Plausible Event-Tree Networks for Knowledge Representation in Real-Time GIS-Based Decision Support Systems

Volodymyr Sherstjuk, Maryna Zharikova

Kherson National Technical University, Kherson, Ukraine
vgsherstyuk@bigmir.gmail.com, marina.jarikova@gmail.com

Abstract. Event-based knowledge representation models providing sufficient detail in space and time are often necessary for real-time GIS-based decision support systems. The paper is devoted to developing such a model based on a plausible event tree network, which is built over a spatial model of a terrain discretized with a grid of uniform-sized cells. Each event has not only time reference, but also spatial reference, and describes a transition of the cell from one state to another. It combines different kinds of likelihood assessments (probability, fuzzy, or rough) using various plausibility models. The paper describes the event tree network-based knowledge representation, which can be used to describe a multitude of interacting processes on the terrain. An experiment based on real data describing a forest fire cascade has been conducted and has confirmed the validity and usability of the proposed model for the considered class of GIS-based decision support systems.

Keywords: Knowledge Representation Model, Event Tree Network, Likelihood, Spatial Configuration, Decision Support System

1 Introduction

Natural systems include a multitude of interacting processes, which evolve in space and time. Some of them are destructive and cause deaths, injuries, and a huge damage to property and infrastructure. Now, people face a problem of real-time decision making in conditions of natural destructive processes. However, developing the decision support systems (DSS) is a complex and non-trivial task because most of such processes arise unexpectedly, proceed fleetingly, evolve in space and time transiently, non-linearly, and have a stochastic nature. Processes of a destructive nature distributed over a confined terrain give rise to a variety of hazards, threats, and risks to various objects [1]. Often they can lead to emergencies. Thus, a GIS-based real-time DSS for the natural emergencies response is a topic of current interest.

Giving the fact that natural emergencies are poorly modeled and unpredictable, well-studied classical decision support approaches cannot be used for such kind of processes

[2]. The efficiency of response operations strongly depends on the availability of observations of destructive processes, as well as on the validity and usefulness of the observed information representation.

The most frequently observed information is represented as event streams, which represent series of time-stamped events [3]. Usually, event sequence analysis looks at the sequence of events and time gaps between events [4].

Knowledge representation about a multitude of events occurring jointly and simultaneously has been studied in many fields of knowledge. The greatest number of formal ways of presenting knowledge about events was proposed in the field of natural language processing, such as Rich Entities, Relations, and Events (Rich ERE); Light Entities, Relations, and Events (Light ERE); Event Nugget (EN); Event Argument Extraction (EAE); Richer Event Descriptions (RED); and Event-Event Relations (EER) [5]. Those representation models are focused on event processing and event-event relations aimed at inference, causal relationships, and anomaly detection across several languages [6].

Event-based structures are considered as building blocks for creating and updating situation models related to comprehension [7]. The definition of an event is "a segment of time at a given location that is conceived by an observer to have a beginning and an end" [8]. All of the above-mentioned approaches are based on semantic meaning, use strictly-defined notions of events, time and space and do not use the hierarchical structures of time and space necessary for GISs-based real-time DSSs [9].

Approaches, where representations do not take on semantic meaning, include Causal events, Force dynamics, Stochastic Context-Free Grammars, and Spatio-Temporal Derivatives [10]. Despite the fact that they differ in methods, they employ the event structure representations, which are very hard for decision-maker interpretation.

Taking into account the nature of the events, we can emphasize the existence of two basic approaches, probabilistic and non-probabilistic. Probabilistic approaches include Variance propagation, Monte Carlo sampling, variations of the dynamic Bayesian network, Hidden Markov Models, and imprecise probabilities [11]. Non-probabilistic methods include those based on fuzzy sets theory and possibility theory [12].

However, an insufficiency of statistical data for probabilistic models, a lack of well-known membership functions or degree of possibility for non-probabilistic models, as well as their common feature of a high computational complexity prevent their efficient use. Extensive reviews on this topic with respect to a considered class of DSSs have been presented in the literature [13, 14].

The event trees allow modeling of a sequence of events, forming the structures of any level of complexity [15]. The event trees can be adapted to different ways of uncertainty accounting (probabilistic, fuzzy, rough, etc.). The limitation of the event trees used in the existing works lies in the fact that the events are referenced to time points rather than to spatial locations. However, this method is very flexible, is open to using hierarchical structures and have a big potential for evolution.

The above-mentioned review enables to conclude that existing event-based knowledge representation models correspond very weakly to the systems of the considered class, and do not provide the acceptable efficiency of DSS. This paper is dedicated to developing the event-tree knowledge representation model, where events are referenced to certain time and locations and can be organized into hierarchical structures with respect to time and space. Such model should be suitable for solving the problems of decision-making on natural emergencies response in real-time GIS-based DSS providing required efficiency.

2 Spatial Model

Assume that the destructive processes spread over a certain area of interest (AOI).

Consider a three-dimensional Euclidean space C, which contains the AOI as an openly connected subspace $X \subseteq C$. Suppose that each point $x \in X$ has a non-empty finite set of attributes $A = \{a_1,...a_m\}$, V_{a_i} is a domain of $a_i \in A$, $V = \cup_{a_i \in A} V_{a_i}$, and f is an attribute value function such that $f : X \times A \to V$ for each $x \in X$.

A spatial model is discretized at three levels: the lower level contains cells of equal size, the middle level consists of spatial regions of different sizes, and the upper level represents large spatial areas.

At the lower level, we impose a metrical grid of coordinate lines with size δ on C using a linear map ϕ such that coordinate lines form a set D of the cubic cells with the size being $\delta \times \delta \times \delta$, $\phi : D \to C$. Thus, space C is discretized by a grid $D = \{d_{xyz}\}$ of isometric cubic cells d_{xyz}. A cell $d \in D$ is a spatial object of a minimal size. Each cell $d \in D$ is associated with a set of attribute values, which is called the cell state, via the value function $f(d, A)$. The proposed discretization assigns the equal values of the attributes to each point belonging to a certain cell d, therefore each cell $d \in D$ represents a homogeneous area of the AOI in the sense of the attribute values A. Thus, each cell $d \in D$ can be reduced to a point of X, and all points of this cell are A-indiscernible: $(\forall d_1, d_2 \in D)(\forall a \in A)\left[f(d_1, a) = f(d_2, a)\right]$.

At the middle level, the subspace X can also be divided into a finite set of disjoint objects having geometric shapes, which represent the certain homogeneous areas of the AOI. Consider a non-empty subset of attributes $A_i \subseteq A$. Define an A_i-indiscernibility relation [10] $R_D^A = \left\{(d_m, d_n) \in D \times D \middle| \forall a_j \in A_i, f(d_m, a_j) = f(d_n, a_j)\right\}$ on the set of cells D. If $(d_m, d_n) \in R_D^A$, it means that all pairs of different points y, z that belong to the different cells d_m, d_n have the same values of attributes $a_j,...a_m \in A_i$ as all pairs of different points x, y of each cell d_m, d_n. Thus, we define a middle-level structural element of the spatial model as the homogeneous spatial area that is uniform in the sense of attribute's values and can be represented by the approximating set of cells. Such element is called a region and denoted by h. All the cells belonging to the region h are A_i-indiscernible. Each region can represent the object of a certain class on a GIS map that is named as geotaxon. The geotaxons cannot overlap or cover one another, but they can be adjacent or adjoin to one another. Thus, a continuity and a connectivity are their important features (spatial concentration of the underlying cells).

However, we often need to analyze the spatial areas containing a plurality of objects with the certain relations between them. Such spatial areas may consist of a plurality of separate regions spatially distributed over X, and represent zones homogenous in the sense of definite assessments of some indicators (e.g. danger, threat, and risk), which depend on the values of attributes $A_j \subseteq A$. Obviously, they do not have the property of the continuity. Consider a set of regions $H = \{h_1,...h_k\}$. Define the A_j-indiscernibility relation $R_H^{A_j} = \left\{\forall h_l, h_q \in H,\ \forall d_m, d_n \in D,\ \exists d_m \in h_l, d_n \in h_q \middle| \forall a_k \in A_j, f\left(d_m, a_k\right) = f\left(d_n, a_k\right)\right\}$ on the set of regions H [16]. Obviously, all regions belonging to $R_H^{A_j}$ are A_i-indiscernible in the sense of the same values of attributes $a_l,...a_p \in A_j$. Thus, we define distributed spatial area H as an upper-level structural element of the spatial model that is uniform in the sense of attribute's values and is represented by the approximating set of regions.

3 Events and States

Suppose the set of attributes A can be divided into subsets: not changing over time (static) attributes A_S, time-varying (dynamic) attributes A_D, slowly changing (environmental) attributes A_E, $A = A_S \cup A_D \cup A_E$. We next define the cell state categories.

Suppose $W = \{w_0,...w_i,...w_F\}$ is an ordered set of the cell state categories (modes), where w_0 is the initial mode, w_F is the final mode, and w_i is the transitional mode. Suppose ϑ is a category function such that $\vartheta : D \times A \to W$. Each category $w \in W$ has three subcategories: the cell status $w_S = \vartheta(A_S)$, the cell condition $w_C = \vartheta(A_S \cup A_E)$, and the cell stage $w_D = \vartheta(A_D)$. Each state category (and subcategory) is a certain subspace of the n-dimensional attribute values space $V_{a_1} \times ... \times V_{a_i} \times ... \times V_{a_n}$.

Each random change of values of any subset of dynamic parameters $A_k \subseteq A_D$ can change the cell condition w_C in such a way that the cell stage w_D must also change. This change is not necessary, but possible. If the cell condition w_C changes, the cell possibly goes into another state category w_i (mode of behavior).

We define an accessibility relation $R_{ACC} \subseteq W_D \times W_D$ (reflexive, asymmetric, and non-transitive) on a set of stages W_D and a compatibility relation $R_{COM} \subseteq W_C \times W_D$ (non-reflexive, symmetric, and non-transitive), taken the cell condition $w_C \in W_C$ onto the cell stage $w_D \in W_D$. The accessibility can be determined by a function $f_{acc} : W_D \times W_D \to \{\xi\}$, which returns the possibility of the transition from one stage to another, whereas the compatibility can be determined by a function $f_{com} : W_C \times W_D \to \{\mathit{true, false}\}$ taking into account that some stages may be incompatible with the certain conditions (e.g., sandy areas, which are not covered with vegetation, usually cannot be exposed to burning).

We consider each significant (perhaps, jump-like [17]) change of the cell attribute's value, which forces the cell to change its state, as an event, and denote it by y, so that $y : w_i \to w_j$. It is clear that the model of the destructive process can be represented as a model of dynamic change of states of a subset of cells covered by the process within the spatial model. Assume, during the destructive process, the cell moves through a sequence of qualitatively different classes (categories) of states. The states should be evaluated during continuous observations (monitoring) that allow obtaining time-ordered sequences of events (random event flow without consequences).

We formalize the event, the events model, and the event stream, taking [18] as a basis and introducing parameters of time and space into the model.

4 Event Model

Consider a time point set T with the initial moment $t_0 \in T$ and an order relation $<_T$, which sets a fully ordered timescale $\langle t_0, T, <_T \rangle$ over T.

In order to consider various aspects of a recorded event depending on the accuracy of its observation, it is possible to classify the events using taxonomy hierarchy. This hierarchy is denoted by I_1 and corresponds to a set of classes $Class = \{c_i\}_{i=1}^{n}$, where c_i is an event class. A partial order relation $\prec_1$ over I_1 arranges observed information in order from abstract to detailed, e.g. $c_1 \prec_1 c_2$ means that the event class c_1 contains less information than c_2.

Notice, that any event can be a part of one or more complex event structures, such as coupled, concatenated, or triggered events and their chains. Thus, it is advisable to build a composition hierarchy I_2 that corresponds to a certain composition of events such as $y_j = y_k \,\text{B}\, y_l$, where B is disjoint union. The partial order relation $\prec_2$ over I_2 arranges the inclusion of the observed events in the complex chains, where $y_k \prec_2 y_j$, means that y_j is a composition with y_k being its constituent.

Besides that, we can build a spatial hierarchy I_3 within the spatial model with a set of elements like {*cells*, *regions*, *areas*} and the partial order relation $\prec_3$ over it, as well as a time hierarchy I_4 with the set of elements like {*seconds*, *minutes*, *hours*, *days*...} and a full-order relation $<_T$ over it. The spatial and temporal hierarchies are the basis for building the adequate space-and-time-referenced event model.

Thus, the basic element for developing the event model is the hierarchy.

Suppose each hierarchy $\Im_i$ is a triple: $\Im_i = \langle \perp_i, I_i, \prec_i \rangle$, where I_i is a set of some elements, each of which corresponds to a certain relation υ_i among them (e.g. taxonomic υ_1, inclusion υ_2, spatial υ_3, temporal υ_4, accessibility υ_5, compatibility υ_6, and so on), $\prec_i$ is the order relation over I_i, and $\perp_i$ is the least element of $\prec_i$.

Define an event signature $\mathfrak{z}$ as a tuple $\mathfrak{z} = \left\langle A, \{\Im_i\}_{i=1}^{m} \right\rangle$, where A is a set of parameters, and $\{\Im_i\}_{i=1}^{m}$ is a set of hierarchies $\Im_i$ induced by the corresponding relations υ_i.

Consider the event model $E = \langle \upsilon, \rho, \mathfrak{z} \rangle$, where υ is a variable set, ρ is a set of restrictions, and $\mathfrak{z}$ is the signature.

Define the event y in the model E as a structure $\mathrm{y} = \langle Y, c, t, d, A_{\mathrm{y}} \rangle$, where $Y \in E.\upsilon$ is a unique label, $c \in E.\mathfrak{Z}.I_1$ is an event class, $t \in T$ is a time point (observation moment, $t \in E.\mathfrak{Z}.I_4$), $d \in D$ is a spatial point (observation georeference, $d \in E.\mathfrak{Z}.I_3$), and A_{y} is a set of event parameters, $A_{\mathrm{y}} \in A$. Thus, the event y is a complex object that belongs to a certain class of events $c \in Class$ and gets the values of parameters of the corresponding cell $d \in D$, which contains the georeference point. Since the conditions and the accuracy of observation differ at various time points, each event y can be described by the parameters that also differ in values and accuracy.

An event sequence S in the model E is an aggregate of events ordered by $<_T$, $S = [\mathrm{y}_1, \mathrm{y}_2, \ldots \mathrm{y}_n]$, such that it holds $\mathrm{y}_1.t \le_T \mathrm{y}_2.t \le_T \ldots \le_T \mathrm{y}_n.t$ for all events. One should refer to the event y of the sequence S_i with the index j as y^i_j, $\mathrm{y}_i.t = t_j \in T$.

To formalize any kinds of relations established between the events and the variables in the event model E, we introduce the notion of an event path. The event path $\rho(d_i, d_j)$ is a sequence of events spreading along a chain from the cell d_i to the cell d_j. Two or more paths can be combined using a concatenation operator.

5 Assessment of Event Likelihood

Suppose L is a non-empty carrier set with multiplicative $\otimes$ and additive $\oplus$ operators. If they satisfy the conditions of idempotency, commutativity, associativity, and distributivity for any $x, y, z \in L$, we obtain a distributive quasi-lattice $R = \langle L, \otimes, \oplus \rangle$. Setting two zero-ary operations 0 and 1, as well as their absorption conditions, we obtain a bounded distributive lattice (semiring) $Z = \langle L, \otimes, \oplus, 0, 1 \rangle$ that is not closed with respect to the union [19]. Therefore, $\langle L, \otimes, 1 \rangle$ is a monoid, $\langle L, \oplus, 0 \rangle$ is a commutative monoid.

Taking into account the idempotency of the additive operator $\oplus$ with respect to L and the properties of the semiring Z, we can define an order relation $\prec_Z$ such that $\forall x, y, z \in L \quad x \prec_Z y \leftrightarrow x \oplus z = y$. Thus, $\prec_Z$ is the partial order on L that makes it possible to compare the various elements of the carrier set, so that $x \prec_{\mathrm{P}} y$ means that the value y is preferable than x.

The proposed semiring model and the partial order relation can serve us as the framework. Choosing the corresponding initial carrier set L, defining the additive $\oplus$ and multiplicative $\otimes$ operators over Z, and equipping the lattice with operations of taking the exact lower edge *inf* and upper edge *sup*, we can build a likelihood model ℓ, which can express the degree of likelihood as a measure of belonging to the carrier set.

Thus, we use a relative likelihood assessment on the scale [0,1] instead of probability to evaluate events as more possible, less possible, equally possible based on the numerical value (degree) of their likelihood. It makes possible to combine estimates of probabilities based on statistical observations, with assessments of the possibilities formed by experts, within the single framework, taking into account the possible incompleteness and inaccuracy of information represented by fuzzy or rough sets.

6 Plausible Event Tree Networks

Let us build an event tree network over the model E, based on the joint mapping of a set of induced hierarchies into the event tree structure [20].

Consider oriented connected multigraph $g = \langle v, e \rangle$, which doesn't contain cycles, where v is a set of vertices, and e is a set of arcs. Subdivide a nonempty set of vertices v into three disjoint sets: a set of leaf nodes $r \subset v$, a set of roots, which contains vertices of higher level $u \subset v$, and a set of nodes b, which are neither roots nor leaves, such that $g = u \cup b \cup r$, $u \cap b = u \cap r = b \cap r = \varnothing$. Subdivide a nonempty set of arcs e into subsets $e = e_{\prec \upsilon 1} \cup ... \cup e_{\prec \upsilon n}$, each of which reflects a certain relation υ_i over v.

We can use the likelihood model ℓ for labeling arcs with the coefficients of likelihood, which belong to the carrier set L. The presence of a transition (arc $e_{\prec \upsilon i} \in e$) between two nodes is a reflection of some relationship υ_i between the events represented by the nodes. Based on ℓ, the measures $\lambda_i \in L$ can be assigned to the arcs, expressing the degree of the existence of a relationship between two nodes connected by this arc.

The model ℓ (Fig.1) can be based on the deterministic formalism (*D*), or the probabilistic theory (*P*) for evaluation of probabilities of transitions between nodes, or the fuzzy set theory (*F*) for evaluation of possibilities of transitions between nodes, or the rough set theory (*R*) for evaluation of necessities/possibilities of the transition between nodes, of the temporal theory (*T*) for evaluation of transition times between nodes.

Likelihood model, ℓ	L	$\otimes$	$\oplus$	**0**	**1**
deterministic, D	$\{0,1\}$	$\wedge$	$\vee$	0	1
probabilistic, P	$[0,1]$	$\cdot$	$1-\prod_{i=1}^{k}(1-p_i)$	0	1
fuzzy, F	$[0,1]$	$\min_{i=1}^{k}(\mu_i)$	$\max_{i=1}^{k}(\mu_i)$	0	1
rough, R	$[0,1]\times[0,1]$	$\left(\bigcap_{i=1}^{k}\underline{d}_i, \bigcap_{i=1}^{k}\overline{d}_i\right)$	$\left(\bigcup_{i=1}^{k}\underline{d}_i, \bigcup_{i=1}^{k}\overline{d}_i\right)$	$\langle 0,0\rangle$	$\langle U,U\rangle$
time, T	$[0,\infty[$	$+$	$\min_{i=1}^{k}(t_i)$	0	∞
combined, PT	$[0,1]\times[0,\infty[$	$\left(\prod_{i=1}^{k}p_i, \sum_{i=1}^{k}t_i\right)$	$\left(1-\prod_{i=1}^{k}(1-p_i), \min_{i=1}^{k}(t_i)\right)$	$(0,0)$	$(1,\infty)$
combined, FT	$[0,1]\times[0,\infty[$	$\left(\min_{i=1}^{k}(\mu_i), \sum_{i=1}^{k}t_i\right)$	$\left(\max_{i=1}^{k}(\mu_i), \min_{i=1}^{k}(t_i)\right)$	$(0,0)$	$(1,\infty)$
combined, RT	$[0,1]\times[0,1]\times[0,\infty[$	$\left(\left(\bigcap_{i=1}^{k}\underline{d}_i, \bigcap_{i=1}^{k}\overline{d}_i\right), \sum_{i=1}^{k}t_i\right)$	$\left(\left(\bigcup_{i=1}^{k}\underline{d}_i, \bigcup_{i=1}^{k}\overline{d}_i\right), \min_{i=1}^{k}(t_i)\right)$	$(\langle 0,0\rangle,0)$	$(\langle U,U\rangle,\infty)$

Fig. 1. The likelihood models

A composite likelihood model ℓ^* is a model built on the base of models ℓ_i and ℓ_j so that the semi-ring Z^* of the compositional model is the Cartesian product of semi-rings $Z^* = Z_i \times Z_j$. A complex likelihood model μ is a shared set $\{\ell_i\}_{i=1}^{n}$ of simple and composite models ℓ_i.

Let us introduce a multigraph g over the model E.

The plausible event tree network (PETN) in the model E over the event stream $S=[y_1,...y_n]$ such that $y_i \in E.\mathcal{Y}$ is a structure $G=\left\langle g,\varphi,\gamma,\{\prec_i,\phi_i\}_{i=1}^{k},\tau,\delta,\mu\right\rangle$, where g is an acyclic connected multigraph, $\varphi: r \to S$ is a mapping of each leaf node r into a certain event from the sequence S, $\tau: r \to T$ is a mapping of each leaf node r into a set of time values $E.\mathcal{Y}.I_4$, $\delta: r \to X$ is a mapping of each leaf node r into a spatial location $E.\mathcal{Y}.I_3$, $\varsigma: r \to \rho$ is a mapping of each leaf node r into a certain restriction from a set of restrictions ρ, $\gamma: v \to 2^{E.\mathcal{Y}.A}$ is a mapping of each node v into a set of parameters from $E.\mathcal{Y}.A$, and $\{\prec_i\}_{i=1}^{k}$ is a set of partial order relations $\prec_i$ over a set of nodes v, each of which is induced by a relation υ_i and represented by a subset of arcs $e_{\prec\upsilon i} \in e$, μ is the complex likelihood model, $\phi_i : e_j \xrightarrow{\lambda_j} \ell$ is a mapping that labels each arc $e_j \in e_{\prec\upsilon 1}$ with a likelihood estimate λ_j within the model $\ell \in \mu$.

The root nodes $h \in u$ of PETN G determine a set of sequences consisting of all events in the leaf nodes $\mathrm{r} \in r$ ordered by $\{\prec_i\}$. Each leaf node $\mathrm{r} \in r$ corresponds to certain parameters γ, a time point τ, and a spatial location δ. Each non-leaf node represents a significant event sequence, where each event is represented by the leaf node $\mathrm{r} \in r$. Thus, each PETN node corresponds to the event sequence represented by h, and its descendants represent a certain chain of the parent sequence.

PETN can be used to formalize partially ordered sets of events that occur simultaneously and jointly. In this case, the leaf nodes represent the elementary (observed) events, and the non-leaf nodes correspond to the aggregate events (sequences of elementary events). An important feature of the formalized PETN is its ability to expanding the set of representable relations between events by any new relations υ_k adding a corresponding subset of arcs e_k to e and implementing a corresponding order relation $\prec_k$. A complex composition structure can be created in the presence of several event sequences within the model E and some different relations between them. Fortunately, the composition hierarchy I_2 allows us to consider PETN as a composite structure, which includes the other tree structures as elements.

A composite event tree network within the model E with respect to the event sequence set $\{S_j\}_{j=1}^{n}$, $S_j=[y_1^j,...y_m^j]$ such that $y_i^j \in E.\mathcal{Y}$ is a structure like $G^{\mathrm{e}} = G_1 \text{ в } ... \text{ в } G_n$. The structure G^{e} can be represented as a forest of poly-multi-trees. A fragment of the fuzzy-temporal (*FT*-)PETN is shown in Fig. 2.

We can notice that an aggregate event y_{111} is a sequence with an event y_{1111} and possibly (with a degree 0.8) the event y_{1112} following with an interval of 1 to 5 minutes. Also, if the event y_{111} is a part of an abstract event y_{11}, it is possible (with the degree 0.5) that event y_{112} should occur within an interval of 2 to 6 minutes.

The proposed formalization of the PETN can be considered as an abstraction of the known model of the Belief Network [14].

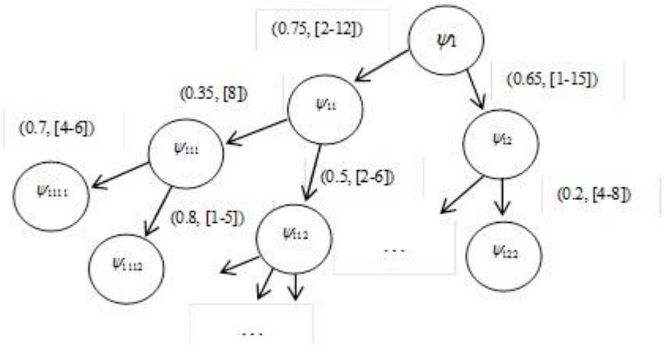

Fig. 2. A fragment of fuzzy-temporal (*FT*) PTEN

7 PETN Implementation

The proposed plausible event-tree networks were implemented as PETN dynamic library written in Python, which provides a high-level programming interface to the developer.

The PETN library includes a set of classes and methods that allow creating complex structures of events on the fly and maintaining them. The structure of the PETN Library is illustrated in Fig. 3.

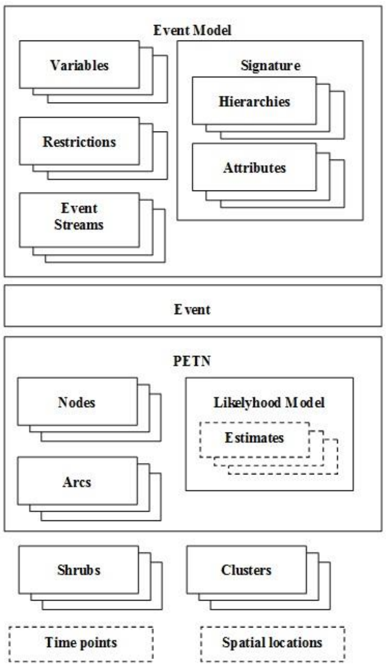

Fig. 3. Structure of PETN dynamic library

The main class of the library is the **Event** class. The sequences of events are collected in ordered lists that representing event streams. Thus, objects of the **event** class are building blocks for the objects of the **Event streams** class.

Event model class allows describing a multitude of event streams, variables, and constraints, as well as defining necessary hierarchies for events and setting their attributes. Thus, a certain event model environment can be constructed.

Hierarchies can be described as the ordered lists of elements, where the first element is always the least element.

PETN class describes interconnected sets of nodes and arcs, where each node object is associated with a certain event object from **Event model**, as well as each arc object gets its estimate from the corresponding likelihood model.

The main feature of the PETN object is that each event and, consequently, each node of PETN has reference to the corresponding time point and geolocation.

All classes of PETN library have a unified CRUD interface that implements the corresponding set of methods (create, read, update, and delete). In addition, all ordered lists of objects have an additional interface, which takes into account their indexing.

Classes **Shrubs** and **Clusters** are very important because they allow cutting and separately handling a certain part of the tree-like structure. The **Shrubs** class allows getting some node of PETN as root, and then pull out the entire set of nodes and arcs connecting them from the PETN structure. The resulting subtree growing from the root node can be further considered as an independent PETN structure with all the possibilities for its processing and analysis. On the contrary, the **Clusters** class allows to simultaneously specifying a non-empty set of root nodes, so the whole bundles of bushes growing from these roots can be processed and analyzed.

Objects of **Event** and **Event stream** classes can be serialized and deserialized using a data source associated with a certain PostgreSQL database. Objects of PETN class can also be serialized and deserialized but its data source is associated with another PostgreSQL database. Thus, event streams are accumulated and stored in one database, while the corresponding event-tree networks are grown and stored in another, separate database. This makes it much easier to monitoring events.

The goal of PETN Library is to make the development of PETN structures using Python as quick, flexible, and elegant as possible using its programmable interface.

However, for use in the DSS, it is necessary to implement an interface that allows the operator to perform certain operations with PETN structures without directly programming. Therefore, a PETN description and manipulation language (PETN-DML) interpreter were developed and used to translate requests of DSS into series of method calls for PETN objects through the PETN library.

8 PETN Model Implementation in the DSS

The proposed knowledge representation model based on the plausible event tree network was implemented using the PTEN Dynamic libraries.

To make the development of GIS-based DSS quick and flexible, it uses Python programming language as well as the following tools (Fig. 4):

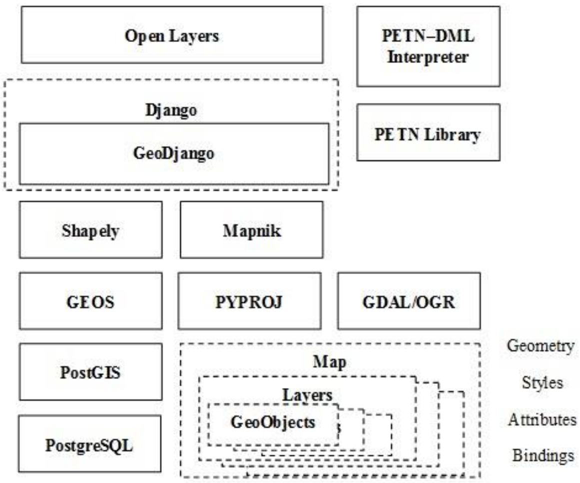

Fig. 4. Structure of the DSS implementation

- DBMS PostgreSQL;
- Geospatial extension PostGIS for PostgreSQL;
- PostgreSQL adapter psycopg2 for Python;
- Framework Django;
- GIS extension GeoDjango;
- GDAL/OGR geospatial libraries;
- GEOS Geometry Engine library;
- Mapnik – open source toolkit for rendering maps;
- pyproj – library for cartographic transformations and geodetic computations;
- Shapely – planar geometry library;
- PETN dynamic library;
- PETN interpreter;
- OpenLayers mapping library.

The web-oriented GIS-based DSS use the WGS84 datum and GeoJSON as an open format for encoding geographic data structures, as well as GML (geographic markup language) as an open XML-based standard for the exchange of GIS data.

The OGR library is used to read and write geodata in vector format, which was obtained from the Natural Earth website.

The Mapnik library takes the geodata from the PostGIS database and turns it into clearly visualized images. It provides a Map object, representing the map as a whole, Layer objects representing thematic layers with the content of the map, and Style objects that tell how to draw various layers.

Shapely is based on the dynamic library GEOS (the engine used in PostGIS) and manage spatial rather than geospatial data. It assumes that the geodata is already projected to a two-dimensional coordinate plane before they can be manipulated, and the results can be converted to geographic coordinates if desired.

The geospatial extension PostGIS is used for storing spatial data and working with them in the object-relational database PostgreSQL.

Using PostGIS and Mapnik, we ensure that the geospatial data is split into a large number of regular cells organized into rows and columns that are elements of a regular cubic grid in a certain data model. Thus, each cell is geographically referenced, that is, it has its own coordinates within the Map object.

Applying PostGIS from the Python programs, we have to use PostgreSQL DBMS as well as the psycopg2 PostgreSQL database adapter for Python.

9 The Results of the Research

The developed GIS-based real-time DSS is based on the PETN model implementation and allows evaluating a number of indicators, e.g. danger degrees, threats, and risks, for a given set of target objects, as well as providing the geospatial analysis of emergencies in real time disaster situations.

Fig. 5 depicts the map of Tsurupinsk forestry (Kherson region, Ukraine), which has been implemented using the proposed spatial model grid with the variable cell size.

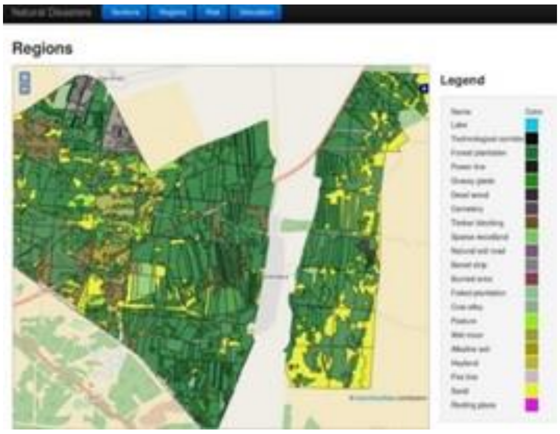

Fig. 5. Representation of the Tsurupinsk forestry in GIS-based DSS Forest Project

To examine a validity and an efficiency of the proposed PETN model, we have conducted an experiment based on the information on series of large-scale forest fires, which had been taken place in Tsyurupinsk and Golopristan forestries (Kherson region, Ukraine) on July 20-31, 2007.

We have modeled the ongoing processes via PETN and evaluated the total time of decision-making as well as the losses at the end of the processes. We have also varied the number of ignitions investigating its impact on the DSS assessment time, which enabled the evaluation of the influence of these parameters upon a risk assessment performance. The number of ignitions has been varied from 1 to 8, and the corresponding number of nodes in obtained PETN has been respectively varied from 1,438 to 18,582.

Although the performed experiment would benefit from comparing our results with results based on some alternative implementation, we did not have an adequate software library for comparison. Therefore, we have compared the results obtained with the implemented DSS, with the results obtained with manual decision-making.

The results of the experiment are depicted in Fig. 6.

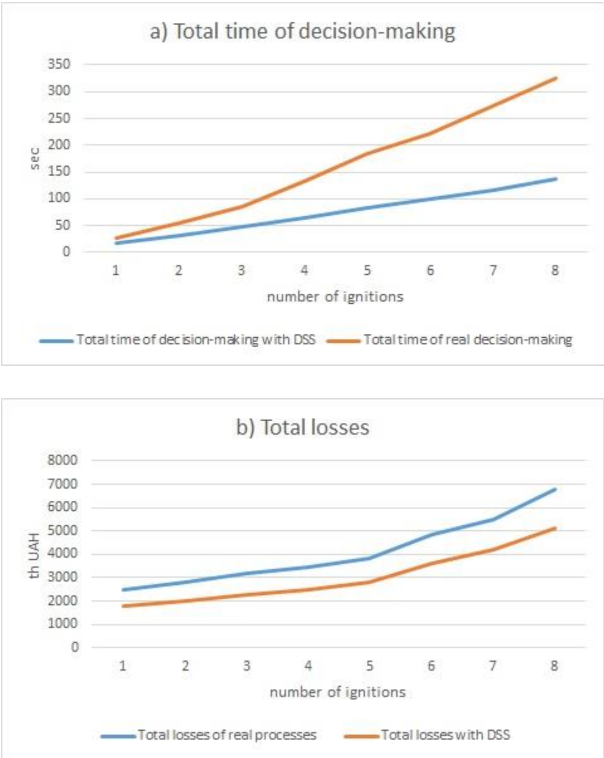

Fig. 6. Total time of decision making (a) and total losses (b) vs. number of ignitions

As our results indicate, the PETN model provides an opportunity to represent the dynamics of destructive processes adequately. The use of the proposed model makes it possible to accelerate the decision-making process under destructive process conditions by about 60% (for 5 ignitions) and above. The graph Fig. 6(a) shows the tendency to a significant increase of DSS efficiency with an increase of the number of ignitions (from 6 and above) as a consequence of the limitation of human heuristic capabilities (without DSS). The achieved acceleration of the decision-making process leads to decreasing the total losses by 35% and above in the same conditions.

A general view of the DSS performance curve in Fig. 6 shows that the dependence of the decision-making time on the ignition points number is linear. Since the dependence of the number of PETN nodes on the number of ignition points is also linear, it is confirmed that PTEN inference time depends linearly on the time with respect to the number of nodes.

10 Conclusions

The knowledge representation model based on the plausible event tree networks is built as a result of conducted research. The model represents a sequence of cell state transitions considered as events within the spatial model. Unlike the other known approaches, PETN allows referencing events to both the time and spatial points over the digital terrain model. Thus, we can closely integrate the proposed PETN model into GIS-based DSS, and, as a result, provide an adequate mapping of the dynamics of the spatially-distributed processes. Another important advantage of the proposed PETN model is its satisfactory performance, which ensures DSS operability in real time.

The proposed knowledge representation method based on plausible event tree networks also gives the opportunity to describe events based on incomplete and unreliable observed parameters as the likelihood assessments of transitions of the cell from one state to another. Unlike to well-known probabilistic or possibilistic approaches, PETN formalism can combine time measures and various likelihood measures (probability, possibility, fuzzy, or rough) in one frame depending on the specific conditions of the uncertainty of observations.

The proposed PETN model is applicable in all domains, which can be considered as a multitude of interacting spatially-distributed processes evolving in space and time. Such processes often give rise to a danger and risk due to dynamic locations and dynamic spatial relationships of interacting objects. In most cases, this causes their destructions and can lead to critical situations or emergencies. Solving such problems requires real-time GIS-based DSS containing a digital model of confined space (terrain). Thus, the proposed PETN model can be the basis of real-time GIS-based DSSs.

The proposed PETN model has been used in GIS-based DSS for a natural emergency response. Currently, intensive research is also being conducted on the use of the PETN model for solving problems of safety assessment in vehicle-onboard control systems and threat assessment in computer security systems. Research results show that the proposed approach significantly reduces the computational complexity of problem-solving in the above-mentioned domains.

References

1. Zharikova, M., Sherstjuk, V.: Threat assessment method for intelligent disaster decision support system. Advances in Intelligent Systems and Computing 512, 81–99 (2016). doi: 10.1007/978-3-319-45991-2_6
2. Newman, J., Maier, H., Riddell, G., Zecchin, A., Daniell, J., Schaefer, A., van Delden, H., Khazai, B., O'Flaherty, M., Newland, C.: Review of literature on decision support systems for natural hazard risk reduction: current status and future directions. Environmental Modelling & Software 96(C), 378–409 (2017). doi: 10.1016/j.envsoft.2017.06.042
3. Wongsuphasawat, K., Guerra Gómez, J. A., Plaisant, C., Wang, T., Taieb-Maimon, M., Shneiderman. B.: LifeFlow: visualizing an overview of event sequences. In: SIGCHI Conf. on Human Factors in Comp. Systems, pp. 1747–1756. New York, USA (2011). doi: 10.1145/1978942.1979196

4. Burch, M., Beck, F., Diehl, S.: Timeline trees: visualizing sequences of transactions in information hierarchies. In: Working conf. on Advanced visual interfaces, pp. 75–82. New York, USA (2008). doi: 10.1145/1385569.1385584
5. Bies, A., Song, Z., Getman, J., Ellis, J., Mott, J., Strassel, S., Palmer, M., Mitamura, T., Freedman, M., Ji, H., O'Gorman, T.: A Comparison of Event Representations in DEFT. In: 4th Workshop on Events: Definition, Detection, Coreference, and Representation, pp. 27–36. San Diego, California (2016). doi: 10.18653/v1/W16-1004
6. Broad Agency Announcement: Deep Exploration and Filtering of Text (DEFT). Defense Advanced Research Projects Agency, DARPA-BAA-12-47 (2012).
7. Ferretti, T. R., Kutas, M., McRae, K.: Verb aspect and the activation of event knowledge. Journal of Experimental Psychology: Learning, Memory, and Cognition 33(1), 182–196 (2007). doi: 10.1037/0278-7393.33.1.182
8. Zacks, J. M., Tversky, B.: Event structure in perception and conception. Psychological Bulletin 127(1), 3-21 (2001). doi: 10.1037/0033-2909.127.1.3
9. Sinha, C., Gärdenfors, P.: Time, space, and events in language and cognition: a comparative view. Annals of the New York Academy of Sciences, Flow of Time, 1–10 (2014). doi: 10.1111/nyas.12491
10. Manshadi, M., Swanson, R., Gordon, A.S.: Learning a probabilistic model of event sequences from internet weblog stories. In: 21st FLAIRS Conf., pp. 159–164 (2008).
11. Varacca, D., Völzer, H., Winskel, G.: Probabilistic Event Structures and Domains. In: CONCUR 2004 – Concurrency Theory. LNCS, vol. 3170, pp. 481–496. Springer (2004). doi: 10.1007/978-3-540-28644-8_31
12. Jacquin, A. P.: Possibilistic uncertainty analysis of a conceptual model of snowmelt runoff. Hydrology and Earth System Sciences 8(14), 1681–1695 (2010). doi: 10.5194/hess-14-1681-2010
13. Matott, L. S., Babendreier, J. E., and Purucker, S. T.: Evaluating uncertainty in integrated environmental models: A review of concepts and tools. Water Resour. Res., 45, W06421 (2009) doi: 10.1029/2008WR007301
14. Montanari, A., Shoemaker, C. A., van de Giesen, N.: Introduction to special section on Uncertainty Assessment in Surface and Subsurface Hydrology: An overview of issues and challenges, Water Resour. Res., 45, W00B00 (2009) doi: 10.1029/2009WR008471
15. Peila, D., Guardini, C.: Use of the event tree to assess the risk reduction obtained from rock fall protection devices. Natural Hazards Earth System Sciences 8, 1441–1450 (2008).
16. Pawlak, Z., Grzymala-Busse, J., Slowinski, R., Ziarko, W.: Rough sets. Communications of ACM 38(11), 88–95 (1995). doi: 10.1145/219717.219791
17. Sherstjuk, V., Zharikova, M.: Approximate model of spatially distributed Markov process for GIS-based decision support system. In: IEEE 12th Int. Conf. on Computer Sciences and Information Technologies (CSIT-2017), pp. 300–304. Lviv, Ukraine (2017). doi: 10.1109/STC-CSIT.2017.8098791
18. Martin, F.: Case-Based Sequence Analysis in Dynamic, Imprecise, and Adversarial Domains. Tesi doctoral, Universitat Politecnica De Catalunya, Barcelona (2004).
19. Wirsching, G., Huber, M., Kölbl, C.: The confidence-probability semiring. In: Report 2010-04, Institute of Informatics, Augsburg University (2010).
20. Rajangam, E., Annamalai, C.: Graph Models for Knowledge Representation and Reasoning for Contemporary and Emerging Needs – A Survey. International Journal of Information Technology and Computer Science 8(2), 14–22 (2016). doi: 10.5815/ijitcs.2016.02.02