SANDIA REPORT

SAND2016-9071
Unlimited Release
Printed September, 2016

Development of a Multi-physics Capability for Predicting Residual Stress in a GTS Reservoir (U)

Kevin L. Manktelow, Lauren L. Beghini

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

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

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

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

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

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

SAND2016-9071
Unlimited Release
Printed September, 2016

Development of a Multi-physics Capability for Predicting Residual Stress in a GTS Reservoir (U)

Kevin L. Manktelow, Org. 08259
Lauren L. Beghini, Org. 08259

Multiphyics Modeling and Simulation
Sandia National Laboratories
P.O. Box 0949
Livermore, CA 94550-MS9042

klmankt@sandia.gov

Abstract (U)

This report documents completion of a Level 2 Milestone on the development of a multi-physics capability to predict the evolving material state through the manufacturing process of a Gas Transfer Systems (GTS) reservoir. We present details on new developments and capability improvements that address the following completion criteria: (i) validation of a microstructure evolution model, including recrystallization and strain aging, (ii) demonstration of the capability to remesh, map and transfer material state (internal state variables) and residual stress from forging to machining to welding processes, and (iii) formal V&V characterization and quantification of uncertainties of material parameters and manufacturing process parameters on residual stress.

Acknowledgement

The major advancements in modeling and simulation capabilities described in this report were only possible because of previous work by others and continued support from ASC P&EM, V&V, and IC programs. ASC P&EM and ASC V&V project management and funding support were provided by Jim Redmond, Jeff Payne, and Arthur Brown.

Previous developments in process modeling for forging, machining, and welding come from many individual and team contributions of the past several years. Arthur Brown, Tim Kostka, Bonnie Antoun, Mike Maguire, Lisa Deibler, Mike Veilleux, Jay Foulk, and Tom Reynolds are a few of the many individuals that have contributed towards the success of this milestone.

Many of the material model developments including dynamic strain aging and recrystallization are a result of research done by Arthur Brown, Bonnie Antoun and others. Other facets of the microstructure evolution model have a rich history of development, with credit given to D. Bamman, M. Chiesa, G. Johnson, D. Regueiro, M. Marin, and several others.

The remeshing and state variable mapping aspects of this milestone could not have been successful without the programs, processes, and mathematical formalisms tested and developed by Jay Foulk, Mike Veilleux, John Emery, Alejandro Mota, Sofie Leon, and Jake Ostien. Their tools served a foundational role in milestone developments that enabled state variable mapping in complex multi-physics contact problems. Support from Mike Veilleux and Jay Foulk on the use and implementation of these tools was particularly valuable in this regard. Stacy Nelson also supported the verification and validation efforts via statistical analysis, post-processing, and documentation.

We would also like to acknowledge the support and direction provided by the L2 Milestone review committee including Scott Peterson (chair), Kendall Pierson (code expert), Bill Scherzinger (constitutive model expert), Patty Hough and Sophia Lefantzi (V&V experts), Dorian Balch (GTS customer), and Arthur Brown (multi-physics modeling and simulation (Org. 8259) manager).

Although this project was defined as a joint P&EM and V&V effort there were significant contributions from the integrated code (IC) program. The SIERRA development teams – particularly Solid Mechanics and Thermal/Fluid – provided major support through bug fixes, feature implementation, office hours, and general troubleshooting. Their work on improving solution robustness and implicit contact, in particular, resulted in major improvements during the course of this work that enabled the solution of challenging multi-physics problems that were previously difficult or impossible to solve. This partnership was critical to successful completion of the milestone objectives, and should serve as a model for future milestones.

Finally, the programs supporting this milestone are implicitly part of a larger Predictive Capability Framework Pegpost research effort led by Jay Foulk to simulate the GTS cradle-to-grave life cycle. The Pegpost effort and the individuals involved have been valuable in coordinating research efforts that support immediate milestone objectives, discussing and addressing technical challenges, as well as in scoping work to support long-term visions for modeling and simulation capability development.

Contents

List of Figures

List of Tables

This page intentionally left blank.

Chapter 1

Introduction

Modeling and simulation analysis is an increasingly necessary tool for designing new products, refining existing designs, and understanding complex or difficult-to-measure phenomena. Analysis demands presented to the modeling and simulation community are increasingly challenging and require continued capability development. Mechanical failure prediction modeling is one area of interest to designers and product managers that has seen particular growth over the last several years. Residual stress evolution through a series of steps is a particular concern in failure prediction analysis. This document describes a collection of related efforts to support a Level 2 (L2) Milestone for predicting residual stress evolution through a series of complex manufacturing steps that are inherently multi-physics (thermo-mechanical, or thermo-mechanical-electrical) in nature. Residual stress evolution in high pressure vessels is the particular motivating example discussed later, but the simulation procedures and capabilities developed as part of this work are broadly applicable to a variety of products and systems.

Residual stresses introduced through manufacturing and testing processes can be large enough to drive crack propagation and eventual failure in metallic components. The combined effect of these processes on residual stresses is not well-understood. For example, residual stresses introduced during a forging process may be relieved during a subsequent machining or heat treatment process. New residual stresses may be introduced during a welding process. Furthermore, processes involving high-temperatures or large deformations such as forging and welding fundamentally change material properties in ways that are difficult to experimentally assess.

Figure 1.1 schematically depicts a series of manufacturing steps for a typical high pressure vessel that may be used, for example, in a gas transfer system (GTS). A bulk material such as steel ingot or bar stock is forged to obtain desirable material characteristics. The forging is later machined and welded to produce a pressure vessel for storing high-pressure gas. Modeling and simulating the sequence of forging, machining, welding, and residual stress evolution is the primary focus of this report. These developments were jointly supported by the ASC project through a P&EM project aimed at improving high-temperature plasticity modeling, and a V&V project aimed at understanding uncertainty propagation through combined processes. This work also supports a longer-term PCF Pegpost effort to develop a complete “cradle-to-grave” simulation capability for assessing GTS reservoirs; additional process steps include simulating the effects of gas storage on material properties and an uncertainty assessment for potential failure mechanisms such as yield and crack growth.

Each of the manufacturing or life cycle steps depicted in Fig. 1.1 induces or relieves residual stresses. Furthermore, the large deformations and temperature ranges encountered may fundamentally change microscopic material structure in ways that affect the macroscopic behavior. Forging, for example, can introduce yield strength and hardness changes at a macroscopic scale as a result of grain growth and recrystallization. Heat treatment and non-uniform cooling can further evolve the material properties in unintuitive ways, and machining can relieve or redistribute residual stresses. Various welding methods introduce high temperatures and sharp gradients which further alter material properties and introduce new residual stresses, all of which may impact final design quality. It's worth noting that modeling and simulation has already been very successfully used to simulate individual elements of the GTS life cycle [16, 3, 4, 13]. The work described in this report builds on these previous efforts by developing, verifying, and validating tools that extend simulation capabilities.

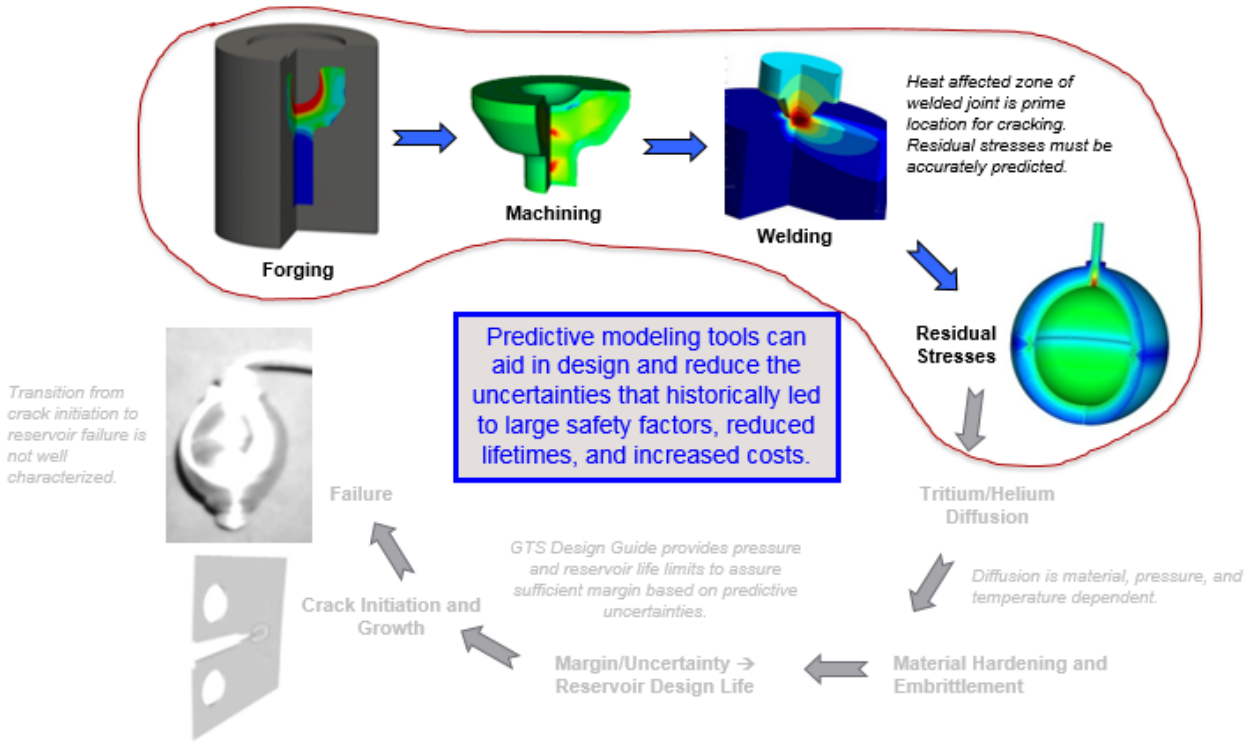

Figure 1.1. Cradle-to-grave life cycle of a typical GTS high-pressure vessel.

An assessment of the processes forming the GTS life cycle helped identify gaps in existing simulation capabilities. The three largest gaps identified were selected for future development: high-temperature plasticity, material state mapping, and residual stress uncertainty quantification. High-temperature plasticity needs involve gaining a better understanding of plastic flow across a broad range of temperatures and developing robust constitutive models to capture necessary behaviors. Even with accurate models for high-temperature plasticity, there are fundamental limitations in the amount of [mesh] deformation that can be sustained during a typical finite-element analysis.

Material state mapping processes are required to transfer residual stresses, deformations, and material internal state variables from a deformed mesh to an undeformed mesh so that deformation processes can resume. This state mapping must be robust, accurate, and minimally-invasive to gain acceptance from the modeling community and to provide impactful results to designers. The introduction of new tools and simulation processes requires a formal assessment of model form assumptions and errors. These gaps were succinctly captured as L2 Milestone completion criteria:

1. Validate a microstructure evolution model, including recrystallization and strain aging. The model shall also capture the history effects of strain rate and temperature changes
2. Demonstrate the capability to remap and transfer material state (internal state variables) and residual stress from forging to machining to welding processes
3. Characterize and quantify uncertainties of material parameters and manufacturing process parameters on residual stress

New developments and evidence of improved capabilities specifically supporting each of these criteria are provided in Chapters 2, 3, and 4. Chapter 2 discusses validation and enhancements made to the Bamman-Chiesa-Johnson microstructure evolution model (BCJ_MEM) made to improve high-temperature plasticity modeling. This effort involved a full renovation of the existing model, including conversion from FORTRAN to C++, solver robustness improvements, bug fixes, and a new dynamic strain aging capability. Chapter 3 describes a new state variable mapping and remeshing approach needed to simulate large deformation and machining processes. A representative resistance forge weld geometry was selected to demonstrate the process, guide development efforts, and demonstrate residual stress evolution through forging, machining, and [resistance] welding. Finally, Chapter 4 briefly describes efforts to characterize and quantify uncertainties of material parameters and manufacturing processes on residual stresses. Several representative and proof-of-capability geometries were used to leverage historical validation experiments and demonstrate mesh convergence. A formal V&V assessment supporting the last completion criteria is documented in a companion report [1].

Developments supported by this L2 Milestone are major step forward in enabling predictive modeling and simulation for manufacturing processes. Examples include the first simulation of a complex, coupled multi-physics resistance forge weld on representative geometry and formal V&V assessment of residual stresses through a sequence of manufacturing processes that includes forging, machining, welding, and subsequent cooling. As with all research efforts, there were growing pains and challenges. Several opportunities remain for future research and development efforts including implicit contact algorithms, solution robustness, and multi-physics code coupling capabilities. These and other recommendations for future work are discussed in Chapter 5, along with some "lessons learned" that may be beneficial for future analysis efforts.

This page intentionally left blank.

Chapter 2

Validation of Microstructure Evolution Model

In this chapter, we address the first completion criteria "Validate a microstructure evolution model, including recrystallization and strain aging. The model shall also capture the history effects of strain rate and temperature changes." Here, we provide evidence to support the first milestone success criteria:

- Incorporate dynamic strain aging into the existing BCJ_MEM material model (development code)
- Validation - formal assessment per V&V guidelines of how the BCJ-MEM model compares to experimental data available to date for 304L material. This can be built upon previous validation efforts, with references included in the L2 Milestone documentation.

An "exceeds" criteria was to incorporate the BCJ_MEM model with dynamic strain aging capability in SIERRA. These efforts are described next.

2.1 Material Modeling Overview

The DSA material model was originally developed as a basis for implementing a dynamic strain aging (DSA) capability into the existing BCJ_MEM (Bamman-Chiesa-Johnson microstructure evolution model). The BCJ_MEM material constitutive model provides microstructure evolution modeling capabilities required to model hardening, recrystallization, grain growth, and recovery mechanisms. The BCJ_MEM model has long been part of the verified, validated, and tested production code. However, the original BCJ_MEM model was also written in FORTRAN and had some fundamental implementation details that hindered further development. In some problems, BCJ_MEM could not solve the complex evolution equations and instead used a simple radial return approximation, leading to incorrect solutions. For these reasons, the BCJ_MEM material model was converted to C++ (the language used for a majority of SIERRA) and the resulting development model was named *DSA* since the original intent was to add a dynamic strain aging (DSA) capability. As the BCJ_MEM model was converted from FORTRAN to C++, several of the other implementation details were also improved to help provide robust solutions in situations where the

internal BCJ_MEM solver may have previously stalled. A few of these developments are discussed in more detail in Section 2.2; other modifications are left as code comments for future developers.

The highly successful conversion effort provided a clean development foundation for implementing, improving, and testing new capabilities such as dynamic strain aging. A function (method) specifically devoted to implementing dynamic strain aging evolution equations was added and used to implement a basic explicit integration of DSA evolution equations; some of this work is described in Section 2.4. However, a robust and correctly-implemented solution is necessary before adding additional physics knobs so emphasis was placed on general model improvements and code architecture before DSA development began. Hence, although a dynamic strain aging capability was added to the BCJ_MEM constitutive model (via DSA) it is still considered a development capability that requires future development.

The DSA material model is also more general than the BCJ_MEM material model, allowing for more complex rate and temperature-dependencies, element death criteria, and even variable number of recrystallization cycles. The more complex rate and temperature-dependencies implemented are identical to those implemented for the Elastoviscoplastic material model. Hence, the DSA material model consolidates the legacy FORTRAN code of multiple material models while simultaneously providing a more robust solution. The result of this consolidation effort is a broader impact since more users benefit from the improved capabilities. Furthermore, the DSA material model is also serving as a collaborative platform for future hydrogen diffusion modeling. Development is predicted to continue into FY17 as new features are added, old ones are removed, and maintenance continues.

This chapter discusses some of these efforts in greater detail. A complete discussion of the plasticity model, flow rules, evolution equations, and numerical algorithm implementation is desirable but cannot be adequately addressed within the scope of this manuscript. These details are worthy of an entire report on their own and as such no attempt is made to include all details; it is assumed the reader is already familiar with the constitutive model. Readers interested in specific material model details such as the evolution equations, plastic flow rules, and material properties should consult excellent references such as [5] and [11]. The SIERRA/SM user manual [14] also includes a good discussion regarding options available to users.

Finally, analysts interested in testing the DSA material model will find switching from BCJ_MEM or Elastoviscoplastic only requires changing the material model name and nothing else. The improved solver behavior must be manually enabled by setting the additional option `Newton_Solver_Method=1` at the present time, although this may become the default in future versions.

2.2 Solver Modifications & Improvements

The DSA material model includes several solver and algorithm updates. One of the most persistent issues with the BCJ_MEM and Elastoviscoplastic models arises when the semi-implicit plastic strain solver cannot converge on a solution for the plastic increment. The issue is manifested to

users through a warning printed to the analysis log file:

```
WARNING: semi-implicit integration for plastic strain failed to
converge ... so radial return solution will be used, for this
element in this model problem.
```

This warning message means that although a user requested an implicit (or semi-implicit) solution for the plastic strain increment, the internal material model solver could not obtain a solution without further iteration. The warning message further informs the user that a radial return solution will be used instead. The degree of error incurred as a result of using a radial return approximation will vary depending on the time step and resulting excursion from the yield surface, but usually the end result is poor or impossible convergence behavior in implicit solution schemes. In other cases, even though convergence was eventually obtained the solution required over 20000 iterations to solve for the plastic strain increment to even modest tolerances.

After close analysis of the solver, the root-cause of the implicit solution failure was traced to a singularity in the residual function gradient provided to the Newton-Raphson solver. The residual function used for Newton-Raphson iterations was modified to eliminate the singularity, thus improving the overall iterative behavior of the Newton-Raphson solution. A [very] brief description of the changes follows, intended only to introduce the general approach without including lengthy algorithmic details[1]. Note that BCJ_MEM (and therefore, DSA) uses a decoupled solution procedure for damage and the plastic increment $\Delta\varepsilon^p$. Therefore, without loss of generality, only the solution for the plastic increment is discussed. Note that the development that follows is presented in the context of an incremental solution.

The plastic increment solve is semi-implicit, using a Newton-Raphson implicit solution strategy for the consistency parameter $\Delta\gamma$ (proportional to the plastic increment) while updating the isotropic hardening variable κ each Newton iteration using a midpoint integration scheme. The Newton-Raphson algorithm is used to enforce the consistency condition – that is, the constraint that the converged solution lies on the rate-dependent yield surface:

$$f(\boldsymbol{\sigma}, \dot{\varepsilon}^p) = 0. \tag{2.1}$$

Using the isotropic BCJ_MEM yield surface

$$\sigma_y = (Y(\theta) + \kappa)\, F(\theta; \dot{\varepsilon}) \tag{2.2}$$

and replacing $\boldsymbol{\sigma}$ with an expression for the deviatoric effective stress $\boldsymbol{\xi}$, the consistency condition can be written as

$$f(\boldsymbol{\xi}, \dot{\varepsilon}^p) = \sqrt{\frac{3}{2}}\, \|\boldsymbol{\xi}\| - \sigma_y(\dot{\varepsilon}^p; \theta) = 0. \tag{2.3}$$

The iterative incremental form implemented in the material model uses relations that permit Eqs. (2.2) and (2.3) to be re-written only as a function of the consistency parameter $\Delta\gamma$, so that (2.3) can

[1] A more comprehensive document formally outlining the material model and implementation details is planned for future work.

be expressed instead as

$$f(\Delta\gamma) = \sqrt{3/2}\,\|\boldsymbol{\xi}\| - (Y + \kappa(\Delta\gamma))\,F(\Delta\gamma) = 0. \tag{2.4}$$

The rate-dependence of the yield surface is mostly contained within the function $F(\Delta\gamma)$, where the relationship between the consistency parameter and plastic strain rate $\dot{\varepsilon}^p = \sqrt{2/3}\Delta\gamma/\Delta t$ has been substituted

$$F(\Delta\gamma) = \left\{ 1 + \sinh^{-1}\left[\left(\frac{\sqrt{2/3}\Delta\gamma/\Delta t}{f(\theta)} \right)^{1/v(\theta)} \right] \right\}. \tag{2.5}$$

A rate-parameter χ is introduced for notational convenience so that $F(\Delta\gamma)$ is expressed as

$$F(\Delta\gamma) = 1 + \sinh^{-1}(\chi), \quad \text{where} \quad \chi = \left(\frac{\sqrt{2/3}\Delta\gamma/\Delta t}{f(\theta)} \right)^{1/v(\theta)}. \tag{2.6}$$

The consistency condition is enforced using a Newton-Raphson algorithm which seeks roots for the residual function

$$R(\Delta\gamma) = \frac{\sqrt{3/2}\,\|\boldsymbol{\xi}\|}{Y + \kappa} - F(\Delta\gamma), \tag{2.7}$$

The residual function $R(\Delta\gamma)$ is identical to the consistency condition, but is normalized by the expression $Y + \kappa$ so that semi-implicit solutions converge using a relative tolerance. Therefore, a root $\Delta\gamma$ solving $R(\Delta\gamma) = 0$ also enforces the consistency condition. An analytical residual derivative is supplied to the Newton-Raphson solver to determine update direction

$$R' = \sqrt{\frac{3}{2}} \left(\frac{(Y + \kappa)\,\|\boldsymbol{\xi}\|' - \|\boldsymbol{\xi}\|\,(Y + \kappa)'}{(Y + \kappa)^2} \right) - F', \tag{2.8}$$

where $(\,)'$ denotes derivative with respect to the consistency parameter $\Delta\gamma$. The yield parameter Y is independent of the consistency parameter, so Eq. (2.8) reduces to

$$R' = \sqrt{\frac{3}{2}} \left(\frac{(Y + \kappa)\,\|\boldsymbol{\xi}\|' - \|\boldsymbol{\xi}\|\,\kappa'}{(Y + \kappa)^2} \right) - F'. \tag{2.9}$$

The second term $F' = \mathrm{d}F/\mathrm{d}\Delta\gamma$ is responsible for derailing the Newton-Raphson algorithm when the desired root is near $\Delta\gamma = 0$. The derivative of $F(\Delta\gamma)$ is given by

$$F' = \frac{1}{\sqrt{1 + \chi^2}}\chi', \quad \text{where} \quad \chi' = \frac{1}{v(\theta)} \left(\frac{\sqrt{2/3}}{\Delta t f(\theta)} \right) \Delta\gamma^{(1/v(\theta) - 1)}. \tag{2.10}$$

Without further discussion of the temperature-dependent functional forms for $v(\theta)$ and $f(\theta)$, it is clear from the last expression that for $v(\theta) > 1$ the exponent of $\Delta\gamma$ becomes negative. Small plastic strain increments associated with $\Delta\gamma \approx 0$ result in a singularity whereby $\chi' \to \infty$ as $\Delta\gamma \to 0$. Figure 2.1 illustrates the residual function used in BCJ_MEM and the singularity at $\Delta\gamma = 0$. The problem in the BCJ_MEM semi-implicit solver resulted from situations where the consistency parameter $\Delta\gamma$ was evaluated too close to 0 and produced a NaN value for the derivative χ'. A

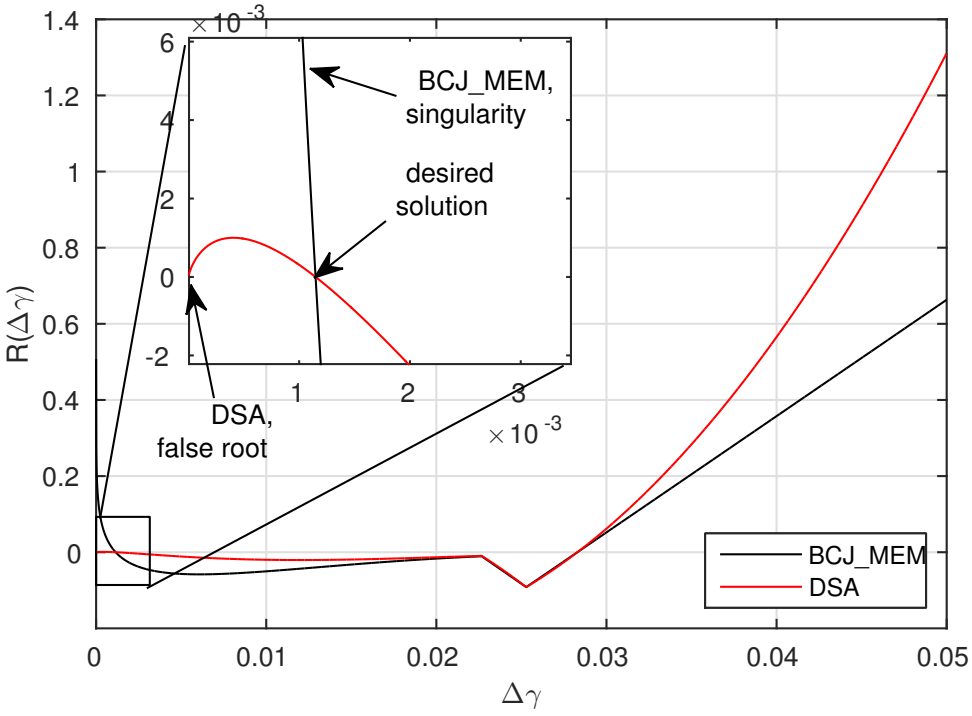

Figure 2.1. BCJ_MEM and DSA residual functions from application example exhibiting difficult material model convergence

check for NaN's within the solver attempted to catch these and compensate accordingly, but the solution robustness as a result of this singularity still suffered in many cases such as when the true root was very near zero. This unfortunately, is a particularly common scenario in situations where small time step may be necessary (e.g., contact, buckling regimes, or implicit dynamics), or simply during the transition from elastic to plastic loading.

The DSA implementation differs from BCJ_MEM in two key ways. First, the DSA material model incorporates the slightly more general yield surface used in the elasto-viscoplastic model. In this case, the rate-dependencies of the initial yield $Y(\theta)$ and hardening behavior κ may be described independently, according to the expression

$$\sigma_y = Y(\theta)F(\theta;\dot{\varepsilon}) + \kappa G(\theta;\dot{\varepsilon}), \tag{2.11}$$

where both $F(\theta;\dot{\varepsilon})$ and $G(\theta;\dot{\varepsilon})$ have similar functional forms, but whose parameters are independent. Second, a benign $\Delta\gamma$ factor is used to introduce a 'false root' at $\Delta\gamma = 0$ to eliminate the singularity. The modified DSA residual function takes the form

$$R(\Delta\gamma) = \Delta\gamma\left(\sqrt{3/2}\,\|\boldsymbol{\xi}\| - YF(\Delta\gamma) + \kappa(\Delta\gamma)G(\Delta\gamma)\right), \tag{2.12}$$

where again the the algorithmic dependence on the consistency parameter increment $\Delta\gamma$ is explicitly annotated. The residual derivative in this case is

$$R' = \sqrt{3/2}\,\|\boldsymbol{\xi}\| + \Delta\gamma\sqrt{3/2}\,\|\boldsymbol{\xi}\|' - (\Delta\gamma Y F)' - (\Delta\gamma\kappa G)' \tag{2.13}$$

$$= \sqrt{3/2}\,\|\boldsymbol{\xi}\| + \Delta\gamma\sqrt{3/2}\,\|\boldsymbol{\xi}\|' - \Delta\gamma Y F' - Y F - \Delta\gamma(\kappa G' + G\kappa') - \kappa G \tag{2.14}$$

$$= \sqrt{3/2}\,\|\boldsymbol{\xi}\| + \Delta\gamma\sqrt{3/2}\,\|\boldsymbol{\xi}\|' - Y(F + F'\Delta\gamma) - \kappa(G + G'\Delta\gamma) - G\kappa'\Delta\gamma \tag{2.15}$$

As with BCJ_MEM, the derivatives F' and G' appearing in the residual derivative R' have singularities at $\Delta\gamma = 0$. However, the additional $\Delta\gamma$ multiplier in the terms $F'\Delta\gamma$ and $G'\Delta\gamma$ eliminates the singularity for all $v(\theta) > 0.0$[2]. The remaining terms are well-conditioned and do not present any singularities at $\Delta\gamma = 0$.

Figure 2.1 schematically depicts the effect of the modified residual form. The residual functions for BCJ_MEM and DSA have been scaled so that their relative magnitudes are similar. Note that both residual functions have a root near $\Delta\gamma = 0.001$ in this example, but while the DSA residual function is well-behaved near $\Delta\gamma = 0$, the BCJ_MEM residual function approaches infinity. The residual function should ideally only have two roots as the consistency parameter is increased. The first root is the desired solution and is generally near zero. The second root corresponds to the opposite side of the yield surface. In the DSA algorithm, there is a third root at zero. The trade-off here is that a false root is introduced in exchange for numerical stability. This poses no problem since the root is always at $\Delta\gamma = 0$ and a custom Newton-Raphson algorithm can account for it.

These modifications were implemented in a new semi-implicit solver algorithm that is selectable in the DSA material model by specifying `Newton_Solver_Method = 1`. The new semi-implicit Newton-Raphson algorithm uses a standard algorithm to solve for the independent variable $\Delta\gamma$. For each Newton-Raphson iteration $k < k_{max}$, $\Delta\gamma = \Delta\gamma^{(k)}$ is updated according to

$$\Delta\gamma^{(k)} = \Delta\gamma^{(k-1)} - \frac{R(\Delta\gamma^{(k-1)})}{R'(\Delta\gamma^{(k-1)})}. \tag{2.16}$$

The hardening state variable $\kappa(\Delta\gamma)$ is also updated within each Newton-Raphson iteration. Note that $R(\Delta\gamma)$ can be viewed as a measure of distance from the current estimate to the yield surface so convergence is checked relative to a constant C which is proportional to the radius of the yield surface in previous steps. As with any Newton-Raphson algorithm, there are several other possible issues that may be encountered and must be handled appropriately for a robust capability. A few of these issues are:

- Large residual derivatives R' paired with small residuals can cause the solver to stall since incremental changes may approach numerical precision.
- Minima and maximum of the residual function result in $R' = 0$, causing NaN's due to divide-by-zero errors.
- Algorithm *overshoot* may result in evaluating $R(\Delta\gamma)$ with negative values of $\Delta\gamma$ (that violate the Kuhn-Tucker conditions)

[2]There is actually a $v_Y(\theta)$ and $v_K(\theta)$ for functions F and G, respectively

- Algorithmic loops due to discontinuous derivatives may stall the iterative solution.

The new DSA Newton-Raphson solver (again, currently selectable with `Newton_Solver_Method = 1`) checks for situations which may potentially result in any of these catastrophic scenarios and takes a reasonable course of action to avoid numerical issues. Note that original BCJ_MEM solver used a similar Newton-Raphson update scheme with a line-search to improve convergence; however, the singularity near $\Delta\gamma = 0$ was still problematic in many of these scenarios.

In some cases, even the updated Newton-Raphson solver is still not robust enough to find the desired $\Delta\gamma$ solution; a fall-back "brute force" adaptive bisection solver was added to specifically handle these scenarios. The solver passes control to the brute force solver if the maximum Newton-Raphson iterations are reached without convergence. The brute force solver adaptively searches for a root $\Delta\gamma > 0$ using a bisection algorithm. The search interval starts near $\Delta\gamma = 0$ and adaptively increases in size and location if a root is not initially found. In every scenario tested thusfar, the Newton-Raphson or fallback bisection solver have been successful in returning a converged root for $\Delta\gamma$. Although more difficult cases may exist, the robustness has been thoroughly tested by several analysts in many challenging application problems that include element death failure criteria, resistance welding, tension test simulation, and more.

2.3 Hardening and Misorientation Stability

The DSA material model also introduces a small but important modification intended to improve the numerical integration of the evolution equations for the hardening and misorientation state variables κ and ζ. The complete details regarding the derivation of the evolution equations for κ and ζ are beyond the scope of this text; for that the reader is referred to [5]. The objective of this section is simply to make potential users of BCJ_MEM or DSA aware of some numerical artifacts that result from the integration procedure implemented for the evolution equations.

Hardening and misorientation differential equations describe the evolution of these variables over time. Furthermore, their evolution is coupled through terms governing recrystallization state variables. As discussed previously, the plasticity model is implicitly integrated to obtain the plastic strain increment, but other variables such as the recrystallization evolution equations and terms in the hardening evolution equation for κ evolve explicitly.

The coupling between the equations can potentially result in unstable evolution behavior[3] since competing hardening and softening effects are not integrated fully implicitly. Figure 2.2 depicts the evolution of κ averaged over eight Gauss points for an element which is experiencing unstable behavior. During each time step, the algorithm in BCJ_MEM prevents negative values of κ and ζ from occurring by detecting $\kappa < 0$ and $\zeta < 0$ and prescribing a small value ($\kappa =$ 1e-6) instead, before continuing the algorithm. The result of this was wild oscillations resulting in values of 1e-6 during one time step, 1e+12 during the next time step, and back to 1e-6 in the following step. Not

[3] Unstable evolution behavior in this case just means large non-physical oscillations from one time increment to the next.

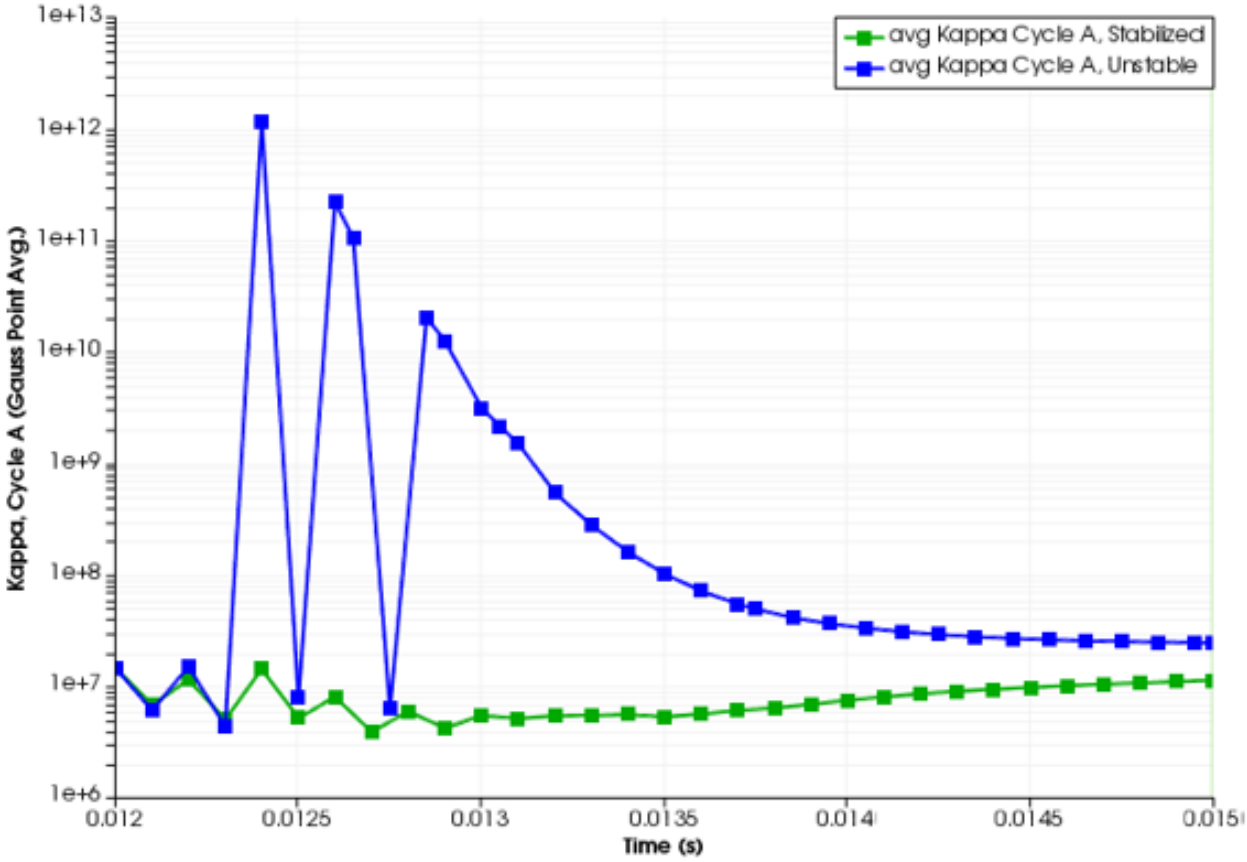

Figure 2.2. Hardening variable (κ) oscillations during resistance forge weld. Values shown are Gauss point average values; some individual Gauss points have more severe oscillations (from 1e-6 to more than 1e12).

all Gauss points in a single element experience these oscillations at the same time, so the Gauss-point averaged results depicted in Fig. 2.2 do not return to 1e-6 at the bottom of the oscillations. An alternative (but still imperfect) adjustment procedure was implemented in the DSA algorithm to somewhat reduce the oscillations by limiting the amount κ or ζ can change during a single time step. Rather than setting $\kappa =$ 1e-6 when negative values are detected, κ and ζ are limited to a 90% reduction in value over a single time step. This ensures negative values are never computed, and reduces the magnitude of the unstable oscillations that are likely due to the explicit integration and coupling effects of ζ, κ, and recrystallization state variables.

The procedure in BCJ_MEM somewhat stabilizes the results eventually as seen in Fig. 2.2, but for several time steps some elements may behave very stiff (effectively elastic) due to an extremely large yield surface. Figure 2.3 illustrates the effect that very large κ can have on a real problem. The left sub-figures (top and bottom) depict the deformation that results from using the DSA material model, whereas the right sub-figures (top and bottom) depict the deformation that results

from using the BCJ_MEM material model. Both figures depict the same instant in time and are colored by the Gauss-point averaged hardening variable κ. Three dark red elements are visible in the lower-right figure with very large values for κ, effectively making these elements perfectly elastic. This effectively stiffens the elements and prevents them from plastically deforming like the surrounding elements.

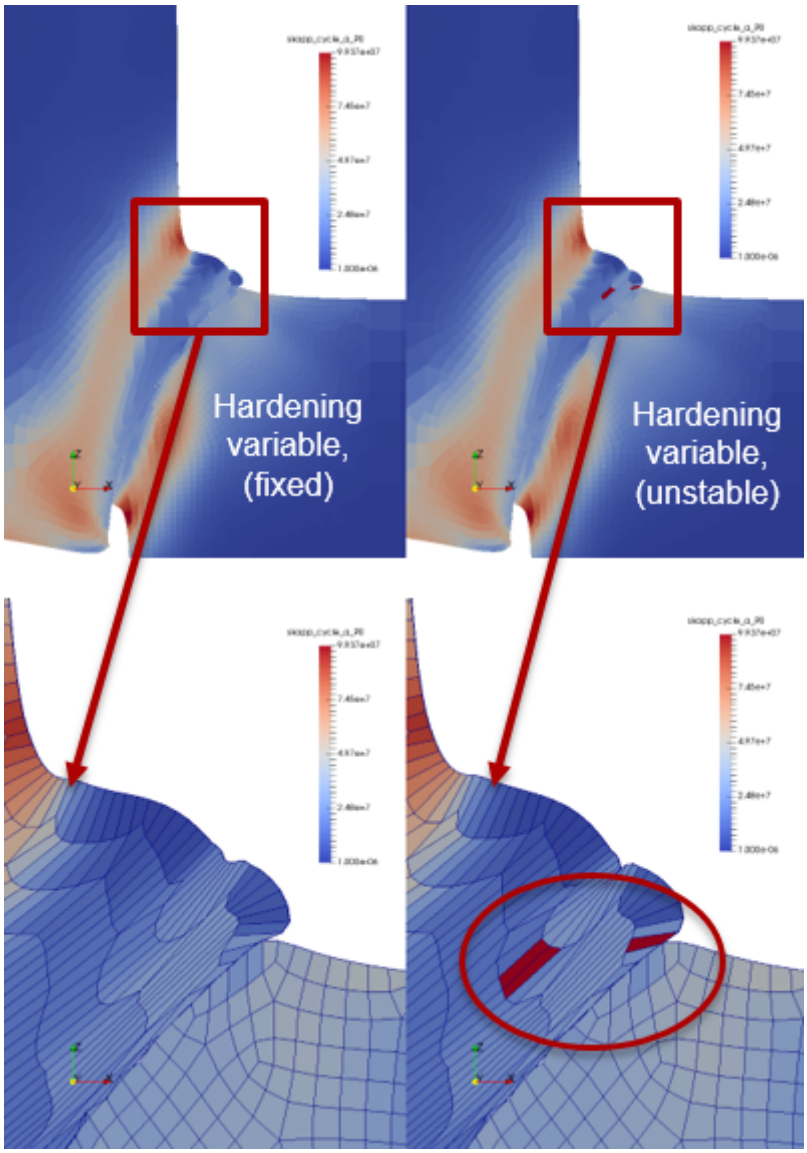

Figure 2.3. Hardening variable (κ) instability during a resistance forge weld simulation

The imperfect, but improved, hardening behavior in DSA has further implications on contact solutions and remeshing/mapping processes. The unstable and oscillatory behavior can have detrimental effects on contact iterations, since predicted configurations and tangent preconditioners rely

on accurate and smooth evaluations of the material model. After implementing this fix, the number of failed contact iterations was reduced and contact convergence robustness was significantly improved as a result. Remeshing and mapping procedures are impacted non-physical large values for κ distributed randomly throughout the mesh effectively creates discontinuities in an otherwise smooth field. Projection and interpolation procedures used to map state variables from a deformed mesh to a new reference configuration have no knowledge of the incorrect field and will thus map the incorrect field over to the next mesh. Although a new reference configuration may continue to work for a short period, the results are corrupt and will pollute any future solutions.

2.4 Dynamic Strain Aging Implementation and Status

The dynamic strain aging (DSA) model was originally constructed as a development model for implementing a dynamic strain aging capability into the existing BCJ_MEM constitutive model. This section briefly describes the state of that effort and areas for future work. The dynamic strain aging phenomena is a microstructure evolution phenomena associated with the Portevin-Le Chatelier effect [10]. The effects of dynamic strain aging are observed only in specific temperature ranges where dislocation motion and solute pinning lead to inverted rate sensitivities, serrated stress-strain behavior during tension tests, and stress overshoot during re-loading [10].

An option was added to the DSA model as an initial effort to include dynamic strain aging effects in a material model accessible to the SIERRA finite-element production code. This work is largely based on research and development efforts by Arthur Brown during FY15 as part of the high-temperature plasticity P&EM project. It is emphasized that while the current implementation of the dynamic strain aging physics does provide a mechanism for some DSA processes (e.g,. overshoot), it is still considered a development effort that is not yet ready for general production use.

The current implementation of dynamic strain aging utilizes three state variables to evolve dislocation densities. Mobile and immobile dislocations are represented by ρ_m and ρ_i, respectively. A third variable describing the fraction of trapped mobile dislocation densities is described by ω. These internal state variables influence the yield stress, and evolve during loading and unloading. The overshoot mechanism, for example, results from dislocations trapped by solutes which are eventually freed with increases in plastic strain. The specific forms for dislocation density evolution and coupling to the yield stress are still in flux, and the current implementation is not ideal so no evolution equations are provided herein. Future work is still needed to design the evolution equations so that all necessary phenomena can be captured.

Figure 2.4 depicts the evolution of these internal state variables during a strain-controlled simulation for a single element. The element is deformed with a constant strain rate, unloaded slightly and held fixed for a period of time, and then strained again. This process repeats with various holding times. As the fraction of trapped dislocations increases, more pronounced overshoot is observed upon re-loading.

A primary reason for implementing DSA into a SIERRA was to make the material model testable within the Sierra/SM architecture. Previous work (by Arthur Brown) implemented strain aging in a custom program used for developing evolution equations and explicitly integrating them for a single element, so the implementation into a production finite-element code is a significant step forward in implementing DSA phenomena. Figure 2.5 depicts a displacement-controlled tension test. Wedge-like boundary conditions were applied to approximate axisymmetry. The image on the left depicts the evolution of the equivalent plastic strain variable, whereas the images on the center and right depict the the immobile and mobile dislocation densities at a snapshot in time. These smooth tension specimen tests should also exhibit overshoot (and other) phenomena as depicted in 2.5 as the DSA implementation matures.

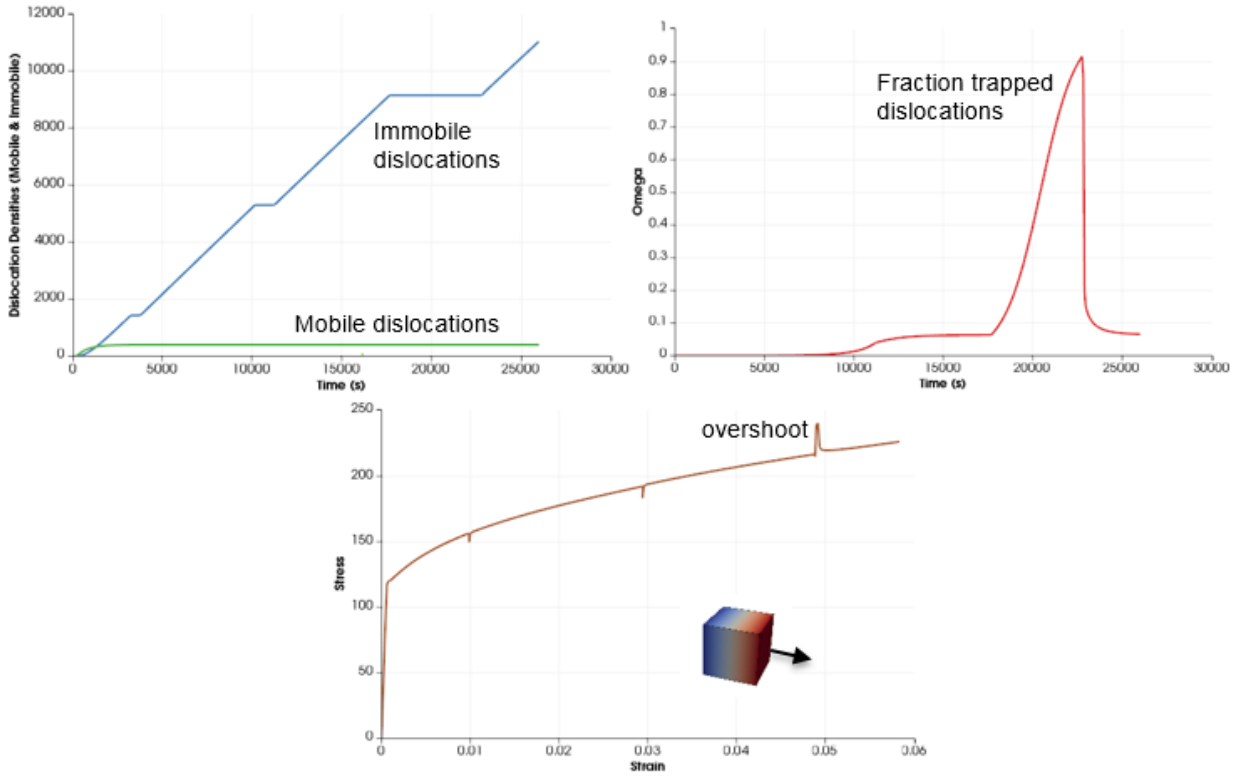

Figure 2.4. (top left) Evolution of internal state variables representing trapped and mobile dislocation densities. (top right) Evolution of internal state variable ω representing fraction of trapped mobile dislocations. (bottom) The net effect on the global stress-strain behavior during a sequence of strain-and-hold events, demonstrating the overshoot phenomena when a large fraction of mobile dislocations become trapped.

The conversion of the BCJ_MEM material model to C++ (DSA model) was a critical step forward in implementing DSA. Several new parameters have been added specifically for use in the DSA evolution equations. The evolution equations are currently integrated explicitly, though future efforts should revisit the numerical integration scheme. Implicit solutions remain a challenge for any simulation attempting to use the DSA physics because the semi-implicit plastic strain solver requires additional work to be compatible.

2.5 Material Model Regression Tests

Material constitutive model regression tests were necessary to guide the conversion of the BCJ_MEM constitutive model from FORTRAN to C++, and also to ensure new features did not impact previ-

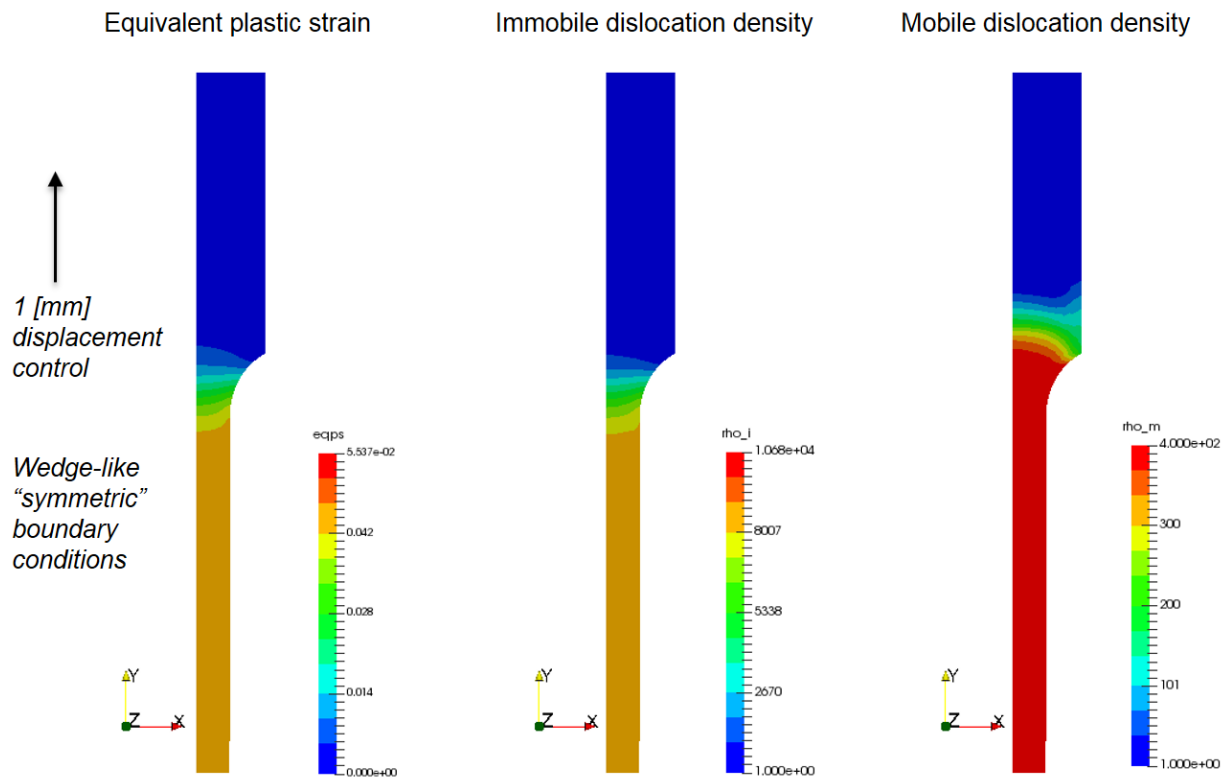

Figure 2.5. Tension specimen simulation and resulting internal state fields for equivalent plastic strain, immobile dislocation density, and mobile dislocation density.

ous behavior in unintended ways. Several verification and regression tests were added to the nightly SIERRA test repository during this process, making the DSA material model more tested than any similar model (e.g., BCJ_MEM, BCJ, elastoviscoplastic, EMMI). The DSA material model was developed originally using all of the previous verification and regression tests for the BCJ_MEM material model. “Gold” results files generated from the original BCJ_MEM material model were used as a basis, ensuring that any new code produced an identical solution to within very tight tolerances (often less than 1e-10 relative tolerance). A summary of these additional tests follows:

- 5 abnormal usage tests transferred from BCJ_MEM ensure valid model input and parameter specification
- A localization patch test transferred from BCJ_MEM to test average pressure formulation for locking
- 2 localization tests transferred from BCJ_MEM for a uni-axial tension condition (with and without initial grain size, recrystallization, hardening, and damage parameters)
- 2 damage evolution tests transferred from BCJ_MEM
- Grain size effects regression test transferred from BCJ_MEM
- Recrystallization evolution test transferred from BCJ_MEM, and modified to include multi-

cycle recrystallization
- Four element-level shear regression tests testing multiple shear directions in both explicit quasi-static and implicit solution modes with the selective-deviatoric element formulation.
- 4 element-level tension regression tests testing multiple constitutive model solvers (original re-coded BCJ_MEM solver as well as a new, more robust DSA solver), implicit solution, explicit solution, and explicit quasi-static solution options
- 4 additional tension and shear tests at multiple strain rates to verify that the DSA material model also replicates the elastoviscoplastic material model.

These verification and regression tests played a critical role in the DSA material model development process, and even helped find (and fix) several bugs in the production BCJ_MEM material model. As a result, the conversion and model development of DSA can also be regarded as a thorough peer-review of the BCJ_MEM model. Note also that since the DSA material model has become a platform for collaborative research efforts that new tests are being added regularly.

2.6 Validation

A formal assessment per V&V guidelines has been prepared where the multi-physics capability presented here utilizing the BCJ_MEM model is compared to experimental data to date in [1]. This work has been built upon previous efforts that can also be found in [4, 13, 3]. In summary, for the L2 Milestone, four separate validation activities were leveraged: validation of the material model, a wedge in which tensile specimens were machined away where the displacement was validated after relaxation, a u-cup which was forged and validated using the resulting residual stress contours both from the simulation and experimental measurements utilizing the contour method, and another u-cup of a different geometry which was forged, machined and welded where the thermal history, yield strength and recrystallization were validated. Detailed information on each of these activities is given in [1].

2.7 Other Impact Areas

The conversion and improvement efforts described herein primarily supported L2 Milestone deliverables through a high-temperature plasticity P&EM project. However, a brief discussion on the positive impacts this work has had on unrelated projects and programs is certainly warranted. The BCJ_MEM material model is a highly effective and widely used material model, so the work done to revitalize the material model has impacted several other programs.

Figure 2.6 illustrates just a few of the modeling efforts that have benefited from the work done as part of this Milestone. The first sub-figure illustrates the canonical *pipe bomb* problem. The pipe bomb problem uses the DSA material model to include temperature and rate dependencies. As heat and pressure are applied to the interior of the cylinder, major plastic deformations take place before

ultimately triggering material failure criteria via element death (collaboration with K. Karlson, J. Ostien, and A. Hanson). In this application, improved robustness have resulted in increased speed and solution accuracy. The middle figure illustrates a smooth tension test simulation in which the DSA material model was used to fit experimental test data (K. Karlson, B. Collins). Again, the DSA material model is being used for its increased speed and robustness and enables better parameter fitting than previously possible. The DSA material model has also fostered several new collaborative efforts. For example, the element death criteria was added to the material model by J. Ostien. The last sub-figure depicts an on-going effort to add evolution equations that model the effects of hydrogen diffusion on the yield surface (G. de Frias and J. Foulk).

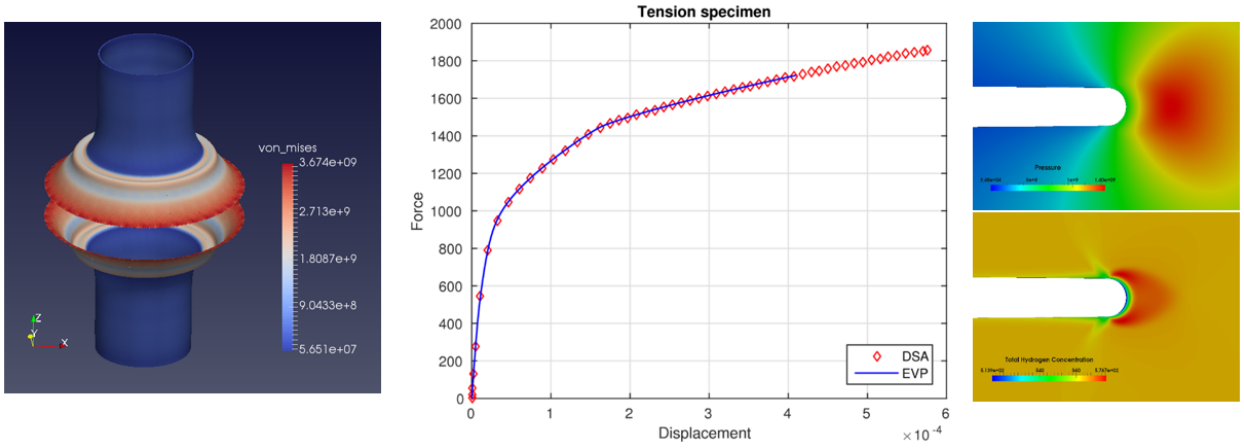

Figure 2.6. Application areas impacted by DSA

This page intentionally left blank.

Chapter 3

Remap and Transfer State from Forging to Machining to Welding

This chapter provides details that address the second L2 milestone completion criteria: demonstrate the capability to remap and transfer material state and residual stress from forging, to machining, and finally to [resistance] welding processes. The deliverable is considered to exceed requirements if a solution can be demonstrated on a representative GTS forging, including initial machining steps, and using implicit contact. A state variable remeshing and mapping tool was developed to perform the necessary material state transfers and machining operations. A representative resistance forge weld problem was used as an "acceptance problem" to anchor and guide development efforts in way that should be broadly applicable to future users. Finally, a complete forging-machining-welding sequence is completed (using representative GTS forging and implicit contact) to demonstrate capabilities meeting the *exceeds* criteria.

3.1 Remeshing and Mapping to Transfer State

3.1.1 Background and Motivation

The requirement to remap and transfer state variables supports large deformation analyses and machining operations. The GTS life cycle involves large deformation processes such as forging and resistance welding, as well as subsequent machining operations. Simulations of these mechanical processes utilize finite-element analysis tools such as the the SIERRA codes to obtain (for example) residual stress and deformation approximations.

Finite element analysis methods require a high-quality mesh to discretize and integrate the governing partial differential equations[1]. Varying quality metrics can be used to describe a mesh, but in general a high-quality mesh has minimal skew (element interior angles near 90 degrees for a hexahedral element, for example) and hence is far from inversion. During the solution of a large deformation process such as metal forming or forging, the initially high-quality mesh deteriorates and, with enough deformation, may become inverted. This poses a numerical issue in finite ele-

[1] Here, the term "high-quality" is employed in a somewhat ad-hoc fashion with the assumption that the reader is familiar with meshing and mesh quality.

ment analysis including reduced convergence rates due to ill-conditioning and ultimately solution termination due to element inversions.

Machining operations can be simulated with varying degrees of complexity. In the most general case, machining is simply the removal of material from a bulk to obtain a more desirable geometry. Here, the term *machining operation* refers to a simulation process or step in which the desirable geometry is "cut" from a bulk material, usually with a subsequent dynamic relaxation step in which residual stresses come to equilibrium. The physical machining process (e.g. cutting on a lathe or mill) along with all of the complexities that go along with it such as surface friction, tribology, tool sharpness, feed rates, etc. is not modeled. However, the end result of these processes is simulated: a new geometry with all of the original residual stresses from forging or heat treatment, but redistributed to maintain equilibrium.

The need to transfer residual stresses, deformations, and internal state variables is common to both large deformation simulations and machining operations. The idea of transferring these variables is not new; previous toolsets (e.g. MAPVAR) have been used to achieve similar means [15] and with varying levels of success. Previous tools have used a straight-forward interpolation of element and node fields onto a new mesh. The algorithms used to interpolate neglected element formulation (e.g., number of Gauss points) and aggregated all data at element centroids. Furthermore, variables for which interpolations are not valid (e.g., rotation matrix components) were directly mapped from source to target mesh. After mapping, the components of a rotation matrix in general will not constitute a proper rotation matrix. Accumulated errors from this process repeated many times (e.g., during a very large deformation or a sequence of machining operations) introduces inaccuracies into the simulation results, which can compound and produce erroneous artifacts in the simulated results. The deficiencies of past mapping processes spurred a revitalization of the mapping and remeshing process. A more formal and rigorous approach to mapping state variables was developed by the authors of [8] and is briefly described in the following section.

3.1.2 Updated Mapping Process

A new set of mapping tools was developed by J. Foulk, A. Mota, M. Veilluex, J. Emery, et al. The new tools are designed to reduce or eliminate inaccuracies of previous tools. A summary and detailed description of the process steps can be found in [8]. Nodal variables map from a source mesh to a target mesh much like previous mapping algorithms: the source field defines a scattered data set, and the target field is evaluated by interpolating within the source fields. However, Gauss point variables and element-centroid variables are mapped more accurately and precisely. A global L^2 projection matrix is formed which projects element (Gauss point or element centroid) data out to a nodal field. Element shape functions are used in this process to accurately resolve gradients within an element. One can show that the L^2 projection minimizes the volume-integrated error – for a given discretization – in the projected fields [12]. Furthermore, by properly interpolating the Lie algebra for rotation matrices (and possibly other fields) we minimize the errors incurred during the mapping process. The new mapping process introduces additional complexity along with the increased accuracy.

A flow chart for the remeshing and mapping process of a typical large deformation simulation is illustrated in Fig. 3.1. The flow chart depicts a sequence of finite-element simulations indexed from $i = 0...N$. Simulation run 0 is the initial simulation with resulting deformation that requires remeshing. The unrotated stress tensor, rotation matrices, displacements, and internal state variables are outputs from the initial simulation step which may have terminated due to poor element quality, or a specified finish time. A logarithm of some variables (e.g., rotation matrices) is necessary so that subsequent algebraic operations such as projections and interpolations are admissible. A program called `logVars` was developed by the authors of [8] and [12] specificially for this purpose. The next step is an L^2 projection performed within the SIERRA/SM code that projects variables from Gauss points or element centroids, out to the nodes. This projection is done at a global level and takes into account the location of Gauss points within an element. There are a number of intricacies within this process that need be carefully handled; in particular, some variables need not have continuous fields across material or contact boundaries. After the projection a second utility is used to update the model coordinates by applying displacements to the reference configuration. The output should be a mesh with node field variables to be interpolated onto the new mesh. At the time the initial simulation is finished, the results must also be imported into Cubit or some other software for re-meshing. Cubit reads in a deformed mesh and corrects any deformed elements via smoothing or completely re-meshing. The new mesh from Cubit is used as a target mesh for interpolating the new fields. The output of the interpolation step is a new mesh with mapped state variables at nodes and gauss points. Any fields (such as rotations) that were log()'d during a previous step are exponentiated to return them to their original space. The reference configuration is updated by applying rotations (e.g., to stress tensors) and resetting the reference configuration. Finally, a third SIERRA simulation is used to ensure the projected fields represent an equilibrium state to ensure state variables, such as plastic strain, lie on the yield surface, contact-induced stresses are in equilibrium, and boundary conditions are re-established. After the equilibrium solve, new and correctly map fields are ready for initializing the next simulation (run 1). The process is repeated until the final deformation state of interest is reached. Note that the process outlined in Fig. 3.1 is specific to completely remeshing a quasi-static solution using 8-node hexahedral elements. Additional steps are required for remeshing and mapping state variables on higher-order elements such as the 10-node composite tetrahedral or 20-node hexahedral element, or for the case where only a small portion (e.g. a single element block) needs to be remeshed. Specifically, mid-side nodes need a special treatment that is beyond the scope of this text. Interested readers are referred to [8] for a more complete and formal explanation of the mapping process.

The dashed boundary indicates a sequence of remeshing and mapping steps that from a user experience point-of-view, should be completely hidden from the user. A re-meshing script was developed that automates this sequence of steps for the user that is briefly described in subsequent sections. The re-meshing step may or may not need to be hidden from the user. In some cases, such as when a hexahedral mesh is desirable, automatic remeshing is simply not possible[2] and so this process should be controlled by the user. In other cases, such as when tetrahedral meshes are used, the process is ideally opaque to the user. The analyst should not be concerned with the remainder of the process in a final production-ready capability. SIERRA developers added many

[2]With some small exceptions. Current progress on CUBIT's mesh scaling and sculpt tools show some promise in this application area.

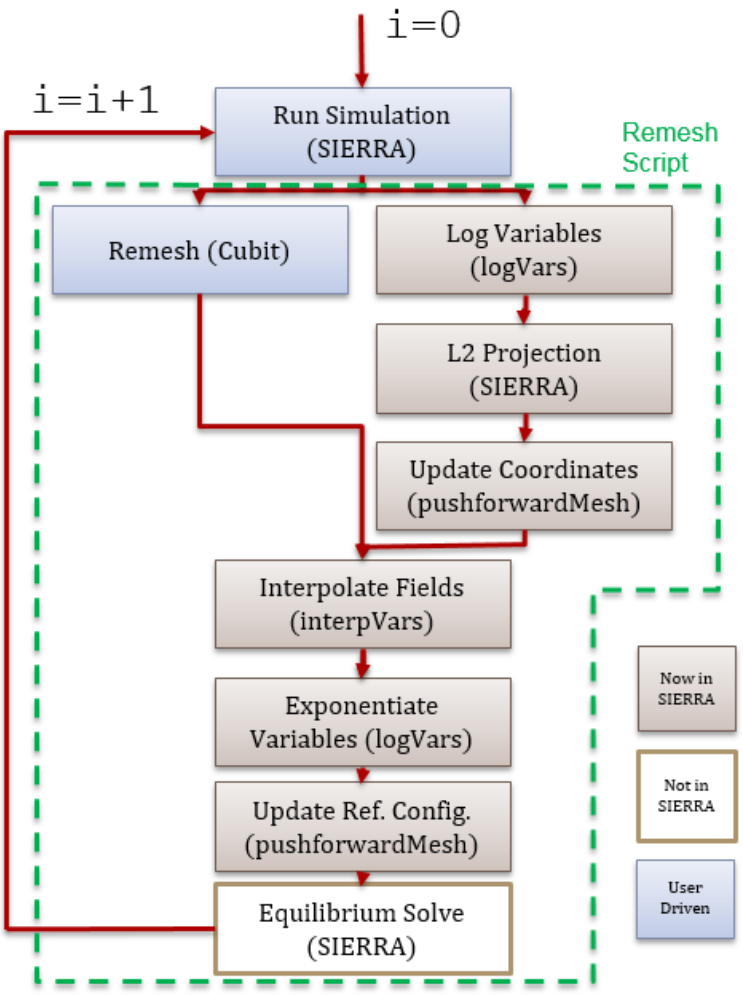

Figure 3.1. Remeshing and mapping process for hexahedral elements.

of these capabilities to the production code to support this milestone. However, a fully-functional capability was not available until the end of FY16 and so the individual programs (logVars, pushforwardMesh, interpolateVars) were used in mapping processes. An equilibrium step is not yet implemented, and several issues still remain in coupled thermo-mechanical analyses run using Arpeggio. These issues remain as future work.

3.1.3 Remeshing and Mapping Tool

`Remesh.py` is a program written in the Python language to simplify the remeshing and mapping process. `Remesh.py` wraps Sierra/SM executables Adagio (and Arpeggio[3]), Cubit, and a set of mapping tools so that the interface exposed to the user is minimal and, theoretically, intuitive. The program can be run locally, or on the high-performance computing clusters with additional input.

The basic input for `Remesh.py` is a set of minimally-modified SIERRA input files, and a 'Remesh Settings Input File'. The 'Remesh Settings' input file requires only a few basic python-style keywords. The basic keywords point to the main input file and mesh, the material model to be used for mapping, a list of times to call the remeshing and mapping tools, and a cubit script to perform the requisite remeshing. In fact, the remesh settings file is a python file and so regular python syntax can be used to perform some operations such as creating a list of remesh times by importing `numpy`.

The box below lists all of the required input commands to start a remeshing and mapping process.

```
input_deck = <string>       # [required] the main SIERRA input file
t_finish = <list>           # [required] a set of analysis finish times
initial_mesh = <string>     # [required] the initial mesh
material_model = <string>   # [required] one of the supported material models
cubit_file = <string>       # [required] a cubit python remeshing file
```

Optionally, three additional parameters can be specified to run jobs on multiple processors (either locally, or on high-performance clusters)

```
analysis_time_limit = <dd:hh:mm:ss, string>     # [optional], queue time
account = <string>                              # [optional], WCID account
num_proc = <int>                                # [optional], number of processors
```

A brief description of each keyword follows.

input_deck A main input file that contains the `begin sierra` block. The file location can be specified relative to the remesh settings file, or can be an absolute path.

t_finish A python-style list of simulation termination times. `Remesh.py` assumes a start time of zero; the first entry should be the first time a remesh and mapping step should be triggered. `Remesh.py` will stop execution after the last time; this is the primary way to control the number of remeshing and mapping steps.

initial_mesh The initial, undeformed mesh to be used for beginning the simulation. Like the input_deck keyword, this can be specified relative to the remesh settings file, or can be an absolute

[3] Arpeggio is the SIERRA executable used to couple Adagio (Solid Mechanics) and Aria (Thermal and Electrical) physics regions

path.

material_model This is a string keyword that specifies the material model being used in the input deck. The material model keyword is used to identify which variables need to be mapped. Currently supported keywords are `elastic`, `elastic_plastic`, `thermoelastic_plastic`, and `bcj_mem` (which can also be used for the in-development DSA model as well).

cubit_file A python scripted cubit file that contains a function with the definition:

```
def remesh(exodus_in, genesis_out, map_args=None):
    return 0
```

The function name should be defined as 'remesh'. The first parameter is the file name that must be imported (in a deformed state) into Cubit. The second parameter is used in the export step. The last parameter is an optional input that can be used to pass additional arguments into the remesh script; this option has limited testing.

analysis_time_limit If the analysis time limit keyword is specified the simulation is automatically queued up using the SIERRA job submission script. This is typically only used when running remeshing jobs on a computing clusters.

account This is the same WCID that would be used (e.g., FY140xxx) when submitting jobs to the clusters. Both `account` and `analysis_time_limit` must be specified if running a job on the computing clusters.

num_proc The `num_proc` keyword is used to run a simulation on multiple processors. If this keyword is provided, SIERRA automatically decomposes the mesh into the appropriate number of processors and submits the jobs. After the job is complete, mesh parts are joined automatically before subsequent operations.

The cubit remeshing script is one of the most critical aspects of the remeshing and mapping process, and a robust implementation may become complex quickly so it deserves a little more explanation. A basic, bare-bones 'null' remesh script example is as follows:

```
import os, sys, cubit
def remesh(exodus_in, genesis_out, map_args=None):
    cubit.init(['', '–nobanner', '–noecho', '–nojournal'])
    cubit.cmd('import mesh geometry "%s" block all use nodeset sideset feature_angle 135.00
        linear deformed last merge' % exodus_in)
    #
    # Meshing command go here, e.g.
    #
    # cubit.cmd('mesh volume 1')
```

```
    cubit.cmd('export genesis "%s" overwrite' % genesis_out)
    cubit.destroy()
    return 0
```

This remeshing script simply initializes cubit, reads in the deformed input mesh, does nothing (null remesh) and outputs the same mesh. In practice this isn't useful at all; but it demonstrates the basic machinery required for a successful remeshing. Almost all Cubit commands can be used within this script to delete, reset, and remesh volumes as necessary. This remeshing script can be as complicated as needed to achieve the desired goals. For example, to remesh the resistance forge weld some additional functions were defined which check hexahedral quality before accepting a mesh; if the quality is too poor, a smaller mesh size is used to improve the quality. Robustly and *automatically* remeshing with hexahedrals is a difficult task at present. There are a number of more advanced 'tricks' that one can used to carefully remesh mesh-based geometry. For example, Cubit will automatically create curves and surfaces for nodesets and sidesets that are defined during the import process. These curves and surfaces can be intelligently selected using Cubit's extended entity selection tools to achieve some surprisingly complex results such as curve biasing, edge refinement, size setting, and interval control in specific areas. Further details are beyond the scope of this report; suffice it to say that the combination of Python and APREPRO script in Cubit can be used to great extent, although significant room for improvement exists.

In principle, the material model could be identified from the user input deck, and a default Cubit remeshing file could be created using a newly-developed "mesh scaling" command in Cubit which can be used to automatically remesh hexahedral meshes. These are possible areas for improvement in a SIERRA-based implementation of remeshing and mapping.

`Remesh.py` also supports a limited restart capability.

```
manual_remap_idx = <int>  # [for restart, required] run index to restart in
manual_remap_mesh = <str> # [for restart, optional] manually remeshed file to use
```

There are some significant limitations in `Remesh.py` that currently cannot be overcome with current versions of SIERRA. Specifically, multiple element types are not supported and multiple material models are not supported. This may not be a major concern for many problems, but certainly is not ideal. However, as the general objective is to implement as much of the remeshing and mapping process as possible directly into SIERRA these functionalities are not likely to be supported in the near term, nor is any significant future development to `Remesh.py` .

`Remesh.py` supports a few additional options that will not be discussed in any great detail, including options to modify paths and output directories, as well as a machining mode. The "machining" mode ran only the process steps necessary to transfer variables from one mesh to another. This capability was added as a short-term solution until capabilities were added into the production code. Nevertheless, it may still be a viable and efficient option for many users.

3.1.4 Machining

The machining process consists of a special use case of the remeshing previously described. In this case, a source results file and a target mesh are provided, where the target resides inside the source (see Fig. 3.2). The state from the source is transferred to the target mesh and an analysis is performed where the new mesh with the imposed state is allowed to relax and come to equilibrium. This process was recently converted over from the process previously described to now be performed entirely within one Sierra input deck with additional support from the integrated codes (IC) team in Q3.

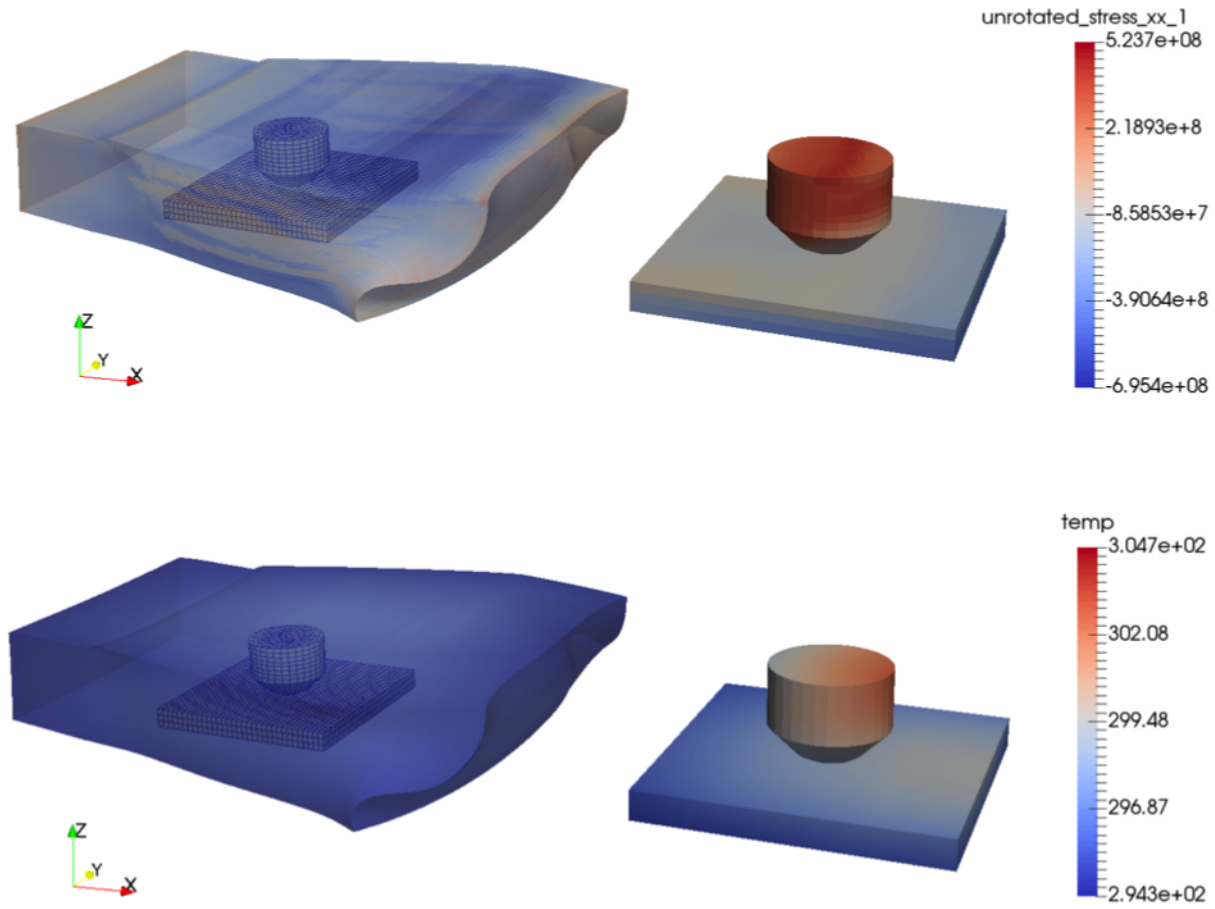

Figure 3.2. Illustration of maching process from forging to RFW surrogate

3.2 Resistance Forge Weld Problem

3.2.1 Motivation and Background

Resistance welding is a common manufacturing process used at Sandia to attach a stem to a high pressure vessel as illustrated in Fig. 3.3. High pressure vessels are complex and expensive to manufacture so replacing stems on existing pressure vessels for reclamation is of particular interest. The resistance welding process forms a solid state bond as a result of grain growth and recrystallization growth across an interface subjected to near-melt temperatures. High temperatures and interference fits required for achieving a solid state weld are also the responsible mechanisms for severe deformations and residual stresses. Cracks along the welded interface have the potential to grow, causing pre-mature pressure vessel failure. Potential crack flaws may be driven by a complex, but unknown, residual stress state following the resistance forge weld and subsequent cooling process. Furthermore, a complex initial stress and material state may already be present

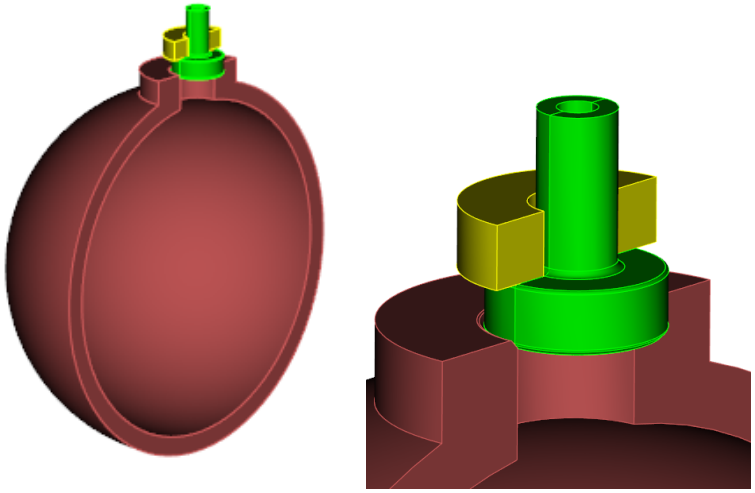

Figure 3.3. Cross-subsection of a generic high pressure vessel with a reclamation stem (green) ready for insertion. The stem is attached via a resistance forge weld in which large electrical currents and high pressure are applied through an electrode (yellow).

before a resistance forge weld even begins. The initial state resulting from material processing and manufacturing steps including initial forging, heat treatments, machining (e.g., the stem from a forging), and cooling all contribute to the final deformation and residual stress state in a non-trivial way. For example, initial forging and heat treatment process parameters can leave some parts with non-uniform yield strength throughout the body. Hence, the goal of studying the resistance forge weld is to understand many of these complexities through modeling and simulation of the entire manufacturing process.

The challenges of the resistance forge welding process are well-known and have been studied

experimentally at Sandia since at least the early 1980s in an attempt to characterize critical process parameters [7, 9]. Through a sequence of experiments, Formisano studied how displacement, current, voltage, and forge applied during a resistance forge weld process related to weld quality. Resistance forge welds were initiated and stopped pre-maturely after one, two, three, four, and five current pulses [7]. Some "cold" studies evaluating contact resistance as a function of applied pressure were conducted in 1989 [9]. In these experiments, resistance weld samples were subjected to static loads while resistance and displacement were measured. Higher contact resistances were observed at lower load levels. Contact resistance decreased and eventually leveled out (presumably near a perfect-conductance limiting value) as the load was increased. A later experimental study by Samuel B. Johnson investigated several processing parameters including weld current, closing force, weld time, plug fit, and interfacial cleanliness [9]. He concluded that the side-bonding that occurs during the resistance forge weld is a "remarkably robust process, extremely tolerant of major variations in processing parameters." Additional experimental studies have helped refine the manufacturing process over time, but limitations on measurable quantities and a desire to better understand the process has largely motivated this work.

Modeling and simulation efforts toward improving our understanding of the resistance forge welding process increased around 2000. The BCJ_MEM model (and its predecessor, the BCJ model) was a critical and necessary development for simulating rate-dependent and temperature-dependent processes in addition to modeling the microstructure evolution process. At the same time, the finite-element codes making up the SIERRA software suite were in full-scale development and evolving to handle more complex and larger problems. About 2003, the Accelerated Strategic Computing Initiative (ASCI) program supported a three year effort intended to position the SIERRA computational tools for the solution of resistance welding problems [16].

A 2005 SAND report [16] documents the first successful solution in SIERRA using a three-way coupled finite-element code[4]. The coupled-physics problem solved was the (now canonical) "tapered bar" acceptance problem. Figure 3.4 illustrates the tapered bar geometry and several cross-sectional snapshots of the temperature and plastic strain fields. A companion C6 experiment resistance welding two symmetric tapered bars at the waste provided experimental data for comparison.

[4]The code was named CALAGIO, coupling the solid mechanics code ADAGIO and the thermal and electrical codes CALORE and eCALORE, respectively.

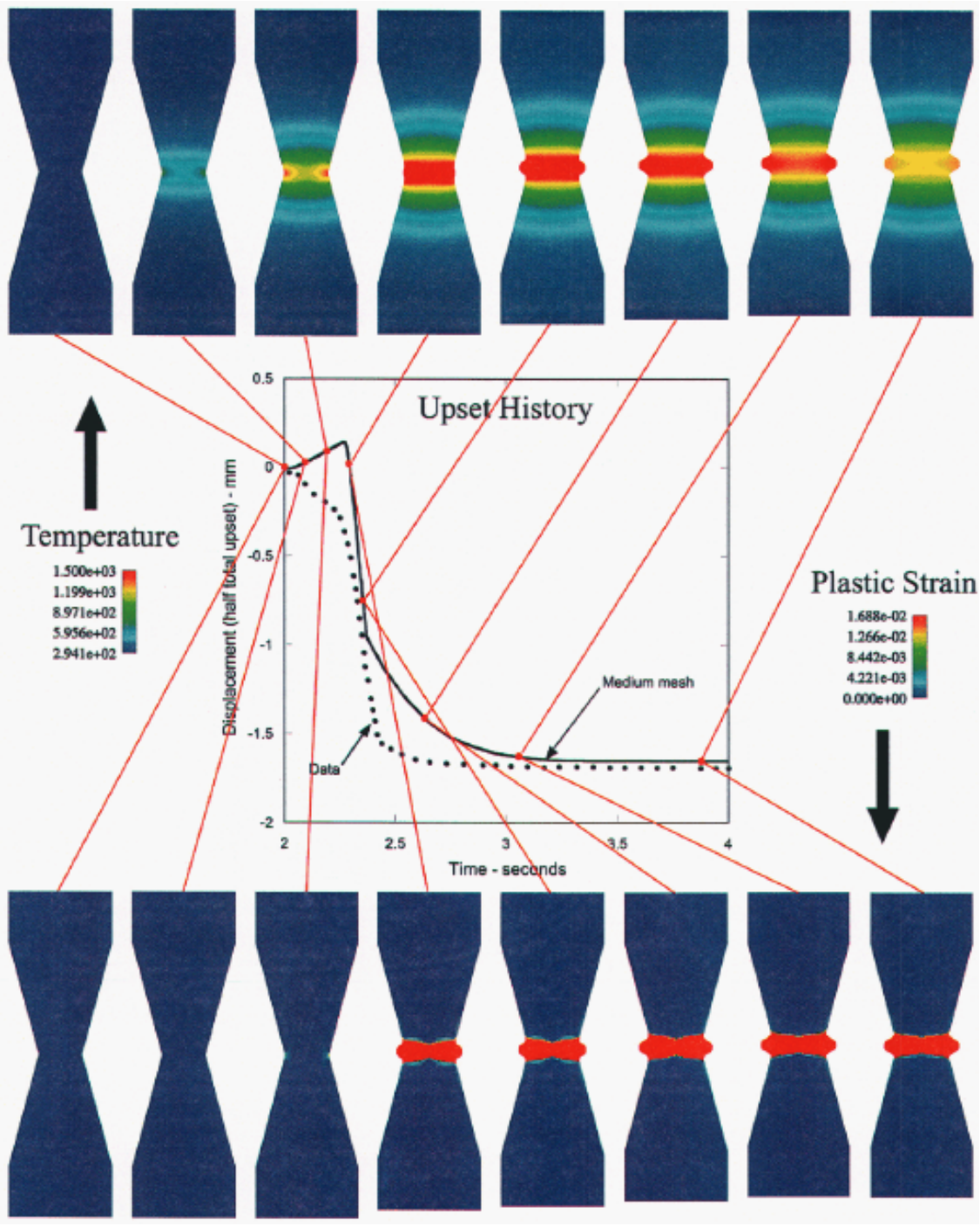

Figure 3.4. Temperature (K) and plastic strain history during upset. Figure reproduced from SAND Report authored by W.S. Winters, A.A. Brown, D. J. Bammann, J. W. Foulk, and A. R. Ortega [16].

Recrystallization kinetics and a rudimentary resistance welding capability were also added to the SIERRA code around this time to support this effort, also described in [16]. The rudimentary resistance welding capability has not changed much in the last 10 years and remains an area for potential future improvements. Resistance welding is assumed to have taken place once the first recrystallization cycle exceeds some critical value (hard-coded as a 50% volume fraction). Once this occurs, a node variable is created that instructs SIERRA to switch from a sliding friction model to tied contact in an attempt to approximate the kinematics of a welded interface.

These models and code features continued to be developed along with other modeling tools, scripts, and codes with various levels of funding and effort. The tapered bar problem was revisited with the more recent Arpeggio finite element coupling code (a replacement for CALAGIO), the BCJ_MEM model with welding and recrystallization features was applied to a pinch-weld simulation, several state variable mapping tools and scripts were produced, and verification and validation efforts continually pushed capabilities forward. However, it was not until around 2013 that any resistance forge weld simulation remotely resembling the geometry of interest (e.g., Fig. 3.3) was seriously attempted. The fact that a serious modeling and simulation effort on representative resistance forge weld geometry did not occur until very recently should indicate the level of complexity involved. All of the aforementioned research efforts (and more) were absolutely necessary before any reasonable attempt could be made and simulating the process.

Still, even with many of the tools in place, procuring a believable result still involves complex physics and numerical solution challenges. The simulation must accurately approximate coupled thermo-mechanical-electrical physics. Joule heating through applied current flux must couple with changing geometry. Heat transfer through all of the necessary mechanisms (conduction, convection, etc.) must be included, along with all of the complexities that go along with discontinuous interfaces (pressure-dependent conduction, mesh tieing, etc.). The solid mechanics portion must deal with rate-dependent and temperature-dependent materials, recrystallization kinetics, resistance weld modeling, and contact at frictional interfaces (among other things). The severe deformations taking place near the welded interface pollute the numerical solution when element quality deteriorates, *requiring* remeshing and state variable mapping onto a better quality mesh.

Figure 3.5 illustrates a typical resistance forge weld simulation result, and some electrical, thermal, and mechanics fields at a snapshot in time during the resistance forge welding process. Three distinct parts are shown: an electrode, a portion of the stem, and a portion of the base (representing the bore in a pressure vessel). The left half of the figure shows steamlines representing the electric current flux passing through from the top electrode to a grounded boundary; colors indicate flux magnitude, increasing near corners and in narrow areas. Joule heating provides a source term that drives temperature gradients, indicated by the white-to-orange-to-black color map on the left side. The rainbow color map on the right half of the figure illustrates the pressure field and typical deformation that takes place as a result of the axial pressure on the electrode used to drive the stem into the base bore.

The remainder and primary objective of this section is intended to describe a representative resistance forge weld simulation done in support of the L2 milestone objective. Specifically, the representative resistance forge weld problem was chosen as a canvas for demonstrating the "capability to remap and transfer material (internal state variables) and residual stress from forging

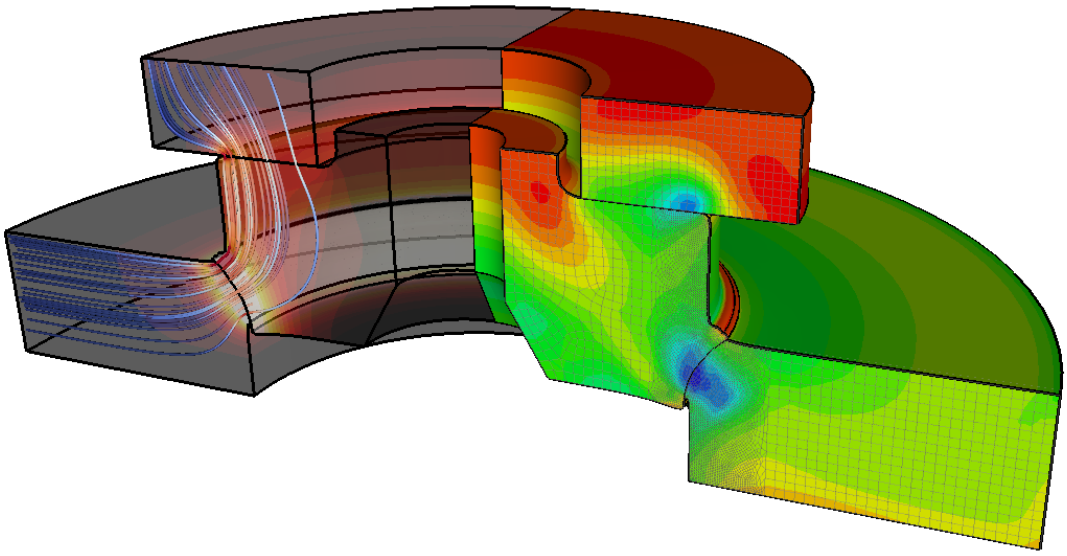

Figure 3.5. Typical simulation of the resistance forge welding process. The cross-subsection shown is a small portion of the stem, base, and electrode surrounding the welded region. The left half shows steamlines to represent the electric current flux and temperature field resulting from Joule heating. The right half shows a pressure field and deformation occuring as a result of simultaneous heating and applied axial pressure on the electrode.

to machining to welding processes." This section documents the first successful attempt at simulating the resistance forge weld process on representative geometry all the way through bonding and subsequent cooling periods. Of course, although this effort was highly successful, solution of the resistance forge weld process is still considered a development capability that requires careful shepherding through the solution by an analyst. The required tools (e.g., remeshing and mapping) and numerical solution capabilities (e.g., implicit contact, code coupling) are still evolving to improve robustness so that verification and validation efforts can be successful. In the following sections, some detail regarding the geometry, meshing (and re-meshing), boundary conditions, solution process, and results are presented.

3.2.2 Model Geometry and Boundary Conditions

Geometry A geometrically-simplified model was used to assess the dominant physics and solution parameters. Figure 3.6 depicts a 90 degree section of the simplified model. Compared with a more complete pressure vessel and stem geometry as depicted in Fig. 3.3, the geometry of the weld surfaces is essentially unchanged; changes are primarily in the geometry extents. The weld stem is shortened, and the base is truncated. The electrode geometry is unchanged.

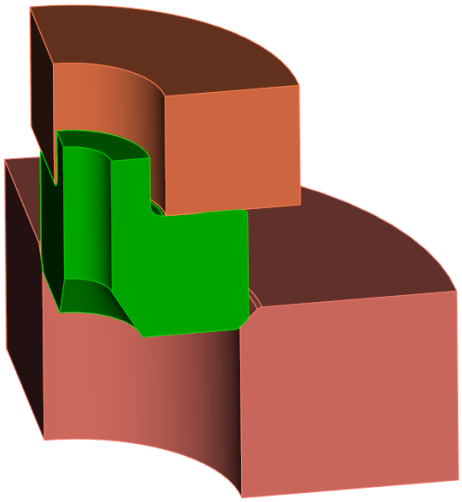

Figure 3.6. Cross-subsection of simplified geometry for modeling the dominant physics processes.

This simplified model represents the combined thermal-heating and large deformation process closely in an area local to the major deformation that occurs, while reducing element count to a level that allows expedient evaluation of relevant process and solution parameters. Of course, the global electrical, thermal, and displacement fields may differ. However, fields local to the forging interface will be similar enough to represent the resistance forge welding process such that parametric studies relevant to the simplified model will also be relevant to larger full-scale models.

An overview of the boundary conditions and component nomenclature (electrode, stem, base) applied to the model is depicted in Fig. 3.7. Three separate and distinct boundary conditions sets are depicted: mechanical BC's, thermal BC's, and electrical BC's.

Mechanical Boundary Conditions The mechanical boundary conditions consist of external loads and kinematic constraints. The only mechanical load is an axial pressure boundary condition applied to the top surface of the electrode. The pressure is defined so that a specific net force, provided by experimental measurements, is obtained throughout the analysis. A linear pressure ramp (starting from zero load) during the first millisecond of deformation added as part of an

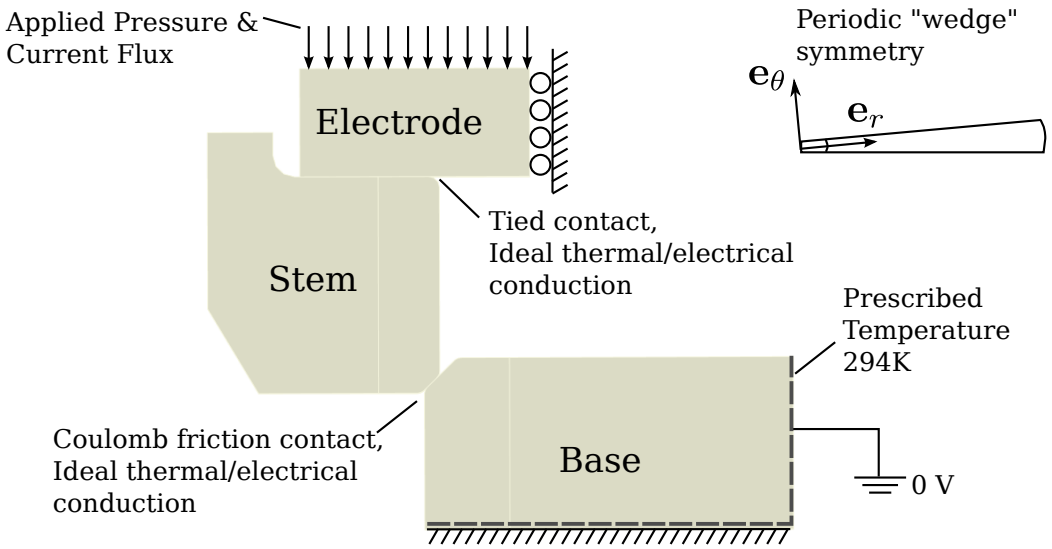

Figure 3.7. Boundary conditions overview for the axisymmetric resistance forge weld

attempt to improve implicit convergence.

The kinematic boundary conditions are slightly more complex, including simple fixed boundaries as well as a complex set of multi-point constraints designed to accurately enforce axisymmetry. All degrees of freedom on the bottom of the base are fixed to prevent rigid body motion and to approximate physical test fixturing constraints. A roller boundary preventing radial motion was added to the stem to prevent non-physical rotations observed during portions of the simulation when high temperatures dramatically decrease stiffness in the stem.

Finally, a set of multi-point constraints was applied to nodes to enforce correct axisymmetry. The SIERRA Solid Mechanics codes are primarily intended for use with 3D geometry and do not contain a proper 2D axisymmetric element. Despite this, Sandia is often interested in geometries that are nominally axisymmetric. Analysts often resort to modeling a much larger portion of a geometry – for example, 90 degrees – and then applying constraints to prevent out-of-plane motion at $\theta = 0^\circ$ (perhaps the XZ-plane) and $\theta = 90^\circ$ (perhaps the YZ plane). This approach has a number of limitations that make it less tha ideal: this is only an approximation for axisymmetry that only truly enforces quarter-symmetry, and a huge number of elements may still be necessary.

For the large majority of resistance forge weld simulations a $\theta = 1^\circ$ wedge was modeled with one face aligned with a coordinate plane. The mesh consisted of a single layer of hexahedral elements constructed with a sweep through-the-thickness. Nodes on that coordinate plane were constrained in the out-of-plane direction, leaving only two remaining displacement degrees of freedom in the radial $\mathbf{e}_r$ and axial $\mathbf{e}_z$ directions. Each node on the opposite wedge face (on the

$\theta = 1^\circ$ plane) requires all three displacement degrees of freedom constrained for true axisymmetry:

$$u_\theta|_{\theta=1^\circ} = u_\theta|_{\theta=0^\circ} = 0 \tag{3.1}$$
$$u_r|_{\theta=1^\circ} = u_r|_{\theta=0^\circ} \tag{3.2}$$
$$u_z|_{\theta=1^\circ} = u_z|_{\theta=0^\circ}. \tag{3.3}$$

These axisymmetry constraints were converted to Cartesian coordinates and enforced with multi-point constraints to ensure that the solution space consists only of potential solutions that are truly axisymmetric. Before a simulation (and after remeshing), a custom Python script read in the mesh and looped over all nodes in a specified nodeset. The script finds a matching node on the opposite surface and constructs the appropriate multi-point constraint. This boundary condition strategy resulted in a more robust contact solution and remeshing capability and was a critical step in the success of this work.

Contact constraints were required at the stem-to-electrode interface, and also at the stem-to-base interface. A simple Couloumb friction model with a somewhat arbitrary friction coefficient of $\mu = 0.1$ was initially used for both interfaces. However, later simulations were eventually run with a tied stem-to-electrode interface to improve contact convergence and reduce the model complexity. The stem-to-base contact interface was specified with the `nodal field dependent friction model`. This friction model reads a `contact_transition` node variable and switches from a Coulomb friction model to tied contact to approximate resistance welding. The `DASH` face-face contact enforcement algorithm (with self-contact search enabled) was used with mostly default settings.

Thermal Boundary Conditions The set of thermal boundary and initial conditions is straightforward. The assembly is initially at room temperature assumed to be 294 K. Two surfaces on the base (indicated with a dashed line in Fig. 3.7 maintain room temperature to provide some heat flow away from the welded area as an approximation of the heat transfer to test fixtures and the remainder of the truncated pressure vessel. Heat generation arises from both thermal-mechanical coupling as well as thermal-electrical coupling. Thermo-mechanical heat generation results from plastic energy dissipation, whereas thermal-electrical heat generation results from Joule heating. Heat conduction across interfaces is accomplished through perfectly tied interfaces. Previous attempts had used more complex interface definitions that included pressure-dependent thermal conductance, but this additional level of detail was removed to reduce extraneous details and simplify the solution process.

Electrical Boundary Conditions The electrical boundary conditions consist of an applied current flux and tied conduction interfaces. Like the thermal interfaces, perfectly tied constraints enforce a continuous voltage field across interfaces. Previous attempts had used more complex pressure-dependent electrical conduction interfaces, but the complexity did not justify the additional level of detail at this point in the solution investigation process. The current flux is applied to the top of the electrode and passes through to ground, governed by electrostatics equations.

A measured weld current-time function from one experiment is shown in Fig. 3.8 (normalized time and current), and was used as the basis for defining a functional current form for modeling purposes. The waveform results from an AC current supply controlled by an SCR phase angle switching [9]. The weld current is nominally periodic with a minor transient during the first three

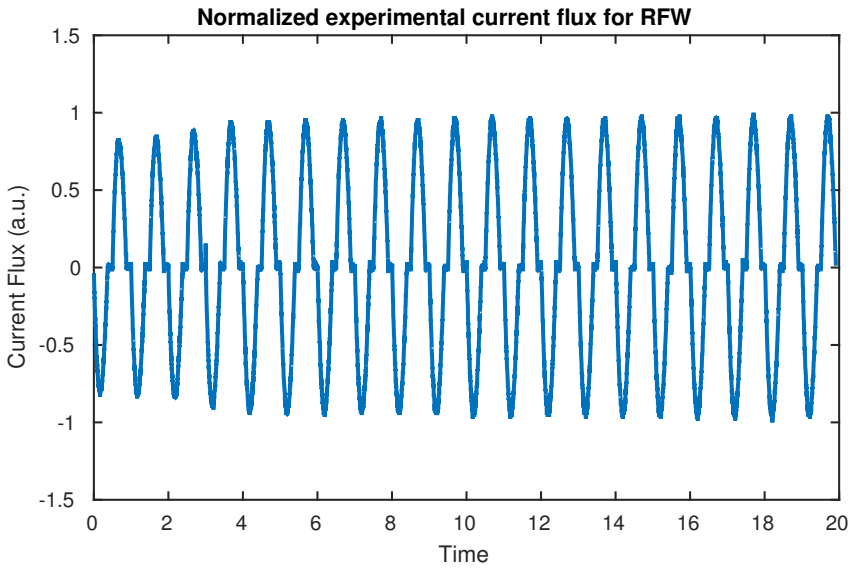

Figure 3.8. Measured weld current vs. time

cycles. Neglecting the initial transient ramp-up, the weld current can be described with a simple periodic signal. The signal is composed of opposing haversine-like functions spaced by a brief dwell at zero current as shown in Fig. 3.9. The periodic weld current model has a number of advantages:

1. The signal properties are easily modified within an input deck (period, amplitude, dwell time, etc.);
2. A Fourier series can completely characterize the signal, so we can provide SIERRA with an analytical function to evaluate at any time and for any duration; and,
3. There is no need to post-process noisy piecewise-linear measured signals which can make convergence difficult for implicit solvers.

A single period of the weld current model is shown in Fig. 3.10 along with a 7 term Fourier series approximation. The amplitude, pulse width, and dwell time were manually selected so that the wave form closely matched the measured current function as shown in Fig 3.9. The wave form is odd ($f(-t) = -f(t)$) so that only odd terms $k = 1,3,5...$ need be included. The Fourier Series approximation of the weld current flux $J(t)$ is given as

$$J(t) = \sum_{k=1}^{N} A_k \cos(k\omega_1 t + \phi_k), \quad k = 1,3,5,...N, \tag{3.4}$$

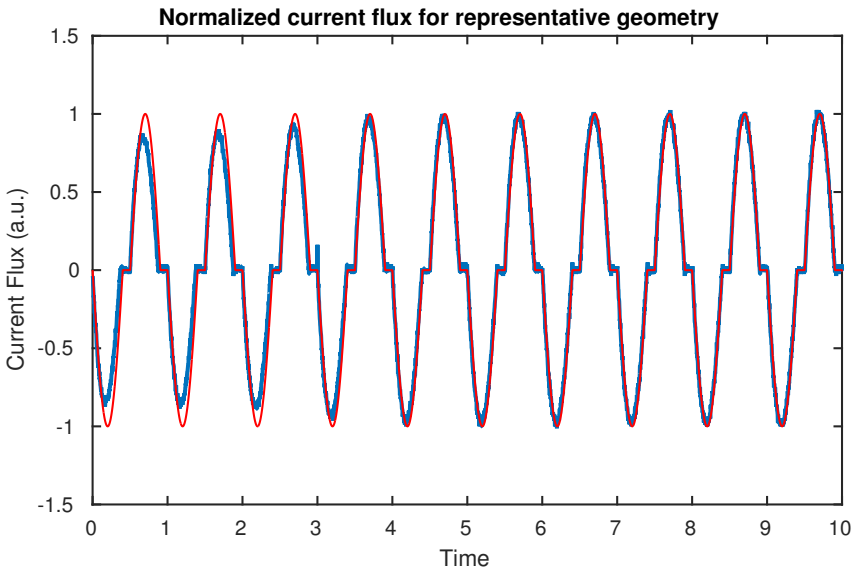

Figure 3.9. Measured weld current compared with a simple periodic weld current model.

where the fundamental frequency is ω_1 and the first few amplitude and frequency pairs are listed in Table 3.1. The first three terms ($k = 1, 3, 5$) account for a majority of the weld current, whereas higher-order terms effectively create the "dwell" phase and contribute very little to the overall heat generation.

Table 3.1. First few amplitude and frequency pairs for the weld current Fourier Series approximation. Note that even harmonics have an amplitude of zero.

Harmonic	**Current Flux** (relative)	**Phase** (rad)
1	0.875	1.88
3	0.172	-0.64
5	0.067	0.02
7	0.027	0.60
9	0.007	1.22

3.2.3 Material Properties: Solid Mechanics Region

All components of the resistance forge weld assembly are made of 304L stainless steel. The BCJ_MEM constitutive model described and validated in [5] was assigned to each of these parts, selected for the ability to model rate-dependent and temperature-dependent plasticity, and recrystallization effects. The parameters for the BCJ_MEM constitutive model were calibrated to test

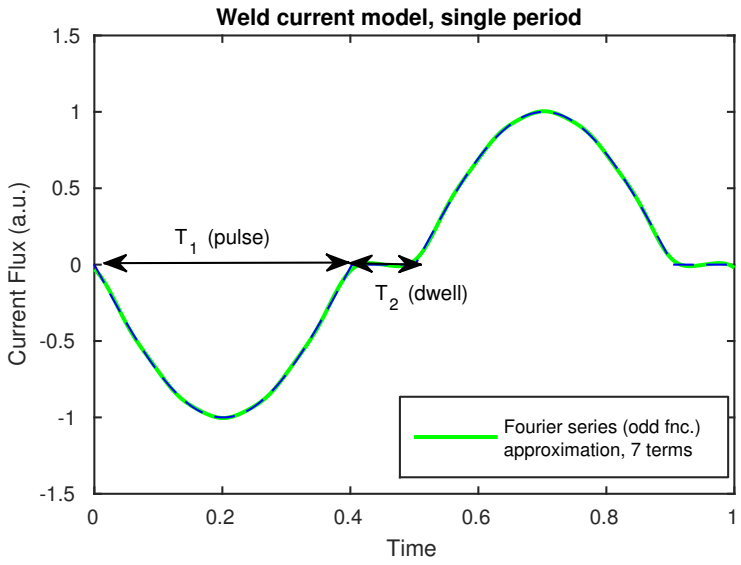

Figure 3.10. Period weld current model and a Fourier series approximation. The signal is odd, so the 7 included terms are $k = 1, 3, 5, 7, 9, 11, 13, 15$.

data from B. Antoun, et al. [4, 3]. The calibrated material model matches rate and temperature-dependent tests and also models the static and dynamic recrystallization necessary for resistance weld models. The copper electrode was also assigned the same 304L mechanical properties due to current limitations in the remeshing and state variable mapping approach. This limitation should be alleviated in the near future as the SIERRA support for internal state variable mapping increases.

The calibrated material parameters used for the BCJ_MEM model are provided in Box 3.1. All material model parameters are defined with regard to the Newton-meter-second system of units, with temperatures in Kelvin. It's also worth noting that the BCJ_MEM model requires an absolute temperature scale to perform correctly. There is no explicit check, warning, or error catching done by SIERRA to verify that the user has chosen an absolute temperature scale. However, the material model will adjust any negative temperatures to a near-zero value.

Box 3.1. BCJ_MEM material model parameters for 304L stainless steel. The unit system is Newtons, meters, and seconds, with temperatures in Kelvin.

```
# temperature−dependent elastic properties
youngs modulus function = stainless_steel_304l_youngs_modulus_multiplier
poissons ratio function = stainless_steel_304l_poissons_ratio_multiplier
youngs modulus = 200e9
poissons ratio = 0.249

# yield and hardening
rate_independent_yield_constant = 1.0528e10
```

```
rate_independent_yield_temperature_dependence = 2.688e+05
rate_independent_yield_temperature_dependence2 = 1.870e−03
rate_independent_yield_temperature_dependence3 = 8.683e+02
rate_independent_yield_temperature_dependence4 = 3.316e+01
flow_rule_coefficient_constant = 9.178e−02
flow_rule_exponent_temperature_dependence = 5.699e+03
isotropic_dynamic_recovery_constant = 8.565e+02
isotropic_dynamic_recovery_temperature_dependence = 5.419e+03
isotropic_hardening_shear_coefficient = 0.01
misorientation_variable_hardening_constant = 1.670e−03
misorientation_variable_hardening_exponent = 1.000e+00

# recrystallization
initial rex volume frac = 0.0001
recrystallization_kinetics_temperature_dependence = 5.e+04
recrystallization_kinetics_mobility_coefficient = 8.846e+16
recrystallization_kinetics_mobility_exponent = 5.431e+00
recrystallization_kinetics_boundary_energy_dependence = 1.1e16
recrystallization_kinetics_multiple_cycle_correction_factor = 1.000e+00
recrystallization_kinetics_boundary_area_exponent1 = 0.6667
recrystallization_kinetics_boundary_area_exponent2 = 1.333

# thermo−mechanical coupling for heat generation
temperature_option = 0
plastic_dissipation_factor = 9.500e−01

# material model solver parameters
newton_solver_method = 1
semi implicit plastic strain solver number of iterations = 100
semi implicit plastic strain solver residual tolerance = 1.e−12
```

A partial description of some parameters follows, but a full discussion is well beyond the scope of this report. Interested readers should refer to the SIERRA Solid Mechanics user manual [14] and references therein, as well as open literature (e.g., [5]) for a more complete description of the material model and its parameters.

Temperature-dependence for the elastic modulus and Poisson's ratio are introduced via temperature-dependent scaling factors. These functions are illustrated in Figs. 3.11 and 3.12, respectively. In defining these functions, the solidus and liquidus temperatures are taken as 1673K and 1723K, respectively. Room temperature is assumed to be approximatley 294K. The elastic modulus and Poisson's ratio scale factors are piece-wise linear, decreasing from the nominal value (at room temperature) to approximately 40% of nominal at the solidus temperature. The elastic modulus decreases rapidly in the 50K degree transition temperature range between the solidus and liquidus temperatures, dropping to very near zero as a very rough approximation of the molten state. Poisson's ratio follows a similar trend, increasing from its nominal value by approximately 50% of nominal at the solidus temperature. Poisson's ratio increases sharply to almost 0.5 at the liquidus temperature as a rough approximation of an incompressible molten state. High-temperature plasticity modeling is still an open area of research and the modeling approximations within these regimes are not without questions. The scale factors depicted were used in initial simulations, but slightly modified in the final simulations used to generated the results presented later on. Specif-

ically, the elastic modulus was allowed to only decrease to 20% of its room-temperature value at the liquidus temperature and Poisson's ratio was only allowed to increase by 70% of it's room-temperature value. These adjustments were made to prevent numerical ill-conditioning with the understanding that results should be scrutinized if these temperature ranges are observed.

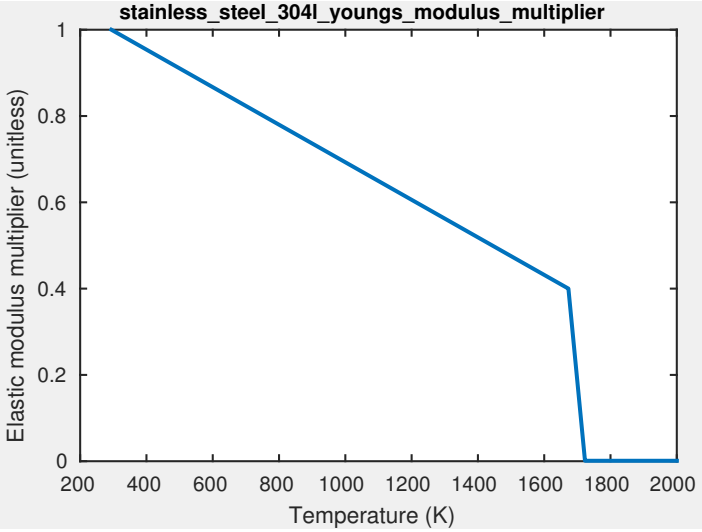

Figure 3.11. Stainless steel 304L youngs modulus multiplier vs. temperature

Linear and isotropic thermal expansion is modeled using an empirical fit based on thermal expansion data for 304L in [2]. An analytic function is provided to SIERRA for evaluating thermal strains ε^T, also depicted in Fig. 3.13, of the form:

$$\varepsilon^T = \frac{ax^{(1+b)}}{(1+b)}, \tag{3.5}$$

where coefficients a and b are fit for temperature ranges between 200K and 1400K

$$a = 2.918\text{e-}6 \tag{3.6}$$
$$b = 0.2543, \tag{3.7}$$

and extrapolated beyond that range. The experimental data fits quite well (see Figure 6 of Ref. [2]) and the authors suggest the analytical fit is accurate to within a $\pm 5\%$ uncertainty.

Finally, it is worth noting that initial simulation attempts used the formal production BCJ_MEM material model but later efforts switched to the development DSA material model. The DSA material model is simply a re-implementation of the BCJ_MEM constutitive laws and behaves nominally identical. However, the DSA material model contains a much improved semi-implicit plastic strain solver that is activated by specifying `newton_solver_method = 1`. DSA also contains some stability enhancements, bug fixes, speed improvements, and a much larger regression test library.

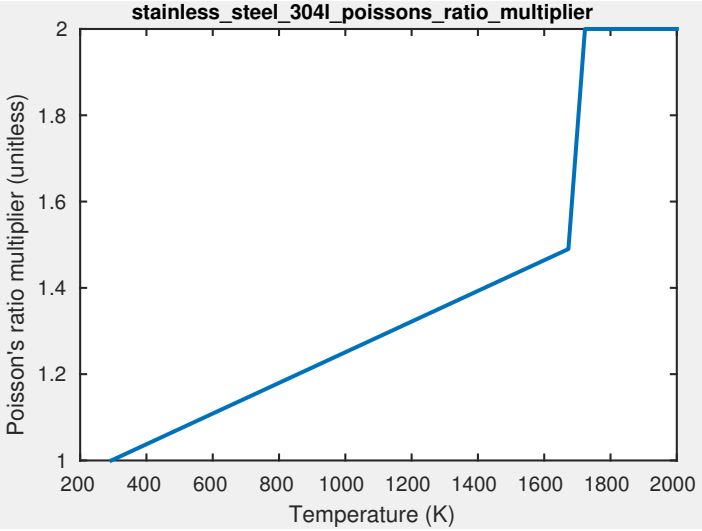

Figure 3.12. Stainless steel 304L Poisson's ratio multiplier vs. temperature

It was observed that using the DSA model with implicit contact resulted in fewer failed contact iterations and generally improved convergence behavior. It is the author's recommendation that the DSA material model eventually replace the BCJ_MEM material model for these reasons.

3.2.4 Material Properties: Thermal and Electrical Regions

The thermal and electrical material models employ basic Fourier's law and Ohm's law for heat conduction and electrostatics, respectively. Functions for temperature-dependent specific heat, thermal conductivity, and electrical conductivity were specified based on data for 304L from [6]. Figures 3.14, 3.15, and 3.16 illustrate the functional forms for thermal conductivity, electrical conductivity, and specific heat, respsectively.

The authors of [6] suggest that the measured values for specific heat and electrical resistivity (or conductivity) are within $\pm 3\%$ and $\pm 5\%$, respectively. No uncertainty approximation is provided for thermal conductivity, as the authors of [6] reference data from another source in their article.

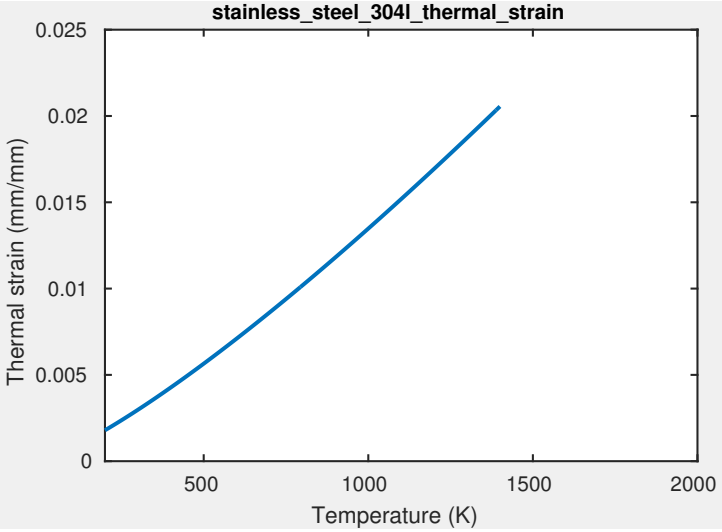

Figure 3.13. Stainless steel 304L thermal strain vs. temperature

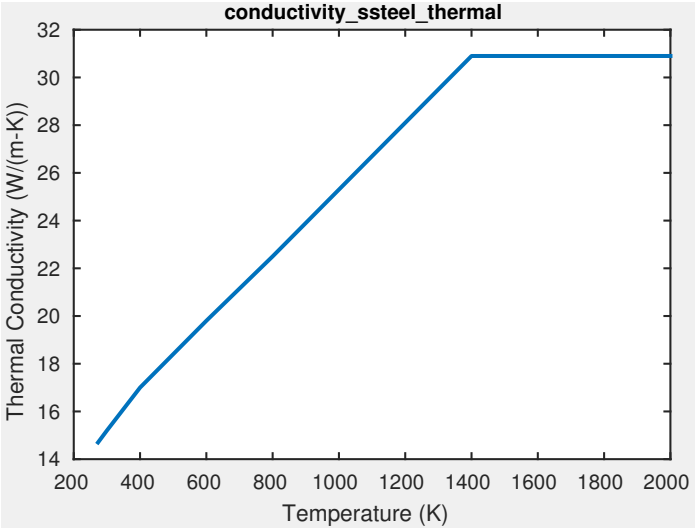

Figure 3.14. Stainless steel 304L thermal conductivity vs. temperature

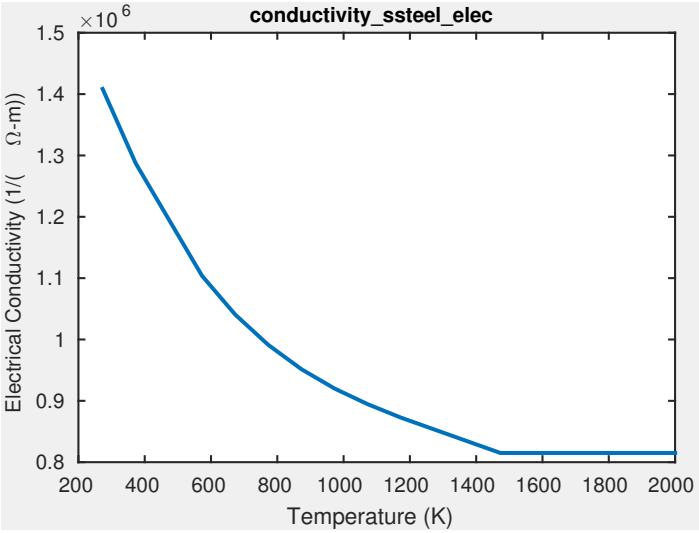

Figure 3.15. Stainless steel 304L electrical conductivity vs. temperature

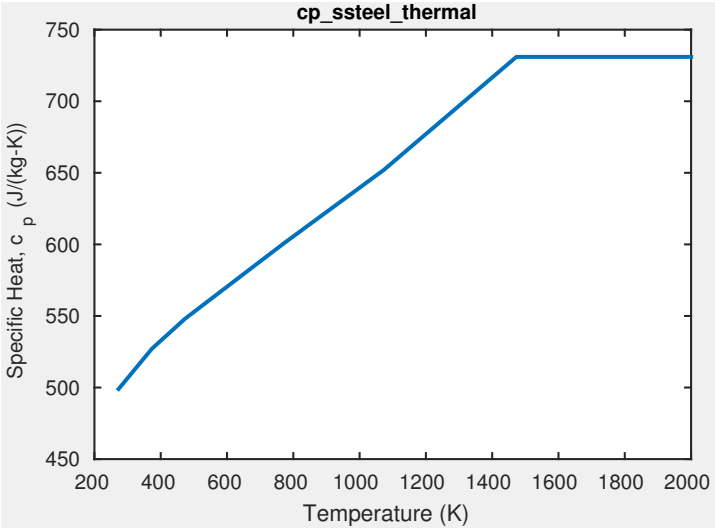

Figure 3.16. Stainless steel 304L specific heat vs. temperature

3.2.5 Numerical Formulation

This section details some of the numerical parameters and formulation choices in solving the resistance forge weld.

Implicit Solution Scheme The results presented here were obtained entirely with an implicit quasi-static integration scheme. Achieving convergence and reasonable solution robustness in this manner is regarded as a huge success and step forward in modeling a complete life cycle process for GTS forgings.

Prior to this work, many resistance forge weld simulations relied on an explicit time integration. An explicit solution steps forward in time using only information known in a previous time step, and as such does not require a converged solution. Explicit solution schemes are generally quite robust in enforcing contact due to the very small changes in contact configuration encountered over a time step. However, explicit time stepping schemes are conditionally stable and suffer from a critical time step limitation. The time scale of forging, welding, and subsequent cooling are long enough that an explicit solution can become prohibitive. Furthermore, as deformation progresses element quality degrades and further reduces the stable time step. There exist well-known ad-hoc methods for speeding up solution times such as *mass scaling* techniques and *time compression*. Mass scaling invovles artificially increasing the material density to increase the stable time step. However, large density increases lead to inertial effects that lead to artificial jetting and prevent correct application of force-type boundary conditions. Time compression is not an effective solution in this case because an adjusted time scale interferes with rate-dependent material properties and generally convoludes the entire solution because of adjustments needed to time-dependent functions (e.g., current flux), diffusion coefficients, etc. Therefore, although an explicit time-integration solution is ideal for robustness – and in particular, contact robustness – the drawbacks are large enough that an implicit solution scheme was pursued, instead.

An implicit solution scheme, unlike an explicit one, is unconditionally stable for large time steps. However, a converged solution for a system of equations is required that dramatically decreases solution robustness, particularly in contact problems. The benefits of an implicit solution are primarily that the solution is obtained to within a specified degree of accuracy (the solution tolerance) and large time steps are stable. Further, because an implicit solution is unconditionally stable in time, localized mesh refinement can be used to resolve sharp gradients without adversely affecting the time step[5].

SIERRA/SM solves implicit quasi-static problems using a conjugate-gradient algorithm with an optional (but almost-always necessary) full tangent preconditioner [14]. Problems involving contact require a multi-level iterative solution strategy that SIERRA/SM terms “control contact.” The reason for this is that the contact configuration of a future step is unknown, so an iterative approach is necessary to solve the contact problem. The implicit solver parameters ultimately used to obtain a reasonably robust solution are provided for reference in Box 3.2.

[5]However, dramatic differences in element size or quality can adversely affect the numerical conditioning of the implicit system of equations

Box 3.2. Implicit solver parameters for SIERRA/SM

```
begin solver
    begin loadstep predictor
        type = scale_factor
        scale factor = 0.0
    end
    level 1 predictor = none

    begin cg
        target relative residual = 0.5e−3
        acceptable relative residual = 1000

        reference = belytschko
        minimum iterations = 1
        maximum iterations = 50
        iteration print = 1

        begin full tangent preconditioner
            iteration update = 30
            small number of iterations = 25
            minimum smoothing iterations = 5
        end full tangent preconditioner
    end cg

    begin control contact
        target relative residual = 1e−3
        acceptable relative residual = 1e−2
        maximum iterations = 50
    end
end solver

begin adaptive time stepping
    cutback factor = 0.5
    maximum failure cutbacks = 10
end
```

Mostly default solver parameters were used with a few exceptions. The load step predictor was turned off by setting the scale factor to 0.0 to improve contact convergence. The conjugate gradient solver iterates within a contact iteration which was solved to a relative residual tolerance of 1e-3, referenced to a Belytchko force norm. Lower tolerances can be used, but often resulted in stalled convergence and eventually a failed solution. The full tangent preconditioner was critical to obtaining any solution and the author recommends that it is used for any implicit problem. A maximum of 50 conjugate gradient iterations were allowed, with a tangent update forced after 30 iterations. Five "smoothing iterations" were used for each model problem and usually resulted in a rapid decrease of the residual by at least an order of magnitude with very little cost. Finally, a basic adaptive time stepping scheme was specified. In the event a converged solution could not be obtained, the time step was halved before attempting the time step again.

Element Formulation The resistance forge weld was meshed with only hexahedral elements. The default element formulation in SIERRA/SM is the uniform gradient element [14]. This formulation uses a single integration point located at the element centroid, but suffers from well-known hourglassing instabilities. It is a relatively robust and efficient element and in many situations instabilities can be remedied through hourglass stiffness and viscosity terms. However, these spurious zero energy modes quickly manifested in the resistance forge weld simulations and so the selective deviatoric element formulation was used instead. The selective deviatoric element uses eight Gauss points but does not exhibit any spurious hourglass modes, and also performs quite well with nearly incompressible materials. The selective deviatoric element formulation used default values for the `deviatoric parameter` (`1.0`) value and `strain incrementation` scheme (`midpoint_increment`). The former is required to eliminate possible zero energy modes, since it effectively blends a mean quadrature scheme and a selective deviatoric stress integration scheme [14]. The `strain incrementation` scheme deserves some attention. The default strain incrementation scheme uses a midpoint-increment formulation; an alternative option available to users is a strongly objective scheme which is generally preferred for problems involving large rotations over time. However, the strongly objective formulation should not be used in problems involving remeshing and state variable mapping because current schemes update the reference configuration. The updated reference configuration is problematic for the strongly objective algorithm because the initial deformation gradient is reset after each remesh step, leading to incorrect results and poor (or impossible) convergence.

Multiphysics Coupling The numerical coupling approach to solve the multiphysics of the resistance forge welding problem deserves some discussion. There several approaches to solving a coupled physics problem, each with pros and cons. A *loosely* three-way coupled approach was applied to this problem in which electrical, thermal, and solid mechanics governing equations were solved in a serial, segregated fashion at each time step. Solution fields from previous steps are passed forward and used in subsequent steps (e.g., updating a temperature field and using it to update material properties in a solid mechanics step). The solution sequence begins with an implicit electrostatic solution, followed by an implicit heat transfer solution, and finally an implicit solid mechanics solution step. Joule heating in the thermal step is informed by the solution of the previous electrostatics equation, and temperature-dependent material properties in the solid mechanics step are informed by the temperature field of the heat transfer step. Finally, the solid mechanics step provides updated model coordinates a heat generation term for use in the new time step. A time step of 0.1 ms was used for all physics regions. Each of the phsyics regions (electrical, thermal, solid mechanics) is solved implicity to a converged tolerance; however, there is no outside loop to ensure an overall converged solution is obtained. Although such a solution strategy is desirable in many cases, very slow convergence rates in such iterative multiphysics coupling schemes can prevent obtaining any solution at all. In practice, the loose coupling scheme has been an effective strategy for solving thermo-mechanical modeling where coupling effects can often be approximated as one-way (temperature fields have the most effect on material properties, but not vise-versa) as long as the time step is chosen appropriately.

3.2.6 Simulation Results

This section describes simulation results for the representative resistance forge welding problem. As noted previously, these results are intended to illustrate an application of the tools and capabilities developed as part of the L2 milestone requirements. Specifically, the successful simulation of the resistance forge welding problem repeatedly demonstrates a "capability to remap and transfer material state (internal state variables) and residual stresses" during a welding process. The same capability can be (and has been) successfully applied to a simulated machining process. The results described next were all obtained using implicit solution methods (including implicit contact) on a representative GTS forging to support the "exceeds" case.

First, we illustrate a scenario in which remeshing (and subsequent mapping) is necessary. Then, the evolution of several selected material state variables and nodal fields is illustrated, including the evolution of a residual stress state following cooling. Finally, the same problem is simulated using an initial residual stress state mapped from previous forging, machining, and heat treatment steps. Emphasis is placed on comparing these results with those of an uninitialized simulation (e.g., assuming annealed stress-free components).

Figures 3.17, 3.18, 3.19, and 3.20 illustrate geometry deformation snapshots at 13 ms simulation time in both the reference mesh and remeshed states. The initial resistance forge weld geometry is as depicted in Fig. 3.7 (and other previous figures). The initial mesh consists of high-quality hexahedral elements with nodal jacobian ratios greater than 0.25. After approximately 13 ms, mesh deforms as depicted in Fig. 3.17. Figure 3.19 provides zoomed view of Fig. 3.17 near the stem-to-base interface wehre the deformation is the greatest. The shaded (red) element along the boundary has almost collapsed due to severe plastic shear deformation and is nearing inversion. Decreasing element quality poses numerical convergence challenges, and will ultimately cause the solution to fail. The solution is stopped at this point and remeshed; the existing stress state and internal state variables are mapped onto the new reference configuration depicted in Figs. 3.18 (global) and 3.20 (zoom), respectively. The new high-quality mesh is used to continue the simulation process.

Figures 3.21 (radial stress field), 3.22 (shear stress), 3.23 (temperature field), 3.24 (hardening field κ), and 3.25 (welded regions) have been selected as representative results for describing the complete resistance forge welding process. Each sequence is viewed left-to-right, and top-to-bottom. Five remeshing and mapping steps were required to resolve the large deformation; these steps were initiated by solution termination due to poor element quality. Poor element quality was defined to have a nodal jacobian ratio of less than 0.15.

The resistance forge weld begins in an initially stress-free state, with all parts having equal yield strength. During the first current pulse large shear and compression stresses develop. Joule heating drives localized temperature gradients near the stem-to-base interface develop due to higher current fluxes as a result of the smaller cross-section. High temperatures soften the 304L near the interface, but plastic deformation-induced hardening (seen as increases in κ) provides an opposing effect. During current pulse dwells, sharp temperature gradients near the welding interface diffuse into the neighboring bulk material. Local heat diffusion near the welding interface drives

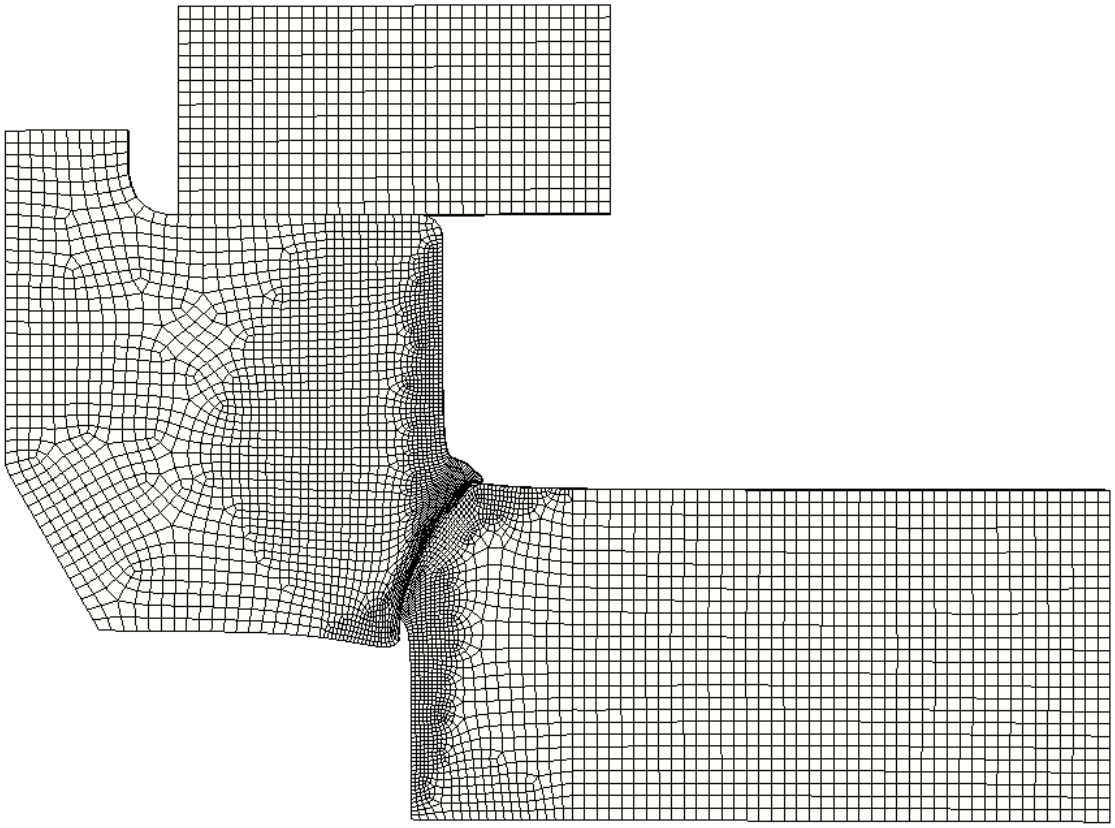

Figure 3.17. Deformed mesh of initial geometry

the recrystallization process, which ultimately leads to the simulated resistance weld bond. The 'ibond' internal state variable is binary and takes on a value of 1 only when a sufficient amount of recrystallization has taken place (0.5 recrystallized volume fraction at a Gauss point). Physically, an ibond value of 1 indicates the potential of material to form a solid state bond across the interface; technically, a value of one communicates to the contact algorithm that a change in contact enforcement is necessary. The last subfigure of Fig. 3.25 illustrates the that a resistance weld bond initializes after about 0.13 ms and is fully bonded shortly thereafter at about 0.040 ms.

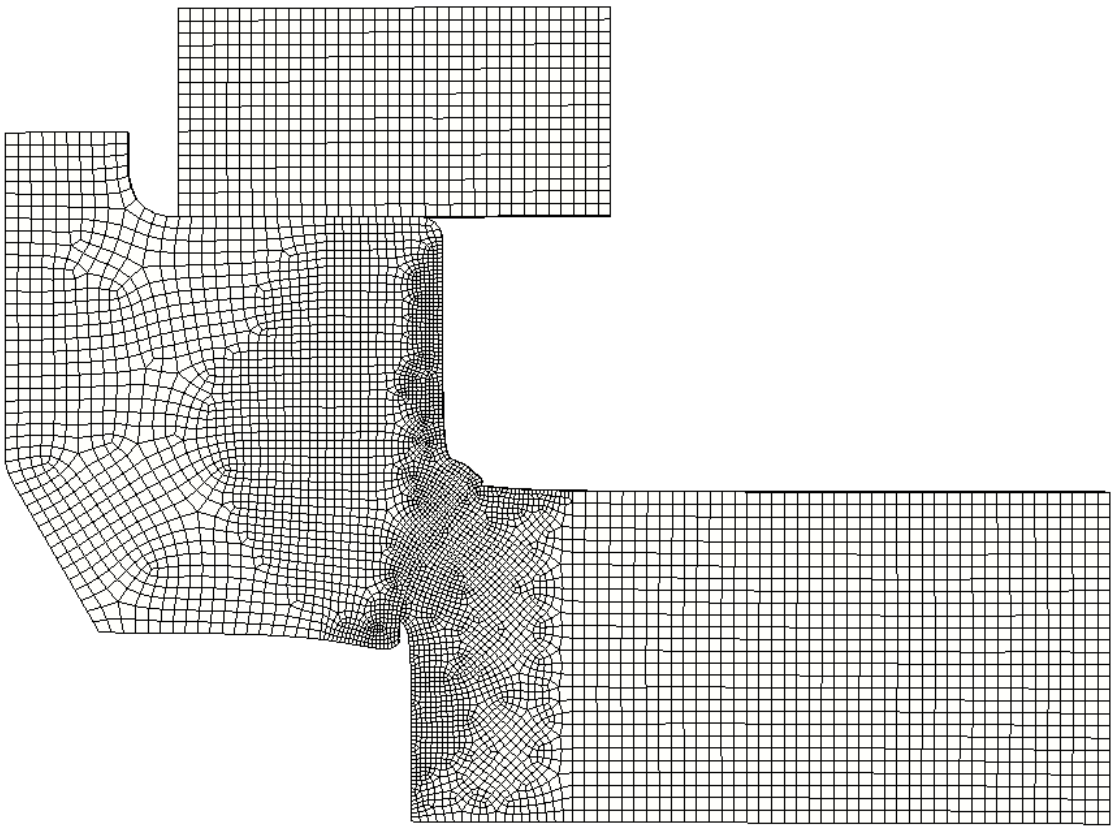

Figure 3.18. Remesh of deformed configuration (compare with Fig. 3.17)

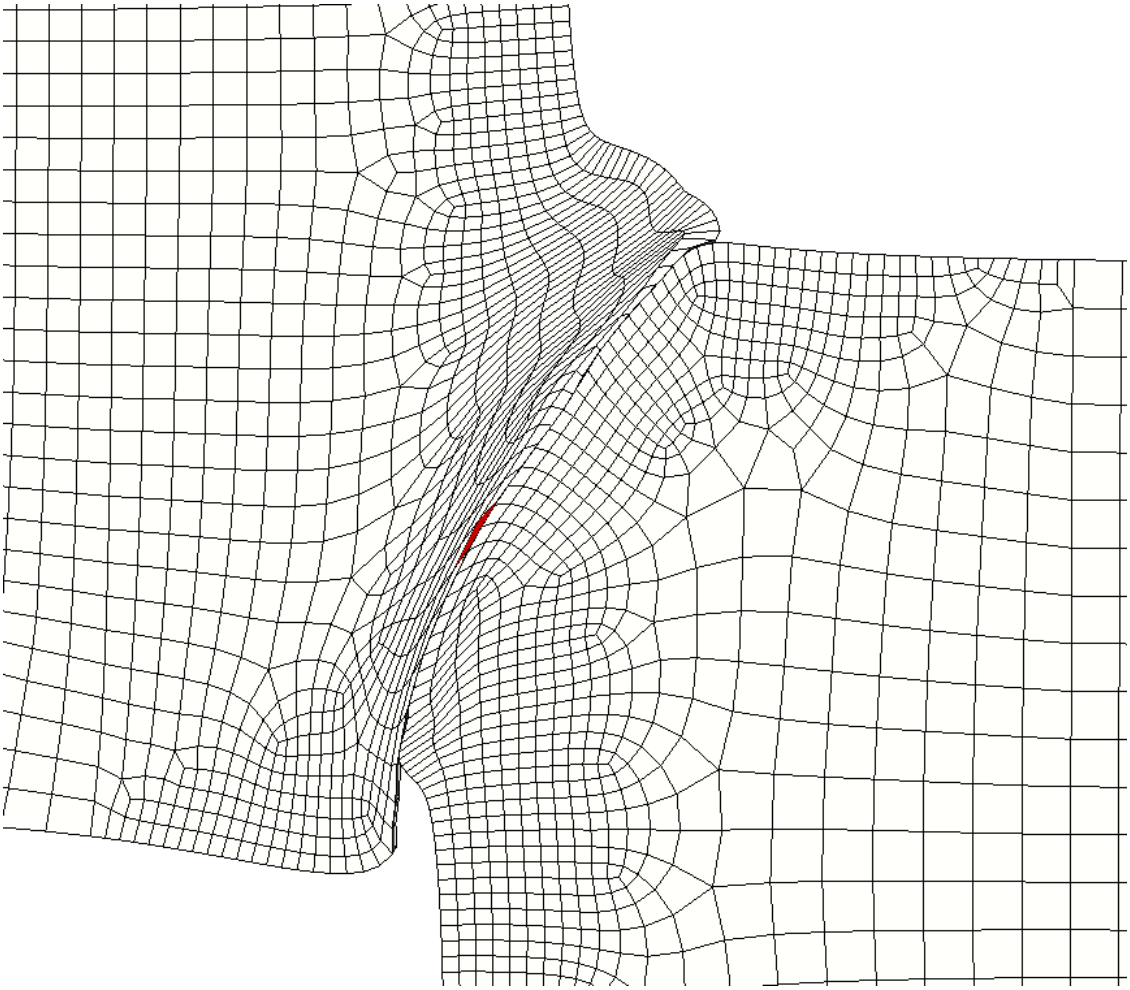

Figure 3.19. Zoom-in view of initial geometry deformed mesh. Shaded element along stem-to-base interface has poor element quality (nodal jacobian ratio ≈ 0.147) and is near inversion.

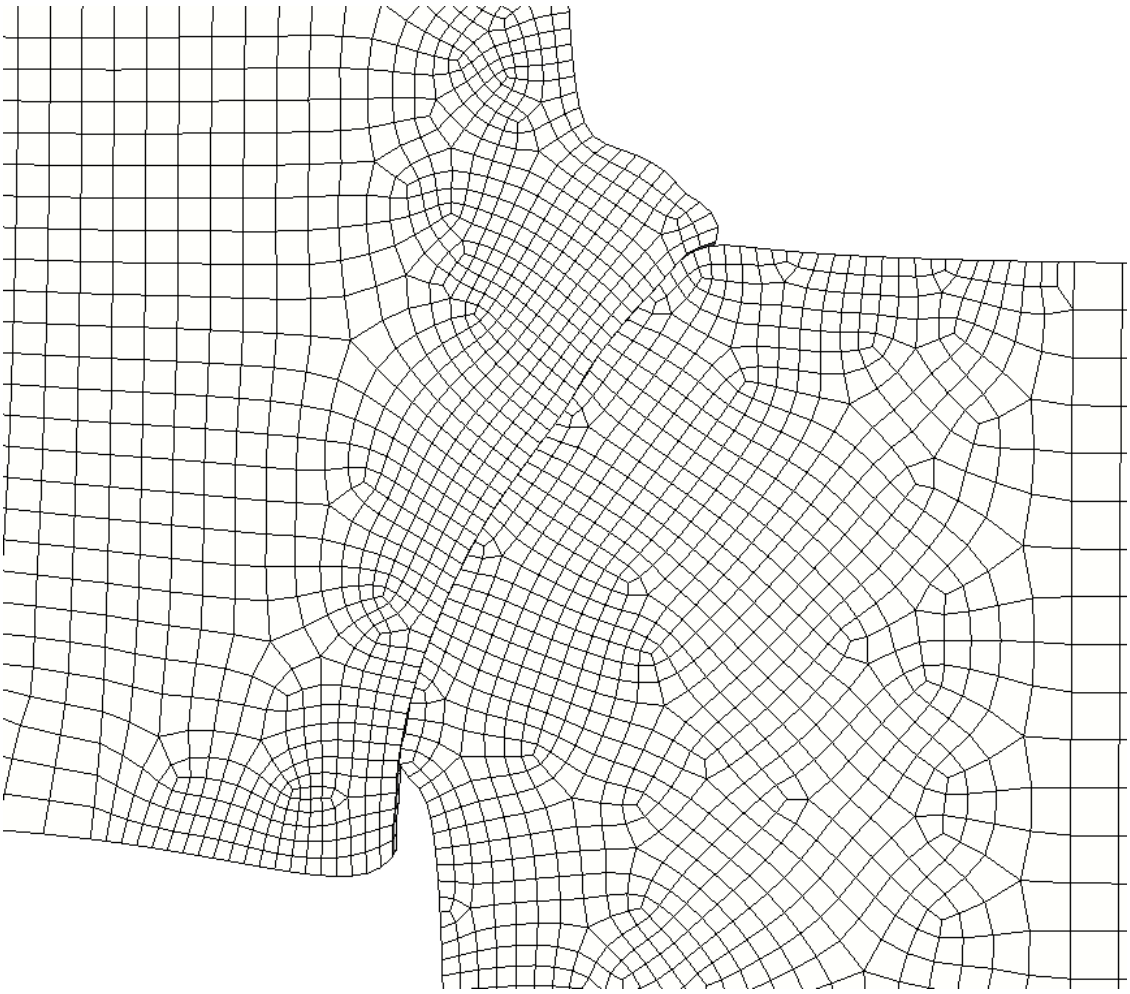

Figure 3.20. Zoom-in view of deformed configuration after remeshing (compare with Fig. 3.19)

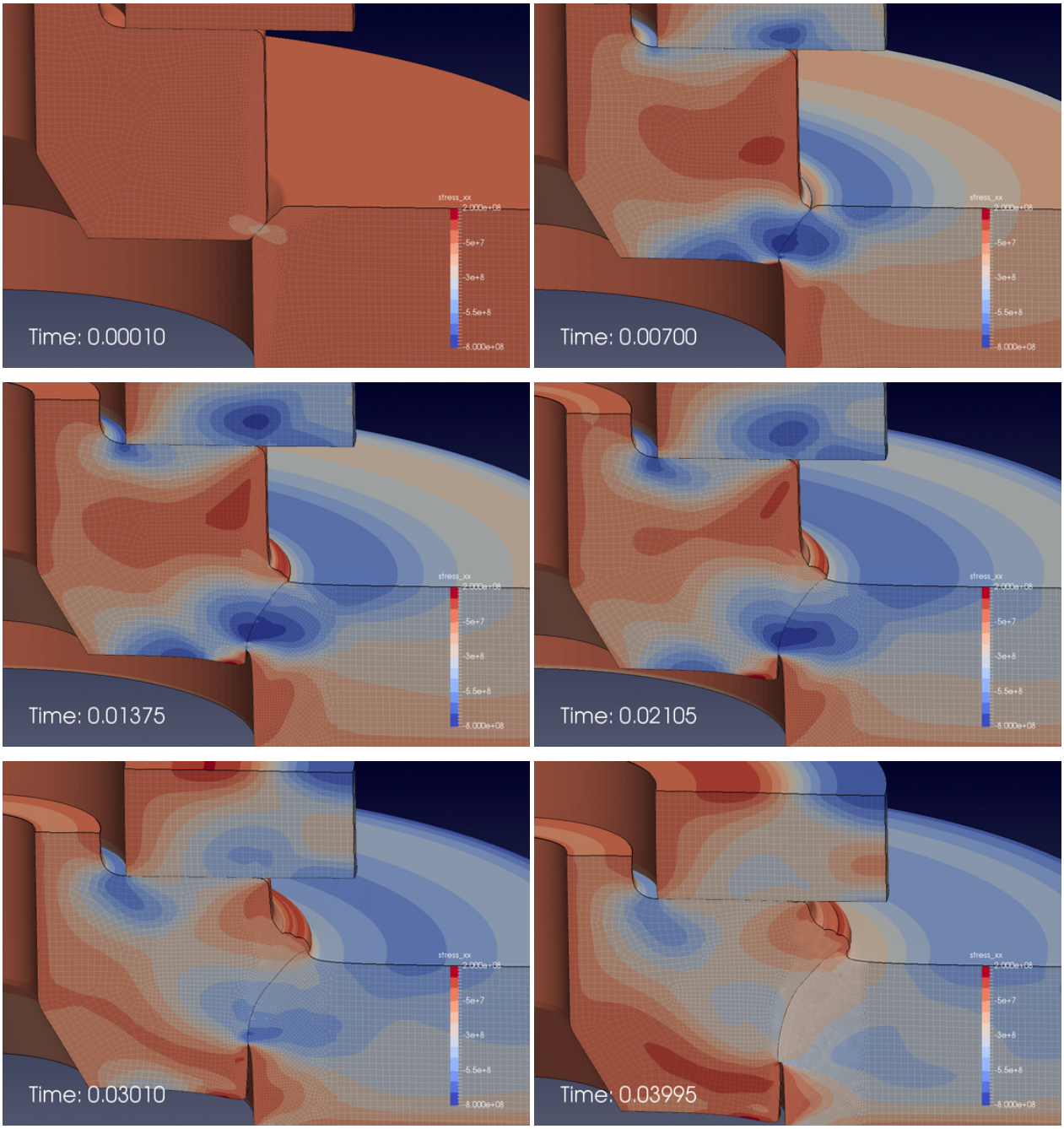

Figure 3.21. Residual stress evolution (in-plane, radial component) during a resistance forge weld simulation *(color map range -800 MPa to 200 MPa)*. The sequence is viewed left-to-right, top-to-bottom. Five remeshing/mapping steps were required to resolve the large deformation. Notice the decrease in stress magnitude as high temperatures and deformations lead to softening and recrystallization.

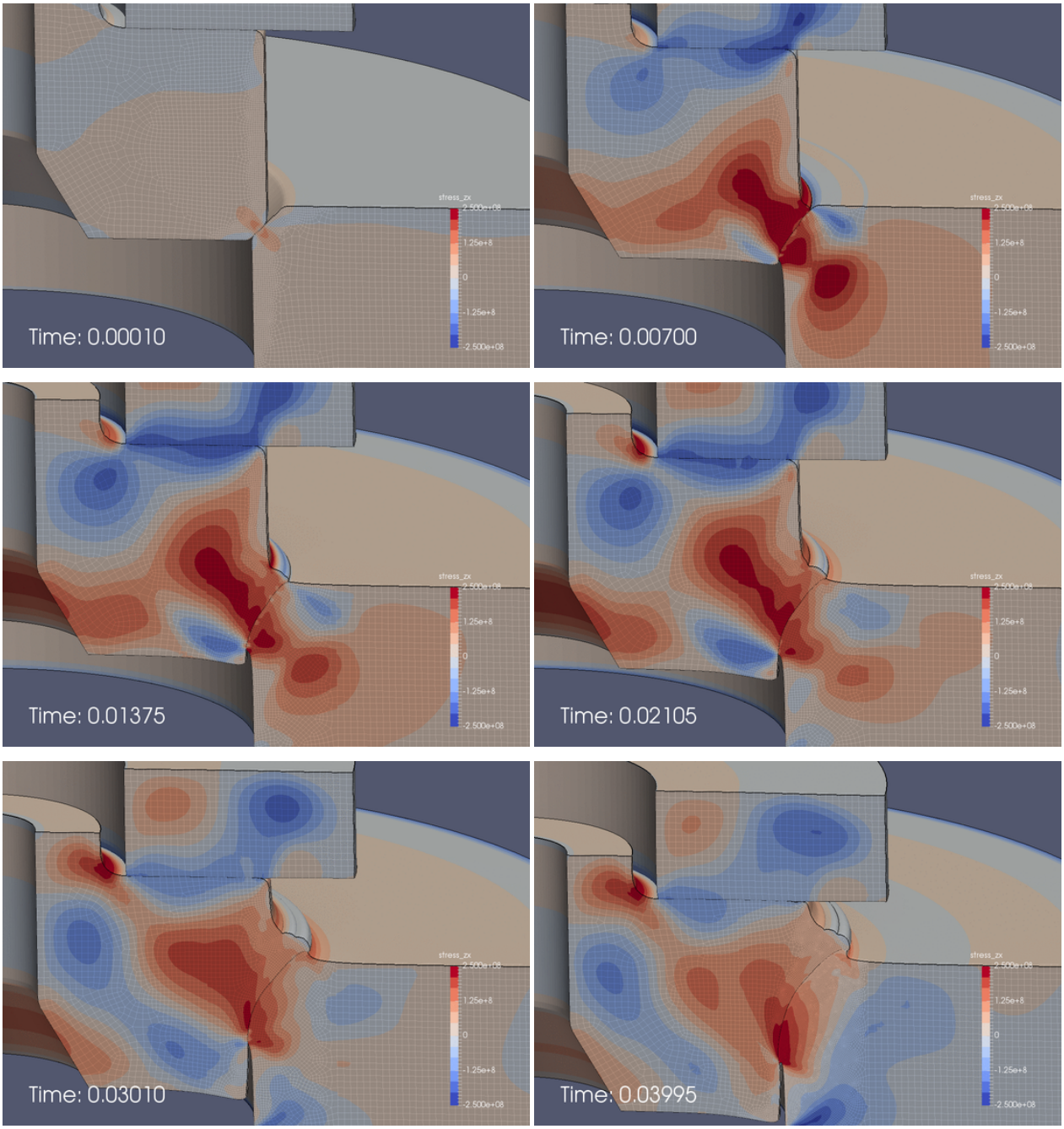

Figure 3.22. Residual stress evolution (in-plane shear component) during a resistance forge weld simulation *(color map range -250 MPa to 250 MPa)*. The sequence is viewed left-to-right, top-to-bottom. Five remeshing/mapping steps were required to resolve the large deformation. Note the shear stress sign reversal at the bottom of the stem-to-base interface. Stresses introduced by the interference fit and forging are immediately relieved when the contacting interface ends, resulting in a sharp stress gradient.

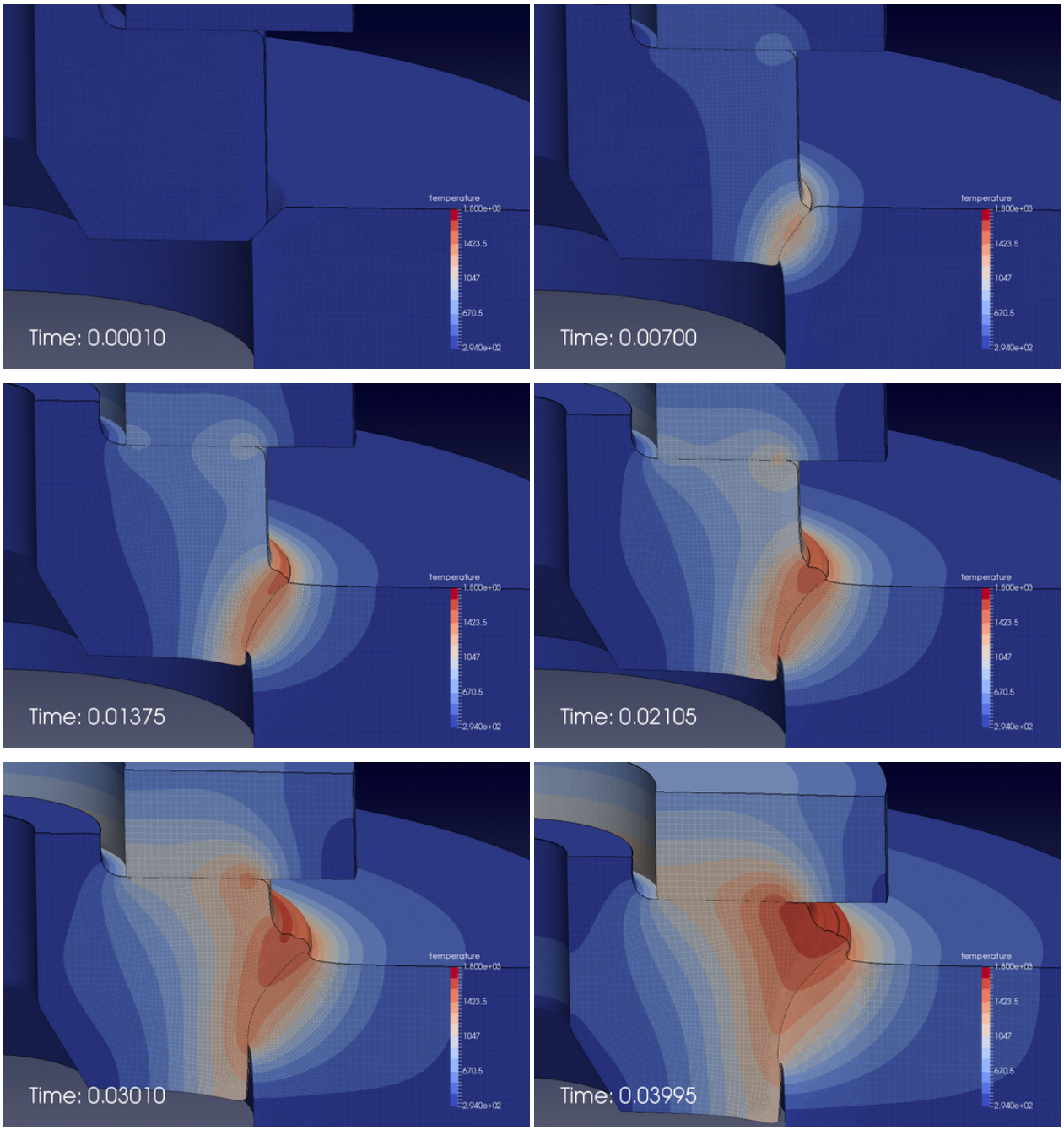

Figure 3.23. Temperature evolution during a resistance forge weld simulation *(color map range 294K to 1800K)*. The sequence is viewed left-to-right, top-to-bottom. Five remeshing/mapping steps were required to resolve the large deformation. Temperatures increase during current pulses, especially near the stem-to-base interface.

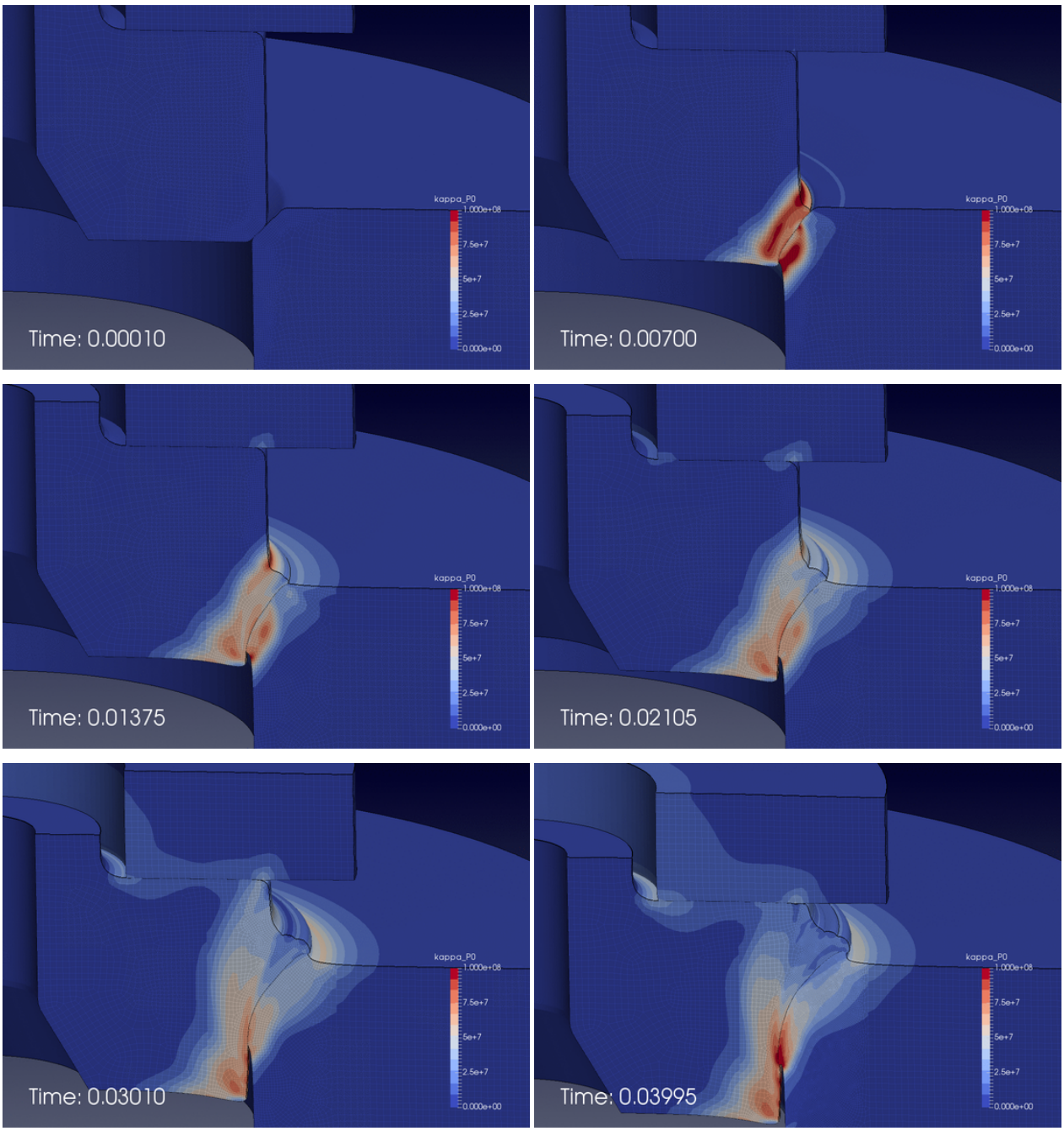

Figure 3.24. Hardening evolution (κ) during a resistance forge weld simulation *(color map range 0 to 100 MPa)*. The sequence is viewed left-to-right, top-to-bottom. Five remeshing/mapping steps were required to resolve the large deformation. Initial work-hardening causes a rise in kappa. Subsequent recrystallization softens the material again until large shear strains lead to local hardening along the bond interface.

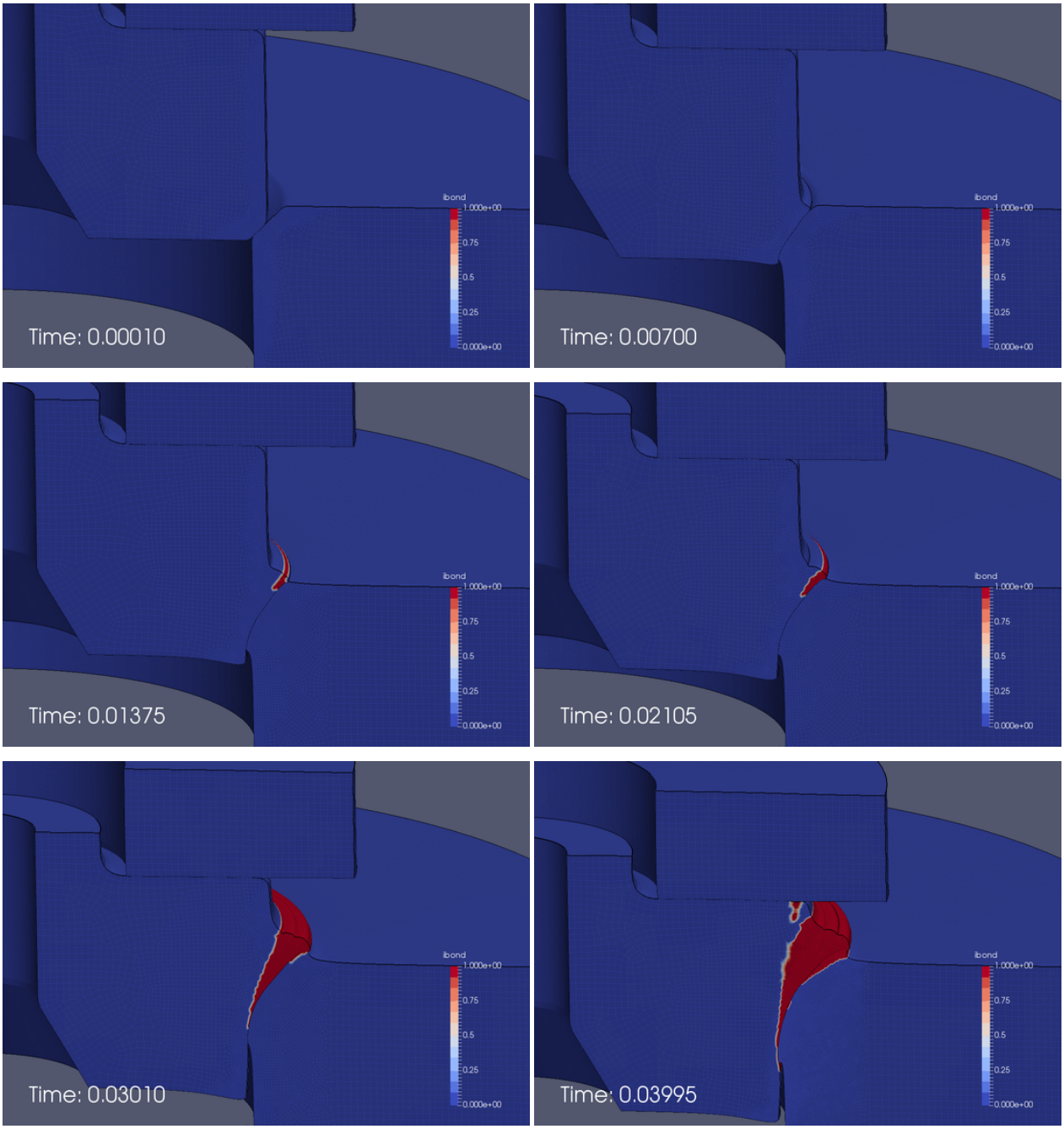

Figure 3.25. Solid state weld evolution during a resistance forge weld simulation *(color map range 0 to 1)*. The sequence is viewed left-to-right, top-to-bottom. Five remeshing/mapping steps were required to resolve the large deformation. Red regions indicate a simulated bond has formed, tieing adjacent surfaces together; blue indicates no bond has formed yet. The initial bond forms during the end of the first current cycle and is almost complete by the end of the second second cycle.

The residual stress state after resistance forge welding is complex and provides a significant and previously unknown insight into the process. Subsequent cooling and material handling processes can have further (significant) implications on the evolution of residual stresses. Figure 3.26 illustrates a cooling process that takes place during the approximately two seconds immediately after the resistance forge weld simulation terminates at 0.040 ms. The cooling process depicted is only intended to be a nominal representation of the additional residual stresses induced by thermal expansion and contraction during cooling stages.

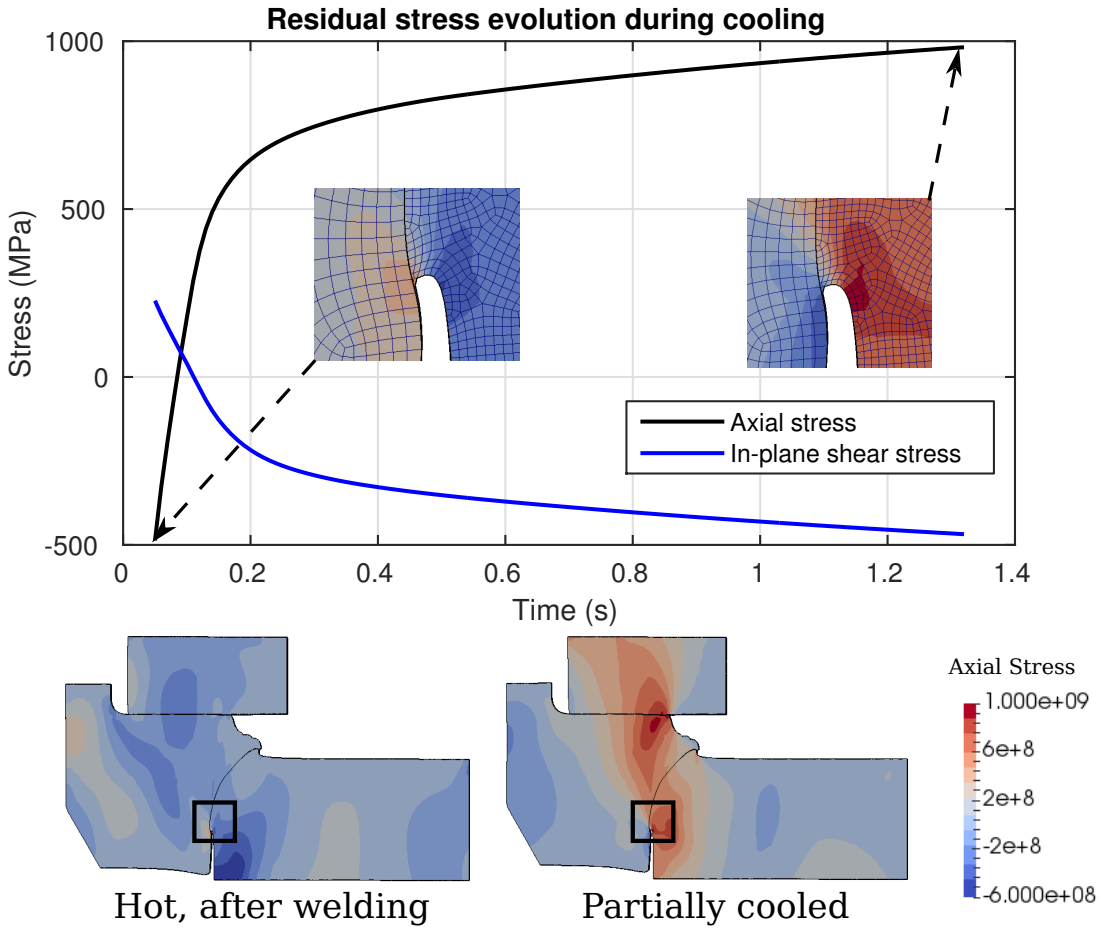

Figure 3.26. Evolution of residual stresses during cooling

The electrode is held fixed and room-temperature (294K) boundary conditions are applied to portions of the base to cause heat transfer. Current flux and applied pressures are removed and the material state after the resistance weld is transferrred to initialize the cooling step. A rapid decrease in maximum temperature from around 1600K to 600K occurs during the simulated time and is accompanied by a dramatic change in the residual stress field. Figure 3.26 depicts axial and in-plane shear stress fields near a stress concentration developed during forging. It is remarkable to note that thermally-induced residual stresses are sufficiently large to cause stress sign reversals;

areas with axial compression stress change to tension stresses, and shear stresses reverse direction. This change happens early on and reflects changes in both boundary conditions that represent changes in fixturing as well as cooling processes due to heat conduction. Effects of radiation and convection can also be included with appropriate knowledge of the environment. It is emphasized that the results of Fig. 3.26 are illustrative only and intended to demonstrate a process capability. Future work using actual geometries and improved process simulation steps can provide additional insight into the residual stress evolution process.

3.3 Representative GTS Problem

For the "exceeds" criteria, the forging, machining, welding process was performed on a more representative GTS problem where a cup and stem was forged (as in previous efforts) and then a resistance forge weld stem was machined away from the forged geometry (see Fig. 3.27). Then, the resistance forge weld was initialized with the residual stresses left over from the forging and machining operations and run with implicit contact.

A snapshot of simulation results comparing the stress fields (σ_{xx}) with and without state initialization is depicted in Fig. 3.28. The simulation snap shot is taken at 6.2 ms. Differences in the stress field contour are attributable to the residual stress state as well as the material internal state. The initial cup and stem forging process as depicted in Fig. 3.27 also introduces work hardening, manifested by increases in the state variable κ. As a result, the stem does not yield as readily as the base. The increased stem hardness causes more localized deformation in the base as can be seen at the lower edge of the stem-to-base interface. As before, the results described herein are intended to demonstrate capability only. Further exploration of specific manufacturing process parameters and simulated outcomes are left for future work.

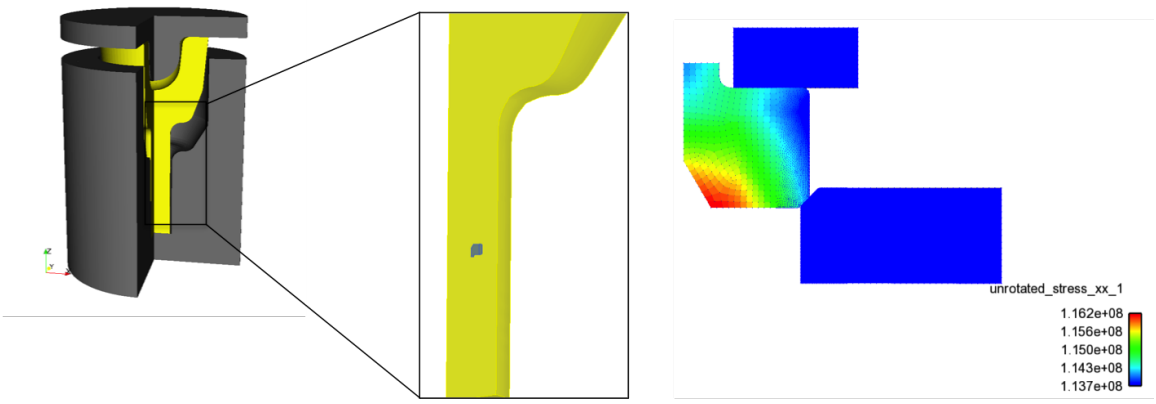

Figure 3.27. Illustration of the forging, machining, welding process for a representative GTS problem. Initial state of residual stress for the RFW setup is shown on the right.

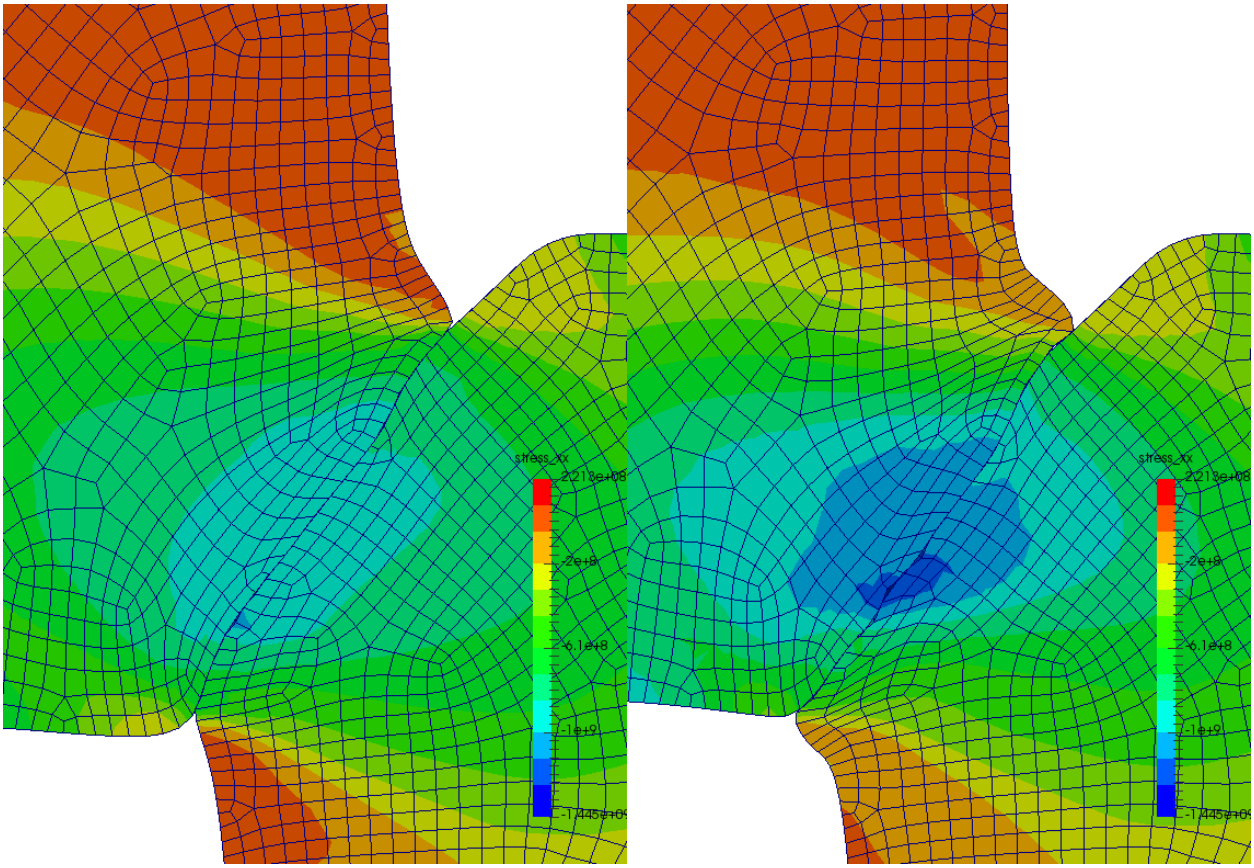

Figure 3.28. Comparison between resistance forge weld simulation without residual stress initialization (left) and with residual stresses due to machining (right). The initialized model (right) includes hardening effects of the manufacturing process due to forging the stem, and results in larger base deformations.

Chapter 4

Characterize and Quantify Uncertainties

The third completion criteria was to "characterize and quantify uncertainties of material parameters and manufacturing process parameters on residual stress" with the criteria success evaluated by performing uncertainty quantification of the forging process effects on residual stress. While the details on this work can be found in our corresponding formal V&V report [1], a summary of the process that was performed is as follows:

- Perform mesh convergence on series of steps (forging, machining and welding)
- Improve robustness of the forging, machining, welding modeling process. Includes improvements to solvers, run time, convergence, machining, contact, etc.
- Identify process and material parameters to study with the GTS customer
- Determine plausible ranges for the parameters
- Perform sensitivity analysis on forging, machining, welding process using Latin Hypercube Sampling (allowing for maximum reuse of analyses)
- Modify ranges on the critical parameters using the sensitivity analysis results (if needed)
- Quantify uncertainties (leveraging sensitivity analysis results) to understand the distribution of stresses and whether any exceed the thresholds
- Summarize and document results

4.1 Mesh Convergence

A mesh convergence study was conducted to ensure the sensitivity analysis and subsequent uncertainty quantification utilized a mesh in which the quantities of interest, or residual stresses in this work, were converged. To assess the convergence a series of meshes were considered with refinement levels of approximately 8, 16, and 24 elements per inch for the coarse, medium and fine meshes, respectively for the forging, and 40, 80, and 120 for the machining and welding. A Richardson's extrapolation was performed to compute the exact solution and assess the numerical

error associated with each mesh. For the forging, we observed less than 10% error on the coarsest level, but only less than 10% error on the finest level of the machining and welding meshes. Unfortunately, the computational run time of the finest mesh was on the order of days, which necessitated some improvements to the robustness of the process.

4.2 Improvements of the Robustness

One of the most critical issues to perform the V&V work presented in [1] was to reduce the overall simulation run time, evident in performing the mesh convergence study. Due to prior issues with implicit contact convergence, the forging process was broken down into a series of explicit analyses, many of which utilized mass scaling to achieve a realistic run time. Unfortunately, even with the mass scaling, the time steps were too small and overall run time too demanding to complete the V&V. Thus, the longest step of the forging process, the quenching, was explored in more detail to understand the bottleneck. Traditionally, the quench process modeling utilized Presto (explicit dynamics) with mass scaling to achieve convergence. However, to improve the performance, this step was converted to Adagio and both the implicit and explicit quasistatic capabilities were performed. After calibration of the solver, we were able to reduce the run time from around 2 days to about 5 hours. Similar performance gains were seen in conversion of the explicit dynamic relaxation after the machining to implicit quasistatics, allowing us to move forwards with the V&V efforts.

4.3 Parameters of Interest and Ranges

To understand possible sources of uncertainty in the residual stresses, it was crucial to examine the process model inputs and assumptions. After meeting with the customer to discuss, the key sources of uncertainty in the forging, machining, welding process were identified as follows:

furnace temperature The temperature of the furnace in which the ingot was heated was not well controlled. Thus, a study was conducted for temperatures ranging from 1500F to 1700F, with 1600F as our nominal value.

initial platen temperature The initial platen temperature was not monitored or controlled. The range was varied from room temperature to 500F.

post forge die chill duration The time in which the hot ingot sat on the die was not controlled. A range of 1s to 30s was studied.

transfer time to quench bath The time between the end of the forging compression and the quenching was variable, and had significant effects on the recrystallization due to thermal softening (i.e. the longer the time, the more softening that would occur).

quench bath temperature The quench bath temperature was not controlled, and several forgings were quenched one after another, which could have significantly raised the temperature. Thus, we studied temperatures in the range of room temperature to 500K.

quenching convection coefficient Assumptions were made in the model on the quenching convection coefficient, as data did not exist. The range was varied between 3000 and 7000, with 5000 as the nominal value.

forging distance The distance the plates were compressed was fairly well controlled, but slightly variable. Differences in final forged heights of the wedge were observed between different lots, thus necessitating a study.

machining location The location of the machined geometry inside the original geometry was fairly well controlled, but the location could have a significant effect on the initial state of stress for subsequent process modeling steps.

welding friction coefficient With no data supplied on the frictional contact, the coefficient was varied in the simulation from 0.1 to 0.5.

displacement rate The applied displacement boundary condition for the simplified resistance weld was not based on experimental data, as it did not exist. Thus, it was varied 10% above and below the nominal value.

room temperature Assumptions were made regarding the temperature of the room in which the welding would be conducted, thus it was varied between 50F and 90F.

flux The flux boundary condition required some assumptions, as an experiment was not conducted for the welding yet at this point in time. A factor of 10% was considered.

4.4 Sensitivity Analysis

The parameters and corresponding ranges given above were used to conduct a formal sensitivity analysis of their effects on the final residual stresses after each of the forging, machining and welding steps. A more detailed description of the sensitivity analysis is given in [1]. In summary, a 12-dimensional parameter space was studied using a Latin Hypercube Sampling (LHS). According to the mesh convergence study, ideally, the forging, machining and welding process models corresponding to the finest mesh would be used during this sensitivity study to ensure a verified solution. However, the high computational costs associated with the most refined models was deemed prohibitive for a significant number of LHS designed simulations. Alternatively, the parameter distributions were each sampled 40 times, creating 40 sets of sampled parameters, and these 40 parameter sets were processed through simulations of the forging/machining/welding processes using both the coarse and medium refinements. Upon completion of these two LHS sensitivity studies, the results corresponding to the two different meshes were assessed, as the same stress metrics upon which mesh convergence was measured were compared for the 40 predictions

resulting from each of the coarse and medium refined models. For each of the forging, machining, and welding processes, a comparison of the coarse and medium mesh refinements yields an approximately linear relationship with a heavier weight favoring neither the medium nor coarse discretizations. This indicates that trends regarding parameter sensitivity are perhaps not mesh dependent and that conclusions of which potential sources of uncertainty are most critical need not be made with the most refined mesh. Therefore, given its low computational cost, 40 additional LHS samples of the coarsely meshed forging/machining/welding model were processed, resulting in 80 total samples of the 12-dimensional parameter space.

From the study, we identified seven sensitive parameters, which can intuitively be justified as critical to the formation of residual stresses throughout the forging/machining/welding process with engineering judgment. First, regarding the forging distance, this parameter outlines the extent to which the metal ingot is compressed. Instinctively, it is understood that greater amounts of compression will result in the formation of higher residual stresses upon conclusion of the forging process. Next, with respect to both the furnace and initial platen temperatures, it is expected that greater temperature differentials between the ingots pre- and post- quench conditions could affect the quantity of the developed stresses. Specifically, it is possible that, with large thermal excursions, the ingot dissipates less heat away and a greater residual stress state forms as a consequence. Next, the quenching convection coefficient governs how quickly heat if transferred from the ingot during the quenching process. Therefore, it is expected that as this coefficient increases, the rate at which heat is dissipated from the ingot changes and the magnitude of the residual stresses formed through the process is reasonably affected. Next, regarding the post forge die chill duration, this parameter defines the duration of time over which the ingot remains in contact with the die prior to introduction into the quench bath. Typically, it is understood that longer pre-quench time durations promote increased recrystallization and softening within the ingot that effectively reduce the existing residual stresses. Next, the machining location represents the site from which the subset mesh is taken from the original forging geometry. Since the post-forging residual stresses are not constant over the ingots volume, it is intuitively understood that the subset mesh will exhibit higher or lower residual stresses according to the location from which they are mapped from the original ingot. Lastly, with regards to the displacement rate, this parameter defines how quickly the welding boundary conditions are applied. While the criticality of this parameter could perhaps be solver dependent, the metallic material being modeled is rate dependent. Therefore, it is possible that increasing the displacement rate will cause the material to demonstrate rate dependent behaviors with a resulting net increase in the residual stresses formed during the welding process.

4.5 Uncertainty Quantification of the Forging

Using the five of the parameters identified from the sensitivity analysis, a preliminary uncertainty quantification was conducted on the forging process with 100 new LHS samples. These 100 samples resulted in 100 parameter sets that were processed through 100 new forging process simulations in order to propagate the input parameters uncertainties through to the predicted response and fully understand what effect these five critical parameters have on the estimated residual stresses. Upon completion of the 100 forging process simulations, the results were examined statistically

in order to characterize the resulting distribution of 100 residual stress predictions. The distribution created by the 100 simulated responses was tested for normality with the adjusted Anderson-Darling statistic, which in this case showed a normal distribution with a mean around 37.0 ksi and standard deviation around 15.6 ksi. More details on the UQ process are given in [1].

4.6 Uncertainty Quantification of Ucup Geometry for Welding

Additionally, for the Ucup geometry discussed in Section 2.6, a uncertainty quantification study was conducted to understand the effects of the geometry used for the welding. Previously, we observed that the forging slice used for the autogenous weld differed from the nominal dimensions. Further investigation revealed uncertainties in the initial ingot geometry, the point at which the forging was annealed and the extent to which the ingot was flattened before the final forging. Thus, we have conducted some preliminary simulations to better understand how such uncertainties affect the microstructural evolution. While the general trends remain the same, the most noticeable difference occurs if the ingot is flattened to a 2" diameter (as opposed to 1" previously used), as displayed in Fig. 4.1.

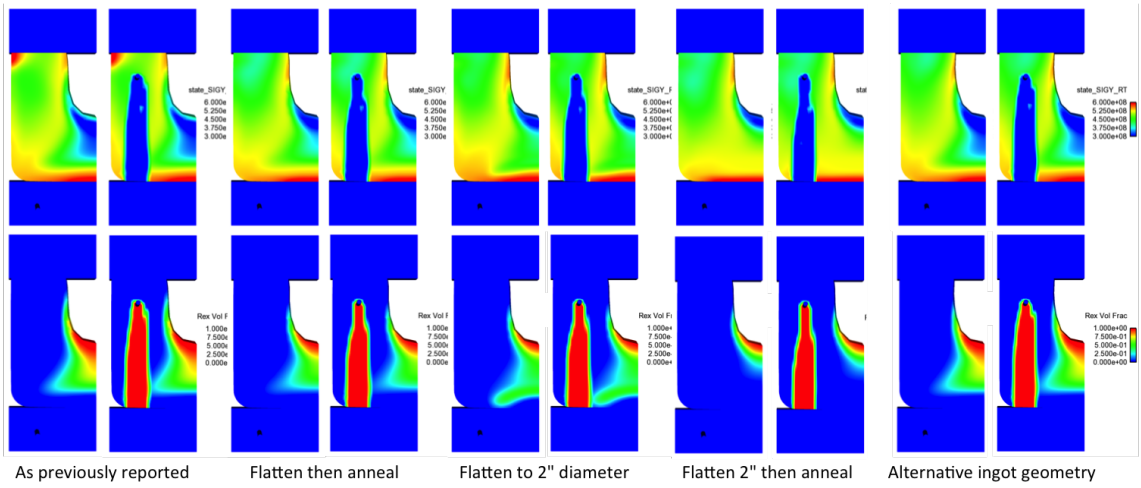

Figure 4.1. Investigation of the uncertainties in the Ucup geometry for the autogenous weld process

This page intentionally left blank.

Chapter 5

Conclusions

5.1 Summary

An enormous amount of effort has gone into achieving highly successful completion of each L2 milestone objective. As a result, modeling and simulation capabilities necessary to support GTS manufacturing and life cycle processes – from "cradle to grave" – have dramatically improved in both capability and robustness. Close collaboration and goal alignment between P&EM and V&V programs was key and should serve as a model for future efforts. Early planning, regular communication, and continued support from SIERRA developers throughout the program duration was critical in extending new capabilities (e.g. state variable mapping) and refining existing capabilities (e.g. implicit contact).

New tools to remesh and map state variables from highly deformed configurations have enabled high-temperature plasticity simulations to progress beyond what has been previously possible. Solution robustness has increased, particularly for large deformation simulations utilizing the BCJ_MEM material model in implicit contact environments. Solution confidence has also increased through V&V efforts that investigated uncertainty and error propagation through a series of manufacturing steps such as forging, machining, and welding. Statistical and mesh convergence analyses quantified errors in the life cycle processes using formal V&V practices, helping to focus future efforts in appropriate areas.

The resistance forge weld problem served as a classical representative example of the GTS cradle-to-grave process. It involves coupled thermal-mechanical-electrical coupling, high-temperature plasticity, resistance welding, and large deformations. Accurately resolving the complex physics required the developments and capabilities outlined in the L2 milestone criteria. It is also a meaningful problem to the GTS customers motivating this work. Hence, having the resistance forge weld as an "acceptance problem" provided a datum to measure progress and to assess, debug, and improve existing capabilities.

Close collaboration with the SIERRA development team played a critical role in the successful completion of this milestone. Early notification of deliverables and code development needs are important in fostering this collaboration to allow for necessary planning and resource allocation. A huge amount of time, resources, and effort was provided by the code team to address bugs, implement new features, and improve algorithms. Much of this effort was documented and tracked through the SIERRA-help ticket system. The resistance forge weld application example, in partic-

ular, would not have been successful without continued support for the implicit contact algorithm and code coupling capabilities.

5.2 Future Research Opportunities

There are still a number of immediate needs necessary to provide efficient, accurate, and robust modeling and simulation to support for the GTS cradle-to-grave life cycle. A few areas encountered during the course of this work are outlined next, and should be considered for future work from both research and code development perspectives.

Thermo-mechanical analysis capability Physical processes, such as the GTS life cycle, requiring simulation are increasingly complex and multi-physics in nature. Modeling assumptions to justify de-coupled thermal or mechanical analyses are no longer considered adequate. Analysts need a thoroughly-vetted multi-physics coupling capability with continued support and development. The current development model with segregated Solid Mechanics and Thermal/Fluid code teams places thermal-mechanical analyses in the void between these teams. This statement is not intended to mean analysts are without support; it is a suggestion for increased and targeted thermo-mechanical efforts in the future given analysis needs of the present.

Implicit contact Implicit contact is a challenging problem. However, contact physics are present in the vast majority of problems for which simulation is requested and so a robust capability and continued, dedicated support is a real need. Explicit contact enforcement may resolve many of the challenges faced by implicit contact algorithms, and may be the right choice for certain problems. But, there are still many reasons and a large class of problems that are best solved with implicit contact. Ironically, the number of analysts attempting to use *implicit* contact in their work may be relatively low despite the ubiquity of contact analysis problems. The explanation for this is simple: analysts need to obtain the best solutions possible within a reasonable time frame. Implicit contact is arguably the best solution option, but a number long-standing problems have deterred analysts from using the capability. Instead, alternative methods using *explicit* contact enforcement are used in combination with non-physical strategies such as mass scaling and time-compression. These non-physical strategies involve creating simulation objects with densities greater than any substance known to man or compressing the real time scale of the relevant physics processes into milliseconds, or other circuitous approaches that often deteriorate solution quality but are opaque to the customer requesting the analysis. A few long-standing problems with implicit contact enforcement are (i) difficulty in obtaining a converged solution, (ii) unstable contact enforcement when used in combination with non-contact boundary conditions, (iii) large number of solver iterations leading to long solution times (often counter-acting benefits of implicit solutions), (iv) non-physical pressure oscillations along contact interfaces, and (v) a need for better friction interface models. Some of these problems are algorithmic in nature, others are more research-oriented. Focused effort and dedicated resources will be required to bring improvements to this challenging

area from ASC P&EM, V&V, and IC program perspectives if we are to keep up with the ever-increasing demands of modeling and simulation.

Internal state variable mapping Large deformation processes simulated within a Lagrangian framework inevitably require remeshing and mapping, and the process used for this work has been quite successful (thanks to the work of many individual and team efforts). This process was driven by external executables (i.e., not in SIERRA). Many of the required features for projecting variables, interpolating to a new mesh, and initializing states for a subsequent simulation have been added by the SIERRA/SM code team. These capabilities have been tested for solid mechanics, but have seen very limited testing on the therm-mechanical front. Incidentally, the authors would suggest that a large majority of scenarios in which re-meshing may be required involve heat transfer or generation either due to external sources (e.g., furnace heating), or due to plastic dissipation. The fundamental components needed to streamline the remeshing and internal state variable mapping process are in place. But, additional effort is required to streamline and test the procedure on multiple relevant acceptance problems in a thermo-mechanical coupled code such as Arpeggio. The user interface for re-meshing and mapping also needs to be simplified to the point that analysts are not required to have expert knowledge of the fundamental algorithms driving the process. Analysts need to exercise the capability on acceptance problems and provide feedback to the code team.

CUBIT coupling with SIERRA Successful process simulations utilizing re-meshing and mapping will require a better synergy between meshing tools (CUBIT) and simulation tools (SIERRA). The ability to call CUBIT from within SIERRA would enable a more stream-lined state variable mapping capability. Improved tetrahedral meshing could automate the remeshing process. Lastly, improved tools for handling mesh-based geometry will be required in the future.

Material modeling Material models are in some ways one of the most fundamental physics that can be specified in a solid mechanics model. SIERRA/SM contains a large and diverse library of material models. Experimental tests continue to elucidate physical phenomena that cannot be accounted for with the current material modeling capabilities. The initial implementation of a dynamic strain aging model is one example. Anisotropy and location-dependent properties observed during additive manufacturing process may also motivate future material modeling needs. Because material models are so fundamental to obtaining a finite-element solution there are other less-exciting material modeling needs. The stability and robustness of material models is often manifested to the analyst as a failed conjugate-gradient solve, failed implicit contact iterations, or non-physical solution. Many such solution issues were resolved over the course of this work, particularly in the resistance forge weld application example, because of the author's intimate knowledge of the material model algorithm. However, this is an exception rather than the rule since many analysts will not have the access, knowledge, or time to remedy these problems. Continued funding to support fundamental and often unseen algorithmic improvements is just as necessary as efforts introducing new and exciting capabilities.

Element technology Perhaps even more fundamental than material models is the underlying finite-element technology used in the discretization process. SIERRA/SM has supported a number of element formulations including various tetrahedral elements and formulations, the uniform gradient hexahedral and the selective-deviatoric hexahedral, among others. However, the as the default and often fastest element the uniform gradient hexahedral has received the most work in terms of performance. In many cases, however, a different element formulation such as the selective-deviatoric hexahedral element is more appropriate. The selective-deviatoric element may benefit from performance enhancement studies. In addition, development of an element that linearly interpolates the pressure field, in addition to displacements, could provide a better solution and convergence behavior in some plasticity, contact, and incompressible problems. Additional research and development is necessary to scope out the potential benefits of such an element. Furthermore, tetrahedral elements such as the "composite tet" may be even more appropriate in some situations where large-deformation and re-meshing is required. With more complex geometries that could be involved in abnormal environment simulations, for example, a robust tetrahedral remeshing and solution capability becomes critical. Re-meshing strategies with hexahedral elements are simply not possible with current tools due to the lack of a geometrical representation after a finite-element solution. Mesh-based geometry generated by meshing tools such as CUBIT is not amenable to decomposition, and the time required to obtain a high-quality hexahedral mesh is prohibitive.

Periodic boundary conditions Many problems of interest are axisymmetric in nature. This axisymmetry can be exploited to obtain significantly reduced run-times, promote formal V&V studies, and obtain higher-quality solutions as a result of increased discretization capability. However, there is currently only a very limited periodic boundary condition in SIERRA/SM. Analysts typically enforce a symmetry constraint to prevent out-of-plane motion on wedge boundaries, but this does not enforce a truly-periodic solution. Additional development of the existing periodic boundary condition is necessary to fully-realize all of the benefits previously identified.

5.3 Lessons Learned

This section presents several "lessons learned" during the course of this work. These lessons were learned primarily in studying the resistance forge weld and several other representative example problems, but may be more broadly applicable in the scope of large deformation contact problems.

- Remeshing and mapping capabilities have significantly improved, but all state variables and Gauss points should still be carefully scrutinized after a solution.
  - Ensure all Gauss points have appropriate state variables and field data, and that it is being initialized correctly in subsequent simulations.
  - State variables may not transfer correctly, particularly in thermo-mechanical coupled analyses using Arpeggio and `Solution Control` blocks. Some issues were observed particularly with the `Interpolate Volume Elements` transfer type.

  - Loose tolerances on L2 projections may introduce non-physical artifacts along processor boundaries. It is critical that the L2 projection is solved very accurately to prevent the propagation of algorithmic artifacts during subsequent mesh decompositions. Default tolerances were tightened as a result of observations made during this work, but analysts should be aware of the possibility in future analyses.
  - The `Strongly Objective` formulation for the selective deviatoric hexahedral does not currently work correctly with re-meshing and mapping algorithms. The initial deformation gradient state is not tracked through remeshing and mapping steps, so attempting to use it beyond the initial simulation is futile. The recommended approach is to use the default `Midpoint Increment` formulation.

- The nodal field dependent friction models used to simulate welding (via the `ibond` state variable) transition from tied contact to Coulomb friction. However, the contact behavior is not well defined when one side of the interface indicates bonding should occur (`contact_transition=1`) while the other side indicates the opposite (`contact_transition=0`). For the default non-symmetric `DASH` contact algorithm, SIERRA/SM simply enforces the tied constraint only when the slave surface transitions. However, the slave surface is determined somewhat ambiguously when opposite sides of the interface have similar mesh size. Furthermore, if symmetric contact is used it is possible that multiple, conflicting, friction models (e.g., tied and friction) could potentially be enforced simultaneously. One possible solution, though not implemented during the course of this work, is to project contact transition from one side of the contact interface to the other so that elements in the same neighborhood have identical contact transition values.

- Implicit wedge boundary conditions are *essential* for accurate and robust solutions for a large class of problems, including the resistance forge weld. Axisymmetric boundary conditions reduce the solution space the contact algorithm must search to a subset of more desirable configurations. Non-axisymmetric solutions quickly lead to configurations which cannot be sweep-meshed in CUBIT. Axisymmetry was painstakingly enforced with a custom script that generated multi-point constraints, given two node sets. A critical setting required in the Adagio region is the `resolve multiple mpcs = enforce all`. This setting was added to ensure all manual MPCs were enforced.

- The DASH option `Developer Command: Dice Angle` can mean success or failure of a contact problem. A good default value and an understanding of its effects on the contact algorithm are needed. This option was introduced by SIERRA developers in support of this milestone.

- `Initial Overlap Removal` is recommended for remeshing and mapping processes. Remeshing can (and often will) introduce overlapping facets that were not present in the deformed mesh. This is particularly the case for problems with contact. The equilibrium solve will introduce nonphysical stresses when trying to remove this overlap, and may simply not converge.

- Sometimes one can manually recover from a failed contact solution by adjusting the contact interaction type from symmetric to non-symmetric. A better contact algorithm or adaptive solution procedure could accommodate this and resolve the issue for the user “on-the-fly”.

- Use the SIERRA version-of-the-day when possible; developers are constantly increasing speed and solution robustness. Contact problems, in particular, have seen major improvements between 4.34 and 4.40 releases.
- If convergence fails, increase or decrease the number of processors being used by 1 or 2. Often, the modified decomposition and resulting parallel solution strategy is sufficiently different that the solution can proceed. These parallel inconsistencies are difficult to resolve, but should also be reported to SIERRA developers for additional troubleshooting.

References

[1] L.L. Beghini, S.M. Nelson, and K.L. Manktelow. V&V of residual stress for GTS. *Sandia Report (SAND2016-xxxx)*, 2016. Unclassified/Unlimited Release.

[2] RH Bogaard, PD Desai, HH Li, and CY Ho. Thermophysical properties of stainless steels. *Thermochimica Acta*, 218:373–393, 1993.

[3] A.A. Brown, L.A. Deibler, L.L. Beghini, T.D. Kostka, and B.R. Antoun. Process modeling and experiments for forging and welding. In *SEM*, 2015.

[4] A.A. Brown, T.D. Kostka, B.R. Antoun, M.L. Chiesa, D.J. Bammann, S.A. Pitts, S.B. Margolis, D. O'Connor, and N.Y.C. Yang. Validation of thermal-mechanical modeling of stainless steel forgings. In *Proc. XI International Conference on Computational Plasticity Fundamentals and Applications*, 2011.

[5] Arthur A. Brown and Douglas J. Bammann. Validation of a model for static and dynamic recrystallization in metals. *International Journal of Plasticity*, 32:17–35, 2012.

[6] GM Eggert and PR Dawson. Assessment of a thermoviscoplastic model of upset welding by comparison to experiment. *International journal of mechanical sciences*, 28(9):563–589, 1986.

[7] B. D. Formisano. Evaluation of the effects of current impulses in a side bonding resistance forge weld (U). *Sandia Report (SAND84-8003)*, June 1984. Unclassified/Unlimited Release.

[8] James W. Foulk and et. Al Veilleux, Michael G. Capturing finite deformations in metallic structures through robust remeshing and mapping of internal state variables. *In preparation*.

[9] S. B. Johnson. A characterization of the side bonding, upset plug welding process (U). *Sandia Report (SAND89-8242)*, August 1989. Unclassified/Unlimited Release.

[10] LP Kubin and Yu Estrin. Evolution of dislocation densities and the critical conditions for the portevin-le chatelier effect. *Acta metallurgica et materialia*, 38(5):697–708, 1990.

[11] E. B. Marin, D. J. Bammann, Regueiro R. A., and G. C. Johnson. On the formulation, parameter identification and numerical integration of the EMMI model: Plasticity and isotropic damage. *Sandia Report (SAND2006-0200)*, January 2006. Unclassified/Unlimited Release.

[12] A. Mota, W. Sun, J.T. Ostien, J.W. Foulk, and K.N. Long. Lie-group interpolation and variational recovery for internal variables. *Computational Mechanics*, 52(6):1281–1299, 2013.

[13] T.B. Reynolds, A.A. Brown, L.L. Beghini, T.D. Kostka, and C.W. San Marchi. Development of Residual Stress Simulation and Experimental Measurement Tools for Use in Lifetime Assessment of Stainless Steel Pressures Vessels. In *Proceedings of the 14th International Conference on Nuclear Engineering*, 2015.

[14] SIERRA Solid Mechanics Team. Sierra/solidmechanics 4.40 user's guide. Technical report (UUR) SAND2016-2707, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550, March 2013.

[15] G.W. Wellman. MAPVAR - A Computer Program to Transfer Solution Data Between Finite Element Meshes. Technical Report SAND99-0466, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550, 1999.

[16] W.S. Winters, A. A. Brown, and A.R. Ortega. Progress report for the ASCI-AD resistance weld process modeling project AD2003-15. *Sandia Report (SAND2005-3000)*, May 2005. Unclassified/Unlimited Release.

DISTRIBUTION:

1	MS	9042	Brown, Arthur, 08259
1	MS	9042	Manktelow, Kevin, 08259
1	MS	9042	Beghini, Lauren, 08259
1	MS	9042	Nelson, Stacy, 08259
1	MS	9042	de Frias, Gabriel, 08259
1	MS	9042	Veilleux, Michael, 08259
1	MS	9159	Hough, Patricia, 08954
1	MS	9152	Lefantzi, Sophia, 08954
1	MS	9042	Nilsen, Curt, 08250
1	MS	0825	Payne, Jeffrey, 01513
1	MS	9042	Peterson, Scott, 08343
1	MS	9042	Foulk, James, 08343
1	MS	9042	Ostien, Jake, 08343
1	MS	0845	Pierson, Kendall, 01542
1	MS	0840	Scherzinger, William, 01554
1	MS	0899	Technical Library, 8944 (electronic copy)

This page intentionally left blank.


Sandia National Laboratories