Early adaptive evaluation scheme for data-driven calibration in forest fire spread prediction⋆

Edigley Fraga, Ana Cortés, Andrés Cencerrado, Porfidio Hernández, and
Tomàs Margalef

Universitat Autònoma de Barcelona, 08193 Cerdanyola del V. (Barcelona), Spain
http://www.uab.cat
edigley@gmail.com
{ana.cortes,andres.cencerrado,porfidio.hernandez,tomas.margalef}@uab.cat

Abstract. Forest fires severally affect many ecosystems every year, leading to large environmental damages, casualties and economic losses. Established and emerging technologies are used to help wildfire analysts determine fire behavior and spread aiming at a more accurate prediction results and efficient use of resources in fire fighting. Natural hazards simulations need to deal with data input uncertainty and their impact on prediction results, usually resorting to compute-intensive calibration techniques. In this paper, we propose a new evaluation technique capable of reducing the overall calibration time by 60% when compared to the current data-driven approaches. This is achieved by means of the proposed adaptive evaluation technique based on a periodic monitoring of the fire spread prediction error ϵ estimated by the normalized symmetric difference for each simulation run. Our new strategy avoid wasting too much computing time running unfit individuals thanks to an early adaptive evaluation.

Keywords: Forest Fires · Urgent Computing · Data Uncertainty · Data Driven Prediction.

1   Introduction

Fire is a natural element of many ecosystems and even large wildfires are part of a defined disturbance regime [20]. For that reason, the challenge from both a prevention and a suppression point of view is to anticipate and reduce the spread potential of large wildfires, and the succeeding risk for lives, property and land use systems [7]. Wildfires have a relatively unpredictable nature as their spread can vary based on the flammable material and can differ by their extent and wind speeds. Forest fire prevention strategies for detection and suppression have improved significantly through the years, both due to technological innovations and the adoption of various skills and methods. Nowadays, wildfire researchers

---

use technologies that integrate data on weather prediction, topography and other factors to predict how fires spread.

Nevertheless, wildfires still occur widely and represent a permanent threat whose consequences may be catastrophic both in terms of human fatalities, ecosystem degradation and economic losses. For example, years marked by intense drought and dry conditions contribute to the high levels of wildfire activity which could be turned into natural disasters. In the last years, unusually large wildfires severely damaged forests in Indonesia, Canada, United States, Spain, Chile, Portugal, and Australia, just to name a few. Actually, some studies suggests that over the past few decades, the number of wildfires has indeed increased [9, 15, 17, 18].

In light of this situation, forest fire prediction, prevention and management measures have become increasingly important over the decades. Systems for wildfire prediction represent an essential asset to back up the forest fire monitoring and extinction phase, to predict forest fire risks and to help in the fire control planning and resource allocation.

When dealing with the extinction phase, an accurate prediction of the fire propagation is a critical issue to minimize its effects. Actually, in order to be used in a fire extinction activity, a wildfire spread prediction is a *hard-deadline-driven task*. For instance, a complex wildfire simulation that could accurately predict the perimeter of a wildfire, can drive firefighters to put firebreaks where they would be most effective to stop the fire propagation. In this particular case, an accurate prediction that comes up late compared to the actual event is useless to the task of fire suppression. These characteristics represent an urgent computing system, from which the simulation results are needed by relevant authorities in making timely informed decisions to mitigate financial losses, manage affected areas and reduce casualties [14]. The following three urgent computing requirements can be found in the mentioned forest fire propagation system:

a The computation operates under a strict deadline after which the computation results may give little practical value (”*late results are useless*”);
b The beginning of the event that demands the computation is unpredictable;
c The computation requires significant resource usage.

To fulfill those requirements, a solution must be *deadline-driven*, *on-demand provisioned* and *scalable*. To deal with those kind of applications, High Performance Computing (HPC) community usually rely on high-end clusters, on supercomputers or on distributed computing platforms. As deadline-based resource management like resource reservation schedulers and minimum latency dispatch are key-components to urgent computing procedures, much work have been conducted to address this issue [5, 11–13, 23]. In particular, HPC resources involved into simulation should be provisioned and managed in a way that the computation process will be finished within a defined time limit. Those are technologies that aim to support urgent computation dynamically and yet preserve overall machine utilization and the productivity of the other users working daily on the platforms being used.

The use of simulators for forest fire propagation models requires sufficient time for the processing, a precise fuel model data and an accurate knowledge of small and large-scale interaction of weather and topography [7]. To start a simulation, it is necessary a plethora of input parameters, which include spatial data describing the elevation, slope, aspect, and fuel type. Figure 1 illustrates the input data commonly used for fire simulations.
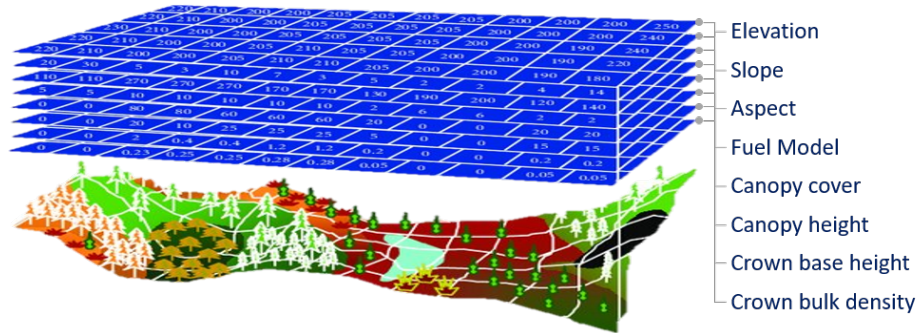

Fig. 1: Fuel and topographic grid data used for wildfire simulations.

In reality, the input data describing the actual scenario where the fire is taking place is usually subject to high levels of uncertainty that represents a serious drawback for the correctness of the prediction [22]. In this paper, we build up our work on a calibration phase that uses a *Genetic Algorithm* (GA) as an optimization technique, proposing a new evaluation strategy capable of reducing the overall calibration time by 60% when compared to the current state-of-the-art data-driven approach for the problem at hand [2]. The new adopted strategy avoid wasting too much computing time running unfit individuals thanks to an early adaptive evaluation for the fitness function.

The remainder of this document is organized as follows. Related works are discussed in Section 2 whereas Section 3 describes the fire spread simulation model and the two-stage prediction method employed to deal with input-data uncertainty. Section 4 presents the proposed evaluation strategy. A case study and computational experiments are presented in Section 5. Finally, Section 6 contains some concluding remarks.

2   Related Works

In a scenario with uncertainties regarding input data, to accurately predict a fire spread under a strict deadline constraint represents a challenge for any designed solution. Current state-of-the-art presents different approaches to tackle data-uncertainty problem, ranging from applying ensemble strategies to soften the uncertainty of input parameters effects to apply Kalman filter to certain input

variables in order to tune their values [21]. Another approach is to resort on computing-intensive methods to relieve such data uncertainty effects, namely a Two-Stage prediction method, composed of a *Calibration* and a *Prediction* stage, wherein the former the input parameters values are adjusted to better reproduce the observed past behaviour of the fire, and the latter where those calibrated parameters are used to forecast the forest fire spread evolution [8].

With regard to fire simulators, FARSITE [10] is a well-known fire growth simulation modeling system which uses spatial information on topography and fuels along with weather and wind inputs. To improve the accuracy of wildfire spread predictions, Srivas [21] extended FARSITE to incorporate data assimilation techniques based on noisy and limited spatial resolution observations of the fire perimeter. The adjustment is calculated from the Kalman filter gain in an Ensemble Kalman filter, based on a Monte-Carlo implementation of the Bayesian update problem. Uncertainty on both the measured fire perimeter and the simulated fire perimeter is used to formulate optimal updates for the prediction of the spread of the wildfire.

In order to cope with the input data uncertainty related with fire spread simulation, Abdalhaq [1] proposed a two-stage methodology to calibrate the input parameters in an adjustment phase so that the calibrated parameters are used in the prediction stage to improve the quality of the predictions. Cencerrado [6] applied Genetic Algorithm as the calibration technique in the adjustment phase, which requires the execution of many simulations to generate the best calibrated set of input parameters. Similar work was also carried over by Méndez-Garabetti et al [16]. Cencerrado also devised one strategy based on *Decision Trees* to identify long running execution individuals of a fire spread simulation. Such strategy was the base for a classification method that allows to estimate in advance the execution time of a simulation given a certain set of input parameters.

More recently, Artés [2] proposed and evaluated a set of resource allocation policies to assign more computing resources to estimated long running executions and less resources to the fast ones, allowing to reduce the adjustment stage time to a more acceptable deadline. That was possible due to the use of a parallel version of FARSITE model that could reduce long running execution times by 35% [3]. To work in a time-constrained fashion, a hybrid MPI-OpenMP application based on the Master-Worker paradigm was developed to take advantage of the execution in a parallel HPC cluster environment. The Two-Stage framework has been proved to be a good methodology to deal with the input data uncertainties and it is leveraged in this current work.

3   Two-Stage Prediction Method

Usually, to predict forest fire behavior a simulator takes the initial state of the fire front perimeter (P_0) along with other parameters as input. As output, the simulator then returns the fire front spread prediction for a later instant in time ($\hat{P}_1$). After comparing the simulation result with the actual advanced fire front
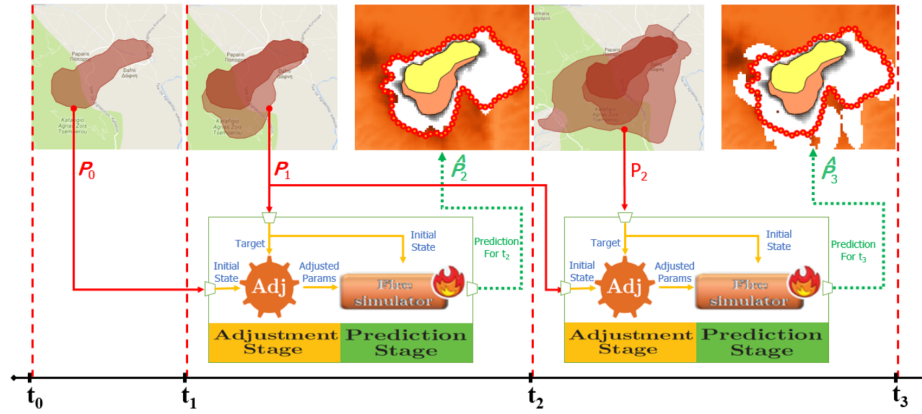
Fig. 2: Two-Stage Prediction Method.

(P_1), the predicted fire line tends to differ from the actual one. Besides the natural phenomena modeling complexity uncertainty, the reason for this mismatch is that the classic scheme calculation is based solely on a single set of input parameters, affected by the aforementioned data uncertainty. To overcome this drawback, a simulator independent data-driven prediction scheme was proposed to calibrate model input parameters [8].

Introducing a previous adjustment stage (see Figure 2), the set of input parameters is calibrated before every prediction step. Thus, the solution comes from reversing the problem, coming up with a parameter configuration such that the fire simulator would produce predictions that match the actual fire behavior. After detecting the simulator input that better reproduces the observed fire propagation, the same set of parameters is used to describe the conditions for the next prediction $(\hat{P}_2)$, assuming that meteorological circumstances remain constant during the next prediction interval. Then, the final prediction becomes the result of a series of automatically adjusted input configurations. The process can be applied again for subsequent fire perimeters $(\hat{P}_3)$, $(\hat{P}_4)$, $(\hat{P}_5)$ and so on.

In order to enhance the quality of the predictions, as a data-driven scheme, the two-stage method is applied continuously, providing calibrated parameters at different time intervals and taking advantage of observed fire behavior and helping to reduce the negative effects related to the input-data uncertainty. This approach has been proven to be appropriate in order to enhance the quality of the predictions. In particular, a Genetic Algorithm based adjustment technique gives accurate results [8] although not being able to give fast response times even when using multi-core allocation strategies, as showed in Section 5.

3.1 Forest Fire Spread Prediction Model Simulator

In the field of forest fire behavior modeling, there is a few fire propagation simulators, based on different physical models, whose main objective is to predict

the fire evolution. Among those, FARSITE [10] is a well-known fire growth simulation modeling system which uses spatial information on topography (terrain) and fuels along with weather and wind inputs. With FARSITE it is possible to compute wildfire growth and behavior for long time periods under heterogeneous conditions of terrain, fuels, and weather. It incorporates existing models for surface fire, crown fire, spotting, post-frontal combustion, and fire acceleration into a two-dimensional fire growth model. A FARSITE simulation generates a sequence of fire perimeters representing the growth of a fire under given input conditions. For that purpose, it incorporates, among others, the simple but effective Rothermel's surface fire spread behavior model [19] along with the Huygens's Principle of wave propagation [10]. Although being a deterministic modeling system, a forest fire spread simulated with FARSITE is a process inherently complex, from which a long execution time for an individual simulation is not atypical.

3.2   Genetic Algorithm Implementation

The adjustment stage is based on a genetic algorithm implemented in a Master-Worker paradigm. The calibration starts from an initial random population of individuals, each one representing a scenario to be simulated. An individual is composed of a set of different genes that represent input variables such as dead fuel moisture, live fuel moisture, wind speed and direction, among others.

Each individual is simulated and it is evaluated comparing the predicted and the real fire propagation by estimating the fitness function (or prediction error function) based on the normalized symmetric difference between predicted and real burned areas. Equation 1 defines how such difference is calculated, where *Real* is the area burned by the real fire at a certain time and *Pred* is the area burned by the predicted fire at the same time instant.

$$SymDiff = \frac{\bigcup(Real, Pred) - \bigcap(Real, Pred)}{Real}$$
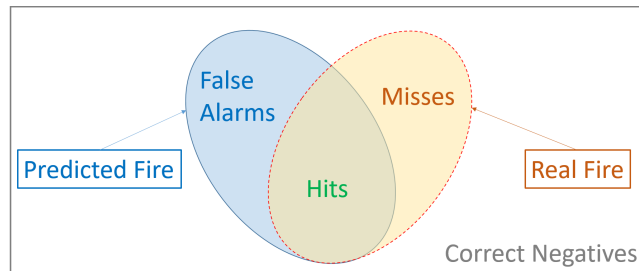$$= \frac{Misses + FalseAlarms}{Misses + Hits} \tag{1}$$

Fig. 3: Different categories present in forecast verification.

As illustrated in Figure 3, the areas around the simulation map that have not been burned by neither the real fire nor the simulated fire are considered *Correct Negatives*. Areas that have been burned by both fires are called *Hits*. The areas that have been burned only in the real fire are called *Misses* whereas areas that have been burned only in the simulated fire are called *False Alarms*.

Due to the high parallelizable characteristics of GAs, its implementation in a Master-Worker paradigm is straightforward. One can take advantage of stable tools, platforms and programming language support to exploit the execution in a parallel HPC cluster environment. Figure 4 illustrates how the adjustment phase may be implemented in a Master-Worker paradigm, with one worker node per individual and the master node acting as a coordinator.
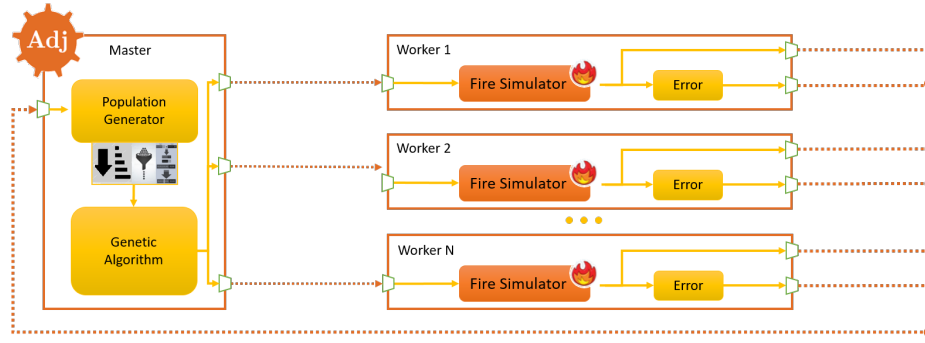

Fig. 4: Genetic Algorithm implemented in a Master-Worker paradigm.

In the current implementation, those individuals whose execution time is estimated to be longer than a preset value are discarded. In this type of situation, if the GA drives the system to a search space associated with parameter settings whose correspondent simulation takes longer than the preset value, this search space will never be considered. In the following section, a technique to overcome this drawback is proposed.

4   Early Adaptive-Evaluation Strategy

In order to overcome the slow time-to-result characteristic of the genetic algorithm, we propose an ***adaptive evaluation*** technique based on a periodic monitoring of the fire spread prediction error ε estimated by the normalized symmetric difference, as defined in Equation 1, for each execution of the individuals. Figure 5a shows the steps done by the monitor process to determine whether the monitored individual should be early terminated or not. Figure 5b depicts two individuals being executed and monitored according to the monitor flow defined in Figure 5a. The one described in Figure 5b (I) terminates normally whereas the other described in Figure 5b (II) is early terminated by the monitoring agent due to its unfitness based on its ongoing prediction error.

Individual Execution
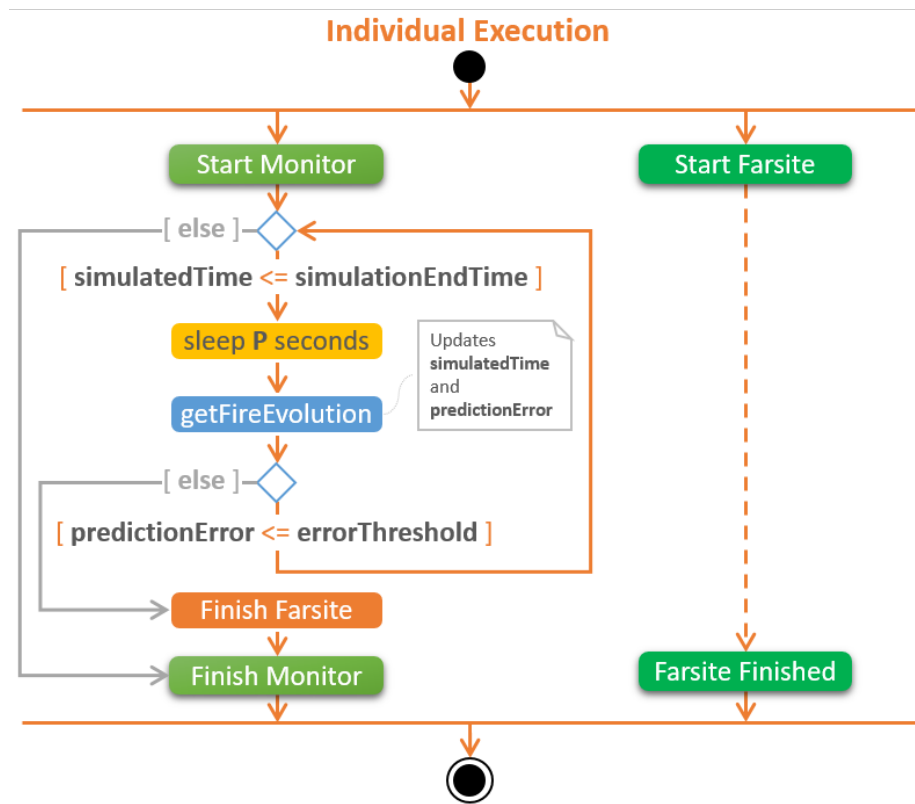
(a) Monitor activity diagram.
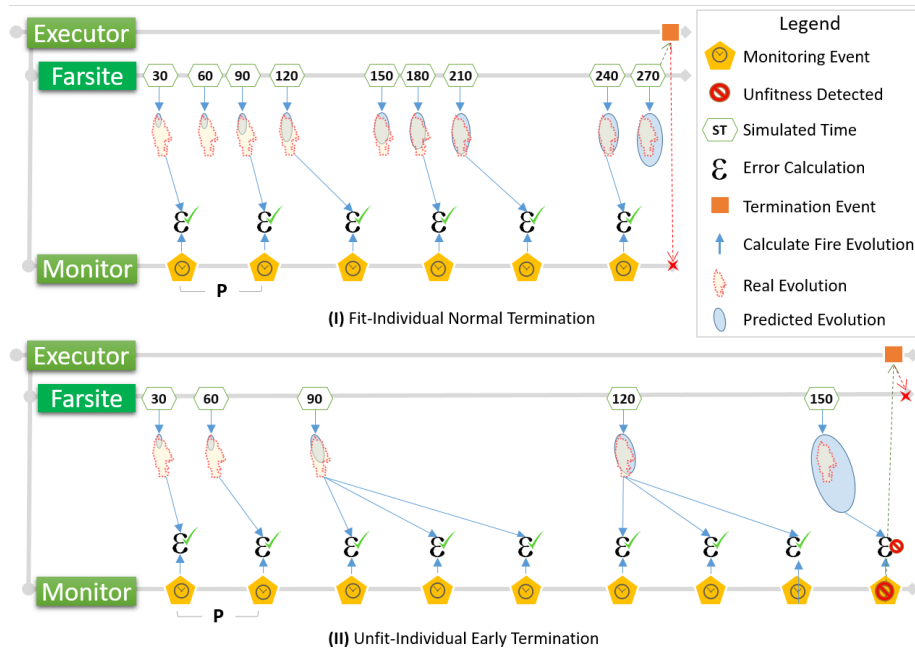
(b) Termination Scenarios.

Fig. 5: Individual termination strategies.

The reasoning behind the early termination is to avoid wasting computing time running individuals that are doomed to unfitness. If along its execution the monitor agent detects that the prediction deviates too much from the actual fire spread, it is considered safe to early terminate the individual. Each monitoring agent is launched with a prediction error threshold T, i. e. a maximum tolerable error above which the monitoring agent must early terminate the individual, and a monitoring period P, usually defined in the order of dozen of seconds.

For example, if we consider the canonical scenario of a fire prediction evolution described in Figure 6, together with their corresponding normalized symmetric difference (prediction error ε) calculated as described in Equation 1, we notice that as the prediction overspread beyond the real fire, the prediction error increases at a rate higher than the ones circumscribed to the real fire perimeter. Therefore, if at an earlier stage of the prediction a monitor configured with a threshold $T=2.0$ detects a fire evolution similar to the one showed in Figure 6f, then, the execution can be terminated as the individual is considered unfit to represent the real fire.
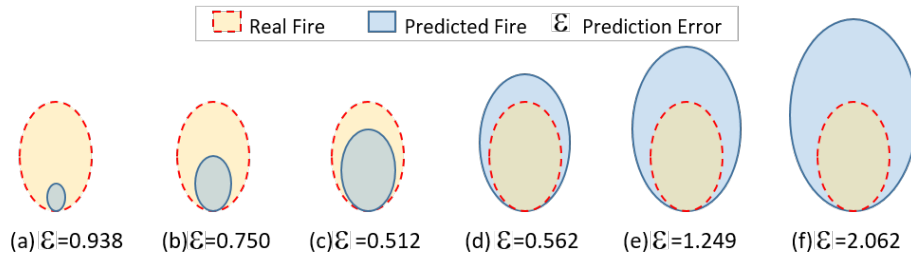

Fig. 6: Normalized Symmetric Difference calculated along fire spread prediction.

In the current implementation, as described in Section 2, the strategy used to early terminate and then discard individuals was based on a deadline previously defined to avoid delaying an entire generation due to longer individuals. Or, even worse, previously filtering out those individuals whose execution time is estimated to be longer than a preset value. The problem regarding early termination is its impact on the population diversity, a crucial characteristic to the genetic algorithm's ability to continue fruitful exploration of the search space.

Another difference from previous works, where the normalized symmetric difference was directly used as the fitness function for the evaluation of the individual, is that, in this proposal, we use a weighted version of it, in order to take into account early terminations. The formula showed in Equation 2 is a weighted version for the normalized symmetric difference in which *PredictionTime* is the total time to be simulated whereas *SimulatedTime* is the total time simulated until a normal or an early termination takes place.

$$fitness = \frac{PredictionTime}{SimulatedTime} \times SymDiff \qquad (2)$$

In Figure 5b(I), in the scenario of a normal termination, for a total simulated time equals to 270 minutes, we can see that the simulated and prediction time are equals, then the fitness function is equals to the normalized symmetric difference. On the other hand, in the early termination scenario showed in Figure 5b(II), the penalty is directly proportional to the simulated time left to a normal termination, i. e., $fitness = \frac{270}{150} \times SymDiff = 1.8 \times SymDiff$. The idea is to be able to compare unfinished individuals results against the actual burned area, considering how much has been simulated until the moment of the early termination: the sooner the termination, the greater should be the penalty. Therefore the main advantage of such strategy is to avoid wasting too much computing time running indubitably unfit individuals. Nevertheless, thanks to the weighted normalized symmetric difference those individuals can still be evaluated and be unlikely selected to the following generations, ensuring more diversity to the evolution process.

4.1   Adaptive-Evaluation and Deadline-Driven strategies comparison

Due to the *fork-and-join* characteristic of the evolution stream defined by a genetic algorithm, the response time of each generation and therefore for the entire evolution is limited by the slowest individual's execution time. Previous works [6] have identified that the execution time distribution of randomly generated individuals follows the characteristic of a power law, with a few small values dominating the distribution, complemented by a long tail representing the slowest individuals. Unfortunately, like in all unusual extreme events modelled by power laws, those infrequent occurrences are responsible for the worst damage to the calibration phase capacity to provide shorter response time.

Then, in order to decrease the overall calibration time, it is crucial to be able to *consistently* decrease the random individuals execution time. The violin plots in Figure 9 (a) compare the FARSITE execution time for a 1 hour deadline-driven strategy and the adaptive-evaluation for a real forest fire scenario (see Section 5). Considering those two execution time distributions, it would be expected that the new strategy helps to calibrate the input data approximately three times faster than any strategy based solely on a deadline-driven approach set to 1 hour per generation. In Section 5 we evaluate the adaptive strategy applied to the aforementioned real fire scenario.

5   Case Study and Computational Experiments

The Mediterranean area is one of the European regions most affected by forest fires during high risk seasons. The selected case study corresponds to a region within the Mediterranean coast that is frequently affected by forest fires. In particular, we used a wildfire that occurred in La Jonquera (North-East of Catalonia, Spain - see Figure 7) in July 2012 that devastated over 13,000ha and caused the death of two people. This wildfire scenario has been thoroughly studied and compared in previous works [2, 4].
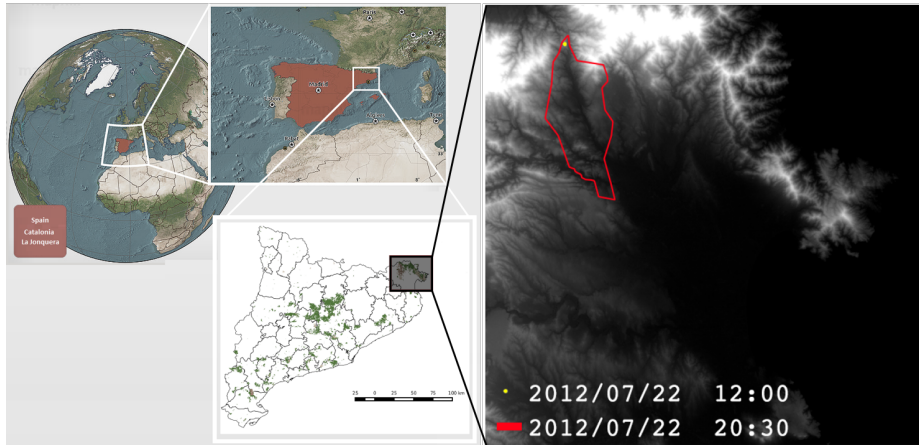
Fig. 7: La Jonquera' Fire (Catalonia - Spain), and its calibration perimeter

The simulated calibration time period was set to match exactly the real observed evolution that goes from 22 July 2012 at 12:00 to 22 July 2012 at 20:30, as depicted in Figure 7. The genetic algorithm has been configured to evolve for 10 generations, each one with population size set-up to 100 individuals. Probabilities of crossover and mutation used in the executions are 0.7 and 0.3, respectively. Each individual is simulated and the resulting simulated fire perimeter is compared to the actual fire spread using the weighted normalized symmetric difference between the real burned area and the predicted one, as described in Section 4. Moreover, each monitoring agent is launched with a prediction error threshold set to 1.5. This implies that the maximum ongoing tolerable error above which the monitoring agent must early terminate the individual is 1.5. Or saying in other words: when the predicted area is one and a half times the size of the one defined by the real fire perimeter, the corresponding simulation will be terminated.

Figure 8 shows an overview of a single calibration execution using the adaptive-evaluation strategy and the deadline-driven approach set to 1 hour per generation for a population of 64 individuals where each individual has been executed sequentially in a single core.

Confronting the two figures, we can notice that the overall calibration time is reduced from above 8 hours to 1 hour on average, representing an improvement of impressive 85%. When compared to the current best data-driven Hybrid MPI-OpenMP Parallel approach [2, 4] dubbed TAC (*Time Aware Core allocation*), the calibration time is reduced from above 3 hours to the same 1 hour on average, resulting in a 60% time-to-result improvement. The TAC approach, which applies classification techniques to detect in advance those individuals that will last longer and allocate more cores to them, was implemented with strict time restrictions: a time lapse of 4 hours was considered to carry out the calibration stage and the final prediction. This time window was split into two intervals: 3
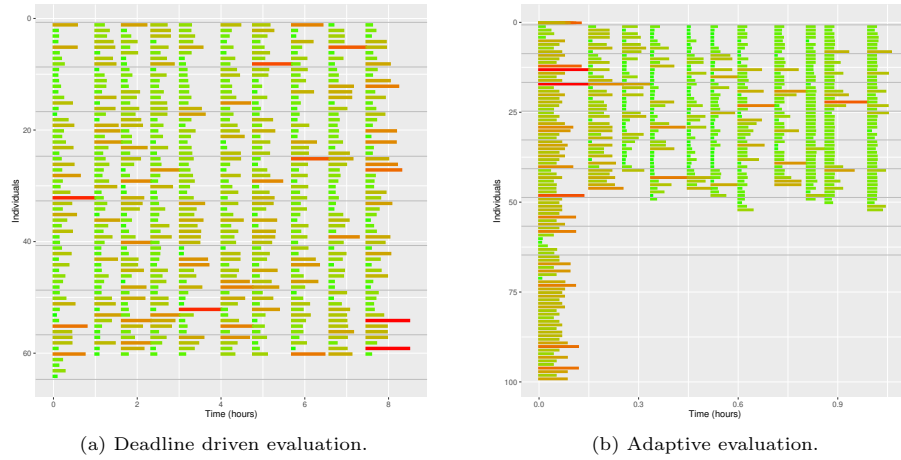
(a) Deadline driven evaluation.

(b) Adaptive evaluation.

Fig. 8: Comparison between different evaluation strategies.

(a) Runtime distributions.
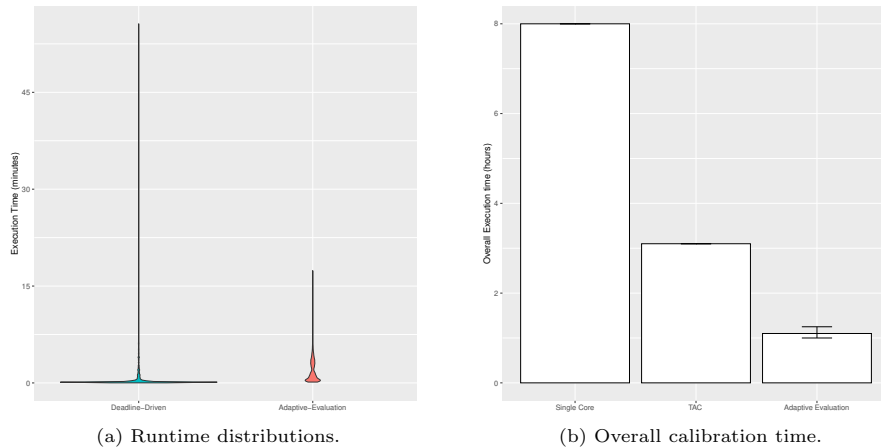
(b) Overall calibration time.

Fig. 9: Comparison between different evaluation strategies.

hours for the calibration, and 1 hour for the simulation that will result in the final prediction.

Figure 9b shows the overall comparison between the three calibration strategies. In relation to the adaptive evaluation metric, the error bars represent the minimum and the maximum values obtained for 20 different calibrations whereas the intermediate point is the average value. For the other scenarios, the bar height represent the average value as reported by Artés et al [2]. As expected, the adaptive evaluation indubitably outperforms the other two strategies, representing a major improvement even when compared to the TAC strategy.

6   Concluding Remarks

Forecasting forest fire spread is a complex problem by itself, considering the plethora of factors capable of influence in this natural phenomena. Apart from that, input data uncertainty represents an additional challenge to devise an accurate model. When applied to a real case scenario in production, there is also an extra key factor that challenge the forecast process, the response time. In such situation, late results are useless.

For this reason, any approach that focuses on accelerating forest fire spread prediction without losing accuracy is more than welcome. In this work, we propose a new strategy able to, at the same time, simplify implementation and improve response time of the critical adjustment stage of a Two-Stage Prediction Method, implemented as a High Performance Computing framework to deal with the input data uncertainties. In that methodology, the prediction phase is preceded by an adjustment stage, the latter being the most time consuming one. In this stage, a genetic algorithm is carried out to calibrate unknown parameters such as fuel humidity and meteorological data.

Previous works exploited heavily cluster environments, classification algorithms and core allocation techniques to be able to deliver acceptable response time under strict time constraints. In contrast, we resort to a monitoring strategy that, apart from being much simpler to implement and easier to productionize, managed to achieve better calibration response time results. In order to validate our proposed strategy, we analyzed a real forest fire scenario that took place in La Jonquera (Catalonia, Spain) in 2012. Results show that the new technique is capable of reducing the overall calibration time by 60% when compared to the current best data-driven time aware core allocation approach.

As future work we intend to propose and evaluate other fitness functions together with the early termination technique and its impact in the overall prediction accuracy.

References

1. Abdalhaq, B., Cortés, A., Margalef, T., Luque, E.: Enhancing wildland fire prediction on cluster systems applying evolutionary optimization techniques. Future Generation Computer Systems **21**(1), 61 – 67 (2005)
2. Artés, T., Cencerrado, A., Cortés, A., Margalef, T.: Relieving the effects of uncertainty in forest fire spread prediction by hybrid mpi-openmp parallel strategies. Proc. Comp. Science **18** (2013), Intern. Confer. on Comput. Science, ICCS 2013
3. Artés, T., Cortés, A., Margalef, T.: Large forest fire spread prediction. Procedia Computer Science **80**(C), 909–918 (Jun 2016), International Conference on Computational Science, ICCS 2016
4. Brun, C., Artés, T., Cencerrado, A., Margalef, T., Cortés, A.: A high performance computing framework for continental-scale forest fire spread prediction. Proc. Comp. Science **108** (2017), Intern. Confer. on Comp. Science, ICCS 2017
5. Brzoza-Woch, R., Konieczny, M., Kwolek, B., Nawrocki, P., et al.: Holistic approach to urgent computing for flood decision support. Procedia Computer Science **51** (2015), International Conference On Computational Science, ICCS 2015

6. Cencerrado, A., Cortés, A., Margalef, T.: Genetic algorithm characterization for the quality assessment of forest fire spread prediction. Procedia Computer Science **9**, 312 – 320 (2012), International Conference on Computational Science - ICCS

7. Costa, P., Castellnou, M., Larrañaga, A., Miralles, M., Kraus, D.: Prevention of Large Wildfires using the Fire Types Concept. European Forest Institute (01 2011)

8. Denham, M., Cortés, A., Margalef, T.: Computational steering strategy to calibrate input variables in a dynamic data driven genetic algorithm for forest fire spread prediction. In: Comp. Science – ICCS 2009. Springer Berlin Heidelberg (2009)

9. E. Dennison, P., Brewer, S., D. Arnold, J., Moritz, M.: Large wildfire trends in the western united states, 1984-2011. Geophysical Research Letters **41** (04 2014)

10. Finney, M.: FARSITE: Fire Area Simulator-Model. Development and Evaluation. USDA Forest Service Research Paper, RMRS-RP-4 (1998)

11. Knyazkov, K.V., Nasonov, D.A., Tchurov, T.N., Boukhanovsky, A.V.: Interactive workflow-based infrastructure for urgent computing. Procedia Computer Science **18**, 2223 – 2232 (2013), Intern. Confer. on Comput. Science, ICCS 2013

12. Kovalchuk, S.V., Smirnov, P.A., Maryin, S.V., et al.: Deadline-driven resource management within urgent computing cyberinfrastructure. Procedia Computer Science **18**, 2203 – 2212 (2013), Intern. Confer. on Comput. Science, ICCS 2013

13. Leong, S.H., Frank, A., Kranzlmüller, D.: Leveraging e-infrastructures for urgent computing. Procedia Computer Science **18**, 2177 – 2186 (2013), International Conference on Computational Science, ICCS 2013

14. Leong, S.H., Kranzlmüller, D.: Towards a general definition of urgent computing. Procedia Computer Science **51**, 2337 – 2346 (2015), International Conference On Computational Science, ICCS 2015

15. Moreno, M.V., Conedera, M., Chuvieco, E., Pezzatti, G.B.: Fire regime changes and major driving forces in spain from 1968 to 2010. Environmental Science & Policy **37**, 11 – 22 (2014)

16. Méndez-Garabetti, M., Bianchini, G., Tardivo, L., Scutary, P.: Comparative analysis of performance and quality of prediction between ess and ess-im. Electronic Notes in Theoretical Computer Science **314**, 45–60 (06 2015)

17. Pausas, J.G., Fernández-Muñoz, S.: Fire regime changes in the western mediterranean basin: from fuel-limited to drought-driven fire regime. Climatic Change **110**(1), 215–226 (Jan 2012)

18. Pausas, J.G., Llovet, J., Rodrigo, A.A., Vallejo, R.A.V.: Are wildfires a disaster in the mediterranean basin? In: International Journal of Wildland Fire (2008)

19. Rothermel, R.: A mathematical model for predicting fire spread in wildland fuels. USDA Forest Service research paper INT, Intermountain Forest & Range Experiment Station, Forest Service, U.S. Dept. of Agriculture (1972)

20. San-Miguel-Ayanz, J., Moreno, J.M., Camia, A.: Analysis of large fires in european mediterranean landscapes: Lessons learned and perspectives. Forest Ecology and Management **294**, 11 – 22 (2013), the Mega-fire reality

21. Srivas, T., de Callafon, R.A., et al.: Data assimilation of wildfires with fuel adjustment factors in farsite using ensemble kalman filtering. Procedia Computer Science **108** (2017), International Conference on Computational Science, ICCS 2017

22. Thompson, M.P., Calkin, D.E.: Uncertainty and risk in wildland fire management: A review. Journal of Environmental Management **92**(8), 1895 – 1909 (2011)

23. Yoshimoto, K., Choi, D., Moore, R., Majumdar, A., Hocks, E.: Implementations of urgent computing on production hpc systems. Procedia Computer Science **9**, 1687–1693 (2012), International Conference on Computational Science, ICCS 2012