Letter Report
TLR-RES/DE/REB-2022-11

Graphite Degradation Modeling and Analysis

December 2022

Joseph Bass
Idaho National Laboratory

Christopher Ulmer,
Raj Iyengar
U.S. Nuclear Regulatory Commission

Division of Engineering
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555–0001

Prepared as part of the Task Order 31310019N0006-31310020F0060, "Technical Support for DOE Component Integrity Codes, Graphite for ANLWRs, and AMT"

This report does not contain or imply legally binding requirements. Nor does this report establish or modify any regulatory guidance or positions of the U.S. Nuclear Regulatory Commission and is not binding on the Commission.

INL/EXT-21-65240

Graphite Degradation Modeling and Analysis

December 2022

Idaho National Laboratory

INL is a U.S. Department of Energy National Laboratory
operated by Battelle Energy Alliance, LLC

INL/EXT-21-65240

Graphite Degradation Modeling and Analysis

December 2022

Joseph Bass
Idaho National Laboratory

Idaho National Laboratory
Idaho Falls, Idaho 83415

www.inl.gov

Prepared for the
U.S. Nuclear Regulatory Commission
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

CONTENTS

FIGURES

ACRONYMS

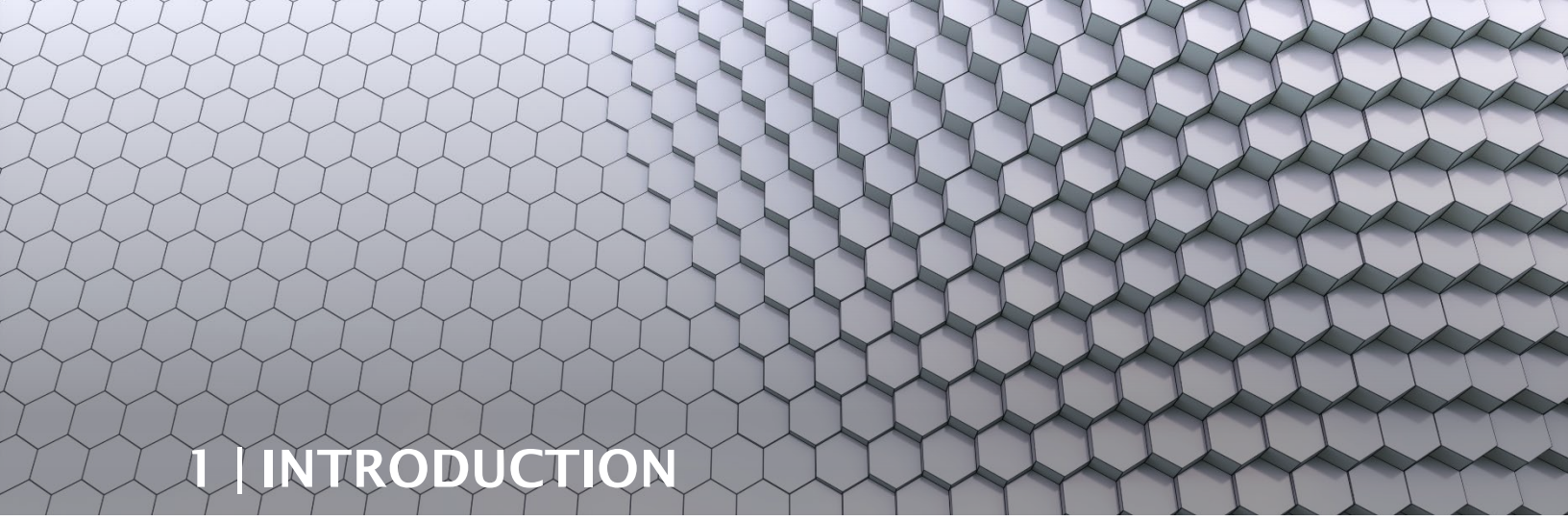
ASA	Active Surface Area
ASME	American Society of Mechanical Engineers
BPVC	Boiler and Pressure Vessel Code
CTE	Coefficient of Thermal Expansion
INL	Idaho National Laboratory
MLE	Maximum Likelihood Estimate
MOOSE	Multiphysics Object-Oriented Simulation Environment
NRC	Nuclear Regulatory Commission
POF	Probability of Failure
SRC	Structural Reliability Class

EXECUTIVE SUMMARY

Graphite is present in the reactor core of several advanced nuclear reactors as a moderator and as core support structures. Because of the significant radiation effects on graphite components, modeling of graphite aging and degradation would help predict graphite behavior throughout a reactor's life cycle. With the U.S. Nuclear Regulatory Commission's (NRC) upcoming endorsement of Section III, Division 5 of the American Society of Mechanical Engineers Boiler and Pressure Vessel Code (ASME BPVC), it is of particular interest to develop modeling capabilities relevant to the Division 5 assessment methodologies for graphitic components. The Multiphysics Object-Oriented Simulation Environment (MOOSE) developed at Idaho National Laboratory provides an ideal platform to develop graphite modeling capabilities. The models discussed in this report are currently in GRIZZLY, which is built in the MOOSE framework and is used for modeling the degradation of nuclear power plant systems, structures, and components.

A graphite component in a nuclear reactor core is subjected to a variety of stresses and can experience degradation during normal and off-normal operation. Understanding how a graphite component will behave in service is essential to ensuring core structural stability and safe reactor operation. This report summarizes a graphite modeling tool currently under development at Idaho National Laboratory. The tool incorporates the effects of several loads which are prominent during service as well as oxidation and irradiation prior to turnaround. This tool is intended to be used to help assess the design of graphite components by utilizing design code rules found in Section III, Division 5 of the ASME BPVC. Specifically, the tool is intended for use with the methodologies found within the Full and Simplified assessments from Article HHA-3000 to verify that a graphite component has an acceptably low probability of failure.

The graphite modeling capabilities discussed in this report are primarily concerned with computing stresses as well as the effects of oxidation and irradiation on a graphite component. These capabilities are applicable for many reactor designs, but additional modeling capabilities may be required based on the reactor design. For example, molten salt reactors have the added complication that the salt may penetrate or abrade the graphite. Further development of the graphite tool presented in this report to account for phenomena relevant to molten salt reactors is currently in progress.



1 | INTRODUCTION

Graphite is used as a moderator, reflector, and structural component in many high temperature nuclear reactor designs [1]. When assessing the design of a graphite core component, a designer must consider the anticipated stresses imposed upon the component and any degradation issues which may be experienced during normal and off-normal operation. Developing models, like the ones discussed in this report, is essential for understanding how a component will behave in a reactor environment and for ensuring safe operation throughout the component's life cycle. One of the goals of this work is to provide a tool which can be readily accessed and used by interested parties to assist in the design of graphite components. Consequently, Section III, Division 5 of the American Society of Mechanical Engineers (ASME) Boiler and Pressure Vessel Code (BPVC) graphite component design code rules can be used with this tool.

The purpose of this report is to provide an outline for the graphite modeling tool being developed at the Idaho National Laboratory (INL). The tool is comprised of multiple graphite behavior models as well as a reliability analysis. This report will cover the graphite physics which is included in the behavior models as well as some of the limitations of the tool. The report will also include the inputs, outputs and numerical formulations associated with the various graphite models and how they interact with a reliability analysis.

This report is laid out in the following way. Section 1.1 provides a very brief background on nuclear graphite. This background is intended to highlight physics and phenomena which effect graphite behavior in nuclear applications. Section 1.2 will discuss methodologies for assessing a graphite component according to the ASME BPVC. Section 2 outlines the graphite modeling capabilities developed in this project and Section 4 presents conclusions. A brief user guide is provided in Appendix A. It outlines how to generate input files in order to run the graphite modeling capabilities discussed in the body of the report.

1.1 Graphite Behavior

Research on graphite properties has been conducted for decades. General trends in graphite properties as a function of temperature, oxidation, and irradiation are reasonably well substantiated, although, many different graphite grades exist, and every grade of graphite behaves slightly differently under similar operating conditions [2,3]. Therefore, while general trends in the material properties as a function of environment are known, data specific to a particular grade are not necessarily readily available.

A designer must understand graphite behavior to appropriately assess how a component may fail. This begs the question, "what is failure?" This question is deceptively complex, and the answer will vary depending on the specifics associated with each component's function. Although, one sure way to fail any

component is to have stresses which are too high. In a nuclear environment, stress can be generated in a graphite component from a multitude of sources. These include mechanical stresses such as those induced by the weight of stacked graphite components or pressure from the cooling fluid (i.e., dense molten salt coolant). However, the most significant stresses anticipated during nuclear operations will result from temperature and received irradiation dose gradients within a component. Of specific concern are stresses generated from non-uniform irradiation-induced dimensional changes. Stresses can be partially alleviated by irradiation-induced creep. While this is not an exhaustive list of what may cause stress within a graphite component in a reactor environment, these are some of the most prominent sources. A more complete list of loads to consider can be found in HHA-3122 in Section III, Division 5 of the ASME BPVC.

Some other important considerations in the discussion of graphite core component behavior are degradation and property changes. Graphite will readily oxidize if it is exposed to oxidants at sufficiently high temperature. The oxidation process is controlled by two main factors. One is the oxidant's diffusivity though the graphite and the second is the local kinetic rate. The diffusivity is a function of temperature as well as the graphite microstructure, and the local kinetic rate is a function of the temperature and the available active surface area (ASA) on the graphite crystallite. These two factors control when oxidation occurs, where oxidation occurs, and the extent of oxidation damage. Oxidation damage causes dramatic changes to a graphite's properties and understanding where and how much these local properties change is key to understanding how a component's bulk behavior is affected. Irradiation-induced graphite property changes are complex and can be significantly affected by the irradiation temperature as well as received dose. Accounting for oxidation, dose, and temperature effects on properties is necessary for understanding a component's behavior.

1.2 ASME Graphite Component Assessment

Article HHA-3000 in Section III, Division 5 of the ASME BPVC provides guidance on designing graphite core components [4]. Throughout the remainder of this report, text from Section III, Division 5 of the ASME BPVC will be referenced using their HHA article designations. HHA-3000 provides three methodologies for assessing a component: design by test, Simplified assessment, and Full assessment. The design by test is outlined in HHA-3240 through HHA-3243, and it provides guidelines on how to verify acceptable reliability from experimental testing. The Simplified assessment is contained in HHA-3220 and is performed by comparing an allowable stress value to the expected peak stress calculated in a component. The allowable stress is computed based on the graphite's material properties as well as the required reliability which is determined from the Structural Reliability Class (SRC). The Full assessment is slightly less conservative than the Simplified assessment and considers the full stress distribution in the component. The difference in conservatism between the two methods is not easily quantified and will depend on the variance in the stress distribution in the component. In the Full assessment, the probability of failure (POF) is computed using the method in HHA-3217. Then, the computed POF is compared to the allowable POF determined from the SRC. This report will further discuss the application of the Simplified and Full assessments in Section 2.3. The design by test methodology will not be discussed further, because unlike the Simplified and Full assessments which make use of computing the internal stress in a component, the design by test methodology is experimentally based.

2 | GRAPHITE COMPONENT ASSESSMENT

The purpose of this graphite modeling tool is to help assess the structural integrity of a graphite component's design. The tool accomplishes this by calculating stresses in a graphite component and potentially implementing those stresses in accordance with Section III, Division 5 of the ASME BPVC. The model which computes the stresses in graphite can accommodate mechanisms like oxidation and irradiation which may cause property changes during the life cycle of a component. A schematic of how this assessment methodology is structured is shown in Figure 1. This schematic includes all necessary inputs and outputs of the tool with the ultimate result coming from the ASME BPVC assessment. The following discussion will use the schematic as a road map to outline how the tool works.

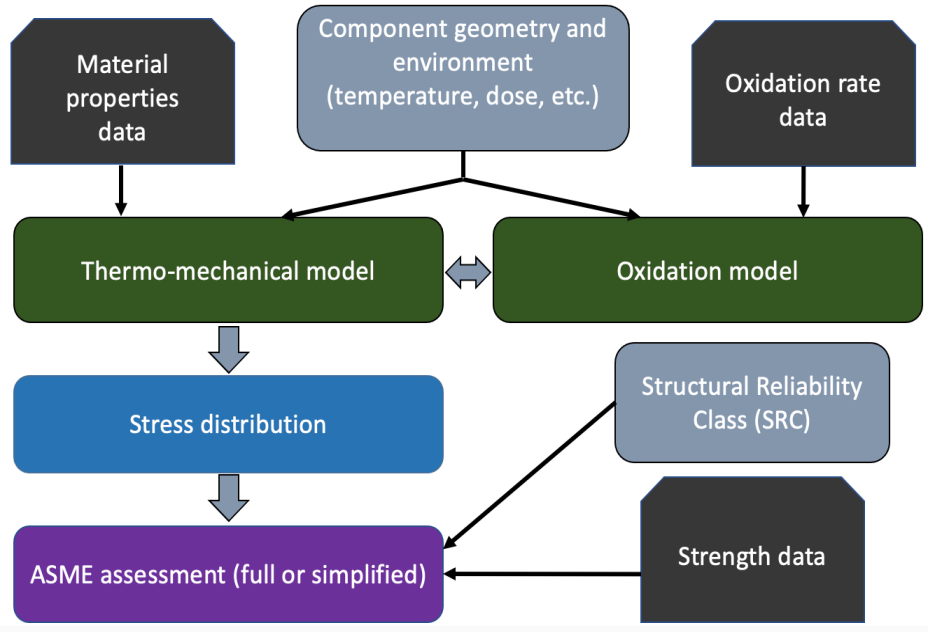


Figure 1. Schematic for assessing a graphite component based on the ASME BPVC.

In the flow diagram, there are three distinct models which require inputs. These models are labeled "Thermo-mechanical model", "Oxidation model" and "ASME assessment (Full or Simplified)" which will be referred to as the "Reliability model" for the remainder of this report. The Thermo-mechanical model and Oxidation model are implemented in the Multiphysics Object-Oriented Simulation Environment (MOOSE) which is an open-source, parallel finite-element framework. The Reliability model is implemented using Python. The other blocks in the schematic besides the "Stress distribution", which is an output of the Thermo-mechanical model, are required inputs. These required inputs mainly correspond to material property data as well as component geometry and the environment experienced in the reactor (temperature, dose, oxidant concentration, etc.).

This tool is not intended to provide a comprehensive analysis of a graphite component. The tool is only intended to model stress generation in graphite which has been oxidized or is undergoing irradiation but prior to the turnaround dose. If applicable, these stresses can be used in the Full or Simplified assessments in the ASME Code. The turnaround dose is identified on a dimensional change *versus* dose plot as the dose value where the graphite stops shrinking and begins to expand. The model is limited to pre-turnaround doses because the scatter in the graphite behavior and properties becomes much larger after turnaround and limited data exists. The model is only as good as its parameterization, so this large uncertainly in post-turn-around data creates a large uncertainty in the simulation results and quantifying that uncertainty is beyond the scope of this work. Modeling the combined effect of irradiation and oxidation is possible using the Thermo-mechanical and Oxidation models, but it is not an intended application. At this time, the property relationships of graphite with combined oxidation and irradiation effects have not been well explored, which means material property data required by the models is not available. The tool's reliability analysis is also limited to the application of the 2021 edition of the ASME BPVC rules. This means that topics which are not directly addressed in the ASME BPVC will not be included in the Reliability model.

2.1 Thermo-mechanical Model

2.1.1 Thermo-mechanical Model Introduction

The evolution of the stresses and temperatures within a graphite component are computed by the Thermo-mechanical model. In the Full and Simplified assessment methodologies, computing stresses is necessary to assess the reliability of a component. The inputs for the Thermo-mechanical model are the component geometry, environment experienced throughout the life cycle (temperature, dose, mechanical loads), and material properties. The material properties which are required are the elastic modulus, thermal conductivity, coefficient of thermal expansion (CTE), and Poisson's ratio. The material properties of graphite change as a function of temperature, irradiation dose, and oxidation damage. It is suggested that input material properties are implemented as a function of these states.

2.1.2 Thermo-mechanical Model Numerical Formulation

The state variables which can evolve in the Thermo-mechanical model are the strains, ϵ, temperature, T, and dose, γ. The constitutive relation which governs the strain in the model is

$$\epsilon_{total} = \epsilon_{therm} + \epsilon_{irr} + \epsilon_{creep} + \epsilon_{elastic} \tag{1}$$

where ϵ_{total} is the total strain, ϵ_{therm} is strain from thermal expansion, ϵ_{irr} is strain from irradiation-induced swelling, ϵ_{creep} is strain from irradiation-induced creep, and $\epsilon_{elastic}$ is the elastic strain. The functional form of the thermally induced eigenstrain is

$$\epsilon_{therm} = CTE(T - T_o) \tag{2}$$

where CTE is the coefficient of thermal expansion, and T_o is the temperature where the strain is assumed to be zero. The irradiation-induced dimensional change is traditionally expressed as a function of temperature and dose.

$$\epsilon_{irr} = f(T, \gamma) \tag{3}$$

Experimental data is available to fit the relationship between the strain, dose, and temperature for multiple grades of graphite. MOOSE is flexible, and the functional form of the strain relationship can incorporate almost any function of the state variables. The irradiation-induced creep is comprised of a primary and secondary component.

$$\epsilon_{creep} = \epsilon_{Primary} + \epsilon_{Secondary} \tag{4}$$

The functional form of the primary- and secondary-creep strains is

$$\epsilon_{Primary} = A\sigma\big(1 - exp(-b\gamma)\big)/E \approx \sigma/E \tag{5}$$

$$\epsilon_{Secondary} = K\sigma \tag{6}$$

Where A and b are primary creep parameters, σ is the stress, E is the elastic modulus, and K is the secondary-creep coefficient. The stresses in the component can be computed from

$$\sigma = E\epsilon_{elastic} \tag{7}$$

2.1.3 Thermo-mechanical Model Parameterization

In this model, the thermal conductivity, elastic modulus, CTE, and irradiation-induced swelling should be implemented as a function of the temperature, dose, and, if oxidation is being simulated, density. As each grade of graphite behaves differently, a different parameterization will be required for every grade. Implementing the parameterization is relatively straight forward in MOOSE. The eigenstrains from the irradiation-induced swelling and temperature can be implemented using the "ComputeVariableEigenstrain" material. The "prefactor" input in the "ComputeVariableEigenstrain" material is a "DerivativeParsedMaterial" object which is formulated as a function of the states. The dose and time-derivative of the dose can be implemented as AuxVariables which have associated "FunctionAux" AuxKernels. The thermal conductivity function is implemented in the GraphiteThermalGaseous.C source file by editing the kT variable. An isotropic elastic modulus which is a function of the states can be implemented by inputting the elastic modulus function as a "DerivativeParsedMaterial" and using the "DerivativeParsedMaterial" as an input to the "ComputeVariableIsotropicElasticityTensor" material object. The graphite irradiation creep behavior is built into the "GraphiteIrradiationCreep" material. A more in-depth discussion of the code necessary for running the Thermo-mechanical model is available in the user guide in Appendix A.

A parameterization for IG-110 is provided in equations 8 - 11 below. This parameterization was computed using MATLAB's optimization toolbox to fit experimental data. It should be noted that this parameterization is not intended to be used past turnaround. In the following equations, T is temperature in degrees Celsius and γ is the dose in dpa. The CTE, with units of 1/Kelvin, irradiation-induced swelling ϵ_{irr}, elastic modulus, E, in GPa and thermal conductivity, K, in W/(m °C) can be expressed as:

$$CTE = 4.827e-6 - 3.9413e-11\,T - 1.149e-7\,\gamma - 2.648e-11\,T\,\gamma + 3e-9\,\gamma^2 \quad (8)$$

$$\epsilon_{irr} = 0.0006351 - 6.23e-7T - 0.003476\,\gamma - 4.26e-7\,T\,\gamma - 0.0002324\gamma^2 \quad (9)$$

$$E = 12.41 - 0.0007386\,T + 2.838\,\gamma - 0.00102\,T\,\gamma - 0.0753\,\gamma^2 \quad (10)$$

$$K = 117.8 - 0.08176\,T - 2.819\,\gamma \quad (11)$$

Images of the parameterization function fits on top of the experimental data are shown in Figure 2. This parameterization is provided in an example input file which is available in GRIZZLY which is part of the MOOSE framework.

To compute stresses in an oxidized component, the material properties must be computed as a function of mass loss. Currently, very little data exist on the combined effects of oxidation and irradiation, so a parameterization combining the two effects is not advisable. The material properties as a function of mass loss are shown in equations 12-14 and the fits on top of the experimental data are shown in Figure 3. In the following equations X is the normalized mass loss and T is in Kelvin.

$$CTE = 3.397e-6 + 1.446e-9T - 1.861e-8X - 3.54e-13T^2 + 8.08e-12\,XT \quad (12)$$

$$E = 10.67\,exp(-0.05256\,X) \quad (13)$$

$$K = 135.9 - 0.06048T - 148.9\,X + 0.06049\,X\,T + 14.92\,X^2 \quad (14)$$

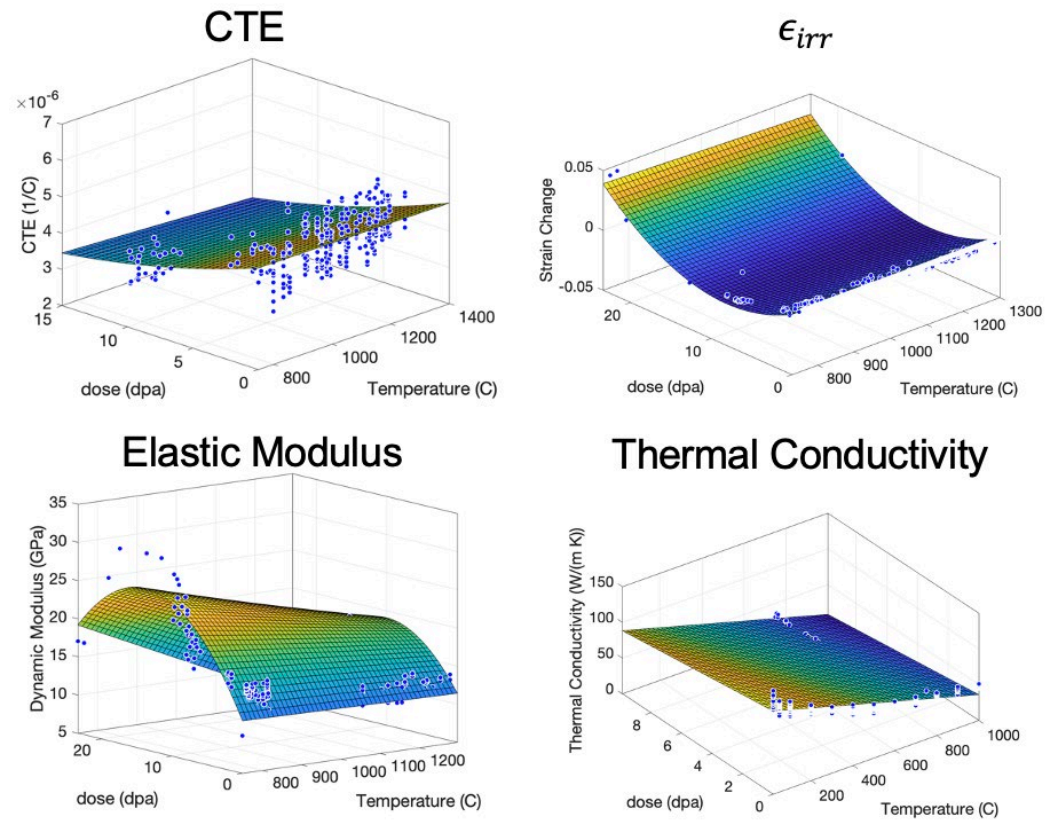


Figure 2. IG-110 parameterization of material properties as a function of temperature and dose.

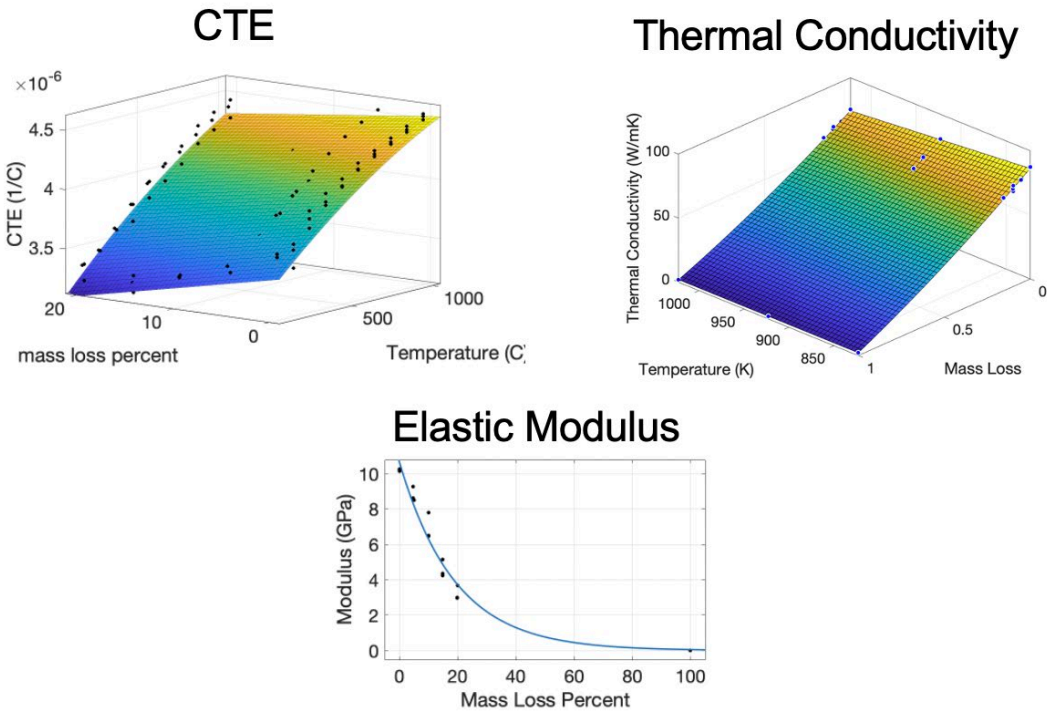


Figure 3. IG-110 parameterization of material properties as a function of temperature and mass loss.

2.1.4 Thermo-mechanical Model Discussion

The constitutive strain relationship allows for all loads mentioned in Section 1.1 to be incorporated. One of the advantages of MOOSE is a simple method for incorporating additional physics. Incorporating additional eigenstrains or modeling physics associated with a specific reactor type can be added in MOOSE with reduced effort relative to other modeling platforms. The user guide in Appendix A provides guidance on setting up an input file to run the Thermo-mechanical model.

The Thermo-mechanical model requires parameterization for multiple inputs including the material properties as well as the relationships between the strains and state variables. As each graphite grade requires a grade-specific parameterizations, it is not feasible to account within the model for the variation in grades. Although, parameterizations for select grades will be provided in the tool. This Thermo-mechanical model is based upon the work done by Andrea Nicolas, so the interested reader will find a more in-depth discussion of a similar model in her work [5].

2.2 Oxidation Model

2.2.1 Oxidation Model Introduction

The Oxidation model computes when, where, and to what extent oxidation damage occurs. Perhaps most importantly, this model determines the density profile which is generated within a graphite component. The required inputs for this model are the component geometry, oxidant concentration (boundary conditions), environment (temperature), and material properties.

The two nontrivial material properties needed to parameterize the Oxidation model are related to the chemical species diffusion and the ASA of the graphite. Note that both parameters will vary as a function of the local density in the graphite. The model will provide parameterizations for medium-grained NBG-18 and fine-grained IG-110. Unlike the Thermo-mechanical model, material property parameterization of the Oxidation model for additional grades may be difficult due to scarcity of experimental data combined with a more complex parameterization procedure.

2.2.2 Oxidation Model Numerical Formulation

The state variables in the Oxidation model are the density, ρ, chemical species concentration, and temperature, T. Note that the Oxidation model can be coupled to the Thermo-mechanical model. The shared state variables between the models are the temperature and the density. Coupling through the density state variable requires that material properties in the Thermo-mechanical model are implemented as a function of density. The global reaction between graphite and oxygen in the Oxidation model is

$$C(s) + (1 - 0.5x)O_2 \rightarrow xCO + (1 - x)CO_2 \tag{15}$$

where $C(s)$ is the carbon in the graphite and x is the molar fraction of CO produced in the reaction products. The value of the x can change as a function of the states. The change in density of the graphite is computed by

$$\frac{\partial \rho}{\partial t} = k_{eff}^{"} S_A [O_2] \tag{16}$$

where $k_{eff}^{"}$ is the effective reaction rate normalized to the ASA, S_A is the ASA per unit volume and $[O_2]$ is the concentration of oxygen. The mass transfer and flux are computed by the following five equations

$$N_i \cong -[C_T] D_{eff} \nabla y_i + y_i (N_i + N_m) \tag{17}$$

$$\frac{\partial \varepsilon [CO2]}{\partial t} = -\nabla N_{CO_2} + (1 - x) k_{eff}^{"} S_A [O_2] \tag{18}$$

$$\frac{\partial \varepsilon [CO]}{\partial t} = -\nabla N_{CO} + x k_{eff}^{"} S_A [O_2] \tag{19}$$

$$\frac{\partial \varepsilon [O2]}{\partial t} = -\nabla N_{O_2} - \left(1 - \frac{x}{2}\right) k_{eff}^{"} S_A [O_2] \tag{20}$$

$$\frac{\partial \varepsilon [I]}{\partial t} = -\nabla N_I \tag{21}$$

Here, C_T is the total gas concentration, D_{eff} is the effective diffusivity, y_i is the mole fraction of chemical species i, N_i and N_m are the flux of specific chemical species i and arbitrary species m, ε is the porosity by volume fraction and i can be any additional chemical species. Equations 17-21 also account for the generation of carbon dioxide and carbon monoxide as well as the depletion of oxygen.

The reaction between graphite and oxygen is exothermic. At low temperatures where the reaction rate is slow, the effect of this exothermic reaction will likely be negligible. This is because at low temperatures, heat production is slow, and the generated heat can be dissipated before any significant temperature change occurs. At higher temperatures, the heat release from this reaction may cause a significant temperature change. The temperature evolution is computed by

$$\frac{\partial (\rho C_p T)}{\partial t} = \nabla \cdot (k_T \nabla \mathrm{T}) + k_{eff}^{"} S_A [O_2] \Delta H_{rx} \tag{22}$$

where C_p is the specific heat of graphite, k_T is the thermal conductivity, and ΔH_{rx} is the heat of reaction. The thermal conductivity in the model is implemented as a function of the states.

2.2.3 Oxidation Model Parameterization

The oxidation behavior of each graphite grade is different and therefore each grades requires its own parameterization. There are parameters associated with two physical features that must be parameterized. These are the diffusivity of the oxidants and the ASA parameters. Both will vary as the microstructure changes from oxidation mass loss. Currently, parameterizations are provided for IG-110 and NBG-18. Selecting from one of these grades can be done by editing the "graphite_type" input in the "GraphiteThermalGaseous" material.

To parameterize the model for additional grades, two sets of experimental data are needed. The first set is an effective diffusivity analysis. In the model it is assumed that the oxidant diffusivity varies with the square of the mass loss. This relationship between mass loss and diffusivity is based on theory and has been used in previous oxidation models [6]. Currently, experiments are being performed at INL to confirm this behavior. To parameterize the diffusivity, the "Z" parameter in the GraphiteThermalGaseous.C source file should be set equal to the ratio of the effective diffusivity over the bulk diffusivity. The second needed experimental data set is the oxidation mass loss versus time at very low temperature. This should be determined using the methodology from ASTM D7542. This low-temperature data will allow for diffusion contributions to be minimized and therefore any change in the reaction rate can be attributed to a change in the ASA density. Measurements of the relative ASA have been performed previously at INL [7]. The evolution of the ratio of the ASA density over the initial ASA should be input in the GraphiteThermalGaseous.C source file in the _SA parameter. The final step to parameterize the model is to fit the "rate_scaling_factor" in the GraphiteThermalGaseous material. The parameter should be adjusted until the mass loss versus time trend matches the experimental data. An example of the successful validation is shown in Figure 4. The user guide in Appendix A provides guidance on setting up an input file to run the Oxidation model.

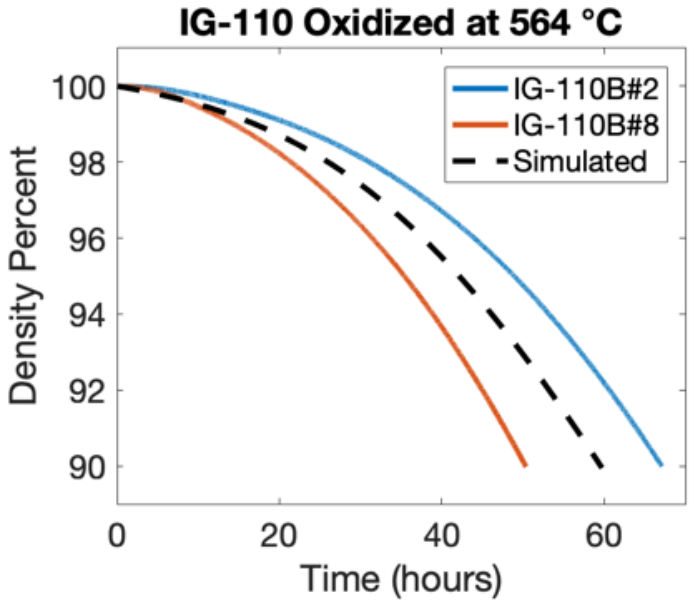


Figure 4. IG-110 Oxidation model parameterization result.

2.2.4 Oxidation Model Discussion

Oxidation in a reactor may be categorized as either acute or chronic. Acute oxidation is likely to occur only in an accident scenario when a large amount of oxidant enters the reactor. Chronic oxidation can occur due to trace amounts of oxidants present in the reactor. The Oxidation model is capable of modeling both types of oxidation by varying the oxidant concentration boundary conditions. It is worth mentioning that graphite can be oxidized by multiple chemical species including oxygen, carbon dioxide, steam as well as others. The Oxidation model was developed to investigate oxidation due to pure oxygen. Oxidation from oxygen is often the primary concern because the activation energy for the reaction between graphite and oxygen is lower than most other oxidants likely to be present. While the model was only parameterized for IG-110 and NBG-18 these parameterizations may be used to provide insights for additional grades. For example, if a grade has a similar grain size, density, and oxidation rate as to IG-110 or NBG-18, then it is likely that they will have a similar oxidation profile. For a more in-depth discussion of oxidation and the model discussed above, the interested reader should investigate the work performed by Joshua Kane [8,9].

2.3 Reliability Model

2.3.1 Reliability Model Introduction

The Reliability model is directly taken from Article HHA-3000 in Section III, Division 5 of the ASME BPVC [4]. The Full and Simplified assessments in the 2021 edition are implemented in Python. The program uses outputs from the Thermo-mechanical model along with material properties and component reliability requirements to assess a graphite component. The Reliability model does not attempt to provide analysis outside of the ASME BPVC. During testing of this model, multiple test problems were run using the ASME methodologies. The following discussion will cover the Full and Simplified assessments and will attempt to highlight potential issues which may arise during application of the assessments.

2.3.2 Reliability Model Numerical Formulation

This section outlines both the Full and Simplified assessments which are found in the ASME Code. Both the Full and Simplified assessments derive an allowable POF from a component's SRC as shown in Table HHA-3221-1 of the ASME BPVC. Designation of the SRC is based on the components service life and is defined in HHA-3111.

In the Simplified assessment, the required inputs are a two parameter Weibull distribution of the graphite's tensile strength, the flexural strength, and the maximum equivalent stress which is computed from FEA modeling. The ASME Code uses a method for determining an equivalent stress based on maximum deformation theory. The methodology for computing the equivalent stress is outlined in HHA-3213. A method for computing the Weibull shape, $m_{95,}$ and scale, Sc_{95}, parameters which correspond to a 95 percent one sided confidence interval and can be found in HHA-II-3100. The design allowable stress, S_g, is computed by

$$S_g = \mathrm{Sc}_{95}\left(-ln(1-POF)\right)^{\frac{1}{m_{95}}} \tag{23}$$

The allowable stress is compared directly to the peak equivalent stress computed in the Thermo-mechanical model. If the peak equivalent stress is lower than the allowable stress then the design passes the Simplified assessment; otherwise, the design fails the Simplified assessment.

For the Full assessment, the required inputs are a three parameter Weibull distribution of the graphite's strength, the element volumes from the mesh used in the finite-element analysis, and the equivalent stress distribution. A method for computing the Weibull shape, $m_{95,}$ scale, Sc_{95}, and threshold, S_0, parameters can be found in HHA-II-3200. Note that while the text and parameter notation in the Code refers to these as the lower bound values of a 95% confidence interval, the supplied equations appear to be maximum likelihood estimates (MLE)s for the parameters. The calculation for the POF is outlined in HHA-3217. The seven-step procedure for determining the POF has seen changes over the last few editions of the Code and will likely be seeing changes in the upcoming editions. As such, it is important for anyone implementing the code to take care to use the appropriate Code edition. Ultimately, if the allowable POF taken from table HHA-3221-1 is higher than the computed POF then the design passes the Full assessment; otherwise, the design fails the Full assessment.

2.3.3 Reliability Model Discussion

Section III, Division 5 is written in a generic way to encompass a variety of reactor types. To this point, the Forward in Section III, Division 5 states "The Code does not address all aspects of these (construction) activities". In the case of analyzing a graphite component, the Full and Simplified assessments do not fully address degradation caused by oxidation or irradiation. HHA- 3141 states that "Oxidation analysis shall be carried out in detail to estimate the weight loss profiles of graphite structures" and provides guidelines for analysis but does not specify how to account for the oxidation gradient in the Simplified or Full assessments. Therefore, the designer must provide an appropriate assessment methodology because material property gradients caused by oxidation and irradiation are outside the scope of the Full and Simplified assessments.

Beyond the limitations in the Code mentioned above, there are some important considerations when implementing the Code. First, while the methodologies in the code are tied to a POF, the assessments are only determining a probability of crack formation in a component. Crack formation does not necessarily directly correlate to a loss of function. This is a topic which will likely be addressed in future versions of the Code. Second, it is important to use a sufficiently fine mesh in the analysis of a component. If a coarse mesh is used, the assessments become less conservative. Currently, the Code does not place any restriction on the mesh size. Third, in the Full assessment, if only a portion of a component is used in an assessment, the POF will be affected. For example, if a component is symmetric about a center plane and a designer only models half the components with the understanding that the stress distribution will be symmetric about the plane, then the designer will need to account for the missing stressed volume in the POF calculation.

3 | CONCLUSION

Understanding graphite behavior is necessary for ensuring safe operation of a nuclear power plant. In this report, a tool was outlined which can model some primary sources of stress generation in a graphite component subjected to a reactor environment. This model strives to account for the effects of oxidation and irradiation prior to turnaround in graphite. One topic which was not addressed in this report was uncertainty and variance in the material property inputs and how those may affect the results. This model does not attempt to account for uncertainty in the model inputs. Although, the model can be used to perform sensitivity studies to determine if uncertainty in the input parameters may lead to significant changes in the results. Although, even if the stress distribution in a graphite component is known, developing a methodology for assessing a graphite component is not a trivial task. Graphite can serve multiple purposes in a reactor which makes defining a systematic method for assessing a component very difficult. This work has used the Full and Simplified assessments from Section III, Division 5 of the ASME BPVC as an assessment methodology.

REFERENCES

[1] Arregui-Mena, J. D., R. Worth, G. Hall, P. Edmondson, A. Giorla, T. Burchell, "A Review of Finite Element Method Models for Nuclear Graphite." Archives of Computational Methods in Engineering, 2018, pp. 331-350.

[2] Matthews, A., J. Kane, D. Swank, W. Windes, "The Degradation of Strength under Varying Oxidizing Conditions for Nuclear Graphite." INL/EXT-19-53723, 2019.

[3] Irradiation damage in graphite due to fast neutrons in fission and fusion systems, IAEA-TECDOC-1154, International Atomic Energy Agency, 2000.

[4] ASME Boiler and Pressure Vessel Code, Section III, Div. 5, 2021 Edition.

[5] Nicolas, A., M. Messner, T. Sham, "Preliminary design analysis workflow for Division 5 HHA-3200 requirements for graphite core components". Applied Materials Division Argonne National Laboratory. ANL-ART-197. August 2020.

[6] Wei, X., Lei, S., Yanhua, Z., "Transient analysis of nuclear graphite oxidation for high temperature gas cooled reactor". Nuclear Engineering and Design. Volume 306, 2016, pp. 138-144.

[7] Kane, J., A. Matthews, C. Orme, C. Contescu, W. Swank, W. Windes, "Effective gaseous diffusion coefficients of select ultra-fine, super-fine and medium grain nuclear graphite". Carbon, Volume 136, 2018, pp. 369-379.

[8] Kane, J., A. Matthews, W. Swank, W. Windes, "Effects of air oxidation on the evolution of surface area within nuclear graphite and the contribution of macropores" Carbon, Volume 166, 2020, pp. 291-306.

[9] Kane, J., C. Contescu, R. Smith, G., Strydon, W. Windes, "Understanding the reaction of nuclear graphite with molecular oxygen: Kinetics, transport, and structural evolution". Journal of Nuclear Materials, Volume 493, 2017, pp. 343-367.

[10] Spencer, B. W., Backman, M., Chakraborty, P., Schwen, D., Zhang, Y., Huang, H., Bai, X., Jiang, W., "Grizzly Usage and Theory Manual". Light Water Reactor Sustainability Program, 2016, INL/EXT-16-38310.

APPENDIX A | USER GUIDE

The user guide portion of this report covers how to implement the graphite model discussed in the previous sections. Accessing the graphite modeling capabilities discussed in this report requires a user to obtain access to GRIZZLY which is one of the animals in the MOOSE framework. Due to the nuclear aspects of the code, GRIZZLY is not open-source and access must be requested. The process for obtaining access has been documented in the GRIZZLY user manual [10]. Example input files for the graphite tool are available in GRIZZLY, and these input files can serve as a good starting point for someone interesting in using the model. This section of this report does not provide a thorough discussion of modeling using MOOSE. There is a plethora of modeling capabilities in the MOOSE framework, and it is beyond the scope of this report to provide a user guide to MOOSE. For someone unfamiliar with modeling using the MOOSE framework, a background can be obtained by exploring the moose framework website. Also, some established codes built in MOOSE, such as GRIZZLY, provide well developed user manuals which contain documentation on how to use MOOSE [10].

The input files run in MOOSE are comprised of "blocks". These blocks make up different aspects of a simulation. For example, these aspects include, defining the mesh, specifying the variables, designating the material behavior, choosing parameters associated with the numerical solve as well as a multitude of other options. In MOOSE not all blocks are required in all simulations. For example, in some cases a Postprocessors block can be useful for generating quantities to output, but this block is not required in all simulation. In the following discussion, only blocks which have inputs specific to graphite modeling will be discussed. These blocks include Variables, Functions, AuxVariables, Kernels, AuxKernels, Materials and VectorPostprocessor. These will be discussed in more detail in the remainder of this guide. Blocks which do not have inputs specific to the graphite model will not be discussed. Although, the example problems in GRIZZLY include all necessary blocks to run a simulation.

A.1 Oxidation Model Input Files

A.1.1 Variables Block: Oxidation Model

The oxidation model tracks the concentrations of multiple chemical species during a simulation (O_2, N_2, CO, CO_2 and He). These species concentrations are included in the input file within the Variables block. Each of these chemical species can be included in the input file using the following syntax.

```
[./Species]
  initial_condition = 0.0
[../]
```

In the above text, Species is the variable's name, and the initial value of the variable is set at 0. The units of this initial value are not specified by MOOSE and are therefore dependent on the model parameterization. In our case, since this is a species concentration variable, the model has been parameterized using moles per cubic meter. Variables need to be generated for all of the tracked species, O_2, N_2, CO, CO_2 and He. The temperature, T, is also defined in the Variables block. Therefore, the Variables block can be written as

```
[Variables]
  ### Molar concentration of oxygen gas [mol/m^3]
  [./O2]
    initial_condition = 0.0
  [../]
  ### Molar concentration of nitrogen gas [mol/m^3]
  [./N2]
    initial_condition = 35
  [../]
  ### Molar concentration of carbon monoxide gas [mol/m^3]
  [./CO]
    initial_condition = 0.0
  [../]
  ### Molar concentration of carbon dioxide gas [mol/m^3]
  [./CO2]
    initial_condition = 0.0
  [../]
  ### Molar concentration of helium gas [mol/m^3]
  [./He]
    initial_condition = .1
  [../]
  ### Temperature [K]
  [./T]
    initial_condition = 837.15
  [../]
[]
```

For the purposes of modeling oxidation, in the Variables block, the only changes required are altering the initial conditions. These initial conditions should be set equal to the expected initial chemical species concentrations and gas temperature. The addition or reduction of chemical species in the Variables block is not suggested. For example, if a user wished to add H_2O to the tracked chemical species, it would require editing the source code to add the species as well as reparametrizing the oxidation model.

A.1.2 AuxVariables Block: Oxidation Model

The auxiliary variables in the oxidation model typically do not require any alteration by a user. The AuxVariables block is where quantities which are computed separate from the main system of partial differential equations are introduced. For example, in the oxidation model, the diffusivity of the chemical species is computed as a function of the density of the graphite. These diffusivities are considered auxiliary variables because they are not computed in the partial differential equation system. Beyond the chemical species diffusivities, auxiliary variables are included which track the system bulk density,

effective reaction rate, CO versus CO_2 production, specific heat, heat generation from the reactions and thermal conductivity. The code for the AuxVariables block can be viewed in the example problems found in GRIZZLY. The general syntax for the auxiliary variables is

```
[./variable_name]
  order = <string>
  family = <string>
[../]
```

A.1.3 Kernels Block: Oxidation Model

The Kernels block is used to define and solve the partial differential equations in the system. The kernels which are used in this oxidation model control the evolution of the chemical species concentrations as well as the temperature evolution. Generally, the kernels do not need to be edited by the user. The diffusion kernels which control the evolution of the chemical species are input as

```
[./Species_time]
  type = GraphiteGasMixtureTimeDerivative
  variable = Species
[../]
[./Species_diffusion]
  type = GraphiteGasMixtureDiffusion
  variable = Species
  diffusion_coefficient_name = diffusivity_of_Species
[../]
```

Each chemical species in the simulation requires the two inputs above to evolve the species concentration during a simulation. Note that Species should be changed to the chemical species name. The CO, CO_2, and O_2 concentration can also change from the reaction of oxygen and the graphite. To account for this an additional kernel is required which has the form

```
[./Species_source_sink]
  type = GraphiteReactionSourceSink
  variable = Species
[../]
```

Therefore, an example of the kernels associated with CO would be

```
[./CO_time]
    type = GraphiteGasMixtureTimeDerivative
    variable = CO
[../]
[./CO_diffusion]
  type = GraphiteGasMixtureDiffusion
  variable = CO
  diffusion_coefficient_name = diffusivity_of_CO
[../]
[./CO_source_sink]
  type = GraphiteReactionSourceSinkKernel
```

```
  variable = CO
[../]
```

Using T as the temperature variable, the temperature evolution, in the input file can be written as

```
[./heat_time]
  type = GraphiteHeatTimeDerivative
  variable = T
[../]
[./heat_conduction]
  type = HeatConduction
  variable = T
[../]
[./heat_reaction]
  type = GraphiteReactionHeatSource
  variable = T
[../]
```

The above kernels control the heat conduction as well as heat generation from the reaction between graphite and oxygen. An example of the complete kerels block can be found in the example problems in GRIZZLY.

A.1.4 AuxKernels Block: Oxidation Model

The auxiliary kernels solve for the values of the auxiliary variables. Typically, a user will not need to edit the AuxKernels block. All the auxiliary kernels used in the oxidation model are of type MaterialRealAux which is used to output material properties. The syntax for the AuxKernels block is

```
[AuxKernels]
  [./aux_kernel_name]
    type = MaterialRealAux
    variable = aux_variable
    property = aux_property
  [../]
[]
```

In the above code, the variable is the name of the auxiliary variable which holds the result from the auxiliary kernel and property is the material property name. For example, the text for the bulk density auxiliary variable could be written as

```
[./bulk_density]
  type = MaterialRealAux
  variable = bulk_density
  property = bulk_density
[../]
```

There are many auxiliary variables, each of which require the above input in the AuxKernels block. An example of the complete AuxKernels block can be found in the example problems in GRIZZLY.

A.1.5 Materials Block: Oxidation Model

The Materials block is used to designate material properties and material models. The user will select the graphite type being analyzed here. Currently the two options are IG-110 and NBG-18. The addition of other grades is possible but requires using the process outlined in Section 2.2.3 of the report. The Materials block used in the oxidation model has the form

```
[Materials]
  [./porous]
    type = GraphiteThermalGaseous
    #System Inputs
    system_pressure = 101325.0
    temperature_var = 'T'
    #Specify graphite type
    graphite_type = IG-110 # two types considered: IG-110 or NBG-18
    #Parameters for specified chemical species
    gas_mixture  = 'O2 N2 CO CO2 He'
    molecular_weights = '31.9988 28.0134 28.0101 43.9987 4.0026'
    e_kb = '106.7 71.4 91.7 195.2 10.22'
    sigma = '3.467 3.798 3.69 3.941 2.551'
    dipole_moment = '0.0 0.0 0.122 0.0 0.0'
    polarizability = '1.562 1.71 1.9532 2.5070 0.2080'
  [../]
[]
```

In the text above, there are three parameters which should be edited or checked by a user. The system_pressure is the pressure in pascals, temperature_var is the temperature variable name, in the example above it is T, and graphite_type which can be set to IG-110 or NBG-18.

A.2 Thermo-mechanical Model Input Files

The Thermo-mechanical model is developed for computing stresses in a graphite component. Unlike oxidation modeling, MOOSE has many well developed tools for performing these computations. This includes an action system which will automatically setup much of the input file with limited input from the user. Therefore, the main inputs the user must consider are the material properties and boundary conditions. The following discussion will go over each of the important blocks in the Thermo-mechanical model and show how the input file should be setup. Note there are example input files in GRIZZLY which can be used as a good starting point for modeling the stresses in graphite.

A.2.1 TensorMechanics Action: Thermo-mechanical Model

The TensorMechanics action allows for much of the input file to be set up in a simplified manner. There are many options in the action, but for the purposes of modeling the stresses in graphite, a simple and effective input is

```
[Modules/TensorMechanics/Master]
  [./all]
    strain = SMALL                          # Small strains
```

```
    add_variables = true                  # Automatically add the
    displacement variables
    eigenstrain_names = 'eigen_swell_per eigen_swell_par eigen_thermal_per
    eigen_thermal_par'
    generate_output = 'max_principal_stress mid_principal_stress
    min_principal_stress’
  [../]
[]
```

In the action above we specify the type of strain calculation (SMALL), generate multiple variables, identify the eigenstrains and the outputs of interest. In this example the selected outputs of interest are the principal stresses, because those will be used in the ASME assessments. The variables that are set up by the action are the displacement variables, and the eigenstrains are the dimensional change from temperature and irradiation-induced swelling. As some graphite grades show anisotropy in their dimensional change, we include terms for the perpendicular and parallel to the grain directions.

A.2.2 Variables Block: Thermo-mechanical Model

In a graphite component, stresses can be generated from temperature gradients. Therefore, it is important to include a temperature variable, like in the Variables block shown below.

```
[Variables]
  [./T]
    order = FIRST
    family = LAGRANGE
    initial_condition = 1000.0
  [../]
[]
```

In the example above, the temperature variable is named T, the family and order of the variables are specified, and an initial condition is set. It should be noted that the mechanical model does not attempt to evolve the temperature, so it would also be reasonable to use an auxiliary variable to contain the temperature value. Also, if the oxidation model is being used in conjunction with the Thermo-mechanical model, then the temperature will need to be defined as a variable because the oxidation model does evolve the temperature during a simulation.

A.2.3 AuxVariables Block: Thermo-mechanical Model

The auxiliary variables in the Thermo-mechanical model are the dose profile, time-derivative of the dose profile, irradiation-induced swelling perpendicular to the grain, irradiation-induced swelling parallel to the grain, and the coefficient of thermals expansion perpendicular and parallel to the grain. The Thermo-mechanical model is intended to be able to account for the effects of irradiation, but the dose profile must be determined elsewhere and used as an input. The AuxVariables block can be defined using the following syntax.

```
[AuxVariables]
  [./dose]
    order = FIRST
```

```
    family = LAGRANGE
  [../]
  [./ddose_dt]
    order = FIRST
    family = LAGRANGE
  [../]
  [./swell_per]
    order = FIRST
    family = LAGRANGE
  [../]
  [./swell_par]
    order = FIRST
    family = LAGRANGE
  [../]
  [./cte_per]
    order = FIRST
    family = LAGRANGE
  [../]
  [./cte_par]
    order = FIRST
    family = LAGRANGE
  [../]
[]
```

In the above example, dose and ddose_dt are the variable names of the dose profile and dose profile time-derivative. The swell_per and swell_par auxiliary variables are the irradiation-induced dimensional change perpendicular and parallel to the gain, and the cte_per and cte_par are the coefficients of thermal expansion perpendicular and parallel to the grain. A typical user will not need to alter the AuxVariables block.

A.2.4 Functions Block: Thermo-mechanical Model

A Functions block can be used to input functions used in the model. This can be useful for setting initial conditions or specifying the evolution of an auxiliary variable. The dose profile and derivative of the dose profile with respect to time can be input using the Functions block in the following way

```
[Functions]
  [./fluence]
    type = ParsedFunction
    value = (x+y+z)*t
  [../]
  [./dfluence_dt]
    type = ParsedFunction
    value = (x+y+z)
  [../]
[]
```

In the above example, the dose profile function is named fluence, and the derivative of the dose profile function with respect to time is named dfluence_dt. The value option takes in an equation which can be a

function of the dimensions (x,y,z) as well as time, t. A user will have to determine an appropriate dose profile and dose profile time-derivative from another source and input the functions.

A.2.5 Kernels Block: Thermo-mechanical Model

Most of the kernels in the Thermo-mechanical model are setup through TensorMechanics action. Although, if temperature is defined as a variable, a corresponding kernel is required. There are multiple kernels in MOOSE which can be used to evolve the temperature. These include the kernels identified in the oxidation model. Although for simple isotropic heat conduction we could use the following Kernels block

```
[Kernels]
  [./heat]
    type = HeatConduction
    variable = T
  [../]
  [./heat_dt]
    type = HeatConductionTimeDerivative
    variable = T
  [../]
[]
```

The above heat conduction kernels are applicable to isotropic or near isotropic grades of graphite. Although kernels in MOOSE exist for anisotropic heat conduction. A user should determine if they believe anisotropic heat conduction will influence their simulation results and choose the appropriate kernels.

A.2.6 AuxKernels Block: Thermo-mechanical Model

The dose and dose time-derivative auxiliary variables require corresponding auxiliary kernels. Since the auxiliary variables are defined by parsed functions, the necessary auxiliary kernels are of type FunctionAux. The syntax for the auxiliary kernels associated with the dose and time-derivative of the dose is

```
[./dose_Aux]
  type = FunctionAux
  variable = dose
  function = fluence_func
[../]
[./df_dt_Aux]
  type = FunctionAux
  variable = ddose_dt
  function = dflunece_dt
[../]
```

The other auxiliary variables can be computed by ParsedAux auxiliary kernels. These auxiliary kernels can be input as

```
[./swell_per]
  type = ParsedAux
  args = ' T dose ’
  variable = swell_per
  function = f(T, dose)
[../]
[./swell_par]
  type = ParsedAux
  args = ' T dose'
  variable = swell_par
  function = f(T, dose)
[../]
[./cte_par]
  type = ParsedAux
  args = 'T dose'
  variable = cte_par
  function = f(T, dose)
[../]
[./cte_per]
  type = ParsedAux
  args = 'T dose'
  variable = cte_per
  function = f(T, dose)
[../]
```

The args are a list of variable and auxiliary variable which make up the functional form of the ParsedAux function. In the example above the ParsedAux function will include temperature and dose profile. A user will need to input the functions, f(T, dose), for the irradiation-induced swelling and coefficient of thermal expansion. These relationships for IG-110 are provided in this report and are available in the example problems in GRIZZLY.

A.2.7 Materials Block: Thermo-mechanical Model

In the Thermo-mechanical model, there are three main material behaviors which are being computed. These are the eigenstrains from temperature and irradiation swelling, the elasticity tensor, and the irradiation creep behavior. The eigenstrains are computed using the following code

```
[./var_dependence_swellPar]
  type = DerivativeParsedMaterial
  f_name = var_dep_swellPar
  args = swell_par
  function = 'swell_par*1'
  output_properties = 'var_dep_swellPar'
  enable_jit = true
  derivative_order = 2
[../]
[./var_dependence_swellPer]
  type = DerivativeParsedMaterial
  f_name = var_dep_swellPer
```

```
  args = swell_per
  function = 'swell_per*1'
  output_properties = 'var_dep_swellPer'
  enable_jit = true
  derivative_order = 2
[../]
[./eigenstrainSwellPer]
  type = ComputeVariableEigenstrain
  eigen_base = '0 0 1 0 0 0'
  prefactor = var_dep_swellPer
  args = 'swell_per'
  eigenstrain_name = eigen_swell_per
[../]
[./eigenstrainSwellPar]
  type = ComputeVariableEigenstrain
  eigen_base = '1 1 0 0 0 0'
  prefactor = var_dep_swellPar
  args = 'swell_par'
  eigenstrain_name = eigen_swell_par
[../]
[./var_dependence_thermalPer]
  type = DerivativeParsedMaterial
  f_name = var_dep_thermalPer
  args = ' cte_per T'
  function = 'cte_per*T’
  output_properties = 'var_dep_thermalPer'
  enable_jit = true
  derivative_order = 2
[../]
[./var_dependence_thermalPar]
  type = DerivativeParsedMaterial
  f_name = var_dep_thermalPar
  args = ' cte_per T'
  function = 'cte_per*T’
  output_properties = 'var_dep_thermalPar'
  enable_jit = true
  derivative_order = 2
[../]
[./eigenstrainThermalPer]
  type = ComputeVariableEigenstrain
  eigen_base = '0 0 1 0 0 0'
  prefactor = var_dep_thermalPer
  args =  'cte_per T’
  eigenstrain_name = eigen_therm_per
[../]
[./eigenstrainThermalPar]
  type = ComputeVariableEigenstrain
  eigen_base = '1 1 0 0 0 0'
  prefactor = var_dep_thermalPar
  args =  'cte_par T’
```

```
  eigenstrain_name = eigen_therm_par
[../]
```

A user should confirm that eigen_base aligns with their model's parallel and perpendicular to the grain directions but should not need to make any other changes to the above code. There are multiple ways to define the elasticity tensor within MOOSE, but for isotropic graphite the following code can be used.

```
[./ym]
  type = DerivativeParsedMaterial
  f_name = ym
  function = 'f(dose,T)'
  args = 'dose T'
[../]
[./elasticity_tensor]
  type = ComputeVariableIsotropicElasticityTensor
  args = 'dose T'
  youngs_modulus = ym
  poissons_ratio = .2
[../]
```

In the above code we define a DerivativeParsedMaterial which allows us to input the elastic modulus as a function of the variables and auxiliary variables (dose, T). The elasticity tensor is defined by the ComputeVariableIsotropicElasticityTensor input. A user will need to input the appropriate Youngs modulus function in the DerivativeParsedMaterial and the correct Poisson's ratio in the ComputeVariableIsotropicElasticityTensor. The creep behavior is input using the following text

```
[./radial_return_stress]
  type = ComputeMultipleInelasticStress
  inelastic_models = 'graphite_creep'
  tangent_operator = elastic
[../]
[./graphite_creep]
  type = GraphiteIrradiationCreep
  coefficient = K
  temperature = T
  fluence_dT = df_dt
[../]
```

In the above code, the coefficient is the secondary-creep rate. The user will need to input the secondary-creep rate in place of K in the code above.

A.3 Outputting Data for ASME Assessments

The ASME assessments requires the principal stresses in order to compute an effective stress which is used in the analysis. The Full assessment also requires the volumes of the elements in the mesh. The following steps can be taken to have MOOSE compute these parameters and output the values in CSV files. To compute the element volumes, in the AuxVariables block we can include a volume auxiliary variable using the following code

```
[./volume]
  order = CONSTANT
  family = MONOMIAL
[../]
```

A corresponding entry is required in the AuxKernels block to compute the element volume values.

```
[./volume_aux]
  type = VolumeAux
  variable = volume
[../]
```

To generate the principal stresses, the TensorMechanics action needs to include the line, generate_output = 'max_principal_stress mid_principal_stress min_principal_stress'. An example of this is shown above is Section 5.2.1**.** The principal stresses and volumes can be output in CSV files by using the VectorPostprocessors and Outputs blocks as shown below.

```
[VectorPostprocessors]
  [max_principal]
    type = ElementValueSampler
    variable = max_principal_stress
    sort_by = id
    execute_on = TIMESTEP_END
  []
  [mid_principal]
    type = ElementValueSampler
    variable = mid_principal_stress
    sort_by = id
    execute_on = TIMESTEP_END
  []
  [min_principal]
    type = ElementValueSampler
    variable = min_principal_stress
    sort_by = id
    execute_on = TIMESTEP_END
  []
  [volume]
    type = ElementValueSampler
    variable = volume
    sort_by = id
    execute_on = TIMESTEP_END
  []
[]
[Outputs]
  exodus = true
  csv = true
[]
```

This will generate CSV files at the end of each time step. The above user guide discussion is intended to introduce the blocks used in the graphite modeling tool. It is the authors opinion that the best way to understand MOOSE and the graphite modeling capabilities is to go through the example problems provided in GRIZZLY. The Python code which implements the ASME assessments, and an associated user guide is available upon request.