Hardening of Arches for Commercial Simulation of Industrial Flares

Marc Cremer, Minmin Zhou, Matthew McGurn, Adam Cowley, Dave Wang
Reaction Engineering International
cremer@reaction-eng.com
801-875-4314

Jeremy Thornock
University of Utah
j.thornock@utah.edu

Abstract

With funding from the Department of Energy (DOE), to further the use of government funded high-performance computing (HPC) software for engineering analysis of industrial problems, Reaction Engineering International (REI) is working with the University of Utah to leverage the Uintah Computational Framework (UCF) for commercial simulation of industrial flares. The Arches component of the UCF provides a reacting large eddy simulation (LES) capability, which is a more fundamentally accurate description of turbulent mixing and combustion than is obtained in conventional Reynolds Averaged Navier Stokes (RANS) approaches. Since the application of Arches to the simulation of commercial flares presents many challenges to a potential user, including software compilation, case definition, case setup, simulation and post-processing in an HPC facility, there is a need for streamlining this process to make Arches and commercial HPC facilities more accessible to flare designers and end-users. This paper provides an update on the results of our DOE program focusing on three areas:

- Evaluation of mesh sensitivity
- Evaluation of subgrid scale models
- Development of a web-based interface

The computational demands associated with LES simulations performed on meshes that resolve 80% of the turbulent kinetic energy are extremely high. It is expected that in many potential industrial applications, the computational demands associated with this level of resolution may exceed available resources. Evaluation of the impacts of mesh refinement and the chosen subgrid scale model is an important element in hardening Arches for commercial simulation of industrial flares.

Results in this paper show that quantitative predictions of combustion efficiency from a simple flare tip vary as the mesh size resolves up to 69% of the turbulent kinetic energy. Further work will be necessary to evaluate the predictive capability of Arches flare simulations that are completed on meshes of this size. Simulations have also been completed to evaluate impacts of two different subgrid scale models: 1) the commonly used dynamic Smagorinsky model, and 2) the novel Sigma model. The simulations indicate that the Sigma model provides quantitatively similar combustion efficiency predictions compared to the dynamic Smagorinskly model at a three-fold reduction in computer run-time. The paper also describes the front-end and back-end design of a web-based interface that is under development to streamline case definition, simulation, and post-processing of flare simulation results from a commercial HPC facility.

1 Introduction

Visual examination of an industrial flare clearly demonstrates the highly unsteady behavior of flare flames, and the large range of turbulent length scales that are at play within a flare, fundamentally impacting the overall flare performance and flare emissions. Due to the inherent unsteadiness and the large range of turbulent length scales involved with industrial flare operation, it is challenging for Reynolds Averaged Navier Stokes (RANS) models to provide robust predictions of flare performance over a large range of design, environmental, and operational parameters. Large-eddy simulation (LES) may provide a feasible approach to high fidelity computational evaluation of industrial flares.

To leverage the more fundamental representation of turbulence that is offered with LES, high performance computing (HPC) resources are typically necessary. HPC resources exist but have traditionally been limited to expert-level users at universities and national labs. Their use by US industry for engineering and manufacturing has been severely lagging [1, 2]. In most cases, small to medium sized companies do not have the resources to develop and maintain the in-house expertise or hardware to support these advanced modeling and simulation capabilities [3]. However, economical access to these high-fidelity modeling and simulation tools as well as to the HPC facilities able to support them could have significant impact on the ability to improve design, manufacturing, and operational know-how. This access will improve the ability of small to medium sized US businesses to meet long term US goals associated with improved air quality while maintaining an economical advantage in the global market.

Researchers at the University of Utah have previously demonstrated the application of Arches to simulation of industrial flares using DOE and University of Utah HPC facilities [4] [5]. The computational requirements in terms of data storage and computational time to carry out these simulations can be daunting. In addition, the tasks associated with code compilation, case setup, simulation, and post processing on sophisticated HPC hardware requires expertise with the LES software as well with the HPC hardware. The key objective of our U.S. Department of Energy (DOE) sponsored program is to harden the application of Arches to commercial simulation of industrial flares to make the software and commercial HPC facilities more accessible to designers and end users.

This paper describes steps that have been taken and that are in process to harden Arches for this purpose. The first key area of development is the integration of Arches with a web-based user-interface. REI continues to develop the front-end and back-end design. An example showing the functionality of the interface for case definition, simulation, and post-processing results from a commercial HPC facility is provided in this paper.

Two key areas of focus, which directly impact the computational requirements of an Arches flare simulation are mesh size (i.e. resolution) and choice of subgrid scale model. This paper describes results of flare simulations evaluating two subgrid-scale models over three different mesh sizes. The novel Sigma model has been seen to reduce the computer run-time by three-fold with similar predictions of combustion efficiency as the conventional dynamic Smagorinsky model. In all cases, the simulations were carried out for a simple flare tip geometry which accommodated an accurate geometrical representation over the range of grid refinements. The most resolved case was calculated to resolve an average of 69% of the turbulent kinetic energy and the coarsest grid was calculated to resolve an average of 53% of the turbulent kinetic energy.

2 Methods, Assumptions, and Procedures

2.1 User Interface

A focus of the prototype user interface development has been to build a flexible foundation that will serve for future growth. A variety of platforms and paradigms were investigated in order to select the components for the Flare UI. This effort included development of fully functional proof of concepts of simplified applications in AngularJs and React with both Bootstrap and Semantic UI stylings. After developing the prototype UIs and exploring multiple other frameworks it was decided to build the Flares UI using the single page application paradigm shown in Figure 1.

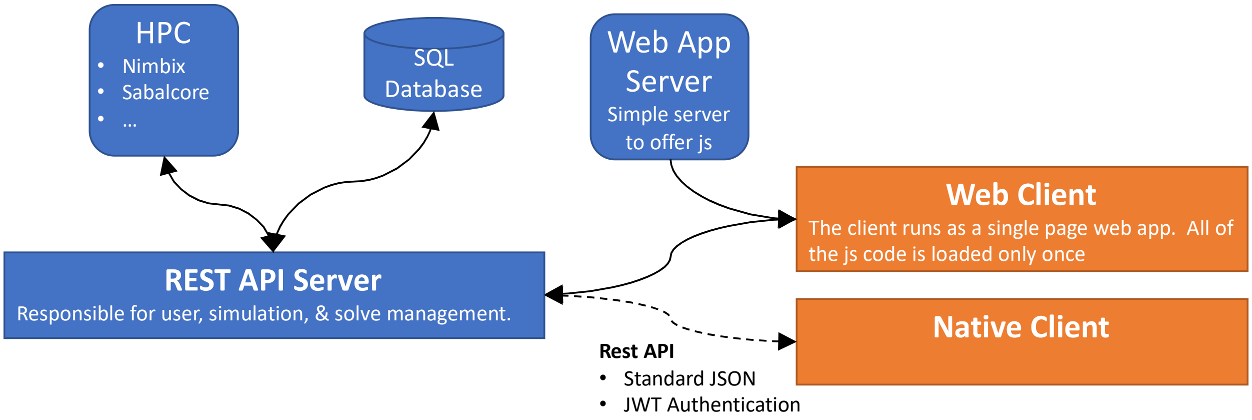

HPC
Nimbix
Sabalcore
…
SQL Database
Web App Server
Simple server to offer js
Web Client
The client runs as a single page web app. All of the js code is loaded only once
REST API Server
Responsible for user, simulation, & solve management.
Native Client
Rest API
Standard JSON
JWT Authentication

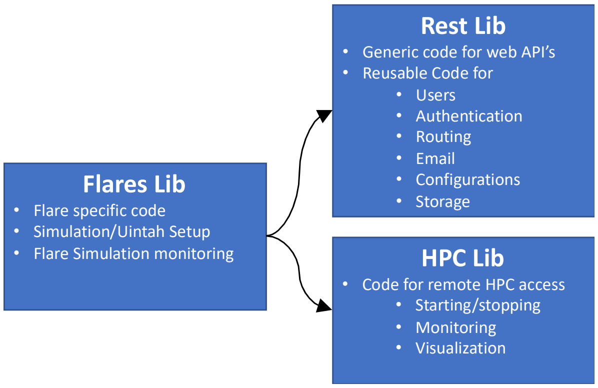
Figure 1: Single page application paradigm used for the Flares UI

A single page application (e.g. Google Docs or Microsoft Office Online) operate and behave more like traditional desktop applications. As shown in Figure 1, single page applications load the entire Java Script (JS) program into the client's web browser where the rendering is done on the client. This results in a more responsive interface. The web client communicates with the REST API Server to store and retrieve user specific information. This communication is accomplished using the industry standard JavaScript Object Notation (JSON) file format and the REST architecture. An added benefit of the utilizing the REST architecture is that it allows the same back-end server to be used by multiple applications including native client applications (illustrated with a dotted line for future development). The REST server handles most of the application logic including user, simulation, and solve management. The server also combines information and communicates with multiple sources including the user, simulation databases, and HPC resources. Available HPC servers communicate with the REST API Server using a custom daemon running for each simulation solve.

The current user interface of the Flares UI is a single page application that is run in a client's web browser. Multiple open-source frameworks are available to help speed development and ensure reliable code. After testing multiple possibilities REI selected to utilize the React-Redux framework using TypeScript programming language. TypeScript is a superset of JavaScript that compiles to JavaScript capable of running in any modern web browser. TypeScript allows for statically typed classes and objects which result in a framework easier to develop and maintain by multiple people. The React-Redux frameworks provide a system to build component-based UI that allow for components to be reused within the same project and across applications. The paradigms and React Library simplify the rendering and updating the user interface based upon user interactions. The Redux library supports an application state that ensures that local and server-side changes stay in sync and are reflected throughout the application.

The front-end single page application communicates to the backend REST server using JSON file. JSON is a lightweight near universal data-interchange format. Information of formatted within the local single page application as JSON and then sent to the server. The data is parsed on the server, acted upon, and a response formatted and returned. Individual images and other files are handled using standard web protocols.

The Go programming language and associated standard library were chosen to serve as the basis of the back-end Rest API Server. Go was selected for simplicity and standard library tailored for web applications. The server is broken into three primary libraries in order to allow reuse amongst different projects, outlined in Figure 2. The Flares Lib is comprised of code specific to the Flares UI API including application specific simulation setup, monitoring, and geometry. The Flares Lib references both the Rest and HPC libraries. These libraries contain code and methods that can be utilized by multiple projects.



Flares Lib
- Flare specific code
- Simulation/Uintah Setup
- Flare Simulation monitoring

Rest Lib
- Generic code for web API's
- Reusable Code for
  - Users
  - Authentication
  - Routing
  - Email
  - Configurations
  - Storage

HPC Lib
- Code for remote HPC access
  - Starting/stopping
  - Monitoring
  - Visualization

Figure 2: Overview of the REST API Server Components.

2.2 HPC facilities

Our efforts to leverage commercial HPC facilities have focused thus far on Nimbix. The Nimbix platform was originally selected due to its containerized cloud computing capabilities in addition to its industry standard HPC hardware with high speed interconnects. A variety of node types is available to the end user for an appropriate workload, from low CPU count/high memory to high CPU count/high memory, and various combinations with multiple GPU types. However, developers can customize their application to guide the user into using appropriate node types. The JARVICE platform encompasses several components: a web interface (Material Compute), application deployment (PushToCompute), a web-based Application Programming Interface (API) for job control/monitoring, and several job UI access methods.

For our evaluation of grid sensitivity and subgrid model sensitivity, REI was also able to utilize an allocation from the Argonne Leadership Computing Facility (ALCF) Discretionary allotment. An initial allocation of 750,000 core hours was awarded on Mira.

2.3 Flare Simulations

2.3.1 Grid Resolution

From a computational scaling perspective, Arches has been successful in scaling to very large numbers of computational cores (O(100K)) mainly due to a structured meshing strategy. The structured mesh strategy, along with a domain patch decomposition allows the framework to equally distribute work across the total number of cores to obtain a high-level of efficiency. In the case of flare simulations, the structured meshing strategy leads to limitations in the ability to accurately resolve the geometry of the flare tip. Increased resolution of the flare tip geometry through increasing mesh size comes at the expense of increasingly large grids covering the simulation domain. Along with increasing the mesh resolution, an increased fraction of turbulent kinetic energy is resolved.

To focus our evaluation on the impact of increasing the fraction of resolved turbulent kinetic energy, without sacrificing accuracy in the representation of the geometry of the flare tip, a geometrically simple flare tip geometry was selected (see Figure 3). This simplified flare tip always conserves the mass and momentum of inlet streams over the range of grid refinements. In our evaluation, a series of different mesh resolutions was chosen to describe this simplified flare tip. The operating conditions were based on the A2.1 test in the TCEQ 2010 study [6].

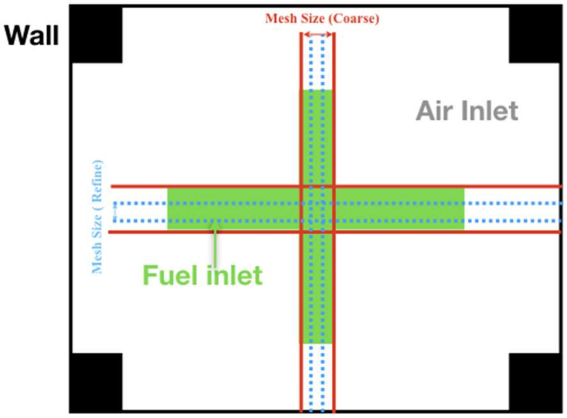


Figure 3: Geometry for the simple flare tip

The operating conditions of A2.1 Run 2 in the TCEQ 2010 study is listed in Table 1. A2.1 Run 2 test fires pure propylene at a flow rate of 355 lb/hr, ~0.25% of maximum capacity (144,000 lb/hr). This fuel gas flow rate is in the range of operation for typical flow rates (less than 0.5% of maximum capacity) used in industry.

Table 1: Operating Condition of A2.1 Test and Designed Case

Parameter	A2.1 Test
Fuel	
Fuel C_3H_6 Mole Fraction	1.00
Fuel Flow Velocity (m/s)	0.152
Fuel Inlet Area (m^2)	0.176
Fuel Initial Temperature (K)	300.
Assisted Air	
Air Inlet Area (m^2)	0.697
Air Flow Velocity (m/s)	13.2
Ambient Temperature (K)	300.
Stoichiometric Ratio (SR)	15.9

Figure 4 shows the overall chosen simulation domain ([2.88m, 2.88m, 8m]). The simulation domain is constructed as following:

a) Wind direction is chosen as the direction from –y to +y
b) Domain height is chosen as about the height of flare (8m), starting from the ground level
c) In the xy-plane, the length is chosen as three times of flare bottom diameter (3*0.96 flare tip width = 2.88m), the flare center location is placed in the center of that plane.
d) Uniform resolution was chosen in three directions.

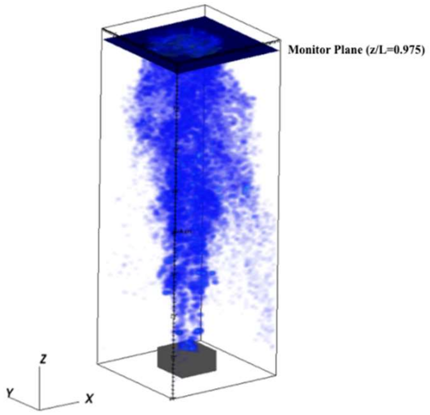


Figure 4: Simulation domain

Numerical details of the simple flare simulations for different mesh resolution are listed in Table 2. For the same physical time of the simulations, the fine mesh resolution will be computationally more costly compared to the coarse mesh resolution. For example, the computational costs of a refined grid with two-fold refinement will be about 16 times of that of the coarse grid [7]. Table 3 lists the estimated computational costs at a commercial HPC facility ($0.09/core-hour) as a function of mesh resolution. These estimates are based on scaling of the previously completed detailed simulation of the A2.1 test condition, and they also correspond to use of the more computationally expensive dynamic Smagorinsky SGS model [8]. Refining the 3*Delta mesh size to 1.5*Delta increases the computational cost 16-fold to approximately $359,424. This change in grid resolution corresponds to an increase in the fraction of resolved turbulent kinetic energy from approximately 65% to 72.8%. This table demonstrates the trade-off associated with grid resolution (i.e. fraction of resolved turbulent kinetic energy) and computational cost and the economic need to evaluate what level of grid refinement is sufficient for reliable commercial simulations of industrial flares with Arches.

Table 2: Resolution and cost details of flare simulations

	Mesh Size(m)	Resolution	Timestep (s)	Relative core-hour
8*Delta	0.08	[36,36,100]	0.0004	1
4*Delta	0.04	[72,72,200]	0.0002	16
2*Delta	0.02	[144,144,400]	0.0001	256
1.5*Delta	0.015	[192,192,536]	0.000075	1296
Delta	0.01	[288,288,800]	0.00005	4096

Table 3: Commercial computational cost estimates ($0.09/core-hour) of flare simulations

	Mesh Size(m)	Ratio of Resolved TKE to total TKE	Core-hours	Cost
4*Delta	0.04	0.60	78,975	$7,108
3*Delta	0.03	0.65	249,600	$22,464
2*Delta	0.02	0.688	1,263,600	$113,724
1.5*Delta	0.015	0.728	3,993,600	$359,424

2.3.2 Subgrid Model

In addition to evaluation of the impacts of mesh resolution on predicted flare performance (e.g. combustion efficiency), simulations were carried out to evaluate impacts of two specific subgrid-scale (SGS. Choice of the SGS model has a direct impact on computational cost and on predictive accuracy.

The two SGS models that we evaluated are the dynamic Smagorinskly model [9] and the Sigma model [10]. At the subgrid scale, LES modeling assumes an eddy-viscosity assumption as shown in Eq.1.

$$\tau_{ij} - \frac{1}{3}\tau_{kk}\delta\delta_{ij} = -2\nu_{SGS}\bar{S}_{ij}. \qquad \text{Eq. 1}$$

The turbulent viscosity, is defined as

$$\nu_{\mathrm{SGS}} = (\mathrm{C_m}\bar{\Delta})\bar{\mathrm{D}}_{\mathrm{S}}, \qquad \text{Eq. 2}$$

where $\mathrm{C_m}$ is the model specific constant, $\mathrm{D_S}$ is the differential operator of the model and $\bar{\Delta}$ is the filter width. The differential operator of the Smagorinksy model is based on the resolved strain rate as

$$\mathrm{D_S} = \sqrt{2\bar{\mathrm{S}}_{\mathrm{ij}}\bar{\mathrm{S}}_{ij}}\,, \qquad \text{Eq. 3}$$

where $\overline{\mathrm{S}}_{\mathrm{ij}} = \frac{1}{2}\left(\frac{\partial \bar{\mathrm{u}}_{\mathrm{i}}}{\partial x_{\mathrm{j}}} + \frac{\partial \bar{\mathrm{u}}_{\mathrm{j}}}{\partial x_{\mathrm{i}}}\right)$. The Sigma model adopts the singular values of the velocity gradient tensor of $\frac{\partial \bar{\mathrm{u}}_{\mathrm{i}}}{\partial x_{\mathrm{j}}}$. The differential operator is defined as

$$\mathrm{D}_{\mathrm{S}} = \frac{\delta_3(\delta_1 - \delta_2)(\delta_2 - \delta_3)}{\delta_3^2}, \qquad \text{Eq. 4}$$

where δ_i is the singular values of the velocity gradient tensor [10].

3 Results and Discussion

3.1 User Interface

3.1.1 Case Setup

The web-based user interface can run in any modern browser and is being constructed to be fast and responsive. Much of the front-end interface has been constructed and is described here for the John Zink air-assisted flare tip, which was previously simulated [8]. Figure 5 shows a detailed description of specific information regarding flare geometry and operating information, which is communicated by the end user through the web interface, which is translated by the REST API Server to construct the Uintah Problem Specification (UPS). The UPS contains the necessary case set-up information for the UCF (i.e. Arches) simulation.

The User Interface (UI) includes user login/registration and user permissions (i.e. a specific user is limited to access of their own files) and the login page is the entry point of the user to the application. The UI allows the user to create a new flare template from scratch or to start with an existing flare template, which resides in the library. Figure 6 shows the template selection window. The paradigm that we have used to design the UI allows for additional templates to be added with little effort.

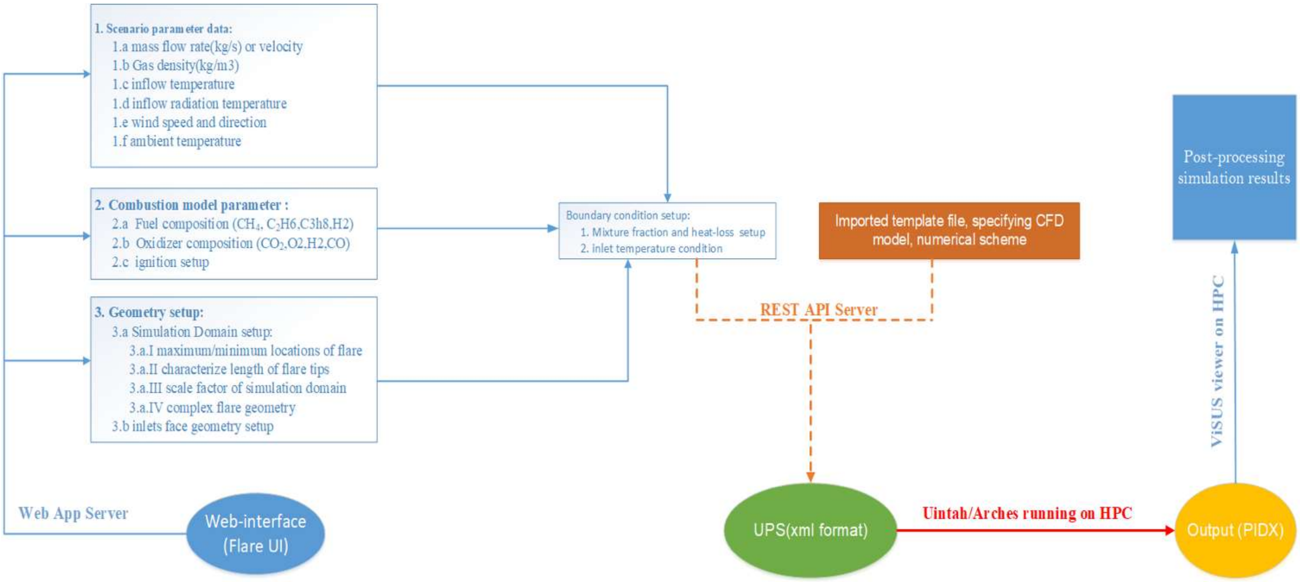
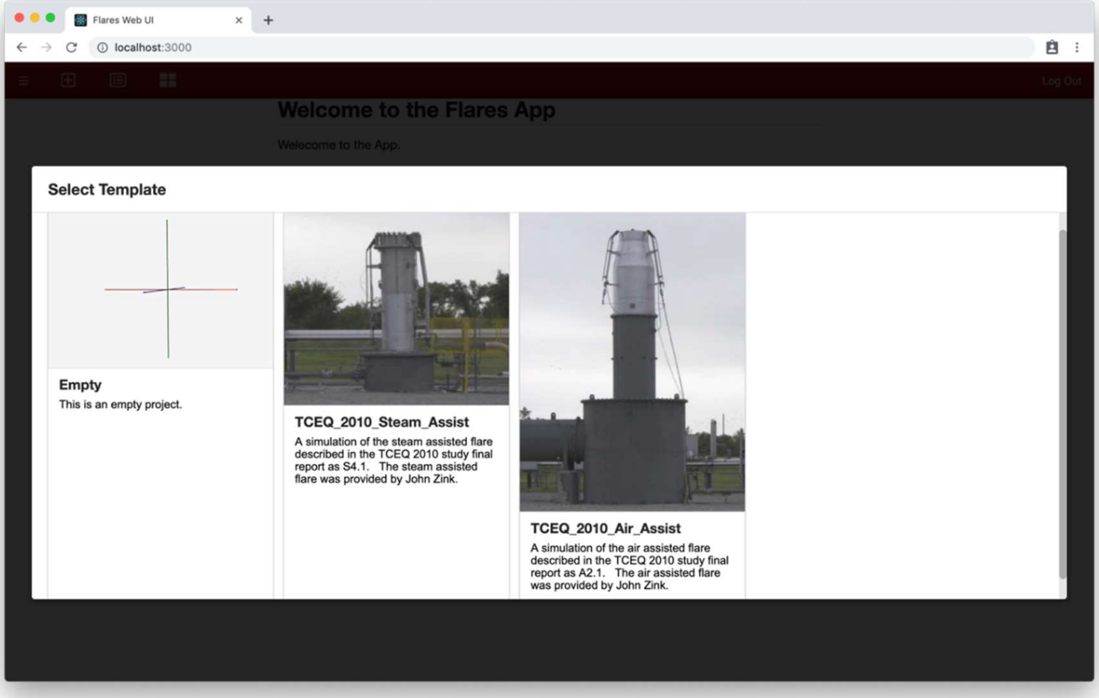

1. Scenario parameter data:
1.a mass flow rate(kg/s) or velocity
1.b Gas density(kg/m3)
1.c inflow temperature
1.d inflow radiation temperature
1.e wind speed and direction
1.f ambient temperature
2. Combustion model parameter :
2.a Fuel composition (CH_4, C_2H6,C3h8,H2)
2.b Oxidizer composition (CO_2,O2,H2,CO)
2.c ignition setup
3. Geometry setup:
3.a Simulation Domain setup:
3.a.I maximum/minimum locations of flare
3.a.II characterize length of flare tips
3.a.III scale factor of simulation domain
3.a.IV complex flare geometry
3.b inlets face geometry setup
Boundary condition setup:
1. Mixture fraction and heat-loss setup
2. inlet temperature condition
Imported template file, specifying CFD model, numerical scheme
REST API Server
Post-processing simulation results
ViSUS viewer on HPC
Web App Server
Web-interface (Flare UI)
UPS(xml format)
Uintah/Arches running on HPC
Output (PIDX)

Figure 5: Workflow from web-interface to CFD simulation



Welcome to the Flares App

Welecome to the App.

Select Template

Empty
This is an empty project.

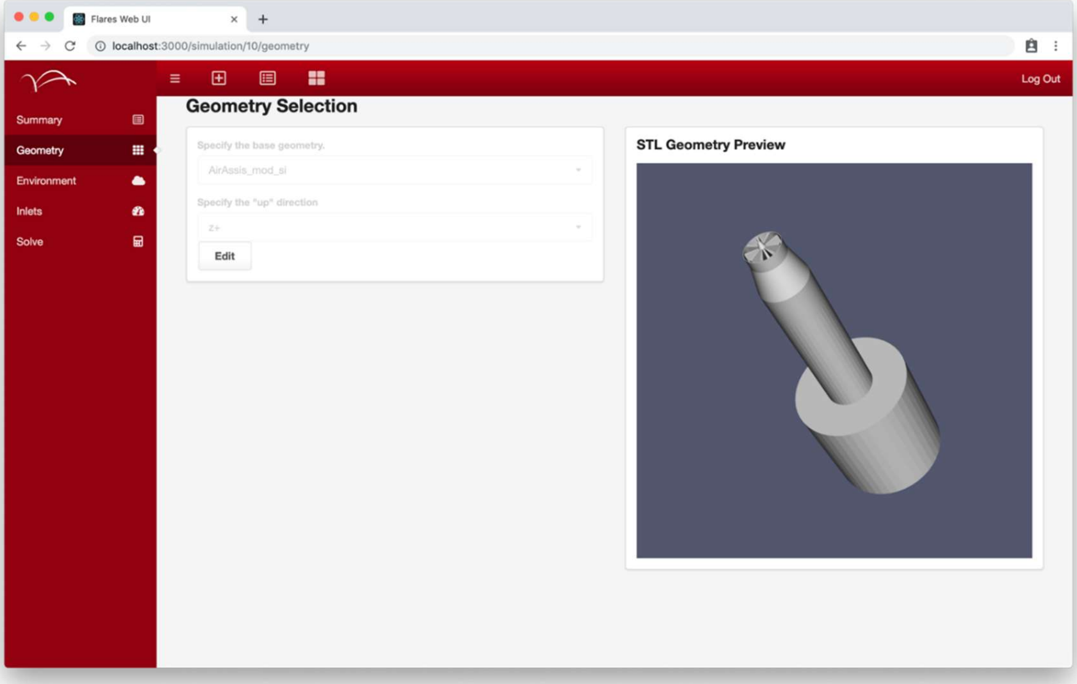
TCEQ_2010_Steam_Assist
A simulation of the steam assisted flare described in the TCEQ 2010 study final report as S4.1. The steam assisted flare was provided by John Zink.

TCEQ_2010_Air_Assist
A simulation of the air assisted flare described in the TCEQ 2010 study final report as A2.1. The air assisted flare was provided by John Zink.

Figure 6: Template selection page

Once a template is chosen, a new case set-up is generated and pre-filled with the template values. For example, when the TCEQ_2010_Air_Assist is selected, the geometry is automatically set to the corresponding flare tip (see Figure 7). The geometry preview pane features a 3D preview of the CAD geometry (STL format), which can be controlled with the mouse including the ability to orbit, pan, zoom, and rotate the 3D preview. The 3D geometry viewer is built on vtk.js which is a JavaScript implementation of the popular open source VTK (Visualization Tool Kit) library by Kitware [11].

The user navigates through the various simulation set-up pages using the collapsible sidebar menu (see left side of Figure 7). When operating conditions or geometrical information is updated, it is automatically synced with the REST API server. Geometry specification of the flare tip and of the flare simulation domain is a very important component of the UI and continues to be developed. As shown in Figure 8, multiple inlets can be specified along with their chemical compositions (e.g. air or fuel gas). When in edit mode, the 3D rendering tool (also built with vtk.js) allows for the selection of the inlet faces by using the right mouse button (see the red and blue colored faces). These inlets are then used in conjunction with the CAD file to define the corresponding inlet boundary conditions in the Flares simulation.



Geometry Selection

Specify the base geometry.

AirAssis_mod_si

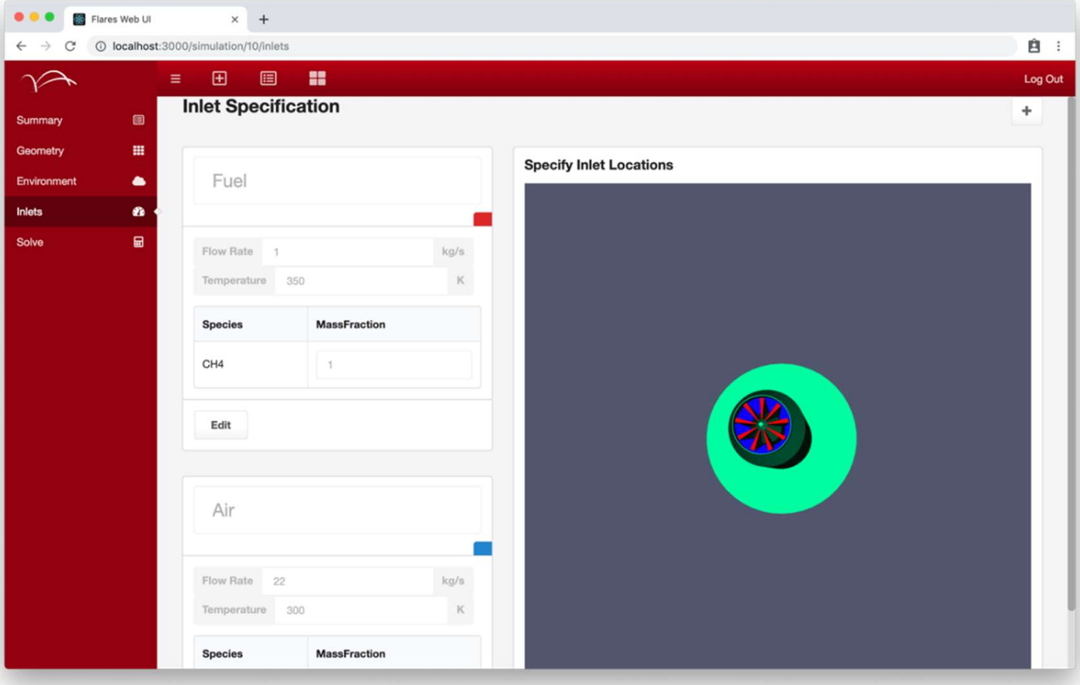
Specify the "up" direction

z+

Edit

STL Geometry Preview

Figure 7: Geometry selection and preview page



Inlet Specification

Fuel

Flow Rate 1 kg/s

Temperature 350 K

Species	MassFraction
CH4	1

Edit

Air

Flow Rate 22 kg/s

Temperature 300 K

Species	MassFraction

Specify Inlet Locations

Figure 8: Inlet specification and selection page

The primary purpose of the front-end UI, with regards to the case setup, is the construction of the UPS file. The UPS file is written in Extensible Markup Language (XML) format, and it provides all case setup information (inlet conditions, boundary conditions, geometry, numerical model parameters, etc.) that is necessary to execute an Arches simulation. Using Go language (by Google), we have developed the necessary code to interpret the information retrieved from the user's input from the web-interface to output the correct XML in the UPS.

As described in the previous section, much of the geometrical, inlet, and boundary information for the flare simulation is communicated by the user in the web interface. Other parameters, that are necessary components of the UPS file, will use default values. Examples of this include various LES parameter and simulation options, such as the numerical method, transport equations, and radiation model. Another example is the specification of the number of patches used in the Arches simulation, which translates to the number of cores that will be used for parallel processing. These default specifications along with the information retrieved by the web interface are written in Go language and used within the REST API server to construct the correct XML in the UPS file.

The Arches software currently adopts a rate-controlled constrained equilibrium (RCCE) approach to represent the combustion chemistry. This approach requires the use of a pre-generated chemistry table which depends on the fuel composition. Currently, we have automated the development of the chemistry table within the REST API server for fuels that include a mixture of CH_4, C_2H_6, and C_3H_8.

A custom daemon is currently being developed that allows cases on the commercial HPC facilities to communicate with the REST API server. The daemon running for each simulation controls the start/stopping of the software as well as communicating with the server. The communication includes updating the server with the current status, sharing reduce logs, and two-dimensional images. The REST server provides these updates to the single page web application.

3.2 Flare Simulations

3.2.1 Mesh Resolution Impacts

Figure 9 shows the temporal evolution of CO_2 mass fraction shown through the central plane of the simulation domain for the 2*Delta simulation (0.02 m cell size). The snapshot at 1.5 sec shows the residence time is approximately 1.5 sec. Figure 10, showing the time-averaged and instantaneous spatial averages of CO_2 concentration and temperature at the monitor plane indicate that steady state conditions are reached after approximately 8 sec, which is four times the residence time (i.e. the large eddy turn over time is much greater than that of the residence time [12]). Figure 9 shows that most of the combustion happens in the region near the flare fuel tip, which is consistent with the high temperature near the flare fuel tip. Aerodynamic interactions associated with the flare stack and flare tip geometries and the ambient air conditions, along with interactions with the plume of hot combustion gasses cause eddies to strip unburnt fuel away from the combustion zones. Subsequent dilution with air and cooling of the stripped fuel permanently results in combustion inefficiency. This physical behavior, fundamentally predicted by Arches (i.e. LES model), is a notable challenge to RANS based turbulence models.

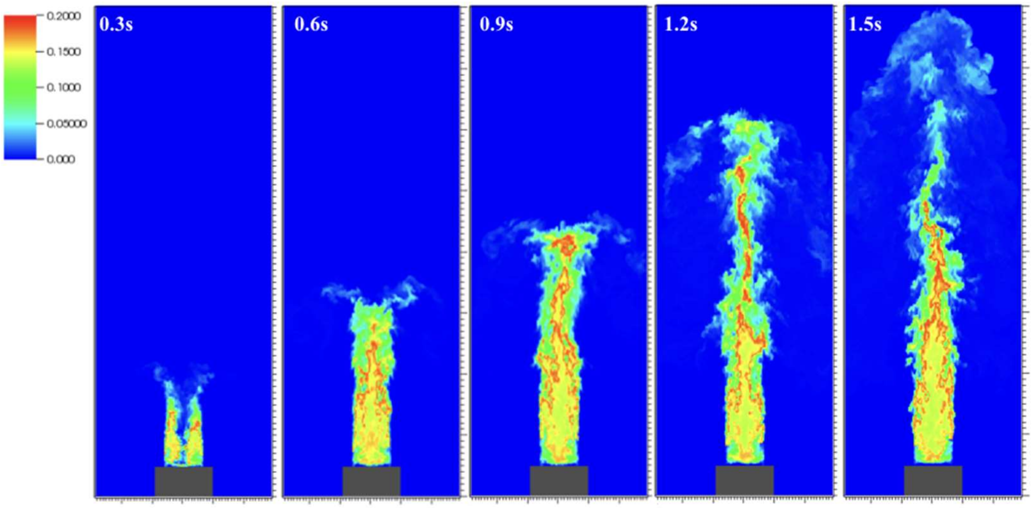


Figure 9: Temporal evolution of unsteady CO2 mass fraction field for diverse timesteps

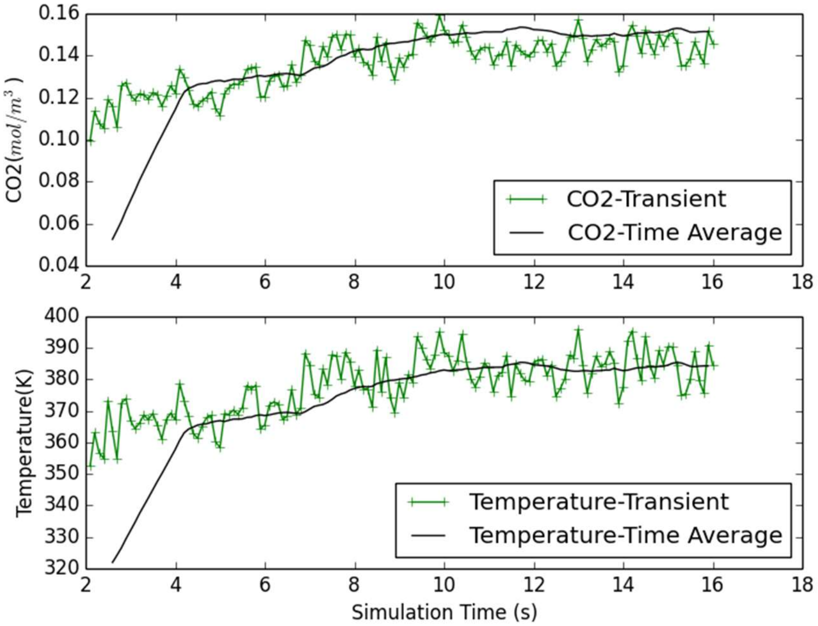


Figure 10: The plane-averaged CO2 and Temperature as a function of time at monitor surface

Combustion Efficiency (CE) is computed as the percentage of the total hydrocarbon stream entering the flare that burns completely to form only carbon dioxide and water. Numerically, if CO_2 in the air is assumed to be negligible, this is represented as

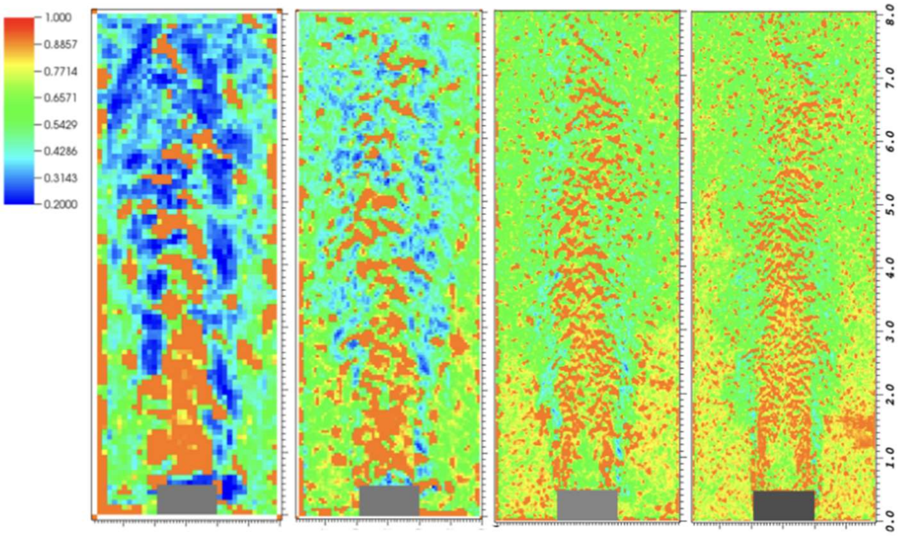
$$\mathrm{CE}(\%) = \left(\frac{CO_2(plume)}{CO_2(plume)+CO(plume)+\sum hydrocarbons(plume)}\right) \times 100 \qquad \text{Eq. 5}$$

where, CE (%) = combustion efficiency (%); CO_2 (plume) = volume concentration of carbon dioxide in the plume (ppmv) after combustion has ceased; CO (plume) = volume concentration of carbon monoxide in the plume (ppmv) after combustion has ceased; ∑hydrocarbons (plume) =

volume concentration of all the unburned hydrocarbons in the plume after combustion has ceased multiplied by the number of carbons in the hydrocarbon (ppmv).

For this simulated flare geometry, the monitor surface was selected as Z = 7.8 m, where the spatial time-average and instantaneous values of CO_2 and temperature are plotted in Figure 10. The plane-average CO_2 mass fraction and temperature at the monitor surface oscillate around 0.12 and 368 K, respectively, after a physical time of 5 seconds. Figure 10, together with the corresponding calculations of CE were used to determine that steady-state conditions were achieved by a physical time of 10 seconds.

One measure for assessing the quality of LES is based on the ratio of resolved turbulent kinetic energy (TKE) to total turbulent kinetic energy (TKE). A target ratio is 80% as referenced in many previous studies [13, 14, 15, 16]. Figure 11 shows the contours of the calculated fraction of resolved TKE for each of the mesh sensitivity simulations. The finest mesh size of 0.015 m achieves an overall spatial average of 73% whereas the coarsest mesh size of 0.08 m achieves an overall spatial average of 60% (Table 3). The lower ratios are observed to surround the flare plume. This is the shear layer where the flare plume entrains the surrounding combustion air into the reaction-zone and where accurate representation of turbulent mixing is critical.



Mesh Size=0.08m, LES Ratio=0.5265

Mesh Size=0.04m, LES Ratio=0.6034

Mesh Size=0.02m, LES Ratio=0.688

Mesh Size=0.015m, LES Ratio=0.728

Figure 11: Contours of LES quality assessment at the central plane

Figure 12 shows the corresponding time dependent spatial averages of gas temperature, CE, fuel mass fraction and CO2 mass fraction at the monitor plane. For the 0.08 m mesh resolution, the results show that even after 16 sec of physical time, the simulation has not yet reached steady state as the fuel mass fraction and combustion efficiency continue to change with time. Furthermore, the CE for this case is very low and is continuing to decrease. Decreasing the mesh size from 0.08 m to 0.4 m and subsequently to 0.02 m monotonically increases the predicted CE to approximately 80% and then to 92%.

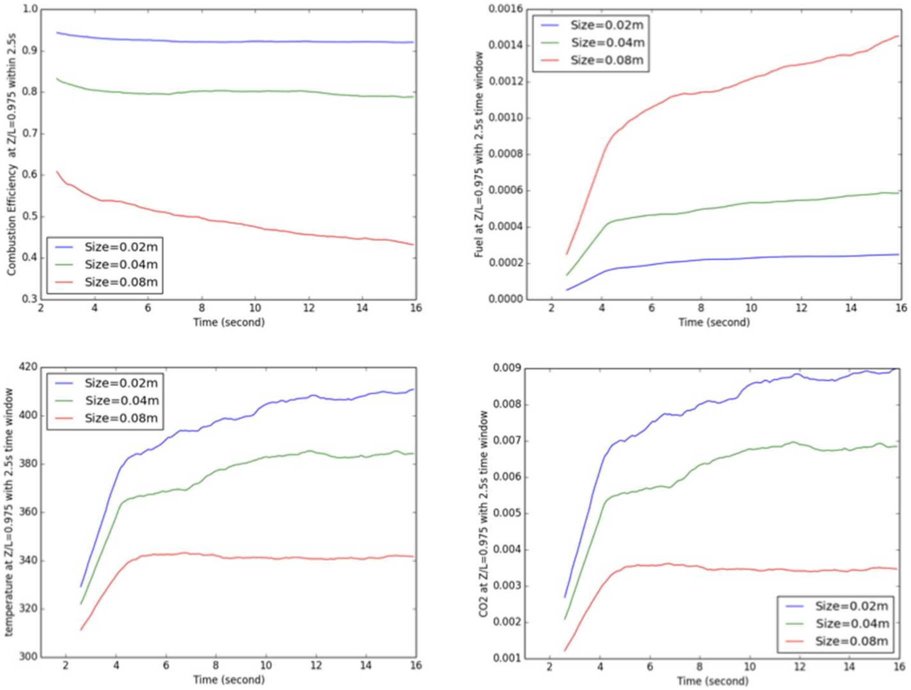


Figure 12: Temporal evolution of plane-averaged variables at the monitor surface

3.2.2 Subgrid-scale model impacts

The sigma model does not use a dynamic procedure to determine the model constant (C_m) in Eq.4, as carried out for the dynamic Smagorinsky model. This difference has been observed to reduce the computer run-time by a three-fold compared to the dynamic Smagorinsky model. The computed LES ratio profiles based on the Sigma model shown in Figure 13 evolve similarly to the Smagorinsky model in Figure 11, which indicates a similar distribution of the resolved TKE distribution between these two models.

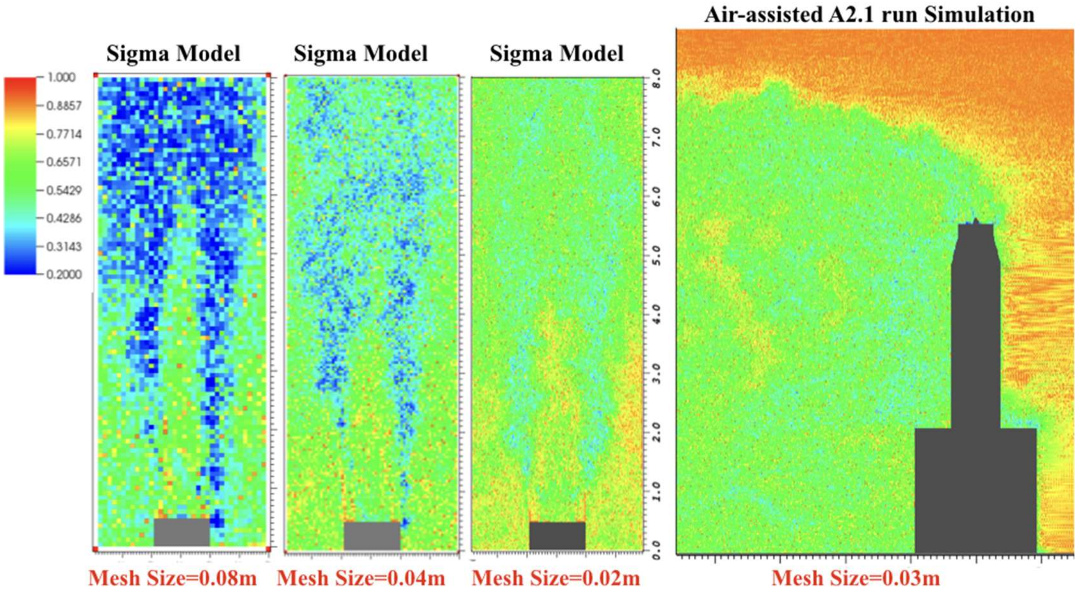


Figure 13: Contours of LES quality assessment at the central plane

Comparison of time-averaged velocity profiles for the two SGS model results are shown in Figure 14. Overall, the Sigma model results for axial velocity (W) compare similarly to the Smagorinsky model, since the axial motion is mainly driven by the injected momentum. Significant differences in the x-direction (U) velocity occurs at approximately z/D=1, where the peak value of the Smagorinsky model is three times of the value of Sigma model. Similar differences are seen with the radial velocity profiles shows a noticeable variation brought by these two SGS models.

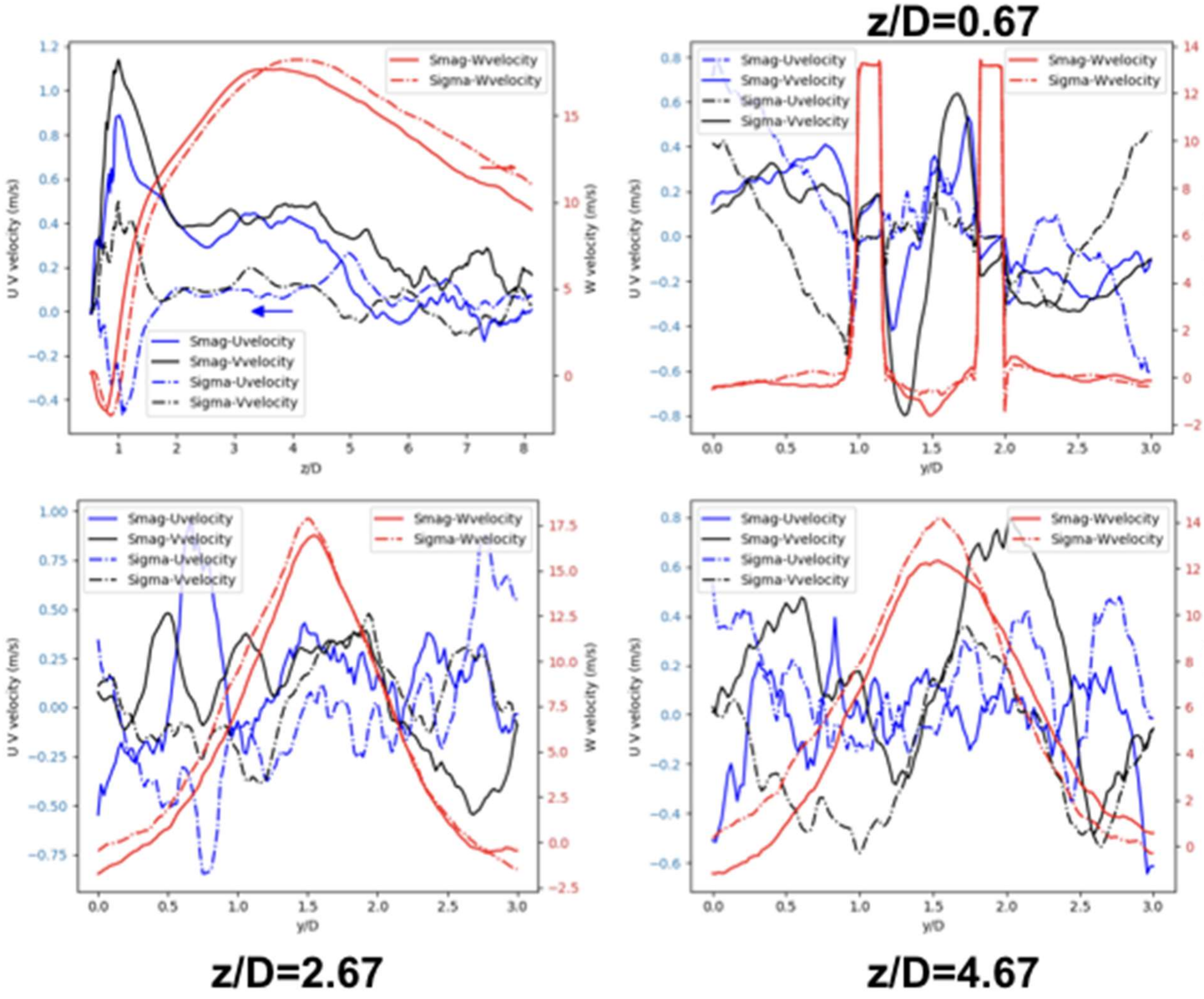


Figure 14: Time-averaged velocity (U, V, W) distribution along the axial and radial direction

Comparisons of the predicted gas temperature and CO_2 mass fraction are shown in Figure 15. Profiles in the radial and axial directions are remarkably similar for the two SGS models despite the U and V velocity differences seen in Figure 14.

Figure 16 shows the temporal evolution of monitor plane-averaged CE, gas temperature, CO_2 mass fraction and unburned fuel mass fraction at the monitor surface for the sigma model and dynamic Smagorinsky model. The response of increasing the mesh resolution on combustion results is very similar for these two SGS models. Increasing the mesh resolution increases the CE, temperature and CO_2. The instantaneous scalar dissipation rate of Sigma model results at the central plane is shown in Figure 17. It can be seen that within the flare plume, the scalar mixing rates are higher for the Sigma model. This is likely playing a significant role in the increased CE predicted for the Sigma model shown in Figure 16, where the CE for the Sigma model results always exceeds that of the dynamic Smagorinsky results.

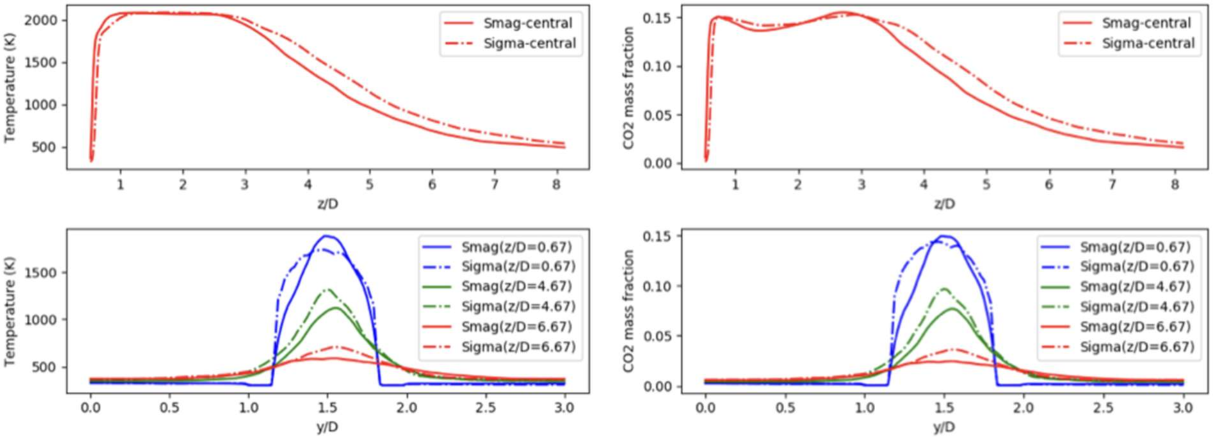


Figure 15: Time-averaged temperature (K) and CO_2 mass fraction distribution along the axial and radial direction

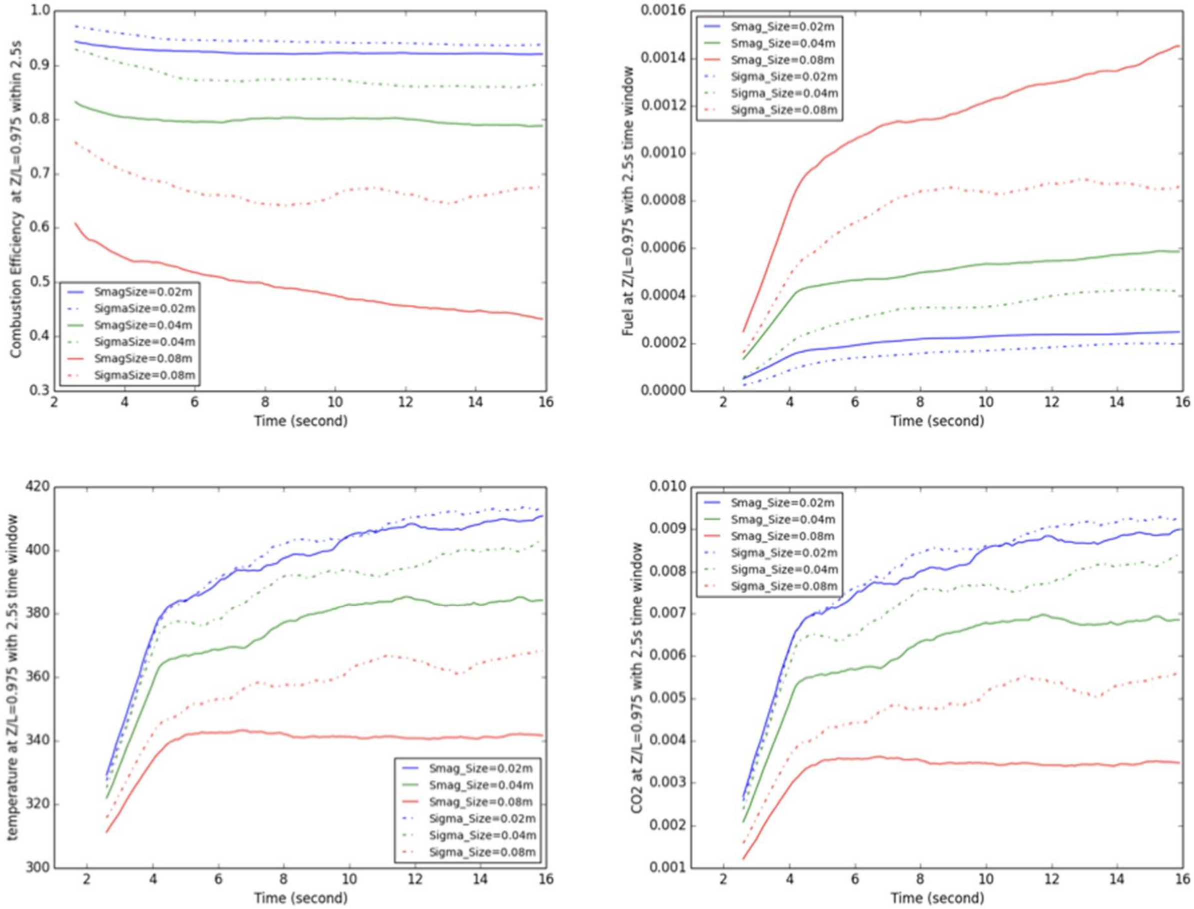


Figure 16:Temporal evolution of plane-averaged combustion parameters

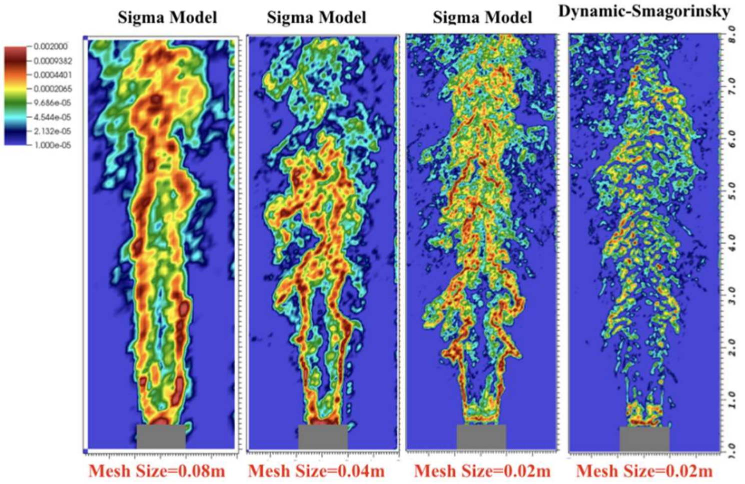


Figure 17: Instantaneous scalar mixing rate at the central plane for Sigma model

4 Conclusions

The key objective of this research is to harden the UCF, particularly the Arches LES model component, to the commercial simulation of industrial flares. Our efforts have focused on two specific areas: 1) Evaluation of trade-offs between model accuracy and computational requirements, and 2) Development of a web-based interface to make the software and HPC hardware accessible to non-expert end users. Regarding item 1, particular focus has been devoted to examination of mesh resolution and selection of SGS model. This research shows that key combustion results including combustion efficiency are highly dependent on mesh resolution even when the average fraction of resolved turbulent kinetic energy exceeds 60%. Although mesh resolution capable of resolving 80% of the TKE may be a desirable target, computational costs associated with this level of resolution are significant, and may not be generally feasible in a commercial environment. In addition, this research has shown that the choice of the SGS model also impacts key combustion results including CE. Comparisons between the widely used dynamic Smagorinsky model and the Sigma model show that the Sigma model exhibits a three-fold decrease in run-time. Further work is necessary to evaluate if there is a significant impact on predictive capability associated with this observed reduction in computational cost.

Progress has been made regarding item 2 and the development of a user-friendly web-based interface for Arches. The Flares UI has been demonstrated to: 1) specify geometry, inlet conditions, and boundary conditions for an Arches simulation, 2) monitor case progress, and 3) post process case results. Work is in process with development of the HPC daemon to initiate the corresponding Arches simulations at a commercial HPC facility (i.e. Nimbix).

References

[1] Council on Competitiveness, "US Manufacturing-Global Leadership through Modeling and Simulation," 2009.

[2] President's Information Technology Advisory Committee, *Computational Science: Ensuring America's Competitiveness. Report to the President, June 2005,* 2005.

[3] "What's Behind the HPC Justification Problem," HPCwire, 9 October 2014. [Online]. Available: http://www.hpcwire.com/2014/10/09/behind-hpc-justification-problem.

[4] P. Smith, A. Jatale, J. Thornock and S. Smith, "A validation of flare combustion efficiency simulations," in *AFRC Industrial Combustion Symposium*, Salt Lake City, 2012.

[5] J. Thornock, P. Smith and A. Chambers, "LES Simulations of Sour Gas Flares in Western Canada," in *AFRC Industrial Combustion Symposium*, Houston, 2009.

[6] D. T. Allen and V. M. Torres, "TCEQ 2010 Flare Study Final Report," Texas Commission on Environmental Quality. PGA No. 582-8-862-45-FY09-04, August 1, 2011.

[7] M. Berzins, J. Schmidt, Q. Meng and A. Humphrey, "Past, Present, and Future Scalability of the Uintah Software," Proceedings of the Blue Waters Workshop, 2012.

[8] M. Cremer, D. Wang, J. Thornock and M. McGurn, "Leveraging the Uintah Computational Framework for Commercial Simulation of Industrial Flares," in *AFRC Industrial Combustion Symposium*, Salt Lake City, 2018.

[9] M. Rieth, F. Proch, O. Stein, M. Pettit and A. Kempf, "Comparison of the Sigma and Smagorinsky LES models for grid generated turbulence and a channel flow," vol. 99, 2014.

[10] F. Nicoud, H. Toda, O. Cabrit, S. Boose and J. Lee, "Using singular values to build a subgrid-scale model for large eddy simulations," vol. 23, 2011.

[11] [Online]. Available: https://kitware.github.io/vtk-js/docs/index.html.

[12] D. Geyer, A. Kempf, A. Dreizler and J. Janicka, "Scalar dissipation rates in isothermal and reactive turbulent opposed-jets: 1-D-Raman/Rayleigh experiments supported by les," vol. 30, no. 1, 2005.

[13] A. Doost, F. Ries, L. Becker, S. Burkle, S. Wagner, V. Ebert, A. Dreizler, F. di Mare, A. Sadiki and J. Janicka, "Residence time calculations for complex swirling flow in a combustion chamber using large-eddy simulations," *Chemical Engineering Science,* vol. 156, pp. 97-114, 2016.

[14] A. Sadiki, S. Agrebi, M. Chrigui, A. Doost, R. Knappstein, F. Di Mare, J. Janicka, A. Massmeyer, D. Zabrodiec, J. Hees and R. Kneer, "Analyzing the effects of turbulence and multiphase treatments on oxy-coal combustion process predictions using LES and RANS," *Chemical Engineering Science,* vol. 166, pp. 283-302, 2017.

[15] I. Celik, Z. Cehreli and I. Yavuz, "Index of resolution quality for large eddy simulations," *Journal of Fluids Engineering,* vol. 127, no. 5, pp. 949-958, 2005.

[16] S. B. Pope, "Ten questions concerning the large-eddy simulation of turbulent flows," vol. 6, 2004.

[17] H. Pitsch, "Large eddy simulation of turbulent combustion," vol. 38, no. 2, 2006.

5 Acknowledgement and Disclaimer

This material is based upon work supported by the U.S. Department of Energy, Office of Science, under Award Number DE-SC0017039.

This paper was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

The authors would like to acknowledge the Argonne Leadership Computing Facility for their allocation of core hours from their discretionary allotment.

The authors would also like to acknowledge the contributions of Dr. Ralph Price of CP Chem, Mr. Markus Martin of Dynamite Digits, and Mr. Scott Evans and Mr. Daniel Pearson of Clean Air Engineering.