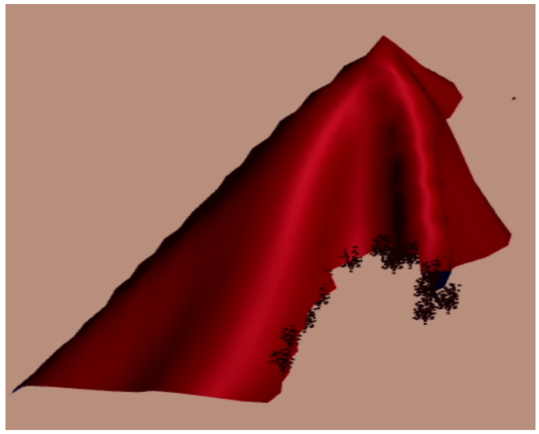
Simulating Burning Cloth

Angelica Loshak & Joseph Killian Jr.
Rensselaer Polytechnic Institute



***Abstract* -** Simulation is one of the strengths of computer graphics, whether it be for representing the physical world, or for artistic/entertainment purposes. This paper provides a method for simulating fire burning different types of cloth materials. It combines a mass-spring system with a particle system based off a fluid simulation to simulate the erosion and deformation of the cloth, and the smoke/ash particles respectively. The simulation is capable of eating away a given cloth completely, by using predefined ignition points, and manipulating springs to show the cloth crumpling as it burns. Smoke particles are present where the cloth is ignited. The results we obtained look fairly reasonable, but adding more elements to make the simulation more physically based would be beneficial for future work.

1 Motivation

When it comes to simulations of real-life phenomenon, there are various ways that one can approach the problem. One option is to try to make the simulation as realistic as possible, without concern for time constraints, or efficiency of the algorithms used. On the other end of the spectrum, the simulation can be very artistic and meant to be conducted only for the visual quality above all else for the particular domain. The simulation of burning cloth in this paper falls somewhere in the middle of these two. Some of the algorithms and methods used are done to mirror some real life aspects of how fire spreads, while some of the other choices were made to try and get a visually appealing result that does not necessarily reflect how a given material would burn in real-life. The goal was to get a functioning simulation and then test it out with some different material parameters, and see how the speed of the spread of the heat changes, as well as the different deformations/crumpling that occurs. Our fire simulation consisted of two components: A mass spring system to simulate the cloth, and a particle simulation for the smoke/ash being released from the cloth material. The next section goes through some related work that motivated the use of certain algorithms, or gave inspiration for how to solve some of the problems that we ran in to. The following sections will then describe the cloth and particle simulations, results, and limitations/future work.

2 Related Work

A fire simulation is far from a novel idea, and there has been work going on in computer graphics with some of the first work with fire simulations being conducted in the early 1990s. Chiba [4] as well as Perry and Picard [5] tackled the problem of trying to simulate fire, smoke, and how to accurately represent how fire spreads. On the particle simulation side of things when it comes to simulating smoke, much of the basis for our simulation is built upon the work from Foster and Metaxas [6] and their simulation for fluids and liquids.

Larboulette [2] looked at burning paper, and the how the fiber of paper affects how it burns. Although we are not looking at burning paper in particular, this paper utilizes a mass-spring system combined with a particle simulation, which is very similar to how we approached the problem. When looking further at modeling the smoke that the fire produces from burning the cloth Losasso [1] provided some insight as to how to store certain parameters at each node of the mesh, things like temperature, and whether a particle is currently ignited. For some of the other properties of how cloth is affected by being burnt, Liu [3] discussed fire and how it spreads, but in particular, how it crumples and deforms as it burns.

3 Cloth Burning

The cloth simulation consists of a collection of cloth particles all connected by springs, with one or more points designated as fixed points to prevent the cloth from falling off the screen due to gravity. The core challenges for these simulations were: How to manipulate the springs, how to handle spread of heat, where to store temperature and ignition status, how to smoothly remove triangles from the mesh, and how to change the force calculations as the cloth particles increase in temperature. There were three iterations that this part of the simulation went through, and this section will detail the challenges that were faced, and how the solutions to these problems came about.

3.1 First Iteration: Shrinking Cloth

The first attempt of trying to burn the cloth started with trying to just manipulate the springs without removing any triangles from the mesh. This would be done by having the triangles of the mesh shrink as they burn and then have the mesh shrink in on itself, without actually removing them. The key equation for manipulating the spring lengths was the below equation from [2] that described how to subtract from the rest spring length based on temperature over time.

$$l(t+dt) = l(t)\frac{factor * l_{init} * (T_a + T_b)}{k_s * c_s}$$

C_s is the material's dimensional stability, which describes the materials ability to keep its structure as heat and humidity change [2]. k_s is the spring stiffness that is a property of the material. Factor is another scaling constant that Larboulette [2] uses when spring stiffness and dimensional are low. Both C_s and factor are very small constants, so the approach that was taken for our purposes was to wrap them into just a single constant. This constant varied between $1e-6$ and $1e-8$, and required some trial and error to find reasonable values. So the equation for the rest lengths that our simulation uses is a simplified version of the one proposed above.

$$l(t+dt) = l(t)\frac{factor * l_{init} * (T_a + T_b)}{k_s}$$

As can be seen in this formula, it also requires values for T_a and T_b which are the temperatures for particles a and b connected by a given spring. The approach that was taken was to have each cloth particle in the system have its own temperature value that was initially set to be $60°$. To start the simulation, the cloth starts with any number of fixed points that are offset to allow the cloth to slowly begin to bend and hang from those fixed points. Once the cloth has come to rest a single particle was set to be $1000°$ and the heat was spread to all adjacent particles (Smoke/Particle simulation later added the functionality to start the fire at any point, not just when the cloth comes to rest). Heat was spread by scaling it by the time step, and in this first attempt, it was naively spread equally to all 8 particles adjacent (those particles surrounding the current particle in the cardinal directions). Once the temperature rose above the initial temperature heat would be spread. There were many problems that came about from this first attempt. The first big one came about when we introduced two fixed points. The cloth can't shrink into a single point if there are two points that are fixed. So, this added the necessity for some of the springs to be able to be "cut" off once they are burnt. Fire also burns upward due to various factors (pressure/buoyancy) so spreading heat equally does not make sense since the flame would be heating up the above particles more that those below and to the side. The other big error that we did not consider on this first attempt was keeping spring rest lengths consistent for each particle. The global rest lengths were changed as opposed to the local ones, which cause some issues. The images below in Figure 1 are what were obtained from this first iteration. At the beginning it looks somewhat convincing as the cloth crumples and deforms, but it quickly spins out of control. It also burned significantly faster than expected.

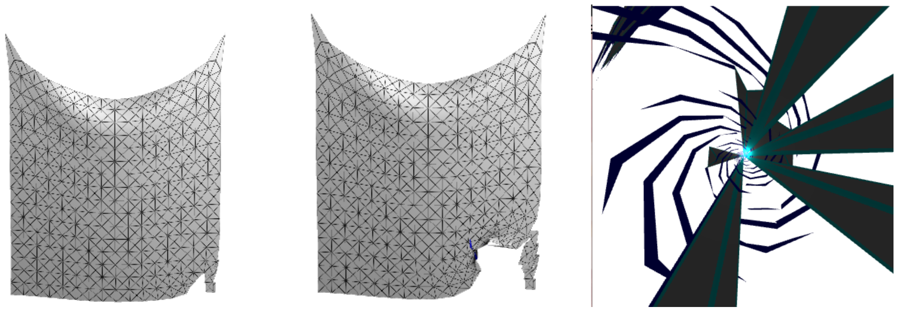


Figure 1: The first two images show how the simulation starts off with the fire starting slightly offset from the bottom right corner. The simulation however quickly spins out of control as shown in the last image.

The next step that was taken was to try and simplify some of the parameters instead of trying to solve everything at once. So, the next iteration focused on making sure temperature propagates more accurately and removing triangles from the mesh instead of trying to alter the springs at all.

3.2 Second Iteration: Removing Cloth

The main objective of this attempt was to try and get the cloth to slowly disappear by removing triangles from the mesh until there is nothing left. To make things more approachable, the springs were reverted back to how they worked before manipulating the rest length based on heat. Instead, we added other parameters for the cloth particles that indicated whether or not they were "burnt" or not and whether that particle was ignited. To do this we looked up the points at which different cloth materials would be lit on fire, which resulted in the table below in Figure 2.

Ignition Thresholds

Cloth	Silk	Denim	Paper	Cotton
482ºF	289ºF	410ºF	451ºF	410ºF

Figure 2: Shows the temperature values at which each cloth particle will be designated as ignited for different material types.

Once a particle exceeded this threshold, it would be set as ignited. To determine when a particle should no longer be considered in the simulation we set an arbitrary high temperature of 2000° to indicate that the particle was burnt and removed. To fix the problems we had with how temperature was spreading too evenly and quickly, some of the parameters were changed for how it is scaled by the time step were altered. The formula that was used for temperate spread was defined as:

$$P_T += \Delta T * (timestep * 0.0001)$$

To fix the problem of spreading heat evenly, more heat was spread to the particle directly above the current particle, and to those in its upward diagonals. Instead of spreading heat once a particle's temperature exceeds it's initial value, it spreads heat once it has been ignited as defined by the table in Figure 2. The particle above received $4\Delta t$ and those on the upward diagonal received $2\Delta t$. These values are not physically accurate, and are instead what was found to be a good value for how the simulation visually appeared. The distribution

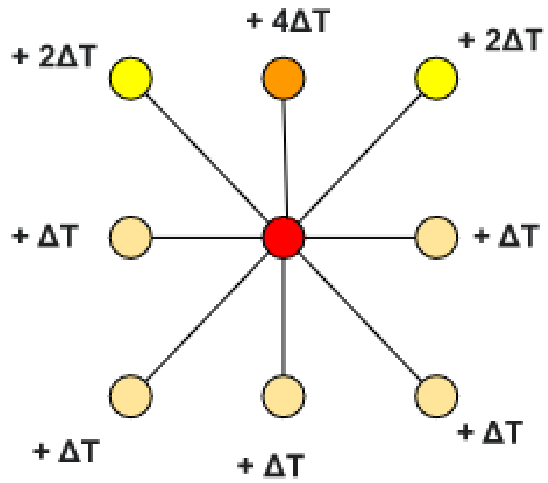


Figure 3: Diagram shows how heat is distributed to the adjacent particles to an ignited particle.

of heat from an ignited particle is represented in Figure 3, and note that this ignited particle does not lose any heat when spreading heat to the adjacent particles.

When deciding when to render triangles of the mesh, once a particle was designated as burnt (exceeding the 2000° threshold) then any triangles that the particle is a member of are deleted from the screen. An odd bug that is still present that was run into during this process is that triangles were still being rendered even after the check for whether one of their member particles was burnt. It would remove the desired triangle, and then draw another one somewhere else. The fix for this bug was to draw a triangle where each point is set to (0,0,0) instead of trying to remove it. So, the number of particles and triangles remains constant in the simulation, even though it looks as if they are being removed. The results of this simulation were very promising as we did end up with the whole cloth being consumed completely, regardless of the number of fixed points that are present in the given cloth. Figure 4 shows some images from this iteration.

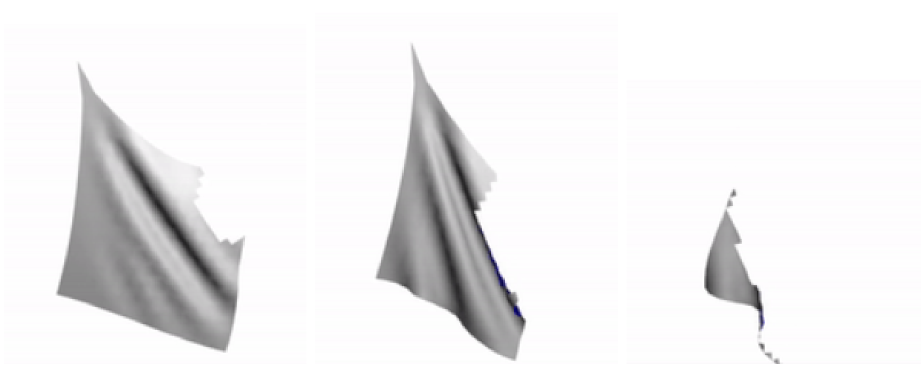


Figure 4: These images show the stages as the triangles from the mesh are removed as the fire spreads. The last image shows the cloth as it is falling when both fixed points have been detached.

This iteration was a step in the right direction but there were still some problems that were present. Heat is only being spread to adjacent particles, but when the cloth folds over, heat should be spreading to the particles that are close together in proximity, not just adjacency. The removal of triangles is also very jarring, at the edges of the cloth are very jagged. A smoother edge where the cloth is being burnt would make things look a lot better. So, this leads into the last and current iteration of this portion of the project, which combines elements from the first two iterations. Mainly adding back in the spring manipulations, and making sure things are calculated more consistently and correctly.

3.3 Corrected Springs & Heat Spread

The way to solve the jagged removal of triangles is to add back in the logic regarding changing the spring lengths based off of temperature. The way that this was solved was by keeping track of 12 unique springs for each particle that needed to be kept consistent with the adjacent particles. Before the spring rest lengths were universal, with this iteration, each spring has its own dynamic rest length. Each particle can have up to 12 springs depending on its location in the mesh. Four structural springs, four shear springs, and four bending springs. The particles on the corners and the edges of the mesh may have fewer springs due to lacking adjacent particles in certain directions. Figure 5 shows off the location and labels for each of these springs connected to a particle.

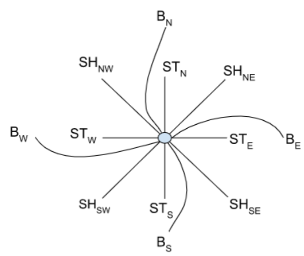


Figure 5: ST stands for structural springs, SH stands for shear springs, and B stands for bending springs, with the subscripts corresponding to the cardinal direction.

The spring forces being calculated had to be consistent, so for example the $N_{Structural}$ spring of one particle had to match the $S_{Strucutral}$ spring of the particle directly above it. So when going through the loop for force and rest length calculations, a check is made to make sure this rest length for the spring is identical, and if it is not, they are set to be the same. Since the force calculations go through each particle, only half of the contribution to reducing the rest length is counted for each particle to avoid double counting this reduction. There is also a check to prevent the rest length of the springs from reaching zero. Combining this with the removal of triangles resulted in a much smoother removal of triangles. Two other small fixes that were made during this iteration was adding the heat contribution of particles close to each other by having a distance threshold, and removing the force contributions of a particle once it has been burnt. For the proximity fix, once particles are close enough, they begin to spread heat to one another. For stopping the force calculations, once either of the particles being used for calculation is burnt, the force function returns (0,0,0). Along the way there were still some hiccups which resulted from having negative or incorrect rest lengths. triangles would stretch too much, or shrink into the location of the fire as shown in Figure 6.

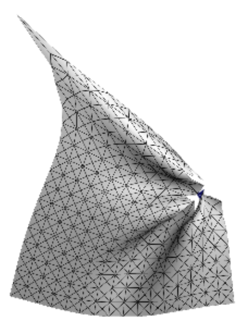


Figure 6: Blooper image showing what happens with incorrect spring rest lengths, with the cloth seemingly being consumed by a black hole.

The outputs of this final iteration are shown in the results section combined with the particle simulation for the smoke. There are still problems associated with this element of the fire simulation, and those will be discussed further in the limitations section along with the limitations of the smoke simulations as well.

4 Particle Simulation

Adding particles of fire to our simulation was another important task to complete the visual aspects of the project. The particles make it look realistic and help convey to the viewer that the cloth is actually on fire. Along with that, they serve a functional purpose of showing the viewer where the fire is located and the direction it is moving.

The first challenge was figuring out how to get openGL to render objects on top of the cloth. We decided to simplify fire into a fluid object because fire is a plasma and can exhibit fluid-like behaviors. Since we had already written functional fluid code for homework 2, it was easy to modify the code to suit our needs.

Once we figured out how to render fluid particles on top of the cloth, we moved on to allowing the fire to interact with the cloth object. This way, the fire object had information about the amount of heat in any given position on the cloth and how the heat was spreading. This helped the fire particles determine where they should be rendered and how long they should stay on the screen.

Initially, the particles were appearing in positions where the cloth had already burnt away because the cloth mesh was not being properly deleted. In order for the cloth and fire to work together, they had to communicate where the burning had already occurred and where there was no longer any cloth.

Finally, we had to render more particles simultaneously on all areas of burning. Our starting fluid code could only support rendering one body of fluid at a time. Since there were multiple spots of burning, we had to adapt the code to render fire at every point on the cloth that was expected to be on fire. Rendering so many particles could cause lagging. However, we were able to overcome this by adjusting the density, point sizes, and time steps. After the particles met our standards, we experimented with the lighting and shading of the cloth to help the particles stand out and suggest fire.

Although there are things we need to improve about the fire particles, such as their texture and motion, overall, they significantly enhance the visual appeal of our project.

5 Results

Once we combined the two simulations, many different shapes and materials of cloth were tested out to see what the output would be. There are many more images and videos than what are shown in this report, and these are just a selection of a few of them. Figures 7, 8, and 9 all show off some of the aspects of both the particle and cloth elements of the simulation.

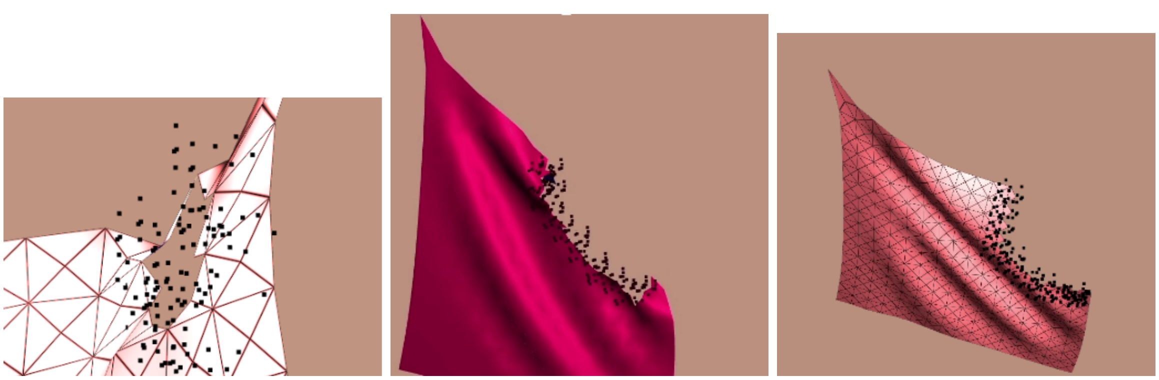


Figure 7: The first image shows a closer look into the how some of the triangles deform as they burn. The second an third images shows the smoother frontier of the fire as it burns.

Figure 8: Some other assorted images showing different types of cloth shapes being burnt.

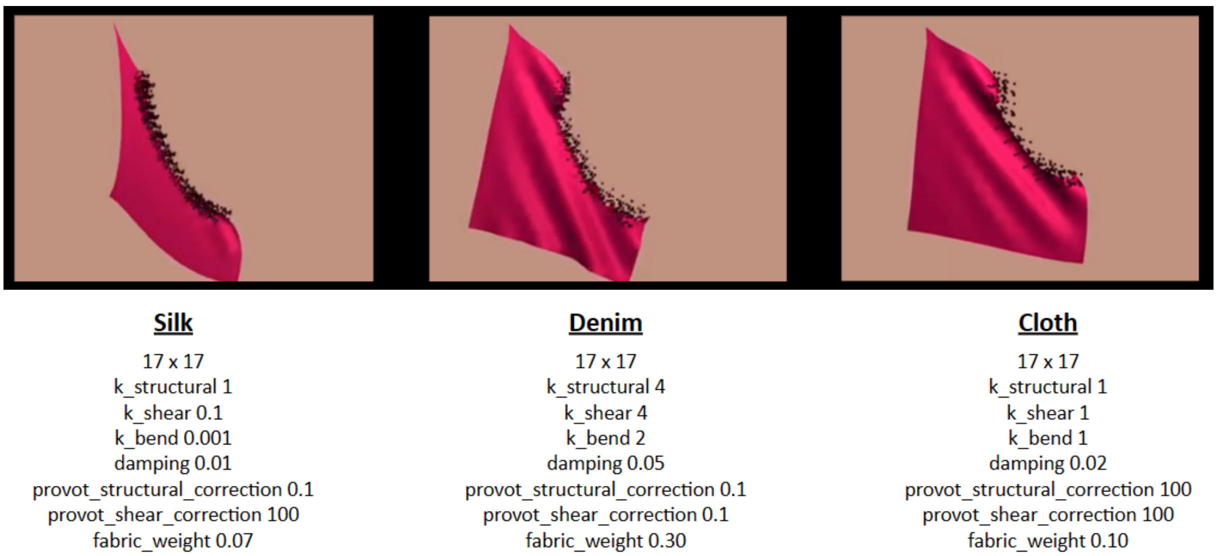


Figure 9: Shows off the comparison of different cloth materials being burnt, with silk on the left, denim in the center, and a more general cloth on the right. The parameters used to represent each of these cloths is displayed beneath each cloth.

6 Limitations

Our simulation takes elements of how fire burns in reality, and then some other artistic liberties, but there is still plenty of room for improvement. In general, incorporating more elements that we neglected or skimmed over from the physics of fire would be helpful in making the fire look better. The spring adjustment still possesses some bugs with some springs overstretching when they are not supposed to, and we assumed that the stiffness of the springs remains constant while being burnt. The crumpling that is present in our current iteration of the simulation is also accidental in a sense, since it originates from the altering of the spring rest lengths. One crucial element that was neglected in terms of spreading heat was the heat equation. In our simulations the amount of heat in the system is always increasing. This causes problems with how the spread of heat is scaled by the time step and requires some trial and error. Connecting the heat equation to our simulation would solve some of these problems. The meshes that we used for each material are also identical. Although we varied the parameters for the spring stiffness, weight, and provot correction values, the structure of these cloths is very different and that should probably be reflected in the meshes as well. Cloth materials are also multi-layered in real life, so doing a simulation with thin-shelled cloth would also be more representative of real cloth. When it comes to how smoke is represented, our simulation is very limited. We lack visual flames, and the smoke has to be rendered in bounding boxes along the frontier of the flame. The texture of the cloth also remains the same even when being burnt, so we would need some sort of change in texture to be blackened as it burns to better reflect reality.A big issue with this simulation is that as the cloth scales up in size, it starts to run much slower. We loop through the particles once to calculate all the forces, again to check for particles close in proximity, and then multiple times for provot correction. Utilizing a different methodology other than Euler for force calculations, and finding shortcuts for some of the other steps of the process would help the simulation run smoother for larger cloths. Most of these issues and some of the decisions that were made originated from the time pressure, and with more time to work on these simulations, the ideas discussed in this section would be the direction that we would go down.

7 Conclusion

For this paper, we wanted to create a simulation to model different cloth materials as they burned. The methods we used to achieve that were a mass-spring system to manipulate the cloth, combined with a fluid simulation to represent the smoke being produced. The cloth simulation part changes the spring lengths as a function of the the temperature of the current particles, and removes triangles from the mesh based off the particle temperature values. The smoke particles part renders fluid particles wherever a fire should be at that point in the mesh. Some shortcuts from how fire works in reality were taken, but future work could be done to incorporate more of the physically based aspects of both fire and cloth to produce more convincing and accurate results.

7.1 Division of Work

When starting work on the project, both members worked cooperatively on the same set of code, but we soon realized the two separate directions of work to be completed for the project. Below is a more detailed description of who completed which aspects of the project.

Angelica & Joseph: Both members worked cooperatively on the first iteration of the cloth burning simulation. Once we obtained some initial results and bloopers, we decided to split the work between the cloth and particle simulations. Both contributed equally to the creation of this report and the accompanying presentation given in class.

Angelica: Did the work with smoke particles, and combining the two simulations so that the cloth and fluid simulations can run at the same time. Also experimented with the color of the particles, cloth, and background.

Joseph: Did the work for iterations two and three for the cloth burning portion of the assignment. Worked on making sure cloth burned completely, heat was spread consistently and correctly, and that the springs were manipulated for a smoother removal of mesh triangles.

References

[1] F. Losasso, G. Irving, E. Guendelman, and R. Fedkiw, "Melting and burning solids into liquids and gases," IEEE Transactions on Visualization and Computer Graphics, vol. 12, no. 3, pp. 343–352, 2006.

[2] C. Larboulette, P. Quesada, and O. Dumas, "Burning paper," Proceedings of Motion on Games, 2013.

[3] S. Liu, Q. Liu, T. An, J. Sun, and Q. Peng, "Physically based simulation of thin-shell objects' burning," The Visual Computer, vol. 25, no. 5-7, pp. 687–696, 2009.

[4] Chiba N, Muraoka K, Takahashi H, Miura M (1994) Two-dimensional visual simulation of flames, smoke and the spread of fire. J Vis Comput Anim 5:37–53.

[5] Perry CH, Picard RW (1994) Synthesizing flames and their spreading. Massachusetts Institute of Technology, Cambridge

[6] Foster N, Metaxas D (1996) Realistic animation of liquids. Graph Model Image Process 58:471–483.