

A System-Level Approach to Fault Progression Analysis in Complex Engineering Systems

Manzar Abbas¹, and George J. Vachtsevanos¹

¹ School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332, USA
manzar.abbas@gatech.edu
george.vachtsevanos@ece.gatech.edu

ABSTRACT

Complex engineering systems consist of many subsystems. Each of the subsystems is composed of a large number of components. While faults arise at component level, sensing capabilities are limited to subsystem level, and system operations and maintenance practices are scheduled based on system level parameters. This paper presents a hierarchical architecture to analyze the effects of system level parameters on component level faults of dominant failure modes of a complex system. An aeropropulsion system of turbofan type has been used as the application domain. In most of the cases, engine life is limited due to cracks in high-pressure turbine blades. In this paper, it is assumed that creep is the only active failure mechanism. Based on a finite-element model of the turbine blades available in the open literature, design of experiments (DoE) methodology is used to build a subsystem-level model. A simulation package of a commercial aircraft engine is then used to obtain system-level results.

1 INTRODUCTION

Condition-based maintenance (CBM) and prognostics and health management (PHM) technologies aim at improving the availability, reliability, maintainability, and safety of systems through development of fault diagnostics and failure prognostic algorithms. Failure prognostics has been approached via a variety of techniques ranging from probabilistic/statistical methods to artificial intelligent tools. In general, prognosis algorithm approaches can be categorized broadly into model-based (Saha *et al.*, 2007; Orchard *et al.*, 2005; Saha *et al.*, 2009) and/or data-driven (Heimes, 2008; Wang *et al.*, 2008). Most of the CBM/PHM methodologies, whether model-based or data-driven, have been limited to component/subsystem level. System-level approaches, such as model-based reasoning

This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

(MBR), have been used to generate fault propagation models (Saha, 2007). Although MBR methodologies are successful in identifying faulty system components, yet they lack the ability to predict the remaining useful life (RUL) of the system. Moreover, these approaches are based on qualitative reasoning tools rather than the actual physics of the failure mechanisms.

This paper aims at developing a system-level prognostic approach, which can assist in making decisions about successful mission planning and optimum maintenance practices. It is obvious that this decision making process must identify potential failure of a component/subsystem, determine the effects of this failure, and identify actions that can eliminate or reduce the likelihood of potential failures to occur. The standard failure modes and effects analysis (FMEA), and failure modes, effects, and criticality analysis (FMECA) are based on these ideas (Bowles, 2003). However, the FMEA and FMECA procedures do not identify the product failure mechanisms, and thus have a limited applicability to provide a meaningful input to critical procedures, such as root cause analysis, accelerated test programs, and remaining life assessment (Ganesan *et al.*, 2005).

Failure modes, mechanisms, and effects analysis (FMMEA) is a methodology that has been developed to address weaknesses in the traditional FMEA and FMECA processes (Mathew *et al.*, 2008). The purpose of FMMEA is to identify potential failure mechanisms and models for all potential failures modes, and to prioritize failure mechanisms. FMMEA as proposed in (Mathew *et al.*, 2008), investigates failure mechanisms of each component in isolation. It does not consider the effects of system-level operating parameters on the failure mechanisms, and hence is limited upto component/subsystem-level.

In this research, an aeropropulsion system of turbofan type has been used as the application domain. In most of the cases, engine failure occurs as a result of HPT blade cracks. In HPT turbine blade, there exists two dominant failure mechanisms, i.e., creep and fatigue. In this paper, it is assumed that creep is the only active mechanism. Based on the finite element (FE)

models of the blades (made of GTD-111 material) available in the open literature, design of experiments (DoE) methodology is used to build a subsystem-level model. A simulation package of a commercial aircraft engine is then used to obtain system-level results.

Before discussing the methodology, various terms are explained in the following section to avoid any confusion arising due to use of the terminology. The methodology is based on several models which are arranged in a hierarchical architecture presented in section III. Then, the application domain is introduced in section IV, and the results are discussed in section V.

2 BACKGROUND

2.1 Fault Progression Vs. Fault Propagation

When a fault condition arises in one of the components/subsystems, it may result in change in the operating conditions of the neighboring subsystems, and hence spreading its effects to those subsystems. To distinguish the fault evolution within a subsystem from the spreading of effects of the fault to other subsystems, hereafter, the former is referred to as fault progression/evolution and the latter as fault propagation. Fault progression modeling aims at estimating evolution of a fault in a subsystem under given operating conditions/usage pattern, while fault propagation modeling estimates the effect of a fault on another fault, both in the same system.

This paper presents an architecture which is applicable to both fault progression and fault propagation. However, the results being presented are limited to an example of fault progression.

2.2 Operating Cycle/Drive Cycle/Mission Profile

In different systems, these terms have been used interchangeably, identifying the same parameter. For example, in the case of an aircraft, the often used term is *mission profile*, while in case of ground vehicle, the prevalent term is *operating cycle/drive cycle*. However in each case, these terms mean the same, i.e., a time sequence of system-level operating conditions during an entire mission or cycle. For example, the operating cycle of a gas turbine engine being used in a commercial aircraft consists of following sequence of operating modes.

1. Takeoff
2. Climb
3. Cruise
4. Descent
5. Landing

During each operating mode, system-level operating conditions will change as a time sequence vector. This statement will be explained in the subsequent paragraphs.

2.3 System-Level Operating Conditions

System-level operating conditions are a combination of system ambient conditions and operator settings. For example, in the case of a commercial aircraft, system-level operating conditions consist of:

1. Altitude
2. Mach
3. Ambient temperature
4. Throttle resolver angle

Altitude, mach, and ambient temperature come from system ambient conditions while throttle resolver angle comes from operator settings.

2.4 Subsystem-Level Operating Conditions

System-level operating conditions are subsequently translated into subsystem-level operating conditions. For example, a typical gas turbine engine (system) consists of following subsystems: fan, low-pressure compressor (LPC), high-pressure compressor (HPC), combustor, high-pressure turbine (HPT), and low-pressure turbine (LPT). In this case, there will be subsystem-level operating conditions for each of these subsystems, as shown in Fig. 1. For example, subsystem-level operating conditions of HPT consist of:

- gas temperature ($T_{g,HPT}$)
- gas pressure ($P_{g,HPT}$)
- gas flow ($W_{g,HPT}$)
- turbine rotational speed (N_{HPT})
- torque developed by turbine (τ_{HPT})

2.5 Load Conditions

The term "load" has been used in engineering literature in different contexts. Sometimes, stress and load are used interchangeably. In the field of electrical engineering, load refers to the amount of current being supplied by the system. Mechanical engineers, when using the term load, are mostly implying the torque being applied. Since the objective of this work is to develop a unified methodology, which is applicable to systems belonging to different domains, the terminology was defined in such a way that can be used across different domains while avoiding confusion. Fig. 1 shows that load conditions act as an intermediate layer between subsystem-level operating conditions and stress conditions. In the subsequent work, load will be referred to as a subset of subsystem-level operating conditions, which are responsible for generating stresses inside that subsystem.

For example, in a gas turbine, three types of stresses are: thermal stress (σ_T), centrifugal stress (σ_w), and bending stress (σ_p). These stresses are being generated by the following subsystem-level operating conditions: gas temperature (T_g), turbine rotational speed (N), and gas pressure (P) respectively. Similarly, windings of an

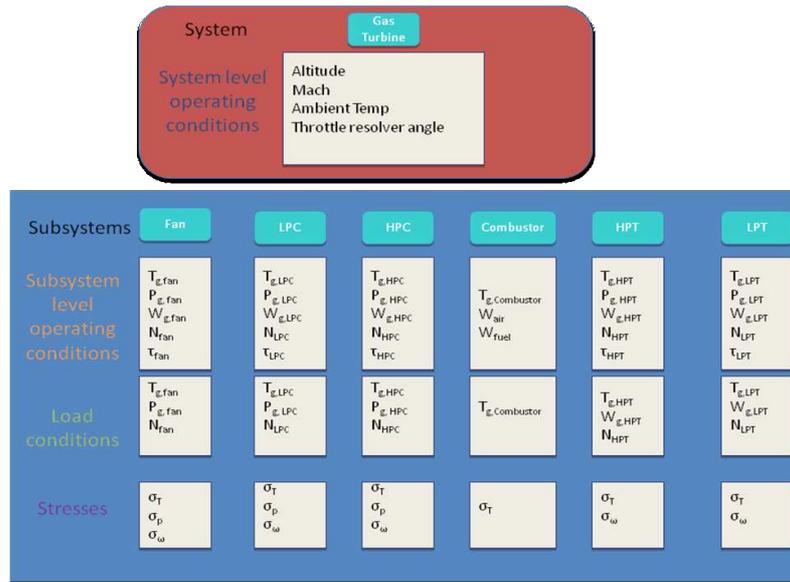


Figure 1: Hierarchical structure of the terminology used in the proposed approach.

electromechanical system deteriorate under the effect of thermal stress (σ_T), which is generated by the temperature being applied on the windings. Thus, in this case, temperature is included in the category of load variables since it is generating thermal stress, which is driving the failure mechanism. Similarly, in the case of an electrochemical system, grid corrosion is one of the dominant failure mechanism. There are two type of stresses controlling this failure mechanism; chemical stress and thermal stress. Chemical stress is generated by acid concentration and thermal stress by system temperature. Thus, in this case, load variables are acid concentration and system temperature corresponding to chemical stress and thermal stress.

2.6 Stresses

Stresses are the parameters which directly drive failure mechanisms. It is pertinent here to state the difference between load conditions and stress conditions. While load conditions are a subset of subsystem-level operating conditions, these are translated into stress conditions through the subsystem properties (discussed in the subsequent sections). For example, turbine rotational speed is the load condition, which is translated into centrifugal stress using turbine geometry and mechanical properties. Typically, several types of stresses are acting simultaneously on a subsystem. In this work, stress vector has been represented as σ and load vector as U .

2.7 Subsystem Properties Model

As shown in Fig. 1, the subsystem model translates the applied load into stresses, which drive the failure mechanism of respective subsystems. Translation of load vector U into stress vector σ is determined by

properties specific to that subsystem. These properties include, but are not limited to:

- Subsystem geometry (size, shape)
- Mechanical (stress-strain) properties
- Thermal properties (conductivity, melting point, thermal expansion)

Typically, finite element analysis is performed in conjunction with the material properties data to construct the subsystem properties model. In this work, it is assumed that the subsystem properties model, which translates the applied load vector to stress vector is available for the given system.

2.8 Failure Mechanisms

Failures in engineering systems occur due to specific causes. Failure mechanisms are the physical processes by which stresses cause damage to the elements comprising the system, and ultimately lead to failure. Physics-based prognostic models rely on identifying the failure mechanisms that could be activated by the applied stresses during the life cycle of the system. Failure mechanisms are broadly grouped into overstress mechanisms and wearout mechanisms (Dasgupta and Pecht, 1991).

Overstress failures are catastrophic events that result due to application of excessive stresses. This work focuses on failures that are caused by gradual deterioration. These types of failures are driven by wearout failure mechanisms.

In gas turbines, most of the failures occur due to fatigue and creep (Liu, 2002). Creep/fatigue damages start with deformations, and then eventually lead to ruptures.

3 METHODOLOGY

High-pressure turbine (HPT) is a subsystem exposed to extremely harsh conditions. In most of the cases, engine failure occurs as a result of HPT blade cracks. In HPT turbine blade, there exist two dominant failure mechanisms, i.e., creep and fatigue. Both of these failure mechanisms are driven by thermal and centrifugal stress. However, the ways these two failure mechanisms are driven by the stresses is different for each case.

In this research, it is assumed that:

- Creep is the only failure mechanism acting on the turbine blades (the effect of the other dominant failure mode, i.e. LCF, is not being considered in this paper).
- HPT stage has suffered from a loss in efficiency and flow due to creep strain.

As discussed in the previous sections, translation of load vector \mathbf{U} (which is a subset of subsystem-level properties) into stress vector $\boldsymbol{\sigma}$ is determined by the properties (geometry, mechanical, and thermal) specific to that subsystem, as shown in Fig. 2. Usually this step is carried out by running finite element analysis. For each set of load conditions, stresses are evaluated for the entire structure. The highest level of stresses are subsequently used as the life limiting parameter. In this paper, results obtained by (Liu, 2002) are used. A simple creep model is used to derive a creep equation, represented in Fig. 2 as *failure mechanism model*:

$$\epsilon_{creep} = \int_0^t \beta \sigma^2, \quad (1)$$

where ϵ_{creep} is creep strain, β is a temperature dependent constant, σ is the stress, and t is time over which the loading conditions continued.

The *failure mechanism model* is then combined with the *subsystem properties model* and a *response surface metamodel* for creep rate is constructed (Fig. 2).

Design of experiments (DoE) methodology is used in (Liu, 2002) to develop response surface equation (RSE) of the following form,

$$R = b_0 + \sum_{i=1}^k b_i x_i + \sum_{i=1}^k \sum_{j=i+1}^{k-1} b_{ij} x_i x_j. \quad (2)$$

where R is the response (ϵ_{creep} in this case), and x_1, x_2, \dots are the following variables:

- x_1 : temperature of hot flow
- x_2 : temperature of cooling flow
- x_3 : rotational speed
- x_4 : diameter of cooling hole
- x_5 : Young's modulus
- x_6 : thermal conductivity
- x_7 : thermal expansion at 0 degree
- x_8 : specific heat

Results in (Liu, 2002) show that hot flow temperature (x_1) and rotational speed (x_3), are the major contributors to the response variable.

Taking 0.5% creep strain as failure criteria and ignoring the insignificant terms, creep life is expressed as

$$\log(\text{creep life}) = 5.321 - 1.694(x_1) - 1.30(x_3). \quad (3)$$

Table 1 shows the DoE settings used to obtain this creep life metamodel.

Table 1: DoE settings used to obtain creep metamodel in Eq. (3).

	-1	0	1
Hot flow temperature (x_1) (C)	500	1000	1500
Rotational speed (x_3) (rpm)	1000	5000	9000

The response surface metamodel (Eq. (3)) considers constant load conditions. However, in actual operation, load conditions vary during each phase of operation. Life-fraction model, also known as Robinson's accumulation of creep (AC) rule, (Woodford, 1997; Dowling, 1993) proposes a method for variable loading conditions. The linear life-fraction model states that:

At failure

$$\sum_{i=1}^m \frac{t_i}{L_i} = D, \quad (4)$$

where

t_i = time interval at constant loading for i^{th} case ,

L_i = creep life at constant loading,

D = material dependent constant (Ainsworth *et al.*, 1994; Danzer, 1992),

m = number of intervals at constant loading.

4 APPLICATION DOMAIN

Results were obtained for a turbofan engine using recently released Commercial Modular Aero Propulsion System Simulation (C-MAPSS) package (Frederick *et al.*,). C-MAPSS is a tool for simulating a realistic large commercial turbofan engine. The software is a combination of Matlab and Simulink (The Mathworks, Inc.) with a number of editable fields. In addition to the engine model of 90,000 lb thrust, the package includes an atmospheric model capable of operation at (i) altitudes from sea level to 40,000 ft, (ii) Mach numbers from 0 to 0.90, (iii) sea level temperatures from -60 to 103 F, and (iv) a wide range of thrust levels throughout the full range of operating conditions. C-MAPSS has about 14 inputs that include fuel flow and a set of 13 health-parameters inputs. These inputs can be used to simulate the effects of faults and deterioration in any of the engine's five rotating

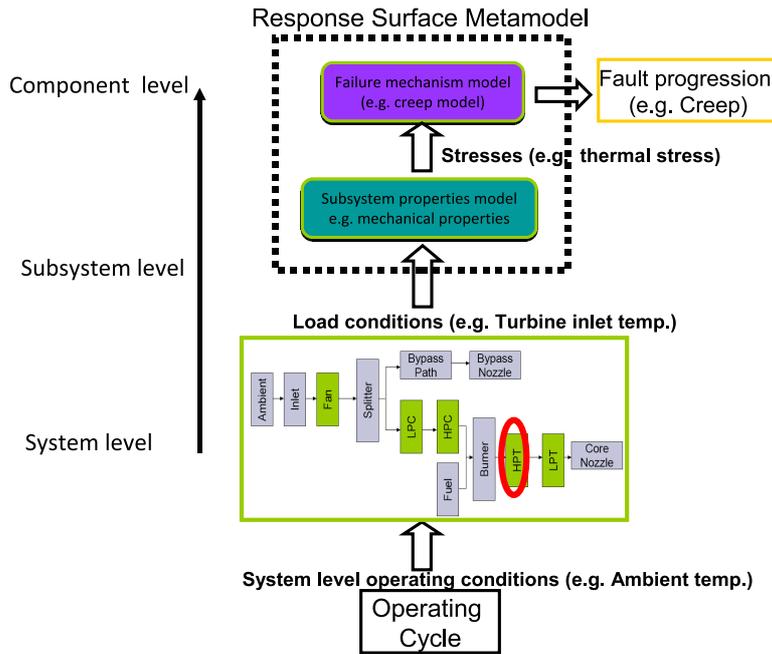


Figure 2: Architecture of the models used in the approach.

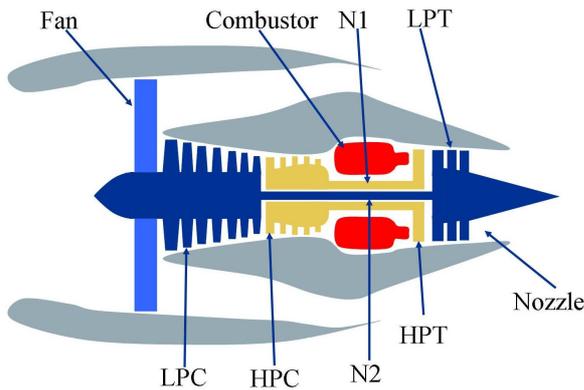


Figure 3: Simplified diagram of the engine simulated in C-MAPSS.

components, i.e., fan, low-pressure compressor (LPC), high-pressure compressor (HPC), high-pressure turbine (HPT), and low-pressure turbine (LPT). The engine diagram in Fig. 3 shows the main elements of the engine model, and the flow chart in Fig. 4 shows how various subroutines are assembled in the simulation.

There are 4 variables which constitute the system-level operating conditions.

1. Throttle resolver angle (TRA)
2. Altitude (Alt)
3. Mach number (Mach)
4. Ambient temperature (T_{amb})

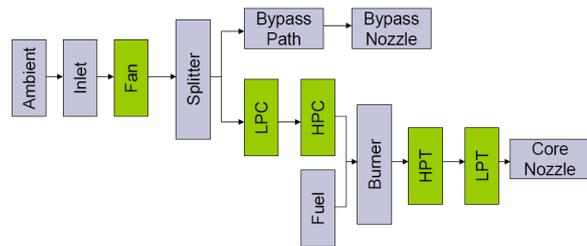


Figure 4: A layout showing various modules and their connections as modeled in the simulation.

5 RESULTS

In this paper, the operating cycle of a commercial aircraft is simulated by combining 5 phases of the cycle.

1. Takeoff
2. Climb
3. Cruise
4. Descent
5. Landing

Fig. 5 shows the system-level operating conditions (Alt, TRA, Mach, T_{amb}) in one of these phases of operation, i.e., takeoff.

C-MAPSS is an engine model, which is being used in this research to translate the system level operating conditions into subsystem level operating conditions. Initially, we assume that all the subsystems are healthy. Eq. (3) shows that hot flow temperature and turbine rotational speed are the principal contributors to the

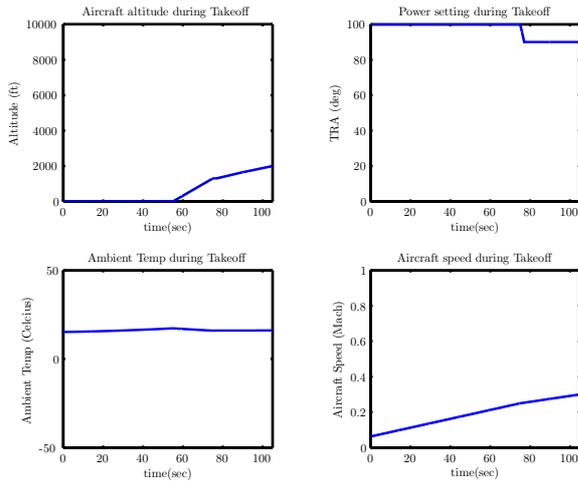


Figure 5: System-level operating conditions during takeoff.

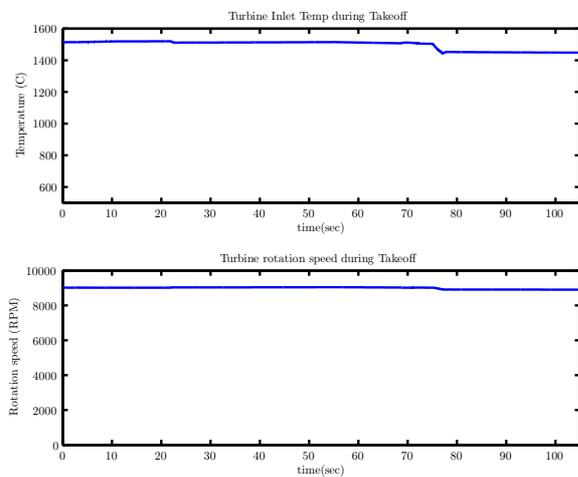


Figure 6: HPT load conditions during takeoff.

creep phenomenon. Fig. 6 shows these subsystem-level operating conditions (hot flow temperature and rotational speed) of HPT during the takeoff phase of the simulated operating cycle.

Using the linear life fraction model, damage accumulation due to creep is calculated for the entire cycle. Fig. 7 compares damage accumulation due to creep for the 2 cases. The first case is when the effects of creep have not started showing on the HPT blades yet. In HPT, creep manifests itself as loss in efficiency and flow. Next, the damage accumulation due to creep is calculated for the case when creep has started manifesting itself as HPT creep strain. The severity of the fault is simulated by modifying the HPT efficiency by -2% and HPT flow by -3% of the corresponding nominal values.

Fig. 8 quantifies the contribution of each phase of operation to the damage accumulation. It shows that

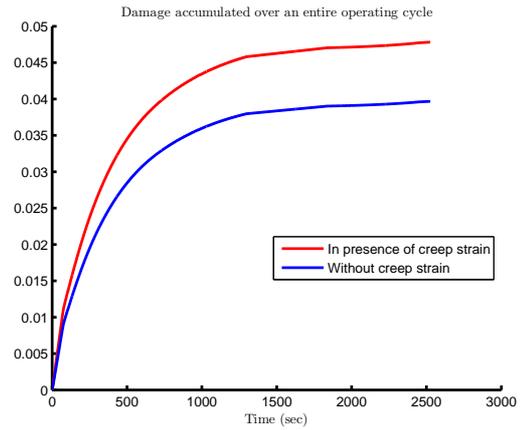


Figure 7: Damage accumulation due to creep during an entire operating cycle.

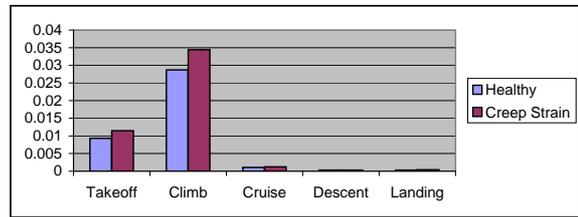


Figure 8: Damage contribution due to creep in each phase of operation.

only climb and takeoff phase contribute significantly to the damage due to creep.

The results show that there is a significant increase in damage accumulation over the operating cycle when creep has developed into creep strain. These results assume that all the other subsystems are in healthy state, i.e., no other subsystem has been subjected to deterioration yet. Thus, this case is an example of fault progression.

6 CONCLUSIONS

In a large system such as an aircraft engine, failure prognostics can be performed at various levels, i.e., component level, subsystem level, and system level. Though, component-level models yield more accurate results, yet the variables involved in such low-level models are not always available to system operators and maintenance personnel. This paper presented a hierarchical architecture, which was used to investigate the effects of system-level operating conditions on failure mechanism-based component models. A typical operating cycle of a commercial aircraft was simulated, and damage caused due to creep during various phases of the operation was estimated. Furthermore, creep damage without creep strain was compared to the case when creep strain had appeared on the HPT blades.

ACKNOWLEDGMENTS

The authors would like to thank Jonathan S. Litt in Glenn Research Center, NASA, for providing valuable insight into the operation of C-MAPSS.

REFERENCES

- (Ainsworth *et al.*, 1994) R.A. Ainsworth, P. J. Buden, and R. Ohtani. Design and assessment of components subjected to creep. *Journal of Strain Analysis*, 29(3), 1994.
- (Bowles, 2003) J.B. Bowles. Fundamentals of failure modes and effects analysis. *Tutorial Notes Annual Reliability and Maintainability Symposium*, 2003.
- (Danzer, 1992) R. Danzer. The influence of creep on the fatigue life of nickel based superalloys. *Proceedings of the 5th International Conference on Creep of Materials*, 1992.
- (Dasgupta and Pecht, 1991) A. Dasgupta and M. Pecht. Material failure mechanisms and damage models. *IEEE Transactions on Reliability*, 40:531 – 536, December 1991.
- (Dowling, 1993) N.E. Dowling. *Mechanical Behavior of Materials*. Prentice-Hall Inc., Upper Saddle River, NJ, 1993.
- (Frederick *et al.*,) D. Frederick, J. DeCastro, and J. Litt. *User's Guide for the Commercial Modular Aero-Propulsion System Simulation (CMAPSS)*. NASA/ARL.
- (Ganesan *et al.*, 2005) S. Ganesan, V. Evely, D. Das, and M. Pecht. Identification and utilization of failure mechanisms to enhance fmea and fmeca. *Proceedings of the IEEE Workshop on Accelerated Stress Testing & Reliability (ASTR)*, 2005.
- (Heimes, 2008) F.O. Heimes. Recurrent neural networks for remaining useful life estimation. *International Conference on Prognostics and Health Management*, 2008.
- (Liu, 2002) Z. Liu. A methodology for probabilistic remaining creep life assessment of gas turbine components. Technical report, School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, 2002.
- (Mathew *et al.*, 2008) S. Mathew, D. Das, R. Rossenberger, and M. Pecht. Failure mechanisms based prognostics. *International Conference on Prognostics and Health Management*, 2008.
- (Orchard *et al.*, 2005) M. E. Orchard, B. Wu, and G. Vachtsevanos. A particle filtering framework for failure prognosis. *Proceedings of WTC2005 World Tribology Congress III*, Sep 2005.
- (Saha *et al.*, 2007) B. Saha, K. Goebel, S. Poll, and J. Christophersen. An integrated approach to battery health monitoring using bayesian regression and state estimation. *IEEE Autotestcon Proc.*, 2007.
- (Saha *et al.*, 2009) B. Saha, K. Goebel, S. Poll, and J. Christophersen. Prognostics methods for battery health monitoring using a bayesian framework. *IEEE Transactions on Instrumentation and Measurement*, 58:291 – 296, February 2009.
- (Saha, 2007) B. Saha. A model-based reasoning architecture for system-level fault diagnosis. Technical report, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, 2007.
- (Wang *et al.*, 2008) T. Wang, J. Yu, D. Siegel, and J. Lee. A similarity-based prognostics approach for remaining useful life estimation of engineered systems. *International Conference on Prognostics and Health Management*, 2008.
- (Woodford, 1997) D.A. Woodford. Creep analysis of directionally solidified gtd111 based on stress relaxation testing. *Materials at High Temperatures*, 14(4):413 – 420, 1997.

Manzar Abbas earned the B.E. degree from the National University of Sciences and Technology (NUST), Pakistan in 1999 and the Masters of Electrical Engineering from the Georgia Institute of Technology in 2007. He joined intelligent control systems laboratory (ICSL) in summer 2005 and is currently a Ph.D. candidate in Electrical Engineering at the Georgia Institute of Technology. In the past, he carried out applied research in the areas of fault diagnostics and failure prognostics of electrical, electrochemical, electromechanical and electronics systems. Currently, he is working on developing a methodology to analyze fault propagation from one subsystem to the other subsystems with a specific focus on turbo machinery

George Vachtsevanos is a Professor Emeritus of Electrical and Computer Engineering at the Georgia Institute of Technology. He was awarded a B.E.E. degree from the City College of New York in 1962, a M.E.E. degree from New York University in 1963 and the Ph.D. degree in Electrical Engineering from the City University of New York in 1970. He directs the Intelligent Control Systems laboratory at Georgia Tech where faculty and students are conducting research in intelligent control, neurotechnology and cardiotechnology, fault diagnosis and prognosis of large-scale dynamical systems and control technologies for Unmanned Aerial Vehicles. His work is funded by government agencies and industry. He has published over 240 technical papers and is a senior member of IEEE. He was awarded the IEEE Control Systems Magazine Outstanding Paper Award for the years 2002-2003 (with L. Wills and B. Heck). He was also awarded the 2002-2003 Georgia Tech School of Electrical and Computer Engineering Distinguished Professor Award and the 2003-2004 Georgia Institute of Technology Outstanding Interdisciplinary Activities Award.