DEVS formalism and vertex method-based fuzzy modeling

P.-A. Bisgambiglia, E. De Gentili, (Pr.) P.A. Bisgambiglia and J.F. Santucci
University of CORSICA - CNRS UMR SPE 6134
Faculté des Sciences et Techniques Batiment 018 Quartier Grossetti 20250 Corte, Corse
{bisgambiglia/gentili/bisgambi/santucci}@univ-corse.fr

Abstract— **In this paper, we present our work on fuzzy modeling, and in particular an approach based on the integration of the uncertain theories in the formalism of multi-modeling and simulation with discrete events DEVS. The goal of this approach is to help the expert of a field to specify in a simple way the behavior of a complex system with parameters badly defined, fuzzy, etc. This approach can be employed in multiple fields, an application to the study of the fires forest propagation is presented in order to validate the models.**

I. Introduction

The study of the complex phenomena, like the natural systems, can be facilitated by the use of modeling and simulation formalisms which reduce the complexity of the system to the analysis of the most important parts. The modeling of natural systems generates the treatment and the analyze of information and variables for which the values are often vague, dubious, badly definite, etc. The traditional approach consists in approximating the values of the fuzzy variables, which can generate, during the simulation, erroneous results.

In order to allow the simplification and the study of such systems as well as the taking into account of these fuzzy data, we propose to define an approach linking the work of professor Zeigler [1] on modeling and simulation in discrete events (DEVS), and "the uncertain theories" introduced by professor L.A Zadeh [2].

The modeling and simulation DEVS formalism (Discrete EVent system Specification) has been developed for more than thirty years by a scientific community [3], [4], [5], [6], [7], [8]. This work is integrated in the development of an approach making it possible to facilitate the phases of modeling, simulation and validation, at the time of the study of complex systems. This approach rests on the development of a software architecture making it possible on one hand to use the same environment of multi-modeling to analyze different systems or fields, and on the other hand, to implement techniques of generic simulation in order to be able to simulate the corresponding models.

There are many couplings between the modeling and the treatment of the uncertain [5], [4], [9], but none is appropriate to us perfectly. All the methods making it possible to represent, handle, and exploit fuzzy data, like fuzzy logic or the fuzzy set theory introduces by L.A. Zadeh [10], or the fuzzy arithmetic and the vertex method of professor D. Dubois [11], [12], etc., are gathered under the name "treatment of the uncertain" or "uncertain theories".

In this paper our generic approach is applied to the study of the propagation of forest fires.

The first two parts of this article deal with the basic concepts of our work, fuzzy logic and DEVS formalism. In the third part we describe our method of modeling, finally before concluding we present an example of application and the obtained results.

II. Fuzzy logic

Within the framework of traditional logic, a proposal is either true, or false, or unspecified. However, human reasoning often uses some confuse knowledge and some imperfect data.

Fuzzy logic [13] was established with an aim of treating the inaccuracy and the uncertainty of the data. For a given information element, the inaccuracy concerns the value and the uncertain is related to its truth. Fuzzy logic, or more generally the treatment of inaccuracy, has, as an aim of study, the representation of uncertain knowledge and the approximate reasoning.

Many tools or methods were developed in this direction, like the fuzzy sets theory [10], the extension principle [10], the fuzzy arithmetic or the vertex method [12]...

A. *The fuzzy sets theory*

Fuzzy logic is based on the mathematique concept of fuzzy sets. The definition of a fuzzy set meets the need to represent inaccuracy or uncertain knowledge [11]. These data are fuzzy, either because they are expressed in natural language by an observer which gives few precision or which is not very reliable, or because they are obtained using observation instruments which produce errors or which are not very precise. The concept of fuzzy set makes it possible to treat:
- of some categories in badly boundaries definite;
- of the intermediate situations between the whole and nothing;
- the progressive passage of a property to another;
- of the approximate values;
- of the classes by avoiding the arbitrary use of rigid limits.

The concept of fuzzy set constitutes an easing of the one of a given subset.

A.1 Definition

In a reference set X, a fuzzy set of this reference is characterized by a membership function λ of X in the interval of the crisp number $[0,1]$ [10]. This function is the extension of the characteristic function of a traditional set. The purpose of the concept of fuzzy set is to authorize an element to, belong more or less strongly, to a class.

A fuzzy set $\tilde{A}$ on the field of variation X of x is defined by the triplet: $(\tilde{A}, \tilde{a}, \mu_{\tilde{A}})$, where:

- $\tilde{A}$ is a subset of X;
- $\tilde{a}$, a linguistic label, characterizing qualitatively part of the values of X;
- $\lambda_{\tilde{A}}$, the function x of X $x \in X \rightarrow \mu_{\tilde{A}}(x) \in [0;1]$, which gives the degree of membership of an observation of X to fuzzy set $\tilde{A}$.

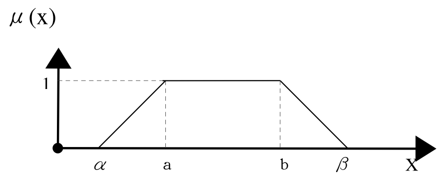

Fig. 1. Membership function example

A.2 Extension principle

The extension principle (eq.1), proposed originally by Pr. L.A. Zadeh [10], is one of the fundamental tools of the theory of the fuzzy sets. It makes it possible to extend traditional functional relations to fuzzy quantities. Let F be an application from a universe X to a universe Y. Where A is a definite fuzzy subset of X. The principle of extension stipulates that the image by F of A, $F(A)$, is a fuzzy subset of Y which membership function membership is defined by:

$$f_B(y) = sup_{(x=x_1,...x_r) \in X | y=f(X))} \times min(f_{A_1}(x_1), ...f_{A_r}(x_r)) \quad (1)$$

The extention principle allows to obtain the image of fuzzy sets by a function.

A.3 Fuzzy arithmetic

A fuzzy number (fig.1 and 2), as Dubois and Prade defined it in [11], is a fuzzy interval at compact support having only one modal value. To make it simplier and more effective their handling, certain classes of numbers and fuzzy intervals were defined using a parametrical representation known as $L-R$. We take other two functions of form, L (left) and R (right), of $\mathbb{R}^+$ in $[0,1]$, symmetrical, not decreasing on $[0,+\infty[$; such as: $L(0) = R(0) = 1$, $L(1) = R(1) = 0$ where $L(x) > 0 \, \forall \, x$ with $lim_{x \to \infty} L(x) = 0$ and $R(x) > 0 \, \forall \, x$ with $lim_{x \to \infty} R(x) = 0$.

They are noted :

$$\mu_A(x) = \begin{cases} L(\frac{a-x}{\alpha}) \, if \, x \leq a \\ 1 \, if \, a < x < b \\ R(\frac{x-b}{\beta}) \, if \, x \geq b \end{cases} \quad \text{for } A = [a, b, \alpha, \beta] \text{ (fig.1).}$$

There are many methods for handling of such numbers, presented in [14] in particular the Vertex method [12]. A vertex is a function to $[0;1]$ in $\mathbb{R}$ which enables to model a bound of the fuzzy interval.

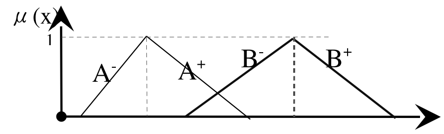

Fig. 2. Fuzzy interval example defined by $[A^-, A^+]$ with (a=b)

For the $L-R$ fuzzy intervals type: A and B two fuzzy intervals defined by $[A^-, A^+]$ and $[B^-, B^+]$ (see fig.2). A^+ representing the equation of the half-line $(B, beta)$ defined by equation : $A^+(\lambda) = \beta - \lambda \times (\beta - b)$. A^- representing the equation of the half-line $(alpha, A)$ defined by equation $A^-(\lambda) = \lambda \times (m - \alpha) + \alpha$.

From that, to carry out operations between intervals, it is enough to add, withdraw, multiply and divide the equations of A and B between them.

- $A + B = [A^+ + B^+, A^- + B^-]$
- $A - B = [A^- - B^+, A^+ - B^-]$
- $A \times B = [min(A^- \times B^-, A^+ \times B^-, A^- \times B^+, A^+ \times B^+), max(A^- \times B^-, A^+ \times B^-, A^- \times B^+, A^+ \times B^+)]$

B. *Fuzzy modeling*

Fuzzy modeling, i.e. the design of fuzzy systems, is a difficult task, requiring the identification of many parameters. According to the Pr. L.A. Zadeh: "fuzzy modeling provides approximate but efficient means to describe the behaviour of the systems which are too complex or too badly defined to admit the use of a precise mathematical analysis".

One of the most important problems in fuzzy modeling is the problem of dimension i.e. that the computing conditions develop exponentially with the number of variables. However for systems such as the propagation of forest fires, where many parameters difficult to define enter into account, fuzzy modeling proves to be relatively interesting.

III. Discrete event modeling: DEVS

Modeling can be defined like an operation by which the model of a phenomenon is established, its setting in equation, in order to have of it a simplified representation, interpretable and which can be simulated.

Since the 1970's, formal work have been undertaken to develop the theoretical bases of discrete events modeling and simulation.

DEVS (Discrete EVent system Specification) [15] makes it possible to the modeling specialist to be completely abstracted from the creation of the simulators implementing the model system. DEVS Simulation is based on the taking into account of events and not on a progression according to time.

A. *DEVS modeling principle*

DEVS formalism can be defined like a universal and general methodology which provides tools to model and simulate systems whose behaviour is based on events. It is based on the systems theory, the concept of component and allows the specification of discrete events complex systems in a modular and hierarchical form. Nevertheless DEVS must be adapted and extended when it is replaced in the specific context of applicability field.

DEVS [15] is based on the definition of two types of components: atomic components or atomic model and the composition model or coupled model.

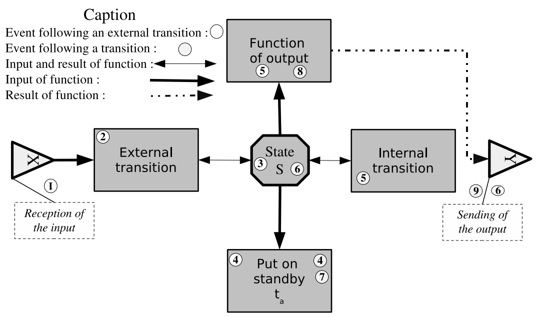

Fig. 3. Behaviour of an atomic model

A.1 Atomic model

The atomic model (AM, fig.3, eq.2) provides an autonomous description of the behaviour of the system, defined by states, by input/output functions, and by internal/external transitions functions of the component. It is characterized by:

$$AM =< X, Y, S, t_a, \delta_{int}, \delta_{ext}, \lambda > \qquad (2)$$

with :
- X the input ports set, through which external events are received;
- Y the output ports set, through which external events are sent;
- $S$the states set of the system;
- t_a: $S \rightarrow \mathbb{R}^+$ the time advance function;
- δ_{int} : $S \rightarrow S$ the internal transition function;
- δ_{ext}: $Q \times X \rightarrow S$ the external transition function, with :
  - $Q = \{(s, e)|s \in S, 0 \leq e \leq t_a(s)\}$ state set;
  - $e =$ the time passed since the last transition;
- $\lambda : S \rightarrow Y$ the output function.

A.2 Coupled model

The coupled model (eq.3) is a composition of atomic models and/or coupled models. It is modular and presents a hierarchical structure, which allows the creation of complex models from basic models. It is described in the form:

$$CM =< X, Y, C, EIC, EOC, IC, L > \qquad (3)$$

with :
- X the input ports set;
- Y the output ports set;
- C the set of all component models;
- EIC the external input coupling relation which connects the input ports of the coupled model to one or more of the input ports of its internal components;
- EOC the external output coupling relation which connects the output ports of the internal components to the output ports of the coupled model;
- IC the internal coupling relation which connects the output ports of the internal components to the input ports of other components;
- L the list of priorities between components.

In DEVS each component is independent and can be regarded as a whole entity with of the system, or as the component of a larger system. It is shown in [15], [7] that DEVS formalism is closed under coupling, i.e. that for each atomic or coupled DEVS model, it is possible to build an equivalent DEVS atomic model.

There are many environments which integrate DEVS methodology; we chose to work with the PowerDEVS environment [16]. PowerDEVS is a tool for the modeling and the simulation of hybrid systems, free, including GCC compiler of GNU. It allows to define its own models and libraries and provides a graphic interface for the design and simulation models. The simulation processes of DEVS are presented in [15], [7].

B. *DEVS extensions*

DEVS was developed for the study of electronic systems; its use in many other fields leads to the development of many extensions. We can quote DSDE [17] and dynDEVS [18] for the dynamic systems. The dynamic systems have a structure which can evolve in time...

fuzzy-DEVS [4] is a first extension of DEVS which tries to take into account fuzzy data. The fuzzy-DEVS formalism introduced by Y. Kwon into [4] is derived from DEVS formalism while preserving its semantics, a part of its concepts and its modularity. It is based on fuzzy logic, the "Max-Min" rule and the methods of fuzzyfication and defuzzyfication. The fuzzy atomic model, of fuzzy-DEVS, contrary to DEVS atomic model, is not determinist, i.e. it does not answer the two following conditions:

1. The internal transition function is launched ($\delta_{int}(s_t) = s_{t+1}$) when the lifespan of the state is passed ($t_a = 0$) and the external transition function ($\delta_{ext}(s_t, X_t) = s_{t+1}$) is carried out when an external event is received before time is passed.
2. The output function ($\lambda(s_t) = Y_t$) is launched when the lifespan of a state is finished ($t_a = 0$).

In fuzzy-DEVS the following state ($S \leftarrow S_{t+1}$ is not determined with δ_{int} and δ_{ext} but with the rule "Max-Min" [4]. The various possibilities of input, output and state update are represented by matrices and the evolution of the model by possibilities trees [4]. The algorithms which generate all the possible trajectories are not very effective and are always in research phase.

This approach does not appear completely coherent to us with DEVS formalism. A fuzzy-DEVS model does not have the property of closing under coupling defined in the DEVS formalism [15], [7]. Moreover to allow simulation the fuzzy parameters must be transformed into crisp parameters (defuzzyfication), and finally to be able to exploit the output data, those are again transformed into fuzzy data (fuzzyfication).

IV. Discrete event methodology of fuzzy modeling

We saw that fuzzy logic, as well as modeling and simulation formalism, could be an important tool for the study of natural phenomena. Their association could be beneficial in this way. The goal of this part is to present our method of modeling based on fuzzy logic and DEVS, as well as the tools developed to allow this association.

For its modeling part DEVS formalism is based on two types of elements: the atomic and coupled model. As we saw in the first part, these elements have of input ports, output ports and variables. The exchange of the data is established through the different ports of a model, thanks to two types of fundamental events: external events and internal events:

- An external event expected at the date t represents a modification of the value of one or more input ports belonging to a given element M. This has as a consequence a modification of the variables of M, at the date t;
- An internal event expected at the date t, represents a modification of the variables of M, without any external event intervening. Moreover, the arrival of an internal event causes, at the moment t, a change of value on one or more output ports of the model M.

A DEVS event is characterised by the formula: $E_{event} = (time, port, value)$.

The first field represents the time of occurrence of the event, the second indicates the port on which the event intervenes, and the third symbolizes the value of the event.

In DEVS an event takes place at a given time, and modifies the state (the value) of only one variable. As we already saw it, DEVS formalism does not allow the taking into account of fuzzy data. This one passes by an evolution in the definition of the DEVS events, which are at the base of simulation.

A. Fuzzy events treatment

For the events there can be two types of fuzzy parameters, time and/or value. For the values, whatever their form is, interval or real, they are defined like an object of the FuzzyInterval type with $[a, b, \alpha, \beta]$ (fig.1), and the methods of classes make it possible to handle them. On the level of time it is problematic. An event is sent and placed in the simulation schedule on a given date; if we don't know this date the event cannot take place and thus be taken into account. Such events are not treated yet with our method. We work at the present time on the events with fuzzy value. Afterwards our method will be enriched by techniques of fuzzy simulation which should make it possible to treat these parameters.

Another method can be used; it consists in transforming the fuzzy on the dates in fuzzy on the values.

For example, for the proposal: "a fire traverses 2 kilometers in an interval from 40 to 60 minutes", thanks to a transfer field function defined by an expert, it can be transformed into:
1. At the least the fire courses 2 km in 60 minutes, therefore in 50 it would have traversed 1,6 km of them;
2. At the most thee fire courses 2 km in 40 minutes, therefore in 50 it would have traversed 2,5 km of them.
3. At the end our data has this form: "in 50 minutes fire can traverse between 1,6 and 2,5 km".

That brings us to treat a fuzzy on a value.

B. Fuzzy atomic model

Our fuzzy atomic model (eq.4) presented in [19] the same form as a standard DEVS atomic model but the values of the state variables ($\widetilde{S}$), the life span of the state ($\widetilde{t_a}$) and the input output variable ($\widetilde{X}, \widetilde{Y}$) can be fuzzy, it is there one of the forces of our approach. It thus preserves all its properties. A fuzzy atomic model can be regarded as an interval of fuzzy data which includes an interval of real data representing the DEVS atomic model. A fuzzy model is a generalization of a standard model ($\tilde{AM}_{fuzzy} \subset AM_{DEVS}$).

Fuzzy atomic model example:

$$\tilde{AM} :< \tilde{X}, \tilde{Y}, \widetilde{S}, \widetilde{t_a}, \widetilde{\delta}_{int}, \widetilde{\delta}_{ext}, \widetilde{\lambda} > \quad (4)$$

with:
- $\tilde{X} : \{fI_1, fI_2\}$: the input ports, receive the fuzzy intervals;
- $\tilde{Y} : \{fI_R\}$: the output port, send the result of the operation ϕ
- $\widetilde{S} : \{Status, \sigma, R\}$ or:
  - $Status$: the status of the model, $\{ACTIVE/INACTIVE\}$, initialy the status is $INACTIVE$;
  - σ is a lifespan of a state;
  - R : the result of the operation ϕ;
- $\widetilde{t_a}(S) :\rightarrow return(\sigma)$
- $\widetilde{\delta}_{ext} : \begin{cases} Status \leftarrow ACTIVE \\ \sigma \leftarrow 0 \\ R \leftarrow fI_1 \phi fI_2 \end{cases}$ with $\phi = \{+, -, \times, \backslash\}$
- $\widetilde{\delta}_{int} : \begin{cases} Status \leftarrow INACTIVE \\ \sigma \leftarrow INFINITY \end{cases}$
- $\widetilde{\lambda} :\rightarrow send(R)$

C. Data handling

We chose to treat all the data in the form of interval of the $L-R$ type $[a, b, \alpha, beat]$ (fig.1) with α the lower limit of the interval, β the higher limit and a, b the most possible values.

From each interval we can define two functions:
1. $f^+(\lambda) = \alpha + (a - \alpha) \times \lambda$ with $\lambda \in [0, 1]$;
2. $f^-(\lambda) = \beta - (\beta - b) \times \lambda$.

For example to model the information: "the wind blows at $20km$ per hour at more or less 10%".

Speed of the wind $= [a \leftarrow 20, b \leftarrow 20, \alpha \leftarrow 18, \beta \leftarrow 22]$.

This method also makes it possible to define an integer $If_{value} = 4$ in the form $[a \leftarrow 4, b \leftarrow 4, \alpha \leftarrow 4, \beta \leftarrow 4]$.

Once having modelled the data it is necessary to be able to handle those using operations. For that we based ourselves on the work of the Pr. Dubois [12], presented in the second part. Among all the methods which allow the handling of fuzzy data, this one appears to us to be the most

Algorithm 1 Fuzzy Interval class

$class$FuzzyInterval{
int $\lambda \in [0,1]$
float a,b
float α, β
function $left(\lambda) = \lambda \times (a - \alpha) + \alpha = \psi \times (\lambda - 1) + a$ for $\alpha = a - \psi$
function $right(\lambda) = \beta - \lambda \times (\beta - b) = \pi \times (1 - \lambda) + b$ for $\beta = b + \pi$
FuzzyInterval *operator* $* (FuzzyInterval)$
FuzzyInterval *operator* $= (FuzzyInterval)$
FuzzyInterval *operator* $+ (FuzzyInterval)$
}

optimized and the most flexible. This method was integrated in a library to allow the definition and the handling of objects of the interval fuzzy type (example algorithm 1).

The FuzzyInterval class implemented in C++ was integrated into the PowerDEVS [16] environment as a library. A first application example is given in [14].

Our method has several interests, at first, it makes it possible to extend the applicability fields of DEVS formalism to the systems at badly definite parameters; then the use of the tools related of the fuzzy logic makes it possible to build new types of models with the possibility of defining as fuzzy the the events (time and value) and the parameters of the model ($\tilde{X}, \tilde{Y}, \widetilde{S}, \widetilde{t}_a, \widetilde{\delta}_{int}, \widetilde{\delta}_{ext}, \widetilde{\lambda}$), by using known data under the name of linguistic data. This type of data, very near to the mode of human representation, is adapted to the dialogue with the experts of a field. Moreover, we do not use fuzzyfication and defuzzyfication functions for the modeling part because the data result directly from the domain specialists. For the simulation part they can be used to transform an uncertainty over a time, but we currently work on an fuzzy ranking algorithm to cure this problem.

Initially and in order to be validated, our method was applied to solve fuzzy first order differential equations [14]. The following part presents another application, the study of the forest fires propagation.

V. Fire spreading application

There are several methods for the study of the fires propagation. Some get busy to describe in a more or less pointed way all the mechanisms implemented using physical or mathematical equations [20], [21], [22]. Others closer to reasoning on a land scale consider than a great number of parameters can not be taken into account.

Our laboratory accentuated its work on two aspects: the fight and the prevention, in these two fields of the tools of decision aid are essential.

In front of the extent of work, it is necessary to make the bet of the effectiveness of the committed actions. That implies to permanently adjust the strategy of prevention and fight according to the means available. Accordingly and in order to tally with realities of land we began a work in collaboration with the Departmental Service of Fire and help from High-Corsican (SDIS). Several tracks of work emerged from this collaboration; those remain very close to their needs and concerns.

A. Problems

One of the problems proposed by the SDIS is the need for quickly envisaging the possible fire evolutions, in order to set up a fight adequate policy. This one must take account of several requirements: the accessibility of the land, provision and use on site of the material or the men, method of intervention H-B-E for man-building-environment. They deal with initially the men then the dwellings and finally the vegetation.

Initially, to meet their needs, we directed our work towards the definition of a propagation model of on a land scale. The first stage was the identification of the parameters.

B. Parameters identification

We identified three groups of parameters to be taken into account, the vegetation, the topology of the land and the weather. These groups can be divided into several under parts, with:

- for the weather, the wind: power and direction, and the moisture of the air;
- for the vegetation: type, density, height, moisture, inflammability;
- and for the topology of the land: the slope and configuration.

By assumption, and following the various discussions with the firemen who use like approximate speed of propagation three percent the speed of the wind, we consider that on a land scale only the wind has a true influence on the propagation.

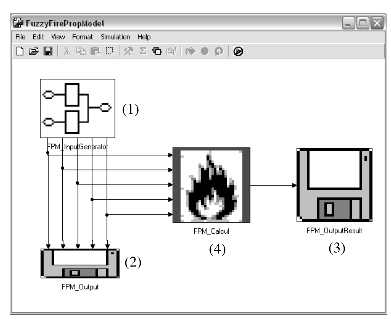

Fig. 4. powerDEVS model

C. Simple model

Our propagation model is based on several assumptions.

- The propagation velocity is regarded as constant between two events. For its calculation, we suppose that it is more or less equal to 3% (prC algo.2) of the speed of the wind, and the width and the height of the front of flame are not taken into account.
- The front of flame is schematized by a line. To represent its evolution, we calculate the coordinates of the point

of intersection between the middle of this line and a perpendicular segment representing the direction of the wind. The front of flame evolves according to the direction of the wind.

- The vegetation and the topology of the land are represented by a coefficient fixed at 1, which influences the propagation velocity.

The model of the figure 4 is a PowerDEVS coupled model which implements our propagation model. The model (1) (fig.4) is a coupled model with 5 outputs, which generates the parameters of the land: coordinates (x, y) of the point representing the front of flame, coefficient of vegetation, power and direction of the wind. The models (2) and (3) (fig.4) allow printing in a csv file the output parameters of the models (1) and (4). The model (4) describes our spreading model and calculates the distance covered by fire.

The model (4) takes in input (eq.5) the parameters of the model (1) (fig.4):

$$X_{input} = \{double\ C_X, C_Y, Vege_{coef}, W_{pwr}, W_{dir}\} \quad (5)$$

At the reception of an input, it updates its state (eq.6) variables by launching the function of external transition δ_{ext} (algo.2).

$$S = \{double\ Coor[x, y], Vege_{coef}, Wind[pwr, dir]\} \quad (6)$$

The external transition function δ_{ext} put also the lifespan of the state at $(t_a = 0)$, which, according to the DEVS method, launch the internal transition function δ_{int} and the output function λ.

Algorithm 2 δ_{ext} : External transition function

function $\delta_{ext}(C_X, C_Y, Vege_{coef}, W_{pwr}, W_{dir})\{$
$Coor[x] \leftarrow C_X$;
$Coor[y] \leftarrow C_Y$;
$Wind[pwr] \leftarrow W_{pwr}$;
$Wind[dir] \leftarrow W_{dir}$;
$Vege_{Coef} \leftarrow Vege_{coef}$;
$ta = 0;\}$

The internal transition function δ_{int} (algo.3) calculates the new coordinates $Coor[x, y]$ and launches the output function λ, which sends the coordinates $Coor[x, y]$ towards the model (3) (fig.4). The results of simulation are presented in the table I.

Algorithm 3 δ_{int} : internal transition function

function $\delta_{int}()\{$
$double\ prC \leftarrow 0.03$; // 3%
$double\ angle \leftarrow \Pi/180$; // angle in degree
$double\ distance \leftarrow time \times Wind[pwr] \times prC \times Vege_{Coef}$;
$Coor[x] \leftarrow Coor[x] + distance \times cos(Wind[dir] \times angle)$;
$Coor[y] \leftarrow Coor[y] + distance \times sin(Windt[dir] \times angle)$;
$ta = time$; // time before a new update$\}$

D. Fuzzy model

The fuzzy model has the same behaviour as the simple model. The only changes which are brought are the type of the treated data.

$$\tilde{S} = \left\{ \begin{array}{c} FuzzyInterval\ Coor_X, Coor_Y \\ FuzzyInterval\ Wind_{Pwr}, Wind_{dir} \\ FuzzyInterval\ Vege_{coef}, \end{array} \right\} \quad (7)$$

The simple model uses *double* whereas the fuzzy model uses objects of the $FuzzyInterval$ type (see eq.7 and algo.4).

Algorithm 4 $\tilde{\delta}_{int}$: fuzzy internal transition function

function $\tilde{\delta}_{int}()\{$
$FuzzyInterval\ prC \leftarrow [0.03, 0.03, 0.026, 0.034]$; // 3%
$double\ angle \leftarrow \Pi/180$; // angle in degree
$FuzzyInterval\ distance$;
$distance \leftarrow Wind_{pwr} \times prC \times Vege_{Coef} \times time$
$Coor_X \leftarrow Coor_X + distance \times cos(Wind_{dir} \times angle)$;
$Coor_Y \leftarrow Coor_Y + distance \times sin(Wind_{dir} \times angle)$;
$ta = time$; // time before a new update$\}$

To make it possible to carry out the various types of operations it was necessary, in the $FuzzyInterval$ class, to overload a great number of operators such as $+, -, \times, \div, =$, ... (see algo.1).

A problem remains all the same on the level of the angles (fig.4). We did not overload the functions $sin()$ and $cos()$ of the library "$math.h$", and thus we could not define a fuzzy propagation direction.

The results of the simulation of this model, with like fuzzy parameter the vegetation coefficient, the power wind and prC (algo.2) the approximate speed of propagation, are presented in the table II.

(a)

$Time$	$Vege$	$Wind_{pwr}$	$Wind_{dir}$	X	Y
0	1	2.7	10	4	4

(b)

$Time$	X	Y
0	4	4
1800	147.5	29.3
3600	291.1	54.6
5000	434.7	79.9

TABLE I

Input and output data of the simple model

E. Results

Tables (I.a) and (II.a) are drawn from the model (2) figure 4. They present the input data.

It is noticed, for table (II.a), that the fuzzy data (the vegetation and the power wind) are printed in the form of an interval $[a, a = b, \alpha, \beta]$ (see figure 1). The approximate speed of propagation (prC) is a model state value. The tables (I.b) and (II.b) show the results of the simulation of the model (4). It is noted that the outputs of the fuzzy model are fuzzy interval and that for the most possible values the two results are the same. We can conclude from these

	$Time$	0	
	$Vegetation$	$[1, 0.6, 1.4]$	
	$Wind_{pwr}$	$[2.7, 2.5, 2.9]$	(a)
	$Wind_{dir}$	10	
	X	4	
	Y	4	

$time$	x	y	
0	$[4, 4, 4, 4]$	$[4, 4, 4, 4]$	
1800	$[147.5, 78.7, 231.8]$	$[29.3, 17.1, 44.1]$	(b)
3600	$[291.1, 153.3, 459.6]$	$[54.6, 30.3, 84.3]$	
5000	$[434.7, 227.9, 687.4]$	$[79.9, 43.4, 124.5]$	

TABLE II

INPUT AND OUTPUT DATA OF THE FUZZY MODEL (WITH $a = b$)

results that simulation from the simple model (I) gives precise but surely erroneous results on the ground scale, the results of table two (II), although they are vague, are likely great to fall right or on a scenario which will occur.

VI. Conclusion and Remarks

In this paper we presented a part of our work on fuzzy modeling; in particular our approach based on the fuzzy arithmetic, the vertex method, and on the multimodeling and simulation at discrete event DEVS formalism. Our approach makes it possible to take into account an uncertainty on the value of the parameters of a model $(\widetilde{X}, \widetilde{Y}, \widetilde{S}, \widetilde{t_a}, \widetilde{\delta}_{int}, \widetilde{\delta}_{ext}, \widetilde{\lambda})$ like on the parameters of an event $(time, value)$; it is based on no function of fuzzyfication or defizzyfication, the data are directly modelled and treated like fuzzy. The aim of this method is to help the experts of a field, like the firemen for forest fires, to specify in a simple way the behaviour of a complex system with badly definite parameters.

The basic idea of our methodology is to make it possible for the modeling specialist to specify parameters of fuzzy models, using interval or linguistic variables. In order to make the simulation of these data possible, a library of fuzzy functions was added to DEVS formalism.

Although it is already exploitable, our method is always in phase of research, development and validation. For its improvement we work on several works, such as its integration in a specific environment allowing carrying out fuzzy simulations. The use of a not dedicated environment limits its exploitation possibilities. We showed that for the moment it is not possible to simulate events at fuzzy time. From this perspective an environment is under implementation progress, nd we work on the algorithms of fuzzy simulation [23], [24].

On the level of the validation, we must prove that our method is generic and usable in several fields; the realization of a wind model could be a first step. Lastly, it will be also necessary to see how the firemen will be able to use it. About this subject, we follow a project which aims at integrating DEVS formalism and several modeling techniques like the Multi-Agents Systems in a Geographic Information System.

References

[1] B. P. Zeigler, *Theory of Modeling and Simulation.* 1976.

[2] L. Zadeh, *The Role of Fuzzy Logic in the Management of Uncertainty in Expert Systems.* 1983.

[3] L. Capocchi, D. Federici, F. Bernardi, and P. Bisgambiglia, "Behavioral fault simulation for VHDL descriptions using the DEVS formalism," 2004.

[4] Y. Kwon, H. Park, S. Jung, and T. Kim, "Fuzzy-DEVS Formalisme : Concepts, Realization and Application," *Proceedings AIS 1996*, pp. 227–234, 1996.

[5] N. Giambiasi and S. Ghosh, ""min-max-DEVS: A new formalism for the specification of discrete event models with min-max delays, "," pp. 616–621, 13th European Simulation Symposium, 2001.

[6] Y.-B. Lin and P. A. Fishwick, "Asynchronous parallel discrete event simulation," vol. 4, no. 26, pp. 397–412, 1995.

[7] H. Vangheluwe, "The discrete event system specification DEVS Formalism," tech. rep.

[8] G. Wainer, "Using DEVS for modeling and simulation of computer networks," 2004.

[9] P. Fishwick, O. Balci, R. Sadowski, and R. Nance, "Fuzzy set method for qualitative and natural language oriented simulation," *Winter Simulation Conference*, pp. 513–519, 12 1990.

[10] L. Zadeh, "Fuzzy sets," *Information Control*, vol. 8, pp. 338–353, 1965.

[11] D. Dubois and H. Prade, "Fuzzy set in approximate reasoning," *part 1, Fuzzy set and Systems, vol 40*, 1993.

[12] D. Dubois, H. Fargier, and J. Fortin, "A generalized vertex method for computing with fuzzy intervals," in *Proc. of the International Conference on Fuzzy Systems* (IEEE, ed.), pp. 541–546, IEEE, July 2004. Budapest, Hungary.

[13] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, 1988.

[14] P.-A. Bisgambiglia, L. Capocchi, E. de Gentili, and P. Bisgambiglia, "Manipulation of incomplete or fuzzy data for DEVS-based systems," pp. 87–92, International Modeling and Simulation Multiconference (IMSM) - Conceptula Modeling Simulation (CMS), 2 2007.

[15] B. P. Zeigler and S. Vahie, "DEVS formalism and methodology - unity of conception diversity of application," in *Proceedings of the 1993 Winter Simulation Conference* (S. Editions, ed.), pp. 573–579, 1993.

[16] E. Kofman, M. Lapadula, and E. Pagliero, "PowerDEVS: A DEVS-based environment for hybrid system modeling and simulation," tech. rep., 2003. LSD0306, Rosario National University.

[17] F. Barros, "Abstract simulators for the dsde formalism," in *Proceedings of WSC 1998*, pp. 407–412, Winter Simulation, 1998.

[18] A. Uhrmarcher, "Dynamic Structures in Modeling and Simulation : A Reflective Approach," *ACM Transactions on Modeling and Computer Simulation vol. 11 2001*, pp. 206–232, 2001.

[19] P.-A. Bisgambiglia, E. de Gentili, J. Santucci, and P. Bisgambiglia, "DEVS-Flou: a discrete events and fuzzy sets theory-based modeling environment," pp. 95–100, Systems and Control in Aerospace and Astronautics, (ISSCAA), 1 2006.

[20] A. Grishin, *Mathematical modelling of forest fires and new methods of fighting them.* House of the Tomsk State University, 1997.

[21] R. Weber, "Modeling fire spread through fuel beds," *Prog. Energy Combust. Sci, vol. 11*, pp. 67–82, 1991.

[22] R. C. Rothermel, "A mathematical model for predicting fire spread in wildland fuels.," *Research Paper INT- 115*, vol. Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station., p. 40p, 1972.

[23] G. Kim and P. Fishwick, "Validation method using fuzzy simulation in an object-oriented physical modeling framework," *Enabling Technology for Simulation Science II*, vol. 3369, pp. 143–153, 1998. SPIE.

[24] A. Anglani, A. Grieco, F. Nucci, G. Semeraro, and T. Tolio, "A sound analysis of fuzzy simulation results," (citeseer.ist.psu.edu/496399.html), Intelligent Computation in Manufacturing Engineering 2000, June 2000. Capri, Naples, Italy.