SANDIA REPORT

SAND2017-9463
Unlimited Release
Printed June 2017

Rubble Fire Multi-Phase Model Development

Heeseok Koo, Alexander L. Brown, Tyler Voskuilen, Flint Pierce

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology and Engineering Solutions of Sandia, LLC.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2017-9463
Unlimited Release
Printed June 2017

Rubble Fire Multi-Phase Model Development

Heeseok Koo and Alexander L. Brown
Fire Science and Technology, 01532

Tyler Voskuilen and Flint Pierce
Computational Thermodynamics and Fluid Mechanics, 01541
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185

Abstract

This report presents multi-phase modeling approaches that are developed for simulating rubble fire scenarios similar to a large-scale rubble pool fire test at Sandia National Laboratories using composite materials and jet fuel [1]. The rubble pool fire test burnt oddly shaped combustible solid objects submerged in liquid fuel. As an intermediate step toward a full scale rubble fire simulation, various model improvement tasks were performed. For modeling solid decomposition, a multi-step degradation model was used for canonical verification problems and the Chemical Percolation for Devolatilization (CPD) approach was implemented. Capabilities of Lagrangian particle approach has been extended such that a group of particles may represent a solid bulk. For gas-liquid interface, the volume of fluid (VOF) technique was implemented and relevant physics were added. The developed tools offer a potential for simulating three-phase (gas, liquid, and solid) combustion applications.

Contents

List of Figures

List of Tables

Nomenclature

`p0`, (p)	Initial(current) fraction of intact bridges
`c0`, (c)	Initial(current) fraction of stable (char) bridges
`sigp1`, $\sigma + 1$	Cluster coordination number/number of bridges per cluster
`ab`, A_b	Rate prefactor for bridge breaking
`eb0`, E_b	Activation energy for bridge breaking
`ebsig`, σ_b	Standard deviation of activation energy for bridge breaking
`mw1`, M_{clust}	Average molecular weight of cluster
`mdel`, M_{del}	Average molecular weight of sidechains
`ac`, ρ	Ratio of rates of bridge breaking to char bridge formation
`ec0`, E_c	Difference in activation energies (bridge breaking and char bridge formation)
`ag`, A_g	Rate prefactor for gas release
`eg0`, E_g	Activation energy for gas release
`egsig`, σ_g	Standard deviation of activation energy for gas release
`acr`, A_{cross}	Frequency factor for cross linking
`ecr`, E_{cross}	Activation energy for cross-linking
$f_{char}, f_{tar}, f_{gas}$	Fraction of initial particle mass that remains or has be converted into char, tar, and gas
$\omega, \dot{\omega}_k$	Reaction rate, rate of kth elementary reaction
f_m, f_r, f_p, f_c	Multiplier, rate, pressure, and concentration functions for gas chemistry reaction rate
E_a	Activation energy for gas chemistry reaction
A	Arrhenius prefactor for gas chemistry reaction
R	Gas constant
ρ_g, ρ_s	Density of gas or solid

$Y_a, Y_{a,b}$	Mass fraction of species a in phase b
α	Volume fraction of liquid
u	Velocity
S_α	Source of liquid
u_c	Interface compression velocity
n	Interface normal
ϵ	Arbitrarily small number
α_s	Smoothed volume fraction of liquid
Fo	Diffusive smoother Fourier number
$\dot{m}_{evap}$	Volumetric mass source from evaporation
m''_{evap}	Evaporative mass flux
ρ_L	Liquid density
ρ_g	Gas density
A	Volumetric interface area (area per volume)
W	Area weighting factor
T_{sat}	Saturation temperature
T	Temperature
β	Evaporation model accommodation coefficient
h_{fg}	Phase change formation enthalpy
K	Evaporation rate constant
R	Universal gas constant
u_{est}	Velocity estimate from the momentum equations
D_p	Pressure correction equation scaling coefficient (mass-based)
A_p	Pressure correction equation scaling coefficient (volume-based)
p	Pressure
$\dot{m}$	Mass flux
γ	Surface tension
κ	Interface curvature
D_u	Diagonal term from the momentum equations
α_r	Radiative absorptivity

Chapter 1

Introduction

Fires involving a mix of solid and liquid fuels are realistic hazards. These may occur in the case of a wrecked aircraft (aviation fuel mixed with aircraft materials), or in the case of a rubbish fire where combustible liquids may be co-mingled with combustible solids. The former is highly relevant to the National Nuclear Safety Administration (NNSA) safety environments for engineered components, while the latter is a classic case in a domestic or industrial setting where oily rags or similar materials are often found to be major fire hazards. There are not many mature computational methods that can be employed to model such an event with any expectation of accuracy. Such a problem can be broken down into component problems that do have existing modeling methods with reasonable predictive capability. To model a mixed fuel fire, one might bring together the various individual model components to create a general model that can predict the fire behavior in these complex environments.

Liquid fuel mixed with solid fuel creates a fire environment full of physical complexity. Liquids fuels are normally much more volatile than solids, and are often semi-transparent. The high volatility would lend to them burning earlier in the fire. Liquids also typically burn with minimal residue. Solids on the other hand often leave significant residue, volatilize at higher temperatures, and can vary from semi-transparent to opaque. When the two are mixed, the burn rate is not merely a weighted fraction of the burn rate of the two materials. There are complexities relating to the mixture that can be expressed in terms of the heat transport, mass transport, fluid flow, and reaction rates. Fundamental understanding of all of these phenomena exists, although the complexity of the system is such that it has not been found to ever previously have been resolved. The assimilation of the appropriate methods to model such an event is the crux of the technical problem addressed in this effort.

Sandia National Laboratories (SNL) develops and maintains the SIERRA architecture, which is a generalized architecture for simulating engineering physics problems. The architecture enables much of the computing functionality that is required to perform calculations of engineering physics. The architecture is employed in the design of several fluid mechanics tools that can be used to predict heat transport and fluid behavior. One of the SIERRA/Fluid Mechanics capabilities is the Fuego code, which is designed to predict low-Mach number reacting flows [4]. It is generally described as a fire computational fluid dynamics (CFD) code, and has many modules that enable multi-physics calculations. Other Fluid Mechanics capabilities exist, and coupling exists between many of the SIERRA modules to enable multi-physics computations. To model a rubble fire, one needs the ability to

model both the liquid and the solid. There are presently three SIERRA methods that can be employed to model solid reacting materials:

- A particle reaction capability exists that has been used to model propellant particle reactions. This capability has also been used with non-inertial reactive particles for modeling wildland fires.
- A 1-D solid reacting boundary conditions allows prediction of the behavior of a solid at a boundary interface by constructing heat transport and reactive equations at a 1-D surface [5].
- A 3-D new solid reacting capability exists through coupling with Aria, another SIERRA tool [6]. The Aria Fluid Mechanics capability permits detailed prediction of decomposing materials in a fire environment. A surface coupling exists whereby the Aria and Fuego capabilities can co-exist in the same calculation. This permits the details of the 3-D solid reacting material to be predicted with Aria, while the gas phase reactions, transport and thermal radiation heat transport are predicted with Fuego.

In Aria, the solid degradation capability has been extensively tested and several different mass transfer and heat diffusion models are available. Therefore, coupling Fuego with Aria would offer a better solution for the rubble fire scenario. Due to the time and resource constraints, however, current work utilizes the existing solid reacting features in Fuego.

Regarding the gas/liquid interface model, additional capabilities also exist relevant to this effort. There are methods for modeling liquid fuels. Two such capabilities presently exist:

- Within Fuego, there is a 1-D reacting fuel pool boundary condition. The incident convective and radiative flux are used to predict the energy flux to a surface boundary condition. This is used to predict the evolution of a liquid layer existing on the surface. The quantity of liquid is tracked through a fuel height variable, while the energy balance is managed through some assumptions relating to energy conservation that allow the pool to locally heat in addition to evaporate fuel [7].
- Within Aria, there is a level-set capability to model free surface dynamics of gas/liquid interfaces. This capability employs a surface re-meshing algorithm for localized refinement to resolve the surface with increased resolution to be able to more accurately represent the interface [8].

While two liquid modeling capabilities presently exist, neither is considered particularly well suited to adapt to modeling mixed fuel conditions. The 1-D capability in Fuego sacrifices resolution in the liquid layer, and cannot be expected to resolve a heterogeneous mixture of fuels. It was deemed generally adequate for predicting many fuel fire conditions not involving significant liquid spread. The level-set capability in Aria could be adapted to work for this

problem. However, the local refinement capability is only in Aria and does not presently couple well to the Fuego reaction code. The fluid motion in the stated problem of interest is comparatively subtle, and does not require the detailed resolution that is typically needed for design basis problems. Furthermore, the level-set capability struggles to maintain mass conservation in scenarios that do not maintain high resolution at the surface interface. Since the scale of fires is often tens of meters and the resolution requirements for the level-set capability is less than a millimeter, the mixed liquid and solid fuel problem could become intractable at larger scales due to the resolution costs and requirements.

This report summarizes the efforts to improve existing model capabilities to be able to more appropriately model a fire involving a mix of liquid and solid combustible materials. Solid decomposition models (Chap. 2) and a particle based bulk representation approach (Chap. 3) are presented followed by liquid interface models (Chap. 4). Combination of the developed models do have a potential to replicate the September 2014 rubble fire burning test where 900 lbs of aviation composite materials were burned in a rubble pool fire of 324 gallons of jet fuel at Sandia National Labs [1]. The test used carbon-fiber epoxy composite which is the target material for the solid degradation modeling work.

Chapter 2

Solid Combustion

Two solid decomposition approaches are introduced in this chapter. Implementation and verification of a new model (called Chemical Percolation Devolatilization, or CPD) is presented first followed by a new method to handle generalized chemistry associated with gas-phase tar cracking. Later, solid degradation is represented by a multi-step chemistry which was used to simulate and analyze thermogravimetric analysis (TGA) and radiant panel experiments.

2.1 Chemical Percolation Devolatilization Particle Model

2.1.1 CPD Description and Motivation

The thermal decomposition of solid organic material ranging from coal to biomass (cellulose, etc.) to organic composites represents a challenging class of multi-physics problems. Many organic solid phase materials can be effectively represented as a network of organic molecules (aromatic and aliphatic) joined through thermally decomposable bonds (labile bonds). Fletcher's CPD mechanism [9, 10, 11, 12, 13, 14] provides a tractable way to model the thermal decomposition and devolatilization processes while avoiding the complexity of fully parameterizing the enormous number of specific chemical interactions that actually occur within the material. Microstructural data from nuclear magnetic resonance (NMR) and other experimental methods are used to parameterize the model. This is particularly useful for cases where individual reactions and reaction parameters are either not known or difficult to obtain through experimentation. Fletcher and others [9, 10, 11, 12, 13, 14] have successfully applied CPD to a variety of materials and have found good agreement with experimental results.

2.1.2 CPD Code Development

Fletcher's CPD source code is written in Fortran and is readily available from his website [15]. The full CPD model has been described previously [9] and will not be exhaustively described in this work. Fletcher's CPD code can be used to provide the reaction history of

a thermally decomposing organic particle. Such decomposition includes the evolution of the percolation parameters and relative mass fractions of the particle components due to the thermal history. These mechanisms are dependent on percolation and reaction parameters that are partially determined though NMR spectroscopy and other methods. The CPD code requires a number of input parameters. Percolation parameters include the cluster coordination number/number of bridges per cluster (σ+1), the initial fraction of intact and stable (char) bridges (p_0,c_0), and average molecular weight of a cluster and sidechains (M_{clust}, M_{del}). Reaction parameters include a rate prefactor, an activation energy, and standard deviation of activation energy for bridge breaking (A_b, E_b, σ_b), ratio of rates of bridge breaking to char bridge formation and difference in their activation energies (ρ, E_c), rate prefactor, activation energy, and standard deviation of activation energy for gas release (A_g, E_g, σ_g), frequency factor and activation energy for cross-linking (A_{cross}, E_{cross}). Additional parameters include the pressure (P), and maximum size of n-mers used to specify the tar molecular weight distribution. Tar here is defined as species that condense at ambient temperature and pressure, while it could be in the gas phase at the high-temperature, bridge-breaking moment. As mentioned above, the user also inputs times and temperatures specifying the thermal history of the particle as well as the size of the integration timestep. Output includes the fraction of the initial particle mass that remains or has been converted into char, tar, and gas (f_{char}, f_{tar}, f_{gas}). Also reported are the metaplast fraction, the fraction of intact, labile, and char bridges (p, L, c), and the cross-link fraction.

SIERRA low Mach module Fuego was developed to model a variety of fire environments. It models incompressible, buoyancy driven, turbulent flow using CVFEM (control-volume finite-element method) [4]. Historically, Fuego has been used successfully for simulating a wide range of fluid and fire scenarios including coupled Eulerian(fluid)/Lagrangian(particle) problems in which Lagrangian particles are coupled to the background fluid through the momentum, continuity, and energy equations. Until recently, a limited range of reacting particle capabilities existed in Fuego, including wildland fire particles and single component evaporation.

For the current work, Fletcher's CPD method was incorporated into Fuego as a new reacting particle type. Input to Fuego for the CPD particle type is consistent with Fletcher's CPD implementation but contains additional particle parameters including particle size, position, velocity, density, infrared radiation absorptivity, and specific heat. Fuego assumes homogeneous particles so that the density, absorptivity, and specific heat of a particle are determined through the values of the constituent materials and their relative mass fractions. The user can specify whether particles are to be mobile or immobile. If mobile, particle positions are updated based on momentum coupling to the background fluid, while if immobile, particle positions are static. Particles are thermally coupled to the background fluid so that they can experience a heat flux to or from the background fluid. As a CPD particle devolatilizes, it produces a light gas (species identified by the user) which becomes a source term in the fluid phase. Particle sizes are assumed to be constant. For Fletcher's CPD code, the sole interaction between a particle and the environment is through the temperature time history provided in the user input, which is in contrast with Fuego where fluid temperature is directly solved.

For the sake of this work, we have compared the Fuego implementation of CPD with Fletcher's CPD code. For verification studies, Fuego simulations are initialized with an isothermal environment with boundary conditions having the same temperature. As a result, particle temperatures are "almost" invariant over the course of the simulation. For verification, we have used a number of CPD parameter sets and for each, several temperature values.

2.1.3 Example Fuego CPD input

In order to make use of CPD particles in a Fuego simulation, we are required to specify a number of CPD particle properties. The first of these are the particle materials. We see an example of this below, where we have defined the materials char and tar, including their various thermal properties.

```
BEGIN PARTICLE MATERIAL char
      DENSITY = 1200.e-3
      THERMAL_CONDUCTIVITY = 10.e3
      SPECIFIC_HEAT = 20.05e6
      ABSORPTIVITY = 0.8
END PARTICLE MATERIAL char
BEGIN PARTICLE MATERIAL tar
      DENSITY = 1200.e-3
      THERMAL_CONDUCTIVITY = 40.30e3
      SPECIFIC_HEAT = 50.81e6
      ABSORPTIVITY = 0.6
END PARTICLE MATERIAL tar
```

Next we define a CPD particle interface which contains the various particle reactions relevant to the CPD mechanism [9]. Each reaction indicates the stoichiometric coefficient for the gas phase product for each gas phase species present in the simulation.

```
BEGIN PARTICLE INTERFACE refStuff
      begin particle reaction labile_to_intermed #no gas evolution
      gas species H2O 0.0
      gas species CH4 0.0
      gas species CO2 0.0
      gas species CO 0.0
      gas species O2 0.0
      gas species H2 0.0
      gas species N2 0.0
end particle reaction labile_to_intermed
begin particle reaction intermed_to_side #no gas evolution
      gas species H2O 0.0
```

```
      gas species CH4 0.0
      gas species CO2 0.0
      gas species CO 0.0
      gas species O2 0.0
      gas species H2 0.0
      gas species N2 0.0
end particle reaction intermed_to_side
begin particle reaction side_to_g1 #CO_2
      gas species H2O 0.0
      gas species CH4 0.0
      gas species CO2 1.0
      gas species CO 0.0
      gas species O2 0.0
      gas species H2 0.0
      gas species N2 0.0
end particle reaction side_to_g1
begin particle reaction intermed_to_char_g2 #CH_4
      gas species H2O 0.0
      gas species CH4 1.0
      gas species CO2 0.0
      gas species CO 0.0
      gas species O2 0.0
      gas species H2 0.0
      gas species N2 0.0
      gas species He 0.0
end particle reaction intermed_to_char_g2
```

The CPD parameter set, consistent with Fletcher's Fortran version is then populated:

```
# use as activation energy the parameter Ea/R, which has units of T
REACTION RATE PARAMETER ab = 2.00e16
REACTION RATE PARAMETER eb0 = 27881
REACTION RATE PARAMETER ebsig = 2063
REACTION RATE PARAMETER ac = 100.0
REACTION RATE PARAMETER ec0 = 0.0
REACTION RATE PARAMETER ag = 3.0e15
REACTION RATE PARAMETER eg0 = 30800
REACTION RATE PARAMETER egsig = 4076
REACTION RATE PARAMETER acr = 3.0e15
REACTION RATE PARAMETER ecr = 32712
REACTION RATE PARAMETER p0 = 1.0
REACTION RATE PARAMETER c0 = 0.0
REACTION RATE PARAMETER sigp1 = 3.0
REACTION RATE PARAMETER mw1 = 81.
```

```
REACTION RATE PARAMETER mdel = 22.7
REACTION RATE PARAMETER particleMobile = 0.0 # if > 1.0 , particle is mobile
END PARTICLE INTERFACE refStuff
```

Finally, the CPD_PARTICLE type is defined using the above parameters and materials:

```
BEGIN PARTICLE DEFINITION CPDWC
     PARTICLE TYPE IS CPD_PARTICLE
     ADD PARTICLE MATERIAL char
     ADD PARTICLE MATERIAL tar
     ADD PARTICLE INTERFACE refStuff
END PARTICLE DEFINITION CPDWC
```

The above parameter set is for cellulose as parameterized by Fletcher.

2.1.4 CPD Verification Problems

A number of verification studies have been run with the CPD model against Fletcher's Fortran implementation. We find good agreement. Two such cases are found below in Fig. 2.1, one for Xylan® and the other for cellulose, both at T = 750K. The fractions of char, tar, gas, and the labile bond fraction are plotted over time. Results from Fletcher's CPD Fortran code are shown in black, with Fuego CPD results shown in red. Minor differences occur at late times in the char and gas fractions due to the fact that we currently do not transport tar from the particle to gas phase. Following introduction of the general chemistry gas reaction capability, we plan to implement this transfer and add a subsequent tar to light gas reaction to the gas phase for further combustion. At that point we expect those differences to vanish.

The model is further compared against laminar entrained flow reactor experimental results for cellulose pyrolysis [2]. In the experiment, cellulose particles experience devolatilization while traveling through a wall-heated, helium-filled tube (Fig. 2.2). Particles are fully vaporized for higher wall temperature cases, while conversion to gas phase is limited for low heating (lower temperature) scenarios. The conversion rates from various controller temperatures are compared to the experiment in Fig. 2.3. The CPD model implemented in SIERRA/Fuego offers results close to the experimental data, and superior to several other pyrolysis models. This modeling capability enables higher accuracy and increased physical fidelity predictions of pyrolysis.

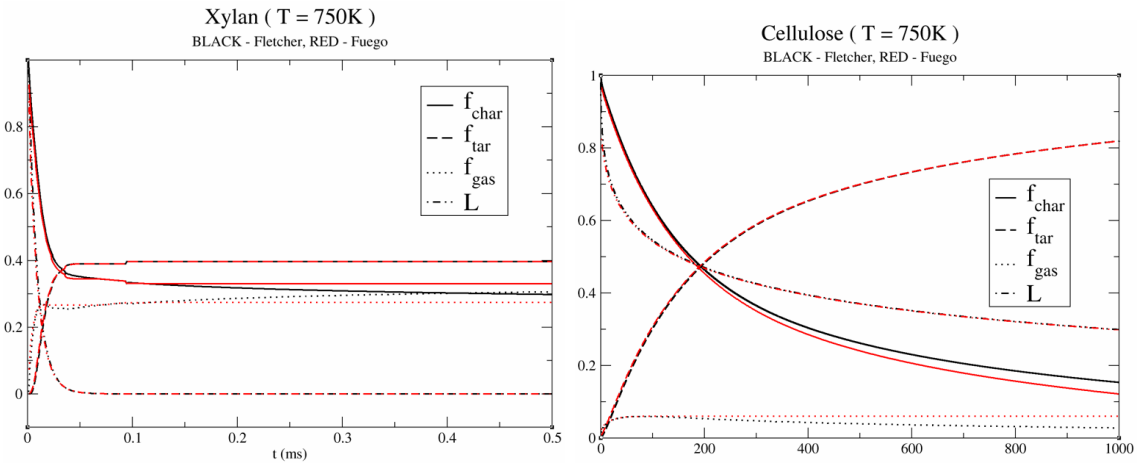


Figure 2.1. A verification model comparison between the stand-alone CPD model and the Fuego implementation.

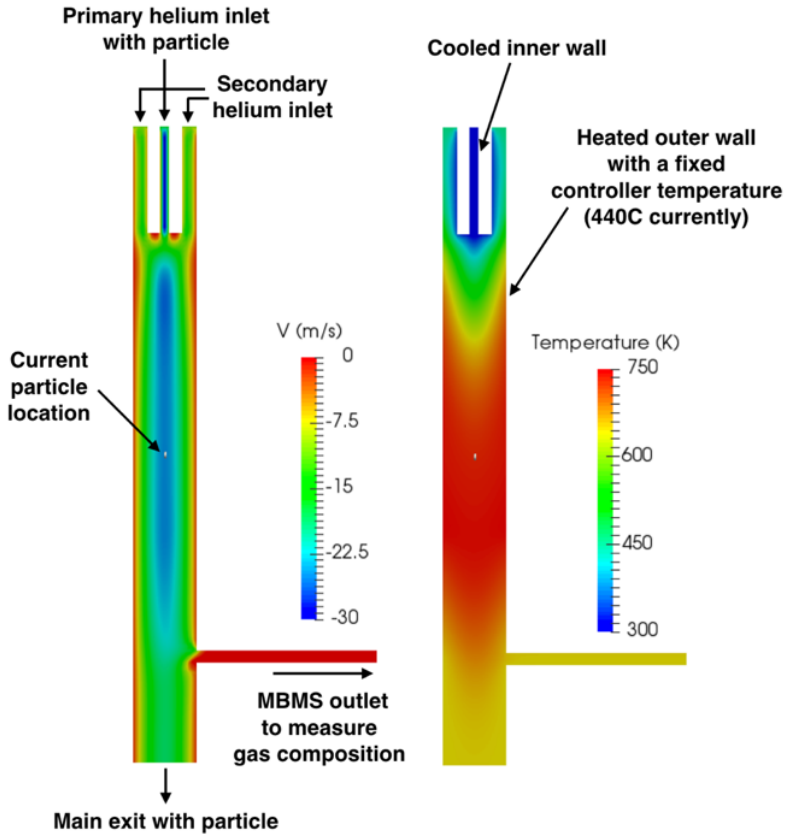


Figure 2.2. Velocity and temperature contours of a laminar flow-entrained reactor used for model verification. MBMS stands for molecular beam mass spectrometer.

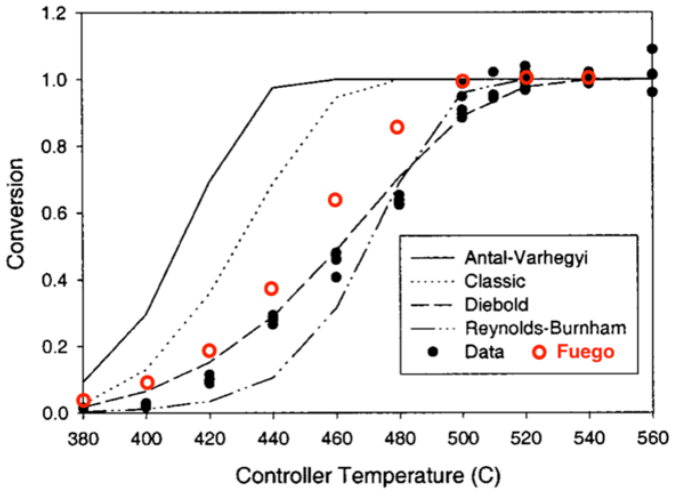


Figure 2.3. Conversion rate from solid to gas and liquid phases for different heating scenarios from Fuego (red circle) compared to the experiment data (dots) as well as other pyrolysis models (various lines). Conversion profiles of experiment and other models are extracted from [2].

2.2 Gas Phase General Chemistry Model and Implementation

2.2.1 Gas Phase Reactions in Fuego

Combustion in Fuego has traditionally been modeled using the EDC (Eddy Dissipation Concept) model, though a more recently developed mixture fraction model is also available [4]. EDC is analogous to a turbulent, mixing-driven perfectly stirred reactor (PSR) where combustion chemical reactions occur only a small fraction of a turbulent control volume [16]. Each control volume in the mesh is divided into two zones, a reaction zone and the surroundings. This reaction zone is considered to be much smaller than the control volume of the mesh element. In EDC, only local properties affect the combustion rate.

Additionally, fast chemistry is assumed in the EDC implemented in Fuego. An arbitrary hydrocarbon fuel is decomposed into CO and H_2 at the first step, while the intermediate species, CO and H_2, are further oxidized in the second step, provided oxygen is still sufficient. This approach models CO, and potentially soot, in the fuel-rich case. However, only a single-fuel species is allowed, which limits our ability to describe gas phase reactions with multiple-fuel components.

To address the need to model other gas phase reactions, we have recently enhanced Fuego's ability to specify Arrhenius and non-Arrhenius type, kinetically limited gas phase chemical mechanisms. The particle region was previously enhanced to allow a similar reaction chemistry mechanism. This new capability allows reactions to exist alongside the EDC reaction set. It enables us to describe specific reactions for the gas phase products of particle phase reactions or composite surface type reactions that produce combustible or otherwise reactive gas phase species. This is particularly useful due to the single fuel limitation of EDC. General gas phase reactions are specified in the Fuego input deck in the following manner with appropriate numerical quantities replacing $*$'s:

```
Begin Chemistry Description MyGasChem
     Begin Reaction R1
          Reaction is a1X1 + a2X2 + ··· + aNXN -> b1X1 + b2X2 + ··· + bNXN
          Rate Function = Arrhenius A = * beta = * Ea = * R = *
          Concentration Function = Standard mu = Automatic
          Heat of Reaction = *
     End Reaction R1
End Chemistry Description MyGasChem
```

In the above description, R1 is the name of the reaction, which can be any valid string. In the line beginning "`Reaction is`", `a1` through `aN` are the Stoichiometric components for the reactant side of this chemical reaction, `X1` through `XN` being the names of the chemical species present (`CH4, O2, N2, CO2, CO,` $\cdots$). `b1` through `bN` are likewise the Stoichiometric

coefficients for the product species. Note that there is no requirement for all species present in the simulation to have coefficients specified on either the left or right side of a specific chemical reaction mechanism. Thus we might specify the burning of H_2 as:

```
Reaction is 1.0H2 + 0.5O2 -> 1.0H2O
```

without any need to specify coefficients for CO, CO_2, or other species that are present in the simulation. The general form for a reaction rate is:

$$\dot{\omega} = f_m(t,T,P,Y) f_r(t,T,P,Y) f_p(P,T) f_c(Y) \tag{2.1}$$

where f_m, f_r, f_p, and f_c are multiplier, rate, pressure, and concentration functions, which have dependencies on the time t, temperature T, pressure P, and species mass fractions Y. Each of these functions is described in detail in the Fuego user manual [4] and will not be exhaustively discussed here. In the above example, the multiplier and pressure functions are unity valued. The concentration function f_c is the "standard" type being the product of the mass fractions of the reactants raised to the power of their Stoichiometric coefficients. The temperature-dependent rate function used in the example above has a general Arrhenius form rate of reaction:

$$f_r(t,T,P,Y) = AT^{\beta} \exp\left(-\frac{E_a}{RT}\right) \tag{2.2}$$

where A is the Arrhenius prefactor, β is the temperature scaling exponent, E_a is the reaction's activation energy, R is the ideal gas constant, and T is the temperature. Several other forms for f_r, f_m, f_p, and f_c are available through the general chemistry mechanism, including the use of a time/temperature dependent prefactor function to increase or decrease the reaction rate at a specified time or temperature, distributed Arrhenius, pressure dependence, and concentration dependence (standard, Prout Tompkins, and evaporation). These are all described in detail in the Fuego user manual. The Heat of Reaction (per mass) is also defined in this section and is used as a source for the Energy/Enthalpy equation. Multiple reaction mechanisms can be designated in an input deck.

Note that the reaction rate in Eq. 2.1 did not include turbulent-chemistry interaction. Use of the general chemistry feature is therefore limited to the cases where a laminar combustion model is applicable. Those cases includes, not limited to, when the control volume is as small as the flame thickness, or if turbulent-chemistry interaction is expected to be small.

2.2.2 Example General Chemistry Scenarios

We demonstrate this generalized reaction capability in the following scenario. For this purpose, we have created a mesh of 33,280 linear hexagonal elements as shown in Fig. 2.4. The full extent of the simulation domain is 1x1x1.25m^3 with a 0.25x0.25x0.25m^3 rectangular

inflow duct impinging from the top (+z). The inflow species is pure N_2 at a temperature of 300K, and the inflow velocity is varied between 0.1 and 1.0 m/s. The simulation domain is initialized as pure O_2 at T = 300K. The side and top boundaries (except for the inflow) are modeled as no slip isothermal wall boundaries at a temperature of 300K. The lower surface of the simulation domain is an open boundary also at T=300K. The one-equation, KSGS turbulence model has been utilized for this scenario. A 4th order Discrete Ordinates PMR solver (Thurgood quadrature) with 128 ordinate directions is used to solve for the transport equation of the thermal radiation. The emissivities of all boundaries are set to 1.0.

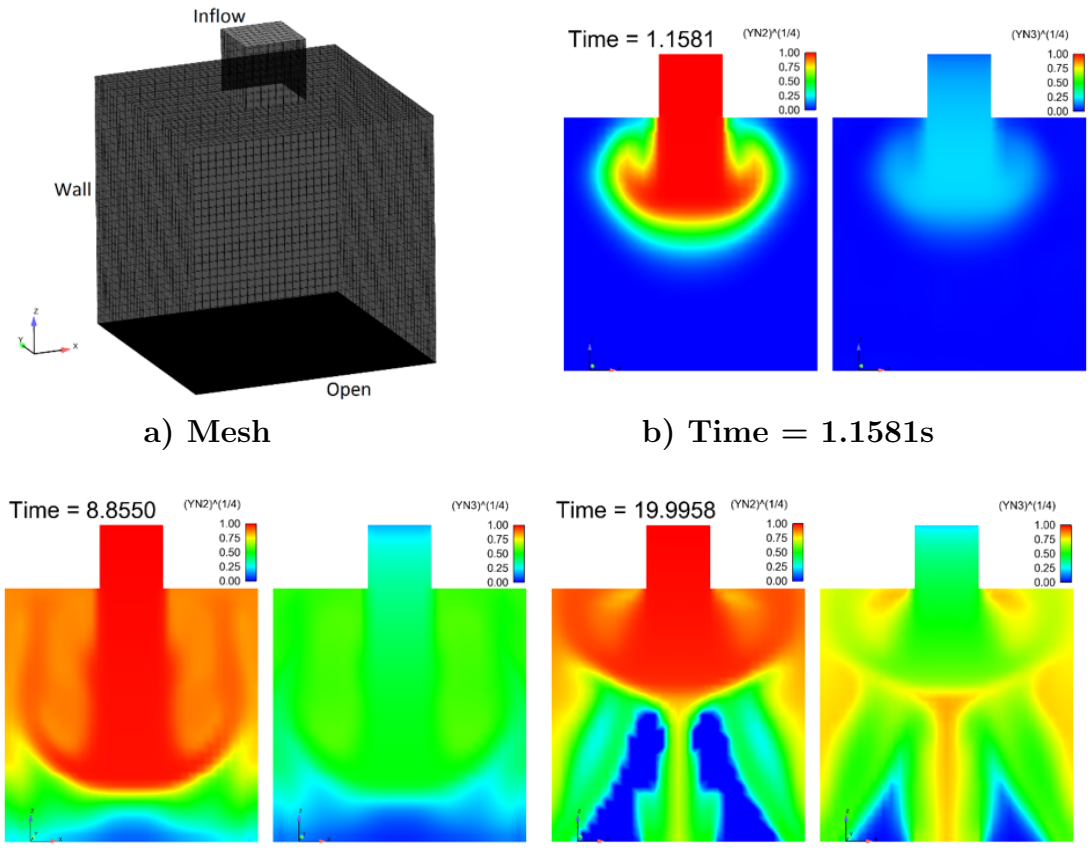
In our example, we utilize a single general chemistry reaction:

```
Begin Chemistry Description MyGasChem
      Begin Reaction R1
            Reaction is N2 → N3
            Rate Function = Arrhenius A = 3.34e30 Ea = 2.2714e5 R = 8.314
            Concentration Function = Standard mu = Automatic
            Heat of Reaction = 0.0
      End
End
```

Here, we are using a standard Arrhenius form as shown above with $\beta = 0$. For clarity, this first case has a simple reaction mechanism which converts the inflow N_2 into a fake species N_3 which has been generated with identical thermal properties to N_2. These properties are found within a chemistry xml file that must be present to run a Fuego simulation with chemistry. In Fig. 2.4, the inflow velocity is 1.0 m/s, and we show the concentration of N_2 and N_3 on a vertical cross section plane through the domain normal to the y axis at various simulation times. Again for clarity, we color this cross plane by the 1/4 power of either $Y(N_2)$ and $Y(N_3)$. This scaling is used rather than $Y(N_2)$ and $Y(N_3)$ directly due to the higher visibility of low mass fraction of N_3 during the early stages of the simulation. Clearly, as the N_2 inflow propagates across the domain, N_3 is gradually formed due to the specified reaction. Additionally, at late stages we see that there are regions where N_2 has previously been at a higher concentration but has reacted away to form N_3, now at higher concentration (Fig. 2.4).

A couple of additional scenarios are tested to lend confidence to the implementation. For the next scenario (Fig. 2.5), we keep the mesh unchanged, but modify the inflow to be 0.1 m/s instead of 1.0 m/s. We also reduce the Arrhenius coefficient A to 3.34e10 (a factor of 10^{20}). Additionally, we change the temperatures of the side walls and lower open boundary to 600K. The results are similar to that of the previous figure except that the N_2 jet now easily transports across the entire simulation domain. Again, we see N_3 production occurring where N_2 concentrations have been high.

To demonstrate the effect of the Arrhenius rate prefactor A on the rate of production of N_3, we take this second scenario and vary the value of A between 3.34e9 and 3.34e10. In Fig. 2.6 we observe the mass fraction of N_3 at two times in the simulation. For both times,

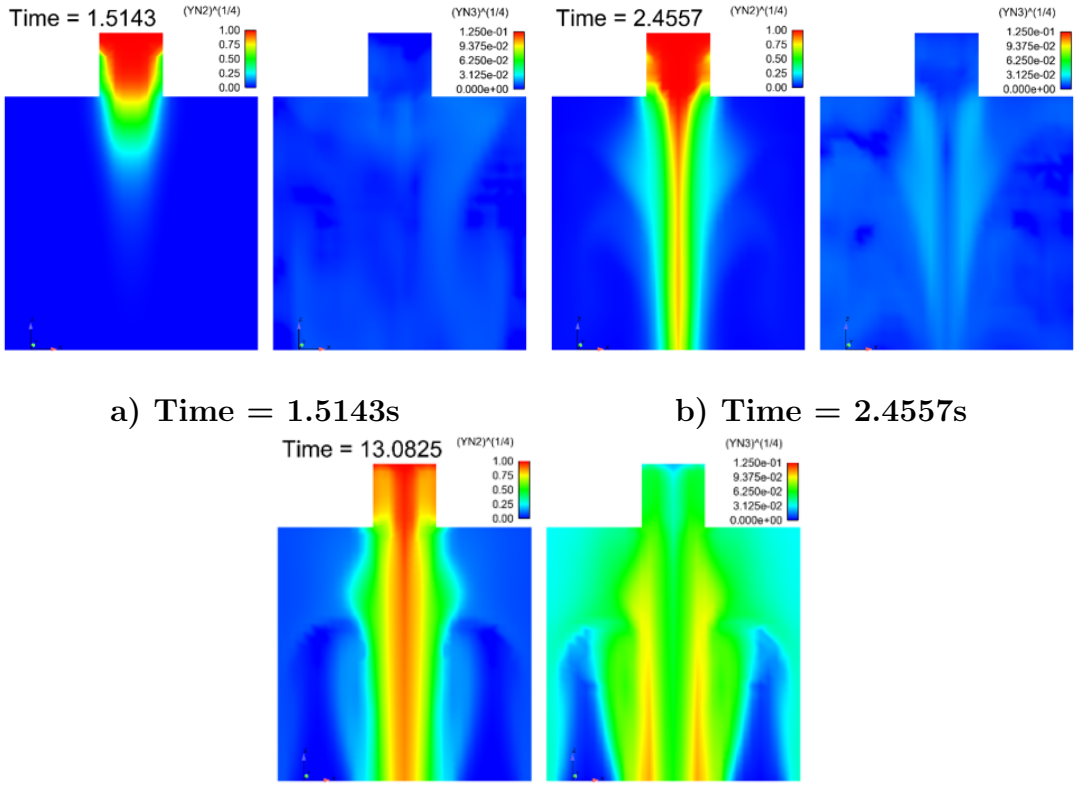


a) Mesh **b) Time = 1.1581s**

c) Time = 8.8550s **d) Time = 19.9958s**

Figure 2.4. a) Mesh for general chemistry reaction example. b)-d): Mass fractions (Y) of N_2 and N_3 species at three different times.

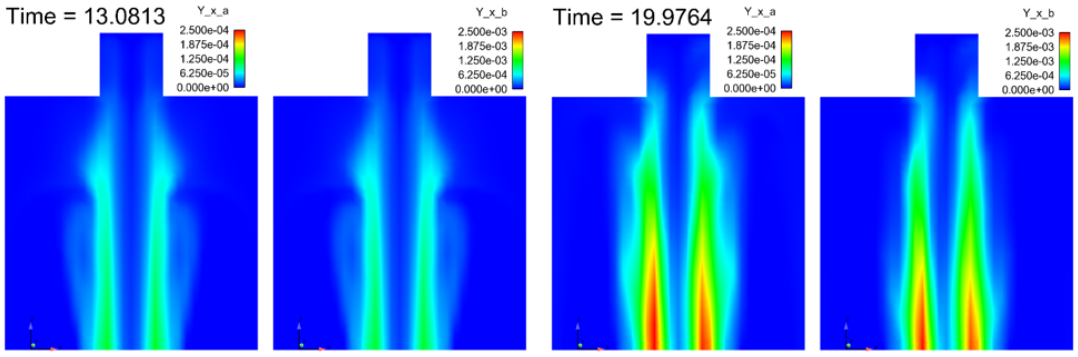
the mass fractions of N_3 for the two different values of the Arrhenius coefficient are visually indistinguishable, though the scales vary by a factor of 10, just as the values of A do, as one would predict.



a) Time = 1.5143s **b) Time = 2.4557s**

c) Time = 13.0825s

Figure 2.5. Second General Chemistry Reaction simulation with similar conditions to that of Fig. 2.4, but lower inflow rate (0.1 m/s) and higher wall temperatures (600K).



a) Time = 13.0813s **b) Time = 19.9764s**

Figure 2.6. Same conditions as scenario in Fig. 2.5, but with rate prefactors that vary by a factor of 10. Results shown at two simulation times.

2.3 Multi-step Decomposition Mechanism for Carbon-Fiber Epoxy

The CPD model was implemented on several composite materials such as polyurethane foam and removable epoxy foam [17, 18]. Different from single component CPD models, bond structure is complex and bond-breaking procedure needs to be described in several steps for composite materials. Furthermore, side chains (initially broken out structures) are heavily non-uniform and their later stages cannot be modeled using the same approach in the single component CPD. To develop a CPD mechanism, a full understanding of the detailed chemical structure and pyrolysis steps is required. Epoxy pyrolysis is typically modeled proceeding the following steps: 1. dehydration, 2. low-temperature volatilization due to weakened C-O bond and unsaturated C-C, 3. high-temperature volatilization due to breakdown of C-phenyl bonds of bisphenol-A [19]. Based on this procedure, a detailed Arrhenius-based pyrolysis mechanism was developed [20]. To develop a CPD model, however, one needs more knowledge such as molecular structures and bond statistics for each process. An alternative approach, or an approximated multi-step representation for the decomposition, is used in this chapter.

Different from the CPD model, single- or multi-step representation of a decomposition does not need any structural background or known pyrolysis procedure of a solid material. That becomes useful for the current problem where curing material and composition detail of the tested carbon-fiber epoxy is unknown. In general, single- or multi-step Arrhenius based decomposition chemistry is derived from TGA results. Recently, several TGA tests using carbon fiber-epoxy were published [3, 21], along with fitted pyrolysis mechanisms. The fitted reaction mechanism from [3] is used for simulating various reactive composite scenarios in this chapter.

Material conversion is modeled on two scenarios: on a panel located on the domain boundary and on a Lagrangian particle. Conversion into a different material or a different phase occurs following a set of reaction rates listed in Table 2.1 which includes one pyrolysis and four oxidation steps. Note here that for both phases, mass concentration is used for rate calculations whereas general gas-phase reaction mechanisms involve molar concentration. Moreover, oxidation reaction rates in Table 2.1 are presented in unconventional form that involves different phases. For instance, when both solid and gas phases participate in a reaction as reactants, the pre-exponent factor is not a constant as it varies with gas density. If a reaction mechanism involves a 3rd reactant, the pre-exponent factor needs further modification reflecting the density of the 3rd reactant. This is a common issue when the mechanism is developed using mass fractions separately defined in each phase [3, 21].

In Fuego, Eqns. 2.3 and 2.4 are the methane and epoxy evolution equations. In the equations, $\dot{\omega_{EDC}}$ is gas-phase combustion source, and ρ_g and ρ_s are density in gas and solid phases, respectively.

$$\frac{\partial \rho_g Y_{CH_4}}{\partial t} + \nabla \cdot (\rho_g \mathbf{u} Y_{CH_4} - \rho_g D \nabla Y_{CH_4}) = \dot{\omega_{EDC}} + 0.5\dot{\omega_1} + \dot{\omega_2} \tag{2.3}$$

Table 2.1. Pyrolysis and oxidation mechanism and rates for carbon-fiber epoxy [3].

Mechanism	A (1/s or m^3/kg-s)	E_a/R (K)	ΔH (kJ/kg)	Equation
Epoxy $\rightarrow$ 0.5CharA + 0.5CH_4	3.33E+15	27200	0	(2.5)
Epoxy + O_2 $\rightarrow$ CharB + CH_4	5.30E+15/ρ_g	27200	0	(2.6)
CharA + O_2 $\rightarrow$ Residue + CO	7.58E+02/ρ_g	10000	12730	(2.6)
CharB + O_2 $\rightarrow$ Residue + CO	7.58E+02/ρ_g	10000	12730	(2.6)
Carbon-Fiber + O_2 $\rightarrow$ Residue + CO_2	3.79E+15/ρ_g	38000	24770	(2.6)

$$\frac{d\rho_s Y_{epoxy}}{dt} = -\dot{\omega_1} - \dot{\omega_2} \tag{2.4}$$

$\dot{\omega_k}$ is the reaction source from the kth mechanism in Table 2.1, which is in either Eq. 2.5 or Eq. 2.6 depending on the number of reactant phases. $Y_{r,s}$ and $Y_{r,g}$ are the mass fractions of the reactant solid material and the reactant gas phase species, respectively. A is the pre-exponent factor, E_a is the activation energy, R is the gas constant, and T is the temperature.

$$\dot{\omega_1} = \rho_s Y_{r,s} A e^{-E_a/RT} \tag{2.5}$$

$$\dot{\omega_{2-5}} = \rho_s Y_{r,s} \rho_g Y_{r,g} A e^{-E_a/RT} \tag{2.6}$$

Note that some reactions are exothermic. ΔH in Table 2.1 alters the temperature of the solid material. Density of the carbon fiber-epoxy is $1779 kg/m^3$, and the emissivity is assumed as 0.85. We model the exothermic reaction by adding the energy to the solid phase.

2.3.1 Single Particle Evolution

TGA tests were used to calibrate the pyrolysis and oxidation mechanism of carbon fiber-epoxy [3, 21]. The test was replicated in a full 3D fluid calculation to understand interactions of the sample with surrounding flows. A non-movable composite particle sample with a diameter of $50\mu m$ was placed in a 4cm×2cm×2cm domain (Fig. 2.7). The mesh size was 2mm. The air stream velocity was fixed at 2cm/s and the temperature slowly increased until the sample was depleted. For simplicity, gas density is at 1 atm and 700^{o}C, and was used to correct the pre-exponent factors in Table 2.1. Heat release from the oxidation was neglected, which is an acceptable assumption since TGA is designed to occur in an isothermal system (slow heating).

A full two-way coupling with the gas-phase was included for the study, and a point-particle assumption was used. Since the particle represents a tiny sample in this study, uniform properties are assumed throughout the particle.

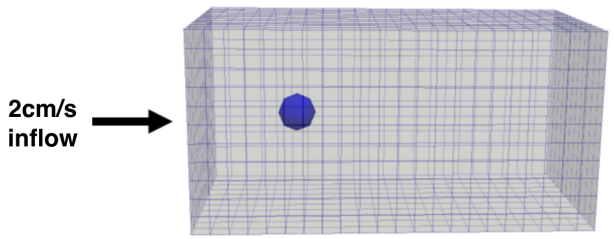


Figure 2.7. The mesh for the single particle TGA calculation. Particle size is exaggerated.

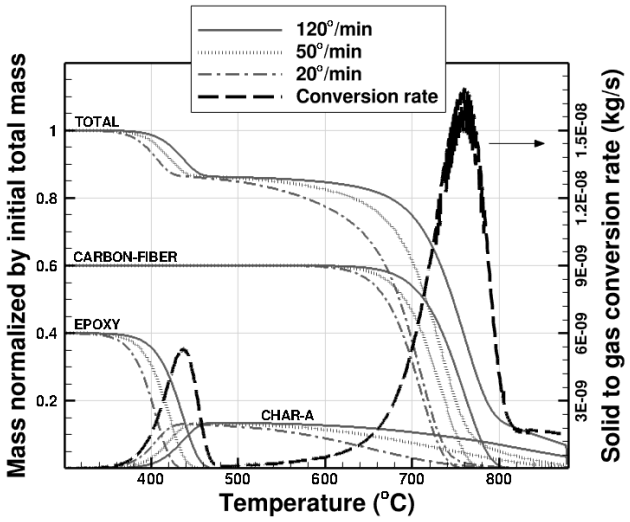


Figure 2.8. Predicted mass changes of the single particle evolution for different heating rates. Conversion rate is from the fastest heating rate case.

Figure 2.8 shows the predicted mass change of the material being consumed or generated during pyrolysis. As shown in Fig. 2.9, the profile closely matches the reference in [3] since the same mechanism is used. Before reaching 450^{o}C, epoxy pyrolysis creates char A and epoxy oxidation generates char B (only char A is plotted in Fig. 2.8). The carbon-fiber and the chars oxidize at a substantially higher temperature. Pyrolysis response to different heating rates is correctly predicted as plotted in Fig. 2.8. In particular, both pyrolysis and oxidation are delayed as the heating rate increases. The phase conversion rate is three times higher in the oxidation process than the initial epoxy pyrolysis for the fastest heating rate case (120^o/min).

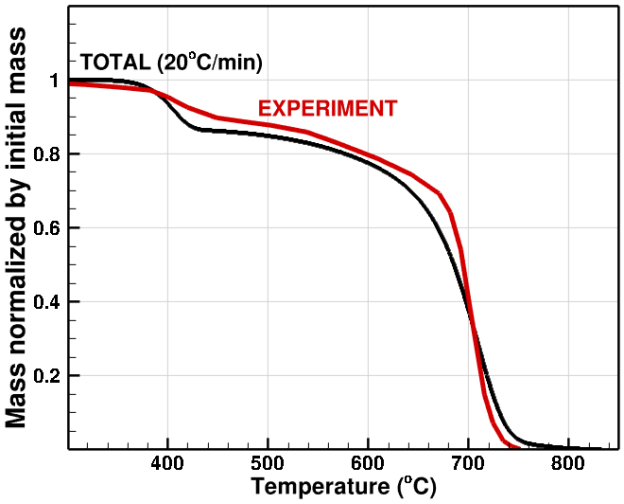


Figure 2.9. Total mass change compared to the experiment data from [3]

2.4 Composite Panel under Radiant Heat Source

The multi-step degradation mechanism in the previous section is applied on the uni-directional boundary condition (named 1D BC modeling in Fuego [4]) to simulate a solid panel exposed to strong fire-like radiant heat (from an experiment by Hubbard et al. [22]). In the panel fire simulation, the pyrolyzed fuel components participate in gas-phase combustion using a turbulent combustion model. Panel backside temperature and pyrolysis mass are compared to test results [22].

In the 1D modeling approach, thickness, density, and other properties of the composite panel are given as initial conditions. Temperature and material composition profiles inside the panel are modeled by dividing the material into a given number of nodes. As decomposition takes place, the composition changes and mass transfer between gas and solid phases occurs. Surface regression is not considered in the 1D modeling approach. Details of the model are available in Brown et al. [5].

The domain shown in Fig. 2.10 is a simplified representation of an experiment performed by [22]. A composite panel with a size of 0.102m×0.076m was located in the middle of the bottom surface. The thickness of the panel was 3.2mm. Radiative heat flux originated from a hot 0.184m×0.184m surface on the top. The distance between the heat source and the composite panel was 0.14m. An adiabatic condition was assumed for all the other walls, including the back side of the panel. The mesh size was 5mm and the total number of elements was approximately 100,000. This effort demonstrates coupling capabilities between flow fields, heat transfer, pyrolysis, and radiation. Temperature variation in the panel depth direction was ignored and the combustion model was turned off.

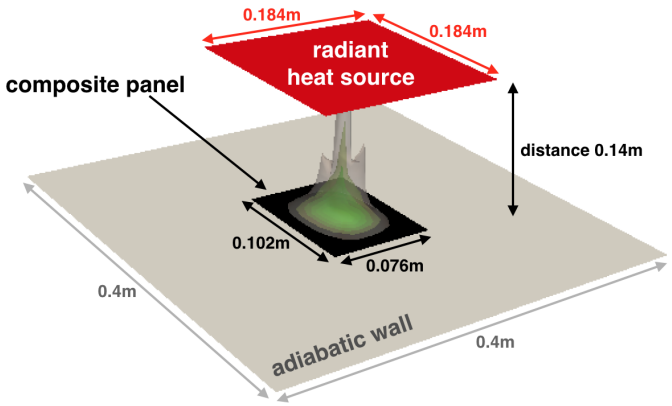


Figure 2.10. The computational domain for the composite panel case. Omitted here is the wall at the top surface around the heat source. Isocontours of a decomposed gas from the panel is shown in the middle.

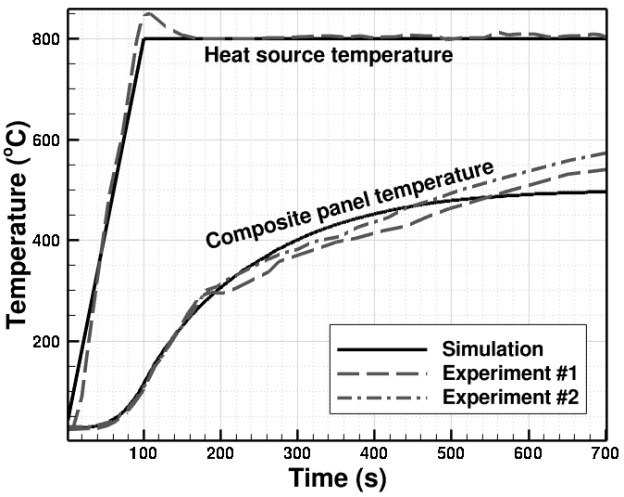


Figure 2.11. Temperature profiles of the controlled heat source and the heated panel.

2.4.1 Panel under a Radiant Heat Simulation

In the panel under a radiant heat simulation, the temperature of the heat source (as shown in Fig. 2.11) follows close to the experimental condition. The simulation started from zero flow velocity. The conversion mass rate and temperature profile at the back side of the panel are plotted in Figs. 2.11 and 2.12. As radiant heat flux increases, the panel begins to pyrolyze until all of the epoxy is consumed (Fig. 2.12). Oxidation, however, is very slow, resulting in minimal conversion rate once epoxy pyrolysis is over. As shown in Fig. 2.13, lack of oxygen near the panel significantly delays oxidation process. Instead, produced gas such as CH_4, CO, and CO_2 are temporarily accumulate near the composite panel (Figs. 2.13 and 2.14). Moreover, the panel does not reach high enough temperature for an active oxidation.

The back side temperature profile compares well to two experiments with thermocouple temperatures that were measured under the same condition [22] (Fig. 2.11). It is reported that visible off-gassing occurred between 165s and 660s in one experiment (marked as #1 in Fig. 2.11), and between 100s and 520s in the other (#2 in Fig. 2.11). Note that experimental variability related to non-uniform composite resulted in different results, such as #1 and #2 profiles in Fig. 2.11, even with the identical experimental condition. The simulation shows higher mass flow rate (10% of peak conversion rate, or 1.2E-5kg/s) between 240s and 400s, which begins later and finishes earlier than the experiment (Fig. 2.12). The conversion that continues after 400s appears weaker, although it is not clear whether the small conversion rate such as 1E-5kg/s would be visible in the experiments. Reported panel shape changes (such as swelling) and their effects were ignored in the simulation [22, 23].

Burning was not observed in the experiment. Since the simulation predicts ignition as soon as the gas-phase reaction model is turned on, combustion was turned off for the

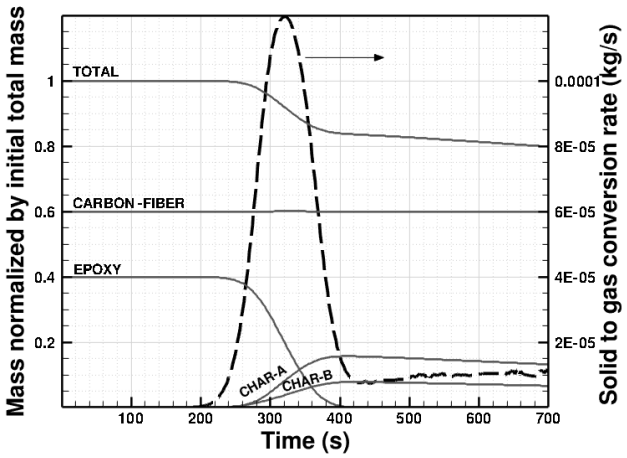


Figure 2.12. Panel mass changes and conversion rate with a radiant heat source and quiescent flow.

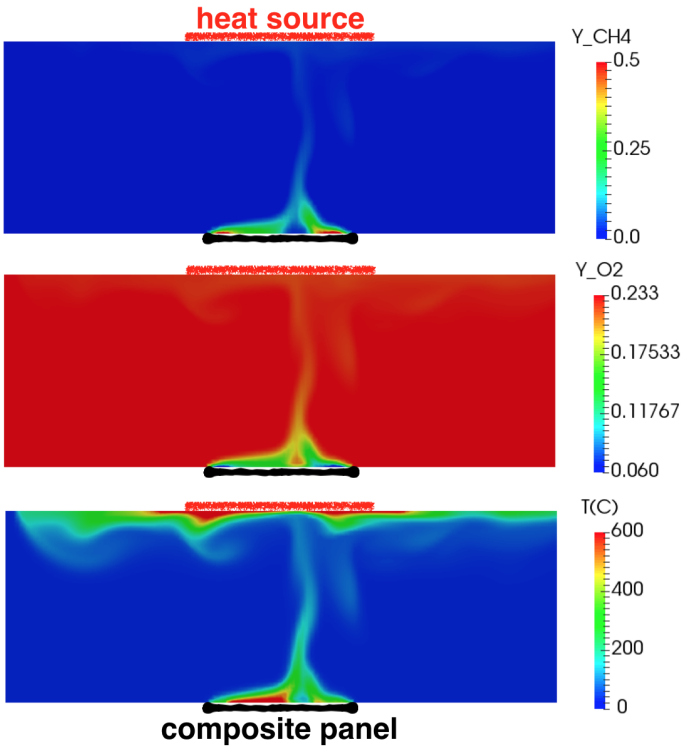


Figure 2.13. (Top to bottom) Methane mass fraction, oxygen mass fraction, and temperature contours of the center plane at 310s.

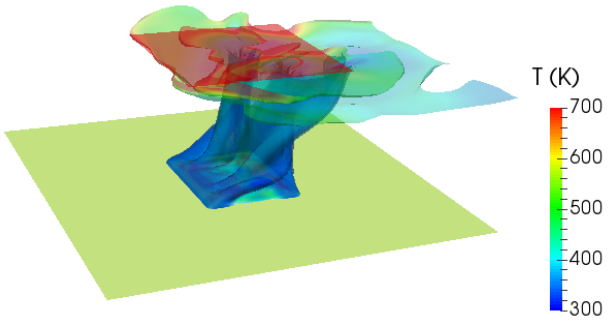


Figure 2.14. Isocontour of sum of the produced gas (leveled at a mass fraction of 0.025) at 310s, colored by temperature.

calculations in this chapter. Poor ignition predictability along with the lack of tar cracking attributes to the discrepancy. Note that the pyrolysis and oxidation mechanisms focus on the solid-phase response while generated gas-phase species were not measured.

2.4.2 Panel Exposed to Slowly Heating Air

The panel was exposed to a slow heating rate by controlling temperature of the global flow. The setup is similar to the TGA case except that the composite is now exposed only on one side in this study while all sides of the particle are exposed to the gas in TGA. A thinner panel (0.32mm) was used for this calculation and a rather fast 2m/s horizontal flow was imposed. Temperature of the flow was increased by 120^o/min. The result is plotted in Fig. 2.15. Compared to the single particle calculation (Fig. 2.8), epoxy pyrolysis and oxidation initiation occur at similar temperature ranges. However, the composite panel oxidizes much slower than the composite particle. Similar to the radiant heat case, the delay is attributed to oxygen blockage above the panel due to the gas generated from pyrolysis and oxidation. Due to the slower oxidation, peak conversion rates between epoxy pyrolysis and oxidation are comparable (Fig. 2.15).

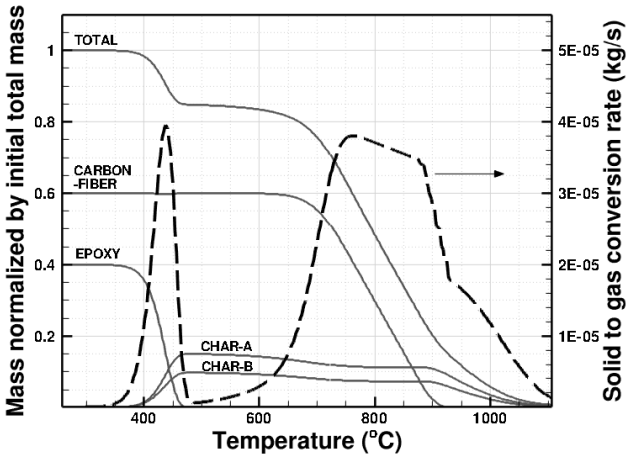


Figure 2.15. Mass change of the composite panel under a slow heating rate, 120^{o}C/min, and a faster flow, 2m/s.

Chapter 3

Extended Particle Capabilities

In this chapter, recent particle modeling is presented to explore the possibility of extending Fuego's Lagrangian particle feature to represent thermal interactions within a solid bulk. Rubble pieces are expected to be described by a group of Lagrangian particles once all the relevant models are developed.

SIERRA low Mach module Fuego [4] includes the capability to model coupled Lagrangian particle/ Eulerian fluid simulations where coupling between the two phases occurs through various physical mechanism. These include the momentum, energy, and species (chemistry) equations. Until recently, Lagrangian particles in Fuego were non-interacting, meaning that particles did not exert forces or exchange thermal energy with other particles except indirectly through the Eulerian background fluid. We have recently enhanced this capability by adding interaction terms for both momentum and energy equations that allow particles to exert forces and conduct heat to other particles in a pairwise fashion. We describe the interparticle thermal interaction model below. This capability relies on the interaction model previously introduced to Fuego under a different project which included force interactions between particles. The force interactions will not be described in detail here but are described in Brown and Pierce [24].

3.1 Lagrangian Inter-particle Thermal Conduction

In order to allow for transport of thermal energy through a bed or aggregate of interacting particles, we have recently added a model of thermal conduction between contacting particles in Fuego. This particle-particle thermal contact is idealized in Fuego in the following manner. Each particle within a set of interacting Lagrangian particles possesses a set of different "sizes" with uses that depend on the physical mechanism of interest. For the purposes of momentum coupling to the Eulerian fluid (drag), Fuego uses the particle diameter defined in the input deck or the particle insertion file. For Lennard-Jones force interactions, the Lennard-Jones diameter (σ above), is used. For thermal considerations, particles have an additional thermal contact diameter that serves to define a thermal contact area with other particles. This diameter tends to be larger than the Lennard-Jones diameter by a factor on the order of unity which facilitates attractive Lagrangian particles coming into contact with an equilibrium thermal contact area. The factor of proportionality between the nominal

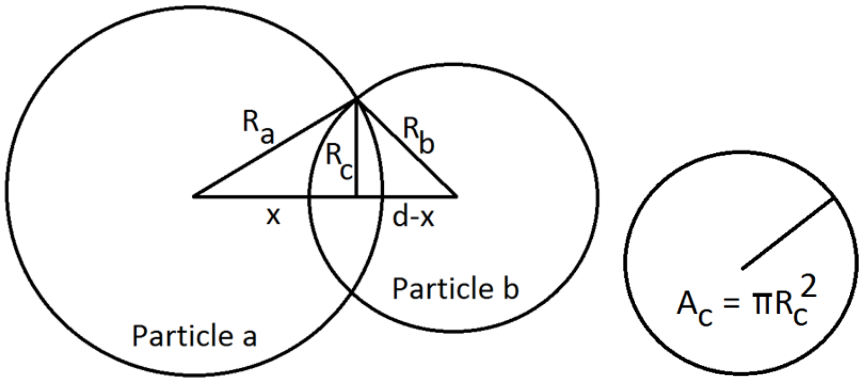


Figure 3.1. Thermal contact between two Lagrangian particles of thermal radii R_a and R_b separated by a distance d. Simple geometry defines a contact circle of area A_c and radius R_c. In this Figure, x is the distance between the center of particle a and the contact circle, d-x the distance between particle b and the contact circle.

particle diameter and the thermal contact diameter can be set by the user, as seen in the Thermal Contact Diameter Multiplier in the Fuego input example shown earlier. Figure 3.1 demonstrates this scenario. Two particles, a and b, are in positions that allow thermal overlap since the separation between the particle centers is less than the sum of the thermal radii of the two particles (R_a and R_b). Figure 3.1 shows a cross-sectional view through the center of the particles. The separation distance between the particle centers is d. Due to this configuration, a contact circle is generated with a radius of R_c and cross-sectional area A_c normal to the line between the centers of the spherical particles as seen in Fig. 3.1. In this scenario, x is the distance between the center of particle a and this contact circle. In this geometry, we find:

$$x = \frac{d^2 + R_a^2 - R_b^2}{2d} \tag{3.1}$$

with

$$R_c^2 = R_a^2 - x^2 \tag{3.2}$$

and

$$A_c = \pi R_c^2 \tag{3.3}$$

The thermal conduction rate between the particles is then:

$$P_{a,b} = k_a k_b A_c \frac{T_a - T_b}{k_a(d - x) + k_b x} \tag{3.4}$$

where k_a and k_b are the thermal conductivities and T_a and T_b are the temperatures of particles a and b, respectively. In Fuego, Lagrangian particles have a single scalar temperature. Notice that the thermal power $P_{a,b}$ has the correct limiting behavior. If the two particles are at nearly the same temperature, $T_a \approx T_b$, the thermal rate is $P_{a,b} \approx 0$. If the particles do not significantly overlap, the contact area $A_c \approx 0$ and $d \approx R_a + R_b$ so that we also have $P_{a,b} \approx 0$. In the limit that x << d, that is $R_a << R_b$, we find $P_{a,b} \approx k_b A_c (T_a - T_b)/d$, indicating that the thermal transfer rate is dominated by conduction through the material in particle b, as expected. For each pair of particles a and b in thermal contact, a transfer of thermal power $P_{a,b}$ occurs at each computational timestep.

3.2 Example Particle Interaction Simulation

In Fig. 3.2, we show the results of a simulation of Lagrangian particles that exhibit both force and thermal interactions as defined above. Results are shown at five different times throughout the simulation. In this scenario, a spherical droplet composed of 935 heated inertial particles of diameter d = 1cm, σ = 1cm, ϵ = 2erg, $U_0 = 3\epsilon$, and $r_0 = 0.9\sigma$, and $d_{thermal}$ = 1.3d is placed at the center of the cubical computational domain. Particles are initially placed at rest at nearest-neighbor distances equal to the minimum of the LJ potential ($2^{1/6}\sigma$) and arranged in positions consistent with an FCC (face-centered cubic) lattice where each internal particle has when there are 12 nearest neighbors. This process creates a droplet that is already at *nearly* equilibrium density without any significant interparticle stress to relax. The Eulerian fluid mesh is composed of 1000 hexagonal elements and is 25x25x25 cm^3 in volume. Particles are composed of a single material with the following properties defined in the particle material section of Fuego input file:

```
BEGIN PARTICLE MATERIAL material1
      DENSITY = 791.3e-3
      THERMAL_CONDUCTIVITY = 150000.0
      SPECIFIC_HEAT = 4.184e5
      ABSORPTIVITY = 0.75
      FILM_PRANDTL_NUMBER = 10.0
      VISCOSITY = 1.0
END PARTICLE MATERIAL material1
```

All material parameters are in cgs units, consistent with the interaction parameters. A number of surface particles have been initialized to a temperature of $T_{p,hot}$ = 1000K (red) with the balance of the particles initialized to $T_{p,cold}$ = 300K. The Eulerian fluid begins at an isotropic temperature of $T_{fluid,i}$ = 350K. The lateral and top surfaces of the domain are open boundary conditions at 0 thermodynamic pressure and with mass fractions of nitrogen Y_{N2} = 0.77 and oxygen Y_{O2} = 0.23 (airlike). The simulation volume is initialized with the same concentrations. The lower surface is a no-slip wall (0 fluid velocity) with a fixed temperature of T_{wall} = 350K. Gravity is directed in the -x direction and is set to 1g = -9.8 m/s^2.

Several features regarding particle motion can be noted. As indicated above, due to the initial relative placement of particles, little residual stress is present in the droplet, so that while particles move relative to one another throughout the simulation as expected for a liquid droplet, the droplet volume is nearly unchanged as the droplet falls toward the lower surface. On contact with the lower boundary, the droplet deforms and rebounds, also expected for a liquid droplet. At this impact velocity and due to surface tension resulting from the particle interactions, the droplet stays intact instead of shattering. Of course, features below the size of the constituent particles cannot be resolved with this model.

The thermal response of particles in the droplet is also apparent. The initial distribution of hot surface particles (1000K) as seen in Fig. 3.2a) has resulted in a conduction of heat

through part of the droplet by 1.0s which continues to proceed throughout the droplet as the simulation time advances, including during and after the impact of the droplet with the lower boundary. For the duration of the simulation, particles are also thermally coupled to the background Eulerian fluid, which stays approximately steady at T=350K. At the final time shown (9s), the droplet is nearly isothermal at a temperature of $T_{p,final} = 350K$.

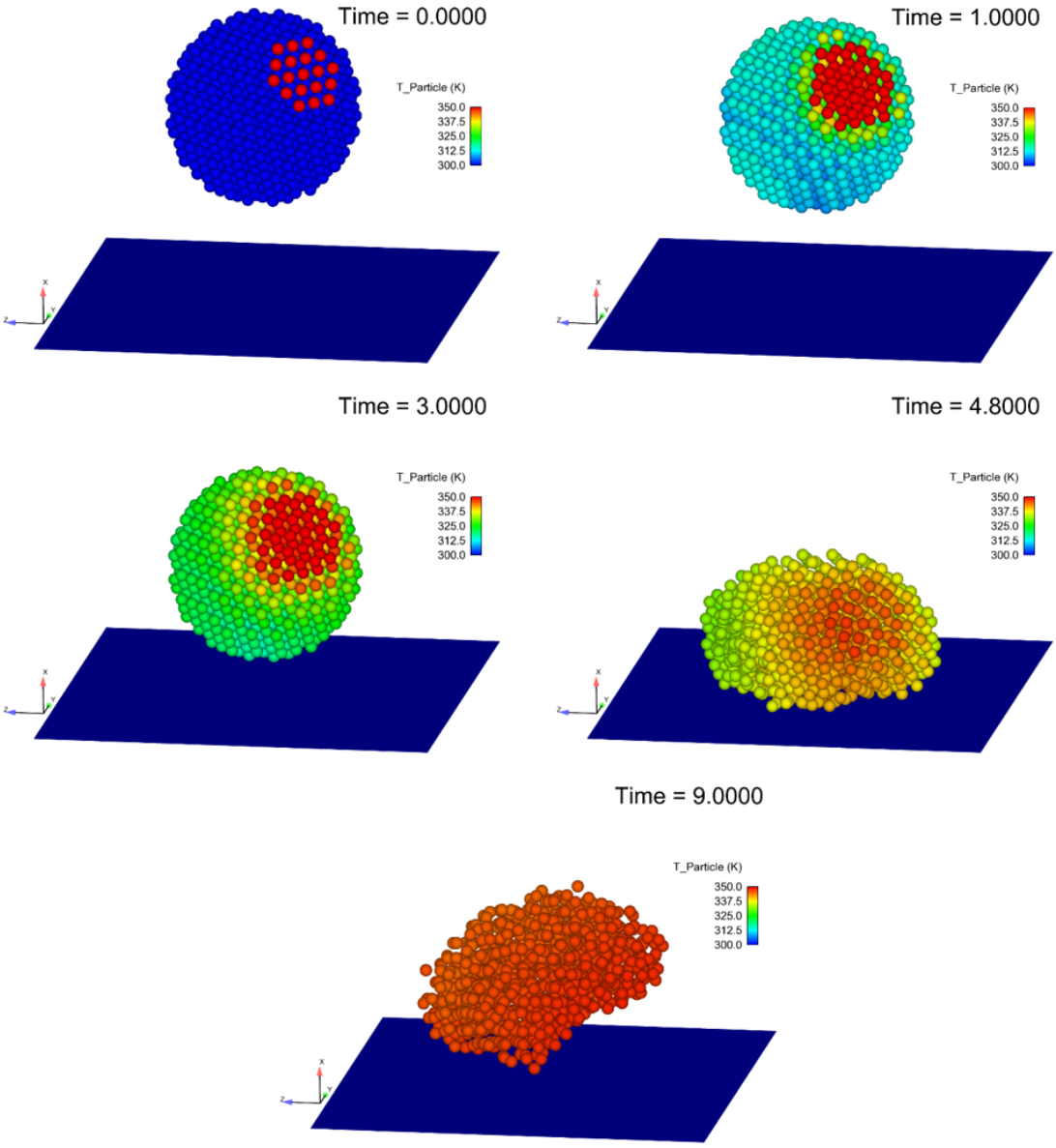


Figure 3.2. (a-e) Droplet of 953 interacting spherical particles. Force interactions are governed through the modified LJ potential. Thermal transport is controlled by the thermal transport equation above for $P_{a,b}$. Particles are shown at various times throughout the simulation. Individual particles are colored by their temperature.

Chapter 4

Liquid-Gas Interphase Model

A liquid-gas interphase model is essential for simulating a rubble pool fire scenario. As discussed in the introduction chapter, neither of the two existing liquid modeling capabilities in Fuego is well suited for mixed fuel conditions. The Volume Of Fluid (VOF) interface model is therefore implemented in Fuego. In this chapter, implementation details and several verification tests of the VOF capability are presented.

4.1 Volume of Fluid

4.1.1 Governing Equations

The basic advection equation for the volume fraction (α) in VOF equations is

$$\frac{\partial \alpha}{\partial t} + u \cdot \nabla \alpha = S_\alpha \tag{4.1}$$

This can be expressed in corrected conservative form as

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (u\alpha) = \alpha(\nabla \cdot u) + S_\alpha \tag{4.2}$$

where the right hand side term is 0 for an incompressible flow with no phase change. The standard advection operator is typically too diffusive for practical use, so one of two options are typically used:

1. A geometric advection operator [25, 26, 27, 28]

2. An additional interface compression term [29, 30]

Several varieties of geometric advection operators have been proposed in the literature [25, 26, 27, 28], many of which are only applicable for 2D quadrilateral meshes. Those that can be extended to higher dimensions often do so at considerable complexity. For a more

detailed discussion of the different options, see [31]. The form implemented here uses the interface compression approach to modify the governing equation to be

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (u\alpha) + \nabla \cdot (u_c \alpha(1-\alpha)) = \alpha(\nabla \cdot u) + S_\alpha \tag{4.3}$$

where u_c is the compressive velocity. A common form for this is

$$u_c = C_\alpha |u| n \tag{4.4}$$

$$n = \frac{\nabla \alpha_s}{|\nabla \alpha_s| + \epsilon} \tag{4.5}$$

where α_s is a field that varies smoothly away from the interface and C_α is a constant. The simple approach to use $\alpha_s = \alpha$ results in considerable noise in the interface normal vector calculation, particularly in regions of high curvature relative to mesh sizes. In the coupled level set VOF approach, or CLSVOF, α_s is the companion level set field that is either advected along with the volume fraction or redistanced from the volume fraction contour at 0.5.

For this work we implement both a CLSVOF approach and a diffusive approach, where α_s is a diffusively smoothed version of the volume fraction field, α, using a user-specified Fourier number (Fo) and number of iterations.

$$\alpha_s^0 = \alpha \tag{4.6}$$

$$\alpha_s^{n+1} = \alpha_s^n + Fo \Delta x^2 \nabla^2 \alpha_s^n \tag{4.7}$$

When phase change is present, the source term S_α is non-zero. Nominally, the volume source of liquid is

$$S_\alpha = \frac{-\dot{m}_{evap}}{\rho_L} \tag{4.8}$$

but when the gas and liquid have different densities, this also introduces a velocity divergence so the overall source term becomes

$$S_\alpha = \frac{-\dot{m}_{evap}}{\rho_L} + \alpha \dot{m}_{evap} \left(\frac{1}{\rho_g} - \frac{1}{\rho_L} \right) \tag{4.9}$$

The evaporation term is calculated using the method described by Hardt and Wondra [32] which converts an evaporative mass flux to a volume source using the interphase area, A, and the mass flux predicted by the evaporation model, m''_{evap}, as

$$\dot{m}_{evap} = m''_{evap} A \tag{4.10}$$

$$A = W|\nabla\alpha| \frac{\int |\nabla\alpha| dV}{\int W|\nabla\alpha| dV} \tag{4.11}$$

$$W = \alpha \tag{4.12}$$

One variation of this that we explored is $W = (1-\alpha)^2$, which is well suited for a diffuse interface approach. We find that the form by Hardt and Wondra [32] is sometimes destabilizing to the diffuse interface and leads to an artificially high surface regression rate. For a very simple 1D problem or a sharp interface method, the $W = (1-\alpha)^2$ weighting scheme places all the evaporation outside the liquid and does not correctly predict the surface regression rate. However, it can be better for a diffuse interface representation since evaporation acts as a stabilizing liquid sink to counteract interface diffusion. When using a sharp interface method or a 1D problem where there is no interface diffusion, the weighting of [32] ($W = \alpha$) should be used.

Different evaporation models can be used to define m''_{evap}. The model proposed at [32] uses deviation from the saturation temperature to define the rate as

$$m''_{evap} = K(T - T_{sat}) \tag{4.13}$$

$$K = \frac{2\beta}{2-\beta} \frac{h_{fg}}{\sqrt{2\pi R}} \frac{\rho_g}{T_{sat}^{1.5}} \tag{4.14}$$

This form is only physically applicable to either fully saturated gasses, or interface temperatures above the boiling point since it does not take into account the gas composition. As such, this model only allows positive evaporation rates (so no condensation) in order to produce realistic behavior during heat-up and away from the hot regions of the simulation. When using a multi-species gas, the more complete Hertz-Knudsen-Langmuir model can be used, where the rate is driven by the difference between the partial pressure of the vapor and the saturation pressure. This model will be added to the available evaporation models once the multi-species gas treatment is complete.

4.1.2 Pressure Stabilization

The standard order pressure correction algorithms used in Fuego are not compatible with density ratios greater than about 10. For VOF simulations, a density ratio of at least 1000

is required. In the original form, the pressure correction equation (for incompressible fluids) is

$$\nabla \cdot (\overline{D_p} \nabla p) = \nabla \cdot \dot{m} \tag{4.15}$$

where D_p is a pressure diffusion term, often selected as the time step, and overbar quantities are interpolated linearly from nodes to faces. Another way of considering this equation is to define the continuity-satisfying mass flux as

$$\dot{m}_f = \overline{u_{est}\rho} + \overline{D_p \nabla p} - \left(\overline{D_p^{-1}}\right)^{-1} \nabla p_f \tag{4.16}$$

The velocity estimate here, u_{est}, is obtained by solving the momentum equations, and is not guaranteed to be continuity-satisfying. The addition of the first two terms provides a version of the "pressure-free" velocity estimate. The method most commonly used in VOF [33] includes the density in the pressure diffusion term, D_p, which we will refer to as A_p when it includes density.

$$\nabla \cdot (\overline{A_p} \nabla p) = \nabla \cdot u_{est} \tag{4.17}$$

For this application, the 4th order stabilization (which includes the pressure term in the momentum equations and adds the projected nodal gradient, $\overline{D_p \nabla p}$) is problematic because the 2-way interpolation near the sharp density gradients causes velocity oscillations on either side of the phase interface. The pressure gradient is interpolated from nodes to faces, then for this work, we use the 2nd order stabilization scheme with the pressure-free momentum equation.

Additionally, instead of time step scaling ($A_p = \Delta t/\rho$), we use momentum scaling, $A_p = 1/|D_u|$, where D_u is the diagonal term from the momentum equations. These changes allow the use of density ratios up to at least 1,000,000 (the highest tested) without exhibiting the instabilities present in the original form.

4.1.3 Additional Forces

When considering additional body forces, such as surface tension or gravity, the above equation is modified to

$$\nabla \cdot \left(\overline{A_p} \nabla p\right) = \nabla \cdot (u_{est} + u_F) \tag{4.18}$$

where u_F is the velocity correction due to the applied surface tension force [34]. For surface tension, this velocity correction is

$$u_F = \overline{A_p}\overline{(\gamma\kappa)} \nabla \alpha_f \tag{4.19}$$

where the curvature is defined by the divergence of the interface normal vector

$$\kappa = \nabla \cdot n \tag{4.20}$$

This could also be written as an adjustment to the pressure gradient, as

$$\nabla \cdot \left[\overline{A_p}\left(\nabla p - \overline{(\gamma\kappa)}\nabla\alpha_f\right)\right] = \nabla \cdot (u_{est}) \tag{4.21}$$

Gravity forces are included in the momentum equation, and show up here as a part of the velocity estimate term, u_{est}.

4.1.4 Phase Change

The effect of phase change on the velocity divergence must also be included in the pressure correction equation. Adding this in gives the full equation as

$$\nabla \cdot \left[\overline{A_p}\left(\nabla p - \overline{(\gamma\kappa)}\nabla\alpha_f\right)\right] = \nabla \cdot (u_{est}) + \dot{m}_{evap}\left(\frac{1}{\rho_g} - \frac{1}{\rho_L}\right) \tag{4.22}$$

4.2 Verification Tests

4.2.1 Advection Tests

There are a number of commonly used advection tests for various VOF methods described in the literature [35, 36]. These tests do not involve surface tension or coupling strategies between the VOF equation and momentum/continuity solves.

4.2.2 Planar Advection

The simplest test is the advection of a sharp interface plane down a channel with a uniform velocity. These tests show the diffusiveness of different advection operators by measuring the interface width over time (defined as the distance between the 0.025 and 0.975 contours of α). A snapshot from this test is shown in Fig. 4.1, comparing the effect of the interface compression scheme with an upwind advection operator. The interface widths of all the methods tests are shown in Fig. 4.2, where it is clear that the compression scheme is needed to stabilize the interface width as it moves. In the un-compressed schemes the interface width continues to grow unbounded, although at a slower rate with the higher order MUSCL advection scheme. The most diffuse interfaces begin to leave the domain after about 160 time steps, which is the cause of the decrease in interface width.

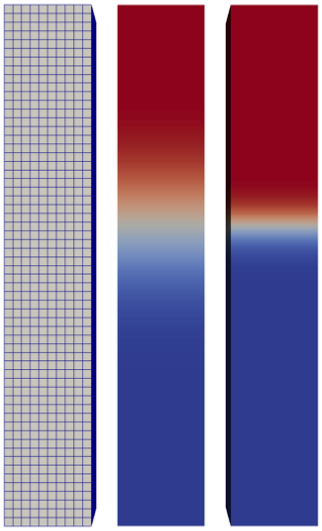


Figure 4.1. Advected VOF field with upwind advection (center) and compression (right).

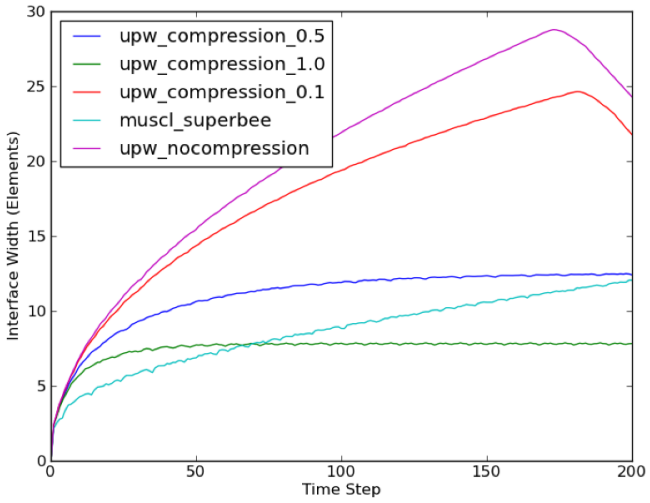


Figure 4.2. 1D interface width with different advection operators. Purple (no compression) and green (compression 1.0) lines correspond to the contours shown in Fig. 4.1.

4.2.3 Circle Advection

The next common set of advection tests involve using a fixed velocity field to advect a prescribed shape and determine how well the shape is maintained and how well mass is conserved. The domain for this test is a 2x2 square (SI units) with a 200x200 mesh and a CFL number ($u\Delta t/\Delta x$) of 0.25 for 800 time steps. The shape advected is a circle with a radius of 10 mesh elements. The velocity is in the <1,1> direction with a constant magnitude of 1 m/s.

For comparison, the level set results are shown using a pseudo-VOF field reconstructed from the level set (LS) field (ϕ) with a tanh operator as

$$\alpha_{LS} = \frac{1}{2}\left[\tanh\left(\frac{\phi}{\Delta x}\right) + 1\right] \tag{4.23}$$

For this advection test and the following tests, we compare the new SIERRA/Fuego VOF implementation against the SIERRA/Aria level-set implementation using streamwise upwind control volume stabilization (SUCV) and level-set redistancing [4, 6].

Table 4.1. Translating circle test results

Code	Method	Final Mass Loss	Results
Aria	CVFEM LS with SUCV	34.2%	Fig. 4.3
Aria	CVFEM LS Redistanced	1.87e-3%	Fig. 4.4
Fuego	VOF UPW Compressed	1.74e-5%	Fig. 4.5

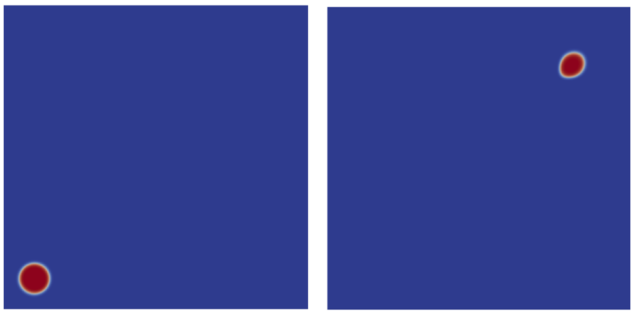


Figure 4.3. Pseudo-VOF field reconstructed from CVFEM level set advection at time steps 0 and 800

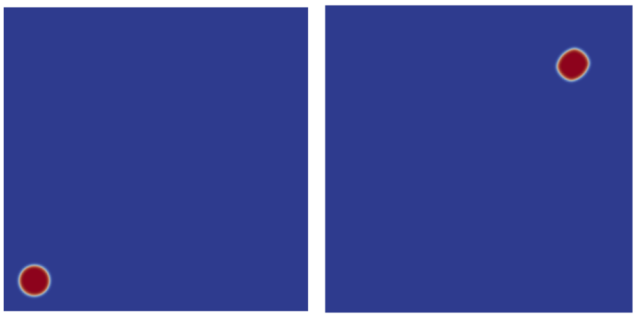


Figure 4.4. Pseudo-VOF field reconstructed from redistanced CVFEM level set at steps 0 and 800

Figure 4.5. VOF field with compression at time steps 0 and 800

4.2.4 Hollow Square Advection

The domain for this test is a unit square (SI units) with a 200x200 mesh and a CFL number of 0.25 for 500 time steps. The shape advected is a hollow square with a constant velocity field of <2,1>, and the size of the square is 40x40 elements with a 20x20 element hole (as described at [36]).

Similar to the previous test case, conserving mass and maintaining shape are two biggest concerns. The shape error is defined to quantify the shape change for a given VOF field. To calculate the shape error, the difference in VOF fields between the calculated solution and the exact solution is normalized by the initial solution (Eq. (9) in [36]).

There are two sources of shape error in these tests: interface distortion and interface diffusion. Assuming a tanh-shaped interface profile, the error due to interface diffusion can be approximated as

$$E_{diff} \approx 0.757 \frac{wP}{4A} \tag{4.24}$$

where w is the width of the diffuse interface, P is the shape perimeter, and A is the shape area. The 0.757 pre-factor is an approximation from the integration of the tanh function.

For the hollow square test with an interface width of 6 elements, this value is 0.23. This is the primary difference between the diffuse interface methods used here and the geometric methods reported at [36]. Since we use a diffusion interface approximation for the level set, the same level of error in that result can be attributed to the diffuse interface.

The results of this test are shown in Table 4.2 and Fig. 4.6. The compressed VOF scheme has the lowest final shape error of the methods compared, and the lowest rate of error growth. Also, the mass conservation of the VOF method is several orders of magnitude better than even the redistanced level set approach.

Table 4.2. Hollow square test results. *Note*: the MUSCL case was diffuse enough that some fluid diffused out through the boundaries, which is why its mass loss is higher than most VOF results

Code	Method	Final Mass Loss	Final Shape Error	Results
Aria	CVFEM LS with SUCV	16.3%	57.6%	Fig. 4.7
Aria	CVFEM LS Redistanced	2.38e-3%	80.2%	Fig. 4.8
Fuego	VOF UPW Compressed	4.17e-6%	32.3%	Fig. 4.9
Fuego	VOF MUSCL+SuperBee^{Note}	1.18e-2%	113.9%	Fig. 4.10

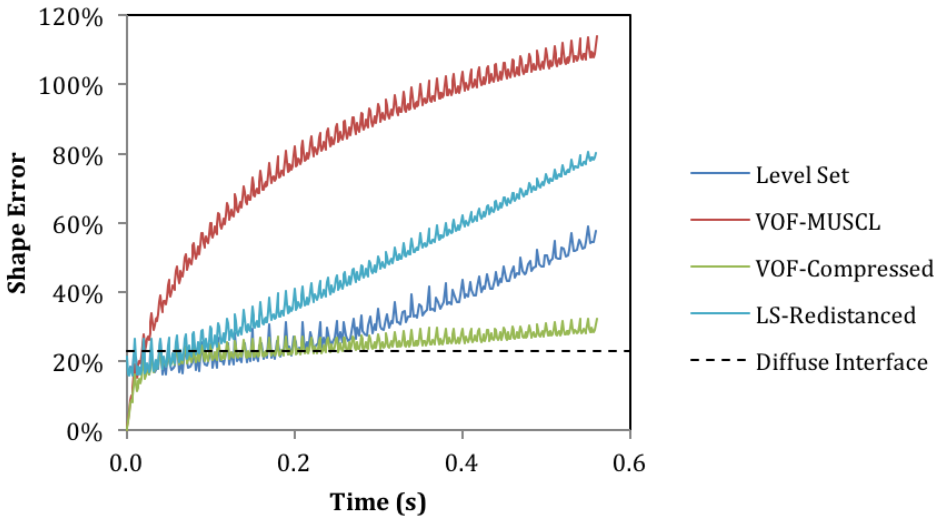


Figure 4.6. Shape error of hollow square test over time, compared with the estimated error from the diffuse interface treatment (23%). Four lines correspond to the cases listed in Table 4.2.

Figure 4.7. Pseudo-VOF field reconstructed from the CVFEM level set solution at time steps 0 and 500

Figure 4.8. Pseudo-VOF field reconstructed from the re-distanced CVFEM level set at time steps 0 and 500

Figure 4.9. VOF result with interface compression at time steps 0 and 500

Figure 4.10. VOF result with MUSCL+SuperBee advection at time steps 0 and 500

4.2.5 Shearing Advection - Reversing Vortex Test

The final type of advection test involves a divergence-free shearing flow and has been used by many authors, but was originally described by Rider and Kothe [37]. The prior tests used shear-free velocity fields, which is not necessarily representative of real flows. This test uses a unit square (SI units) domain with a 100x100 mesh and a velocity field of

$$u =< \cos(\pi x)\sin(\pi y), -\sin(\pi x)\cos(\pi y) > \tag{4.25}$$

which reverses sign after 1000 time steps. The circle has a radius of 20 elements and is initialized with its center 25 elements below the domain centroid. For a perfect advection scheme, the initial condition should be returned exactly after 2000 time steps. The CFL number is 0.25 based on the maximum velocity in the domain (1 m/s) giving a time step of 2.5 ms. The calculation results are outlined in Table 4.3.

The results of these tests are shown in Table 4.3 and Figures 4.11-4.15. While the non-redistanced level-set runs have the lowest shape error, their mass loss is significantly higher than the other methods. On the other hand, the VOF methods have mass loss orders of magnitude lower than the redistanced level-set approach, which also incurred more significant shape distortion as a result of the redistancing.

Table 4.3. Reversing vortex test result

Code	Method	Final Mass Loss	Final Shape Error	Results
Aria	CVFEM LS with SUCV	4.88%	6.69%	Fig. 4.11
Aria	GFEM LS with SUPG	4.71%	6.22%	Fig. 4.12
Aria	CVFEM LS Redistanced	1.17e-2%	35.4%	Fig. 4.13
Fuego	VOF UPW Compressed	2.7e-6%	10.85% (E_{diff}(Eq. 4.24)$\sim$9.4%)	Fig. 4.14
Fuego	VOF MUSCL+SuperBee	2.4e-7%	113.1%	Fig. 4.15

Figure 4.11. Pseudo-VOF field reconstructed from CVFEM level set result at 0, 1000, and 2000 time steps

Figure 4.12. Pseudo-VOF field reconstructed from GFEM level set result at 0, 1000, and 2000 time steps

Figure 4.13. Pseudo-VOF field reconstructed from CVFEM redistanced level set result at 0, 1000, and 2000 time steps

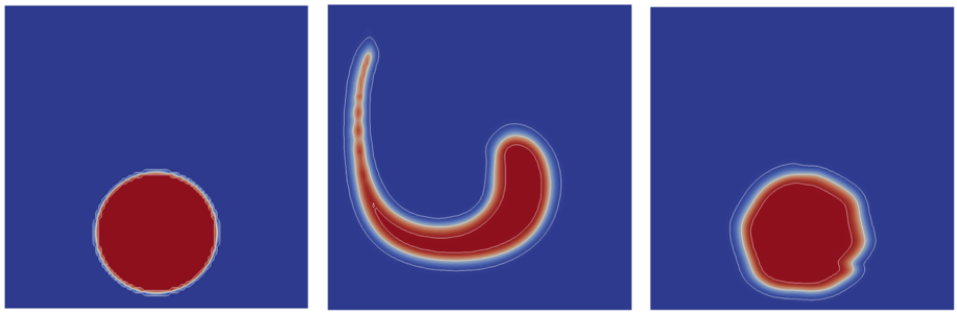


Figure 4.14. Compressed VOF results at 0, 1000, and 2000 time steps (white contours at 0.025, 0.5, and 0.975 isosurfaces)

Figure 4.15. VOF with MUSCL+SuperBee advection at 0, 1000, and 2000 time steps

Mesh convergence was tested using the reversing vortex test above at a constant CFL number and at a constant time step (Table 4.4). This means that both the mesh spacing and time step were changed simultaneously and the number of time steps increased accordingly. The computed order of convergence is 1.02 (Fig. 4.16), which is consistent with the upwind advection scheme used. Results of different mesh resolutions are shown in Fig. 4.17.

Table 4.4. Mesh convergence studies for reversing vortex test

Mesh	Time Step	CFL	Final Shape Error	Results
100x100 (0.01m)	2.5ms	0.25	10.85%	Fig. 4.17a)
200x200 (0.005m)	1.25ms	0.25	5.29%	Fig. 4.17b)
400x400 (0.0025m)	0.625ms	0.25	2.64%	Fig. 4.17c)

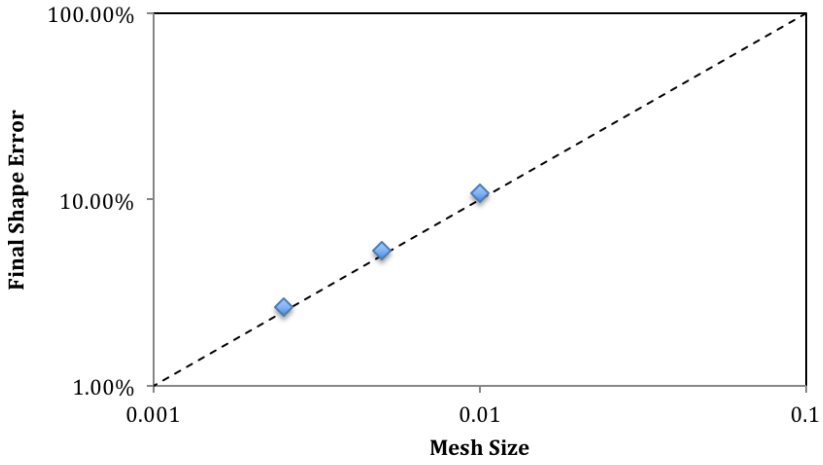


Figure 4.16. Mesh convergence study, showing line for 1st order convergence. Three points correspond to the cases listed in Table 4.4.

The clear conclusion from these advection tests is that the compressed upwind advection scheme is the best performing of those examined here. As such, all subsequent tests use this advection scheme.

Figure 4.17. Final circle shape with a) standard, b) medium, and c) fine meshes

4.3 Modified Pressure Stabilization

4.3.1 Drop in a Channel Flow

The nominal test case for variable density ratios is a drop in a channel flow. This is done without surface tension or gravity forces, so the drop is deformed by the flow around it and moves based on the force applied by the gas flow. This test was conducted up to a density ratio of 1,000,000 to test for stability. Results at a density ratio of 1000 are shown below, where the channel inlet velocity was 0.5 m/s and the channel dimensions were 15 cm by 15 cm by 40 cm. The drop initial radius was 3 cm. The volume fraction of fluid in this case is initialized using a stair-stepped approach, which produces the rough initial surfaces seen in the first image. A better approach, which has been implemented since this work was performed, is to use a planar intersection algorithm to initialize the drop based on where the specified sphere intersects the mesh control volumes. An example from this series of tests with a density ratio of 1000 and the compressed upwind advection scheme is shown in Fig. 4.18, where the white isosurfaces for the drop are drawn at the volume of fluid contour of 0.5. The metrics for success in these tests were: transfer of inertia to the drop and an appropriate reduction when the drop density was higher; stability and boundedness of the volume fraction field; and stability of the pressure field (which is tightly coupled with stability of the velocity field).

4.3.2 Falling Drop Integration Test

We selected a falling drop test as an integration test involving gravity, surface tension, and inertial transfer between multiple liquid entities. The fluid properties used in the test were constant and equal to those of water and air at ambient conditions (Table 4.5). The domain is a 5 cm cube with an 8 mm diameter drop initialized with its center 3.5 cm above the surface of a 5 mm deep pool, with a 50x50x50 hex mesh and an automatic time step set by a CFL number limit of 0.5. Since this is an integration test rather than a validation test, the main metrics for success were formation of the drop splash crown, propagation of the crown into a wave, and reflection of the wave off the domain walls. A sequence of images showing these events in the test are shown in Fig. 4.19.

Table 4.5. Falling drop test fluid properties

Property	Gas	Liquid
Density (kg/m^3)	1	1000
Viscosity (Pa-s)	1.98e-5	1e-3
Surface Tension (N/m)	0.07	0.07

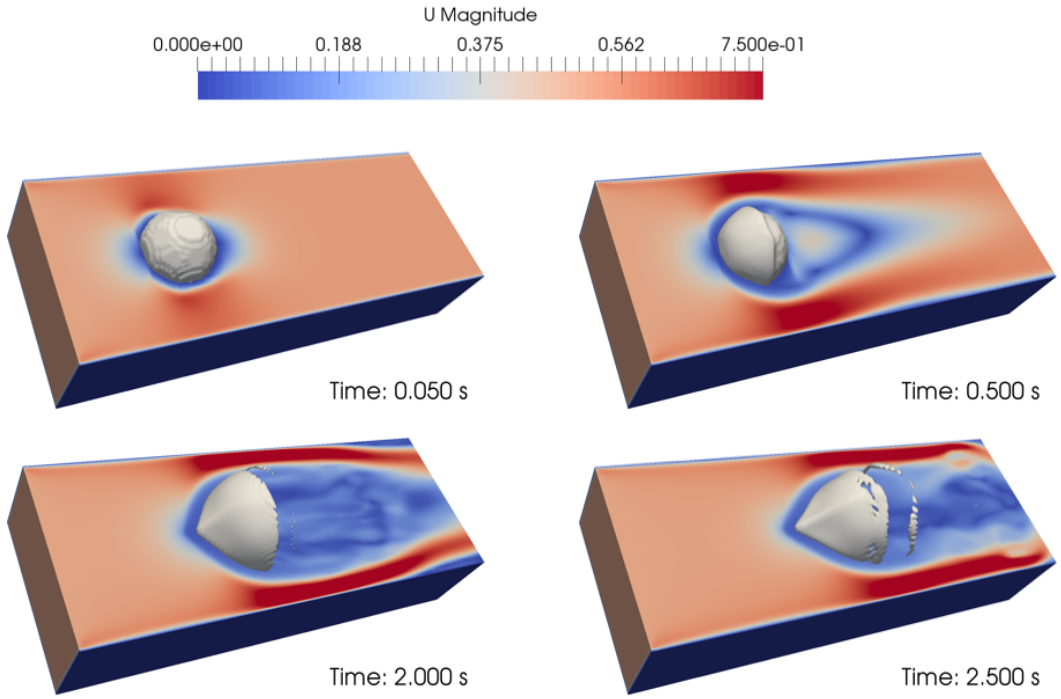


Figure 4.18. Isosurface at the volume of fluid contour of 0.5 for different times for the drop channel test at a density ratio of 1000

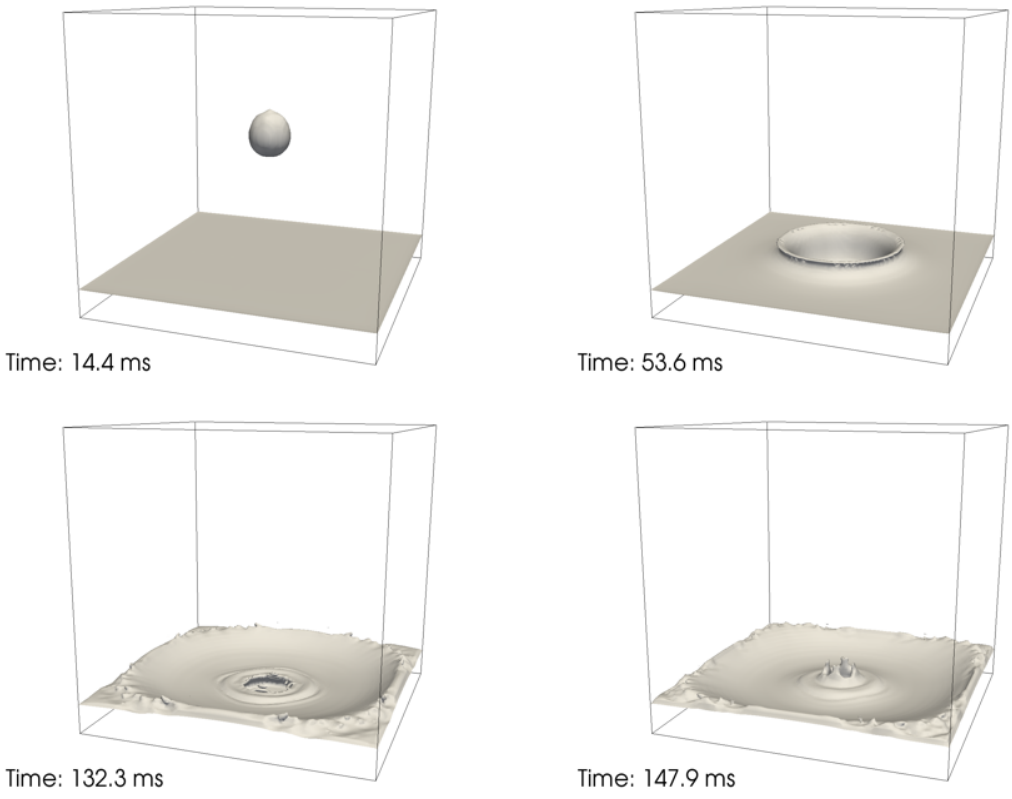


Figure 4.19. Progression of volume fraction of liquid in pool splash test showing splash crown and reflected waves

4.3.3 Parasitic Current Test

Once the effect of surface tension is included, the persistent problem of parasitic currents around the VOF interface becomes an issue. To verify our implementation, we use the test cases described by Francois et al. [34].

The first test consists of a spherical droplet in a quiescent environment using the analytical value for curvature (rather than calculating it). Francois et al. [34] show that for a proper surface tension force implementation the parasitic currents in this case can be reduced to numerical noise levels ($<10^{-16}$) after one time step. Using this test, our implementation produces parasitic currents of approximately 10^{-18} m/s after one time step using the same conditions as the reported tests with a density ratio of 1000. We use the same mesh, time step, and fluid properties as used by Francois et al. [34].

The second test involves calculating the curvature rather than using the analytical expression. Francois et al. [34] compare the parasitic current after 1 and 50 time steps using five different curvature calculation methods. The best case has parasitic currents of $4.02e^{-3}$ cm/s and $4.02e^{-2}$ cm/s after 1 and 50 time steps, respectively. Using our CLSVOF approach we get parasitic currents of $2.3e^{-2}$ cm/s after 50 time steps, so our method produces better results than the best performing height function approach presented by Francois et al. [34].

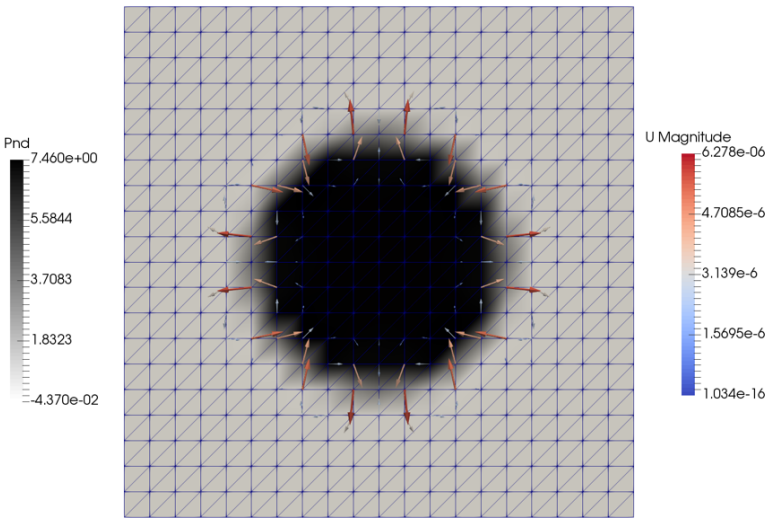


Figure 4.20. Center plane slice of static drop pressure (P_{nd}) and parasitic currents after 1 time step (SI units, Pa and m/s). The analytical Laplace pressure is 7.3 Pa for this configuration.

4.4 Heat Transfer Tests

4.4.1 Phase Change Stefan Problems

As validation for the phase change implementation, we use a classical Stefan problem. The Stefan problem is as described by Samkhaniani and Ansari [38], where the liquid and vapor have the same density so it is purely a heat conduction and moving interface problem. In this case, heat conduction is through the vapor phase to the liquid-vapor interface and causes the interface to move away from the heated wall. In this form, there is an analytical solution for the phase interface position in time that is documented in [38]. The comparison between the simulation and analytical solution for this problem is shown in Fig. 4.21. We use the exact same problem setup as Samkhaniani and Ansari [38] here, so the reader is referred to their paper for further details on the problem.

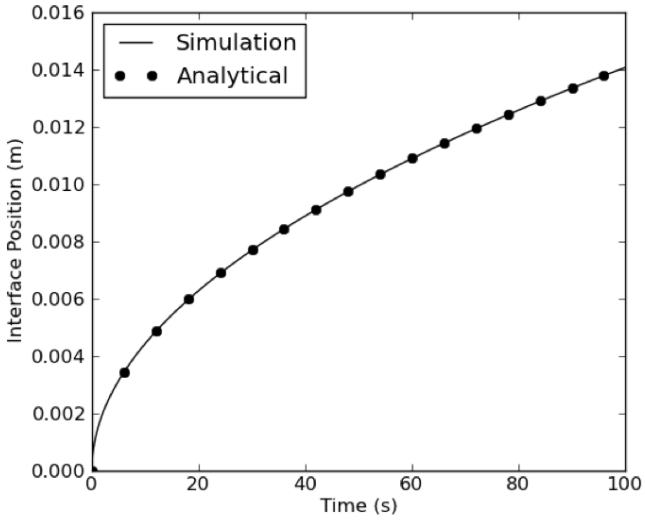


Figure 4.21. Comparison between simulation and analytical solution for Stefan problem

4.4.2 Radiation Driven Evaporation

To model the increased radiation absorption at the phase interface, we introduce an interface absorption term, defined as

$$\alpha_r = \alpha_{r,0} + \alpha_{r,I}\alpha(1-\alpha) \tag{4.26}$$

where $\alpha_{r,0}$ is the nominal absorption and $\alpha_{r,I}$ is a user-specified enhanced absorption at the phase interface. To test this capability, we use $\alpha_{r,0} = 0$ m^{-1} and $\alpha_{r,I} = 1000$ m^{-1}. A hot block (3000 K) is placed above a 10 cm square pool of liquid water initially at 370 K using the thermal evaporation model. The pool regression from evaporation is shown in Fig. 4.22. For an actual fire simulation, $\alpha_{r,0}$ would be calculated by the existing combustion model and would be non-zero in the gas phase flame region.

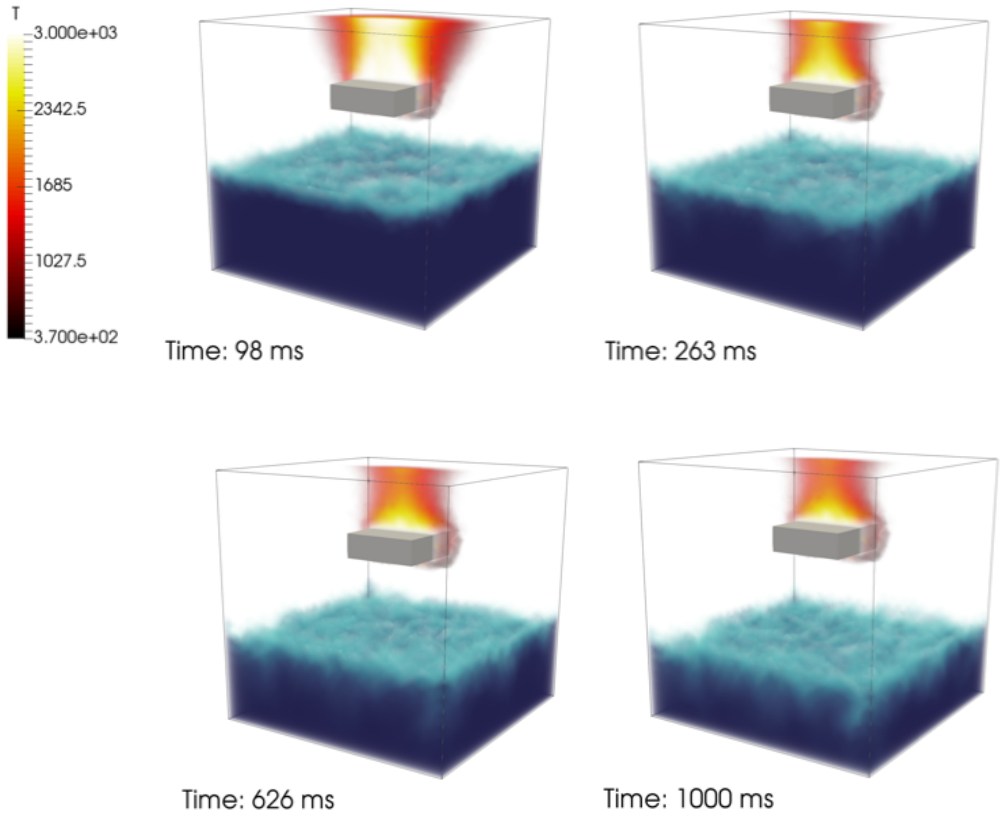


Figure 4.22. Volume rendered isocontours of pool by the liquid volume fraction and gas by the temperature, at selected times.

4.5 Dam Burst Simulation

As a test of the volume of fluid implementation, comparisons to data provide confidence. One classical scenario that provides a good challenge is the dam burst scenario which involves the rapid removal of a dam-like structure (gate valve) and the re-adjustment of the fluid due to the gravity force. The test of Kleefsman et al. [39] was simulated by Crespo et al. [40], and seems to be a good test problem. It involves a 3.22 m long tank with a 1 m square cross-section. The initial water height was 0.55 m, and the length 1.228 m. A rectangular obstruction was located 0.6635 m from the down-stream end, and was 0.161 high, 0.403 m wide, and 0.16 in the length direction. Heights were sampled at 0.582, 1.732, and 2.228 m downstream.

The simulation assumed a gas density of 1 kg/m3, fluid density of 1000 kg/m3, surface tension of 0.07 N/m, liquid viscosity of 0.001 Pa-s, gas viscosity of 1.98e-5 Pa-s, and a contact angle of 45. A coarse mesh was formed, and progressive refinements were performed in Cubit software to split the coarse mesh an integral number of times, resulting in the mesh refinement as illustrated in Table 4.6. The mesh was very close to orthogonal and square, with the maximum aspect ratio for the coarse mesh being 1.087. The time step was managed to keep the CFL number mostly below 1.0 as indicated in the table. Typical maximum CFL numbers were between 0.1 and 0.5 at each timestep.

Table 4.6. Dam burst meshes

No.	Mesh	Nodes	Nominal Mesh Spacing	Time Step (s)
1	coarse	28,600	0.05000 m	0.00250
2	med	216,400	0.02500 m	0.00125
3	fine	716,500	0.01667 m	0.00100
4	xfine	1,682,000	0.01250 m	0.00100
5	xxfine	3,266,300	0.01000 m	0.00050
6	xxxfine	5,622,400	0.00833 m	0.00050
7	xxxxfine	8,903,500	0.00714 m	0.00050

Kleefsman et al. [39] and Crespo et al. [40] demonstrate model accuracy by comparing the height of the fluid at the above-mentioned downstream locations. Data were extracted from [40] and are replicated in Figures 4.23 through 4.25 therein, along with predictions from the newly implemented model. Kleefsman et al. [39] and Crespo et al. [40] use different conventions to identify downstream measurement locations. This effort follows the convention used in [40].

Figure 4.23 shows fluid height predictions and data for the H1 location which is just upstream of the obstruction. Models track the data well, up to one second. At later times, the arrival of the wave peaks is slightly delayed compared to the data. As the mesh is refined, the magnitude of the delay generally decreases. Figure 4.24 shows H2 location results, which exhibits similar results to the H1 location. One difference is that the arrival of the first peak is slightly in advance of the data for the highest resolution model case. Figure 4.25 shows

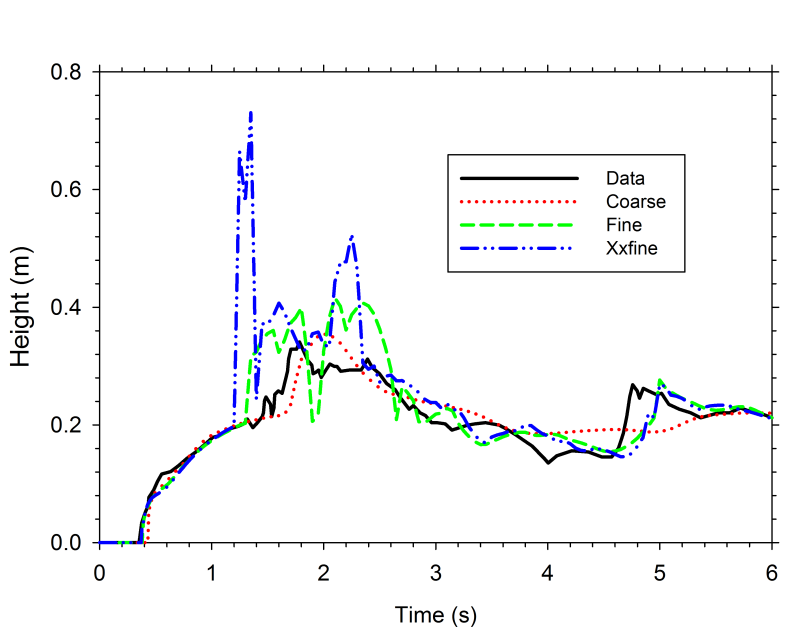


Figure 4.23. Fluid height results at the H1 location

the H3 location results. This location is inside the initial liquid column, so at early times the liquid level is high. Early time simulation results all are similar to the data, diverging at around 2.5 seconds. At this time, there is a reflection wave that arrives. It arrives at a very close time for the data and the two refined model simulations. At about 4 seconds, there is a second peak caused by the second reflection of the wave. The peak height tends to be under-predicted, although the two more refined scenarios correspond better to the data. Peak height was calculated numerically by identifying the maximum height where the volume of fluid was at least 0.5. Locations were determined by the variable clipping algorithm in Paraview (v. 5.1.2), which presumably locates this precise height using a linear interpolation function. The early peaks in the xxfine predictions for the H1 and H2 locations are caused by splash-back, which is likely ignored in the data and is under-resolved in the coarser simulations.

In addition to the height data, the experiments produced images of the spreading liquid. Crespo et al. [40] and Kleefsman et al. [39] extracted images at particular times for comparison with their models. Selected images of the predictions at the same times are provided in Figures 4.26 and 4.27. The source material may be consulted for corresponding data images. The images in Figure 4.26 show predictions for the coarse, fine, and xxfine simulations (all odd numbered meshes in Table 4.6). These are arranged with the coarser mesh predictions higher in the image group. At 0.32 through 0.64 seconds there is a clear difference in the location of the leading edge of the spreading liquid. The increased refinement scenarios predict a faster spread, which is not particularly unexpected given that there is improved resolution on the mesh to resolve the small features of the leading edge of a spreading liquid.

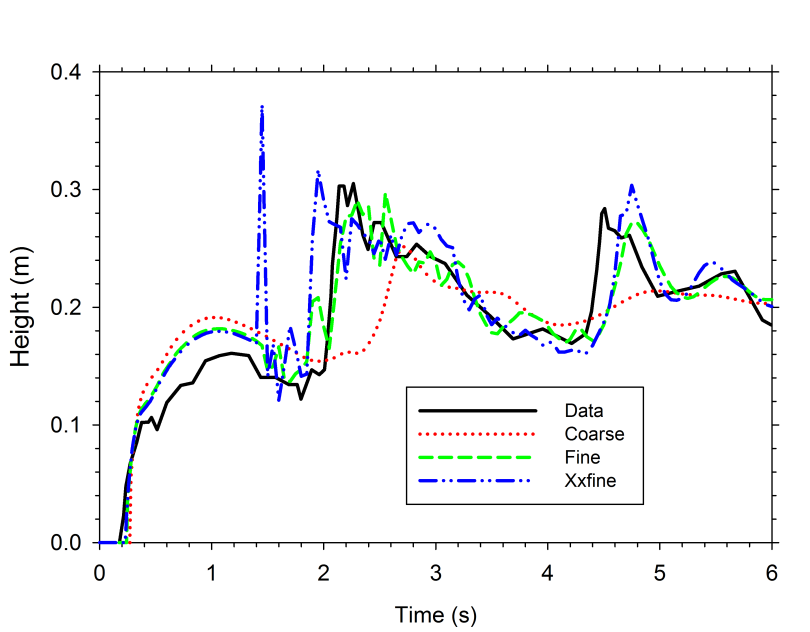


Figure 4.24. Fluid height results at the H2 location

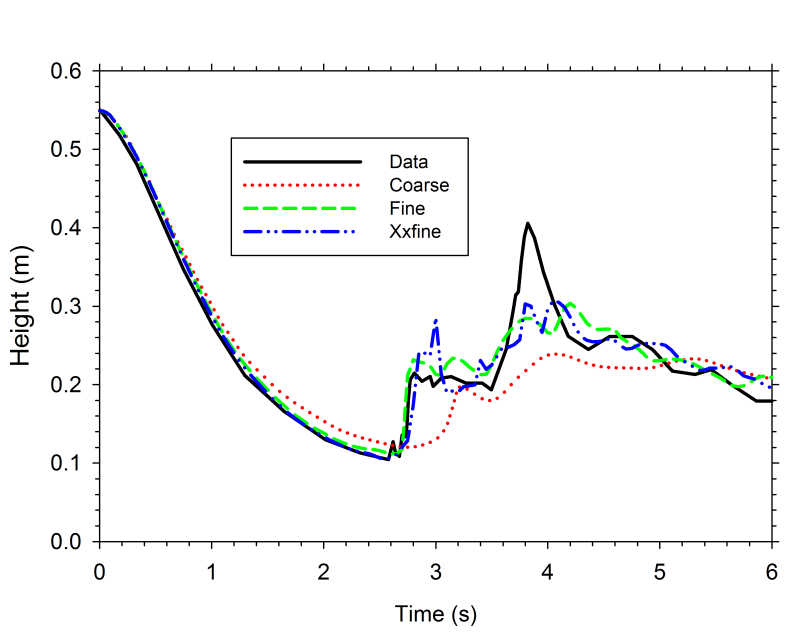


Figure 4.25. Fluid height results at the H3 location

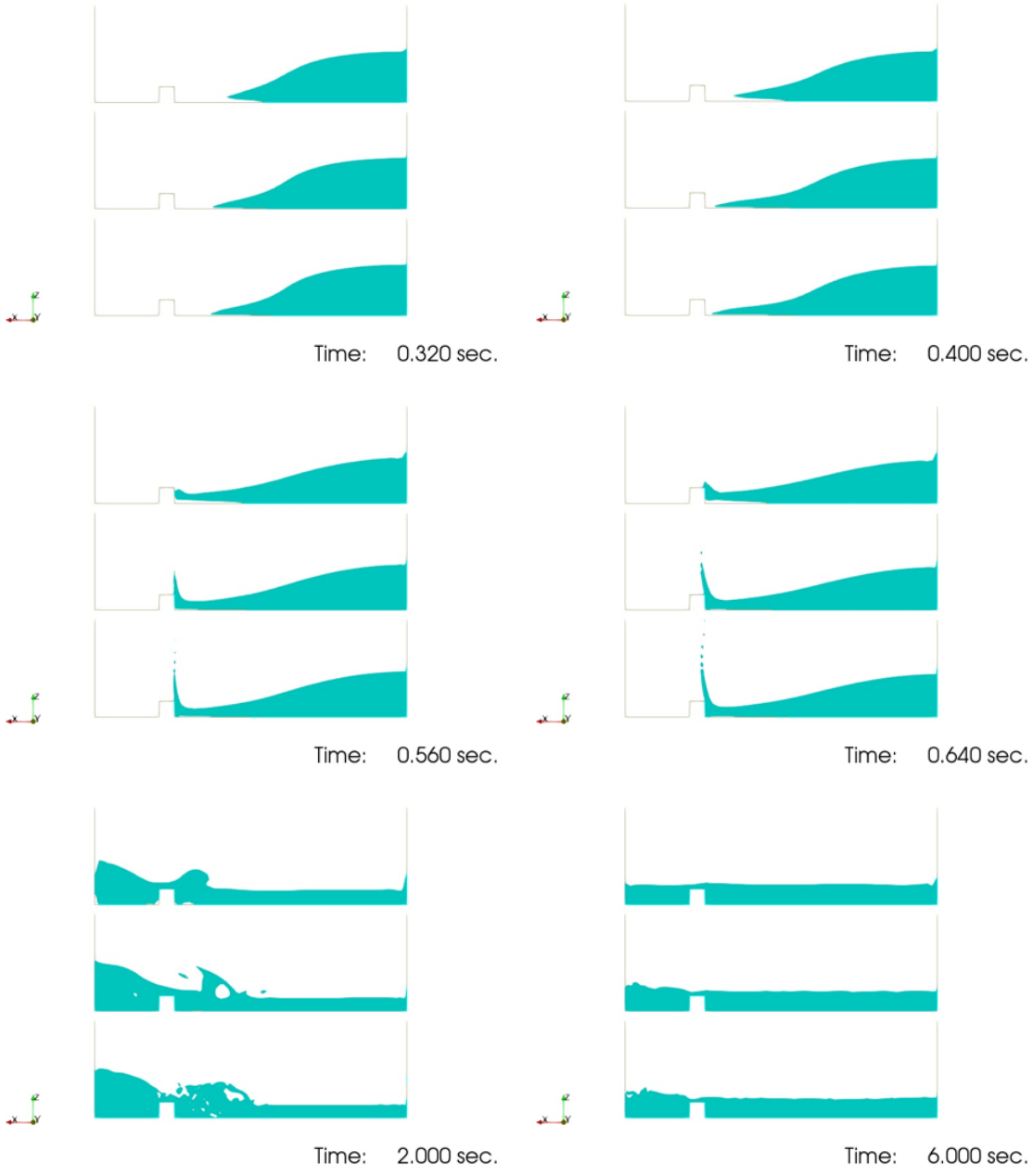


Figure 4.26. Center-plane images of the fluid at various times for three mesh resolutions; coarse, fine, xxfine (top to bottom)

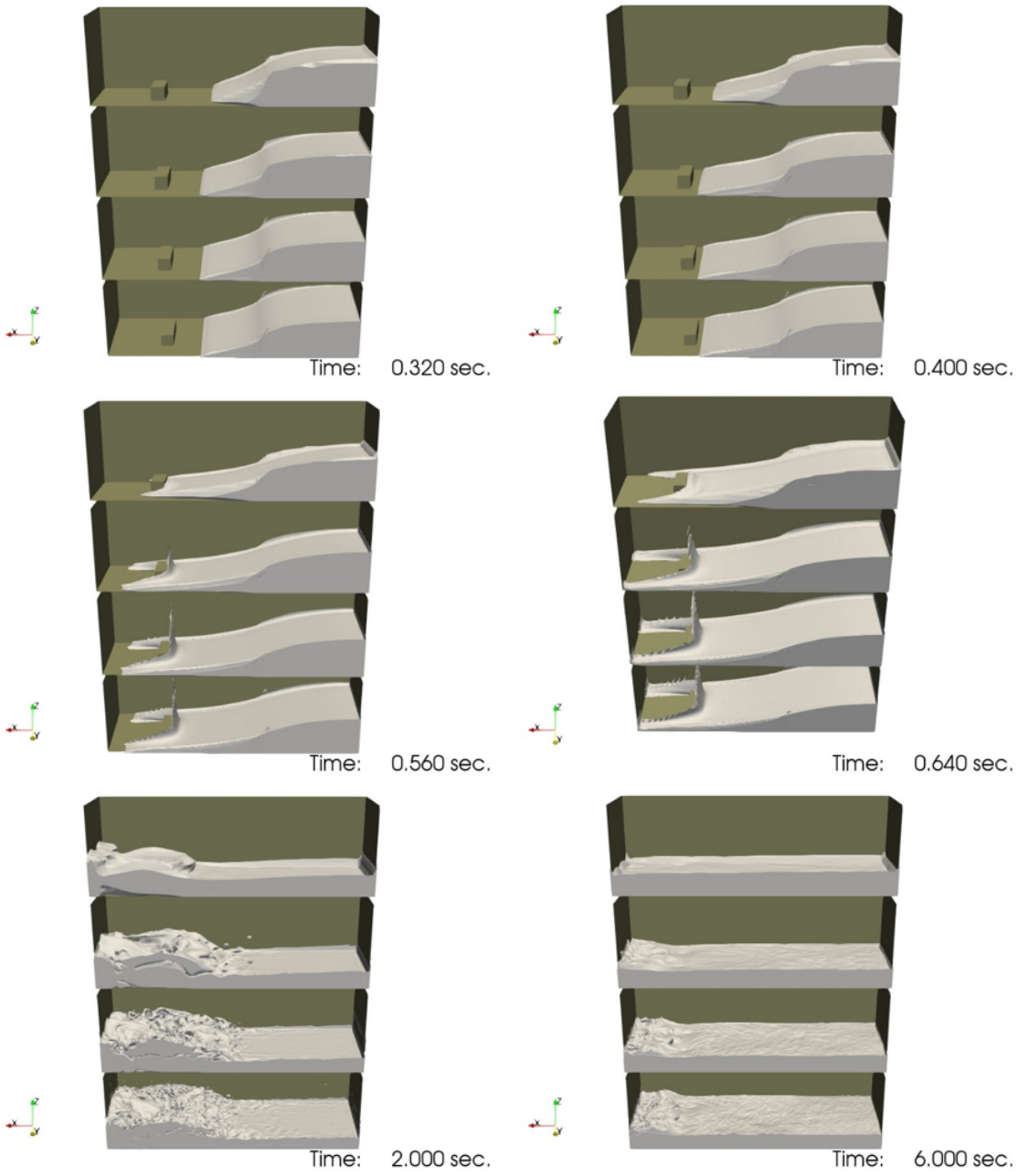


Figure 4.27. Images of the rendered fluid at various times for four mesh resolutions

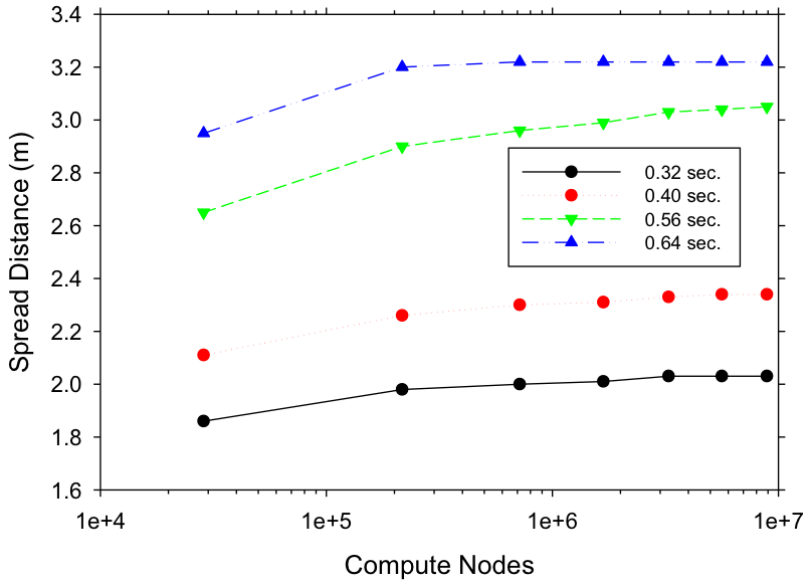


Figure 4.28. Predicted spread distance for various numbers of mesh nodes

Figure 4.27 shows 3D views of the flow and include one higher level of resolution. They show a similarity in answer, and improved similarity with increasing resolution.

The leading edge is extracted at similar times for the various mesh resolutions to demonstrate the convergence of the algorithm. Figures 4.28 and 4.29 show the predicted length of spread for either the total node count or the resolved length scale (mesh length unit; both indicated in Table 4.6). The results exhibit good length-scale convergence for the earlier times, with results showing poorer convergence at later time (0.56 sec.). The 0.64 second results appear convergent, but the fine mesh and above results all agree that the liquid reaches the back wall (not an indicator of convergence). The 0.56 second results appear slightly non-convergent, which is probably due to the fact that the spread has to navigate the obstacle at this time. The flow around the obstacle is likely under-resolved, resulting in increased differences in the flow prediction.

This simulation exercise provides some confidence in the ability of the implemented multiphase algorithms to reproduce physical data. The simulations were performed with idealized fluid properties, and give a reasonably adequate representation of the complex behavior of the liquid as determined in the experimental tests. Mesh resolution is shown to affect the quality of the prediction for very coarse meshes. Besides these quantitative simulation issues, there is also the question of the dam removal process and its effect on the initial conditions of the liquid as well as the temporal component of the removal when compared to the instantaneous assumption made in the model. Given these sources of uncertainty, the model provides a reasonably good reproduction of the measured behavior of the liquid for this scenario.

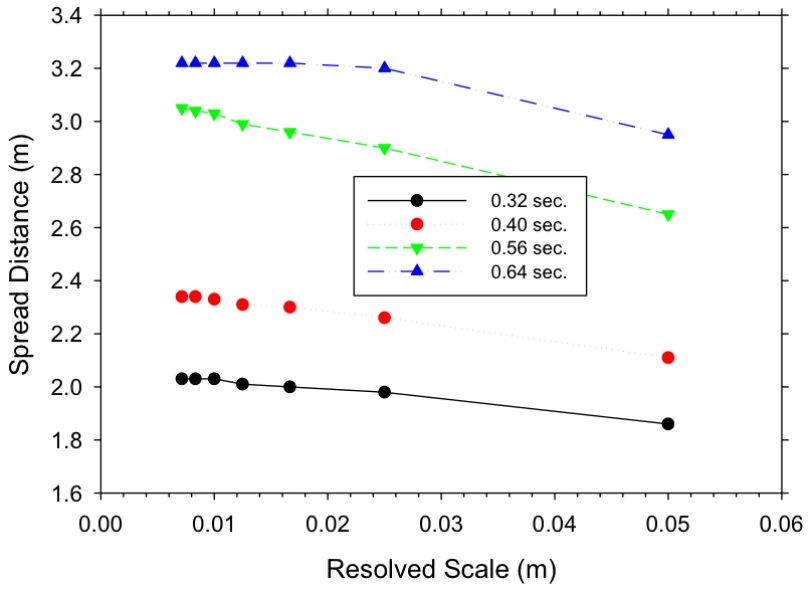


Figure 4.29. Predicted spread distance versus resolved length scale (mesh spacing)

Chapter 5

Conclusion

Various model development activities have been performed related to the rubble fire scenario. A detailed solid decomposition model, CPD, is incorporated in Fuego which assumes an organic solid as a percolated network of molecules joined by labile bonds. The model is well validated over two different coal particle evolution scenarios. CPD application to the target composite material (carbon fiber epoxy) or even pure epoxy requires further development of modeling parameters, which prohibits the use of the model at the moment.

A multi-step decomposition mechanism was fitted to a TGA profile and the model is simulated using Fuego. After revising some of the model parameters, Fuego successfully replicates the TGA degradation rate when a particle approach is used. The model is used to simulate an exposed panel experiment that was heated by a radiation source. While solid decomposition was delayed, panel temperature was predicted well compared to the experiment data. When simulating the panel with the 1D capability, oxidation is slower when compared to using the particle method. This is due to one-sided exposure to the gas phase. For the same mass, the exposed area of the panel is smaller than that of the particle, which discourages oxidation. Moreover, gas generated due to pyrolysis reduces oxygen mass fraction near the panel. Therefore, the panel type simulation cannot be done without solving surrounding fluids. Note that this is a distinct difference from a TGA-type small particle case where fluid-solid interaction is frequently omitted.

In the studies, composition of pyrolysis gas components is simplified as methane. However, it is expected that char and epoxy will produce heavier gas than methane, such as tar. At present, Fuego does not handle more than a single fuel species. A new model was developed to allow different fuel species in Fuego. This enhancement offers an ability to crack heavier tar into methane. Moreover, more accurate species transportation is expected since different diffusivity and buoyancy effects between fuel compositions can be included. Detailed gas composition from carbon fiber epoxy degradation, however, will be found at a later study.

Particle capability has been extended to offer a potential replacement for solid bulk. For a network of particles to represent a rubble fraction, the interparticle thermal interaction is important. Heat transfer is designed to be proportional to the contact area between the particles such that heat transfer becomes smaller when particle shrinks due to decomposition. Moreover, directional differentiated heat transfer, which is important for the composite, can be easily included with the developed approach.

To enable modeling of the liquid transport for the three-phase pool fire problem we implemented a volume-of-fluid capability (VOF) in SIERRA/Fuego. We have validated this capability using comparison with both published multiphase tests and experimental data. This capability is integrated with the existing enthalpy equation and includes thermally-driven phase change in order to capture the fuel evaporation. We have also added interaction between radiation and the liquid interface, which to our knowledge has not been done before. This will be key for the fully integrated fire test since radiative flux to the liquid interface is expected to be a key driver of evaporation.

Modeling methods, verification, and validation for the different material phases are shown here. For submerged solid phase, we expect a competition between delayed decomposition (due to cool surrounding liquid) and heat conduction along the composite from the ignited gas phase above. Liquid fuel near the composite will experience enhanced vaporization due to the conducted heat. The methods developed separately will be brought together to enable simulating a complex fire scenario involving all three phases involved in liquid rubble mixed combustion.

References

[1] A. L. Brown and B. Servantes. Composite rubble fire test results. Number SAND2017-(in preparation), 2017.

[2] A. L. Brown, D. C. Dayton, and J. W. Daily. A study of cellulose pyrolysis chemistry and global kinetics at high heating rates. *Energy & Fuels*, 15:1286–1294, 2001.

[3] A. B. Dodd, B. Shelden, and K. L. Erickson. Numerical simulation of decomposition and combustion of an epoxy-carbon-fiber composite. In *13th International Conference and Exhibition on Fire Science and Engineering*, 2013.

[4] Sandia National Laboratories. *SIERRA Low Mach Module: Fuego User Manual – Version 4.42, SAND2016-10164*, 2016.

[5] A. L. Brown, D. Glaze, and F. Pierce. Sensitivity analysis and verification of a 1-D surface solid combustion model for a fire CFD boundary condition. In *11th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, number 2014-2256, 2014.

[6] Sandia National Laboratories. *SIERRA Multimechanics Module: Aria User Manual – Version 4.42, SAND2016-10166*, 2016.

[7] A. L. Brown and B. E. Vembe. Evaluation of a model for the evaporation of fuel from a liquid pool in a cfd fire code. In *Proceedings of the ASME International Mechanical Engineering Congress and Exposition*, 2006.

[8] D. R. Noble, E. P. Newren, and J. B. Lechman. A conformal decomposition finite element method for modeling stationary fluid interface problems. *International Journal for Numerical Methods in Fluids*, 63(6):725–742, JUN 2010.

[9] D. Grant, R. Pugmire, T. Fletcher, and A. Kerstein. Chemical-model of coal devolatilization using percolation lattice statistics. *Energy & Fuels*, 3(2):175–186, MAR-APR 1989.

[10] T. Fletcher, A. Kerstein, R. Pugmire, and D. Grant. Chemical percolation model for devolatilization. 2. temperature and heating rate effects on product yields. *Energy & Fuels*, 4(1):54–60, JAN-FEB 1990.

[11] T. Fletcher, A. Kerstein, R. Pugmire, M. Solum, and D. Grant. Chemical percolation model for devolatilization. 3. direct use of C-13 NMR data to predict effects of coal type. *Energy & Fuels*, 6(4):414–431, JUL-AUG 1992.

[12] T. H. Fletcher, A. R. Kerstein, R. J. Pugmire, M. S. Solum, and D. M. Grant. A chemical percolation model for devolatilization: Milestone report. In *SAND92-8207*, 1992.

[13] S. Niksa and A. Kerstein. Flashchain theory for rapid coal devolatilization kinetics .1. formulation. *Energy & Fuels*, 5(5):647–665, SEP-OCT 1991.

[14] P. Solomon, D. Hamblen, R. Carangelo, M. Serio, and G. Deshpande. General-model of coal devolatilization. *Energy & Fuels*, 2(4):405–422, JUL-AUG 1988.

[15] https://www.et.byu.edu/∼tom/cpd/cpd.html.

[16] B. F. Magnussen. On the structure of turbulence and a generalised eddy dissipation concept for chemical reactions in turbulent flow. In *9th AIAA Aerospace Science Meeting*, 1981.

[17] M. Hobbs, K. Erickson, and T. Chu. Modeling decomposition of unconfined rigid polyurethane foam. *Polymer Degradation and Stability*, 69(1):47–66, 2000.

[18] M. Hobbs. Modeling epoxy foams exposed to fire-like heat fluxes. *Polymer Degradation and Stability*, 89(2):353–372, AUG 2005.

[19] D. Bikiaris. Can nanoparticles really enhance thermal stability of polymers? Part II: An overview on thermal decomposition of polycondensation polymers. *Thermochimica Acta*, 523(1-2):25–45, AUG 2011.

[20] E. Kandare, B. K. Kandola, and J. E. J. Staggs. Global kinetics of thermal degradation of flame-retarded epoxy resin formulations. *Polymer Degradation and Stability*, 92(10):1778–1787, OCT 2007.

[21] J. G. Quintiere, R. N. Walters, and S. Crowley. Flammability properties of aircraft carbon-fiber structural composite. Number DOT/FAA/AR-07/57, 2007.

[22] J. A. Hubbard, A. L. Brown, A. B. Dodd, S. Gomez-Vasquez, and C. J. Ramirez. Carbon fiber composite characterization in adverse thermal environments. Number SAND2011-2833, 2011.

[23] M. B. McKinnon, Y. Ding, S. I. Stoliarov, S. Crowley, and R. E. Lyon. Pyrolysis model for a carbon fiber/epoxy structural aerospace composite. *Journal of Fire Sciences*, 35(1):36–61, JAN 2017.

[24] A. L. Brown and F. Pierce. Modeling aerodynamic breakup of liquid drops in a gas flow with molecular dynamics analogy methods. In *Proceedings of the 2nd Thermal and Fluids Engineering Conference*, 2017.

[25] J. Mencinger and I. Zun. A PLIC-VOF method suited for adaptive moving grids. *Journal of Computational Physics*, 230(3):644–663, FEB 2011.

[26] J. Martinez, X. Chesneau, and B. Zeghmati. A new curvature technique calculation for surface tension contribution in plic-vof method. *Computational Mechanics*, 37(2):182–193, JAN 2006.

[27] M. Sun. Accuracy improvement of PLIC-VOF volume-tracking method using the equation of surface normal vector. *Advances in Pure Mathematics*, 3(1A):219–225, 2013.

[28] M. Huang, L. Wu, and B. Chen. A piecewise linear interface-capturing volume-of-fluid method based on unstructured grids. *Numerical Heat Transfer Part B-Fundamentals*, 61(5):412–437, 2012.

[29] H. Lee and S. H. Rhee. A dynamic interface compression method for VOF simulations of high-speed planing watercraft. *Journal of Mechanical Science and Technology*, 29(5):1849–1857, MAY 2015.

[30] A. Q. Raeini, M. J. Blunt, and B. Bijeljic. Modelling two-phase flow in porous media at the pore scale using the volume-of-fluid method. *Journal of Computational Physics*, 231(17):5653–5668, JUL 2012.

[31] L. Jofre, O. Lehmkuhl, J. Castro, and A. Oliva. A PLIC-VOF implementation on parallel 3d unstructured meshes. In *Proceedings of the V European Conference on Computational Fluid Dynamics*, 2010.

[32] S. Hardt and F. Wondra. Evaporation model for interfacial flows based on a continuum-field representation of the source terms. *Journal of Computational Physics*, 227(11):5871–5895, MAY 2008.

[33] Z. Tukovic and H. Jasak. A moving mesh finite volume interface tracking method for surface tension dominated interfacial fluid flow. *Computers & Fluids*, 55:70–84, FEB 2012.

[34] M. Francois, S. Cummins, E. Dendy, D. Kothe, J. Sicilian, and M. Williams. A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework. *Journal of Computational Physics*, 213(1):141–173, MAR 2006.

[35] D. Gerlach, G. Tomar, G. Biswas, and F. Durst. Comparison of volume-of-fluid methods for surface tension-dominant two-phase flows. *International Journal of Heat and Mass Transfer*, 49(3-4):740–754, FEB 2006.

[36] M. Rudman. Volume-tracking methods for interfacial flow calculations. *International Journal for Numerical Methods in Fluids*, 24(7):671–691, APR 1997.

[37] W. Rider and D. Kothe. Reconstructing volume tracking. *Journal of Computational Physics*, 141(2):112–152, APR 1998.

[38] N. Samkhaniani and M. Ansari. A vof method to phase change modeling. In *1st Iranian Conference on Heat and Mass Transfer*, 2012.

[39] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, and B. Buchner. A volume-of-fluid based simulation method for wave impact problems. *Journal of Computational Physics*, 206(1):363–393, JUN 2005.

[40] A. C. Crespo, J. M. Dominguez, A. Barreiro, M. Gomez-Gesteira, and B. D. Rogers. GPUs, a new tool of acceleration in CFD: Efficiency and reliability on smoothed particle hydrodynamics methods. *PLOS ONE*, 6(6), JUN 2011.

DISTRIBUTION:

1 MS 0899 Technical Library, 9536 (electronic copy)


Sandia National Laboratories