Intelligent bush fire spread prediction using fuzzy cellular automata

Miha Mraz, Nikolaj Zimic and Jernej Virant
Faculty of Computer and Information Science, University of Ljubljana, Tržaška 25, 1000 Ljubljana, Slovenija
E-mail: miha.mraz@fri.uni-lj.si

In this paper, we wish to present a practical approach to modelling the shape of a wild bush fire using fuzzy logic and cellular automata (CA). In this way, the time consuming measurements, which are used as input data for the statistical modelling approach, could be exchanged with uncertain knowledge built in to the decision structure.

1. Introduction

The main goal of models for the wild bush fire spread and shape simulation is to make a prediction of the area which will catch fire. They are used in the domains of fire intervention and fire prevention education. In the sources [2, 3, 5] we can find the basic approaches used for modelling fire spread shape and size. These are

- Physical-chemical models,
- Statistical models and
- Cell models.

The latter approach uses the network of cells and the rule base composed of rules which describe how the disturbance (fire) is spread through the cells under the observed conditions. The disadvantage of this approach is in determining the exact rules, which are based in most cases on iterative equations and not on experienced knowledge of the system's behaviour provided by a system expert.

Our approach is based on the cell model. Instead of crisp rules, we used the concept of fuzzy logic rules. This enabled us to use the descriptive and uncertain knowledge of the system's behaviour provided by firemen, who have practical experience of fire spread. The use of fuzzy logic also enables us to formulate a decision process with uncertain or approximate input data instead of exact values.

2. The basics of cellular automata

The cellular automaton (CA) is a structure built from cells in n-dimensional space. The dimension n of space in real-life applications is usually 2 or 3. Every cell can be treated as an independent computing device which captures input data from its neighbouring cells, calculates and changes to a new state, and this state is used as input to other cells in the neighbourhood in the next processing step. The "program" of a cell's behaviour is unique for all cells in the space. A generalised definition of CA [1] can be seen in Definition 1.

Definition 1. Cellular automaton M is a quadruplet $\{A, Q, u, F\}$, where A is an n-dimensional array, Q a finite non-empty set of possible states for all cells, $u(x)$ a function which returns a neighbourhood of cell x unique for all cells, and $F(u(x), q(x, t))$ a set of exact rules for local transitions (a cell behaviour program).

The pattern is the global state of CA and it is observed as a set of all states of the cells. Other main characteristics of CA are:

- Different cells can be in different states in step t.
- On the base of the initial pattern and pattern's dynamics throughout the processing steps we can divide the models into two groups: the patterns which change slowly are called "cold", and the patterns which change very rapidly are called "gassy".

Journal of Intelligent and Fuzzy Systems 7 (1999) 203–207
ISSN 1064-1246 / $8.00

- Some self-organisation possibilities which could lead the system from initial global pattern to next possible global states known from real-system's behaviour are:
  - pattern "dies"
  - pattern becomes stable or occurs cyclically with constant period
  - pattern grows with constant speed infinitely.

Often, when we create the model of a real-life system we do not know the exact rules for local transitions and that is the main problem in programming CA's behaviour. For this reason we suggest a fuzzy approach for cell-state transitions.

3. A fuzzy cellular automata approach

Cells can also be treated as fuzzy automata. The definition which follows was partially developed by [4, 6, 8, 9].

Definition 2. Fuzzy automaton $\tilde{A}$ is defined as a 7-tuple $\tilde{A} = (I^n, Q^m, F^m, f_A, h, \pi, O)$. I represents n-parametric finite unempty fuzzy set of inputs, Q m-parametric finite unempty fuzzy set of possible states, F m-parametric fuzzy set of final states and O output fuzzy set of automata $\tilde{A}$. f_A and h are time dependent fuzzy functions defined as

$$\forall\ (q, x, q', t),\ q, q' \in Q^m,\ \ x \in I^n,\ \ t \in T:$$

$$f_A:\ Q^m \times I^n \times Q^m \times T \rightarrow [0, 1],$$

$$h:\ I^n \times Q^m \times T \times O \rightarrow [0, 1]. \tag{1}$$

π represents the distribution vector of memberships and it could be treated as a start state from crisp automata definition. Transitions of states in automaton are defined as

$$\forall\ (e, x, y) \in I^*,\ \ \forall\ (q_1, q_2) \in Q:$$

$$f_A(q_1, e, q_2) \ = \ \begin{cases} 1 & \textit{if}\ \ q_1 \ = \ q_2, \\ 0 & \textit{if}\ \ q_1 \ \neq \ q_2, \end{cases}$$

$$f_A(q_1, xy, q_2)$$

$$= \max_{q \in Q} \min\ [f_A(q_1, x, q), f_A(q, y, q_2)] \tag{2}$$

and I^* is defined as a set of all final strings under I.

A fuzzy approach in the field of CA enables us to program the cell's behaviour on the basis of uncertain and descriptive knowledge of the real system's behaviour. This enables us to:

1. Provide imprecise descriptive rules directly from the system-behaviour expert.
2. The modelled system's dynamics are often processed as a result of parallel triggering of opposite rules.

In the processing phase at step t we try to decide, for every cell, its next state $(t + 1)$ on the basis of input data and cell-state in step t. A pseudo-code sequence to explain the decision process is as follows (3):

1. `fuzzify(global input data);` (for instance *wind*)
2. `for all cells do:`
   - `fuzzify(local input data);`
   - `new_stage[x,t+1] =`
     `F(u(x),q(x,t), global_data);`
   - `defuzzify(new_stage[x,t+1]);`
3. `for all cells do:`
   `fire_stage = new_stage;` (3)

The space dimension n of our application mentioned in [5] is 2 and the neighbourhood function used in our model is

$$u([x, y])$$

$$= ([x - 1, y], [x + 1, y], [x, y + 1],$$

$$[x, y - 1], [x + 1, y - 1], [x - 1, y + 1],$$

$$[x - 1, y - 1], [x + 1, y + 1]). \tag{4}$$

The basic schemes of the neighbourhood function and possible directions of spread are shown in Fig. 1.

From the fire-spread interpretation point of view,

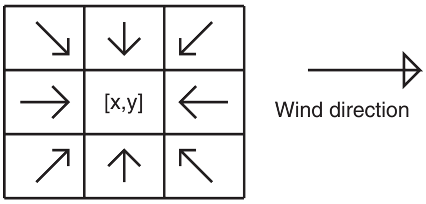

Fig. 1. Basic schemes of neighbourhood function and spread directions.

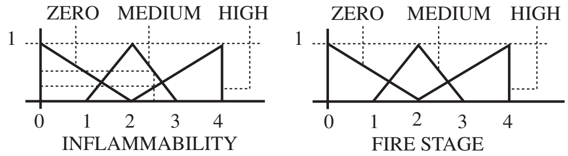

Fig. 2. Fuzzy sets *inflammability* and *fire stage.*

we can presume that the spread of disturbance (fire) is more or less accelerated in the direction of the wind. The acceleration of the spread depends on input data. Every cell is determined with an internal data structure presented in Exp. (5).

```
struct cells
{ int fire_stage; /* q(x,t)*/
  int inflammability; /* q(x,t)*/
  int new_stage; /* q(x,t+1)*/
} cell[max_x,max_y];                (5)
```

The first variable represents the stage of fire in one cell in step t (inference step index), the second one represents the inflammability stage of the cell, and the third one represents a temporary location in the decision process for determination of the *fire_stage* in the step $t+1$. All input variables are normalised in expected intervals. The last input variable used in the decision process is *wind_speed*(t), which represents the speed of the wind. In terms of fire spread, greater wind speed means greater fire spread and lower wind speed produces lower fire spread.

3.1. Fuzzifying input data

In binary logic, every cell x in step t can be in exactly one state q, $(q \in Q)$. In fuzzy logic the cell can be simultaneously in a number of states, depending on their values and membership functions. From our application's point of view [5], we have used a state description of two parameters. The first one is the *inflammability* of the cell's internal property and the second one is *fire_stage* in step t. Figure 2 shows an example of the fuzzification process for both input variables. From Fig. 2, we can can see that if *inflammability* = 2.5 (it can be calculated or estimated from the real system), then $\mu(t, i)_{BigInflamm} = 0.25$, and $\mu(t, i)_{MediumInflamm} = 0.5$. These expressions represent the membership values for different terms of *inflammability*. Explanation: *Inflammabilty* 2.5 belongs to the class "*Medium Inflammability*" with membership 0.5 and to class "*High Inflammability*" with membership 0.25. In the same way other input variables are fuzzified. These are:

- *fire_stage* = {*zero, medium, high*};
- *inflammability* = {*zero, medium, high*};
- *wind_speed* = {*zero, medium, high*};

Every exact input value can belong to more terms (descriptive classes) with calculated membership from interval $[0, 1]$, which depends on the definition of membership functions. The decision-making phase as the next step uses terms as input variables.

3.2. Fuzzy decision process

Rule Exp. (6) shows an example of a fuzzy rule from the application [5]. It is used to determine the fire stage in the cell $[x, y]$ (2-D model) in step $t+1$, on the basis of input data which are neighbourhood states and the cell's $[x, y]$ fire state in step t.

```
if (cell[x,y].fire_stage = "medium")
AND (cell[x,y].inflammability =
     "high")
AND (wind_speed = "high")
AND (cell[x-1,y].fire_stage = "high")
then (cell[x,y].next fire stage
     = "high").                       (6)
```

The logic interpretation of the presented rule is: *if fire in the cell is medium and if the cell's inflammable stage is high and if the wind speed is high and the fire in the left neighbouring cell is high then the fire in the cell on the next step will be high.*

The rule set consists of 288 rules. All of them are of the same type as the example in Exp. (6). They describe how the disturbance is carried from neighbouring cells to the central one.

After the sequence of events over activations of rules we get an output fuzzy value. It means in fuzzy logic terms that membership of output terms is assigned with max. value during iterations. It is presented with the same three terms as the *fire_stage* variable. The last step of the algorithm (see Exp. (3)), which has to be performed is the defuzzification process of the output variable. In our experiments, the COG method was used [7, 10].

4. Results

In this section we wish to present the results of the simulation compared with the statistical method's results. The latter method is widely used in practice. The shape of the fire is found on two semi ellipses. The parameters of these depend on the wind speed (global dynamic variable) and on a constant, which depends on the inflammability of the area. Fire statistics performed on real fire data show that a ratio between the length (direction of wind) and the width of the fire spread shape is between 1:1 (no wind, equally inflammable area) and 1:6 (strong wind in the height of flames, equally inflammable area). The results of our fuzzy method show similar ratios. Figure 3 shows the shape (grey shape) as a result of the proposed fuzzy method compared to the statistical result (2 semi ellipses). Figure 4 shows the statistical shape compared to the real fire spread shape and finally, Fig. 5 presents the results of the proposed fuzzy method compared to real data. The real and statistical data are resumed from [2].

The results presented in Figs 4 and 5 show us that the statistical method gives slightly better results in comparison to our results. The main reason for this is a lack of data on the change of wind (speed and direction) not mentioned in the source data [2]. The main disadvantage of the statistical method is in the large number of experiments which must be performed on a specific area to get the parameters of the semi ellipses. We suppose that in the case of a randomly chosen wild fire location (in which case we lack specific statistical data) our method can give better results than the statistical approach.

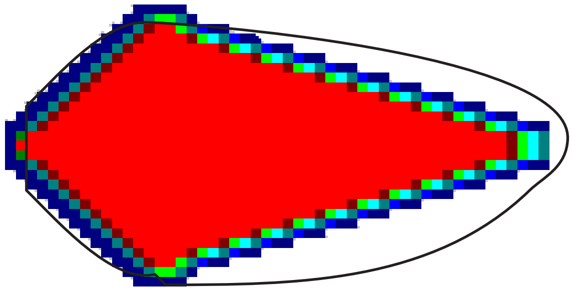

Fig. 3. Statistical approach results compared to fuzzy CA.

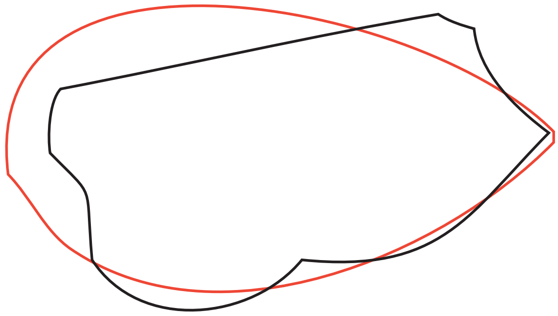

Fig. 4. Statistical approach results compared to real data.

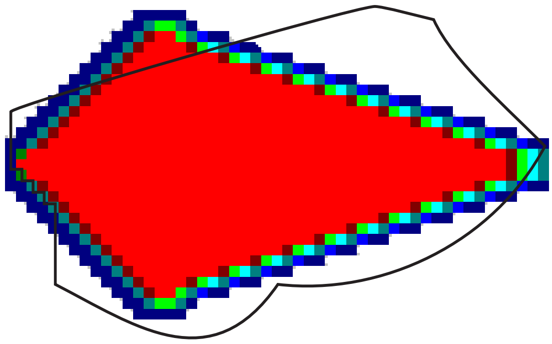

Fig. 5. Fuzzy CA approach compared to real data.

5. Conclusion

We have built a relatively simple cell network, with a fuzzy rule based spread of disturbance. Fuzzy logic enables us to use descriptive uncertain knowledge about the simulated system's behaviour. The proposed modelling approach could also be used in the field of healing wounds, the spread of diseases, social phenomena etc. The only disadvantage of processing fuzzy rules is the increased processing time. Our main intention in future is to build a universal application for cell space behaviour modelling. It will be built with improvements in terms of:

- optimised speed of processing fuzzy arithmetic operations

- graphic interface to provide interactive work with decision details
- the target platform has to be a high speed performance workstation
- fuzzy logic alternative approaches built in (fuzzification, operators,..)
- distributed processing of cell's state transition on more platforms.

Acknowledgements

This article presents part of doctorial thesis being prepared by M. Mraz at the Faculty for Computer and Information Science in Ljubljana.

References

[1] A.I. Adamatzky, Hierarchy of fuzzy cellular automata, *Fuzzy sets and Systems* **62** (1994), 167–174.

[2] H.A. Anderson, *Predicting Wind Driven Wild Land Fire Size and Shape*, USDA Forest Service, Intermountain Research, USA Ogden, Research Paper INT 305, USA, 1983.

[3] B. Grm, B. Stevanović and M. Mraz, *Wild Land Fire Spread Simulation and Mathematical Models* (MORS's Final Project Report), Slovenija, 1995.

[4] M. Mizumoto, J. Toyoda and K. Tanaka, Some considerations on fuzzy automata, *Journal of Computer and System Sciences* **3** (1969), 409–422.

[5] M. Mraz, N. Zimic and J. Virant, Computer aided predicting wind driven wild land fire size and shape using fuzzy logic, in: *9th International Symposium Computer Science for Environment Protection,* Berlin, 1995, pp. 726–732.

[6] E.S. Santo, Maximin automata, *Information and Control* **13** (1968), 363–377.

[7] J. Virant, *Using Fuzzy Logic in Modern Systems*, Didakta, Radovljica, Slovenija, 1992.

[8] J. Virant and N. Zimic, Fuzzy automata with fuzzy relief, *IEEE Trans. on Fuzzy Systems* **3** (1) (February 1995).

[9] W.G. Wee and K.S. Fu, A Formulation of fuzzy automata and its application as a model of learning systems, *IEEE Trans. on Systems Science and Cybernetics* **5** (5) (July 1969), 215–223.

[10] H.J. Zimmermann, *Fuzzy Sets Theory and Applications*, Kluwer Acad. Press, 1992.