AGENT-BASED MODELING AND SIMULATION OF WILDLAND FIRE SUPPRESSION

Xiaolin Hu
Yi Sun

Department of Computer Science
Georgia State University
Atlanta, GA 30303 U.S.A.

ABSTRACT

Simulation of wildland fire suppression is useful to evaluate deployment plans of firefighting resources and to experiment different fire suppression strategies and tactics. Previous work of fire suppression simulation uses analytical models based on a continuous space. This paper presents a design of fire suppression simulation using a discrete event agent model based on a discrete cellular space. We present a framework of wildland fire suppression simulation and describe how firefighting agents in direct attack, parallel attack, and indirect attack are modeled. Experiment results are provided to demonstrate the agent models and to compare them in different fire suppression scenarios.

1. INTRODUCTION

Wildland fires pose great challenges in the management of forests. This ecological problem raises significant concerns and calls for research for understanding the underlying causes, the effect of land management on fire ecology, wildfire risk, the dynamics of vegetation fuel, and how to reduce the likelihood of large-scale fires. In the event of a wildfire, decision-making of fire suppression, including firefighting resource deployment, plan generation, and dynamical tactics adaptation, is needed. Such decision-making is difficult due to the short decision-making time, dynamical fire behavior, and the limited firefighting resources that are available. Simulation of fire spread and suppression provides a valuable tool for real-time decision-making of fire suppression and containment.

The major wildland fire simulation environments, such as FARSITE (Finney 1998) and BehavePlus (Andrews et. al. 2005), are based on discrete time approaches. In these environments, fire spread is simulated as a time-based propagation process in a continuous space, and fire suppression is simulated by solving an analytical model that leads to the desired fireline to be constructed. Recently, discrete event models are also developed for simulating

wildland fire. A major paradigm of discrete event simulation of wildland fire (see examples (Vasconcelos 1993; Muzy et. al. 2005; Ntaimo et. al. 2004)) is based on the cellular space approach using Discrete Event System Specification (DEVS) (Zeigler et. al. 2000). This approach models a forest as a cellular space, where each cell is a discrete event model and can change its state and influence its neighboring cells. Fire spread is simulated as a propagation process from cell to cell (e.g., from cell center to cell center) when burning cells ignite their unburned neighbors. Significant advances have been made for using discrete event models to simulate fire spread. However only limited research is conducted for fire suppression simulation based on discrete event cellular space models.

This paper presents a discrete event agent-based approach for modeling and simulating wildland fire suppression. Firefighting resources such as engines and bulldozers are modeled as agents that build a fireline in a discrete cellular space to suppress or contain a fire. Consistent with the fire suppression simulations in previous works such as FARSITE and (Fried and Fried 1996), three firefighting strategies are modeled here: direct attack (head and tail), parallel attack, and indirect attack. The fire spread simulation model that supports this work is a discrete event cellular space model that was developed in (Ntaimo et. al. 2004). Simulating fire suppression in a discrete cellular space means the effects of agents' firefighting will also be discrete, i.e., agents' fireline construction will proceed discretely both in space and in time. Algorithms that govern how an agent constructs a fireline in direct attack, parallel attack, and indirect attack are developed respectively.

This work is part of a large project that develops a dynamical data driven framework integrating simulation and stochastic optimization for wildland fire management. The framework uses simulation to predict the fire spread behavior, stochastic optimization to compute optimal plans for firefighting resource deployment, and fire suppression simulation to evaluate deployment plans as well as firefighting strategies and tactics. Integrating these components together gives rise to a real-time decision support

system that can assist fire managers to effectively bring under control a potentially catastrophic fire, and allow for a timely warning and a well-coordinated evacuation plan. Within this framework, this paper focuses on the development of fire suppression simulation that ultimately will be integrated with the optimization component for firefighting resource deployment. The modeling and simulation environment used by this work is the DEVSJAVA environment (Zeigler and Sarjoughian 2003).

The remainder of the paper is organized as follows. Section 2 describes the related work of fire suppression simulation. Section 3 presents the overall framework within which this work is developed. Section 4 presents the modeling of firefighting agents in direct attack, parallel attack, and indirect attack. Section 5 gives some experiment results to demonstrate the agent models and section 6 concludes this work.

2. RELATED WORK

Simulation of wildland fire suppression is studied in several research areas. First, research in fire spread simulation generally studied fire suppression simulation. For example, FARSITE has considered ground attack simulation where three types of attack are supported: direct attack, parallel attack and indirect attack. The effect of direct attack on an active fire front is simulated using the known fire perimeter positions at two successive time steps and an attack crew building line is defined based on the quadrilateral formed by perimeter vertices in the two time steps (FARSITE Technical Reference 1998). The work of (Fried and Fried 1996) developed a mathematical model for direct attack and parallel attack, where parallel attack is modeled in the same way as direct attack for a "super" free burning fire boundary (*fbfb*) that has a fixed safe distance to the actual *fbfb*. This model is also used by BehavePlus (Andrews et. al. 2005) for fire containment simulation. In all these simulations, fire spread is a propagation process with fire front represented by a smooth curve, typically in an elliptical shape, that evolves over time in a self-similar fashion. In every time step, fire suppression is simulated by solving an analytical model that leads to the desired fireline that needs to be constructed. In discrete event simulations of fire spreading, (Ntaimo et al. 2004; Ntaimo and Zeigler 2005) developed a cellular DEVS model of forest fire spread that includes fire suppression control measures. (Ameghino et al. 2001) also incorporates fire suppression into a cellular Cell-DEVS model. These works focus on the modeling of a forest cell itself by adding new states (such as *burning_wet* and *unburned_wet* as in (Ntaimo et al. 2004)) to account for the effect of fire suppression. They do not model how a fireline is constructed nor do they consider realistic fire suppression tactics. Our previous work (Hu et. al. 2005) developed a hybrid agent and cellular space modeling approach for supporting firefighting agents

to interact with forest cells. It builds a ground for this work but did not model agents with realistic tactics.

Research in artificial agents also studied fire suppression as an application area. These works focus on the autonomous decision-making, learning, and group cooperation among a team of agents in the context of firefighting. For example, Phoenix (Cohen et al. 1989; Greenberg and Westbrook 1990) is an agent simulation environment in the domain of forest fire fighting. Agents include watchtowers, fuel trucks, helicopters, bulldozers and a fireboss that coordinates the efforts of all. The goals of the Phoenix project were to challenge AI agents with difficult problems to solve under time pressure, with dynamical environmental conditions, communication and cooperation among agents, and tradeoffs between planning time and plan quality. The work of (Hogg and Jennings 1997) studied how socially rational agents can be used to achieve "best" individual and also overall system's objectives in the context of firefighting scenarios. Although based on the application of fire suppression, these works emphasize on the AI aspect thus are different from the main concerns of this paper.

Fire suppression as part of fire management is also researched in forest science and operations research. A major effort of this research is to choose the (optimal) resource deployment for initial attack (see, e.g., Fried et al. 2006; Dimopoulou and Giannikos 2001). Software tools are also developed. For example, The California Fire Economics Simulator version 2 (CFES2) is a sophisticated stochastic simulation model designed to facilitate quantitative analysis of the potential effects of changes in many key components of most wildland fire systems, e.g. availability and stationing of resources, dispatch rules, criteria for setting fire dispatch level, staff schedules, and deployment and line-building tactics. IIAA (Interagency Initial Attack Assessment) is a tool used to develop budget requests as part of the National Fire Management Analysis System (NFMAS) process, which is an analytical model used to identify the most economically efficient level of the fire management organization.

3. SIMULATION AND OPTIMIZATION FOR WILDLAND FIRE SUPPRESSION

Simulation of fire suppression relies on the existence of a fire spread model. This interaction between the production of suppression fireline and a fire's capacity to spread is a feature that must be considered in simulating wildfire containment (Fried and Fried 1996). Meanwhile, a simulation of fire suppression must take the inputs of what types of firefighting resources will be deployed and when and where to be deployed. Such deployment plans of firefighting resources need to be optimized to ensure that a fire will be contained with least operating cost and result in least damage to the forest area. This relationship among

fire spread simulation, fire suppression simulation, and firefighting resource deployment optimization motivates us to develop an integrated framework for supporting wildland fire management.
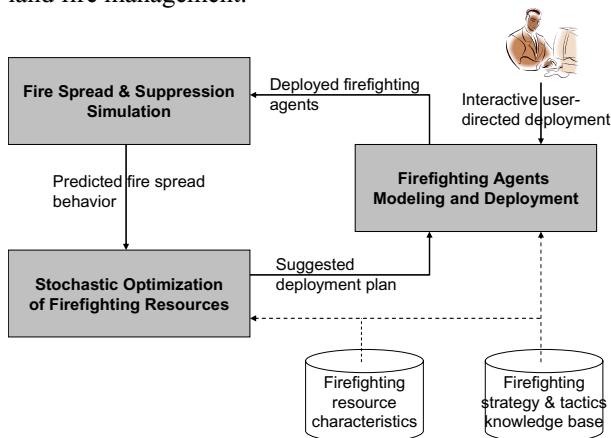

Figure 1: Optimization, deployment, and simulation for wildland fire suppression

Figure 1 shows the three functional components of this framework, among them fire spread simulation and fire suppression simulation belongs to the same functional component because they are closely related, i.e., a fire suppression simulation cannot exist without fire spread simulation. Note that the reverse is not true as a fire spread simulation can run by itself. The other two components are stochastic optimization of firefighting resources, and firefighting agent modeling and deployment. The optimization component takes inputs, including time-indexed burned areas and fire front perimeters, from multiple runs of fire spread simulations (they represent the predicted fire spread scenarios). It also takes information of firefighting resource characteristics such as operating cost and time to deploy and then computes the optimal deployment plan for containing the fire. Based on the deployment plan suggested by the optimization component, the firefighting agent modeling and deployment component is responsible to (dynamically) create agents and deploy them. It also supports interactive user directed deployment. This is especially useful for experimenting and comparing different fire suppression strategies and tactics. The component of firefighting agent modeling and deployment utilizes two knowledge databases: a firefighting resource characteristics database including information such as the resource types and their production rates; and a firefighting strategy and tactics knowledge base including information such as direct attack, parallel attack, and indirect attack and different configurations of them. The three components, when working together, allow a fire manager to predict how a fire will spread, to generate firefighting resource deployment plans based on the predicted fire, and to dynamically create and deploy firefighting agents according to the plans and then run fire suppression simulations to evaluate them.

The simulation models of both fire spread and fire suppression are based on Discrete Event System Specification (DEVS) (Zeigler et. al. 2000). Specifically, the fire spread model is extended from the model developed in (Ntaimo et. al. 2004). This is a two dimension cellular space model where each cell represents a sub-area of the forest. Fire spread is simulated as a propagation process where burning cells ignite their unburned neighbors. A cell is coupled to its eight neighboring cells (the Moore neighborhood). This allows cells to send messages and thus affect each other. A forest cell has several discrete states such as *unburned, burning,* and *burned* and transits from one to another based on external or internal events. The fire spreading speed of a burning cell is calculated using Rothermel's model (Rothermel 1972) and then decomposed according to eight directions corresponding to the eight neighboring cells. More information of this model can be found in (Ntaimo et. al. 2004). Different from the cellular space model used in fire spread simulation, the simulation of fire suppression is modeled by agent models. The design principles of how an agent model works with a cellular space model is described in (Hu et. al. 2005). To support the interactions between an agent and its environment (the cellular space), couplings are added between the agent and the corresponding cell where the agent locates. These couplings are dynamically added/removed during the simulation when the agent changes its location from one cell to another. In the current implementation, each agent is an atomic model and is responsible to add/remove its couplings to forest cells when it moves from one cell to another. Finally, the model of stochastic optimization is an integer programming model. Interested readers can refer to (Ntaimo et. al. 2006; Ntaimo et. al. 2007) for more information of this model.

4. MODELING AND SIMULATING FIREFIGHTING AGENTS

Fire suppression is a process for firefighting agents to construct a fireline to suppress (or contain) a burning fire. Similar to FARSITE and (Fried and Fried 1996), we consider three types of fire suppression strategies: direct attack, parallel (indirect) attack, and indirect attack. Direct attack refers to the strategy in which fireline is constructed on the flaming fire front, the region where combustible fuels are igniting. Two specific direct attack tactics are direct head attack and direct tail attack where the attacks start from the head and tail of the fire respectively. Parallel (indirect) attack refers to the strategy in which fireline is constructed parallel to, but at a safe distance (offset) away from, the fire perimeter. This is usually applied when the fire is intensive and fast spreading thus having the potential for causing serious injuries or fatalities to the firefighters. Indirect attack refers to the strategy in which fireline is

constructed according to a predetermined route. More discussions about these different strategies can be found in (Fried and Fried 1996).

In modeling the firefighting agents, we make the following two assumptions that are common for all three attack strategies. These assumptions are made because of the discrete nature (discrete space and discrete event) of the fire spread simulation model. 1). In a cellular space model where the space is divided into discrete cells, a firefighting agent can only proceed, i.e., construct the fireline, from center to center between two cells that neighbor each other. Each cell has eight neighboring cells. 2). The effect of fire suppression is a Boolean effect (true of false). When an agent reaches the center of a cell, the cell is considered suppressed. Otherwise, the cell is treated as a normal cell. In other words, we do not model intermediate fire suppressing stages such as half suppressed or ninety percent suppressed.

Simulating fire suppression mainly deals with how to simulate the dynamical process of fireline construction that is carried out by firefighting agents. Each agent, after completing a fireline segment, needs to decide how to proceed for constructing the next fireline segment. This decision of how to proceed next is either based on a pre-defined plan (such as in indirect attack) or on the dynamical behavior of fire spread (such as in direction attack and parallel attack). In this work, since an agent can only move from a cell to a neighboring cell in a discrete fashion, the modeling of firefighting agent concerns how an agent chooses an appropriate neighboring cell to construct the next fireline segment. Below we present the agent models for direct attack, parallel attack, and indirect attack respectively.

4.1. Firefighting Agent in Direct Attack

In direct attack, agents build a fireline along the fire front where cells are burning. Since it takes time for a fireline segment to be constructed, a burning cell may ignite its neighboring cells even if it is being attacked by an agent. Thus in choosing which neighboring cell to construct the next fireline segment, an agent needs to ensure that the to-be-constructed fireline segment will be completed before any neighboring cells "outside" this fireline segment are ignited. In our design, this is achieved by a "predict-and-scan" schema that is illustrated in Figure 2 and described below.

The basic idea of the "predict-and-scan" schema is that an agent needs to predict how far the current fire can spread based on how fast the agent itself can complete the next fireline segment, and then uses the predicted fire front as a guidance to decide the direction for constructing the next fireline segment. This is illustrated in Figure 2(a), where O represents the agent's current location, line segment $C1$-O represents the already completed fireline, line segment O-$C2$ represents the current fire front, and line segment O-$C3$ represents the predicted fire front. The "look-ahead" time window that is used for fire spreading prediction is based on how fast the agent can finish the next fireline segment. This is calculated by dividing the segment length by the agent's production rate. After predicting the fire spread, the agent then chooses the predicted fire front (*e.g.*, O-$C3$ in Figure 2(a)) as the direction for constructing its next fireline segment. Because this is the predicted fire front, it ensures that during the time when the agent builds the chosen fireline segment, no area "outside" the fireline perimeter (*e.g.*, the top left area in Figure 2(a)) will be ignited. To detect the predicted fire front, the agent applies a "scan" process. Specifically, starting from the completed fireline segment and taking the circular direction that goes away from the burning side of that segment, the agent scans around until it meets the first burning area. This is illustrated by the dashed circular arrow in Figure 2(a). Figure 2(b) illustrates the "predict-and-scan" schema in a cellular space. In this figure, the completed fireline segment comes from the southwest neighbor of the current cell (both the southwest cell and the current cell are thus suppressed). The east neighboring cell represents the current fire front and the northeast neighboring cell represents the predicted fire front. By "scanning" its neighboring cells, the agent chooses the northeast neighboring cell as the destination cell for constructing the next fireline segment.
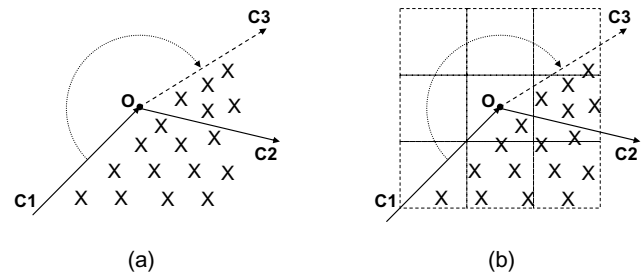

(a)   (b)

Figure 2: The "predict-and-scan" schema to choose the next suppression cell

Based on the design idea described above, a two stage "predict-and-scan" schema is developed for the agents in direct attack. The two stages are needed because the lengths for a diagonal fireline segment and that for a non-diagonal fireline segment are different, i.e., the length to a diagonal cell is $\sqrt{2}$ as long as the length to a non-diagonal neighboring cell. This difference results in different time that is needed for an agent to construct the fireline segments. Because of this, two different "look-ahead" time windows, thus two stages of prediction, should be used to predict fire spread for the agent to decide the next fireline segment. The algorithm that implements the two stage "predict-and-scan" schema is described below (informally). Specifically, after an agent reaches the center of a cell, the agent goes through the following steps to choose a

neighboring cell for constructing the next fireline segment. In the following, *cellSize* is the size of a cell, and *production_rate* is the agent's production speed (meters per second) for building a fireline.

1. Calculate the time for building a fireline to a non-diagonal neighboring cell: *T_lookahead = cellSize/production_rate*.
2. Use *T_lookahead* as the look-ahead time window to predict the fire spreading situation of the neighboring cells. Mark them as *burning, burned, unburn, or suppressed*.
3. Apply the "scan" process described above to scan its neighboring cells until meet the first burning cell. Two situations may happen:
   i. If this cell is a non-diagonal cell, the cell is chosen as the destination cell.
   ii. If this cell is a diagonal cell, aborts the scan and starts the second "prediction-and-scan" stage (Step 4).
4. If a destination cell is chosen, skip the rest of the steps. Otherwise, calculate the time for building a fireline to a diagonal neighboring cell: $T\_lookahead = \sqrt{2} * cellSize/production\_rate$.
5. Use *T_lookahead* as the look-ahead time window to predict the fire spreading situation of the neighboring cells. Mark them as *burning, burned, unburn, or suppressed*.
6. Apply the "scan" process described above to scan its neighboring cells until meet the first burning cell. Choose this cell (diagonal or non-diagonal) as the destination cell.

The algorithm that is used to predict fire spread is the fire spreading simulation itself. Specifically, an agent creates a new cell space model that duplicates the local area of the "original" cellular space. The states of the cells in this new cell space are initialized to the current states of the corresponding cells in the original cell space. This new cell space model is then simulated until the given look-ahead time window is reached. Creating a cell space model that represents only the local sub-area, instead of the entire cell space, for simulation is due to performance considerations. Also note that in the two stage "predict-and-scan" schema, for simplicity we use the same "look-ahead" predicting time window for all non-diagonal (or diagonal) cells. However in a space with non-uniformed fuel models, the fireline construction time to different cells will be different and is dependent on the fuel model of the cell. To have more precise predictions in such cases, the two-stage "predict-and-scan" schema can be extended to a multi-stage "predict-and-scan" schema (not presented in this paper) to account for the non-uniformed fuel models.

4.2. Firefighting Agent in Parallel Attack

In parallel attack, agents build a fireline parallel to the fire front perimeter and maintain a fixed safe distance to it. The same algorithm, i.e., the "predict-and-scan" schema, that is used in direct attack is employed to compute the fireline path in parallel attack. The major difference is that an agent in direct attack finds the first cell that is burning as the destination cell when it "scans" its neighboring cells. But in parallel attack, the agent finds the first cell whose distance to the closest fire front equals to (or is close to) the pre-defined safe distance as the destination cell. Specifically, after an agent reaches the center of a cell, the agent goes through the following steps to choose a neighboring cell for building the next fireline segment.

1. Calculate the time for producing a fireline to a diagonal neighboring cell: $T\_lookahead = \sqrt{2} * cellSize/production\ rate$.
2. Use *T_lookahead* as the look-ahead time window to predict the fire spreading situation of the agent's *distance bounded neighboring cells*. Here the *distance bounded neighboring cells* are those cells whose cell-distances (distance based on difference of cell IDs) are less than or equal to $\left\lceil \dfrac{D\_safe}{cellSize} \right\rceil + 1$, where *D_safe* is the desired safe distance. Based on the prediction, mark the cells as *burning, burned, unburn, or suppressed*.
3. Apply the same "scan" process as described in direct attack to scan the agent's direct neighboring cells. For each cell, calculate the distance *D_fire* from this cell to the closest fire front (a cell that is burning). Choose the first cell (diagonal or non-diagonal) whose *D_fire* equals to (or is close to) *D_safe* as the destination cell to construct the next fireline segment.

Two things are worthy to mention here. First, the firefighting agents in parallel attack use a single stage, instead of two-stage, "predict-and-scan" schema to calculate the fireline path. This is a simplified treatment and is used in our current implementation. Because a longer distance (i.e., the distance to a diagonal cell) is used to calculate the look-ahead time window for prediction, this treatment guarantees no burning cell will be left out of the fireline perimeter. Second, an agent in parallel attack needs to predict the fire spreading situation of its *distance bounded neighboring cells*. The value of the "*distance bound*" is dependent on the desired safe distance in parallel attack. This is different from that in direct attack, which only predicts fire spreading of the agent's direct neighboring cells.

4.3. Firefighting Agent in Indirect Attack

In indirect attack, agents build a fireline according to a predetermined route. Such a predetermined route can be explicitly specified by a fire manager, or be computed from some algorithm that considers factors such as current and predicted fire spreading behavior and available firefighting resources. How to generate an effective predetermined route for indirect attack is out of the scope of this paper. Below we assume such a route has been generated somewhere else and describe how a firefighting agent chooses a neighboring cell in indirect attack.

Because the existence of a predetermined route, the design of firefighting agents in indirect attack is simpler than that in direct attack and in parallel attack. Specifically, each agent has a copy of the predetermined route. After an agent reaches the center of a cell, the agent uses its current location to check the route, and then chooses a neighboring cell that is consistent to what the route suggests. It is assumed that an agent will always follow the predetermined route in an indirect attack, independent of how the real fire spreads.

4.4. Multiple Agents

It is common for multiple firefighting resources to work together to suppress a fire. In our design, dependent on weather the multiple agents work on the same fireline segment or not, we handle the multiple agents in two different ways. First, if multiple agents work on the same fireline segment, they are treated as a single agent with its fireline production rate being the aggregated production rate of all the agents. For example in a direct tail attack, if two agents stay together and build a fireline in the same direction, the effect of fire suppression by these two agents is simulated by using a single agent whose fireline production rate equals to the sum of those of the two agents. On the other hand, if multiple agents work on different fireline segments (either on different locations or following different directions), they are treated as independent agents without influencing each other. For example, if one agent works in the clockwise direction and the other in the anticlockwise direction, or if one works in the head of the fire and the other in the tail, then the two agents work independently. In general, multiple agents are divided into several groups, each of which works on its own fireline segment. In this case, the agents that work on the same fireline segment are simulated as a single agent, and different groups (simulated by single agents) work concurrently.

5. EXPERIMENT RESULTS

This section presents three experiments that were conducted to demonstrate the firefighting agents designed in this paper. These experiments demonstrate and compare different fire suppression strategies and tactics. In all the experiments, fire spread simulation are based on the model developed in (Ntaimo et. al. 2004). The dimension of the cellular space is 50 x 50, with each cell's size being 15 meter x 15 meter. For simplicity, a uniformed fuel model is used for all the cells. Fire is firstly ignited at cell (25, 8) at time 0. A constant wind is applied with wind speed being 1.1 (1.117600004 is corresponding to 2.5kph) and direction from south to north. In the following figures, the color of green means the cell is *unburned*, red means *burning*, black means *burned out*, gray means *suppressed when burning*, and blue means *suppressed when unburned*.

(a) Before attack   (b) Direct tail attack
(t = 2712.26s)   (t = 6141.02s)

(c) Direct head attack   (d) Direct head & tail attack
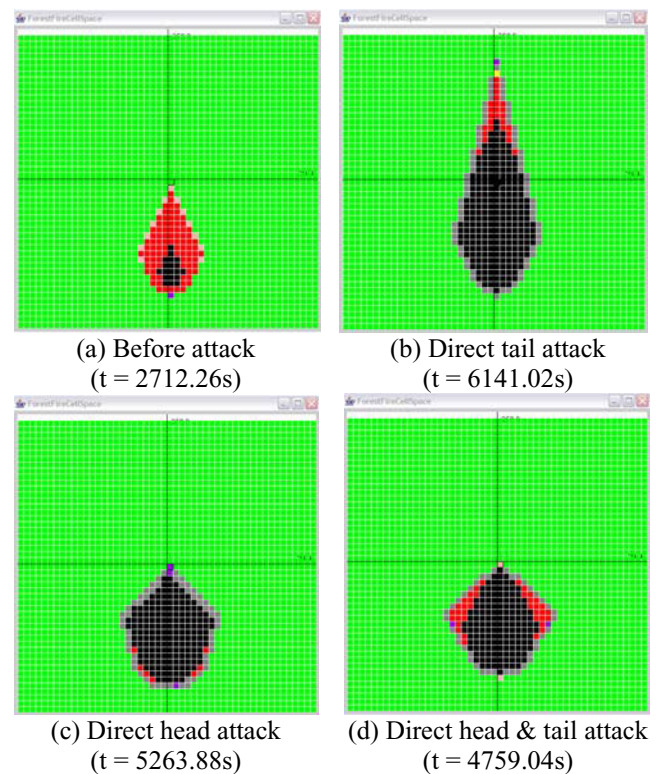(t = 5263.88s)   (t = 4759.04s)
Figure 3: Experiment 1- three tactics of direct attack

The first experiment (Figure 3) compares direct head attack, direct tail attack, and combined direct head and tail attack. In this experiment, we maintain a fixed overall production rate of firefighting agents, and set the same start time of fire suppression (see Figure 3(a)) for all three tactics. In both direct tail attack and head attack (see Figure 3(b) and 3(c) respectively), two agents (or two groups of agents, each of which is simulated by a single agent) start the suppression from the same location with one constructing the fireline clockwise and the other anticlockwise. Each agent has fireline production rate of 0.25 meter/second. The initial position of suppression is cell (25,6) in direct tail attack, and cell (25, 25) in direct head attack. The case of combined tail and head attack (see Figure 3(d)) combines the configurations of direct head attack and direct tail attack mentioned above. It employs four agents,

each of which has production rate of 0.125 meter/second. Figure 3 shows the results of these different types of fire suppressions.

Table 1: Three direct attack tactics in experiment 1

0.5 overall fireline production rate for all agents	Suppression time (s)	Fireline length (m)	Burning/burned area (m^2)
Before attack		681.78 (fire front length)	27225
Direct tail attack (2 agents, each rate is 0.25)	3428.76	1720.8	63000
Direct head attack (2 agents, each rate is 0.25)	2551.62	1255.65	40725
Combined tail & head attack (4 agents, each agent is 0.125)	2046.78	1023.3	41625

Table 1 shows the suppression time (the time duration for agents to suppress the fire), total fireline length that is constructed by agents, and burned/burning area after the fire is suppressed (the area that is surrounded by the fireline) for the three direct attack tactics in experiment 1. It can be seen that compared to direct head attack, direct tail attack takes longer time and ends up with more fireline length and more burned area. The two other cases, direct head attack and combined head & tail attack, result in roughly the same fireline length and burned area. But the tactic of combined head & tail attack takes less time to suppress the fire.

(a) Before attack
(t = 2712.26s)

(b) One agent
(t=9755.89s, not completed)

(c) Two agents
(t = 6141.02s)

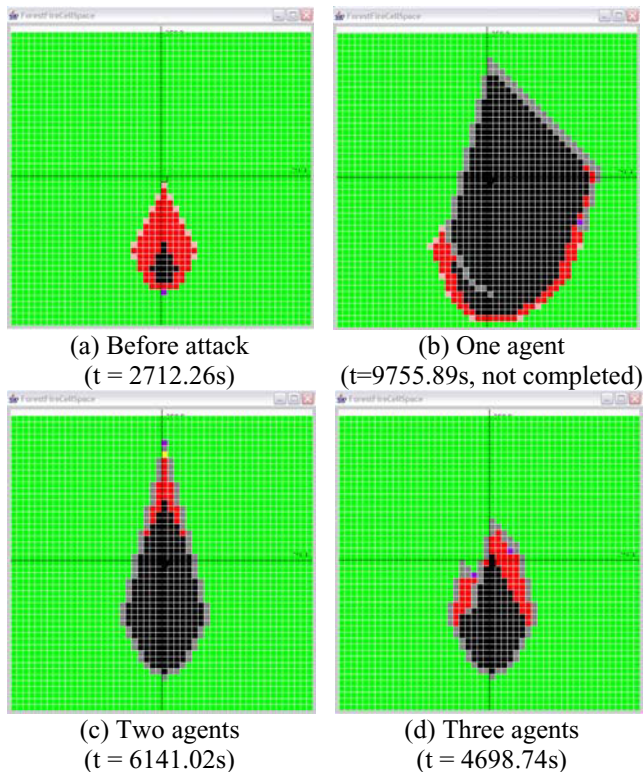(d) Three agents
(t = 4698.74s)

Figure 4: Experiment 2 - different number of agents

The second experiment (Figure 4) varies the number of deployed firefighting agents (1 agent, 2 agents, and 3 agents) and compares their fire suppression results. In this

experiment, each agent has the same fireline production rate of 0.25 meter/second. The start time of fire suppression for all three cases is the same as that in experiment 1 (see Figure 4(a), which is the same as Figure 3(a)). Figure 4(b) shows the result when one agent is deployed for direct tail attack (initial position (25, 6)) in a clockwise direction. Figure 4(c) shows the result when two agents are deployed for direct tail attack with the same initial position (25, 6). Note that this is the same configuration as that in Figure 3(b). Figure 4(d) shows the result when three agents are deployed. Two of them start from (25, 6), and the third starts from (21, 17) and builds fireline in the clockwise direction.

Table 2: Different number of agents in experiment 2

0.25 production rate per agent	Suppression time (s)	Fireline length (m)	Burning/burned area (m^2)
before attack		681.78 (fire front length)	27225
1 agent	N/A	1761.15 (not completed)	N/A
2 agents	3428.76	1720.8	63000
3 agents	1986.48	1490.1	40950

Table 2 shows the suppression time, fireline length, and burned/burning area for the three different cases where different number of agents are deployed. It can be seen that one agent was not enough to suppress the fire as the fire escaped from the fireline. Compared to the case of using two agents, the case of three agents results in less suppression time, less fireline length, and less burned area.
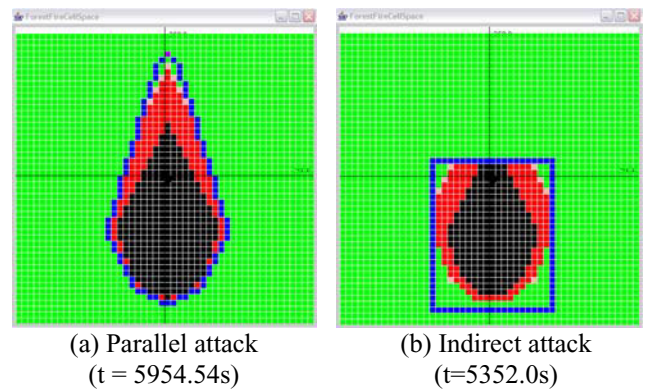

(a) Parallel attack
(t = 5954.54s)

(b) Indirect attack
(t=5352.0s)

Figure 5: Experiment 3 - parallel attack and indirect attack

Table 3: Parallel attack and indirect attack in experiment 3

2 agents, each with 0.25 production rate	Suppression time (s)	Fireline length (m)	Burning/burned area (m^2)
before attack		681.78 (fire front length)	27225
parallel attack (distance = 2 cell size)	3242.28	1621.2	104625
Indirect attack (rectangle route)	2639.74	1350.0	102600

The third experiment (Figure 5) demonstrates parallel attack and indirect attack. In both cases, two agents were used, with each one having production rate of 0.25. The start time of suppression is the same as that in the first two experiments. In parallel attack (Figure 5(a)), the two agents starts from the tail of the fire and maintain a safe distance of two cell size (30 meters). In indirect attack, the two

agents start from the head of the fire and follow a pre-defined rectangle route as shown in Figure 5(b). Table 3 shows the suppression time, fireline length, and burned/burning area for these two cases. It can be seen that in both cases, the agents were able to suppress the fire. However, compared to direct attack (see Table 1), parallel attack and indirect attack result in much more burned areas.

6. CONCLUSIONS

This paper presents a design of fire suppression simulation using a discrete event agent model based on a discrete cellular space. Firefighting agents in three different fire suppression strategies, i.e., direct attack, parallel attack, and indirect attack, are developed. Preliminary results are provided to demonstrate the agent models and to compare them in different fire suppression scenarios. Initial analysis of these results shows that they are consistent with those in previous works. Future work includes integrating the developed fire suppression simulation with the stochastic optimization component of firefighting resource deployment, and to apply and evaluate the agent models in more complex fire suppression scenarios.

ACKNOWLEDGMENTS

REFERENCES

Ameghino, J., A. Troccoli, G. Wainer, 2001. Models of complex physical systems using Cell-DEVS. *Proceedings of Annual Simulation Symposium*, Seattle, WA. U.S.A, p. 266-273.

Andrews, P.L., C.D. Bevins, R.C. Seli. 2005. BehavePlus fire modeling system, version 3.0: User's Guide Gen. Tech. Rep. RMRS-GTR-106WWW Revised. Ogden, UT: Department of Agriculture, Forest Service, Rocky Mountain Research Station. 132p.

Cohen, P. R., M. L. Greenberg, D. M. Hart, and A. E. Howe. 1989. Trial by fire: Understanding the design requirements for agents in complex environments. *AI Magazine* 10(3): 32 - 48.

Dimopoulou, M., and I. Giannikos. 2001. Spatial optimization of resources deployment for forest-fire management, *Intl. Trans. in Op. Res*. 8 (2001) 523-534

FARSITE Technical Reference. 1998. (from FARSITE 4.0 help), available at http://www-laep.ced.berkeley.edu/~itr/literature/farsite/index.html

Finney, Mark A. 1998. FARSITE: Fire area simulator–Model development and evaluation. Research Paper RMRS-RP-4. Ogden, UT: U.S.Department of Agriculture, Forest Service, Rocky Mountain Research Station.

Fried, J.S., and B.D. Fried. 1996. Simulating wildfire containment with realistic tactics. *For. Sci.* 42(3):267–281.

Fried, J.S., Gilless, J.K., and J. Spero. 2006. Analysing initial attack on wildland fires using stochastic simulation. *International Journal of Wildland Fire* 15:137-146

Greenberg, M. L., and D. L. Westbrook. 1990. *The Phoenix testbed*. Technical Report COINS TR 90-19, Computer and Information Science, University of Massachusetts at Amherst.

Hogg, L. M. and Jennings, N. R. 1997. Socially Rational Agents. In *Proceedings of AAAI Fall Symposium on Social Intelligent Agents*, pp. 61-63

Hu, X., A. Muzy and L. Ntaimo. 2005. A Hybrid Agent-Cellular Space Modeling Approach for Fire Spread and Suppression Simulation, *Proceedings of 2005 Winter Simulation Conference*, December, 2005

Muzy, A., E. Innocent, A. Aiello, J-F. Santucci and G. Wainer. 2005. Specification of Discrete Event Models for Fire Spreading. *SIMULATION* 81 (2): 103-117.

Ntaimo, L., B. Khargharia, B. P. Zeigler and M. J. Vasconcelos. 2004. Forest fire spread and suppression in DEVS, *SIMULATION*, 80 (10): 479-500.

Ntaimo, L., and B.P. Zeigler. 2005. Integrating Fire Suppression into a DEVS Cellular Forest Fire Spread Model, *Proceedings of the Spring Computer Simulation Conference*, San Diego, CA, April 3-7, 2005.

Ntaimo, L., W-J. Lee and A. Jalora. 2006. A stochastic mixed-integer programming approach for wildfire containment. *Proceedings of IIE Annual Conference*, Orlando, FL, May 21-24.

Ntaimo, L., W.J. Lee and E. Zwierzykowski. 2007, A Stochastic Programming Approach to Resource Deployment for Wildfire Containment under Uncertainty, submitted

Rothermel, R. C. 1972. *A mathematical model for predicting fire spread in wildland fuels*. Research Paper INT-115. Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station. 40 p.

Vasconcelos, J. M. 1993. *Modeling Spatial Dynamic Ecological Processes with DEVS-Scheme and Geographical Information Systems*, Ph.D. Dissertation, Dept. of Renewable and Natural Resources, University of Arizona, Tucson, U.S.A.

Zeigler B. P., H. Praehofer, and T.G. Kim 2000, *Theory of modeling and simulation*, 2nd Edition, Academic Press.

Zeigler, B.P., H. S. Sarjoughian, 2003, Introduction to DEVS Modeling & Simulation with JAVA: Developing Component-based Simulation Models, available at http://www.acims.arizona.edu/PUBLICATIONS/publications.shtml

AUTHOR BIOGRAPHIES

XIAOLIN HU is an assistant professor in the Computer Science Department at Georgia State University. His research interests include modeling and simulation, agents, and simulation-based design.

YI SUN is a Ph.D. candidate in the Computer Science Department at Georgia State University. Her research interests include modeling and simulation and high performance simulation.