Injecting Dynamic Real-time Data into a DDDAS for Forest Fire Behavior Prediction*

Roque Rodríguez, Ana Cortés, and Tomás Margalef

Departament d' Arquitectura de Computadors i Sistemes Operatius,
Universitat Autònoma de Barcelona, 08193 - Bellaterra (Barcelona) Spain.
dario.rodriguez@caos.uab.es
{ana.cortes,tomas.margalef}@uab.es
http://www.caos.uab.es

Abstract. This work presents a novel idea for forest fire prediction, based on Dynamic Data Driven Application Systems. We developed a system capable of assimilating data at execution time, and conduct simulation according to those measurements. We used a conventional simulator, and created a methodology capable of removing parameter uncertainty. To test this methodology, several experiments were performed based on southern California fires.

Key words: Dynamic Data Driven Application System, Parallel computing, Forest fire prediction, HPC, Evolutionary computing.

1 Introduction

Forest fires are one of the nature's most serious threats. Actually, there exist several tools for mitigating damages caused by fires such as fire propagation simulators, based in some physical or mathematical models, being Rothermel's the most recognized one [13]. However, most simulators of natural phenomena such as hurricanes and fires, are very computing demanding and they required as inputs a wide set of variables whose values are either not well known or estimated prior to execution including a considerable uncertainty degree. In fact, this static restrictions (variable inputs are set up only at the very beginning of the simulation process) is an important drawback because as the simulation time goes on, variables previously initialized could dramatically changed producing misleading simulations results. Therefore, to overcome these restrictions, we need a system capable of either obtaining or estimating the values of the input parameters needed by the underlying simulator correctly and, furthermore, this system must be able of adapting itself dynamically to the constant environment conditions changes, by means of real-time measurements. Those characteristics matches the definition of Dynamic Data Driven Application Systems (DDDAS)[5].

Furthermore, nowadays there is a huge computer power available around the world because of distributed systems such as Grid environments and emerging technologies improvements such as multiple cores and new parallelization

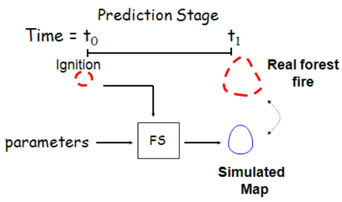
* This work is supported by the MEC-Spain under contract TIN 2007-64974.

techniques. However, most of the current simulation tools are both off-line and sequential presenting slightly time restriction, such as most of the scientific applications. This work represents a step forward to make use of the available computing resources in order to drive this kind of applications to the dimension of the Urgent HPC applications [2].

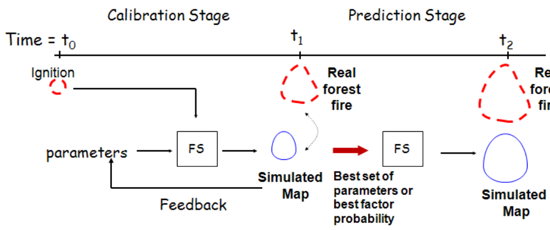
In section 2, we describe the proposed prediction strategy $SAPIFE^3$ a two stage fire prediction method that overcomes time restrictions while reducing the skew in simulation results caused by sudden changes in the weather conditions. A brief description of the module responsible to inject data at run time is included in section 3. In section 4, we present the experimental study and, finally, the main conclusions are reported in section 5.

2 $SAPIFE^3$: a Two Stage Prediction method

Our research team has proposed, in previous works, a paradigm change in forest fire prediction, coming from the classic prediction to DDDAS methods [3] [6]. The classic fire prediction scheme sets up only once the simulator's input variables at time instant t_0 (seen figure 1(a)) keeping them constant for the whole prediction phase (also called Prediction Stage). We include another phase, previous to the prediction one (called the Calibration stage), where the simulator's input parameters are calibrated, depending on the observed fire's behavior from t_0 to t_1. The calibrated values obtained at this Calibration stage will be used in the Prediction stage as it can be seen in figure 1(b).



(a)



(b)

Fig. 1. Predicction Methods

In this work, we propose a two stage prediction scheme called $SAPIFE^3$, this is the spanish acronym for *Adaptive System for Fire Prediction Based in Statistical-Evolutive Strategies* (Sistema Adaptativo para la Prediccin de Incendios Forestales basados en Estratgias Estadstico-Evolutivas) [12]. This method couples two prediction schemes: a genetic algorithm and a statistical approach. Subsequently, both methods are described.

Genetic Algorithm: This prediction method uses a Genetic Algorithm (GA) in the Calibration stage. The population used in the GA is composed of n individuals each of them being composed by a particular setting of the underlying simulator input parameters. We call each input parameter's combination a scenario. The $fitnessfunction$ is an error formula that returns the error between the real observed fire propagation map and the simulated propagation map. This error function will be evaluated for each scenario in order to rank them in terms of prediction quality. Since our system is based on a cell automaton, the error function used is the one defined in equation 1 where $InitCells$ are the cells where fire begun, $Cells \cup$ is the union between real and simulated fire spread, $Cells \cap$ is the intersection between real and simulated fire and $RealCells$ are the cells burnt by the real fire. Once all scenarios have been ranked, they will be updated according to elitism, selection, crossover and mutation operation and an improved population will be obtained. Once the evolution process (Calibration stage) is finished, the best population will be used in the statistical module.

$$Error = \frac{(Cells \cup -InitCells) - (Cells \cap -InitCells)}{RealCells - InitCells} \tag{1}$$

$Statistical\ Integration$: Originally, this method was called Statistical System for Forest Fire Management (S^2F^2M)[3]. This method is based on probabilities and has the aim of sweeping the whole search space exhaustively by considering almost all possible combinations of the simulator's input parameters. Obviously, this method generates a huge number of possible scenarios. When S^2F^2M is coupled to the GA, the number of scenarios used is reduced because it will receive as input population the one provided by the GA. Afterwards, S^2F^2M will evaluate the probability of any cell to be burnt or not, i.e. it merges each propagation map generated for all scenarios in a global probabilistic map. It is important to notice that this method uses the same error function (equation 1) as the GA used.

As we have mentioned, $SAPIFE^3$ merges the two above described prediction methods including their advantages and demising their drawbacks. In particular, $SAPIFE^3$ reduces the number of total scenarios from a number such as hundreds of thousands to several hundreds, by optimizing the set of scenarios through the use of a GA. The combination of several individuals improves the results of the GA in case of sudden changes. That is, when conditions change hardly, the best individual found by the genetic algorithm at Calibration stage could be a very bad one in the Prediction stage. Nonetheless, if we consider the whole population, some individuals referred as to bad individuals during the Calibration stage, may be useful in the Prediction stage.

In the following section, we shall introduce the data assimilation module for the proposed architecture.

3 Data Assimilation

The Data Collection System component - (see figure 2) is the responsible to gather all information regarding the fire's environment, such as weather, topography and terrain composition data (the combustible). This module must

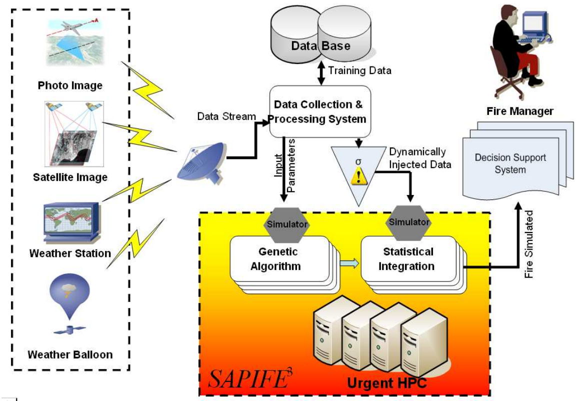


Fig. 2. Conceptual Design

work alongside GIS (Geographical Information System) tools, i.e. MIRAMON [9]. This module must also be well connected to a network of weather stations such as the Network of Automatic Weather Stations of Catalonia's government (XEMA - Xarxa d'Estacions Meterologics Automatiques in catalan).

This module also injects data in real-time. Data is read from the weather stations through *ftp* connections, and then copied to a file inside the execution environment. The process responsible for the statistic integration monitors this file, and in case of changes on it, the changes are introduced in the form of replacing the worst individual who came from the *GA*.

4 Experimental study

In november 2008, southern California was hit by devastating fires. The extreme conditions of the Santa Ana's winds [11], combined with the environment's low humidity, created the ideal conditions for fires as, for instance, the one known as "Freeway Complex Fire", which destroyed around 850 houses, and burnt more than 40.000 acres. The losses due to this fire were about 16 million dollars [7].

In order to test our DDDAS forest-fire propagation prediction system, we performed a series of postmortem experiments based in the conditions of the Freeway Complex Fire. The main objective of these experiments were to demonstrate the benefits of DDDAS for forest fire prediction, specially when environmental conditions are quite dynamic showing suddenly changes in wind speed and wind direction. This way, we are demonstrating the importance of the DDDAS systems for forest fire prediction, and in what way they affect the fire simulators' output, when conditions are dynamic and changes are sudden.

The Freeway Complex Fire happened between the cities of Corona, Chino Hills, Yorba Linda, Brea and Anaheim, in the state of California. In this region, there are several weather stations, property of the Weather Underground [14]. The one chosen to gather data for our experiments was the KCAYORBA4 weather station, located at latitude 33.88 and longitude -117.79, inside the area affected by the fire.

This station was chosen because it monitors humidity, wind speed and wind direction every five minutes. We also used the MODIS Hotspot detection system [10], which allows fire data to be visualized into Google Earth using KML languaje [8] , so it is possible to verify the situation of the fires. Figure 3 shows data for november 16, 2008 where it is possible to visualize the KCAYORBA4 weather station at the bottom of the image.

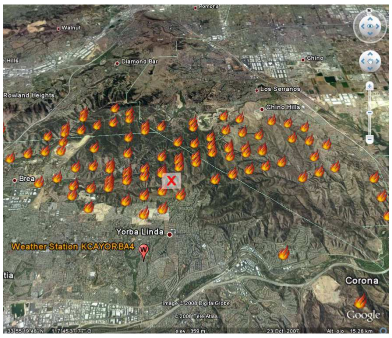


Fig. 3. Freeway Complex Fire view using Google Earth and MODIS Hotspot

The data available for this fire is quite extensive, therefore, in order to have reasonable experimentation times, we cropped the data region into a one square kilometer plot, with a slope of five degrees. This selected area is marked in figure 3 with X. In the reported experiments we recreate the wind conditions according to the data gathered by the KCAYORBA4 weather station in november 16, 2008, between 4:00 and 5:20 a.m. During this time spam, relevant changes in enviromental conditions occurs in a small period of time what allows us to show how sudden changes can affect traditional fire simulators, such as FireLib [4]. We also show how to improve spread prediction results applying DDDAS methods.

The evolution of wind speed and wind direction for the selected time interval is shown in figure 4. As we can observe, the behavior patterns both for wind direction and wind speed are quite fluctuating denoting huge variabilities in five minutes intervals as, for instance, between minutes 10 and 15, where wind speed changes from 2.7mph to 10.1mph. The same effect can be observed in wind direction, which changes in almost every time interval. Those changes are impossible to be predicted and that is the reason why the real-time dynamic data injection could became a crucial point.

As it was described in previous sections, the proposed DDDAS for forest fire behavior prediction needs to be fed with the map of the real observation of the fire propagation, for calibrating purposes. Since our experiments are dealing with a recreation of the real fire, we generate a synthetic real map propagation

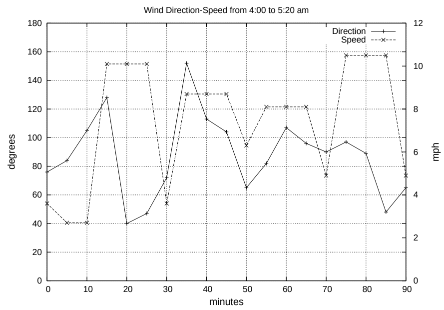


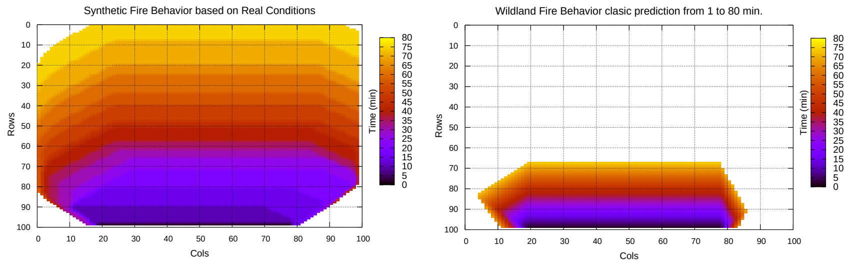
Fig. 4. Changes in wind speed and wind direction during one hour and twenty minutes

by manually varying wind speed and wind direction every 5 minutes following the KCAYORBA4 weather station data pattern. Consequently, we obtained a different propagation map every five minutes. All these maps were joined together, in one single propagation map, that goes from minute 0 to 80. We call this map the "synthetic propagation map based on real conditions" - see figure 5(a). However, in future work, this real map would be an aerial or satellite image of the fire's evolution.It is important to note that the only parameters that changed at each time interval were wind speed and wind direction. All other simulator parameters, such as vegetation model number 4 (Mixed Chaparral - typical from southern California [1]), slope and humidity, were kept constant during the whole simulation.

Figure 5(b) reproduces the propagation map generated by the simulator, when wind speed and wind direction are introduced at time zero and kept constant throughout the simulation (from minute 0 to minute 80). This case shows the prediction results provided by the classical prediction method where a single input parameters measurement are used for the complete simulation process. Comparing this propagation map to the "synthetic propagation map based on real conditions" we can state the bad prediction quality provided by the classical prediction scheme because of the lack of considering dynamic conditions. In particular, the prediction error rate obtained in this case is more than 90% what is clearly unuceptable.

4.1 Experimental results

As it was described in section 2, SAPIFE^3 is composed by the Genetic Algorithm (GA) and the statistical scheme S^2F^2M. In the proposed experiment, the real-time data injection is done after the GA stage and just in the beginning of S^2F^2M. This modification of the basic SAPIFE^3 has been called SAPIFE$^3_{rt}$. The particular GA's configuration for the reported experiments is: population size 500, generation 5, elitism 20, crossover probability 0.2 and mutation probability 0.01. The experimental results shown in this section, include the prediction results provided by those three dynamic data driven schemes.



(a) (b)

Fig. 5. Dimensions: 1000x1000 meters, $10m^2$ cells, 5 slope, model 4(Mixed Chaparral)

Slope and vegetation model are assumed to be known, therefore they are set up as constants inputs for all experiments and schemes. As it has been previously described, the measurement of wind speed and wind direction are available, not only at the very beginning of the fire, but also every 5 minutes (recorded by KCAYORBA4 weather station). Although these data availability, the only dynamic data driven prediction scheme that can take advantage of such information is SAPIFE$^3_{rt}$ because of its ability to receive real-time data at execution time.

Since all compared methods work in a two stage scheme, we shown in two different graphics the results provided at the first stage (calibration stage, figure 6) and after the second stage where the whole prediction process has been finished (prediction stage figure 7). In figure 6, we can observe that GA is the scheme that provides better error adjustments, that means that it can find a set of parameters (individual) capable of reproducing the fire's propagation in a very similar fashion of the real fire for the time interval used for calibration. SAPIFE^3 also presents good adjustment ratios, however with slightly more error ratios than GA. It happens because when integrating each individual, some of them may not be adequate. Nevertheless, the error ratio is still low. The third column shows SAPIFE$^3_{rt}$ which is the method that denotes higher error rates at the calibration stage. That's happens because SAPIFE$^3_{rt}$ not only injects data at the prediction stage but also applies the same behavior at the calibration stage and, therefore, the data injected during the current calibration stage does not correspond to the time interval effectively used for the calibration stage masking the results. That means that to perform the calibration stage for time interval 0-10 minutes, the system is injecting data measured at a time posterior to minute 10. Although it looks like a disadvantage, when sudden change occurs it can be a very useful characteristic as it will be reported subsequently.

Figure 7 depicts the prediction error provided by each method once the prediction stage has finished. In this comparison we included the FireLib results representing the classical prediction method. As FireLib has no calibration stage, it

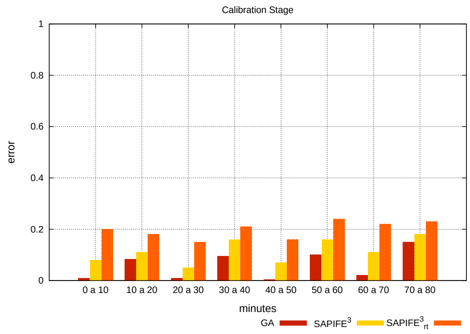


Fig. 6. Comparison between three methods at Calibration Stage

was not included in the previous figure. It is important to notice that, although we are depicting time intervals that exactly last 10 minutes, in fact, the prediction results are provided before reaching the end of the corresponding interval time. However, we can not evaluate the goodness of the obtained prediction until reaching the end of the underlying time interval. That is the reason we plot as prediction interval the exact times. For example, in a time instant previous to t_{20}, the system will deliver the prediction fire behavior for time t_{20}, however, the real prediction validation will be performed only when fire propagation will reach time instant t_{20}. The same happen at each prediction step as shown in figure 7.

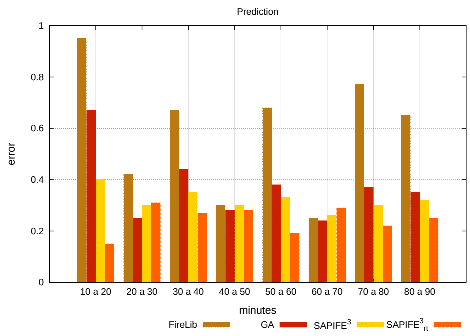


Fig. 7. Comparison between three methods in Prediction Stage

An immediate conclusion obtained from observing figures 7 is that FireLib prediction results are for all time intervals the worst. This fact states that the classical prediction scheme where no dynamic data driven approach is included, is a clear drawback of such a scheme. Furthermore, and taking into account the result discussed from figure 6, one can see that there is no a direct correlation between the results obtained in the calibration stage and the results provided by the prediction stage, in fact, they apparently tend to have an inverse relation. For example, from minute 10 to 20, GA denotes a high error ratio, although at the calibration stage (from minute 0 to 10) it provided the best error adjustment. The same behavior also appears in most of the interval times. GA has an intrinsic drawback related to its impossibility of being aware of drastic changes in environmental conditions from calibration stage to the prediction stage. This penalty is more incident for the case studied bacause of the wind variability pattern used. Therefore, we can affirm that in the presence of sudden changes, the conditions in the moment of the calibration stage does not determine the prediction stage specially when either a classical approach or the GA scheme are applied. $SAPIFE^3$ and $SAPIFE^3_{rt}$ denote the the best prediction results and, in particular, $SAPIFE^3_{rt}$ is shown to be the best. The ability of injecting real-time data allows, for the case studied, to keep bounded the error ratio below 20% although in presence of drastic wind changes.

If we observe wind behavior in figure 4, we can see that it suffers from extreme change on its speed in minute 15. This change is taken into account by $SAPIFE^3_{rt}$ when performing the prediction stage. This fact represents a big advantage, because of this change will generate an increase in fire spread velocity that will not be considered otherwise. Consequentely, $SAPIFE^3_{rt}$ gets almost 50% less error than $SAPIFE^3$, who doesn't have any runtime data insertion mechanism - and who is going to notice the changes only in the range from 20 to 30 minutes. Besides, we can see that the improvement over GA is almost 70%, and more than 90% over FireLib.

In the time frame between 20 and 30 minutes, GA is the one who better performs. This happened because the wind conditions keeps quite similar between the adjustment and prediction phases. This turns the individual found in the 10 to 20 minutes time frame to be very good also for the next period. However, $SAPIFE^3$ and $SAPIFE^3_{rt}$ are very close to it, even in those stable conditions, which keep quite constant until minute 35, when again, they change a lot. This affects seriously the prediction of all methods except for $SAPIFE^3_{rt}$, which is able to get results with error ratios less than 30%.

5 Conclusions and Future Work

In this work, we presented a DDDAS for forest fire spread prediction with real time data injection. We performed a series of experiments based on the behavior of two most variable parameters: wind speed and wind direction. The data used to set up those experiments has been gathered from southern California fires.

The experiment results obtained shown that runtime data insertion improve prediction when conditions change suddenly during a fire. However, this dynamic data insertion must be performed only in the presence of sudden changes, to not disturb simulation results. This will be taken into account for future $SAPIFE^3$ versions, where it will be able to detect sudden changes automatically, and it will be able to decide whether data is going to be inserted or not. This work also demonstrate that a conventional simulator can easily being ported to the proposed DDDAS system having a considerable improvement in its prediction quality. For this reason, we are developing a a general DDDAS framework for any kind of simulator on High Performance Computing (HPC) platforms. In order to introduce the Urgent factor into the systems (Urgent-HPC) we will use SPRUCE (Special PRiority and Urgent Computing Environment)[2] as a authorization system for allocation urgent sessions. This approach will provide new challenges such as dynamic data injection in grid environments.

References

1. Anderson H. E., Forest I., Station R. E., Ogden U.: Aids to Determining Fuel Models for Estimating Fire Behavior, tech. report INT-122, Agriculture Dept. Intermountain Forest and Range Experiment Station,U.S. Forest Service, 1982.
2. Beckman P., Nadella S., Trebon N., Beschastnikh I.: SPRUCE A System for Supporting Urgent High-Performance Computing, Proc. IFIP International Federation for Information Processing, Springer Boston, Arizona, pp 295-311, Jul. 2006.
3. Bianchini G., Cortés A., Margalef T., Luque E.: Improved Prediction Methods for Wildfires Using High Performance Computing: A Comparison, Proc. 6th Int'l Conf. Computer Science (ICCS 2006), LNCS 3991, Springer Berlin, pp. 539-546, 2006.
4. Collins D. B.: FireLib User Manual & Tecnhical Reference, Nov, 2006; http://www.fire.org/
5. Douglas C.: Dynamic Data Driven Application Systems homepage, Jan 2008; http://www.dddas.org
6. Denham M.: Cortés A., Margalef T., Luque E., Applying a Dynamic Data Driven Genetic Algorithm to Improve Forest Fire Spread Prediction, Proc. 8th Int'l Conf. Computer Science (ICCS 2008), LNCS 5103 Springer Berlin, pp. 36-45, 2008.
7. Orange County Fire Authority - OCFA homepage: Freeway Complex Fire Premliminary Report, Febr, 2008. http://www.ci.yorba-linda.ca.us/Fire_Prelim_Report.pdf
8. KML Support homepage: KML Tutorial, Nov, 2008; http://code.google.com/apis/kml/documentation/mapsSupport.html
9. MIRAMON Geographic Information System and Remote Sensing software homepage: What is MiraMon?, Jan, 2008; http://www.creaf.uab.es/MiraMon/what_mm/ENG/index.htm
10. Justice C.O, Giglio L., Korontzi S.,Owens J., Morisette J.T., Roy D., Descloitres J., Alleaume S., Petitcolin F., Kaufman Y.: The MODIS fire products, Remote Sensing of Environment, ELSEVIER., vol. 83, pp. 244-262, 2002
11. Raphel M. N.: The Santa Ana Winds of California, Journal Earth Interactions. vol. 7 no. 8, pp. 1, 2003.
12. Rodriguez R., Cortés A., Margalef T., Luque E.: An Adaptive System for Forest Fire Behavior Prediction,Proc. 11th IEEE Int'l Conf. Conference on Computational Science and Engineering (CSE 2008), IEEE Computer Society, Washington, DC, USA, pp. 275–282, 2008.
13. Rothermel R. C.: How to predict the spread and intensity of forest and range fires, tech. report INT 143, Agriculture Dept. Intermountain Forest and Range Experiment Station Ogden,U.S. Forest service, 1983.
14. Weather Underground homepage: History for KCAYORBA4, Nov, 2008; http://www.wunderground.com/weatherstation/