatio-temporal prediction by means of fuzzy rule-based approach, in: 2006 IEEE International Conference on Fuzzy Systems, 2006, pp. 1271–1278.
[44] M.-C. Su, A fuzzy rule-based approach to spatio-temporal hand gesture recognition, IEEE Transactions on Systems, Man and Cybernetics—Part C: Applications and Reviews 30 (2) (2000) 276–281.
[45] S. Zaboli, S. Naderi, A.M.E. Moghaddam, Application of image mining for knowledge discovery of analyzed traffic images, in: IEEE International Conference on Industrial Technology (ICIT 2006), 2006, pp. 1066–1070.
[46] G. Ligozat, Reasoning about cardinal directions, Journal of Visual Languages and Computing 9 (1998) 23–44.
[47] J. Allen, Maintaining knowledge about temporal intervals, Communications of the ACM 26 (11) (1983) 832–843.
[48] C. De Runz, E. Desjardin, E. Piantoni, M. Herbin, Using fuzzy logic to manage uncertain multi-modal data in an archaeological GIS, in: 5th International Symposium on Spatial Data Quality, 2007, pp. 114–119.
[49] A. Kaufmann, M. Gupta, Introduction to Fuzzy Arithmetic, Van Nostrand Reinhold, New York, 1991.
[50] S-M. Chen, J.-S. Ke, J.-F. Chang, Knowledge representation using fuzzy Petri nets, IEEE Transactions on Knowledge and Data Engineering 2 (3) (1990) 311–319.
[51] S. Ribarić, T. Hrkać, TeMAS—a multi-agent system for temporally rich domains, Knowledge and Information Systems 15 (1) (2008) 1–30.
[52] G. Nagypál, B. Motik, A fuzzy model for representing uncertain, subjective and vague temporal knowledge in ontologies, International Conference on Ontologies, Databases and Application of Semantics (ODBASE) 2003, Lecture Notes in Computer Science, vol. 2888, Springer, 2003, pp. 906–923.
[53] D. Dubois, H. Prade, Fuzzy Sets and Systems, Theory and Applications, Academic Press, New York, 1980.
[54] D. Papadias, Y. Theodoridis, Spatial relations, minimum bounding rectangles, and spatial data structures, International Journal on Geographic Information Systems 11 (2) (1997) 111–138.
[55] D.A. Randell, Z. Cui, A. Cohn, A spatial logic based on regions and connection, in: B. Nebel, C. Rich, W. Swartout (Eds.), The 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR'92), Morgan Kaufmann, San Mateo, California, 1992, pp. 165–176. ⟨citeseer.ist.psu.edu/randell92spatial.html⟩.
[56] W. Reisig, A Primer in Petri Net Design, Springer Verlag, Berlin, 1992.