2010 IEEE/ASME International Conference on
Advanced Intelligent Mechatronics
Montréal, Canada, July 6-9, 2010

Artificial Intelligence for Forest Fire Prediction

George E. Sakr, Imad H. Elhajj, George Mitri and Uchechukwu C. Wejinya

Abstract— **Forest fire prediction constitutes a significant component of forest fire management. It plays a major role in resource allocation, mitigation and recovery efforts. This paper presents a description and analysis of forest fire prediction methods based on artificial intelligence. A novel forest fire risk prediction algorithm, based on support vector machines, is presented. The algorithm depends on previous weather conditions in order to predict the fire hazard level of a day. The implementation of the algorithm using data from Lebanon demonstrated its ability to accurately predict the hazard of fire occurrence.**

Index Terms— **Machine Learning, SVM, Forest Fire Prediction.**

I. Introduction

Forest fires are an integral part of many terrestrial ecosystems such as boreal forests, temperate forests, Mediterranean ecosystems, savannas and grasslands, among others. Fires in the Mediterranean basin account for a significant percentage of total fires occurring worldwide [14].

Forest fire prediction, prevention and management measures have become increasingly important. Systems for forest fire-danger prediction represent an essential tool to predict forest fire risks, back up the forest fire monitoring and extinction phase, and to assist in the fire control planning and resource allocation [1].

At present many fire risk models make use of forest fire databases to construct and assess probabilistic models. Brillinger et al. proposed a model for each specific location based on its fire history, its elevation above sea level and the corresponding dates of fire days and non-fire days [3]. The system tries to fit different probabilistic models to data from different locations. When the fitting is accomplished the system is used to estimate the probability of a forest fire taking place at a particular location and time. No weather parameters are used and the main output is the probability of the number of fires being greater than a certain threshold. This probability estimation is helpful for fire fighters' resources allocation. Experiments show that the assessed risk is accurate for a specific area. A shortcoming of this model is its close dependency on the trained area and its inability to generalize the model to nearby areas. Jaiswal et al. implemented a GIS-based forest fire risk model, to study the relationship between vegetation, climate, topography and their associated factor to causing forest-fires [11]. A forest fire risk zone map was constructed using a four-category risk scale. The risk range is from very high to low. The resulting map was found to have a strong correlation with the highly affected fire sites. This method is also dependent on the studied area and the model cannot be generalized to other lands. It is also notable that the use of satellite images and GIS is imperative in the construction of the models. Estimation of forest fire risk on a global scale was introduced by Iliadis who developed a decision support system (FFIR-EDESSYS) that implements fuzzy logic and fuzzy algebra concepts [10]. The system was implemented in Greece, and it showed a good estimation of the forest fire risk areas but it has no indication of the estimated area of the fire which might compromise the resource allocation problem. Some important clustering techniques using partitioning methods like k-means or density based clustering like DBSCAN that are normally used on spatial data, can be extended to spatio-temporal data. Many models implement Spatio-temporal Data Mining (STDM) techniques, and demonstrate that these techniques have great potential in forest fire prediction. Their application to forest fire is described by cheng et al. as follows [4]:

1) Forecasting and trend analysis: basically one can make use of historical data related to burnt area to predict future forest fires. These methods can also provide the ability to forecast the burnt area and the length of fire field.
2) Association rule mining for prediction of ongoing forest fire development: This is based on spatial forest data like the slopes or position of the slope as well as weather data (precipitation, wind speed and direction, temperature) and fuel type, to predict the spreading of the fire. It will add the ability to create logical conditions such as: if a fire occurs in W then it is very likely to spread towards S and allows forest fire fighters to obtain an optimal plan.
3) Pattern Detection for sequence of fire events: using spatio-temporal data to discover sequential patterns

This research was funded by the American University of Beirut University Research Board, Dar Al-Handassah (Shair & Partners) Research Fund, the Rathman (Kadifa) Fund and the Association for Forests, Development and Conservation.

G. E. Sakr is a PhD candidate at the American University of Beirut, Beirut, Lebanon. (Phone: +961-1-350000; e-mail: ges07@aub.edu.lb)

I. H. Elhajj is with the Department of Electrical and Computer Engineering, American University of Beirut, Beirut, Lebanon. (Phone: +961-1-350000; fax: +961-1-744462; e-mail: ie05@aub.edu.lb)

G. Mitri is the Director of Biodiversity Program, Institute of Environment, University of Balamand, Tripoli, North Lebanon. (P.O.Box 100, Tripoli, North Lebanon. E-mail: george.mitri@balamand.edu.lb)

U. C. Wejinya is with the Department of Mechanical Engineering, University of Arkansas, Fayetteville, Arkansas. e-mail: uwejinya@uark.edu

that occur frequently, and hence having the ability to generate logical rules.

4) Cluster analysis and identification of fire spots: Spatio-temporal clustering may discover the cells (hot spots) that have a high probability of starting a fire. The discrimination of fire spots will have a direct implication on the probability of forest fires.

Li et al. developed a system for automatic identification of fire smoke using artificial neural networks applied to advanced very high resolution radiometer imagery [13], while Wiering and Dorgio were interested in knowing where to cut fire-lines to minimize the damage done by the forest fire [19]. The idea was to build a fire spread simulator and to search for good decision policies. The parameters used to build the spread index are the fuel-type and the wind speed. The system was achieved using evolving neural networks. Naturally this system has to be calibrated on every type of terrain and helps in making policies for fire fighters. Han et al. used predictive geospatial data mining methods and constructed mathematical models to predict the forest fire hazardous areas, based on the FHR (Forest Fire Hazard Rate) and a PRC (Prediction Rate Curve) [9]. Furthermore, new techniques have emerged like cellular automata or agent-based modeling. In the cellular automata models, a forest is considered to be a cellular space that evolves with time. Each cell has an independent and a dependent state. The independent state is its own evolution with time, while the dependent one is the influence of neighboring cells. A burning cell is considered to be able to conduct fire to neighboring unburned cells, which will be useful to model fire propagation process between cells [16]. Cellular automata were applied by Clarke et al. who made use of fractal geometry to predict wildfire propagation and extinction [6]. While Muzy et al. implemented fire spread prediction using cellular automation model (Cell-DEVS)[15] and they combined it with a physical model of fire spread proposed by Rothermel [17]. A new technique was introduced by Dunn and Milne to overcome the problem of terrains of heterogeneous structure [8]. In the old techniques such terrains must be split into sub terrains and each one has to be modeled by itself. This method encodes the spread of wild land fire in a set of interacting automata. The system was able to simulate the fire spread on an irregular field. Cheng and Wang made use of spatio-temporal data mining techniques and the model was focused on predicting the burnt area [5]. Their model uses recurrent neural network to combine historical fire data of the studied landscape as well as weather data. The model was validated on Canadian soil and it achieved errors as low as 0.5ha for an area of 200ha. Another relatively recent approach used data mining and meteorological data to predict forest fires and to estimate the burnt area corresponding to that fire [7]. They tried out several inputs in the aim of finding the best parameters that can be used for prediction. As a result of this they found out that the useful parameters were: wind speed, temperature, humidity and precipitation. They used 5 different data mining algorithms: multiple regression (MR), Decision Tree, Random Forest, Neural Networks and Support Vector Machines. The best results were achieved using support vector machines and a Gaussian kernel function. The overall performance was measured using two techniques: the Mean Absolute Deviation and the Root Mean Square Error. The SVM implementation gave a MAD of 12.71ha and a root mean square error of 64.7ha.

Overall the challenge for a prediction system is how to combine the different indicators in order to make a decision and how to predict a large number of unseen patterns from a few known ones. The prediction has to be accurate, consistent and computationally effective. This paper deals with the prediction problem, it presents an algorithm for fire risk classification over four classes based on the historical number of fires and certain weather conditions. This algorithm is based on support vector machines which is presented in the next section.

II. Proposed Prediction Algorithm and Architecture

The proposed method introduces a fire risk index on a scale of 1 to 4, where 1 corresponds to the lowest fire risk and 4 to the highest fire risk. This index corresponds to the potential number of fires that could occur on a specific day and hence can be used to estimate the actual number of fires on that day. In order to perform prediction, it is required to specify the parameters or features monitored during the day that are used in the prediction algorithm.

A. *Feature selection*

Any prediction mechanism bases its prediction on a continuous observation of a number of specific features. In this paper the aim is to reduce the number of monitored features, and to eliminate the need for weather prediction mechanisms. The reason is to reduce the potential sources of error. If weather prediction is not accurate then it becomes a source of erroneous data, and based on the data mining principle of "junk in, junk out", it will lead to erroneous prediction even if the prediction algorithm was optimal. Another challenge is to choose easily measurable features in the aim of reducing the cost of the system. At the same time, the chosen features must have a high correlation with the risk of fire occurrence: The feature selected are the following:

- The minimal temperature of the day, T_{min}.
- The maximal temperature of the day, T_{max}.
- The average humidity of the day.
- The solar radiation over the day.
- The average wind speed over the day.
- The cumulative precipitation level: taken as precipitation year-to-date starting from October 1. It is worth noting that this value is constant for all points of a certain year considered since the algorithm is applied only for the Lebanese fire season months of June, July, August, September and October, where typically no precipitation occurs.

The strategy used to eliminate weather prediction is discussed in section III. In order to fuse the above features and to make fire predictions Support Vector Machines (SVM) are used. The mechanism is introduced in the next section.

B. Support vector machines

In the simplest form, SVM uses a linear hyperplane to create a classifier with a maximal margin [12]. In other cases, where the data is not linearly separable, the SVM maps the data into a higher dimensional space called the feature space. This task could be achieved using various nonlinear mappings: polynomial, sigmoid and RBF such as gaussian RBF. After the nonlinear transformation, SVM finds a linear separating hyperplane in this new feature space. Not like other techniques, probability model and probability density functions need not be known before building the machine. This is very important for generalization purposes, as in practical situations, there is not enough information about the underlying probability laws and distributions. Since SVM has been recording high accuracies in many fields, and since it has an excellent generalization ability, it is used in the course of this paper.

What follows is an introduction to the theory of SVM and the general equation of the hyperplane that will separate the two classes. In the case of linearly separable data the approach is to find among all the separating hyperplanes the one that maximizes the margin. Any other hyperplane will have a greater expected risk than this hyperplane.

During the learning stage the machine uses the training data to find the parameters $\mathbf{w} = [w_1 w_2 ... w_n]^T$ and b of a decision function $d(\mathbf{x}, \mathbf{w}, b)$ given by:

$$d(\mathbf{x}, \mathbf{w}, b) = \mathbf{w}^T \mathbf{x} + b = \sum_{i=1}^{n} w_i x_i + b \qquad (1)$$

The separating hyperplane follows the equation $d(\mathbf{x}, \mathbf{w}, b) = 0$. In the testing phase, an unseen vector x, will produce an output y according to the following indicator function:

$$y = sign(d(\mathbf{x}, \mathbf{w}, b)) \qquad (2)$$

In other words, the decision rule is: if $d(\mathbf{x}, \mathbf{w}, b) > 0$ then x belongs to class 1 and if $d(\mathbf{x}, \mathbf{w}, b) < 0$ then x belongs to class 2.

The weight vector and the bias are obtained by minimizing the following equation:

$$L_d(\alpha) = 0.5\alpha^T H \alpha - f^T \alpha \qquad (3)$$

subject to the following constraints:

$$y^T \alpha = 0;$$
$$\alpha \geq 0$$

Where H denotes the Hessian matrix given by:$H = y_i y_j (x_i x_j)$ and f is the unit vector $f = [1, 1...1]^T$. Having the solutions α_{0i} of the dual optimization problem will be sufficient to determine the weight vector and the bias using the following equations:

$$\mathbf{w} = \sum_{i=1}^{l} \alpha_{0i} y_i x_i \qquad (4)$$

$$b = \frac{1}{N} \sum_{i=1}^{N} (\frac{1}{y_i} - x_i^T \mathbf{w}) \qquad (5)$$

where N represents the number of support vectors.

The linear classifier presented above has limited capabilities since it is only used with linearly separable data while in most practical applications data is random and is not linearly separable. The nonlinear data has to be mapped to a new feature space of higher dimension using a suitable mapping function $\Phi(x)$ which is of very high dimension, potentially infinite. Fortunately, in all the equations, this function appears only in the form of a dot product.

From the theory of reproducing kernel Hilbert spaces [2], which is beyond the scope of this paper, a kernel function is defined as:

$$K(x_i, x_j) = \Phi(x_i)^T \Phi(x_j). \qquad (6)$$

Equation (3) has this form in the feature space:

$$L_d(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \qquad (7)$$

subject to

$$\alpha_i \geq 0,$$
$$\sum_{i=1}^{l} \alpha_i y_i = 0.$$

The decision hyper surface in equation (1) is given in the nonlinear space by the following equation:

$$d(\mathbf{x}) = \sum_{i=1}^{l} y_i \alpha_i K(x_i, \mathbf{x}) \qquad (8)$$

The solution of equation (7) yields the hard margin classifier. In general, it is useful to use a soft margin classifier to preserve the smoothness of the hyperplane and prevent α_i from tending to infinity. This classifier is obtained using the same minimization process by just adding one more constraint to equation (7). The constraint is: $0 \leq \alpha_i \leq C$, where C is defined by the user. If C tends to infinity, the soft margin classifier tends towards the hard margin.

C. Detection architecture

The above method has the ability to classify 2 classes and hence it is necessary to introduce an architecture that allows the classification of more than 2 classes. The proposed architecture shown in figure 1 is well know for 4 classes classification [12], [18]. The training and optimization of this architecture is described below. In training SVM 1, points of scale 1 and 2 are put together to form class (-1) while points of scale 3 and 4 are put together to form class (+1). SVM 2 is trained by scale 1 as class (-1) and scale 2 as class (+1). SVM 3 is trained by scale 3 as class (-1) and scale 4 as class (+1). Hence a point that gets classified by SVM 1 as (-1) is passed to SVM 2, which will output its corresponding scale. While a point classified as (+1) by SVM 1 is passed to SVM

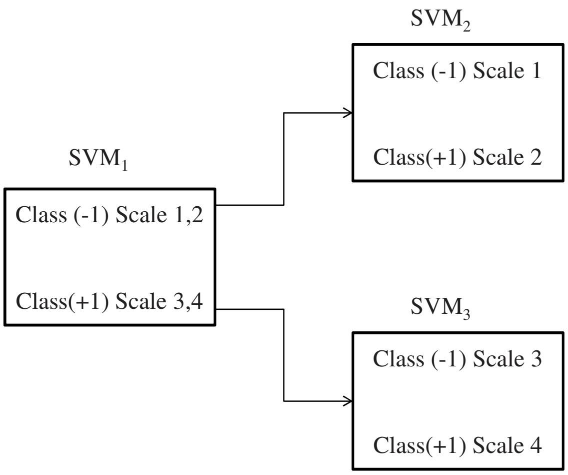

Fig. 1. Four classes prediction architecture.

TABLE I
OUTPUT SCALE.

SVM_1	SVM_2	SVM_3	Output Scale
-1	-1	NA	1
-1	1	NA	2
1	NA	-1	3
1	NA	1	4

TABLE II
CLASS BOUNDARIES.

	Scale 1	Scale 2	Scale 3	Scale 4
June	0	1-3	4-7	≥ 8
July	0	1-4	5-8	≥ 9
August	0	1-3	4-15	≥ 15
September	0	1-4	5-16	≥ 17
October	0	1-7	7-11	≥ 12

3, which will output its final scale. Table I shows the output scale based on the decision of each SVM. As described in section III, each month has its own risk scale definition. For example a day in September where 15 fires have occurred is considered to be a day of scale 3 while 15 fires for June or July is considered a day of scale 4. As a result of this discrepancy, each month has its own architecture trained by training points taken only from that month.

III. EXPERIMENT AND RESULTS

The weather data provided by the Lebanese Agricultural Research Institute (LARI) covers the Lebanese territory and spans the nine years between 2000 and 2008. The data is collected using fixed weather stations across the country. The weather parameters provided are the daily minimum temperature, maximum temperature, average humidity, average wind speed, solar radiation and the cumulative annual precipitation. In addition to the weather parameters, the daily number of forest fire was provided by the Lebanese Ministry of Environment. For each day of weather data from LARI corresponds a specific number of fires that is extracted from the fire list. The daily number of fires over the nine years is used to create the four scales of danger on which the prediction takes place. The proposed method introduces a fire risk index on a scale of 1 to 4, where 1 corresponds to the lowest fire risk and 4 to the highest fire risk. This index is based on the number of fires that occurred on a specific day and hence can be used to estimate the actual number of fires that could happen on that day. Class 1 always corresponds to a no fire day. As for class 2, it corresponds to any number of fires that falls between the first quartile and the third quartile of the number of fire distribution that is given by the historical data. Class 3 corresponds to an increase of risk by 10% from the third quartile and class 4 corresponds to any risk greater than 10%. Table II shows the corresponding number of fires per month per scale which defines the four classes. A challenge is to have this index independent of weather prediction mechanisms; and thus avoid the problem created by potential erroneous weather forecasting. To avoid weather prediction mechanisms, the training points are created as follows: the weather parameters of a given day are associated with the scale of the next day and hence the relationship that is being learned by SVM is the relationship between the weather parameters of today and the number of fires of the following day. This architecture is used for daily prediction. The above architecture could be used for monthly prediction, by associating the average weather parameters of a month with a scale for the following month. Or it could be used for annual prediction by associating the average weather of the year with the scale of the next year. Due to the limited amount of data which corresponds only to 9 years, it is not possible to create an annual classifier or a monthly classifier. The architecture is then tested only for daily prediction. After labeling the points using table II the training of the algorithm is performed. Although each month has different parameters, the training is conducted the same way for all months. Half of the points of a month are taken as training points. The points are chosen in a way to be equally distributed between the four classes. Then the architecture is tested on the remaining points of that same month. This procedure is performed for all the months. The performance of the architecture is tested by computing the average error on the number of fire predicted. Denote by N_i the true number of fires that occurs on day i and by d_i the predicted scale for that same day. Denote also by $q_{min,c}$ the lower boundary of class c and by $q_{max,c}$ the upper boundary for that same class. Then the error on the decision made on day i is given by:

$$Ef_i = \begin{cases} 0 & if\ q_{min,d_i} \leq N_i \leq q_{max,d_i} \\ q_{min,d_i} - N_i & if\ N_i < q_{min,d_i} \\ N_i - q_{max,d_i} & if\ N_i > q_{max,d_i} \end{cases} \tag{9}$$

Another error parameter is the scale error. If C_i is the true class of day i then the scale error is given by:

$$Es_i = |C_i - d_i| \tag{10}$$

The results are presented in table III as follows: the second line corresponds to the expected value of the fire error

TABLE III

PREDICTION RESULTS FOR THE FIVE-MONTH FIRE SEASON.

(a) June

Scale	1	2	3	4
$E[Ef_i]$	0.53	1.3	7.8	2.83
$E[Es_i]$	0.40	0.57	0.78	0.55

(b) July

Scale	1	2	3	4
$E[Ef_i]$	0.58	1.14	7.21	2.77
$E[Es_i]$	0.680	0.60	0.77	0.64

(c) August

Scale	1	2	3	4
$E[Ef_i]$	0.88	0.48	4.45	2.08
$E[Es_i]$	0.46	0.42	0.76	0.64

(d) September

Scale	1	2	3	4
$E[Ef_i]$	0.98	0.97	5.1	5.0
$E[Es_i]$	0.56	0.56	0.93	0.2

(e) October

Scale	1	2	3	4
$E[Ef_i]$	1.21	0.97	2.4	5.63
$E[Es_i]$	0.57	0.61	1.2	1.54

TABLE IV

TWO-CLASS CLASSIFIER RESULTS.

Month	Accuracy
June	89%
July	85%
August	96%
September	90.2%
October	78.4%

computed over all the days of the month and the third line shows the expected value of the scale error taken also over all days of the month. For example, in June the average error on the number of fires when the predicted scale was equal to 1 is 0.53 fires, while the average scale error is 0.4. It is worth noting that the average scale error is less than 0.8 except for four cases and drops as low as 0.2, which shows that the decision on average is very close to the true scale. This 4-class classifier can be specialized for the case of binary classification. By grouping scale 1 and 2 together and scale 3 and 4 together, it is possible to discriminate a high risk day from a low risk day with a high accuracy. The two class accuracy is shown in table IV.

IV. CONCLUSION

The paper presented a forest fire risk prediction mechanism, based only on meteorological data and independent of any weather prediction mechanism. The results demonstrates the ability to predict forest fire risk with a limited amount of data and has shown that support vector machines can be used for a two-class prediction of fire risk with a very high accuracy of up to 96% for August as well as four classes prediction with a low error on the number of fires as well as on the predicted scale.

ACKNOWLEDGMENT

This research was funded by the American University of Beirut University Research Board, Dar Al-Handassah (Shair & Partners) Research Fund, the Rathman (Kadifa) Fund, and the Association for Forests, Development and Conservation. We would like to acknowledge the Lebanese Agricultural Research Institute (LARI) and the Ministry of Environment for providing the data used in this work.

REFERENCES

[1] A. Alonso-Betanzos, O. Fontenla-Romero, B. Guijarro-Berdiñas, E. Hernández-Pereira, M. Inmaculada Paz Andrade, E. Jiménez, J. Luis Legido Soto, and T. Carballas, "An intelligent system for forest fire risk prediction and fire fighting management in Galicia," *Expert systems with applications*, vol. 25, no. 4, pp. 545–554, 2003.

[2] N. Aronszajn, *Introduction to the theory of Hilbert spaces*. Stillwater, Oklahoma: Reasearch [sic] Foundation, 1950.

[3] D. R. Brillinger, H. K. Preisler, and J. W. Benoit, "Risk assessment: A forest fire example," in *Statistics and Science: a Festschrift for Terry Speed*, ser. IMS Lecture Notes Monograph Series, D. Goldstein and T. Speed, Eds. Beachwood, OH: Institute of Mathematical Statistics, 2003, vol. 40, pp. 177–196.

[4] T. Cheng and J. Wang, "Applications of spatio-temporal data mining and knowledge for forest fire," in *ISPRS Technical Commission VII Mid Term Symposium, Enschede*, 2006, pp. 148–153.

[5] ——, "Integrated Spatio-temporal Data Mining for Forest Fire Prediction," *Transactions in GIS*, vol. 12, no. 5, pp. 591–611, 2008.

[6] K. Clarke, J. Brass, and P. Riggan, "A cellular automaton model of wildfire propagation and extinction," *Photogrammetric Engineering and Remote Sensing*, vol. 60, no. 11, pp. 1355–1367, 1994.

[7] P. Cortez and A. Morais, "A data mining approach to predict forest fires using meteorological data," in *Proceedings of the 13th Portugese Conference on Artificial Intelligence*, 2007, pp. 512–523.

[8] A. Dunn and G. Milne, "Modelling wildfire dynamics via interacting automata," *Lecture notes in computer science*, vol. 3305, pp. 395–404, 2004.

[9] J. Han, K. Ryu, K. Chi, and Y. Yeon, "Statistics Based Predictive Geo-spatial Data Mining: Forest Fire Hazardous Area Mapping Application," *Lecture notes in computer science*, pp. 370–381, 2003.

[10] L. Iliadis, "A decision support system applying an integrated fuzzy model for long-term forest fire risk estimation," *Environmental Modeling and Software*, vol. 20, p. 613, 2005.

[11] R. Jaiswal, M. Saumitra, D. Kumaran, and S. Rajesh, "Forest fire risk zone mapping from satellite imagery and gis," *International Journal of Applied Earth Observation and Geo-information*, vol. 4, pp. 1–10, 2002.

[12] V. Kecman, *Learning and soft computing: support vector machines, neural networks, and fuzzy logic models*. London: The MIT press, 2001.

[13] Z. Li, Y. Kaufman, C. Ithoku, R. Fraser, A. Trishchenko, L. Giglio, J. Jin, and X. Yu, "A review of AVHRR-based active fire detection algorithms: Principles, limitations, and recommendations," *Global and Regional Vegetation Fire Monitoring from Space, Planning and Coordinated International Effort*, pp. 199–225, 2001.

[14] G. Mitri and I. Gitas, "A semi-automated object-oriented model for burned area mapping in the Mediterranean region using Landsat-TM imagery," *International Journal of Wildland Fire*, vol. 13, no. 3, pp. 367–376, 2004.

[15] A. Muzy, T. Marcelli, A. Aiello, P. Santoni, J. Santucci, and J. Balbi, "An object oriented environment applied to a semi-physical model of fire spread across a fuel bed," in *Actes de la conférence ESS 2001 conference*, 2001, pp. 641–643.

[16] L. Ntaimo and B. Khargharia, "Two-dimensional fire spread decomposition in cellular DEVS models," *SIMULATION SERIES*, vol. 38, no. 1, p. 103, 2006.

[17] R. Rothermel, “A mathematical model for predicting fire spread in wildland fuels,” *USDA Forest Service Research Paper INT (USA)*, 1972.

[18] V. Vapnik, *Statistical learning theory*. New York: Wiley, 1998.

[19] M. Wiering and M. Dorigo, “Learning to control forest fires,” in *Proceedings of the 12th international Symposium on Computer Science for Environmental Protection*. Citeseer, 1998, pp. 378–388.