

UC Riverside

UC Riverside Electronic Theses and Dissertations

Title

Smoldering Combustion: Animation and Control

Permalink

<https://escholarship.org/uc/item/6k38j5s6>

Author

Celly, Bhrigu

Publication Date

2015

Supplemental Material

<https://escholarship.org/uc/item/6k38j5s6#supplemental>

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
RIVERSIDE

Smoldering Combustion : Animation and Control

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

by

Bhrigu Celly

August 2015

Dissertation Committee:

Dr. Victor Zordan, Chairperson

Dr. Eamonn Keogh

Dr. Walid Najjar

The Dissertation of Bhriugu Celly is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

Working on my dissertation has taken me through a long journey, full of enriching experiences and support from a lot people, who helped me achieve my vision, and to whom I would ever be indebted.

First and foremost I wish to express my immense gratitude to my advisor Dr. Victor Zordan. His support and encouragement over several years during the course of my Ph.D., especially during the tough times during this pursuit, was invaluable. I appreciate the time he contributed towards our insightful discussions about this research which made my Ph.D. experience productive and stimulating.

For this dissertation I also appreciate the involvement of my committee members, Dr.Eamonn Keogh and Dr. Walid Najjar and thank them for their helpful comments and interest in my work.

My gratitude is also extended to Amy Ricks. I couldn't have completed all the required paperwork and delivered it to the correct place without her.

I gratefully acknowledge the funding source that made my Ph.D. work possible. I was funded by my employer Sony Pictures Entertainment.

No acknowledgments would be complete without giving thanks to my parents Varsha and Manu celly for all their love and encouragement, for giving me a good foundation and supporting me all my pursuits. In addition my mother-in-law Dr. Aruna Batra was the one who helped me in the editing of this thesis. Thank you for the time spent in proof reading .

I would also like to thank my 5-year old son, Ojas Celly who has been the bright light in my life. His presence always re-energized me after a long day of work and school. In his own little ways he pushed me to finish this dissertation.

Last, but certainly not the least, I would like to express appreciation to my beloved wife Navnidhee. She has patiently endured many, many long hours alone while I worked on my dissertation. We both have learned a lot about life and have strengthened our commitment and determination to each other and to live life to the fullest. I am grateful to her for her faithful support during the final stages of this Ph.D. Thank you.

This work is dedicated to the Un-manifest Supreme Force
That energised me and all those who supported me
To complete it successfully

ABSTRACT OF THE DISSERTATION

Smoldering Combustion : Animation and Control

by

Bhrigu Celly

Doctor of Philosophy, Graduate Program in Computer Science
University of California, Riverside, August 2015
Dr. Victor Zordan, Chairperson

This work presents a model to simulate, animate and control smoldering combustion. Smoldering is a slow, flame-less, low temperature combustion that is sustained by chemical heat generated by fuel oxidation in an oxygen limited environment. In most cases under the right conditions it is the pathway to flaming. Researchers so far have focused on flaming combustion and we introduce smoldering to the graphic community.

In this work we use chemical oxidation reaction, chemical heat generation, heat diffusion, and oxygen diffusion to model smolder propagation. We simulate fuel oxidation, in voxelized space, which forms the basis of chemical heat release. This heat release is used to sustain and propagate smolder. As this combustion is flame-less, chemical heat is the only energy generating mechanism that drives this phenomenon. On oxidation the combustible portion of fuel converts to ash. Ash formed acts as an insulating layer on the fuel thus reducing heat loss, which helps in sustenance and propagation of smolder. The smoldering has a feature of being porous, and permeable to flow of diffused air. This porosity acts as a way to get oxygen through the fuel to facilitate and sustain fuel oxidation. We model ash and wind as they play a very important part in propagation, flaming and extinguishing of smolder. Coupled with

these thermal changes we also propose a voxel based volumetric deformation model based on springs to model smoldered fuel. To further that based on the simulation we also propose a visualization scheme that allows us to render these effects.

We hereby extend realism to combustion by introducing smoldering, oxidation effect on smolder, transition to flaming, fuel deformation and ash formation to the graphic community.

Contents

List of Figures	xi
List of Tables	xiii
1 Introduction	1
1.1 Motivation	2
1.2 Goal	2
1.3 Methodology	3
1.4 Outline	4
2 Smolder and Burning	6
2.1 Combustion	6
2.1.1 Flaming	7
2.1.2 Smoldering	7
2.2 Previous Research and Motivation	8
3 Smoldering Simulation	12
3.1 Voxels	13
3.1.1 Voxel Classification	13
3.1.2 Voxel properties	14
3.2 Heat Transportation	15
3.2.1 Heat Transfer	15
3.2.2 Heat Diffusion	16
3.3 Oxygen Transportation	18
3.3.1 Surface Oxygen Transfer	19
3.3.2 Oxygen Diffusion	20
4 Reaction Zone	23
4.1 Reaction Zone	23
4.1.1 Pyrolysis	25
4.1.2 Chemical Oxidation	26
4.1.3 Reaction Fuel Limit	32
4.1.4 Pyrolysis propagation	32
4.2 Reaction Front propagation	33
4.2.1 Oxidation Front propagation	33
4.2.2 Oxidation Front Detection	35
4.3 Forward and Reverse Smolder	36

4.4	Ash	38
5	Smolder Propagation	39
5.1	Self sustained smolder	39
5.1.1	Assisted smolder	40
5.2	Transition to Flaming	43
5.2.1	Accelerated Smolder Propagation	44
5.2.2	Smolder Extinguishing	44
5.2.3	Heat Insulation and Flaming	45
6	Secondary Deformation	46
6.1	Mass-Spring Mesh	48
6.2	Tetrahedral Mass Spring Voxel	48
6.2.1	Automatic Voxel Based Spring Mesh Generation	50
6.2.2	Connecting Simulation Voxels To Spring Voxels	51
6.2.3	Deformed Marching Cube	52
6.3	Controlling Deformation	54
6.3.1	Temperature Based Deformation	54
6.3.2	Gravity based forces	56
6.3.3	Mass Spring Force Transfer	57
7	Rendering of Smolder	59
7.1	Reaction Zone	61
7.2	Oxidation Front	65
7.3	Ash	66
8	Implementation	68
8.1	Cigarette	68
8.2	Cigar	73
8.3	Bunny	74
8.4	Sheet	75
8.5	Briquette	76
8.6	Torus	77
8.7	Leaf	78
8.8	Matchstick	79
9	Conclusions and Future Work	80
9.1	Conclusion	80
9.2	Future Work	81
9.2.1	Sooth and Char Formation	81
9.2.2	Fluid Model for Wind	81
9.2.3	Fuel and Ash Fracture	81
9.2.4	Forest Fires	82
9.2.5	Internal collision detection	82
9.2.6	Climate Effect on Smolder	82
	Nomenclature	83
	Bibliography	86

List of Figures

2.1	Smolder Material	7
2.2	Smolder Simulation	8
3.1	Voxel Classification	13
3.2	Voxel Lifecycle	14
3.3	Voxel Properties	15
3.4	Cigarette Ash	17
3.5	Cigar Burn Cone and Ash Formation	18
3.6	Cigarette and Cigar Oxidation cones	19
3.7	Oxidation front with ash diffusion parameters	22
4.1	Pyrolysis and Chemical Oxidation	25
4.2	Reaction Zone and Oxidation Heat	26
4.3	Level Set using normals	34
4.4	Fuel showing forward and reverse smolder	37
4.5	Cigarette Cutout for Voxel arrangement showing forward and reverse smolder	37
5.1	Self Sustained Smolder	40
5.2	Assisted Smolder Airflow parameters	41
5.3	Assisted Smolder	42
5.4	Wind Mechanics	43
6.1	Tetrahedrons in a Spring Voxel	49
6.2	Spring Types	50
6.3	Spring arrangement on a Sheet	50
6.4	Automatic Spring Mesh Generation Steps	50
6.5	Data Exchange To Deform	52
6.6	Simulation Voxel Mapping to Spring Voxel	52
6.7	Spring arrangement on a Sheet	53
6.8	Cigarette Spring Mesh Showing Ash Deformation	54
6.9	Matchstick Simulation Comparison	55
6.10	Sheet Simulation Comparison	56
6.11	Cigarette Spring Mesh Showing Deformation	57
6.12	Gravity based deformation render	57
6.13	Voxel Expansion	58

7.1	Texture Flow	60
7.2	Cigar Reaction Zone	61
7.3	Briquette - Light Source Mapping to Reaction Zone	62
7.4	Cigarette - Light Source Mapping to Reaction Zone under Wind	64
7.5	Cigarette Side Burn Rendered Reaction Zone	64
7.6	Texture Property Mapping	65
7.7	Heat Release from Oxidation Front	66
7.8	Briquette with Oxidation front rendered	66
7.9	Briquette showing Ash Texture Formation	67
8.1	Cigarette Self Smolder Comparison	69
8.2	Pyrolysis with different tobacco properties	70
8.3	Draw In and Out Simulation	71
8.4	Cigarette Draw in and Out Motion	71
8.5	Side Burn Actual vs Simulation	72
8.6	Cigarette Simulation With Side Blow.	72
8.7	Cigarette Simulation With Ash Deformation	73
8.8	Cigar Simulation With Self Smoldering	73
8.9	Bunny Self Smolder Simulation	74
8.10	Bunny with No Ash	74
8.11	Sheet under secondary deformation	75
8.12	Briquette with ash	76
8.13	Briquette no ash	76
8.14	Torus with Ash formation	77
8.15	Torus with oxidation front	77
8.16	Leaf Rendered with Deformation	78
8.17	Leaf showing pyrolysis and Ash formation	78
8.18	Matchstick with deformation and Ash formation	79

List of Tables

4.1	Pseudo Code Heat Simulation	24
4.2	Units for Constants Used	31
4.3	Pseudo Code - Oxidation Front Generation	36
6.1	Pseudo Code - Spring Mesh Auto Generation	51
8.1	Cigarette	69
8.2	Cigar	73
8.3	Bunny	74
8.4	Sheet	75
8.5	Leaf	78
8.6	Matchstick	79

Chapter 1

Introduction

This dissertation describes research and development undertaken as a portion of studies in attaining the degree of Doctor of Philosophy in Computer Science. This work contributes towards advancement in the animation and control related to smoldering combustion.

Smoldering is an important phenomena and an underlying cause of several fires. These are not limited to wildfires, subsurface fires, biomass and aerospace fires. It has been a cause of concern in safety in different areas such as aerospace and commercial flights. Smoldering is of interest because of its capability of producing harmful and damaging results.

The study of smoldering helps in environmental science, forest management to name a few as stated by Rein [51] and Ohelmler[45]. Advanced animation studies can help in preventing and in some case also recreating some of these damaging results.

1.1 Motivation

The present work began with a quest for greater realism in simulation with the desire to introduce smoldering to the graphics community that would allow simulation of a wider range of visual effects with visually compelling models. This could be of immense value not only in motion picture and gaming industry, but in a wide array of technological fields related to environment and energy, in which smoldering combustion would have direct applicability.

1.2 Goal

In computer animation modeling, natural phenomenon such as flames, burning and object decomposition always present a challenging problem. Not only expensive, creating these burning effects could be potentially dangerous. Simulating these effects can thus make it easier to overcome some of these issues.

There have been multiple attempts to model surface burning coupled with flame propagation. As flame is the driving force behind any current simulation, there has always been a requirement to have a flame as a heat generating mechanism to allow combustion spread and heat transfer.

Smoldering is a slow, low temperature, flame-less form of combustion that is sustained by the heat generated by the chemical reaction of oxygen reacting with the fuel. Smoldering involves many physical processes like heat distribution, pyrolysis, oxidation chemical reaction, oxygen diffusion and fuel to ash transition.

Most of the research identified has been done in rendering flames and modeling flame spread, whereas smoldering has not had much attention.

In this dissertation slow flame-less smoldering combustion is simulated and

visualized using chemical energy as the driving process to create visually credible results, with the goal to add more realism to the current combustion models, which can allow us to study other aspects like flaming start that initiates flameless combustion and fire safety.

1.3 Methodology

In this project we have used a simple system that simulates oxidation of fuel in voxelized space. Each voxel contains fuel that is oxidized at pyrolysis temperature to release energy and create ash. The heat released during the oxidation process is used to increase the temperature of the neighboring voxels. As the temperature in the voxel increases, the fuel gets pyrolysed and awaits sufficient diffused oxygen or surface oxygen to get oxidized. Over time oxygen diffuses in, to oxidize the fuel and generates heat to sustain pyrolysis.

Due to the nature of this form of combustion the formation of ash plays an important role in this process. It helps in retaining oxidation chemical energy, and directs the combustion. The portion of fuel that is under combustion has been introduced as the reaction zone. This reaction zone is the zone between fuel and ash. This zone has two fronts, one where reaction zone connects to un-burnt fuel called pyrolytic front and second where reaction zone connects to ash called oxidation front.

A volumetric model is used to model the combustion in the solid fuel and track the two fronts over time. While under combustion, fuel in the reaction zone, while being turned into ash, loses moisture and becomes porous. This ash then acts as insulation around the reaction zone and simultaneously permitting the flow of oxygen to the reaction zone by diffusion.

The oxygen diffusion is also modeled to reaction zone and heat retention by the reaction zone due to ash formation. As by definition smoldering is an oxygen deficient process, the heat energy released on oxidation in the reaction zone has to be sufficient to balance the heat required for smolder propagation.

A voxel based spring mass deformation model is coupled with the simulation system. This forms the basis of modeling secondary deformation. Heat release from the simulation forms the input and driving mechanism to the secondary deform system.

In this dissertation the following physical phenomenon related to smoldering are addressed:

- Chemical Reaction
- Chemical Heat Energy Release
- Heat Distribution
- Oxygen Diffusion
- Wind Oxygen Transport
- Wind Effect on Heat Distribution
- Heat to light conversion
- Smolder Extinguishing
- Transition to Flaming
- Thermal Deformation

1.4 Outline

Literature reviews conducted during the course of this work are briefed in Chapter 2. It focuses on burning and smoldering, their differences and similarities. We

also discuss the background work done in combustion and go over our approach. Chapter 3 explains our model and the various parts of the voxel model. Chapter 4 describes the smolder zone, and different reactions that go into smolder creation and sustenance. In Chapter 5 the different elements associated with simulation are added, such as wind and flaming that helped us widen the effects we could produce with our simulation. It also describes the factors that would cause smolder propagation and extinguishing. Chapter 6 discusses the secondary deformation that is caused by smolder. In Chapter 7 the visualization details are presented. Chapter 8 discusses the implementation results. Chapter 9 has conclusions and potential future work.

Chapter 2

Smolder and Burning

This section presents the results of literature reviews conducted during the course of this work, focusing on burning and smoldering, their differences and similarities and examines the multiple aspects of combustion that closely relates to the present work.

2.1 Combustion

Combustion is a process in which the fuel is heated by ignition and mixes with oxygen to start an exothermic process [60]. This exothermic process is followed by decomposition of the fuel by release of carbon dioxide and moisture under extreme heat.

Combustion process can be roughly classified in two basic categories i.e., detonation and deflagration, based on the reaction rate. Detonation is a very high speed reaction followed by shock waves and compression effects. Deflagration is the slower of the two and is accompanied by fire. It can be further divided into Flaming and smolder. In both these cases the spreading takes place when heat is released on chemical oxidation.

2.1.1 Flaming

Flaming combustion is a widely discussed component of combustion. All the earlier research on combustion in computer animation and control has focused only on this aspect. Fire spread is an area of high interest in the graphics community and it forms an important component for any of the combustion models. This form of combustion is high temperature, high rate and occur on the surface of the fuel, whereas smoldering that is flameless occurs at lower temperatures.

2.1.2 Smoldering

Smoldering is a slow, low temperature form of combustion that derives its heat from the oxidation chemical reaction. A common occurrence of this flame-less form of combustion is found in materials like coal, tobacco, decaying wood and sawdust as shown in figure 2.1.



(a) Polyethylene



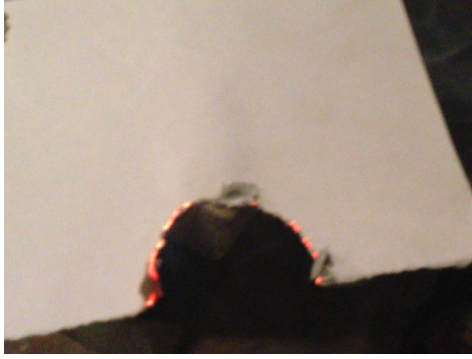
(b) Saw Dust



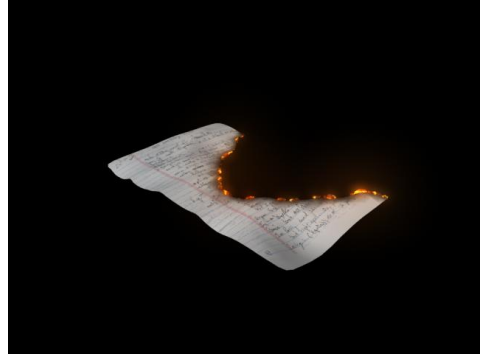
(c) Charcoal

Figure 2.1: Smolder Material

Smolder is initiated by weak sources of heat that would not be enough to start a flame. They produce smoke that is a low temperature and hence would be cooler than normal flaming smoke. As the process is oxygen deficient it produces harmful gases. It eventually will result in flaming form of combustion under critical conditions.



(a) real smolder



(b) simulated smolder

Figure 2.2: Smolder Simulation

Smolder reaction is characterized by two fronts. In one front the exothermic reaction of oxidation of fuel takes place and the other where heat is removed due to heat transfer to surroundings. In order for the front to move, the next heat released should be enough to balance the amount of heat required for propagation. Figure 2.2 shows the appearance of oxidation front.

2.2 Previous Research and Motivation

Most work done in computer graphics involving combustion revolves around fire and flames. Heat transfer revolves around flames and flame spread. Smoldering adds more realism to the current combustion models and allows us to study other aspects like flaming start that initiates flame-less combustion and fire safety. However, it has not had much attention.

Early models involving combustion propagation have always been coupled with flame generation. Some of the initial approaches, such as by Reeves were modeled procedurally using particle systems and were driven by how they looked with no physical process driving them [50]. Chiba et al[13] model a two dimensional grid based flame

spread and smoke simulation. Some later models like Perry et al [49] were physically driven models that are modeling combustion spread on the surface of solids where combustion was driven by fuel density and oxygen. Most of these early models were two dimensional surface models. Pegoraro and Parker [48] model physically based flames and rendering soot by changing reflective properties. Barer et al in [5], [6] using a physically based model on a multi-resolution grid to model flames and its turbulent effect on the surrounding gas.

Some of the later models used solid fuels as source of energy to create and sustain flame production.

Inakage in one of the earlier attempts to model flames, created a simple model to render flames using a physical reaction [19]. Nguyen et al [42] modeled the chemical reaction of flames but do not pay too much attention to the solid fuel producing the flame. The fuel surface boundary is used as the reaction front, that injects gasified fuel to start and stop the simulation. Malek and Keyser [35] propose a physically based model with its main focus on achieving interactive rates so the model is simplistic. They also introduced the idea of pyrolysis and time varying pyrolysis [36].

Ishikawa et al [20] modeled burning with a physically driven simulation using chemical energy to drive their simulation. Lasasso et al [32] treated the solid objects as a highly viscous liquid and melting it over time. Riensche and Lewis [53] added properties like moisture and wood grain to the solid fuels to influence combustion deformation while using level sets to decompose. Min and Metaxas [38] have a voxel based model that is a more detailed combustion model with its primary focus is fire simulation. Ihm et al [17] used chemical kinetics to drive gas based animation. Jeong and Kim [22] model physically based combustion waves with no geometry decomposition.

Nguyen et al, Min and Metaxas and Ishikawa et al [42],[38],[20] do not account

for solid decomposition in their models. All these models treat pyrolysis and oxidation in a single step. If the temperature is right the fuel gets consumed as soon as it gets pyrolysed. In order to model smoldering we need both the processes to be treated separately as smoldering is oxygen limiting and pyrolysed fuel will only be consumed if sufficient oxygen is available. Due to slowness and lower temperature during smoldering the process of oxidation and pyrolysis will happen at different times, as discussed by Rein[51]. Other decomposition processes such as corrosion and erosion can also be modeled using decomposition as shown by Wojtan et al[61].

Smoldering is not a fully surface reaction and to accommodate subsurface combustion a volumetric model is required. Zhao et al [63] introduced a non physical approach using LBM. They model the burning solid as a volumetric object and all combustion is set to the iso surface they select. Even though their model is volumetric they do not account for any decomposition. Zhu et al [64] have a physically based model that defined burning solid object as a volumetric data set using GPU acceleration with a multi grid approach. They model ash as only a texture change that does not have any bearing on the simulation. In the discussed models ash residue either gets dissolved or gets fully consumed. We model and simulate ash formation to incorporate its effect on fuel oxidation.

Secondary deformation plays an important part in making the simulations more realistic. There have been several attempts to deform fuel while it is still under combustion. Liu et al [30] had described a model that was based on gradient, to deal with thin shells. Their model is not totally physically based. They model large secondary scale deformation using temperature and heat based stress that are empirically generated. Melek and Keyser [37] also model secondary thermal object deformation applying physical processes to the edges of a proxy using Free Form Deformation and barycentric

coordinates. Both these models are based on thermal stresses caused during combustion and forms the basis of their deformation. Larboulette et al [26] base their two dimensional model on papers fibers composition and use particle mass springs couple with that distribution equation. They do model paper fracture based on weakening of internal fibers. Jeong et al in [23] and [21] model two dimensional crumpling in shrinking of leaves due to loss of water and also cloth and paper under thermal stress, Narine et al [41] model crumpling of sheets. We model fuel under thermal stress and ash as it deforms, using a voxel based modified mass-spring system coupled with a modification to marching cube algorithm.

Air flow has been an important controlling parameter in progression of flame spread. Perry and Picard [49] use air with the curvature and surface geometry to determine fire spread. Beaudoin et al [8] model growth of the burning zone using multiple parameters such as, fuel, oxygen, wind and surface orientation. The air field effects the direction of movement of flame front. Lee et al [27] use a pre-calculated air field to provide control to the animator. Air field and terrain slope determine direction of the fire spread. Zhao et al [63] use LBM to model wind to control speed of the spread of the fire front. All combustion models are fuel dependent with having sufficient oxygen supply all times. Due to nature of smoldering any increased air flow increases oxygen diffusion and thus increase fuel oxidation increasing heat release.[44]

Smolder requires energy to sustain, and if it is not sufficient to drive, the oxidation front would not be able to sustain and smolder will extinguish. Airflow while increasing the oxygen available to smolder could also increase convection heat losses that remove heat from the smolder zone. This could eventually cause a heat imbalance leading to extinguishing smolder.

Chapter 3

Smoldering Simulation

Different parts of smoldering process are described in this section, followed by what goes into creating a smoldering simulation. Fuel gets heated, reacts with oxygen and starts the oxidation reaction. Oxidation of fuel produces heat, light and ash. Heat generated will produce a self sustaining reaction until oxidation stops or the fuel loses heat and gets cooled.

Fuel needs to be raised to a particular temperature where it is ready for chemical reaction. In order for a fuel to react with oxygen and reach a self sustaining state heat generated should be more than the heat losses. The main driving mechanism for smoldering is heat from the oxidation of fuel. Smoldering combustion will propagate when there is a self-sustaining reaction and oxidation of fuel continues to produce heat. Fuel on being completely oxidized produces ash. Ash has very low thermal conductivity causing it to act as insulation, thus reducing heat losses from oxidation. Ash is porous allowing oxygen diffusion from the surface. This property of ash allows smolder to sustain and propagate.

In our simulation, fuel is voxelized and each voxel is a fuel source, having the

ability to fully turn to ash. Any voxel containing fuel has a potential to get pyrolyzed, and then oxidized if there is sufficient oxygen available. Oxidation of fuel produces heat to increase the temperature of the current and neighboring voxels; and as a by-product of oxidation, the fuel turns to ash.

3.1 Voxels

3.1.1 Voxel Classification

We classify voxels into four categories as listed below as shown in figure 3.1. We do this so we can differentiate their properties during the simulation. They would differ in thermal conductivity, oxygen diffusion, heat diffusion, volatility and heat generation capabilities.

$$V_{i,j,k}^S = \{\text{Fuel, Reaction Zone, Ash, Air}\}$$

Air voxels do not contain any fuel. They act as an oxidation source and heat distribution sink to voxels that have fuel. They also have different thermal conductivity and porosity properties as compared to other type of voxels.

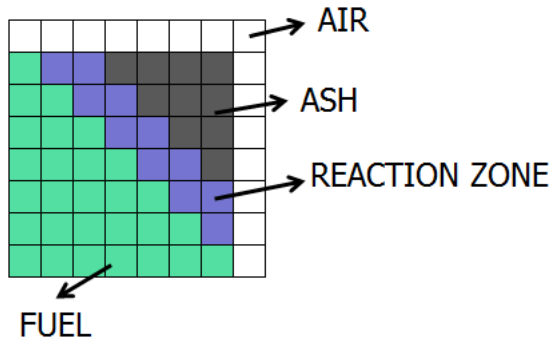


Figure 3.1: Voxel Classification

Fuel voxel are a fuel source that have not chemically reacted. These are at this stage not active and cannot get oxidized. When a fuel voxel reaches pyrolysis temperature it moves to a reaction zone.

Reaction zone voxels are either ready or in process of getting oxidized. They liberate heat and light when they are being oxidized. Since heat is the driving force behind smolder, this category of voxels play the most important part in the simulation. In order for a fuel voxel to turn to ash it needs to be part of reaction zone before it converts to ash.

Once a voxel gets pyrolyzed it gets active and changes state to be in the reaction zone. It may be temperature ready to be oxidized but if the voxel does not have enough oxygen it remains in pyrolyzed state until it is either cooled down or get oxidized. Figure 3.2 below shows a voxel life-cycle showing progression of states.

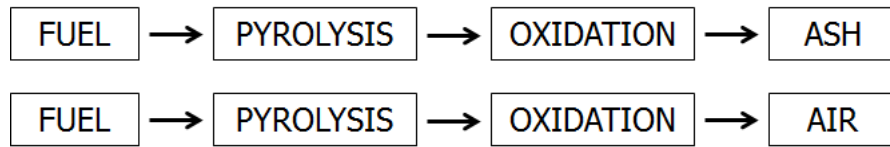


Figure 3.2: Voxel Lifecycle

Ash is not capable of igniting or sustaining chemical oxidation. It has very low thermal conductivity and high porosity. These properties effect smolder heat retention and propagation. All fuel eventually becomes ash on complete oxidation.

3.1.2 Voxel properties

Each voxel has multiple properties assigned to them: fuel, temperature, heat, material constants and signed distance we can see in figure 3.3. On complete oxidation, pyrolyzed fuel turns to ash. Smoldering is oxygen limited, so will discuss oxygen and

heat diffusion later in this chapter.

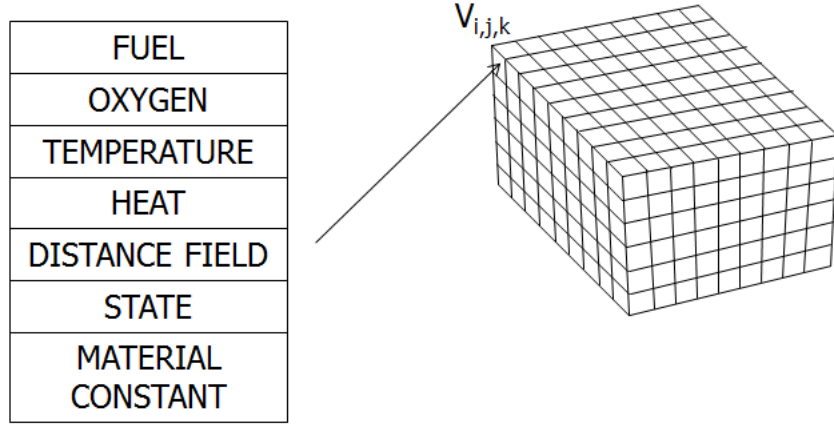


Figure 3.3: Voxel Properties

$$V_{i,j,k} = \{\text{Fuel, Oxygen, Signed Distance, Heat, Temperature, Material Constants, State}\}$$

Material constants store values for different constants like mass fraction, specific heat and diffusion constants for fuel and oxygen.

$$V_{i,j,k}^{MC} = \begin{cases} V_{i,j,k}^{MC_{fuel\ specific\ heat}}, V_{i,j,k}^{MC_{oxygen\ specific\ heat}}, \\ V_{i,j,k}^{MC_{fuel\ diffusion\ coefficient}}, V_{i,j,k}^{MC_{oxygen\ diffusion\ coefficient}}, \\ V_{i,j,k}^{MC_{fuel\ mass\ fraction}}, V_{i,j,k}^{MC_{oxygen\ mass\ fraction}} \end{cases}$$

3.2 Heat Transportation

3.2.1 Heat Transfer

For simulating the heat transfer model, we have heat transfer within the solid with heat diffusion that is discussed later. We also transfer heat from solid to air which we model using the equations below. As smoldering is a slow and low temperature there

will be very little heat that could be transferred using convection and radiation. We in this case ignore heat transferred by convection and radiation. We use direct neighbors to transfer heat from solid to air. In our case heat is generated using chemical heat so we have only heat loss to air.

$$V_{i,j,k}^S = \begin{cases} 1 & \text{if } i, j, k \text{ is air} \\ 0 & \text{if } i, j, k \text{ is fuel, ash, reaction zone} \end{cases}$$

$$\frac{d}{dt} V_{i,j,k}^{T_i} = V_{i+1,j,k}^S (V_{i+1,j,k}^T - V_{i,j,k}^T) + V_{i-1,j,k}^S (V_{i-1,j,k}^T - V_{i,j,k}^T)$$

$$\frac{d}{dt} V_{i,j,k}^{T_j} = V_{i,j+1,k}^S (V_{i,j+1,k}^T - V_{i,j,k}^T) + V_{i,j-1,k}^S (V_{i,j-1,k}^T - V_{i,j,k}^T)$$

$$\frac{d}{dt} V_{i,j,k}^{T_k} = V_{i,j,k+1}^S (V_{i,j,k+1}^T - V_{i,j,k}^T) + V_{i,j,k-1}^S (V_{i,j,k-1}^T - V_{i,j,k}^T)$$

$$\frac{d}{dt} V_{i,j,k}^T = k_{HE} \left(\frac{d}{dt} V_{i,j,k}^{T_i} + \frac{d}{dt} V_{i,j,k}^{T_j} + \frac{d}{dt} V_{i,j,k}^{T_k} \right)$$

3.2.2 Heat Diffusion

Heat diffusion can be modeled using the equation below. In smolder the rates of heat diffusion across voxels is different, as on oxidation the thermal properties of the voxels change. Ash is more insulating than fuel and will be slow to diffuse heat. k_{DT} represents thermal conductivity. In smolder as the ash has different thermal conductivity the equation below needs to be extended.

$$\frac{dT}{dt} = k_{DT} \nabla^2 T$$

In order to accommodate different thermal conductivity for per voxel, we use the equation below where k_{DT} is the constant that determines conductivity at a particular voxel. This helps us accommodate variable conductivity in ash and allows us to control heat diffusion from oxidation.

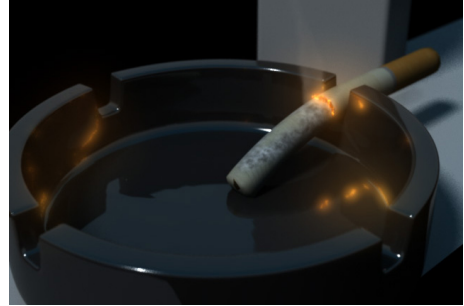
$$\frac{\delta}{\delta t}V^T = k_{Tx} \frac{\partial^2 V^T}{\partial x^2} + k_{Ty} \frac{\partial^2 V^T}{\partial y^2} + k_{Tz} \frac{\partial^2 V^T}{\partial z^2}$$

$$V_{i,j,k}^{k_T} = (k_{Tx}, k_{Ty}, k_{Tz})$$

Ash has lower conductivity than fuel and for this ends up insulating heat. $V_{i,j,k}^{k_T}$ is used to control thermal conductivity property of different voxels enabling us to re-regulate heat transfer across voxels and control direction of smolder. Zhu et al [64] discuss the ability of fuel to have different thermal conductivity but only model it for fuel.



(a) Cigarette



(b) Cigarette Simulation

Figure 3.4: Cigarette Ash

The direction of smolder can be controlled by altering the diffusion properties of ash. Different thermal conductivity is one of the factors for different shapes for the cones and ash formation. The figures 3.4 and 3.5 have different tobacco in cigar and cigarette, hence having variation with their ash formation and oxidation cones.



(a) Cigar



(b) Cigar Simulation

Figure 3.5: Cigar Burn Cone and Ash Formation

3.3 Oxygen Transportation

Oxygen transportation plays a very important part in smolder propagation. Smolder is low temperature combustion reaction and requires oxygen to produce heat. We here describe two ways in which smolder can receive oxygen and propagate. We break the transport into two parts. The first oxygen that can be received from the surface air and second oxygen that gets diffused from the surface to the oxidation front. Smolder is common in materials that are porous, where diffusion of oxygen is a real possibility. Ash is porous and allows diffused oxygen transport.

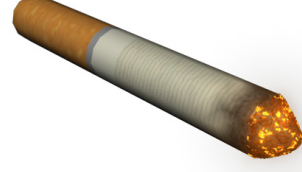
Different forms of Oxygen available to a voxel:

- Surface Oxygen Transfer
- Diffused

In figure 3.6 on the next page you can see the shape of oxidation front is dependent on the porosity of ash that is created when the cigar or a cigarette burns.



(a) Cigar Oxidation Cone



(b) Cigarette Oxidation Cone

Figure 3.6: Cigarette and Cigar Oxidation cones

3.3.1 Surface Oxygen Transfer

In order to model oxygen transport in our simulation we use similar approach like we do in heat transfer. Oxygen from the air voxels which would be the atmospheric oxygen is the source to any voxel that has air as a connecting neighbor. We use the same direct neighbor technique we used for heat transfer to transport oxygen to voxels that has direct connection to air. Any voxel connected as a direct neighbor has oxygen available from the air voxels.

$$V_{i,j,k}^S = \begin{cases} 1 & \text{if } i, j, k \text{ is air} \\ 0 & \text{if } i, j, k \text{ is fuel, ash, reaction zone} \end{cases}$$

$$\frac{d}{dt} V_{i,j,k}^{O_{surface}_i} = V_{i+1,j,k}^S (V_{i+1,j,k}^{O_{surface}} - V_{i,j,k}^{O_{surface}}) + V_{i-1,j,k}^S (V_{i-1,j,k}^{O_{surface}} - V_{i,j,k}^{O_{surface}})$$

$$\frac{d}{dt} V_{i,j,k}^{O_{surface}_j} = V_{i,j+1,k}^S (V_{i,j+1,k}^{O_{surface}} - V_{i,j,k}^{O_{surface}}) + V_{i,j-1,k}^S (V_{i,j-1,k}^{O_{surface}} - V_{i,j,k}^{O_{surface}})$$

$$\frac{d}{dt} V_{i,j,k}^{O_{surface}_k} = V_{i,j,k+1}^S (V_{i,j,k+1}^{O_{surface}} - V_{i,j,k}^{O_{surface}}) + V_{i,j,k-1}^S (V_{i,j,k-1}^{O_{surface}} - V_{i,j,k}^{O_{surface}})$$

$$\frac{d}{dt} V_{i,j,k}^{O_{surface}} = k_{O_{surface}} \left(\frac{d}{dt} V_{i,j,k}^{O_{surface}_i} + \frac{d}{dt} V_{i,j,k}^{O_{surface}_j} + \frac{d}{dt} V_{i,j,k}^{O_{surface}_k} \right)$$

$k_{O_{surface}}$ is the surface absorption coefficient. This allows the model to accommodate for different fuels by changing the ability of a fuel to absorb atmospheric oxygen on the surface in contact.

3.3.2 Oxygen Diffusion

The fundamental difference between smoldering and flaming combustion, in case of flaming the oxidation reaction and heat release is mostly surrounding the fuel. Smoldering is fundamentally in porous fuels so allows oxygen to diffuse through the ash to oxidation voxels to help in heat release. We use a similar approach that we use for heat diffusion and describe them in the equations below. To ease rendering we simplify our simulations to make fuel minimally porous, causing oxidation to occur at the ash/reaction zone boundary.

There have been several attempts to model diffusion in computer graphics by You et al[62], Morimoto et al [39], Huber et al [16], Lenaerts et al[28] and Stam and Fiume [59]. You et al, use diffusion to model paint mixing that is fluid to fluid diffusion. Morimoto et al [39], Lenaerts et al [28] and Huber et al [16] model textile tie dye, wet cloth simulation and porous flow, all being examples of solid and liquid diffusion. Stam and Fiume use diffusion blobs to model many physical phenomenon like smoke and steam.

The oxygen gets diffused in the fuel through ash as discussed by Buckmaster and Lozinski [11], and use Ficks second equation [57] to model oxygen diffusion. Here we model mass transfer where oxygen diffuses into ash and fuel to help in fuel oxidation. Jon Stam models real-time diffusion for games [58] and we model our gas diffusion on this approach.

$$\frac{\delta}{\delta t} V^{O_{diff}} = k_{Ox} \frac{\partial^2 V^{O_{diff}}}{\partial x^2} + k_{Oy} \frac{\partial^2 V^{O_{diff}}}{\partial y^2} + k_{Oz} \frac{\partial^2 V^{O_{diff}}}{\partial z^2}$$

Each voxel has controlling diffusion coefficient that help in modeling different material properties.

$$V_{i,j,k}^{k_O} = (k_{Ox}, k_{Oy}, k_{Oz})$$

We also define total oxygen in a voxel being the sum of surface oxygen transfer and diffusion. At the start of simulation the only oxygen available is from the surrounding air voxels. The diffused oxygen gets updated as the simulation goes on. Material properties for fuel and ash are different as ash being more porous allow more oxygen to diffuse through.

$$V_{i,j,k}^O = V_{i,j,k}^{O_{diff}} + V_{i,j,k}^{O_{surface}}$$

The amount of oxygen that is available for oxidation reaction is $V_{i,j,k}^O$.

In figure 3.7 we can see cones in a cigarette that were created with different diffusion and thermal conductivity parameters for ash.

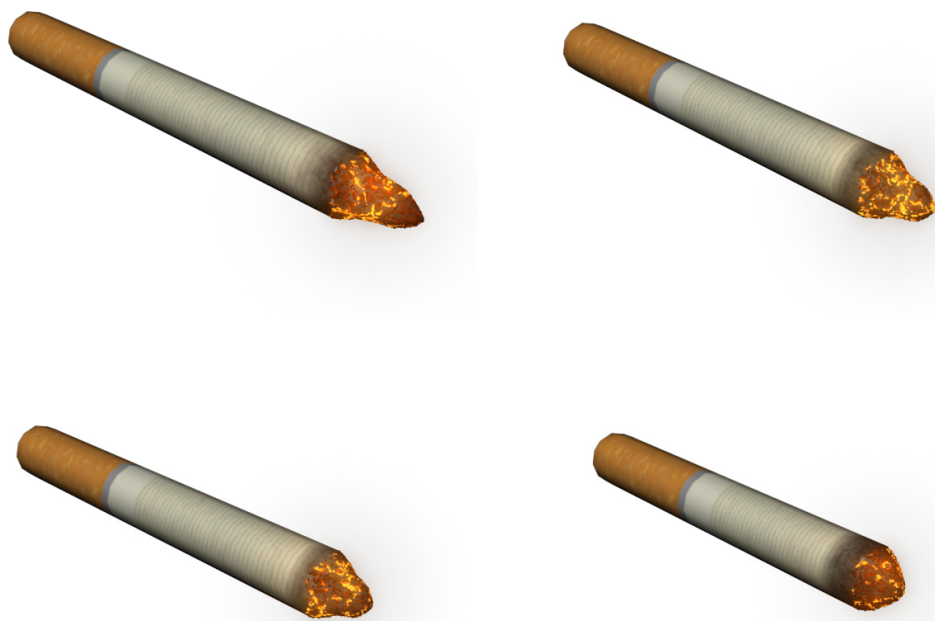


Figure 3.7: Oxidation front with ash diffusion parameters

Chapter 4

Reaction Zone

Reaction zone are the voxels that drive the smolder reaction by creating the heat that is necessary and is required to drive smolder, oxidizing the fuel, releasing chemical energy as the mechanism to maintain temperatures of the fuel in this zone, In this section we describe different parts of the reaction zone and the heat generating mechanism that allows smolder propagation. Rein describes the mechanism of smolder to be driven by heat and the process taking place in the reaction zone[51].

Pseudo code for the Heat simulation model is given in table 4.1 on the adjoining page.

4.1 Reaction Zone

We here introduce a concept of reaction zone to accommodate the slowness of smoldering process. Rein and Ohlemiller [51] [45] describe smoldering primarily being heat driven process and the energy exchange taking place in the reaction zone. This is the region where pyrolysis and oxidation take place to release heat. Fuel first undergoes the process of pyrolysis and then gets oxidized. As smolder is a low temperature, slow,

Table 4.1: Pseudo Code Heat Simulation

1.	Ignite keep heat source until fuel reaches T_p
2.	Calculate Heat Diffusion
3.	Calculate Surface Oxygen
4.	Calculate Diffused Oxygen
5.	Calculate Cumulative Oxygen
6.	if Voxel Temperature $> T_p$
7.	Use Oxygen Mass Conserve Equation to calculate Reaction Rate
8.	Use Reaction Rate and Reaction Equation to Calculate Fuel Used
9.	Use Reaction Rate, Fuel and Oxygen to Calculate Heat
10.	If (Voxel Fuel $< k_f$)
11.	Calculate Heat Released
12.	Update Distance Field/Level Set
13.	Else
14.	Calculate Oxygen based on Estimated Max Fuel
15.	Go to Step 7

oxygen deficient process, the pyrolysed fuel takes a little while to get oxidized. When the fuel gets pyrolysed it also gets preheated and can be oxidized as the next step.

In previous research for flaming combustion the temperature for pyrolysis is constant and provides a gateway to oxidation at the time of pyrolysis. In past research by Melek and Keyser [35] [36], Zhu et al [64] and Min and Metaxas [38], if a voxel entered pyrolysis state it would be oxidized with the assumption there is enough oxygen available would trigger oxidation. There have been attempts where the pyrolysis is variable [36] but even in these cases it forms the trigger for combustion oxidation reaction to take place. It is assumed that oxidation is only capable of happening on the surface of the fuel. In all these cases the source of rise in temperature is an external heat or a flame that releases heat. With all the research done by Ohlemiller [45] and Buckmaster [11] in this area, we assume that this is a two step process consisting of pyrolysis and then

oxidation. Pyrolysis and oxidation are two independent process and have to be treated independently.

Essential difference between flaming and smoldering is captured by breaking this process. If the fuel voxel is at a temperature to be oxidized with no oxygen available it will not participate in the smolder reaction as it will not be able to produce chemical heat.

Voxel falling in Reaction Zone are from two parts of the simulation as shown in figure 4.1:

- Pyrolysis
- Oxidation

$$V_{i,j,k}^{RZ} = \begin{cases} \text{oxidation} \\ \text{pyrolysis} \end{cases}$$

$$V_{i,j,k}^{RZ} = \sum_{i=1}^n V_{i,j,k}^{SO} + \sum_{i=1}^k V_{i,j,k}^{SP} \begin{cases} n = \text{oxidation} \\ k = \text{pyrolysis} \end{cases}$$

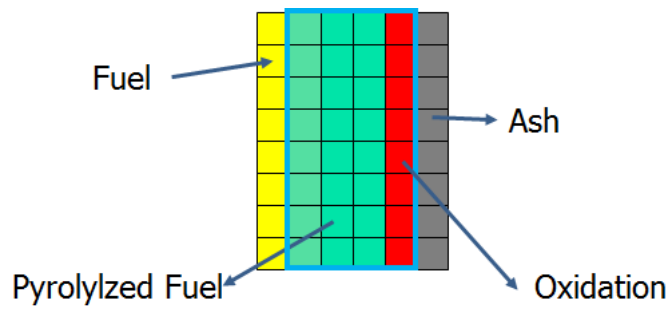


Figure 4.1: Pyrolysis and Chemical Oxidation

4.1.1 Pyrolysis

Pyrolysis is a process where the fuel heats up, loses moisture and is ready for chemical oxidation as discussed by Rein and Ohlemiller [51][45]. If fuel is below this

temperature, it is not capable of oxidation and will not react. The temperature at which pyrolysis starts is denoted by T_p .

If temperature falls below T_p the pyrolyzed voxel moves out of reaction zone and at that point is not capable of being oxidized. In this case the temperature is not sufficient for the voxel to be oxidized. This temperature can be lower or higher depending on type of fuel. More volatile fuels would have lower temperatures.[51]

$$V_{i,j,k}^T > T_p$$

If a voxel has a temperature value over the T_p it is considered to be pyrolyzed and it moves to the reaction zone. In figure 4.2 shows the two ends of the reaction zone as pyrolysis forms the entry and oxidation the exit. Collection of voxels undergoing pyrolysis and oxidation make up the reaction zone.

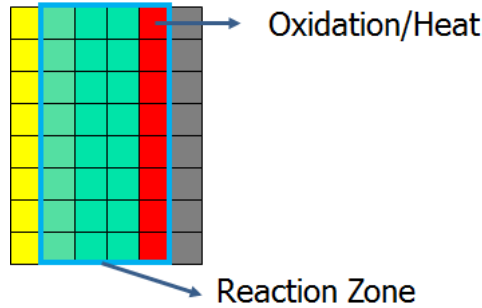


Figure 4.2: Reaction Zone and Oxidation Heat

4.1.2 Chemical Oxidation

Chemical oxidation is the most important part in smoldering as it is the energy source for the reaction. Heat generated during the oxidation process is what sustains the smolder. Fuel at pyrolysis temperature reacts with oxygen to produce carbon dioxide, heat and ash. The rate of heat generation is described by the equation below and is

dependent on fuel and oxygen. Oxidation forms one end of the reaction zone, where fuel gets consumed to generating heat.

Once a voxel starts oxidation it generates chemical heat that is dissipated to the neighboring voxels. The equation below shows the dissipation of chemical energy. Chemical energy released on oxidation will raise the temperature of all components involved in the chemical reaction plus heat transfers to any of the unreacted components.

$$ChemicalOxidationEnergy > Heat Loss$$

Heat gained in this reaction is due to oxidation, which is the only form of energy available in the simulation to sustain smolder and should be greater than all heat losses.

The diffused oxygen reacts with the fuel to generate heat at pyrolysis temperatures. The chemical heat generated is used to generate heat and helps raise the temperature of the fuel. Heat generated raises the temperature of the neighboring voxels. We describe the reaction using the equation below where Mass of oxygen and fuel determine the heat released that is used to rise the temperature and release heat to surrounding in the form of heat and light.

In the equations below we describe the process of smolder using conservation of energy and conservation of mass equations [11] [33]. We use the following energy equation that is based on the work done by Buckmaster and Lozinski [11] and Lu and Dong [33] on smoldering.

$$C_p \rho_s \frac{\partial T}{\partial t} + C_p (M_s - M_g) \frac{\partial T}{\partial x} = \lambda \nabla^2 T + Q \Omega + \frac{H}{d} (T - T_a)$$

In order to describe the energy conservation equation above we break it to

explain different energy transfer mechanisms.

$$C_p \rho_s \frac{\partial T}{\partial t}$$

This is the component of heat retention in the solid fuel. Where C_p is specific heat of the fuel and gas per Buckmaster and Lozinski [11] , ρ_s is density of solid, T is the temperature for both solid and gas and is at thermodynamic equilibrium.

$$C_p(M_s - M_g) \frac{\partial T}{\partial x}$$

This is the chemical heat generation part of the equation. As discussed by Buckmaster and Lozinski [11] we neglect the difference between the specific heat of solid and gas . M_s is mass flux of solid and M_g is of gas involved in smoldering.

We define M_o and M_f below as mass fraction of oxygen and fuel in the solid and gas, their relationship with fuel and oxygen in the simulation:

$$M_g = M_o + M_{g-o}$$

$$M_s = M_f + M_{s-f}$$

Where M_{g-o} is mass flux of parts in air other than oxygen and M_{s-f} is in-combustible mass flux part of the fuel undergoing combustion. Using the above set of equation we can break the equation up into heat generation and heat loss.

$$C_p(M_s - M_o) \frac{\delta T}{\delta x} - C_p M_{g-o} \frac{\delta T}{\delta x}$$

The first part of the equation is chemical oxidation heat generation and the second part is the heat loss due to unreacted gas flux.

$$\lambda \frac{d^2 T}{dx^2}$$

λ is the heat conductivity constant and is the conduction part of the energy equation.

$$\Omega Q$$

This component is the heat release part of the energy equation. Ω stands for the reaction rate from the Arrhenius equation. Q is heat release in the reaction.

$$\frac{H}{d}(T - T_a)$$

This component is the heat in the convective heat loss which is caused by airflow over the surface, where H represents the coefficient of convective heat transfer, and d represents the solid fuel thickness, T is temperature of fuel and oxygen and T_a is ambient temperature. This heat loss is introduced as the air flowing over the fuel has a cooling effect. This equation can be simplified by assuming that the coefficient of heat transfer correlates with the square root of the average velocity of airflow over surface, as described by Li and Dong [33].

$$\frac{H}{d} = K\sqrt{M_g}$$

Using the previous two equations we can derive the equation below.

$$K\sqrt{M_g}(T - T_a)$$

Here K is constant factor and the above term is the convective heat loss.

In the above described model there are a few assumptions we make with support of experimental data and previously done research. It neglects thermodynamic expansion effects of gas and assumes constant pressure in the reaction zone. The above equations do not take into account any momentum changes or state changes. Specific heat for solid and gas have been taken as one value. Per Buckmaster and Lozinski [11] and Lu and Dong [33] Simplification of the model by neglecting thermodynamical expansion and hydrodynamic effects is based on assumptions made from data using the

research mentioned by Lu and Dong[33] and experimental data collected by Matalon [34].

$$M_o = M_g \frac{\partial Y_o}{\partial x}$$

$$M_f = M_s \frac{\partial Y_f}{\partial x}$$

Where Y_o represents mass fraction of oxygen in gas and Y_f represents mass fraction of fuel in solid that is undergoing combustion. Using conservation of mass for fuel we get the equation below.

$$\rho_s \frac{\partial Y_f}{\partial t} - M_s \frac{\partial Y_f}{\partial x} = -\Omega$$

ρ_s and ρ_g are densities of gas and solid. Here the density is defined as mass per unit volume. Here we use conservation of mass to solid reactants and oxygen. μ is the stoichiometric constant to regulate amount of oxygen. Using conservation of mass for oxygen and using oxygen diffusion we get the equation below.

$$\rho_g \frac{\partial Y_o}{\partial t} + M_g \frac{\partial Y_o}{\partial x} = \rho_g D \nabla^2 Y_o - \mu \Omega$$

Rate law determining reaction rate using T_p and T_a ambient temperature, where rate is determined by the reacting fuel and oxygen.

$$\Omega = k_r Y_o Y_f e^{-\frac{T_p}{T}}$$

With the above information the above equation becomes :

$$C_p \rho_s \frac{\partial T}{\partial t} + C_p (M_f - M_o) \frac{\delta T}{\delta x} + C_p (M_{s-f} - M_{g-o}) \frac{\delta T}{\delta x} = \lambda \frac{d^2 T}{dx^2} + Q \Omega$$

As part of the reactant requirements stemming from the rate equation both $M_o > 0$ and $M_f > 0$ in order for the smolder reaction to proceed. $V_{i,j,k}^O$ is the oxygen available in the voxel $V_{i,j,k}$ and $V_{i,j,k}^F$ is the fuel available in the voxel $V_{i,j,k}$.

$$M_g = M_o + M_o V_{i,j,k}^{MC_{oxygen\ mass\ fraction}}$$

$$M_g = V_{i,j,k}^O + V_{i,j,k}^O V_{i,j,k}^{MC_{oxygen\ mass\ fraction}}$$

Table 4.2: Units for Constants Used

variable	unit
Y_o, Y_f	<i>dimensionless</i>
μ	<i>dimensionless</i>
k_r	$g/cm^3/sec$
v_g	cm/s
M_g, M_s, M_f, M_o	g/cm^2
C_p	$cal/g/K$
T	K
λ	$cal/cm/s/K$
ρ, ρ_s, ρ_g	g/cm^3
q	W/m^2
Q	cal/g
D	cm^2/sec

$$M_s = M_f + M_f V_{i,j,k}^{MC_{fuel\ mass\ fraction}}$$

$$M_s = V_{i,j,k}^F + V_{i,j,k}^F V_{i,j,k}^{MC_{fuel\ mass\ fraction}}$$

Quantity of fuel can result in incomplete combustion which can have an adverse effect on heat generation. In order to completely use up the fuel at a time-step a little extra oxygen is required. If there is too much air it results in excessive heat loss as the Nitrogen does not get oxidized and only ends up adding to heat loss.

$$dH = C_p \rho_s \frac{\partial T}{\partial t} + C_p (M_s - M_o) \frac{\delta T}{\delta x} - C_p M_{g-o} \frac{\delta T}{\delta x}$$

Heat retained by the system is given by the above equation. First term is the chemical heat generated by the system and the second is the heat lost due to heating unusable portion of air.

In table 4.2 on the next page we show the units for all the constants used in the model described.

4.1.3 Reaction Fuel Limit

Fuel is required for the chemical reaction to take place and the fuel quantity for a time step increases with the availability of oxygen. There is a limit on how much air can be contained in a given volume. Experimental data indicates requirement of stoichiometric ratio to enable full combustion oxidation as discussed by Buckmaster[11].

Fuel and oxygen need to be mixed before they can actually react and the mixing of fuel to enable oxidation could take some time. In order to accommodate this mixing we limit the amount of fuel fraction that is available to react per time-step.

$$\frac{dF}{dt} < k_f$$

where k_f is the maximum amount of fuel that can be consumed in a time step.

When we hit this limit in a voxel oxygen, usage is re-calculated using the rate equation using this cap value. The energy equation is re-run to calculate heat loss with the new oxygen value. Even though there is excess oxygen available k_f will be the maximum amount of fuel available at any time-step.

4.1.4 Pyrolysis propagation

Any voxel entering pyrolysis becomes part of the reaction zone as described before. During pyrolysis there are changes to the fuel where it loses moisture and gets ready to be oxidized. We track this surface as we render, as color changes to the fuel. There is loss in fuel as it is not getting oxidized. In order to show this change the color of the fuel changes. Once color changes it only adds to the change and is irreversible.

In order to track the surface we employ a modified spherical meatball model using the oxidation front as the starting point as done by Ilic et al [18]. Our purpose of tracking the Pyrolysis propagation is to be able to render its effect on the reaction zone

to show how the fuel reacted to the change.

As described earlier pyrolysis forms one end of the reaction zone and as we are modeling smolder both sides of the reaction zone play a vital role to the final render.

In order to track the pyrolysis we create spherical meatballs to the centroid of each triangle in the triangulated oxidation front. This is described in more detail in chapter VII.

4.2 Reaction Front propagation

4.2.1 Oxidation Front propagation

We in our simulation need to track the reaction zone. Reaction zone has a set of moving interface for oxidation. Level set is a widely adopted approach and we use it to track surfaces using distance field see figure 4.3. Melek and Keyser [35], Zhu et al[64] have used this in order to model decomposition. There have been attempts by others such as Zhao et al [63] representing fronts using an enhanced distance field multi-band model. They use it to track a volumetric dataset. Liu et al [29] propose a method to re-input fire energy to alter temperature fields and effect the level-set. As a starting point to calculate the distance field from a polygonal mesh we use Batty’s library [7].

The oxidation and ash boundary can be defined as below.

$$\phi_t + V \cdot \nabla \phi = 0$$

In order to track the oxidation surface we use level set method similar to Malek and Keyser [35] and Zhu [64] . As smolder moves slowly we believe that it will travel in the direction of the surface normal and will be dependent on fuel consumed. As fuel consumption is dependent on oxygen usage and following that we use this energy release

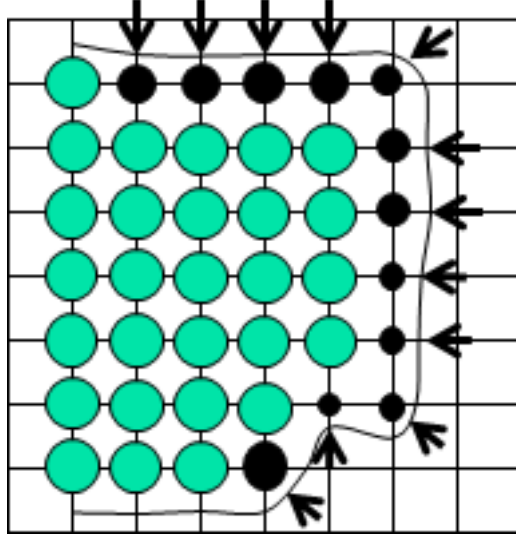


Figure 4.3: Level Set using normals

equation to model the velocity.

$$q = v_s \rho C_p \Delta T$$

Where ρ is the combined density of both solid and gas which is described in the equation below.

$$\rho = \rho_s + \rho_g$$

typically ρ_s/ρ_g is around 30 as shown by Buckmaster and Lozinski [11] and we can approximate ρ . Melek and Keyser [35], [36] in order to find moving boundary used a vector field to define the burning solid deformation. Zhu et al [64] use a normal vector field to calculate the moving boundary. We extend their approach where in, we combine heat conservation equation [64] with [11] Buckmaster and Lozinski's work to calculate velocity of smolder movement.

$$q = v_s \rho_s C_p \Delta T$$

In order to calculate the velocity of smolder movement we combine the chemical

heat generation flux equation with the above equation in conjunction with heat laws.

$$v_s = \frac{(C_p(M_s - M_o)\frac{\delta T}{\delta x} - C_p M_{g-o}\frac{\delta T}{\delta x})}{\rho_s \Delta H dt}$$

v_s represent speed of smolder. Keeping it in line with other work done in this area [64] the speed of travel is normal to the direction of the front.

$$Vel = -N(\vec{x})kv_s$$

Vel is the velocity of the level set.

$$Vel = -N(\vec{x})k\frac{(C_p(M_s - M_o)\frac{\delta T}{\delta x} - C_p M_{g-o}\frac{\delta T}{\delta x})}{\rho_s \Delta H dt}$$

k here is a constant that can control degree of decomposition. As described by Zhu [64] the movement of the surface can be calculated in the normal direction, the direction can be given by the equation below.

$$N(\vec{x}) = \frac{\nabla\phi(\vec{x})}{|\nabla\phi(\vec{x})|}$$

We reconstruct the polygonal mesh ash and reaction surface using distance field and then use marching cubes to reconstruct the polygons using only the voxels that have been modified in the reaction zone. There is a mesh created at every time-step that only accounts for changes made in the reaction zone and reconstruct the part of mesh corresponding to the zone and then update the distance fields and rest of the voxels are left unchanged. [9] propose a technique to model smoothing of level-set mesh using animated particles. Their technique could be used as an extension to smoothen the above created mesh.

4.2.2 Oxidation Front Detection

For each frame we need to detect the surface that comprises of the oxidation front. As discussed before, marching tetrahedron is used to determine the oxidation

surface deformation from distance field. Zhu et al [64] use refined marching cube algorithm to determine change at every time-step. With view to determine current oxidation surface we use a similar algorithm that calculates the part of mesh that changes. As they propose modification of only the effected distance fields and using only the changed voxels.

The following pseudo-code shows the algorithm:

Table 4.3: Pseudo Code - Oxidation Front Generation

1.	Current time is t, Create 4 lists S,M,T,C, S - Triangle list from t-1 time-step M - Triangle List from t time-step, T - New Triangles created in t time-step, C - Triangles created due to Reaction Zone
2.	if (S is empty) Create a list of triangles using the starting mesh and store in S
3.	Get all Voxels in the reaction zone
4.	Use distance field to create mesh M
5.	Compare mesh M with S and get list of updated triangles. Store in T
6.	for every triangle on M do
7.	if triangle is created by a voxel in the reaction zone add to C
8.	end for
9.	end do
10.	combine all triangles in C and make a Mesh
11.	New mesh is the oxidation front mesh for current time step
12.	Save mesh M in memory as S

4.3 Forward and Reverse Smolder

Smoldering can be classified into two main configurations on the basis of the direction of airflow and the direction of smolder movement:

- Forward Smolder
- Reverse Smolder

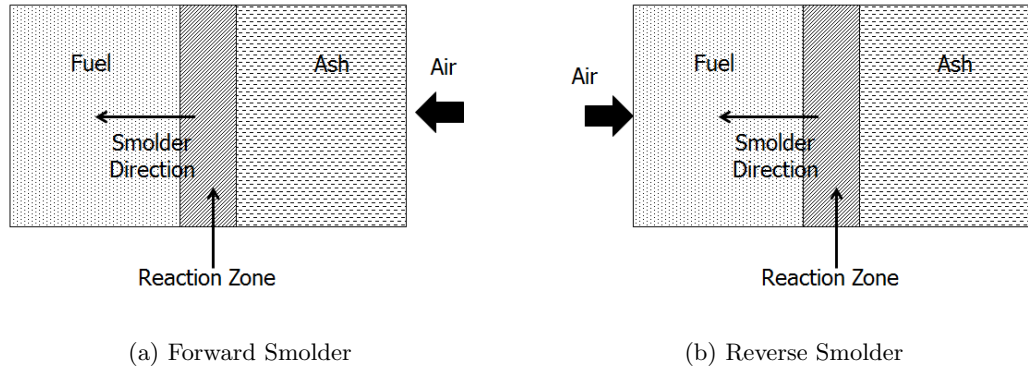


Figure 4.4: Fuel showing forward and reverse smolder

These phenomenon are studied in great detail by Rein in foam [52], cigarette [51] and by Ohlemiller and Lucca [46] in fuel beds. We generalize the configurations to show generic fuel in figure 4.4 from the work of Rein, and Ohlemiller and Lucca. As described by Rein [52] [51], Ohlemiller and Lucca [46], if the smolder reaction propagates in the same direction to the direction of air flow, is called forward smolder. If the direction of air flow and direction of smolder movement is opposite it is called reverse smolder. These do play an import part in smoldering propagation.

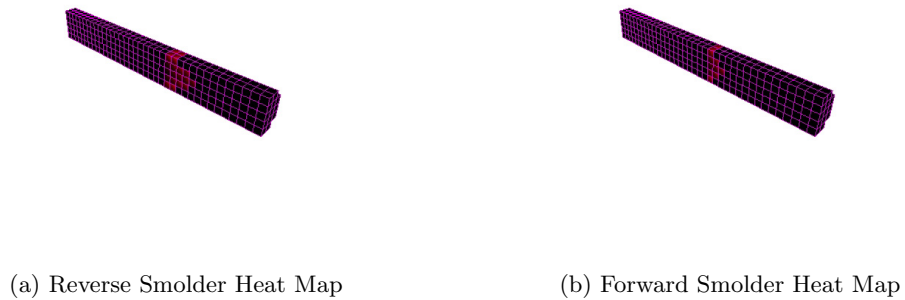


Figure 4.5: Cigarette Cutout for Voxel arrangement showing forward and reverse smolder

A cigarette draw would be considered as forward smolder where the direction of airflow is in the same direction of smolder movement. Later on in this dissertation we will show example of cigarette draw. Figure 4.5 on previous page shows the heat release for the different smoldering processes.

4.4 Ash

In smoldering thermal properties of ash play an important part in providing insulation and reducing heat loss when chemical heat is generated at the oxidation front. Fuel voxels get oxidized to produce ash and heat. We track the oxidation front that is introduced in a previous section. There have been several attempts to model secondary deformation under thermal stress. One such attempt is to use proxy object around the object under combustion by Melek and Keyser [37]. In their attempt the creation of the proxy object has no connection to the physical simulation driving the animation. Liu et al [30] models deformation of thin shells on combustion by mapping thermal stresses on vertex and face of the thin shell mesh. There also have been attempts by Larboulette et al [26] and Jeong et al [21] to model paper and cloth using spring systems. We propose a simple model for ash deformation based mass-spring mesh in a later chapter.

Ash is at a lower temperature than the oxidation front. After formation of ash it forms an insulation layer around the oxidation front. This layer prevents most heat losses during smoldering, then during flaming combustion and we think that thermal stresses do not play a significant part in ash deformation. To model ash we propose a simple model based on deformation under gravitational forces over time. This model is explained in a later section.

Chapter 5

Smolder Propagation

Here we discuss self propagating smolder where oxygen diffusion is the main source of oxygen and is responsible for maintaining smolder. We also consider factors such as airflow that help in either accelerating, converting to flaming or extinguishing smolder movement.

Smolder can propagate either on its own or by help of an external agent. Airflow is one of the agents that help in increasing reaction rate that accelerates the oxidation reaction. Under certain circumstances this acceleration could lead to flaming or putting out smolder. We also discuss the ability of smolder to propagate on its own in the self sustained section. We also discuss the ability of smolder to flame and extinguishing.

5.1 Self sustained smolder

This form of smolder is where the oxygen available by diffusion is enough for generating heat that can sustain the smolder reaction. This type of smolder is slow. A lighted cigarette with no interaction could be considered a self sustaining smolder. Simulation results to actual cigarette are shown in figure 5.1.



(a) Cigarette



(b) Cigarette Simulation

Figure 5.1: Self Sustained Smolder

Self sustained smolder is slow to travel as the energy released during chemical oxidation is trapped due to the insulation provided by ash. The path will also depend on a few other factors like the porosity and thermal properties of fuel, ash to name a few. We can see the shape of the oxidation front cone different for a cigarette and cigar due to some of the mentioned property differences.

5.1.1 Assisted smolder

Smolder can be accelerated by adding an additional oxygen supply. This oxygen could be in the form of airflow. This airflow accelerates the smoldering reaction where chemical heat is gained using the energy equation. We use the equation below to increase the chemical heat when there is excess airflow. Kingman and Palmer [25], Buckmaster and Lozinski [11], Lu and Dong[33] discuss elementary and experimentally the effect of airflow on smoldering. The equations below are derived from their work.

Lee et al [27] use a techniques where they add wind velocity on a grid. We use a similar technique where we add density to create a density map and velocity to a voxel to control the airflow supplied to the simulation.

The mass flux of gas M_g is increased using the equation below. We can see this

in figure 5.4 where heat loss behavior determines outcome of the assistance provided to smolder.

$$M_g = v_g \rho_g$$

v_g is velocity of gas and ρ_g is density of gas. As the amount of M_g increases the chemical heat component releases more chemical heat on oxidation.

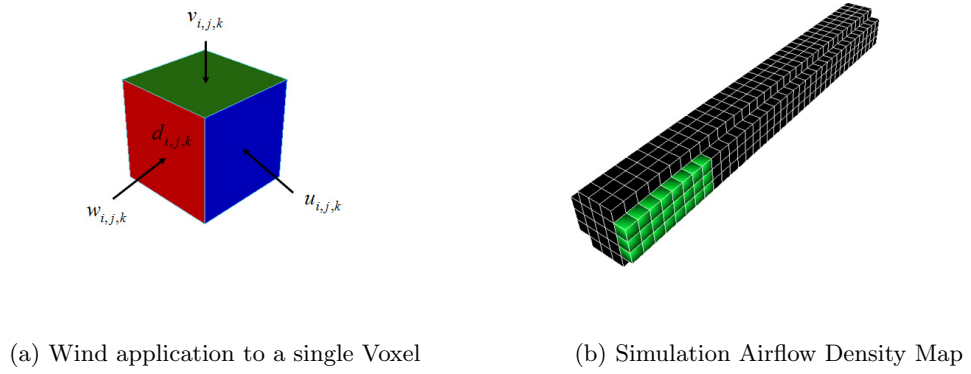


Figure 5.2: Assisted Smolder Airflow parameters

In the simulation each voxel can be a source of airflow and the possibility of having a velocity and density element. This helps to determine the airflow that the simulation will receive. This helps to calculate M_g . We have shown direction and density component in figure 5.2 where 5.2a shows a single voxel with direction of velocity and density of airflow supplied to the voxel, 5.2b shows the density map showing the density of airflow being added per voxel. In the figure the amount of airflow per voxel is the same. The two parameters allow us to determine mass flux of gas to enable calculation of heat release.

We can also assume that air flow per voxel $Vel_{ijk}(t) = (u_i, v_j, w_k)$ is the velocity vector at $Voxel_{ijk}$. In continuation of the previous equation we can calculate airflow at $Voxel_{ijk}$, which can be expressed as $Air_{ijk}(t) = d_{ijk}Vel_{ijk}(t)$ for any voxel.

The effect of airflow on smolder has been a major area of research in the mechanical engineering community as it results in subsequent phenomenon of flaming and smolder extinguishing.

In figure 5.3b we can see the effect of 5.2b density map. There is extra heat released during the oxidation process and also increases fuel usage on the side where the airflow is applied. The rendered results are shown in 5.3b as the cone created by the oxidation reaction gets accelerated in one direction hence skewing the cone.

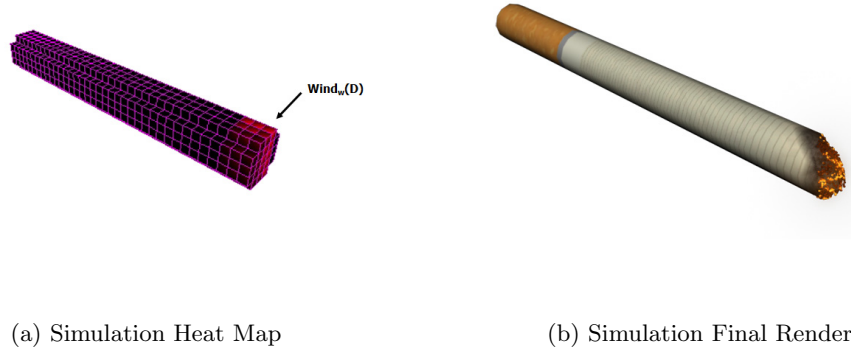


Figure 5.3: Assisted Smolder

Chemical heat in the equation below can be seen in the heat map shown in 5.3a. This heat helps in maintaining the heat energy balance in smolder and in some case helps to accelerate the smolder reaction.

$$Chemical\ Heat\ Gain = C_p(M_s - M_o) \frac{\delta T}{\delta x}$$

Airflow plays an important part in smolder propagation as it acts as a source of oxygen to increase rate of reaction. Ohlemiller [43] discusses the effect of increased air flow on cellulosic insulation and its relationship with flaming and impulsive heat release.

We model airflow to have two effects here; one it alters the oxygen to the chemical oxidation front 5.3a and second to have a thermal effect on fuel. If the heat generated

is high initially it will accelerate the smolder propagation and if the heat released cannot be dissipated quickly causing spontaneous exponential rise in temperature, it will make the fuel to flame.

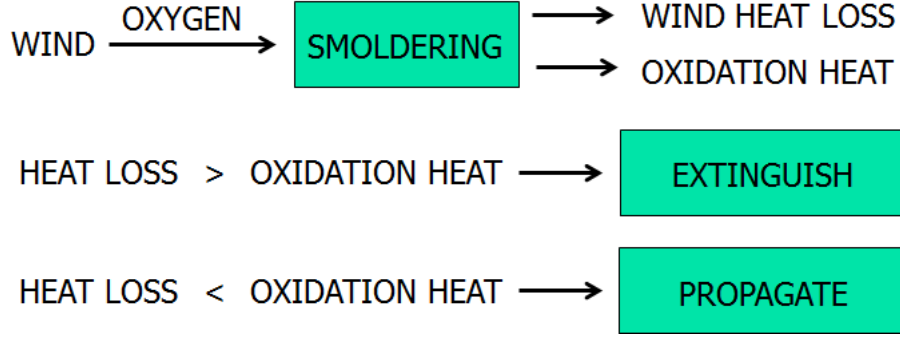


Figure 5.4: Wind Mechanics

5.2 Transition to Flaming

We here consider the ability of a material to transition from smolder to flaming under some of the above factors. Flaming is an important phenomena where a fuel under the right conditions with excess heat buildup bursts into flames. This is also a major hazard and a cause of concern for fire safety.

Aldushin et al [1] discuss the relationship between flaming and impulsive heat release. We here study the mechanism and different factors that would trigger flaming. We also propose a model where a smolder reaction would transition to flaming. When smolder is under increased oxygen the heat released becomes faster and the fuel will become hotter. With this extensive release of heat and requiring in a very short time to dissipate can causes flames to erupt [1].

$$C_p(M_s - M_o)\frac{\delta T}{\delta x} > H_{flaming}$$

As oxidation is the main control parameter during smoldering due to increased airflow rate of heat generation increases. The above equation forms the basis of initiation to flaming. Figure 5.4 shows multiple ways in which smolder gets effected due to application of wind.

5.2.1 Accelerated Smolder Propagation

Air flow has multiple effects on smolder. In flaming combustion the flame will help in heat transfer to the fuel in the direction of wind. There have been multiple attempts to model wind with flames using particles and on meshes. In smolder there is no flame so heat transfer is not associated with it. In smolder air flow has an oxidation source that accelerates the oxidation of fuel. Oxidation leads to generating more heat and accelerate smolder propagation.

Direction of airflow determines how the oxidation of the fuel is going to take place. As smolder material are porous air flow passing through fuel will be warmer when it reaches the oxidation zone. There will be less heat losses and more heat generated. In case of wind not passing through, fuel will be cooler and hence more heat loss, less heat generated. Smolder speed will be depending on the direction of wind. The direction of air flow forms the basis of forward and reverse smolder

5.2.2 Smolder Extinguishing

The flow of oxygen can enhance or reduce the smolder process depending on how much oxygen is supplied to the smolder process. This is also influenced by the amount of convective heat loss that happens during the process. We here discuss the energy equation terms that produce heat loss and can cause smolder to die down.

If the heat losses exceed the heat release the reaction zone temperature will

fall below T_p and will stop smolder propagation.

$$V_{i,j,k}^{T_{total}} < T_p$$

In cases where the temperature of the reaction zone falls below T_p there are two parts of the energy equation that cause the loss. There is energy required to heat the unused gas in the oxidation reaction.

$$Heat\ Loss - Unreacted\ Gas\ Flux = C_p M_{g-o} \frac{\delta T}{\delta x}$$

Convectional heat loss is proportional to the gas mass flux. As the gas flux increases with velocity of airflow and so will heat loss do to convection.

$$Heat\ Loss - Convection = K \sqrt{M_g} (T - T_o)$$

When the gas velocity increases the above two parts of the energy equation form the heat loss and could lead to lowering of the temperature of the voxel to below T_p causing smolder to extinguish.

5.2.3 Heat Insulation and Flaming

Heat in the simulation is either dissipated or is used to propagate smolder. If the heat is not allowed to escape it will cause the temperature of fuel and reaction zone to rise. A cigarette for example will not flame if the heat is being dissipated. Either more oxygen must be added, the temperature has to be raised or the heat produced by the cigarette must not be dissipated. We can create an example where a cigarette can be dropped onto a sofa. This could reduce the dissipation of the heat from the cigarette and allow the smoldering to flame. We can control the insulation level on the oxidation zone to cause a heat build up in the reaction zone. In this case the heat is getting insulated and that causes it to start flaming without the wind oxidation effect.

Chapter 6

Secondary Deformation

Smolder is a low temperature process and the solid object undergoes physical deformation but as the temperature is lower than flaming combustion the deformation due to thermal changes is limited. In this section we describe the deformation process with how the simulation effects the deformation at these low temperatures.

In quest for making simulation more realistic there have been multiple attempts to deform objects using internal physical reactions as the driving process. Malek and Keyser [37] proposed a method where they create a proxy object and map simulation properties to that proxy using Free Form Deformation(FFD), byacentric coordinates and a mass-spring system.They are limited to one proxy per object and is created by the animator. The proxy object, is then modified by the simulation and the deformation is passed to the actual burning object. Malek and Keyser do not account for the possibility of having multiple independent physical reactions and the use of multiple proxies. Defining of the proxy object as it is not being part of the simulation is some manual ask for the animator. If an animator is not familiar with their system it will require some doing to set the proxy up.

This problem is addressed by Larboulette et al. [26] and Liu et al. [30]. In both the frameworks the deformation is part of the simulation and is represented as a unified framework. The simulation parameters are directly used to drive deformation. There is no concept of proxy objects and deformation is created automatically as part of the simulation. Their work only account thin shells thus their models cannot be extended to normal three dimensional objects.

Melek and Keyser [37] and Liu et al.[30] use FFD as the underlying method to determine structural changes and decomposition. Jeong et al [21] and Larboulette et al [26] use a spring mass system to model crumpling of paper and cloth. They have a two dimensional system that cannot be extended to help model the deformation we require.

We propose a method based on voxelized space created during the physically based simulation, with a combination of marching cubes [31] and a tetrahedral based mass spring mesh to create deformed marching cubes. This offers a simulation integrated deformation model in three dimension with no requirement for external proxy objects. Melek and Keyser’s [37] process has a key component where the proxy creation is the key and deformation lattice being associated to a cosine function to allow generation of crumpling and deformation effects. We simplify all this with the object shape creating the proxy automatically, limiting parameters required for creating deformation and then having the simulation drive the deformation directly. A voxel based mesh is generated from the simulation system to create a volumetric spring mass system. The mass-spring system is deformed under the influence of forces that are derived from the simulation presented earlier. The forces causing deformation could be thermal or gravitational. Basic deformation of solid primitives are described by Barr [4]. VoxMorph by Firaj et al [15] is a voxel grid based cage deformation tool. We use a voxel based technique but using a version of a modified marching cube algorithm.

6.1 Mass-Spring Mesh

Spring conventionality have been used to model deformation in 2D for thin shells and cloth in the work done by Liu et al [30]. We extend the use of mass-spring meshes based on a voxel configuration to model deformation of two and three dimensional objects. 3D data is volume based and there may be information inside the object verses the surface. We when creating the mass-spring system create a three dimensional structural re-presentation of the object. This kind of approach was used to model muscles by Chen et al [12]. They propose a voxel based mass spring system to model muscle movement and display a real time animation mesh. We base our approach on a similar method. We explain below the force equation governing out spring mesh system. Diziol et al use [14] a tetrahedral based method to solve the volume conservation problem. Jeong et al [21] and Larboulette et al [26] use a spring mass system but use it for two dimensional objects.

$$M \frac{\partial^2 \vec{x}}{\partial t^2} + C \frac{\partial \vec{x}}{\partial t} + K(\vec{x}) = \vec{f}(\vec{x}, t)$$

We use the above equation to solve the spring system . M represents mass matrix from the simulation, C represents damping matrix and $K(\vec{x})$ are internal forces and $\vec{f}(\vec{x}, t)$ are external forces like gravity and thermal forces that we explain in a later section. We use the work done Barbic et al, using the Vega framework [3] that almost preserves volume and more details can be found in their paper.

6.2 Tetrahedral Mass Spring Voxel

We here define a spring voxel which form the basic building block for our mass-spring system. In order to create the deformable volumetric system we use the

tetrahedral mesh. We use work done by Hang Si [56] mesh created to help in creation of the tetrahedral mesh structure from a voxel mesh system.

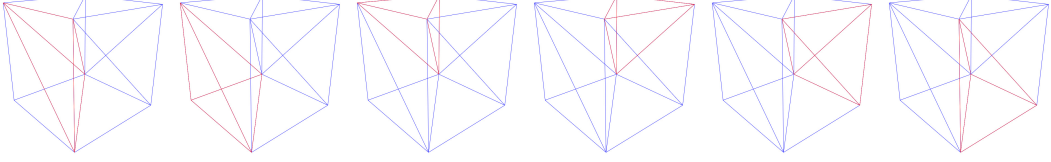


Figure 6.1: Tetrahedrons in a Spring Voxel

We create a tetrahedral mesh using the voxel so we can base our volumetric structure on it. Each edge of the voxel cube forms a spring and the corner vertex have mass particles attached. During the process of breaking down the voxel we create six small tetrahedrons. Figure 6.1 shows the six tetrahedrons. The total amount of fuel mass from the simulation is mapped to particles on the corners of this tetrahedral voxel mesh. Each of these masses are connected to their neighboring masses by edges that end up being springs. We use tetrahedral meshes as they allow better structural integrity and conservation of volume.

As we go on to discuss the Spring voxel, it consists of three type of springs: Edge, Face and Body. There are twelve edges, six face and one body spring. The randomness of body springs lets us maintain the random organic structure. Figure 6.2a shows the arrangement of how we connect the particle masses in a spring voxel.

Figure 6.3 shows the arrangement of spring on a simple sheet. We can see edge, body and face springs colored in orange.

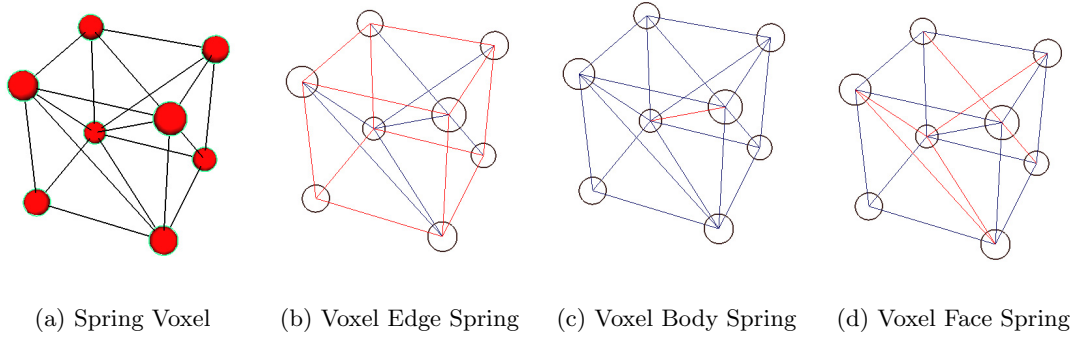


Figure 6.2: Spring Types

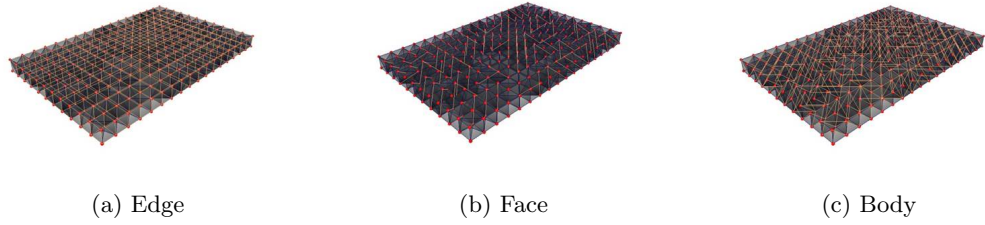


Figure 6.3: Spring arrangement on a Sheet

6.2.1 Automatic Voxel Based Spring Mesh Generation

Generating a proxy object has always been a manual task. Malek and Keyser [37] have a technique where the proxy object needs to be created manually outside their simulation. In order to make it easy for the animator and reduce his work, we in our simulation want to generate a mass-spring system tied to the simulation automatically where the voxel properties can drive the force system on the mass-spring mesh voxel.



Figure 6.4: Automatic Spring Mesh Generation Steps

First step to this process is to be able to break the object into voxelized space. For our purpose we are able to use a much simpler intuitive technique where each spring

voxel is mapped to a simulation voxel. The algorithm and steps to create the mesh automatically are described in Table 6.1 and figure 6.4.

Table 6.1: Pseudo Code - Spring Mesh Auto Generation

1.	Input Mesh
2.	Calculate Distance Field with grid using pre-defined Voxel Dimensions
3.	Voxelize Mesh using Distance Field and record distance field in each voxel
4.	Create Complete voxel List C
5.	for each voxel tetrahedralize creating body, face and edge springs
6.	Calculate Mass and position of each particle mass calculated by equally dividing the simulation voxel mass to particles All fractional masses added to get total mass of vertex particle Position of spring mesh masses using corners of the simulation voxel
7.	if ($\text{voxel} \in C$)
8.	Redo process
9.	Update and initialize the spring mesh system with the masses and spring stiffness.

Once the spring mesh is ready it gets automatically connected to the simulation.

The connection criteria is described in the next section.

6.2.2 Connecting Simulation Voxels To Spring Voxels

In order to control secondary deformation each voxel connects to a spring voxel cube. Voxels in the simulation as discussed earlier also form the basis of the marching cubes [31] used to generate the surface from the distance field value. We use [10] implementation of Marching Cubes as a starting point. Figure 6.5 shows the communication mechanism between different parts of the simulation. Simulation voxel is connected to the spring voxel where it transfers information necessary to calculate deformation forces. Primary exchange from simulation to springs is transferring of material and thermal properties.

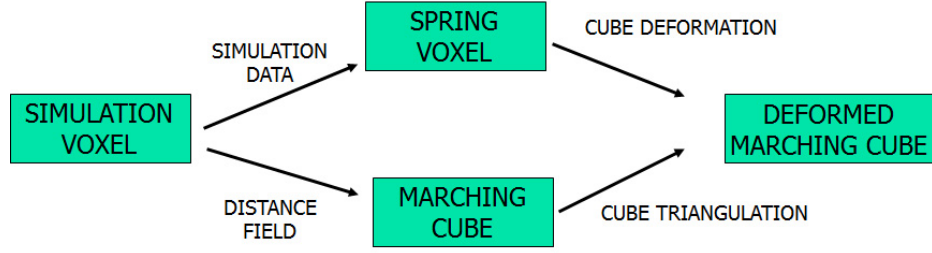
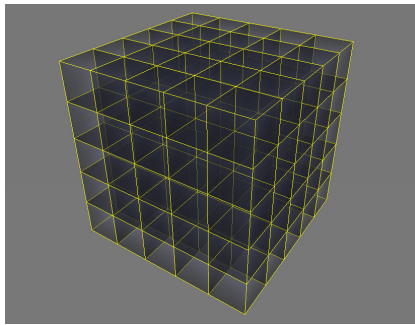


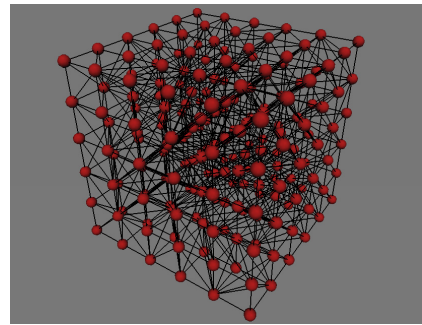
Figure 6.5: Data Exchange To Deform

6.2.3 Deformed Marching Cube

Our simulation is connected to the deformation spring mesh system at the voxel level. Each simulation voxel connects to its corresponding spring voxel in the mass spring system, when we recreate the output polygonal mesh from the simulation with marching cubes. Figure 6.5 shows how the exchange occurs. The simulation drives the decomposition using marching cube and the deformation using the spring voxel mesh. As the triangulation is calculated while processing the marching cubes, a new value for each cube is interpolated using the corresponding spring mesh cube. This is described later in this section.



(a) Simulation Voxels



(b) Spring Mass Voxels

Figure 6.6: Simulation Voxel Mapping to Spring Voxel

We in figure 6.6 show the voxel system being connected to the spring mesh. The voxel system coordinates form the basis of the cube structure when we calculate

marching cubes to reconstruct our surface mesh. Next on calculating the triangle positioning on the triangulated cubes new interpolated values from the spring voxels are used to get new positioning. Intersection on the edges of marching cube move to accommodate the deformation but the triangles formed during the process are maintained. This allows us to maintain the same mesh structure but deformed position. We see the algorithm at work in figure 6.7 where the same marching cube mesh is transposed using the spring voxel mesh.

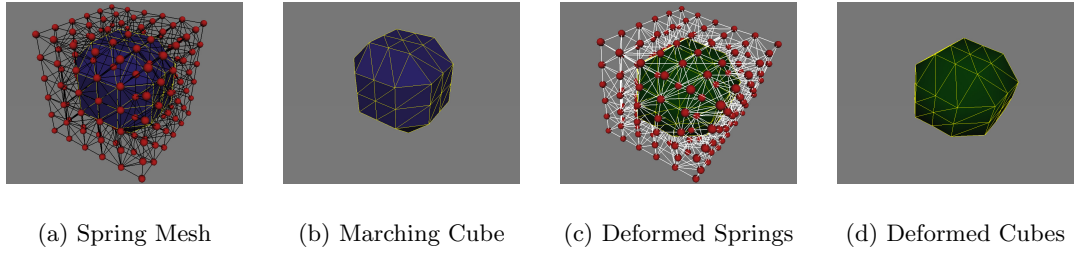


Figure 6.7: Spring arrangement on a Sheet

In the equation below if P_1 and P_2 are the vertices of a cut edge on the marching cube and V_1 and V_2 are the scalar value at each vertex.

$$P = P_1 - \frac{(V_1)(P_2 - P_1)}{(V_2 - V_1)}$$

Now on deformation the points on the edge corresponding to P_1 and P_2 are mapped to the edge SM_{P_1} and SM_{P_2} on the spring mesh and the above equation gets modified to the following:

$$P_1 \rightarrow SM_{P_1}$$

$$P_2 \rightarrow SM_{P_2}$$

With the above translation the new intersection point P_{new} of the deformed cube would be :

$$P_{new} = SM_{P_1} - \frac{(V_1)(SM_{P_2} - SM_{P_1})}{(V_2 - V_1)}$$

The results of the above equation are depicted in figure 6.7d.

Figure 6.8 shows the deformed marching cube algorithm at work when we calculate the new position on the mesh during the deformation of cigarette ash.

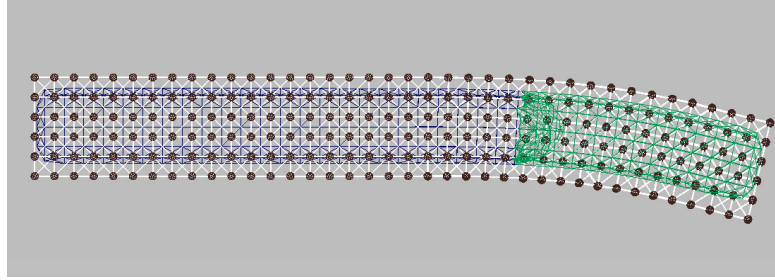


Figure 6.8: Cigarette Spring Mesh Showing Ash Deformation

6.3 Controlling Deformation

In this section we describe the forces that come into play to deform when smoldering takes place. Two major ones are thermal and gravitational forces. We have two examples of this type of deformation. Gravity based deformation is exemplified by the ash deforming when a cigarette burns for a long time, as seen in figure 6.8. The second would be dominant thermal based, an example of which would be a sheet of material bending under thermal stress due to heat produced because of oxidation or a matchstick buckling under thermal stress in figure 6.9.

6.3.1 Temperature Based Deformation

External force applied to the spring mesh system using Hook's law can be in the form of thermal forces defined as F_t and the extension to the spring as x with the

spring constant being k .

$$F_T = kx$$

In our simulation with the increase in temperature in a voxel the spring will deform and will increase in length by dx . This is proportionate to the temperature increase. Higher temperatures cause more expansion as we would expect. L_o is the rest length of the spring, which in this case would be the grid size. T is the temperature of the attached voxel and T_a is the ambient temperature.

$$dx = \tau L_o(T - T_a)$$



(a) Simulation



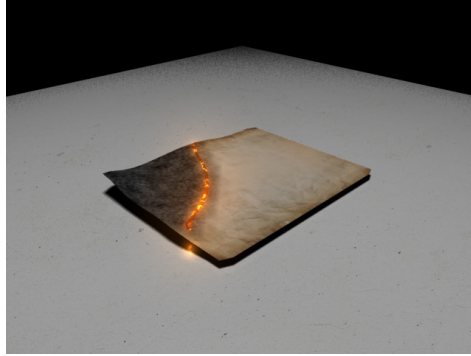
(b) Actual

Figure 6.9: Matchstick Simulation Comparison

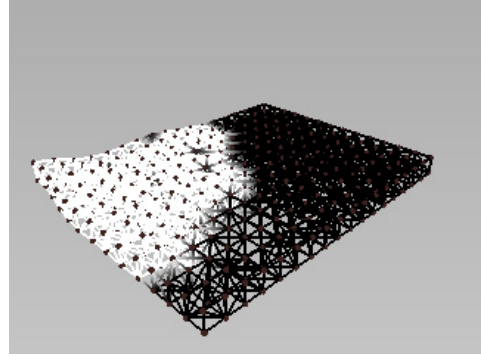
The extension to each of the voxel spring can be calculated where L_o is the size of the voxel and the expansion of each spring can be calculated by the expression below.

$$dx = \tau x(T - T_a)$$

We in the figure 6.10 can see a sheet under thermal stress. Render result is shown in figure 6.10a and the areas of thermal stress on the spring mass system in figure



(a) Render Results



(b) Areas Deformed on the Spring Mass System

Figure 6.10: Sheet Simulation Comparison

6.10b.

6.3.2 Gravity based forces

Thermal forces are one of the force that makes an object deform. Smolder is accompanied by fuel loss. As fuel gets transformed into ash it loses moisture and as fuel gets consumed ash will loose its structural strength. This effect can be captured in some of the ash formation. Ash in most cases is structurally weak and can deform under its own weight. Gravitation force is more dominant than thermal forces.

We can define extension of a spring using Hook's law with external force due to gravity on a voxel with the equation below.

$$F_g = kx$$

We can extend the change in spring length dx as a function gravitational constant and mass of the voxel.

$$dx = gM_o$$

Figure 6.11 shows the marching cubes that are being deformed with the oxi-

dation front. Figure 6.12 shows the final render image from the above spring mesh.

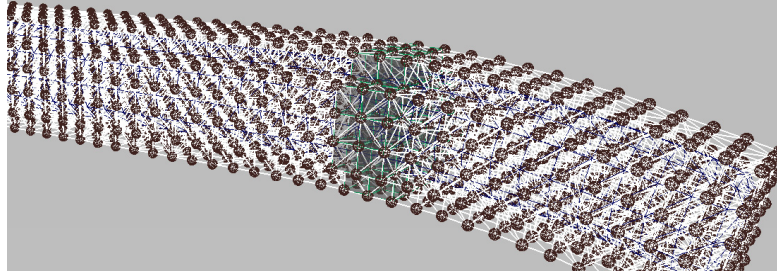


Figure 6.11: Cigarette Spring Mesh Showing Deformation



Figure 6.12: Gravity based deformation render

We define two different types of forces in the above sections. It is possible to add other type of forces to the system in addition to gravity and thermal based. We only explore these two options as they seem relevant to our case. Other forces can be applied to effect the deformation.

6.3.3 Mass Spring Force Transfer

We in our model apply extension to the edge spring of the spring voxel. This extension translates to a force that the spring mass system exerts in resistive force to balance the system. An external force is applied to the center and that is translated to a an edge length increment for the edge spring. This extension is translated to a resisting force from the other springs in the system causing the spring voxel to change size and

deform. The body and face springs are not expendable but are there to maintain support structure of the spring mass system.

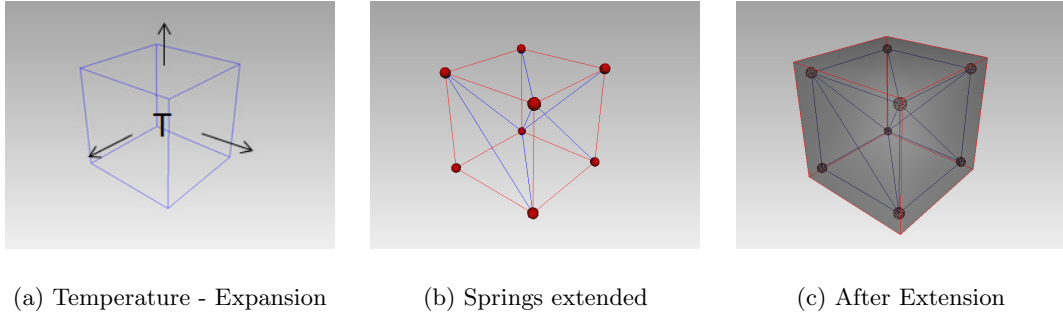


Figure 6.13: Voxel Expansion

In figure 6.13a temperature from the voxel is driving the expansion. Figure 6.13b shows the edge springs being effected on application of this force. Figure 6.13c shows the resulting spring voxel after expansion.

Chapter 7

Rendering of Smolder

Visualization consists of multiple parts as we need to render the oxidation front, reaction zone and ash. There have been multiple attempts to render burnt meshes focusing on the combustion area. One such attempt was by Lee et al[27] in which portion of meshes turned to a blacken color on burning. Malek and Keyser [37] use a projection method with a texture strip that is used to gradually change the burnt surface as it transitions from fuel to being burnt. Their model is only dependent on fuel used in a cell and that result being mapped to a polygonal face on the burning mesh. In their model stages of the combustion process is mapped to a texture which is then mapped to a polygonal face changing it red to black. In our simulation we require frame to frame tracking as we need a texturing method that allows textures to vary with properties of the simulation. We use same method to render effect of air flow on smolder. Zhu et al[64] proposed a method that allows frame by frame tracking on a polygonal surface using distance fields and linearly interpolated textures. This allows modification of textures to the oxidation front so we can render changes to the front during smoldering.

Zhu et al [64] proposed a model for mapping texture coordinates. In their

model surface textures coordinates are generated using the original texture coordinates from the unburnt mesh as a starting point. At each frame using the distance field as a measure and original texture coordinates, new texture coordinates are calculated using linearly interpolation. Each vertex in the voxel inherits texture coordinates interpolated from the original fuel mesh texture map. Smolder is a low temperature process for

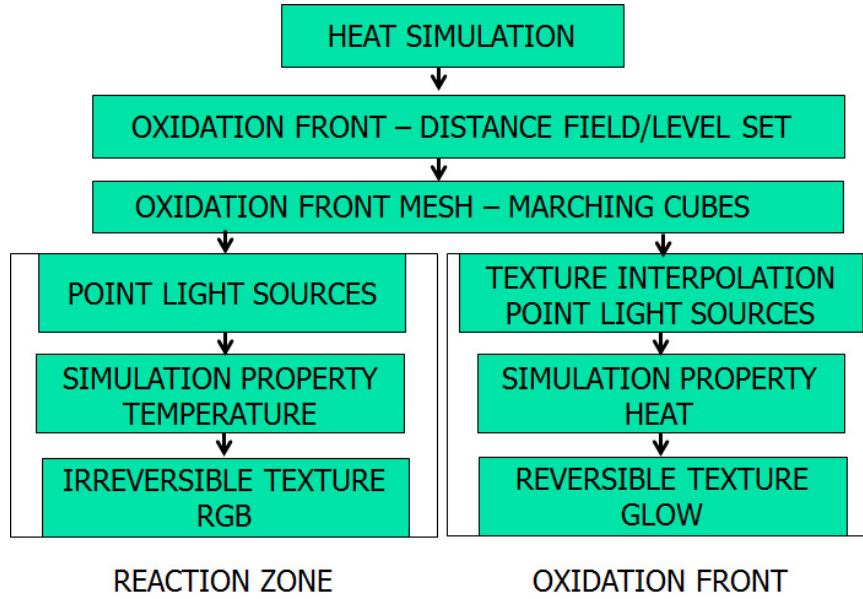


Figure 7.1: Texture Flow

which we need to map multiple properties to the polygonal mesh to render oxidation front and reaction zone. Oxidation front tends to change color and illuminate light as it is composed of small light sources driven by heat generated by the simulation system. Details of these models are described in the sections below. We describe two type of textures one additive that allows change in color with property and the color change is irreversible, second a reversible texture that allows changing of color depending on mapped properties and being reversible. As the simulation properties that are mapped change, texture will also follow that change. Figure 7.1 shows the two texture tracks for reaction zone and oxidation front.

7.1 Reaction Zone

We got inspired by the figure 7.2 showing the cigar oxidation zone. In order to render reaction zone we create a texture that changes color with simulation properties in this case temperature and heat released from the voxel. We model that by a texture darkening to black or a predefined color and then retain that color. When undergoing a change in color under combustion the change in the reaction zone are defined by us as irreversible changes. Zhu et al[64] model does not allow for this as their model can only model reversible change. In the case of fuel cooling it will regain its old texture. Smoldering is low temperature so fuel can cool regularly. With voxel in the reaction zone we model changes to fuel as being irreversible. Texture once changed will not change back.



Figure 7.2: Cigar Reaction Zone

In smolder the reaction zone is where fuel changes color and with moisture evaporating, and it being pyrolyzed it gets ready to be oxidized. Using a spherical meatball used by Ilic and Fua[18] as a point light source we identify and render the region in the reaction zone. We can define the region using the equations below where T_p defines the reaction zone. We use the oxidation front as the starting from a triangulated oxidation front to model the reaction zone. It is a simple technique to model an animated

implicit surface from a mesh. It also allows to create a three dimensional texture.

$$\|\vec{r'}\| = \sqrt{(x - rC_x)^2 + (y - rC_y)^2 + (z - rC_z)^2}$$

$$\|\vec{r_0'}\| = \sqrt{(P_{1x} - rC_x)^2 + (P_{1y} - rC_y)^2 + (P_{1z} - rC_z)^2}$$

Spherical Meatballs as they define an iso surface with radius are shown below. Each sphere forms an area of influence for a point light source. The radius of the sphere is a function of temperature from the corresponding voxel in simulation. We identify the voxels that are currently in the reaction zone as these voxels are mapped to a polygonal mesh using marching cubes. During the course of calculating this mesh we identify the triangles created that match the simulation voxels in the reaction zone. We add random light sources proportionate to the area of triangle from the marching cube. Figure 7.1 gives the flow of the process.

$$\phi_p(x, y, z) = T_p - \sum_{i=0}^N \exp(-k\|\vec{r'}\| - \|\vec{r_0'}\|)$$

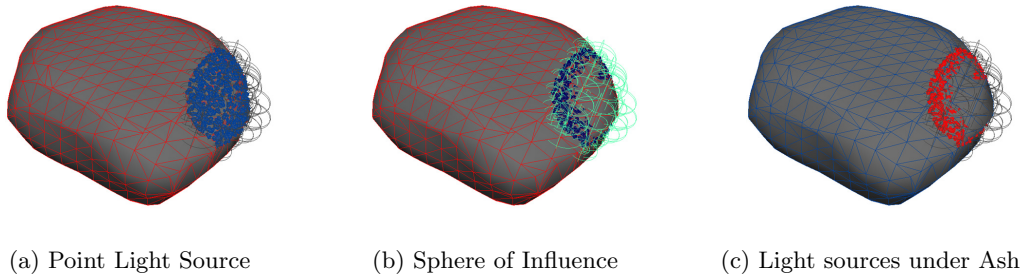


Figure 7.3: Briquette - Light Source Mapping to Reaction Zone

Figure 7.3a shows the point light sources as they would be generated on the oxidation front. The light source will have two outcomes one brightness due to heat generated and the other radius due of influence of temperature increase and its effect

on the fuel. As the temperature rises the fuel gets pyrolyzed thus causing it to change of color to unburnt fuel. Rendering is done in maya using their API [2] and to model coloration on the polygonal fuel mesh we use the soup library [55].

Figure 7.3b shows the sphere of influence for the point source formed and those are used to create the reaction zone. As we move away from the light source, the texture becomes lighter.

Figure 7.3c Shows the light sources visible with ash being visible. Ash formation at fuel gets burnt and covers most of the oxidation front.

We define an additive type of texture which is used to render the reaction zone. The texture color for a vertex falling within the voxels in the reaction zone is rendered by the following equation where $color_T$ as its proposed color and $color$ is what finally gets rendered. We define color with Red, Blue and Green colors for the reaction zone. Only the oxidation front is capable of chemical oxidation and rest of the reaction zone only darkens to change texture and show pyrolysis on the fuel.

$$color = \begin{pmatrix} Red \\ Green \\ Blue \end{pmatrix}$$

$$color_T = color + K_{color} \frac{(T - T_p)}{(T_{max} - T_p)}$$

$$color = \begin{cases} color, T < T_p \& T > T_{max} \\ \max(color_T, color), T_p > T > T_{max} \end{cases}$$

max function we propose here returns with the color closest to max color value for the reaction zone.

Figure 7.4 shows a cigarette being blown from one side. Due to the increased oxygen supply there is more heat generated on that side of the cigarette. Figure 7.5a shows the heat map and we can see there is more heat produced in-turn causing increased temperature. We can also see that in the simulation results shown in figure 7.5b. The reaction zone showing pyrolysis is more spread then the side opposite to airflow. Textures as described earlier in case of pyrolysis are irreversible.

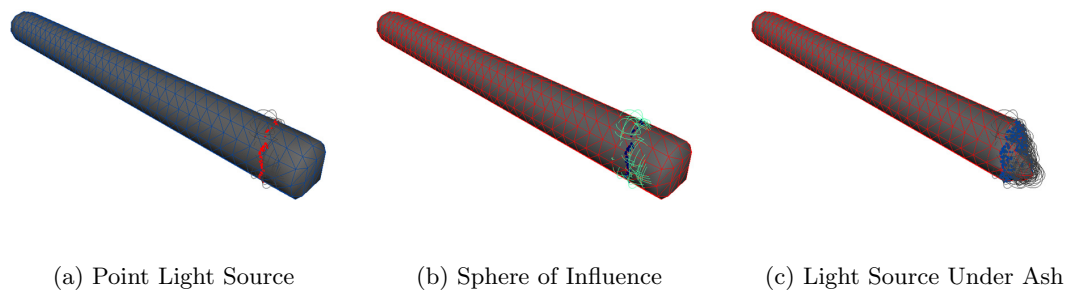


Figure 7.4: Cigarette - Light Source Mapping to Reaction Zone under Wind

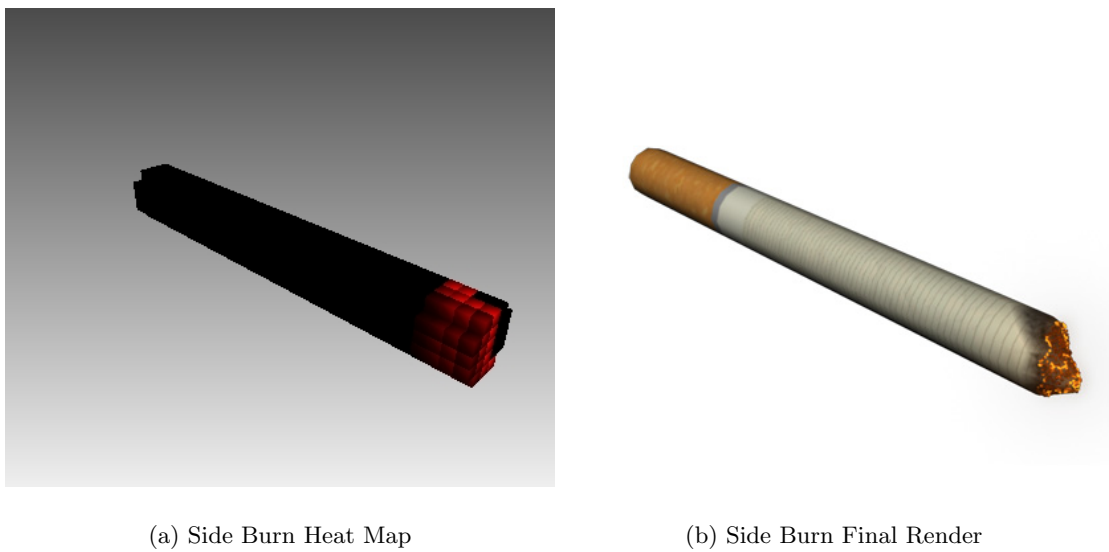


Figure 7.5: Cigarette Side Burn Rendered Reaction Zone

TEMPERATURE $\longrightarrow$ RGB

REACTION ZONE

TEXTURE

HEAT $\longrightarrow$ Glow

OXIDATION FRONT

TEXTURE

(a) Reaction Zone Texture Property Map

(b) Oxidation Front Texture Property Map

Figure 7.6: Texture Property Mapping

In figure 7.6 we show how the texture will behave in the reaction zone and on the oxidation front.

7.2 Oxidation Front

As described earlier we generate the oxidation front using level set and create a polygonal mesh using marching cubes. At every time step we recreate the oxidation front and the triangulation is different. As discussed earlier we create randomly points depending on the triangle area of the corresponding marching cube. These point light sources as described earlier help us to model a texture shown in figure 7.7. Voxels releasing more heat glow than the others, appear to be brighter as shown in figure 7.7b.

At each time step new light sources are created to match the newly generated oxidation front mesh. Updated to a new heatmap from the simulation voxels, we create a texture map from the heat values of these light sources. Figure 7.7 shows examples of such heat maps where the glow value of each point light source is given below.

$$Glow = K_{\text{glow}} \frac{\Delta H}{H_{\text{max}}}$$

H_{max} is the maximum heat release possible for the material. K_{glow} is the heat

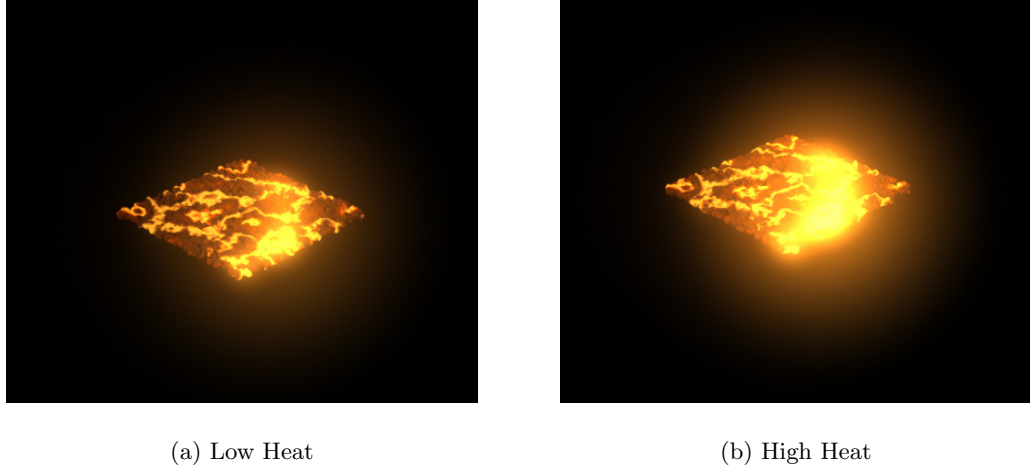


Figure 7.7: Heat Release from Oxidation Front

glow release constant depending on material that is releasing the heat. ΔH change in heat in a voxel at a time-step.

7.3 Ash

Ash is the byproduct of combustion and here we make an attempt to model it. We use a similar approach as rendering reaction zone. We track the voxels where the fuel has been used and texture the surface to reflect formation of ash. Ash forms a layer over the oxidation front, we color it apply some noise to it. Figure 7.8 shows the ash formation on a briquette as it smolders.

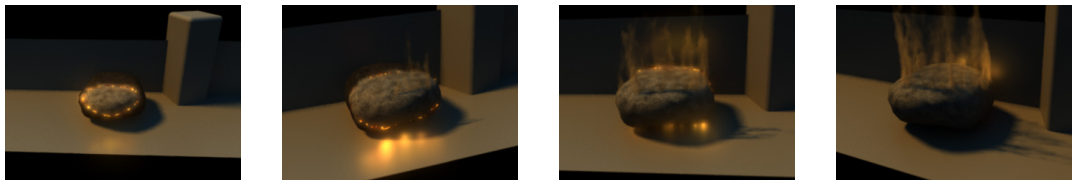


Figure 7.8: Briquette with Oxidation front rendered

On the next page in figure 7.9a and 7.9b we show ash texture formation in green

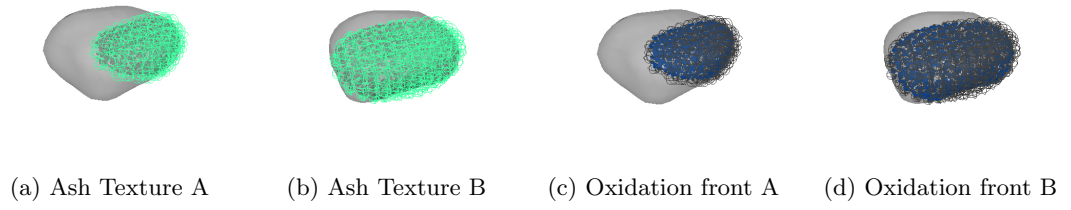


Figure 7.9: Briquette showing Ash Texture Formation

showing where all the ash formation exists. Figure 7.9c and 7.9d show the oxidation front in blue, highlighting its position relative to ash formation. All the four figures show the influencing light sources to create the ash texture.

Chapter 8

Implementation

We here describe multiple examples of smoldering combustion using the previously described model. The examples involve different materials, shapes, airflow and deformation conditions to give multiple outcomes. We also discuss standard examples like cigarette, charcoal briquette and cigar that are fundamentally associated with smolder. Later on in this section we describe more examples that deform like paper and leaf. Smolder plays an important part in many day to day processes. Cigarette is the most common example of this process. Smolder propagation is based mainly on heat retention.

8.1 Cigarette

Cigarette is one of the most common examples that is associated with smoldering combustion. Here we simulate cigarette smoldering and have a few examples.

- Self sustained smolder
- Draw
- Side Blow

First we simulate a self sustained smoldering reaction with the parameters listed in the table. To start smolder we ignite the cigarette and keep the heat source till the voxels in contact reach pyrolysis temperature (T_p). In self sustained the diffused oxygen is controlling the oxidation mechanism. Smolder will continue to propagate until oxidation heat is greater than heat losses. Once the heat losses are more than the oxidation heat, the smolder will extinguish.

In figure 8.1 the simulation results are compared to actual self smoldering photograph. In this case the cigarette is left on an ashtray for some time till the cigarette goes off.

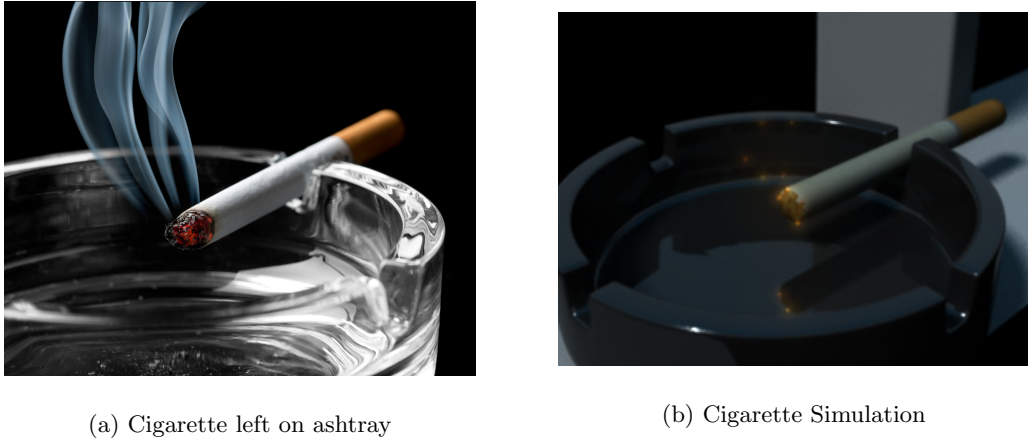


Figure 8.1: Cigarette Self Smolder Comparison

Table 8.1: Cigarette

Object	D	λ	C_p	K	v_g	k_r	μ
self-sustain	0.453	4×10^{-4}	1.6	5×10^{-3}	0.0	0.5×10^5	0.50
draw	0.453	4×10^{-4}	1.6	5×10^{-3}	0.24	0.5×10^5	0.50
side blow	0.453	4×10^{-4}	1.6	5×10^{-3}	0.5	0.5×10^5	0.50

We show in figure 8.2 the effect of pyrolysis due to different types of tobacco. Various tobacco can have different material properties governing the heat released during

smolder and that can govern the pyrolysis spread. In our simulations we have used ρ_g as 1.0×10^{-3} and ρ_s as 0.34 taken from the works of Lu and Dong [33], Buckmaster and Lozinski [11], and Ohlemiller, [44]. Heat of reaction is taken from work done by Nagao et al [40] and Rostami et al [54] .



Figure 8.2: Pyrolysis with different tobacco properties

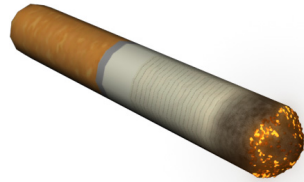
Second we simulate a cigarette draw by adding airflow in the direction of smolder travel. The velocity for the airflow and other simulation details are shown in table 8.1. This is an example of forward smolder where the direction of airflow is in the same direction as smolder travel. We model the airflow being in one direction with the magnitude of velocity. First the cigarette is ignited so the self sustained smolder reaction starts. Airflow is applied for four seconds and then removed in order to simulate a draw. We can see in figure 8.4 actual photographs for a draw action and normal action.

We show the draw in and out action simulation in figure 8.3 and also the actual in figure 8.4. We compare the shape of the cones on the cigarette due to the direction of airflow in draw in and out action.

Figure 8.5 shows a comparison of a person blowing on the side of the cigarette

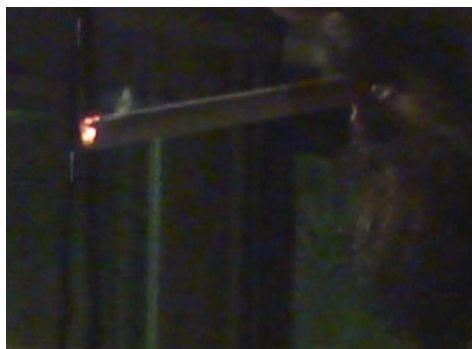


(a) Draw In

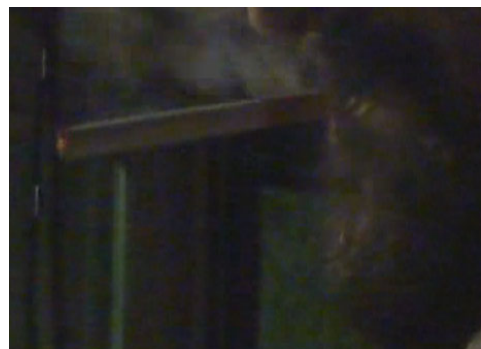


(b) Draw Out

Figure 8.3: Draw In and Out Simulation



(a) Cigarette Draw In



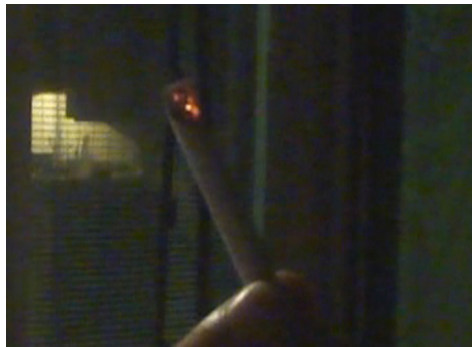
(b) Cigarette Draw Out

Figure 8.4: Cigarette Draw in and Out Motion

and a simulation screen shot.

Then we simulate a cigarette that is being blown on from the side and due to excess oxidation there are changes in the shape in the oxidation front cone. Figure 8.6a and figure 8.6b is at the onset of air flow stops you can see the change in the oxidation cone in 8.6c and figure 8.6d

Occasionally similar to this behavior could be seen in very cold areas where the cigarette gets cooled on one side and changes the oxidization zone cone shape.



(a) Side Burn



(b) Side Burn Simulation

Figure 8.5: Side Burn Actual vs Simulation



(a) Airflow From Side



(b) Airflow continued



(c) Airflow Stopped



(d) Cone after sometime

Figure 8.6: Cigarette Simulation With Side Blow.

Cigarette ash deforming due to structural changes with the formation of ash is shown in figure 8.7 on the next page.

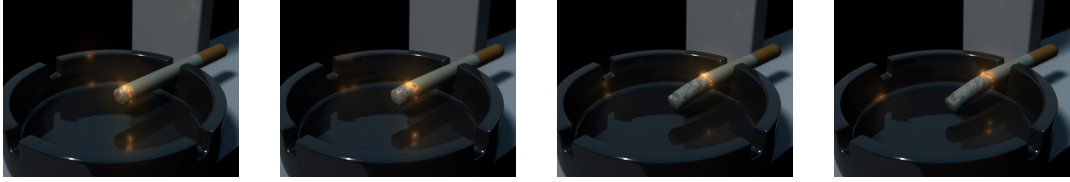


Figure 8.7: Cigarette Simulation With Ash Deformation

8.2 Cigar

We discuss smoldering of cigar as it has different material properties than a cigarette. It tends to burn slower as it has different material construction than a cigarette. Here we capture self sustained smolder with different material properties. We can see from the simulation the shape of the cone is different from a cigarette undergoing smolder. The cone is very shallow as the tobacco has a slower rate of burning and has ash that is more porous than a cigarette. Table 8.2 will show different properties used for cigar in the self smolder simulation. High diffusion constant of ash allows more air to diffuse through to the oxidation front causing the cone to be shallower. After some time of smoldering, the cigar tends to extinguish. Figure 8.8 shows results of the simulation.



Figure 8.8: Cigar Simulation With Self Smoldering

Table 8.2: Cigar

Object	D	λ	C_p	K	k_r	μ
self-sustain	0.723	4×10^{-4}	1.5	5×10^{-3}	0.2×10^5	0.04

8.3 Bunny

Bunny is an example of a nonstandard shape under smolder combustion using parameters from table 8.3. The screen shots from the simulation are taken at different times in the simulation. Figure 8.9 shows simulation with no assistance. In figure 8.10 we start the ignition from the ears and let it follow down to the body. As we see the outside of the bunny smolders and converts to ash but the inside is still unaffected. Ash around the bunny forms a insulation layer and prevents heat loss, thus creating a blanket around it and allowing smolder to sustain longer. Figure 8.10 shows how the oxidation front moves with time.



Figure 8.9: Bunny Self Smolder Simulation



Figure 8.10: Bunny with No Ash

Table 8.3: Bunny

Object	D	λ	C_p	K	k_r	μ
self-sustain	0.15	4×10^{-4}	8.5	5×10^{-3}	0.1×10^5	0.04

8.4 Sheet

We here show an example of a sheet where the material properties are described in table 8.4. We show deformation when the sheet is self smoldering in figure 8.11 showing secondary deformation. The plate is ignited on one of the corners and is allowed to smolder. As we see the plate deforms as the oxidation front moves forward.

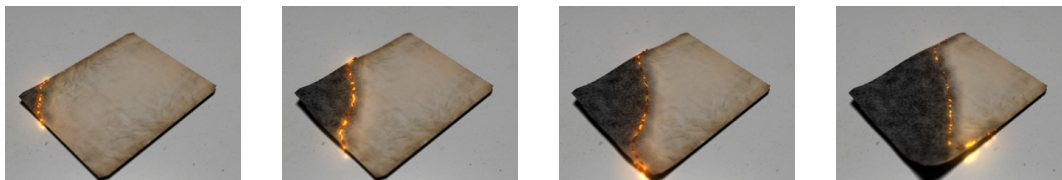


Figure 8.11: Sheet under secondary deformation

Table 8.4: Sheet

Object	D	λ	C_p	K	k_r	μ
self-sustain	0.8	1×10^{-4}	9.5	5×10^{-3}	0.4×10^5	0.02

8.5 Briquette

Here is an example of a set of briquette under self smolder. Figure 8.12 shows multiple screen shots from the briquette simulation with ash formation. Table 8.3 on the previous page is used as the parameters for the simulation.

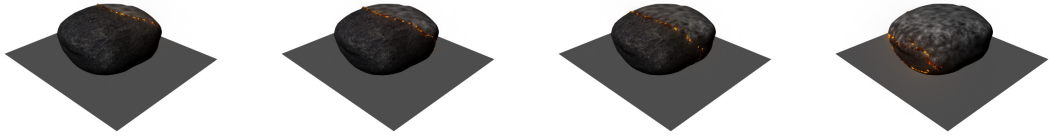


Figure 8.12: Briquette with ash

We use the example of a standard briquette under self smolder and as discussed earlier with the bunny example this is another shape with the same properties as the bunny. The outer portion of the briquette tends to become ash insulating the inside of the briquette hence trapping the heat.

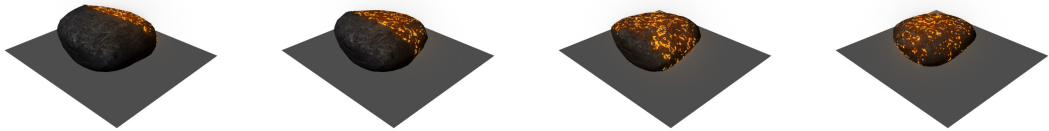


Figure 8.13: Briquette no ash

Figure 8.13 shows multiple screen shots from the briquette simulation with no ash showing the shape of unused fuel.

8.6 Torus

We start the smolder initiation in two different parts of the torus. As smolder propagates it ends up breaking the torus into two parts. Figure 8.14 shows the torus with ash formation. We end up using Table 8.3 as the parameters for the bunny for this shape.



Figure 8.14: Torus with Ash formation



Figure 8.15: Torus with oxidation front

Figure 8.15 shows the torus with breaking in two parts. As the ash is removed there is no fuel material left under the ash.

8.7 Leaf

We test our model with a leaf and it smoldering. Below the leaf is ignited on one side of the leaf and allowed to smolder. We use similar texture for both pyrolysis and ash formation. As we can see in figure 8.17 the leaf shows ash, oxidation front and the pyrolysis zone. Figure 8.16 shows the leaf deforming. Table 8.5 shows the properties we used for the leaf smoldering.

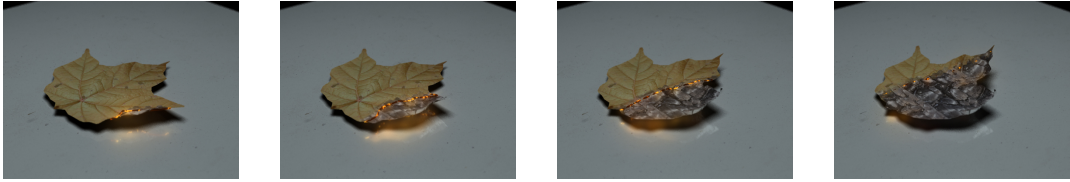


Figure 8.16: Leaf Rendered with Deformation

Table 8.5: Leaf

Object	D	λ	C_p	K	k_r	μ
self-sustain	0.3	1×10^{-4}	9.5	5×10^{-3}	0.5×10^5	0.04



Figure 8.17: Leaf showing pyrolysis and Ash formation

8.8 Matchstick

Here we capture wood based thermal deformation with change in temperature.

Figure 8.18 shows the matchstick deform with table 8.6 showing simulation parameters.

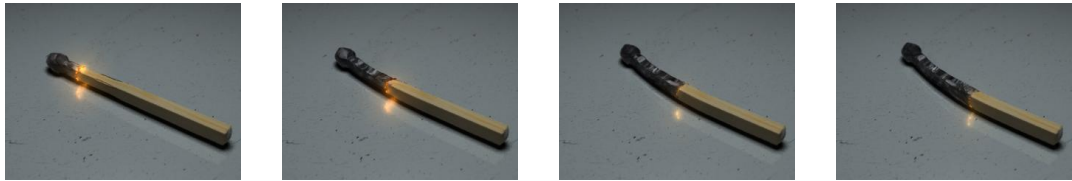


Figure 8.18: Matchstick with deformation and Ash formation

Table 8.6: Matchstick

Object	D	λ	C_p	K	k_r	μ
self-sustain	0.1	2×10^{-3}	3	5×10^{-3}	0.3×10^5	0.02

Chapter 9

Conclusions and Future Work

9.1 Conclusion

The innovative smoldering combustion model developed and described in this dissertation introduces smoldering to the graphic community and presents a promising approach to simulate, animate and control slow flame-less smoldering combustion. The model uses chemical energy as the driving process to create visually incredible results. Our methodology uses a simple system to simulate oxidation of fuel in voxelized space. It also is the first attempt to create an automatic three dimensional spring based secondary deformation model with direct coupling to a physical simulation system and surface generation. We also introduce a new rendering scheme involving heat maps and point light sources to give a realistic feel to our smolder render.

We also demonstrate with multiple application that it is a model that is viable. It meets the goal of adding more realism to the current combustion models by introducing air flow effect on smolder, transition to flaming, smolder extinguishing, fuel deformation and ash formation, which allows us to study other aspects that initiate flame less smolder combustion and fire safety.

9.2 Future Work

We have shown here a model that is a proof of concept and can be explored to add new and extend some of our functionality. We in the section below describe a number of possible refinements to our model and extension to accommodate new areas of research. We also explore where our system can be used to add value to other existing systems.

9.2.1 Sooth and Char Formation

We model ash as a part of the fuel mesh with rendering it very simply. This model could be extended to accommodate soot formation. Soot and Char are a by product of smoldering. That coupled with ash creation that could help modeling a number of interesting physically accurate effects.

9.2.2 Fluid Model for Wind

We in our model use a simplified approach to show the airflow effect on smolder. Traditional mechanical engineering simulates the effect of air flow on smolder simply. Multidimensional flow model is still an import problem to solve that has not been explored in great detail in the engineering community. There could be a number of interesting effects that could become part of this simulation. The fluid model could also include its effect on soot formation that is discussed in the previous sub section. The interaction of sooth with wind would also produce more interesting effects.

9.2.3 Fuel and Ash Fracture

A further extension of this work could be modeling of fuel, or ash fracture. Fuel on excessive heating leads to potential structure weakening of fuel and that could

lead to fracture even before reaching oxidation.

9.2.4 Forest Fires

Forest fires is based on smoldering and this work can extend to creating smolder patterns for them. This could also be extended to show starts of a forest fire.

9.2.5 Internal collision detection

A good extension to our system would be a collision detection system that allows for collisions on deformation. Once there is breaking of ash or fuel the collision detection there is not calculated. In our current system collisions on the spring mesh are not calculated. Deformation can cause springs to collide and create undesired results.

9.2.6 Climate Effect on Smolder

There are several climatic factors like air-pressure, humidity and atmospheric temperature that could influence the propagation of smolder. The above factors could also initiate the start of smolder. There could be many interesting effects that could be designed around these factors that could expand this research to a number of extra and pleasing effects.

Nomenclature

Δt	time-step for the simulation
Δx	Simulation grid size
$\frac{\delta}{\delta t} V^T$	Heat diffusion during smolder
∇	partial differential operator
∇^2	Second order partial differential operator
ϕ	Level Set Distance Field
C	Damping Matrix
$H_{flaming}$	Heat required to achieve flaming
$H_{burnfront}$	Heat released from burnfront
k_r	Chemical Reaction Rate Constant
k_{DT}	Thermal Conductivity Constant
Q	Heat
r	Rate of Reaction
T	Temperature

$V_{i,j,k}^T$	Cumulative Temperature gradient
$V_{i,j,k}^{MC_{fuel\ diffusion\ coefficient}}$	Material Constant Fuel Diffusion
$V_{i,j,k}^{MC_{fuel\ mass\ fraction}}$	Material Constant Fuel Mass Fraction
$V_{i,j,k}^{MC_{fuel\ specific\ heat}}$	Material Constant Fuel Specific Heat
$V_{i,j,k}^{MC_{oxygen\ diffusion\ coefficient}}$	Material Constant Oxygen Diffusion
$V_{i,j,k}^{MC_{oxygen\ mass\ fraction}}$	Material Constant Oxygen Mass Fraction
$V_{i,j,k}^{MC_{oxygen\ specific\ heat}}$	Material Constant Oxygen Specific Heat
$V_{i,j,k}^{k_T}$	Heat Diffusion constants or different conductivity values
$V_{i,j,k}^{T_i}$	Heat transfer from surrounding air to fuel
$Vxl_{reactionzone}$	Voxels in the reaction zone
λ	Temperature Diffusion Constant
μ	Stoichiometric Constant for Reaction Rate
ρ_g	density gas
ρ_s	density solid
ρ	Combined density solid and gas
$V_{i,j,k}^S$	Voxels during simulation
C_p	Specific heat of Solid and Gas combined
D	Diffusion Coefficient for Oxygen

FFD	Free Form Deformation
k_f	Maximum Fuel Consumed per Time Step
k_{HE}	Heat diffusion between air and fuel constant
$k_{O_{surface}}$	Surface Absorption Coefficient for Oxygen
K	Convection Constant
M_g	Mass Gas
M_s	Mass Solid
N	Direction of Level Set
T_p	Temperature of Pyrolysis
T_a	Ambient Temperature
v_g	Velocity of gas
$V_{i,j,k}^{RZ}$	Voxels in the reaction zone
Vel	Velocity of Level Set
Y_f	Mass Fraction Fuel in Solid
Y_o	Mass Fraction Gas in Solid

Bibliography

- [1] A.P. Aldushin, A. Bayliss, and B.J. Matkowsky. On the transition from smoldering to flaming. *Combustion and Flame*, 145(3):579 – 606, 2006.
- [2] Autodesk. Maya 2014 api. <http://www.autodesk.com/>, 2013-2014.
- [3] Jernej Barbič, Fun Shing Sin, and Daniel Schroeder. Vega FEM Library, 2012. <http://www.jernejbarbic.com/vega>.
- [4] Alan H. Barr. Global and local deformations of solid primitives. pages 221–230, 1998.
- [5] Daniel Barrero, Mathias Paulin, and Ren Caubet. A Physics Based Multi-Resolution Model for the Simulation of Turbulent Gases and Combustion. In N. Magnenat-Thalmann and D. Thalmann, editors, *Eurographics Workshop on Computer Animation and Simulation, Milano, Italy, 07/09/99-09/09/99*, pages 177–188, Wien-NewYork, septembre 1999. Springer.
- [6] Daniel Barrero, Mathias Paulin, and Ren Caubet. Simulating turbulent Combustion. In *Graphicon 2000, Moscou, 28/08/00-02/09/00*, pages 181–188. ISBN 5-317-00048-3, aot 2000.
- [7] Christopher Batty. Sdfgen. <https://github.com/christopherbatty/SDFGen>, 2012.
- [8] Philippe Beaudoin, Sébastien Paquet, and Pierre Poulin. Realistic and controllable fire simulation. In *Proceedings of Graphics Interface 2001*, GI '01, pages 159–166, Toronto, Ont., Canada, Canada, 2001. Canadian Information Processing Society.
- [9] Haimasree Bhattacharya, Yue Gao, and Adam W. Bargteil. A level-set method for skinning animated particle data. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Aug 2011.
- [10] Paul Bourke. Polygonising a scalar field. <http://paulbourke.net/geometry/polygonise/>, 1994.
- [11] J. Buckmaster and D. Lozinski. An elementary discussion of forward smoldering. *Combustion and Flame*, 104(3):300 – 310, 1996.
- [12] Y. Chen, Qing-Hong Zhu, A. Kaufman, and S. Muraki. Physically-based animation of volumetric objects. In *Computer Animation 98. Proceedings*, pages 154–160, Jun 1998.

- [13] Norishige Chiba, Kazunobu Muraoka, Hiromichi Takahashi, and Mamoru Miura. Two-dimensional visual simulation of flames, smoke and the spread of fire. *The Journal of Visualization and Computer Animation*, 5(1):37–53, 1994.
- [14] Raphael Diziol, Jan Bender, and Daniel Bayer. Volume conserving simulation of deformable bodies. In *Proceedings of Eurographics*, Munich (Germany), March 2009.
- [15] Noura Faraj, Jean-Marc Thiery, and Tamy Boubekur. Voxmorph: 3-scale freeform deformation of large voxel grids. *Computer & Graphics Journal (Special Issue; Shape Modeling International 2012)*, 2012.
- [16] Markus Huber, Simon Pabst, and Wolfgang Straer. Wet cloth simulation. In *SIGGRAPH Posters’11*, pages 10–10, 2011.
- [17] Insung Ihm, Byungkwon Kang, and Deukhyun Cha. Animation of reactive gaseous fluids through chemical kinetics. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA ’04, pages 203–212, Aire-la-Ville, Switzerland, Switzerland, 2004. Eurographics Association, Eurographics Association.
- [18] Slobodan Ilic and Pascal Fua. From explicit to implicit surfaces for visualization, animation and modeling. In *Proc. of Inter. Workshop on*, 2003.
- [19] M. Inakage. A simple model of flames. In *Proceedings of the eighth international conference of the Computer Graphics Society on CG International ’90: computer graphics around the world*, CG International ’90, pages 71–81, New York, NY, USA, 1990. Springer-Verlag New York, Inc., Springer-Verlag New York, Inc.
- [20] Tomokazu Ishikawa, Ryo Miyazaki, Yoshinori Dobashi, and Tomoyuki Nishita. Visual simulation of spreading fire.
- [21] SoHyeon Jeong, Tae hyeong Kim, and Chang-Hun Kim. Shrinkage, wrinkling and ablation of burning cloth and paper. *The Visual Computer*, 27(6-8):417–427, 2011.
- [22] SoHyeon Jeong and Chang-Hun Kim. Combustion waves on the point set surface. *Computer Graphics Forum*, 32(7):225–234, 2013.
- [23] SoHyeon Jeong, Si-Hyung Park, and Chang-Hun Kim. Simulation of Morphology Changes in Drying Leaves. *Computer Graphics Forum*, 2013.
- [24] Yootai Kim. Rough interface reconstruction using the level set method. In *In VIS 04: Proceedings of the conference on Visualization 04*, pages 251–258. IEEE Computer Society, 2004.
- [25] F.E.T. Kingman and K.N. Palmer. Smouldering in dusts and fibrous materials part i. beech and deal sawdusts in still air. Fire Research Note 6, Fire Research Station, Borehamwood, England, 1952.
- [26] Caroline Larboulette, Pablo Quesada Barriuso, and Olivier Dumas. Burning paper: Simulation at the fiber’s level. In *Proceedings of the ACM SIGGRAPH Conference on Motion in Games*, ISBN: 978-1-4503-2546-2, pages 25–30. ACM New York, NY, USA 2013, ACM Press, nov 2013. <http://dx.doi.org/10.1145/2522628.2522906>.

- [27] Haeyoung Lee, Laehyun Kim, Mark Meyer, and Mathieu Desbrun. Meshes on fire. In *In EG Workshop on Computer Animation and Simulation*, pages 75–84, 2001.
- [28] Toon Lenaerts, Bart Adams, and Philip Dutré. Porous flow in particle-based fluid simulations. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08, pages 49:1–49:8, New York, NY, USA, 2008. ACM.
- [29] Shiguang Liu, Tai An, Zheng Gong, and Ichiro Hagiwara. Physically based simulation of solid objects? burning. In Zhigeng Pan, AdrianDavid Cheok, Wolfgang Mller, Maiga Chang, and Mingmin Zhang, editors, *Transactions on Edutainment VII*, volume 7145 of *Lecture Notes in Computer Science*, pages 110–120. Springer Berlin Heidelberg, 2012.
- [30] Shiguang Liu, Qiguang Liu, Tai An, Jizhou Sun, and Qunsheng Peng. Physically based simulation of thin-shell objects& burning. *Vis. Comput.*, 25(5-7):687–696, April 2009.
- [31] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21(4):163–169, August 1987.
- [32] Frank Losasso, Geoffrey Irving, Eran Guendelman, and Ron Fedkiw. Melting and burning solids into liquids and gases. *IEEE Transactions on Visualization and Computer Graphics*, 12(3):343–352, May 2006.
- [33] Zhanbin Lu and Yong Dong. Fingering instability in forward smolder combustion. *Combustion Theory and Modelling*, 15(6):795–815, 2011.
- [34] Moshe Matalon. Flame dynamics. *Proceedings of the Combustion Institute*, 32(1):57–82, 2009.
- [35] Zeki Melek and John Keyser. Interactive simulation of burning objects. In *Pacific Conference on Computer Graphics and Applications*, pages 462–466. IEEE Computer Society, IEEE Computer Society, 2003.
- [36] Zeki Melek and John Keyser. Multi-representation interaction for physically based modeling. In *Proceedings of the 2005 ACM symposium on Solid and physical modeling*, SPM '05, pages 187–196, New York, NY, USA, 2005. ACM, ACM.
- [37] Zeki Melek and John Keyser. Driving object deformations from internal physical processes. In *SPM '07: Proceedings of the 2007 ACM symposium on Solid and physical modeling*, pages 51–59, New York, NY, USA, 2007. ACM.
- [38] Kyungha Min and Dimitris Metaxas. A combustion-based technique for fire animation and visualization. *Vis. Comput.*, 23(9):679–687, August 2007.
- [39] Yuki Morimoto and Kenji Ono. Computer-generated tie-dyeing using a 3d diffusion graph. In George Bebis, Richard Boyle, Bahram Parvin, Darko Koracin, Ronald Chung, Riad Hammoud, Muhammad Hussain, Tan Kar-Han, Roger Crawfis, Daniel Thalmann, David Kao, and Lisa Avila, editors, *Advances in Visual Computing*, volume 6453 of *Lecture Notes in Computer Science*, pages 707–718. Springer Berlin Heidelberg, 2010.

- [40] Atsushi Nagao, Yoshiyuki Yamada, Keigo Miura, and Masayuki Horio. Evaluation of combustion of a shredded tobacco bed using a reactor to produce reverse smolder: Influence of properties related to mass transfer on combustion of the bed. *Beiträge zur Tabakforschung/Contributions to Tobacco Research*, 21(6):350–357, 2005.
- [41] Rahul Narain, Tobias Pfaff, and James F. O’Brien. Folding and crumpling adaptive sheets. *ACM Transactions on Graphics*, 32(4):51:1–8, July 2013. Proceedings of ACM SIGGRAPH 2013, Anaheim.
- [42] Duc Quang Nguyen, Ronald Fedkiw, and Henrik Wann Jensen. Physically based modeling and animation of fire. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, SIGGRAPH ’02, pages 721–728, New York, NY, USA, 2002. ACM.
- [43] Thomas J. Ohlemiller. Forced smolder propagation and the transition to flaming in cellulosic insulation. *Combustion and Flame*, 81(3-4):354 – 365, 1990.
- [44] T.J. Ohlemiller. Modeling of smoldering combustion propagation. *Progress in Energy and Combustion Science*, 11(4):277 – 310, 1985.
- [45] T.J. Ohlemiller. Smoldering combustion. *SFPE Handbook of Fire Protection Engineering. 3rd Edition*, pages 200 – 210, 2002.
- [46] T.J. Ohlemiller and D.A. Lucca. An experimental comparison of forward and reverse smolder propagation in permeable fuel beds. *Combustion and Flame*, 54(1?3):131 – 147, 1983.
- [47] K.N. Palmer and M.D. Perry. Smouldering in dusts and fibrous materials part x. cork dust under airflow conditions. Fire Research Note 123, Fire Research Station, Borehamwood, England, 1954.
- [48] Vincent Pegoraro and Steven G. Parker. Physically-Based Realistic Fire Rendering. In *Proceedings of the 2nd Eurographics Workshop on Natural Phenomena*, pages 51–59, 2006.
- [49] C. H. Perry and R. W. Picard. Synthesizing flames and their spreading. In *Proceedings of the Fifth Eurographics Workshop on Animation and Simulation*, pages 1–14. -, 1994.
- [50] W. T. Reeves. Particle systems technique for modeling a class of fuzzy objects. *ACM Trans. Graph.*, 2(2):91–108, April 1983.
- [51] Guillermo Rein. Smouldering combustion phenomena in science and technology. *International Review of Chemical Engineering*, 1:3–18, 2009.
- [52] Guillermo Rein, A. Carlos Fernandez-Pello, and David L. Urban. Computational model of forward and opposed smoldering combustion in microgravity. *Proceedings of the Combustion Institute*, 31(2):2677 – 2684, 2007.
- [53] Roderick M. Riensche and Robert R. Lewis. Modeling and rendering physically-based wood combustion. In *Proceedings of the 5th International Symposium on Advances in Visual Computing: Part I*, ISVC ’09, pages 896–905, Berlin, Heidelberg, 2009. Springer-Verlag.

- [54] Ali Rostami, Jayathi Murthy, and Mohammad Hajaligol. Modeling of a smoldering cigarette. *Journal of Analytical and Applied Pyrolysis*, 66(1-2):281 – 301, 2003.
- [55] Peter Shipkov. Soup node framework. <http://www.soup-dev.com/>, 2011.
- [56] Hang Si. Tetgen, a delaunay-based quality tetrahedral mesh generator. *ACM Trans. Math. Softw.*, 41(2):11:1–11:36, February 2015.
- [57] W. Smith and J. Hashemi. *Foundations of Materials Science and Engineering*. McGraw-Hill Education, 2009.
- [58] Jos Stam. Real-time fluid dynamics for games, 2003.
- [59] Jos Stam and Eugene Fiume. Depicting fire and other gaseous phenomena using diffusion processes. pages 129–136, 1995.
- [60] W.C. Strahle. *An Introduction to Combustion*. Number v. 1 in An Introduction to Combustion. Gordon and Breach Science Publishers, 1993.
- [61] Chris Wojtan, Mark Carlson, Peter J. Mucha, and Greg Turk. Animating corrosion and erosion. In *Proceedings of the Third Eurographics Conference on Natural Phenomena*, NPH’07, pages 15–22, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.
- [62] Mi You, Taekwon Jang, Seunghoon Cha, Jihwan Kim, and Junyong Noh. Realistic paint simulation based on fluidity, diffusion, and absorption. *Computer Animation and Virtual Worlds*, 24(3-4):297–306, 2013.
- [63] Ye Zhao, Xiaoming Wei, Zhe Fan, Arie Kaufman, and Hong Qin. Voxels on fire. In *VIS ’03: Proceedings of the 14th IEEE Visualization 2003 (VIS’03)*, page 36, Washington, DC, USA, 2003. -, IEEE Computer Society.
- [64] Jian Zhu, Yuanzhang Chang, and Enhua Wu. Realistic, fast, and controllable simulation of solid combustion. *Journal of Visualization and Computer Animation*, 22(2-3):125–132, 2011.