Interactive Urban and Forest Fire Simulation with Extinguishment Support

Aitor Moreno, Álvaro Segura, Anis Korchi, Jorge Posada, Oihana Otaegui

Vicomtech, Mikeletegi Pasealekua 57, 20009 San Sebastian, Spain
{amoreno, asegura, akorchi, jposada, ootaegui}@vicomtech.org

Abstract. Fires and other related disasters provoke great destruction of high valuable environments and economical losses, especially when they are located in urban areas. In this work, we present a combined urban and forest fire spreading algorithm to be used in real time and interactive Virtual Simulations. The algorithm is pedagogical oriented and its purpose is not focused in achieving precise results that could be used to predict the fire evolution. The main objective is to obtain a fast, interactive and quasi-realistic Virtual Simulation to be used in the simulation of virtual scenarios where fire-fighters and controllers will be trained. The algorithm supports the main variables involved in the fire spreading (slope and wind) and the radiation effect. An additional method has been added to extinguish the fire.

Keywords: Forest Fire, Urban Fire, 3D City Models, Fire Spreading Simulation, Real Time Algorithm, Environment Process Simulation.

1- Introduction

Fire is one of the most complex and destructive phenomena in Nature. When they are out of control, they can devastate large extensions of forest area or burn buildings provoking economical losses, environmental impacts and even human casualties (see **Fig. 1**).

The preventive measures are very important, but eventually, the fire will start. Whether in urban or in forest areas, we can stop or limit a fire by having a skilled and experienced Fire-Force, from the firemen to the management staff who will organize the available resources to fight the fire.

Emergency Training Centres are a key factor to successful fighting against fire. The firemen and controllers learn all the relevant aspects, theoretical concepts and methodologies. The practical knowledge is achieved thanks to controlled real fire fighting exercises or drills (see **Fig. 2**). But, due to obvious limitations, the Training Centres can't recreate all the possible scenarios (a forest fire with specific weather conditions, a city scale urban environment, etc.) because they are too risky and fire-fighters could be injured during the training.

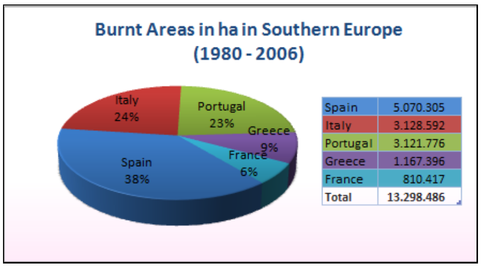

Fig. 1. Burnt Areas in ha in Southern Europe between 1980 and 2006 [6][16].

The most suitable solution to fill this gap is the use of Virtual Simulations, where the fire-fighters and controllers can safely collaborate to fight a virtual fire and check if the concepts learnt in the Training Centre have been applied correctly.

In order to create high quality Virtual Simulations, two main aspects must be fulfilled: *i)* a good reconstruction of the control centre and a proper emulation of the communication process that will be operated by the controllers to make the decisions about the correct deployment of available resources, and *ii)* a highly interactive and collaborative virtual simulation for fire-fighters, controllers, where all the available resources must be represented and operated to extinguish or limit a virtual fire and also, protect the civil people.

Fig. 2: Aircraft Rescue and Fire Fighting Basic Training [13].

One of the main algorithmic elements in such Virtual Simulations is to simulate how fire spreads as simulation time advances. If a very unrealistic or simplistic algorithm is used, the simulation could not be precise enough. This makes the virtual fire behaviour totally predictable during training sessions, which reduce the pedagogical effect. In contrast, introducing the most complex mathematical model in the simulation will increase the realism of the spreading algorithm, but a real time implementation will be less feasible and will also reduce the user interactivity required for a Virtual Simulation.

Although a simplification of the algorithms is needed in order to achieve interactive rates, it is desirable that it could deal with different types of forest and urban fires react to different types of terrain and buildings, slopes and changing weather conditions. The simulation should support the capability of fire to be extinguished by itself (fuel combustion) or by the fire-fighters (fire extinguish agent).

In this work, we present a fire spreading algorithm that can be implemented to be used in real time and interactive Virtual Simulations. The algorithm is pedagogically oriented and its purpose is not to produce accurate results that could be used in scientific or engineering applications. The main objective is to obtain a fast, interactive and quasi-realistic system to be used in the simulation of virtual scenarios where fire-fighters and controllers will be trained.

In the next section, the main fire spreading algorithms will be reviewed. Next, the proposed algorithm and methodology will be introduced, followed by some results. Finally, the conclusions and future work will be addressed.

2- State of the Art

There are two major models of fire simulation, i.e., empirical models and physical models.

The empirical models are made thanks to the experience of real fire, i.e., those models use statistical relationships found between the fire evolution and different parameter tested on the field [14]. In this case, we can mention FARSITE [5], which use Huygens principle of wave propagation.

The second type, physical-based models, use convection and heat transfer mechanism, but also computational fluid dynamics methods. The main mathematical tools used here are partial differential equations and reaction diffusion systems. Fire Dynamic Simulator (FDS, National Institute of Standards and Technology - NIST) or FIRETEC [10] follow this approach.

The advantage of those models is their accuracy in the fire prediction. But the computational effort is very high. The mathematical models are too complex and computers can only provide approximate solutions [4]. Another consequence of the model's complexity is that the spatial resolution required is too high.

Unlike the two previous models, other research works have taken a different direction from the complex mathematical models. Their objective is to reduce computation time and to implement a real time simulation.

Gary L. Achtemeier [1] presented the Rabbit Model, a collection of basic rules of fire evolution, which are implemented as autonomous agents (the rabbits). The scope of the Rabbit Model is limited to the evolution of forest fire.

Hamada [7] is one of the first researchers in urban fire models. His model provided empirical equation describing the speed of the fire spread depending on the wind speed and direction. Hamada's model defines a field compound by identical building blocks separated with the same distance and the fire spread has an elliptical shape [15].

More recently, some physics based models were made [9]. This type of model gives more accuracy in the simulation, using equations to describe the heat transfer between buildings (radiation, convection), the temperature modification, and flame shape (direction, length) coming out of building [8] through the windows.

The physics based models are also used to simulate the fire spread in non-homogeneous cities (on contrary with Hamada's model) with a higher resolution [19].

Other physical based models were proposed using cellular automata with a 9 m² grid cells [11] or a vector based approach in which each building is a vector object (Iwami [8], Tanaka [17]). One limitation of both models is that they limit the fire spread to entire buildings or individual floors, not taking into account the inner structure of the buildings (rooms).

The proposed algorithm in this work presents an urban and forest fire spreading simulation, whose main characteristics are:

- The fire evolution in forest areas is based on the terrain topology, material and weather conditions. In urban areas, the different buildings characteristics are taking into account to simulate more accurate results.
- The fire is allowed to cross rivers, firewall or other barriers by introducing the radiation effect. The radiation effect is also introduced to allow the fire spread between near buildings. In a similar way, urban fire can spread to forest areas and vice versa.
- Very low complexity, allowing real time simulation even with standard computing power.

In the next sections, the details of the proposed method are explained, showing some basic results of the implementation of the algorithm. First, the general approximation of the forest fire and the field definition is introduced. Next, this model is extended with the urban fire model, followed by the extinguishment support.

3- Proposed Method for Forest Fire Spread Simulation

The proposed algorithm main goal is to simulate how the fire spreads in urban and forest environment under different circumstances. In order to explain the method, the field where the fire will be must be defined.

Introduction and Field definition

The forest fire simulation method is based on a divided field in autonomous cells. The field is regularly divided in a grid of square cells and is defined by its geometrical information (origin, size and elevation).

Every square cell contains all the necessary information to support the fire spreading, being divided in two main categories: *i)* static information, which characterizes the cells and will be used in the algorithm rules, and *ii)* runtime information, which comprises all the information that will be modified or calculated during the simulation.

The static information of each cell corresponds with the geometrical data (position, size and altitude of the cell in the field) and the characterization of the cell (semantics), i.e., type of cell which will determine the nature of the cell (dry grass, tall trees…) and the starting quantity of combustible

The fire spreading algorithm is a step-by-step process, i.e., every time that the algorithm is run, a new state of the fire evolution is calculated taking into account the elapsed time.

The runtime information will be continuously calculated during each simulation step, and includes some important variables like the quantity of combustible, the power or intensity of the fire and the state of the fire. All this information is required in order to produce the final rendering of the fire in the Virtual Simulations.

Additional information to support the fire extinguish method are required, but they will be introduced in the corresponding section.

Basics elements of the fire spreading algorithm

The objective is to reduce the algorithmic complexity and the processing time while keeping a correct behaviour of the spreading of the fire. To reach this goal, among the whole set of variables and physical parameters which can influence the fire behaviour, we have chosen the most significant ones.

The most relevant external parameters are the wind and the slope [19]. Other variables, although necessary for precise simulations or predictions have been discarded.

All the cells in the field have an internal state which corresponds with the state of the fire that exists in such cell. The different states are *Safe*, *Activated*, *Burnt*, *Survived*, *RiverCrossing* and *FireStopped* (see **Fig 3**).

When a cell is in the *Safe* state, there is no fire in it. The *Activated* state indicates that there is an active fire in the cell.

The *Burnt* state is the final state of a cell after being completely consumed by the fire. The *Survived* state is pseudo-final state, when the cell is burnt, but still has residual heat. Even being completely consumed by the fire, the cell can irradiate some heat to others cells. Eventually, the cell will pass to the *Burnt* state.

The *RiverCrossing* state is a hidden state used in the computation of the fire spread through a river or firewall, using the radiation effect.

Finally, the *FireStopped* state represents an intermediate state of a cell, when some external fire extinguisher has stopped the fire.

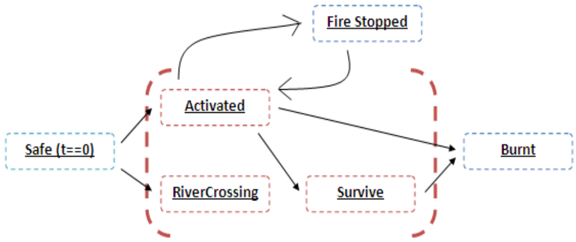

Fig 3. The different states of a cell and the available transitions.

The algorithm that defines how the fire spreads in wild land is based on the cells' initial data of the whole field (combustible, states ...). Following a similar approach to Gary L. Achtemeier in the *Rabbit Model* [1], we define the fire spread as a displacement of *mice*. The combustible on the field is considered as the *mice*'s food (*cheese*). Thus, the basic concept is that when a *mouse* eats a *cheese*, it is equivalent to when a combustible unit is burnt by the fire.

A fire is started by positioning a *mouse* in a square cell, changing its corresponding internal state to *Activated*. The added *mouse* will interact with the neighbour cells following some rules.

Every *mouse* is born in a square with a given power to eat a quantity of *cheese* per simulation step (fire intensity), which changes depending on the type of the cell. When a *mouse* eats a given quantity of *cheese*, its power is increased in the same amount, which simulates how the fire intensity is continuously growing while there is available combustible.

If there is no *cheese* remaining, the *mouse* dies. When a *mouse* dies, up to 8 new *mice* can be born, since there are 8 potential neighbour squares. The algorithm uses the local slope and wind in order to determine which neighbours will be the destiny of the new born *mice*.

The *mouse* will give birth to all the squares which are in an area of +/- 45° of the wind angle. If there is no wind (or it is too slow), the *mouse* will give birth to other *mice* in all the 8 neighbours.

The *mouse* will always try to give birth to other *mice* in squares that are in higher altitude. This behaviour simulates the fact that the fire goes up if there is a slope. Only if there is not a non-burned square at a higher altitude, the *mouse* can go down, as it has no option to go up.

A square can be the destination of multiple *mice* births, as long as it contains *cheese*. Thus, a square can have more than one *mouse*, but the food contained in the square will be more quickly eaten, since their eating power is summed up.

Every square cell has a parameter named *Fire Power*, which is in fact the sum of the *mice's* power of the cell. Similarly, the *Fire Power* of a given cell will be stronger or more intense with a higher number of *mice*.

Elliptical shape support

The Huygens implementation [12] uses differential equations to describe the expansion of an elliptical wave. As the fire evolution is calculated, a typical elliptic shape is obtained.

In this method, the ellipse is simulated by combining the effect of the slope and wind and giving a probability to the mice to give birth in a wrong direction (See **Fig 4**). Those new mice (named *Black Mice*) reduce their power in each step, instead of the normal behaviour. The consequence is that the fire will evolve very slowly in the wrong direction.

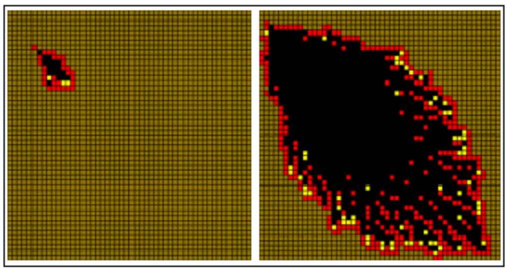

Fig 4. Elliptical shape of a fire in a flat scenario. The wind goes from Top-Left corner to Bottom-Right corner. Black cells are totally burnt, red ones are *Activated*.

Radiation Effect

The radiation effect is added to support the fact that a fire can cross a river, firewall or other barriers like roads. It is based on the heat radiation.

In the proposed algorithm, a fire is stopped when it encounters a barrier. Depending on the *Fire Power* in the area, and considering the width and type (streets, road, river ...) of the barrier, it can be overridden.

The fire is able to bypass the river only if the existing vegetation is composed of trees or any other high vegetation. The main idea is the difference of size: the more a tree is high, the more the wind can spread heat/radiation to the other side of the barrier.

As the river is part of the field, it is also divided in cells (*Water Squares*). The spread of the fire in those squares follows the wind rule, but *mice* in those squares don't eat combustible so the *Fire Power* is never increased. Moreover, when a *mouse* born in water squares, the *Fire Power* is not raised in that square but decreased, i.e., the new *mouse* will have less power than its mother (see **Fig 5**).

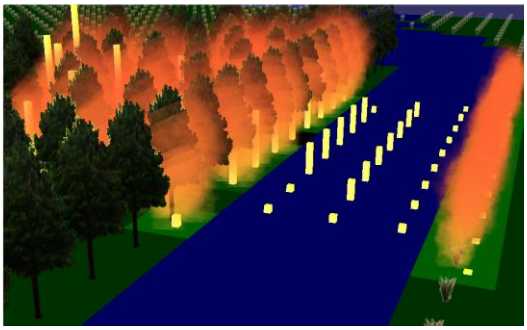

Fig 5. The Fire has bypassed successfully a river. The sticks in the water represent the *Fire Power* and it can be seen that it is being reduced gradually. In the presented case, the fire reached the opposite bank, spreading it.

As the fire is stopped when the power is zero or negative, there are not too many possibilities to reach the other side of the barrier if it is wide enough. In this case, it will be required a very strong wind in the proper direction.

4- Proposed Method for Urban Fire Spread Simulation

The proposed algorithm to simulate the fire spread in an urban environment is an extension of the previously field definition. Thus, the information of the square cell is now enhanced with the urban data needed to simulate the fire spread.

The proposed method introduces the concept of Building, which can be defined as a 2.5 D representation of the real buildings using the approximation of the 2D projection of the building in the field and the number of floors.

In order to fill the field with the information about the buildings, the proposed algorithm splits all the buildings, following the grid, in vertical cells called Building Units, which are split in other small units called Floor Unit.

Different types of buildings

Following the approach of Iwami [8], the proposed method differentiates three types of *Floor Units*: The *Wooden Unit*, *Secure Building Unit* (normal building), *Shanty Unit* (see **Fig 6**).

The main difference between the buildings types are their behaviour when they are exposed to fire. *Secure Building Unit* is the most secure type, being the most difficult type of building to get burned. By contrast, the *Shanty Unit* is the more dangerous one, too easy to get burnt, but with few quantity of combustible. The *Wooden Unit* is an intermediate structure, with the flash over effect, i.e., when a determined heat level is reached, the wooden structure itself is burnt very quickly, producing a blast which will increase the heat and the fire intensity.

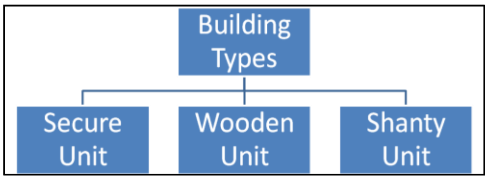

Fig 6: Different types of *Floor Unit*'s. The *Secure Unit* is equivalent to a modern flat, very difficult to get burnt and isolated enough to reduce the probability to spread to contiguous floors. The *Wooden Unit* is normally used in building with up to 4 floors, very dangerous if they catch fire as they can literally explode in a blast. The *Shanty Unit* are normally referred to one floor low quality structures. They normally catch fire very easily, but the fire tends to extinguish itself very quickly as the available combustible is burnt almost instantaneously.

The fire evolution in a *Floor Unit*

The evolution of the fire in a *Floor Unit* is characterized by four states. At the beginning of the fire, the ignition moment, the *Floor Unit* is in the state *FloorSafe*. When the fire begins to evolve in the unit, we are in the state *FireOnlyInterior*. This state is characterized by flames rising only from the openings, e.g., windows). Then, when the fire takes more power and begins to rise from the roof, we reach the state *FireWholeFloor*. At this moment, the fire is burning the whole *Floor Unit*. The last step is the third state characterized by a burned *Floor Unit*.

The evolution of the fire depends on the type of the *Floor Unit*. In fact, a *Secure Building Unit* will have a fire evolution from state *FloorSafe* to *FireOnlyInterior* and then, to the final state *FloorBurnt*. The state sequence for the *Wooden Unit* is *FloorSafe*, *FireOnlyInterior*, *FireWholeFloor* and finally, *FloorBurnt*. The sequence of the *Shanty Unit* is *FloorSafe*, *FireWholeFloor* and *FloorBurnt*.

Life cycle of a *Floor Unit*

In order to create a specific behaviour where different entities (*Floor Unit*'s) will be able to interact between them, a life cycle has been set.

When the fire starts, it only affects to the *Floor Unit* where it belongs. During this step, the state of the fire changes accordingly to the previously presented states.

When the fire is burning inside the *Floor Unit*, heat is being released continuously, what could lead to spread the fire to the upper *Floor Unit*, but also to the neighbour buildings and *Floor Unit*'s (depending on the distance between them).

Finally, the original *Floor Unit* will be totally burned, but the other ignited *Floor Unit*'s, contaminated by radiation will follow at their own the same life cycle.

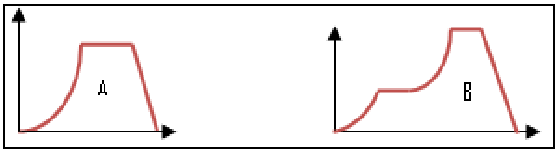

Fig 7: Heat evolution in different *Floor Unit*'s. The left side corresponds to the *Secure Building Unit*, where the evolution of the fire is slower. The right side is the heat released of a *Wooden Unit*, where an intermediate ignition burst can be found. In both cases, the heat released reaches to a maximum, and when the available combustible is consumed and the fire stops, the heat starts to cease, reaching to zero in a given amount of time.

Heat Released and state transitions

The evolution of the fire in the *Floor Unit*'s is based on the released heat during the ignition [18]. The evolution from one state to other depends on the measured changes of the heat.

In the proposed method, the heat evolves proportionally to the square of the time as long as the maximum heat is not reached.

The left side of the **Fig 7** represents the heat evolution for a *Secure Building Unit* and a *Shanty Unit*. The difference between both *Floor Unit*'s is the time scale for every part of the curve and the maximum released heat. The right side of the **Fig 7** represents the heat evolution for the *Wooden Unit*.

The heat of the *Floor Unit*'s does not exceed a maximum value, which depends on the surface of the *Floor Unit*'s. The proposed method takes into account several geometric characteristic to calculate the approximated evolution of the heat inside a *Floor Unit*: *i)* the amount of combustible is directly proportional to floor surface; *ii)* the size of the windows openings, which influences the strength of the combustion, since it is proportional to the summed air flux between the exterior and the interior of the floor; and *iii)* the height of the Floor Unit.

In the case of the *Wooden Unit,* it has another heat level of importance (*maxHeat/2*) to simulate the *flash over* effect.

The last phase of the fire evolution starts when 90% of the combustible is consumed, and therefore, the heat begins to decrease linearly

Urban fire spread between *Floor Unit*. Radiation effect

The algorithm simulates the radiation effect between different Floor Unit's, allowing the fire to spread to adjacent floors of a building or the ignition of other surrounding buildings.

The proposed algorithm implements three types of radiation effect, *i)* between floors of the same building, *ii)* between two different Buildings and *iii)* between a *BuildingUnit* Façade and an *OutsideUnit* Area

Inside a building, the fire goes generally up, trying to reach to the superior levels. But it is also possible to be spread towards lower floors. In order to allow this possibility, a small probability of spreading to lower levels (10%) is given to the fire.

The instant when a *Floor Unit* spreads to another floor is also randomly calculated, but always when the released heat is at its maximum, avoiding the spread when the fire is not strong enough.

The radiation effect is also considered when two buildings are not in direct contact. In our implementation of the radiation effect, given two *Floor Unit*s of different buildings, the fire can be spread between them if the linear distance is less than 12 meters and the release heat is higher than the 80% of its maximum value. Another factor to take into account is the existing wind (speed and direction), which will increase or decrease the spread probability.

An existing fire on a *Floor Unit* can spread to the outside field through the *Floor Unit*'s exterior openings, located in its façade. If the *Floor Unit* has no outside façade, the fire will never go out directly from that floor.

5- Fire Extinguisher Model

The proposed methods allow trying to extinguish an ongoing Urban/forest fire. Its main purpose is to allow the fire-fighters to interact with the fire evolution and see the consequences of their decisions on the field.

We have two approaches for this model, one for the outside fire simulation, and another for the building fire simulation.

Fire extinction for an outside fire

This model is conceptually the antagonist of the fire spreading algorithm, where the Fire Power amount is replaced by Fire Extinguisher Agent amount (e.g., water). When, a given quantity of water is thrown into a cell, the same quantity of Fire Power is decreased.

Thus, as long as there is water in a cell, its Fire Power will be equal to zero and the fire will be stopped (changing to the state *FireStopped*).

In order to simulate the evaporation (see **Fig 8**) of the water when it is thrown in an active fire, we use a similar approach of the radiation effect. A burning cell can reduce the quantity of water of the surrounding cells, depending on the distance between them and the Fire Power of the cell.

In consequence, if a cell is in the state *FireStopped* but it is still containing food (combustible), the fire in that area could be restarted thanks to an evaporation effect due to the surrounding burning areas.

The presented model is a very simplified version of how the fire agents fight the fire, but it is also generic. This model fits very well to simulate the actions of the ground fire-fighter, with a given quantity of water being thrown locally, or the fire fighting aircraft's, which throw a big amount of an especial extinguisher agent in a big area but for a short period of time.

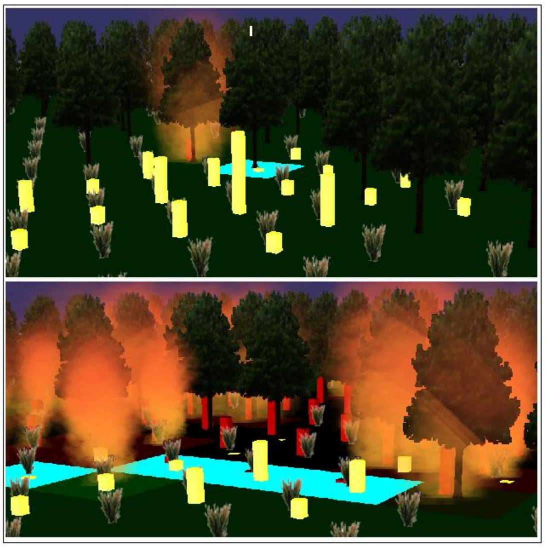

Fig 8. Evaporation effect. A reduction of water on the field is shown in the picture sequence (from the Up to the Down). The yellow sticks correspond to the water quantity.

Fire extinction for a Building Fire

The main concept behind the simulation of the extinguishment of the fire is that a fire extinguisher agent (or for clarity, water) can be thrown inside the building through the openings of a *Floor Unit*. Depending on the power of the fire extinguisher jet, it will be able to reach only the targeted *Floor Unit* or maybe, some internal *Floor Unit* of a Building.

When the water reaches a burning *Floor Unit*, its heat is reduced accordingly to the amount of received water. The fire will be stopped when the heat is equal to zero, but, in a similar way to the forest model, the *Floor Unit* will remain some residual heat that could eventually trigger some fire in the neighbours.

If the water is not enough to stop the fire, the water will be evaporated almost instantaneously, but the heat in the room will be lowered. If the water is not thrown constantly, the fire will continue its evolution after the evaporation of the water.

6- Implementation and Results

A C++ implementation of the algorithm has been done to test the algorithm behaviour and outputs.

The tests have been performed using a field of 2.25 hectares, choosing a tile side size equals to 3 meters, which is half the distance of possible spread of fire when there is no wind [2].

To simplify the tests, the wind is uniform and constant in the entire field and during all the simulation.

Forest Fire results

The simplest test is to run the fire simulator without any external factor which can influence the fire spread (Wind, Slope). The fire evolves producing a in a circular shape (see **Fig 9**, top-left).

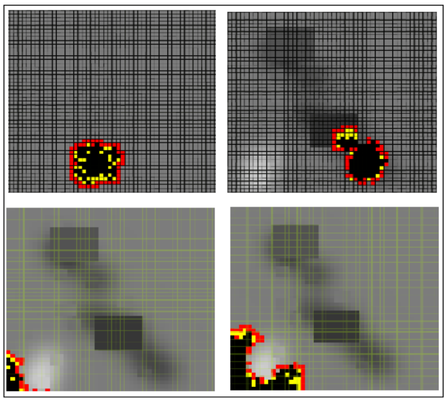

Fig 9. Some results of the Fire Spreading algorithm. Grey level shows the height map of the field: white, low height cells; black, high height cells. *Top-Left)* No Wind, No Slope. *Top-Right)* Slope, the fire goes up the hill. *Bottom)* Slope, the fire avoids enter the valley, surrounding it.

The slope effect has been tested under two scenarios, one to test how the fire can evolve towards higher positions (see **Fig 9**, top-right) and to test how the fire avoids go towards lower positions (see **Fig 9**, bottom left and right).

The Radiation Effect has been tested using the scenario shown in the Fig 10. The burning trees (pure green squares) facilitate the fire crossing river effect. Once the fire has crossed the river, it continues spreading in a normal way.

It is remarkable that the fire spreads to the other side of the river, but in a much lower speed. After some time, the fire becomes stronger and recovers its maximum intensity.

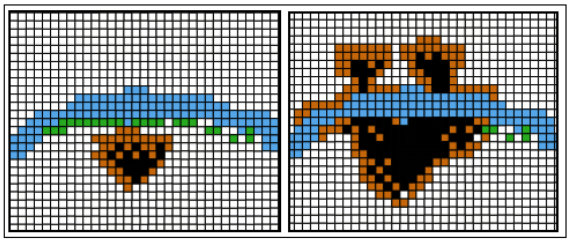

Fig 10. Forest fire crossing a river (blue cells). Green cell is a tree. Black/Orange cells correspond to the fire spread.

Urban Fire Results

Just before testing the urban fire simulator using an urban model, some preliminary test was performed to single building in order to check if the simulated fire evolution corresponds with the expected output.

This preliminary test (Fig 11) simulates the evolution of a fire in a *Wooden Floor Unit* of 9 m^3. The unit burned in 600 seconds with the heat evolution shown in Fig 11. Although the heat evolution curve (and especially the maximum heat) depends on the type, height and openings size of the *Floor Unit,* the simulation behaves in the expected way, including the burst when the released heat reached at its half value.

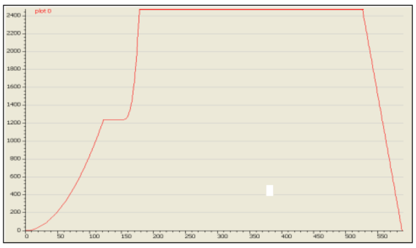

Fig 11: Heat (measured in MW) evolution of a wooden unit burning. The unit burnt in approximately 10 minutes. The flash over effect is around 150s.

The relationships between buildings are tested by using a sample field, with buildings and forest areas, which will be used to test the radiation effect.

If a fire is started in a building, if is not stopped, it will eventually spread. When the heat in a floor reaches a certain level, the other surrounding *Floor Unit*'s of the same level start to burn. In the Fig 12, we can see two examples of how the fire spread to other floors.

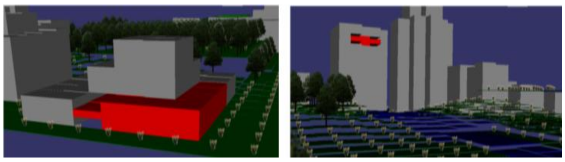

Fig 12: Both images represent how the fire spread in a building. The fire origin is located in the red box with more height, and it has been spread to the neighbour floors (small red box). The height of the red box represents visually the intensity of fire in that floor).

In a similar way, the radiation effect can occur between different buildings or a building and the near forest areas (see Fig 13). In these cases, some wind must be present in order to ease the probability of fire spread.

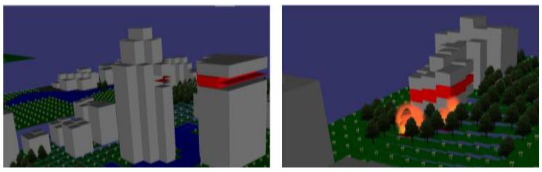

Fig 13: Radiation effect between near buildings (left) and the building and the surrounding vegetation (right). The fire origin is located in the big red box.

7- Conclusions and Future Work

In this paper, a pedagogical real-time fire simulation algorithm has been presented. Its main purpose is to be integrated into interactive Virtual Simulations where fire-fighter and managers can train their skills.

Although the forest fire spreading is a very complex phenomenon, we tried to simulate the most common characteristics of its behaviour by simplifying the model. Only two main physical variables have been used in the algorithm: direction of the wind, and the slope. The same approach has been followed for the urban fire simulation keeping only the heat parameter. The results have shown that the evolution in time of the fire fits the overall behaviour of the fire under different conditions.

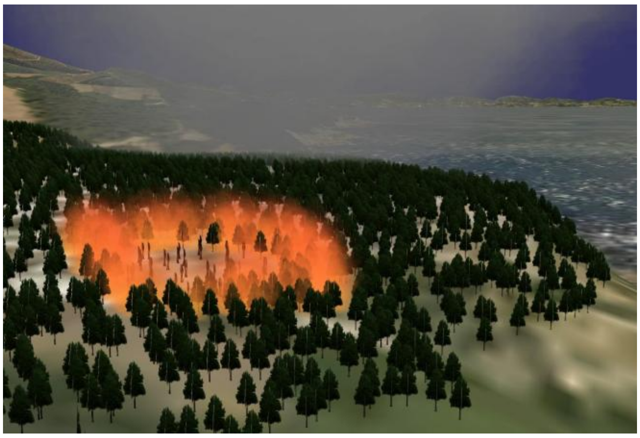

Fig 14: Capture of the prototype of the Forest fire simulator, implementing the proposed method for forest fire spread. The place shown in the picture is an existing zone in the north of Spain.

Some of the used approximations could be improved in the future. We can think about better utilisation of the wind information. The fires modify constantly the local wind, as it is changing the humidity and temperature of the atmosphere. The integration of a low complexity implementation of this phenomenon into the fire simulation would increase the realism of the output.

Also, a comprehensive comparison between different algorithms should be done. These tests will try to measure analytically the efficiency, the memory consumption and other related factors. Since the ultimate objective is to integrate the algorithms in a VR environment, a high quality rendering should be implemented. Some preliminary results are shown in **Fig 14**. After implementing the whole setup, user tests will have to be done in order to validate the algorithm output.

Our preliminary results implemented in a 3D real-time application are based on the OpenSceneGraph graphical API, where some testing 3D urban and forest scenarios were created and loaded. The utilisation of high quality rendering techniques to render the fire, the smoke and the terrain gives a realistic impression of the simulation.

The utilisation of real 3D city models can reduce the complexity of the model creation task and enhance dramatically the outcome of the results. The CityGML [3] standard can deal with all the currently required information by the algorithm, but also, it is also suitable for future needs. As the number of CityGML models grows, they could be used as the field of the urban and forest fire simulation method presented in this work.

Acknowledgements. This work was carried out in the context of project SIGEM, funded by the Spanish Industry Ministry through its *Avanza I+D Programme* and it was supported by *COST Action TU0801 "Semantic Enrichment of 3D City Models for Sustainable Urban Development"*.

References

1. Achtemeier, G. L., 2003. An Application of Stephen Wolfram's "New Kind of Science" to Fire Spread Modeling, USDA Forest Service, Athens, GA.
2. Breton, T., Duthen, Y., 2008. Les simulations de propagation de feu en milieu urbain.
3. CityGML Homepage: Exchange and Storage of Virtual 3D City Models: http://www.citygml.org/.
4. Dumond, Y., 2008. Forest fire growth modelling with geographical information fusion. 11th International Conference on Information Fusion, June 30 2008-July 3 2008 Page(s): 1 – 6.
5. Finney, M. A., 1998. FARSITE: Fire area simulator-model development and evaluation.
6. Forest Fires Website, 2010. Statistics on Forest Fires, http://library.thinkquest.org/07aug/01254/statistics.html . Last time visited: 15/05/2010.
7. Hamada, M., 1951, On Fire Spreading Velocity in Disasters (Tokyo: Sagami Shobo), in Japanese.
8. Iwami, T., Y. Ohmiya, Y. Hayashi, K. Kagiya, W. Takahashi and T. Naruse, 2004, Simulation of City Fire. Fire Science and Technology, 23(2), 132-140.
9. Lee, S., R. Davidson, N. Ohnishi and C. Scawthorn, 2008, Fire Following Earthquake–Reviewing the State-of-the-Art of Modeling. Earthquake Spectra, 24(4), 1-35.
10. Linn, R., Reisner, J., Colman, J. J., Winterkamp, J., 2002. Studying wildfire behavior using FIRETEC.
11. Ohgai, A,. Gohnai, Y., Ikaruga, S., Murakami, M., Watanabe, K., 2005. Cellular Automata Modeling For Fire Spreading As a Tool to Aid Community-Based Planning for Disaster Mitigation. In Book, Recent Advances in Design and Decision Support Systems in Architecture and Urban Planning, 193-209.
12. Richards, GD., R.W. Bryce, 1995. A computer algorithm for simulating the spread of wildland fire perimeters for heterogeneous fuel and meteorological conditions. Int. J. Wildl. Fire. 5(2):73-80.
13. Rocky Mountain website, 2010. Aircraft Rescue and Fire Fighting Basic Training, http://www.rmestc.com/. Last time visited: 15/01/2010.
14. Rothermel, R.C, 1972. A Mathematical model for Predicting Fire Spread in Wildland Fuels, Research Paper. U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range.
15. Scawthorn, C., J. Eidinger and A. Schiff, 2005, Fire Following Earthquake, Technical Council on Lifeline Earthquake Engineering Monograph No. 26 (Reston, VA: American Society of Civil Engineers).
16. Schmuck, G., 2006. Forest Fires in Europe, Report No7 / 2006, European Commission, Joint Research Centre Institute for Environment and Sustainability.
17. Tanaka, T., and Himoto K., 2006, Physics-based Model of Urban Fire Spread and Mitigation of Post-earthquake Fire Risk in Historic Cities, Disaster Prevention Research Institute, Kyoto University, 49(B), 1-5.
18. Vitenys, B., and Richard, D. P., 1991. Heat Release Rate: The single most important variable in Fire Hazard.
19. Weise, D. R., Biging, G. S., 1996. Effects of Wind velocity and slope on flame properties.