

**IMPROVEMENTS AND VERIFICATION OF AN ACCELERATED TECHNIQUE
FOR SIMULATIONS OF CYCLICALLY LOADED STRUCTURES**

by

Songwei Zheng

A thesis submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering

Winter 2013

© 2013 Songwei Zheng
All Rights Reserved

**IMPROVEMENTS AND VERIFICATION OF AN ACCELERATED TECHNIQUE
FOR SIMULATIONS OF CYCLICALLY LOADED STRUCTURES**

by

Songwei Zheng

Approved: _____
Anette M. Karlsson, Ph.D.
Professor in charge of thesis on behalf of the Advisory Committee

Approved: _____
Suresh G. Advani, Ph.D.
Chair of the Department of Mechanical Engineering

Approved: _____
Babatunde A. Ogunnaike, Ph.D.
Interim Dean of the College of Engineering

Approved: _____
Charles G. Riordan, Ph.D.
Vice Provost for Graduate and Professional Education

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor, Dr. Anette M. Karlsson, for her great support and extraordinary guidance during the Master program. She inspires me to think creatively and critically, teaches me skills in technical writing. Her advice and help have always been appreciated.

I am grateful to have Dr. Michael H. Santare and Dr. Joshua L. Hertz serving on my thesis committee. Thanks for their time to evaluate my research.

Special thanks to Dr. Dan Cojocaru. His contributions on the numerical scheme of the “cycle jump technique” serve as the foundation for this work.

I want to thank all my lab-mates: Dr. Liang Cheng, Dr. Mercedes Hernandez, Dr. Zongwen Lu, Dr. Narinder Singh, Dr. Hongzhou Li, Yun Wang, Parag Nittur, Jayanta Kumar, Dattatraya Sahane, Melissa Lugo, Guoliang Ding, Jun Qian. It is great pleasure to work, and have discussions with them.

I also want to thank all the staff in the Mechanical Engineering Department, Lisa Katzmire, Letitia Toto and Ann Connor. They are so nice and helpful.

Special thanks to my father Zhiqiang Zheng, my mother Peihua Zeng, and my wife Queming Qiu, for their constant support.

TABLE OF CONTENTS

LIST OF TABLES	vi
LIST OF FIGURES	vii
ABSTRACT	x

Chapter

1	BACKGROUND	1
1.1	Introduction	1
1.2	Literature Review of Accelerated Numerical Simulation	2
1.3	Existing Modeling Scheme.....	5
1.4	Thermal Barrier Coatings	10
1.5	Experiments	14
1.5.1	Experimental Goal	14
1.5.2	Methodology.....	14
1.5.3	Experiments.....	16
1.5.3.1	Specimens	17
1.5.3.2	Parameter settings and experimental results.....	17
1.5.3.3	Future experiments	19
2	EXTENDING THE CYCLE JUMP TECHNIQUE	20
2.1	Introduction of A Polynomial Extrapolation Scheme	21
2.1.1	Theoretical Framework	21
2.1.2	Numerical Verification.....	25
2.2	Time- and Temperature-dependent Material Behaviors.....	28
2.2.1	Time- and Temperature-dependent TGO Growth	28
2.2.2	Temperature-dependent Material Properties	31
2.2.3	Simulation Results	32
2.2.3.1	Influence of the control parameter	32
2.2.3.2	Comparison between linear and polynomial extrapolation	36

2.2.3.3	Control of maximum jump length	38
2.3	Time-dependent Properties: Creep	40
2.3.1	Creep.....	40
2.3.2	Incorporation of Creep in the Cycle Jump Technique.....	43
2.3.3	Simulation Results for Creep Behavior with the Cycle Jump Technique	43
3	VERIFICATION OF THE CYCLE JUMP TECHNIQUE	45
3.1	Simulation Objectives	45
3.2	Finite Element Model	45
3.2.1	Geometry and Boundary Conditions	45
3.2.2	Thermal Cycling.....	47
3.2.3	Material Behaviors	48
3.3	Parametric Study with Time-independent Plasticity	49
3.4	Reference Simulation Results and Comparisons.....	52
3.5	Verification of the Cycle Jump Technique.....	57
4	SUMMARY AND FUTURE CONSIDERATIONS.....	63
4.1	Summary.....	63
4.2	Future Considerations.....	65
	REFERENCES	67
Appendix		
A	UEXPAN.....	70
B	UMAT	72

LIST OF TABLES

Table 1.1	Experimental parameter settings	17
Table 2.1	Simplified material properties	25
Table 2.2	Parameters of oxidation thickness growth.....	29
Table 2.3	Elastic modulus, Poison's ratio, and thermal expansion coefficient	31
Table 2.4	Temperature-dependent plasticity	31
Table 2.5	Savings of computation with cycle jump technique.....	33
Table 2.6	Relative errors of final radial displacement for selected for selected q_y	33
Table 2.7	Comparison of cycle jump w/o and w/ maximum jump control	38
Table 2.8	Parameters, A and n , in Eq. 2.13	41
Table 3.1	Time-independent plasticity	53
Table 3.2	Deformed data for the HL and LS experiments	54
Table 3.3	Savings of computation with the cycle jump technique.....	59

LIST OF FIGURES

Figure 1.1	Evolution of a state variable at a selected point of a structure subjected to cyclic loading, simulation with the cycle jump technique compared to the cycle-by-cycle reference case.....	6
Figure 1.2	A schematic of a state variable evolution for a structure subjected to cyclic loading (Adopted from Cojocaru and Karlsson [1]).....	7
Figure 1.3	A schematic of the implementation of cycle jumps (Adopted from Cojocaru and Karlsson [1])	9
Figure 1.4	Schematic of thermal barrier coating	10
Figure 1.5	Schematic of oxidation mechanism at high temperature (Adopted from Ref. [6])	12
Figure 1.6	Morphological instability of FeCrAlY due to high temperature oxidation (courtesy Dzodzovic and Bartsch, the German Aerospace Center)	12
Figure 1.7	Geometry of the specimen.....	16
Figure 1.8	Cross section of the grooves.....	16
Figure 1.9	Morphological change from the HL experiment.....	18
Figure 1.10	Morphological change from the LS experiment after 200 cycles	18
Figure 2.1	Evolution of a state variable and its discrete slope within a structure subjected to cyclic loading	22
Figure 2.2	Schematic of polynomial extrapolation for state variables within a structure subjected to cyclic loading	23
Figure 2.3	Basic finite element model of a sample structure.....	25
Figure 2.4	Radial displacement evolution of TGO inner surface, (A) simulation with cycle jump utilizing ‘linear extrapolation’ compared with the reference simulation; (B) simulation with cycle jump utilizing ‘polynomial extrapolation’ compared with the reference case.	27

Figure 2.5	TGO thickness evolution at high temperature, adjusting for the initial thickness	30
Figure 2.6	Radial displacement evolution of TGO inner surface, simulation results with selected control parameters compared with the reference simulation, (A) cycle jump utilizing ‘linear extrapolation’; (B) cycle jump utilizing ‘polynomial extrapolation’.	35
Figure 2.7	Radial displacement evolution of TGO inner surface, $q_y = 0.5$, simulation with cycle jump utilizing different extrapolation scheme compared with the reference simulation, (A) cycle jump with linear extrapolation, (B) cycle jump with polynomial extrapolation.	37
Figure 2.8	Radial displacement evolution of TGO inner surface, $q_y = 0.5$, cycle jump utilizing polynomial extrapolation and a control of maximum jump length = 30 cycles, compared with the cycle-by-cycle reference simulation.	39
Figure 2.9	Schematic of the model for a relaxation test	42
Figure 2.10	Relaxation response in the bond coat for various creep laws, $T=1000$ °C	42
Figure 2.11	Radial displacement evolution of TGO inner surface in creep model, see Figure 2.3, simulation with cycle jump utilizing ‘polynomial extrapolation’ compared with the cycle by cycle reference simulation.	44
Figure 3.1	(A) Schematic of model domain; (B) SEM image of one groove with TGO.....	46
Figure 3.2	Schematic of geometry and boundary conditions	46
Figure 3.3	Schematic of thermal cycles in the experiments and simulations	48
Figure 3.4	Initial and Deformed shape after 180 cycles with two selected T_H	50
Figure 3.5	Deformation for two selected ratios, $\beta = \varepsilon_g/\varepsilon_t$, after 180 cycles with $T_H = 1087$ °C	50
Figure 3.6	Deformation with two different yield strength for FeCrAlY with $T_H = 1087$ °C	51
Figure 3.7	Deformation with two selected yield strength for TGO after 180 cycles with $T_H = 1087$ °C.....	51

Figure 3.8	Morphological change from simulations compared to the experimental results, and contours of plastic/creep strain, for the HL experiment: (a) SEM from the HL experiment; (b) Simulation results using time-independent plasticity; (c) Simulation results using creep properties	55
Figure 3.9	Morphological change from simulations compared to the experimental results, and contours of plastic/creep strain, for the LS experiment: (a) SEM from the LS experiment; (b) Simulation results using time-independent plasticity; (c) Simulation results using creep properties	56
Figure 3.10	Nodes for jump control in simulation with the cycle jump technique, the largest vertical displacement occurs at Node 256 as shown.....	58
Figure 3.11	Contours of vertical displacement, tangential stress and equivalent plastic strain after 200 cycles, cycle jump with $q_y = 3.0$ compared to the cycle-by-cycle reference simulation.....	61
Figure 3.12	Evolution of vertical displacement at Node 256 (see Fig 3.10) in the model for 200 thermal cycles, cycle jump with $q_y = 3.0$ compared with the cycle-by-cycle reference simulation.....	62

ABSTRACT

In engineering applications, structures are commonly subjected to cyclic loadings. This may lead to fatigue and unexpected failures. To prevent the life-time limiting failures, understanding of the failure evolution in these structures during use is very important. For failure prediction due to cyclic loading, finite element analysis (FEA) can be used to simulate and establish the stress and strain distribution as a function of time. However, every single cycle of simulation consists of many computational increments and iterations. The whole process of structural evolution consists of a large number of cycles. Thus, for structures subjected to cyclic loadings, it is extremely time-consuming and inefficient to simulate the whole process of structural evolution. In fact, in most cases, the computational time required is typically prohibiting a complete analysis.

The goal with this work is to improve upon an existing numerical scheme that aims to, in combination with simple testing, predict the life-time of structures subjected to cyclic loading. The numerical scheme is the “cycle-jump technique” developed previously. The fundamental idea of the cycle-jump technique is that there is no need to calculate each individual cycle in a cyclically loaded structure. Utilizing an extrapolation scheme for extrapolating the overall behavior to “jump” over some cycles, the cycle-jump technique may predict the overall structural behaviors with much higher time efficiency.

In this work, the “cycle jump technique” will be modified and improved to eventually be used for simulating realistic designs. To overcome the limitation of the

existing extrapolation scheme, an alternative more general extrapolation scheme is proposed. The numerical code is also improved for applications in simulating time-dependent material behaviors, such as time-dependent oxidation evolution and creep in thermal barrier coatings subjected to high temperature thermal cycles.

To verify that the cycle jump technique can capture real-life experimental results and to demonstrate the power of the method, experimental results, the cycle-by-cycle reference simulation, and the simulation with cycle jump must be compared. Preliminary experiments guided by our preliminary simulations were performed by our collaborators.

Preliminary experimental results are used to compare with the simulation results. Even though the experimental data is limited, it appears as our numerical model can predict the evolution of the test samples, and incorporated the cycle-jump technique will improve the computational efficiency.

Chapter 1

BACKGROUND

1.1 Introduction

It is very common to see failures of structures subjected to cyclic loadings. Investigations of the failure mechanism reveal that they are often influenced by a slow evolution of material and structural properties [1]. For prediction of failures due to cyclic loadings, establishing the state of stress and/or strain is very useful, and even necessary. However, many challenges are involved in the investigation of how the stress and strain in the structures respond as the material or structural properties evolve with time in combination with cyclic loading. Finite element analysis (FEA) [2-4] is a very useful tool that can be used to simulate and establish the stress and strain distribution as a function of time. However, the whole process of structural evolution involves a large number of cycles, and every single cycle of simulation may consist of numerous computational increments and iterations. Thus, for structures subjected to cyclic loadings, it may cost significant computation time and is extremely inefficient to simulate the whole process of structural evolution.

The overall goal with this work is to develop a numerical scheme that can, in combination with simple testing, predict the life-time of structures subjected to cyclic loading. This work is focused on improving the numerical efficiency of simulations with the goal of effectively investigating the behavior of the structure under cyclic loadings. To achieve this goal, we will utilize a numerical scheme and verify the validity by simulating experimentally observed behavior of a material structure. The

numerical tool is the “cycle-jump technique” [1]. The fundamental idea behind the cycle jump technique is that it is too cumbersome and timely inefficient to calculate each individual cycle in cyclically loaded structures. Consequently, with an extrapolation scheme for extrapolating the overall behavior to “jump” over some cycles, the cycle-jump technique may predict the overall structural behavior while significantly saving computation.

Both experimental and numerical efforts are needed in this project. This thesis is organized in the following manner. A review of pertinent literature considering accelerated numerical simulations is given in section 1.2. The extrapolation scheme serving as the basis of the cycle-jump technique is described in section 1.3. Since we are particularly interested in thermal barrier coatings (TBCs), which inspired this work, a short description of TBCs is given in section 1.4. The experimental work by our collaborators in section 1.5 provides realistic material data as input for our numerical model and will serve as reference for the verification of the cycle jump technique. The improvements of cycle-jump technique made to adopt the technique to high temperature materials are discussed in Chapter 2. The reference simulations, which capture the experimental observation are summarized, and the verification of the cycle jump simulating the preliminary experiments are given in Chapter 3.

1.2 Literature Review of Accelerated Numerical Simulation

Only a few investigators have considered accelerated numerical simulations for cyclically loaded structures. A short review will be given in this section.

In the 1980’s, Ladeveze proposed a separated representation of the time and space coordinates [5-6] for efficient solutions of complex non-linear thermo-mechanical problems. Based on this approximation, Ladeveze and co-workers [7-9]

introduced the “Large Time Increments Method”. In this method, the “general” equations for the visco-plastic problems are separated into two groups: (i) “local” equations dealing with the non-linear part of the “general” equations, and (ii) “global” equations dealing with the linear part of the “general” equations. The local equations are solved for every single cycle while the global equations are solved over the total historical loading time. The combined results from the two groups of equations give the final solutions. This idea is very interesting, however, it does not lend itself to be incorporated into commercial finite element software.

In an alternative method for cycle jump developed by Fish et al. [10-11], the time is decomposed into two temporal scales. One is micro-chronological (fast time) scale which represents the cyclic response of the structure, while the other is macro-chronological (slow time) scale which corresponds to the overall trend. The responses of the state variables can be obtained, from combination of the responses in the two time scales. Theoretically, this method works, however, it is based on damage theory. Even though Damage theories can be powerful tools to predict failures, they are built on an assumed pre-knowledge of the failure scenario. Moreover, the complicated formulas and cumbersome iterative calculations raise big challenges for its implementation into commercial finite element software.

In a method by Kiewal and co-workers [12], the internal variables (including displacements, stresses and strains) are extrapolated over a certain number of cycles so as to save computations for a model subjected to cyclic loadings. For each material point in the model, a spline function is obtained based on the evolution equations of the internal variables. The variables are then extrapolated using the obtained spline

functions with an error control function which determines the extrapolation range. A limitation is that this method is customized for visco-plastic model.

Based on the principal method of the extrapolation of internal variable by Kiewal [12], as described above, Wang et al. [13] utilized a linear shape function for extrapolation and made an improvement for the plausibility check of the extrapolation. For the check, the internal variables are extrapolated backwards based on two cycles of FEA after the last extrapolation. Comparing to the backward extrapolated state, whether the last extrapolation is successful or not is determined based on the specified criteria for the relative errors. If the extrapolation fails to fulfill the criteria, the previous extrapolation will be withdrawn. Although they implemented the extrapolation scheme on ABAQUS program, it is also customized for a damage model.

To predict the fatigue damage in structures subjected to cyclic loadings, Bogard et al. [14] developed an accelerated scheme where a cycle jump algorithm is incorporated. This method is similar to the cycle-jump technique described in section 1.3 (to be used in this work). The internal variables are described as functions of time, and expressed in Taylor series up to second order. The jump length are then determined from a number of cycles. At the fatigue damage stage, the damage theory is incorporated and the jump length is then determined by the damage parameter. Thus this method is also mainly based on the damage theory, and is customized for the specific application with specific material properties.

As discussed, all the methods previously developed have some limitations: some of the ideas are hard to be incorporated into commercial finite element software, others could only be used by a small group of structures. A general method to improve

the computational efficiency and provide insight into failure evolution is needed. This is addressed by the method discussed in section 1.3. This method introduced by Cojocaru and Karlsson is a general routine and can be applied on any structure with slowly changing boundary conditions or evolving material properties [1]. For example, it has been customized for application in accelerated fatigue crack growth simulation by Moslemian, Karlsson and Berggreen [15].

1.3 Existing Modeling Scheme

In this section, we will review the ‘cycle-jump technique’ [1], the numerical scheme which serves as the foundation for this thesis.

As already discussed, for structures subjected to cyclic loadings, it can be time-consuming to simulate the structure under a single load cycle; to model multiple cycles can become very lengthy and inefficient. Thus, the “cycle jump” technique is a very useful tool to save computational time. In this work, we will extend a technique previously developed by Cojocaru and Karlsson [1]. We will briefly review this existing algorithm here. In the cycle jump technique, not all individual cycles need to be simulated. By establishing a general evolution of the structure and extrapolating the general evolution, the simulation could skip, or “jump”, over some cycles. By making those jumps, computational efforts can be saved significantly so as to improve the simulation efficiency.

A state variable can be any meaningful measurable quantity in the finite element simulation, for instance, a stress, strain or displacement component at a point in the structure. A typical example of how a state variable evolves as a function of time is shown in Figure 1.1. We can see a global and local evolution where the local change is a cyclic variation during one cycle and the global change is the general trend

that can be extracted over many cycles. In order to reduce computational time by making jumps, the cycle jump technique work as follows [1]:

1. Initial simulation of several cycles with finite element analysis (FEA) to establish the global trend, i.e., global evolution function $y(t)$ for the structural variables;
2. Extrapolation of the state variables by the global evolution functions for the cycle jump;
3. Initialization of the extrapolated state for a new finite element analysis after the jump;

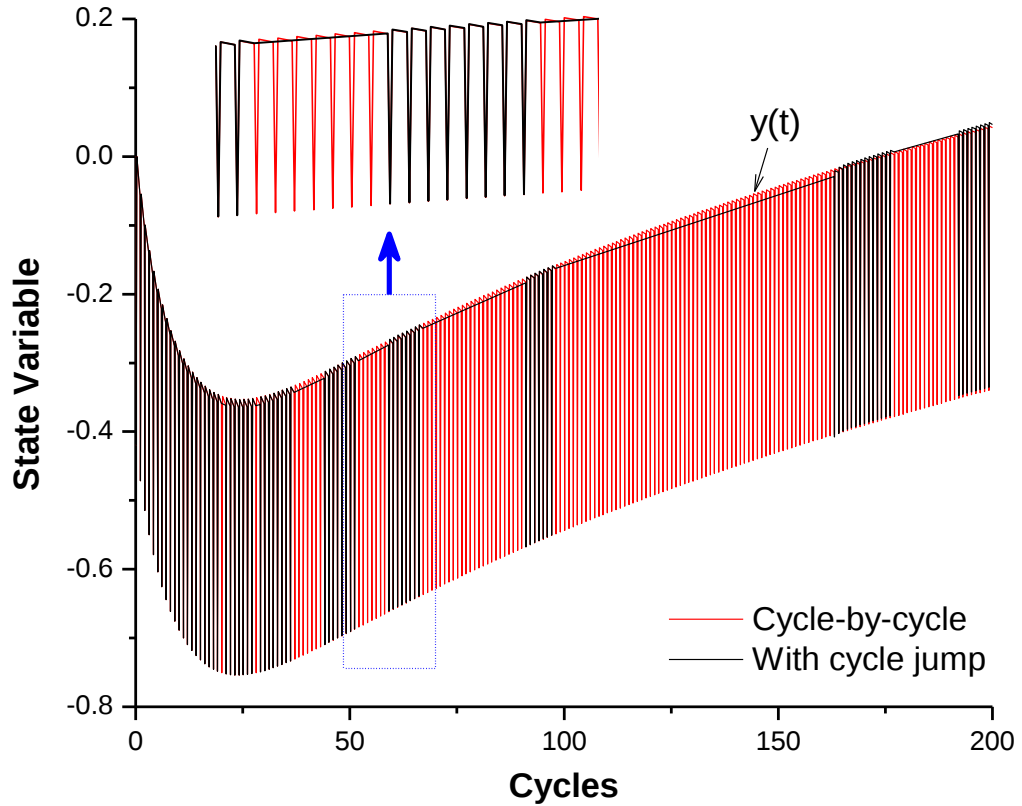


Figure 1.1 Evolution of a state variable at a selected point of a structure subjected to cyclic loading, simulation with the cycle jump technique compared to the cycle-by-cycle reference case.

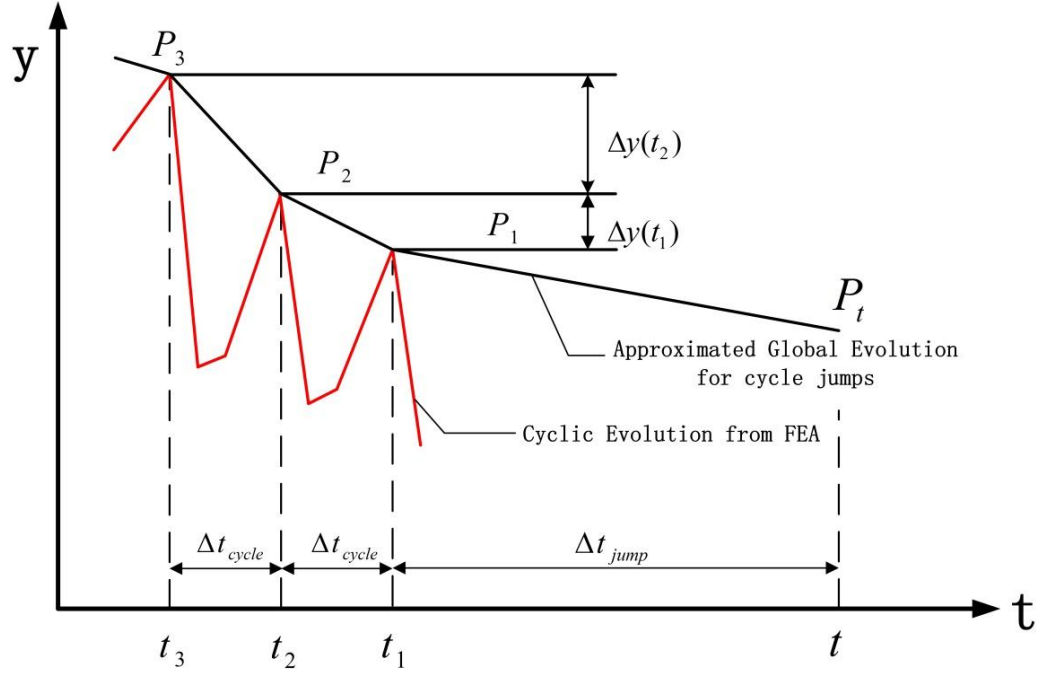


Figure 1.2 A schematic of a state variable evolution for a structure subjected to cyclic loading (Adopted from Cojocaru and Karlsson [1])

The length of the jump (i.e. number of cycles to jump over) is calculated by the ‘control function’ [1]. At least three data points of the state variable (e.g. components of stress, strain, displacement) are extracted after the simulation at the end of each individual cycle. The data points could be at any specific time as long as they correspond to the same relative time in each cycle. These data points are defined as $P_1(t_1, y_1)$, $P_2(t_2, y_2)$, $P_3(t_3, y_3)$, as shown in Figure 1.2. From these points, we can easily obtain the variable change $\Delta y(t_1)$ and $\Delta y(t_2)$, and then define the discrete slopes $s_{12}(t_1) = \Delta y(t_1) / \Delta t_{cycle}$ and $s_{23}(t_2) = \Delta y(t_2) / \Delta t_{cycle}$, where $s_{12}(t_1)$ is the slope of the segment connecting point $P_1(t_1, y_1)$ and $P_2(t_2, y_2)$, $s_{23}(t_2)$ is the slope of the segment connecting point $P_2(t_2, y_2)$ and $P_3(t_3, y_3)$, and $\Delta t_{cycle} = t_1 - t_2 = t_2 - t_3$ is the cycle length (time).

The allowed jump length for each extrapolated variable is dictated by the following criterion [1]:

$$\frac{|s_p(t_1 + \Delta t_{y,jump}^M) - s_{12}(t_1)|}{|s_{12}(t_1)|} \leq q_y \quad (1.1)$$

where q_y is a relative error ($q_y \geq 0$), $\Delta t_{y,jump}^M$ is the time spanned by the jump for the material point M , $s_p(t_1 + \Delta t_{y,jump}^M)$ is the predicted slope at the moment after the jump, obtained by linear extrapolation as [1]:

$$s_p(t_1 + \Delta t_{y,jump}^M) = s_{12}(t_1) + \frac{s_{12}(t_1) - s_{23}(t_2)}{\Delta t_{cycle}} \Delta t_{y,jump}^M \quad (1.2)$$

The value of allowed jump length is now easily obtained [1]:

$$\Delta t_{y,jump}^M = q_y \Delta t_{cycle} \frac{|s_{12}(t_1)|}{|s_{12}(t_1) - s_{23}(t_2)|} \quad (1.3)$$

The value of q_y is specified by the user and may vary as the simulation progresses.

In general, $\Delta t_{y,jump}^M$ will be unique for each material point and variable. Thus, the jump length is set to be the minimum $\Delta t_{y,jump}^M$ as [1]:

$$\Delta t_{jump} = \Delta t_{cycle} \left[\min \{ \Delta t_{y,jump}^M \} / \Delta t_{cycle} \right] \quad (1.4)$$

Finally, the extrapolated state variables could be calculated by [1]:

$$y(t_1 + \Delta t_{jump}) = y(t_1) + \frac{1}{2} [s_{12}(t_1) + s_p(t_1 + \Delta t_{jump})] \Delta t_{jump} \quad (1.5)$$

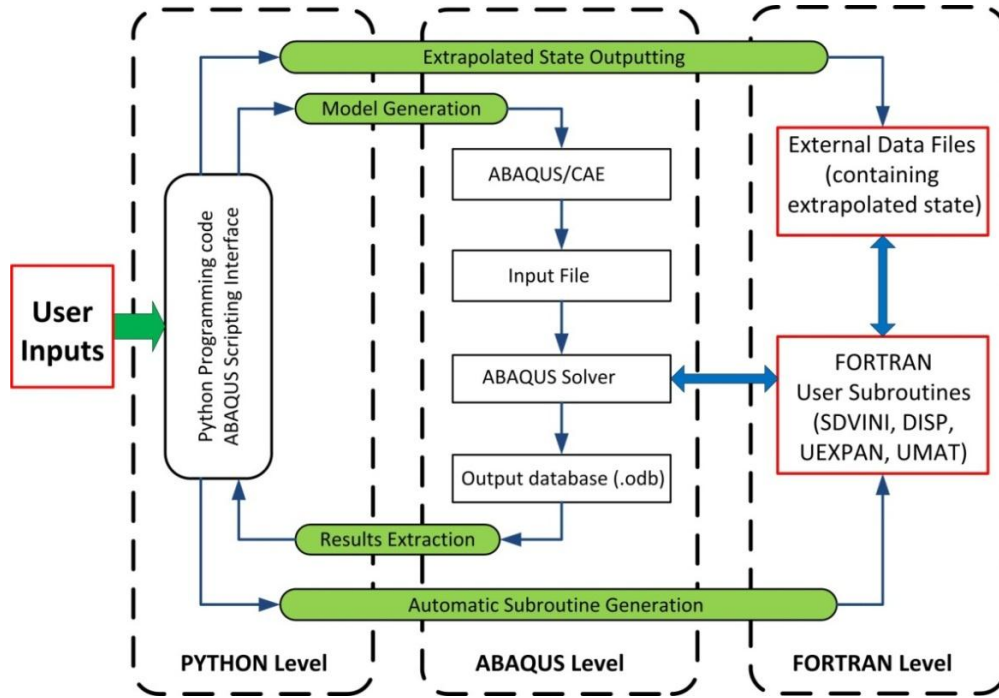


Figure 1.3 A schematic of the implementation of cycle jumps (Adopted from Cojocaru and Karlsson [1])

Figure 1.3 shows the implementation scheme of the cycle jump technique. The finite element model is created by ABAQUS/CAE and Python scripts for the ABAQUS Scripting Interface. A set of cycles is simulated and solved by ABAQUS Solver which interacts with the user subroutines. The solution results are extracted and the states of variables are analyzed by the Python scripts for jump attempts. The analysis establishes the global evolution function and determines the jump length based on the control criteria by Eq. (1.1). The states are then extrapolated based on the global evolution function and output to external data files. The extrapolated states are then initialized in the model by the Fortran user subroutines: the extrapolated displacements for every node are described through DISP, the extrapolated stresses and strains for every integration point are initialized by SDVINI, and UMAT.

This routine had been successfully utilized to conduct cycle jumps in a basic FEA model for a cyclically loaded cylinder [1] where it was implemented using the commercial program ABAQUS and PYTHON code. However, since that work focused on the proof of concept, simplified material behaviors and simple geometry was used. Furthermore, it has not been compared to experimental results. In this work, we will extend this numerical technique for are more realistic structures and material behaviors. Then we will compare to experimental results.

1.4 Thermal Barrier Coatings

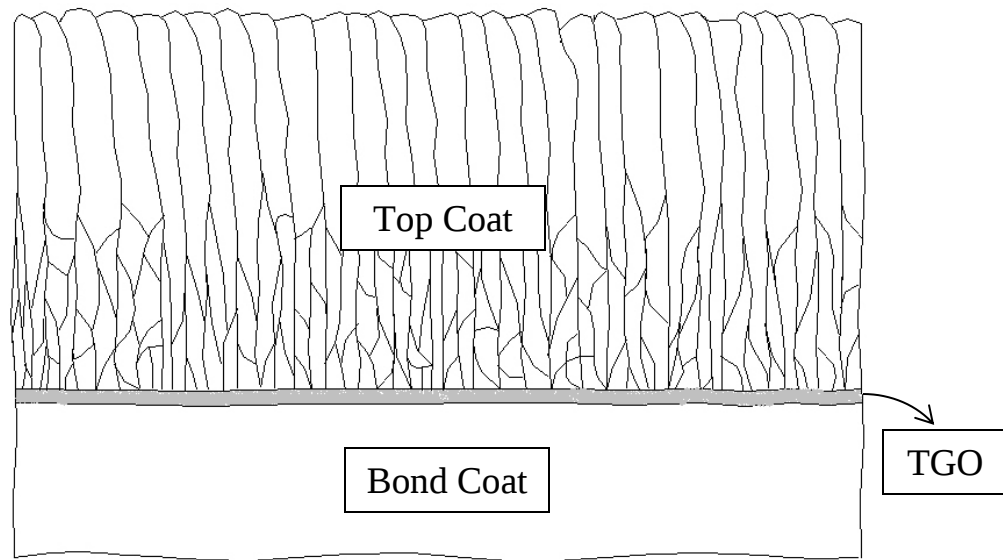


Figure 1.4 Schematic of thermal barrier coating

Thermal barrier coatings are examples of application that have inspired this work. Here we will give a short overview of thermal barrier coatings (TBCs). TBCs are used for high temperatures protection in the hot sections of gas turbines [16].

TBCs are multilayered material systems, typically a nickel based super alloy layered with a bond coat and a top coat as shown in Figure 1.4. The bond coat (BC), a metallic aluminum-rich layer, provides oxidation protection for superalloy structure. The ceramic top coat (TBC) allows a high temperature gradient which provides thermal protection. The gradient is sustained via internal cooling of the superalloy. The system is subjected to complicated cyclic loads, combining both thermal and mechanical loading.

Repeated cyclic loading eventually leads to fatigue and unpredicted failures of the structures. Understanding the failure of these structures requires important information of materials under specific load conditions. Failures in TBCs are mostly driven by the cyclic loading and a slow evolution of material properties, e.g., the yield strength or elastic modulus may change as a function of time at high temperature. Thus, TBCs exhibit a strong interaction between time- and load-dependent degradation.

When exposed to high temperature, between the bond coat and the top coat, a thermally grown oxide (TGO) layer develops due to the oxidation of the bond coat. The TGO mainly consists of aluminum oxide (alumina, Al_2O_3). It is initially less than $0.5 \mu\text{m}$ thick and grows up to $7\text{-}10 \mu\text{m}$ before failure. A schematic of the oxidation mechanism is showed in Figure 1.5: the aluminum from the substrate, q_{Al} , react with the oxygen from air, q_{O_2} , to form alumina, the thermally grown oxide (TGO). Most of the new TGO resides between TGO and bond coat and results in thickening, while some forms between preformed TGO grains and causes lengthening [13]. The induced thickening strain is denoted as ε_t , while the lengthening strain is denoted as ε_g .

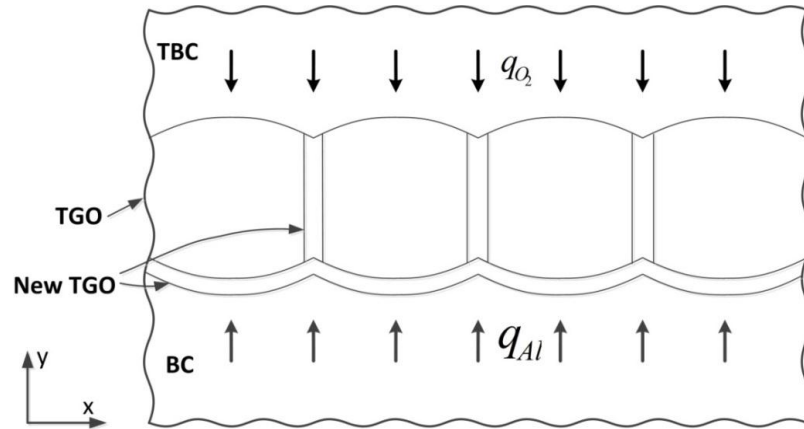


Figure 1.5 Schematic of oxidation mechanism at high temperature (Adopted from Ref. [6])

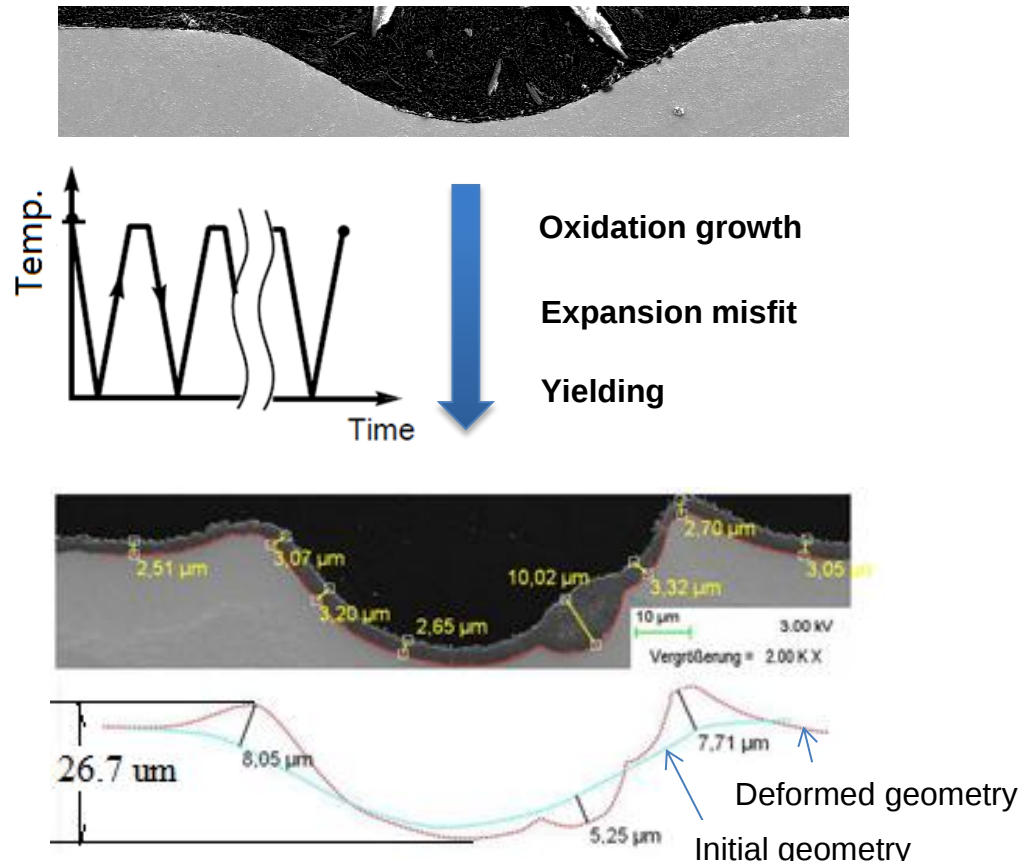


Figure 1.6 Morphological instability of FeCrAlY due to high temperature oxidation (courtesy Dzodzovic and Bartsch, the German Aerospace Center)

Cyclic morphological instabilities in the thermally grown oxide (TGO) are characterized by the local imperfections which grow on a cyclic basis [17]. They represent one source of failures in some thermal barrier systems. Figure 1.6 shows the mechanism of the cyclic morphological instabilities. Observations and simulations have indicated that several factors interact to cause these instabilities to propagate [17]: (i) thermal cycling; (ii) thermal expansion misfit; (iii) oxidation strains; (iv) yielding in the TGO and the bond coat; and (v) initial geometric imperfections. Due to the growth and the presence of the TGO, crack will initiate, grow and coalesce in its vicinity, which eventually results in final failure of the structure by spallation [17].

FeCrAlY is one type of bond coat used in thermal barrier coatings. The material consists of iron (Fe), chromium (Cr), aluminum (Al) and yttrium (Y). Due to the material being prone to a high rate of oxidation, it exhibits a significant tendency for morphological instabilities. Thus, subjected to high temperature oxidation, FeCrAlY will experience significant morphological changes. Therefore, we utilize cyclic oxidation of FeCrAlY as an example for the verification of the cycle jump technique. Hence, for this work, we will investigate the oxidation evolution of FeCrAlY specimen when subjected to high temperature cycles. The morphological instability will be investigated and will serve as an application of cycle jump for the verification of its validity.

1.5 Experiments

1.5.1 Experimental Goal

To verify that the cycle jump technique can capture experimental results and to demonstrate the power of the method, experimental results, the cycle-by-cycle reference simulation, and the simulation with cycle jump must be compared. To provide experimental observation for the verification, we proposed experimental investigation of the morphological instability of FeCrAlY with grooved surface subjected to high temperature thermal cycles. We selected this material and load system mainly for three reasons: (i) it contains few and relatively well defined parameters; (ii) it shows a clear and easy to measure change that only comes with cyclic loading, (iii) these are relatively easy experiments. The experimental work was performed by our collaborators, Dzodzovic and Bartsch at German Aerospace Center (DLR) in Cologne, Germany. The experiments will provide realistic inputs for numerical model, and show the real instability evolution for comparisons, which will ultimately serve as application example to verify that the cycle jump technique works in capturing the experimental observation.

1.5.2 Methodology

Grooves are developed on the specimen surface (shown in section 1.5.3.1) to represent the initial imperfections. The specimens are then hung in the furnace for specified high temperature thermal cycling to develop the morphological instabilities. Each thermal cycle consists of three steps: cooling to low temperature, heating to high temperature, and holding at high temperature. After specified number of cycles, the specimens are investigated under the scanning electron microscope (SEM) to show the morphological shape change.

The experimental parameters must be carefully selected, so that the morphological instabilities can be developed. Based on previous experimental investigations [18] and numerical studies [19,20], we conducted preliminary numerical simulations to guide the parameter selections. If the holding temperature is too high, the oxidation will occur too fast and therefore we can not have sufficient cycles before failures to demonstrate the cycle jump technique. If the cooling rate is too high, there might be cracks in the structure due to ‘thermal shock’. In order to show instability evolution, we proposed interrupted tests where three specimens were subjected to the same operating conditions and then were pulled out of the furnace subsequently at selected time. The series of specimens, subjected to increasing numbers of cycles, were investigated under SEM to show the instability evolution. The time points for interrupted tests were also suggested based on the deforming evolution from our numerical results. The selected thermal cycles, the parameters of holding time, holding temperature, cooling rate, heating rate, and the interrupted tests are summarized in Tabel 1.1.

TGO growth law, which describes how the oxidation rate evolves, is a very important input for the numerical simulations. It can be investigated using two methods. One method is to fit the data of TGO thickness measured on the SEM image from interrupted tests. This method is more straightforward, but many tests are needed to get reliable data. A second method is to establish the growth law from the measurement of mass change of the specimen. In this case, the specimens are weighted at selected time points during the tests. Thus, this is a non-destructive method. The mass change is primarily due to the gain of oxygen during the oxidation.

Knowing the mass change of the specimen, Δm , the mass change of the oxidation, Δm_{TGO} , can be calculated from

$$\Delta m_{TGO} = \frac{\Delta m}{3 \cdot M(O)} \cdot M(Al) \cdot 2 \quad (1.6)$$

where $M(O)$ and $M(Al)$ are the relative molecular mass of oxygen and aluminum respectively. Assuming the oxidation thickness is uniform around the specimen surface, and given the total oxidation surface area and the density of TGO which is $4 \text{ mg} / \text{mm}^3$ [21], the TGO thickness evolution function $h(t)$ can be established.

1.5.3 Experiments

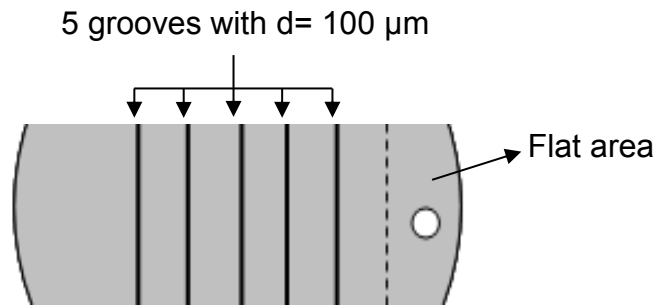


Figure 1.7 Geometry of the specimen

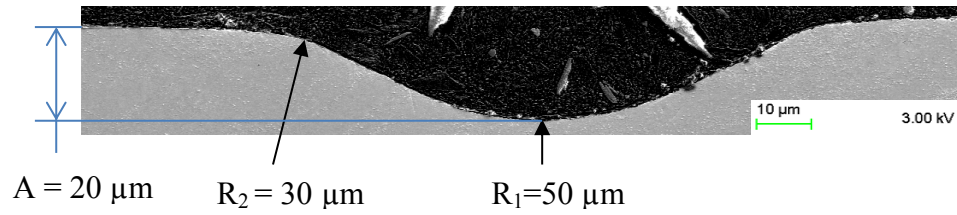


Figure 1.8 Cross section of the grooves

1.5.3.1 Specimens

A schematic top view of the specimen is shown in Figure 1.7. On the specimens, 5 grooves with width of 100 μm are developed to represent the initial imperfection. As shown in Figure 1.8, the initial imperfection is characterized by two radii and an initial amplitude, R_1 is about 50 μm , R_2 is about 30 μm , and A is about 20 μm .

1.5.3.2 Parameter settings and experimental results

Table 1.1 Experimental parameter settings

Experiments	Holding temperature, T_H	Holding time	Cooling rate	Tests
HL	1087 °C	45 mins	870 °C/min	176 cycles + 19 hrs isothermal
LS	1040 °C	18 mins	500 °C/min	200/350/500 cycles
New proposed	1100 °C	30 mins	500 °C/min	60/120/200 cycles

Tabel 1.1 summarizes the parameter settings for the experiments. For the experiment with high temperature and long holding (HL), the specimen was subjected to thermal cycles with 15 minutes for cooling and heating, and 45 minutes for holding at 1087 °C. The cooling rate was about 870 °C/min. The specimen was subjected to 176 cycles of cyclic loading. Due to a technical error, the specimen was then held at high temperature for another 19 hours in isothermal condition. The morphological change is shown in Figure 1.9. The deformed amplitude A is about 26.7 μm compared to the initial amplitude of 20 μm , and the TGO thickness h is about 3 μm compared to zero initially. There is a potential issue that there is spallation of oxidation, which we believe is due to ‘thermal shock’ because of high cooling rate.

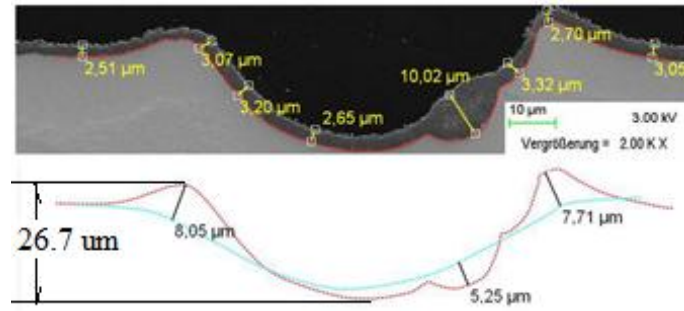


Figure 1.9 Morphological change from the HL experiment

For the experiment with low temperature and short holding time (LS), the specimen was subjected to thermal cycles with 15 minutes for cooling and heating, and 18 minutes for holding at 1040 °C. The cooling rate was about 500 °C/min. The SEM image of morphological change after 200 cycles is shown in Figure 1.10. The deformed amplitude A is about 22.4 μm , and the TGO thickness h is about 1.60 μm . This case gave a non-uniform TGO thickness and minor morphological change. We attribute this to the short holding time.

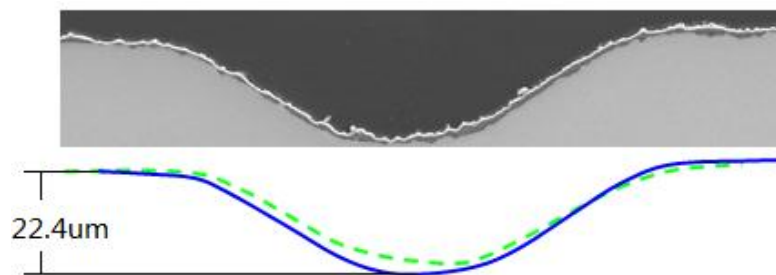


Figure 1.10 Morphological change from the LS experiment after 200 cycles

1.5.3.3 Future experiments

Based on the two initial sets of experiments, we propose a third set of experiment with parameter settings also given in Table 1.1. Based on our preliminary simulation with the new suggested parameters, we believe that this new proposed experiment will overcome the issues limiting the evaluation from previous tests. The specimens will be subjected to thermal cycles with 15 minutes for cooling and holding, and 30 minutes for holding at high temperature of 1100 °C. The holding at 1100 °C will give uniform TGO thickness, and 30 minutes holding per cycle will provide sufficient cycles for simulation with cycle jump. Moreover, we have not obtained reliable TGO growth law from the two initial experiments. In order to get more reliable data, the mass change will be measured more frequently at the beginning when the oxidation grows much faster. Thus, this new experiment is proposed to show the instability evolution and to get reliable the TGO growth law.

Chapter 2

EXTENDING THE CYCLE JUMP TECHNIQUE

Showing the power of the cycle jump technique in accelerated simulation, Cojocaru and Karlsson were looking at a basic model with simplified material properties [1]. The cycle jump was proposed to have wide applications in accelerating simulation of multilayered structures subjected to cyclic loading. The overall goal is to verify that the cycle jump technique is able to capture the experimental results and demonstrate the power of it in computational savings.

Material structures subjected to thermal cycling typically exhibit non-linear time- and temperature-dependent material properties. Thus, we set out to expand the model and extend the cycle jump technique. A more sophisticated extrapolation scheme is developed and utilized. For simulating the experiments described in Section 1.4, a more realistic TGO growth strain law needs to be incorporated. Hence, we will investigate the performance of cycle jump in modeling the structure with time-dependent TGO growth strain rate and temperature-dependent yield strength. Moreover, we will later see that creep is important to be incorporated compared to the previously used time-independent plasticity. Thus, we will also include creep into the model with cycle jump for investigation.

Two sets of simulations will be conducted: cycle-by-cycle reference simulations and simulations with cycle jump. These two sets of simulations are completed separately. Cycle-by-cycle reference simulations only serve as reference for comparison with the results from simulations with cycle jump, so as to evaluate the

performance of the cycle-jump technique. Results from the cycle-by-cycle reference simulations are never used as input in the simulations with cycle jump.

2.1 Introduction of A Polynomial Extrapolation Scheme

2.1.1 Theoretical Framework

The extrapolation scheme that Cojocar and Karlsson [1] used was described in Section 1.3.1. For convenience, this extrapolation method will be named ‘linear extrapolation’ in the discussion that follows. As described previously, the ‘linear extrapolation’ takes discrete slopes based on cycle-by-cycle simulations¹, extrapolates a slope for the jump control, and utilizes these slopes for extrapolating the state variables after the cycle jump. With the simple material properties in a basic model as shown in that work, the ‘linear extrapolation’ scheme works well. However, for a more complicated structure like the one used in our experiments, this extrapolation scheme turns out to have some limitations discussed below.

The solid curve in Figure 2.1 shows the evolution of a state variable, y , where the data are sampled periodically at the end of each cycle from a cycle-by-cycle reference simulation. The dashed curve shows the evolution of the discrete slope, $\Delta y/\Delta N$. We can see that the discrete slopes fluctuate most likely due to numerical issues. Thus the ‘linear extrapolation’ is not suitable since this extrapolation scheme will not allow efficient jumps. For example, from cycle-to-cycle simulations before a jump attempt, we can extract data points, $P_6 - P_1$, as shown in Figure 2.2. If we take the last three data points, P_3 , P_2 , and P_1 , and attempt to make a jump by ‘linear

¹ Note: When the cycle jump technique is used, initial cycle-by-cycle simulations are used followed by a jump. This is not to be confused by cycle-by-cycle reference simulations.

extrapolation' from the data, then the jump control criteria will not allow it. Thus, a more sophisticated extrapolation scheme should be used so that the simulations are not governed by minor numerical fluctuations. Thus, here we developed an alternative more general extrapolation method, called 'polynomial extrapolation'.

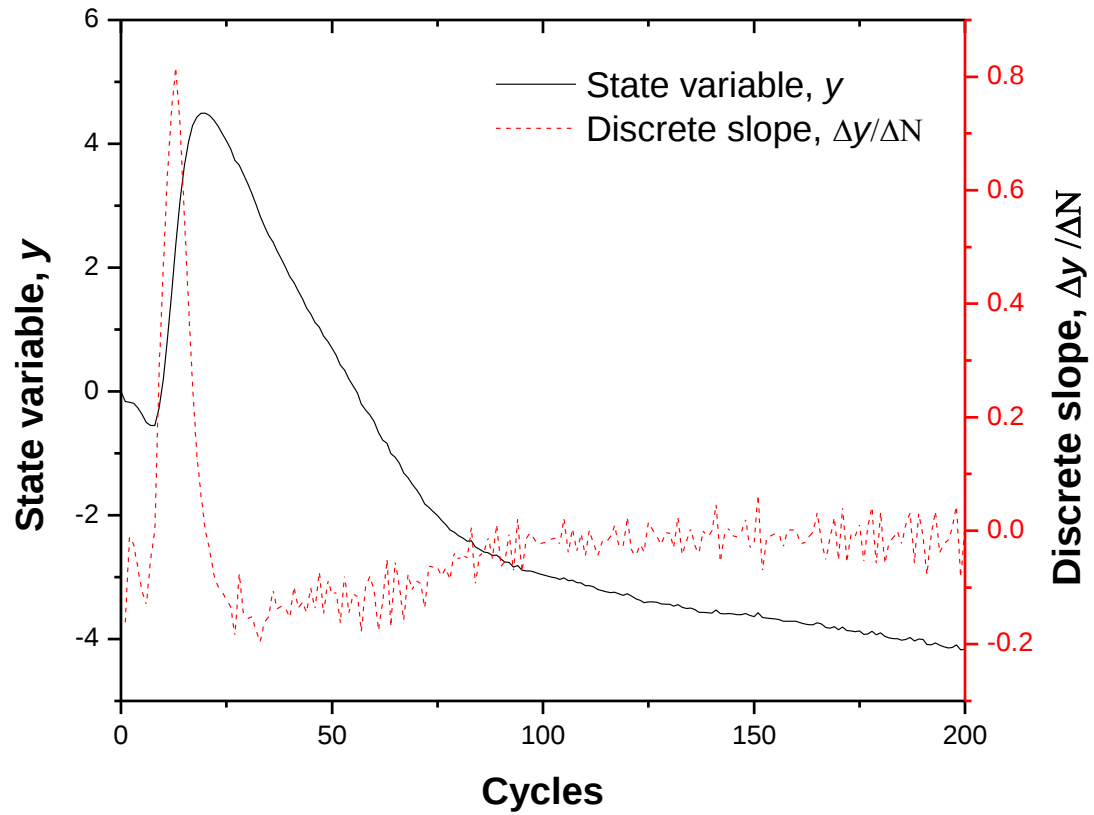


Figure 2.1 Evolution of a state variable and its discrete slope within a structure subjected to cyclic loading

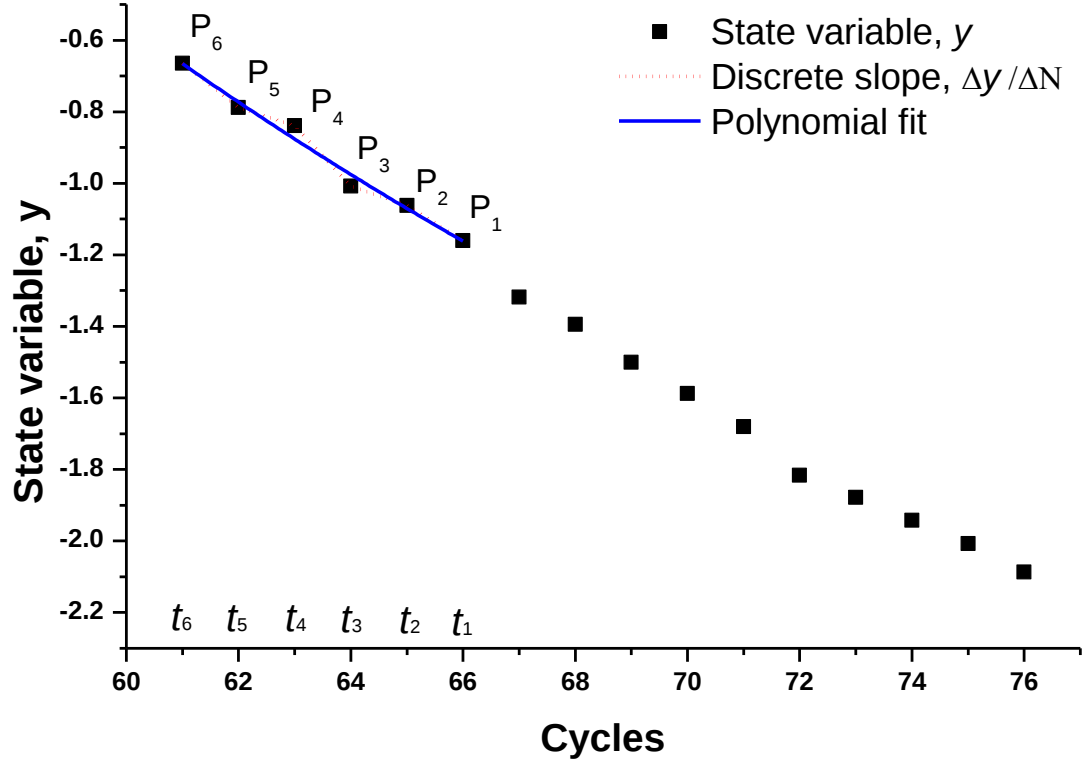


Figure 2.2 Schematic of polynomial extrapolation for state variables within a structure subjected to cyclic loading

Figure 2.2 shows the implementation scheme of the ‘polynomial extrapolation’. For this extrapolation, six data points of the state variable, $P_6 - P_1$, are extracted periodically from the simulation at the end of each cycle (note that the data could also be sampled at any specific time which is relatively the same in each cycle). From a 2nd order polynomial fit with least squares, we can obtain the evolving function of the state variable as:

$$y(t) = A + B \cdot t + C \cdot t^2 \quad (2.1)$$

where t is the time and $y(t)$ stands for any evolving variable. A , B , and C are constants obtained from the polynomial fit.

Taking a derivative of Eq. 2.1 gives the evolving slope function dictated as:

$$s(t) = \frac{dy}{dt} = B + 2C \cdot t \quad (2.2)$$

For control of the cycle jump, the jump length allowed for each extrapolated variable follows bellow criterion:

$$\frac{|s(t_1 + \Delta t_{y,jump}^M) - s(t_1)|}{|s(t_1)|} \leq q_y \quad (2.3)$$

where q_y is the relative error specified by user as the input control parameter, $\Delta t_{y,jump}^M$ is the jump length for material point M, $s(t_1)$ represents the slope at t_1 which is the time point before the cycle jump, $s(t_1 + \Delta t_{y,jump}^M)$ is the extrapolated slope at $t_1 + \Delta t_{y,jump}^M$ after the jump. If the attempted jump length does not satisfy the above control criterion, it will be reduced by one cycle iteratively to ensure that the extrapolated slope is sufficiently close to the slope before the attempted jump.

In general, $\Delta t_{y,jump}^M$ will be unique for each material point and variable. Thus, to ensure that the control criteria are fulfilled, we select the jump length to be the minimum $\Delta t_{y,jump}^M$ as:

$$\Delta t_{jump} = \min \{ \Delta t_{y,jump}^M \} \quad (2.4)$$

Hence, we can get the extrapolated state variable after the jump and express it as:

$$y(t_1 + \Delta t_{jump}) = A + B \cdot (t_1 + \Delta t_{jump}) + C \cdot (t_1 + \Delta t_{jump})^2 \quad (2.5)$$

2.1.2 Numerical Verification

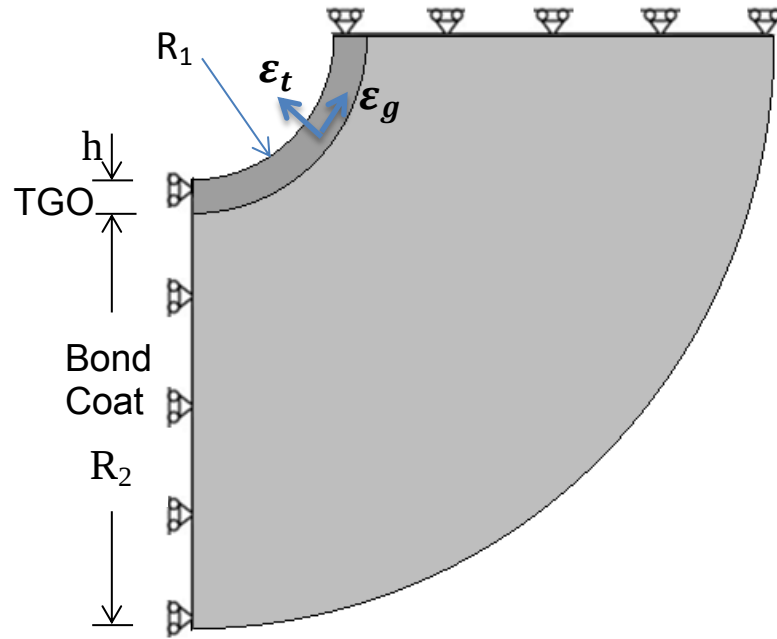


Figure 2.3 Basic finite element model of a sample structure

Table 2.1 Simplified material properties

Material	E (GPa) [18,20]	ν [18]	σ_Y^* (MPa)	Thermal Expansion Coefficient $\alpha \cdot 10^6 / ^\circ\text{C}$ [18,20]
Bond Coat	190	0.3	200	14
TGO	380	0.2	10000, $T \leq 900^\circ\text{C}$ 1000, $T \geq 1000^\circ\text{C}$	8

*The yield strength of the TGO varies linearly between the two temperatures.

For verification of this newly proposed extrapolation scheme, we first investigate the application of the ‘polynomial extrapolation’ on the basic model used in the previous work by Cojocaru and Karlsson [1]. The commercial finite element software ABAQUS [2-4] will be used. The basic model, shown in Figure 2.3, is a two

layer cylinder with dissimilar materials. The TGO thickness h is taken to be $1\text{ }\mu\text{m}$ initially, the inner radius of the structure R_1 is $15\text{ }\mu\text{m}$, and the thickness of the bond coat R_2 is $50\text{ }\mu\text{m}$. The model consists of 5752 nodes and 1800 of Quadrilateral 8-node generalized plane strain elements (CPEG8). The material properties are summarized in Table 2.1. Each thermal cycle consists of three steps: cooling to room temperature, heating to high temperature and holding at high temperature. At maximum temperature during holding, the TGO grows with a thickening rate of $\varepsilon_g = 10^{-3}$, and a lengthening rate of $\varepsilon_t = 10^{-3}$ per cycle. For comparison, the result from application of the ‘linear extrapolation’ scheme [1] is duplicated.

Figure 2.4 shows the application of cycle jump with the two different extrapolation schemes in capturing the radial displacement evolution of the inner TGO surface as a function of cycles when the structure is subjected to high temperature thermal cycles. The graphs show the results from the cycle jump simulations compared to the cycle-by-cycle reference simulations. Figure 2.4.A shows the results from the cycle jump using ‘linear extrapolation’, while Figure 2.4.B shows the results from the cycle jump utilizing ‘polynomial extrapolation’. In simulating 500 cycles, the cycle jump with the ‘linear extrapolation’ saves 307 cycles of simulation by making 23 jumps. This is similar to the results by Cojocaru and Karlsson [1]. Utilizing the cycle jump with the ‘polynomial extrapolation’, we save 310 cycles of simulation by 18 jumps. In this case, compared to the reference simulation, cycle jump with both extrapolation schemes captures the structural behavior very well while save about 60% of the computational efforts. Note that the time cost of data extraction and extrapolation during jump attempts or jumps are negligible compared with the significant savings by “jumping” over a large number of cycle-by-cycle simulations.

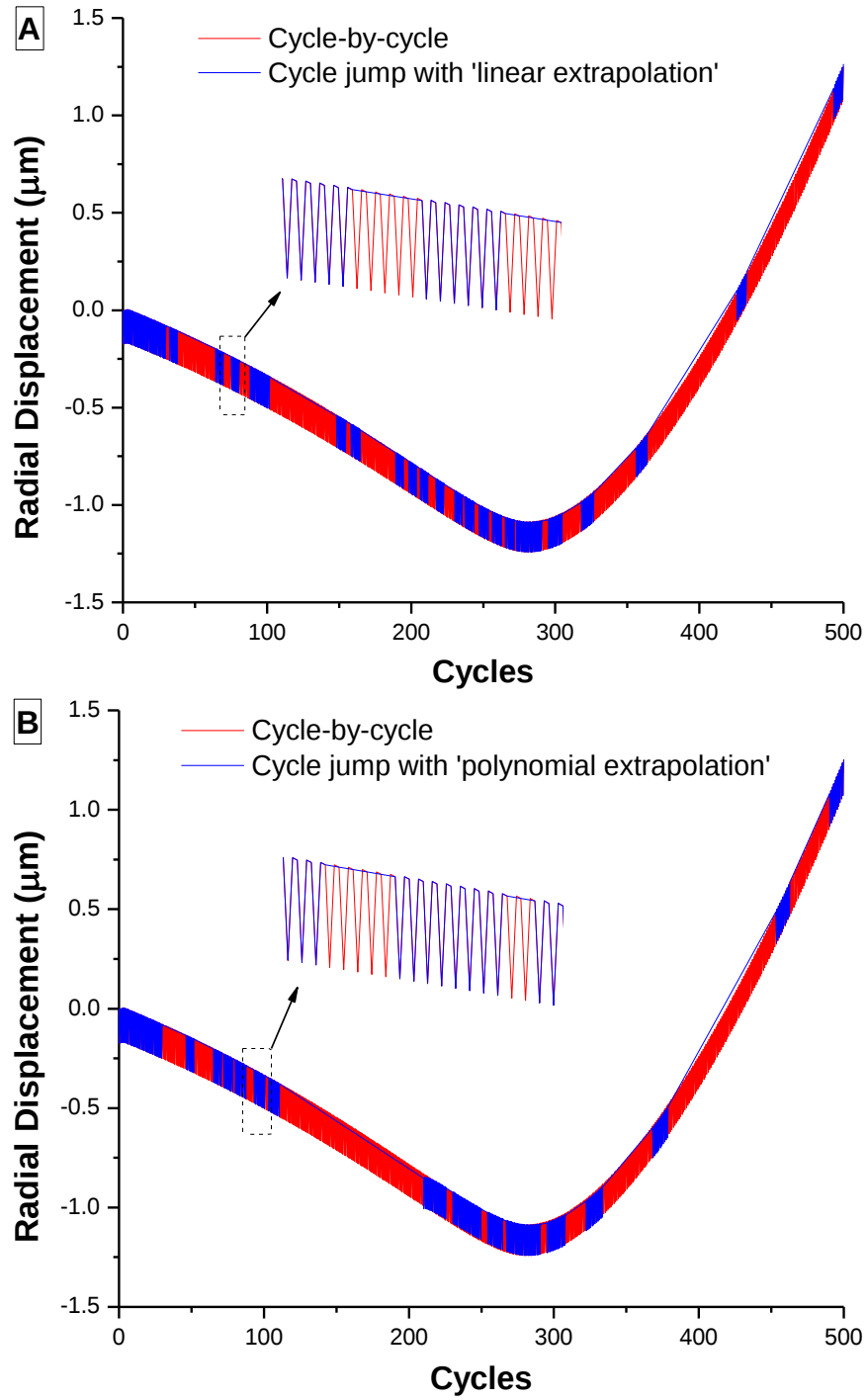


Figure 2.4 Radial displacement evolution of TGO inner surface, (A) simulation with cycle jump utilizing 'linear extrapolation' compared with the reference simulation; (B) simulation with cycle jump utilizing 'polynomial extrapolation' compared with the reference case.

2.2 Time- and Temperature-dependent Material Behaviors

In the previous work by Cojocaru and Karlsson [1], only time-independent material properties were considered. Moreover, the oxidation growth strain rate was taken to be constant, and simplified, time-independent plasticity was considered.

However the oxidation evolution actually is time-dependent [18], and the oxidation growth follows a power law as a function of time [18]. Thus, we will improve the numerical scheme so that it can capture this behavior, and will investigate the performance of cycle jump in modeling the structure with a time-dependent TGO growth strain rate. In addition, we will also impose temperature-dependent yield strength for the bond coat to better represent the behaviour of high temperature material.

With the time-dependent oxidation rate and temperature-dependent yielding behaviour, we will look at a similar sample structure, as shown in Figure 2.3, but with modifications in the dimensions to better represent the experimental geometry. The TGO thickness, h , is taken to be 0.5 μm initially; the inner radius of the structure, R_1 , is 30 μm , and the thickness of the bond coat, R_2 , is 1 mm. The model consists of 4768 nodes and 1504 Quadrilateral 8-node generalized plane strain elements (CPEG8).

2.2.1 Time- and Temperature-dependent TGO Growth

For FeCrAlY subjected to cyclic high temperature, the TGO thickness evolution has been shown to approximately follow [18]:

$$h(t) = k \cdot t^n \quad (2.6)$$

where the parameters, k and n , are temperature-dependent and are given in Table 2.2.a. Taking parameter n to be 0.38 constantly [18], parameter k from polynomial extrapolation for the temperatures of interest in this work is shown in Table 2.2.b.

Table 2.2 Parameters of oxidation thickness growth

Table 2.2.a Data from Ref. [18]

Temp. (°C)	k ($\mu\text{m} \cdot \text{h}^{-n}$)	n
25	-	-
1000	0.255	0.38
1100	0.656	0.40
1200	1.412	0.38
1300	2.630	0.38

Table 2.2.b Extrapolated data from Table 1.a

Temp. (°C)	k ($\mu\text{m} \cdot \text{h}^{-n}$)	n
1000	0.255	0.38
1040	0.362	0.38
1087	0.564	0.38
1100	0.656	0.38

Equation (2.6) and the parameters in Table 2.2.a assume that the initial thickness is vanishingly small. However, the initial thickness is assumed to be 0.5 μm for the TGO layer in the model. By giving a ‘time shift’ for the thickness growth function, the thickness evolution at the four high temperatures is shown in Figure 2.5. The TGO growth is simulated by imposing stress-free [20] growth strain in the lengthening and thickening directions through user subroutine UEXPAN (see Appendix). The incremental true strain for thickening evolution is given as:

$$\Delta\epsilon_t(t) = \ln\left(\frac{h(t + \Delta t)}{h(t)}\right) \quad (2.7)$$

while the incremental lengthening true strain is taken to be proportional to thickening strain as:

$$\Delta\epsilon_g(t) = \beta \cdot \Delta\epsilon_t(t) \quad (2.8)$$

where β is the ratio of lengthening strain rate over thickening strain rate.

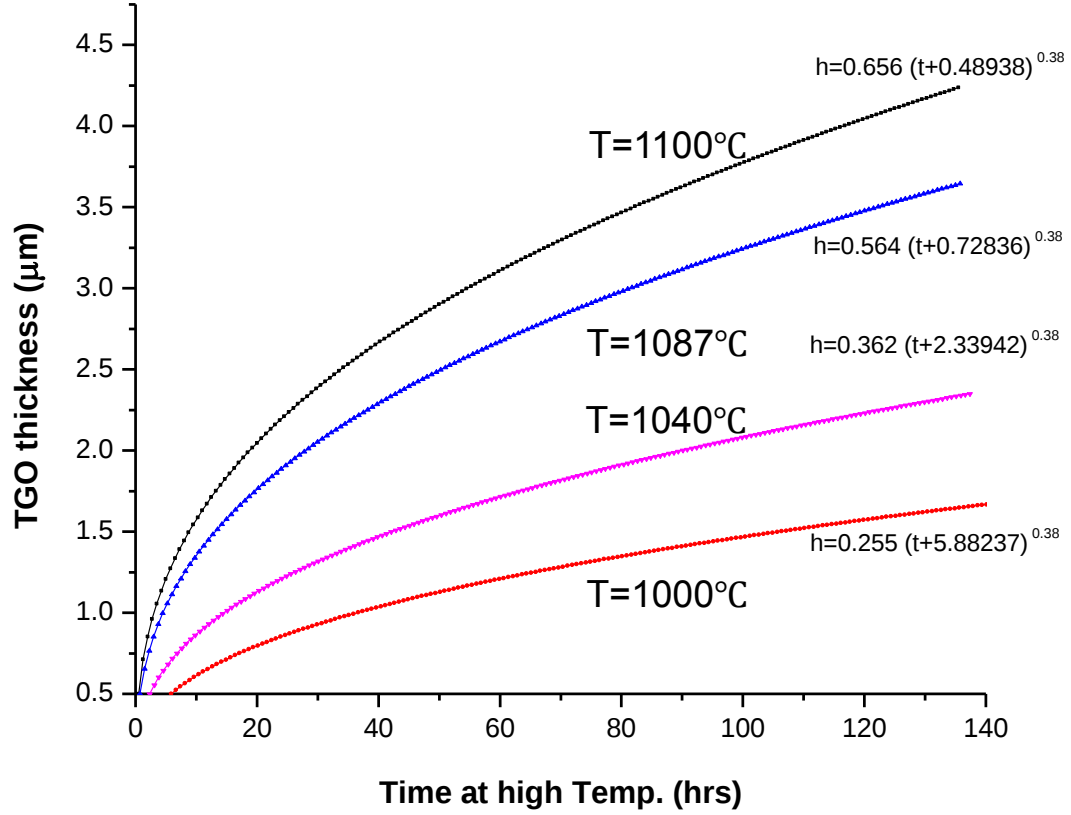


Figure 2.5 TGO thickness evolution at high temperature, adjusting for the initial thickness

In the previous work by Cojocaru and Karlsson [1], the application example considered is not a real-time case. In order to be utilized for simulating real-time problem, corresponding modifications have been made in the cycle jump numerical code. Because the oxidation strain rate is time-dependent, the historic time will be recorded before making a jump. The subroutine UEXPAN will impose the correct time-dependent oxidation strain rate base on the recorded overall historic time.

2.2.2 Temperature-dependent Material Properties

Temperature-dependent material properties are also included. The elastic modulus, poison ratio, and thermal expansion coefficient are summarized in Table 2.3. The elastic modulus and Poisson's ratio for FeCrAlY, tested by Tolpygo and Clarke [18], are assumed to be constant. The elastic modulus for α -Al₂O₃ is temperature-dependent [18], while Poisson's ratio for α -Al₂O₃ is assumed to be constant. The thermal expansion coefficient for α -Al₂O₃ [18] and FeCrAlY [22] are assumed to be linear with increasing temperature.

Table 2.3 Elastic modulus, Poisson's ratio, and thermal expansion coefficient

Material	Elastic Modulus E (GPa) [18]		Poisson's Ratio [18]	Thermal Expansion Coefficient $\alpha \cdot 10^6 / ^\circ\text{C}$ [18,22]
FeCrAlY	190		0.3	$11.73 + 0.0032T$
TGO	25 °C	400	0.25	$7.5 + 0.001T$
	1000 °C	350		
	1100 °C	340		

The yield strengths for FeCrAlY and TGO are summarized in Table 2.4. The yield strength for TGO is approximately 1.2 GPa at high temperature [18], and is very high at lower temperature and is assumed to be 10 GPa at room temperature [20].

Table 2.4 Temperature-dependent plasticity

FeCrAlY [22]		TGO [18,20]	
Temp. (°C)	Yield Strength (MPa)	Temp. (°C)	Yield Strength (MPa)
25	510	40	10000
600	185	1080	10000
800	55	1085	1200
1000	25	1087	1200

2.2.3 Simulation Results

In this section, we will investigate the influence of the control parameter q_y , compare the performance of the two extrapolation scheme, and show the effect of adding a control of maximum jump length.

With the time-dependent oxidation rate and temperature-dependent yielding behavior described in Section 2.2.1 and Section 2.2.2, we will look at a similar sample structure, as shown in Figure 2.3, but with modifications in the dimensions. The TGO thickness, h , is taken to be 0.5 μm initially; the inner radius of the structure, R_1 , is 30 μm , and the thickness of the bond coat, R_2 , is 1 mm. The model consists of 4768 nodes and 1504 elements with the type of Quadrilateral 8-node generalized plane strain (CPEG8).

2.2.3.1 Influence of the control parameter

As discussed earlier in Section 1.3.1, the control parameter q_y defined a criterion for the simulation accuracy. In this section, we will investigate the influence of this parameter on the simulation results. The control parameter is used here to control the acceptable error on the state variable, displacements. Four selected values ($q_y=0.25, 0.5, 1.0, 2.0$) are specified in the jump control for the simulations with cycle jump.

For selected values of the control parameter, q_y , computational savings for simulations with cycle jump utilizing both linear extrapolation and polynomial extrapolation are summarized in Table 2.5. Generally, increasing the value of control parameter q_y , e.g., increasing from 0.25 to 0.5 in this case, will enhance the computational efficiency. However, if the value specified is too high, e.g., $q_y \geq 1.0$ in

Table 2.5 Savings of computation with cycle jump technique

q_y	Linear extrapolation		Polynomial extrapolation	
	Jumps	Saved cycles	Jumps	Saved cycles
0.25	22	324	19	279
0.5	22	342	14	337
1.0	20	342	19	293
2.0	diverged		diverged	

Table 2.6 Relative errors of final radial displacement for selected for selected q_y

q_y	Relative error, E_r	
	Linear extrapolation	Polynomial extrapolation
0.25	0.31%	0.49%
0.5	1.88%	1.88%
1.0	4.66%	1.88%
2.0	diverged	diverged

Table 2.5, the simulations may become slower or even diverge due to convergence difficulties after a jump that is “too large”.

Comparing with reference case, Figure 2.6 shows the influence of control parameter q_y , on the simulation accuracy. The simulation accuracy is described by relative error which is defined as:

$$E_r = \left| \frac{y_{jump} - y_{ref}}{y_{ref}} \right| \cdot 100\% \quad (2.9)$$

For selected q_y , the relative errors of final radial displacement after 500 thermal cycles are summarized in Table 2.6. We can see that the computational accuracy decreases as q_y increases.

Thus, there is a compromise between computational efficiency and accuracy. However with little loss of accuracy, we can significantly enhance the efficiency. Therefore, it is important to select q_y to optimize between accuracy and computational time. In this case, the computation is most efficient with high accuracy when q_y is specified to be around 0.5, where the cycle jump saves about 68% of computational efforts with cost of less than 2% in accuracy.

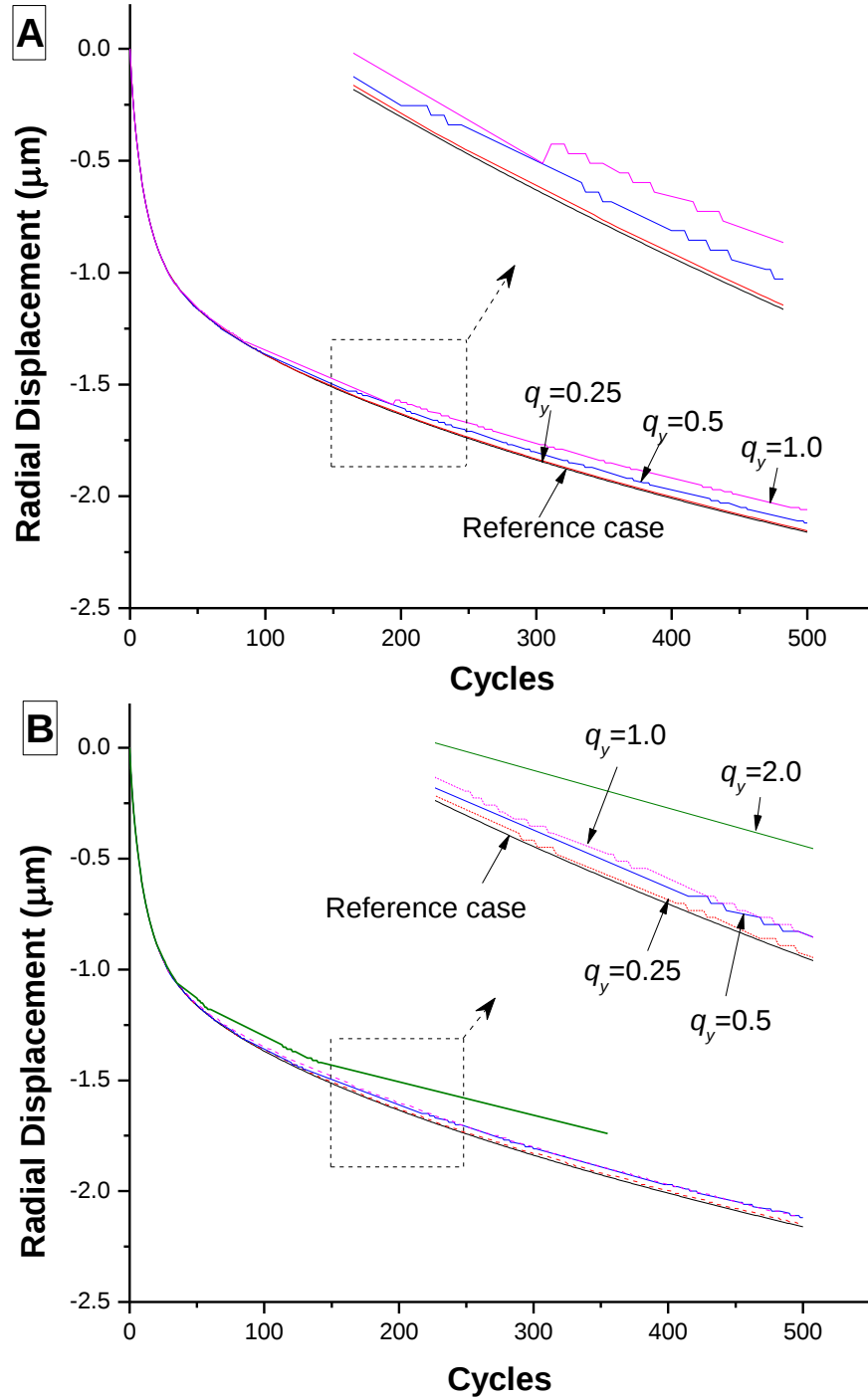


Figure 2.6 Radial displacement evolution of TGO inner surface, simulation results with selected control parameters compared with the reference simulation, (A) cycle jump utilizing 'linear extrapolation'; (B) cycle jump utilizing 'polynomial extrapolation'.

2.2.3.2 Comparison between linear and polynomial extrapolation

Next, we compare the cycle jump technique with the linear and polynomial extrapolation schemes letting $q_y = 0.5$ (Figure 2.7). Over 500 cycles of simulation, the cycle jump with linear extrapolation saves 342 cycles by 22 jumps, while polynomial extrapolation saves 337 cycles by 14 jumps, Tabel 2.5. Their performances are almost the same in computational savings.

For each jump (or jump attempt), it also cost computation time for the data extraction, state extrapolation and initialization after the jump. Thus, in this case, the newly developed polynomial extrapolation is more efficient at making cycle jumps as fewer jumps are needed to finish the same number of cycles.

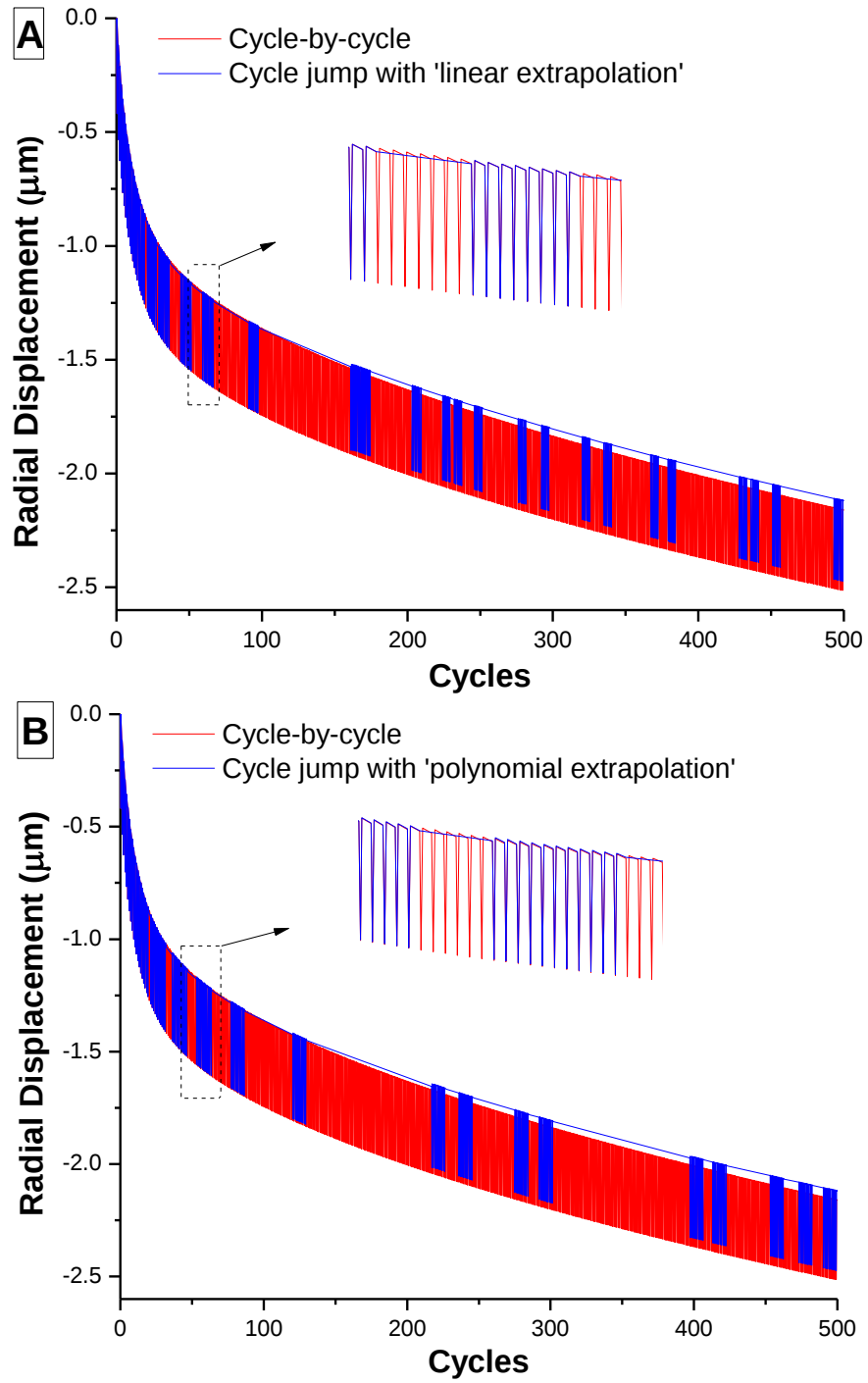


Figure 2.7 Radial displacement evolution of TGO inner surface, $q_y = 0.5$, simulation with cycle jump utilizing different extrapolation scheme compared with the reference simulation, (A) cycle jump with linear extrapolation, (B) cycle jump with polynomial extrapolation.

2.2.3.3 Control of maximum jump length

Next, we will investigate if higher accuracy can be obtained by limiting the maximum possible length allowed in the cycle jump technique.

For example, we can see, in Figure 2.7, that after about 100 cycles, a relatively “linear” region results in a relatively long jump based on jump control with the specified control parameter. It appears that this relatively long jump initiates a small divergence compared to the cycle-by-cycle reference simulation.

Figure 2.8 shows the simulation results of the radial displacement evolution utilizing cycle jump with polynomial extrapolation and a maximum jump control which only allow a jump up to 30 cycles. Comparing with Figure 2.7(B), we can conclude that adding a proper maximum jump control (maximum jump of 30 cycles for this case) can prevent the extrapolation from diverging, thus increase the accuracy. Table 2.7 summarizes the computational savings of the cycle jump technique with ‘polynomial extrapolation’ comparing between cases without and with maximum jump control.

Table 2.7 Comparison of cycle jump w/o and w/ maximum jump control

q_y	Polynomial extrapolation					
	Without maximum jump control			With maximum jump = 30 cycles		
	Jumps	Saved cycles	Relative errors	Jumps	Saved cycles	Relative errors
0.25	19	279	0.51%	18	289	0.44%
0.5	14	337	1.93%	17	307	0.74%
1.0	19	293	1.94%	17	310	1.80%
2.0	diverge			19	283	3.12%

Thus, our results suggest that the accuracy for simulation with cycle jump technique can be improved by adding a proper control over the maximum jump length.

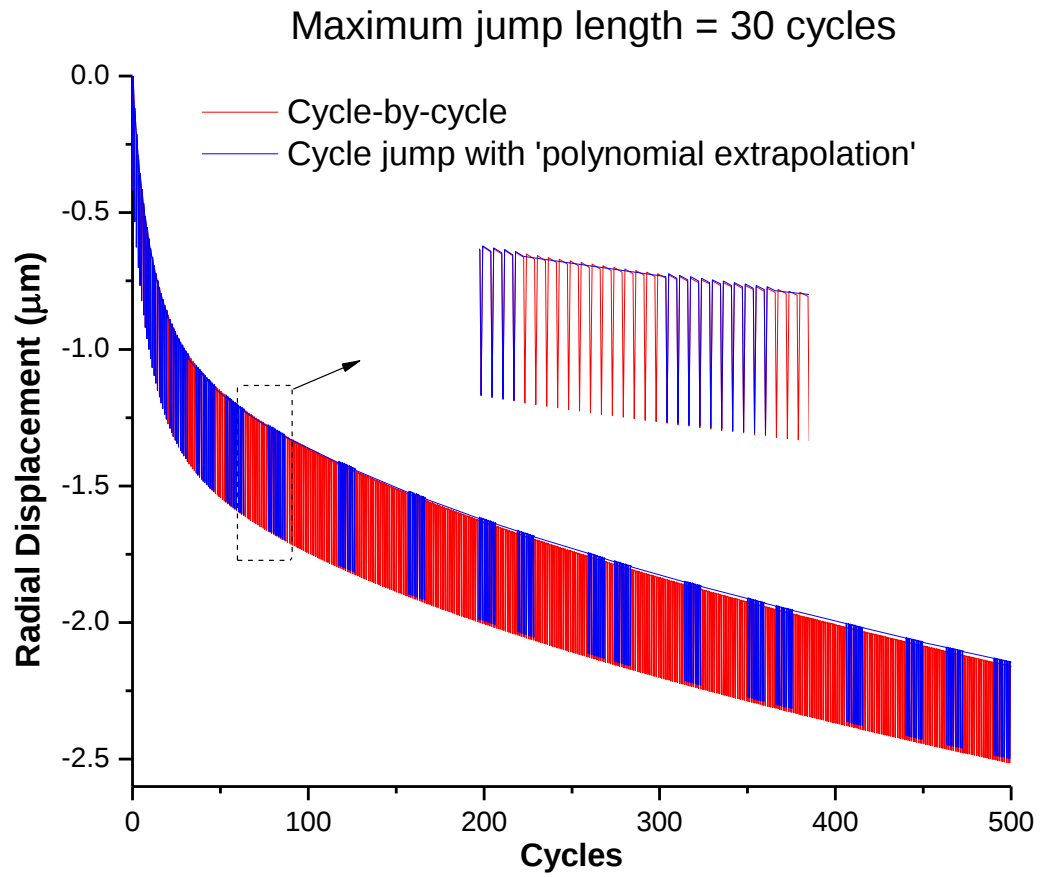


Figure 2.8 Radial displacement evolution of TGO inner surface, $q_y = 0.5$, cycle jump utilizing polynomial extrapolation and a control of maximum jump length = 30 cycles, compared with the cycle-by-cycle reference simulation.

2.3 Time-dependent Properties: Creep

In the previous numerical simulations, simplified time-independent material properties were used. The materials used in our experiments are materials used in high temperature applications with particular focus on TBCs. TBCs exhibit time-dependent material response, thus incorporation of creep into the numerical model is important. Thus, we will now include creep into the model with the cycle jump technique.

2.3.1 Creep

Creep behavior will be incorporated in both the bond coat and the TGO. The creep behaviors are power creep laws based on Refs. [23-25]. Evans, et al [23] summarized the creep behavior for TGO as:

$$\dot{\epsilon} = 1.08 \cdot 10^{-10} \cdot \sigma^{2.3} \cdot \exp\left(\frac{-51000}{T}\right) , s^{-1} \quad (2.9)$$

where $\dot{\epsilon}$ is the strain rate, σ is the material stress value, and T is the temperature. The creep behavior for the bond coat with material of NiCrAlY is described by Evans, et al [23] as:

$$\dot{\epsilon} = 8.96 \cdot 10^{-15} \cdot \sigma^{3.0} \cdot \exp\left(\frac{-35840}{T}\right) , s^{-1} \quad (2.10)$$

The creep behavior for the bond coat with material of FeCrAlY is described in Eq. 2.11 by Saunders, et al [24], and Eq. 2.12 by Echsler, et al [25]:

$$\dot{\epsilon} = 5.96 \cdot 10^{-27} \cdot \sigma^{5.5} \cdot \exp\left(\frac{-47136}{T}\right) , s^{-1} \quad (2.11)$$

$$\dot{\epsilon} = 1.92 \cdot 10^{-17} \cdot \sigma^{3.68} \cdot \exp\left(\frac{-39088}{T}\right) , s^{-1} \quad (2.12)$$

At a given temperature, the creep function can be given as:

$$\dot{\epsilon} = A \cdot \sigma^n , s^{-1} \quad (2.13)$$

Table 2.8 Parameters, A and n , in Eq. 2.13

Temp. (°C)	TGO		Bond coat					
			NiCrAlY		FeCrAlY			
	A	n	A	n	A_1	n_1	A_2	n_2
1000	$2.73 \cdot 10^{-14}$	2.3	$5.33 \cdot 10^{-9}$	3.0	$4.97 \cdot 10^{-10}$	3.0	$1.07 \cdot 10^{-8}$	3.68
1040	$9.25 \cdot 10^{-14}$	2.3	$1.26 \cdot 10^{-8}$	3.0	$1.53 \cdot 10^{-9}$	3.0	$2.73 \cdot 10^{-8}$	3.68
1087	$3.54 \cdot 10^{-13}$	2.3	$3.23 \cdot 10^{-8}$	3.0	$5.31 \cdot 10^{-9}$	3.0	$7.63 \cdot 10^{-8}$	3.68

where, for selected temperatures, the parameters A and n in Eq. 2.9-2.12 are given in Table 2.8.

To capture the experimental results in the simulations, these creep laws were implemented in the finite element model. Simulations with creep laws for FeCrAlY, Eq. 2.11-2.12, failed to converge, while simulations with the creep law for NiCrAlY, Eq. 2.10, converged.

A basic finite element model was created to illustrate the response of the various creep laws for the bond coat. To this end, a uniaxial test was simulated, Figure 2.9, where the structure is subjected to tension and held at constant strain of $\varepsilon = 5 \cdot 10^{-5}$. This strain is selected to result in the instantaneous stress level of about 10MPa, which is the approximate stress level for the material at around 1000 °C based on the our simulations of the experiments. The relaxation response, as shown in Figure 2.10, indicates that FeCrAlY (creep laws in Eq. 2.11 and Eq. 2.12) relaxes much faster than NiCrAlY (creep law in Eq. 2.10). We believe that this is the reason why the creep laws for FeCrAlY raise convergence problem.

We could make significant changes to the models, including changing the mesh and time increments, to accomplish convergence using the FeCrAlY material. However, for simplicity, we will for now implement the NiCrAlY properties to

investigate whether creep properties are possible for use in the cycle jump model. Thus, in this work, we will just utilize the creep law of NiCrAlY, Eq. 2.10, for the bond coat material.

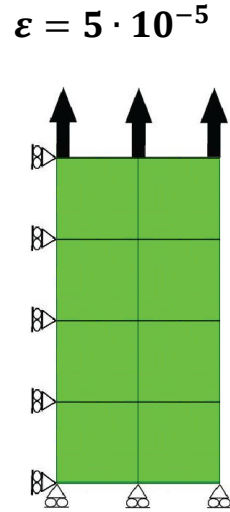


Figure 2.9 Schematic of the model for a relaxation test

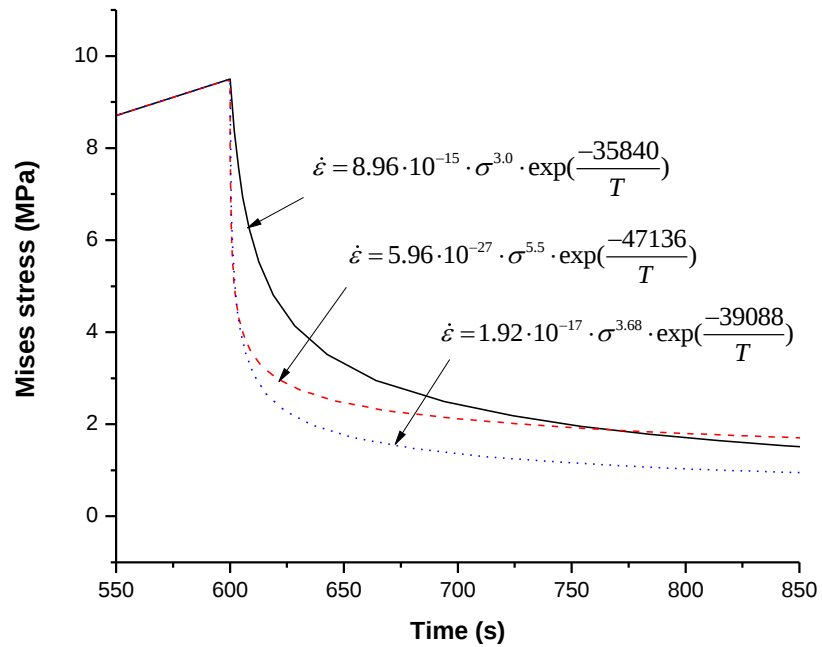


Figure 2.10 Relaxation response in the bond coat for various creep laws, $T=1000$ °C

2.3.2 Incorporation of Creep in the Cycle Jump Technique

For the application of the cycle jump, the extrapolated stress and strain fields are initialized by the subroutine UMAT. Thus, constitutive relations for creep need to be built into UMAT and then incorporated into the scheme of the cycle jump technique. (The subroutine UMAT developed for this work is included in the appendix.)

2.3.3 Simulation Results for Creep Behavior with the Cycle Jump Technique

We utilize the two layer cylinder basic model, Figure 2.3, to investigate the performance of the cycle jump technique when creep behaviors are incorporated. The polynomial extrapolation is used here for the cycle jump technique.

In Figure 2.11, the evolution of radial displacement based on the cycle jump with polynomial extrapolation is compared to that of the reference case. We can see that the results from the cycle jump technique match well with that from the reference case. In this case, with control parameter $q_y = 0.25$, the cycle jump technique is able to save 105 cycles out of 200 cycles by making 7 jumps, which is about 50% in computational efforts. The relative error for the radial displacement at the inner surface by cycle jump is about 0.75% compared to the reference simulation. Thus, in this case, the cycle jump technique saves about 50% of the computation with cost of less than 1% in accuracy.

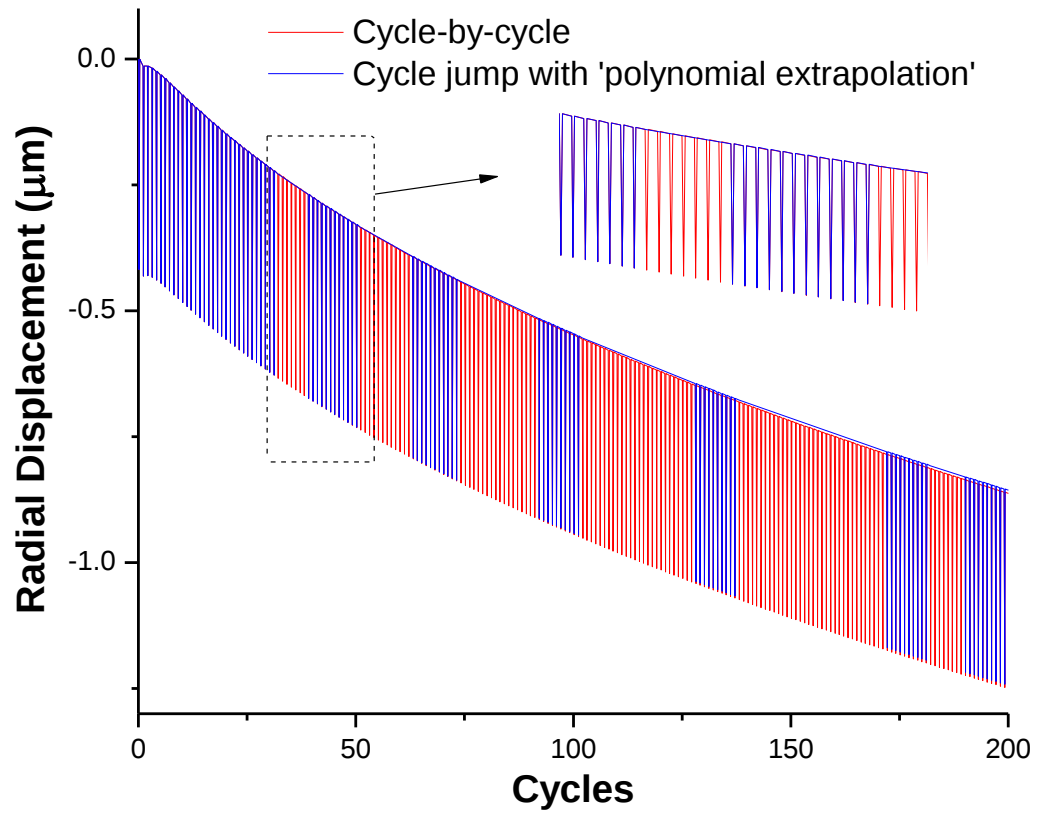


Figure 2.11 Radial displacement evolution of TGO inner surface in creep model, see Figure 2.3, simulation with cycle jump utilizing 'polynomial extrapolation' compared with the cycle by cycle reference simulation.

Chapter 3

VERIFICATION OF THE CYCLE JUMP TECHNIQUE

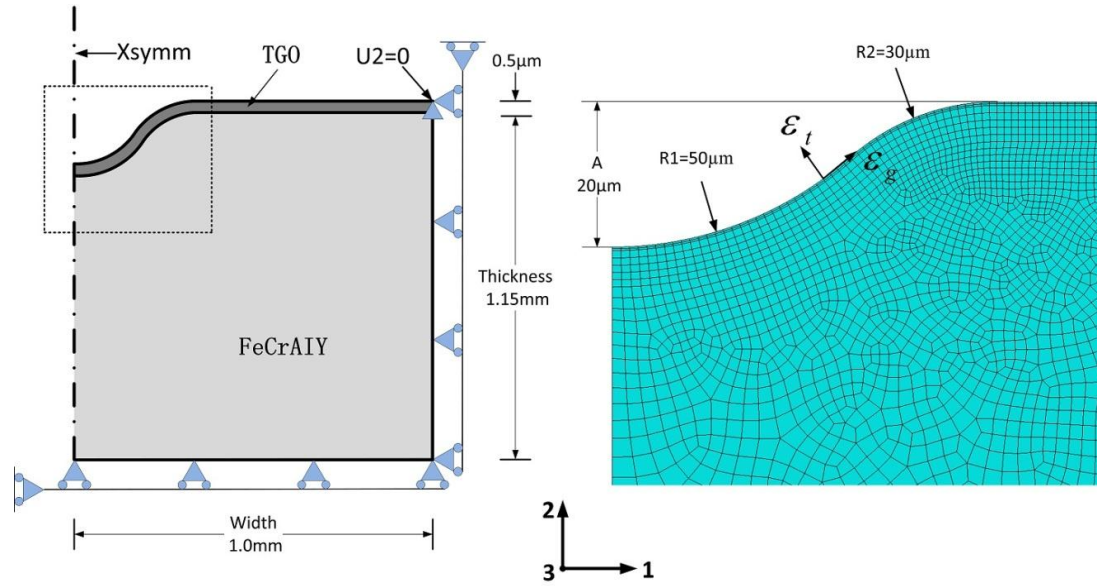
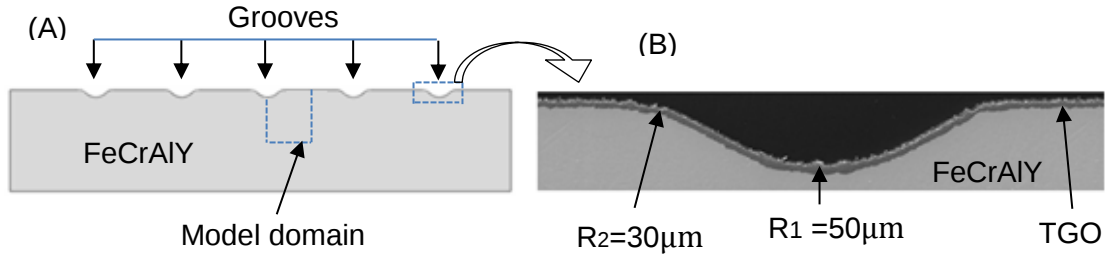
3.1 Simulation Objectives

To verify that the cycle jump technique can capture the real-life experiments and to demonstrate the power of it in accelerating simulation, experimental results, the cycle-by-cycle reference simulation, and the simulation with cycle jump must be compared. Thus, we will investigate the performance of the cycle jump technique in simulating the experiments summarized in Sec. 1.4. The first step is to develop a cycle-by-cycle numerical model which captures the experimental results. During this procedure, the unknown material properties are also investigated through parametric studies. Since the material parameters are unknown in this case, a part of this work provides realistic material data as inputs for the model and will serve as the reference case for verification of the cycle jump technique. Then the cycle jump technique will be imposed for investigations.

3.2 Finite Element Model

3.2.1 Geometry and Boundary Conditions

The geometric dimensions of the specimen are described in the Section 1.4. Figure 3.1 shows the cross section of one groove that is developed on the specimen surface to represent the initial imperfection on the material surface. By assuming symmetry, we can reduce the model size and only need to investigate the domain as



showed in Figure 3.1(a). As described in the Section 1.3.2, when the specimen is subjected to high temperature, a second material layer, the thermally grown oxide (TGO) develops. Thus, the structure has two layers with dissimilar materials. The initial TGO thickness is assumed as $0.5\text{ }\mu\text{m}$ in the model.

The commercially available software ABAQUS [2-4] is used in our simulations. To this end we investigate a two-dimensional finite element model. We utilize the assumption of generalized plane strain providing that the strain in the out of plane direction is assumed to be uniform. Quadrilateral 8-node generalized plane strain elements with reduced integration (CPEG8R) are used for the entire model. The generated finite element model contains 21654 nodes and 6944 elements. The geometric dimensions and boundary conditions in the model are shown in Figure 3.2. A sequence of refined meshes was investigated until convergence of the simulation results was obtained.

3.2.2 Thermal Cycling

The model is stress-free initially at maximum temperature which corresponds to the initial condition for the material. The structure is then brought to room temperature. The stresses at room temperature are the highest due to thermal mismatch. However, we note that the yield strength is significantly higher at room temperature than at high temperature. The structure is subjected to thermal cycling. In the model, each cycle consists of three steps: (1) cooling to room temperature T_L , (2) heating to maximum temperature T_H , and (3) holding at maximum temperature. The loading sequence is illustrated in Figure 3.3, where the dash curve is the thermal cycling in the experiments while the solid curve is the simplified thermal cycling in our numerical model.

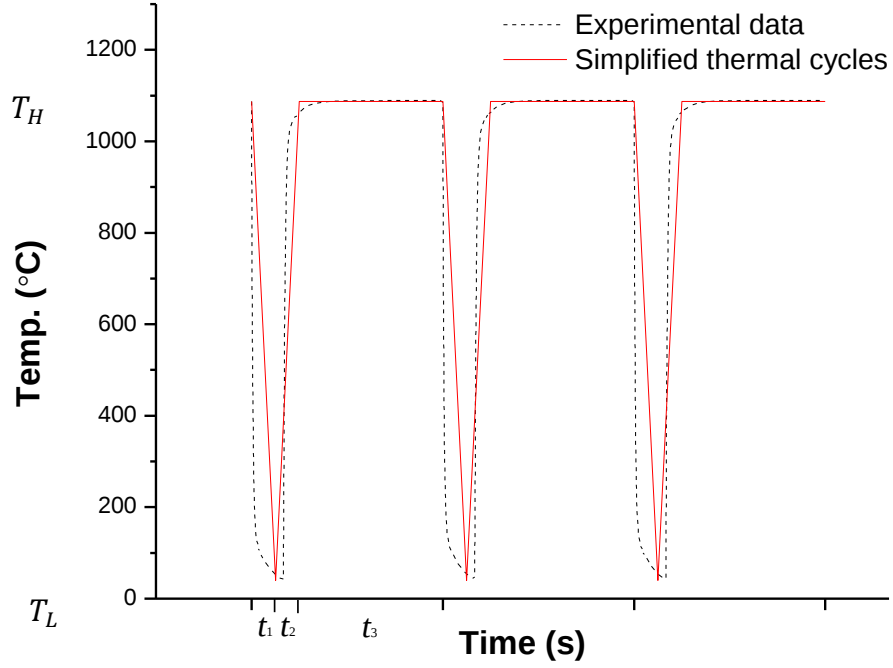


Figure 3.3 Schematic of thermal cycles in the experiments and simulations

3.2.3 Material Behaviors

In previous studies of the morphological instability of TGO [19,20], simplified time-independent material properties were used. The materials used in our experiments are materials used in high temperature applications with particular focus on TBCs. TBCs exhibit time-dependent material response, which is incorporated into the model. Previous work has for simplicity used time-independent yielding to simulate the morphological instabilities. For comparison, we will conduct two groups of simulations, one group utilized time-independent material yield strengths while the other group used creep properties. This will guide future simulations regarding selection of properties.

Inputs for time- and temperature-dependent material behaviors in this finite element model were described previously in Chapter 2. Time- and temperature-

dependent TGO growth was described in Section 2.2.1. Temperature-dependent material properties were summarized in Section 2.2.2, while creep properties were described in Section 2.3.1. The Creep properties are taken from Ref. [6], with Eq. (3.1) for the TGO, and Eq. (3.2) for the bond coat.

$$\dot{\varepsilon} = 1.08 \cdot 10^{-10} \cdot \sigma^{2.3} \cdot \exp\left(\frac{-51000}{T}\right) , \text{ s}^{-1} \quad (3.1)$$

$$\dot{\varepsilon} = 8.96 \cdot 10^{-15} \cdot \sigma^{3.0} \cdot \exp\left(\frac{-35840}{T}\right) , \text{ s}^{-1} \quad (3.2)$$

3.3 Parametric Study with Time-independent Plasticity

Parameters such as maximum temperature during holding, ratio of lengthening strain over thickening strain, and yield strength for both bond coat and TGO have important effects on the evolution of the morphological instabilities [18,20].

The maximum temperature, T_H , governs the oxidation rate. The TGO growth laws depicted in Figure 2.6 show that increasing the holding temperature will accelerate the oxidation, which means higher the oxidation growth strain rate, thus resulting in faster deformation [18]. This has been verified by our numerical results as shown in Fig 3.4. The structure was subjected to 180 thermal cycles with 45 minutes of holding in each cycle, but with two selected holding temperature, $T_H = 1050$ °C and $T_H = 1100$ °C, for comparison. The results suggest that holding at 1100 °C gives significantly larger deformation than holding at 1050 °C.

The lengthening strain (lateral growth strain) has been investigated in previous work [26]. As defined in Section 1.3.2, lengthening strain, ε_g , is the oxidation induced strain in the lengthening direction, i.e. the direction along the structure interface. Thickening strain, ε_t , is the oxidation induced strain in the TGO thickness direction. For simplicity in the numerical model, lengthening strain rate was taken to be

proportional to the thickening strain rate [20]. β stands for the ratio of lengthening strain over thickening strain, thus $\beta = \varepsilon_t/\varepsilon_g$. Taking ε_t to be constant, Ref. [20] showed that increasing the ratio, i.e. increasing lengthening strain, results in increased deformation. This is due to the fact that increasing lengthening strain gives higher tangential stress, thus provides higher driving force towards morphological instabilities. This effect is also verified in our numerical model as shown in Figure 3.5. The figure shows that, holding at 1087 °C for 180 cycles with the same thickening strain rate, the higher β (which means higher lengthening strain) gives increased deformation.

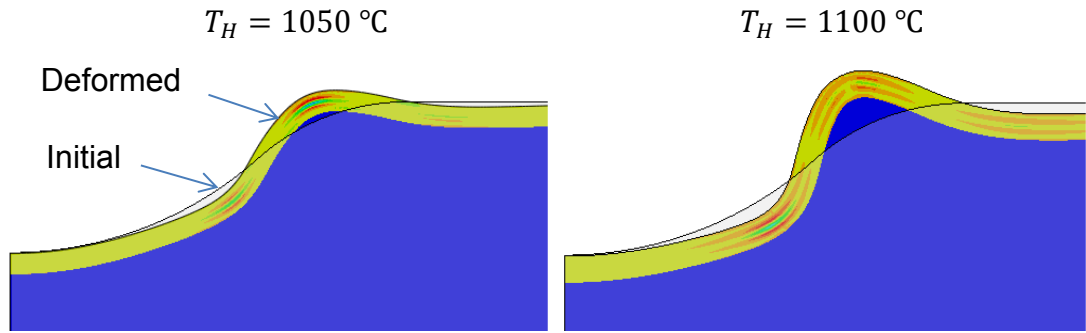


Figure 3.4 Initial and Deformed shape after 180 cycles with two selected T_H

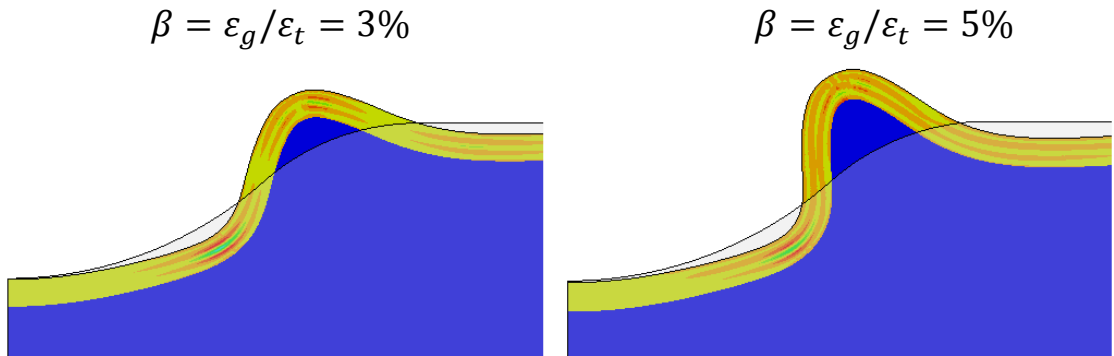


Figure 3.5 Deformation for two selected ratios, $\beta = \varepsilon_g/\varepsilon_t$, after 180 cycles with $T_H = 1087$ °C

We also investigated the effect of yield strength of the FeCrAlY and the TGO. Figure 3.6 shows that increasing yield strength for the bond coat enhances the resistance and thus results in decreased deformation. In contrast, increasing yielding strength for TGO allows higher accumulative driving force and thus gives increased deformation, as shown in Figure 3.7 from our numerical results.

As discussed previously (see Table 2.4), the yield strengths of FeCrAlY and TGO at high temperature are not available. Those unknown material parameters are investigated by a series of simulations. From our parametric study, a better match with experimental shape change occurs when the yield strength of FeCrAlY at temperature over 1000 °C is taken as 25 MPa constant, while the yield strength of TGO at holding temperature is taken as 1.2 GPa, which is consistent with previous study [18].

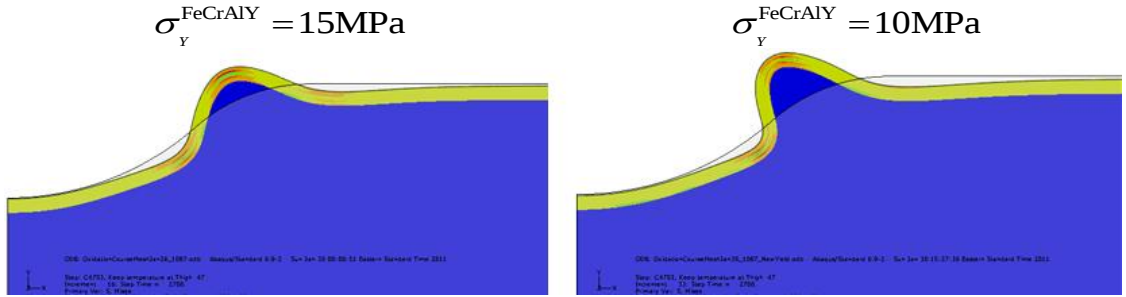


Figure 3.6 Deformation with two different yield strength for FeCrAlY with $T_H = 1087\text{ }^{\circ}\text{C}$

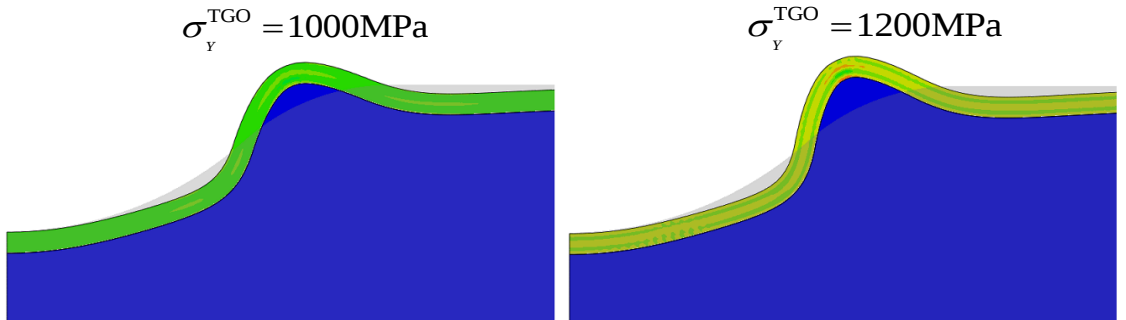


Figure 3.7 Deformation with two selected yield strength for TGO after 180 cycles with $T_H = 1087\text{ }^{\circ}\text{C}$

3.4 Reference Simulation Results and Comparisons

In this section, we will show our numerical work on capturing the two sets of experiments described in Section 1.4. Two groups of simulations are conducted for comparisons, one group with time-independent plastic properties, while the other group with time-dependent plasticity, i.e., creep.

The initial imperfection amplitude on the specimen is $20\text{ }\mu\text{m}$ as shown in Figure 1.8. The experimental details are described in Section 1.4. For the experiment with high temperature and long holding (HL experiment), the specimen was subjected to thermal cycles with 15 minutes for cooling and heating, and 45 minutes for holding at $1087\text{ }^{\circ}\text{C}$. The specimen was subjected to 176 cycles of cyclic loading. Due to a technical error, the specimen was then held at high temperature for another 19 hours in isothermal condition. The SEM image, Figure 3.8.a, shows the result for morphological change. The deformed amplitude, A , is about $26.7\text{ }\mu\text{m}$, and the TGO thickness, h , is about $3.0\text{ }\mu\text{m}$.

For the experiment with low temperature and short holding (LS experiment), the specimen was subjected to thermal cycles with 15 minutes for cooling and heating, and 18 minutes for holding at $1040\text{ }^{\circ}\text{C}$. Figure 3.9.a shows the SEM image of morphological change after 200 cycles of loading. The deformed amplitude is about $22.4\text{ }\mu\text{m}$, and the TGO thickness is about $1.60\text{ }\mu\text{m}$.

As previously discussed, the high temperature material properties are unknown. By conducting parametric studies, the unknown properties may be established. To this end, the material properties that give deformations closest to the experimentally obtained shape change are used. The plastic properties obtained are summarized in Tabel 3.1.

In the simulations with time-independent plastic behavior, in order to get a good match with the experimental results, the ratio of lengthening strain over thickening strain, $\beta = \varepsilon_g / \varepsilon_t$, was taken to be 5% for the HL experiment, but 8% for the LS experiment. The simulation result for the HL experiment is shown in Figure 3.8.b. The deformed amplitude is 26.52 μm and the TGO thickness is 3.09 μm . The simulation result for the LS experiment is shown in Figure 3.8.c. The deformed amplitude is 22.18 μm and the TGO thickness is 1.65 μm .

Table 3.1 Time-independent plasticity

FeCrAlY		TGO [1]	
Temp. (°C)	Yield Strength (MPa)	Temp. / °C	Yield Strength (MPa)
25	510	40	10000
600	185	1080 (1030)	10000
800	55	1085 (1035)	1200
1000	25	1087 (1040)	1200
1100	25		

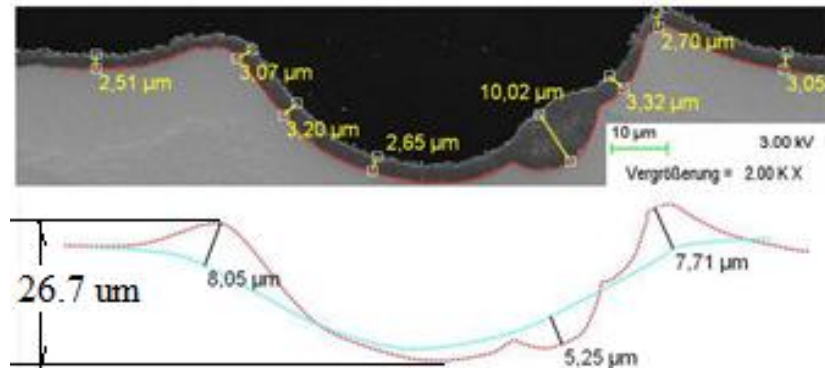
As an alternative to using time-independent yield strength, creep behavior were incorporated and investigated. Based on parametric studies, in order to get a good match with the experimental results, the lengthening oxidation strain was taken to be 2% of the thickening growth strain in this case. Figure 3.8.c and Figure 3.9.c show the simulation results for the HL and LS experiments respectively. For the HL experiment after 180 cycles, the simulation gives a deformed amplitude of 26.56 μm and a TGO thickness of 3.07 μm . For LS experiment after 200 cycles, the model shows a deformed amplitude of 22.63 μm and a TGO thickness of 1.64 μm .

Table 3.2 Deformed data for the HL and LS experiments

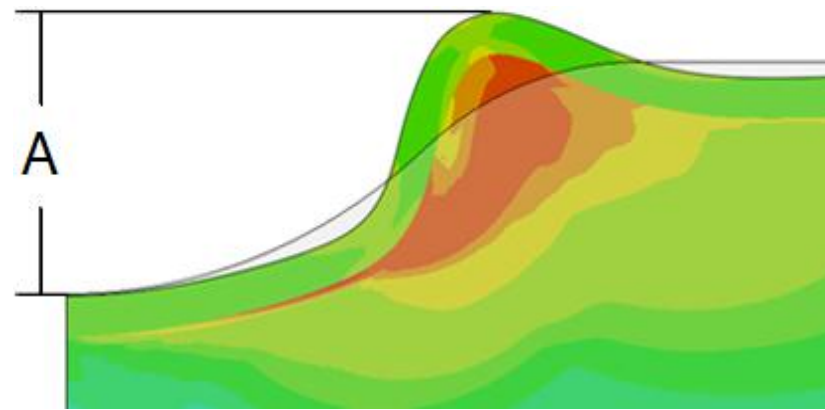
	HL experiment			LS experiment		
	Exp. results	Model with plasticity	Model with Creep	Exp. results	Model with plasticity	Model with creep
$A / \mu\text{m}$	26.70	26.52	26.56	22.42	22.18	22.63
$h / \mu\text{m}$	3.0	3.09	3.07	1.6	1.65	1.64

The TGO thickness, h , and deformed imperfection amplitudes, A , from both experiments and the respective simulations are also summarized in Table 3.2 for comparisons. We can see that the model with time-independent plasticity is able to capture the right amplitude change for the two experiments by adjusting the ratio β , i.e., the lengthening strain over the thickening strain. Using the creep properties, the model captures the right amplitude change well.

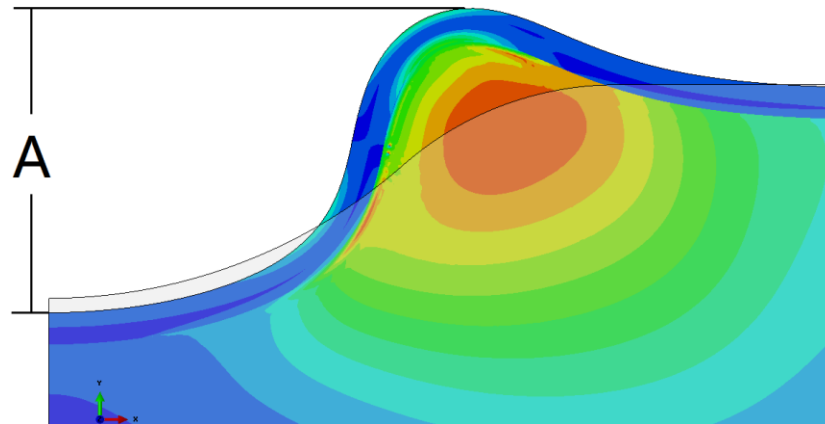
An additional parameter is the surface curvature of the imperfection. The comparisons of the surface deformations are shown in Figure 3.8 and Figure 3.9. We can see that model with creep behaviors gives a smooth shape change, which matches the experimental results better. Figure 3.8.b and Figure 3.9.b shows the contours of plastic strain field, while Figure 3.8.c and Figure 3.9.c show the contours of creep strain field. Evaluating the strain fields, we conclude the following reasons as to why creep behaviors help predict the shape change better. Overall creep in the bond coat around the imperfection area allows more smooth deformation. While in contrast, time-independent plasticity gives significant higher yield strain around the curvature transition point. We believe that this is the reason why there is a kink in the shape change around the curvature transition point. Thus, in order to get the correct shape change, imposing creep is necessary.



(a) SEM image from the HL experiment

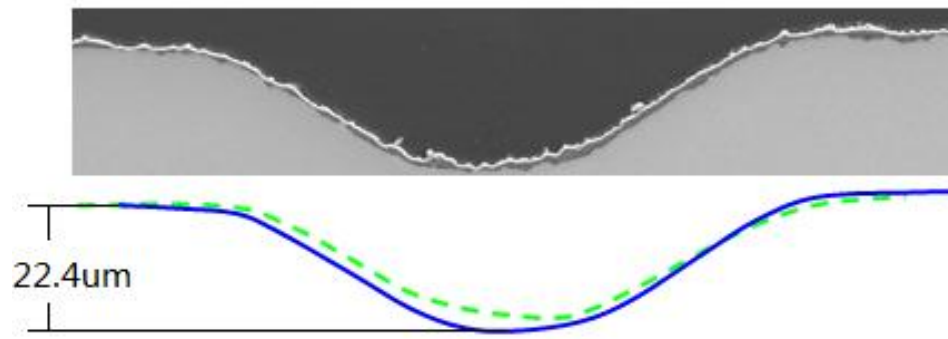


(b) Simulation results using time-independent plasticity

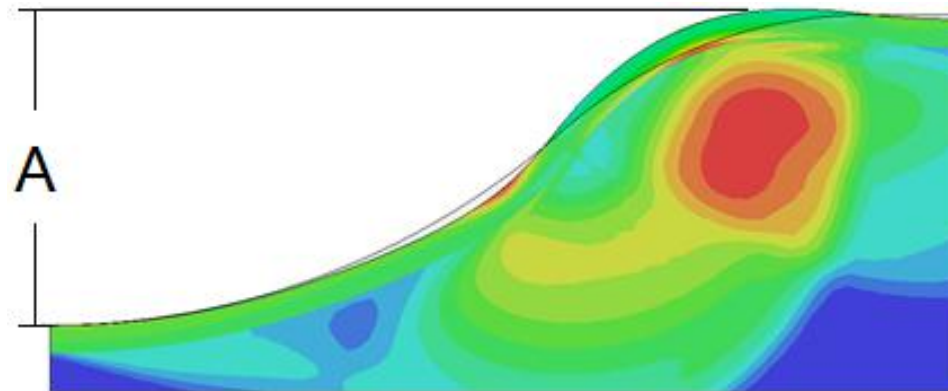


(c) Simulation results using creep properties

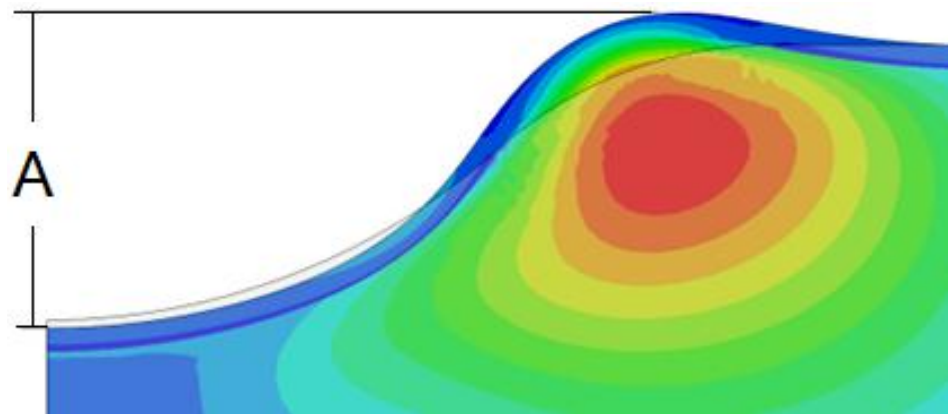
Figure 3.8 Morphological change from simulations compared to the experimental results, and contours of plastic/creep strain, for the HL experiment: (a) SEM from the HL experiment; (b) Simulation results using time-independent plasticity; (c) Simulation results using creep properties



(a) SEM image from the LS experiment



(b) Simulation results using time-independent plasticity



(c) Simulation results using creep properties

Figure 3.9 Morphological change from simulations compared to the experimental results, and contours of plastic/creep strain, for the LS experiment: (a) SEM from the LS experiment; (b) Simulation results using time-independent plasticity; (c) Simulation results using creep properties

Based on the two initial sets of experiments, we have not obtained a reliable TGO growth law. Thus, to this end, the TGO growth laws used are based on data from Ref. [18]. The experimental results may vary due to differences in chemical composition of the specimen. Future work must include a set of experiments to study the TGO growth evolution. Then further parametric study of lateral growth strain will be performed to capture the new experiments.

Never the less, our simulation results suggest that simulations with creep properties may be able to capture the amplitude change obtained experimentally and also provide the correct shape change.

3.5 Verification of the Cycle Jump Technique

In this section, we will investigate the performance of the cycle jump technique aiming to capture the preliminary experimental results. We will impose the cycle jump on simulation of the HL experiment (see Table 1.1) with time-independent inelasticity for performance evaluation. Cycle jump simulations with four selected q_y values (1.0, 2.0, 3.0, 5.0), the jump control parameter, will be compared. The results from cycle jump simulation results with $q_y = 3.0$ will be compared to that from the cycle-by-cycle reference simulation.

The idea of the cycle jump is that not all individual cycles need to be simulated. By establishing and extrapolating a general evolution of state variables, the simulation could skip, or “jump”, over some cycles. By making those jumps, computational efforts can be saved significantly so as to improve the simulation efficiency.

The model is as shown Figure 3.2, which consists of 21654 nodes and 6944 elements with the type of quadrilateral 8-node generalized plane strain elements with

reduced integration (CPEG8R). Imposing the jump control on displacement components at every node in the whole model is not practical. In fact, central to the idea of the cycle jump technique is to follow as few state variables as possible. Since there is significant morphological change occurring in the area around the imperfection, jump control will be applied only on the displacement components at the nodes on the TGO surface around the imperfection. The nodes where we applied the jump control are shown as dots in Figure 3.10.

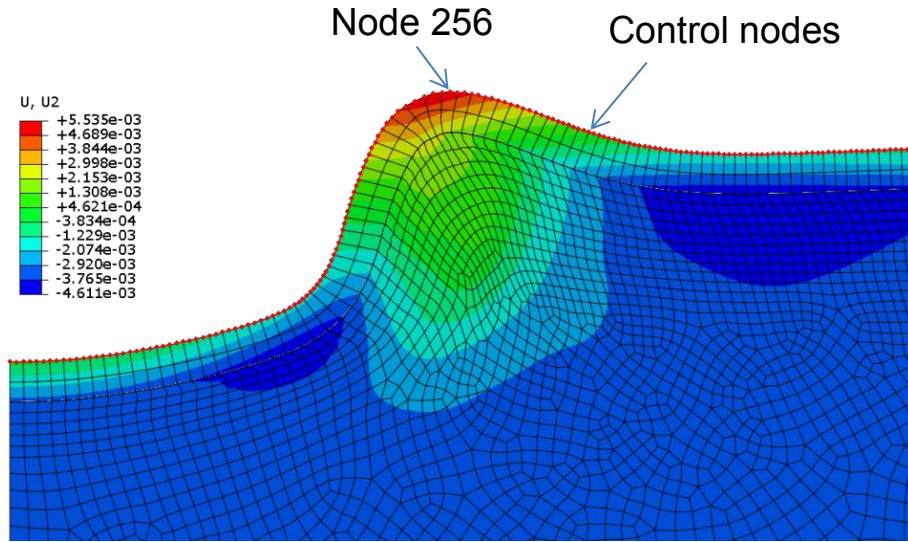


Figure 3.10 Nodes for jump control in simulation with the cycle jump technique, the largest vertical displacement occurs at Node 256 as shown

We will utilize the polynomial extrapolation scheme as discussed in Section 2.1. The savings of computation utilizing cycle jump with selected q_y is summarized in Table 3.3. The computational efficiency increases as q_y increases. This is consistent with the results obtained for the “basic model” simulating a cylinder, discussed in Section 2.2.3.

Table 3.3 Savings of computation with the cycle jump technique

q_y	Jumps	Saved cycles	Relative errors
1.0	11	6	2.9%
2.0	13	31	3.3%
3.0	12	41	3.3%
5.0	diverged		

The largest savings obtained for this case is with $q_y = 3.0$. For simulations of 200 cycles, by making 12 jumps, the cycle jump technique saves 41 cycles of computation in this case. Thus, the cycle jump technique with polynomial extrapolation saves 20.5% of computations with $q_y = 3.0$.

Figure 3.11 shows the comparison of state fields after 200 cycles between cycle jump simulations and the cycle-by-cycle reference simulation. From visual inspection, the state fields of tangential stress, creep strain, vertical displacement from simulation with cycle jump are very similar and close to the results from the cycle-by-cycle reference simulation. More insight is obtained by the evolution of the vertical displacement at one particular node, Node 256 (see Fig 3.9). This node is of particular interest because its vertical displacement is the largest and so it governs the deformation amplitude change in the model. As shown in Figure 3.12, comparing to the results from cycle jump to the cycle-by-cycle reference simulation, the relative error, E_r , is about 3.3% after 200 cycles. Thus, we can conclude that the simulation with the cycle jump technique is able to capture the overall behaviors.

In conclusion, the cycle jump technique saves about 20.5% in computational effort with a cost of about 3.3% in accuracy. Thus, the cycle jump works for this case even though it is modest in computational savings.

The reason why the cycle jump could not save computation significantly is that it is a complicated structure. Some relatively “linear regions” in the structure allow the cycle jumps while some other highly “non-linear regions” are prohibiting the jump attempts.

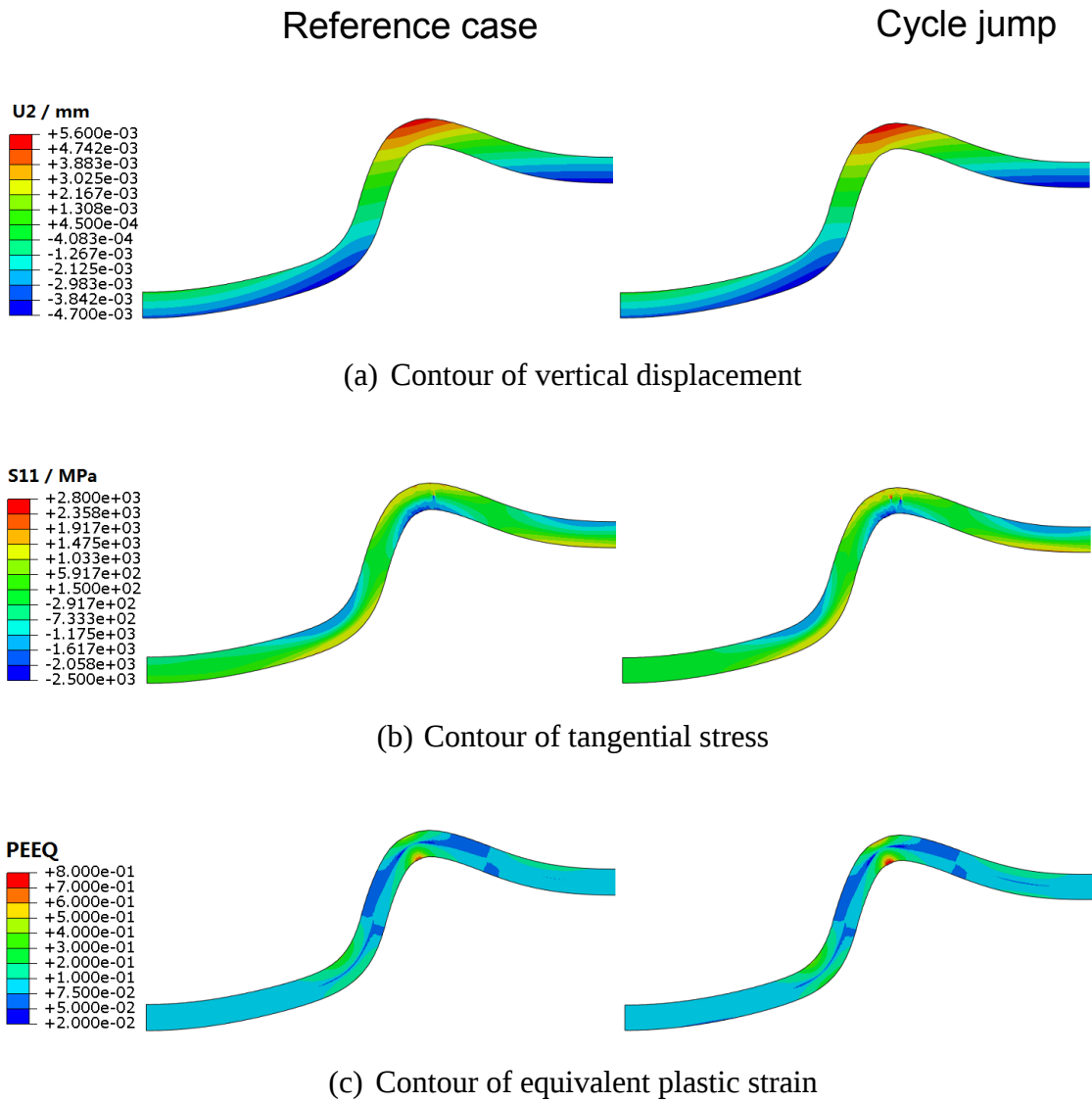


Figure 3.11 Contours of vertical displacement, tangential stress and equivalent plastic strain after 200 cycles, cycle jump with $q_y = 3.0$ compared to the cycle-by-cycle reference simulation

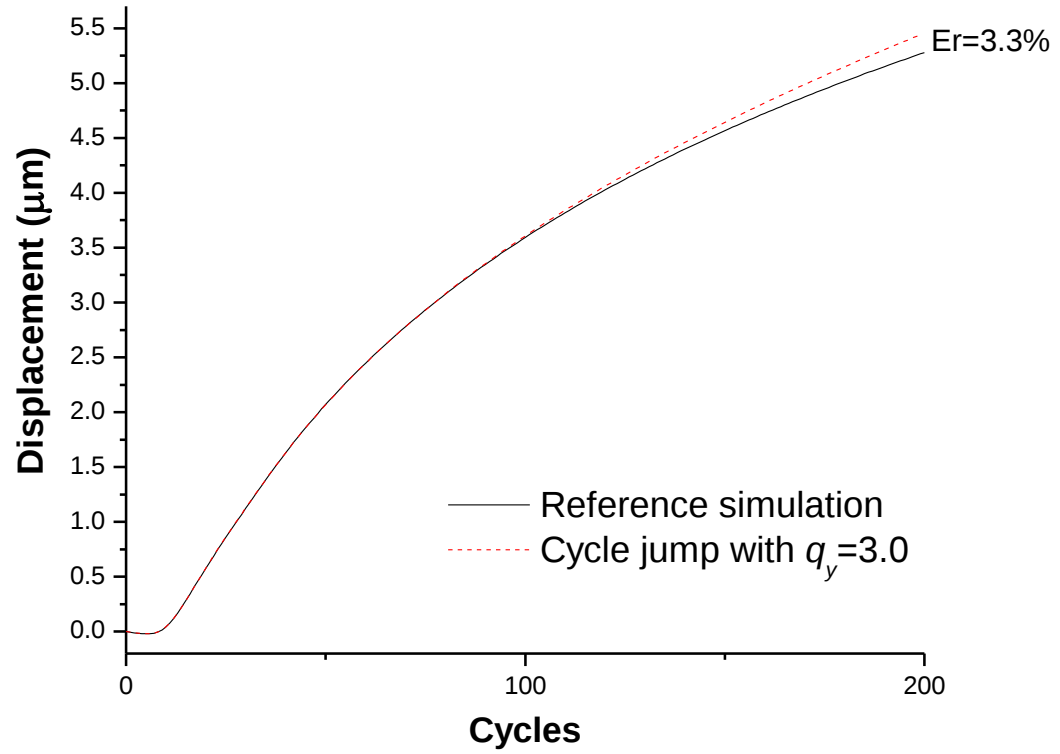


Figure 3.12 Evolution of vertical displacement at Node 256 (see Fig 3.10) in the model for 200 thermal cycles, cycle jump with $q_y = 3.0$ compared with the cycle-by-cycle reference simulation.

Chapter 4

SUMMARY AND FUTURE CONSIDERATIONS

4.1 Summary

Structures in engineering applications are commonly subjected to cyclic loadings, which may result in fatigue and unexpected failures. Understanding of the failure evolution is important to obtain reliable engineering designs. Finite element analysis (FEA) is commonly used to simulate and establish the stress and strain distribution for failure predictions. However, it is time-consuming to simulate the whole process of structural evolution. This work aims to verify and enhance a previously developed numerical technique, the “cycle jump technique” [1], which enables accelerated numerical simulations for structures subjected to cyclic loadings.

Although some studies have been conducted on accelerating numerical simulations, the existing methods [5-15] found in literature have some limitations. The “cycle-jump technique”, previously developed by Cojocaru and Karlsson [1], is a general method to improve the computational efficiency. The idea is that not all individual cycles need to be simulated. By establishing and extrapolating a general evolution of state variables, the simulation can skip, or “jump”, over some cycles. By making those jumps, computational efforts can be saved significantly so as to improve the simulation efficiency. This is a promising technique, and it is relatively easy to be incorporated into commercial finite element software, such as ABAQUS [2-4] (which

is used in this work) and ANSYS. However, to become a reliable tool, the “cycle jump technique” still needs to be improved and its validity needs to be verified.

The thermal barrier coatings (TBCs) system was introduced, since TBCs inspired this work. The TBC systems with corresponding loading conditions served as application examples for the cycle jump technique.

To overcome the limitation of the existing extrapolation scheme ‘linear extrapolation’, a more general extrapolation scheme ‘polynomial extrapolation’ was proposed. Using the basic cylinder model [1], the ‘polynomial extrapolation’ method was first verified and demonstrations showed that it can significantly save computations with little cost in accuracy.

To be used for simulating real-life experiments, both time- and temperature-dependent behaviors (including high temperature time-independent plasticity, time- and temperature-dependent oxidation growth, and time-dependent plasticity i.e. creep) were incorporated into the numerical scheme for the cycle jump technique. Based on a basic cylinder model, the improved code is verified numerically by comparing the results from the cycle jump technique to the cycle-by-cycle reference simulation and excellent agreement is obtained.

To verify the validity of the cycle jump technique that it can capture the experimental observation and to demonstrate the power of the method in computation savings, experimental results, the cycle-by-cycle reference simulation, and the simulation with cycle jump must be compared. Thus, we proposed investigations of morphological instability on FeCrAlY with a grooved surface subjected to cyclic loadings. With a numerical model we developed based on geometry and material properties of the experimental specimen, numerical simulations were conducted and

several groups of parameter settings were suggested to be used in the experiments aimed to be used as an example for verifying the “cycle jump technique”. Initially, 2 sets of experiments were conducted and preliminary experimental results were obtained by our collaborators, Dzodzovic and Bartsch at German Aerospace Center (DLR) in Cologne, Germany.

To capture the experimentally obtained morphological change as a part of our numerical work, two groups of numerical simulations were conducted and compared. For the high-temperature materials, one group used time-independent yielding, while the other group used time-dependent plasticity, i.e. creep. In order to obtain the experimental results, the unknown material properties were established from parametric studies. After obtaining the results of the experiment, the numerical results are compared with the experimental results. Our results suggested that simulations using creep behavior capture the morphological instabilities better than simulations using time-independent plasticity.

Finally, the validity of the cycle jump technique was verified. In conclusion, the simulations utilizing “cycle jump technique” captured the experimental results while improving the computational efficiency over the cycle-by-cycle simulations, even though the savings were not so significant.

4.2 Future Considerations

Based on the two initial experiments, we have not obtained a reliable TGO growth law. Thus the growth laws used are based on data from literature. The experimental results may vary due to the differences in chemical composition of the specimens. Future work must include a set of experiments to study the TGO growth

evolution. Once reliable TGO growth evolution is obtained, the cycle jump technique will be utilized in simulating the new experiments for further performance evaluation.

To this end, the cycle jump technique works well in capturing the real life experiments. However the savings in computation is not significant. Subroutines that are more advanced in making efficient jumps thus saving computations could be used. For example, the concept of substructures can be utilized to separate the whole model domain. When cycle jump on the overall structure is not allowed by the jump control criteria, we can still do cycle jump on the relatively “linear region” of the structure while cycle-by-cycle simulation is executed on the relatively “non-linear region”, so as to further enhance the simulation efficiency.

REFERENCES

- [1] D. Cojocaru and A.M. Karlsson, A simple numerical method of cycle jumps for cyclically loaded structures, *Int J Fatigue* (2006), pp. 1677–1689.
- [2] ABAQUS, ABAQUS 6.9, ABAQUS Inc., Pawtucket, Rhode Island, 2009
- [3] ABAQUS 6.9 Scripting User's Manual. 2009, Providence, RI: ABAQUS, Inc.
- [4] ABAQUS 6.9 Scripting Reference Manual. 2009, Providence, RI: ABAQUS, Inc.
- [5] P. Ladeveze, New algorithms: mechanical framework and development. Technical Report 57, LMT-Cachan, 1985
- [6] P. Ladeveze, On a family of algorithms for structural mechanics, *CR Acad Sci Paris* 300(2):41-44, 1985
- [7] P. Boisse, P. Bussy and P. Ladeveze, A new approach in nonlinear mechanics – the large time increment method, *Int J Numer Meth Eng* (1990), pp. 647–663.
- [8] J.Y. Cognard and P. Ladeveze, A large time increment approach for cyclic viscoplasticity, *International Journal of plasticity* (1993), pp. 141-157.
- [9] J.Y. Cognard, P. Ladeveze and P. Talbot, A large time increment approach for thermo-mechanical problems, *Adv Eng Software* (1999), pp. 583–593.
- [10] J. Fish and Q. Yu, Computational mechanics of fatigue and life predictions for composite materials and structures, *Comput Meth Appl Mech Eng* (2002), pp. 4827–4849.
- [11] C. Oskay and J. Fish, Fatigue life prediction using 2-scale temporal asymptotic homogenization, *Int J Numer Meth Eng* (2004), pp. 329–359.
- [12] H. Kiewel, J. Aktaa and D. Munz, Application of an extrapolation method in thermocyclic failure analysis, *Comput Meth Appl Mech Eng* (2000), pp. 55–71.
- [13] P. Wang, L. Cui, M. Lyschik, A. Scholz, C. Berger, and M. Oechsner, A local extrapolation based calculation reduction method for the application of constitutive

- material models for creep fatigue assessment. *International Journal of Fatigue*, (2012), pp253–259.
- [14] F. Bogard, P. Lestriez, and Y.Q.Guo, Numerical modeling of fatigue damage and fissure propagation under cyclic loadings, *International Journal of Damage Mechanics* (2008), p 173–187
- [15] R. Moslemian, A.M. Karlsson and C. Berggreen, *International Journal of Fatigue*, 33 (2011), p1526–1532.
- [16] T.E. Strangman, Thermal Barrier Coatings for Turbine Airfoils, *Thin Solid Films* (1985), pp. 93–105.
- [17] A. M. Karlsson's personal website:
http://research.me.udel.edu/karlsson/AK_TBC.html
- [18] V.K.Tolpygo and D.R.Clarke, Competition between stress generation and relaxation during oxidation of an Fe-Cr-Al-Y alloy, *Oxidation of Metals*, Vol. 49, Nos. 1/2, 1998
- [19] A. M. Karlsson, J. W. Hutchinson and A. G. Evans, A fundamental model of cyclic instabilities in thermal barrier systems, *J. Mech. Phys. Solids*. 50 (2002) 1565–1589.
- [20] A.M. Karlsson, C.G. Levi, and A.G. Evans, A model study of displacement instability during cyclic oxidation, *Acta Materialia* 50 (2002) 1263-1273
- [21] B. Baufeld, M. Bartsch, S. Dalkilic, M. Heinzelmann *Surf. Coat. Technol.*, 200 (2005), p. 1282
- [22] Material data sheet from material supplier
- [23] Evans HE, Strawbridge A, Carolan RA, Ponton CB. Creep effects on the spallation of an alumina layer from a NiCrAlY coating. *Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing* 1997;225:1
- [24] Saunders S.R.J. , Evans HE, M. Li, D.D. Gohil, and S. Osgerby. Oxidation Growth Stress in an alumina-forming ferritic steel Measured by creep deflection. *Oxidation of Metals*, Vol. 48 189-200

- [25] H. Echsler, E. Alija Martinez, L. Singher, W. J. Quakkers. Residual stress in alumina scales grown on different types of Fe-Cr-Al alloys: effect of specimen geometry and cooling rate. *Materials Science and Engineering A* 384 (2004) 1-11.
- [26] D.R. Clarke, The lateral growth strain accompanying the formation of a thermal grown oxide, *Acta Materialia* 51 (2003) 1393-1407

Appendix A

UEXPAN

```
c user subroutine uexpan

      subroutine uexpan(expan,dexpandt,temp,time,dtime,predef,dpred,
$      statev,cmname,nstatv,noel)
c
c      include 'aba_param.inc'
c
      character*80 cmname
      real(8)::MT      !step time for initialization after jump
      real(8)::GTIME    !cycle length (time)
      real(8)::GPERIOD  !holding time each cycle for TGO growth
      real(8)::TTAJ     !historic time (total time after jump)
      COMMON /GROWTH/ MT,GTIME,GPERIOD,TTAJ

      double precision t0,k,n,t1,t2,Et,Eg
      integer CycleNb   ! number of cycles completed

c
      dimension expan(*),dexpandt(*),temp(2),time(2),predef(*),
$      dpred(*),statev(nstatv)
c
c      Define expansion for FeCrAlY
c      DO be careful, USE Capital Letter for material
c
      IF (CMNAME(1:8).EQ.'BONDCOAT') THEN
c          thermal expansion coefficient
          expCoef=3.2E-9*(TEMP(1)-TEMP(2)/2.0) + 11.73E-6
          thestr=expCoef*temp(2)      ! thermal strain
          expan(1) = thestr
          expan(2) = thestr
          expan(3) = thestr
      ENDIF

c
c      Define expansion for TGO
c      Calculate the thickening rate
c
      t0=0.72863  ! time shift for initial TGO thickness
      k=0.564     ! parameter k in TGO growth law
      n=0.38      ! n in TGO growth law

c
      IF (CMNAME(1:6).EQ.'NEWTGO') THEN
c          thermal expansion coefficient
          expCoef=1.0E-9*(TEMP(1)-TEMP(2)/2.0) + 7.5E-6
```

```

        thestr=expCoef*temp(2)
        expan(1) = thestr
        expan(2) = thestr
        expan(3) = thestr

C
C      Define the TGO growth rate
C
C      compute the periodic time before this increment
        CRGTIME=MOD(T-TOLERANCE,GTIME)

C
C      Impose TGO growth
        IF ((GTIME-GPERIOD) < CRGTIME)
            .AND. (CRGTIME < GTIME)) THEN
! time(1): step time after this increment
! time(2): total time after this increment
! CycleNb: cycles completed
C
C calculate how many cycle completed
        CycleNb=int((T+TTAJ-TOLERANCE)/GTIME)
C t2= how many hrs at hight Temp. after this increment
        t2=t0+CycleNb*GrowthTime/3600.0+time(1)/3600.0
        h2=k*t2**n      C thickness after this increment

C
C t1= how many hrs at hight Temp. before this increment
        t1=t2-DTIME/3600.0
        h1=k*t1**n      ! thickness at before this increment
        Et=log(h2/h1)   ! incremental thickening strain
        Eg=beta*Et      ! incremental lengthening strain
        expan(1)=Eg      ! lengthening dir
        expan(2)=Et      ! thickening dir
        expan(3)=Eg      ! out of plane dir

        ENDIF

C
        ENDIF

C
        RETURN
        END

```

Appendix B

UMAT

Note: A UMAT subroutine for creep behavior. The UMAT for TGO and Bond Coat is similar. Thus only UMAT for TGO is shown here.

```

SUBROUTINE UMAT (STRESS, STATEV, DDSDE, SSE, SPD, SCD,
1 RPL, DDSDDT, DRPLDE, DRPLDT,
2 STRAN, DSTRAN, TIME, DTIME, TEMP, DTEMP, PREDEF, DPRED, CMNAME,
3 NDI, NSHR, NTENS, NSTATV, PROPS, NPROPS, COORDS, DROT, PNEWDT,
4 CELENT, DFGRD0, DFGRD1, NOEL, NPT, LAYER, KSPT, KSTEP, KINC)

C
  INCLUDE 'ABA_PARAM.INC'

C
  CHARACTER*80 CMNAME
  DIMENSION STRESS (NTENS), STATEV (NSTATV), DDSDE (NTENS, NTENS),
1 DDSDDT (NTENS), DRPLDE (NTENS), STRAN (NTENS), DSTRAN (NTENS),
2 TIME (2), PREDEF (1), DPRED (1), PROPS (NPROPS), COORDS (3), DROT (3, 3),
3 DFGRD0 (3, 3), DFGRD1 (3, 3)

C
  IF (CMNAME (1:6) .EQ. 'NEWTGO') THEN

C
    CALL UMAT_NEWTGO (STRESS, STATEV, DDSDE, SSE, SPD, SCD,
1 RPL, DDSDDT, DRPLDE, DRPLDT,
2 STRAN, DSTRAN, TIME, DTIME, TEMP, DTEMP, PREDEF, DPRED, CMNAME,
3 NDI, NSHR, NTENS, NSTATV, PROPS, NPROPS, COORDS, DROT, PNEWDT,
4 CELENT, DFGRD0, DFGRD1, NOEL, NPT, LAYER, KSPT, KSTEP, KINC)

C
  ELSE IF (CMNAME (1:8) .EQ. 'BONDCOAT') THEN

C
    CALL UMAT_BONDCOAT (STRESS, STATEV, DDSDE, SSE, SPD, SCD,
1 RPL, DDSDDT, DRPLDE, DRPLDT,
2 STRAN, DSTRAN, TIME, DTIME, TEMP, DTEMP, PREDEF, DPRED, CMNAME,
3 NDI, NSHR, NTENS, NSTATV, PROPS, NPROPS, COORDS, DROT, PNEWDT,
4 CELENT, DFGRD0, DFGRD1, NOEL, NPT, LAYER, KSPT, KSTEP, KINC)

C
  END IF

C
  RETURN
END SUBROUTINE

C
  HERE STARTS THE SUBROUTINES

SUBROUTINE UMAT_NEWTGO (STRESS, STATEV, DDSDE, SSE, SPD, SCD,
1 RPL, DDSDDT, DRPLDE, DRPLDT,
```

```

2 STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PRED,DPRED,CMNAME,
3 NDI,NSHR,NTENS,NSTATV,PROPS,NPROPS,COORDS,DROT,PNEWDT,
4 CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,KSTEP,KINC)

C
  INCLUDE 'ABA_PARAM.INC'

C
  UMAT FOR NEWTGO

C
  CHARACTER*80 CMNAME
  DIMENSION STRESS(NTENS),STATEV(NSTATV),DDSDDE(NTENS,NTENS),
1 DDSDDT(NTENS),DRPLDE(NTENS),STRAN(NTENS),DSTRAN(NTENS),
2 PRED(1),DPRED(1),PROPS(NPROPS),COORDS(3),DROT(3,3),
3 DFGRD0(3,3),DFGRD1(3,3),TIME(2),DCRSTRAN(NTENS)

C  HERE COMES THE CODE FOR UMAT_NEWTGO
C
  REAL(8)::MT
  REAL(8)::GTIME
  REAL(8)::GPERIOD
C  REAL(8)::TTAJ
  REAL(8)::TOLERANCE=0.00005D0
  COMMON /GROWTH/ MT,GTIME,GPERIOD      !,TTAJ

  REAL(8) CEEQ,DCEEQ
  DIMENSION EELAS(6),EELASR(6),ECREEPR(6),ECREEP(6),FLOW(6)
  PARAMETER (ZERO=0.D0,ONE=1.D0,TWO=2.D0,THREE=3.D0,SIX=6.D0,
1 ENUMAX=.4999D0,NEWTON=10,TOLER=1.0D-6)

C
C  -----
C    UMAT FOR ISOTROPIC ELASTICITY AND CREEP
C    CAN NOT BE USED FOR PLANE STRESS
C  -----
C    PROPS(1) - E
C    PROPS(2) - NU
C    PROPS(3) - SYIELD
C    CALLS AHARD FOR CURVE OF SYIELD VS. PEEQ
C  -----
C
  IF (NDI.NE.3) THEN
    WRITE(6,1)
1    FORMAT(//,30X,'***ERROR - THIS UMAT MAY ONLY BE USED FOR ',
1      'ELEMENTS WITH THREE DIRECT STRESS COMPONENTS')
    ENDIF

C
C  ELASTIC PROPERTIES
C
  EMOD=PROPS(1)
  ENU=PROPS(2)
  TTA=1.08*exp(-51000.0/((TEMP-DTEMP/2.0)+273.15))*10**3.8
  TTN=2.3
  TTM=0.0
C

```

```

IF (ENU.GT.0.4999.AND.ENU.LT.0.5001) ENU=0.499
EBULK3=EMOD/ (ONE-TWO*ENU)
EG2=EMOD/ (ONE+ENU)
EG=EG2/TWO
EG3=THREE*EG2/TWO
ELAM= (EBULK3-EG2) /THREE

C
C   update current time to be considered for growth
T=time (2)+DTIME-MT
CRGTIME=MOD (T-TOLERANCE,GTIME)

C
C   CREEP steps
DO K1=1,NTENS
    DO K2=1,NTENS
        DDSDE (K1,K2)=ZERO
    END DO
END DO
DO K1=1,NDI
    DO K2=1,NDI
        DDSDE (K2,K1)=ELAM
    END DO
    DDSDE (K1,K1)=EG2+ELAM
END DO
DO K1=NDI+1,NTENS
    DDSDE (K1,K1)=EG
END DO

C   RECOVER ELASTIC AND creep STRAINS
DO K1=1,NTENS
    EELAS (K1)=STATEV (K1)
    ECREEP (K1)=STATEV (K1+NTENS)
END DO

C   ROTATE THE RECOVERED STRAINS BECAUSE THE USE OF NLGEOM
CALL ROTSIG (EELAS,DROT,EELASR,2,NDI,NSHR)
CALL ROTSIG (ECREEP,DROT,ECREEPR,2,NDI,NSHR)

C
DO K1=1,NTENS
    EELAS (K1)=EELASR (K1)+DSTRAN (K1)
    ECREEP (K1)=ECREEPR (K1)
END DO

C   RECOVER THE EQUIVALENT CREEP STRAIN
CEEQ=STATEV (1+2*NTENS)

C   WRITE (6,*),"EELAS->",EELAS
C   WRITE (6,*),"ECREEP->",ECREEP
C   WRITE (6,*),"EELASR->",EELASR
C   WRITE (6,*),"ECREEPR->",ECREEPR
C   WRITE (6,*),"DROT->",DROT

C   MISES STRESS
C
SMISES= (STRESS (1)-STRESS (2)) * (STRESS (1)-STRESS (2)) +

```

```

1          (STRESS (2) -STRESS (3)) * (STRESS (2) -STRESS (3)) +
1          (STRESS (3) -STRESS (1)) * (STRESS (3) -STRESS (1))
DO K1=NDI+1, NTENS
    SMISES=SMISES+SIX*STRESS (K1) *STRESS (K1)
END DO
SMISES=SQRT (SMISES/TWO)

C
SHYDRO= (STRESS (1) +STRESS (2) +STRESS (3)) /THREE
RSTRAN1=THREE/TWO*TTA*SMISES** (TTN-1.0) * (STRESS (1) -SHYDRO)
RSTRAN2=THREE/TWO*TTA*SMISES** (TTN-1.0) * (STRESS (2) -SHYDRO)
RSTRAN3=THREE/TWO*TTA*SMISES** (TTN-1.0) * (STRESS (3) -SHYDRO)
RSTRAN4=THREE*TTA*SMISES** (TTN-1.0) *STRESS (4)
RSTRAN5=THREE*TTA*SMISES** (TTN-1.0) *STRESS (5)
RSTRAN6=THREE*TTA*SMISES** (TTN-1.0) *STRESS (6)

C
DCRSTRAN (1) =RSTRAN1*DTIME
DCRSTRAN (2) =RSTRAN2*DTIME
DCRSTRAN (3) =RSTRAN3*DTIME
DCRSTRAN (4) =RSTRAN4*DTIME
DCRSTRAN (5) =RSTRAN5*DTIME
DCRSTRAN (6) =RSTRAN6*DTIME

C
C UPDATE STRAINS
C
DO K1=1, 4
    ECREEP (K1) =ECREEP (K1) +DCRSTRAN (K1)
    EELAS (K1) =EELAS (K1) -DCRSTRAN (K1)
END DO

C
DO K1=1, 4
    STATEV (K1) =EELAS (K1)
    STATEV (K1+4) =ECREEP (K1)
END DO

DCEEQ=SQRT (2.0/3.0* (DCRSTRAN (1) *DCRSTRAN (1)
$      +DCRSTRAN (2) *DCRSTRAN (2) +DCRSTRAN (3) *DCRSTRAN (3)
$      +0.5*DCRSTRAN (4) *DCRSTRAN (4) ) )
CEEQ=CEEQ+DCEEQ
STATEV (1+2*NTENS) =CEEQ

C CALCUALTE UPDATED STRESS
DO K1=1, NTENS
    DO K2=1, NTENS
        STRESS (K2) =STRESS (K2)
$      +DDSDDE (K2, K1) * (DSTRAN (K1) -DCRSTRAN (K1) )
    END DO
END DO

C
C
RETURN
END SUBROUTINE

```