To the Graduate Council:

I am submitting herewith a dissertation written by Donald J. Malloy entitled "Real-Time Fault Identification for Developmental Turbine Engine Testing." I have examined the final copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mechanical Engineering.

Roy J. Schulz

Roy J. Schulz, Major Professor

We have read this dissertation
and recommend its acceptance:

D P Mehta

Roger Crawford

Accepted for the Council:

Lewinkel

Associate Vice Chancellor and
Dean of the Graduate School

REAL-TIME FAULT IDENTIFICATION
for
DEVELOPMENTAL TURBINE ENGINE TESTING

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Donald J. Malloy
May 1999

ACKNOWLEDGMENTS

In the course of this dissertation research, a number of scholars have assisted me generously. In particular, I would like to thank my major professor, Dr. Roy J. Schulz, and my committee members Dr. Firouz Shahrokhi, Dr. Roger Crawford, and Dr. Dinesh Mehta for their guidance, technical support, and constructive review comments. I would like to express my sincere appreciation to Dr. Doug Garrard, Dr. Virgil Smith, Mr. Alvis Turrentine, Mr. Sam Wehofer, Mr. Owen Boals, and Mr. Mark Chappell, from Sverdrup Technology, Inc., AEDC Group, for taking the time to review this research and provide valuable comments. The support of Mr. Mark Chappell, who helped direct this research, and Dr. Csaba Biegl, from Vanderbilt University, who developed the parallel-processing framework, is also acknowledged. Without the support of these individuals, my family, and loved ones, this dissertation would not have been possible.

ABSTRACT

Hundreds of individual sensors produce an enormous amount of data during developmental turbine engine testing. The challenge is to ensure the validity of the data and to identify data and engine anomalies in a timely manner. An automated data validation, engine condition monitoring, and fault identification process that emulates typical engineering techniques has been developed for developmental engine testing.

An automated data validation and fault identification approach employing engine cycle-matching principles is described. Engine cycle-matching is automated by using an adaptive nonlinear component-level computer model capable of simulating both steady-state and transient engine operation. Automated steady-state, transient, and real-time model calibration processes are also described. The model enables automation of traditional data validation, engine condition monitoring, and fault identification procedures. A distributed parallel computing approach enables the entire process to operate in real-time.

The result is a capability to detect data and engine anomalies in real-time during developmental engine testing. The approach is shown to be successful in detecting and identifying sensor anomalies as they occur and distinguishing these anomalies from variations in component and overall engine aerothermodynamic performance.

The component-level model-based engine performance and fault identification technique of the present research is capable of:

- identifying measurement errors on the order of 0.5 percent (e.g., sensor bias, drift, level shift, noise, or poor response) in facility fuel flow, airflow, and thrust measurements;

- identifying measurement errors in engine aerothermodynamic measurements (rotor speeds, gas path pressures and temperatures);

- identifying measurement errors in engine control sensors (e.g., leaking/biased pressure sensor, slowly responding pressure measurement) and variable geometry rigging (e.g., misset guide vanes or nozzle area) that would invalidate a test or series of tests;

- identifying abrupt faults (e.g., faults due to domestic object damage, foreign object damage, and control anomalies);

- identifying slow faults (e.g., component or overall engine degradation, and sensor drift). Specifically, the technique is capable of identifying small changes in compressor (or fan) performance on the order of 0.5 percent; and

- being easily extended to diagnose secondary failure modes and to verify any modeling assumptions that may arise for developmental engine tests (e.g., increase in turbine flow capacity, inaccurate measurement of facility bleed flows, horsepower extraction, etc.).

The component-level model-based engine performance and fault identification method developed in the present work brings together features which individually and collectively advance the state-of-the-art. These features are separated into three categories: advancements to effectively quantify off-nominal behavior, advancements to provide a fault detection capability that is practical from the viewpoint of the analysis, implementation, tuning, and design, and advancements to provide a real-time fault detection capability that is reliable and efficient.

TABLE OF CONTENTS

CHAPTER PAGE

1. INTRODUCTION 1
   - Fault Identification Overview 4
   - Physics-Based Model Overview 6
   - Statement of the Problem 8
     - Limitations to Current Engine Performance Prediction and Fault Identification Techniques 8
     - Contribution to the State-of-the-Art 10
   - Fault Identification Approach 15

2. LITERATURE SURVEY 19
   - System Identification Techniques 19
   - Nonlinear Observer Models 22
   - Fault Identification Techniques 24
   - Model Structure Determination from Test Data 29
   - Steady-State and Transient Gas Turbine Performance: Prediction, Analysis, and Identification 32
   - Real-Time Turbine Engine Models 34
   - In-Flight Estimation Algorithms for Adaptive Performance Optimization 35
   - Gas Turbine Fault Detection/In-Flight Condition Monitoring Systems 38
   - Gas Turbine Performance Deterioration Modeling 42
   - Diagnosis of Turbine Engine Transient Performance with Model-Based Parameter Estimation Techniques 44
   - Contribution of the Present Study to the State-of-the-Art 47

3. MODEL DEVELOPMENT 49
   - Typical Developmental Engine Tests 52
     - Performance Test and Evaluation 53
     - Operability Test and Evaluation 54
   - Test Measurements and Uncertainties 55
     - Mean Flow Properties and Profile Effects 56

Determination of Time Derivatives from Test Data 57
Pressure Measurements 58
Temperature Measurements 59
Speed Measurements 61
Variable Geometry Measurements 62
Structural Clearance Measurements 63
Measurement Uncertainty Influence Coefficients 63
Fuel Flow Measurements 64
Engine Inlet Airflow Measurements 69
Bleed Extraction Measurements 73
Auxiliary Power Extraction Measurements 75
Thrust Measurements 75
Data Filtering Technique to Determine Time-Dependent Values from Test Data 80
Engine Inlet/Isolation Seal 90
Fan Steady-State Analysis Summary 91
Fan Transient Analysis Considerations 94
Fan Analysis Summary 96
High-Pressure Compressor Steady-State Analysis Summary 97
High-Pressure Compressor Transient Analysis Considerations 99
High-Pressure Compressor Analysis Summary 100
Burner Steady-State Analysis Summary 100
Burner Transient Analysis Considerations 102
Burner Analysis Summary 103
High-Pressure Turbine Steady-State Analysis Summary 103
High-Pressure Turbine Transient Analysis Considerations 106
High-Pressure Turbine Analysis Summary 106
Low-Pressure Turbine Steady-State Analysis Summary 108
Low-Pressure Turbine Transient Analysis Considerations 109
Low-Pressure Turbine Analysis Summary 110
Mixer Steady-State Analysis Summary 111
Mixer Transient Analysis Considerations 112
Mixer Analysis Summary 112
Augmentor Steady-State Analysis Summary 113
Augmentor Transient Analysis Considerations 114
Augmentor Analysis Summary 114

Nozzle Steady-State Analysis Summary 114
Nozzle Transient Analysis Considerations 115
Nozzle Analysis Summary 115

4. MODEL FORMULATION AND IMPLEMENTATION 116
Numerical Solution to the Cycle Model Equations 116
Overview of Parallel Computing Processes 124
Model Calibration 125
Step 1- Steady-State Model Calibration 125
Step 2 - Transient Model Calibration 126
Estimation of Heat Transfer Characteristics and Thermocouple Time Constants 129
Development of Non-dimensional Number to Characterize Algorithm Stability and Accuracy 133
Step 3 - Real-Time Model Calibration 150

5. PARALLEL PROCESSING 152
Software/Hardware Engineering Considerations 153
Development Platform 153
Target Platforms 155
Explicit Algorithmic Parallelization 159
Main and Worker Processes 160
Worker: High-Level "Pseudo-Code" 162
Main Process: High-Level "Pseudo-Code" 163
Time Analysis 165
Scalable Performance Improvement 172

6. FAULT DETECTION AND DIAGNOSIS PROCEDURES 176
Fault Detection 178
Sensor Fault Search 181
Performance Degradation Search 182
Human-Computer Interface and Data Visualization System 184

7. IMPLEMENTATION AND VERIFICATION OF MODEL-BASED TECHNIQUES FOR FAULT IDENTIFICATION 185

8. SUMMARY AND CONCLUSIONS 191

9. RECOMMENDATIONS FOR FUTURE WORK 206

REFERENCES 209

APPENDICES 219

APPENDIX A. FIGURES 220

APPENDIX B. REAL-TIME TURBINE ENGINE MODLES 347

APPENDIX C. ADVANCED TURBINE ENGINE SIMULATION TECHNIQUE (ATEST) 353

APPENDIX D. MEASUREMENT UNCERTAINTY TABLES 359

APPENDIX E. THERMOCOUPLE TIME RESPONSE 362

APPENDIX F. PROPAGATION OF ELEMENTAL MEASUREMENT UNCERTAINTIES 364

APPENDIX G. CALCULATION OF AIR FLOW USING VENTURIS AND NOZZLES..... 366

APPENDIX H. FAN PERFORMANCE CHARACTERIZATION..... 367

APPENDIX I. HP COMPRESSOR PERFORMANCE CHARACTERIZATION..... 371

APPENDIX J. HP TURBINE PERFORMANCE CHARACTERIZATION..... 375

APPENDIX K. LP TURBINE PERFORMANCE CHARACTERIZATION..... 379

APPENDIX L. OVERALL ENGINE PERFORMANCE CHARACTERIZATION..... 381

APPENDIX M. SOFTWARE/HARDWARE ENGINEERING CONSIDERATIONS 384

APPENDIX N. SCALABLE SOFTWARE PARADIGM AND MESSAGE-PASSING INTERFACE FOR THE TARGET PLATFORMS 390

APPENDIX O. SIMULATION INTERFACE 393

VITA 399

LIST of FIGURES

Fig. 1.1. Typical Propulsion Development Test Cell 221
Fig. 1.2. Augmented Turbofan Engine 222
Fig. 1.3. Fault Identification Overview 223
Fig. 3.1. Engine Deceleration: Fan Speed vs. Time 224
Fig. 3.2. Engine Deceleration: Fan Speed Derivative vs. Time 225
Fig. 3.3. Hydrodynamic Forces on a Blade Element Subjected to a Step Increase in Flow Rate 226
Fig. 3.4. Fuel Flow Rate Oscillations During an Engine Acceleration 227
Fig. 3.5. Typical Thrust Measurement Configuration 228
Fig. 3.6. Relative Magnitudes of Gross and Net Thrust and Ram Drag 229
Fig. 3.7. Thrust Stand Oscillations During an Engine Acceleration 230
Fig. 3.8. Attenuation of Typical 8-Pole Low Pass Filters 231
Fig. 3.9. Theoretical Frequency Response of Butterworth Filters 232
Fig. 3.10. Theoretical Response of Butterworth Filters to a Step Input Signal 233
Fig. 3.11. Response of 50 Hz 4-Pole Filters to Triangular Input Signal 234
Fig. 3.12. Response of 100 Hz 4-Pole Filters to a Step Input Signal 235
Fig. 3.13. Response of Kalman Filter Based Algorithm to Severe FOD Event 236
Fig. 3.14. Response of Digital Filtering Techniques to a Rapid Engine Deceleration 237
Fig. 3.15. Response of Digital Filtering Techniques to Moderate Fan Speed Variations 238
Fig. 3.16. Inlet Station Designation 239
Fig. 3.17. Isolation Seal Station Designation 240
Fig. 3.18.1. Fan Station Designation for Twin-Spool Turbofan 241
Fig. 3.18.2. Fan Analysis: Steady-State Parameters 242
Fig. 3.18.3. Fan Analysis: Transient Parameters 243
Fig. 3.18.4. Fan Parameters 244

Fig. 3.19.1. High-Pressure Compressor Station Designation for Twin-Spool Turbofan 245
Fig. 3.19.2. High-Pressure Compressor: Steady-State Parameters 246
Fig. 3.19.3. High-Pressure Compressor Analysis: Transient Parameters 247
Fig. 3.19.4. High-Pressure Compressor Parameters 248
Fig. 3.20.1. Burner Station Designation for Twin-Spool Turbofan 249
Fig. 3.20.2. Burner Analysis: Steady-State Parameters 250
Fig. 3.20.3. Burner Analysis: Transient Parameters 251
Fig. 3.20.4. Burner Parameters 252
Fig. 3.20.1. High-Pressure Turbine Station Designation for Twin-Spool Turbofan 253
Fig. 3.21.2. High-Pressure Turbine Analysis: Steady-State Parameters 254
Fig. 3.21.3. High-Pressure Turbine Analysis: Transient Parameters 255
Fig. 3.21.4. High-Pressure Turbine Parameters 256
Fig. 3.22.1. Low-Pressure Turbine Station Designation for Twin-Spool Turbofan 257
Fig. 3.22.2. Low-Pressure Turbine Analysis: Steady-State Parameters 258
Fig. 3.22.3. Low-Pressure Turbine Analysis: Transient Parameters 259
Fig. 3.22.4. Low-Pressure Turbine Parameters 260
Fig. 3.23.1. Mixer Station Designation for Twin-Spool Turbofan 261
Fig. 3.23.2. Mixer Analysis: Steady-State Parameters 262
Fig. 3.23.3. Mixer Analysis: Transient Parameters 263
Fig. 3.23.4. Mixer Parameters 264
Fig. 3.24.1. Augmentor Station Designation for Twin-Spool Turbofan 265
Fig. 3.24.2. Augmentor Analysis: Steady-State Parameters 266
Fig. 3.24.3. Augmentor Analysis: Transient Parameters 267
Fig. 3.24.4. Augmentor Parameters 268
Fig. 3.25.1. Nozzle Station Designation for Twin-Spool Turbofan 269
Fig. 3.25.2. Nozzle Analysis: Steady-State Parameters 270
Fig. 3.25.3. Nozzle Analysis: Transient Parameters 271
Fig. 3.25.4. Nozzle Parameters 272
Fig. 4.1. Measurement and Data Reduction System Schematic 273
Fig. 4.2. Measurement and Data Reduction System Schematic (Continued) 274
Fig. 4.3. Cycle-Balancing and Fault Identification Process (Serial Operation) 275

Fig. 4.4. Engine Model Block Diagram 276
Fig. 4.5.1. Compressor Performance Representation 277
Fig. 4.5.2. Transformed Compressor Map Representation 278
Fig. 4.6. Cycle-Balancing (Parallel Operation: 2 Replicated Workers) 279
Fig. 4.7. Step 1: Initial Steady-State Performance Calibration 280
Fig. 4.8. Step 2: Initial Transient Performance Calibration 281
Fig. 4.9. Step 3: Real-Time Model Calibration and Fault Identification 282
Fig. 4.10. Calculated Component Efficiencies During Snap Engine Deceleration (Before Transient Model Calibration) 283
Fig. 4.11. Calculated Component Efficiencies During Engine Bodie (Before Transient Model Calibration) 284
Fig. 4.12. LP Turbine Exit Temperature During Snap Engine Deceleration (Before Transient Model Calibration) 285
Fig. 4.13. LP Turbine Exit Temperature During Engine Bodie (Before Transient Model Calibration) 286
Fig. 4.14. Assumed Transient Gas and Metal Temperature Variations for Error Analysis ($\tau << 1$) 287
Fig. 4.15. Metal Temperature Error Analysis Model for Transient Operation 288
Fig. 4.16. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Raw Data, 10 Processors, $\tau \equiv 1$ second) 289
Fig. 4.17. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Filtered Data, 10 Processors, $\tau \equiv 1$ second) 290
Fig. 4.18. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Raw Data, 50 Processors, $\tau \equiv 1$ second) 291
Fig. 4.19. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Filtered Data, 50 Processors, $\tau \equiv 1$ second) 292
Fig. 4.20. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Raw Data, 100 Processors, $\tau \equiv 1$ second) 293
Fig. 4.21. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Filtered Data, 100 Processors, $\tau \equiv 1$ second) 294

Fig. 4.22. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Raw Data, 10 Processors, $\tau \equiv 1$ second) 295
Fig. 4.23. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Filtered Data, 10 Processors, $\tau \equiv 1$ second) 296
Fig. 4.24. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Raw Data, 50 Processors, $\tau \equiv 1$ second) 297
Fig. 4.25. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Filtered Data, 50 Processors, $\tau \equiv 1$ second) 298
Fig. 4.26. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Raw Data, 100 Processors, $\tau \equiv 1$ second) 299
Fig. 4.27. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Filtered Data, 100 Processors, $\tau \equiv 1$ second) 300
Fig. 5.1. Interconnections for Eight Processor Parallel Distributed Machine 301
Fig. 5.2. Pentium Platform Effectiveness 302
Fig. 5.3. Parallel Processing Effectiveness (SGI Origin 2000) 303
Fig. 5.4. Parallel Processing Effectiveness (HP-SPP 2000) 304
Fig. 6.1. Normalized Rake 1 Compressor Exit Total Pressures: Nominal and Degraded Operation 305
Fig. 6.2. Normalized Rake 2 Compressor Exit Total Pressures: Nominal and Degraded Operation 306
Fig. 6.3. Normalized Rake 3 Compressor Exit Total Temperatures: Nominal and Degraded Operation 307
Fig. 6.4. Normalized Rake 4 Compressor Exit Total Temperatures: Nominal and Degraded Operation 308
Fig. 6.5. Variation in LP Turbine Discharge Temperature Referred to Fan Inlet Conditions: Nominal and Degraded Operation 309
Fig. 6.6. Variation in Referred Specific Fuel Consumption: Nominal and Degraded Operation 310
Fig. 6.7. Variation in HP Compressor Airflow Pumping Capacity: Nominal and Degraded Operation 311

Fig. 6.8. Variation in HP Compressor Pressure Ratio: Nominal and Degraded Operation 312
Fig. 6.9. Variation in HP Compressor Efficiency: Nominal and Degraded Operation 313
Fig. 6.10. HP Compressor Efficiency Map Scalar: Nominal and Degraded Operation 314
Fig. 7.1. Predicted and Measured Engine Gross Thrust (Initiation of Slow Engine Acceleration) 315
Fig. 7.2. Predicted and Measured Engine Gross Thrust (Part-Way Through Slow Engine Acceleration) 316
Fig. 7.3. Predicted and Measured Engine Gross Thrust (End of Slow Engine Acceleration) 317
Fig. 7.4. Predicted and Measured Engine Gross Thrust (Snap Engine Deceleration) 318
Fig. 7.5. Predicted and Measured Engine Gross Thrust (Engine Bodie) 319
Fig. 7.6. Predicted and Measured LPT Exit Temperature (Initiation of Slow Engine Acceleration) 320
Fig. 7.7. Predicted and Measured LPT Exit Temperature (Part-Way Through Slow Engine Acceleration) 321
Fig. 7.8. Predicted and Measured LPT Exit Temperature (End of Slow Engine Acceleration) 322
Fig. 7.9. Predicted and Measured LPT Exit Temperature (Snap Engine Deceleration) 323
Fig. 7.10. Predicted and Measured LPT Exit Temperature (Engine Bodie) 324
Fig. 7.11. Predicted and Measured LPT Exit Pressure (Initiation of Slow Engine Acceleration) 325
Fig. 7.12. Predicted and Measured LPT Exit Pressure (Part-Way Through Slow Engine Acceleration) 326
Fig. 7.13. Predicted and Measured LPT Exit Pressure (End of Slow Engine Acceleration) 327
Fig. 7.14. Predicted and Measured LPT Exit Pressure (Snap Engine Deceleration) 328
Fig. 7.15. Predicted and Measured LPT Exit Pressure (Engine Bodie) 329
Fig. 7.16. Representative Turbine Exit Temperature Measurements 330
Fig. 7.17. Lag Compensated Turbine Exit Temperature 331

Fig. 7.18. Effects of Propulsion Control Schedules on Airflow Measurement 332
Fig. 7.19. Effect of Slowly Responding Pressure Measurement on Airflow Measurement 333
Fig. 7.20. Slowly Responding LPT Exit Pressure 334
Fig. 7.21.1. Abrupt Change in Compressor Performance During Steady Engine Operation 335
Fig. 7.21.2. Abrupt Change in Compressor Performance During Steady Engine Operation 336
Fig. 7.22. Fault Identification and Estimate of Error 337
Fig. 7.23. Abrupt Change in Compressor Exit Pressure During Steady Engine Operation 338
Fig. 7.24. Seeded Fault: 0.5% Gross Thrust Error 339
Fig. 7.25. Seeded Fault: 0.5% Airflow Error 340
Fig. 7.26. Seeded Fault: 1.0% Fuel Flow Error 341
Fig. 7.27. Seeded Fault: 1.0% Nozzle Area Error 342
Fig. 7.28. Seeded Fault: 5% (50 Lbf) Thrust Error 343
Fig. 7.29. Seeded Fault: 2% Airflow Error 344
Fig. 7.30. Seeded Fault: 2% Fuel Flow Error 345
Fig. 7.31. Multiple Seeded Data Faults 346

NOMENCLATURE

A	Area
A_{2A}	Fan inlet physical area
A_{24}	Compressor inlet physical area
A_3	Compressor discharge physical area
A_{41T}	Thermally corrected HP turbine nozzle discharge area
A_{44T}	Thermally corrected LP turbine nozzle discharge area
A_8	Nozzle throat area
A_9	Nozzle discharge area
A_{ISOD}	Area of male isolation seal
A_{TC}	Surface area of the thermocouple junction
A_w	Effective wetted surface area
$A_{w,\ Augmentor}$	Augmentor effective wetted surface area
$A_{w,\ Burner}$	Burner effective wetted surface area
$A_{w,\ Fan}$	Fan effective wetted surface area
$A_{w,\ HPC}$	HPC effective wetted surface area
$A_{w,\ HPT}$	HPT effective wetted surface area
$A_{w,\ LPT}$	LPT effective wetted surface area
$A_{w,\ Mixer}$	Mixer effective wetted surface area
$A_{w,\ Nozzle}$	Nozzle effective wetted surface area
A/B	Afterburner
ADP_{IS}	Area times pressure difference evaluated at the isolation seal plane
AE	Effective area
AE_{16}	Effective mixing area of the bypass flow
AE_6	Effective mixing areas of the core flow
AEDC	Arnold Engineering Development Center
ALT	Altitude

API	Application program interface
BPR	Bypass Ratio
C_{pfd}	Pressure correction factor
C_w	Discharge coefficient
CF	Flow coefficient
CF_{2AX}	Fan inlet assumed flow coefficient
CF_{24}	Compressor inlet assumed flow coefficient
CF_3	Compressor discharge assumed flow coefficient
CG	Gravitational Constant
CLM	Component Level Model
CP_m	Metal specific heat
$CP_{m,\ Augmentor}$	Augmentor metal specific heat
$CP_{m,\ Burner}$	Burner metal specific heat
$CP_{m,\ Fan}$	Fan metal specific heat
$CP_{m,\ HPC}$	HPC metal specific heat
$CP_{m,\ HPT}$	HPT metal specific heat
$CP_{m,\ LPT}$	LPT metal specific heat
$CP_{m,\ Mixer}$	Mixer metal specific heat
$CP_{m,\ Nozzle}$	Nozzle metal specific heat
$CQUL_{2A}$	Fan inlet coefficient of linear expansion
$CQUL_{24}$	Compressor inlet coefficient of linear expansion
$CQUL_3$	Compressor discharge coefficient of linear expansion
CT_{IS}	Area thermal expansion coefficient
CVV	Compressor variable vane angle
D	Diameter
$\frac{dN}{dt}$	Time rate of change of shaft speed
$\frac{dP}{dt}$	Time rate of change of pressure

$\frac{dT_m}{dt}$	Time rate of change of metal temperature
$\left(\frac{dT_{gas}}{dt}\right)_{max}$	Maximum time rate of change of the gas temperature
E	Efficiency
E2A	Fan efficiency (overall)
E21	Fan efficiency (hub)
E12	Fan efficiency (tip)
E41	HP turbine efficiency
E44	LP turbine efficiency
EAB	Augmentor efficiency
EB	Burner efficiency
ERAM	Inlet ram recovery
ETA	Isentropic efficiency
EUD	Engineering Unit Data
F	Thrust
F_a	Thermal expansion factor
F_G	Engine gross thrust
F_N	Engine net thrust
F_S	Measured scale force with appropriate test cell correction factors
FDDI	Fiber Distributed-Data Interface
FGR2A	Referred gross thrust
FAR	Fuel-air-mass ratio
FNR2A	Referred net thrust
FRAM	Ram drag
FDP	Fault detection parameter
FHV	Fuel lowering heating value
FHV_{IST}	Fuel lower heating value (International steam table calorie)

FHV_{LAB}	Fuel lower heating value obtained per laboratory analysis (thermochemical calorie)
FVV	Fan variable vane angle
FSG	Fuel specific gravity during test
GUI	Graphical User Interface
h_c	Average convective heat transfer coefficient at the junction
h_r	Overall heat transfer coefficient
$h_{Augmentor}$	Augmentor overall heat transfer coefficient
h_{Burner}	Burner overall heat transfer coefficient
h_{Fan}	Fan overall heat transfer coefficient
h_{HPC}	HPC overall heat transfer coefficient
h_{HPT}	HPT overall heat transfer coefficient
h_{LPT}	LPT overall heat transfer coefficient
h_{Mixer}	Mixer overall heat transfer coefficient
h_{Nozzle}	Nozzle overall heat transfer coefficient
H	Total enthalpy
HB	Bleed flow enthalpy
HB_{24A}	Compressor secondary bleed flow rate enthalpy
HB_{24B}	Compressor secondary bleed flow rate enthalpy
HB_{24C}	Compressor secondary bleed flow rate enthalpy
HB_{24D}	Compressor secondary bleed flow rate enthalpy
HB_{24E}	Compressor secondary bleed flow rate enthalpy
HB_{24F}	Compressor secondary bleed flow rate enthalpy
HB_{24G}	Compressor secondary bleed flow rate enthalpy
HB_{24H}	Compressor secondary bleed flow rate enthalpy
HB_{24I}	Compressor secondary bleed flow rate enthalpy
HF	Total enthalpy at the final condition
HFIS	Total enthalpy resulting from an isentropic process
HI	Total enthalpy at the initial condition

HP	High-pressure component
HPC	High-pressure compressor
HPF	High performance FORTRAN
HPT	High-pressure turbine
IS	Isentropic processor or isolation seal
J	Rotor moment of inertia
K	Time constant, $(60/2\pi)^2 550$
K-factor	Frequency/viscosity
L	Loss
LP	Low-pressure component
LPC	Low-pressure compressor
LPT	Low-pressure turbine
$\dot{m}$	Gas flow rate
$\dot{m}_{in}$	Entering gas flow rate
$\dot{m}_{out}$	Exiting gas flow rate
Ma	Gas stream Mach number
M_t	Metal mass
m_{TC}	Mass of the thermocouple junction
$M_{Augmentor}$	Augmentor metal mass
M_{Burner}	Burner metal mass
M_{Fan}	Fan metal mass
M_{HPC}	HPC metal mass
M_{HPT}	HPT metal mass
M_{LPT}	LPT metal mass
M_{Mixer}	Mixer metal mass
M_{Nozzle}	Nozzle metal mass
MIMD	Multiple instruction stream/multiple data stream
MPI	Message passing interface

N	Instantaneous shaft speed
NW	Not available for work
N_L	Percent corrected low rotor speed
P	Total pressure
P0	Free-stream total pressure
P_1	Inlet/engine interface total pressure
P_{12}	Fan tip inlet total pressure
P_{13}	Fan tip discharge total pressure
P_{16}	Mixer inlet (bypass flow) total pressure
P_2	Fan hub inlet total pressure
P_{2A}	Average fan inlet total pressure
P_{21}	Fan hub discharge total pressure
P_{24}	Compressor inlet total pressure
P_3	Compressor discharge total pressure
P_{31}	Burner inlet total pressure
P_4	Burner discharge total pressure
P_{41}	Turbine nozzle discharge total pressure
P_{44}	LP turbine inlet total pressure
P_5	LP turbine exit total pressure
P_6	Mixer inlet (core flow) total pressure
P_{62}	Augmentor inlet total pressure
P_7	Engine/exhaust nozzle interface total pressure
P_8	Nozzle throat total pressure
P_9	Nozzle discharge total pressure
P_{PL}	Engine inlet duct inlet plenum total pressure
P_S	Static pressure
Ps_o	Reference static pressure
P_T	Total pressure
P_v	Vapor pressure

P-A	Area pressure difference
PAMB	Simulated altitude ambient pressure (cell pressure)
PCN2R	Fan percent corrected rotor speed
PCN24R	HP compressor percent corrected rotor speed
PS	Static pressure
PS_{IS}	Static pressure at isolation seal
PS_{2A}	Fan inlet static pressure
PS_{24}	Compressor inlet static pressure
PVM	Parallel virtual machine
PW	Power
PWL_{41}	HP spool power loss
PWX_{41}	HP turbine power extraction
PWL_{44}	LP spool power loss
QP_{2X}	Engine control fan inlet total pressure
QPS_{13}	Engine control fan inlet static pressure
QPS_3	Engine control compressor discharge static pressure
QT_{2A}	Engine control fan inlet total temperature
QT_{21}	Engine control fan hub discharge total temperature
QT_{24}	Engine control compressor inlet total temperature
QT_5	Engine control LP turbine discharge total temperature
QXN_2	Engine control fan rotor speed
QXN_{24}	Engine control compressor rotor speed
R	Gas constant
Re	Reynolds number
R_{IS}	Duct radius
Ro	Roshko number
$RFAW_{2A}$	Fan inlet adiabatic wall recovery factor
$RFAW_{24}$	Compressor inlet adiabatic wall recovery factor
$RFAW_3$	Compressor discharge adiabatic wall recovery factor

RHO_{IS}	Density of air at the isolation seal
RNI	Reynolds number index
RNI_{2A}	Fan Reynolds number index
RNI_{24}	Compressor Reynolds number index
RNI_{41}	HP turbine Reynolds number index
RNI_{44}	LP turbine Reynolds number index
St	Strouhal number
SFC	Specific fuel consumption
SFCR2A	Referred specific fuel consumption
SH	Specific heat
SIMD	Single Instruction Stream/Multiple Data Stream
t	Time
T	Total temperature
T_o	Reference temperature
T_1	Inlet/engine interface total temperature
T_{12}	Fan tip inlet total temperature
T_{13}	Fan tip discharge total temperature
T_{16}	Mixer inlet (bypass flow) total temperature
T_2	Fan hub inlet total temperature
T_{21}	Fan hub discharge total temperature
T_{2A}	Average fan inlet total temperature
T_{24}	Compressor inlet total temperature
T_3	Compressor discharge total temperature
T_{31}	Burner inlet total temperature
T_4	Burner discharge total temperature
T_{41}	Turbine nozzle discharge total temperature
T_{44}	LP turbine inlet total temperature
T_5	LP turbine discharge total temperature
T_6	Mixer inlet (core flow) total temperature

T_{62}	Augmentor inlet total temperature
T_7	Engine/exhaust nozzle interface total temperature
T_8	Nozzle throat total temperature
T_9	Nozzle discharge total temperature
T_{gas}	Gas temperature
T_m	Metal temperature
$(T_m)_{error}$	Error in the estimated metal temperature
T_T	Total temperature
T_{TC}	Temperature indicated by the thermocouple
T5R2A	LP turbine discharge total temperature referred to Station 2A
TAW	Adiabatic wall temperature
$THET_{IS}$	Momentum thickness of boundary layer at the isolation seal
TS	Static temperature
U_{RSS}	Root-sum-square uncertainty
U_{ADD}	Additive uncertainty
UVC	Universal viscosity Curve
V	Volume or velocity
V_{IS}	Free-stream velocity of air at the isolation seal
V0	Ssimulated free-stream velocity
VCF	Volume correction factor
VISTST	Fuel kinematic viscosity during test
VPP	Power law exponent
W	Mass flow rate
W_1	Engine/inlet bellmouth mass flow rate
W_{2A}	Total fan inlet mass flow rate
W_{2AX}	Fan inlet mass flow rate from fan inlet conditions
W_{24}	Compressor inlet mass flow rate
W_{24X}	Compressor inlet mass flow rate from compressor inlet conditions
W_3	Compressor discharge mass flow rate

W_{3X}	Compressor discharge mass flow rate from compressor discharge conditions
W_{41}	HP turbine nozzle discharge mass flow
W_{41FFX}	HP turbine nozzle discharge mass flow function
W_{43}	HP turbine discharge mass flow rate
W_{44}	LP turbine inlet mass flow rate
W_{44FFX}	LP turbine nozzle discharge mass flow function
W_5	LP turbine exit mass flow rate
W_6	Mixer inlet (core flow) mass flow rate
W_{62}	Augmentor inlet mass flow rate
W_7	Engine/exhaust nozzle interface mass flow rate
W_8	Nozzle throat mass flow rate
W_9	Nozzle discharge mass flow rate
W_{IS}	Mass flow rate
W2AR	Fan inlet referred (corrected) mass flow
W24R	HP compressor inlet referred (corrected) mass flow
W41R	HP turbine inlet referred (corrected) mass flow rate
W44R	LP turbine inlet referred (corrected) mass flow rate
WAR	Water-to-air mass ratio
WAR_{2A}	Water-to-air mass ratio at the fan inlet
WB	Bleed flow rate
WB_{24A}	Compressor secondary bleed flow rate
WB_{24B}	Compressor secondary bleed flow rate
WB_{24C}	Compressor secondary bleed flow rate
WB_{24D}	Compressor secondary bleed flow rate
WB_{24E}	Compressor secondary bleed flow rate
WB_{24F}	Compressor secondary bleed flow rate
WB_{24G}	Compressor secondary bleed flow rate
WB_{24H}	Compressor secondary bleed flow rate

WB_{24I}	Compressor secondary bleed flow rate
WB_{13}	Fan bypass duct bleed flow rate
$WB_{13\ \text{Valve Position}}$	Fan bypass duct bleed flow valve position
WB_3	Compressor discharge bleed flow rate
$WB_{3\ \text{Valve Position}}$	Compressor discharge bleed flow valve position
WC	Corrected mass flow rate or cooling flow rate
WC_{LHNW}	LP turbine cooling flow not available for LP turbine work
WC_{LLW}	LP turbine cooling flow available for LP turbine work
WFB	Burner fuel flow
WFGG	Engine gas generator fuel flow
WFMC	Fuel metering valve feedback fuel flow
WFT	Total engine fuel flow
WFTR2A	Referred fuel flow
WHC_{HNW}	HP turbine cooling flow energy not available for HP turbine work
WHC_{HW}	HP turbine cooling flow energy available for HP turbine work
WHC_{LNW}	LP turbine cooling flow energy not available for LP turbine work
WHC_{LW}	LP turbine cooling flow energy available for LP turbine work
W-V	Inlet momentum
X_{N2}	Fan rotor speed
X_{N24}	Compressor rotor speed
XM	Mach number
XMVIS	Inlet momentum evaluated at the isolation seal plane
XN	Rotor speed
XN2AR	Fan referred (corrected) rotor speed
XN24R	HP compressor referred (corrected) rotor speed
XN24R41	HP turbine referred (corrected) rotor speed
XN2R44	LP turbine referred (corrected) rotor speed
Y_a	Adiabatic expansion factor
Z	Compressibility factor

GREEK SYMBOLS

β	Area ratio
Δ	Difference of correction factor
Δt	Elapsed time
δ	Boundary-layer thickness, or ratio of pressure to Standard day pressure $P/14.67$, or scale factor
δ^*	Boundary-layer displacement thickness
γ	Ratio of specific heats
μ	Absolute viscosity
ν	Kinematic viscosity
θ	Ratio of temperature to Standard day temperature $T/518.67$
Θ	Boundary-layer momentum thickness
ρ	Density
τ	Time constant
ϕ	Sonic-flow function (ideal gas)
ϕ_i	Sonic-flow function (real gas)

SCALE FACTORS AND TIME CONSTANTS

δP	Transient total pressure measurement profile scale factor
δP_{12}	Fan tip inlet transient total pressure profile scale factor
δP_{13}	Fan tip discharge transient total pressure profile scale factor
δP_2	Fan hub inlet transient total pressure profile scale factor
δP_{2A}	Average fan inlet transient total pressure profile scale factor
δP_{21}	Transient fan hub discharge transient total pressure profile scale factor
δP_{2A}	Average fan inlet transient total pressure profile scale factor
δP_{24}	Compressor inlet transient total pressure profile scale factor
δP_3	Compressor discharge transient total pressure profile scale factor
δP_5	LP turbine discharge transient total pressure profile scale factor
$\delta_{P/P,FAN}$	Fan efficiency scalar at constant pressure rise
$\delta_{P/P,HPC}$	HPC efficiency scalar at constant pressure rise
δT	Transient total temperature measurement profile scale factor
δT_{12}	Fan tip inlet transient total temperature profile scale factor
δT_{13}	Fan tip discharge transient total temperature profile scale factor
δT_2	Fan hub inlet transient total temperature profile scale factor
δT_{21}	Transient fan hub discharge transient total temperature profile scale factor
δT_{2A}	Average fan inlet transient total temperature profile scale factor
δT_{24}	Compressor inlet transient total temperature profile scale factor
δT_3	Compressor discharge transient total temperature profile scale factor
δT_5	LP turbine discharge transient total temperature profile scale factor

$\delta_{T/T,FAN}$	Fan efficiency scalar at constant work
$\delta_{T/T,HPC}$	HPC efficiency scalar at constant work
$\delta_{T/T,HPT}$	HPT efficiency scalar (or adder)
$\delta_{T/T,LPT}$	LPT efficiency scalar (or adder)
$\delta_{WC,FAN}$	Fan corrected airflow scalar
$\delta_{WC,HPC}$	HPC corrected airflow scalar
$\delta_{WC,HPT}$	HPT corrected airflow scalar
$\delta_{WC,LPT}$	LPT corrected airflow scalar
τ	Time constant
τT	Total temperature response time constant
τT_{12}	Fan tip inlet total temperature response time constant
τT_{13}	Fan tip discharge total temperature response time constant
τT_2	Fan hub inlet total temperature response time constant
τT_{2A}	Average fan inlet total temperature response time constant
τT_{21}	Fan hub discharge total temperature response time constant
τT_{24}	Compressor inlet total temperature response time constant
τT_3	Compressor discharge total temperature response time constant
τT_5	LP turbine discharge total temperature response time constant

1. INTRODUCTION

Turbine engine testing at the United States Air Force's Arnold Engineering Development Center (AEDC) is conducted to evaluate engine operation over a wide variety of simulated altitude conditions. Hundreds of sensors, each producing measurements at rates in excess of one hundred samples per second, are typically installed in the engine and test facility (Fig. 1.1)[1] to measure aerothermodynamic performance. A single sensor can produce nearly 3 million samples of data in 8 hours. Therefore, a couple hundred sensors can produce nearly 300 million samples of aerothermodynamic performance data in 8 hours. The challenge is to ensure the validity of the data, accurately determine engine performance, monitor the condition of the engine, and to promptly identify anomalies. This research is specifically directed at real-time identification of faults during developmental turbine engine tests over a wide variety of simulated altitude conditions.

The countless variations of steady-state and transient engine operation and the necessity to delineate between sensor anomalies and abnormal engine deterioration, combined with the large volume of data, overwhelms the capabilities of traditional data

[1] All figures are in Appendix A.

validation methods (Malloy, 1993). Although traditional methods produce meaningful results, they are labor-intensive and time-consuming. Consequently, application of the methods is typically restricted to a fraction of the available data, which diminishes the ability to detect anomalous data and intermittent events.

An automated approach that emulates and extends traditional data validation and engine condition monitoring processes is needed to ensure a comprehensive assessment. Nonlinear component-matching engine models embody the physical relationships and systems of aerothermodynamic equations employed in the data validation and engine condition monitoring processes, and can provide a basis for automating them. However, in order to provide a sound basis, the mathematical engine performance model must accurately represent the actual test engine.

Calibration of a model to accurately represent a specific engine is also a labor-intensive task. System identification is a technology for determining a mathematical model of a dynamic system from observation of its response to inputs. Although system identification is often regarded as a set of techniques for data processing, the overall technology has a broader scope which includes test planning, instrumentation specification, choice of mathematical model structure, as well as numerical methods of parameter estimation (A. P. Annex, 1990). A two step process was used by Annex, under sponsorship of the Air Force's AEDC, to apply system identification techniques to turbine engine post-stall test and evaluation. First, according to Annex, candidate models

are evaluated by choosing parameters to minimize fit error or mean square prediction error. This evaluation is done using a numerical scheme which accurately evaluates the performance index but which may not accurately evaluate the parameter estimates themselves. Once the model structure is established, the parameter estimates are then refined using a scheme which gives more accurate parameter estimates. According to Annex, a computationally efficient method which is effective for model structure determination is the equation error minimization method; parameter estimates may then be refined using output error minimization or combined state and parameter estimation methods.

System identification techniques surveyed in the following chapter have commonly been applied to reduce the effort required to calibrate a model and have been used most effectively for calibrating simplified models to measurement sets for which data uncertainties are well defined. However, system identification techniques can exhibit numerical divergence if the simulation does not accurately represent the test article or as model complexity increases (e.g., Friedland, 1969 and Trankle, 1982). Consequently, system identification techniques have not been effectively applied to complex transient models in a developmental test environment. Therefore, to take advantage of the relationships inherent in a component-matching model, a more effective calibration technique is required.

Fault identification methods have been demonstrated on systems for which major attributes of the system (e.g., components, configuration, and control) remain unchanged and the same operating conditions are repeated many times. A variety of numerical methods such as expert systems and neural networks and system identification techniques such as Kalman filtering can be used for fault identification. System identification techniques rely on models that are well defined for model-to-data closure and fault identification (Fitzgerald, 1971), while neural network techniques rely on a statistically significant number of samples at a selected operating condition (Doel, 1990) for fault identification. A component-level model for a developmental engine is typically not sufficiently well defined for these techniques and must undergo frequent change to adapt to changes in the engine's attributes. Additionally, because of frequent configuration changes, a developmental engine rarely repeats a specific operating condition, which reduces the applicability of system identification and neural network approaches for fault detection and diagnosis. Consequently, a fault identification approach that is able to adapt to engine system changes is required to enable an automated model-based approach to data validation and engine condition monitoring.

Fault Identification Overview

The model-based fault identification approach described herein relies on an automated real-time model calibration technique and emulates traditional fault identification processes. The technique focuses on single faults as each occurs rather

than on the estimation of an optimal combination of all possible faults (AGARD-CP-448, 1988). However, multiple "error" scenarios are readily evaluated. The approach is adaptable to changes in engine component hardware and control laws that occur in developmental turbine engine testing programs, and relies on a basic component-matching model that represents the turbofan, turboshaft, or turbojet engine cycle. Industry-accepted engine modeling practices are combined with advanced fault diagnostic algorithms and parallel computer techniques to provide real-time fault identification including data validation and engine condition monitoring for steady-state and transient engine operation.

To be effective, gas path analysis tools must accurately characterize component performance and identify component performance deviations and measurement errors. The model-based fault identification process consists of two main phases. The first phase of the fault identification process relies on a real-time model-based evaluation of test data to detect a probable fault resulting from measurement anomalies, engine component events, or both. After a fault is detected, automated model simulation studies are performed to diagnose the most probable cause of the fault in near-real time. This detailed diagnostic information enables the analysis engineer to assess the relative probability of the fault and, in most cases, quickly identify and verify the actual cause of the fault. If necessary, additional test data may be acquired at previously tested stabilized engine operating conditions to increase the fidelity of the diagnosis. The following

sections describe the adaptive engine model and the model-based fault identification approach.

Physics-Based Model Overview

A component-level model (CLM), capable of simulating steady-state and transient engine operation, serves as the basis for the fault identification process. The CLM combines the physical relationships that govern engine operation with empirical relationships that describe individual component performance. The result is an adaptable model in which the effects of changes to engine attributes (e.g., engine hardware, component flows and efficiencies, and controls) are incorporated in the model by making corresponding changes to the model attributes. Additionally, the component matching approach quantifies the changes to engine performance interrelationships, which provide a prediction capability for the fault identification process. In many cases, it is not necessary to predict the absolute values of component and overall engine performance, but rather, to predict trends in behavior as test conditions change. Data faults and engine performance anomalies become readily apparent in the predicted trends, as well as in the absolute behavior.

The CLM is an assembly of components constrained to operate in unison to simulate engine operation (Chappell and McLaughlin, 1993). An augmented turbofan engine, for example, may include a variable-geometry fan and compressor, combustor,

high- and low-pressure turbines, fan bypass duct, mixer, afterburner, and variable exhaust nozzle (Fig. 1.2). The component models combine thermodynamic process equations with empirically determined component performance relationships to simulate component performance (Appendix B). An iterative technique (McLaughlin, 1982) is used to satisfy a set of governing equations in the form of implicit relationships that constrain the assembly to mass, momentum, and energy conservation principles. Measured engine control variables are used as inputs to govern model operation. The effects of rotor acceleration, heat transfer, and off-schedule variable geometry are included, providing a simulation of steady-state and transient engine operation ranging from engine starting conditions to maximum power. The CLM computes temperature, pressure, mass flow, and rotational speed at each inter-component engine station identified in Aerospace Standard 755.

Baseline component performance relationships are an integral element of the CLM; however, they are determined primarily from a baseline component configuration isolated from the engine assembly in a component test rig. Component performance variations that result from operation within the engine assembly are included in the CLM by applying scaling parameters to the baseline performance relationships. The scaling parameters account for the effects of intercomponent interactions, component modifications, off-schedule geometry, and engine-to-engine variations. The scaling parameters are usually applied as multiplicative scalars to flow, pressure ratio, and efficiency relationships and are defined as ratios between measured values and baseline

values. The scalars for the entire engine, comprised of the scalars from all the individual components, are the primary variables used to calibrate the engine model to a specific measurement set. Based on the level of engine development, scalars can be determined from corporate experience, development of the CLM, computational fluid dynamics models, and other sources.

Statement of the Problem

The present research was undertaken to formulate, validate, and apply a CLM-based engine performance and fault identification model to permit real-time fault identification for developmental turbine engine testing. In order for the model to be capable of providing reliable estimates of data quality and component and overall engine performance, the model must effectively quantify measurement error and off-nominal engine behavior; be practical from the viewpoint of the analysis, implementation, tuning, and design; and be reliable and efficient.

Limitations to Current Engine Performance Prediction and Fault Identification Techniques

The author performed a comprehensive review of the state-of-the-art to identify limitations to current engine performance prediction and fault identification techniques. Detailed results of the literature review will be presented in Chapter 2. A brief summary

and discussion of limitations to the current state-of-the-art of engine performance prediction and fault identification techniques, identified in the detailed literature review, is provided below.

- Off-nominal engine behavior is not necessarily quantified and the engine model is sensitive to measurement bias, profile errors, Reynolds number index, and engine-model mismatch.

- Current techniques are insensitive to measured disturbances and relatively small changes in component and overall engine performance.

- The Kalman filter approach is impractical from the viewpoint of the analysis, implementation, tuning, and design. Tuning the filter's dynamic response requires extensive simulations and analysis based on an artificially introduced tuning parameter matrix empirically adjusted to give a good compromise between time response and noise rejection. As a further complication to the analysis, implementation, tuning, and design, the filter's eigenvalues are not directly linked to the tuning parameters and no claim can be made regarding the optimality of the filter. Practicality (or impracticality) depends heavily on one's theoretical training and experience. Therefore, this research serves as a benchmark for the next generation of fault identification approaches, and it will be left to others to argue the practicality of alternative approaches.

- Current techniques require compromises between time response and noise rejection (filter settling times of 8-40 seconds or more for steady-state engine operation).

- Algorithms are model-dependent (typically based on reduced-order linear dynamic models and steady-state engine operation) and, in most cases, model updates are not physically meaningful.

Contribution to the State-of-the-Art:

The CLM-based engine performance and fault identification method developed in the present work brings together features that individually and collectively advance the state-of-the-art. These features are separated into three categories: (1) advancements to effectively quantify off-nominal behavior, (2) advancements to provide a fault detection capability that is practical from the viewpoint of the analysis, implementation, tuning, and design, and (3) advancements to provide a real-time fault detection capability that is reliable and efficient.

The following features of the present method have been found effective to quantify off-nominal engine behavior:

- insensitivity of the engine model to *a priori* assumptions and engine-model mismatch (typical of developmental engine tests);

- insensitivity of the engine model to measurement bias, profile error, and Reynolds number index;

- sensitivity of the engine model to measured disturbances and relatively small changes in component performance;

- incorporation or inclusion of physically meaningful model updates (e.g., multipliers or adders on component flows and efficiencies); and

- identification of small performance changes by incorporating an enhanced component performance map adjustment procedure.

The CLM-based engine performance and fault identification technique of the present research is based on the following design criteria (listed in order of priority). It must be capable of:

- identifying measurement errors (e.g., sensor bias, drift, level shift, noise, or poor response) in facility fuel flow, airflow, and thrust measurements;

- identifying measurement errors in engine aerothermodynamic measurements (rotor speeds, gas path pressures and temperatures);

- identifying measurement errors in engine control sensors (e.g., leaking/biased pressure sensor, slowly responding pressure measurement) and variable geometry rigging (e.g., misset guide vanes or nozzle area) that would invalidate a test or series of tests;

- identifying abrupt faults (e.g., faults due to domestic object damage, foreign object damage, and control anomalies);

- identifying slow faults (e.g., component or overall engine degradation, and sensor drift). Specifically, the technique must be capable of identifying small changes in compressor (or fan) performance resulting from, for example, blade fouling (e.g., accumulation of deposits on the blade surfaces causing an increase in surface roughness due to ingestion of organic or non-organic matter); and

- being easily extended to diagnose secondary failure modes and to verify any modeling assumptions that may arise for developmental engine tests (e.g., increase in turbine flow capacity, inaccurate measurement of facility bleed flows, horsepower extraction, etc.).

The following features of the present method provide a fault identification capability that is practical from the viewpoint of the analysis, implementation, tuning, and design. These features include:

- ease of implementation, tuning, and design;

- methodical approach to analyze available test measurements and uncertainties and component updates to simultaneously detect and diagnose data anomalies and component faults;

- minimization of errors due to simplifications that would otherwise be encountered with a reduced-order linear dynamic engine model;

- analysis of the physics of the system using well-known modeling techniques derived from the fundamental governing equations;

- near-real time determination of the most probable set of measurement errors and component deviations and an estimate of the relative probability of measurement errors or component deviations;

- performance trending.

The following additional features provide a real-time fault identification capability that is reliable and efficient:

- an order of magnitude increase in the model execution speed, as well as an order of magnitude reduction in the settling time (time to convergence), compared to existing techniques;

- real-time fault detection and near-real time fault diagnostics to identify performance anomalies during test maneuvers, giving test engineers the capability to avoid unwanted stalls/surges/collateral damage;

- efficient solution of the governing equations;

- reliable nonlinear model convergence;

- efficient parallel processing algorithms;

- real-time fault detection;

- near-real time fault diagnostics.

Fault Identification Approach

To be effective, gas path analysis tools must identify component performance deviations and measurement errors. The model-based fault identification process consists of two main phases. The first phase of the fault identification process relies on a real-time model-based evaluation of test data to detect a probable fault resulting from measurement errors, engine component events, or both. After a fault is detected, automated model simulation studies are performed in near-real time to determine the source of the failure (e.g. which sensor or actuator has failed) and to estimate the extent and probability of the failure. This detailed diagnostic information enables the analysis engineer to assess the relative probability of the fault and, in most cases, quickly identify and verify the actual cause of the fault. If necessary, additional test data may be acquired at previously tested stabilized engine operating conditions to increase the fidelity of the diagnosis.

In many applications, the problem of fault accommodation is of equal importance to that of fault identification. The fault identification algorithms are designed to process data continually in the presence of faults. For developmental engine testing, it is assumed that the analysis engineer will play a substantial role in reconfiguration of the system in the presence of known faults. Therefore, in order to reduce software complexity associated with automated system reorganization in the presence of known faults, only limited system reconfiguration is performed in real-time.

The fault identification process starts with the best available model of the engine and uses available test measurements to tune the model to account for engine-to-engine variations, engine deterioration, and modeling errors. After the engine model and available test measurements have been identified, the process of determining the best set of sensors and model updates can begin. In general, as the number of sensors and model updates increases, the ability to identify faults improves. However, experience often shows that as more and more sensors are added, more sensor errors are introduced than useful performance information, and the ability to identify faults is reduced (Adibhatla, 1994).

The actual process of selecting the sensor and model updates is based on engineering analysis and an exhaustive computer-based search. Engineering analysis is used to determine all possible test measurements and cycle model updates. The key criterion in the computer-based search is a performance index used to characterize how well measurement errors and changes in component performance can be characterized in addition to how well key performance parameters such as thrust can be predicted in the presence of assumed measurement uncertainties and modeling errors. In order to avoid the need for a detailed model of the engine control system and to accommodate rapidly changing engine control logic and trims associated with developmental engine tests, the fault identification algorithms use measured control variables (e.g., nozzle area, fuel flow, guide vane angles, etc.).

In order to quantify the ability of the fault identification technique to detect and diagnose measurement errors and component performance deviations, an aerothermodynamic model was used to generate data with known errors and component deviations. This aerothermodynamic model, used to develop the fault identification technique, is referred to as a "Truth Model" since it represents the nominal state of the engine used to develop the fault identification technique. Initial studies are performed using a Truth Model with embedded modeling errors and sensor uncertainties to determine the most likely set of sensor/model updates. Follow-on studies and analyses are performed with actual test data, for nominal and degraded engine operation, and "seeded" measurement errors to determine the final set of sensor/model updates. In the absence of any practical constraints such as software complexity, multiple sets of sensor/model updates are required to provide the best estimate of each possible fault. For example, one set of sensor/model updates is required to provide the best estimate of net thrust, and a different set of sensor/model updates is required to provide the best estimate of compressor flow, turbine temperature, stall margin, etc. However, in order to estimate the relative probability of each possible fault, one primary set of sensor/model updates is used. Secondary failure modes which may be associated with an increase in turbine flow capacity or reduction in cooling flow rate, are analyzed using a secondary set of modeling assumptions (e.g. unknown turbine flow area) and a secondary set of sensor/model updates.

An overview of the fault identification process is shown in Fig. 1.3. In Chapter 2, the state-of-the art of real-time fault identification techniques for developmental engine testing is reviewed and key elements of the fault identification approach identified. Chapter 3 provides details of the model development process, test measurements and uncertainties, and steady-state and transient analysis considerations. The engine model of the present method is formulated in Chapter 4 and a three-step calibration process developed. Parallel processing considerations required for real-time model calibration and fault identification are discussed in Chapter 5. Finally, fault detection and diagnosis procedures are developed in Chapter 6. For completeness, the model-based fault identification techniques are verified in Chapter 7, results of the research are summarized in Chapter 8, and recommendations for future work are discussed in Chapter 9.

2. LITERATURE SURVEY

The state-of-the art of real-time fault identification techniques for developmental engine testing is reviewed. The literature survey begins with a brief review of general topics and concludes with a comprehensive review of more specific literature. Accordingly, the literature survey begins with a brief review of system identification techniques, nonlinear state estimators or "observers", and fault identification techniques. Then, literature is presented that addresses model structure determination from test data, along with a review of efforts to model turbine engine compressor and turbine maps. A more detailed review of steady-state, transient, and real-time turbine engine models is then presented. The literature review concludes with a review of in-flight estimation algorithms for adaptive performance optimization, gas turbine fault detection/in-flight condition monitoring systems, gas turbine performance deterioration modeling, and diagnosis of turbine engine transient performance with model-based parameter estimation techniques.

System Identification Techniques

The field of system identification is rather bewildering, even for acknowledged experts. Astrom and Eykhoff (1971) put it best when they state "At many universities the

field has been recognized as a legitimate subject for faculty and Ph.D research. The net result of this development is a large number of publications [they included 230 references in their survey], either accentuating a particular type of approach or describing a particular case study. .... Many different methods and techniques are being analyzed and treated. "New Methods" are suggested en masse, and, on the surface, the field appears to look more like a bag of tricks than a unified subject."

Some of the practical issues that must be considered are reviewed in the present study, but it is left to the reader to review and categorize the limited number of direct comparisons reported in the literature. The effective application of system identification techniques requires the integration of test planning (choice of sensors and of input test signals), monitoring of test execution, and data processing. Many of the practical aspects of system identification which must be considered when choosing and implementing a data processing scheme, have been identified by Trankle (1974). Other considerations in choosing and implementing a data processing scheme include numerical consideration, choosing the model structure and the number of independent parameters to be estimated, and validation of the model. The model structure determination phase outlined by Gupta, Hall, and Trankle (Gupta et al, 1975) consists of processing the input/output data to determine the significant linear and nonlinear equations and associated parameters that are necessary to represent an observed system response. Questions addressed here include the determination of the order of the model (e.g., number of degrees of freedom) and a mathematical form (e.g., polynomial) to represent any nonlinear character or

parameter in the dynamic equations. For linear dynamic systems, the determination of order is of primary importance. For nonlinear systems determination of the mathematical forms to represent the nonlinearities has equal importance.

Many of the practical issues identified by Trankle and Astor and Eykhoff are still valid today. Specifically, how should the data sampling interval be chosen? What is a reasonable model structure? How should an identification experiment be planned? How can it be assured that the *a priori* assumptions required to use a particular method are not violated? Both model structure determination and violation of *a priori* assumptions are treated in greater detail in the present study, but, it is left to the reader to review the literature pertaining to the planning of an identification experiment which includes specifications for both the input signals (Gupta, 1975) and the instrument system (Gupta, 1978).

Even with proliferation of high speed computing systems, Astrom and Eykhoff's conclusions regarding online and real-time identification are worth repeating. They state, "On-line and real-time identification: This is a fiddler's paradise. Much work remains to be done to prove convergence as well as to devise approximation techniques." Their comments regarding the use and impact of available empirical knowledge on the methods are extremely relevant for developmental engine tests. They add, "Most methods seem to work well on simulated data, but not always that well on actual industrial data. This indicates that some methods might be sensitive to the *a priori* assumptions. It therefore

seems highly desirable to develop tests which ensure that the *a priori* assumptions are not contradicted by the experimental data. It also means that it is highly desirable to have techniques which are flexible with respect to *a priori* assumptions. A typical example is the assumption that the measurement errors or the residuals have a known covariance function. It is of course, highly unnatural from a practical point of view to assume that we have an unknown model but that the residuals of this unknown model have known statistics."

A review of process identification and parameter estimation techniques by Nieman et al (1971) reiterates the views of Astrom and Eykhoff. Nieman states, "... there are a great many different methods for process identification and parameter estimation and an abundance of literature dealing with their development and application. ... Discussion of reliability estimates, relationships to other methods, effects of noise, practical aspects of implementation, etc. would make the presentations more complete and the use of standardized examples would facilitate direct comparison of methods."

Nonlinear Observer Models

A more recent review of nonlinear observer models by Misawa and Hedrick (1989) provides an evaluation of the computational requirements and their robustness properties in terms of the extent to which the design requires a "perfect" model. The methods discussed by Misawa and Hedrick are the extended Kalman Filter (Gelb, 1974),

the Constant Gain Extended Kalman Filter (Sorenson, 1985), the Statistically Linearized Filter (Phaneuf, 1968), observers designed using either Global Linearization (Bestle and Zeitz, 1983) or Pseudo Linearization (Nicosia et al, 1986, and Reboulet and Champetier, 1984) methods, observers based on Extended Linearization (Baumann and Rugh, 1986), observers designed according to Thau's method (Thau, 1973), adaptive observers (Gevers and Bastin, 1986), observers that use the set-theoretic point of view, and sliding observers (Slotine, 1984).

Kalman (1960) introduced the concept of an optimal linear filter that minimizes the mean square estimation error. A key assumption is that the model represents the real system exactly. A natural extension for nonlinear systems is the Extended Kalman Filter derived by minimizing the trace of the covariance matrix of estimation errors (Gelb, 1974). In spite of being widely used, the extended Kalman filter has some drawbacks worth stating. Specifically, robustness against modeling errors can not be guaranteed *a priori*, and even with computer speed doubling every year, implementation of the Kalman filter requires a great deal of real-time computer power since the filter and covariance equations are coupled. Misawa (1989) adds, "the lack of guaranteed robustness and difficulties in implementation motivated the investigation of the Constant Gain Extended Kalman Filter".

Detailed development of the Constant Gain Extended Kalman filter is provided by Safanov (1978, 1980) and is not repeated here. Detailed discussions of the robustness of

the Extended Gain Kalman Filter are discussed later in the present work with respect to the estimation algorithms for adaptive performance optimization of turbofan engines. As stated by Misawa (1989), "In spite of their guaranteed robustness, the margins . . . can still be quite restrictive, e.g., when one deals with hard nonlinearities. Also, considering that it is likely that the matrices ... were determined by linearization about operating points, the differentiation of the equation of motion involved in the linearization process might also unfavourably affect the guaranteed robustness properties." In practice, tradeoffs in robustness (the extent to which the nonlinear state estimator or observer requires a perfect model) and settling time (the time required for the observer to converge to the estimated parameters) are required.

Fault Identification Techniques

In general, the choice of an appropriate fault identification design depends heavily on the application. This section provides an overview of a number of the basic concepts in failure detection and, where appropriate, identifies key issues that have been considered for the model-based turbine engine fault identification technique developed herein. The fault identification techniques discussed in this section follow from a survey of design methods for failure detection in dynamic systems completed by Willsky (Willsky, 1976).

In general, failure detection consists of three tasks: alarm (fault detection), isolation (fault isolation), and fault severity estimation (diagnosis). Using Willsky's terminology, the alarm task simply consists of making a binary decision – deciding that either something has gone wrong or that everything is fine. The problem of isolation is that of determining the source of the failure; finding which sensor or actuator has failed. Finally, the fault severity estimation problem involves the determination of the extent of the failure. For example, it must be determined when a sensor becomes completely non-operational, or suffers degradation in the form of a bias or increased inaccuracy.

Willsky (1976) states that "... the most reliable means of failure detection is straight voting among like instruments. In this case one never runs into errors introduced by comparing the outputs of dissimilar devices. However, one pays the price in hardware redundancy in order to implement a voting scheme. In addition, voting has its built-in limitations. Like instruments may not be exactly alike, and one may have to use other instruments to compensate for these discrepancies. ... In addition, voting cannot be used to detect failures that affect both instruments in the same way, e.g. common power supply failures, common thermal effects, etc., and once a single instrument in a triad has failed, we can no longer perform failure detection by voting. Finally, voting techniques may have difficulties in detecting subtle degradations in instrument behavior."

Most fault identification techniques have evolved to address the drawbacks of voting techniques. Key issues that must be considered in comparing the merits of the various fault detection systems include:

- Types of failure modes that must be considered;
- Complexity in implementation;
- Performance, as measured by false alarms, delays in detection , etc.;
- Robustness in the presence of modeling errors.

With respect to performance issues, the fault identification technique developed herein attempts to minimize false alarms and missed detections through simultaneous (or nearly simultaneous) detection, isolation, and estimation of faults. Most fault identification techniques are designed to respond rapidly when a failure occurs. Willsky states, "a system that is designed to respond quickly to certain abrupt changes must necessarily be sensitive to certain high frequency effects, and this in turn will tend to increase the sensitivity of the system to noise, via the occurrence of false alarms signaled by the failure detection system." *A primary design consideration of the present study was minimizing tradeoffs between noise sensitivity and response, and development of an algorithm capable of minimizing false alarms and missed detections while also estimating the probability of a fault.*

For completeness, some of the basic issues regarding failure-sensitive filters, voting systems, and multiple-hypothesis filter-detectors will be discussed. The first-class of failure detection concepts discussed addresses the problem of an "oblivious filter".

As discussed by Willsky and many others, the optimal filter performs well if there are no modeling errors; however, it is possible for the filter estimate to diverge if there are substantial unmodeled phenomena. The problem occurs because the filter learns the state too well, relies on old measurements for its estimates, is oblivious to new measurements, and responds quite sluggishly. Two well-known techniques for keeping the filter sensitive to new data are the exponentially age-weighted filters and limited memory filters. Other techniques include increasing noise covariances or simply fixing the filter gain. As stated by Willsky, "It is important to note a performance tradeoff evident in this method. As we increase our sensitivity to new data by effectively increasing the bandwidth of the Kalman filter, our system becomes more sensitive to sensor noise, and the performance of the filter under no-failure conditions degrades. In some cases this can be rather severe, and one may not be able to tolerate the degradation in overall system performance under no-failure conditions." Willsky adds, "... the tradeoff between detection performance and filter behavior under normal conditions is a characteristic of all failure detection systems and is analogous to the costs associated with false alarms and missed detections in standard detection problems."

Willsky appropriately summarizes voting schemes stating, "Voting schemes are in general relatively easy to implement and usually provide fast detection of hard failures, but they are only applicable in systems possessing a high level of parallel redundancy. They do not in general take advantage of redundant information provided by unlike sensors, and thus cannot detect failures in single or even doubly redundant sensors. In addition, voting techniques can have difficulties in detecting 'soft' failures, such as a small bias shift." *The model-based fault identification technique developed herein takes advantage of the redundant information provided by unlike sensors and can detect failures in singly redundant sensors and 'soft' failures.*

In general, a properly designed failure detection system takes system redundancy into account. *Accordingly, the fault identification technique developed herein is based on the assumption that simpler checks (such as range checks and redundancy checks) are performed prior to initiation of the more complex model-based fault identification algorithms.* More complex analytical redundancy checks such as a comparison of the measured turbine exit temperatures and pressures and the corresponding turbine exit temperatures and pressures determined from the model-based governing equations are, however, an integral part of the fault identification algorithms.

Model Structure Determination from Test Data

Functional forms for the model structure include polynomial, orthogonal polynomial, spline, and B spline fits and should not be confused with the aerothermodynamic turbine engine model referenced throughout this document. Gupta et al (1978) presents a fairly comprehensive review of the critical elements of an advanced model structure determination procedure for dynamic systems. According to Gupta, implementation of methods to develop models from test data must consider the following factors:

1) The number of hypothesized models may be very large (more than a billion is common).
2) All models contain some level of modelling error (i.e., no model of a physical system is perfect). The models are "good" or "poor," not "right" or "wrong."
3) Many physical systems are dynamic. The modelling of system dynamics may improve the model structure estimates.
4) Distribution functions usually are not known and must be approximated.
5) It should be possible to incorporate the analyst's opinion.
6) Computational time should be reasonable (because of factor 1).

The critical elements of such a procedure are 1) specification of functional forms for useful models, 2) criteria for selecting an adequate model against competing models, and 3) efficient numerical algorithms for integrating the conflicting requirements of adequate functional forms and accurate criteria. Gupta's research on the model structure determination process for nonlinear systems focused on three significant issues: 1) specification of classes of *a priori* models, 2) criteria for hypothesis testing of data against models, and 3) numerical procedures.

Criteria for validation of models against test data include fit error (error covariance, whiteness test, and fit errors corrected for degrees of freedom), likelihood approach (likelihood ratio and log likelihood ratio corrected for degrees of freedom), prediction error (direct determination over an independent data set, estimate over same region as data, and estimate over a new region), and F-ratio (equation F-ratio, parameter F-ratio, and Parzen's test). According to Gupta, the F-ratio is perhaps the most widely used statistic for model hypothesis testing. F tests are based on the assumption of normally distributed random disturbances and require *a priori* specification of the acceptance-rejection boundary. Parzen's method is based on the F-statistic and assumes that the "true" model is very complex and has several degrees of freedom (Parzen, 1974). Since the measured data contain noise and do not encompass the entire operating regime, the estimated model must be a simplified version of the true model. Parzen's method then compares the simplified model to a higher order model using an F-statistic.

It should be noted that the ultimate model selection criteria depend mainly on the use of the model (e.g., to detect and diagnose faults in real-time for developmental engine testing, predict data, etc). Typically, turbine engine component performance maps are introduced in the form of tabulated functions requiring some form of interpolation. A continuing effort has been conducted by turbine engine manufacturers to reduce the computer memory required to represent component maps and to increase the accuracy of the representations. In the absence of empirical data or detailed component geometry, consistent off-design performance data can be derived from design point input. Compressor and turbine map fitting systems such as those developed by the General Electric Aircraft Business Group (NASA-CR-174646) are based on similarity parameters derived from the basic physics of the components and design point input. Sieros et al (1997) describe an effort to model the compressor and turbine maps for performance modeling and diagnosis using analytical functions. Mean error (difference between measured value and calculated value by the model) for map mass flow, pressure ratio, and efficiency was on the order of one percent. While the advanced modeling methodology described by Sieros et al results in mean errors on the order of one percent of measured values there are certain areas where the error is substantially higher than the mean error. For early cycle analysis of advanced propulsion systems, component map characteristics are generally not well known. *Improved techniques developed in this research permit identification of changes in component characteristics on the order of one half of one percent.*

Steady-State and Transient Gas Turbine Performance: Prediction, Analysis, and Identification

A component-level model (CLM), capable of simulating steady-state and transient engine operation, serves as the basis for the fault identification process (Chappell and McLaughlin, 1993). The CLM combines the physical relationships that govern engine operation with empirical relationships that describe individual component performance. The result is an adaptable model in which the effects of changes to engine attributes (e.g., components, configuration, and controls) are incorporated by making corresponding changes to the model attributes. Additionally, the component matching approach quantifies the changes to engine performance interrelationships, which provide a prediction capability for the fault identification process.

Chappell and McLaughlin's work, and the literature they review, generally represents the current state of the art with respect to gas turbine engine steady-state and transient simulation capabilities. More recent work has been completed in conjunction with the National Cycle Program (NCP) (Parsons et al, 1998) to develop a flexible system modeling architecture incorporating the most current modeling technologies.

Work done in France (Duponchel et al, 1990) continues to focus considerable attention on steady-state and transient engine performance prediction, analysis, and identification techniques. The remainder of this section provides an overview of the basic identification methodologies highlighted by Duponchel and, where appropriate,

identifies key issues that have been considered for the model-based turbine engine fault identification technique developed herein. The methods outlined by Duponchel include single-point and multi-point identification methods. The choice of method used depends heavily on the level of engine and model development and the available measurements.

The method for computing the component deviation parameters depends on the number of knowns and the number of unknowns. When the number of unknowns is greater than the number of knowns, either a Kalman filter or a "minimum-right-inverse" technique may be used. The Kalman filter estimator minimizes or balances the errors between the knowns and unknowns using weighting factors while the "minimum-right-inverse" estimator finds the best solution by minimizing the unknown errors. When a Kalman filter is used to estimate unmeasurable engine parameters, component deviation parameters can be thought of as complex engine efficiency parameters (i.e., an aggregation of component efficiencies that include the compressor and high turbine, and fan and low turbine, respectively). When the number of knowns equals the number of unknowns, an inverse technique is used to find the solution and component deviation parameters are simple component flow and efficiency scalars. Identification methods, where the number of knowns equals the number of unknowns, are commonly referred to as "square" identification methods.

Square-single-point identification methods typically use only a limited set of measurements corresponding to the number of unknown model parameters (e.g., flow and

efficiency correction factors on the various components). The non-linear equation system comprised of compatibility equations (identity between measurements and simulated parameters, airflow rate continuity, power balance, confluence static pressures, etc.) can then be solved by a Newton-Raphson type method (Broyden, 1967).

Benefits of the square method include use of only a limited number of measurements (leaving excess measurements for validation of test data and modeling assumptions). The square method also has the advantage of being very sensitive to measured disturbances, making it ideal for analyzing small performance deviations. *Accordingly, a variation of the square method has been used in the present study for the fault identification process.*

Real-Time Turbine Engine Models

SAE Aerospace Information Report (AIR) 4548 provides a review of real-time modeling methodologies for gas turbine engine performance. The modeling methodologies addressed in this AIR concentrate on the aerothermal portion of the gas turbine propulsion system. Characteristics of the models, the various algorithms used in the models, and system integration issues are also reviewed. For completeness, overviews of the aerothermodynamic, piece-wise linear, and transfer function real-time modeling methodologies for gas turbine engine performance are provided in Appendix C.

The aerothermodynamic real-time engine model can best be defined as a cycle model, which has been modified for transient simulations with constraints on consistency, size and computational speed. *Improved modeling techniques developed in this research are combined with efficient parallel processing to eliminate the typical constraints on consistency, size, and computational speed and maximize the potential of the aerothermodynamic model for real-time fault identification.*

In-Flight Estimation Algorithms for Adaptive Performance Optimization

The application of fault identification techniques using real-time model-based techniques has been hampered by some challenging problems. Real-time fault identification requires an accurate measurement of basic engine parameters and unmeasurable or poorly measurable parameters. Fault identification studies have typically been performed using engine model simulations. Compounding development problems, flight test evaluations have been limited to existing engine control measurements even though performance could have been improved with additional measurements (Orme and Gilyard, 1992). Key variables that must be estimated from in-flight data include pressures, temperatures, flows, efficiencies, stability margins, and thrust.

Flight test results showed that current propulsion system parameter estimation techniques produce reasonable engine estimates (to recover latent performance from the

propulsion system) when the effects of measurement biases, flight conditions, and engine deterioration are excluded (Maine et al, 1990). In a more recent flight test program (Orme and Gilyard, 1992), test points were specifically designed to measure the effectiveness of a propulsion system parameter estimator. An evaluation of the flight test data determined that well-behaved estimates were produced that accurately reflected the state of the engine and minimized dynamic errors. These dynamic errors were attributed to sources that included approximation of actuator, heat soak and sensor dynamics, errors in the computation of rotor accelerations, and unmodeled dynamics. However, estimates of component deviation parameters did not accurately reflect known engine degradation. The majority of the model estimates reported in the flight test program were within a 3-percent error band at 30,000 feet and high subsonic Mach numbers (Orme and Gilyard, 1992). The error in the estimated high-pressure turbine inlet pressure ranged from 3 percent at 45,000 feet to 5.5 percent at 15,000 feet. For degraded engine operation, the estimated engine thrust was up to 7.5 percent greater than the thrust stand measured value. The sensitivity of engine component deviation parameters to altitude and bias measurement effects indicates the need for improved modeling techniques.

The performance seeking control algorithm (Luppold et al, 1989, Gilyard and Orme, 1992, Nobbs et al, 1992, and Orme and Gilyard 1992, 1993) uses a nominal or baseline propulsion system model and estimates, in-flight, Engine Deviation Parameters (EDPs) to characterize engine deviations with respect to nominal conditions. Espana

(1993) showed that with the measurement system used, unknown biases cannot be estimated independently of the engine deviation parameters.

Espana (1993) summarizes the optimality and practicality of Kalman filter approach (to estimate the engine deviation parameters) from the viewpoint of the analysis, implementation, tuning, and design. "Because of the artificial introduction of the matrix Q_η (a tuning parameter matrix empirically adjusted so as to give a good compromise between time response and noise rejection), no claim can be made concerning the optimality of the resulting filter. Furthermore, . . . the filter's eigenvalues ... are not directly linked to the tuning parameters (entries of the matrix Q_η). This complicates the tuning of the filter's dynamic response, which can only be done after extensive simulations."

The open-loop nature of the performance optimization control scheme makes it very sensitive to model-engine mismatches and, in particular, to measurement biases. Espana (1993) summarizes the effects of model-engine mismatch as follows. "Some recent results show that the other off-nominal conditions not accounted for by the EDPs affects the EDPs' estimates. For instance (Orme and Gillyard, 1992) ... shows, using flight data, that changes in the Reynolds number index and biases in the measured variables have as a consequence that the estimated EDPs do not reflect known levels of engine degradation. Also, (the model) ... may not represent the current linear 'tangent' model of the engine; this fact can result from: (1) the intermediate simplifications

necessary to obtaining it; (2) the discretization of the space of local models; or (3) neglected second order effects of the EDPs on the matrices ..."

In order to improve upon the accuracy of the in-flight estimation algorithms, *fault identification studies are performed in the present study using engine model simulations and actual test data. Additionally, using both test facility and engine control measurements enhances the fault identification process developed in the present work. Moreover, the improved modeling techniques developed in the present research accurately account for altitude effects, measurement bias, and known engine degradation.*

Gas Turbine Fault Detection/In-Flight Condition Monitoring Systems

A summary of engine monitoring activity can be found in SAE Aerospace Standard SP728, "Aircraft Gas Turbine Engine Monitoring Systems".

The following SAE Aerospace Recommended Practices, Information Reports, and Resource Documents provide detailed guidelines for aircraft engine gas turbine monitoring systems:

ARP1587 Aircraft Gas Turbine Engine Monitoring System Guide

AIR1871B Lessons Learned from Developmental and Operational Turbine Engine Monitoring Systems

AIR1900 Temperature Monitoring

AIR4061 Guidelines for Integration of Engine Monitoring Functions with On-Board Systems

AIR4175 Ground Station for Condition Monitoring

ARD50002 A Discussion of Standardized Concepts for Condition Monitoring and Performance Analysis Software.

AIR5120 Engine Monitoring System Reliability and Validity (DRAFT)

Other documents that are of general interest or importance to engine monitoring include:

AIR4176 Cost Versus Benefits of Engine Monitoring Systems

AIR5137 On-Board ACMS Report Guideline (DRAFT)

Urban (1980) and Volpini (1982) developed gas-path analysis techniques to reduce the sensitivity of on-wing gas-path analysis tools to measurement error. As a result, modern gas path analysis techniques use similar or closely related algorithms. Commonly used gas turbine fault detection and condition monitoring systems employed by the Pratt & Whitney Aircraft Company, the Rolls-Royce Engine Company, and the General Electric Aircraft Engine Company are discussed briefly, as follows.

STORM (Self-Tuning On-Board Real-Time Model) is an integral part of the Pratt & Whitney Aircraft Company engine monitoring/data management (EM/DM) system. STORM consists of an Observer (Kalman Filter) and a Predictor which calculate engine process related parameters in real-time. The Observer compares model inputs to engine sensor values and adjusts component efficiencies so that the model estimates match the measured values. The Predictor calculates engine process values based on effector information and adjusted component efficiencies. The Observer and Predictor each consist of a State Variable Model (to model the gas generator) and an aerothermal model (to model the augmentor).

COMPASS (Condition Monitoring and Performance Analysis Software System) is a ground-based engine computer system developed by the Rolls-Royce Engine Company for application on a variety of civil and military turbofan engine applications (Provost, 1988). COMPASS can be considered to consist of four modules: an analysis

module; a trend module; a plot module; and a utility module. Data for COMPASS is generated from four major areas (on-wing data, ground data, test-cell data, and maintenance action data). Module performance analysis and sensor biases are determined using Kalman filtering techniques.

TEMPER (Turbine Engine Module Performance Estimation Routine) is a gas-path analysis tool developed by General Electric Aircraft Engine Company for commercial jet engines. TEMPER uses a weighted least squares algorithm to determine engine state in the presence of sensor errors. Exponential smoothing is used to smooth the performance deterioration and sensor error analysis results. These smoothed analysis results for module deterioration and sensor error (called "running baselines") provide an *a priori* estimate for the analysis of new data.

Commonly used gas turbine fault detection and condition monitoring systems employed by the Pratt & Whitney Aircraft Company, the Rolls-Royce Engine Company, and the General Electric Aircraft Engine Company represent a clear improvement over previous techniques. Nevertheless, there are still significant problems associated with commonly used gas-path analysis tools which can lead to misdiagnosis of problems and expensive and unnecessary repairs. One frequently encountered problem is misdiagnosis of recently overhauled components as deficient. *Problems with commonly used fault detection and condition monitoring systems such as unclear analysis assumptions and difficulty in interpreting the results of the gas-path analysis are addressed in the present*

research. Furthermore, techniques are developed in the current research to select the test measurements, modeling assumptions, and model calibration techniques to lessen the tendency to underestimate (or overestimate) hardware degradation.

Gas Turbine Performance Deterioration Modeling

The present survey of gas turbine performance deterioration data and models is limited to historical results in the open literature. *In order to better interpret the gas-path analysis results and lessen the tendency to underestimate (or overestimate) hardware degradation, gas turbine performance deterioration modeling techniques are an integral part of the CLM-based engine performance and fault identification method developed in the present research.*

A good example of gas turbine performance deterioration modeling work, recently completed by Zaita et al (1997), uses historical data from the JT9D jet engine diagnostic program (Sallee, 1980) and Navy F404 endurance testing. Their work builds on the compressor flow and efficiency adjustment methodologies developed by Aker and Saravanamutto (1989), Tabakoff (1986 and 1987), and Muir et al (1989) and takes into account performance deterioration due to fouling, erosion, and rubbing wear. Compressor flow and efficiency adjustments by Zaita et al are made on a stage by stage basis using the stage stacking methodology developed by Muir et al (1989) and the fouling, erosion, and rubbing wear methodologies developed by Aker and Saravanamutto

(1989) and Tabakoff (1986 and 1987), respectively. Zaita et al's turbine performance deterioration methodology builds upon the works of Mukhtarov and Krichakin (1969), Ainley and Mathieson (1951), Balje' and Binsley (1968) and engine manufacturer's internal loss models. They divide turbine losses into six categories:

1) Profile loss;
2) Profile incidence loss;
3) Secondary flow loss;
4) Blade tip clearance loss;
5) Cooling air injection aerothermodynamic loss; and
6) Cooling air thermodynamic loss.

Deterioration studies performed by Pratt & Whitney Aircraft and General Electric Corporation on the JT9D and CF6 commercial airline engines found that the major causes of turbine performance deterioration were an increase in tip clearance and flow capacity (mass flow rate). Zaita accounts for the decrease in turbine efficiency caused by increased tip clearance using the Egli method which estimates the tip loss by assuming that leakage flow does no work in the turbine. The Egli method calculates the leakage flow over a turbine blade based on pressure ratio and clearance height. The amount of turbine work is then reduced based on the leakage flow to total gas flow ratio.

Diagnosis of Turbine Engine Transient Performance with Model-Based Parameter Estimation Techniques

A recently completed survey by Bird and Schwartz (1994) summarizes the current capabilities with respect to turbine engine transient engine modeling capabilities, system identification modeling capabilities, and limitations to current dynamic modeling and diagnostic capabilities. A summary of their findings is repeated below.

"In summary, component-level performance models based on physical processes have become useful prediction tools. These non-linear models require extensive data on engine and aerothermodynamic and control parameters. Such models have been tuned to match test data but usually only for small perturbations about steady state operating points. For models based on physical processes, the near term apparent needs for further development of diagnosis techniques are prioritized as follows:

1) methods to tune physical models to match test data with and without faults;
2) component models with appropriate functional descriptions (model fidelity vs. data required);
3) non-linear modelling methods for large transients; and
4) methods to select the suite of measurements needed to isolate particular faults.

This apparent need for improved non-linear modelling and validation methods has been confirmed by a general review of research needs by the ASME (Richardson, 1988)."

Bird and Schwartz claim that:

"System identification is becoming a more widely used tool in the modelling of gas turbine components and complete engine systems. SI methods can accommodate the stochastic nature of engine data by including any known characteristics of noise and measurement uncertainty. Some methods estimate the parameters of noise models including biases, scale factors and random errors. MIMO [Multiple Input Multiple Output] and non-linear systems may be handled with some limitations by the selective use of combinations of current techniques. However, selection of an optimal SI model structure is considered an open question and demonstrations of SI methods in the presence of noise levels relevant to engine tests are few. With un-modelled dynamics, control applications using adaptive methods have been shown to have limited robustness to noise.

Little explicit modelling of the physical system has usually been included in the SI studies. Simplified model structures and limited ranges of applicability have been necessary to handle the non-linearities of engine performance. Experimental programs and studies have identified some of these limitations of certain SI methods for gas turbine

modelling and diagnosis. However, integration of physical modelling and SI techniques should provide some improvements, for example, in the selection of model structure."

With the goal of improving predictions usable for diagnostic purposes over wide operating ranges, problems identified in the Bird and Schwartz review include:

1) "physical model development requires extensive data which is not readily derived from tests;
2) difficulty in developing component models with appropriate functional descriptions to balance model fidelity against data and test requirements;
3) faults in control systems and actuators are not usually included;
4) most models use local linearized dynamics limiting the use of the models for large inputs;
5) few methods are available to tune transient physical models to match test data with and without faults;
6) SI applications make limited use of knowledge of physical processes: linear or simplified non-linear SI models have parameters not directly linked to particular component functions to aid fault isolation and maintenance; and
7) Robustness of the existing modelling methods, in the presence of noise, is largely undemonstrated."

The CLM-based engine performance and fault identification method developed in the present work addresses the shortfalls in turbine engine transient engine modeling capabilities, system identification modeling capabilities, and limitations to current dynamic modeling and diagnostic capabilities identified above by Bird and Schwartz.

Contribution of the Present Study to the State-of-the-Art

With this brief review of the literature, it can be argued that the current state-of-the-art does not satisfy the requirements for real-time fault identification during *developmental engine testing*. As previously stated, the CLM-based engine performance and fault identification method developed in the present work individually and collectively advances the state-of-the-art in areas which include:

- system identification techniques;
- nonlinear observer models;
- fault identification techniques;
- model structure determination from test data;

- steady-state and transient gas turbine performance: prediction, analysis, and identification;

- real-time turbine engine models;

- in-flight estimation algorithms for adaptive performance optimization;

- gas turbine fault detection/in-flight condition monitoring systems;

- gas turbine performance deterioration modeling; and

- diagnosis of gas turbine engine transient performance with model-based parameter estimation techniques.

3. MODEL DEVELOPMENT

The scope of this research is limited to steady-state and transient tests of engines that would be performed in ground-based test facilities (either sea-level test cells or altitude cells). The specific problems associated with measurement errors of dynamic conditions are not considered, because they are test dependent problems that require specification of an instrumentation system and test inputs (i.e., test maneuvers to excite the dynamic modes of interest).

Typical developmental engine performance tests used to assess, isolate, and correct deficiencies in engine performance in both "uninstalled" (baseline) and installed engine test configurations and operability tests to verify satisfactory operability characteristics such as engine starting and power transients are outlined in the following sections. Then, an overview of typical data filtering techniques and test measurements and uncertainties used as the basis for the fault identification process is presented. Finally, performance of each engine component is characterized by the aerodynamic and thermodynamic equations relating pressure, temperature, and mass flow and typical measured, known, assumed, and desired parameters are identified.

In general terms, measurements fall into one of three different classifications that describe the unsteadiness of the measurand: steady-state, transient, and dynamic. The classifications for steady-state, transient, and dynamic conditions used in this document follow very closely with the classifications established by AGARD, the Advisory Group for Aerospace Research and Development (AGARD-AR-320).

Steady-state conditions are those where the machine is running at a nominally fixed operating point and the measurand is essentially constant with time. Even in steady-state conditions it must be recognized that that there may be substantial flow unsteadiness in an engine due to the presence of rotor passage wakes, turbulence, etc., which could significantly influence the performance and calibration characteristics of the test measurements. Also, the machine under test cannot normally be held at a precisely constant condition and so over a period of time a sequence of slow variations about a nominal test condition must be expected. Included in this concept of steady-state conditions is the requirement that the machine should be allowed to reach equilibrium conditions in terms of stable aerodynamic and/or structural temperature gradients, and stable component operating point or engine match point, before the definitive measurements are recorded.

Transient conditions are those in which a variation of machine operation, from one steady-state condition to another, is deliberately induced (e.g. engine acceleration) by an operator or controller.

Dynamic measurements are those in which the measurands are varying at a high frequency and for which instantaneous point measurements or their spectral behavior characteristics are required to adequately define the dynamically-varying parameter or flow conditions in the engine. Special instrumentation and data acquisition and storage systems are required to analyze dynamic engine behavior.

Modeling of the engine-cycle during transient operation is based on the conservation of mass, momentum, and energy. For transient engine operation, time-dependent modeling terms are superimposed upon the steady-state component characteristics. The three dynamic effects considered are rotor dynamics, heat transfer, and volume dynamics. At a sea-level static operating condition, engine acceleration from idle to 95% of full power requires from 5-10 seconds, depending upon the particular engine type. Over this short time period, nearly half of the fuel energy added is stored in the dynamic effects. For a typical engine (Gas Turbine Systems Analysis Handbook 1981) 30.6% of the energy is stored in the increased angular kinetic energy (rotor dynamics), 11.6% is stored in the engine components by heat transfer, and 0.5% is stored in the flow compression (volume dynamics). For moderate accelerations, rotor dynamics and heat transfer are the most important of the three dynamic effects. For dynamic engine operation (not specifically addressed in this research), volume dynamics is the most important of the three effects and measurement uncertainties can have a profound effect on the fault identification sensitivity.

The frequency response of the fault identification techniques developed in the present research depends heavily upon the data acquisition and processing systems. Special pretest consideration is required to establish the sampling rate required to accurately characterize transient engine behavior. Dynamic engine events such as surge require that parameters be monitored at rates of up to 500 to 1000 samples per second. It is desirable to sample the measurement at 10 to 20 times the highest frequency of interest. *The fault identification techniques developed in the current research were developed assuming online data sampling rates of 100 samples per second. In general, no special test instrumentation is required for fault identification during developmental turbine engine tests. Moreover, the three-step engine model calibration process described in Chapter 4 can be performed in conjunction with normally scheduled test activities.*

Typical Developmental Engine Tests

This research is specifically directed at developing a method for real-time identification of faults during developmental turbine engine tests. As previously discussed, the scope of this research focuses on typical steady-state and transient engine tests that would be performed in ground-based test facilities. It is usually a requirement, wherever possible, to make the transient instrumentation capable of covering the steady-state requirements. Typical developmental engine performance tests to assess, isolate, and correct deficiencies in engine performance in both "uninstalled" (baseline) and

installed engine test configurations and operability tests to verify satisfactory operability characteristics such as engine starting and power transients are outlined in the following sections.

Performance Test and Evaluation

Current turbine engine performance test and evaluation techniques employed in altitude test facilities during steady-state engine testing are described by Savelle and Prufert (1992). Turbine engine performance test and evaluation is performed in every phase of an engine development program from initial engine development through production verification to component improvement and performance improvement modifications. Steady-state engine performance data are acquired at selected simulated flight and engine operating conditions after the engine has achieved a stable operating condition. As developmental testing proceeds, engine component performance data are typically used to calibrate and validate mathematical models to account for a variety of operational effects such as aircraft installation effects, viscosity corrections, untwisting of the blades under load, extended exhaust system performance, and fuel temperature effects. Performance derivatives for the effects of customer bleed and horsepower extraction on engine cycle performance are evaluated to determine specific engine-airframe installation effects. In addition, engine performance trends as a function of engine operating time are established for engine health monitoring and evaluation of engine performance retention for life assessment. High-accuracy steady-state measurement systems exist for thrust, flow rates, pressure, temperature, and rotational

speeds so an accurate determination of engine performance can be made. Instrumentation capable of quantifying transient behavior is generally not required for performance test and evaluation.

Operability Test and Evaluation

Current turbine engine operability test and evaluation practices employed in altitude test facilities are discussed by Wantland (1991). Comprehensive evaluation of turbine engine operability is a multi-phase task and typically includes an assessment of airstart capability, power and thrust vector transient operation, compression system stability, control authority (including mode transfers and fault accommodation) and combustion system instability. Transient test data are typically used by the engine developer to verify engine operability and handling, determine engine structural changes (blade tip and seal clearance, actuator movements, etc.), verify engine control schedules and settings (to ensure surge free operation, widest relight envelope, minimum acceleration and deceleration time, optimum afterburner thrust augmentation and buzz and screech-free operation), and analyze differences between the actual engine behavior and predicted engine behavior obtained from mathematical models. Of the different types of transient testing discussed, each typically requires a different array of instrumentation. Time-dependent analysis is a fundamental requirement for analysis of transient data. Typically, the measurement objective is to determine the amplitude of a disturbance, and its time dependence (ranging from a few milliseconds to several seconds) and to correlate the time-dependent behavior of two or more measurands, in

time and space. In some cases it is sufficient only to establish the chronology of observed events.

Test Measurements and Uncertainties

Typical altitude test facility developmental turbine engine test measurements and uncertainties have been used as the basis for the fault identification process. The measurement uncertainties reported herein can best be described as a mix of what AEDC and other test facilities have reported in the past. For reference, typical AEDC measurement uncertainties are provided in Appendix D. The most important consideration relative to measurement uncertainty is whether the test objectives can be satisfied with the measurement method used. Turbine engine measurement uncertainties referenced herein are generic, as opposed to application specific. As would be expected, measurement uncertainty is influenced by many variables such as measurement range, type of measuring device, measurement environment, type of recording device, and the method of system calibration. Identification of measurement errors in facility fuel flow, airflow, and thrust measurements is of primary importance in development of the fault identification process described herein.

The range in acceptable measurement uncertainties varies from measurements requiring extensive design and certification efforts to less accurate measurements

obtained when test objectives can be achieved with less costly measurement methods. The uncertainty model utilized in the present research combines random and systematic error components using a root-sum-square uncertainty model (U_{RSS}), which typically implies that the true value is expected to lie within plus or minus U_{RSS} of the measured value 95 percent of the time. Details of the uncertainty model used are provided by Abernathy (1973) and are not repeated here.

Mean Flow Properties and Profile Effects

Mean flow properties at component interface planes are required to accurately identify faults during developmental engine testing. The shape of flow profiles will have a significant influence on the number of sensors required to characterize the mean flow properties and the type of data averaging and the calibrations required to relate local measurements to the mean properties (Doeblin 1975).

In general, transient flow profiles are not assumed to be directly related to steady-state profiles. In the absence of instrumentation and/or calibration data to accurately characterize mean flow properties during transient operation, the transient model calibration process developed in Chapter 4 attempts to characterize the mean flow properties by lumping the unknown transient flow profile effects into the various transient calibration terms (sensor lags, heat soak characteristics, etc.). Any transient flow profile effects which cannot be lumped into the other transient effects, propagate into the component flow and efficiencies determined during real-time calibration of the

engine model. As described in Chapter 6, the sensitivity of the fault detection system is varied to maximize the probability of detecting faults, and to minimize the probability of false alarms. Sensitivity is increased when measurement uncertainties are lower (e.g., stabilized operating conditions at higher power settings). Adjustments to the sensitivity of the fault detection and diagnosis algorithms during transient operation to account for higher uncertainties due, in part, to uncertainties in measurement profiles, sensor lags, heat soak characteristics, and other measurement system considerations are discussed in Chapter 6.

Determination of Time Derivatives from Test Data

The fault detection and diagnostic techniques developed herein require the determination of time derivatives from the transient data for evaluation of non-equilibrium effects such as rotor inertia dynamics and transient lag in the aerodynamic temperature measurements. The effects of sensor lags, rotor dynamics, heat transfer, and volume dynamics are included in the engine model to balance the engine cycle during transient operation. In order to accurately estimate time derivatives from measured test data, the effects of normal measurement noise must be minimized. Therefore, first derivatives are determined from the measured data using a modified sliding least-squares method. The least-squares method (Sokolnikoff et al 1966) applied by Kimzey et al (1977) for short-duration engine testing provides acceptable derivatives for slow and smooth engine accelerations or decelerations (typically 1 minute from idle to maximum power). However, for moderate engine accelerations and decelerations, the least-squares

technique can overestimate or underestimate the instantaneous derivatives by 100% or more. Measured fan speed as a function of time for a snap deceleration is shown in Fig. 3.1. The error in the instantaneous derivative of fan speed is shown graphically in Fig. 3.2, which shows fan speed derivative as a function of time computed using a simple difference technique and the sliding first-order least-square derivative used by Kimzey (referred to herein as a backward least-square derivative). Figure 3.2 also shows a forward-least-squares derivative and *the modified least-square derivative developed in the current research, which is computed as the average of the forward- and backward-least-square derivatives.*

Discussions of test measurements in the following sections will focus on recommended methods for pressure, temperature, and speed measurements and techniques used to minimize the effects of normal measurement noise. These test measurements are the fundamental independent parameters used in the fault detection and diagnostic system and serve as the input, control, and diagnostic measurements.

Pressure Measurements

Steady-state aerodynamic pressure measurements are generally made using a pressure multiplexing system with an in-place calibration system. Recent advances in electronic multiplexing (i.e., one transducer per channel electronic multiplexing systems) can reduce data acquisition time by significantly reducing system stabilization time and thus, reduce test costs compared to slower mechanical multiplexing systems. The

electronically multiplexed aerodynamic pressure uses differential pressure transducers in conjunction with floating, atmospheric, and vacuum reference pressures to extend measurement ranges. In-place pressure calibrations further improve accuracy compared to previous systems used. Typical measurement uncertainties, which can be obtained in altitude test facilities range from 0.04 to 0.16 percent of full scale for electronically multiplexed aerodynamic pressure measurements. Typical transient turbine engine measurement uncertainty for resistance shunt calibrated pressure measurements is +/- 1.5 percent of the reading (applicable over 20 percent to 100 percent of the sensor full-scale range) at 3 Hz. In general, the natural frequency of the pressure probe and transmission tube is assumed to have an acceptable response capability for the pressure amplitude and frequency of interest.

Temperature Measurements

Thermocouple devices are normally used for measurement of temperature in altitude test facilities. Thermocouple probe designs used include single-shielded aspirated, unshielded, semishielded, and bare-wire (crossflow) thermocouples. Thermocouple measurements commonly reference both legs to a copper wire in the temperature reference unit housed in a temperature-controlled environmental enclosure. The thermocouple voltages are converted to temperature using a millivolt substitution calibration. Although thermocouple wire is manufactured to tight tolerances and specifications, the accuracy of the temperature measurement can generally be enhanced by calibrating the wire (testing it at known temperatures). For high-velocity flows, probe

recovery factors can be used to account for the internal-heat-transfer and aerodynamic heating effects, (e.g., Glawe et al (1956) and Willbanks (1971)). In severe thermal environments, radiation and conductive corrections are applied to account for external heat-transfer effects. Moffat (1957) and Glawe et al (1978) provide the basic theory behind evaluation of steady-state and transient velocity, conduction, and radiation errors. Additional theory pertaining to thermocouple time response is discussed in Appendix E.

Steady-state probe calibration is relatively straightforward. Transient probe calibration is typically performed by imposing a step change where the probe is either inserted or withdrawn from a calibrated flow (Stickney, 1955) or by diverting flow from the probe (Glawe et al, 1956). Reference literature on time constant determination is limited and additional publications and/or research in this area would increase the state-of-the-art for turbine engine applications. In the absence of probe design data or transient probe calibration data, time constants (which typically vary as a function of mass velocity and total temperature) can be estimated from the test data using a number of techniques. Two simple approaches that can be used to estimate the time constant for turbine exit temperature (T_5) during a 5-10 second engine acceleration from idle to 95% of full power at sea-level static operating conditions follow. The first approximation of the time constant can be evaluated by noting the slope of the recorded data when the compressor mass flow and speed have stabilized and the time when that slope intersects the peak temperature level. Alternately, the time constant can be evaluated by measuring the time between when the compressor mass flow and speed have stabilized and the time where

the recorded temperature reaches 63 percent of the difference in the temperature between the starting point and the peak temperature measurement.

A more consistent approach to determination of time constants for the fault identification process is to iteratively employ the aerothermodynamic engine model and the diagnostic capability developed herein to estimate fan exit, compressor inlet, compressor exit, and turbine exit time constants and transient profile effects. Typical transient turbine engine temperature measurement uncertainty is +/- 0.75% of reading +/- 4°F at 0.1 Hz (applicable over 10 percent to 100 percent of the sensor full-scale range).

Speed Measurements

Rotating speeds are normally sensed by magnetic transducers that produce an output signal frequency proportional to the number of distinct transducer magnetic field interruptions (metal discontinuities) on the rotating member. Determination of transient rotor speed generally involves a frequency-to-voltage conversion and conversion to speed using a frequency substitution calibration. Extremely accurate speed measurements are obtained for steady-state operation using a pulse-to-digital conversion system based on a period measurement technique in which no time is lost in measuring successive periods. Repeated samples are discarded, and individual periods are summed to obtain an average speed over the duration of the data point. The method used to convert the pulse signal to an average speed over the duration of the data point differentiates this speed

measurement technique from typical pulse-to-digital conversion systems. Typical steady-state measurement uncertainties which can be obtained in altitude test facilities range from 0.02 to 0.3 percent of the reading for speeds. Typical transient turbine engine speed measurement uncertainty is +/- 0.5% percent of reading (applicable over 10 percent to 100 percent of the full-scale range) at 3 Hz. For acceleration time tests and surge margin assessment using a fuel spike, the transient error is typically on the order of 50 RPM. For the limited speed excursion after a fuel spike (50 RPM excursion at a rate of 1000 RPM/second) the transient error is on the order of the speed excursion (+/- 50 RPM) and it is nearly impossible to measure the instantaneous speed.

Variable Geometry Measurements

Measurement of engine power lever angle, exhaust nozzle area, variable inlet guide vanes, compressor inlet variable vanes, variable stator vanes, and bleed valves are the most common variable geometry measurements obtained during tests conducted in altitude test facilities. Typical transient turbine engine geometry system measurement uncertainty is +/- 2% of reading (applicable over 10 percent to 100 percent of the sensor full-scale range) over a frequency response dependent on the mechanical system.

Nozzle position or throat size is typically measured using a position indication. With appropriate calibrations, nozzle throat size (area) can be measured with an accuracy on the order of +/- 0.3 percent of area for non-augmented operation and on the order of +/- 0.5 percent for larger nozzle areas corresponding to augmented operation. During

actual engine operation, errors in nozzle area due to deformation of the nozzle flaps are generally insignificant.

Structural Clearance Measurements

Changes in clearances of neighboring components are most apparent during transient engine operation due to varying mechanical loads, centrifugal forces, and thermal effects. In addition, permanent changes in clearances can result from interference or rubbing of neighboring components. Increases in structural clearances often decrease component efficiency and, in the case of the compressor, decreases surge margin. As developmental engine testing proceeds, theoretical models used to calculate structural clearances evolve and eliminate the need for direct measurement of the clearances. As a result, structural clearances are usually not measured during typical developmental engine tests.

Measurement Uncertainty Influence Coefficients

Many turbine engine measurements like fuel flow, airflow, and force measurements are not directly measured, but instead are calculated from measured data inputs. Discussions of the calculated data inputs in the following sections will focus on the determination of fuel flow, airflow, and force measurements made in altitude test facilities. Elemental measurement uncertainties are propagated to calculated measurement uncertainties using influence coefficients described in Appendix F.

Fuel Flow Measurements

The most common method of measuring steady-state and transient engine fuel flow rates is with calibrated turbine-type flowmeters. Flowmeter frequency is measured using the highly accurate pulse-to-digital conversion system previously discussed. Fuel flow rate is calculated using the meter pulse output, fuel temperature, and fuel and meter characteristics. Typical variables used to calculate fuel flow include fuel viscosity, specific gravity, flowmeter flow constants, and temperature compensation relationships. Fuel viscosity and specific gravity are evaluated using fuel sample analysis, and flowmeter flow constants and temperature relationships are evaluated from fuel meter calibration data. Fuel viscosity corrections as a function of test fuel temperature are made using ASTM Standard Viscosity Temperature Linearization Techniques (ASTM D341-89).

A variety of fuel flow determination methods currently exist (in industry) which result in differences in measured fuel properties and calculated fuel flow beyond the expected experimental uncertainty. SAE ARP 4990, currently in draft form, will provide a common basis for consistent and accurate calculation of fuel flow based on ASTM and API standard test methods and temperature and pressure correction methodologies.

When considering transient flow measurements, errors in measured fuel flow can be introduced by the effects of inertial forces, and by bearing friction and fluid slippage.

Use of close-coupled low inertia turbine flowmeters can overcome some of these effects. In general, the correction of fuel flow for transient effects is applied as a first order lag to simulate the moment of inertia effect. The moment of inertia effect can be quantified by an analysis of the equation of motion for a rotor based on a calculation of the flow-induced force per unit blade length (perpendicular to the effective velocity) of the rotor. The moment of inertia effect can be characterized in terms of known or measurable values for the flowmeter rotor (Fig. 3.3). For the blade element shown, **r** is the radius of the blade element measured from the rotor axis, **c** is the blade element chord, α is the blade angle with the pipe axis (flow direction), $\mathbf{u_2}$ is the final flow velocity after an instantaneous increase in the flow rate, **L'** is the lift force per unit blade length, α_o is the effective angle of attack, $r\frac{d\theta}{dt}$ is the blade element velocity, and **v** is the effective velocity. After trigonometric substitutions and integration the equation for the time constant can be derived. The time constant K, (Grey, 1956) may thus be obtained by

$$\int_0^K dt = \frac{1}{A\cos\alpha} \int_{\frac{u_1}{u_2}\tan\alpha}^{[1-\frac{1}{e}(1-\frac{u_1}{u_2})]\tan\alpha} \frac{dz}{\tan\alpha - z}$$

Integrating and introducing the limits

$$K = \frac{1}{A\cos\alpha}$$

Corrections for the transient lag in turbine-type flowmeters can be accomplished using the equation

$$WF = WF_M + \tau \frac{dWF_M}{dt}$$

where WF is the fuel mass flow rate (corrected for transient lag), WF_M is the measured fuel mass flow rate, and τ is the time constant K.

Bearing friction effects (due to increased loading from inertial forces during transients) and fluid slippage effects are usually small compared to the moment of inertia effect. Errors in measured fuel flow can also be introduced by the sampling process, as suggested by Evans et al (1994) and by the low pass dynamics of the fuel metering valve investigated by Shoukens et al (1994).

For transient speed measurements, pulses are counted over a period corresponding to some multiple of the total number of vanes on the turbine flowmeter. This approach reduces inaccuracies due to non-uniform spacing between the turbine blades but adds an additional delay in the measurement. Improvements to the pulse-to-digital conversion system follow directly from investigations performed by the author to determine the average fuel flow rate during sinusoidal flow variations resulting from pressure variations in the fuel supply system. Asynchronous sampling of the data system (typical of pulse-to-digital conversion systems) during pulsating flow typically overestimates the average fuel flow rate. Additional errors in the fuel flow rate result from averaging a non-integral

number of cycles of a pulsating flow. To minimize the uncertainty of the fuel flow measurement, multiple ranges of flowmeters can be used. Likewise, multiple data channels and processing techniques can be used to minimize the uncertainty of the steady-state and transient flow measurements.

Influence coefficients are used to propagate elemental measurement uncertainties to a calculated measurement uncertainty value. Typical turbine flowmeter measurement uncertainties obtainable in altitude test facilities range from 0.2 to 1.0 percent of reading (applicable over 10 percent to 100 percent of the sensor full-scale range) for fuel flow. Typical transient turbine engine fuel flow measurement uncertainty is +/- 2 percent of reading (applicable over 10 percent to 100 percent of the sensor full-scale range) at 3 Hz using turbine type flowmeters. A typical uncertainty for a system specifically designed to measure fuel pulses would be on the order of +/- 10 percent. For typical installations in altitude test facilities, turbine flowmeter measurement uncertainty (if unaccounted for) can be as high as +/- 100 percent for transient engine operation. Transient corrections can be applied to adequately characterize the initial fuel pulse after a snap acceleration or deceleration. Figure 3.4 illustrates the variation in fuel mass flow rate determined by the engine fuel flow metering valve feedback system and the uncorrected test facility fuel flow measurement during an engine acceleration. For this particular example, the facility flow measurement was evaluated as the difference between two turbine-type flowmeter measurements, one being the fuel flow supplied to the engine and the other being the recirculated fuel flow. Significant errors in the fuel flow measurement (on the order of

+/- 10 percent) can be expected for a few seconds after the initial transient due to subsequent fuel oscillations. Errors on the order of +/- 50% can be expected in the time compensated turbine meter measurement during an engine surge. Provisions have been included in the fault identification algorithms to accommodate increased uncertainties in the fuel measurement using turbine-type flowmeters during fuel oscillations or reconfiguration of flowmeter ranges.

Transient fuel flow measurement can also be accomplished through the use of a fuel metering valve feedback system or by installation of a nozzle or orifice and a differential pressure measurement. The most commonly used fuel metering valve feedback system uses a linear or rotary variable differential transformer (LVDT or RVDT) because it integrates easily into a modern engine Full Authority Digital Electronic Control (FADEC) system. In a typical arrangement, a gear pump delivers the fuel flow rate at a constant pressure maintained by a fuel relief valve. Fuel flow rate is determined by the fuel flow command. A pressure regulator maintains pressure drop across the fuel-metering valve.

Fuel metering valve feedbacks have extremely fast response, depending on the signal conditioning used, and can be used to evaluate the response of turbine type flowmeters during transient operation. Uncertainty of fuel flow feedback systems, excluding temperature effects, are on the order of 2 percent of reading (applicable over 10 percent to 100 percent of the sensor full-scale range). The fuel metering valve

feedback system mass flow rate is calculated using Bernoulli's equation and the principle of conservation of mass. The pressure regulation scheme maintains a nearly constant pressure drop across the flow-metering valve and fuel flow can be correlated directly with stroke through bench calibrations. Additional steady-state calibrations are typically performed in actual test applications using turbine flowmeters.

Nozzle and orifice flow measurement systems typically have a higher rate of response than a turbine flowmeter. Additional research is required to include measurements of transient fuel flow using differential pressure measurement systems, and has not been included in the fault identification process described herein. In theory, Bernoulli's equation and the principle of conservation of mass through the orifice can be used with the differential pressure measurement, fuel temperature compensation, and appropriate calibrations to determine the fuel flow rate. The only disadvantage to nozzle and orifice flow measurement systems is relatively high uncertainty in the flow rate at low flows.

Engine Inlet Airflow Measurements

A common method of measuring engine inlet airflow rates in an altitude test facility is with a choked-throat "critical flow" venturi or with a calibrated engine/inlet bellmouth. Critical flow venturis are used when more accurate flow measurements are required during steady-state operating conditions. Multi-venturi airflow-measuring systems are used to maintain critical flow operation over a wide range of airflow rates.

Throat area is corrected for thermal effects and the flow coefficient is calculated using theoretical techniques (Smith, R. E., et al 1961). Throat area and flow coefficients are combined with pressure and temperature measurements to calculate airflow. Equations to calculate air flow using venturis and nozzles are provided in Appendix G (AGARD-AR-320).

Limitations currently exist with the theoretical method for determining the flow coefficient of venturis operating at critical flow conditions. The most important limitations are that the venturi contour must be properly designed and that the venturi must operate in the turbulent boundary layer flow regime.

The engine/inlet bellmouth flow measurement system is typically calibrated using critical flow venturis. Engine/inlet bellmouth mass flow rate W_1 is obtained from the time-dependent engine-inlet flow total and static pressures, and the flow total temperatures which are obtained from pitot, static pressure, and thermocouple probes, respectively.

Influence coefficients are used to propagate P_T, P_S, and T_T elemental measurement uncertainties into a calculated measurement uncertainty value for W_1. As would be expected, the measurement uncertainty is strongly influenced by the engine operating conditions (altitude, Mach number, and power lever setting). Typical measurement uncertainties obtainable in altitude test facilities range from 0.4 to 1.0

percent of reading (applicable over 10 percent to 100 percent of the full-scale range) for engine/inlet airflow measurements.

Engine inlet airflow rate can change very rapidly and the instrumentation used to determine the engine inlet airflow rate must respond accordingly. Transient airflow measurements using engine/inlet bellmouth pressure and temperature measurements requires pressure measurements that are an order of magnitude more responsive than the desired airflow measurement (Wehofer and Rivir, 1980). Simply stated, a 10 Hz response in airflow measurement requires a 100 Hz response in pressure measurement. With a properly designed measurement system, transient airflow can be determined using quasi-steady assumptions. This is justified, according to Wehofer and Rivir, for pressure measurements because the time-dependent pressure variations are relatively small compared to the acoustic speed at which the pressure waves travel. Air temperatures require longer periods to reach equilibrium but, at the engine/inlet interface, temperature variations rarely exceed 15°F/second. Measured wall or skin temperatures used for determination of thermal growth of the bellmouth typically respond an order of magnitude slower than the air temperatures and do not require time correction.

Critical flow venturis are rarely used for determination of transient airflow. When they are used, volumetric dynamic flow corrections (Wehofer and Rivir, 1980) are required using static pressure and total temperature measurements at the engine face due

to the large volume between the venturis and the engine face required for exit flow conditioning.

Measurement of transient engine airflow can be realized with a properly designed system. A complete assessment of transient airflow measurement uncertainty is beyond the scope of this work. Uncertainty analysis completed at AEDC by Wehofer and Rivir (1980) shows that uncertainty levels of 1 percent of reading are obtainable at the engine face for bellmouth flow Mach numbers greater than 0.25 to 0.3 but on the order of 4 to 8 percent for lower Mach numbers. Provisions in the fault identification algorithms to accommodate increased uncertainties in the airflow measurement for transient engine operation and for bellmouth flow Mach numbers less than 0.25 are discussed in Chapter 6.

Critical flow venturis and engine/inlet bellmouth airflow measurement systems have relatively low measurement uncertainties but they cannot be used for internal engine airflow measurements. A specially designed fore-aft probe, typically referred to as a Mach probe or airflow rake can be used for measurement of internal engine Mach number and airflow measurement. The fore-aft probe uses measured gas path total pressure, static pressure, and temperature to determine local airflow velocity. For Mach numbers less than 0.6, the equation for determination of the Mach number can be reduced by Mclaurin series expansion and elimination of higher order terms to show a linear relationship with $\sqrt{\Delta P / P}$. The aft pressure measurement is not a true static pressure but

a wake pressure and requires wind tunnel calibrations (Dvorak et al 1986) to correct for pressure and temperature recovery by adjustment to the calculated Mach number. Local airflow velocity can then be calculated from the true Mach number using well-known relationships. Calibration against a reference measurement system is required to obtain average Mach number and airflow readings.

Uncertainties of fore-aft probe measurements are larger than those of the engine/inlet bellmouth. In many cases the uncertainty of fore-aft probe measurements are so high that the results can be viewed only on a qualitative basis. Therefore, for the purposes of this investigation, fore-aft probes and other advanced airflow measurement sensors are not specifically used in the fault identification process.

Bleed Extraction Measurements

One of the most widely used means of measuring engine bleed mass flow rates in an altitude test facility is with thin-plate square-edged orifices by methods described in ASME Standard ASME-MFC-3M (1985). Expressions for orifice discharge coefficients are obtained from ASME MFC-3M for pipe diameters exceeding 2 inches and from Marxman et al (1961) for pipe diameters between ½ and 1½ inches with flange taps. Expressions for orifice expansion factors are obtained from ASME MFC-3M for subcritical flow and from Cunningham (1950) for supercritical flow. Pipe diameter, orifice diameter, orifice thermal expansion coefficient, orifice discharge coefficient, orifice expansion factor, and orifice pressure tap configuration are combined with the

total temperature of the fluid stream, pressure measurements upstream and downstream of the orifice taps, and water-to dry-air and fuel-to-air mass ratios to calculate orifice flow rate. Estimated uncertainties for orifice discharge coefficients and expansion factors can be found in ASME MFC-3M.

Problems associated with the effects of pulsations on the accuracy of non-linear differential flowmeters such as the orifice plate can be significant. Quasi-steady theory for the measurement of pulsating flow states that it should be possible to determine the average flow rate by measuring the time-mean square root of the differential pressure and by evaluating the other quantities in the flow equation as if the flow were steady. If the time-mean of the differential pressure is measured first and the square root of this value is taken, the flow rate will be overestimated. The resulting error, referred to as the square root error, is closely related to the overestimation of fuel flow measurements with asynchronous pulse-to-digital conversion system sampling. If, on the other hand, normal measurement noise is not removed from the system, quasi-steady theory predicts that the time-mean square root of the differential pressure will be underestimated along with the average flow rate.

Influence coefficients are used to propagate elemental measurement uncertainties to a calculated measurement uncertainty value. As would be expected, the measurement uncertainty is strongly influenced by the engine operating conditions (altitude, Mach number, and power lever setting). A complete assessment of bleed flow uncertainty is

beyond the scope of this work. For the purposes of this investigation, uncertainty of bleed flow rate is assumed to be +/- 2 percent of reading (applicable over 10 percent to 100 percent of the measurement range) at 3 Hz.

Auxiliary Power Extraction Measurements

One of the most widely used means of measuring the horsepower extracted from the engine in altitude test facilities is with a water brake system. For the particular application under investigation, measured water brake load cell torque is combined with engine (compressor) rotational speed and gear ratio to determine the horsepower extracted from the engine. For the purposes of this investigation, auxiliary power extraction uncertainties are assumed to be on the order of +/- 1 percent of reading (applicable over 10 percent to 100 percent of the measurement range).

Thrust Measurements

Force measurements in an altitude test facility commonly use a thrust stand in a free-body configuration so the net forces acting on the free body provide a measurement of the engine gross thrust. The engine thrust stand measures only a portion of the thrust developed by the engine. High-accuracy measurements of the force on the engine thrust stand are generally obtained using an in-place calibration system. The force system is in-place calibrated as follows: Hydraulic/pneumatic-actuated pistons or electromechanical devices in series with standard load cells are used to apply force directly to the thrust

stand. Facility load cells are read out to determine the magnitude of the applied forces. Force levels are selected and sufficient calibrations are performed to define the repeatability, hysteresis, tare, and linearity of the force system (Hollis, 1997). Force adjustments are typically made to account for ambient pressure, interaction, off-axis, tare, thermal, cell cooling air, and other secondary effects such as load cell internal hysterisis due to strain gage design and bonding. Remaining steady-state forces acting on the engine are determined as provided by Smith, R.E., Jr. et al (1982).

A primary consideration of time varying scale force (F_S) measurements is the dynamic response of the thrust stand. Thrust stands are typically designed for measurement of relatively slow force transients and have a natural frequency on the order of 10 Hz and a damping coefficient of about 3 percent. Data for thrust stand natural frequency and damping coefficient are typically available. In the absence of actual data, thrust stand response characteristics can be evaluated from design specifications considering a simple spring-mass damper system.

For time-dependent engine operation, engine gross thrust F_G (Fig. 3.5) can be expressed as (neglecting secondary terms):

$$F_G = F_S + XMVIS + ADPIS + \partial / \partial t \int \rho V d(Vol)$$

where F_s is the thrust stand measured scale force, and XMVIS and ADPIS are the momentum term and the area-pressure difference term evaluated at the isolation seal for direct connect test configurations. The last term describes the time-rate-of change of momentum stored in the system defined by the control volume shown in Fig. 3.5. Figure 3.6 gives an indication of the relative magnitude of the scale force, momentum term, and area-pressure terms for high and low bypass engines over a range of operating conditions (AGARD-AR-320). For consistency with the reference document, the measured scale force is denoted by F_S, the momentum term is denoted by W-V, the area-pressure difference term is denoted by P-A, and N_L is the percent corrected low rotor speed.

The momentum term used in the gross thrust equation is calculated from the following relationship:

$$XMV_{IS} = \frac{W_{IS} * V_{IS} - \frac{2}{144}\left[\pi * R_{IS} * RHO_{IS} * V_{IS}^{\ 2} * THET_{IS}\right]}{CG}$$

where W_{IS} is the mass flow rate of the air at the isolation seal plane, R_{IS} is the duct radius, V_{IS} is the free-stream velocity of air at the isolation seal, $THET_{IS}$ is the momentum thickness of the boundary layer at the isolation seal, and CG is the gravitational constant. Tabulated data from an approximate calculation of turbulent boundary-layer development in compressible-flow (Tucker, 1951) are used to determine the displacement thickness

and momentum thickness as a function of the power law exponent (VPP) and the free-stream Mach number (XM).

The area-pressure difference term at the isolation seal used in the gross thrust equation is calculated from the following relationship:

$$ADP_{IS} = CT_{IS} * A_{ISOD} * (PS_{IS} - PAMB)$$

where CT_{IS} is the area thermal expansion coefficient of the isolation seal, A_{ISOD} is the area of the male isolation seal measured to the outside diameter, PS_{IS} is the Static pressure at the isolation seal, and PAMB is the simulated altitude ambient pressure (cell pressure).

For the particular application under investigation, F_S, PS_{IS}, and $PAMB$ were determined from transient time-dependent measurements. The mass flow rate W_{IS} was obtained from the time-dependent engine-inlet total and static pressures and the total temperature. For transient operation, the momentum thickness term is typically neglected. The last term in the net thrust equation represents the time-rate-of-change of momentum (per unit volume) stored in the engine control volume, and for small to moderate engine transients has negligible influence, and is ignored. For dynamic engine operation (not specifically addressed in this research), volume dynamics are important and the fault detection sensitivity is reduced accordingly.

For thrust measurement uncertainty, as before, influence coefficients are used to propagate elemental measurement uncertainties to a scale force or calculated thrust measurement uncertainty value. As would be expected, the measurement uncertainty is strongly influenced by the engine cycle, engine thrust level, and the engine operating conditions (altitude, Mach number, and power lever setting). Typical measurement uncertainties obtainable in altitude test facilities range from 0.2 to 1.0 percent of full scale for force measurements. A complete assessment of transient force measurement uncertainty is beyond the scope of this work. For the purposes of this investigation, uncertainty of force measurements is assumed to be on the order of +/- 3 percent of maximum thrust at 3 Hz. Variations in the scale force measurement on the order of +/- 3 percent of maximum thrust are shown in Fig. 3.7 during a relatively slow acceleration to intermediate rated power. This simplistic error analysis assumes a five-fold increase in pressure and thrust system installation effects over steady-state uncertainties and a thrust stand dynamic response error of approximately 2 percent of the scale force measurement. Provisions included in the fault identification algorithms to accommodate increased uncertainties in the force measurement for small (but non-negligible) underdamped oscillations in the thrust signal during normal engine operation and for rapid thrust changes during dynamic engine operation associated with augmentor lightoff/blowout or engine surge are discussed in Chapter 6. More detailed analysis, like the one completed by Runyan (1992) to evaluate the structural dynamics of a small rocket thrust stand, is required to minimize error terms resulting from dynamic measurement with a static

calibration, stand fidelity at the response tested, integrating techniques, damping errors, frequency errors, start/stop timing, pulse duration, resonance beating, thermal expansion, momentum, and side loads.

Data Filtering Technique to Determine Time-Dependent Values from Test Data

The fault detection and diagnostic techniques developed herein require the determination of time-dependent values from the transient data. The effects of normal measurement noise must be minimized for satisfactory determination of the calculated inputs due to the non-linear nature of the calculation methods. The effects of normal measurement noise are minimized using a digital filtering technique to ensure common frequency response for all the measurements and compatibility with the fault identification techniques.

As previously stated, the fault identification techniques developed in the current research assume an online sampling rate of 100 samples per second. Filters are almost always incorporated into altitude test facility data acquisition systems to reject or attenuate frequency components of the signal which are outside the range of interest and to guard against aliasing. Many types of filters are available, each with different operating characteristics (AGARD-AR-320). The main performance features of these filters can be described by:

- The flatness, ripple or droop in the pass band, its cut-off frequency (usually 3 dB), and the attenuation decay rate beyond cutoff.
- The phase shift (between input and output) as a function of frequency.
- The rise time (or time constant), overshoot and settling time in response to a sudden change of input, e.g., step or ramp.

The settling time in response to a sudden change of input is an important consideration for the current research which reduces, by an order of magnitude, the filter response time (or settling time) of Kalman filter based techniques used to estimate engine performance variations. The discussion that follows refers to three distinct filtering techniques. The first filtering technique discussed refers to analog low pass filters typically incorporated in the signal conditioning components or in the data acquisition system; examples include Butterworth, Bessel, Chebyshev, and Cauer-Elliptic filters. The second filtering technique discussed refers to the Kalman filter, which includes a dynamic model to estimate engine performance variations. The final filtering technique discussed refers to the digital filtering technique used in the fault identification process to minimize the effects of normal measurement noise from measurements used in the fault identification process. As will be seen in the following discussion, all filters, whether analog or digital, alter the signal passing through them.

The attenuation of typical 8-pole low pass filters is shown in Fig. 3.8. On one extreme, the Bessel function has a drooping pass band, high fidelity, and short settling time. On the other extreme, the elliptic filter provides a sharp cutoff at the expense of poor fidelity and possible ringing. Additional details of common analog filters are provided in AGARD-AR-320. The theoretical response of 2-pole Butterworth filters (Frequency Responses, Inc.), which are commonly used in altitude test facilities, is shown in Figs. 3.9 and 3.10. Compared to the 8-pole filter, the cut-off for the 2-pole is not as sharp but the fidelity is much higher. The actual response of commonly used 2-pole Butterworth filters is characterized by –3 dB attenuation at the cutoff frequency (selectable bandwidth steps of 10, 20, 50, 100, and 200 Hz) and a 12 dB/octave rolloff. Pressure measurements in altitude test facilities are typically filtered at 50 or 100Hz, fuel flow and speed measurements at 100 Hz, and force measurements at 50 Hz. Temperature measurements at the engine/inlet interface are typically filtered at 10 Hz. Temperature variations within the engine are much more rapid than at the engine inlet and, as a result, 50 Hz filters are typically used. Often times, 2-pole RC filters with a –6 dB attenuation at the cutoff frequency of 10 Hz and a 12 dB/octave rolloff are used in place of the 10 Hz Butterworth filters.

Figure 3.11 shows the response of 50 Hz 4-pole filters to a triangular step input. Figure 3.12 shows the response of 100 Hz 4-pole filters to a step input signal. Here, the settling time is on the order of 7 milliseconds for the Bessel filter, 20 milliseconds for the Butterworth filter, and 40 milliseconds for the Chebyshev (1 dB) filter. Figure 3.13

shows the response time of a Kalman filter based algorithm for estimating in-flight engine performance variations (Luppold et al, 1989) due to a rapid change in component performance. The response time of the Kalman filter based algorithm is on the order of 10 seconds. As shown in Fig. 3.13, the Kalman filter introduces significant errors and/or distortions (attenuation and phase shift) in the estimated gross thrust. As noted by Luppold et al (1989), a faster response time is desirable for event detection applications so that correct actions can be taken to avoid surges and stalls, which are prevalent in these situations.

In order to provide the reader with a better understanding of filtering concepts, the Kalman filter is compared with ordinary least squares on a problem where they might be expected to give the same answer – but they don't. The following example, which follows closely with the example provided by Strang (1986), considers a steady model where all observations $b_1, b_2, \ldots b_m$ measure the same thing. In addition, the example assumes that all errors are independent with variance $\sigma^2 = 1$. Solutions to ordinary least squares and Kalman filter problems are shown below.

1) The least squares solution of the m equations $x = b_i$ is

$$\hat{x} = \frac{b_1 + b_2 + \ldots b_m}{m} \text{ with variance } P = \frac{1}{m}$$

2) The Kalman filter solution comes from the m equations $x_i = b_i$ together with the steady model $x_{i+1} = x_i$.

In order to estimate the unknown vector x_i, at each time step, two systems of equations must be solved. The first system of equations relates the measurements to the true values $b_i = A_i x_i$ (with error ε_i) and the second systems of equations governs the change in state as time goes forward using a known law of the form $-F_i x_i + x_{i+1} = 0$ (also with error ε_i). Assuming the first three measurements are known and we are ready to estimate x_2, the equations can be combined into a single system

$$\begin{bmatrix} A_0 & 0 & 0 \\ -F_0 & I & 0 \\ 0 & A_1 & 0 \\ 0 & -F_1 & I \\ 0 & 0 & A_2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_0 \\ 0 \\ b_1 \\ 0 \\ b_2 \end{bmatrix}$$

The differences between the least squares and Kalman filter solutions can be seen after the second measurement

$$\begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} b_0 \\ 0 \\ b_1 \end{bmatrix}$$

and again after the third measurement

$$\begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_0 \\ 0 \\ b_1 \\ 0 \\ b_2 \end{bmatrix}.$$

If we force x_{i+1} to hold exactly, we are back to ordinary least squares. But if we allow drift errors ε_i, also of variance one, then the equations above are solved by least squares. Forming the equations $A^T Ax = A^T b$ in the two cases gives (where $x_{0/1}$ is the predicted value and $x_{1/1}$ is the new value for the case of two measurements)

$$\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x_{0/1} \\ x_{1/1} \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \end{bmatrix}$$

and

$$\begin{bmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_{0/2} \\ x_{1/2} \\ x_{2/2} \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix}.$$

We can see what happens without the complicated formulas. The inverses of the above equations are

$$\left(A^T A\right)^{-1} = \frac{1}{3}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

and

$$\left(A^T A\right)^{-1} = \frac{1}{8}\begin{bmatrix} 5 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 5 \end{bmatrix}$$

After the measurements b_0 and b_1, the best estimates come from the first equation. Multiplying by $\left(A^T A\right)^{-1}$ gives not the ordinary average, but

$$x_{0/1} = \frac{2b_0 + b_1}{3}, \quad x_{1/1} = \frac{b_0 + 2b_1}{3}$$

The new data b_1 has a heavier weight in the filtered estimate of x_1. And b_1 also appears in the smoothed estimate $x_{0/1}$. There it is weighted less heavily, but still we know more about x_0 after measuring b_1.

With b_2, the filtered and smoothed estimates change to

$$x_{0/2} = \frac{5b_0 + 2b_1 + b_2}{8}, \; x_{1/2} = \frac{2b_0 + 4b_1 + 2b_2}{8}, \; x_{2/2} = \frac{b_0 + 2b_1 + 5b_2}{8}$$

The last is the most important. It is the best estimate of x_2, using b_o and b_1 but emphasizing b_2. The possibility of drift produces, as we continue, an exponential decay of weight to the old measurements.

The whole object of filtering is to find the solution to a system of equations which relates the measurements to the true values and the change in state as time progresses using a known law or more appropriately, an imperfect model. According to Gauss, the problem should be solved using least squares and according to Kalman, we should solve it recursively. In general, it is required that the model be a good representation of the observations in any local segment of time close to the present. For the fault identification process, we will need to compare the present observations to observations which are not very old, in order to detect rapid faults, and also compare the present observations to observations that are old to detect slow faults. The ability of the filtering or smoothing technique to follow, or track changes, in the underlying time dependent model depends

mainly on the smoothing constant. Depending on the smoothing constant, more weight is given to the historical data or current observations. The smaller the value of the smoothing constant, the slower the system response to measured disturbances. For larger values of smoothing constants, the system reacts quickly to real changes in system performance and also to random fluctuations, or noise. A major drawback of exponential smoothing is selecting an optimum value of the smoothing constant. As a result, techniques, such as those described by Trigg and Leach (1967) have been developed for adaptive control of smoothing constants.

As previously stated, analog filters are almost always incorporated into altitude test facility data acquisition systems to reject or attenuate frequency components of the signal which are outside the range of interest and to guard against aliasing. *The digital filtering technique used in the fault identification algorithms for the research presented herein is a simple unweighted average of observations immediately prior to and after the current observation to minimize the effects of normal measurement noise. The unweighted average provides for a settling time on the order of 100 milliseconds for fault detection and yields an unbiased estimate of the true process during steady-state operation.* Clock time for diagnosis of measurement related faults is a function of the speed and number of processors used in the fault identification process. Settling time of the fault identification algorithms for diagnosis of severe engine performance variations (characterized by component efficiency changes of 3% or more) is on the order of 100 milliseconds. Settling time for diagnosis of moderate variations in engine performance

(characterized by component efficiency changes of 1.5% or less) is typically less than 2 seconds.

The response of the digital filtering technique used in the fault identification process to minimize the effects of normal measurement noise is shown graphically in Fig. 3.14 for a rapid engine deceleration and in Fig. 3.15 for steady-state engine operation. The response of the digital filtering technique is also shown in contrast to the first-order auto-regressive digital filter, or exponential filter, employed by Kimzey (1977). Compared to the first-order auto-regressive digital filter, the digital filtering technique employed, for use with the fault identification algorithms, provides superior fidelity and reduced error.

The fault identification algorithms extract reliable values from data with normal measurement noise and permit identification of a wide variety of faults during both steady-state and transient engine operation. Compared to filtering algorithms, smoothing algorithms (where the sampling interval is estimated from data obtained at times both before and after the time being considered) such as have been developed for use in the present work provide a substantial improvement in real-time estimation accuracy and fault detection capability. Exponential-type filters were found to be insensitive to many common sensor failure modes and abrupt changes in system performance which must be identified during developmental engine testing.

As would be expected, the fault identification sensitivity is increased during steady-state engine operation where measurement uncertainties are typically much lower than during transient operation, and uncertainties in the modeling process to account for rotor dynamics, heat transfer, and volume dynamics, are negligible.

Engine Inlet/Isolation Seal

The engine inlet and isolation seal station designations for a twin-spool turbofan engine are shown schematically in Figs. 3.16 and 3.17. The state of the air entering the engine is based on thermodynamic gas properties which account for the effects of water vapor, if the water-to-air dry mass ratio is non-zero. Other arbitrary or test-specific gas mixtures can also be accounted for.

Engine inlet duct inlet plenum total pressure, P_{PL} (from measured values with appropriate correction factors), and inlet/engine interface total pressure, P_1, are compared to the average fan inlet total pressure, P_{2A} and engine control fan inlet total pressure, QP_{2X} in the fault detection and diagnosis process. Similarly, inlet/engine interface total temperature, T_1, is compared to the average fan inlet total temperature, T_{2A} in the fault detection and diagnosis process.

In the absence of engine inlet distortion, fan inlet total pressure measurements, P_{12} and P_2, can be determined from the inlet/engine interface total pressure, P_1, with a correction for the pressure loss (based on experimental data) between the stations. In the

absence of fan inlet total temperature measurements, T_{12} and T_2, fan inlet total temperature is typically assumed to equal the inlet/engine interface total temperature, T_1. This assumption is generally valid due to the relatively high flow rate, low driving potential, and low overall heat transfer coefficient between the stations.

Classical use of the engine model is to determine component and overall engine performance based on fixed components. For the present study, the engine model is used to determine component and overall engine performance based on measured test data. A summary of typical measured, known, assumed, and desired parameters for fault identification during developmental turbine engine testing is presented in the following sections for each engine component.

Fan Steady-State Analysis Summary

Fan performance is characterized by the aerodynamic and thermodynamic equations relating pressure, temperature, and mass flow as described in Appendix H. The station designation for a twin-spool turbofan engine is shown schematically in Fig. 3.18.1. Measured, known, and assumed parameters will vary from engine to engine and installation to installation. A summary of typical measured, known, assumed, and desired parameters for steady-state fan analysis is shown in Fig. 3.18.2.

Fan pressure measurements include fan tip inlet total pressure, P_{12}; fan tip discharge total pressure, P_{13}; fan hub inlet total pressure, P_2; average fan inlet total pressure, P_{2A}; fan hub discharge total pressure, P_{21}; and fan inlet static pressure, PS_{2A}. Engine control fan inlet total pressure, QP_{2X}, and engine control fan inlet static pressure, QPS_{13} are used in the fault detection and diagnosis process. Measurement rakes installed throughout the test engine are designed to have a minimal effect on overall engine performance. Measurement rakes do, however, create pressure losses which are archived or "book kept" in order to estimate component and overall engine performance after the rakes are removed (e.g., when the engine is installed in an operational aircraft). If rake pressure losses are negligible, fan hub discharge total pressure, P_{21} can be assumed to equal the measured compressor inlet total pressure, P_{24}. Although rake pressure losses can typically be estimated with a high level of accuracy using theoretical techniques, for the research completed herein, rake total pressure losses at all stations other than the inlet/engine interface were not separated from component performance.

Fan temperature measurements include fan tip inlet total temperature, T_{12}; fan tip discharge total temperature, T_{13}; fan hub inlet total temperature, T_2; fan hub discharge total temperature, T_{21}; and average fan inlet total temperature, T_{2A}. Engine control fan inlet total temperature, QT_{2A} and engine control fan hub discharge total temperature, QT_{21} are used in the fault detection and diagnosis process. Fan hub discharge total temperature, T_{21} is typically assumed equal to the measured compressor inlet total temperature, T_{24}.

Fan flow measurements include total fan inlet mass flow rate (determined from critical flow venturi measurements during steady-state engine operating conditions), W_{2A}, and fan bypass duct bleed flow rate, WB_{13}. Fan inlet mass flow from fan inlet conditions (W_{2AX}) is used in the fault detection and diagnostic process. Fan bypass duct bleed flow valve position, $WB_{13\ \text{Valve Position}}$ is also used in the fault detection and diagnostic process. In altitude test facilities, engine inlet airflow is typically dehumidified and water-to-air mass ratio at the fan inlet, WAR_{2A}, is typically assumed zero.

Additional measurements include fan variable vane angle, FVV, and fan rotor speed, X_{N2}. Engine control fan rotor speed, QXN_2 is used in the fault detection and diagnostic process.

Parameters taken as **"known"** are determined from the steady-state engine model calibration process described in Chapter 4. A fan inlet assumed flow coefficient, CF_{2AX}, is used in the fault detection and diagnostic process to determine fan inlet mass flow from fan inlet conditions (W_{2AX}). Fan inlet physical area, A_{2A}, fan inlet coefficient of linear expansion, $CQUL_{2A}$, and fan inlet adiabatic wall recovery factor, $RFAW_{2A}$, are taken as known. The adiabatic wall temperature, TAW, is determined from adiabatic wall recovery factor and the static and total temperatures

$$TAW = T_S + RFAW(T - T_S).$$

Corrections to fan flow and efficiency as a function of rotor-to-case clearance changes, Reynolds number index, and off-nominal guide vane schedules are either given or must be determined experimentally from test data (Wantland, 1991). When corrections to flow and efficiency are determined experimentally, data are typically acquired for a variety of rotor-to-case clearances, Reynolds number indices, and off-nominal guide vane schedules. Rotor-to-case clearance changes as a result of rapid throttle transients vary with time but tend to reach their greatest value when case expansion leads rotor expansion several seconds after the throttle transient.

The fan tip inlet total pressure boundary layer profile is taken as known. In the absence of engine inlet pressure distortion, the boundary layer profile can be calculated using theoretical techniques (Tucker, 1951) or characterized from previously acquired boundary layer measurements. In the presence of engine inlet distortion, the effects of boundary layer profiles are assumed negligible.

Fan Transient Analysis Considerations

Fan transient analysis parameters are summarized in Fig. 3.18.3. Parameters taken as **"known"** are determined from the transient model calibration process described in Chapter 4.

Total fan inlet mass flow rate W_{2A} is determined from the calibrated engine/inlet bellmouth calculations for time-dependent engine operating conditions. In general, only a limited number of high response transient measurements are available for analysis of transient data. In order to determine the average pressure or temperature at a measurement station, a number of measurement sensors must be placed across the measurement plane. When a limited number of transient measurements are available to characterize the average station properties, transient measurement profile scale factors can be estimated from the more numerous steady-state measurements and experimentally derived time-dependencies. After completion of the transient calibration described in Chapter 4, transient pressure measurement profile scale factors (δP_{12}, δP_{13}, δP_2, and, δP_{21}) and temperature measurement profile scale factors (δT_{12}, δT_{13}, δT_2, and δT_{21}) are taken as known. Theory pertaining to thermocouple time response is presented in Appendix E. Thermocouple time response time constants (τT_{12}, τT_{13}, τT_2, and τT_{21}) are either assumed known or determined from the transient model calibration process described in Chapter 4. For time-dependent engine operation, the steady-state boundary layer profile at the fan inlet tip is lumped in with the transient pressure measurement profile.

Component degradation can result in a shift in tip-to-hub loading in rotating components that may not be detected using a limited data set. Adjustments to the sensitivity of the fault detection and diagnosis algorithms during transient operation to

account for higher uncertainties due, in part, to uncertainties in measurement profiles, sensor lags, and clearance effects are described in Chapter 6.

Transfer of heat from the fan gas stream to the metal during an engine acceleration or from the metal to the fan gas stream during an engine deceleration is determined from the fundamental heat transfer relationships described in Chapter 4. After completion of an initial transient calibration, fan heat soak characteristics, M_{Fan}, $CP_{m,\,Fan}$, h_{Fan}, and $A_{w,\,Fan}$ are taken as known.

Fan Analysis Summary

Fan analysis parameters are shown in Fig. 3.18.4. The tabulated parameters in Fig. 3.18.4 show typical test cell measurements, model inputs, fault detection and diagnostic inputs, and fan performance map scalars. *Initially, the performance map scalars are set at values predetermined by performance analysis to best represent the baseline component performance.* Map scalars are used extensively in the fault identification process. They are used in the baseline engine model to generate pretest predictions, provide an initial indication of data validity, and provide a baseline for trending component degradation. Finally, these same map scalars are varied in real-time to balance the engine cycle, match test data, and provide the basis for the real-time fault identification process.

Model inputs typically include P_{12}, P_2, P_{2A}, T_{12}, T_2, T_{2A}, T_{13}, T_{21}, FVV, WB_{13}, and XN_2. Fan performance map scalars introduced to account for the combined effects of modeling errors, fouling, erosion, and rubbing wear are denoted $\delta_{WC,FAN}$, $\delta_{T/T,FAN}$, and $\delta_{P/P,FAN}$. Corrections to fan flow and efficiency as a function of rotor-to-case clearance changes, Reynolds number index, RNI_{2A} and off-nominal guide vane schedules are either given or must be determined experimentally from test data (Wantland, 1991). Reynolds corrections to fan flow and efficiency as a function of Reynolds number index, RNI_{2A} are taken as known. All fan map adjustment parameters are used in the fault detection and diagnostic process.

High-Pressure Compressor Steady-State Analysis Summary

Compressor performance is characterized by the aerodynamic and thermodynamic equations relating pressure, temperature, and mass flow described in Appendix I. The station designation for a twin-spool turbofan engine is shown schematically in Fig. 3.19.1. Measured, known, and assumed parameters will vary from engine to engine and installation to installation. A summary of typical measured, known, assumed, and desired parameters for steady-state compressor analysis are shown in Fig. 3.19.2.

Compressor pressure measurements include compressor inlet total pressure, P_{24} and compressor discharge total pressure, P_3. Compressor inlet static pressure, PS_{24} and

engine control compressor discharge static pressure, QPS_3 are used in the fault detection and diagnosis process. Compressor temperature measurements include compressor inlet total temperature, T_{24} and compressor discharge total temperature, T_3. Engine control compressor inlet total temperature, QT_{24} is used in the fault detection and diagnosis process. Compressor discharge bleed flow rate, WB_3 is typically measured directly using a thin-plate square-edged orifice described previously in the section on Bleed Extraction Measurements. Compressor discharge bleed flow valve position, $WB_{3\ \text{Valve Position}}$ is used in the fault detection and diagnostic process. Miscellaneous measurements include compressor variable vane angle, CVV and compressor rotor speed, X_{N24}. Engine control compressor rotor speed, QXN_{24} is used in the fault detection and diagnostic process.

Parameters taken as **"known"** are determined from the steady-state model calibration process described in Chapter 4. Compressor secondary bleed flow rates WB_{24A}, WB_{24B}, WB_{24C}, ... WB_{24I} and enthalpies HB_{24A}, HB_{24B}, HB_{24C}, ... HB_{24I} are typically presumed to be a fraction of the total flow entering the compressor. Similarly, compressor bleed enthalpies are typically presumed to be a fraction of the available enthalpy rise in the compressor. Compressor inlet assumed flow coefficient, CF_{24} and compressor discharge assumed flow coefficient, CF_3 are used in the fault detection and diagnostic process to determine compressor inlet mass flow from compressor inlet conditions (W_{24X}) and compressor discharge mass flow from compressor discharge conditions (W_{3X}), respectively. Compressor inlet physical area, A_{24}, compressor discharge physical area, A_3, compressor inlet coefficient of linear expansion, $CQUL_{24}$,

compressor discharge coefficient of linear expansion, $CQUL_3$, compressor inlet adiabatic wall recovery factor, $RFAW_{24}$, and compressor discharge adiabatic wall recovery factor, $RFAW_3$ are taken as known. Reynolds corrections to compressor flow and efficiency as a function of Reynolds number index, RNI_{24} are taken as known. Clearance corrections to compressor flow and efficiency as a function of build clearance, temperature, and speed are also taken as known.

High-Pressure Compressor Transient Analysis Considerations

Compressor transient analysis parameters are summarized in Fig. 3.19.3. Parameters taken as **"known"** are determined from the transient model calibration process described in Chapter 4.

Compressor inlet mass flow rate, W_{24} is determined from the model-based governing equations described in Chapter 4. Transient pressure measurement profile scale factors (δP_{24} and δP_3) and temperature measurement profile scale factors (δT_{24} and δT_3) are taken as known. Temperature measurement time constants (τT_{24}, and τT_3) are also taken as known. After completion of an initial transient calibration described in Chapter 4, compressor heat soak characteristics, M_{HPC}, $CP_{m,\,HPC}$, h_{HPC}, and $A_{w,\,HPC}$ are taken as known.

High-Pressure Compressor Analysis Summary

Compressor analysis parameters are shown in Fig. 3.19.4. The tabulated parameters show typical test cell measurements, model inputs, fault detection and diagnostic inputs, and compressor performance map scalars or adders. Model inputs typically include P_{24}, P_3, T_{24}, T_3, CVV, WB_3, and XN_{24}. Compressor performance map scalars introduced to account for the combined effects of modeling errors, fouling, erosion, and rubbing wear are denoted $\delta_{WC,HPC}$, $\delta_{T/T,HPC}$, and $\delta_{P/P,HPC}$. Corrections to compressor flow and efficiency as a function of rotor-to-case clearance changes, Reynolds number index, RNI_{24} and off-nominal guide vane schedules are either given or must be determined experimentally from test data (Wantland, 1991). The process of scaling the compressor is described in Appendix I. All compressor map adjustment parameters are used in the fault detection and diagnostic process.

Burner Steady-State Analysis Summary

The burner station designation for a twin-spool turbofan engine is shown schematically in Fig. 3.20.1. Measured, known, and assumed parameters will vary from engine to engine and installation to installation. A summary of typical measured, known, assumed, and desired parameters for steady-state burner analysis is shown in Fig. 3.20.2.

Burner inlet pressure, P_{31} is typically assumed equal to the measured compressor discharge pressure, P_3. As a matter of preference, measurement rake total pressure losses are not separated from component performance. Burner inlet total temperature, T_{31} is typically assumed equal to the compressor discharge total temperature, T_3. Adjustments to burner inlet total temperature and pressure are required in the event compressor inbleed is used.

Fuel flow rate is calculated using the fuel flow meter pulse output, fuel temperature, and fuel and meter characteristics previously presented in the section on fuel flow. Three redundant flow measurements are typically calculated. Total engine fuel flow, WFT is calculated using facility meters; engine gas generator fuel flow, WFGG is calculated using an engine mounted meter; and fuel metering valve feedback flow, WFMC is calculated using the metering valve feedback. For steady-state analysis, burner fuel flow, WFB, is determined from the facility total fuel flow, WFT, for non-augmented engine operation and from the gas generator fuel flow, WFGG, during augmented engine operation. For steady-state engine analysis, fuel metering valve feedback flow, WFMC is used in the fault detection and diagnosis process.

Parameters taken as **"known"** include fuel lower heating value obtained per laboratory analysis (corresponding to thermochemical calorie), FHV, fuel kinematic viscosity during test, VISTST, and fuel specific gravity during test, FSG. Laboratory

samples are analyzed pre-test and verified post-test. Fuel lower heating value corresponding to the International Steam Table calorie, FHV_{IST} is used for thermodynamic analysis. At AEDC, the measured fuel lower heating value corresponds to the thermochemical calorie; the conversion from Thermochemical FHV, FHV_{LAB}, to the International Steam Table FHV, FHV_{IST} is:

$$FHV_{IST} = 0.99933 * FHV_{LAB}.$$

Thermodynamic analysis is based on an assumed burner efficiency, pressure drop, and secondary flows. Dry losses may be calculated as a function of any user-defined parameter such as corrected flow. Alternatively, dry losses may be specified as a function of $\Delta P / P$. Heat addition losses may be determined analytically or input using empirical data. Combustion efficiencies may be tabulated as a function of temperature rise and pressure or fuel-air ratio and pressure. In the present study, thermodynamic analysis was performed using empirical burner data provided by the engine manufacturer.

Burner Transient Analysis Considerations

Burner transient analysis parameters are summarized in Fig. 3.20.3. Parameters taken as **"known"** are determined from the transient model calibration process described in Chapter 4. For transient analysis, burner fuel flow, WFB, is determined from the

burner fuel metering valve feedback flow measurement, WFMC. After completion of an initial transient calibration as described in Chapter 4, burner heat soak characteristics, M_{Burner}, $CP_{m, Burner}$, h_{Burner}, and $A_{w, Burner}$ are taken as known.

Burner Analysis Summary

Burner analysis parameters are shown in Fig. 3.20.4. The tabulated parameters show typical test cell measurements, model inputs, and fault detection and diagnostic inputs. Model inputs typically include P_{31}, T_{31}, WFMC, and TFB. WFT or WFGG are used as model inputs for steady-state engine analysis. Fuel lower heating value corresponding to the International Steam Table calorie, FHV_{IST} is also used as a model input.

High-Pressure Turbine Steady-State Analysis Summary

High-pressure (HP) turbine performance is characterized by the aerodynamic and thermodynamic equations relating pressure, temperature, and mass flow described in Appendix J. The station designation for a twin-spool turbofan engine is shown schematically in Fig. 3.21.1. Measured, known, and assumed parameters will vary from engine to engine and installation to installation. A summary of typical measured, known,

assumed, and desired parameters for steady-state HP turbine analysis is shown in Fig. 3.21.2.

HP turbine power extraction, PWX_{41} is typically measured using a water brake system described previously in the section on Auxiliary Power Extraction. HP turbine pressure and temperature measurements are typically unavailable due to the small clearances and harsh operating environment between the components. If a limited number of HP turbine discharge temperature measurements are available, the temperature profile, in general, cannot be accurately characterized by a limited number of measurements and a correction (predetermined by performance analysis) is required to obtain the average gas path temperature.

A variety of approaches are available to determine core flow through the engine for a twin-spool turbofan and reduce the test measurements to a balanced cycle condition. For a mixed-flow turbofan engine, candidate approaches include:

1) determining HP Compressor inlet mass flow (W_{24}) from HP compressor inlet conditions (P_{24}, T_{24}, PS24, CF_{24X}, CT_{24}, and A_{24}),
2) determining core flow using an HP Compressor-HP turbine work balance,
3) determining core flow using a known HP turbine nozzle discharge mass flow function, and
4) determining core flow using a known LP turbine flow function.

For a twin-spool turbofan engine, the third option is typically used in the model-based governing equations to determine core flow through the engine and reduce the test measurements to a balanced cycle condition.

Parameters taken as **"known"** are determined from the steady-state model calibration process described in section Chapter 4. HP spool power loss, PWL_{41}, thermally corrected HP turbine nozzle discharge area, A_{41T}, and HP turbine nozzle discharge mass flow function, W_{41FFX} are typically provided by the engine manufacturer. In addition, HP turbine cooling flow available for HP turbine work, WC_{LHW}, HP turbine cooling flow not available for HP turbine work, WC_{LHNW}, HP turbine cooling flow energy available for HP turbine work, WHC_{HW}, and HP turbine cooling flow energy not available for HP turbine work, WHC_{HNW}, are assumed. HP turbine cooling flow rates are strongly linked to compressor secondary bleed flow rates WB_{24A}, WB_{24B}, WB_{24C}, ... WB_{24I} and enthalpies HB_{24A}, HB_{24B}, HB_{24C}, ... HB_{24I}. Reynolds and clearance corrections to HP turbine flow and efficiency are taken as known. Area ratio corrections to HP turbine flow and efficiency and cold-to-hot corrections on HP turbine flow are also taken as known.

High-Pressure Turbine Transient Analysis Considerations

HP turbine transient analysis parameters are summarized in Fig. 3.21.3. Parameters taken as **"known"** are determined from the transient model calibration process described in Chapter 4. HP turbine inlet mass flow rate, W_{41}, is determined from the model-based governing equations described in Chapter 4. After completion of an initial transient calibration as described in Chapter 4, HP turbine heat soak characteristics, M_{HPT}, $CP_{m,\,HPT}$, h_{HPT}, and $A_{w,\,HPT}$ are taken as known.

High-Pressure Turbine Analysis Summary

HP turbine analysis parameters are shown in Fig. 3.21.4. The analysis summary denotes typical test cell measurements, model inputs, fault detection and diagnostic inputs, and HP turbine performance map scalars or adders. Test cell measurements are typically limited to PWX_{41} and XN_{24}. Turbine losses include profile loss, profile incidence loss, secondary flow loss, blade tip clearance loss, cooling air injection aerodynamic loss, and cooling air thermodynamic loss. Changes in HP turbine

performance introduced to account for the combined effects of modeling errors and turbine losses is denoted by $\delta_{T/T,HPT}$.

As previously stated, thermally corrected HP turbine nozzle discharge area, A_{41T}, and HP turbine nozzle discharge mass flow function, W_{41FFX} are typically provided by the engine manufacturer. A variety of approaches can be used to determine core flow through the engine for a twin-spool turbofan and reduce the test measurements to a balanced cycle condition. Accordingly, a HP turbine gas flow scalar $\delta_{WC,HPT}$ may be determined in subsequent analysis to diagnose HP turbine anomalies.

Reynolds corrections to HP turbine flow and efficiency as a function of Reynolds number index, RNI_{41} are taken as known. Clearance corrections to HP turbine flow and efficiency as a function of build clearance, temperature, and speed are also taken as known. Area ratio corrections to HP turbine flow and efficiency are assumed to be included in the HP turbine nozzle discharge mass flow function. Cold-to-hot corrections on HP turbine flow are assumed to be included in the thermally corrected HP turbine nozzle discharge area. HP turbine map adjustment parameters are used in the fault detection and diagnostic process.

A key assumption used to evaluate HP and Low-Pressure (LP) turbine performance is an assumed LP turbine flow capacity, discussed in the following section, used to determine LP turbine inlet pressure. HP turbine exit pressure can then be

calculated using an estimated inter-turbine duct loss. As a matter of preference, inter-turbine duct losses are not separated from component performance.

Low-Pressure Turbine Steady-State Analysis Summary

LP turbine performance is characterized by the aerodynamic and thermodynamic equations relating pressure, temperature, and mass flow described in Appendix I. The station designation for a twin-spool turbofan engine is shown schematically in Fig. 3.22.1. Measured, known, and assumed parameters will vary from engine to engine and installation to installation. A summary of typical measured, known, assumed, and desired parameters for steady-state LP turbine analysis is shown in Fig. 3.22.2.

LP turbine temperature measurements typically include LP turbine discharge total temperature, T_5 and engine control LP turbine discharge total temperature, QT_5. Due in part to the harsh operating environment, only a limited number of LP turbine exit transient pressure measurements are typically available. A correction to the LP turbine discharge total temperature, T_5, is typically required to obtain the average gas path temperature. A large variation in the LP turbine exit swirl typically prohibits use of the LP turbine exit pressure, P_5, and LP turbine exit temperature, T_5, in the model-based governing equations. However, available LP turbine exit pressures and temperatures are used in the fault detection and diagnostic process. LP turbine inlet mass flow, W_{44}, is

determined using an assumed LP turbine nozzle discharge mass flow function and thermally corrected LP turbine nozzle discharge area.

Parameters taken as **"known"** are determined from the steady-state model calibration process described in Chapter 4. LP spool power loss, PWL_{44}, thermally corrected LP turbine nozzle discharge area, A_{44T}, and LP turbine nozzle discharge mass flow function, W_{44FFX}, are assumed. In addition, LP turbine cooling flow available for LP turbine work, WC_{LLW}, LP turbine cooling flow not available for LP turbine work, WC_{LLNW}, LP turbine cooling flow energy available for LP turbine work, WHC_{LW}, and LP turbine cooling flow energy not available for LP turbine work, WHC_{LNW}, are assumed. LP turbine cooling flow rates are strongly linked to fan and compressor secondary bleed flow rates and enthalpies.

Low-Pressure Turbine Transient Analysis Considerations

LP turbine transient analysis parameters are summarized in Fig. 3.22.3. Parameters taken as **"known"** are determined from the transient model calibration process described in Chapter 4. After completion of an initial transient calibration described in Chapter 4, LP turbine heat soak characteristics, M_{LPT}, $CP_{m,\,LPT}$, h_{LPT}, and $A_{w,\,LPT}$ are taken as known.

Low-Pressure Turbine Analysis Summary

LP turbine analysis parameters are shown in Fig. 3.22.4. The tabulated parameters show typical test cell measurements, model inputs, fault detection and diagnostic inputs, and LP turbine performance map scalars or adders. Test cell measurements are typically limited to XN_2, T_5 and QT_5. LP turbine losses include profile loss, profile incidence loss, secondary flow loss, blade tip clearance loss, cooling air injection aerodynamic loss, and cooling air thermodynamic loss. Changes in LP turbine performance introduced to account for the combined effects of modeling errors and turbine losses is denoted by $\delta_{T/T,LPT}$.

As previously stated, thermally corrected LP turbine nozzle discharge area, A_{44T}, and LP turbine nozzle discharge mass flow function, W_{44FFX}, are assumed. A variety of approaches can be used to determine core flow through the engine for a twin-spool turbofan and reduce the test measurements to a balanced cycle condition. Accordingly, a LP turbine gas flow scalar, $\delta_{WC,LPT}$, may be determined in subsequent analysis to diagnose LP turbine anomalies.

Reynolds corrections to LP turbine flow and efficiency as a function of Reynolds number index, RNI_{44}, are taken as known. Clearance corrections to LP turbine flow and efficiency as a function of build clearance, temperature, and speed are also taken as known. Area ratio corrections to LP turbine flow and efficiency are assumed to be

included in the LP turbine nozzle discharge mass flow function. Cold-to-hot corrections on LP turbine flow are assumed to be included in the thermally corrected LP turbine nozzle discharge area. All map adjustment parameters are also used in the fault detection and diagnostic process.

Mixer Steady-State Analysis Summary

Mixer performance is characterized by the aerodynamic and thermodynamic equations relating pressure, temperature, and mass flow described below. The station designation for a twin-spool turbofan engine is shown schematically in Fig. 3.23.1. Measured, known, and assumed parameters will vary from engine to engine and installation to installation. A summary of typical measured, known, assumed, and desired parameters for steady-state Mixer analysis is shown in Fig. 3.23.2.

Mixer temperature measurements typically include mixer inlet (core flow) temperature, T_6, determined from LP turbine discharge temperature, T_5, and mixer inlet (bypass flow) temperature, T_{16}. Effective mixing areas of the core and bypass flows, AE_{16} and AE_6 are assumed known.

Mixer Transient Analysis Considerations

Mixer transient analysis parameters are summarized in Fig. 3.23.3. Parameters taken as **"known"** are determined from the transient model calibration process described in Chapter 4.

Mixer Analysis Summary

Completely mixed stream Mach number and gas properties are calculated using one-dimensional fluid flow theory based on conservation of mass, momentum, and energy for viscous mixing of parallel gas streams developed by Kervyn D. Mach from the Air Force Aero Propulsion Laboratory (Mach, 1965). The solution technique developed by Mach provides a unique solution for the thermodynamic properties of the mixed stream in terms of the two initial streams and is sufficiently accurate for development and verification of the fault identification algorithms. The solution method, while computationally efficient, neglects all transverse property gradients and expresses the shear and energy transfer in terms of one-dimensional fluid properties. A two-dimensional flow solution to the equations of fluid motion would yield additional information about the shear layer mixing processes of fan and core flow during steady-state and transient engine operation. For the present study, two-dimensional flow

solutions to the equations of fluid motion were considered too slow to incorporate into the fault identification method.

Mixer analysis parameters are shown in Fig. 3.23.4. The tabulated parameters show typical test cell measurements, model inputs, and fault detection diagnostic inputs. Engine model cycle parameters such as mixer cold chute pressure drop, mixer cold chute exit effective area, augmentor inlet pressure drop, and exhaust nozzle flow coefficient can be varied to match the measured test data. For the present research, bypass duct pressure drop is varied to match measured test data and a scale factor on bypass duct pressure drop is used in the fault detection and diagnostic process.

Augmentor Steady-State Analysis Summary

Augmentor station designation for a twin-spool turbofan engine is shown schematically in Fig. 3.24.1. Measured, known, and assumed parameters will vary from engine to engine and installation to installation. A summary of typical measured, known, assumed, and desired parameters for steady-state augmentor analysis is shown in Fig. 3.24.2.

Augmentor Transient Analysis Considerations

Augmentor transient analysis parameters are summarized in Fig. 3.24.3. Parameters taken as **"known"** are determined from the transient model calibration process described in Chapter 4.

Augmentor Analysis Summary

Augmentor analysis parameters are shown in Fig. 3.24.4. The tabulated parameters show typical test cell measurements, model inputs, fault detection and diagnostic inputs.

Nozzle Steady-State Analysis Summary

Nozzle and overall engine performance are characterized by the aerodynamic and thermodynamic equations relating pressure, temperature, and mass flow described in Appendix L. Nozzle station designations for twin-spool turbofan engines are shown schematically in Fig. 3.25.1. Measured, known, and assumed parameters will vary from

engine to engine and installation to installation. A summary of typical measured, known, assumed, and desired parameters for steady-state nozzle analysis is shown in Fig. 3.25.2.

Nozzle Transient Analysis Considerations

Nozzle transient analysis parameters are summarized in Fig. 3.25.3. Parameters taken as "known" are determined from the transient model calibration process described in Chapter 4.

Nozzle Analysis Summary

Nozzle analysis parameters are shown in Fig. 3.25.4. The tabulated parameters show typical test cell measurements, model inputs, fault detection and diagnostic inputs. Overall engine performance is characterized by the aerodynamic and thermodynamic equations relating pressure, temperature, mass flow, and engine thrust and specific fuel consumption described in Appendix L. Measured engine gross thrust, F_G, net thrust, F_N, and specific fuel consumption, SFC are all used in the fault detection and diagnostic process.

4. MODEL FORMULATION AND IMPLEMENTATION

Implementation of the nonlinear cycle model and analysis of large amounts of data provided for the gas path analysis impose a considerable computational burden, not only in terms of cycle time, but also in terms of divergence control of the nonlinear cycle model. Several approaches were employed to increase computational speed and control divergence. The first was to simultaneously balance the cycle equations and minimize errors between the cycle model and the data. The second was to optimize the combination of perturbation variables and cycle constraints[1] associated with the nonlinear cycle model. This maximized the accuracy of the simulation and ensured convergence with minimal iteration. The third approach, "task level" parallel processing, was used to provide real-time operation and will be described in more detail in Chapter 5.

Numerical Solution to the Cycle Model Equations

The REPLICAS numerical solver (McLaughlin, 1982) was used to solve the systems of equations. REPLICAS is a modified Newton-Raphson solver which provides a converged solution to the systems of cycle model equations described below. An initial coefficient matrix is calculated by perturbing each iteration variable and then measuring

[1] Perturbation variables and cycle constraints associated with the nonlinear model are discussed in detail later in this section.

the changes to the error terms that result. A scaling algorithm is applied and then the matrix of partial derivatives is inverted. The resulting inverted matrix is a linear approximation which characterizes the model in the neighborhood of the initial set of iteration variables. An iterative solution to the nonlinear cycle model is performed using estimates of the iteration variables obtained from the error terms and the current matrix of iteration coefficients. Following each execution of the cycle model, the error terms are used with the previous errors and the previous iteration variables to improve the iteration matrix using Broyden's method (Broyden, 1967). The iteration matrix is inverted upon each Broyden update and provides new scaling values by operating on the updated partial matrix. Effective use of the partials correction in the solver significantly reduced the need to recalculate derivatives and minimized any potential benefits associated with parallelizing the numerical solver. Details of the solver subroutine design are provided by McLaughlin (McLaughlin, 1982) and by Chappell (Chappell, 1991).

There are two model sub-systems to be solved by the numerical solver, the first model sub-system comprises the cycle compatibility equations which typically include airflow rate continuity, shaft power balances, and a static pressure balance for the fan and core flows at the mixer entrance. The unknowns are the operating variables such as operating points on the component maps, inlet airflow rate, and bypass ratio. This sub-system, which balances the cycle and ensures the physical validity of the cycle, must be solved exactly (within a small error tolerance). The second model sub-system is composed of the identity between the various test measurements and the corresponding

parameters simulated by the cycle model which can include identities between measured and simulated net thrust, engine inlet airflow, combustor fuel flow, nozzle throat area, and internal pressures and temperatures throughout the engine. If the number of measurements in the second model sub-system is greater than the number of correction factors on the component characteristics (e.g., flow and efficiency scalars) then the second sub-system cannot be solved exactly and the system is "solved" by minimizing the deviation between the test measurements and the simulated values. As previously discussed in the literature review, a variety of system identification methods exist for estimating unknown model parameters from available data by minimizing a performance index. Examples include a least-squares estimate, which minimizes the sum of the squares of the deviations and a weighted least-squares-estimate, which minimizes the weighted sums of squares of deviations. Alternatively, one may use the maximum likelihood philosophy, which maximizes the probability that a measurement set actually occurred, taking into account known statistical properties. Another approach uses Bayesian estimation (when statistical models are available) and the criterion of optimality is the *a posteriori* conditional density function. When the *a priori* density function is uniform, this estimate is equal to the maximum likelihood estimate. Kalman formulated and solved the Wiener problem for Gauss-Markov sequences through the use of a state-space model representation and the viewpoint of conditional disturbances and expectations (Kalman, 1960). For Gaussian random variables, identical results are obtained by all methods as long as the assumptions are the same in each case (Smith,

G.L. 1965). As a result, researchers are tempted to claim that the estimates they produce are either maximum likelihood or Bayesian minimum variance estimates.

Appropriate test measurements and uncertainties, and component updates and sensor biases must be identified to simultaneously detect and diagnose data anomalies and component faults in real-time. The actual process of selecting the sensor and cycle model updates is based on engineering analysis and an exhaustive computer-based search using a performance index to characterize how well measurement errors and changes in component performance can be characterized in addition to how well key performance parameters such as thrust can be predicted in the presence of assumed measurement uncertainties and modeling errors. As previously discussed, initial studies are performed using a Truth Model with embedded modeling errors and sensor uncertainties to determine the most likely set of sensor/model updates. Follow-on studies and analysis are performed with actual test data, for nominal and degraded engine operation, and "seeded" measurement errors to determine the final set of sensor/model updates.

The unknown parameters for the fault identification process include operating variables (operating points on the component maps, inlet airflow rate, bypass ratio, etc.), sensor biases, profile errors, component performance variations from nominal values (deviations in flows and efficiencies), time constants, sensor lags, pressure losses, etc. The engine cycle equations to be solved include the compatibility equations (airflow rate continuity, shaft power balances, and, for mixed exhaust turbofan cycles, a static pressure

balance for the fan and core flows at the mixer entrance). For completeness, measurement system and data reduction schematics are shown in Figs. 4.1 and 4.2 respectively. Fig. 4.3 outlines the cycle-balancing and fault identification process for serial processing of the model-based algorithms. Typical test measurements used to resolve the second model sub-system for the research described herein were identified and presented in the previous section and are shown schematically in Fig. 4.4. The corresponding perturbation variables and error terms for the real-time fault identification model are provided below.

- Perturbation Variables
  - Bypass Ratio
  - HPT Pressure Ratio
  - LPT Pressure Ratio
  - HPT Efficiency Scale Factor
  - LPT Efficiency Scale Factor
  - Bypass Duct Pressure Ratio Scale Factor

- Error Terms
  - HPT Airflow Continuity
  - LPT Airflow Continuity
  - Static Pressure Balance for the Fan and Core Flows at the Mixer Entrance
  - HPT Power Balance

- LPT Power Balance
- Nozzle Airflow Continuity

As with typical turbofan cycle calculations, continuity is satisfied for each of the rotating components through the use of an error term based on a calculated value of component corrected flow and a target value of component corrected flow that is determined from the component performance map. The combination of perturbation variables and constraints differs considerably from typical turbofan cycle calculations, which require guesses on fan speed, operating point on the fan map, core speed, operating point on the compressor map, and combustor fuel/air ratio. In order to facilitate computer reading of fan and compressor maps, the maps are typically transformed from their usual presentation (Fig. 4.5.1) using ray lines or a similar operating parameter (Fig. 4.5.2). Ray lines are nonintersecting lines that are similar in shape and lie roughly parallel to the compressor stall line. All ray lines intersect all speed lines and tend to spread apart at higher speeds. Entry into the fan and compressor maps is typically through two independent variables, ray line and corrected speed. As previously stated, fan and compressor models contains many levels of complexity and typically include corrections to flow and efficiency as a function of rotor-to-case clearance changes, Reynolds number index, and off-nominal guide vane schedules.

Only a limited number of test measurements (equal to number of correction factors on the component characteristics) are used in the present research to resolve the

second model sub-system which is composed of the identity between the various test measurements and the corresponding parameters simulated by the cycle model. Benefits of using of only a limited number of measurements include:

- increased sensitivity to measured disturbances (compared to rectangular techniques using additional measurements) making it ideal for analyzing small performance deviations;

- availability of redundant measurements for validation of test data and modeling assumptions;

- more reliable nonlinear cycle model convergence resulting, in part, from the limited measurement set and associated perturbation variables and error terms;

- increased ability to identify faults associated with a limited number of sensors; experience shows that as more and more sensors are added, more sensor errors are introduced than performance information, and the ability to identify faults is reduced (Adibhatla, 1994).

As previously stated, identification methods where the number of knowns equals the number of unknowns are commonly referred to as "square" identification methods. In ordinary least squares we are given a "rectangular" system with more measurements

than free variables. The least squares solution to the overdetermined system minimizes the error in the individual measurements. It is clear, without complicated formulas, that if the measurement errors have the same size, every new measurement added to the measurement set decreases the estimation error and decreases the sensitivity to any one measurement. Moreover, it is obvious that adding unreliable sensors will decrease the sensitivity, decrease the estimation accuracy, and reduce the ability to identify faults. As previously noted in the literature review, the overall technology of system identification includes analytic methods of specifying test input signals to minimize the uncertainty of estimates of parameters and of specifying sensor accuracy requirements.

Design criteria considered in the development of the fault identification algorithms were previously presented in Chapter 1. The actual process of selecting the sensor and cycle model updates is based on engineering analysis and an exhaustive computer-based search using a performance index to characterize how well measurement errors and changes in component performance can be characterized in addition to how well key performance parameters such as thrust can be predicted in the presence of assumed measurement uncertainties and modeling errors. There are a number of practical considerations which must be considered in the selection process including requirements for physically meaningful cycle model updates, reliable nonlinear cycle model convergence, and real-time identification of faults. It is unreasonable to assume that the cycle model is exact, or that the errors in the model and measurements have the same magnitude (or the same units), or that the mean-time-between failure of the

measurements and components for developmental engine tests are known *a priori*. In addition, one must consider the influence of the test measurements on the cycle model predictions over the engine's "entire" operating range.

Overview of Parallel Computing Processes

Parallel computing processes shown to greatly enhance the real-time computing capability include simultaneous calibration of component maps, calculation of engine processes (cycle balancing) and integration of dynamic quantities (rotor and gas dynamics) for time-dependent data. The software was parallelized using a "replicated worker" approach described in detail in Chapter 5. Different time samples are sent to different processors for cycle balancing and calculation of engine processes (Fig. 4.6). The approach uses one central processor to collect engineering unit data from the network, filter the raw data, distribute the data between the processors, collect and order the results, identify component performance changes and measurement errors from the cycle model data, and run the GUI. Special care must be taken to correlate time-dependent processes. Surprisingly, use of cycle model segmentation and distribution algorithms from the "Parallel Processor Engine Model Program" (NASA-CR-174641, January 1984) failed to greatly enhance the real-time capability of the parallel technique and tended to increase the effect of numerical errors and their cumulative effects over time.

Model Calibration

The real-time fault identification process requires three distinct cycle model calibration steps (Figs. 4.7 – 4.9) to minimize errors in the engine cycle model that are often significant for developmental engine testing. In general, the model calibration steps described can be performed in conjunction with normally scheduled test activities. First, a steady-state engine calibration is performed at a specified flight condition to document engine health at intermediate and part-power conditions. This is performed to calibrate the steady-state component characteristics and to adjust the transient measurements for steady-state bias and profile errors using the more accurate and numerous steady-state data. Next, a transient calibration is performed to determine time-dependent effects. Finally, continuous real-time calibration of the cycle model is performed to identify faults that may occur during subsequent developmental engine testing.

Step 1- Steady-State Model Calibration

A steady-state performance calibration is performed at a specified flight condition to establish engine health at intermediate and part-power conditions and to update the baseline component characteristics. Estimation accuracy is enhanced using a detailed spatial array of pressure and temperature measurements at each of the measurement stations. The spatial array of measurements is ultimately combined to determine area weighted ring (or circumferential) averages and the mean station average. Corrections for steady-state flow profile errors associated with single-element time-dependent probes

are determined from the test data. After completion of the steady-state cycle model calibration, the cycle model and adjusted component maps serve as a baseline to characterize steady-state engine operation and engine health at intermediate and part-power conditions.

Step 2 - Transient Model Calibration

The transient calibration process estimates the effects of sensor lags, profile errors, rotor dynamics, heat transfer, and volume dynamics. Test data are acquired during a rapid intermediate-to-idle power sweep (engine deceleration), followed by a rapid idle-to-intermediate power sweep (engine acceleration) to characterize time-dependent effects. Likewise, time-dependent augmentor characteristics are estimated during a rapid augmentor power sweep.

The calibration approach must extract reliable values from data with normal measurement noise and permit identification of a wide variety of faults during both steady-state and transient engine operation. Corrections for transient lag in aerothermodynamic temperature measurements can be estimated directly from the test data. All other aerothermodynamic sensors provided satisfactory responses such that corrections for transient time lag were small and could be effectively compensated for in conjunction with other time-dependent effects. When available, multiple temperature and/or pressure sensors are used at each measurement station to minimize the effects of transient profile errors associated with single-element probes.

Rotor dynamics are an essential component of the transient model calibration process. Acceleration or deceleration of a rotor occurs when the turbine horsepower either exceeds or is less than the horsepower required by the fan or compressor. For the engine cycle model, these relationships are calculated from the thermodynamic relationships of the respective components and the mass flow distribution through the cycle. The basic relationship governing rotor dynamics is expressed as

$$\frac{dN}{dt} = \frac{K(ExcessHorsepower)}{J * N}$$

where

$\frac{dN}{dt}$ is the time rate of change of shaft speed,

$K = (60/2\pi)^2 550$,

J is the rotor moment of inertia,

N is the instantaneous shaft speed, and the

Excess Horsepower is the turbine horsepower minus the compressor horsepower minus the extracted horsepower.

Rotor dynamic effects are determined using the above relationship and speed derivatives obtained from the measured test data. The effects of heat transfer, volume dynamics, sensor lag, and other transient effects such as tip and seal clearance changes are also estimated from the test data. An empirical method for estimating the effects of heat transfer and volume dynamics was developed in the present study. The empirical method is well-suited for real-time fault identification and avoids inaccurate estimates relative to metal mass, heat-transfer coefficients, and calculation of metal temperature derivatives.

Inaccuracies in component efficiencies prior to calibration of the transient cycle model are shown in Fig 4.10 for a rapid engine deceleration and in Fig 4.11 for a rapid engine acceleration followed by a rapid engine deceleration. The component efficiencies shown in Figs. 4.10 and 4.11 were computed from the measured test data taking into account rotor dynamics but excluding the effects of sensor lags, heat transfer, and volume dynamics. The anomalous component efficiencies shown in Fig. 4.10 highlight the effects of thermocouple lag during the initial deceleration and the effects of heat transfer after the engine has decelerated to the lower power setting. The data shown at the end of the engine deceleration clearly show an abnormally high LPT efficiency and correspondingly low FAN efficiency. The abnormally low FAN efficiency after the

engine has decelerated is due primarily to the lack of compensation for the effects of FAN heat transfer. The correspondingly high LPT efficiency after the engine has decelerated is primarily due to the lack of FAN heat transfer but also includes the effects of HPC, BURNER, HPT, LPT, MIXER, AUGMENTOR, and NOZZLE heat transfer. Significant deviations between the measured and observed[2] turbine exit temperature are seen in Figs. 4.12 and 4.13 for the uncalibrated transient cycle model.

Often times, knowledge of sensor and engine dynamics is unknown *a priori*, and a system identification experiment must be conducted to obtain the necessary information. System identification experiments might include lab testing of thermocouples to determine their reference time constants, τ_o, and the effects of variations in the gas temperature, T_{gas}, gas stream Mach number, Ma, and gas stream static pressure, Ps, on the time constant. System identification experiments might also require full-scale engine tests, with well-instrumented engines, to determine pressure and temperature profiles and heat sink/soak characteristics during transient engine operations.

Estimation of Heat Transfer Characteristics and Thermocouple Time Constants

In the absence of *a priori* knowledge of thermocouple sensor lags (τ) and component heat transfer characteristics, such as M, the metal mass, C_{pm}, the metal specific heat, h, the overall heat transfer coefficient, and A_w, the effective wetted surface

[2] The term **"observed"** is standard state estimation terminology, whereas, the terms

area, the fault detection and diagnostic algorithms can be used to estimate the effects of these characteristics on engine performance. For the FAN alone, there are sixteen unknown parameters to be estimated: four transient pressure measurement profile scale factors (δP_{12}, δP_{13}, δP_2, and, δP_{21}); four temperature measurement profile scale factors (δT_{12}, δT_{13}, δT_2, and δT_{21}); four temperature measurement time constants (τT_{12}, τT_{13}, τT_2, and τT_{21}); and four FAN heat soak characteristics (M_{Fan}, $CP_{m,\,Fan}$, h_{Fan}, and $A_{w,\,Fan}$). Often times developmental engines include sufficient high response instrumentation to determine transient temperature and pressure measurement profiles.

The model and fault identification results presented herein do not assume *a priori* knowledge of thermocouple sensor lags, transient temperature or pressure measurement profiles, or heat transfer characteristics. Rotor polar moments of inertia, however, were known. In the absence of *a priori* knowledge of heat transfer coefficients, the overall heat transfer coefficients, for the research reported herein, were assumed to be of the form

$$h = f(h_o, Re)$$

"simulated" and **"predicted"** are standard engine simulation terminology.

where, h_o is a reference overall heat transfer coefficient, and Re is a Reynolds number used to correct or adjust h to h_o. The above equation for the overall heat transfer coefficient with Reynolds number correction becomes

$$h = h_o \left(\frac{W}{W_o} \right)^c$$

where, W and W_o are the as tested and reference mass flow rates respectively, and the exponent c, is typically assumed to be a constant on the order of 0.8.

The process of identifying the unknown sensor lags and heat sink/soak characteristics is, by necessity, an iterative process in which the model is refined at each step in the process. There are a number of approaches that can be used to estimate thermocouple time constants. Two simple approaches to estimating the time constant for low pressure turbine exit temperature, T5, were previously outlined in Chapter 3 for a 5-10 second engine acceleration from idle to 95% of full power at a sea-level static operating conditions. The first approximation of the time constant can be evaluated by noting the slope of the recorded data when the compressor mass flow and speed have stabilized and the time when that slope intersects the peak temperature level. Alternately, the time constant can be evaluated by measuring the time between when the compressor mass flow and speed have stabilized and the time where the recorded temperature reaches

63 percent of the difference in the temperature between the starting point and the peak temperature measurement.

A more consistent approach which can be used to determine the thermocouple time constants is to iteratively employ the thermodynamic engine model and diagnostic capability, developed in the present research, to estimate the sensor lags in addition to the heat sink/soak and transient profile effects. Without explicitly accounting for the sensor lags and heat soak/sink effects, the fault diagnostic algorithms, which are described in detail in Chapter 6, will attempt to characterize these effects as measurement errors in the temperature and fuel flow measurements. The probability of any individual measurement error (temperature, fuel flow, pressure, airflow, thrust, etc.) will depend on the engine cycle, the transient maneuver performed, the sensor lag and heat sink/soak characteristics of the components, and other transient effects including profile errors. The model predictions presented in Chapter 7 were produced with only a few iterations to quantify the transient effects. During the iterative calibration process, the diagnostic algorithms are reconfigured to identify the magnitude and probability of errors in the assumed sensor lags and heat sink/soak characteristics.

In turbine engines there are significant changes in pressure, density, and temperature which must be related by the aerothermo equations. Thermodynamic changes in high-speed flow of a gas introduce two dimensionless parameters, Mach number and specific-heat ratio. The effect of the specific-heat-ratio is only slight to

moderate, but Mach number exerts a strong effect on compressible-flow properties if it is greater than ~ 0.3.

Development of Non-dimensional Number to Characterize Algorithm Stability and Accuracy

The foundation of the dimensional-analysis method rests on two basic assumptions: (1) that the proposed physical relation is dimensionally homogenous and (2) that all the relevant variables have been included in the proposed relation. The Malloy number will be derived for the purpose of analyzing the stability and accuracy of the explicitly parallelized model-base algorithms. The derivation to follow proceeds with the knowledge that the literature is abound with dimensionless formulations which do not resolve the process.

With any numerical technique, stability and accuracy criteria must be developed to assure the stability and accuracy of the process (e.g., Leppert (1973) and Fowler (1945)). In order to make an assessment of the stability and accuracy of the parallelized model-based algorithms, it is useful to establish an appropriate non-dimensional number, or numbers, to quantify the stability and accuracy. Methods such as non-dimensionalization of the governing equations, the power-product method, or the Buckingham Pi theorem are often used in the development of non-dimensional numbers.

For a lumped system analysis, the time constant is of importance as is the time-rate-of-change of the gas temperature. In order to determine the differential equation that describes the metal temperature as a function of time, an energy balance is performed on the metal to equate the rate of heat loss by convection to the gas to the rate of decrease in internal energy of the metal.

Letting T_m be the temperature of the metal at time t, we write

$$hA(T_m - T_{gas}) = -\rho CV \frac{dT_m}{dt}$$

Assuming the Biot number is sufficiently small, the lumped system analysis is governed by

$$\frac{dT_m}{dt} + \left(\frac{hA}{\rho cV}\right) T_m = \left(\frac{hA}{\rho cV}\right) T_{gas}$$

or

$$\frac{dT_m}{dt} + \left(\frac{1}{\tau}\right) T_m = \left(\frac{1}{\tau}\right) T_{gas}$$

where, the time constant, τ is

$$\left(\frac{\rho c V}{hA}\right).$$

If the gas temperature $T_{gas,}$ remains constant, we could set the right-hand side to zero and use separation of variables to determine the complementary or transient solution. Then, noting that the non-homogeneous part of the differential equation is a constant, one would assume the metal temperature is constant and substitute back into the differential equation to determine the particular or steady-state solution. An alternate approach is to define a new dependent variable and solve the resulting homogeneous ordinary differential equation. During transient engine operation, the gas temperature varies with time and the particular solution to the differential equation is solved using techniques such as the method of undetermined coefficients or variation of parameters.

If the gas temperature remains constant and the Biot number is sufficiently small, the solution to the lumped system is

$$T_m = (T_{m,i} - T_{gas})e^{-t/\tau} + T_{gas}.$$

If the gas temperature is assumed constant, the metal temperature can be evaluated at any time t, using only the initial metal temperature (i.e., independent of *a priori* knowledge of the time history of the metal temperature). Furthermore, if the time constant $\tau >> 1$, the metal temperature will remain nearly constant and the estimated metal temperature will be insensitive to the time-rate-of-change of the gas temperature.

To fully consider the time-rate-of-change of the gas temperature in a turbine engine, one must consider component operating characteristics including stall margin, burner and augmentor blowout margin and ignition characteristics, and engine control laws. Furthermore, one must consider the variation in the overall heat transfer coefficient, h, which is highly dependent on Reynolds number. Therefore, the dimensional analysis that follows assumes that the maximum time-rate-of-change of the gas temperature for transient engine operation of interest is known and, for the period of interest, the overall heat transfer coefficient remains constant.

In the absence of *a priori* knowledge of a time-dependent gas temperature, one would typically assume that the path between the two points is linear in nature. When considering heat transfer in the FAN or HPC, the gas temperature is typically measured and the time dependent nature of the gas temperature is known. However, when considering the heat transfer characteristics of the BURNER, HPT, and LPT, the gas temperature is typically not measured and the gas temperature must be determined from the model-based aerothermo relations. In the absence of *a priori* knowledge of the time

dependent nature of the gas temperature, a linear relationship between two points in time is assumed for the error analysis presented herein. After completion of the error analysis, techniques are developed to better estimate the time dependent nature of the gas temperature (prior to cycle-balancing) in order to reduce the error in the estimated metal temperature.

For the case where the time constant $\tau << 1$, the metal temperature would be expected to follow the gas temperature very closely (Fig. 4.14) and, assuming a linear relationship for the time-dependent gas temperature, the maximum error in the estimated metal temperature $(T_m)_{error}$, is on the order of

$$\frac{1}{2}\left(\frac{dT_{gas}}{dt}\right)_{max} \Delta t .$$

where, $\left(\frac{dT_{gas}}{dt}\right)_{max}$ is the maximum rate of change of the gas temperature and Δt is the elapsed time. It is important to note however, that at time $t = t_i + \Delta t$ in Fig. 4.14, the error in the estimated metal temperature is negligible for cases where the time constant, $\tau << 1$. For situations where the time constant, τ is on the order of 1, the minimum error in the estimated metal temperature occurs when the change in the actual metal temperature is linear. In that case, the error in the estimated metal temperature is zero. In order to

evaluate the maximum error in the estimated metal temperature, the following functional relationship for the time dependent change in the gas temperature is assumed (Fig. 4.15)

$$\text{for } t < \Delta t/2,\ T_{gas}(t) = \left(\frac{dT_{gas}}{dt}\right)_{max} t$$

$$\text{for } t > \Delta t/2,\ T_{gas}(t) = \left(\frac{dT_{gas}}{dt}\right)_{max} (\Delta t - t)$$

This means that for the period of interest, the gas temperature is assumed to increase (from time $t = t_i \rightarrow t_i + \Delta t/2$) at the maximum possible rate and then decreases at the maximum possible rate from time $t = t_i + \Delta t/2 \rightarrow t_i + \Delta t$.

For the first half of the period under consideration,

$$\frac{dT_m}{dt} + \frac{1}{\tau} T_m = \frac{t}{\tau}\left(\frac{dT_{gas}}{dt}\right)_{max}$$

and for the second half of the period under consideration,

$$\frac{dT_m}{dt} + \frac{1}{\tau} T_m = \frac{\Delta t - t}{\tau}\left(\frac{dT_{gas}}{dt}\right)_{max}$$

letting

$$m_1 = \frac{1}{\tau}$$

and

$$m_2 = \frac{1}{\tau}\left(\frac{dT_{gas}}{dt}\right)_{\max}$$

we have, for the first half of the period of interest

$$\frac{dT_m}{dt} + m_1 T_m = m_2 t\,.$$

The metal temperature is determined from the complementary and particular solutions

$$T_m = c_1 e^{-m_1 t} + \frac{m_2}{m_1} t - \frac{m_2}{m_1^{\,2}}\,.$$

Application of the initial condition gives

$$c_1 = T_{m,i} + \frac{m_2}{m_1^2}$$

assuming $T_{m,i} = 0$ and $t_i = 0$, for the first half of the period of interest,

$$\left(T_m\right)_{t=0 \rightarrow \Delta t/2} = \tau e^{-t/\tau}\left(\frac{dT_{gas}}{dt}\right)_{max} + t\left(\frac{dT_{gas}}{dt}\right)_{max} - \tau\left(\frac{dT_{gas}}{dt}\right)_{max}$$

Before completing the derivation of the error in the estimated metal temperature, we now analyze the error halfway through the period of interest. Based on physical reasoning, the maximum error in the assumed gas temperature occurs when the gas temperature varies at the maximum possible rate from the assumed variation, which is a linear variation (Fig. 4.15). Based, once again, on physical reasoning, the error in the metal temperature will increase throughout the first half of the period of interest, and then decrease slightly during the second half of the period of interest. That is, if the gas temperature increases linearly, the metal temperature will lag behind the gas temperature. Once the gas temperature starts to decrease, the metal temperature will continue to increase until the gas temperature passes below the metal temperature. Then, the metal temperature (and corresponding error in the estimated metal temperature) will start to decrease.

Although the error at any time during the period could be determined analytically, only the error at the end of the period (time $t = t_i + \Delta t$) is of interest since the metal

temperature at all other times will be calculated, in parallel, on other processors. Three examples are presented below to quantify the error halfway through the period of interest assuming a time constant, τ of 1 and a maximum rate of change of the gas temperature, $\left(\frac{dT_{gas}}{dt}\right)_{max}$ of 100 degR/second. In all the examples to follow, Δt is equal to the number of processors divided by the data sampling rate. In the first example, data is sampled at 100 samples/second and processed on a single processor. As expected, the error in the estimated metal temperature for serial operation is negligibly small.

Example 1 (serial processing, data sampled at 100 samples/second)

$\left(\frac{dT_{gas}}{dt}\right)_{max}$ = 100 degR/second

$\tau = 1$

Δt = .01 seconds

$(T_m)_{error}$ = .001 degR

In the second example, data is sampled at 100 samples/second and processed on 10 processors, corresponding to a Δt of 0.1 seconds. As expected, the error in the estimated metal temperature for parallel operation on 10 processors increases, but is still extremely small.

Example 2 (parallel processing, 10 processors, data sampled at 100 samples/second)

$$\left(\frac{dT_{gas}}{dt}\right)_{max} = 100 \text{ degR/second}$$

$\tau = 1$

$\Delta t = .1$ seconds

$(T_m)_{error} = .123$ degR

In the third example, data is sampled at 100 samples/second and processed on 100 processors, corresponding to a Δt of 1.0 seconds. For $\tau = 1$, $\left(\frac{dT_{gas}}{dt}\right)_{max} = 100$ degR/second, and parallel processing on 100 processors, the maximum error in the estimated metal temperature (assuming a linear variation in the gas temperature over a Δt of 1.0 second) is still relatively small with respect to the maximum difference between gas and metal temperatures in gas turbine engines which is on the order of a few hundred degrees Rankine.

Example 3 (parallel processing, 100 processors, data sampled at 100 samples/second)

$$\left(\frac{dT_{gas}}{dt}\right)_{\max} = 100 \text{ degR/second}$$

$\tau = 1$

$\Delta t = 1$ second

$(T_m)_{error} = 10.65$ degR

From before, for the second half of the period of interest

$$\frac{dT_m}{dt} + \frac{1}{\tau} T_m = \frac{\Delta t - t}{\tau} \left(\frac{dT_{gas}}{dt}\right)_{\max}$$

letting

$$m_1 = \frac{1}{\tau}$$

$$m_2 = \frac{1}{\tau} \left(\frac{dT_{gas}}{dt}\right)_{\max}$$

and

$$m_3 = \frac{\Delta t}{\tau}\left(\frac{dT_{gas}}{dt}\right)_{\max}$$

we have, for the second half of the period of interest

$$\frac{dT_m}{dt} + m_1 T_m = m_3 - m_2 t\,.$$

The metal temperature is determined from the complementary and particular solutions

$$T_m = c_1 e^{-m_1 t} - \frac{m_2}{m_1} t + \frac{m_2}{m_1^2} + \frac{m_3}{m_1}\,.$$

The initial condition is evaluated at time t = Δt/2 and the corresponding metal temperature, denoted $T_m(\Delta t/2)$, was previously determined. Applying the initial condition gives

$$T_m(\Delta t/2) = c_1 e^{-m_1 \Delta t/2} - \frac{m_2}{m_1}\frac{\Delta t}{2} + \frac{m_2}{m_1^2} + \frac{m_3}{m_1}$$

or

$$c_1 = \left(T_m(\Delta t/2) + \frac{m_2}{m_1}\frac{\Delta t}{2} - \frac{m_2}{m_1^{\,2}} - \frac{m_3}{m_1} \right) e^{m_1 \Delta t/2}$$

equivalently

$$c_1 = \left(T_m(\Delta t/2) + \frac{\Delta t}{2}\left(\frac{dT_{gas}}{dt}\right)_{\max} - \tau\left(\frac{dT_{gas}}{dt}\right)_{\max} - \Delta t\left(\frac{dT_{gas}}{dt}\right)_{\max} \right) e^{\Delta t/2\tau}.$$

For the second half of the period of interest,

$$\left(T_m\right)_{t=\Delta t/2 \rightarrow \Delta t} = c_1 e^{-t/\tau} - t\left(\frac{dT_{gas}}{dt}\right)_{\max} + \tau\left(\frac{dT_{gas}}{dt}\right)_{\max} + \Delta t\left(\frac{dT_{gas}}{dt}\right)_{\max}$$

The error in the estimated metal temperature at the end point (the point of interest), can now be quantified. As seen in the examples below, the error in the estimated metal temperature for serial operation is negligibly small, the error in the estimated metal temperature for parallel operation on 10 processors extremely small, and the maximum error in the estimated metal temperature (assuming a linear variation in the gas temperature over a Δt of 1.0 second) is still relatively small with respect to the maximum difference between gas and metal temperatures in gas turbine engines.

Example 1 (serial processing, data sampled at 100 samples/second)

$$\left(\frac{dT_{gas}}{dt}\right)_{max} = 100 \text{ degR/second}$$

$\tau = 1$

$\Delta t = .01$ seconds

$(T_m)_{error} = .002$ degR

Example 2 (parallel processing, 10 processors, data sampled at 100 samples/second)

$$\left(\frac{dT_{gas}}{dt}\right)_{max} = 100 \text{ degR/second}$$

$\tau = 1$

$\Delta t = .1$ seconds

$(T_m)_{error} = .24$ degR

Example 3 (parallel processing, 100 processors, data sampled at 100 samples/second)

$$\left(\frac{dT_{gas}}{dt}\right)_{max} = 100 \text{ degR/second}$$

$\tau = 1$

$\Delta t = 1$ second

$(T_m)_{error} = 15.49$ degR

Computed errors in the estimated low-pressure turbine metal temperature for a rapid engine deceleration are shown in Fig. 4.16 using raw data with 10 processors and in Fig. 4.17 using smoothed data with 10 processors. In both cases, the error in the estimated metal temperature for parallel operation on 10 processors (corresponding to a 90% or order of magnitude reduction in computation time) is negligible. Figures 4.18 and 4.19 show the errors in the estimated metal temperature for a rapid engine deceleration using 50 processors. In both cases, the error in the estimated metal temperature for parallel operation on 50 processors (corresponding to a 98% reduction in computation time) is small. Figures 4.20 and 4.21 show the errors in the estimated metal temperature for a rapid engine deceleration using 100 processors. In both cases, the error in the estimated metal temperature for parallel operation on 100 processors (corresponding to a reduction in computation time of 99% or two orders of magnitude) is relatively small (approximately 5% of the maximum difference between the metal and gas temperatures).

Errors in the estimated metal temperature for a rapid engine acceleration and deceleration are shown in Fig. 4.22 using raw data with 10 processors and in Fig. 4.23 using smoothed data with 10 processors. In both cases, the error in the estimated metal temperature for parallel operation on 10 processors (corresponding to a 90% or order of magnitude reduction in computation time) is negligible. Figures 4.24 and 4.25 show the errors in the estimated metal temperature for a rapid engine acceleration and deceleration using 50 processors. In both cases, the error in the estimated metal temperature for

parallel operation on 50 processors (corresponding to a 98% reduction in computation time) is small. Figures 4.26 and 4.27 show the errors in the estimated metal temperature for a rapid engine acceleration and deceleration using 100 processors. In both cases, the maximum error in the estimated metal temperature for parallel operation on 100 processors (corresponding to a reduction in computation time of 99% or two orders of magnitude) is fairly significant and is on the order of 10 percent of the maximum difference between the gas and metal temperatures. Errors in the estimated metal temperature for processing with 50 or more processors can be reduced using a non-linear estimate of the gas temperature over the period of interest or by re-computing the metal temperature using the previous estimate of the gas temperature. Re-computing the metal temperature would, however, increase the computation time.

At this point the author introduces non-dimensional Malloy number defined as follows:

$$MALLOY(Ma) = \frac{ERROR\ \ IN\ \ ESTIMATED\ \ METAL\ \ TEMPERATURE}{MAXIMUM\ \ DIFFERENCE\ \ BETWEEN\ \ THE\ \ GAS\ \ AND\ \ METAL\ \ TEMPERATURES}$$

The significance of the MALLOY number is that it characterizes the theoretical accuracy of an explicitly parallelized lumped parameter analysis. If Ma $\leq$ 0.1the error associated with the lumped parameter analysis is small. The error in the estimated metal temperature could be reduced substantially if the time-dependent nature of the gas

temperature was modeled as a function of one or more measured parameters instead of the linear relationship assumed.

The effects of volume dynamics on the engine cycle are described by the mass, momentum, and energy equations. For time-dependent engine operation, the mass continuity equation has been found to be the most influential of the three equations and was considered in the development of the model-based algorithms. The rate of change of pressure of a volume is related to the difference between the flows entering and leaving the volume. This relationship is represented by:

$$\left(\frac{dP}{dt}\right) = \frac{\gamma RT}{V}(\dot{m}_{in} - \dot{m}_{out})$$

where

$\frac{dP}{dt}$ is the rate of change of pressure

γ is the ratio of specific heats

R is the gas constant

T is the gas temperature

V is the volume

$\dot{m}_{in}$ is the entering gas flow rate

$\dot{m}_{out}$ is the exiting gas flow rate

The error due to the volume dynamics for the explicitly parallelized model-based algorithms is not directly related to the elapsed time between data samples, Δt. It is, however, a weak function of the data sampling rate and the data filtering (smoothing) algorithms. Some attenuation of variation in measured pressures is expected as a result of the data filtering (smoothing) process.

Step 3 - Real-Time Model Calibration

The real-time model calibration approach focuses on estimating changes in engine component flows and efficiencies, which provide the foundation for real-time fault identification. Changes in engine component flows and efficiencies are calculated 20 times per second during steady-state engine operation and 100 times per second during transient engine operation.

The process of choosing an optimal set of test measurements and model calibration factors requires, to a large extent, evaluation of all possible combinations of test measurements and model calibration scale factors. Model-based simulation studies were conducted to choose appropriate test measurements and model calibration scale factors for identification of data and engine component faults. The selected set of test measurements and model calibration scale factors reflects extensive studies conducted using actual engine test data acquired during developmental engine tests. Typical test measurements used for real-time calibration of an augmented turbofan engine were summarized in Chapter 3 and are shown schematically in Fig. 4.4. Typical calibration

scale factors which are used in the fault detection and diagnostic process for a turbofan engine model were summarized in Chapter 3 and include fan and compressor map flow and efficiency scalars and high- and low-pressure turbine map efficiency scalars. The component level model provides an accurate prediction of measured and unmeasured parameters that reflect the current status of the test article and include the effects of off-schedule variable geometry, sensor lags, profile errors, heat transfer, and volume dynamics.

5. PARALLEL PROCESSING

In the discussion that follows, the concepts of load balancing and engine cycle balancing are discussed. The basic concept underlying engine cycle balancing (described in Chapter 4) is that initial guesses are made for the engine cycle model, the engine model then uses these initial guesses to calculate the cycle parameters and error terms component by component (for the engine inlet, fan, compressor, burner, high-pressure turbine, low-pressure turbine, bypass duct, mixer, augmentor, and nozzle components described in Chapter 3); the iteration logic calculates derivatives using the error terms, makes new guesses, and returns to the engine inlet and continues iterating. The iteration continues until convergence or until the maximum allowed iterations has been reached. Load balancing, on the other hand, is defined as the mapping (or re-mapping) of work to processors, with the intent of improving the efficiency of a concurrent computation.

As described in Chapter 4, the software was parallelized using a replicated worker approach. Different time samples are sent to different processors for engine cycle balancing and calculation of engine processes. The scheme uses one central processor to collect engineering unit data from the network, filter the raw data, distribute the data between the processors, collect and order the results, identify component performance changes and measurement errors from the model data, and run the Graphical User Interface (GUI). Special care was taken to correlate time-dependent processes.

Initially, a developmental computing platform consisting of a cluster of Pentium®-based PC's was designed and utilized for parallel processing. Most recently, high performance shared memory parallel computing platforms have been investigated for parallel processing. These two systems are called the Development Platform and Target Platform, respectively.

Software/Hardware Engineering Considerations

Development of the fault identification algorithms began in 1993. At that time, a network of distributed PC processors was chosen as the initial development platform. Engineering considerations relative to the use and parallelization of FORTRAN software are provided in Appendix M, which also includes an overview of a parallelization toolkit developed to aid in explicit software parallelization.

Development Platform

The data validation process with an embedded engine model requires a significant amount of computing power to execute in a real-time test environment. A network of distributed processors was chosen initially (1993) as a development platform and if

required, the development platform can also provide the computing power required for real-time data processing. A prototype platform built and designed by Dr. Csaba Biegl from Vanderbilt University was used to demonstrate the effectiveness of a distributed processor approach in providing the required computing power and as an environment for program development. The prototype platform (parallel machine) consisted of eight 150-MHz Pentium®-based computers interconnected via a dedicated high-speed (100 Mbits/sec) Ethernet network in a cluster configuration. The Pentium® cluster is scaleable since processors may be added or removed as computing requirements vary. As shown in Fig. 5.1, the high-speed ports can also be connected in a ring network (without using a hub) to build an eight-node hypercube.

Each node (or processor) in the cluster, has four network ports: one of these is a standard 10 base-2 (coaxial) port; the other three are high-speed 100 base-T (twisted pair) ports. The coaxial ports are used for "standard" network traffic (remote login, remote shell, network file system, and parallel applications with low communication bandwidth requirements) while the high-speed ports are used for application-level communications. The high-speed ports are connected in a ring network (without using a hub) to build the eight-node cluster. Each processor runs a copy of either the Linux operating system or Windows NT® in a multiboot configuration. This allows users to develop and test parallel code under their preferred operating environment. The parallel machine also includes system management tools (a small collection of shell scripts added to the standard Linux distribution) to facilitate control of the distributed processors. The

Message Passing Interface, MPI, (MPI Standard, 1994) has also been ported to the Linux and Windows NT® environments to provide application-level communication services over the high-speed ports. Special user-level network device drivers were developed for MPI to control the high-speed Ethernet ports without the need to use operating system calls. Other parallel programming environments available on the system include PVM (Parallel Virtual Machine) and a simple TCP/IP-based communication library. The scalable software paradigm and message-passing interface for the development and target platforms, described in the next section, are described in Appendix N.

Target Platforms

As long as the time required to compute the instantaneous aerothermodynamic calculations (i.e., conservation equations, engine dynamics, gas properties, numerical iterations, and cycle-balancing) is significantly greater than the time required to synchronize and exchange data between the processors, a distributed computing platform can significantly reduce computation time. However, if the time required to compute the aerothermodynamic calculations is significantly reduced (either through the development of improved cycle-balancing techniques or through the use of faster processors), machines that can transfer and synchronize data more efficiently offer the greatest reductions in computation time.

Many types of high performance parallel computing platforms are available as target platforms, each with different operating characteristics. Two computational speed tuning parameters (referred to as NUMGROUPING and NUMPENDING), which are used to accommodate different speed processors and to attempt to improve the performance of the Pentium-based development platform (to a level of efficiency comparable to a shared memory machine), are presented in the section entitled **Main and Worker Processes**. Three high performance computing platforms are currently available to process AEDC test data, two Hewlett-Packard HP-SPP2000®'s referred to as "infinity" and "beyond" and a Silicon Graphics SGI Origin 2000® referred to as "montecarlo." An assessment of the scalable performance improvement for the HP and SGI platforms is presented later in this Chapter. All three high performance computing platforms use a Fiber Distributed-Data Interface (FDDI). FDDI is a standard for data transmission on fiber optic lines in a local area network that can extend in range to 100 miles or more. The FDDI protocol is based on the token ring protocol. In addition to being large geographically, an FDDI local area network can support thousands of users. An FDDI network contains two token rings, one for possible backup in case the primary ring fails. The primary ring offers up to 100 Mbps capacity. If the secondary ring is not needed for backup, it can also carry data, extending capacity to 200 Mbps. The main performance features of the three target high performance computing platforms are summarized below:

HP-SPP 2000 (Beyond)

- 64 CPUs
- 46 MegaFlops
- 16 GB Memory
- 216 GB Hard disk
- FDDI connectivity

HP-SPP 2000 (Infinity)

- 48 CPUs
- 35 MegaFlops
- 11 GB Memory
- 162 GB Hard disk
- FDDI connectivity

SGI Origin2000 (Montecarlo)

- 64 CPUs
- 25 MegaFlops
- 32 GB Memory
- 172 GB Hard disk
- FDDI connectivity

In order to effectively employ high performance computing systems in a real-time computing environment, data must be transmitted between the test cell and the high performance computing platform. AEDC data acquisition systems use reflective memory to distribute data among UNIX and NT systems, which then distribute the data in a client server fashion for subsequent display and processing. The reflected memory region is a shared data region and the operating system manages it in that manner. Reflective memory provides three major benefits to real-time applications. First, it solves a problem that causes non-deterministic operation due to random blocking of program execution caused by memory access contention. There is no overhead for memory access with reflective memory because writes to memory are automatically distributed to other connected systems by hardware. Memory reads are executed on the local memory, therefore, eliminating any memory access contention delays. Second, elimination of contention for a shared resource gives determinism (predictable execution speed) to all application software. And finally, data reflection is automatic, concurrent and passive, requiring zero additional application software cycles to update all systems that share the common data. Currently, AEDC uses A UNIX based data distribution server to send data over a Fast Ethernet link to a high performance computing platform. This link uses existing fiber optic cable and inexpensive Fast Ethernet interfaces that are now standard on most systems.

Explicit Algorithmic Parallelization

Appendix M summarized two successful approaches for parallelizing existing FORTRAN software. An alternative method for parallelizing existing software is to analyze the algorithm being used and then develop a method to distribute the different functional blocks of the program explicitly to different processors. This also requires that explicit calls to an underlying communication library be made at the points where data synchronization and/or exchange are necessary. Communication libraries such as the Message Passing Interface, MPI (MPI Standard, 1994), Parallel Virtual Machine, PVM (PVM, 1994), and EXPRESS can be used for this purpose. The advantage of explicit parallelization is that frequently a much higher level of parallelization can be obtained than with instruction-level parallelizing compilers. The disadvantage is that explicit parallelization requires changes to the original program and conditional compilation to maintain serial and parallel versions of the software. Also, the algorithms must be analyzed (or reformulated) to find the best decomposition. The dependency analyzer components of some automatic tools such as the SAGE toolkit (Distribute pC++, 1993) and the FORGE family of compilers (FORGE, 1992) can simplify this process, however, use of global variables and common blocks complicates the analysis.

Main and Worker Processes

For parallel operation, different time samples are sent to different processors **(denoted as worker processors)** for cycle balancing and calculation of engine processes. The scheme uses one central processor **(denoted as main process)** to collect engineering unit data from the network, filter the raw data, distribute the data between the processors, collect and order the results, identify component performance changes and measurement errors from the model data, and run the Graphical User Interface (GUI). A sample packet contains approximately 200 instantaneous model inputs (model initialization variables, measured pressures, temperatures, speeds, etc.) and the corresponding return packet contains approximately 200 model outputs (component flows and efficiencies, predicted thrust, specific fuel consumption, etc.). Two "tuning parameters" (denoted as NUMGROUPING and NUMPENDING) are used to optimize parallel efficiency.

The user selectable "NUMGROUPING" variable controls how many sample packets (time samples) the main processor groups together when sending and receiving data. Increasing the NUMGROUPING variable **reduces** the number of messages passed while **increasing** the size of the messages a corresponding amount (e.g., NUMGROUPING = 2 reduces the number of messages passed by a factor of two and increases the size of the messages passed by a factor of two. Increasing NUMGROUPING reduces overall communication time but generally results in additional

execution time for the worker process (due to an increase in the number of model iterations required for cycle-balancing).

The user selectable "NUMPENDING" variable controls how many sample packets the main processor sends out. As the NUMPENDING variable is increased, additional sample packets are sent to the worker processes before they have completed processing the current sample to reduce idle time. By increasing the NUMPENDING variable, variations in the worker execution time (which is primarily a function of the number of model iterations required for cycle-balancing) can be accommodated. Increasing NUMPENDING reduces idle time but can increase execution time for the worker process (due to an increase in the number of model iterations required for cycle-balancing). The maximum difference between the number of sent and received samples is NUMPENDING * NWORKERS (where NWORKERS is the number of worker processors). The ready process queue initialization uses "NUMPENDING" to preload a certain number of "index tokens" for each process.

High-level "pseudo-code" for worker and main processes is provided in the following sections. The communication software below was developed by Dr. Csaba Biegl from Vanderbilt University under contract to AEDC.

Worker: High-Level "Pseudo-Code"

```
while data packet received do
  case type of data packet
    new datapoint:
      reset model from initialization file;
    solve engine model:
      solve model for all time samples in data packet;
      send back results to main processor;
    exit process:
      exit;
  end case
end do
```

Main Process: High-Level "Pseudo-Code"

```
data
  queue of ready processes          ! stores indices of ready processes
  queue of data samples             ! stores both input samples and solution data
(time ordered)

clear queues;
for i = 1,nworkers
 send new datapoint command to worker i;
end for
for i=1, maxqueuelen
  for j = 1,nworkers
    enqueue worker j in ready process queue;    ! enqueue: append to end of queue
  end for
end for
```

```
while not empty(data queue) or not eof(input file) do
  didsomething = FALSE;
  while pending messages from workers do
    n = next ready worker;
    read results from worker n;
    merge results with input data for the matching time sample stored in the data queue;
    while data is complete at the head of the data queue do
      process the completed time sample;                    ! diagnosis, GUI
      free data queue slot;
    end while
    enqueue worker n in the ready process queue;
    didsomething = TRUE;
  end while
  while not empty(ready queue) and not full(data queue) and not eof(input file) do
    read next input sample;
    n = next ready process dequeued from ready queue;
                          ! dequeue: remove first item from the head of the queue
    send input data to process n;
    store input data samples in data queue;
    didsomething = TRUE;
  end while
```

if not didsomething **then**

wait for data from process which processes the sample at the head of the data queue

end if

end while

Time Analysis

The replicated worker approach developed in the present work requires passing of the same number and size of messages as the number of processors increases (from two, to three, to four, ... , to the total number of processors. As a result, if the main processor can communicate as efficiently with one processor as it can with NP-1 processors, the communication time will remain constant as the number of processors increases. The parallel execution time is governed by four quantities: (1) the serial execution time required to collect engineering unit data from the network, filter the raw data, identify component performance changes and measurement errors from the model data, and run the (GUI), (2) the serial execution time required for cycle balancing and calculation of engine processes, (3) the efficiency of the parallel computations for cycle balancing and calculation of engine processes, and (4) the parallel communication time.

As previously discussed, two "tuning parameters" (denoted as NUMGROUPING and NUMPENDING) are used to optimize parallel efficiency. The time analysis that follows is based on the following observations.

- Total serial execution time is a function of the test maneuver performed and can vary by a factor of three; rapid engine transients (rapid engine decelerations, accelerations, or bodies) typically require the greatest execution time.

- The serial execution time required to collect engineering unit data from the network, filter the raw data, identify component performance changes and measurement errors from the model data, and run the (GUI) is on the order of ten to fifteen percent of the serial execution time during rapid engine transients (this means that the replicated worker approach developed in the present research can reduce execution time by no more than eighty-five to ninety percent for computationally intensive maneuvers). For all other maneuvers where the serial execution time is much lower than for rapid transients, the execution time required to execute the GUI and related processes can be as much as thirty to fifty-five percent of the total serial execution time (this means that the replicated worker approach developed in the present research will reduce execution time by no more than a factor of two or three for the less computationally intensive maneuvers).

In summary, the replicated worker approach developed in the present research, will be much more effective for computationally intensive maneuvers. If, for example, execution time is reduced by a factor of nine for computationally intensive maneuvers and a factor of three for less computationally intensive maneuvers, the maneuvers can be processed in roughly the same time. Based on the stated observations, additional speed increases with the replicated worker approach are possible only if a more efficient GUI is used or if one or more of the main processes (collecting engineering unit data from the network, filtering the raw data, identifying component performance changes and measurement errors from the model data, and running the GUI) are performed in parallel.

The following notation is used for the time analysis:

T_{serial} = serial execution time

$T_{parallel}$ = parallel execution time

$T_{parallel}^{Communication}$ = parallel communication time

T_{GUI} = execution time for calculation of GUI processes (collecting engineering unit data from the network, filtering the raw data, identifying component performance changes and measurement errors from the model data, and running the GUI)

$$\gamma = \frac{T_{GUI}}{T_{serial}}$$

$$\beta = \frac{T_{parallel}^{Communication}}{T_{serial}}$$

NP = number of processors

μ = worker efficiency (ratio of the time required for cycle balancing and calculation of engine processes in serial and parallel). For the time study performed in the present research, the number of model iterations required for cycle balancing was seen to increase by a factor of approximately 1.3 for 3 worker processors, 1.5 for 7 worker processors, and 1.6 for 11 worker processors.

Computational efficiency is characterized as the ratio of serial to parallel execution time as follows:

$$SPEEDUP = \frac{T_{serial}}{T_{parallel}} .$$

Parallel execution time is the larger of two quantities: (1) the total communication time of the main processor plus the time required by the main processor to perform the GUI processes or (2) the total communication time of an individual worker process plus the time required for an individual worker process to perform its cycle balancing and engine processes. For conciseness, the communication time for initial memory paging

(which can be significant for the Pentium system) and communication of initial values to the processors is ignored. It should also be recognized that communication times are expected to vary significantly for blocking and non-blocking architectures, where the communicating process may perform other operations before the communication has completed.

For the first case, the parallel computation time is

$$T_{parallel} = T_{parallel}^{Communication} + T_{GUI}.$$

In general, the parallel communication time of an individual worker process will be a fraction of the total communication time of the main process. For the second case, using a conservative estimate of the worker communication time, the parallel computation time is

$$T_{parallel} = T_{parallel}^{Communication} + \frac{T_{serial} - T_{GUI}}{\mu(NP-1)}.$$

For the second case with

$$\gamma = \frac{T_{GUI}}{T_{serial}}$$

and

$$\beta = \frac{T_{parallel}^{Communication}}{T_{serial}}$$

and dividing the serial execution time by the parallel execution time

$$\frac{T_{serial}}{T^{parallel}} = \frac{T_{serial}}{T_{parallel}^{Communication} + \frac{T_{serial} - T_{GUI}}{\mu(NP-1)}}.$$

The speed up is

$$Speedup = \frac{1}{\beta + \frac{1-\gamma}{\mu(NP-1)}}.$$

The two "tuning parameters" (previously referred to as NUMGROUPING and NUMPENDING) reduce parallel execution time by reducing β, the ratio of parallel communication time to serial execution time and minimizing the effect of variations in worker efficiency μ, during parallel operation.

Sample results from a time analysis using a HP-SPP 2000 for analysis of a rapid engine deceleration with a γ of approximately 0.15 for 1, 4, 8, and 12 processors follow. For the simplified study below, β is assumed to be zero. For execution with 4 and 8 processors, the worker communication time plus the time required for worker cycle balancing and calculation of engine processes is greater than the total communication time plus the time required to perform the GUI processes. For execution with 12 processors, the total communication time plus the time required to perform the GUI processes is greater than the worker communication time plus the time required for worker cycle balancing and calculation of engine processes.

- Number of Processors: 1
  - Execution Time: 71.81 Seconds

- Number of Processors: 4
  - Measured Execution Time: 26.49 Seconds
  - Measured *SPEEDUP* : 2.71
  - Measured Worker Efficiency: 0.75
  - Estimated *SPEEDUP* : 2.64

- Number of Processors: 8
  - Measured Execution Time: 12.44 Seconds
  - Measured *SPEEDUP* : 5.77
  - Measured Worker Efficiency: 0.66
  - Estimated *SPEEDUP* : 5.43

- Number of Processors: 12
  - Measured Execution Time: 10.90 Seconds
  - Measured *SPEEDUP* : 6.59
  - Measured Worker Efficiency: 0.62
  - Estimated *SPEEDUP* : 6.67

Scalable Performance Improvement

The initial version of the model-based fault detection algorithms executed at about 20 samples/sec on a single 150- MHz Pentium platform, whereas 100 samples/sec are required for real-time operation. Attempts to run the software on high-performance single-processor workstations available at that time did not result in real-time operation.

The effectiveness of the Pentium platform and parallelized software was evaluated using an 89-sec test maneuver. Figure 5.2 shows the execution time as a function of the number of worker processors employed. These initial results supported the perceived need to decrease computation time using parallel processing techniques. The results convincingly showed that an eight-processor Pentium system was fully capable of executing the parallelized model-based data validation and fault detection algorithms in real time.

The time required to compute the instantaneous aerothermodynamic calculations can vary by an order of magnitude, or more, depending on the engine cycle, the available test measurements, test measurement uncertainty, and the type of test maneuver performed. As the number of processors is increased (or processor performance increases), faster than real-time processing of test data may be possible for situations where the data must be processed off-line after completion of the test (e.g. post-test processing of data recorded on-board the aircraft).

An evaluation of the scalable performance improvement for an application where the time required to compute the instantaneous aerothermodynamic calculations (i.e., conservation equations, engine dynamics, gas properties, numerical iterations, and cycle-balancing) is relatively small is shown in Fig. 5.3, for processing on an SGI Origin 2000 and in Fig. 5.4 for processing on an HP-SPP 2000. Although not shown in the Figures, a significant variation in the SPEEDUP is seen during processing on the SGI origin and on

the HP-SPP 2000 for cases with 12 processors. The non-deterministic operation, at present, appears to be due to random blocking of program execution caused by memory and processor access contention for the "shared" high-performance computing resources. The results shown in Figs. 5.3 and 5.4 are considered "worst case" and are generally non-repeatable.

Techniques used to increase the effectiveness of the Pentium platform are described below along with techniques which will be evaluated in subsequent research to improve the scalable performance improvement on the shared memory machine.

As the time required to compute the aerothermodynamic calculations is reduced, a corresponding reduction in the effectiveness of the Pentium platform is seen. The efficiency of the Pentium platform is strongly dependent on the efficiency of data transfer and synchronization. While the overall throughput rates for the Pentium system are high, the number of messages passed between the processors can cause some real bottle-necks. Therefore, as previously discussed, multiple data samples were grouped together for processing on the Pentium platform by increasing the NUMGROUPING parameter. In addition, upon completion of processing, output from multiple data samples are also grouped together and sent back to the main processor. The scalable performance improvement is also effected by variations in the time required to compute the instantaneous aerothermodynamic calculations which can be partially offset by increasing the NUMPENDING parameter. This problem is compounded by the fact that, over time,

incremental upgrades have been made to the Pentium platform and, as a result, a wide variety of processors are interconnected. In order to account for differences in convergence time and processor speed, data samples are sent more frequently to the faster processors and to processors in which the aerothermodynamic calculations are performed faster. These two techniques were successfully used improve the scalable performance improvement of the Pentium platform.

On the shared memory machine, the degradation in scalable performance appears to be only slightly affected by the number of messages passed. For selected test maneuvers, increases in scalable performance improvement (over the results shown in Figs. 5.3 and 5.4) are possible by grouping of the data samples. At present, due to the non-repeatability of the timing results on the shared memory machines, the level of performance increase possible by grouping the data samples cannot be quantified.

A higher than anticipated variation in the number of iterations required to balance the engine model-cycle exists between serial and parallel modes of operation. This is due, in part, to degradation of the initial guesses of the model perturbation variables in the parallel processing mode. In order to address this degradation, a simple time-step integration technique, similar to the technique employed by French (1982) in his development of a compact real-time turbofan engine dynamic simulation, will be used to improve the initial guesses. Future research will address the scalable computing performance issues in detail for a variety of platforms and test maneuvers.

6. FAULT DETECTION AND DIAGNOSIS

To be effective, gas path analysis tools must identify component performance deviations and measurement errors. The model-based fault identification process consists of two main phases. The first phase of the fault identification process relies on a real-time model-based evaluation of test data to detect faults resulting from measurement errors, engine component events, or both. After a fault is detected, automated model simulation studies are performed in near-real time to determine the source of the failure (e.g. which sensor or actuator has failed) and to estimate the extent and probability of the failure. This detailed diagnostic information enables the analysis engineer to assess the relative probability of the fault and, in most cases, quickly identify and verify the actual cause of the fault. If necessary, additional test data may be acquired at previously tested stabilized engine operating conditions to increase the fidelity of the diagnosis.

Abrupt engine faults due to domestic object damage, foreign object damage, or control anomalies and measurement errors due to level shifts, noise, or poor response are detected using time dependent transient data. Slow faults however, due to component or overall engine degradation, and sensor drift, are best detected using more numerous and accurate steady-state data averaged over a 10-second time interval. Use of steady-state test data permits identification of relatively small changes in component or overall engine performance such as compressor (or fan) performance due to fouling from accumulation

of organic or non-organic deposits on the blade surfaces that cause an increase in surface roughness, or compressor degradation resulting from tip clearance changes.

The component-level model-based engine performance and fault identification technique is capable of:

1) identifying measurement errors (e.g., sensor bias, drift, level shift, noise, or poor response) in facility fuel flow, airflow, and thrust measurements;

2) identifying measurement errors in engine aerothermodynamic measurements (rotor speeds, gas path pressures and temperatures);

3) identifying measurement errors in engine control sensors (e.g., leaking/biased pressure sensor, slowly responding pressure measurement) and variable geometry rigging (e.g., misset guide vanes or nozzle area) that would invalidate a test or series of tests;

4) identifying abrupt faults (e.g., faults due to domestic object damage, foreign object damage, and control anomalies);

5) identifying slow faults (e.g., component or overall engine degradation, and sensor drift) and relatively small changes in compressor (or fan) performance; and

6) being easily extended to diagnose secondary failure modes and to verify any modeling assumptions that may arise for developmental engine tests (e.g., increase in turbine flow capacity, inaccurate measurement of facility bleed flows, horsepower extraction, etc.).

An overview of the sensor and engine degradation fault searches are provided below. Test data and results used to verify the model-based fault identification algorithms are presented in Chapter 7.

Fault Detection

The issue of detecting faults during developmental turbine engine testing with newly developed hardware at never-before-tested conditions is extremely complex. Therefore, the real-time model-based fault identification approach is used to supplement existing approaches that consider on-site calibrations, online monitoring of instrumentation systems, comparison of redundant measurements, comparison of steady and non-steady measurements, and comparison of measured and predicted responses [(Malloy, 1993) (Davis et al., 1996)].

The fault detection approach relies on a real-time interpretation of measured and predicted responses and interrelationships throughout the propulsion system quantified by

the CLM. A simultaneous multipoint analysis is used to provide a relative assessment of measurement error and changes in component performance. The multipoint analysis includes the following:

1. Interpretation of differences between predicted and measured aerothermodynamic measurements over the time being considered (to validate modeling assumptions and detect measurement errors);

2. Interpretation of changes in component flows and efficiencies using data immediately prior to and during the time being considered (to detect abrupt faults); and

3. Interpretation of changes in component flows and efficiencies using data obtained considerably before as well as during the time being considered (to detect slower faults such as engine degradation or sensor drift).

A weighted root-sum-square fault detection parameter (FDP) is calculated as a measure of the probability of a fault.

$$FDP = \begin{bmatrix} w_1(\delta_{WC,FAN})^2 & +w_2(\delta_{T/T,FAN})^2 & +w_3(\delta_{P/P,FAN})^2 \\ +w_4(\delta_{WC,HPC})^2 & +w_5(\delta_{T/T,HPC})^2 & +w_6(\delta_{P/P,HPC})^2 \\ +w_7(\delta_{DP16,SF})^2 & +w_8(T5_{meas} - T5_{observed})^2 & +w_9(P5_{meas} - P5_{observed})^2 \\ +w_{10}(FG_{meas} - FG_{observed})^2 & +w_{11}(SFC_{meas} - SFC_{observed})^2 & +\cdots \end{bmatrix}$$

The weighting factors (w_1, w_2, w_3, ...) associated with the individual parameter deviations can be determined from pretest uncertainty estimates or determined empirically from the test data. The fault probability is a function of time-dependent changes in component flows and efficiencies, the ability to predict test measurements not directly used in the real-time model calibration process, and changes in component flows and efficiencies relative to baseline values.

The sensitivity of the fault detection system to trigger fault warnings and initiate fault diagnositcs is varied according to the type of engine operation occurring in order to maximize the probability of detecting faults, and to minimize the probability of false alarms. Sensitivity is increased (i.e., FDP is numerically increased) during operation when measurement uncertainties are lower (e.g., stabilized operating conditions at higher power settings). Individual weighting factors can be adjusted to increase or decrease the fault detection sensitivity or, an overall multiplicative parameter can be used to increase or decrease the fault detection parameter. The overall adjustment to the FDP sensitivity

or decrease the fault detection parameter. The overall adjustment to the FDP sensitivity was established during development testing as a function of engine inlet pressure and distortion levels, engine power level, nozzle throat Mach number, rotor speed derivatives, and may also be a function of thrust stand dynamics, and other measurement system considerations.

Sensor Fault Search

Once a fault is detected, a methodical, automated model-based search is performed to isolate the most probable cause of the fault. The model-based search uses the engine cycle model to predict expected behavior with and without the fault occurring. From a comparison of measured and predicted engine parameters, it can be determined whether or not an actual fault did occur or whether one or more sensors failed. The probability of accurately diagnosing sensor faults is increased, as previously stated, by emulating the diagnostic process performed by an analysis engineer. Specifically, the automated model-based fault diagnosis approach concentrates first on identifying the probability of a measurement error. If the fault cannot be attributed to measurement error, a performance degradation search is performed to interpret the fault detection information, including flow and efficiency scale factors.

Once a fault is detected, the fault diagnostic system concentrates on identifying individual sensor faults and tries to identify the erroneous measurement and the

magnitude of the error. The calibrated CLM (calibrated with data immediately prior to detection of the fault) is used to assess the probability of measurement errors. Each measurement is sequentially perturbed, varying magnitudes to determine the most probable cause of the fault (e.g., fuel flow is perturbed ± 5 percent in 0.5-percent increments, then airflow is perturbed, etc.). The model-based diagnostic approach relies on interpretation of measured and predicted responses and interrelationships throughout the propulsion system quantified by the CLM. Due to the highly nonlinear interrelationships throughout the propulsion system, multiple perturbations of varying magnitudes are required to accurately evaluate fault probabilities. The fault identification process is fully automated and the ability to detect and accurately diagnose faults is no longer a function of the experience of the analysis engineer.

Performance Degradation Search

Rapid changes in component flows and efficiencies are diagnosed using time dependent transient data. As previously mentioned, slow faults due to component or overall engine degradation, and sensor drift are identified using steady-state test data averaged over a 10-second interval.

The fault error probability function evaluates differences between predicted and measured parameters and both rapid and slow changes in component flows and

efficiencies. Changes in component flows and efficiencies are the basis of the fault identification process. However, a more detailed performance degradation search is automatically performed using the component level model to identify component anomalies (e.g., misset guide vanes) and secondary failure modes which might include, for example, an increase in turbine flow capacity or a reduction in turbine cooling flow rates.

Techniques used to identify relatively small changes in compressor performance due, for example, to fouling are presented below. An enhanced map adjustment procedure to identify small performance changes is also outlined.

Data typical of compressor performance degradation due to fouling, in the absence of engine inlet pressure distortion, are shown in Figs. 6.1 – 6.10. Variations in compressor exit pressures shown in Figs. 6.1 and 6.2, for a 0.5% degradation in compressor efficiency, are on the order of +/- 0.4 percent and vary as a function of both radial and circumferential location. Variations in compressor exit temperatures shown in Figs. 6.3 and 6.4, for a 0.5% degradation in compressor efficiency, are on the order of +0.25 percent and with minimal variation as a function of radial or circumferential location. Fault detection and diagnostics is aided by proper interpretation of concurrent changes in low pressure turbine exit temperature, specific fuel consumption, and compressor airflow pumping capacity, as shown in Figs. 6.5 - 6.7. For a single-point analysis, differences between measured and observed low pressure turbine exit

temperature, engine gross thrust, and specific fuel consumption are negligible, verifying the validity of the data. However, for multi-point analysis, deviations between nominal and degraded values of compressor efficiency, compressor airflow pumping capacity, low pressure turbine exit temperature, engine thrust, and engine specific fuel consumption verify degraded engine operation and the change in engine performance is attributed to the change in compressor efficiency and airflow pumping capacity. Variations in compressor pressure ratio, and efficiency are shown in Figs. 6.8 and 6.9. Changes in the compressor efficiency map adjustment scalar for nominal and degraded operation are shown in Fig. 6.10.

Human-Computer Interface and Data Visualization System

The fault identification algorithms are written in the FORTRAN programming language and have been incorporated into a real-time graphical user interface. To function efficiently and to minimize demands on analysis engineers, the algorithms are automatically executed as data are acquired. If a fault is detected, the analysis engineer is notified of the event via a computer-generated voice message and the event features and diagnostics are visually displayed for review. All events are logged and permanently recorded on disk. An overview of the graphical user interface is provided in Appendix G and, for illustration purposes, the actual GUI displays available for online viewing by the analysis engineer are presented in the next Chapter.

7. IMPLEMENTATION AND VERIFICATION OF MODEL-BASED TECHNIQUES FOR FAULT IDENTIFICATION

The fault detection and diagnostic algorithms developed in the present investigation were evaluated using data from two distinctly different modern military turbofan engines undergoing ground test development at simulated altitude test conditions. Data from hundreds of sensors were recorded at rates up to 100 samples per second to characterize aerothermodynamic engine performance. Both engines were tested over a wide range of steady-state and transient engine operating conditions enabling a comprehensive assessment of the suitability of the approach for developmental engine testing. For proprietary and other reasons, the engines are not specifically identified and normalized data are presented.

The three-step model calibration process is an integral part of the fault identification process. Accurate predictions from the CLM are a key element of the fault detection and diagnostic processes. Figures 7.1 through 7.3 show a prediction of engine thrust compared to engine test data after model calibration for a slow engine acceleration. Figures 7.4 and 7.5 show a prediction of engine thrust compared to engine test data for a rapid engine deceleration and a rapid engine acceleration followed by a rapid engine deceleration. Predicted thrust provides an independent measure of the real-time model calibration capability, since measured thrust is not used during the calibration process.

The close agreement between measured thrust and predicted thrust demonstrates the accuracy provided by the model calibration capability. Figures 7.6 through 7.10 show a prediction of LP turbine exit temperature compared to engine test data for the same series of engine maneuvers. The close agreement between measured and predicted LP turbine exit temperatures demonstrates the accuracy of the calibrated model. Figures 7.11 through 7.15 show, for the same series of data points, close agreement between measured and predicted LP turbine exit pressure and clearly illustrate the accuracy of the calibrated model.

Representative LP turbine exit temperatures and the average value are shown in Fig. 7.16 to illustrate the magnitude of the transient temperature profile at the LP turbine exit. Figure 7.17 shows a comparison of the raw LP turbine exit temperature and the LP turbine exit temperature after compensation for thermocouple time response.

Propulsion control schedules and engine hardware often vary significantly from basic design conditions during developmental engine testing. The situation is compounded by the use of specific propulsion control laws and schedules developed during engine tests to tune overall system performance to match individual mission roles. The capability to identify measurement errors and anomalous changes in component performance during control logic development is a formidable task because changes to the control laws may result in atypical, but valid, engine operation. False alarms and incorrect diagnoses will result unless the control law changes are reflected in the fault

identification analyses. For example, potentially anomalous engine behavior or anomalous measurement of engine airflow is indicated in Fig. 7.18 by an erratic, rather than uniform, increase in engine airflow for a moderate engine acceleration. However, the erratic change in airflow is a result of control law manipulation and represents valid engine operation. The real-time portion of the calibration process, discussed previously, inherently accounts for the control law changes and results in a confirmation of valid engine operation for the data shown in Fig. 7.18. The fault identification process correctly reported no faults, thus demonstrating its suitability for developmental engine testing.

Sensor faults identified during transient engine operation using the model-based approach include identification of a slowly responding pressure measurement used in the determination of engine inlet airflow. The fault was automatically detected and diagnosed as an engine airflow measurement anomaly using the automated fault identification technique. Variations between the faulty and normal airflow indications are shown in Fig. 7.19. The differences are evident near the end of the deceleration and, to a lesser extent, in the steady portion of the data immediately prior to and following the deceleration. Further investigation by the author revealed that the anomalous airflow measurement was due to a crimped pressure line. The diagnosis demonstrated the applicability of the fault detection technique to transient engine operation by identifying a faulty time response that is not detectable, by any method, during steady-state engine operation. One of the goals of the fault identification technique is to identify errors in

engine control sensors such as a leaking, biased, or slowly responding control pressure that would invalidate a test or a series of tests. Since errors in control sensors occur rather infrequently, no actual test data was available to verify the ability to detect, for example, an error in the control LP turbine exit pressure. However, the ability of the fault detection technique to distinguish between normal and faulty LP turbine exit pressures is clearly seen in Fig. 7.20 in the response of the predicted LP turbine exit pressure and the "steady-state" measurement.

Data anomalies are equally likely to occur during steady-state and transient engine operation. The wide range of engine operating conditions that were encountered enabled detection and diagnosis of a number of sensor faults while demonstrating a low occurrence of false alarms. The fault identification approach is most sensitive during steady-state engine operation at high power settings and is capable of detecting very small (less than 1 percent) errors. Common data anomalies that have been identified using the model-based approach include identification of sensor biases, drifts, level shifts, and excessive noise.

An important function of the fault identification capability is its ability to distinguish between measurement anomalies and engine anomalies. Data for a steady-state engine operating condition were chosen for clarity to illustrate the capability to distinguish between measurement and engine anomalies. For steady-state engine operation, data in Fig. 7.21 seem to indicate the occurrence of an engine anomaly implied

by an abrupt reduction in compressor efficiency and a corresponding reduction in flow capacity. The effects of the observed changes in compressor performance were also seen in other parameters. Superficial inspection of the data leads to a conclusion that a compressor anomaly exists.

However, the fault identification process diagnoses the effects of a faulty P_3 measurement on related cycle parameters, correctly identifies the P_3 measurement anomaly, and distinguishes it from an apparent engine anomaly. The high probability of a P_3 anomaly, as determined by the fault identification process, is shown relative to the probability of other measurements in Fig. 7.22. In contrast, if the probability of a P_3 error had also been low, a compressor anomaly would have been diagnosed. Independent analysis using traditional methods confirmed the anomalous P_3 diagnosis.

Important to note in the anomalous P_3 diagnosis is that the fault was detected early in the event (Fig. 7.23), clearly exceeding the fault detection threshold. Although the P_3 measurement eventually shifted to 95 percent of its true value (i.e., 5-percent error), the fault was identified while the error was on the order of one percent. The P_3 fault, which was indicative of other faults that were identified, demonstrated the capability of the model-based approach to successfully detect and diagnose sensor anomalies as they occur and distinguish these anomalies from variations in component and overall engine performance.

In order to further demonstrate the capability of the model-based algorithms to correctly identify faults during developmental engine tests, seeded fault testing is performed. Seeded fault testing consists of seeding measurement faults (e.g., measurement noise, level shifts, drifts, biases, etc) into otherwise valid test data. Representative results from seeded fault testing are shown in Figs. 7.24 through 7.30 for steady-state and transient engine operation.

Experience indicates that as more and more data are analyzed, the probability of identifying faults increases. For example, the anomalous airflow measurement due to a crimped pressure line was undetectable during steady-state operation but was easily detectable during transient operation. The probability of detecting relatively small errors, such as measurement drifts, is increased if data are acquired as the drift occurs. If numerous errors occur in the data stream simultaneously, the probability of detecting a fault increases dramatically while the probability of correctly diagnosing the faults decreases. In the case of complete failure of the data system, a fault will be detected and the diagnostic process will be inconclusive. Data shown in Figs. 7.24 through 7.30 clearly show the capability of the diagnostic algorithms to identify the failed measurement and provide an estimate of the bias. Figure 7.31 illustrates the capability of the model-based algorithms to detect and diagnose multiple seeded data faults. In cases where one measurement error is significantly greater than the others, its probability will be highest and, the invalid measurement must be removed from the data stream before the smaller errors can be detected and diagnosed.

8. SUMMARY AND CONCLUSIONS

A new capability for real-time fault identification during developmental turbine engine testing has been developed and demonstrated during developmental turbine engine testing at simulated altitude conditions to assess the viability of the approach both in terms of the functionality of the model-based approach and the required computational speed. The model-based fault identification approach successfully detected and identified anomalous measurements and component performance variations during both steady-state and transient engine operation. A Component-Level Model (CLM) provides the basis for the fault identification process and an automated, multi-step, model calibration process ensures adequate model fidelity.

The CLM-based engine performance and fault identification method developed in the present work brings together features which individually and collectively advance the state-of-the-art. These features are separated into three categories: (1) advancements to effectively quantify off-nominal behavior, (2) advancements to provide a fault detection capability that is practical from the viewpoint of the analysis, implementation, tuning, and design, and (3) advancements to provide a real-time fault detection capability that is reliable and efficient.

The following features of the present method have been found effective to quantify off-nominal engine behavior:

- insensitivity of the engine model to *a priori* assumptions and engine-model mismatch (typical of developmental engine tests);

- insensitivity of the engine model to measurement bias, profile error, and Reynolds number index;

- sensitivity of the engine model to measured disturbances and relatively small changes in component performance;

- incorporation or inclusion of physically meaningful model updates (e.g., multipliers or adders on component flows and efficiencies); and

- identification of small performance changes by incorporating an enhanced component performance map adjustment procedure.

The following features of the present method provide a fault identification capability that is practical from the viewpoint of the analysis, implementation, tuning, and design. These features include:

- ease of implementation, tuning, and design;

- methodical approach to analyze available test measurements and uncertainties and component updates to simultaneously detect and diagnose data anomalies and component faults;

- minimization of errors due to simplifications that would otherwise be encountered with a reduced-order linear dynamic engine model;

- analysis of the physics of the system using well-known modeling techniques derived from the fundamental governing equations;

- near-real time determination of the most probable set of measurement errors and component deviations and an estimate of the relative probability of measurement errors or component deviations;

- performance trending.

The following additional features provide a real-time fault identification capability that is reliable and efficient:

- an order of magnitude increase in the model execution speed, as well as an order of magnitude reduction in the settling time (time to convergence), compared to existing techniques;

- real-time fault detection and near-real time fault diagnostics to identify performance anomalies during test maneuvers, giving test engineers the capability to avoid unwanted stalls/surges/collateral damage;

- efficient solution of the governing equations;

- reliable nonlinear model convergence;

- efficient parallel processing algorithms;

- real-time fault detection;

- near-real time fault diagnostics.

In addition, the CLM-based engine performance and fault identification technique satisfies all the design criteria. The real-time fault identification techniques is capable of:

- identifying measurement errors on the order of 0.5 percent (e.g., sensor bias, drift, level shift, noise, or poor response) in facility fuel flow, airflow, and thrust measurements;

- identifying measurement errors in engine aerothermodynamic measurements (rotor speeds, gas path pressures and temperatures);

- identifying measurement errors in engine control sensors (e.g., leaking/biased pressure sensor, slowly responding pressure measurement) and variable geometry rigging (e.g., misset guide vanes or nozzle area) that would invalidate a test or series of tests;

- identifying abrupt faults (e.g., domestic object damage, foreign object damage, and control anomalies);

- identifying slow faults (e.g., component or overall engine degradation, and sensor drift). Specifically, the technique is capable of identifying small changes in compressor (or fan) performance on the order of 0.5 percent; and

- being easily extended to diagnose secondary failure modes and to verify any modeling assumptions that may arise for developmental engine tests (e.g., increase in

turbine flow capacity, inaccurate measurement of facility bleed flows, horsepower extraction, etc.).

Specific efforts completed in the present research to advance the state-of-the art and to provide a CLM-based engine performance and fault identification method to address shortfalls in turbine engine transient engine modeling capabilities, system identification modeling capabilities, and limitations to current modeling and diagnostic capabilities include the following:

- A model-based fault identification technique was developed to take advantage of the redundant information provided by unlike sensors and provide the capability to detect failures in singly redundant sensors and 'soft' failures.

- Real-time modeling methodologies for gas turbine engine performance were reviewed as were the characteristics of the models, the various algorithms used in the models, and system integration issues. Improved modeling techniques developed in this research were combined with efficient parallel processing to eliminate the typical constraints on consistency, size, and computational speed and maximize the potential of the aerothermodynamic model for real-time fault identification.

- A number of "observer models" and methods for computing the component deviation parameters were evaluated including single-point and multi-point identification methods, Kalman filtering techniques, "minimum-right-inverse" techniques, and "square" identification methods where the number of knowns equals the number of unknowns. The square method employed in the present research has the advantage of being very sensitive to measured disturbances, making it ideal for analyzing small performance deviations. Improved techniques developed in this research permit identification of changes in component characteristics on the order of one half of one percent.

- A detailed analysis of steady-state and transient test measurement uncertainties was performed for test measurements used in the fault identification process. Pressure, temperature, speed, variable geometry, structural clearance, fuel flow, airflow, bleed extraction, auxiliary power extraction, and thrust measurement uncertainties were all evaluated.

- Appropriate test measurements and uncertainties, and component updates and sensor biases were identified to simultaneously detect and diagnose data anomalies and component faults in real-time. The actual process of selecting the sensor and cycle model updates was based on engineering analysis and an exhaustive computer-based search using a performance index to characterize how well measurement errors and changes in component performance can be

characterized in addition to how well key performance parameters such as thrust can be predicted in the presence of assumed measurement uncertainties and modeling errors. Initial studies were performed using a Truth Model with embedded modeling errors and sensor uncertainties to determine the most likely set of sensor/model updates. Follow-on studies and analysis were performed with actual test data, for nominal and degraded engine operation, and "seeded" measurement errors to determine the final set of sensor/model updates.

- Design studies were completed to minimize tradeoffs between noise sensitivity and response, and develop an algorithm capable of minimizing false alarms and missed detections while also estimating the probability of a fault.

- A simple digital filtering technique was developed for use with the fault identification algorithms to minimize the effects of normal measurement noise. A simple unweighted average of observations immediately prior to and after the current observation provides for a settling time on the order of 100 milliseconds for fault detection and yields an unbiased estimate of the true process during steady-state operation.

- A modified least-square derivative technique was developed as the average of the forward- and backward-least-square derivatives to more accurately determine time derivatives from test data.

- A methodology was developed by which various parameters and their effects on component and overall engine performance and the fault identification process can be evaluated. The effect of "known", "unknown", and "assumed parameters" was evaluated for each engine component including the engine inlet/isolation seal, fan, high-pressure compressor, burner, high-pressure turbine, low-pressure turbine, mixer, augmentor, and nozzle.

- A real-time fault identification technique was developed in which no special test instrumentation is required for fault identification during developmental turbine engine tests. Moreover, the three-step engine model calibration process developed in the present research can be performed in conjunction with normally scheduled test activities.

- A transient model calibration process was developed which permits effective estimation of heat transfer characteristics and thermocouple time constants from transient test data. In addition, a non-dimensional number was developed to characterize the theoretical accuracy of an explicitly parallelized lumped parameter analysis.

- A detailed assessment of the unknown parameters for the fault identification process include operating variables (operating points on the component maps, inlet airflow rate, bypass ratio, etc.), sensor biases, profile errors, component performance variations from nominal values (deviations in flows and efficiencies), time constants, sensor lags, pressure losses, etc. was completed. The engine cycle equations solved include the compatibility equations (airflow rate continuity, shaft power balances, and, for mixed exhaust turbofan cycles, a static pressure balance for the fan and core flows at the mixer entrance). The corresponding perturbation variables and error terms for the real-time fault identification model were also identified.

- A number of parallel computing processes were evaluated. Parallel computing processes shown to greatly enhance the real-time computing capability include simultaneous calibration of component maps, calculation of engine processes (cycle balancing) and integration of dynamic quantities (rotor and gas dynamics) for time-dependent data. In contrast, cycle model segmentation and distribution algorithms developed in conjunction with the "Parallel Processor Engine Model Program" failed to greatly enhance the real-time capability of the parallel technique and tended to increase the effect of numerical errors and their cumulative effects over time.

- The fault identification algorithms were parallelized using a using a replicated worker approach in which different time samples are sent to different processors for cycle balancing and calculation of engine processes. The scheme uses one central processor to collect engineering unit data from the network, filter the raw data, distribute the data between the processors, collect and order the results, identify component performance changes and measurement errors from the model data, and run the Graphical User Interface (GUI).

- The parallel algorithms were developed on a parallel machine consisting of eight 150-MHz Pentium®-based computers interconnected via a dedicated high-speed (100 Mbits/sec) Ethernet network in a cluster configuration. A detailed time analysis of the parallel computing processes in the replicated worker approach was completed and two "tuning parameters" (denoted as NUMGROUPING and NUMPENDING) identified to optimize parallel efficiency. In addition, an assessment of the scalable performance improvement was completed on two high performance computing platforms (a Hewlett-Packard HP-SPP2000® and a Silicon Graphics SGI Origin 2000®).

- Automated techniques were developed to vary the sensitivity of the fault detection system as a function of measurement uncertainties, engine inlet

pressure and distortion levels, engine power level, nozzle throat Mach number, rotor speed derivatives, and other measurement system considerations to maximize the probability of detecting faults, and to minimize the probability of false alarms.

- A technique was developed which is capable of real-time interpretation of measured and predicted responses and interrelationships throughout the propulsion system quantified by the CLM. A simultaneous multipoint analysis was used to provide a relative assessment of measurement error and changes in component performance. The multipoint analysis includes the following:

    - Interpretation of differences between predicted and measured aerothermodynamic measurements over the time being considered (to validate modeling assumptions and detect measurement errors);

    - Interpretation of changes in component flows and efficiencies using data immediately prior to and during the time being considered (to detect abrupt faults); and

    - Interpretation of changes in component flows and efficiencies using data obtained considerably before as well as during the time being

considered (to detect slower faults such as engine degradation or sensor drift).

- Model simulation studies were automated to determine the source of a fault and to estimate the extent and probability of the fault in near-real time. This detailed diagnostic information enables the analysis engineer to assess the relative probability of the fault and, in most cases, quickly identify and verify the actual cause of the fault. The fault identification process was fully automated and the ability to detect and accurately diagnose faults is no longer a function of the experience of the analysis engineer.

- Problems with commonly used fault detection and condition monitoring systems such as unclear analysis assumptions and difficulty in interpreting the results of the gas-path analysis were addressed in the present research. Accordingly, gas turbine performance deterioration modeling techniques were developed as an integral part of the CLM-based engine performance and fault identification method. In addition, techniques were developed in the current research to select the test measurements, modeling assumptions, and model calibration techniques to lessen the tendency to underestimate (or overestimate) hardware degradation.

- Fault identification studies were performed using engine model simulations and actual test data. Additionally, both test facility and engine control measurements were used to enhance the fault identification process. Moreover, the improved modeling techniques developed in the present research accurately account for altitude effects, measurement bias, and known engine degradation.

- A capability to identify abrupt engine faults due to domestic object damage, foreign object damage, or control anomalies and measurement errors due to level shifts, noise, or poor response and slow faults due to component or overall engine degradation, and sensor drift was developed and proven.

- The fault detection and diagnostic algorithms developed in the present investigation were evaluated using data from two distinctly different modern military turbofan engines undergoing ground test development at simulated altitude test conditions. Data from hundreds of sensors were recorded at rates up to 100 samples per second to characterize aerothermodynamic engine performance. Both engines were tested over a wide range of steady-state and transient engine operating conditions enabling a comprehensive assessment of the suitability of the approach for developmental engine testing.

- Accurate predictions from the CLM are a key element of the fault detection and diagnostic processes. Therefore, comparisons of test data and model predictions were performed for key performance parameters including engine thrust and low-pressure turbine exit pressure and temperature to verify the accuracy of the model predictions. The close agreement between measured and predicted values over a wide range of operating conditions including a slow engine acceleration, rapid engine deceleration, and a rapid engine acceleration followed by a rapid engine deceleration, demonstrated the accuracy of the calibrated model.

Based on these results and efforts, it is also concluded that the CLM-based engine performance and fault identification method developed in the present research is an efficient, practical, online, real-time fault identification technique for developmental engine testing.

9. RECOMMENDATIONS FOR FUTURE WORK

The three-step calibration process developed in the present research is sufficiently accurate for development and verification of the fault identification algorithms and provides an initial capability to identify faults in real-time during developmental engine testing. Additional refinement of the aerothermo model and calibration process would further minimize engine-model mismatch and permit identification of faults which would be otherwise undetectable.

The mixing technique employed in the component-level model (see Mixer Analysis Summary, page 112) provides a unique solution for the thermodynamic properties of the mixed stream in terms of the two initial streams and appeared to be sufficiently accurate for development and verification of the fault identification algorithms developed in this research. However, the quasi-rigorous method, while computationally efficient, neglects all transverse property gradients and expresses the shear and energy transfer in terms of one-dimensional fluid properties. A more rigorous solution to the equations of fluid motion is recommended for inclusion in the engine cycle model to yield additional information about the mixing process and engine performance during steady-state and transient engine operation. Methods to further reduce engine-model mismatch include improvements in the digital filtering technique

and improved models to characterize "known" and "assumed" characteristics during steady-state, transient, and dynamic engine operation.

A higher than anticipated variation in the number of iterations required to balance the engine model-cycle exists between serial and parallel modes of operation. This is due, in part, to degradation of the initial guesses in the parallel processing mode as more and more processors are added. In order to address this degradation, a simple time-step integration technique, similar to the technique employed by French (1982) in his development of a compact real-time turbofan engine dynamic simulation, can be used to improve the initial guesses. Future research will address the scalable computing performance and accuracy issues, in detail, for a variety of

- computing platforms,
- data transfer and synchronization techniques,
- engine and test cell configurations,
- numerical solution techniques,
- measurement uncertainties, and
- test maneuvers.

A variety of interactive tools are currently available for use in developmental engine tests that permit more detailed diagnostics of steady-state data (and if necessary,

backward chaining to identify the fault). Due to observability constraints imposed by the specified instrumentation and test conditions at the time of the identification, some loss of diagnostic resolution of data anomalies results from lumping together component modifiers and measurements used in the model-based diagnostic system (e.g., burner fuel flow (WFB) is a function of measured fuel heating value, specific gravity, viscosity, fuel temperature, turbine flow meter output, etc.). Future work includes linking the model-based fault identification approach with other fault diagnostic systems to better analyze all available information.

The current system is optimized to identify data anomalies, which occur much more frequently than engine anomalies. Additional benefits may also be derived from development of two sets of algorithms: one optimized to detect and diagnose data anomalies, and the other optimized to detect and diagnose engine anomalies.

REFERENCES

REFERENCES

Abernathy, R.B., Pratt & Whitney Aircraft, and Thompson, J. W., ARO, Inc., "Handbook of Uncertainty on Gas Turbine Measurements." AEDC-TR-73-5 (AD-755356), February 1973.

Adibhatla, S. and Johnson, K.L., "Evaluation of a Nonlinear PSC Algorithm on a Variable Cycle Engine", AIAA-93-2077, June, 1993.

Adibhatla, S. and Gastineau, Z., "Tracking Filter Selection and Control Mode Selection for Model Based Control", AIAA-94-3204, June, 1994.

"Aerospace Information Report 1871B: Lessons Learned from Developmental and Operational Turbine Engine Monitoring Systems," Society of Automotive Engineers, May 1994.

"Aerospace Information Report 1900: Temperature Monitoring," Society of Automotive Engineers, Society of Automotive Engineers, November 1997.

"Aerospace Information Report 4061: Guidelines for Integration of Engine Monitoring Functions with On-Board Systems," Society of Automotive Engineers, June 1996.

"Aerospace Information Report 4175: Ground Station for Condition Monitoring," Society of Automotive Engineers, February 1994.

"Aerospace Information Report 5120: Engine Monitoring System Reliability and Validity," (DRAFT), Society of Automotive Engineers.

"Aerospace Information Report 4176: Cost Versus Benefits of Engine Monitoring Systems," Society of Automotive Engineers, October 1995.

"Aerospace Information Report 4548: Real-Time Modeling Methods for Gas Turbine Engine Performance," Society of Automotive Engineers, 1995.

"Aerospace Information Report 5137: On-Board ACMS Report Guideline," (DRAFT), Society of Automotive Engineers.

"Aerospace Recommended Practice 1587: Aircraft Gas Turbine Engine Monitoring System Guide, Society of Automotive Engineers," April 1993.

"Aerospace Resource Document 50002: A Discussion of Standardized Concepts for Condition Monitoring and Performance Analysis Software," Society of Automotive Engineers, November 1992.

"Aerospace Standard 728: Aircraft Gas Turbine Engine Monitoring Systems," Society of Automotive Engineers, April 1993.

"Aerospace Standard 755: Aircraft Propulsion System Performance Station Designation and Nomenclature," Society of Automotive Engineers, 1996.

AGARD Advisory report No. 320, Guide to the Measurement of the Transient Performance of Aircraft Turbine Engines and Components: AGARD-AR-320, March 1994.

AGARD Conference Proceedings No. 448, Engine Condition Monitoring - Technology and Experience: AGARD-CP-448, October 1988.

Ainley, D.G., and Mathieson, G.C.R., 1951, "A Method of Performance Estimation for Axial-Flow Turbines," British ARC, R&M 2974.

Aker, G.F., and Saravanamuttoo, H.I.H., 1989, "Predicting Gas Turbine Performance Degradation Due to Compressor Fouling Using Computer Simulation Techniques," Transactions of the ASME: Journal of Engineering for Gas Turbines and Power, Vol. 111.

Annex, A. P., "Application of System Identification Techniques to Turbine Engine Post-Stall Test and Evaluation," AEDC TR-90-32.

ASME MFC-3M, -Measurement of Fluid Flow in Closed Conduits Using Orifice, Nozzle, and Venturi-, ASME, New York, 1985.

ASTM D341-89, "Standardized Viscosity-Temperature Charts for Liquid Petroleum Products."

Astrom, K.J., and Eykhoff, P., "System Identification – A Survey", Automatica, Vol. 7, pp. 123-162. Pergamon Pres, 1971.

"Automatic FORTRAN Parallelization for Distributed Memory Systems with FORGE Magic/DM," Applied Parallel Research, Inc., Placerville CA, 1992.

Balje', O.E., and Binsley, R.L., 1968, "Axial Turbine Performance Evaluation. Part A – Loss Geometry Relationships," Journal of Engineering and Power.

Baumann, W.T., and Rugh, W.J., "Feedback Control of Nonlinear Systems by Extended Linearization," IEEE Trans. Automatic Control, Vol. AC-31, No. 1, Jan. 1986.

Bestle, D., and Zeitz, M., "Canonical Form Observer Design for Nonlinear Time-Variable Systems," Int. J. Control, Vol. 38, No. 2, 1983, pp. 419-431.

Biegl, C., Malloy, D.J., and Chappell, M. A., "Automated, Real-Time Validation of Turbine Engine Test Data Using Explicit Parallelization on an Eight Processor Pentium® Platform." Thirteenth International Symposium on Airbreathing Engines, September 8-12, 1997, Chattanooga, TN.

Bird, J.W., and Schwartz, H.M., "Diagnosis of Turbine Engine Transient Performance with Model-Based Parameter Estimation techniques", ASME-94-GT-317, June, 1994.

Borgmeyer, C.H., "A Streamlined Control System Development Process", AIAA-79-1344, June, 1979.

Broyden, C.G., "Quasi-Newton Methods and Their Application to Function Minimization," Mathematics of Computation, 21 (1967), pp. 568-581.

Bushman, M. and Gallops, G.W., "In-Flight Performance Identification Capability of an Adaptive Engine Model," AIAA-92-3747, July 1992.

Chappell, M.A., and McLaughlin, P.W., "Approach to Modelling Continuous Turbine Engine Operation from Startup to Shutdown," Journal of Propulsion and Power, Vol. 9, No. 3, May-June 1993, pp. 466-471.

Chisholm, J. D., "In-Flight Optimization of the Total Propulsion System," AIAA-92-3744, July 1992.

Conners, T.R., "Thrust Stand Evaluation of Engine Performance Improvement Algorithms in an F-15 Airplane," AIAA-92-3747, July 1992.

Cunningham, R. G. "Orifice Meters with Supercritical Compressible Flow." ASME Paper No. 50-A45, August 1950.

Davis, J., Bapty, T., Karsai, G., Sztipanovits, J., Tibbals, T., and Malloy, D., "A Model-Based Data Validation System," Society for Machinery Failure Prevention Technology, 1996 Technology Show-case on Integrated Monitoring, Diagnostics and Failure Prevention.

"Distributed pC++: Basic Ideas for an Object Parallel Language," Scientific Computing, Vol. 2, No. 3, Fall 1993.

Doel, D. L., "The Role for Expert Systems in Commercial Gas Turbine Engine Monitoring," ASME Paper 90-GT-374 Gas Turbine and Aeroengine Congress and Exposition, Brussels, Belgium, June 11-14, 1990.

Doebelin, E.O., "Measurement Systems, McGraw-Hill, 1975.

Doel, David L., "TEMPER – A Gas-Path Analysis Tool for Commercial Jet Engines," ASME 92-GT-315, June 1992.

Duponchel, J.P, Loisy, J., and Caririllo, R., "Steady and Transient Performance Calculation Method for Prediction, Analysis, and Identification," AGARD Lecture Series: Steady and Transient Performance Prediction of Gas Turbine Engines," circa 1990.

Dvorak, S.D., Hosny, W.M.; and Steenken, W.G., E_s(3) 10C Compressor Test Analysis of High-Speed Post-Stall Data, NASA CR-179521 R86Aeb564, October 1986.

Espana, M.D., "On the Estimation Algorithm for Adaptive Performance Optimization of Turbofan Engines", AIAA-93-1823, June 1993.

Evans, D.C., D Rees, D.L. Jones and D.C. Hill, 1994, "Time and Frequency Domain Identification of Jet Engine Dynamics: Problems and Solution," 10th IFAS Symposium on System Identification, Session CL3, Copenhagen.

Fitzgerald, R. J., "Divergence of the Kalman Filter," IEEE Trans. Automatic Control, Vol. AC-16, December 1971, pp. 737-747.

Fowler, C.M., "Analysis of Numerical Solutions of Transient Heat Flow Problems, "Quart. Appl. Math, 3, (1945), p. 361.

French, M. W., "Development of a Compact Real-Time Turbofan Engine Dynamic Simulation", SAE Paper 821401.

Friedland, B., "Treatment of Bias in Recursive Filtering," IEEE Trans. Automatic Control, Vol. AC-14, No. 4, August 1969, pp. 359-367.

"Gas Turbine Performance Analysis," United Technologies Corporation, 1994.

Gelb, A., Applied Optimal Estimation, M.I.T. Press, Cambridge, MA, 1974.

Gevers, M., and Bastin, G., "A Stable Adaptive Observer for a Class of Nonlinear Second_Order Systems," Analysis and Optimization of Systems, ed. By Bensoussan, A., and Lions, J.L., Springer-Verlag, June 1986, pp. 143-155.

Gilyard, G. and Orme, J., "Subsonic Flight Evaluation of a Performance Seeking Control Algorithm on an F-15 Airplane," AIAA-92-3743, July 1992.

Glawe, G.E., Holanda, R., Krause, L.N., "Recovery and Radiation Corrections and Time Constants of Several Sizes of Shielded and Unshielded Thermocouple Probes for Measuring Gas Temperature, NASA 1099, 1978.

Glawe, G.E., Simmons, F.S., and Stickney, T.M. "Radiation and Recovery Corrections and Time Constants of Several Chromel-Alumel Thermocouples in High Temperature, High Velocity Gas Streams." NASA-TN-3866, October 1956.

Grey, J. Transient Response of the Turbine Flowmeter, Jet Propulsion, February 1956.

Gupta, N.K., and Hall, W.E., Jr., "Input Design for Identification of Aircraft Stability and Control Derivatives," NASA Contractor Report CR-2493, Feb. 1975.

Gupta, N.K., Hall, W.E., Jr., and Trankle, T.L., "Advanced Methods of Model Structure Determination from Test Data," Journal of Guidance and Control, Vol. 1, No. 3, May-June 1978.

Gupta, N.K., and Hall, W.E., Jr., "Design and Evaluation of Sensor Systems for State and Parameter Estimation," J. Guidance and Control, Vol. 1, No. 6, Nov-Dec., 1978 pp. 397-403.

Hollis, G.D., "The Reduction of Scale Force Uncertainty in Low Thrust Regions," Master thesis, University of Tennessee, December 1997.

Kalman, R.E., "On a New Approach to Filtering and Prediction Problems," ASME Journal of Basic Engineering, Vol. 28, March 1960.

Khalid, S.J. and Faherty, M.F., "Propulsion System Flight Test Analysis Using Modeling Techniques", International Journal of Turbo and Jet Engines, 10, 31-43 (1993).

Kimzey, W.F. and Wantland, E.C., "Short Duration Turbine Engine Testing for Energy Conservation," AIAA-77-991, July 1977.

Leppert, G., "A Stable Numerical Solution for Transient Heat Flow Problems, "J. Am. Soc. Naval Engrs., 65, (1973), p. 741.

Luppold, R. H. et al, "Estimating In-Flight Engine Performance Variations Using Kalman Filter Concepts," AIAA-89-2584.

Mach, Kervyn, D., "Viscous Mixing of Parallel Gas Streams", Research Technology Briefs, Vol. 3, No. 11, Air Force Systems Command, November 1965.

Maine, Trindel A., Gilyard, Glenn B. and Lambert, Heather H., "A Preliminary Evaluation of an F100 Engine Parameter Estimation Process Using Flight Data," NASA TM-4216, 1990.

Malloy, D. J., "Improved Data Validation and Quality Assurance in Turbine Engine Test Facilities," AIAA-93-2178, June 1993.

Marxman, G. A. and Burlage, H. "Expansion Coefficients for Orifice Meters in Pipes Less Than One Inch in Diameter", Journal of Basic Engineering, pp. 289-298, June 1961.

McLaughlin, P.W., "REPLICAS-A New Continuous System Simulation Language," Proceedings of the 15th Annual Simulation Symposium, Association for Computing Machinery, Tampa, FL, March 1982, pp. 1-15.

Misawa, E.A., and Hedrick, J.K., "Nonlinear Observers – A State-of-the-Art Survey," Transactions of the ASME, Vol. 111, September 1989.

Moffat, R.J., "Designing Thermocouples for Response Rate", ASME 57-GTP-8, 1957.

MPI Standard, version 1.1, MPI-2 Forum, June 12 1994.

Mueller, F.D., Nobbs, S.G., and Stewart, J.F., "Dual Engine Application of the Performance Seeking Control Algorithm", AIAA-93-1822, June, 1993.

Muir, D.E., Saravanamuttoo, H.I.H., and Marshall, D.J., 1988, "Health Monitoring of Variable Geometry Gas Turbines for the Canadian Navy," ASME Journal of Engineering and Power, Vol. 111, No. 2 pp. 245-250.

Mukhtarov, M. Kh, and Kichakin, V.I., 1969, "Procedure for Estimating Flow Section Losses in Axial-Flow Turbines when Calculating Their Characteristics," Central Scientific Research Institute of Aircraft Engine Construction, Teploenergtika.

NASA-CR-174641, "Parallel Processor Engine Model Program," January 1984.

NASA-CR-174645, "Extended Parametric representation of Compressor Fans and Turbines", March 1984.

Nicosia, S., Tomei, P., and Tornambe', "Feedback Control of Elastic Robots by Pseudolinearization Techniques," 25th Conference on Decision and Control, Athens, Greece, Dec. 1986.

Nieman, R.E., Fisher, D.G., and Seborg, D.E., "A Review of Process Identification and Parameter Estimation Techniques," International Journal of Control, 1971, Vol. 1, No. 2, 209-264.

Nobbs, S.G., Jacobs, S.W., and Donahue, D.J., "Development of the Full-Envelope Performance Seeking Control Algorithm," AIAA-92-3748, July 1992.

Orme, John S. and Gilyard, Glenn B., "Subsonic Flight Test Evaluation of a Propulsion System Parameter Estimation Process for the F100 Engine," AIAA-92-3745, July 1992.

Orme, J. and Gilyard, G., "Preliminary Supersonic Flight Test Evaluation of a Performance Seeking Control," AIAA-93-1821, June, 1993.

Parsons,F., Kern, R., Ashleman, R., and Lavelle, T., "The National Cycle Program: A Flexible System Modeling Architecture," AIAA-98-3114, July 1998.

Phaneuf, R.J., "Approximate Nonlinear Estimation," Ph.D. thesis, Massachusetts Institute of Technology, May 1968.

Provost, M.J., "COMPASS: A Generalized Ground-Based Monitoring System" AGARD CP-448, pp. 42-1 – 42-13, October 1988.

PVM: Parallel Virtual Machine -- A Users' Guide and Tutorial for Networked Parallel Computing, MIT Press, Cambridge, 1994.

Reboulet C., and Champetier, C., "A New Method for Linearizing Nonlinear Systems: the Pseudolinearization," Int. J. Control, Vol. 40, 1984, pp. 631-638.

Runyan, R.B., Deken, L.R., and Miller, J.T., "Structural Dynamics of a Small Rocket Thrust Stand", July, 1992.

Safanov, M.G., Stability and Robustness of Multivariable Feedback Systems, MIT Press, Cambridge, MA, 1980.

Safanov, M.G., and Athens, M., "Robustness and Computational Aspects of Nonlinear Stochastic Estimators and Regulators," IEEE Transactions Automatic Control,Vol. AC-23, No. 4, August 1978, pp 717-725.

Sallee, G.P., 1980. "Performance Deterioration Based on Existing (Historical) Data: JT9D Jet Engine Diagnostic Program," NASA CR-135448, Pratt & Whitney Aircraft Group, Contract No. NAS3-20632.

Savelle, S. A. and Prufert, M. B., "Turbine Engine Performance Test and Evaluation Techniques," AIAA-92-3977.

Schoukens, J., R. Pintelton and H. van Hamme, 1994, "Identification of Linear Dynamic Systems Using Piecewise Constant Excitations: Use, Misuse and Alternatives," Automatica, Vol. 30. No. 7.

Shaw, P. et al, "A Design Approach to a Performance Seeking Control," AIAA-86-1674, June 1986.

Sieros, G., Stamatis, A., and Mathioudakis, "Jet Engine Component Maps for Performance Modeling and Diagnosis", Journal of Propulsion and Power, Vol. 13, No.5, September-October 1997.

Slotine, J., "Sliding Controller Design for Nonlinear Systems," Int. J. Control, Vol. 40, 1984.

Smith, G. L., "The Scientific Inferential Relationships Between Statistical Estimation Decision Theory, and Modern Filter Theory," Proc. JACC, Rensselaer Polythechnic Inst., June 1965, pp. 350-359.

Smith, R.E., Jr. and Matz, Roy "Verification of a Theoretical Method of Determining Discharge Coefficients for Venturis Operating at Critical Flow Conditions." AEDC-TR-61-8 (AD-263714), September 1961.

Smith, R. E., Jr. and Wehofer, S. "Measurement of Engine Thrust in Altitude Ground Test Facilities." AIAA-82-0572, AIAA 12th Aerodynamic Testing Conference, March 22-24, 1982.

Sokolnikoff, I.S. and Redheffer, R.M. Mathematics of Physics and Modern Engineering, McGraw-Hill, New York, 1966.

Sorenson, H.W., ed., Kalman Filtering: Theory and Application, IEEE Press, New York, N.Y., 1985.

Strang, Gilbert, "Introduction to Applied Mathematics," Wellesley-Cambridge Press, 1986.

Tabakoff, W., 1986, "Compressor Erosion and Performance Deterioration," ASME Publication FED-Vol. 37, presented at the AIAA/ASME 4th Joint Fluid Mechanics, Plasma Dynamics, and Lasers Conference, Atlanta Georgia.

Tabakoff, W., 1987, "Compressor Erosion and Performance Deterioration,"Transactions of ASME: Journal of Fluids Engineering, Vol. 109.

Templeman, W.G. and Gallops, G.W., "Performance Benefits of Adaptive In-Flight Propulsion System Optimization," AIAA-92-3749, July 1992.

Thau, F.E., "Observing the State of Nonlinear Dynamical systems," Int. J. Control, Vol. 17, No. 3, 1973.

Trankle, T .L., "Practical Aspects of System Identification." ASME 79-WA/DSC-23, December, 1979.

Trankle, T .L., Vincent, J. H., and Franklin, S. N., "System Identification of Nonlinear Aerodynamic Models." North American Treaty Organization AGARDOGRAPH, February 1982.

Trigg, D.W., and Leach, A.G., "Exponential smoothing with an Adaptive Response Rate," Operational Research Quarterly, vol. 15, no. 1, pp. 271-274, 1967.

Tucker, M. "Approximate Calculation of Turbulent Boundary-Layer Development in Compressible Flow." NACA-TN-2337, April, 1951.

Urban, L. A., "Gas-Path Analysis – A Tool for Engine Condition Monitoring," presented at the 33rd Annual International Air Safety Seminar, Christchurch, New Zealand, September 1980.

Volpini, A. J., "Gas Path Analysis – An Approach to Engine Diagnostics," presented at the 35th Symposium of the Mechanical Failures Prevention Group, Gaithersburg, Maryland, April 1982.

Wantland, E.C., "Turbine Engine Operabilty Test and Evaluation Techniques", AIAA-91-2277, June 1991.

Wehofer, S. and Rivir., Measurement of Turbine Engine Transient Airflow in Ground Test Facilities, AEDC-TR-80-21, August 1980.

Willbanks, C.E. "Recovery Characteristics of a Single-Shielded Self-Aspirating Thermocouple Probe at Low Pressure Levels and Subsonic Speeds." AEDC-TR-71-68 (AD-882199L), April 1971.

Willsky, A. S., "A Survey of Design Methods for Failure Detection in Dynamic Systems," Automatica, Vol. 12, pp. 601-611, Pergamon Press, 1976.

Zaita, A.V., Buley, G., and Karlsons, "Performance Deterioration Modeling in Aircraft Gas Turbine Engines," ASME 97-GT-278, June 1997.

APPENDICES

APPENDIX A: FIGURES

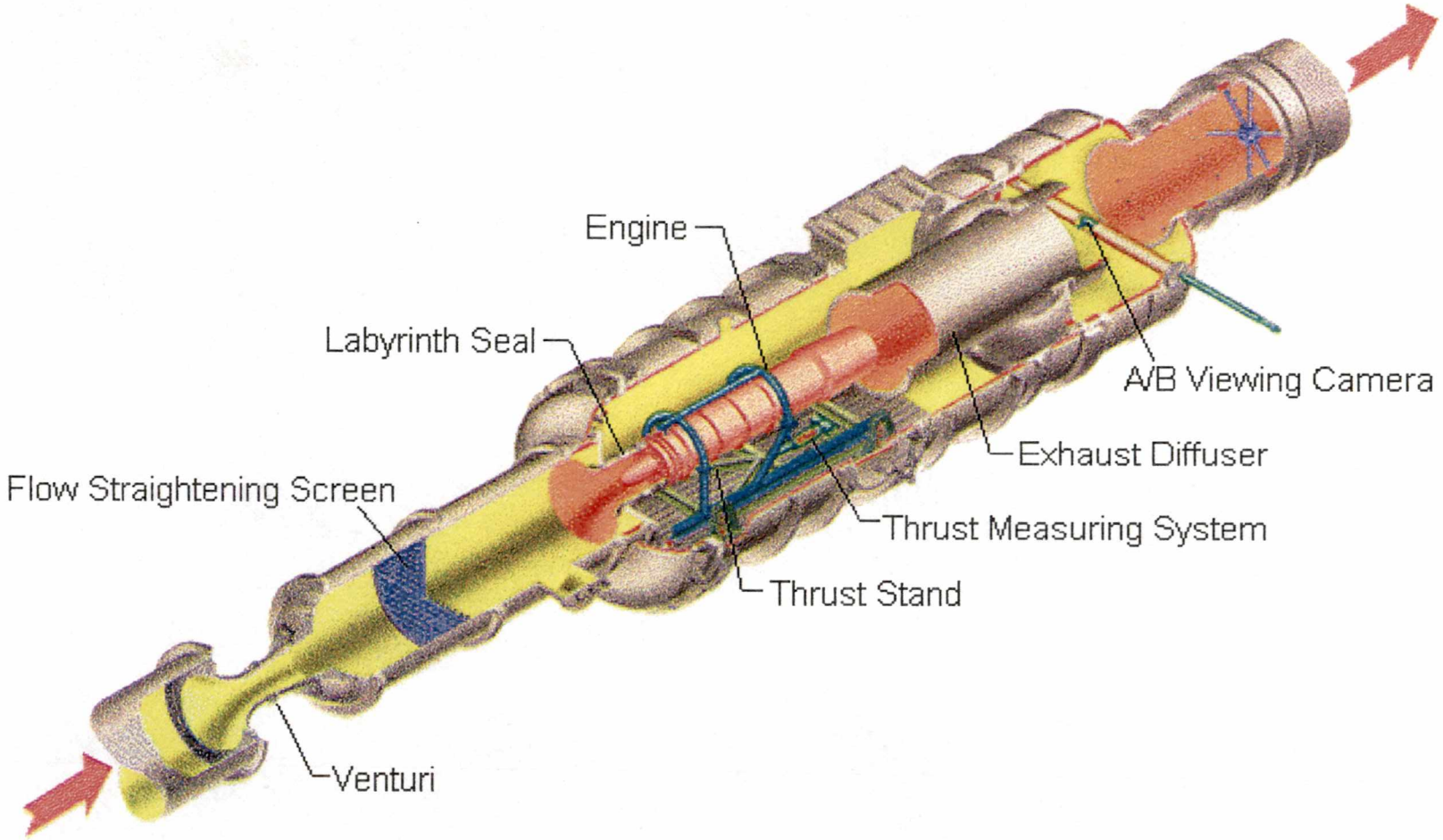


Fig. 1.1. Typical Propulsion Development Test Cell

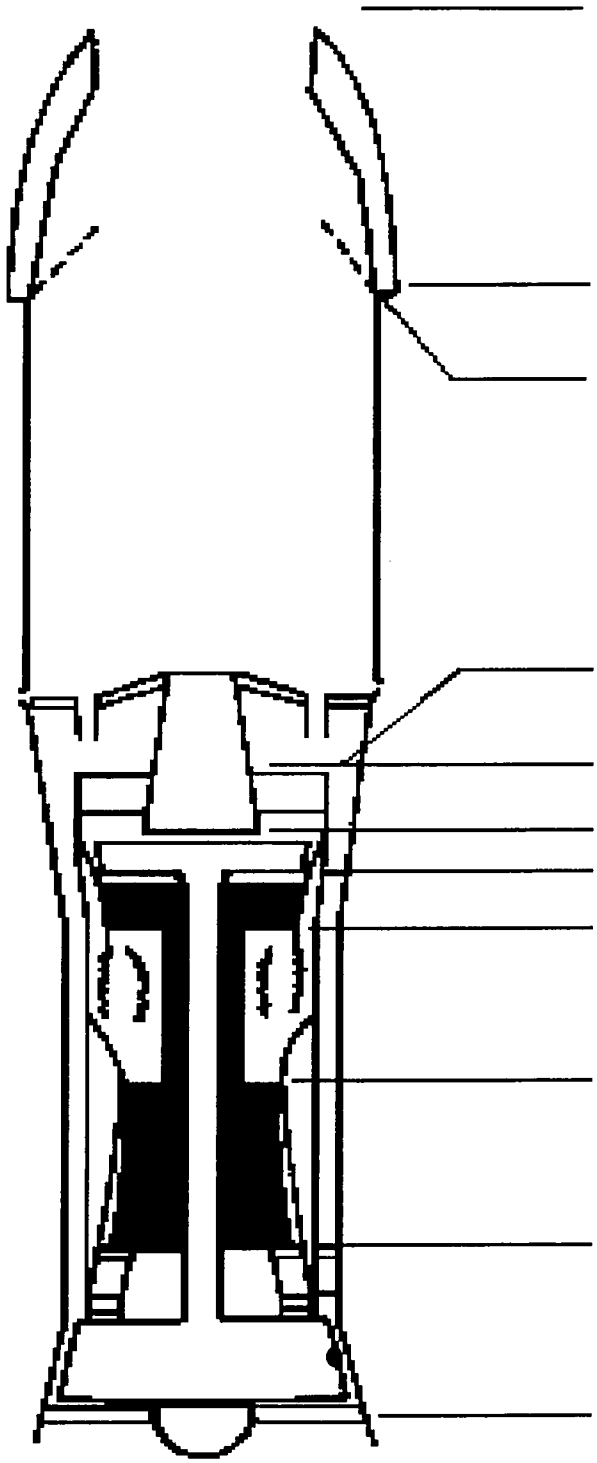


Fig. 1.2. Augmented Turbofan Engine

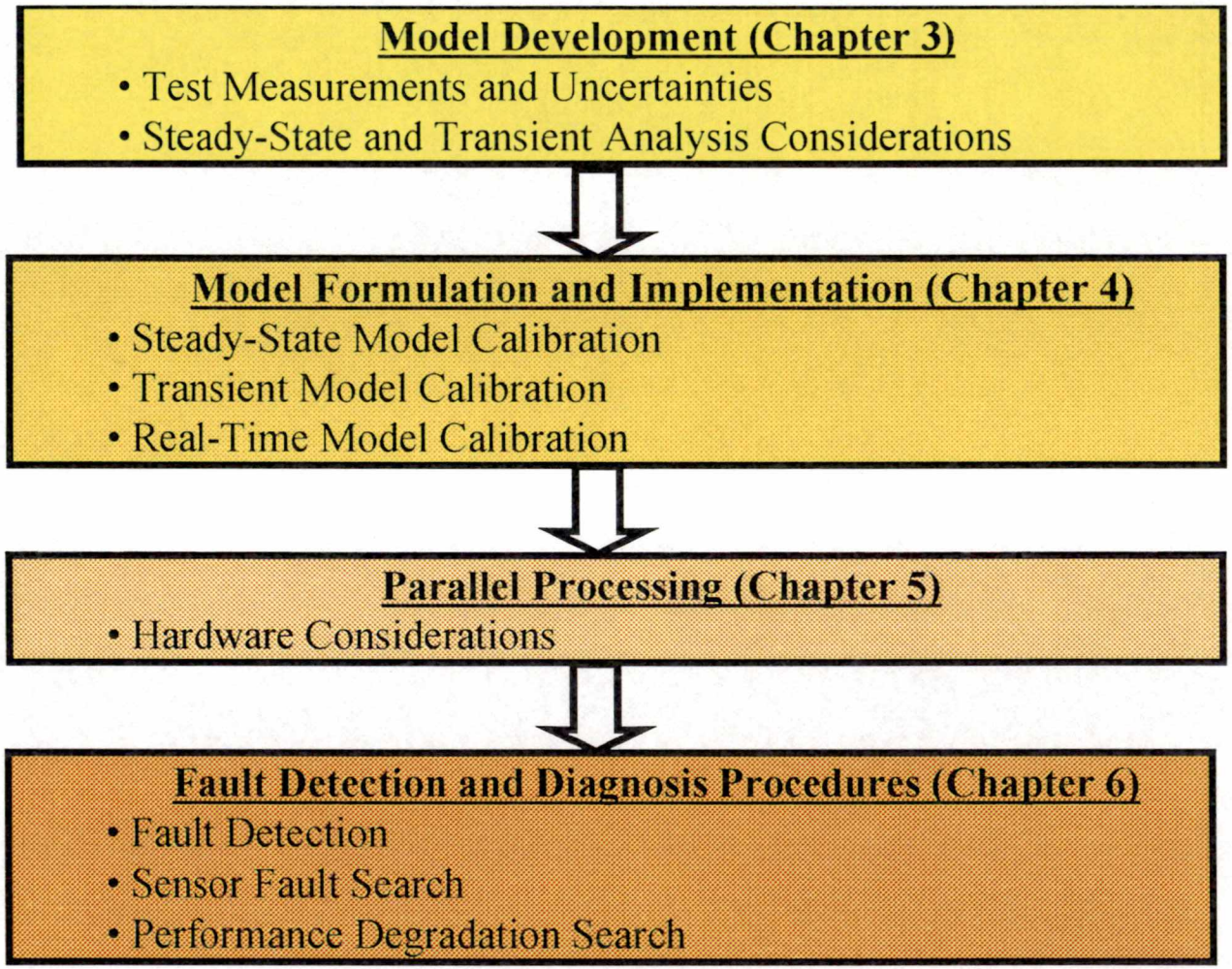


Fig. 1.3. Fault Identification Overview

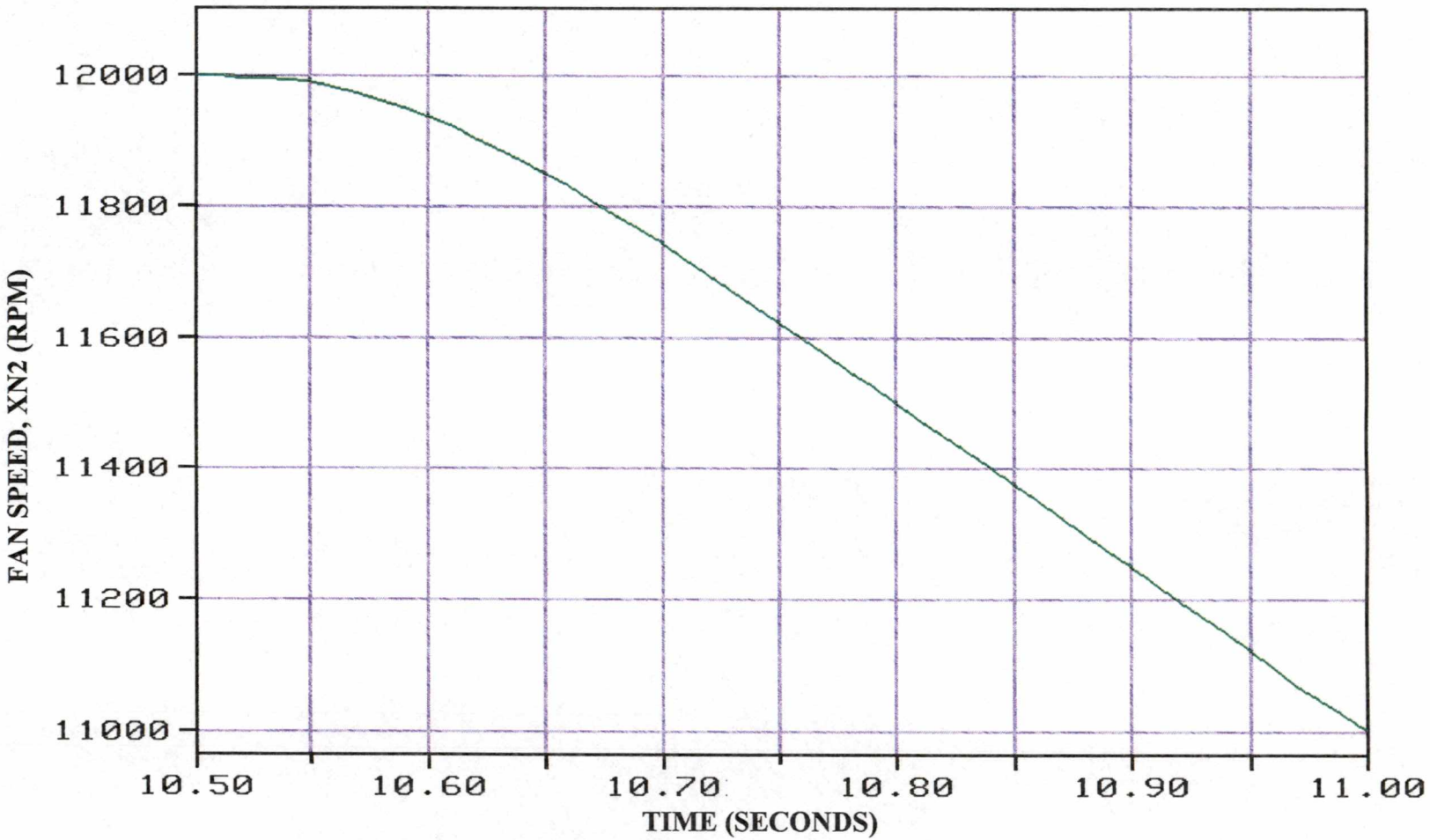


Fig. 3.1. Engine Deceleration: Fan Speed vs. Time

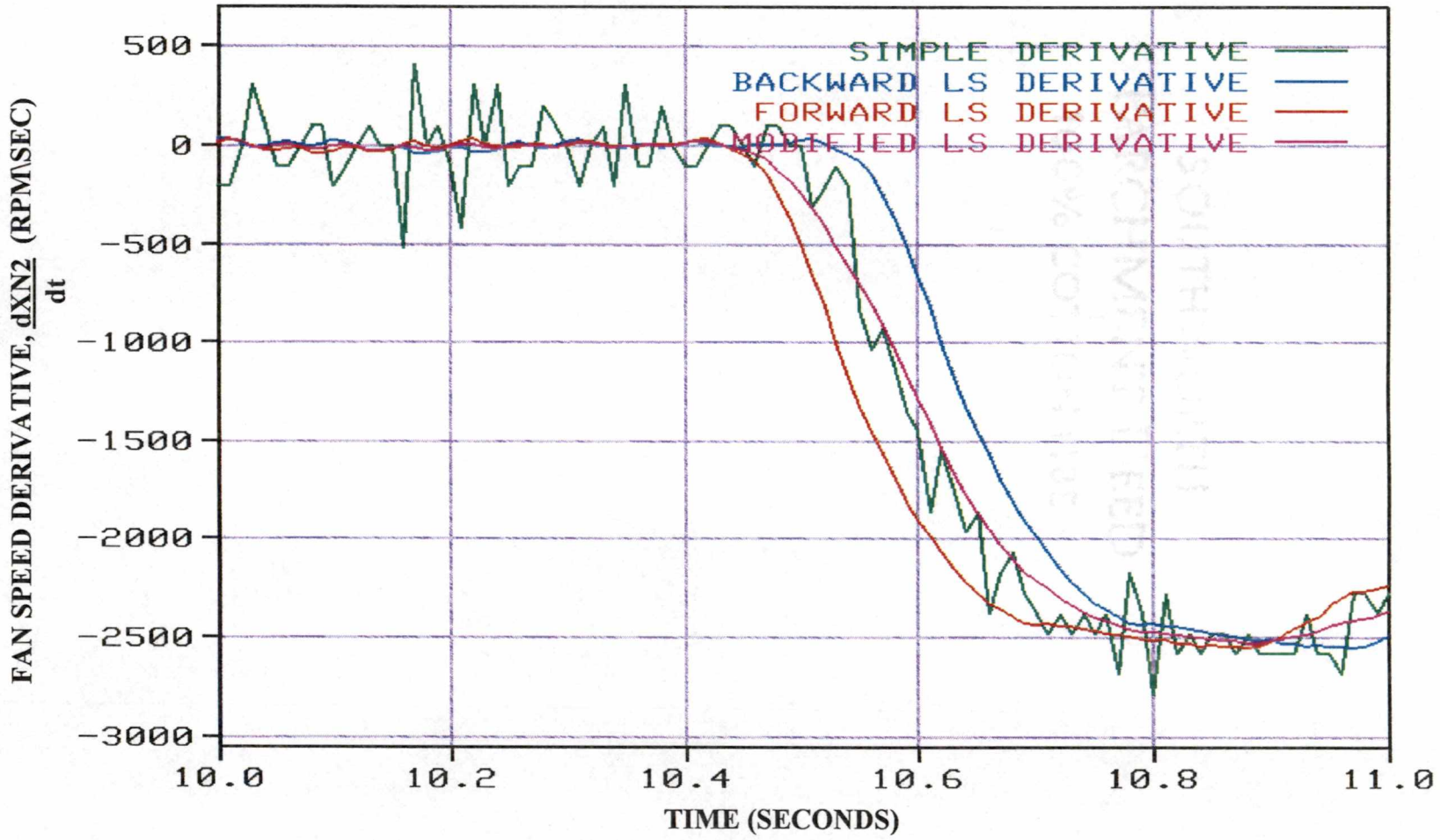


Fig. 3.2. Engine Deceleration: Fan Speed Derivative vs. Time

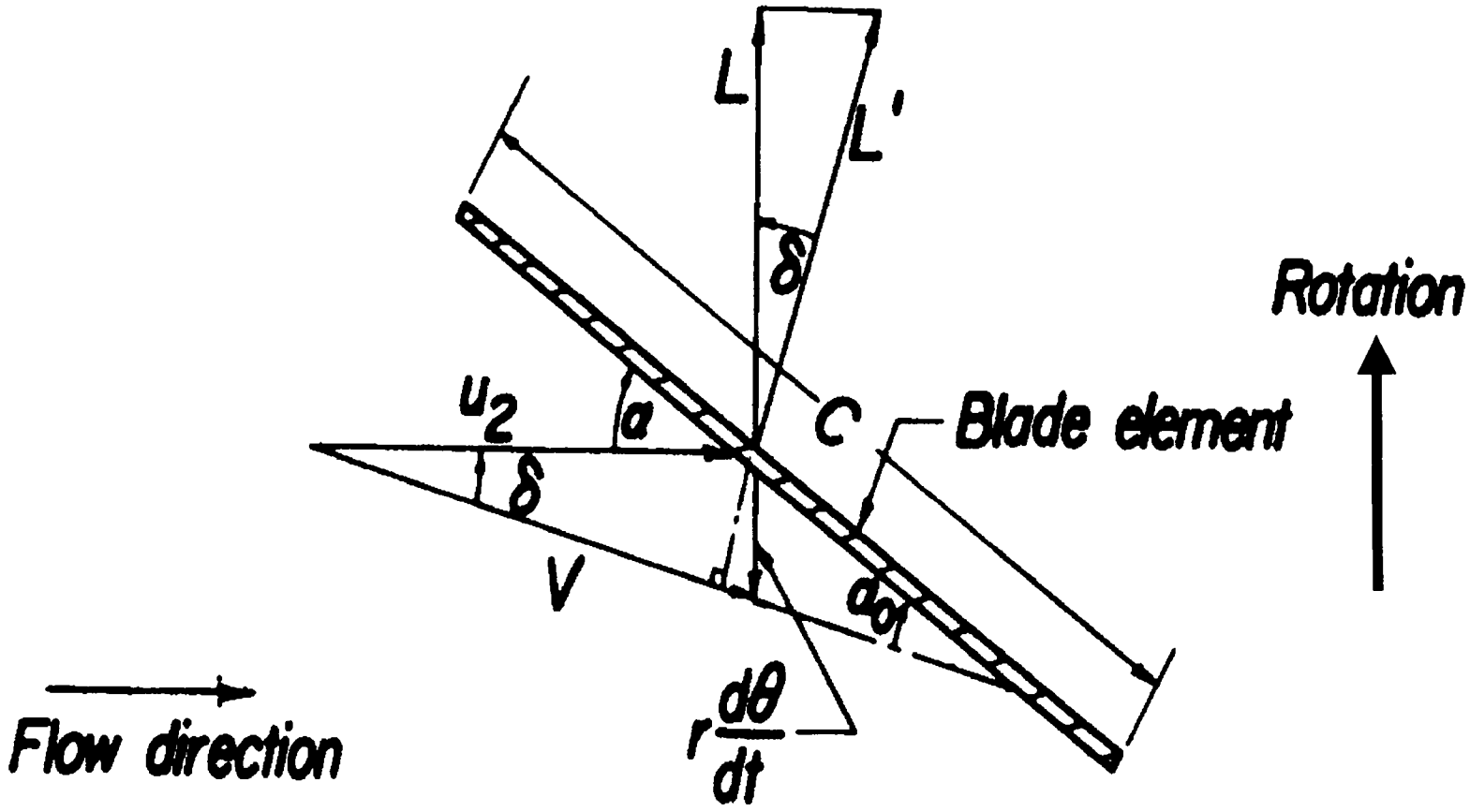


Fig. 3.3. Hydrodynamic Forces on a Blade Element Subjected to a Step Increase in Flow Rate

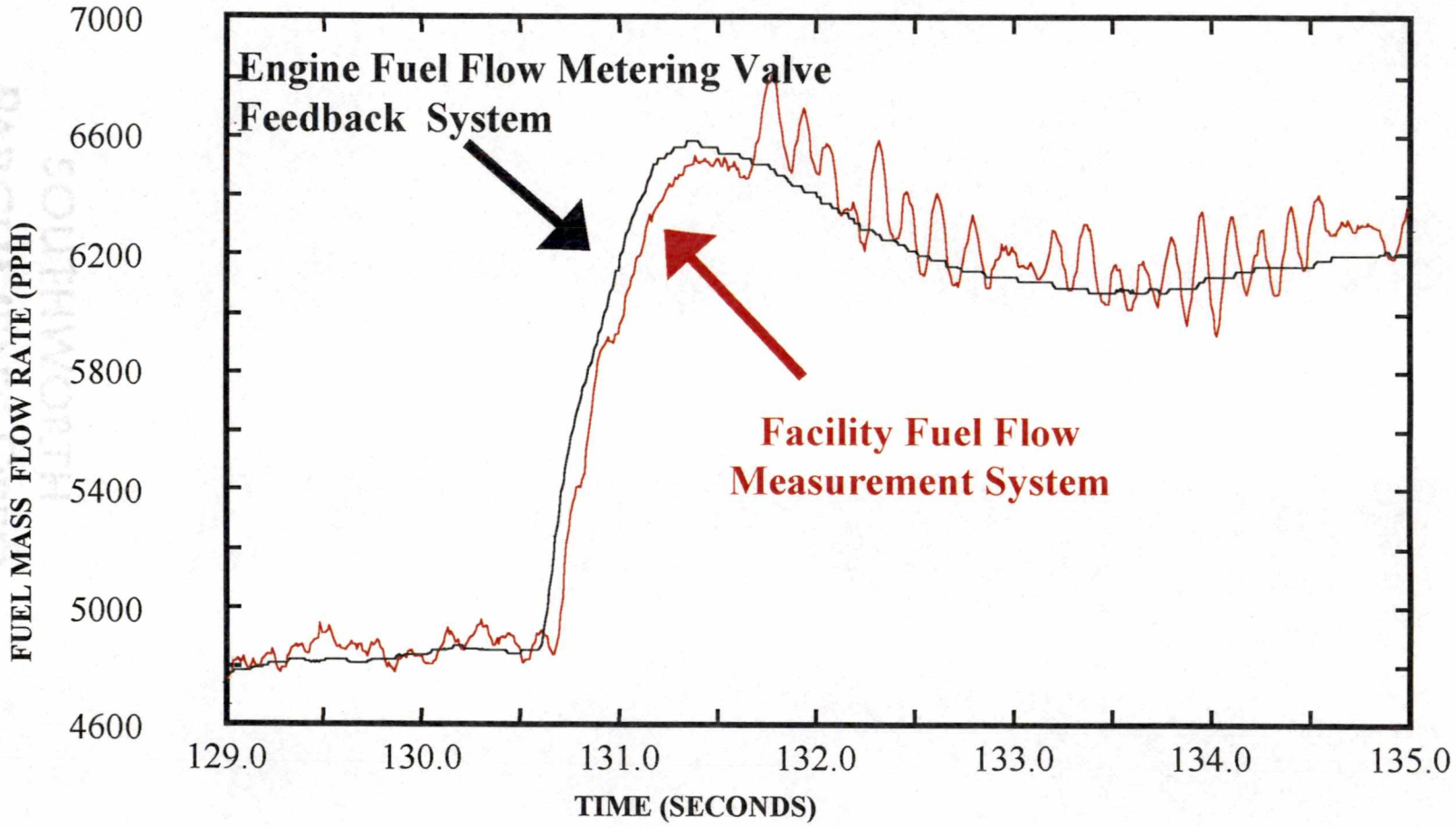


Fig. 3.4. Fuel Flow Rate Oscillations During an Engine Acceleration

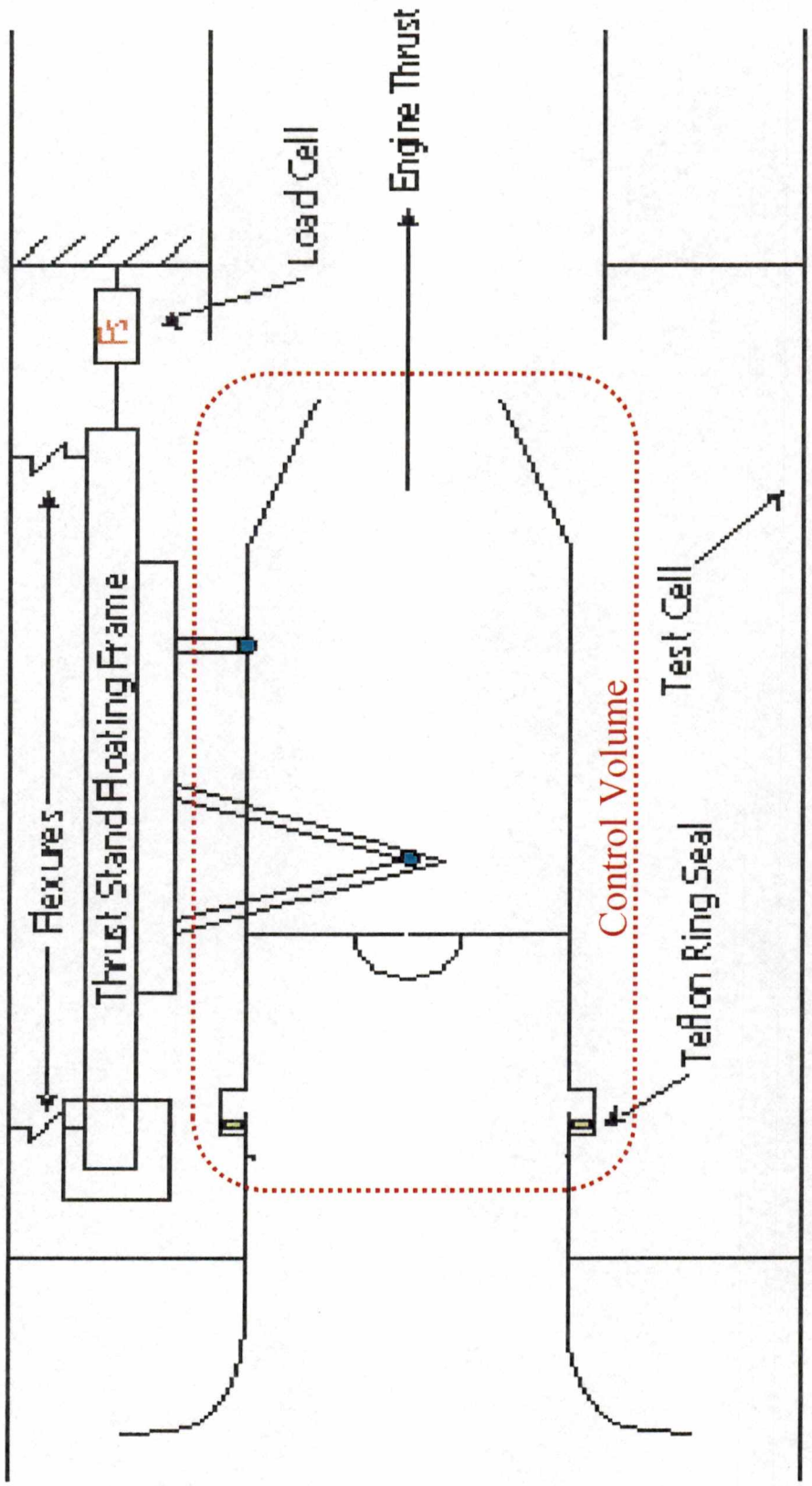


Fig. 3.5. Typical Thrust Measurement Configuration

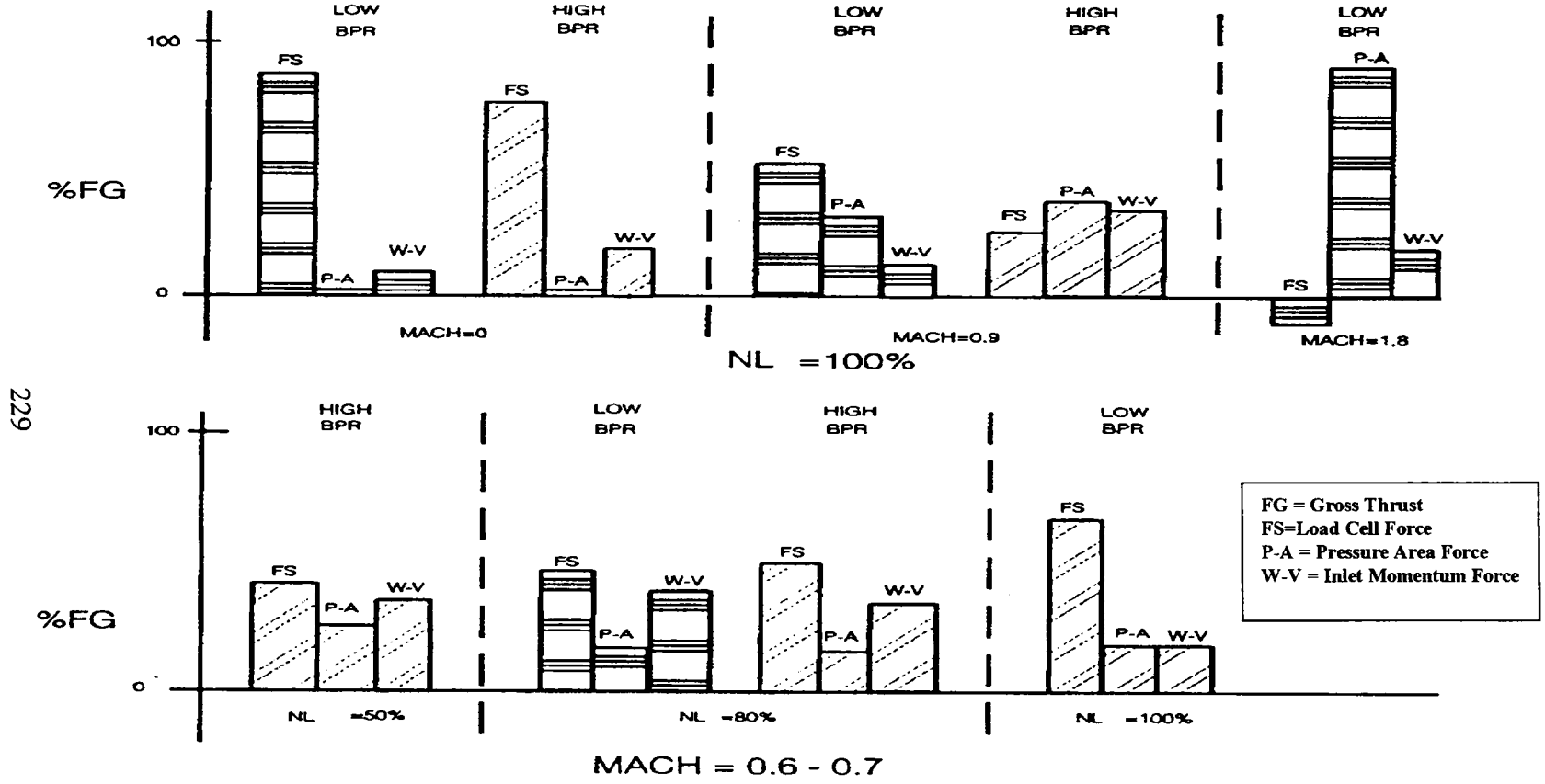


Fig. 3.6. Relative Magnitudes of Gross and Net Thrust and Ram Drag

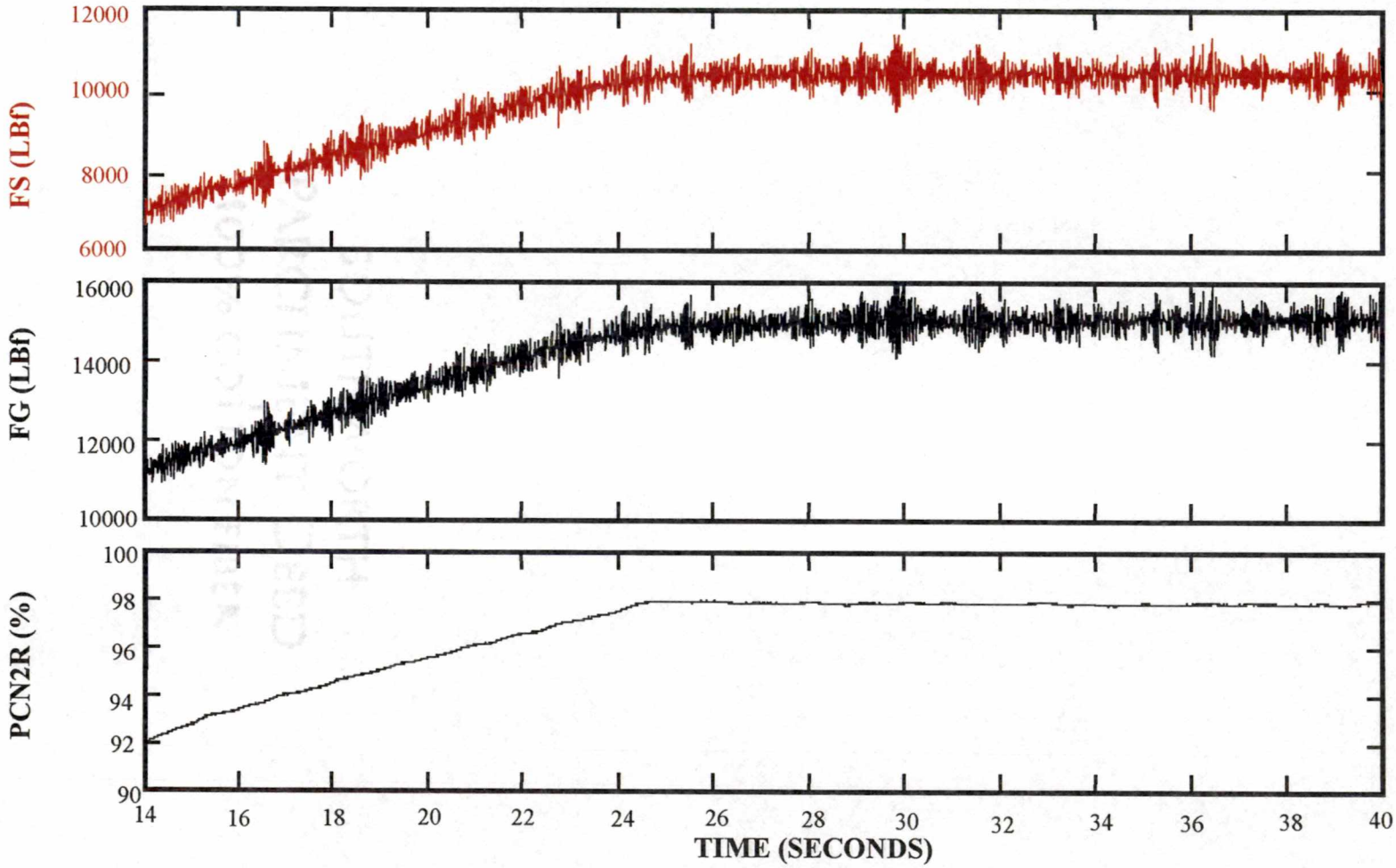


Fig. 3.7. Thrust Stand Oscillations During an Engine Acceleration

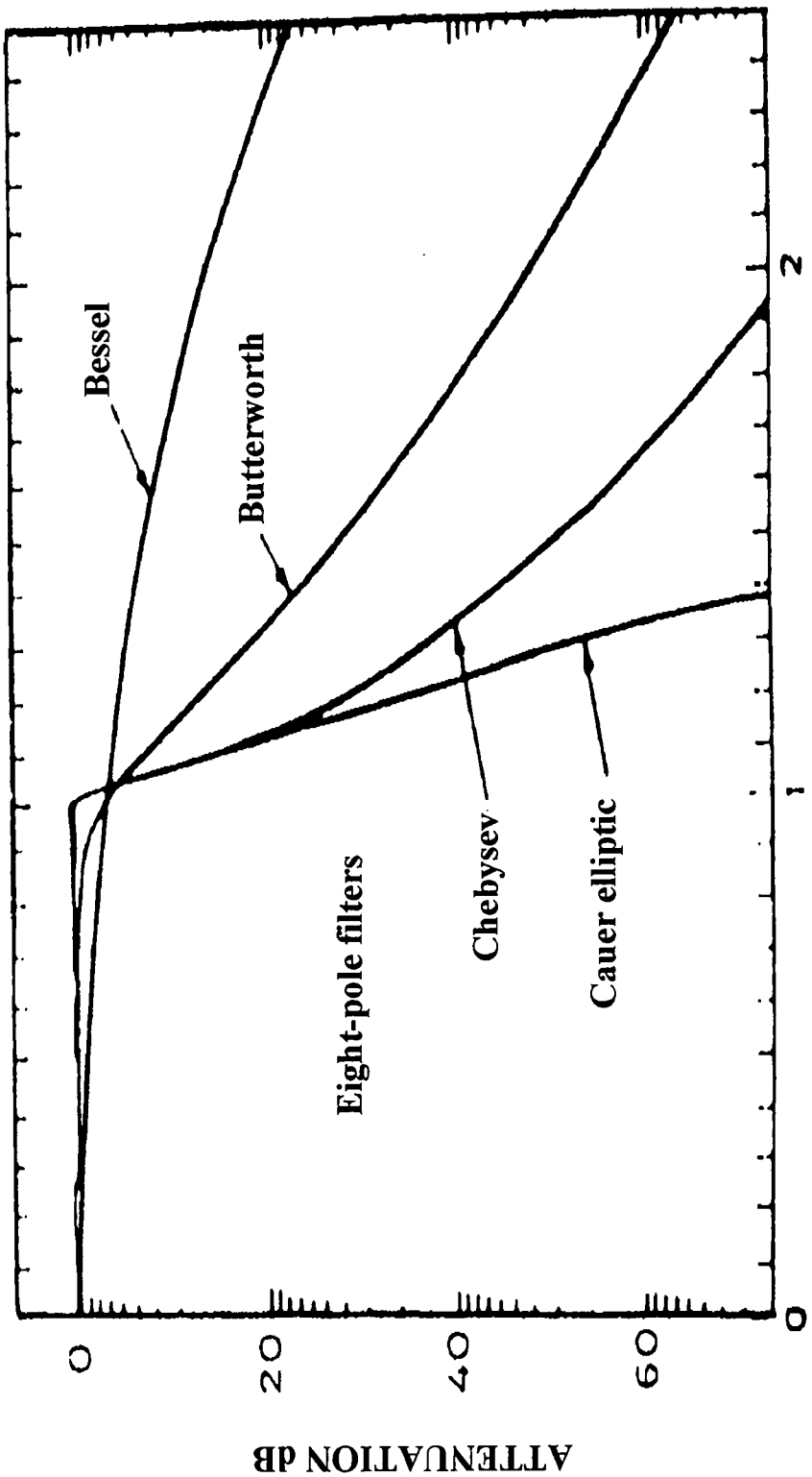


Fig. 3.8. Attenuation of Typical 8-Pole Low Pass Filters

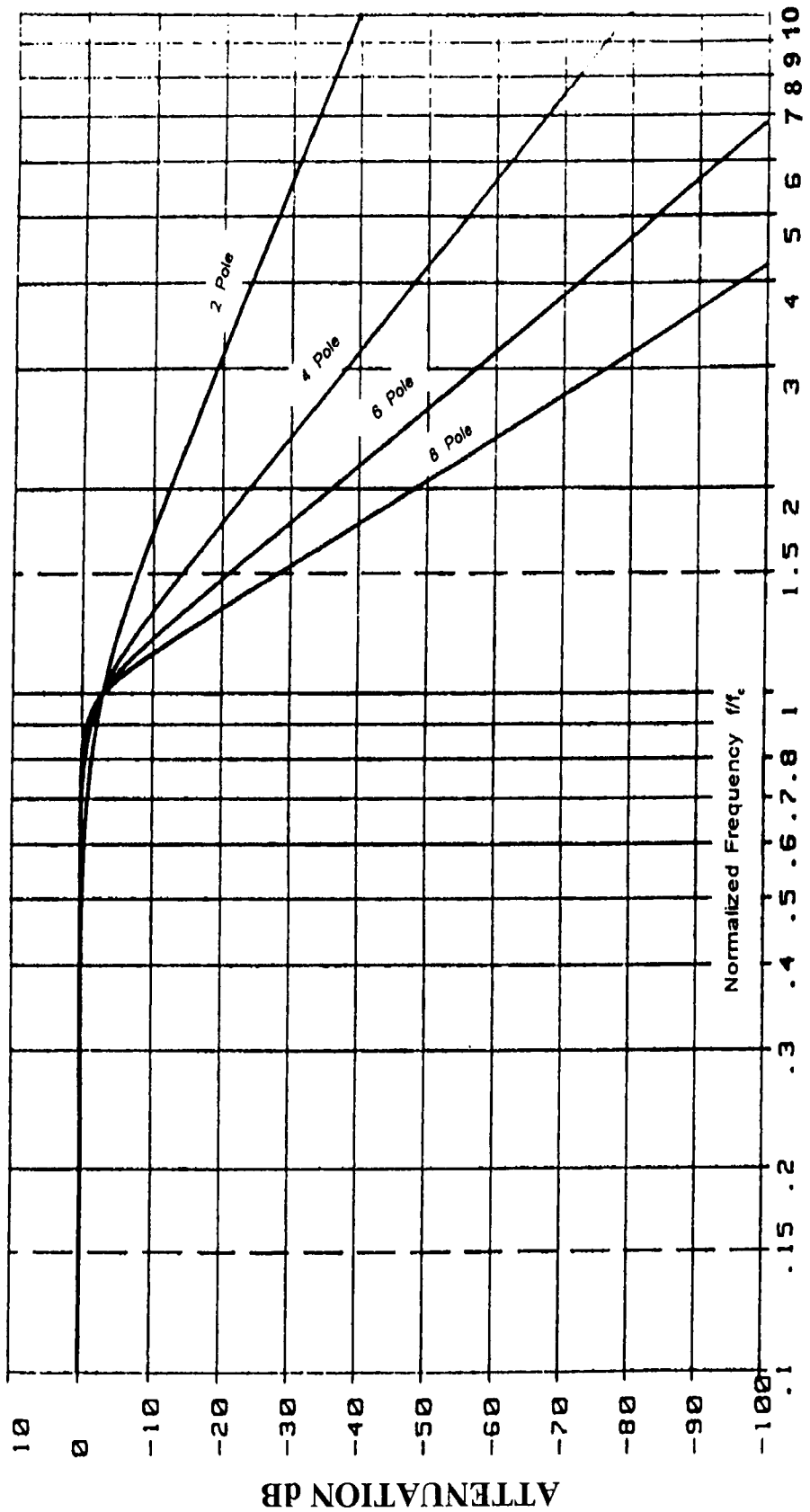


Fig. 3.9. Theoretical Frequency Response of Butterworth Filters

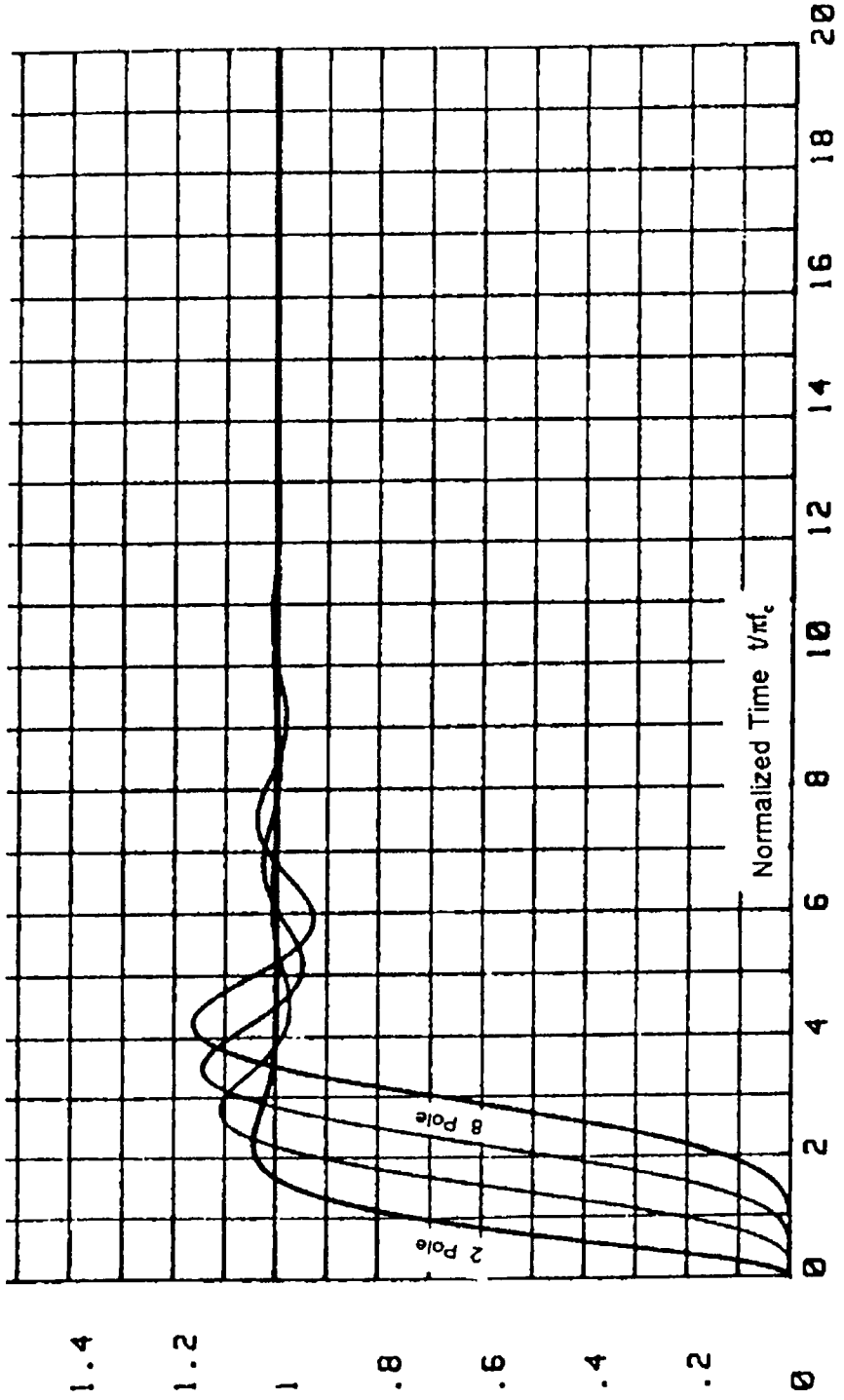


Fig. 3.10. Theoretical Response of Butterworth Filters to a Step Input Signal

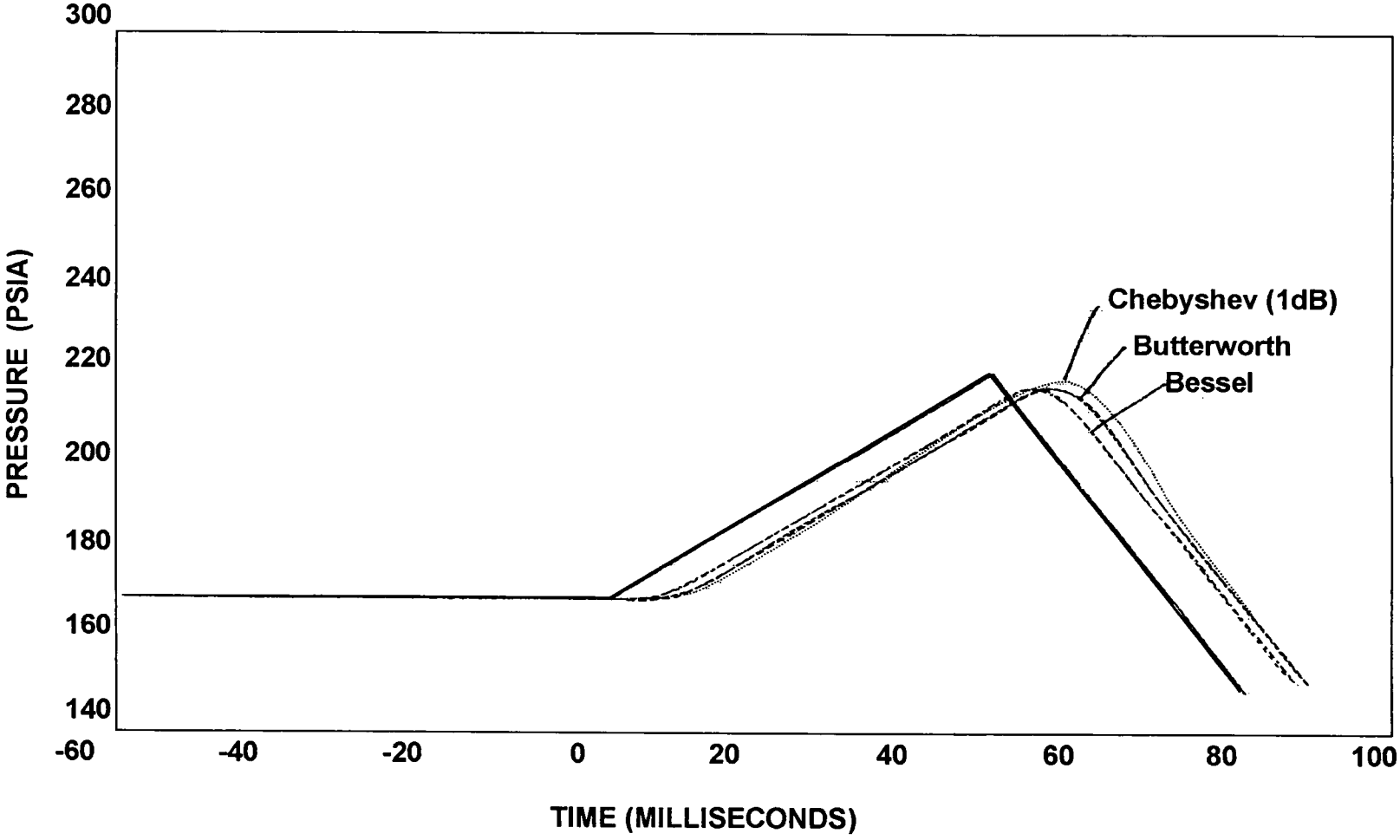


Fig. 3.11. Response of 50 Hz 4-Pole Filters to Triangular Input Signal

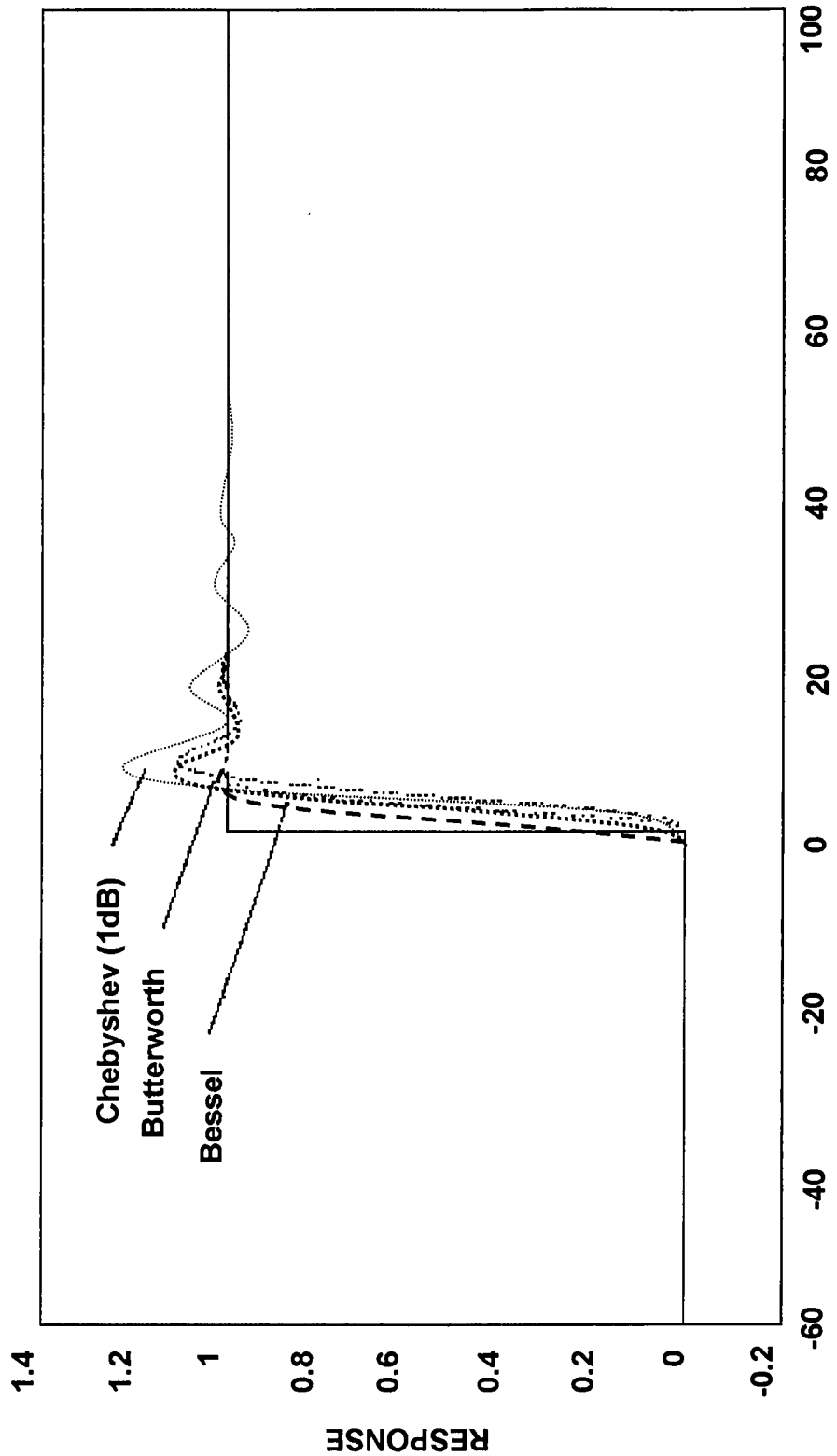


Fig. 3.12. Response of 100 Hz 4-Pole Filters to a Step Input Signal

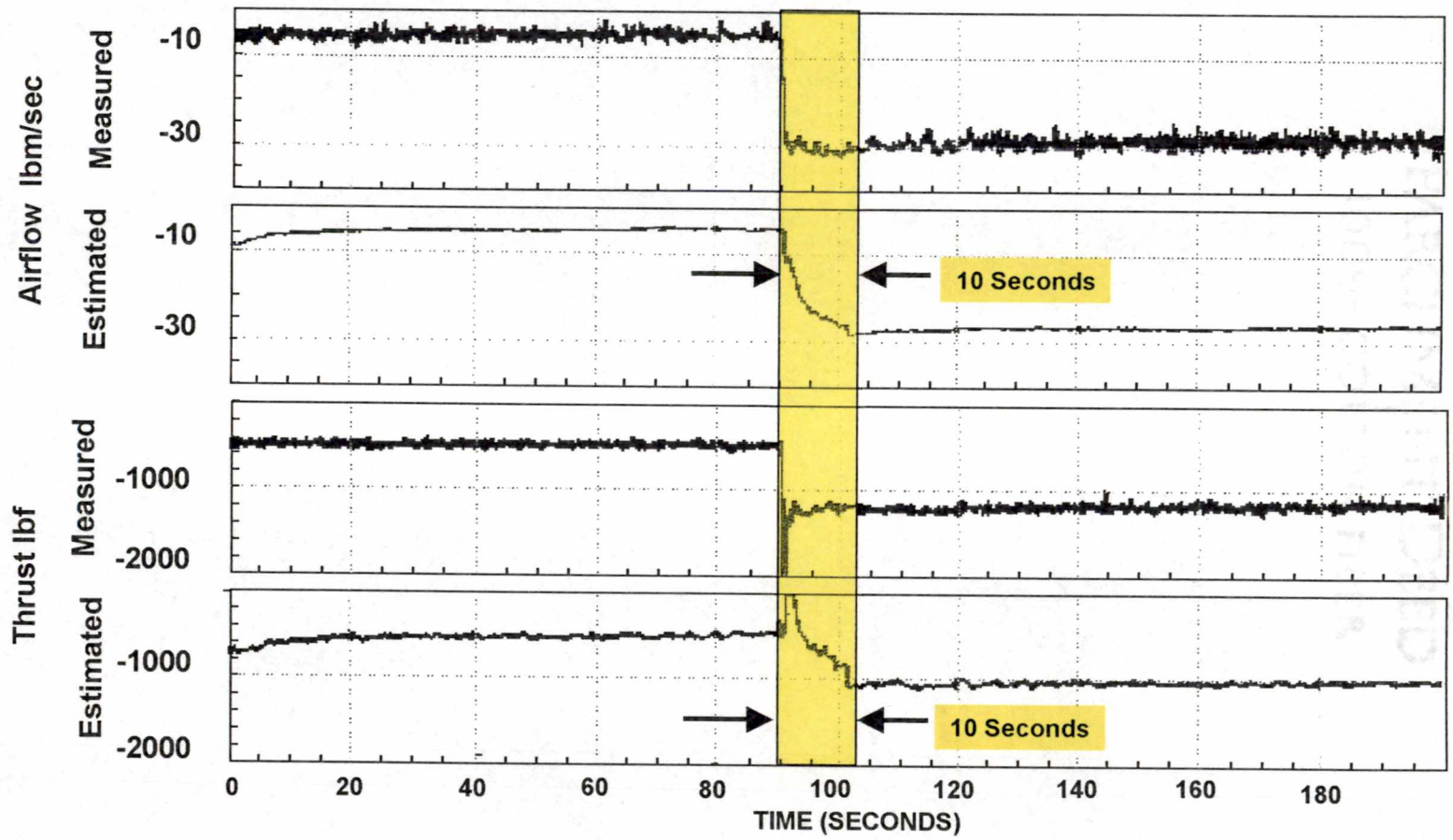


Fig. 3.13. Response of Kalman Filter Based Algorithm to Severe FOD Event

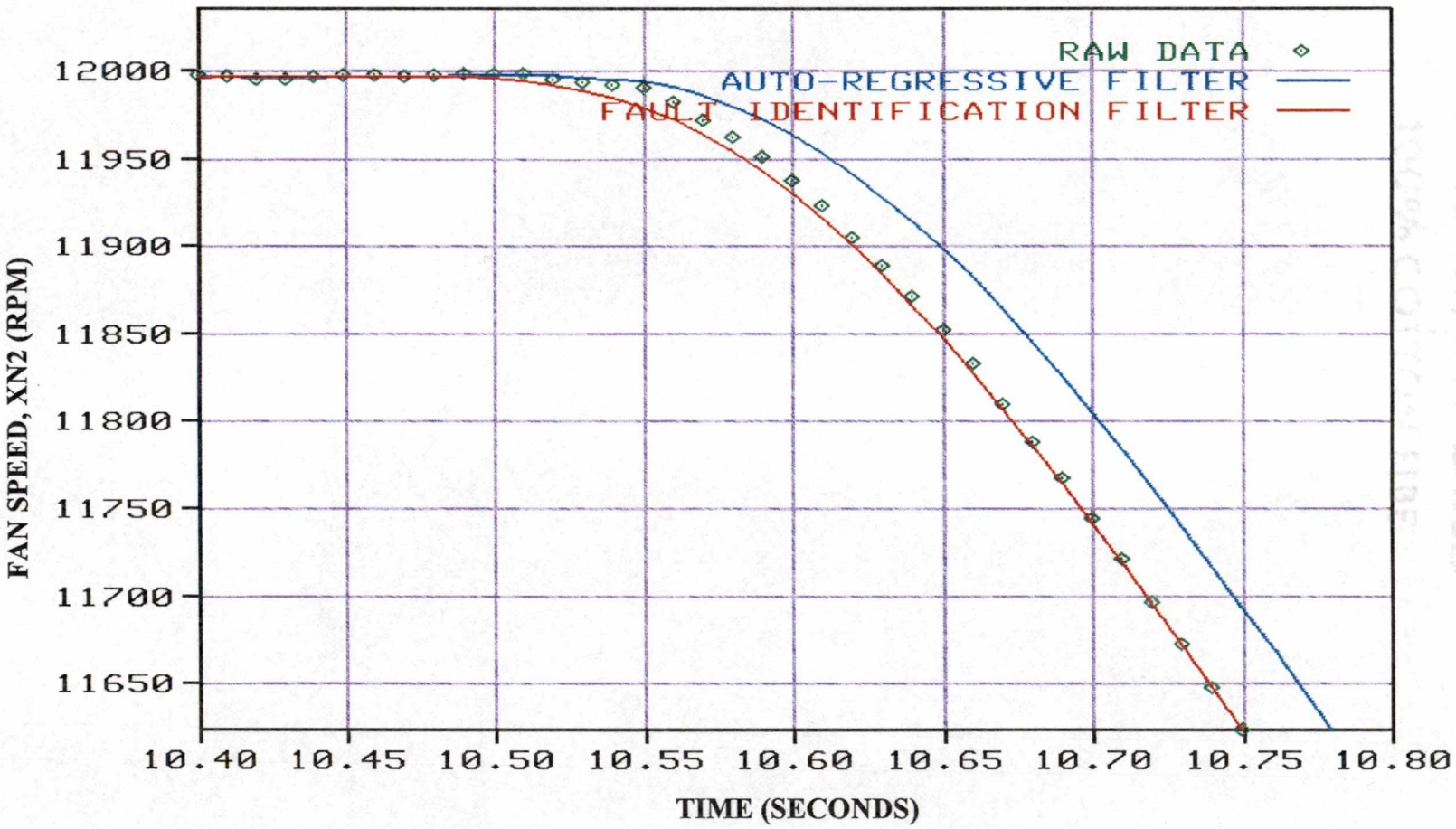


Fig. 3.14. Response of Digital Filtering Techniques to a Rapid Engine Deceleration

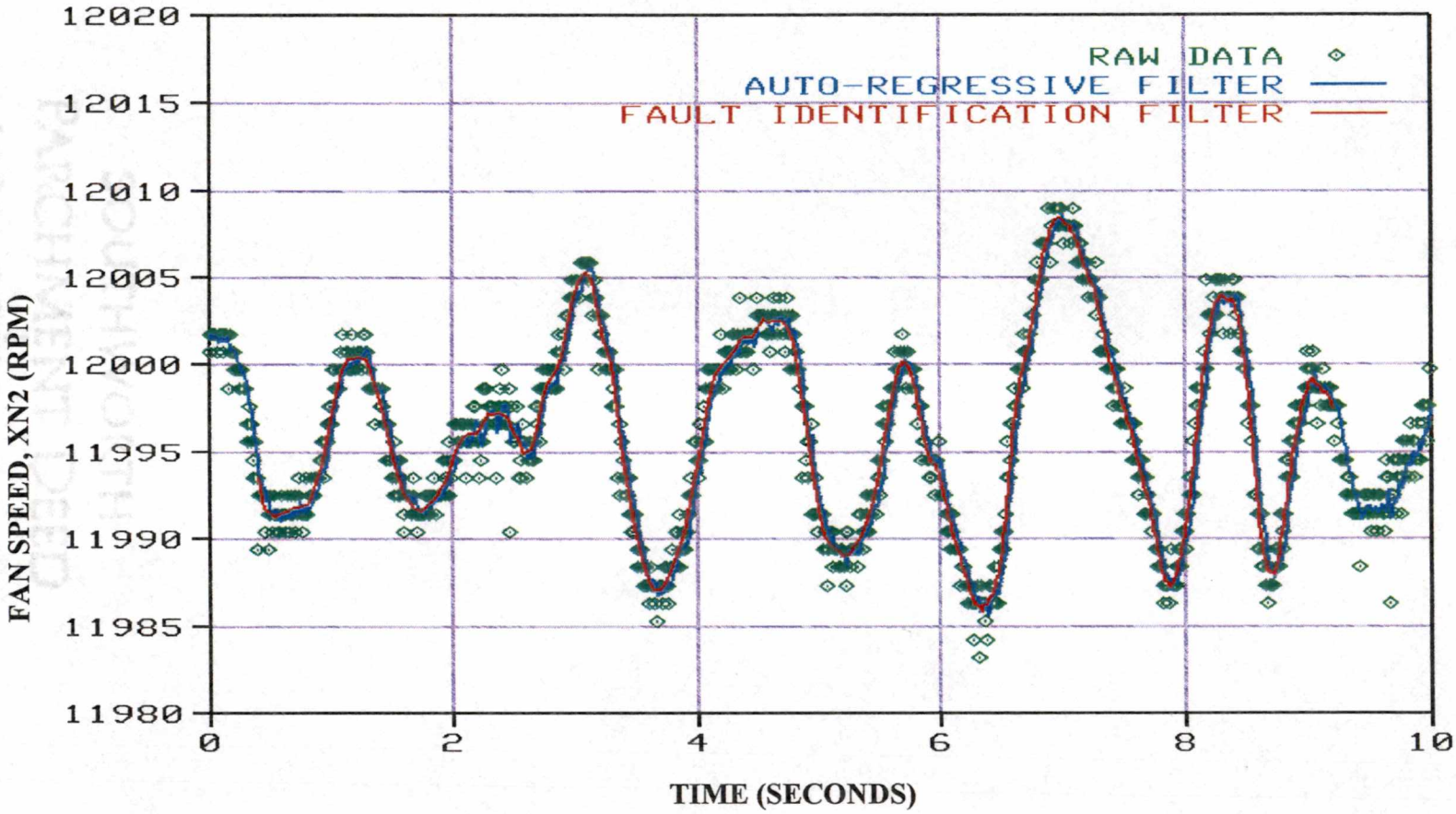
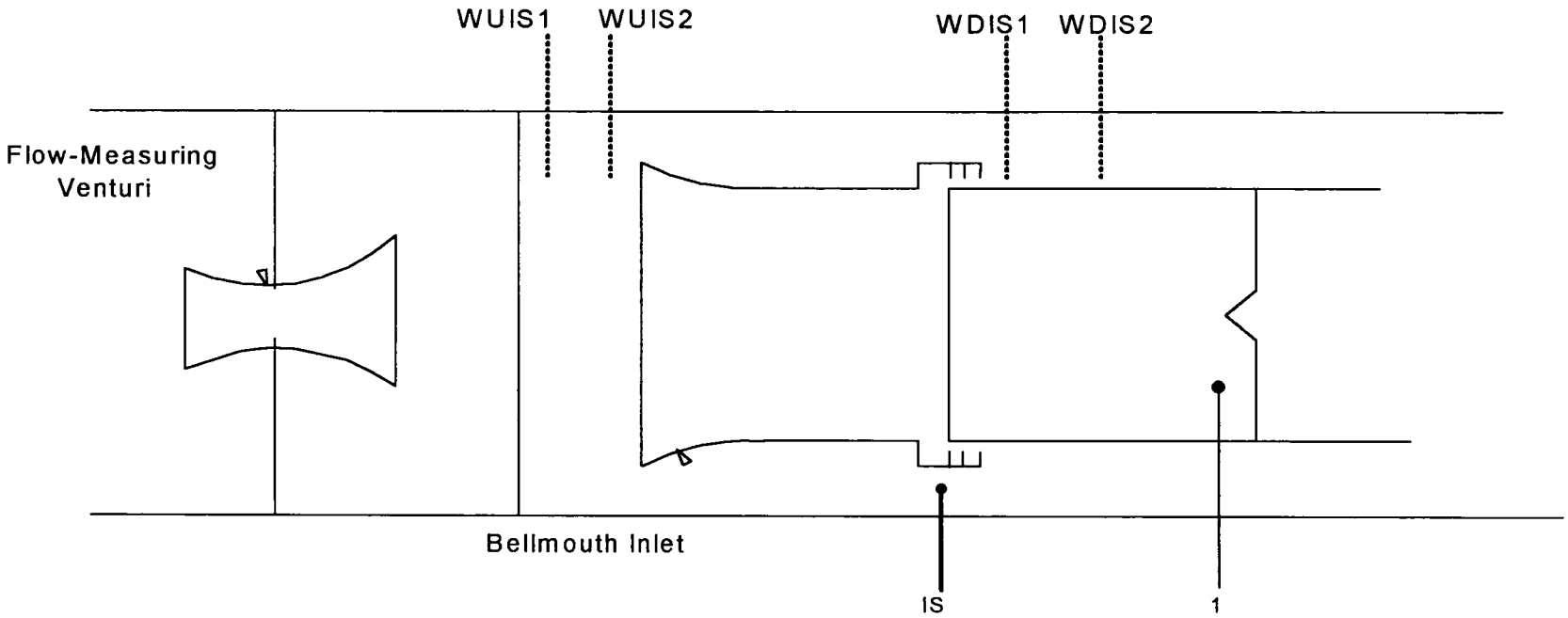


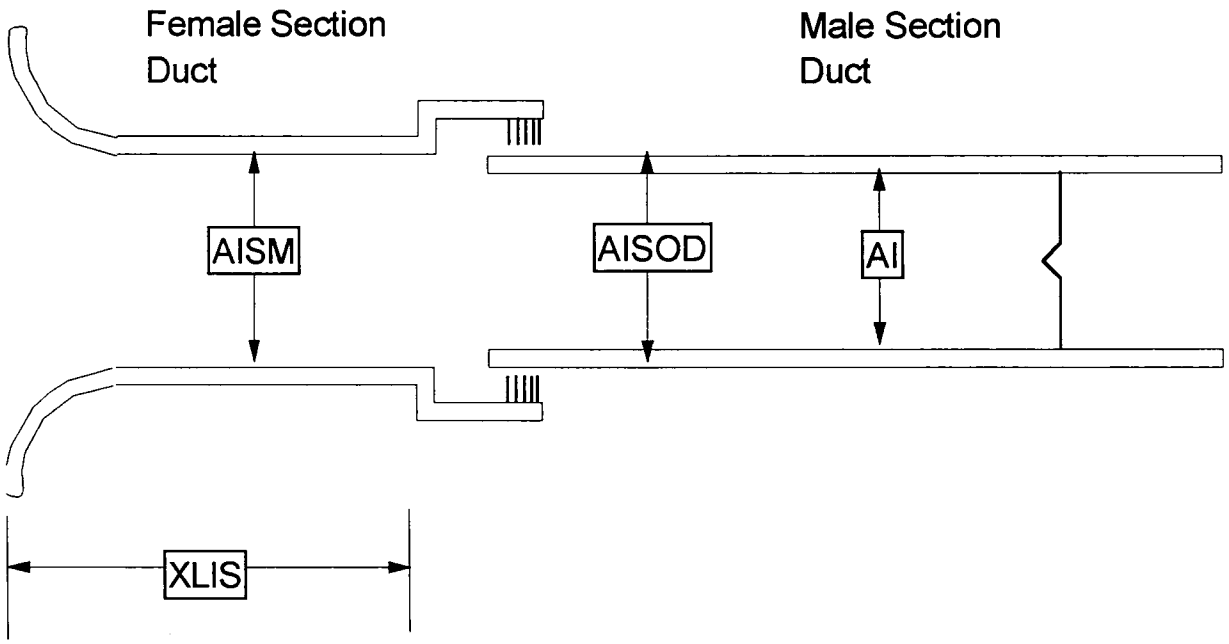
Fig. 3.15. Response of Digital Filtering Techniques to Moderate Fan Speed Variations



Station Number	Designation
IS	Isolation Seal Plane
1	Engine/Inlet Interface Plane

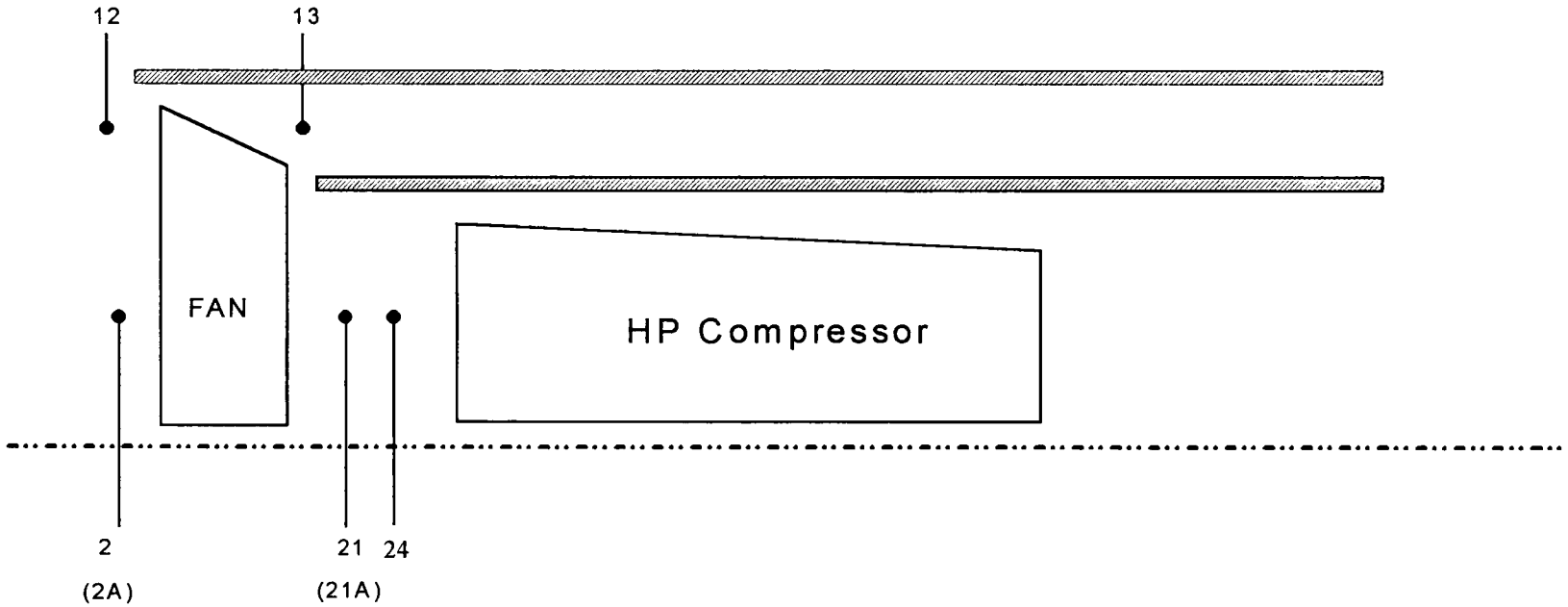
WUIS1	Secondary Mass Flow Addition or Extraction
WUIS2	Secondary Mass Flow Addition or Extraction
WDIS1	Secondary Mass Flow Addition or Extraction
WDIS2	Secondary Mass Flow Addition or Extraction

Fig. 3.16. Inlet Station Designation



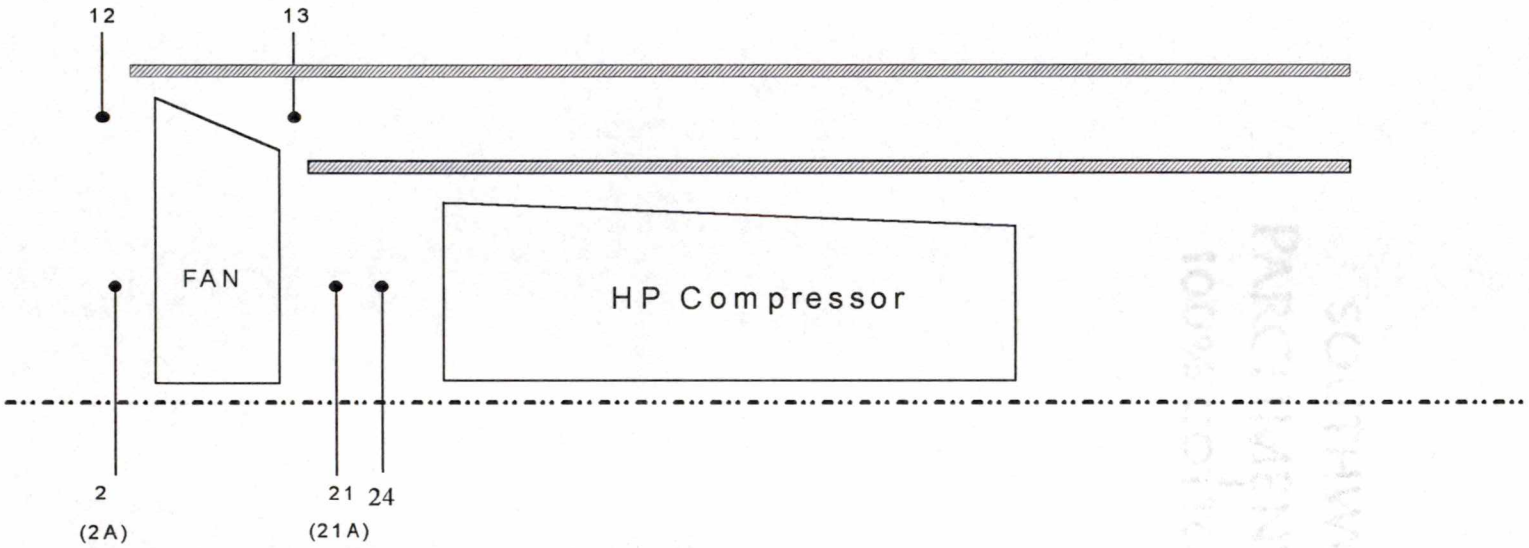
A1	Area of duct at inlet/engine interface plane
AISOD	Area of male isolation seal measured to outside diameter
AISM	Area of duct at PSIS measurement
XLIS	Distance from bellmouth inlet lip to isolation seal

Fig. 3.17. Isolation Seal Station Designation



Station Number	Designation
2A	Fan Inlet (Average)
2	Fan Inlet (Hub Section
12	Fan Inlet (Tip Section)
21A	Fan Discharge (Average)
24	Fan Discharge (Hub Section)
13	Fan Discharge (Tip Section)

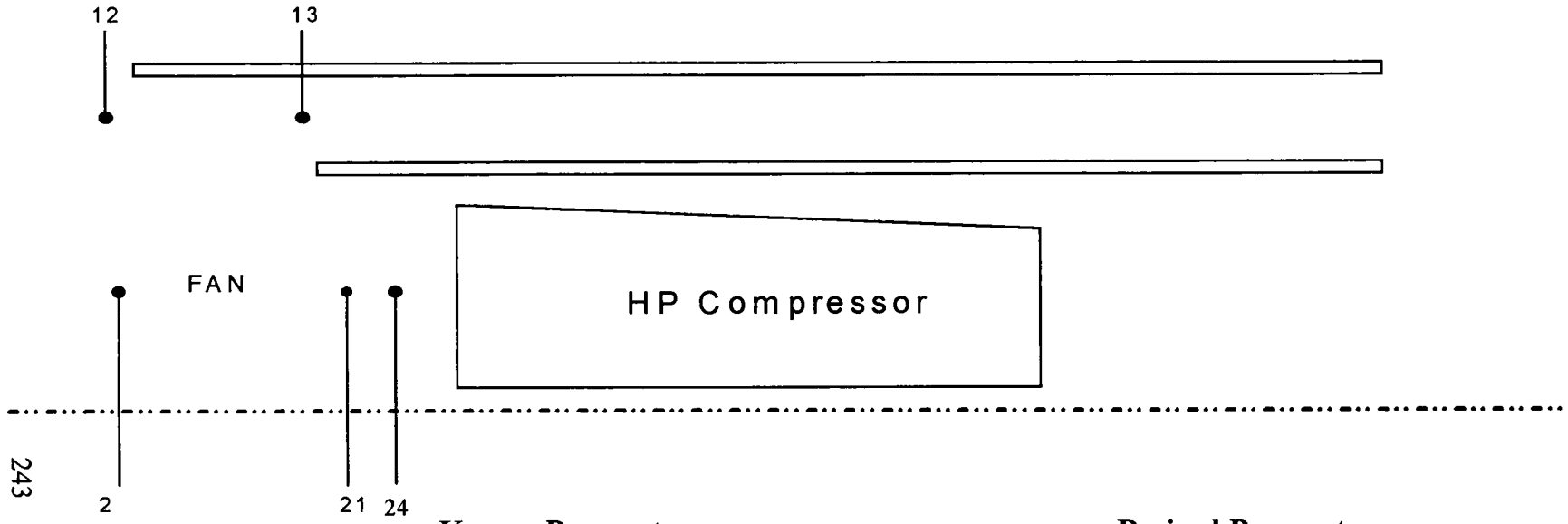
Fig. 3.18.1. Fan Station Designation for Twin-Spool Turbofan



Measured Parameters	+	Known Parameters	+	Assumptions	=	Desired Parameters
T_1, P_1, P_{PL} T_{12}, P_{12}, T_{13}, P_{13} T_2, P_2, T_{21}, P_{21} W_2, FVV, XN_2, WB_{13}		Boundary Layer Profile Secondary Flows Variable Vane Effects RNI_{2A} Effects Clearance Effects T_{24}, P_{24} WAR_{2A}, A_{2A}, $CQUL_{2A}$, $RFAW_{2A}$		CF_{2AX}		T_{12}, P_{12}, W_{12} T_{13}, P_{13}, W_{13} T_{2A}, P_{2A}, W_{2A} T_{21}, P_{21}, W_{21}
W_{2AX}, PS_{2A}, PS_{13} QP_{2X}, QT_{2A}, QT_{21}, QPS_2, QXN_2 WB_{13} Valve Position						

Model Inputs
Diagnostic Inputs

Fig. 3.18.2. Fan Analysis: Steady-State Parameters



Known Parameters

Transient Measurement Profile Scale Factors

δT_{12}, δP_{12}, δT_{13}, δP_{13},
δT_2, δP_2, δT_{21}, δP_{21}

Sensor Lag Time Constants

τT_2, τT_{12}, τT_{13}, τT_{21}

Fan Heat Soak Characteristics

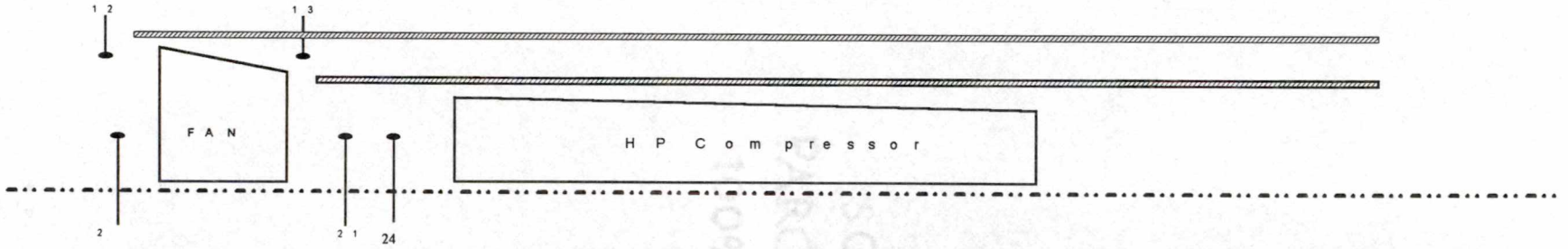
M_{Fan}, $CP_{m,\,Fan}$, h_{Fan}, $A_{w,\,Fan}$

Fan Metal Temperature - TM_{FAN}

Desired Parameters

T_{12}, P_{12}, W_{12}
T_{13}, P_{13}, W_{13}
T_{2A}, P_{2A}, W_{2A}
T_{21}, P_{21}, W_{21}

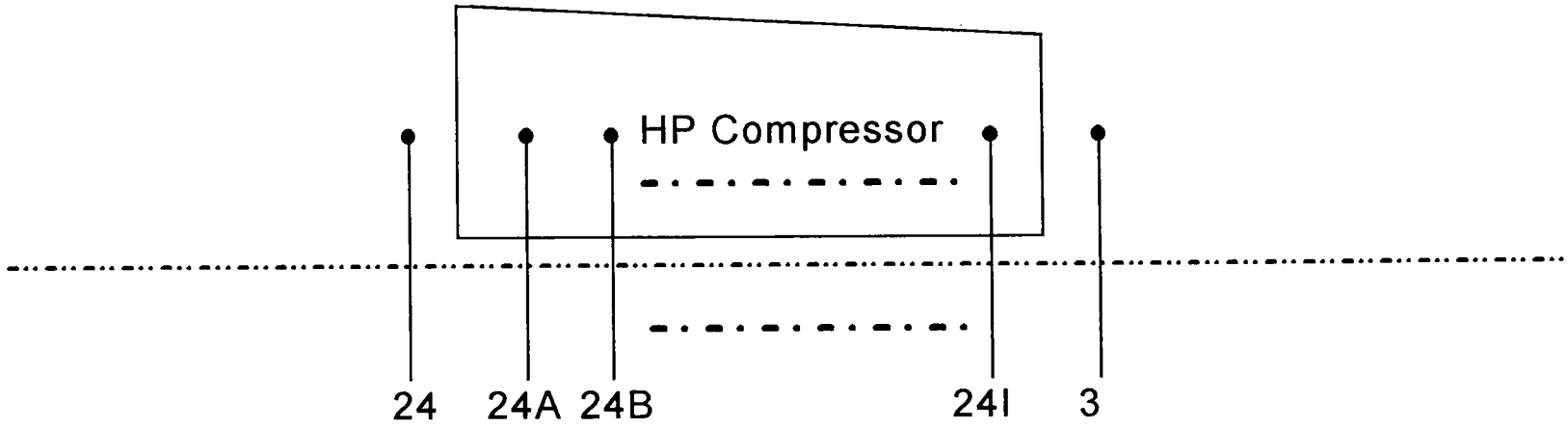
Fig. 3.18.3. Fan Analysis: Transient Parameters



PARAMETER	ADJUSTED MAP	TEST DATA	MODEL INPUT	DIAGNOSTIC INPUT	MAP ADJUSTMENT*
W_{2A}	X	X		*	Fan Flow Scalar ($\delta_{WC,FAN}$)
W_{2AX}		X		X	
P_{12}, P_2, P_{2A}		X	X		
P_1, P_{PL}, QP_{2X}		X		X	
PS_{2A}		X			
T_{12}, T_2, T_{2A}		X	X		
T_1, QT_{2A}		X		X	
W_{21}					
P_{13}, P_{21}	X	X	X	*	Pressure Ratio Scalar ($\delta_{P/P,FAN}$)
PS_{13}		X		X	
QPS_{13}		X		X	
T_{13}, T_{21}	X	X	X	*	Temp. Ratio Scalar ($\delta_{T/T,FAN}$)
QT_{21}		X		X	
FVV		X	X		
WB_{13}		X	X		
WB_{13} Valve Position		X		X	
XN_2		X	X		
QXN_2		X		X	

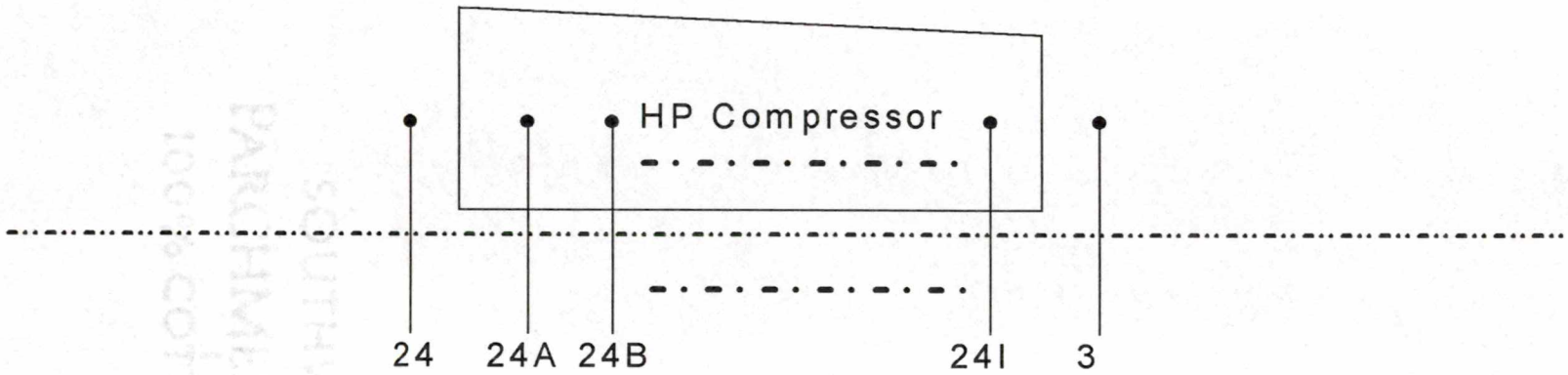
*** All Map Adjustment Parameters are Used as Diagnostic Inputs**

Fig. 3.18.4. Fan Parameters



Station Number	Designation
24	HPCompressor Inlet
24A	
24B	
●	Intermediate HPCompressor Stages Where Bleeds Are Present
●	
●	
24I	HP Compressor Discharge
3	

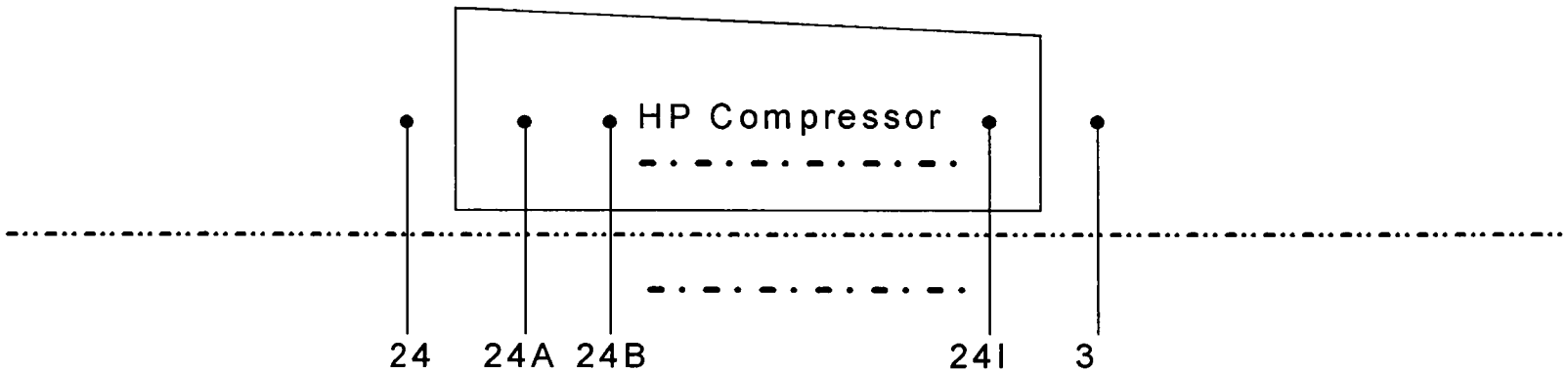
Fig. 3.19.1. High-Pressure Compressor Station Designation for Twin-Spool Turbofan



Measured Parameters	+	Known Parameters	+	Assumptions	=	Desired Parameters
T_{24}, P_{24} **T_3, P_3, CVV** **WB_3, XN_{24}** --- PS_{24}, PS_3 QXN_{24} , QPS_3 , QT_{24} WB_3 Valve Position		**Variable Vane Effects** **RN_{I24} Effects** **Clearance Effects** **A_{24}, $CQUL_{24}$, $RFAW_{24}$** **A_3, $CQUL_3$, $RFAW_3$** W_{24} , PS_{24}, PS_3		**Secondary Flows** **WB_{24A}, WB_{24B}, ...** **HB_{24A}, HB_{24B}, ...** **CF_{24X}, CF_{3X}**		**T_{24}, P_{24}, W_{24}** **T_3 , P_3 , W_3**

Model Inputs
Diagnostic Inputs

Fig. 3.19.2. High-Pressure Compressor: Steady-State Parameters



Known Parameters

Transient Measurement Profile Scale Factors

δT_{24}, δP_{24}, δT_3, δP_3

Sensor Lag Time Constants

τT_{24}, τT_3

HPC Heat Soak Characteristics

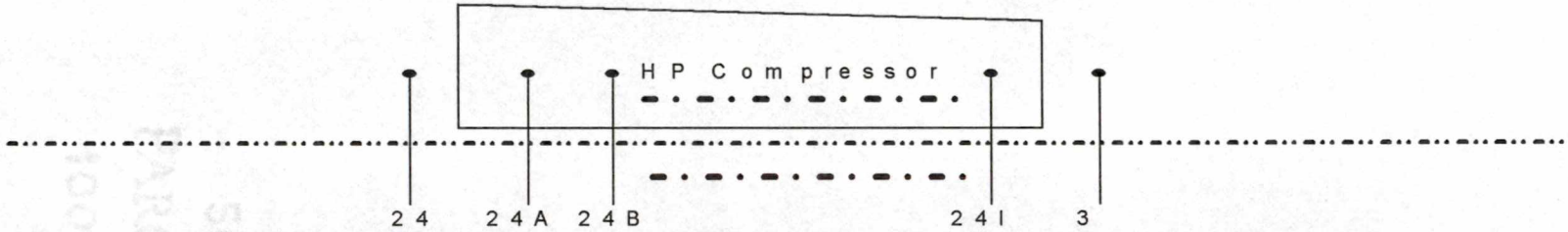
M_{HPC}, $CP_{m,\,HPC}$, h_{HPC}, $A_{w,\,HPC}$

HPC Metal Temperature - TM_{HPC}

Desired Parameters

T_{24}, P_{24}, W_{24}
T_3, P_3, W_3

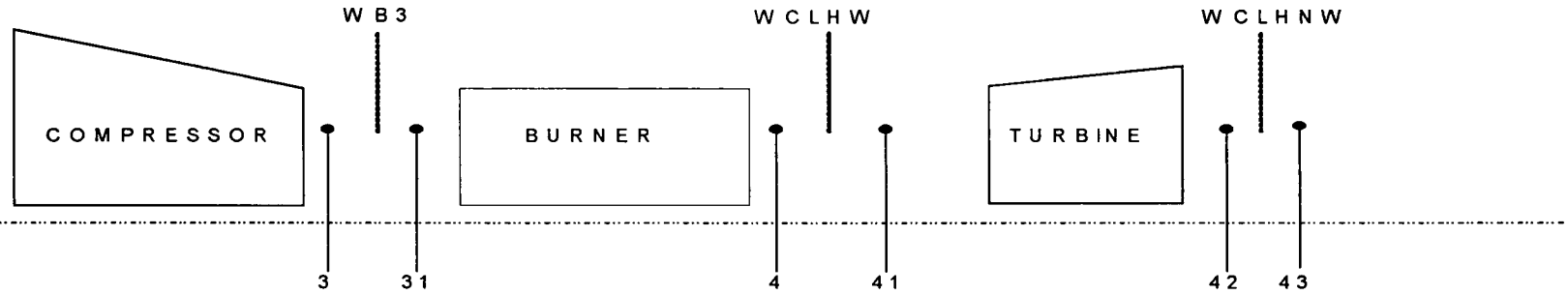
Fig. 3.19.3. High-Pressure Compressor Analysis: Transient Parameters



PARAMETER	ADJUSTED MAP	TEST DATA	MODEL INPUT	DIAGNOSTIC INPUT	MAP ADJUSTMENT*
W_{24}	X			*	HPC Flow Scalar ($^{d}_{WC,HPC}$)
W_{24AX}		X		X	
P_{24}		X	X		
PS_{24}		X		X	
T_{24}		X	X		
QT_{24}		X		X	
W_3					
W_{3ax}		X		X	
P_3	X	X		*	Pressure Ratio Scalar ($^{d}_{P/P,HPC}$)
PS_3		X		X	
QPS_3		X		X	
T_3	X	X	X	*	Temp. Ratio Scalar ($^{d}_{T/T,HPC}$)
CVV		X	X		
WB_3		X	X		
WB_3 Valve Position		X		X	
XN_{24}		X	X		
QXN_{24}		X		X	

*** All Map Adjustment Parameters are Used as Diagnostic Inputs**

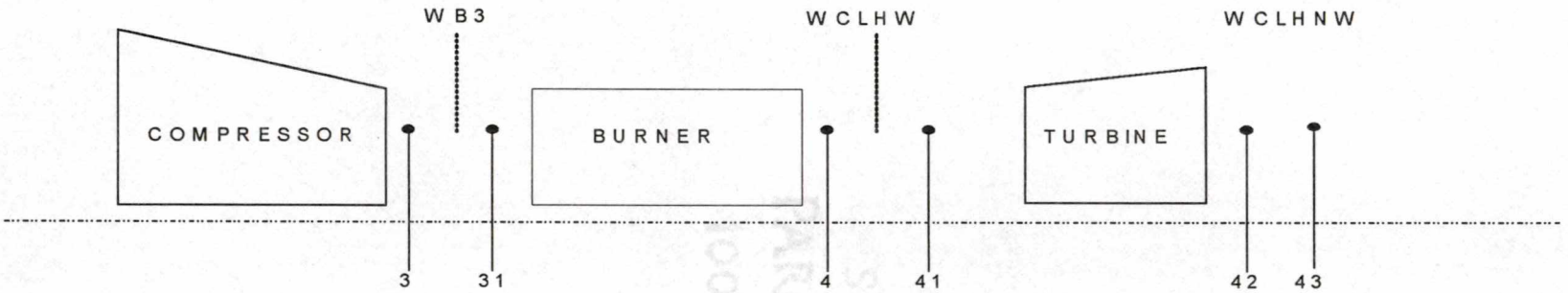
Fig. 3.19.4. High-Pressure Compressor Parameters



Station Number	Designation
3	HP Compressor Discharge
31	*
4	Burner Discharge
41	HP Turbine Nozzle Discharge
42	**
43	HP Turbine Discharge

*	Fictitious station created for compressor discharge/burner inlet for bleed bookkeeping
**	Fictitious station created for turbine cooling flow bookkeeping
WB3	HP Compressor discharge/burner inlet bleeds
WCLHW	HP Turbine cooling-flow available for LP turbine work.
WCLHNW	HP Turbine cooling-flow not available for LP turbine work.

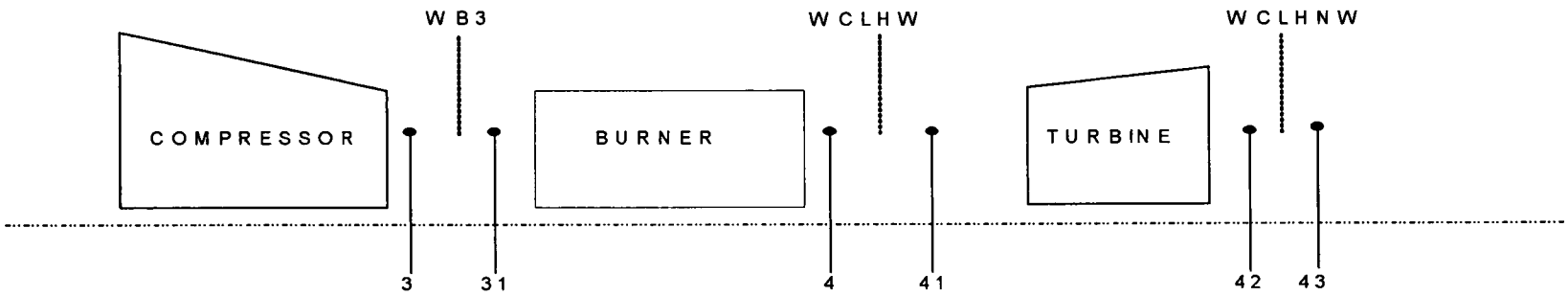
Fig. 3.20.1. Burner Station Designation for Twin-Spool Turbofan



Measured Parameters	+	Known Parameters	+	Assumptions	=	Desired Parameters
T_3, P_3, WFMC, WB_3 TFB, WFT, WFGG		T_{31}, P_{31} Fuel Characteristics FHV, FSG, VISTST		Burner Efficiency Burner Pressure Drop Secondary Flows		T_{31}, P_{31}, W_{31} T_4, P_4, W_4

Model Inputs
Diagnostic Inputs

Fig. 3.20.2. Burner Analysis: Steady-State Parameters



Known Parameters

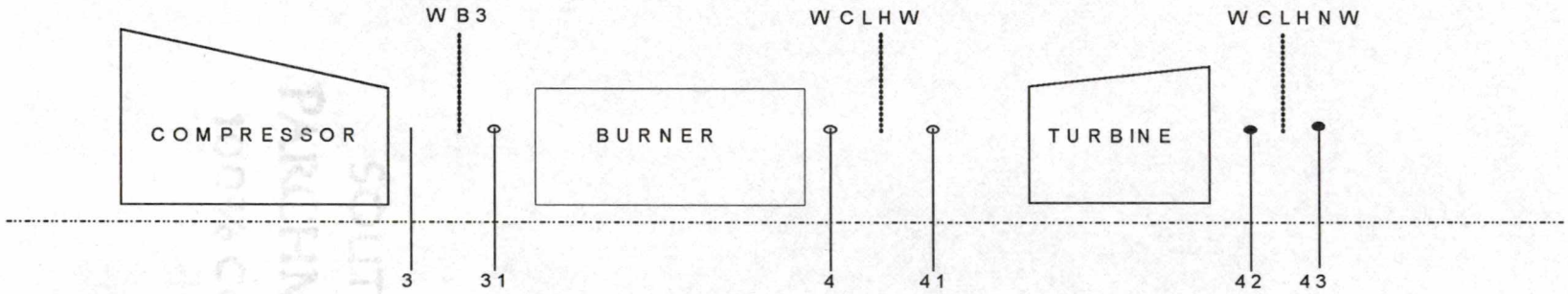
Burner Heat Soak Characteristics
M_{Burner}, $CP_{m,\ Burner}$, h_{Burner}, $A_{w,\ Burner}$

Burner Metal Temperature - TM_{Burner}

Desired Parameters

T_{31}, P_{31}, W_{31}
T_4, P_4, W_4

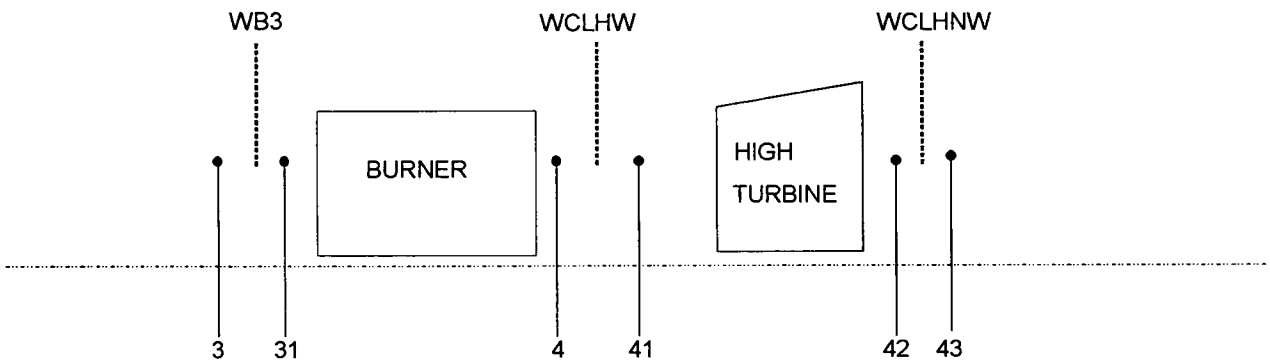
Fig. 3.20.3. Burner Analysis: Transient Parameters



PARAMETER	ADJUSTED MAP	TEST DATA	MODEL INPUT	DIAGNOSTIC INPUT	MAP ADJUSTMENT*
W_{31}					
P_{31}			X		
T_{31}			X		
TFB		X	X		
WFT		X		X	
WFMC		X	X		
WFGG		X		X	
FHV		X	X		
VISTST		X	X		
FSG		X	X		

*** All Map Adjustment Parameters are Used as Diagnostic Inputs**

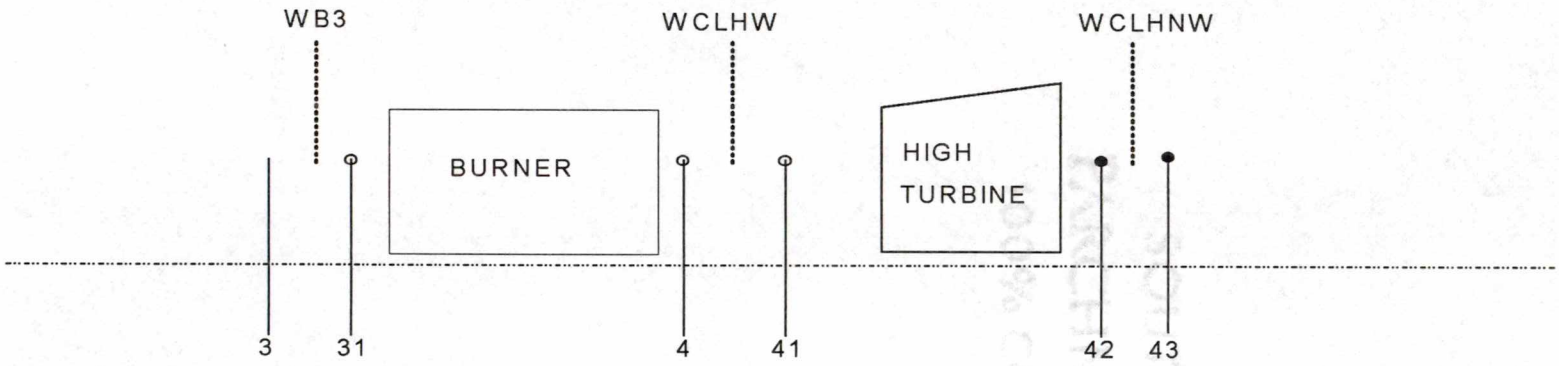
Fig. 3.20.4. Burner Parameters



Station Number	Designation
3	HP Compressor Discharge
31	*
4	Burner Discharge
41	HP Turbine Nozzle Discharge
42	**
43	HP Turbine Discharge

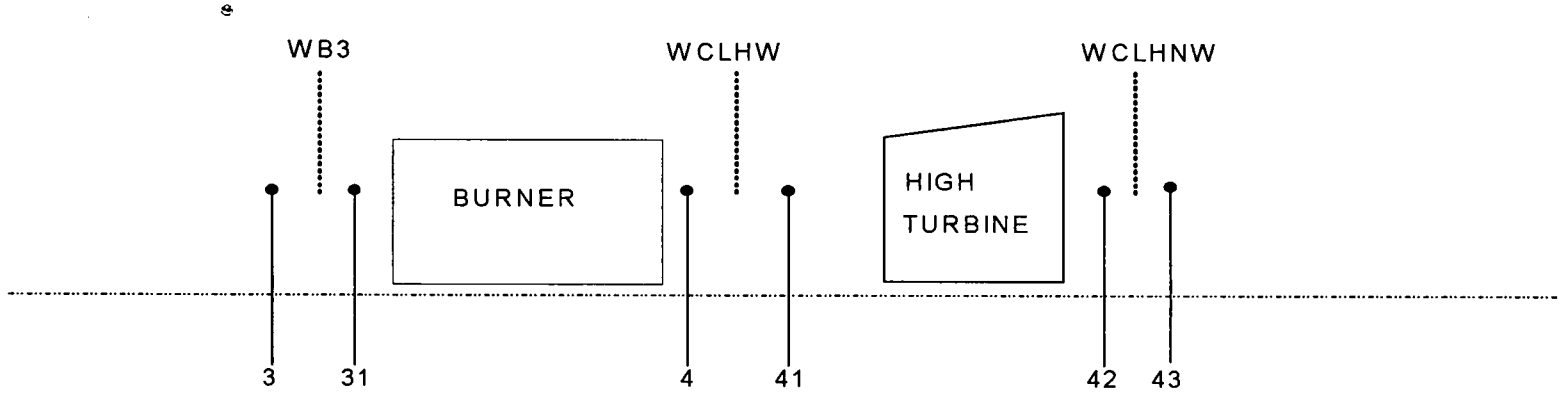
*	Fictitious station created for compressor discharge/burner inlet for bleed bookkeeping
**	Fictitious station created for HP Turbine cooling flow bookkeeping
WB3	HP compressor discharge/burner inlet bleeds
WCLHW	LP turbine cooling-flow available for LP turbine work.
WCLHNW	LP turbine cooling-flow not available for LP turbine work.

Fig. 3.21.1. High-Pressure Turbine Station Designation for Twin-Spool Turbofan



Measured Parameters	+	Known Parameters	+	Assumptions	=	Desired Parameters
Model Inputs: XN_{24}, PWX41 Diagnostic Inputs: QXN_{24}		Compressor Work T_{41}, P_{41} Clearance Effects RNI_{41} Effects		PWL_{41} A_{41T}, W_{41FFX} WC_{LHW}, WHC_{HW} WC_{LHNW}, WHC_{HNW}		T_{41}, P_{41}, W_{41} T_{42}, P_{42}, W_{42}

Fig. 3.21.2. High-Pressure Turbine Analysis: Steady-State Parameters



Known Parameters

High Spool Moment of Inertia

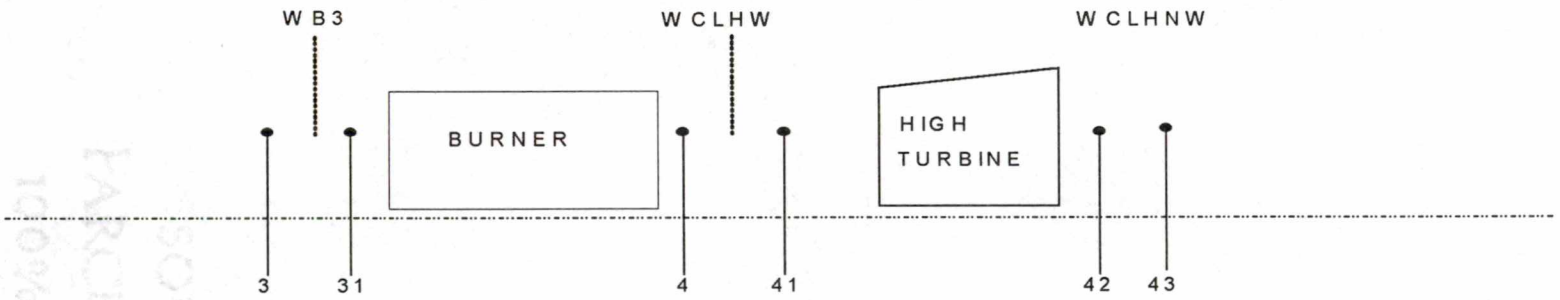
HPT Heat Soak Characteristics
M_{HPT}, $CP_{m,\,HPT}$, h_{HPT}, $A_{w,\,HPT}$

HPT Metal Temperature - TM_{HPT}

Desired Parameters

T_{41}, P_{41}, W_{41}
T_{42}, P_{42}, W_{42}

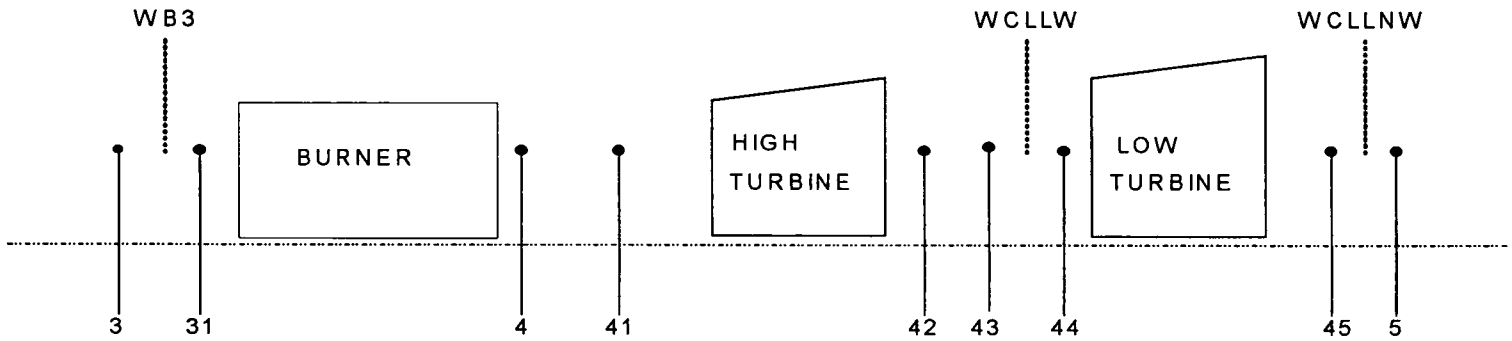
Fig. 3.21.3. High-Pressure Turbine Analysis: Transient Parameters



PARAMETER	ADJUSTED MAP	TEST DATA	MODEL INPUT	DIAGNOSTIC INPUT	MAP ADJUSTMENT*
W_{41}					
P_{41}					
T_{41}					
W_{42}					
P_{42}					
T_{42}	X			*	Temp. Ratio Scalar ($d_{T/T1HPT}$)
PWX_{41}		X	X		
XN_{21}		X	X		
QXN_{21}		X		X	

*** All Map Adjustment Parameters are Used as Diagnostic Inputs**

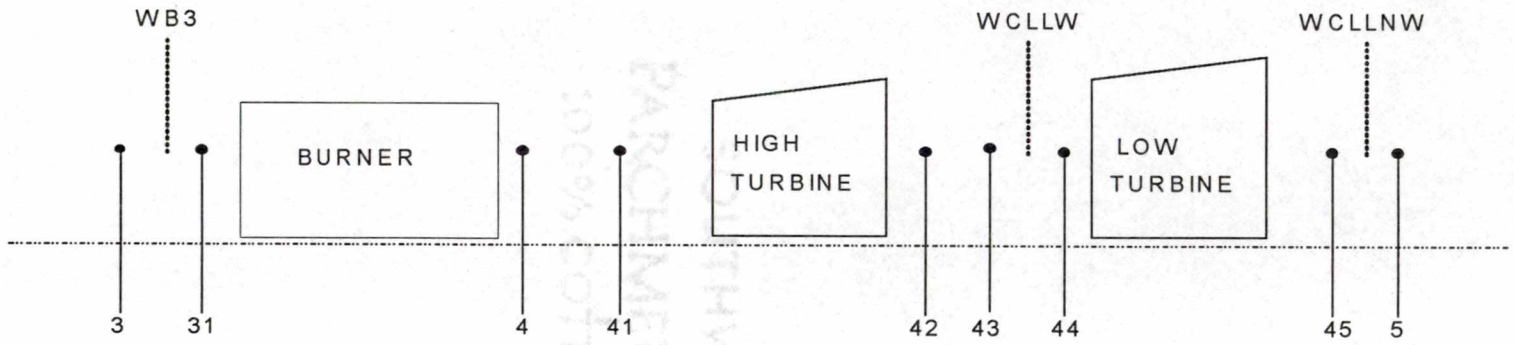
Fig. 3.21.4. High-Pressure Turbine Parameters



Station Number	Designation
3	HP Compressor Discharge
*	*
4	Burner Discharge
41	HP Turbine Nozzle Discharge
42	**
43	HP Turbine Discharge
44	LP Turbine Inlet
45	****
5	LP Turbine Discharge

*	Fictitious station created for compressor discharge/burner inlet for bleed bookkeeping
**	Fictitious station created for HP Turbine cooling flow bookkeeping
***	Fictitious station created for HP Turbine cooling flow bookkeeping
WB3	HP compressor discharge/burner inlet bleeds
WCLLW	LP turbine cooling-flow available for LP turbine work.
WCLLNW	LP turbine cooling-flow not available for LP turbine work.

Fig. 3.22.1. Low-Pressure Turbine Station Designation for Twin-Spool Turbofan

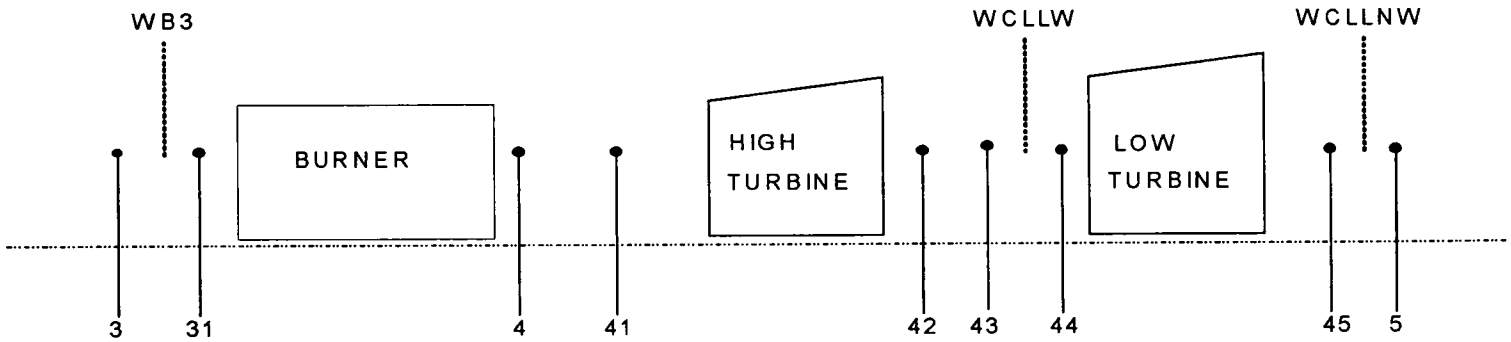


Measured Parameters + **Known Parameters** + **Assumptions** = **Desired Parameters**

Measured Parameters	Known Parameters	Assumptions	Desired Parameters
XN_2	**Fan Work**	**PWL_{44}**	**T_{44}, P_{44}, W_{44}**
T_5, QT_5, QXN_2,	**T_{44}, P_{44}**	**A_{44T}, W_{44FFX}**	**T_{45}, P_{45}, W_{45}**
	T_{45}, P_{45}	**WC_{LLW}, WHC_{LW}**	
	Clearance Effects	**WC_{LLNW}, WHC_{LNW}**	
	RNI_{44} Effects		

Model Inputs
Diagnostic Inputs

Fig. 3.22.2. Low-Pressure Turbine Analysis: Steady-State Parameters



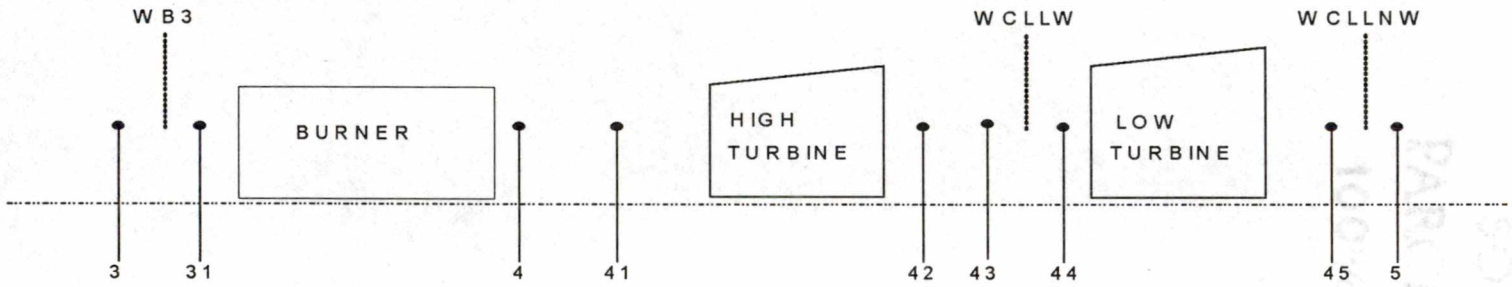
Known Parameters

Transient Profiles
T_5
Sensor Lags
T_5
LP Spool Moment of Inertia
LPT Heat Soak Characteristics
Metal Temperature - TM_{LPT}

Desired Parameters

T_{44}, P_{44}, W_{44}
T_{45}, P_{45}, W_{45}

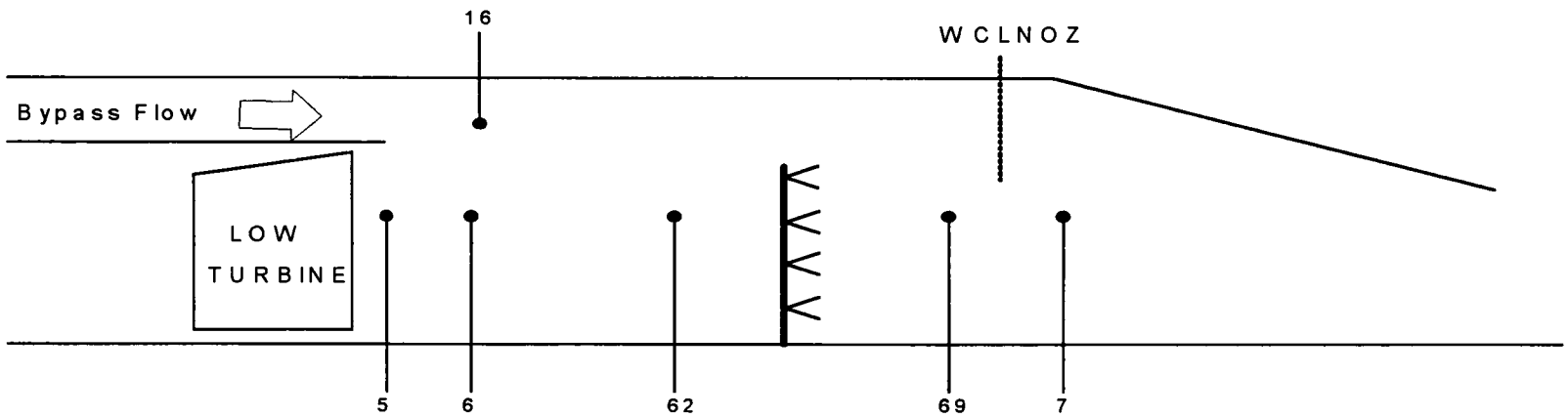
Fig. 3.22.3. Low-Pressure Turbine Analysis: Transient Parameters



PARAMETER	ADJUSTED MAP	TEST DATA	MODEL INPUT	DIAGNOSTIC INPUT	MAP ADJUSTMENT*
W_{44}					
P_{44}					
T_{44}					
W_{45}					
P_{45}					
T_{45}	X			*	Temp. Ratio Scalar ($\delta_{T/T,LPT}$)
T_5		X		X	
QT_5		X		X	
XN_2		X	X		
QXN_2		X		X	

*** All Map Adjustment Parameters are Used as Diagnostic Inputs**

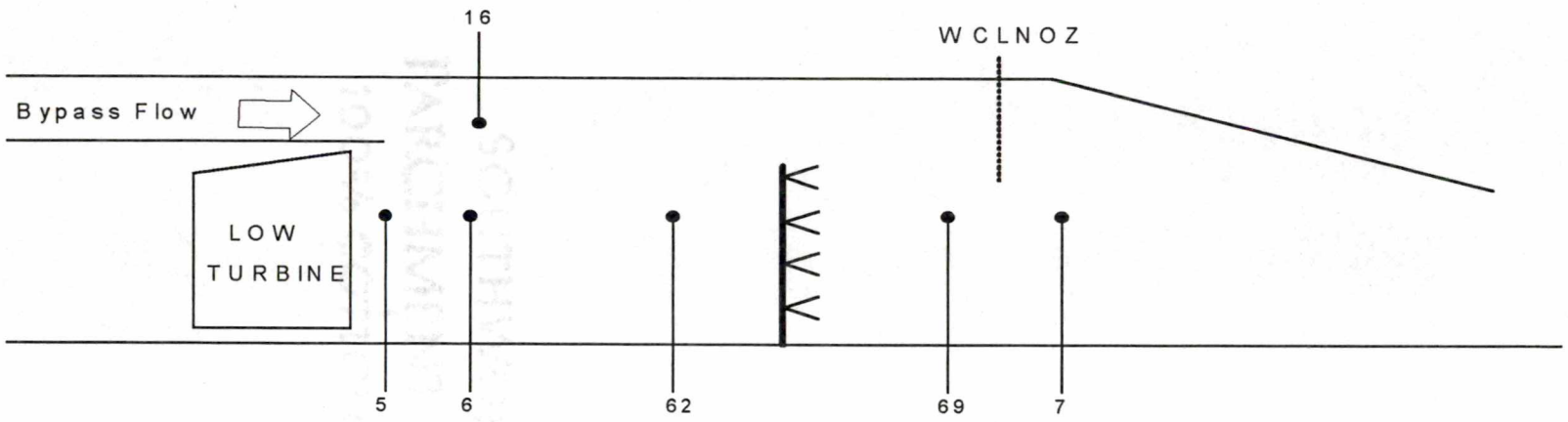
Fig. 3.22.4. Low-Pressure Turbine Parameters



Station Number	Designation
5	LP Turbine Discharge
6	Mixer Inlet (Core Flow)
16	Mixer Inlet (Bypass Flow)
62	Augmentor Inlet
69	*
7	Engine/Exhaust Nozzle Interface

* Fictitious station created for nozzle cooling-flow bookkeeping
WCLNOZ Nozzle cooling flow

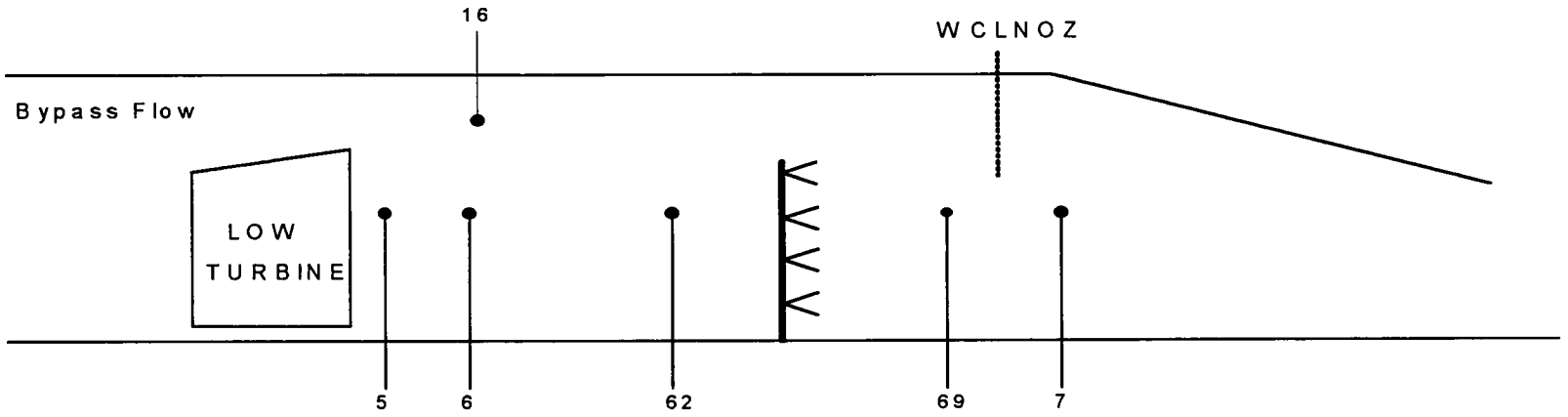
Fig. 3.23.1. Mixer Station Designation for Twin-Spool Turbofan



Measured Parameters	+	Known Parameters	+	Assumptions	=	Desired Parameters
T_{16} P_{16}, T_5, P_5		**Bypass Duct Pressure Loss** W_5, W_{16}, PS_{62} T_5, P_5 T_{16}, P_{16} AE_{16}, AE_6		**Bypass Duct Thermal Energy Absorption**		T_{62}, P_{62}, W_{62}

Model Inputs
Diagnostic Inputs

Fig. 3.23.2. Mixer Analysis: Steady-State Parameters



Known Parameters

Heat Transfer Characteristics

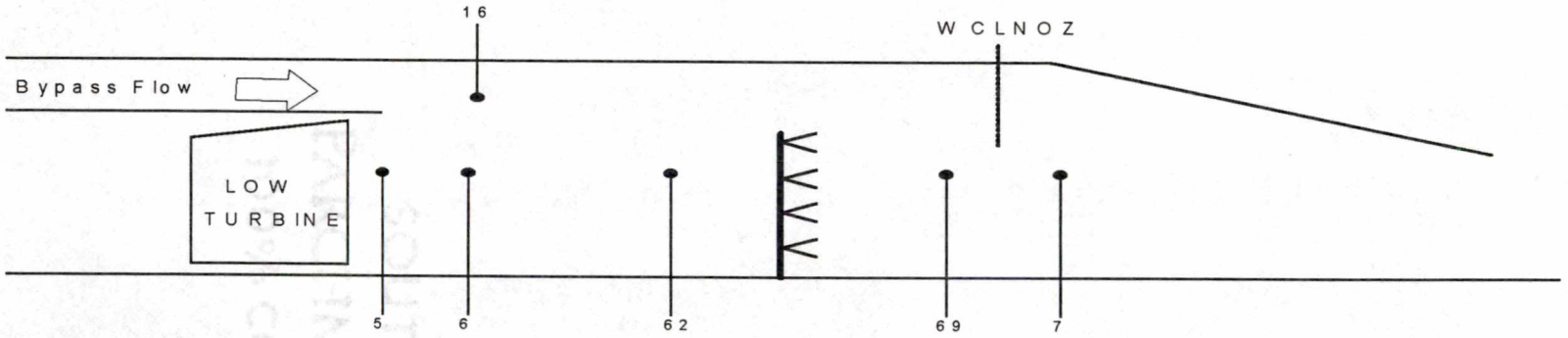
Effective Mixing Areas

- **AE6**
- **AE16**

Desired Parameters

T_{62}, PT_{62}, W_{62}

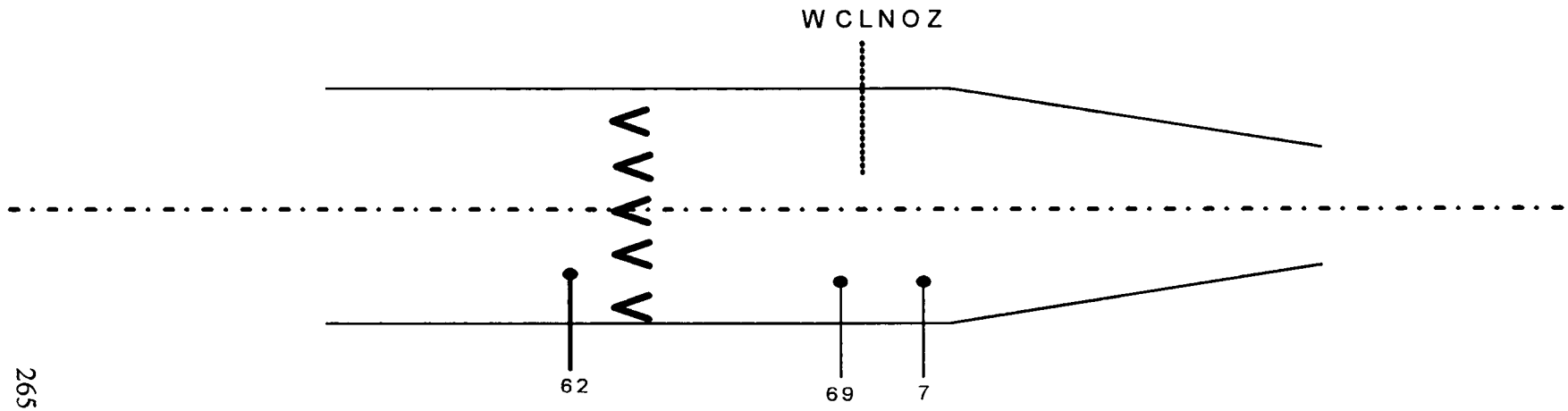
Fig. 3.23.3. Mixer Analysis: Transient Parameters



PARAMETER	ADJUSTED MAP	TEST DATA	MODEL INPUT	DIAGNOSTIC INPUT	MAP ADJUSTMENT*
W_6					
P_6					
T_6					
W_{16}					
P_{16}		SS only		*	Duct Pressure Loss Scale Factor
T_{16}		X	X		
W_{62}					
P_{62}					
T_{62}					

*** All Map Adjustment Parameters are Used as Diagnostic Inputs**

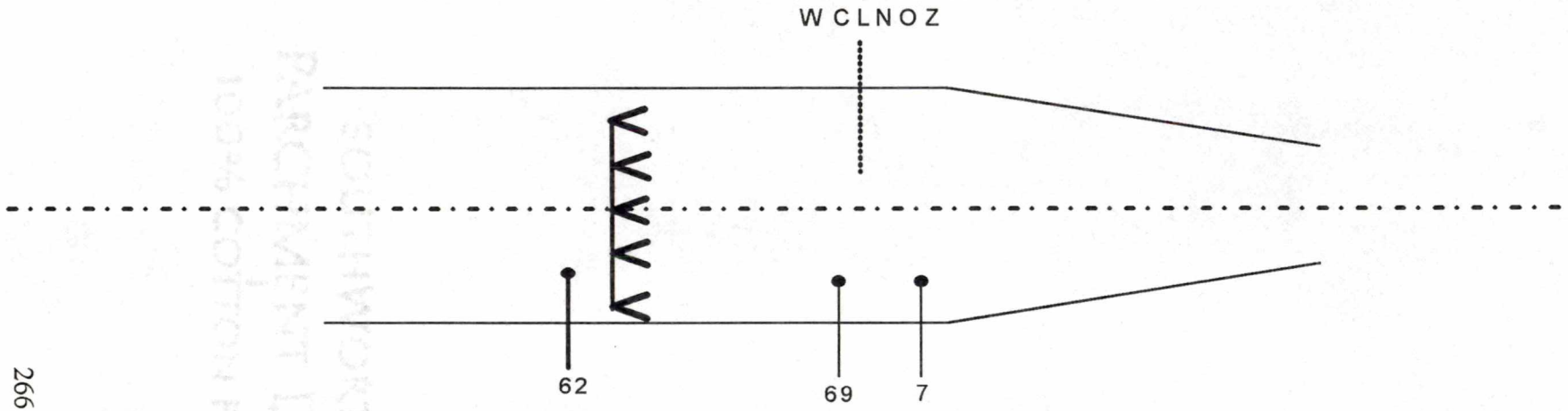
Fig. 3.23.4. Mixer Parameters



Station Number	Designation
62	Augmentor Inlet (Core Flow)
69	*
7	Engine/Exhaust Nozzle Interface

* Fictitious station created for nozzle cooling-flow bookkeeping
WCLNOZ Nozzle cooling flow

Fig. 3.24.1. Augmentor Station Designation for Twin-Spool Turbofan



Measured Parameters	+	Known Parameters	+	Assumptions	=	Desired Parameters
WFAB, *WCLNOZ*		**Reheat Pressure Loss** T_{69}, P_{69}		**Augmentor Efficiency**		T_{62}, P_{62}, W_{62} T_7, P_7, W_7

Model Inputs
Diagnostic Inputs

Fig. 3.24.2. Augmentor Analysis: Steady-State Parameters

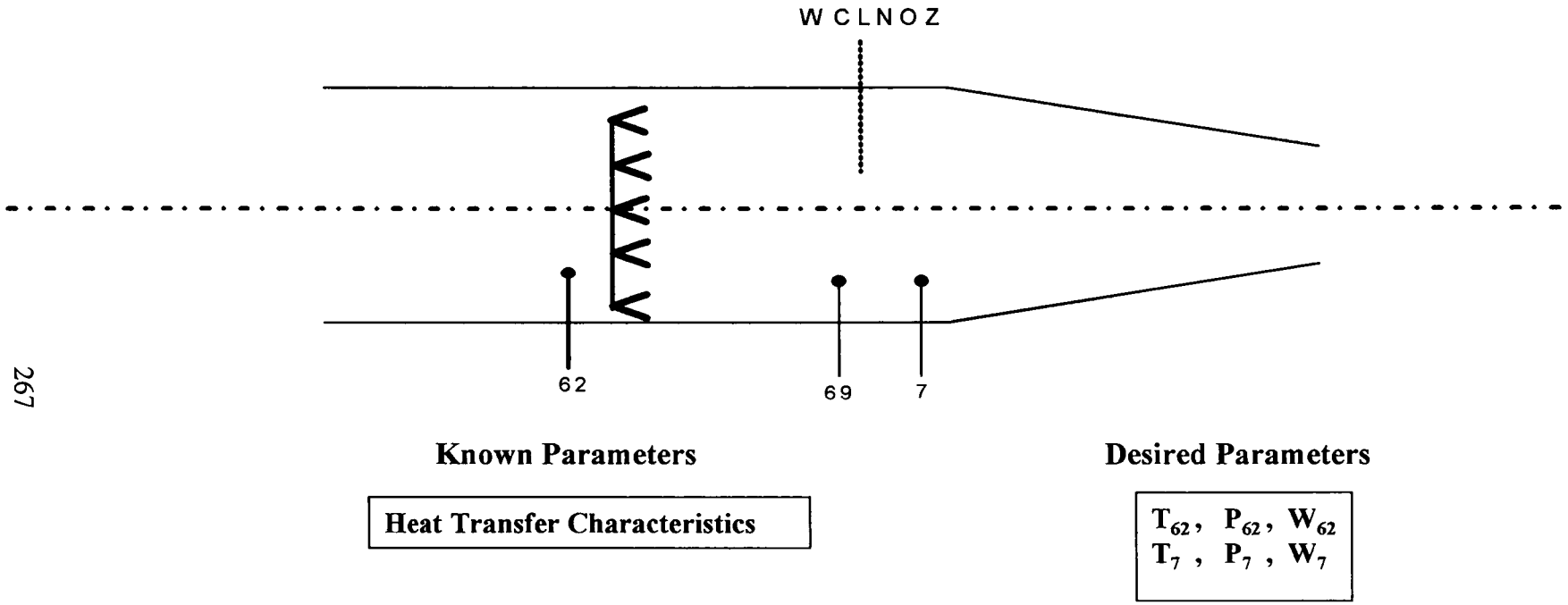
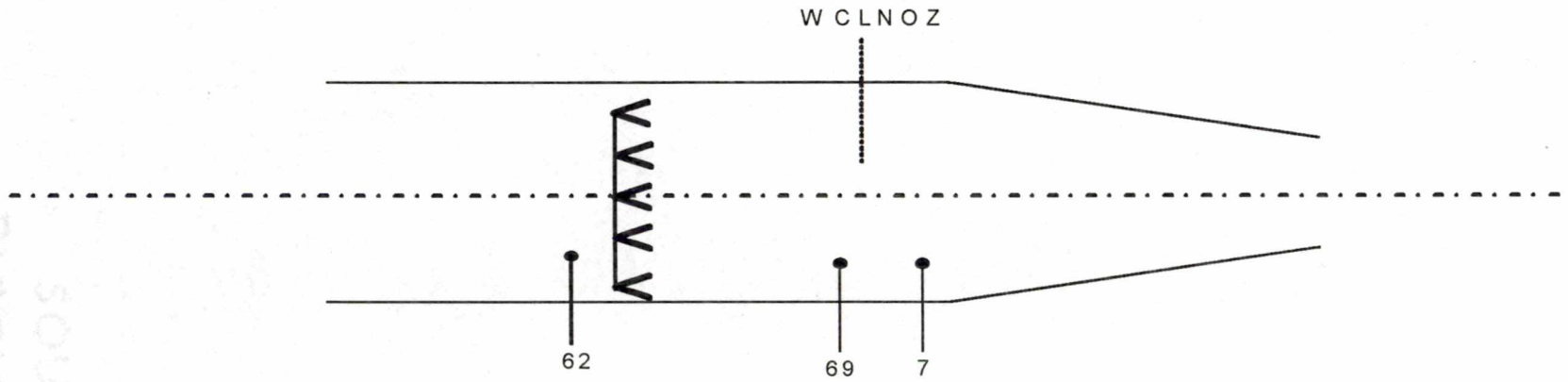
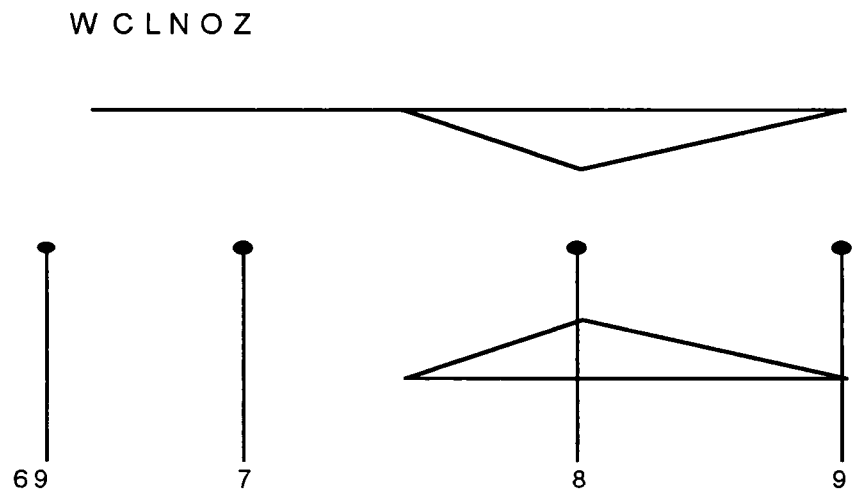


Fig. 3.24.3. Augmentor Analysis: Transient Parameters



PARAMETER	ADJUSTED MAP	TEST DATA	MODEL INPUT	DIAGNOSTIC INPUT	MAP ADJUSTMENT*
W_{62}					
P_{62}					
T_{62}					
WFAB		X	X		
W_7					
P_7					
T_7					

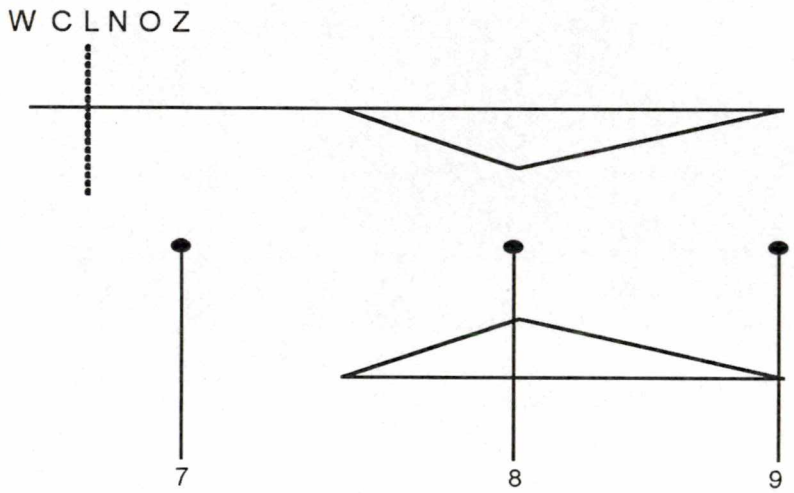
Fig. 3.24.4. Augmentor Parameters



Station Number	Designation
69	*
7	Engine/Exhaust Nozzle Interface
8	Nozzle Throat (CD)
9	Nozzle discharge

* Fictitious station created fornozzle coolingflow bookkeeping
WCLNOZ Nozzle cooling flow

Fig. 3.25.1. Nozzle Station Designation for Twin-Spool Turbofan

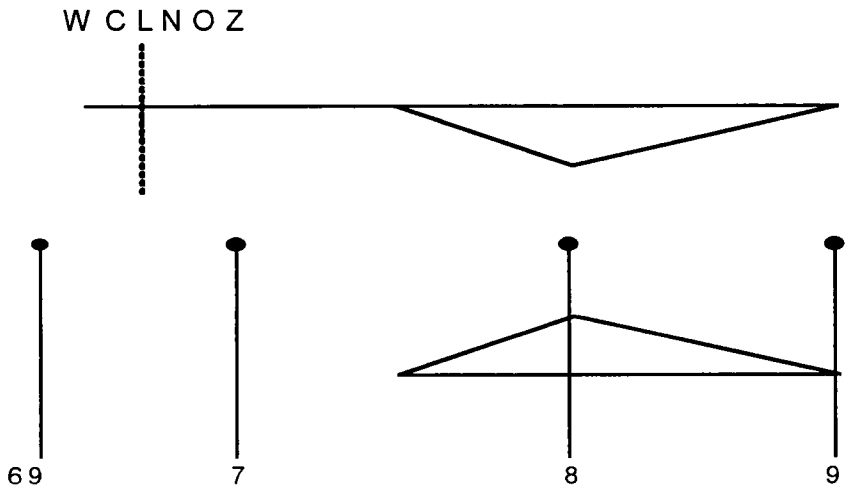


Measured Parameters	+	Known Parameters	+	Assumptions	=	Desired Parameters
A_8		A_9, CF_8, CFG_9		1-D Flow		T_7, P_7, W_7 T_9, P_9, W_9 FG
FG, SFC						

Model Inputs
Diagnostic Inputs

Fig. 3.25.2. Nozzle Analysis: Steady-State Parameters

PARTMENT OF
SOUTH DAKOTA
270



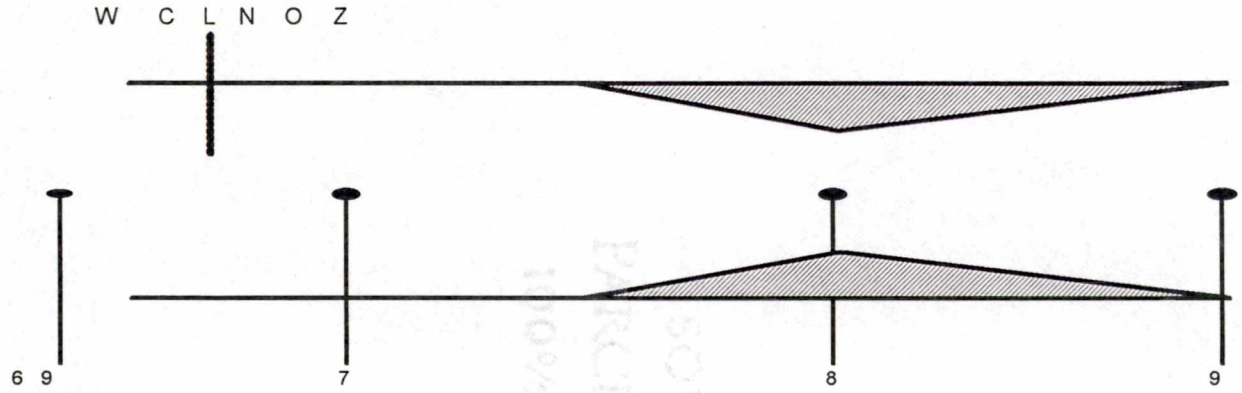
Known Parameters

Heat Transfer Characteristics

Desired Parameters

T_7, P_7, W_7
T_9, P_9, W_9
FG

Fig. 3.25.3. Nozzle Analysis: Transient Parameters



PARAMETER	ADJUSTED MAP	TEST DATA	MODEL INPUT	DIAGNOSTIC INPUT	MAP ADJUSTMENT*
W_7					
P_7					
T_7					
A_8		X	X		
A_9		X	X		
W_9					
P_9					
T_9					
FG		X		X	
SFC		X		X	

Fig. 3.25.4. Nozzle Parameters

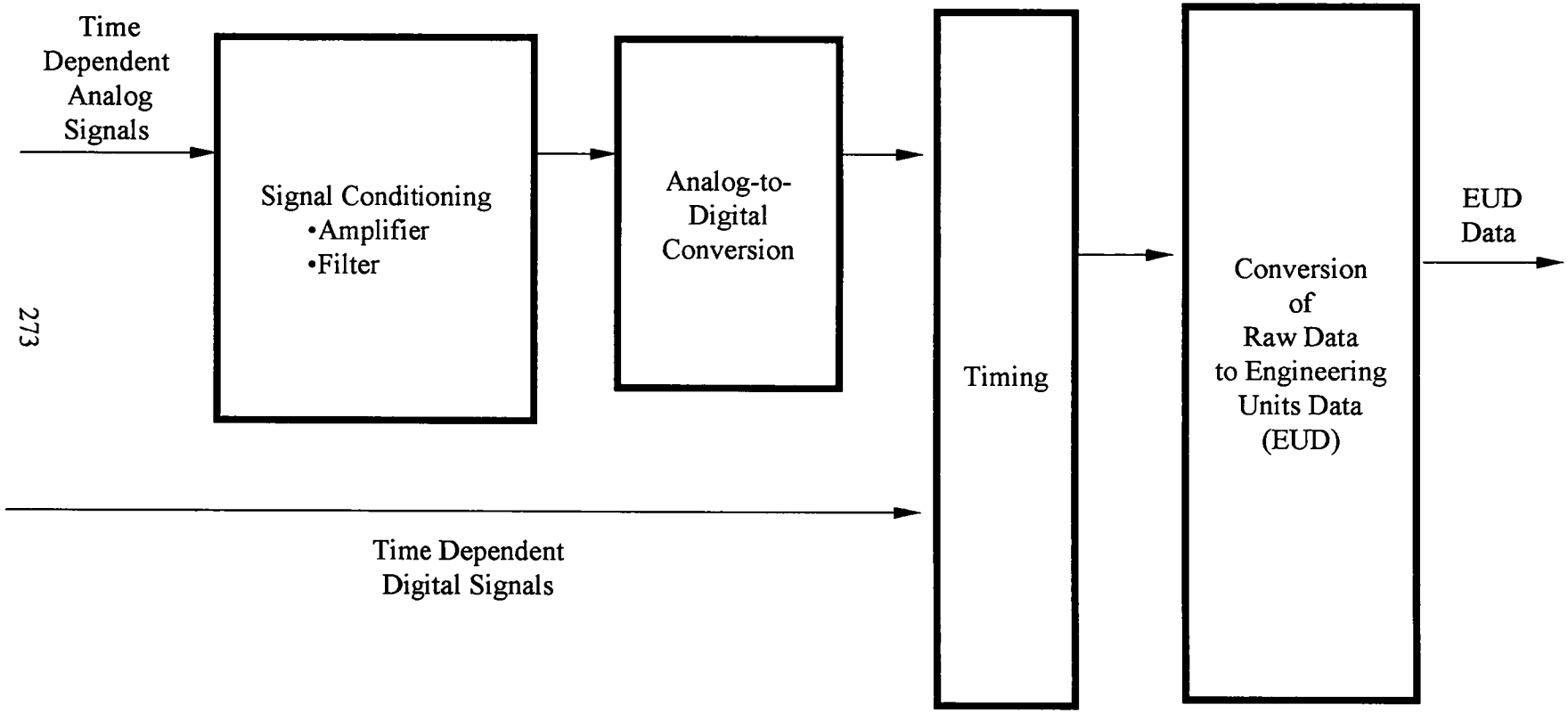


Fig. 4.1. Measurement and Data Reduction System Schematic

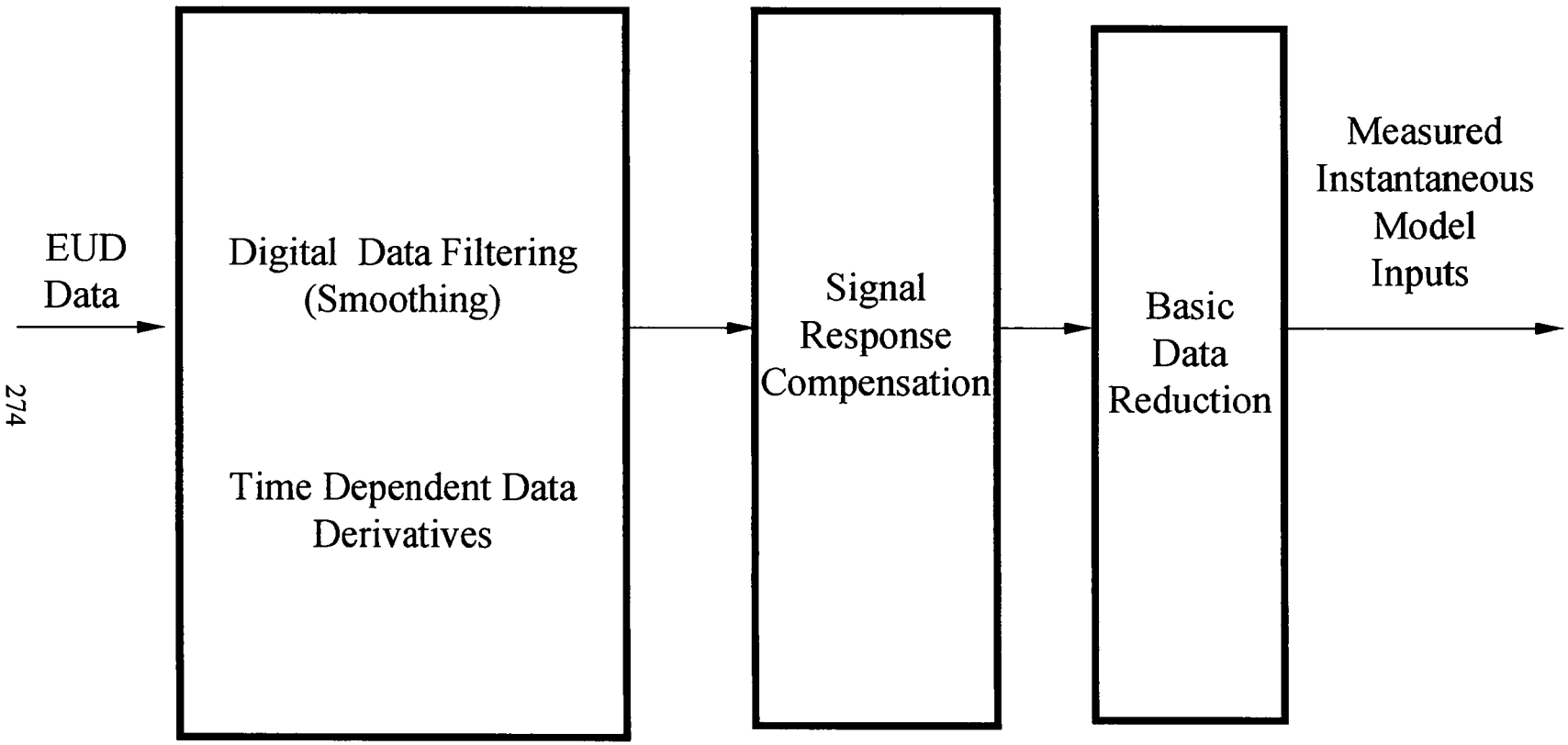


Fig. 4.2. Measurement and Data Reduction System Schematic (Continued)

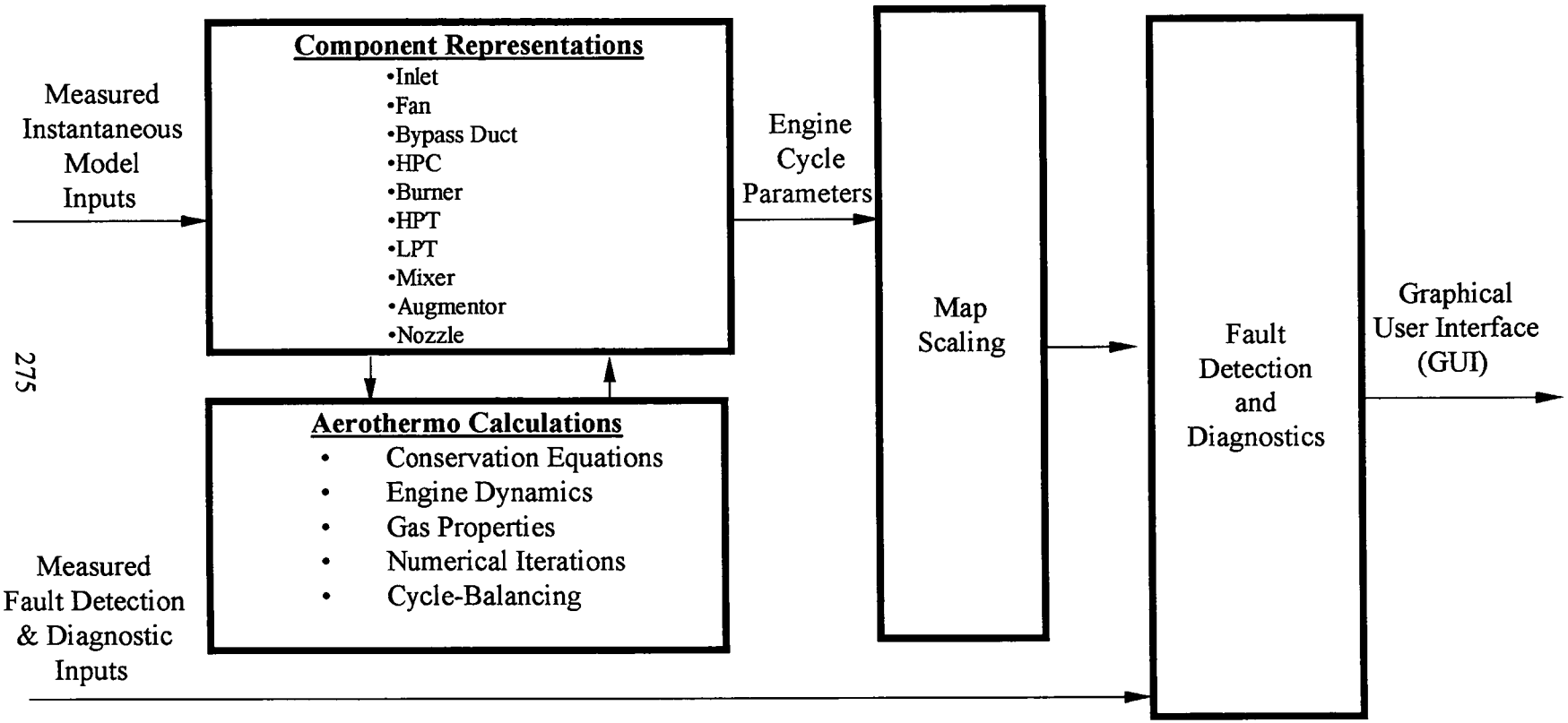


Fig. 4.3. Cycle-Balancing and Fault Identification Process (Serial Operation)

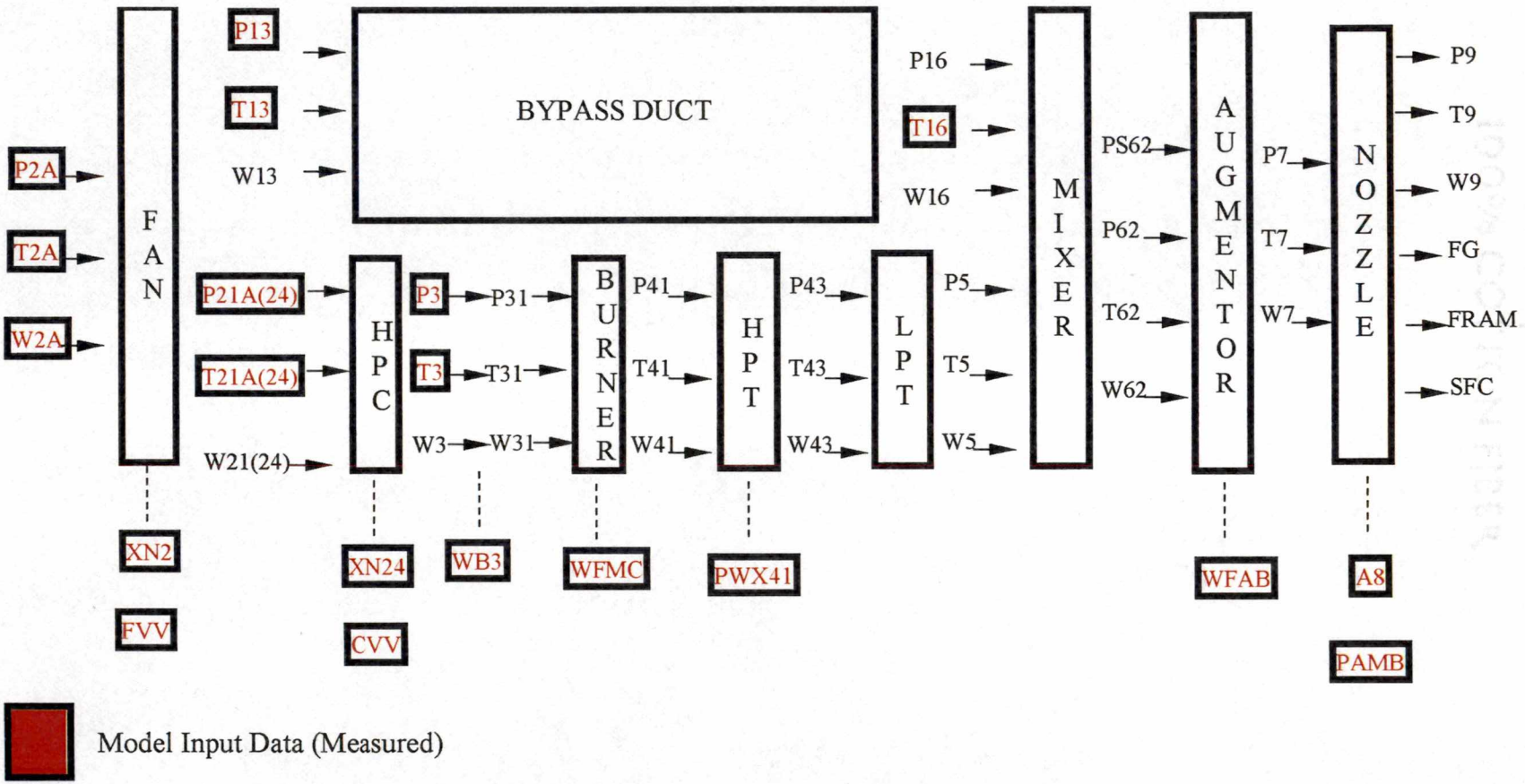


Fig. 4.4. Engine Model Block Diagram

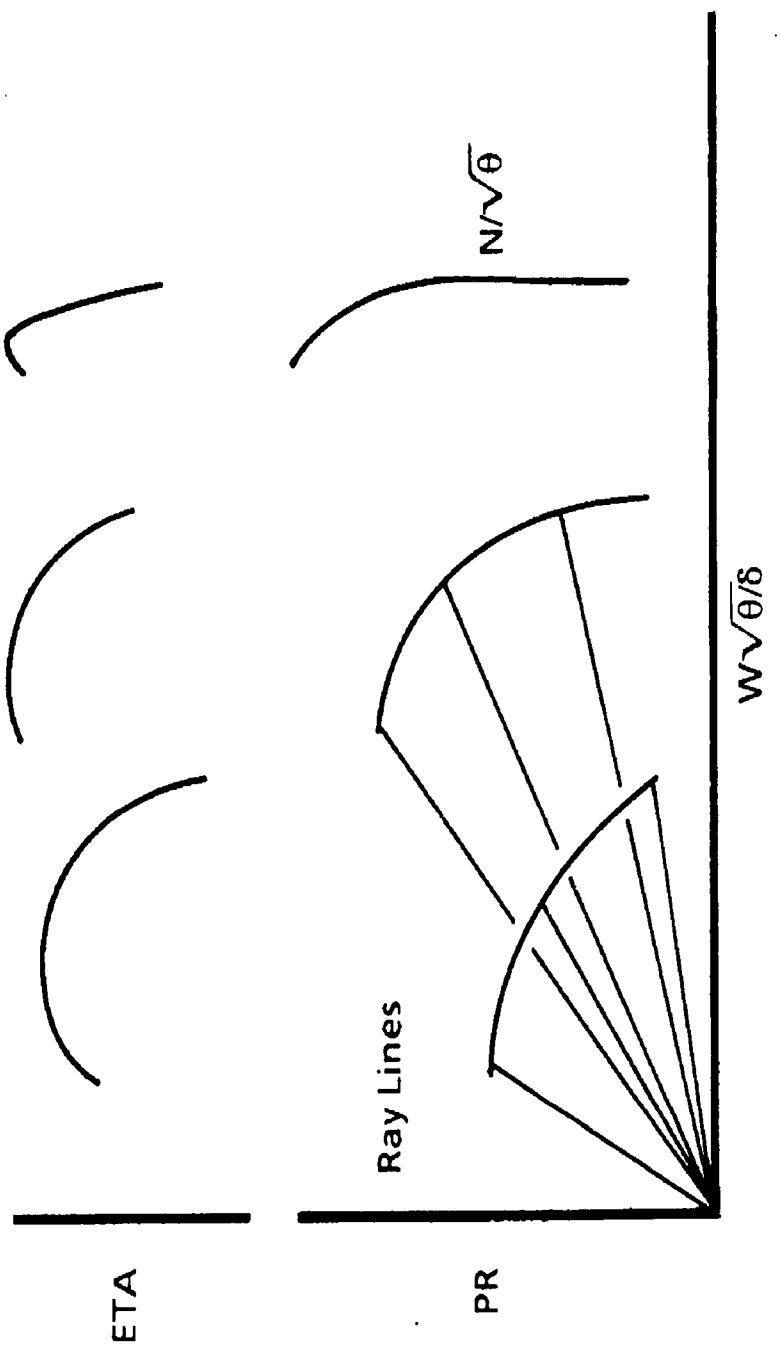


Fig. 4.5.1. Compressor Performance Representation

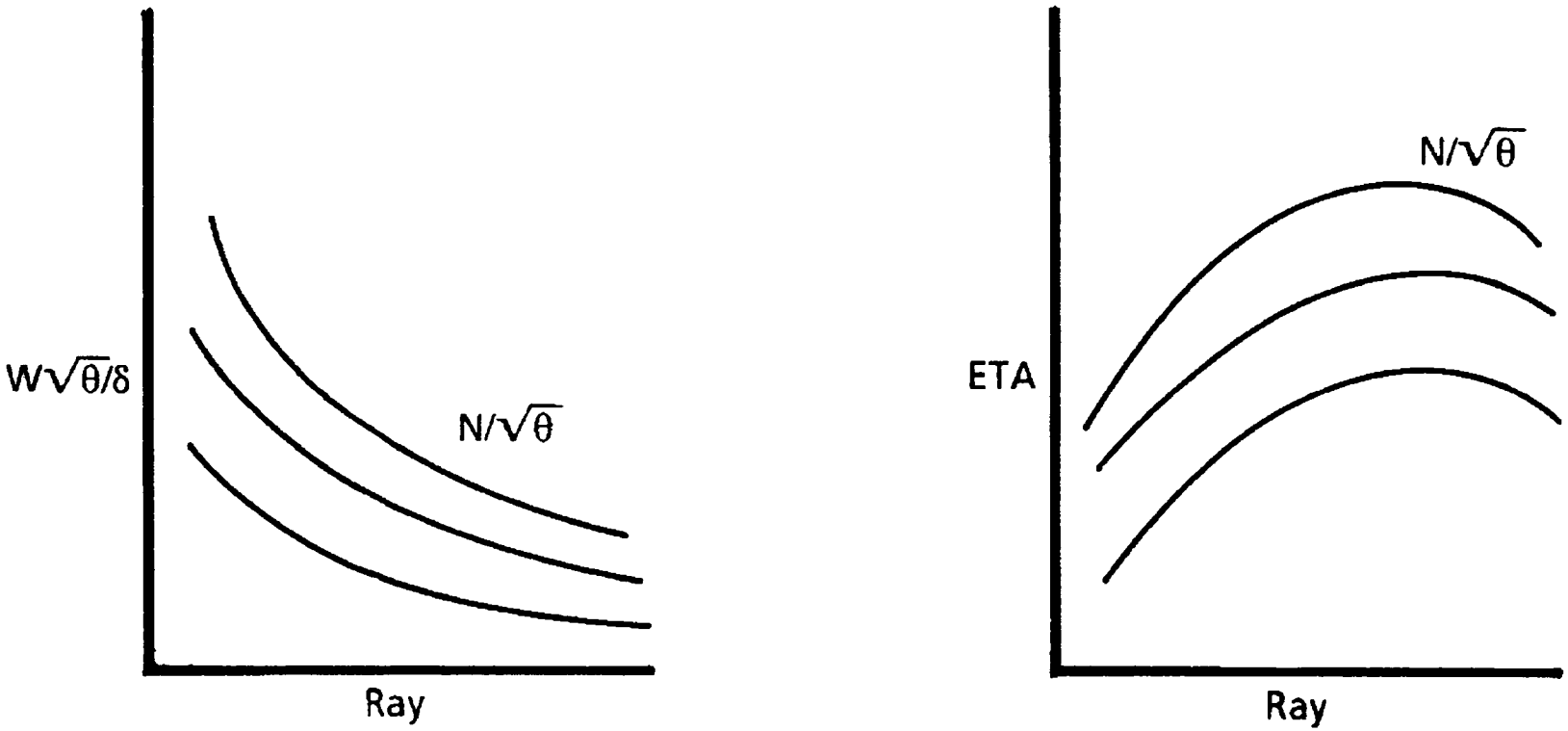


Fig. 4.5.2. Transformed Compressor Map Representation

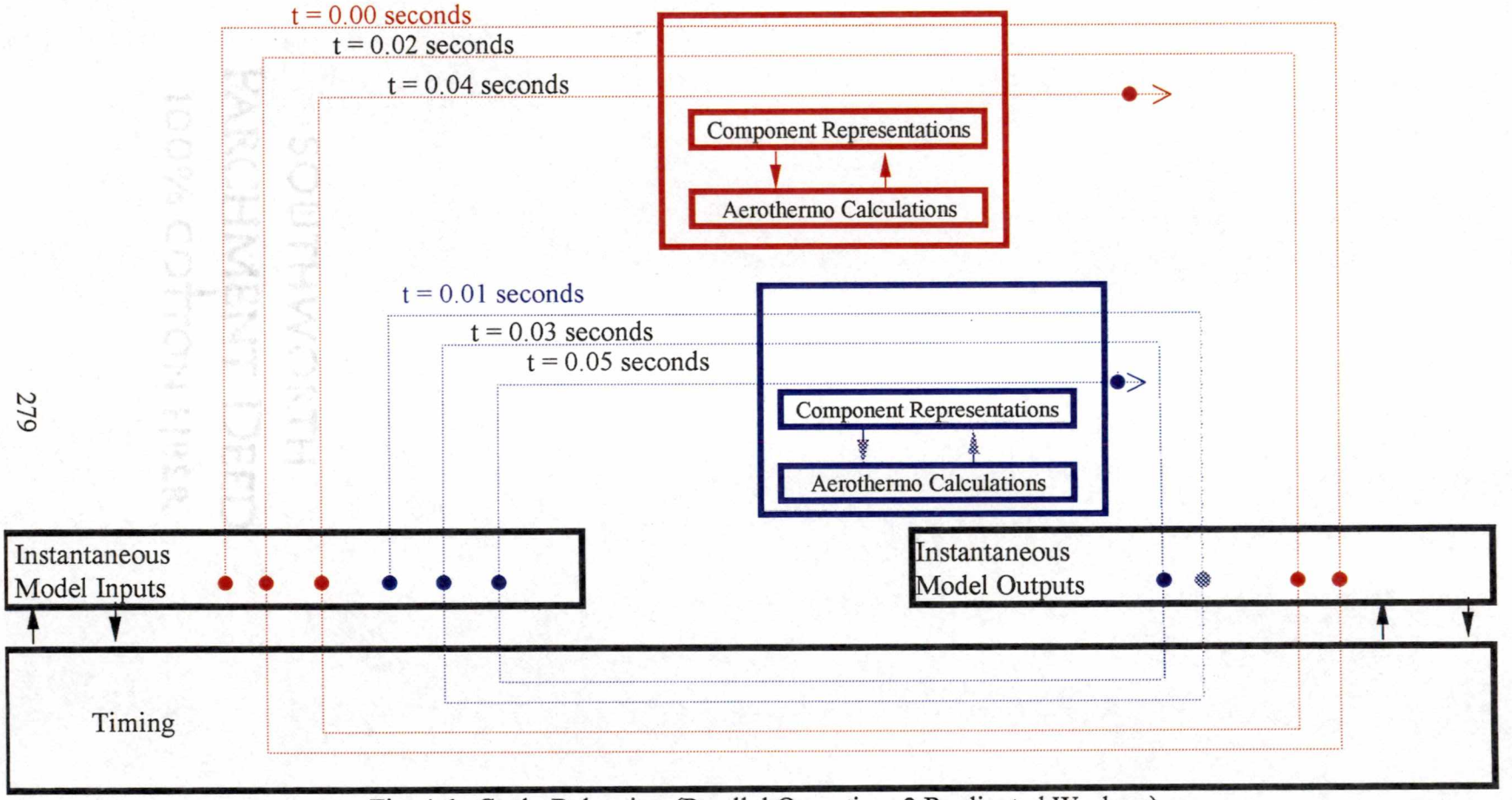


Fig. 4.6. Cycle-Balancing (Parallel Operation: 2 Replicated Workers)

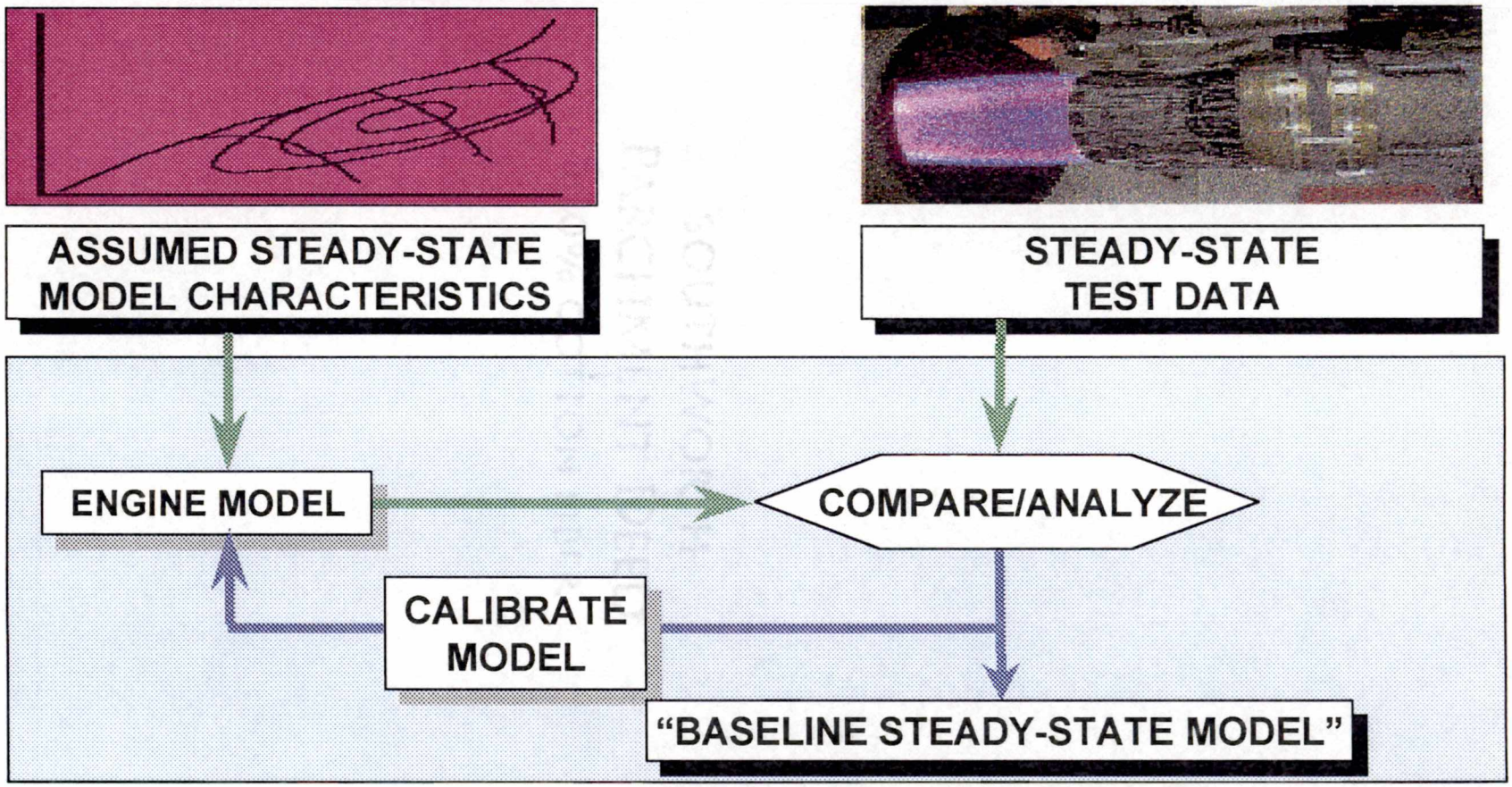


Fig. 4.7. Step 1: Initial Steady-State Performance Calibration

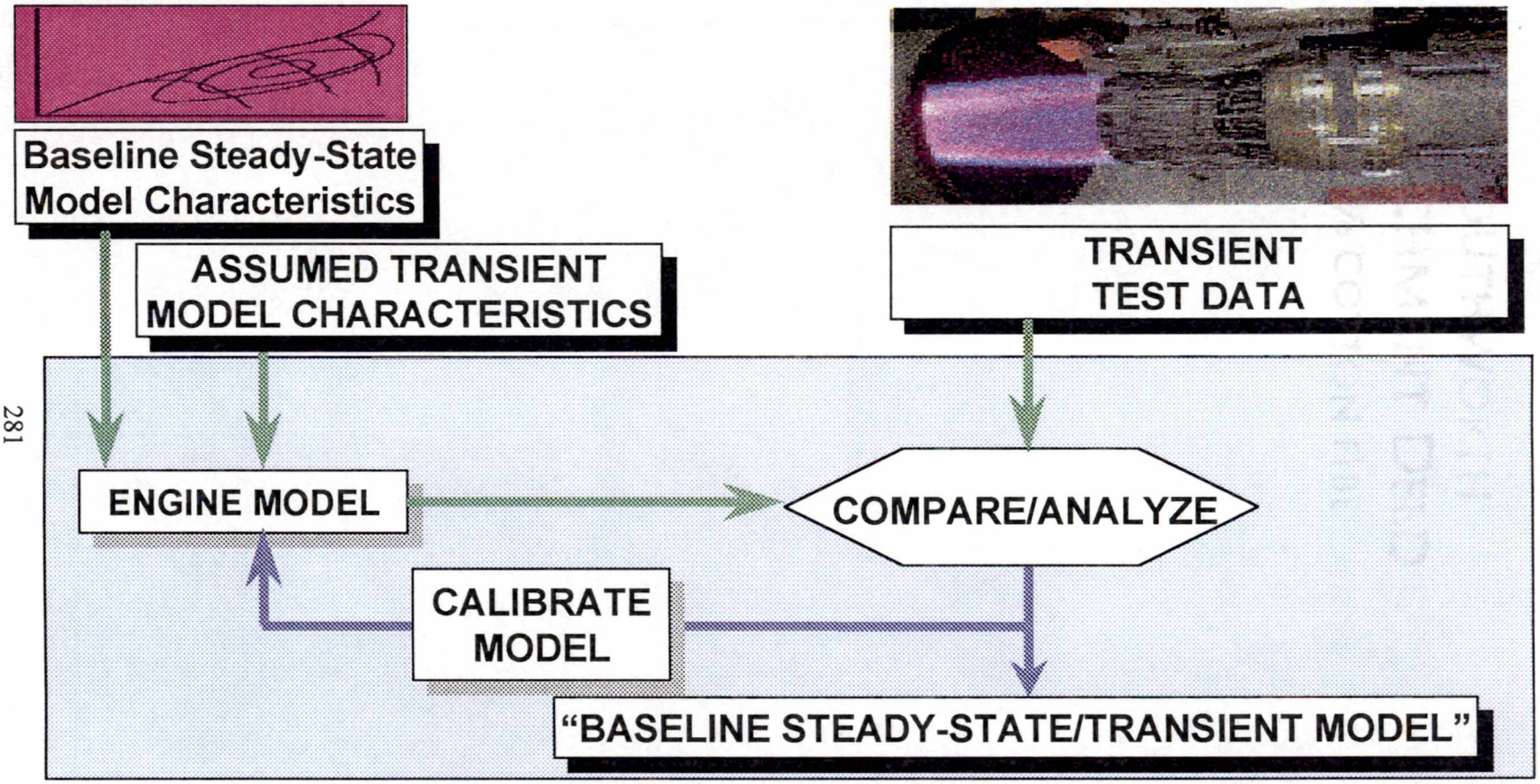


Fig. 4.8. Step 2: Initial Transient Performance Calibration

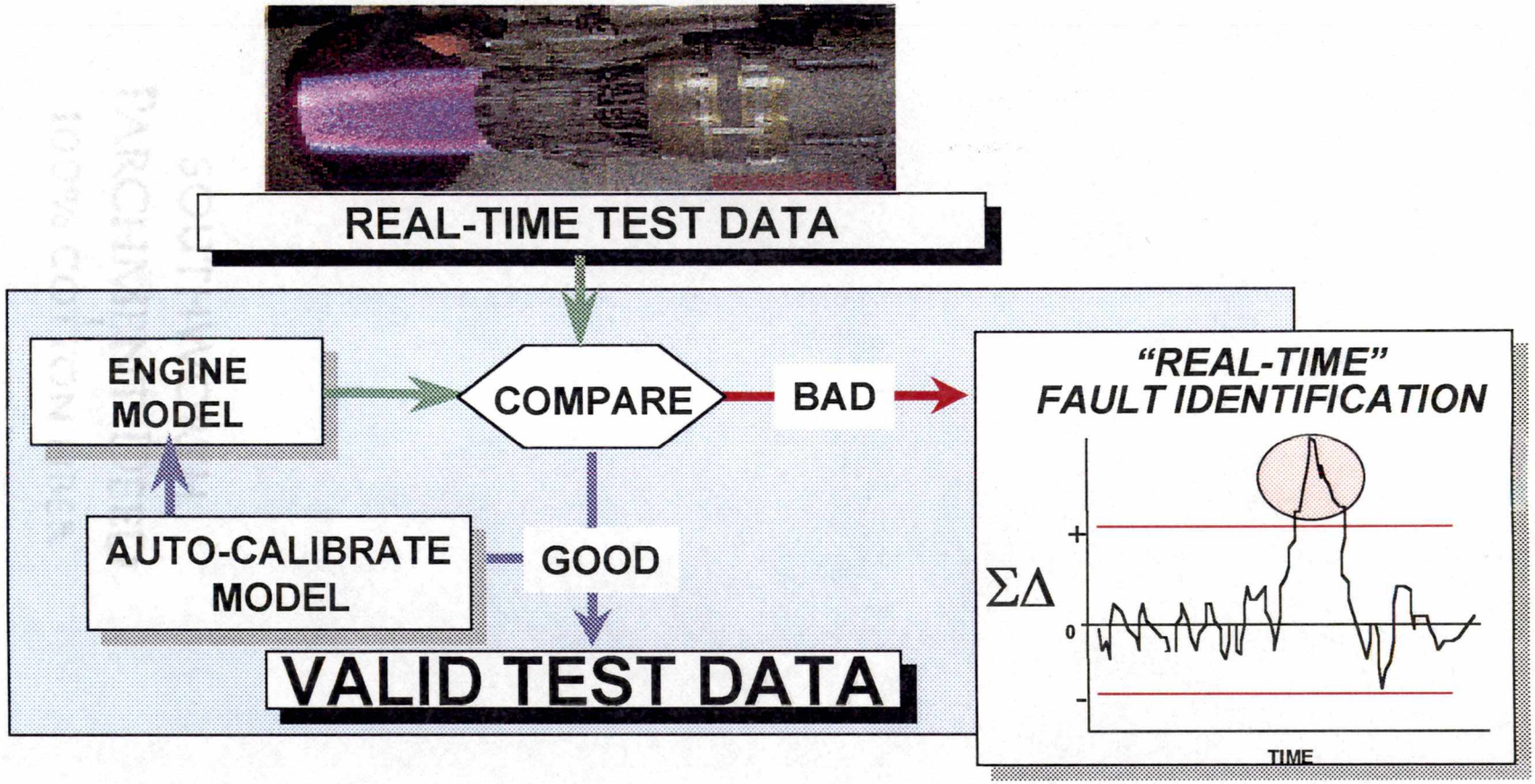


Fig. 4.9. Step 3: Real-Time Model Calibration and Fault Identification

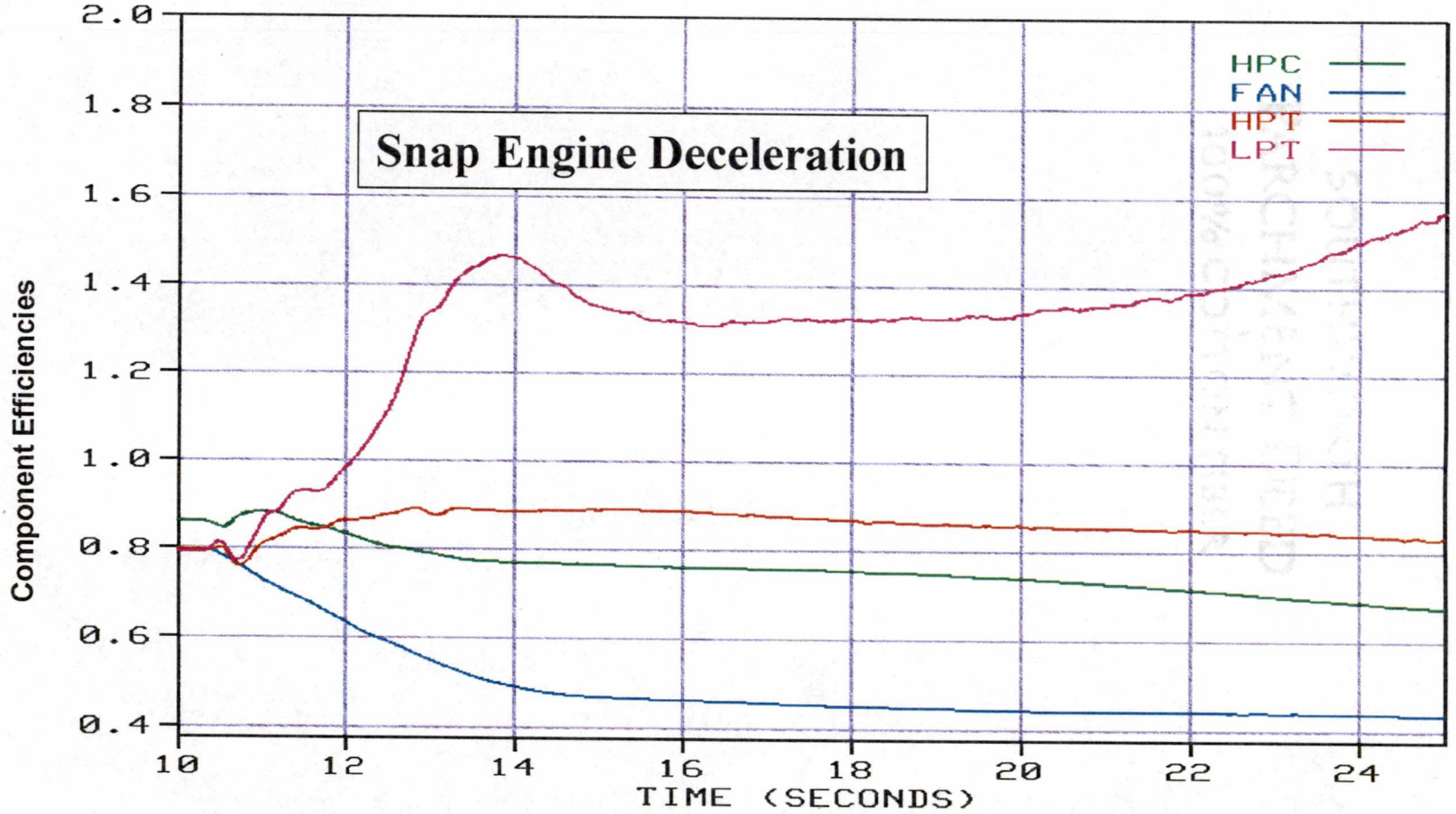


Fig. 4.10. Calculated Component Efficiencies During Snap Engine Deceleration (Before Transient Model Calibration)

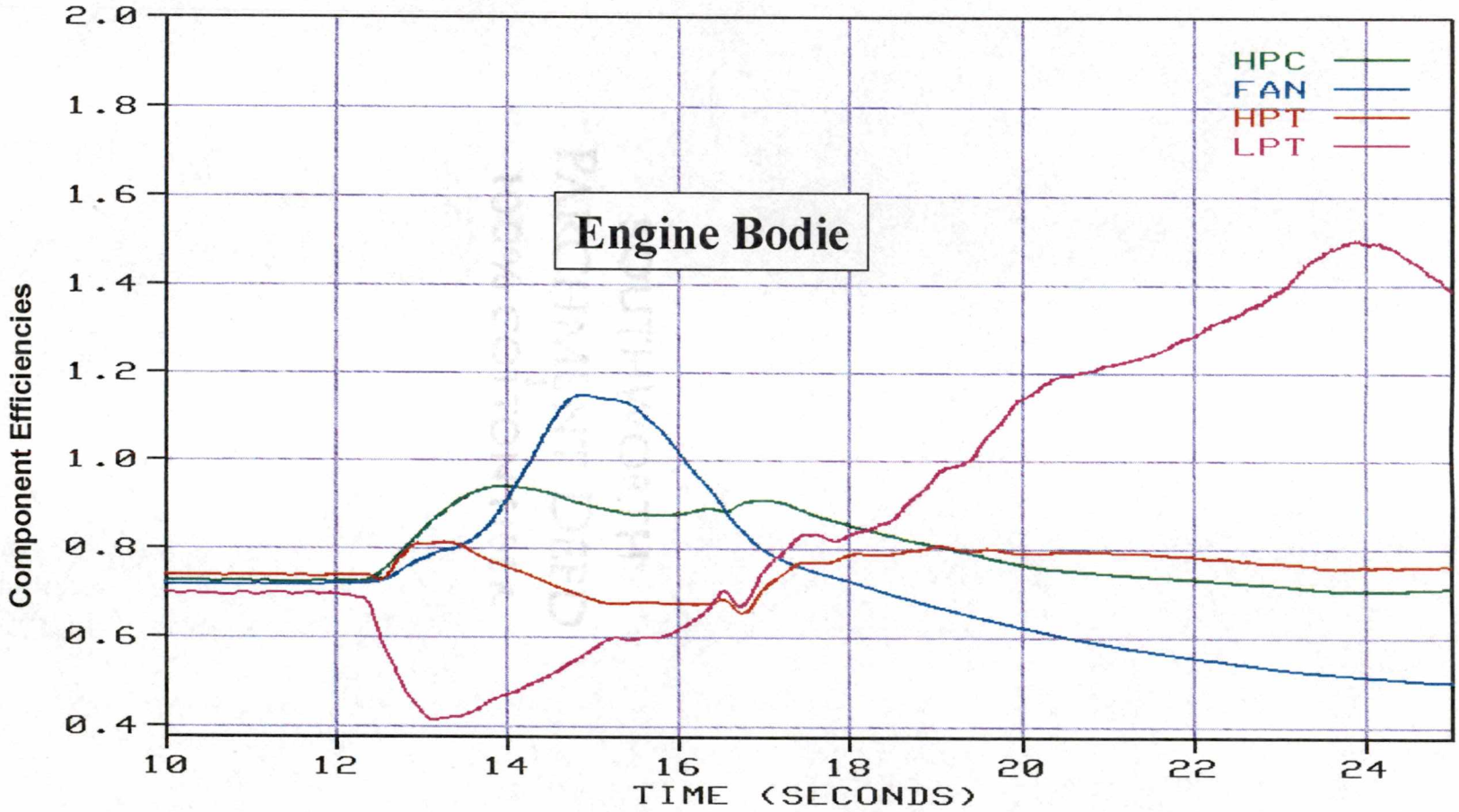


Fig. 4.11. Calculated Component Efficiencies During Engine Bodie (Before Transient Model Calibration)

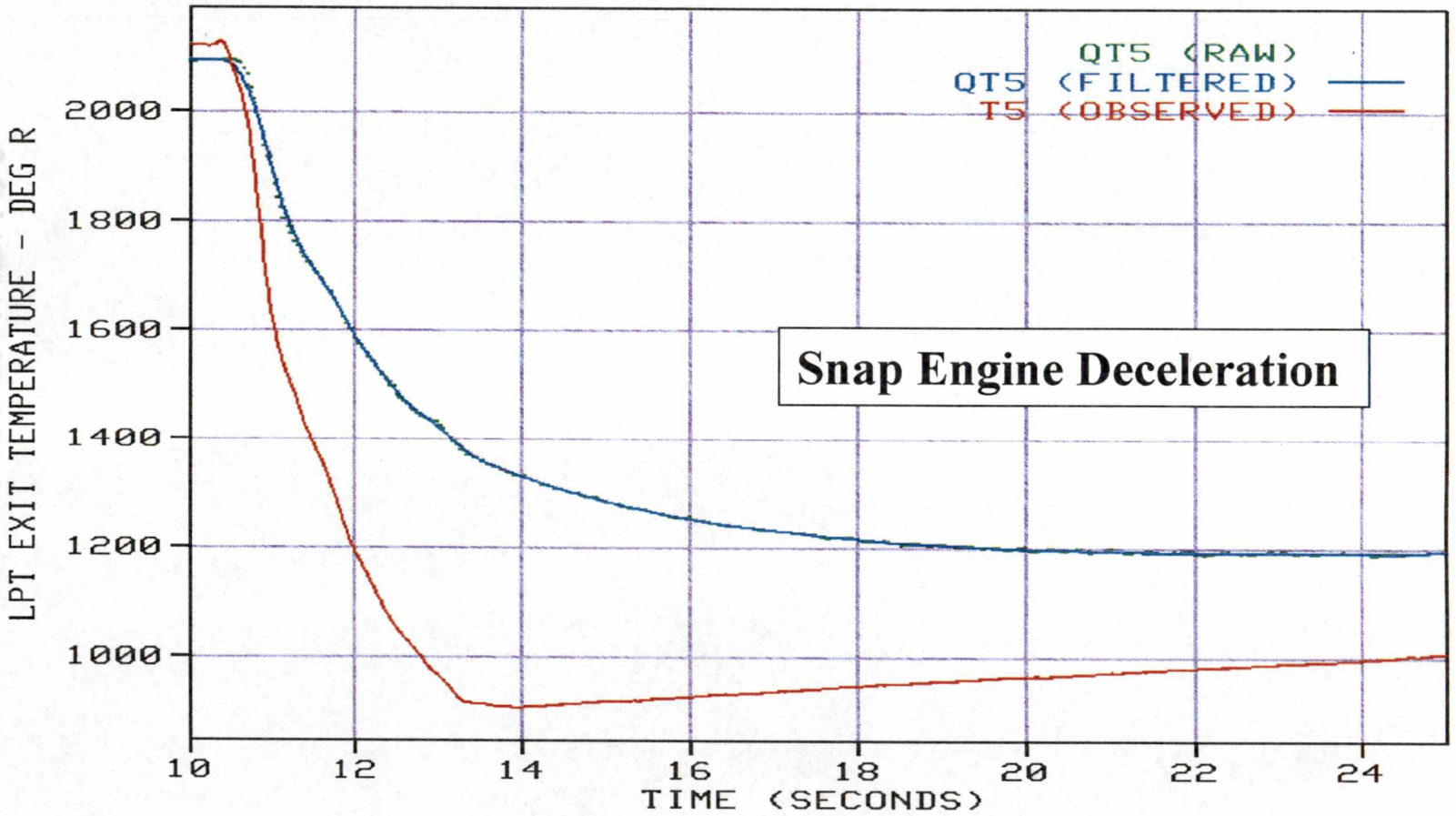


Fig. 4.12. LP Turbine Exit Temperature During Snap Engine Deceleration (Before Transient Model Calibration)

PARCHMENT PAPER
100% COTTON FIBER

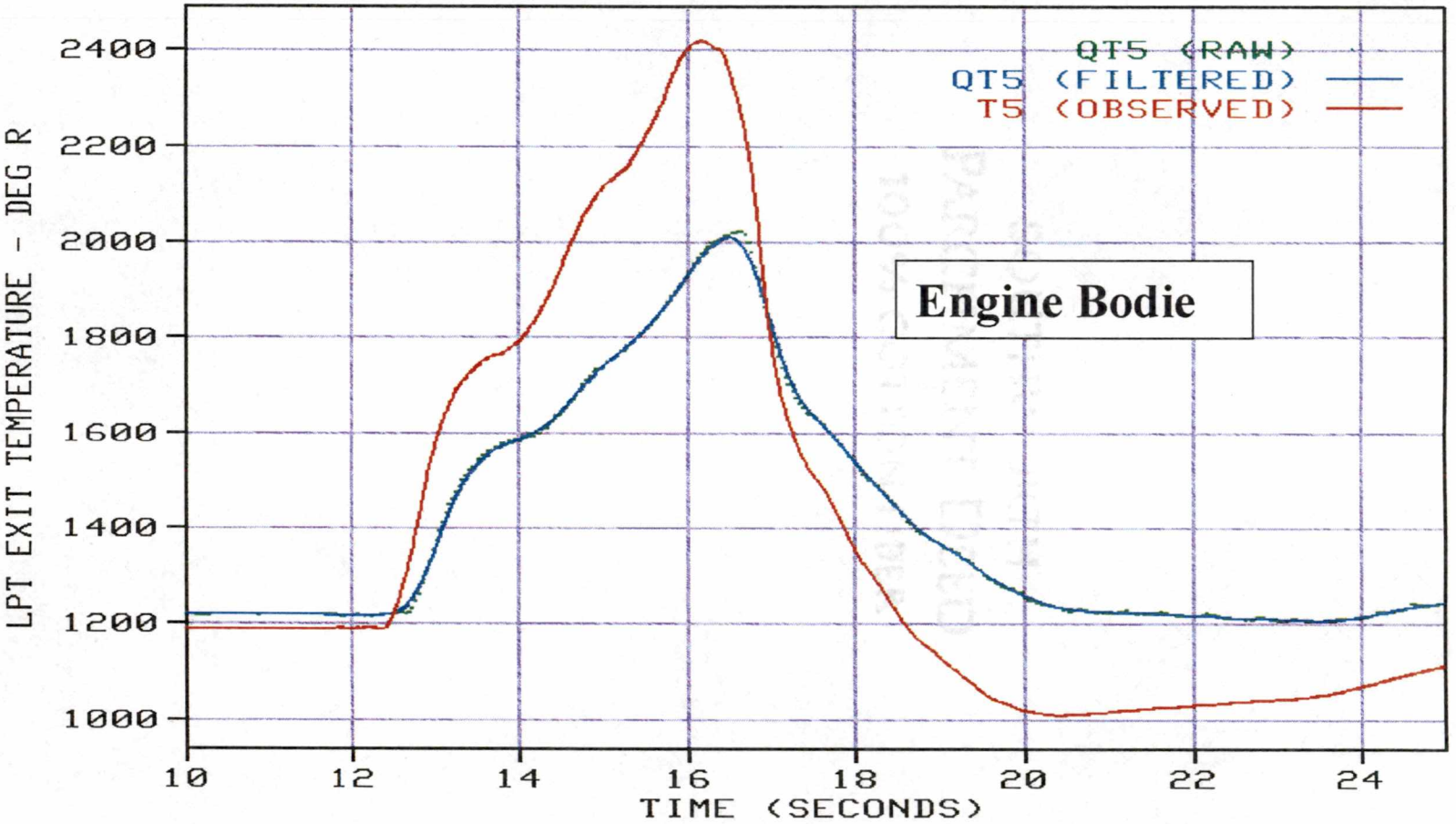


Fig. 4.13. LP Turbine Exit Temperature During Engine Bodie (Before Transient Model Calibration)

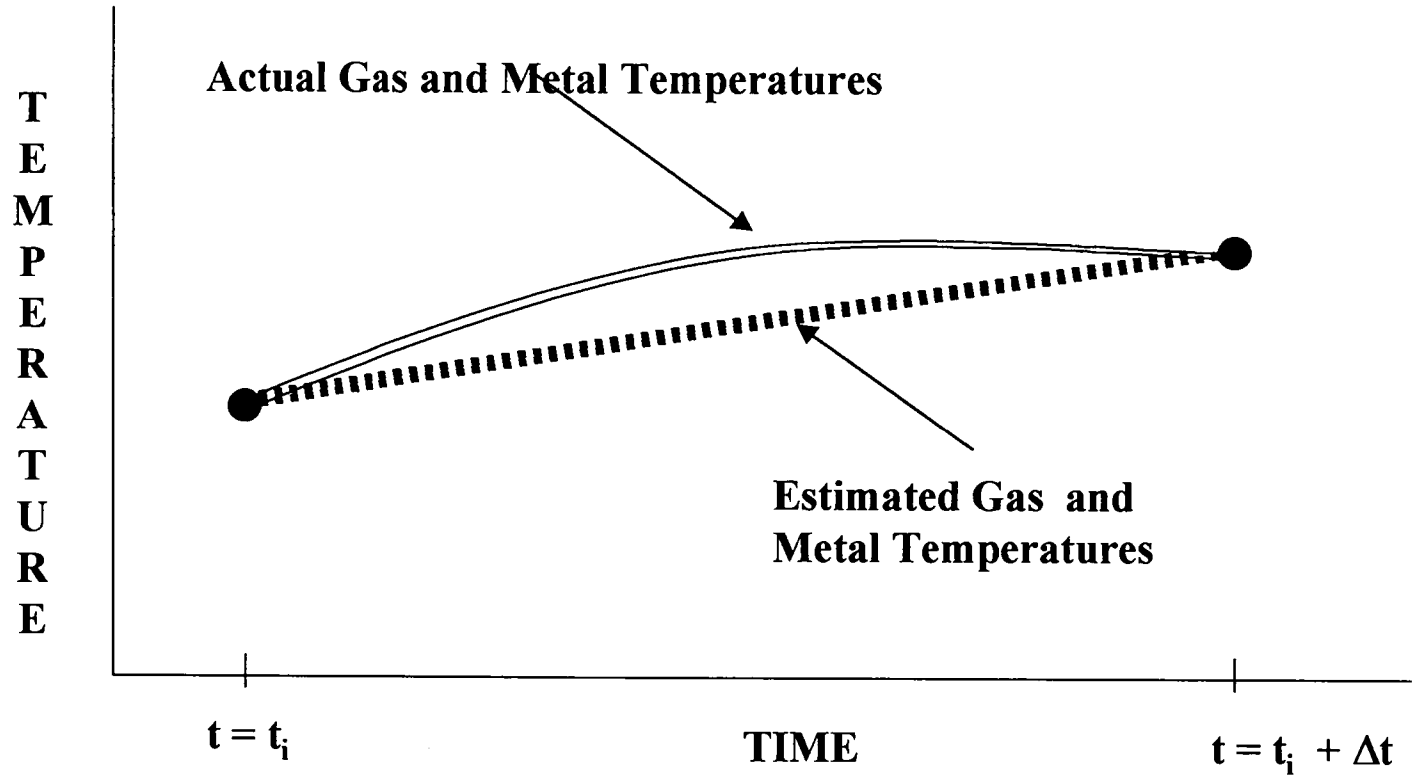


Fig. 4.14. Assumed Transient Gas and Metal Temperature Variations for Error Analysis ($\tau << 1$)

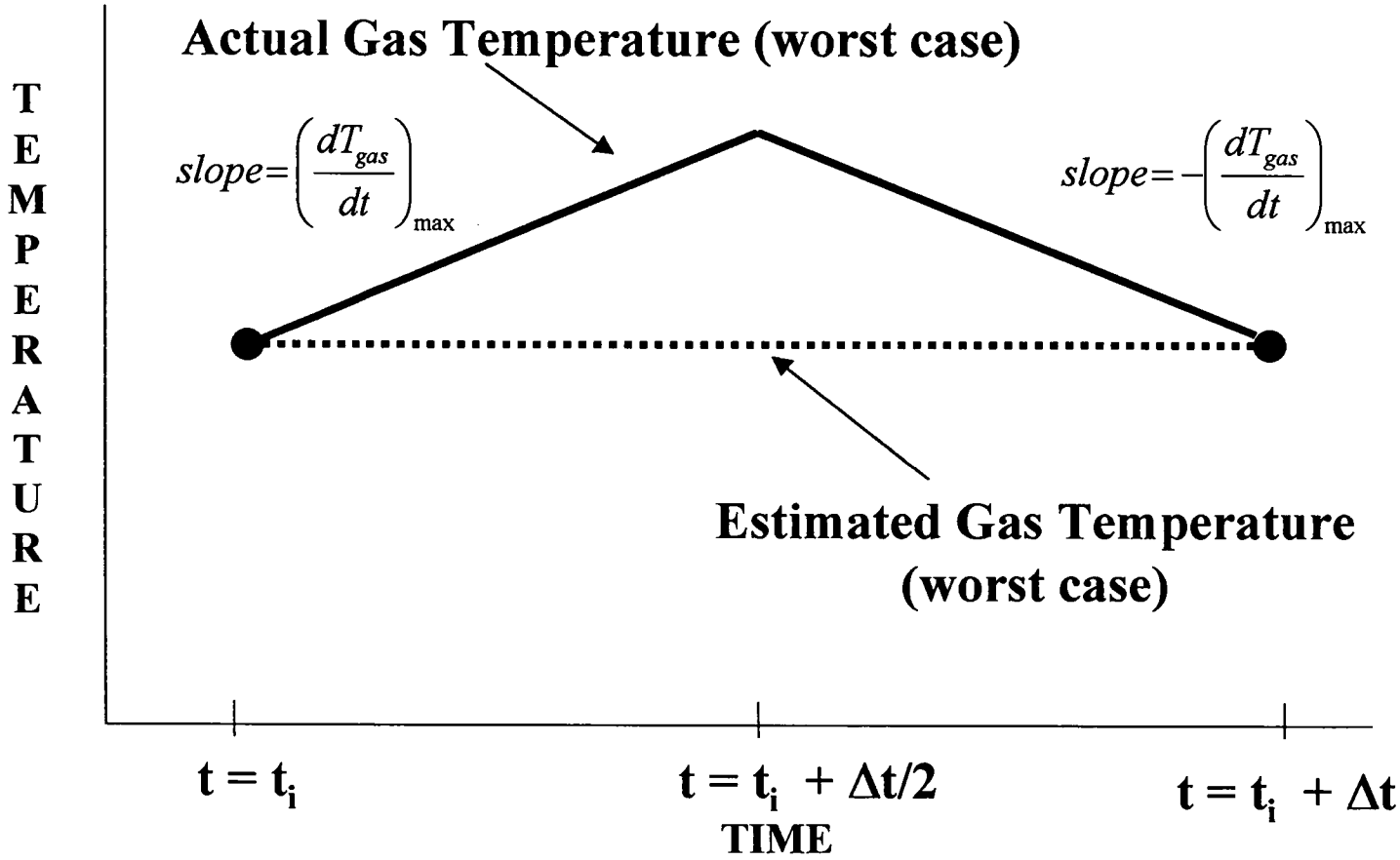


Fig. 4.15. Metal Temperature Error Analysis Model for Transient Operation

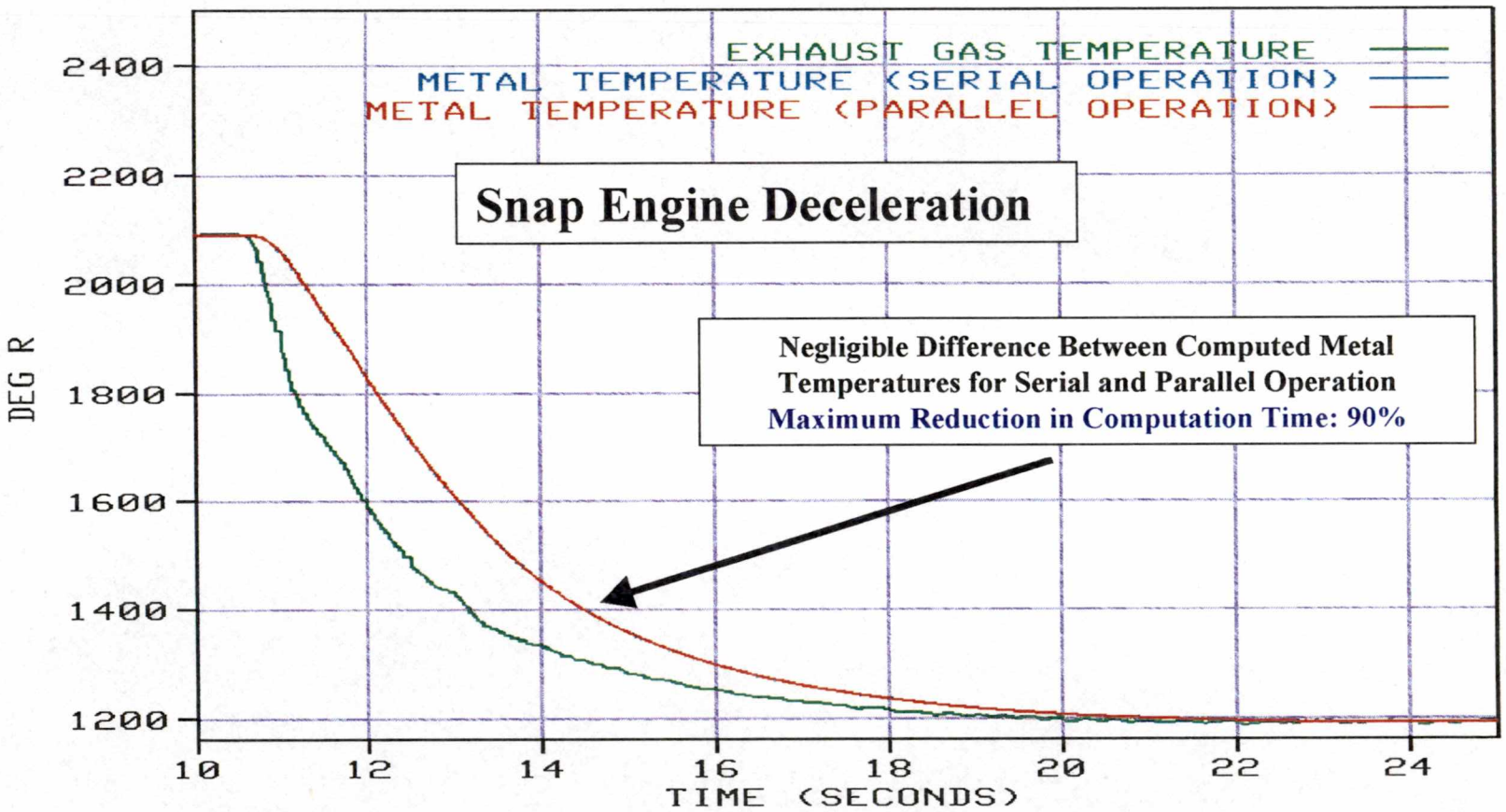


Fig. 4.16. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Raw Data, 10 Processors, $\tau \equiv 1$ second)

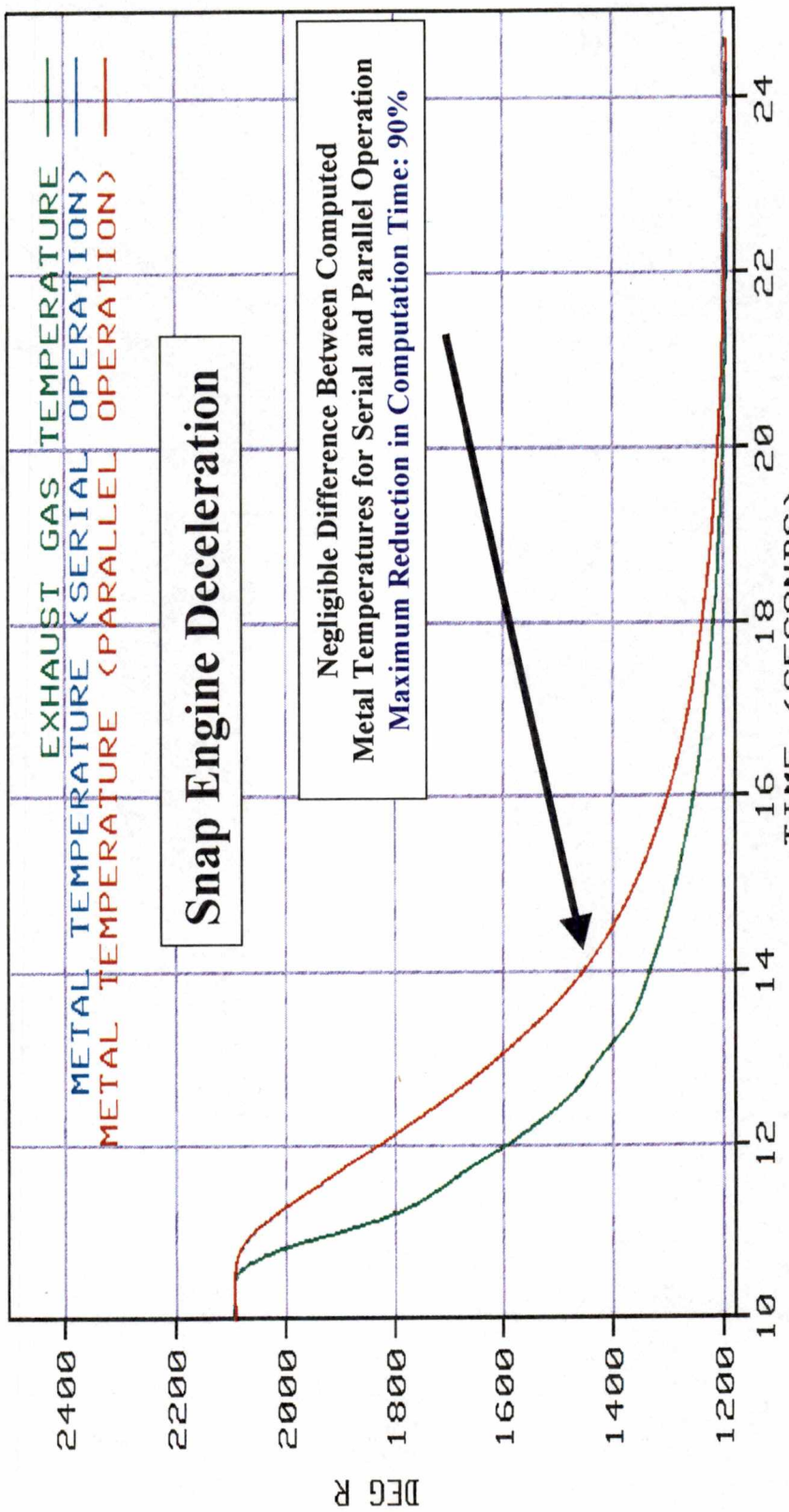


Fig. 4.17. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Filtered Data, 10 Processors, $\tau \equiv 1$ second)

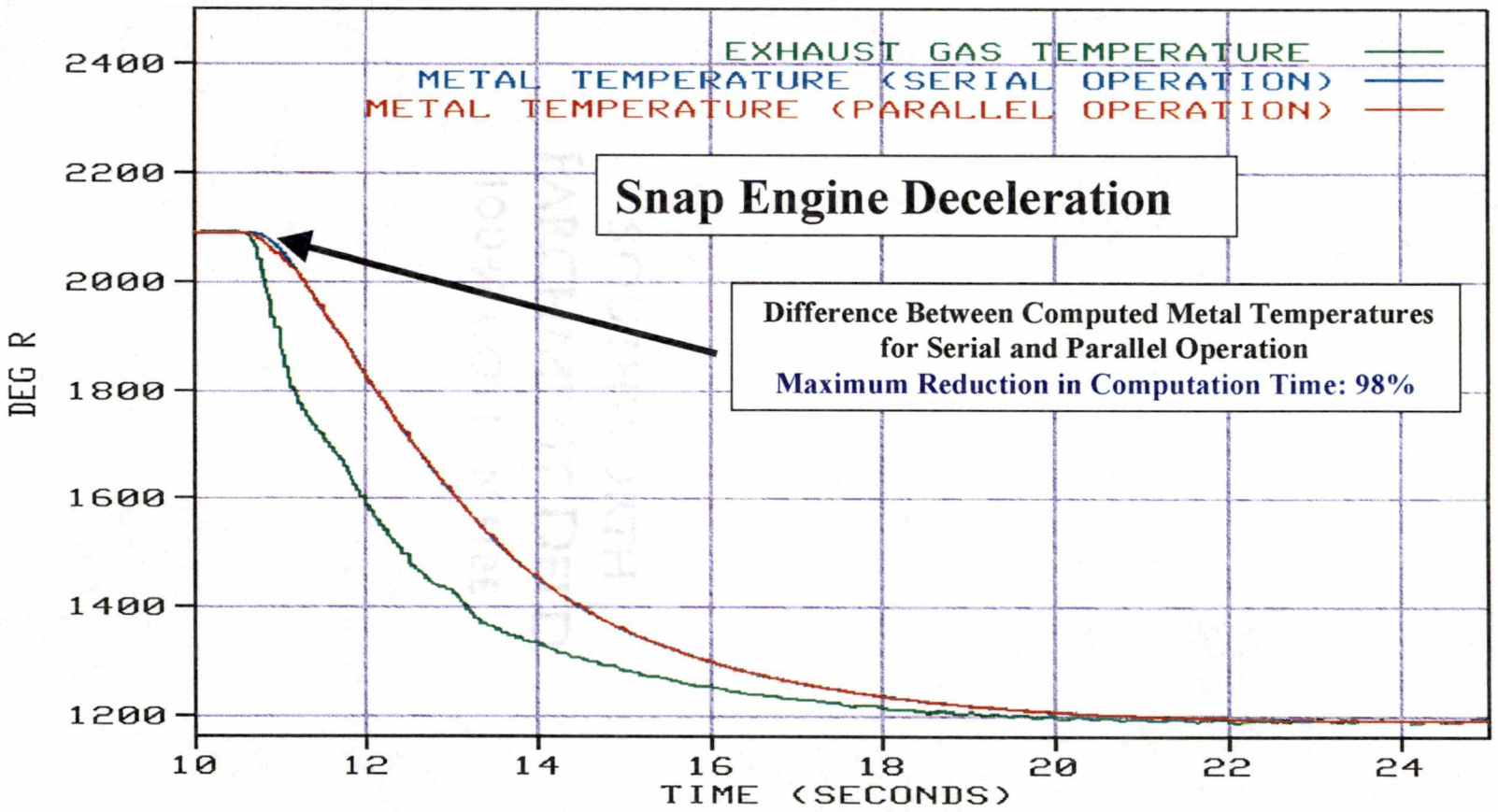


Fig. 4.18. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Raw Data, 50 Processors, $\tau \equiv 1$ second)

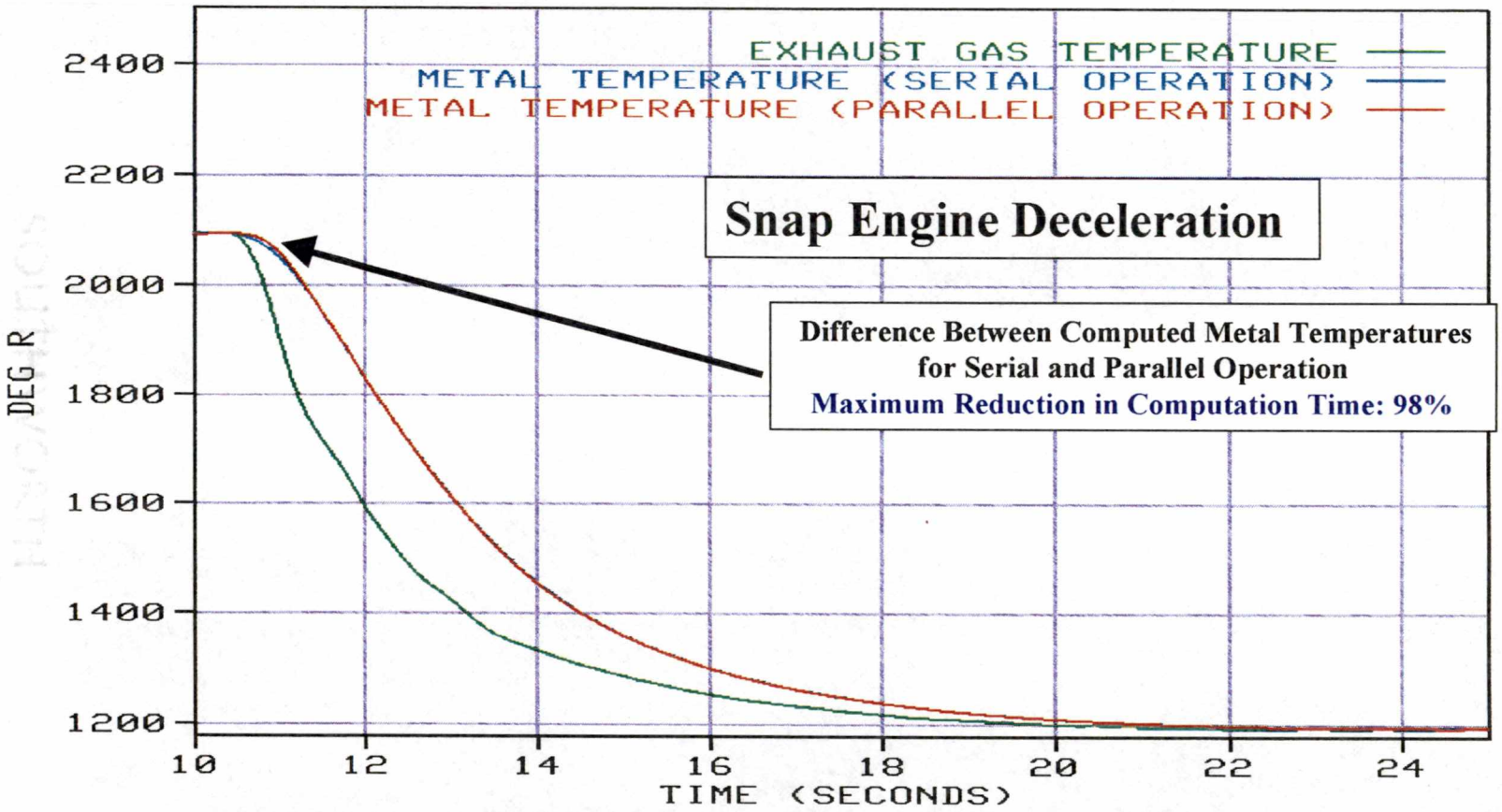


Fig. 4.19. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Filtered Data, 50 Processors, $\tau \equiv 1$ second)

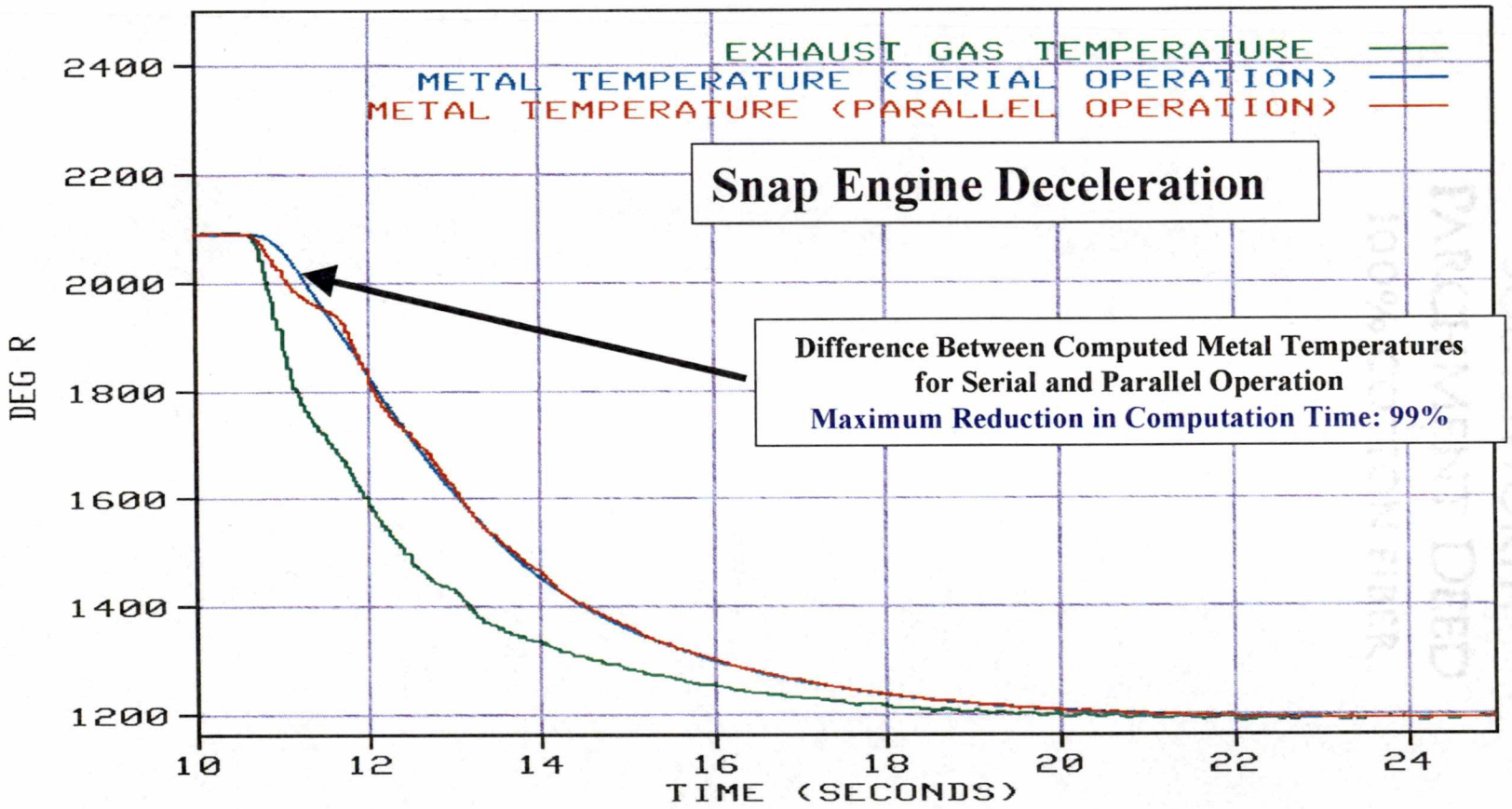


Fig. 4.20. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Raw Data, 100 Processors, $\tau \equiv 1$ second)

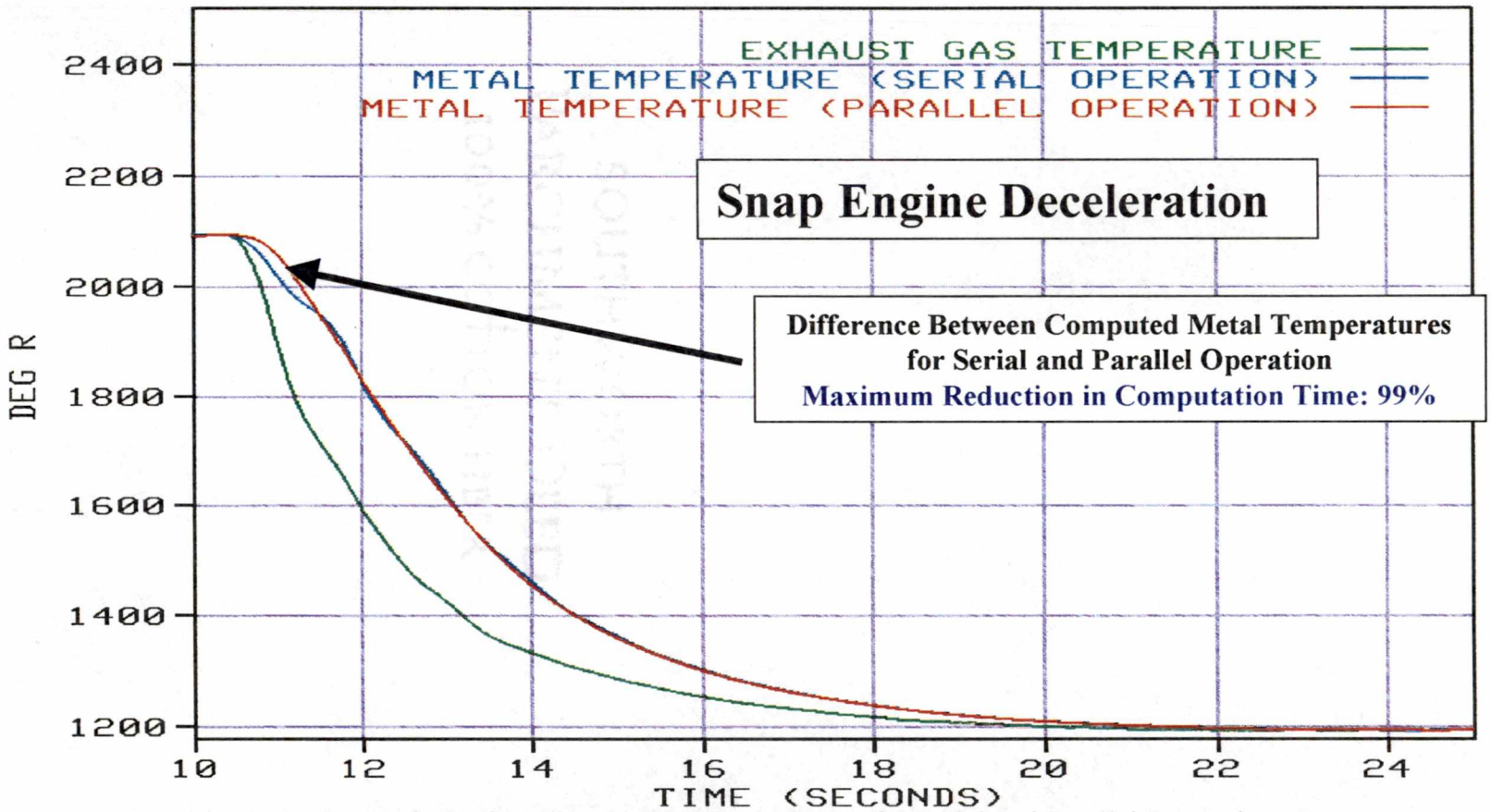


Fig. 4.21. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Filtered Data, 100 Processors, $\tau \equiv 1$ second)

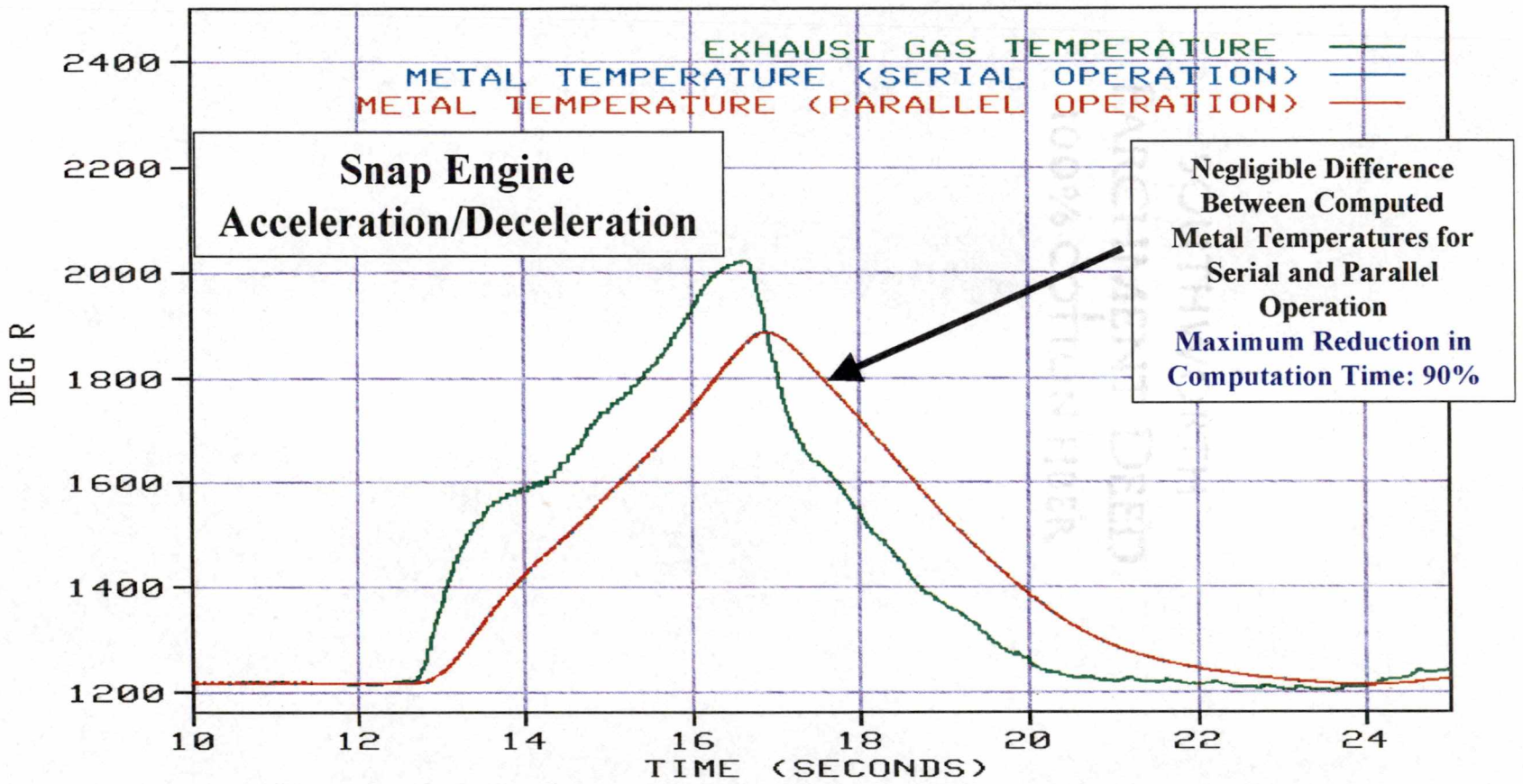


Fig. 4.22. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Raw Data, 10 Processors, $\tau \equiv 1$ second)

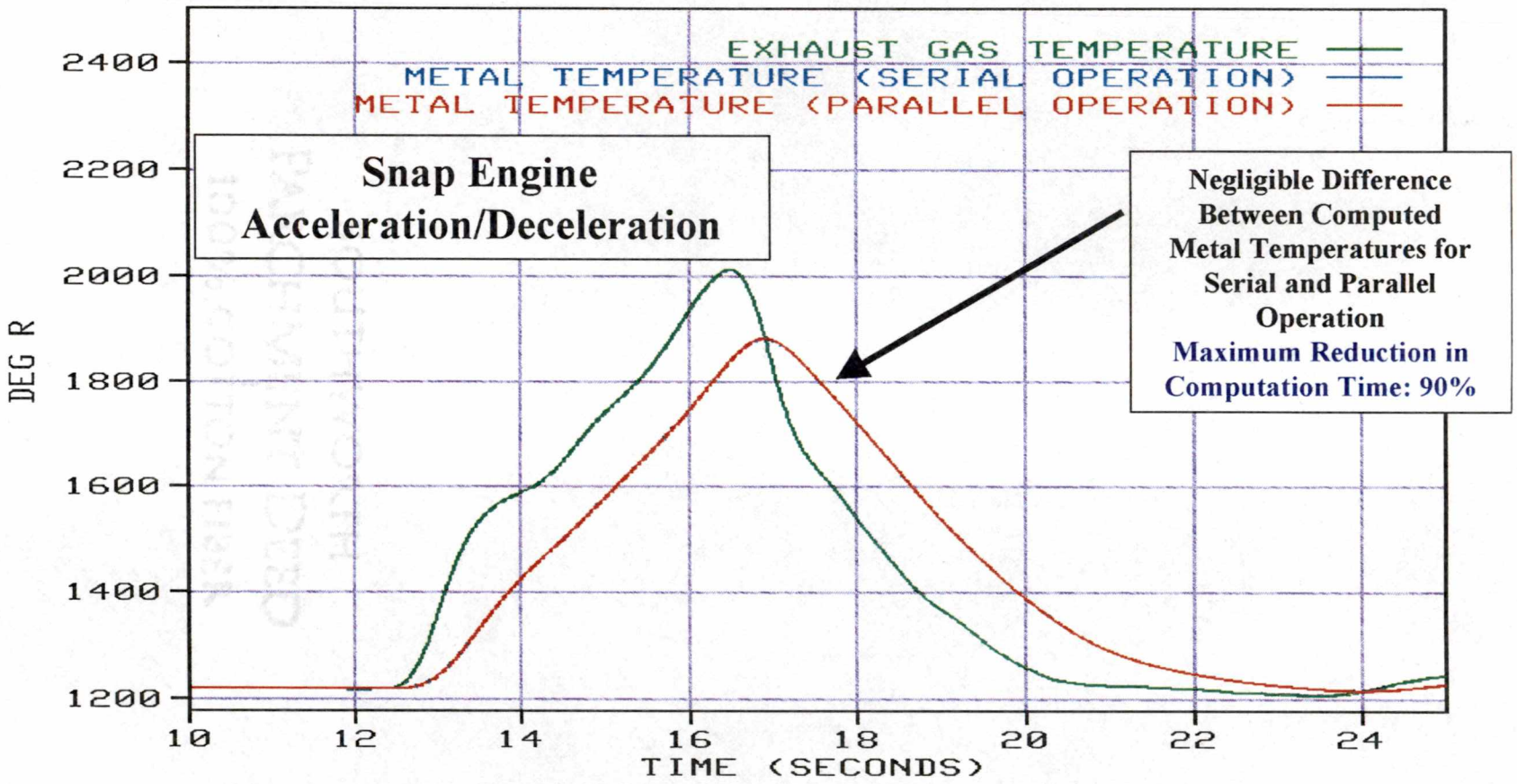


Fig. 4.23. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing (Filtered Data, 10 Processors, $\tau \equiv 1$ second)

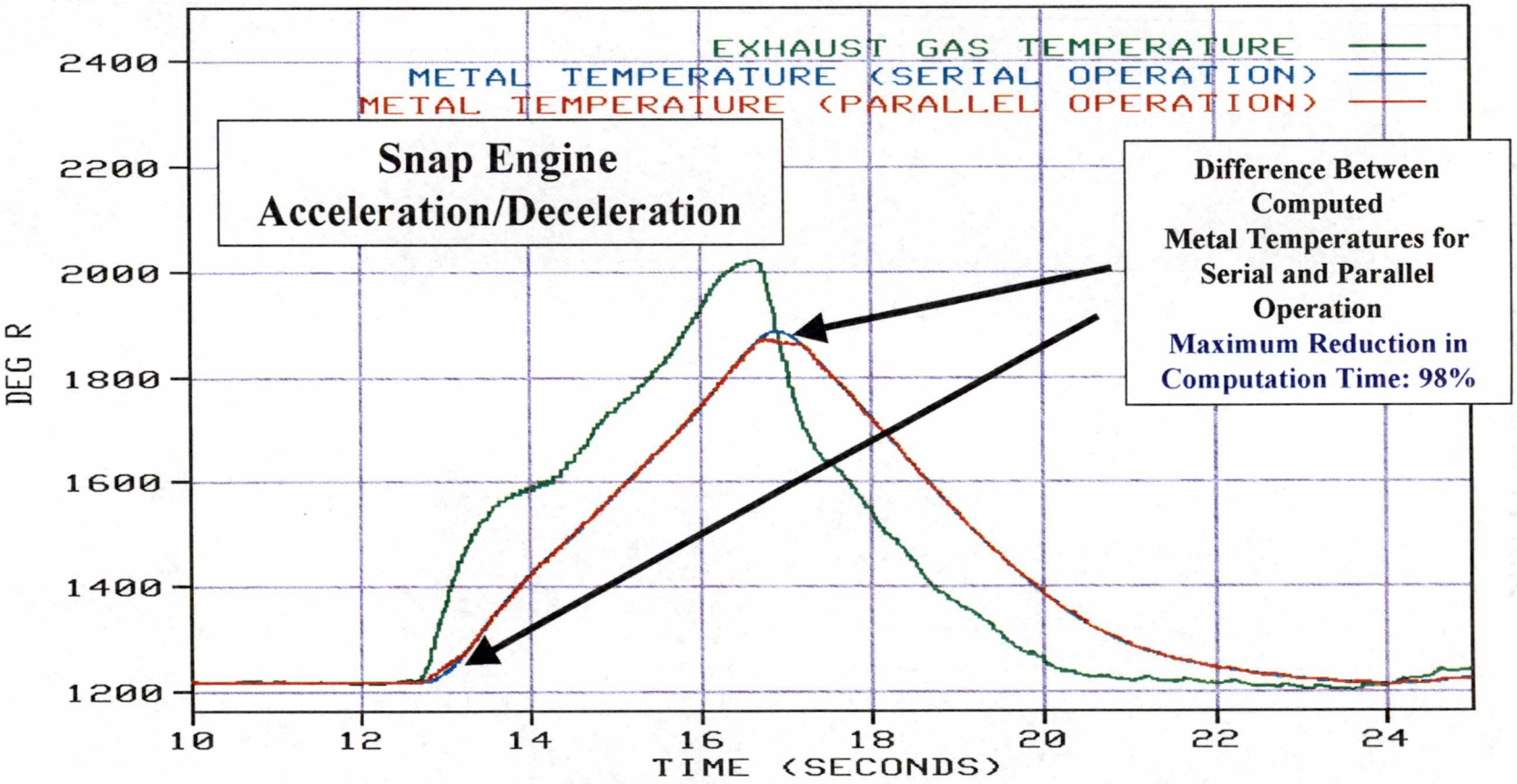


Fig. 4.24. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Raw Data, 50 Processors, $\tau \equiv 1$ second)

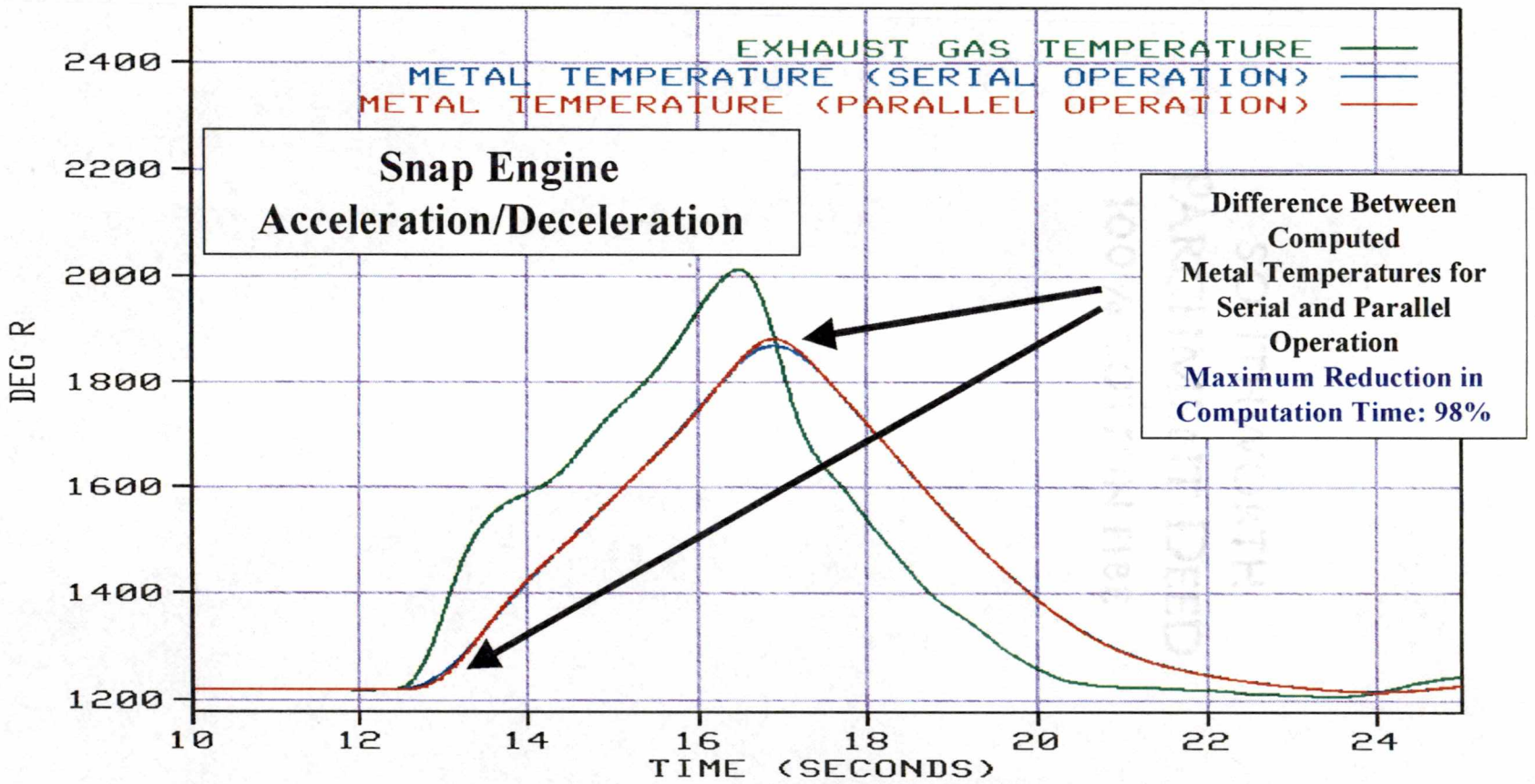


Fig. 4.25. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Filtered Data, 50 Processors, $\tau \equiv 1$ second)

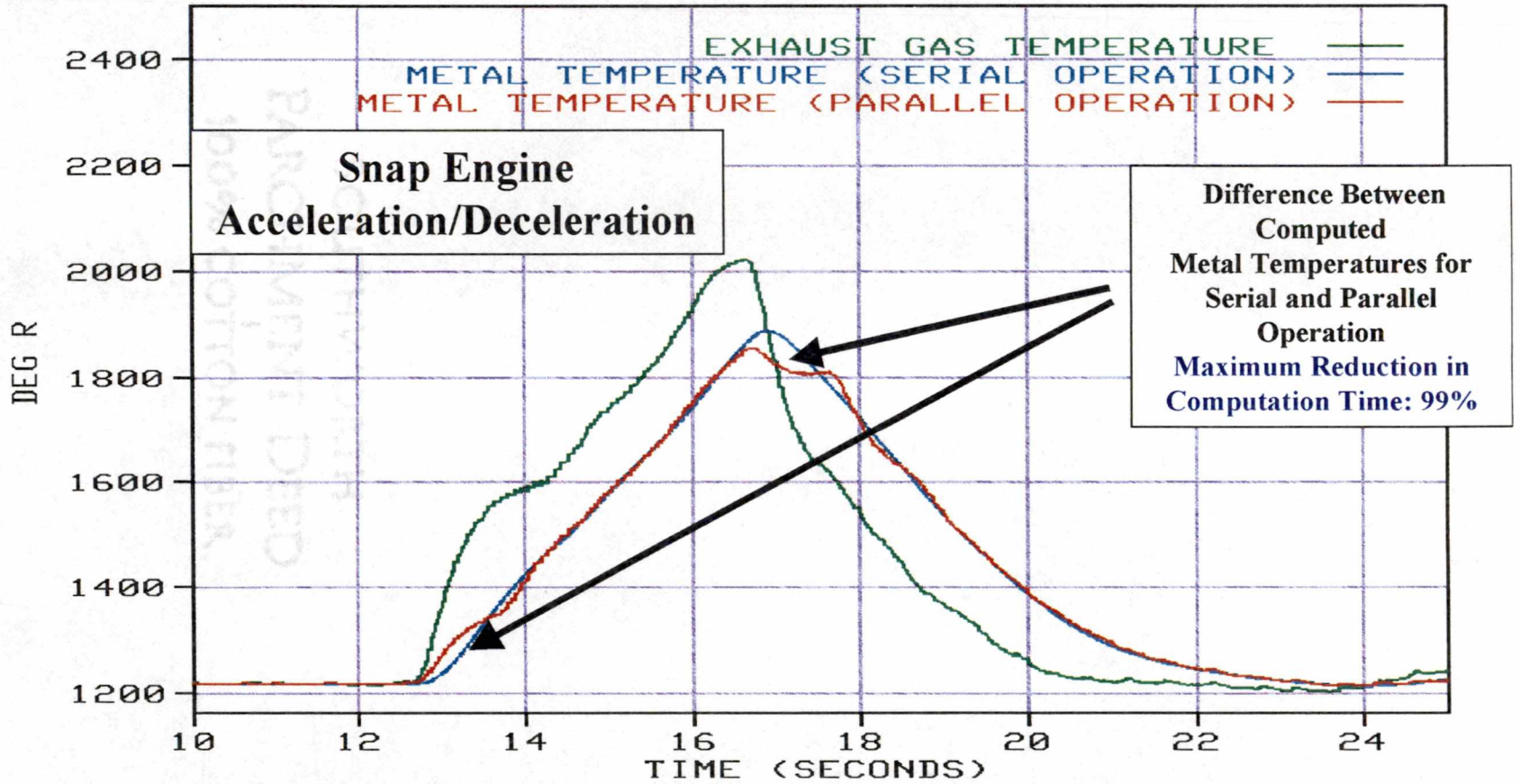


Fig. 4.26. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Raw Data, 100 Processors, $\tau \equiv 1$ second)

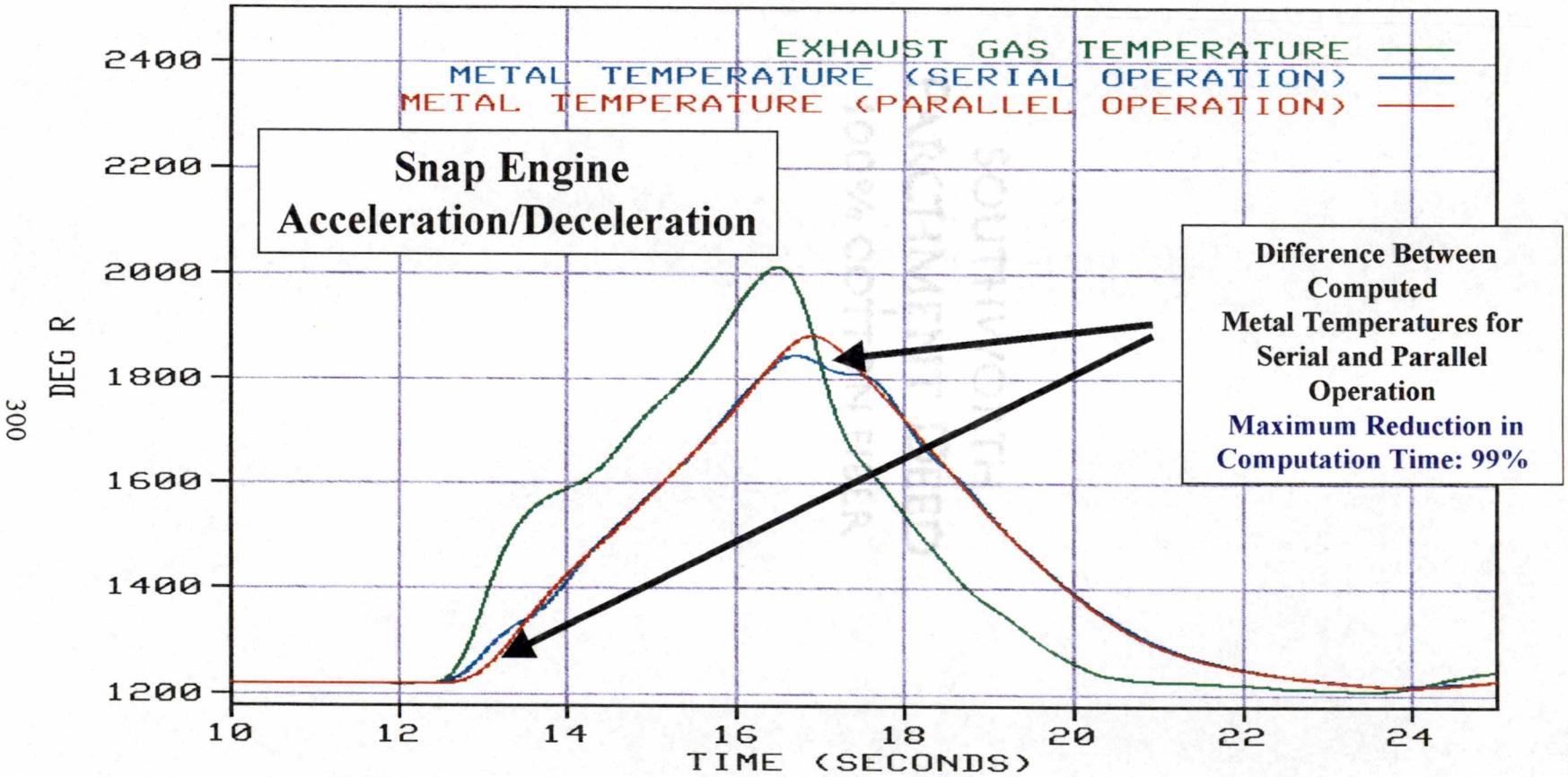


Fig. 4.27. Comparison of Predicted Metal Temperatures for Serial and Parallel Processing
(Filtered Data, 100 Processors, $\tau \equiv 1$ second)

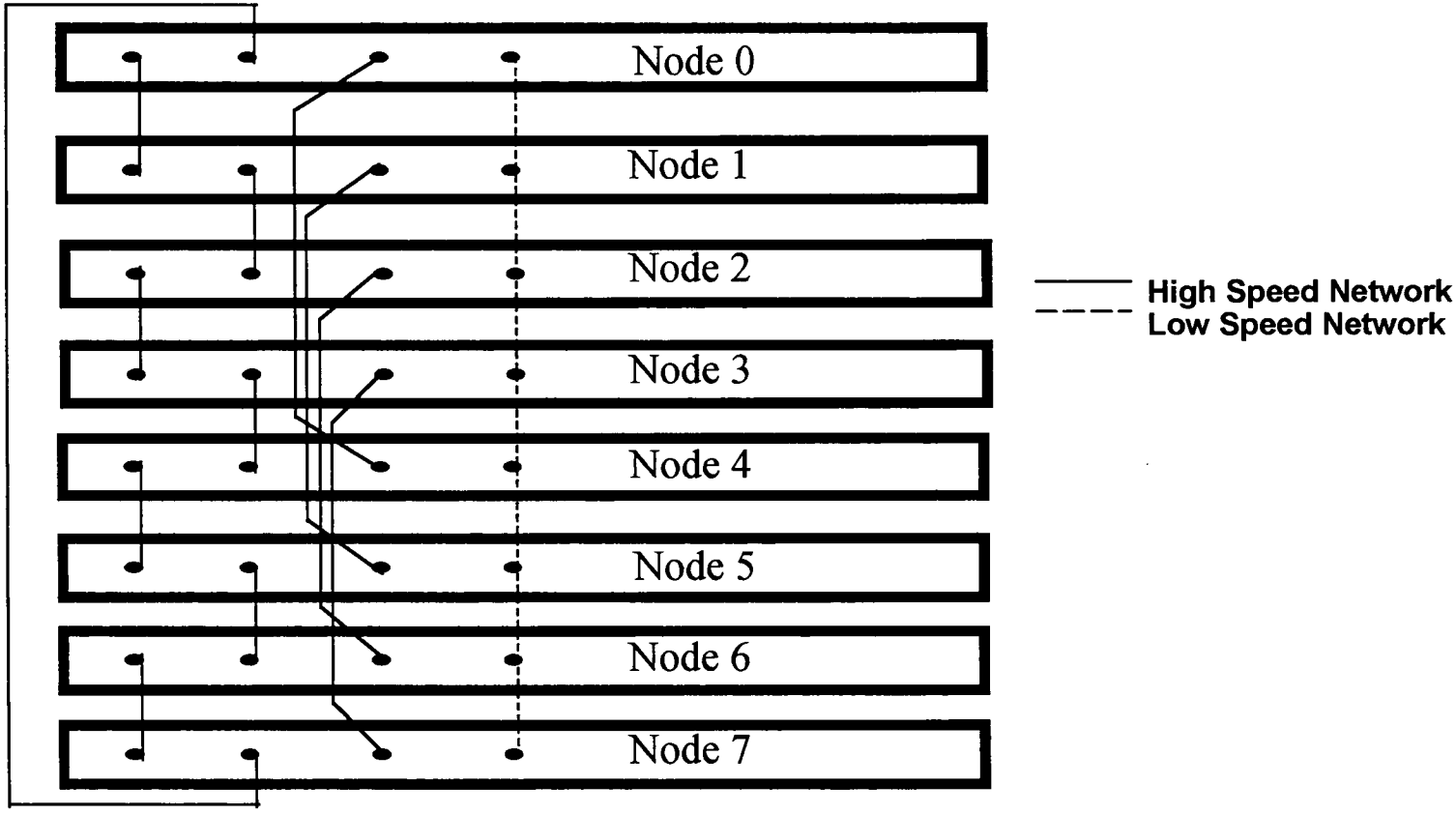


Fig. 5.1. Interconnections for Eight Processor Parallel Distributed Machine

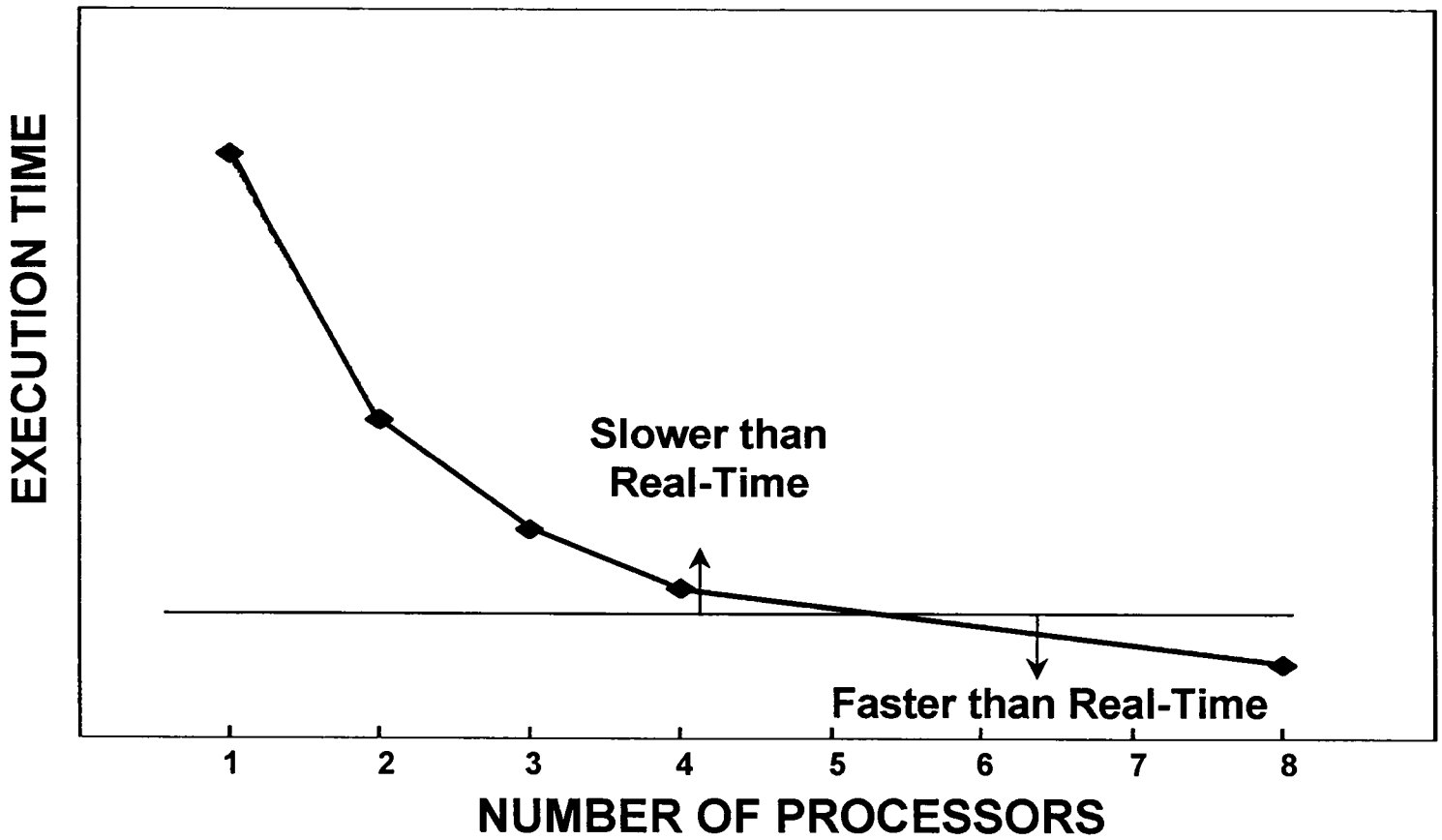


Fig. 5.2. Pentium Platform Effectiveness

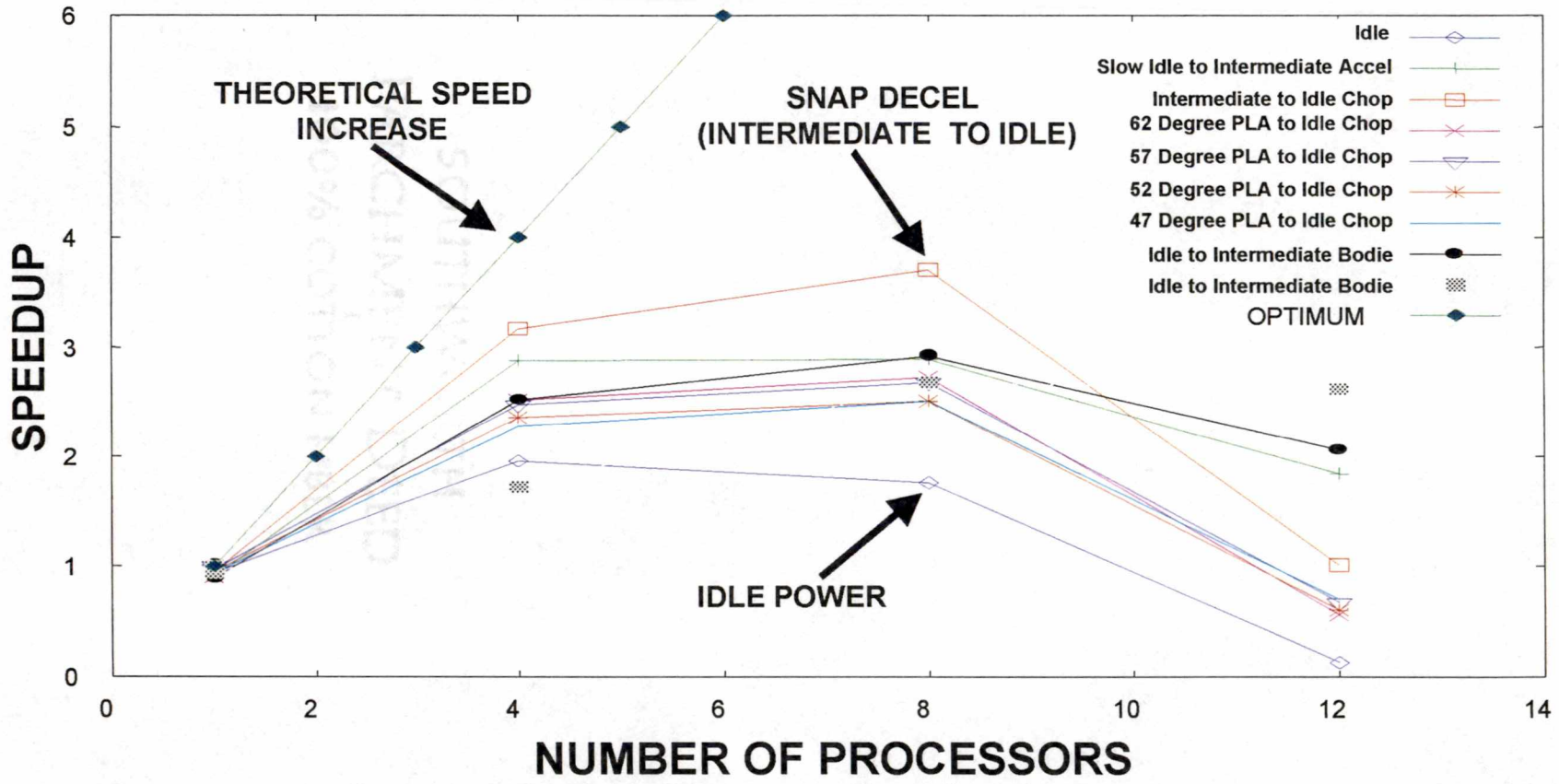


Fig. 5.3. Parallel Processing Effectiveness (SGI Origin 2000)

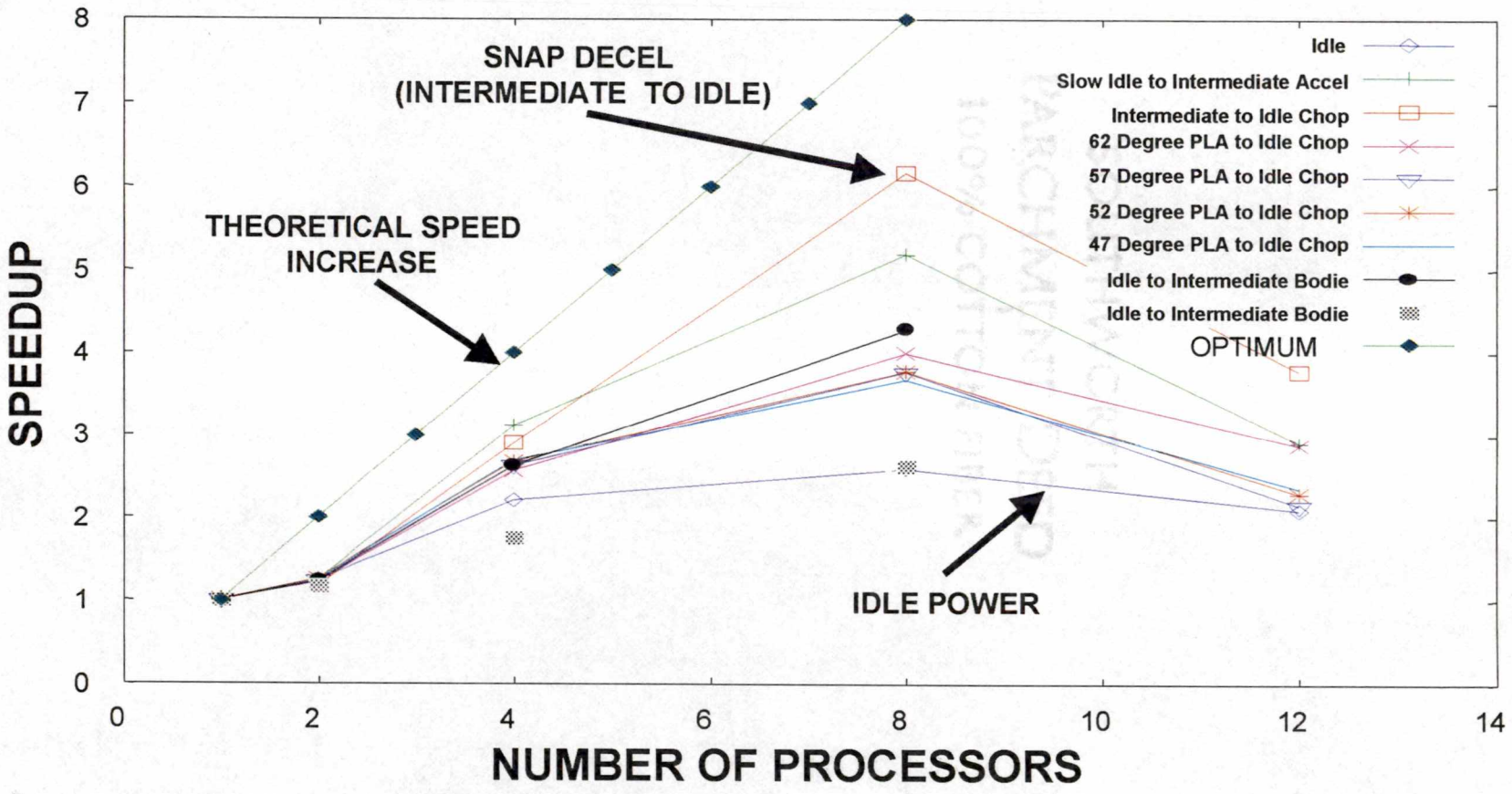


Fig. 5.4. Parallel Processing Effectiveness (HP-SPP 2000)

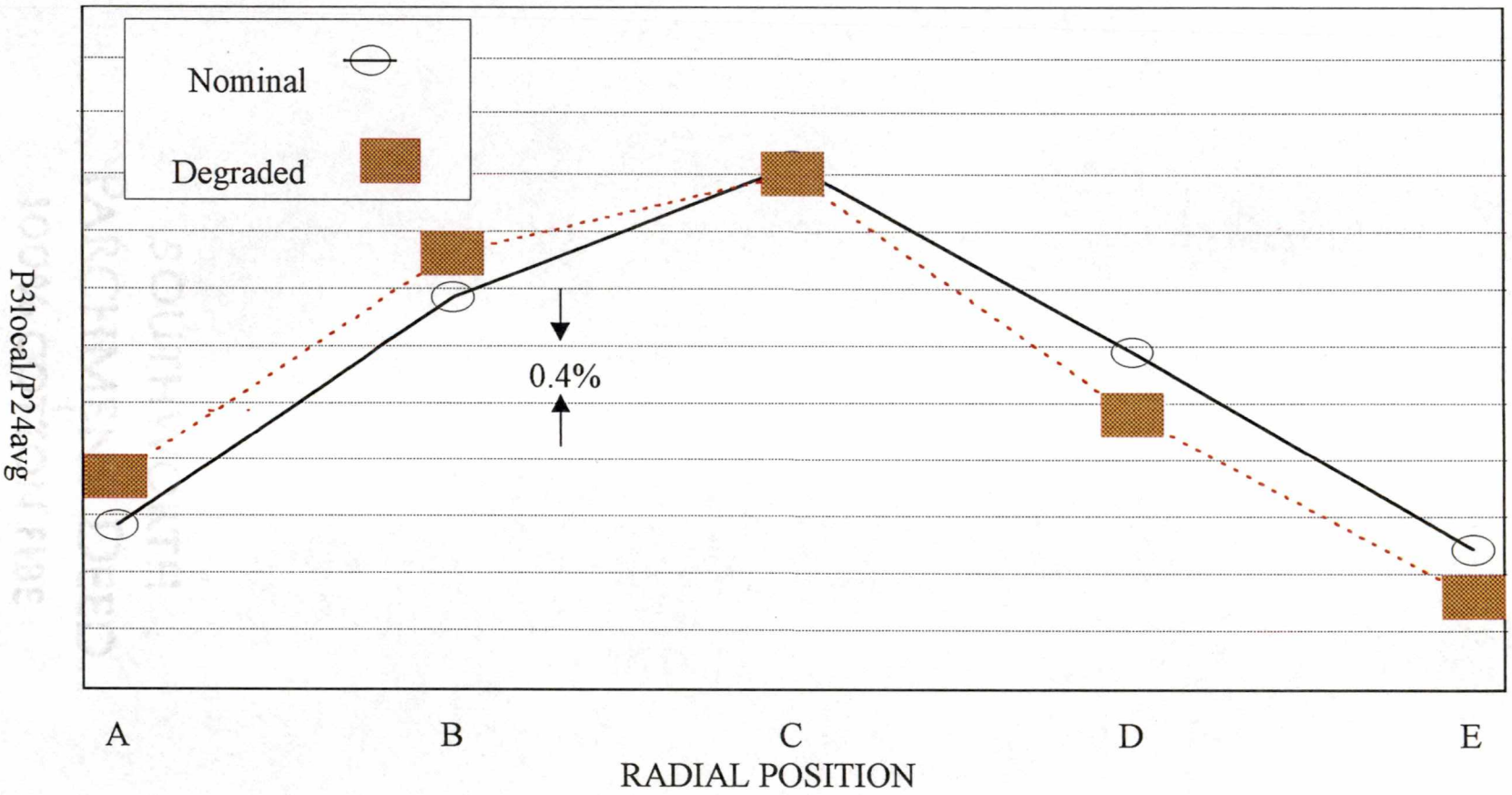


Fig. 6.1. Normalized Rake 1 Compressor Exit Total Pressures: Nominal and Degraded Operation

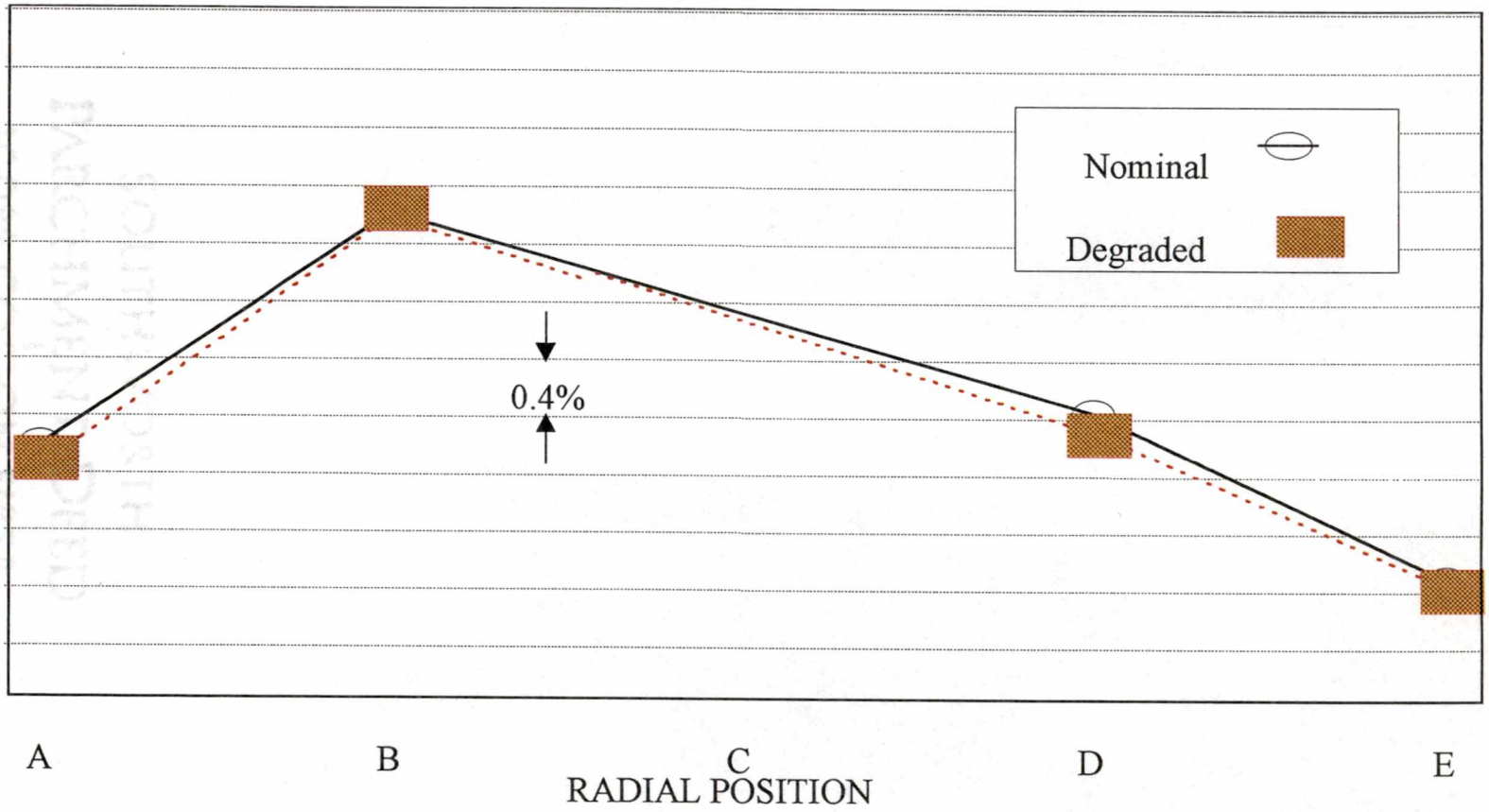


Fig. 6.2. Normalized Rake 2 Compressor Exit Total Pressures: Nominal and Degraded Operation

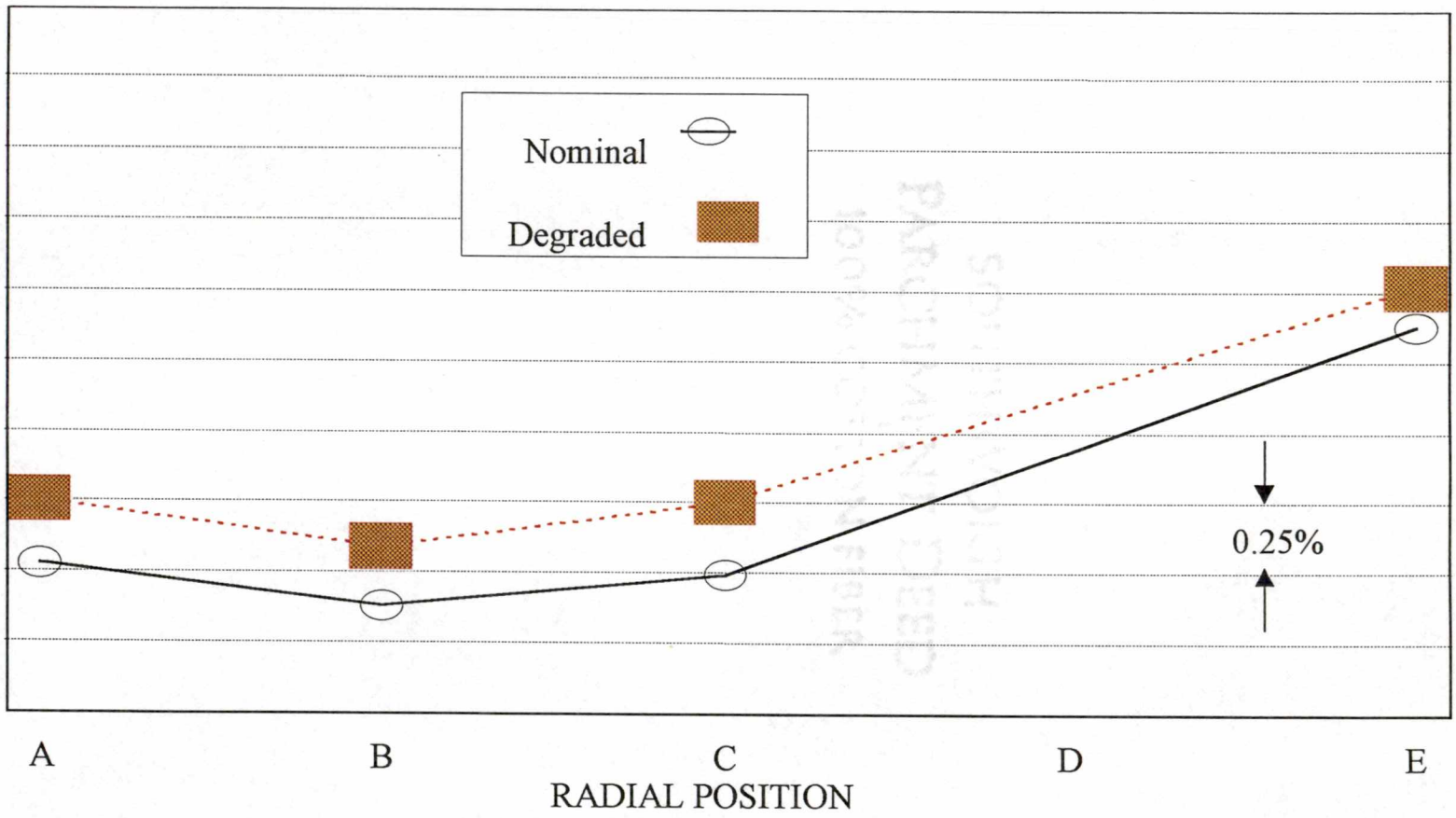


Fig. 6.3. Normalized Rake 3 Compressor Exit Total Temperatures: Nominal and Degraded Operation

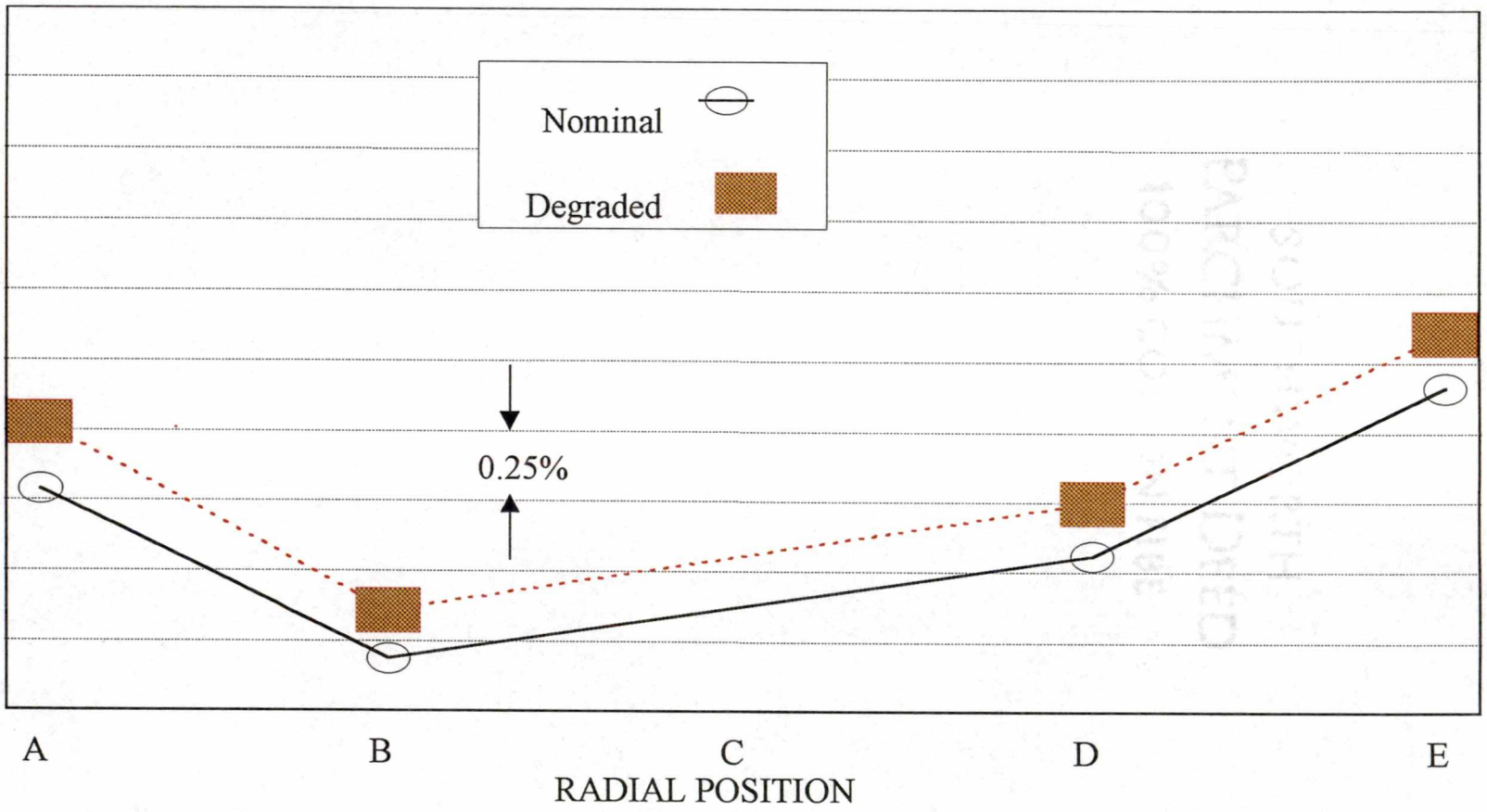


Fig. 6.4. Normalized Rake 4 Compressor Exit Total Temperatures: Nominal and Degraded Operation

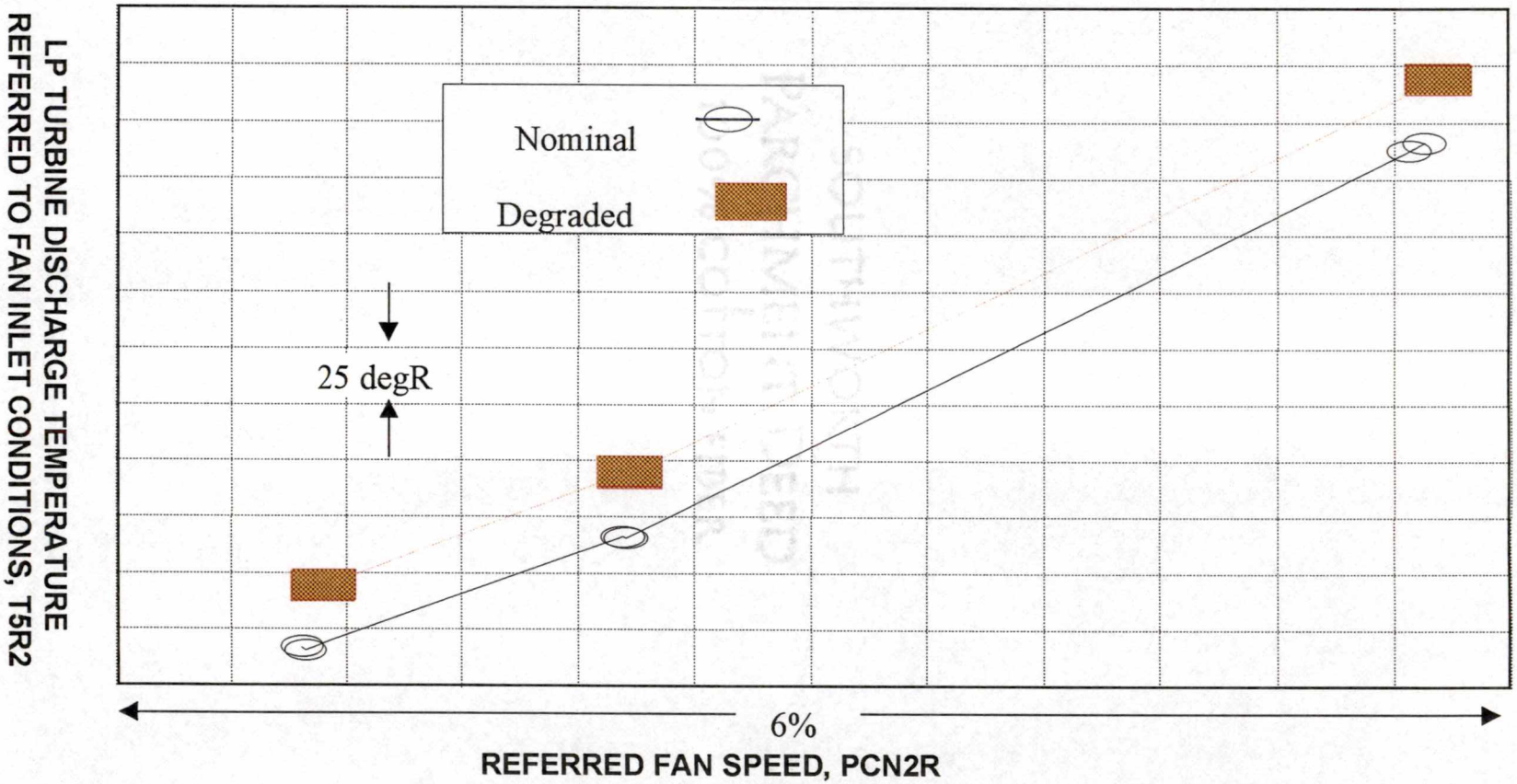


Fig. 6.5. Variation in LP Turbine Discharge Temperature Referred to Fan Inlet Conditions: Nominal and Degraded Operation

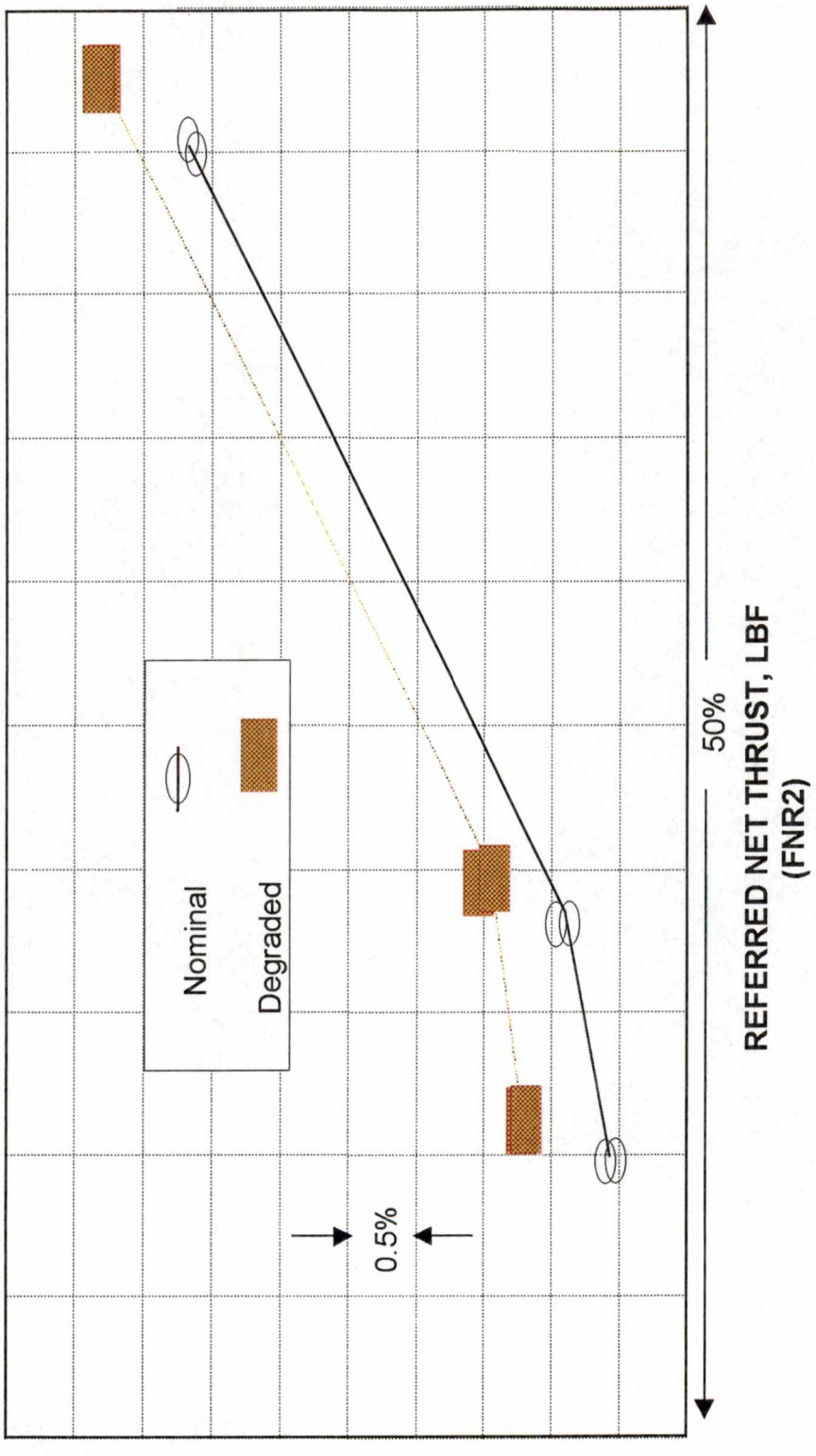


Fig. 6.6. Variation in Referred Specific Fuel Consumption: Nominal and Degraded Operation

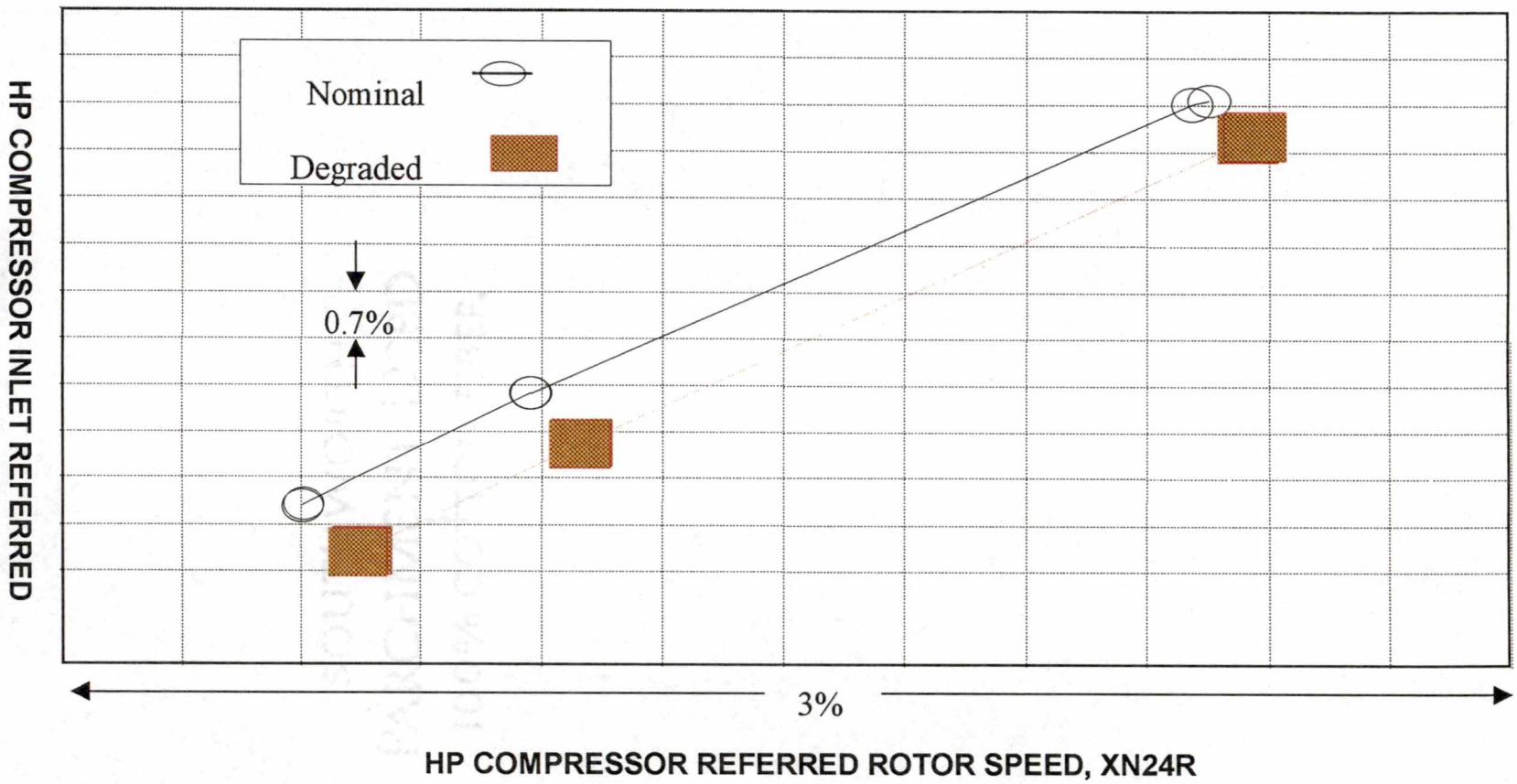


Fig. 6.7. Variation in HP Compressor Airflow Pumping Capacity: Nominal and Degraded Operation

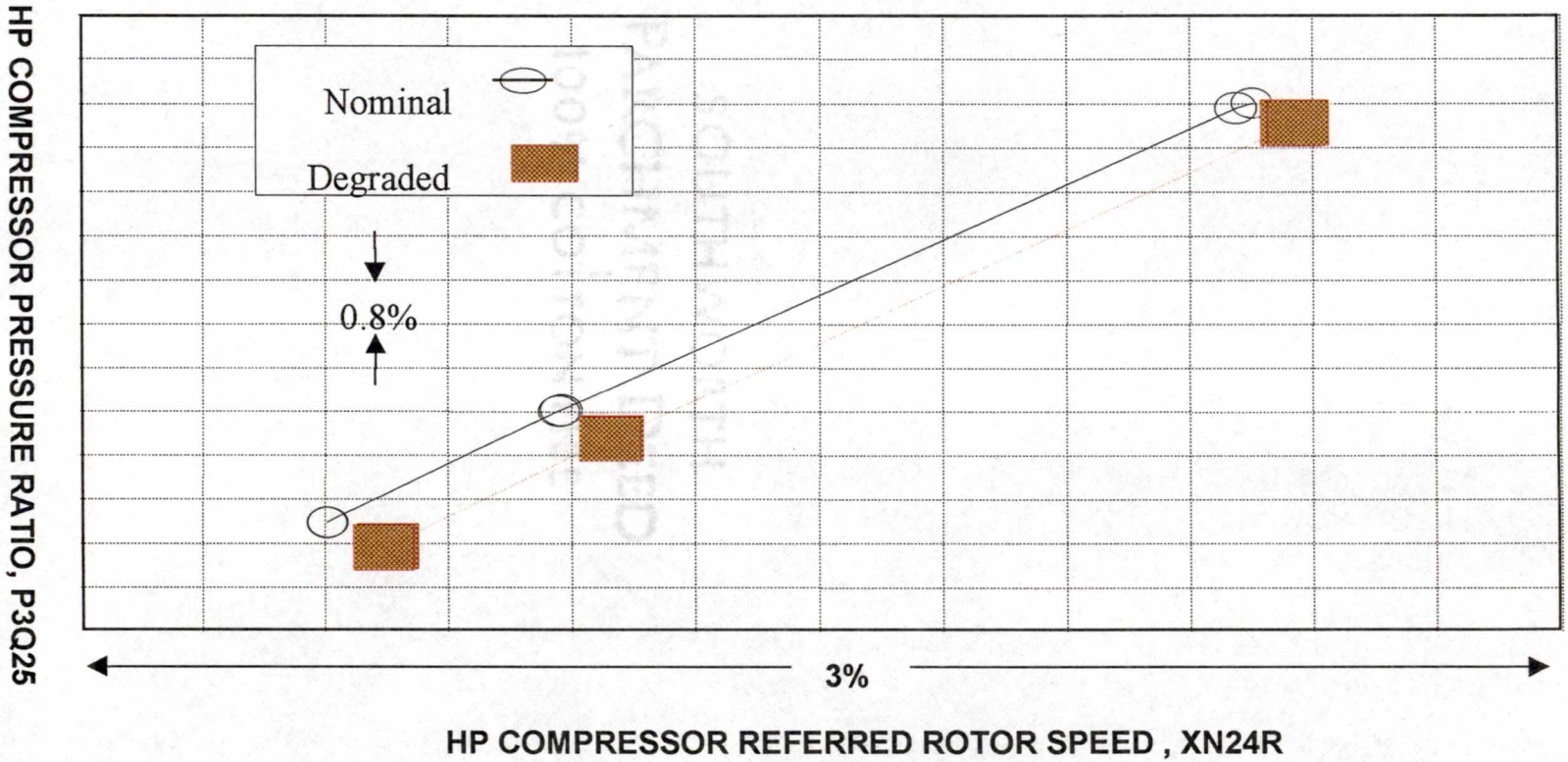


Fig. 6.8. Variation in HP Compressor Pressure Ratio: Nominal and Degraded Operation

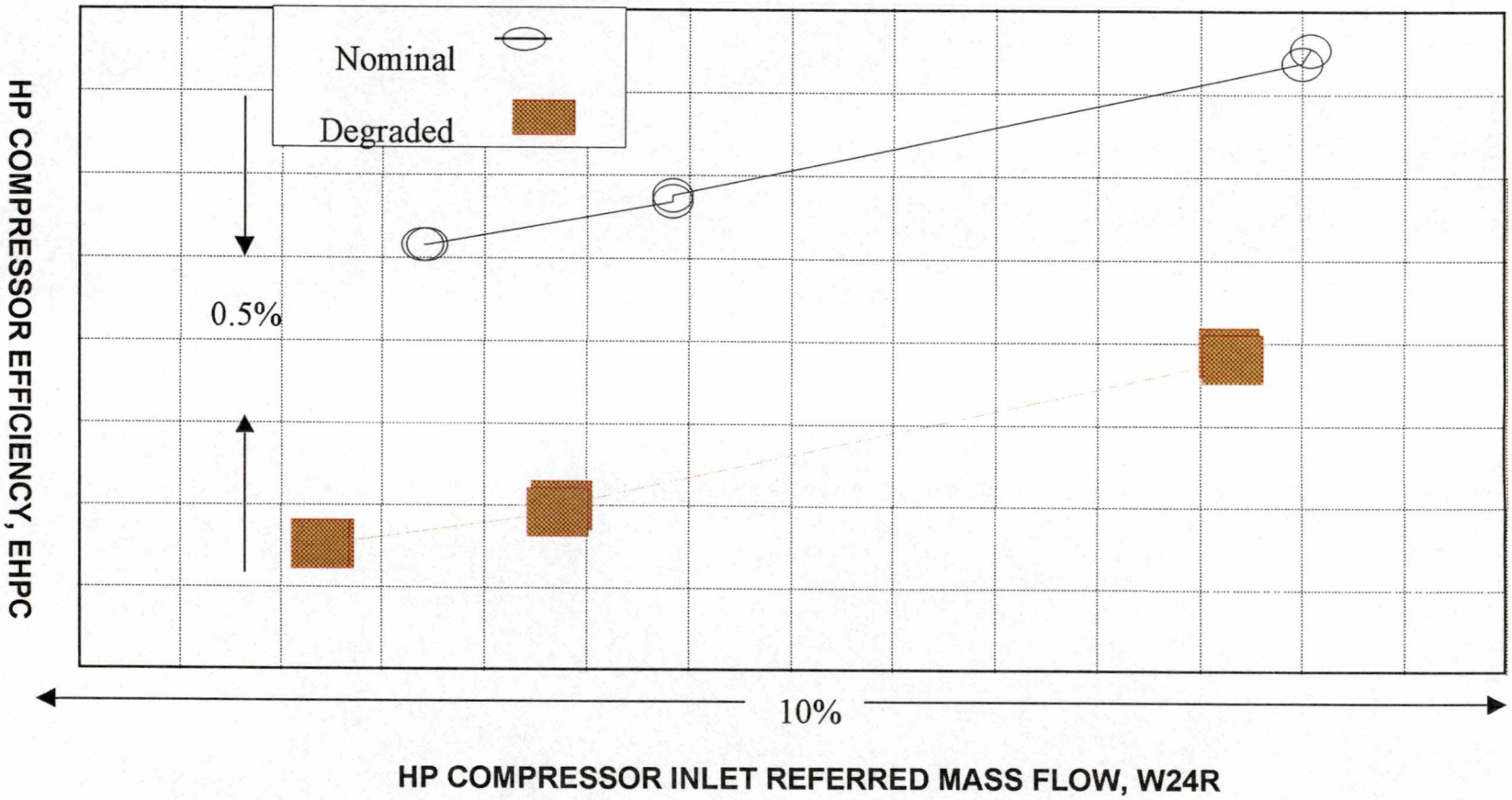


Fig. 6.9. Variation in HP Compressor Efficiency: Nominal and Degraded Operation

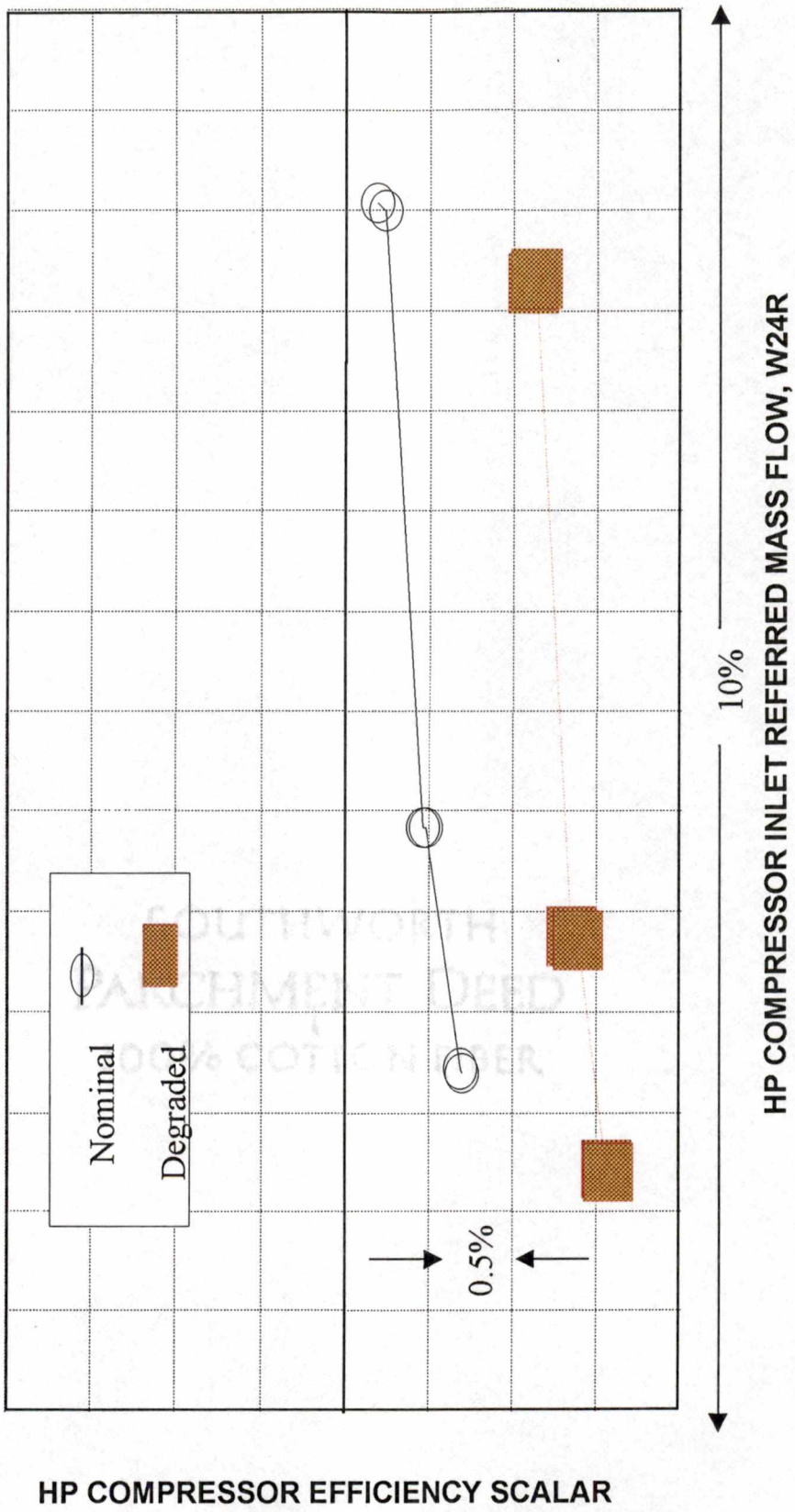


Fig. 6.10. HP Compressor Efficiency Map Scalar: Nominal and Degraded Operation

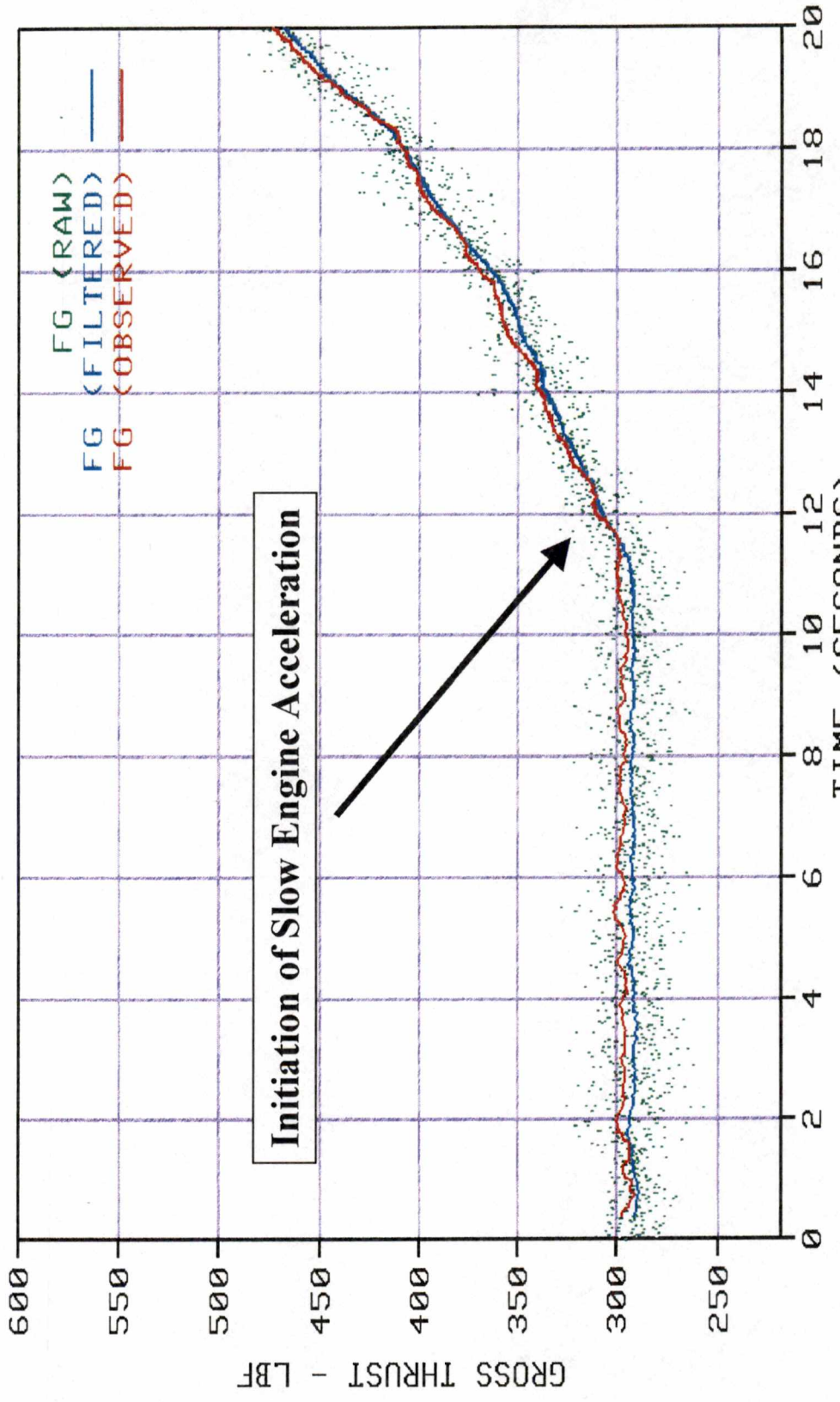


Fig. 7.1. Predicted and Measured Engine Gross Thrust (Initiation of Slow Engine Acceleration)

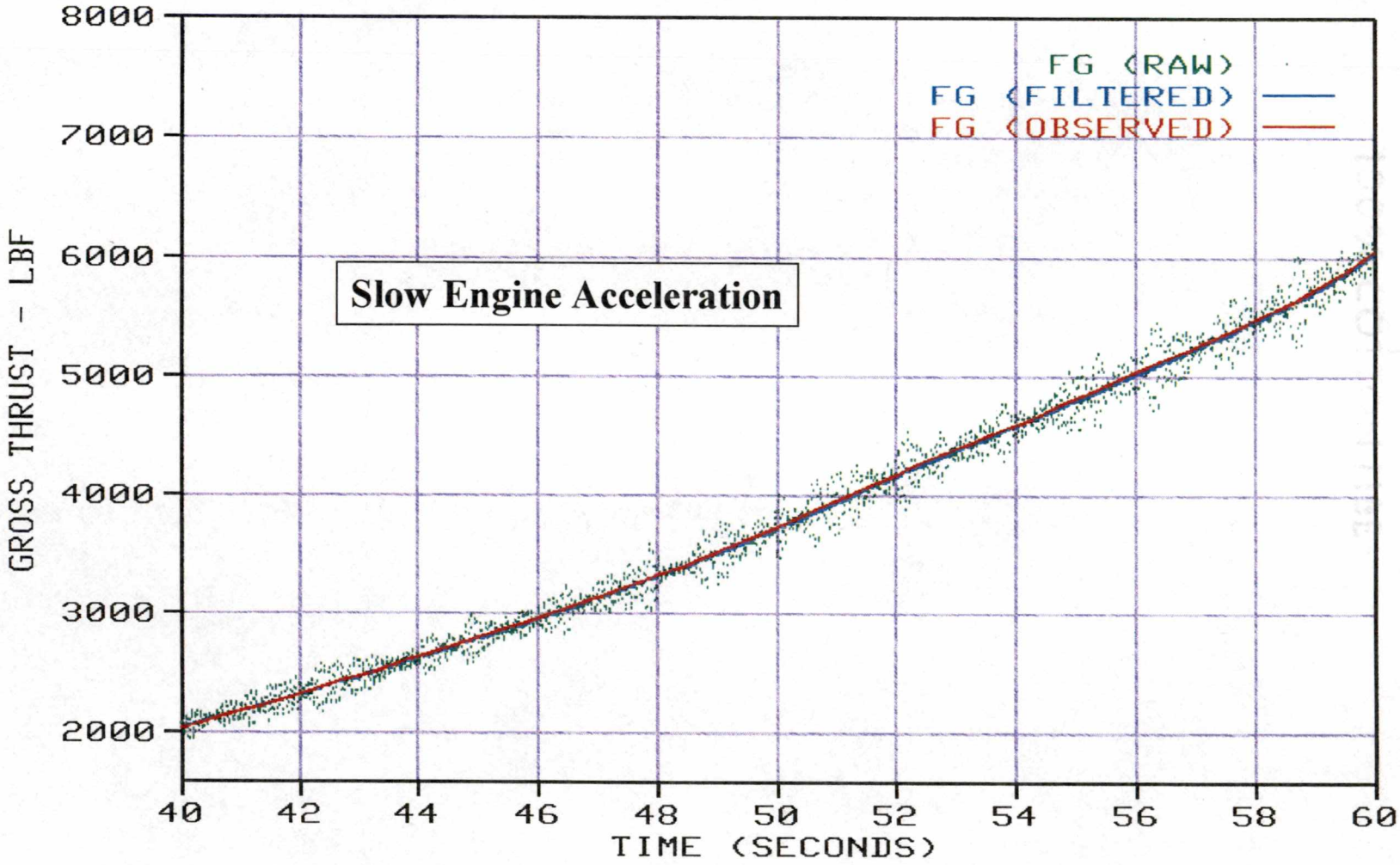


Fig. 7.2. Predicted and Measured Engine Gross Thrust (Part-Way Through Slow Engine Acceleration)

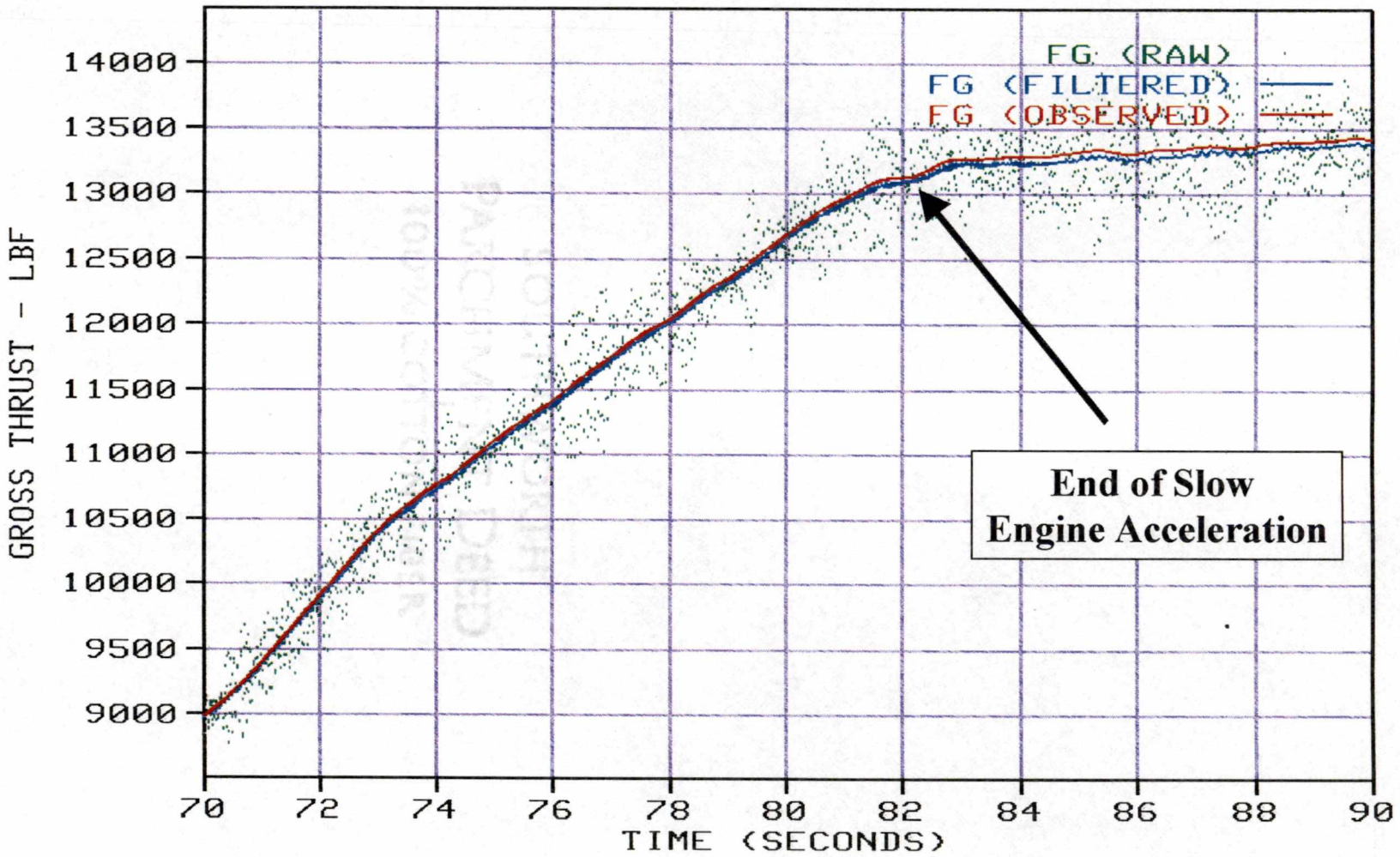


Fig. 7.3. Predicted and Measured Engine Gross Thrust (End of Slow Engine Acceleration)

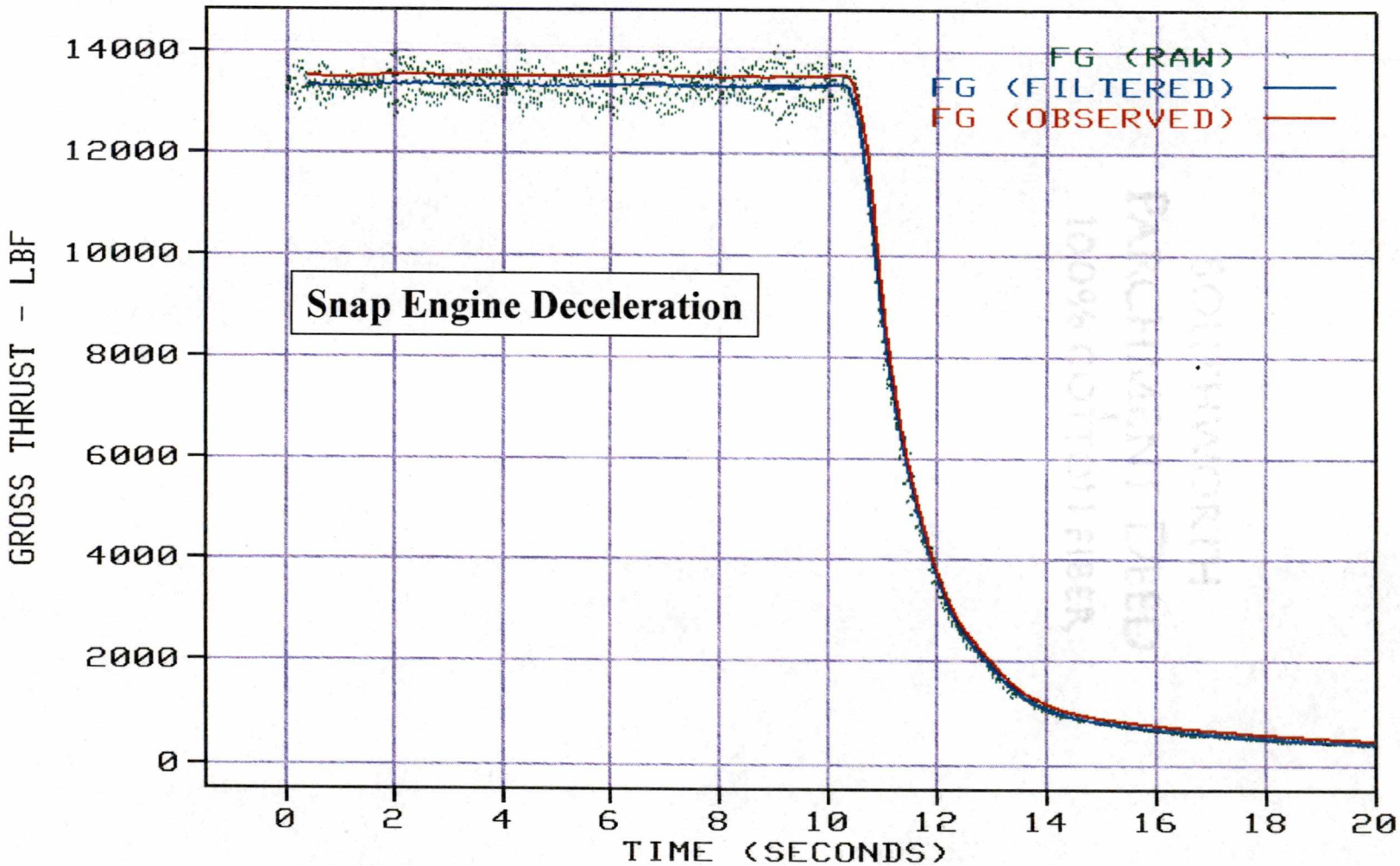


Fig. 7.4. Predicted and Measured Engine Gross Thrust (Snap Engine Deceleration)

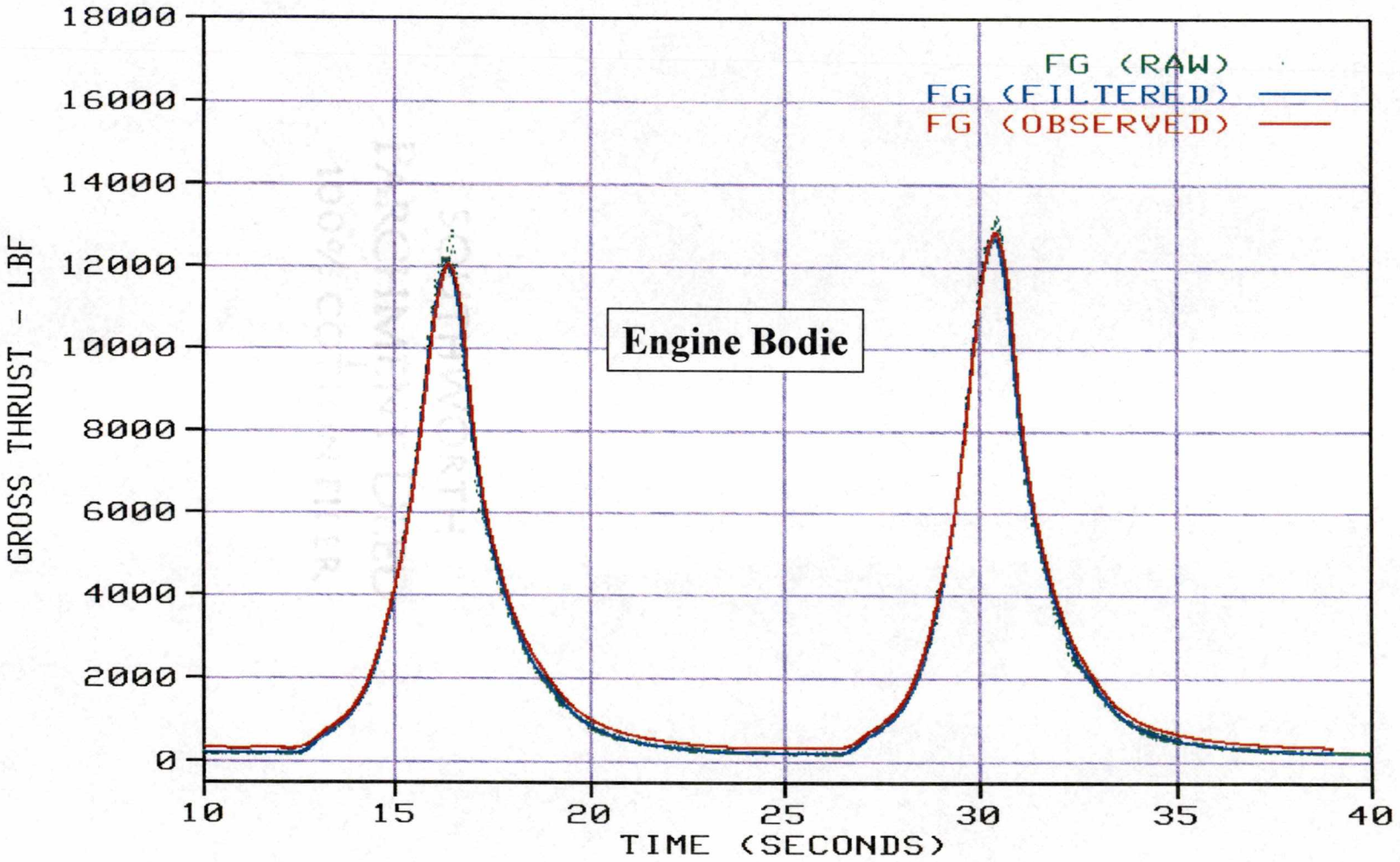


Fig. 7.5. Predicted and Measured Engine Gross Thrust (Engine Bodie)

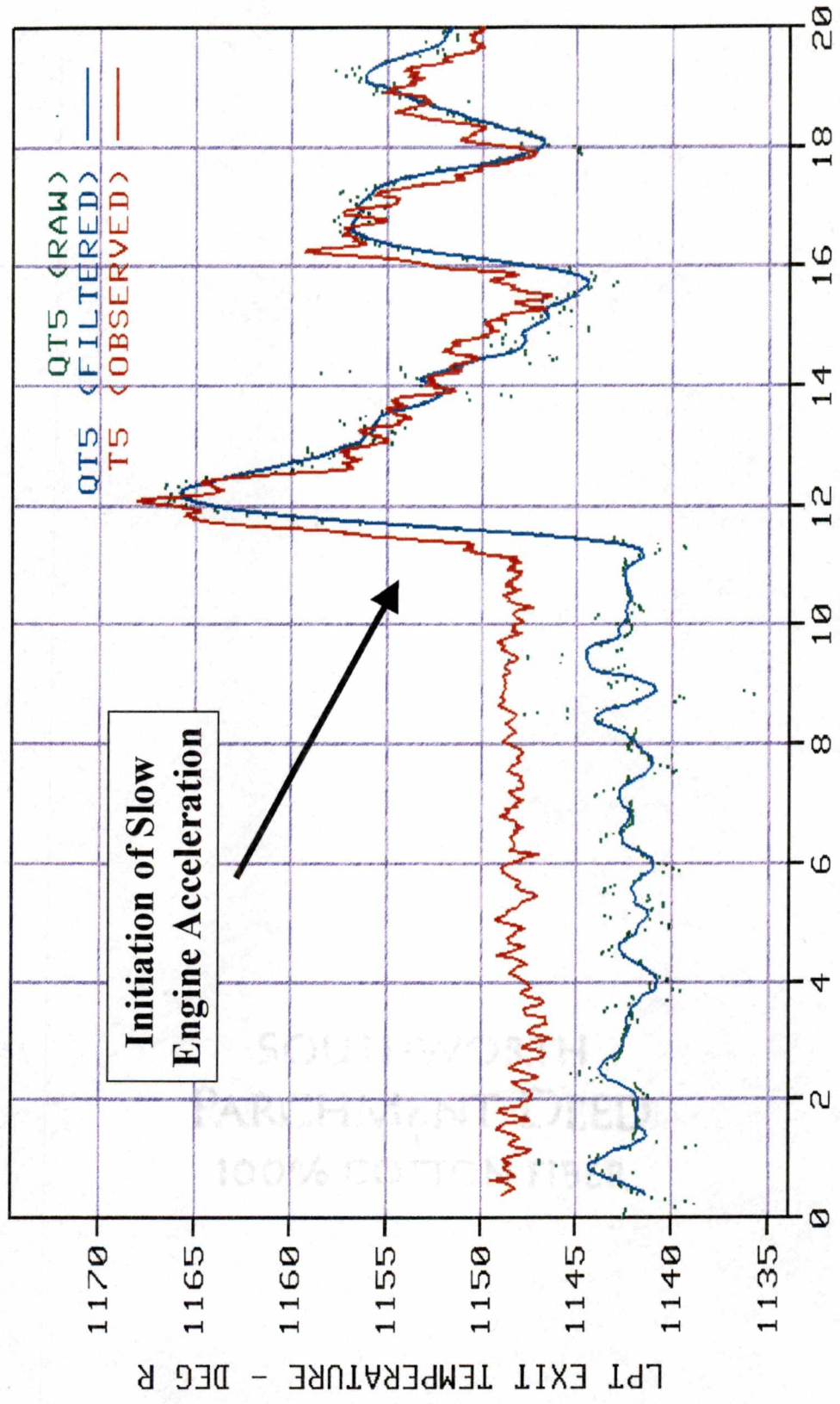


Fig. 7.6. Predicted and Measured LPT Exit Temperature (Initiation of Slow Engine Acceleration)

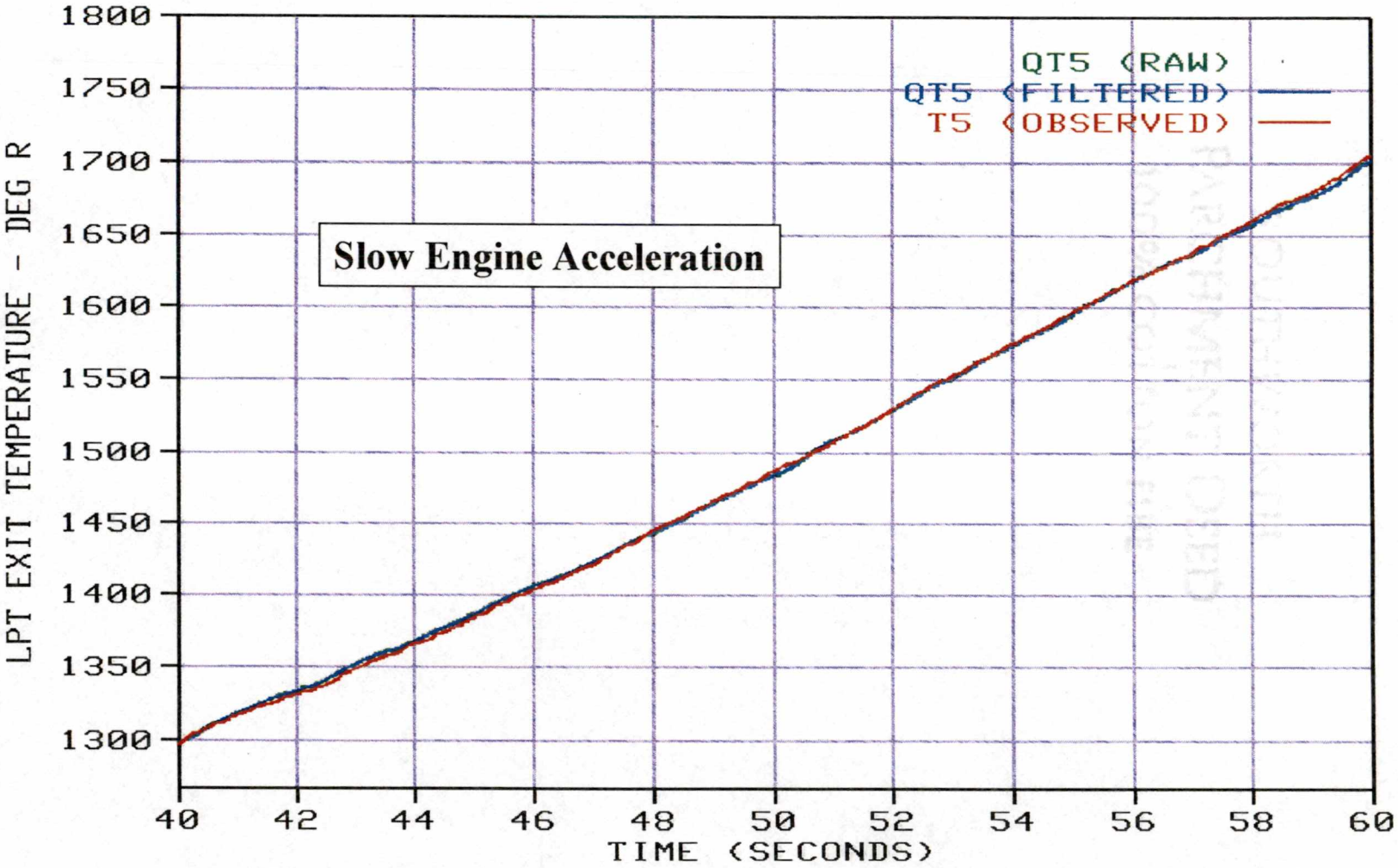


Fig. 7.7. Predicted and Measured LPT Exit Temperature (Part-Way Through Slow Engine Acceleration)

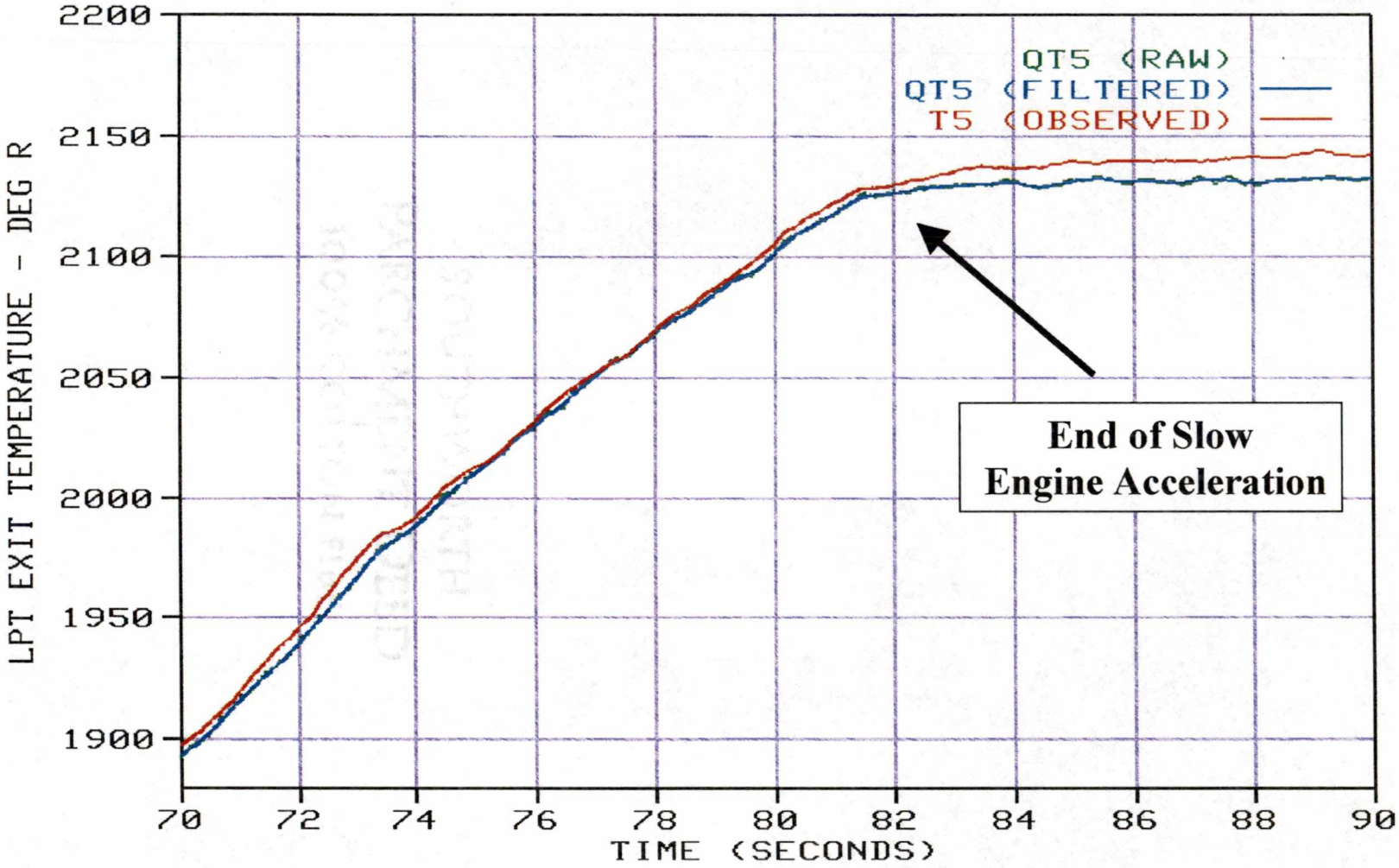


Fig. 7.8. Predicted and Measured LPT Exit Temperature (End of Slow Engine Acceleration)

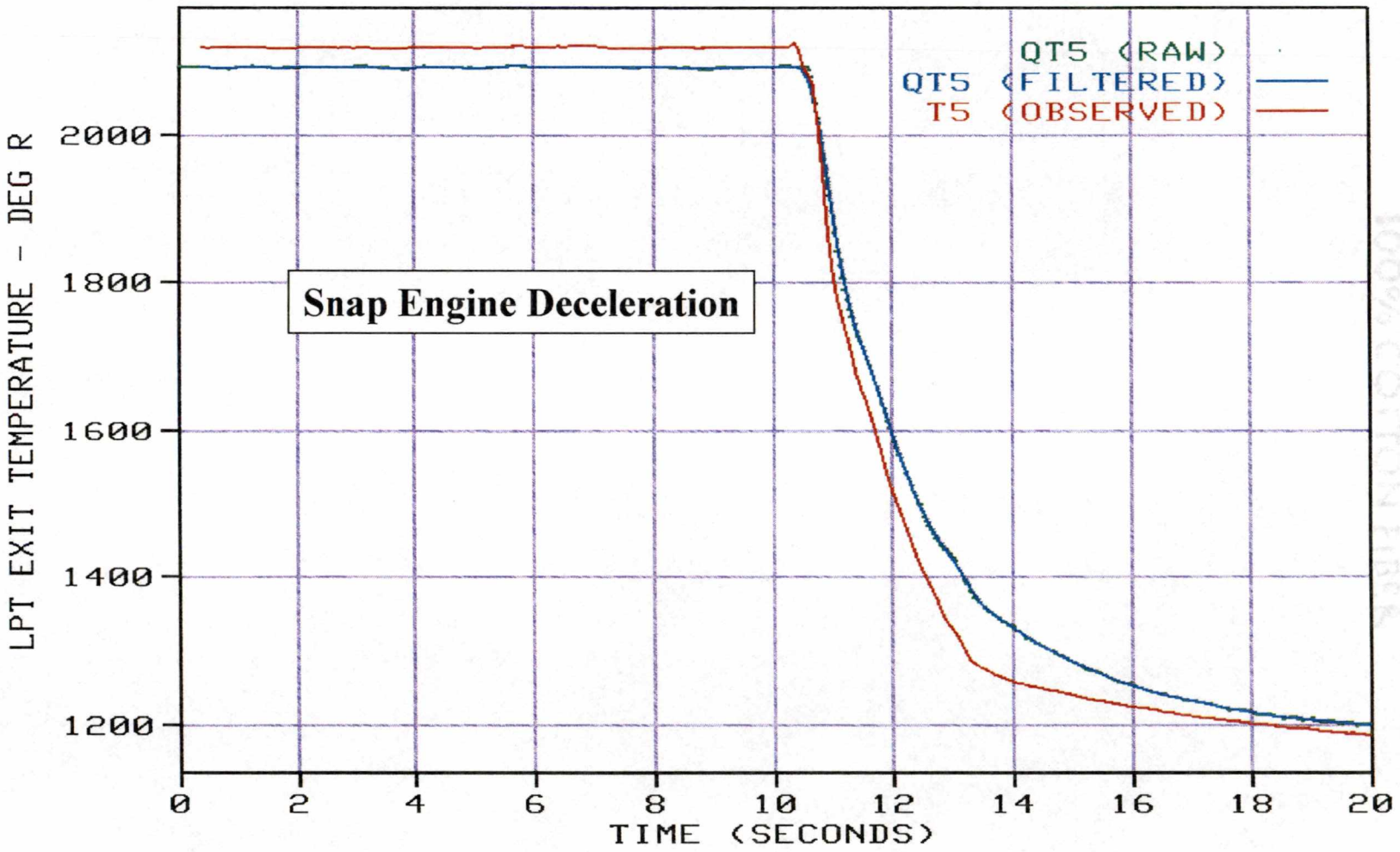


Fig. 7.9. Predicted and Measured LPT Exit Temperature (Snap Engine Deceleration)

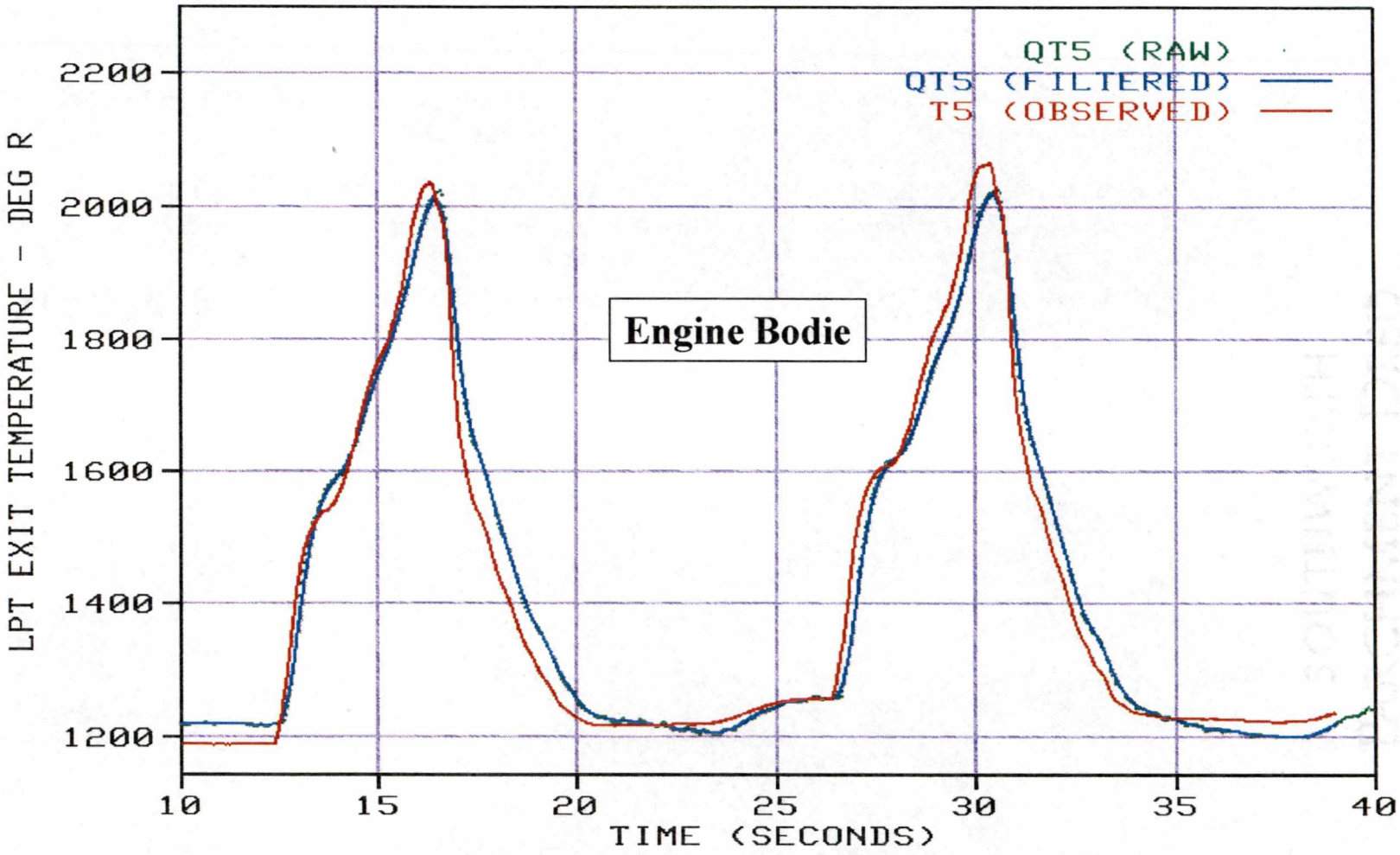


Fig. 7.10. Predicted and Measured LPT Exit Temperature (Engine Bodie)

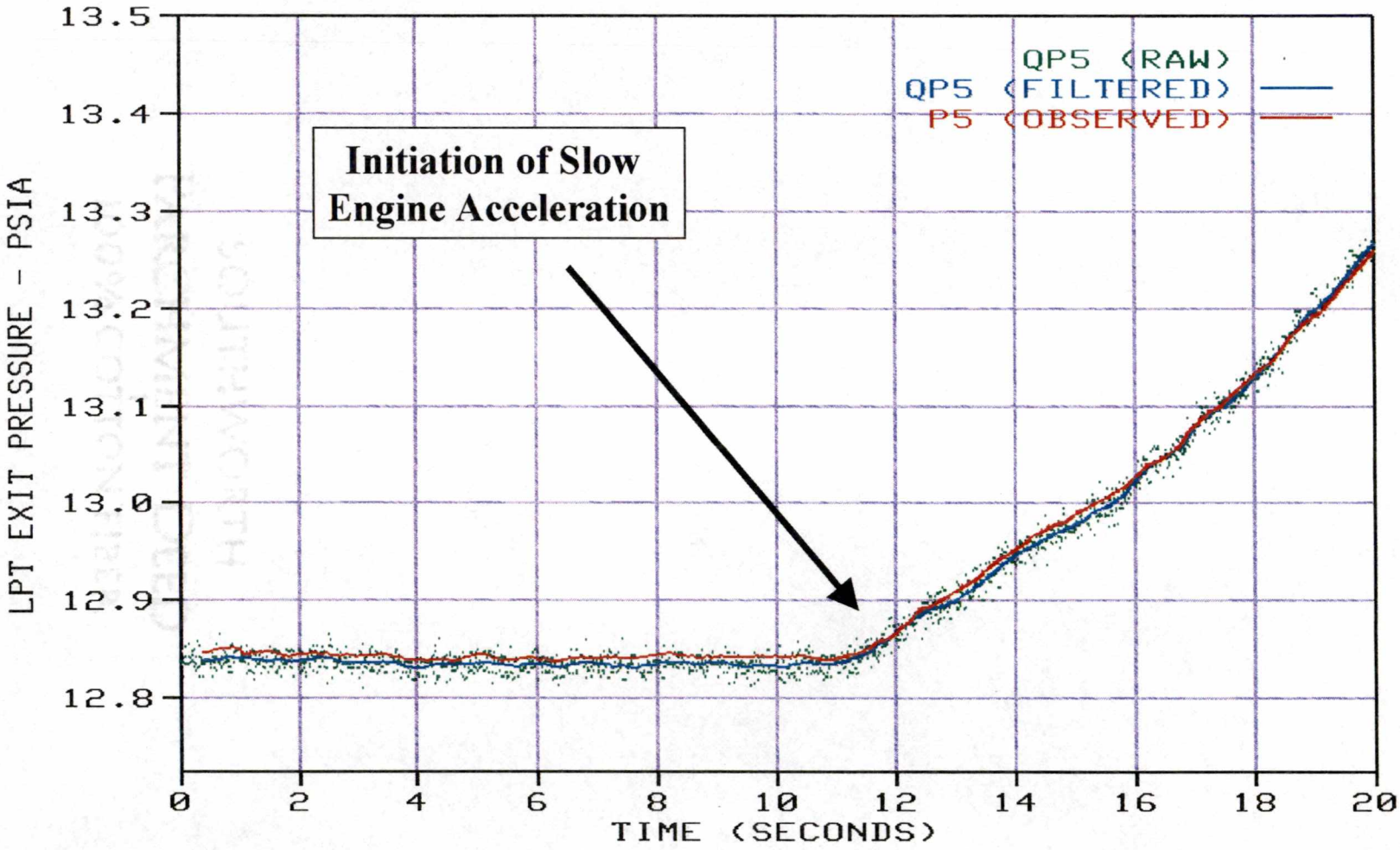


Fig. 7.11. Predicted and Measured LPT Exit Pressure (Initiation of Slow Engine Acceleration)

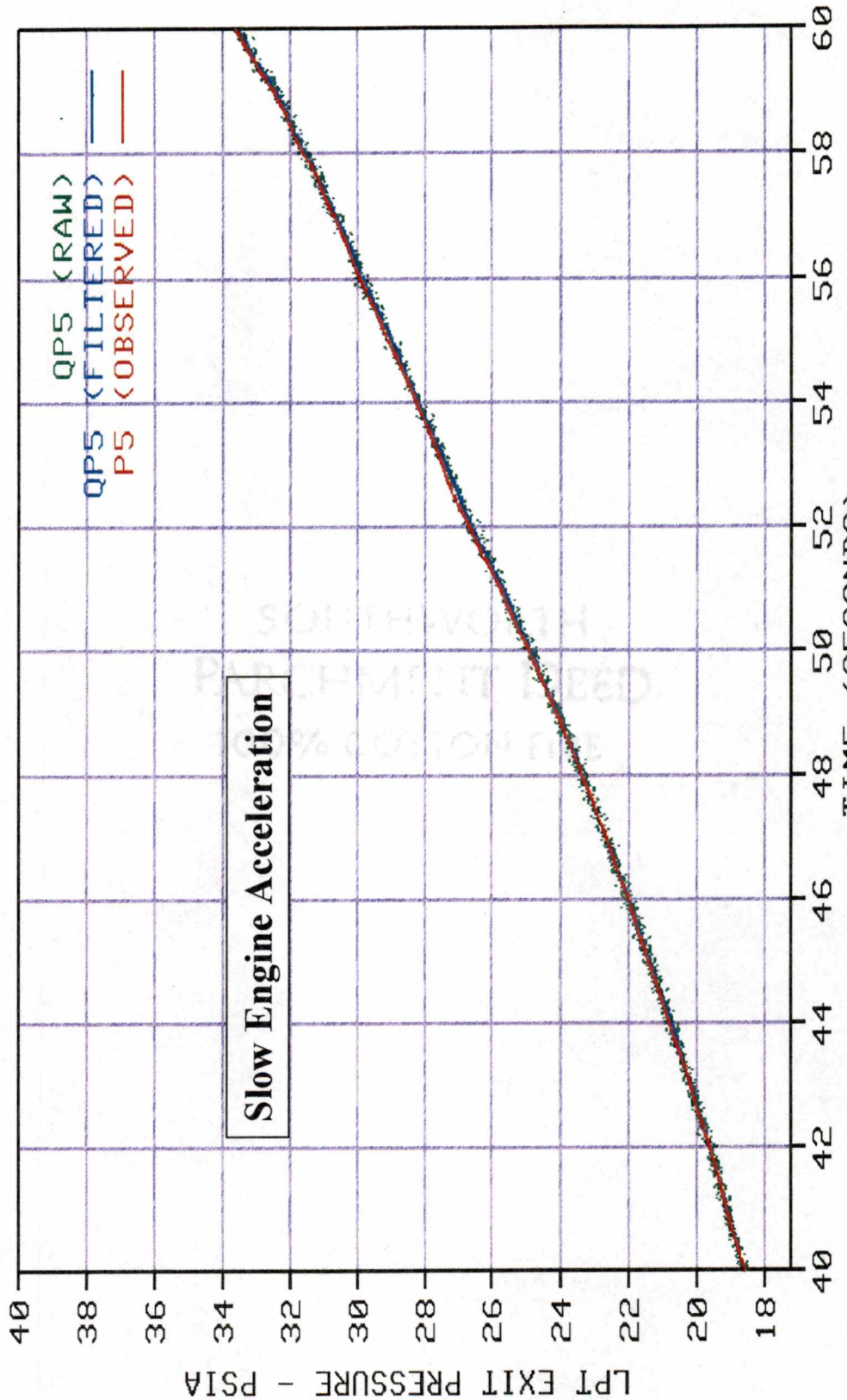


Fig. 7.12. Predicted and Measured LPT Exit Pressure (Part-Way Through Slow Engine Acceleration)

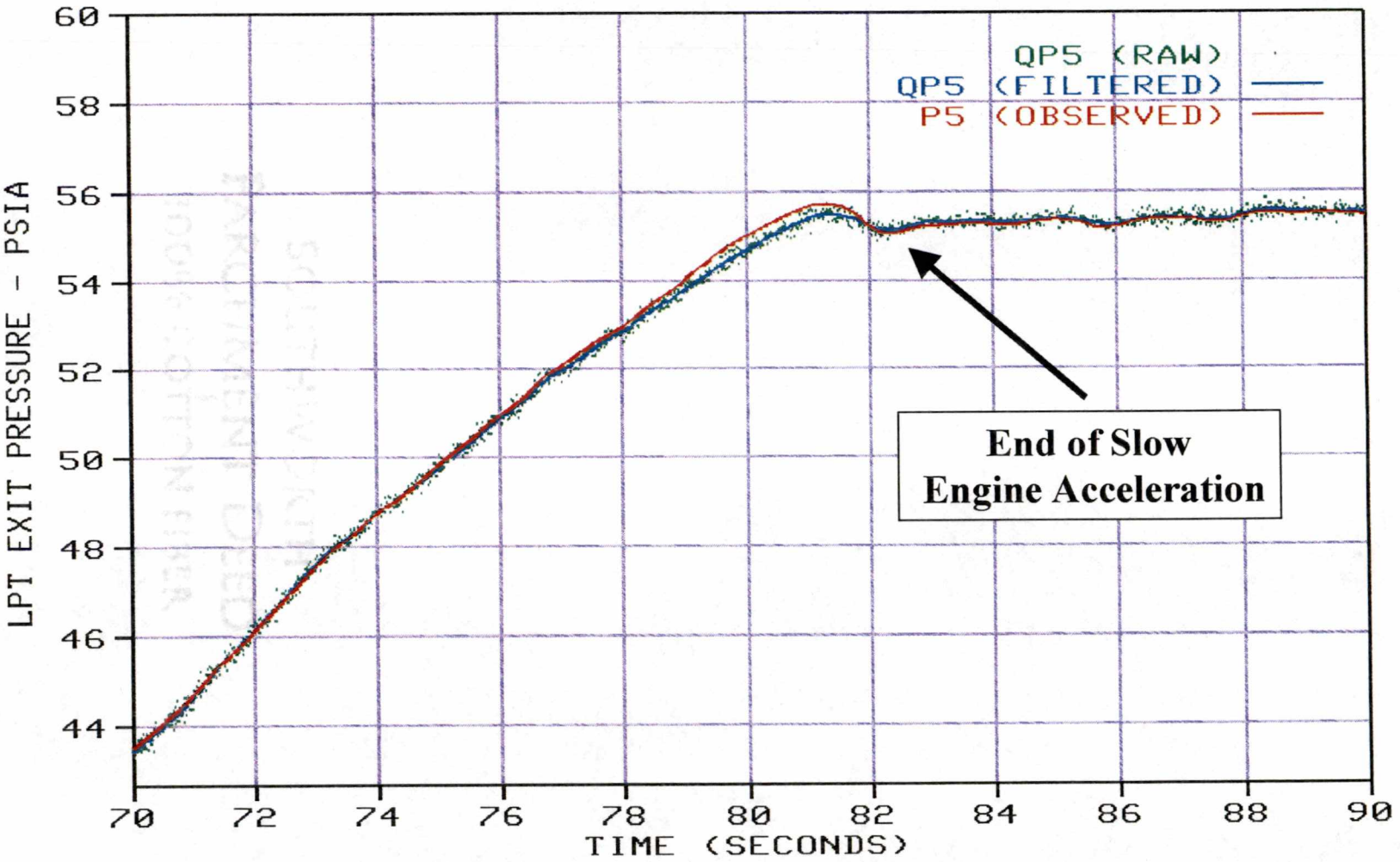


Fig. 7.13. Predicted and Measured LPT Exit Pressure (End of Slow Engine Acceleration)

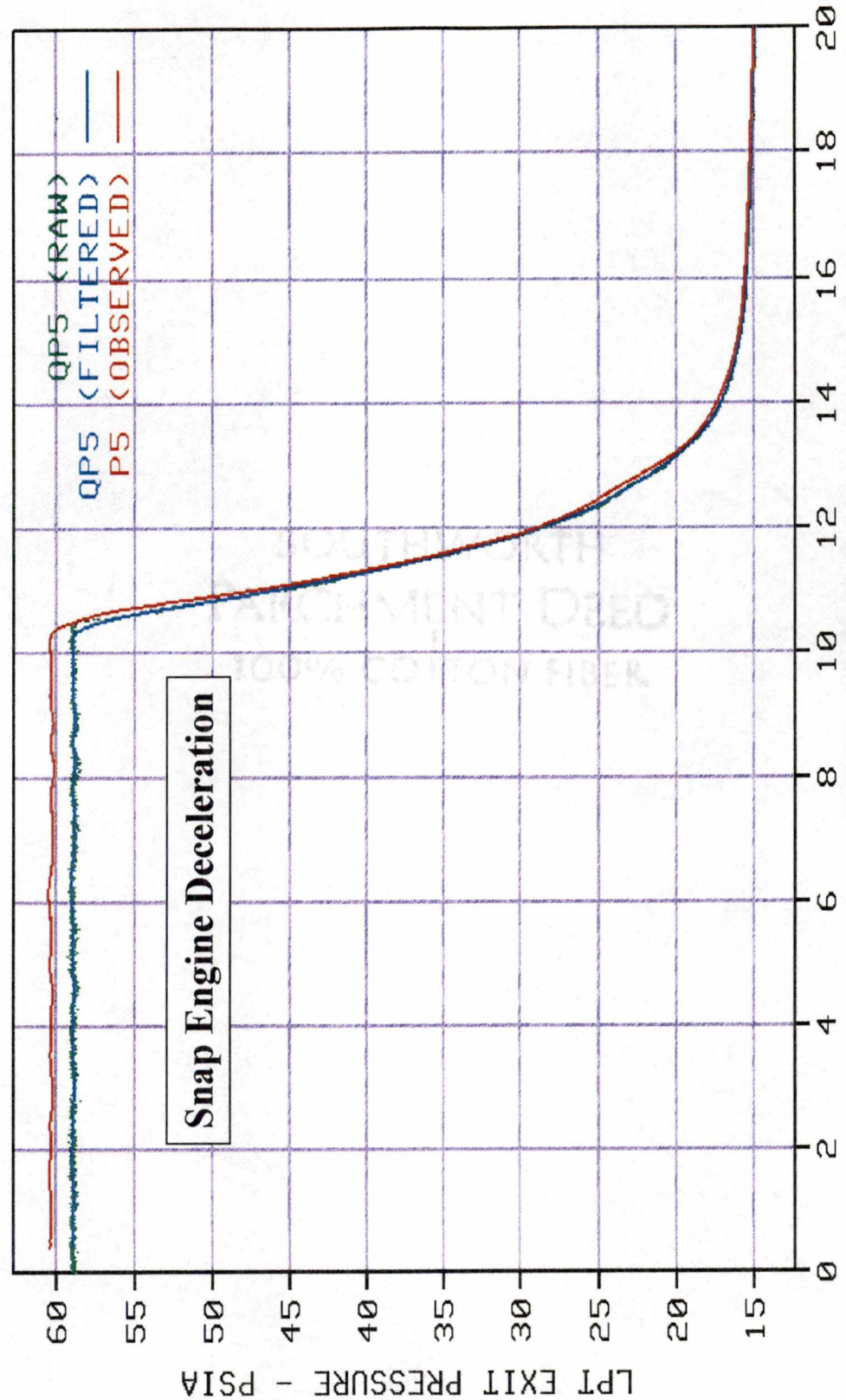


Fig. 7.14. Predicted and Measured LPT Exit Pressure (Snap Engine Deceleration)

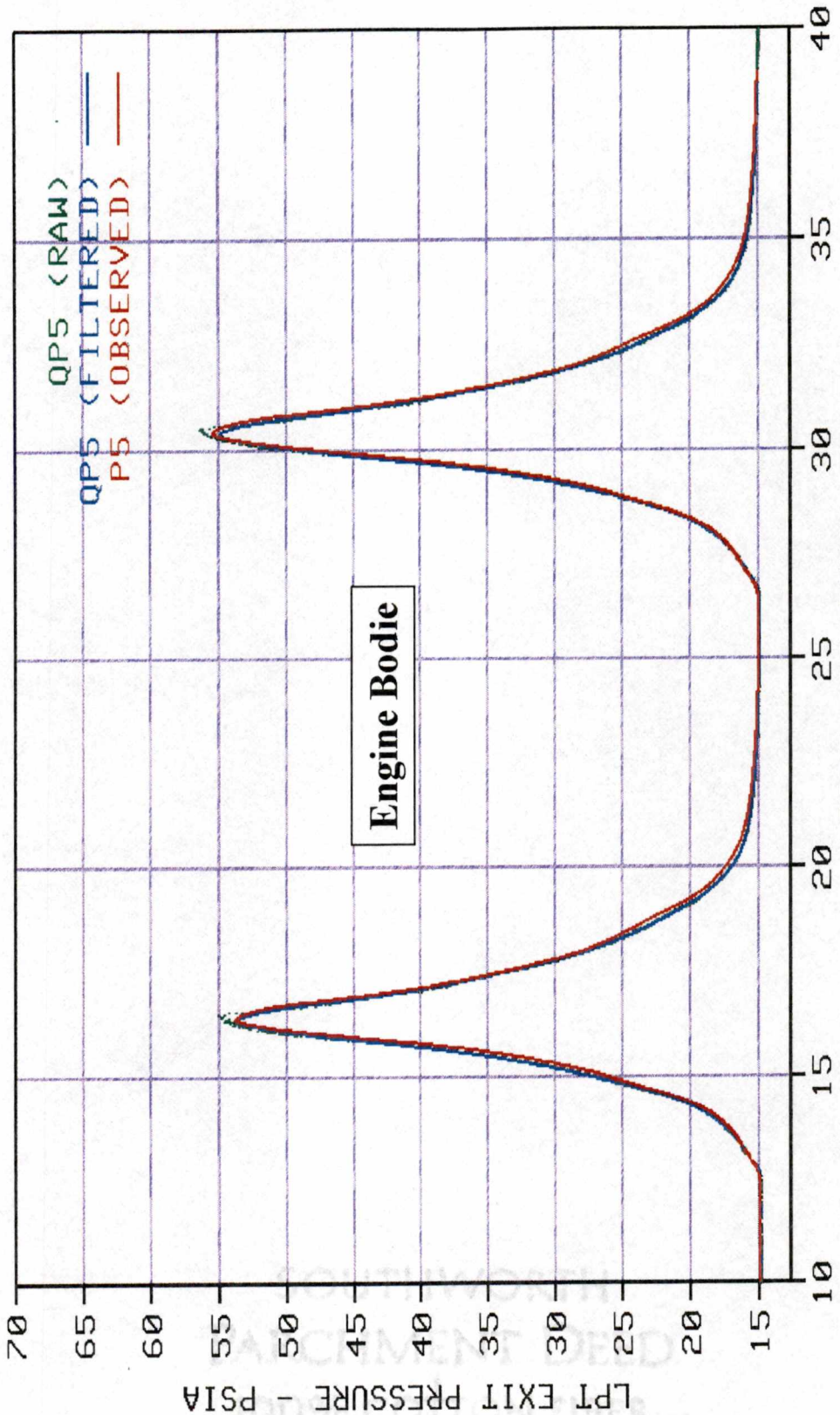


Fig. 7.15. Predicted and Measured LPT Exit Pressure (Engine Bodie)

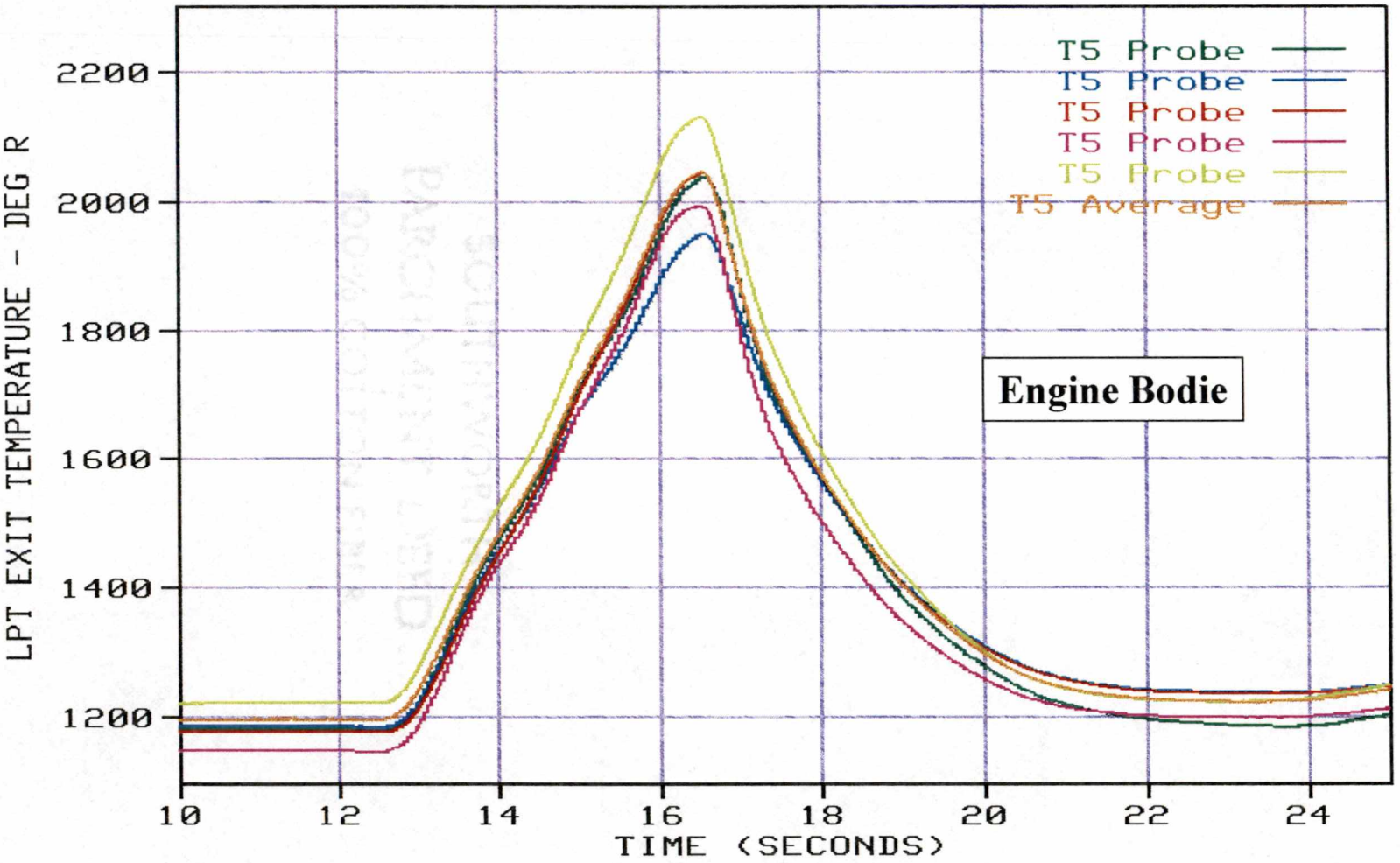


Fig. 7.16. Representative Turbine Exit Temperature Measurements

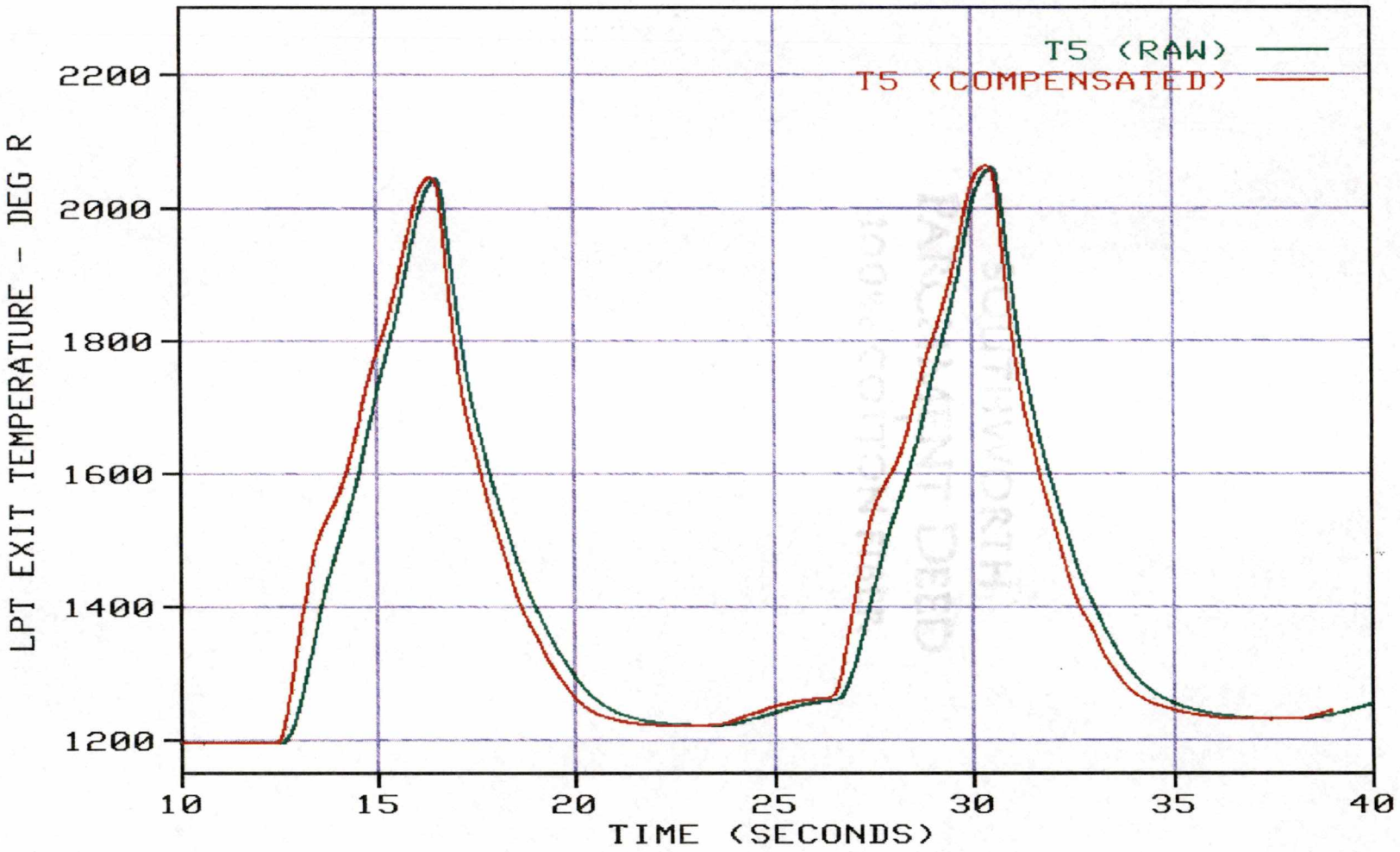


Fig. 7.17. Lag Compensated Turbine Exit Temperature

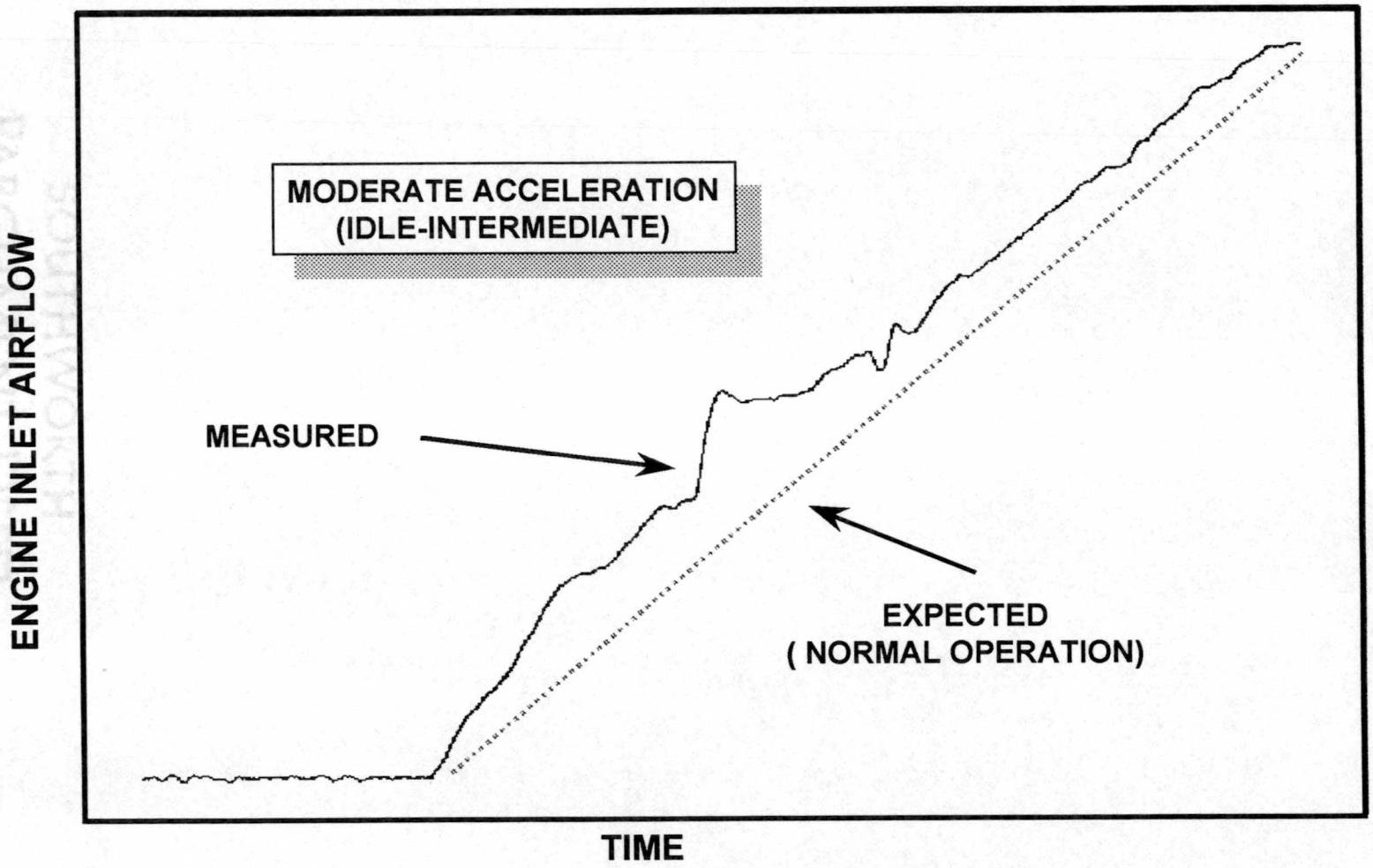


Fig. 7.18. Effects of Propulsion Control Schedules on Airflow Measurement

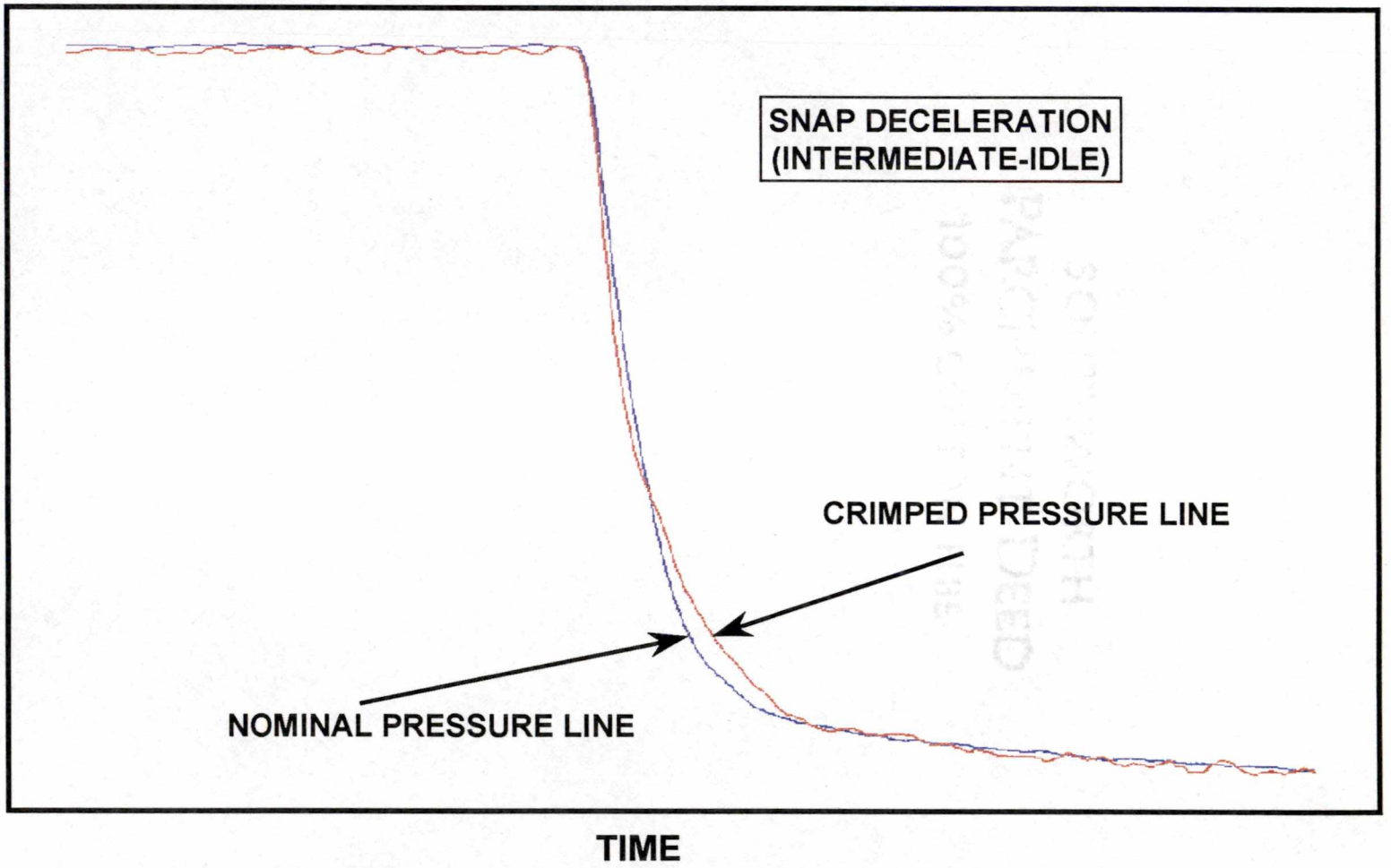


Fig. 7.19. Effect of Slowly Responding Pressure Measurement on Airflow Measurement

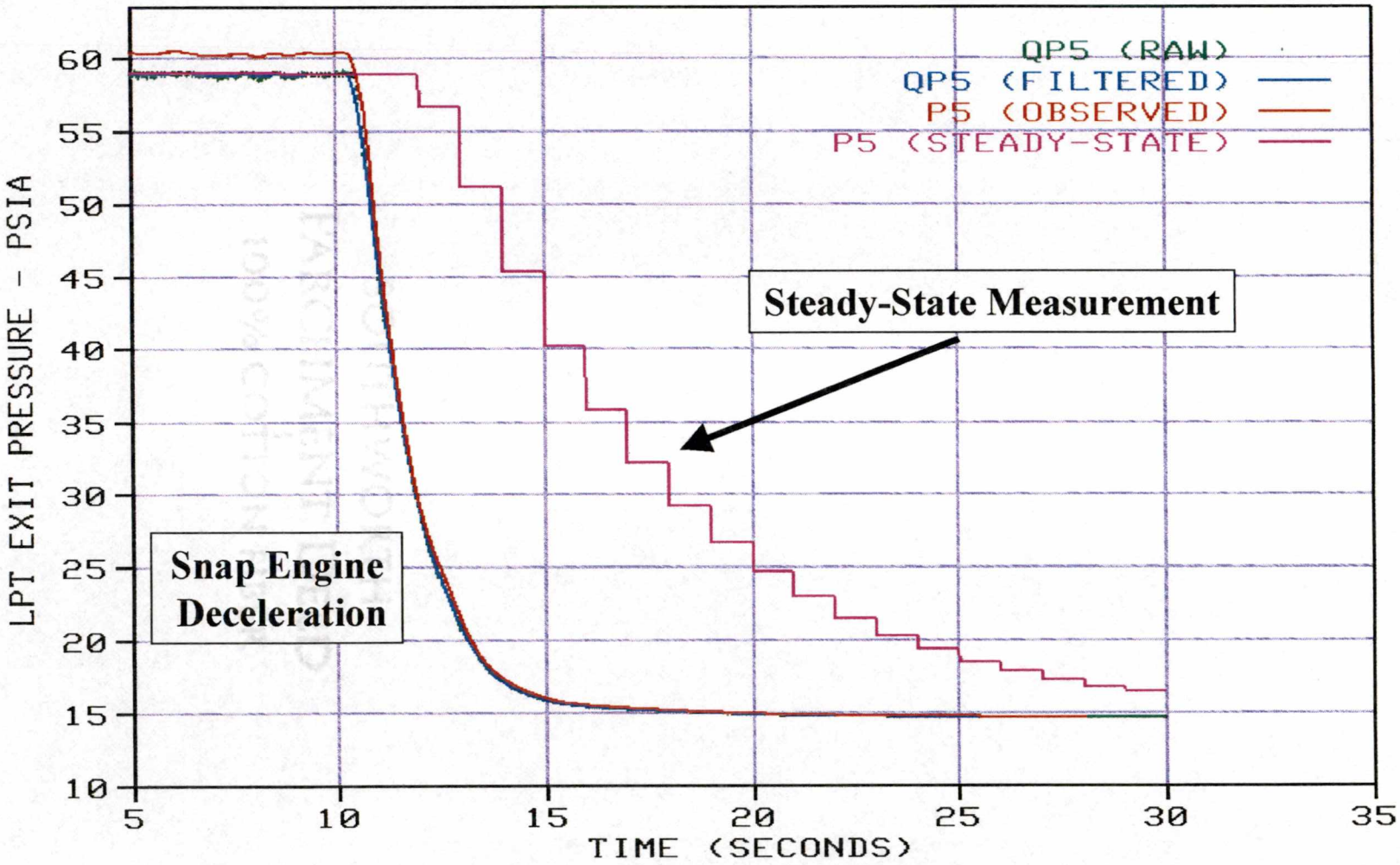


Fig. 7.20. Slowly Responding LPT Exit Pressure

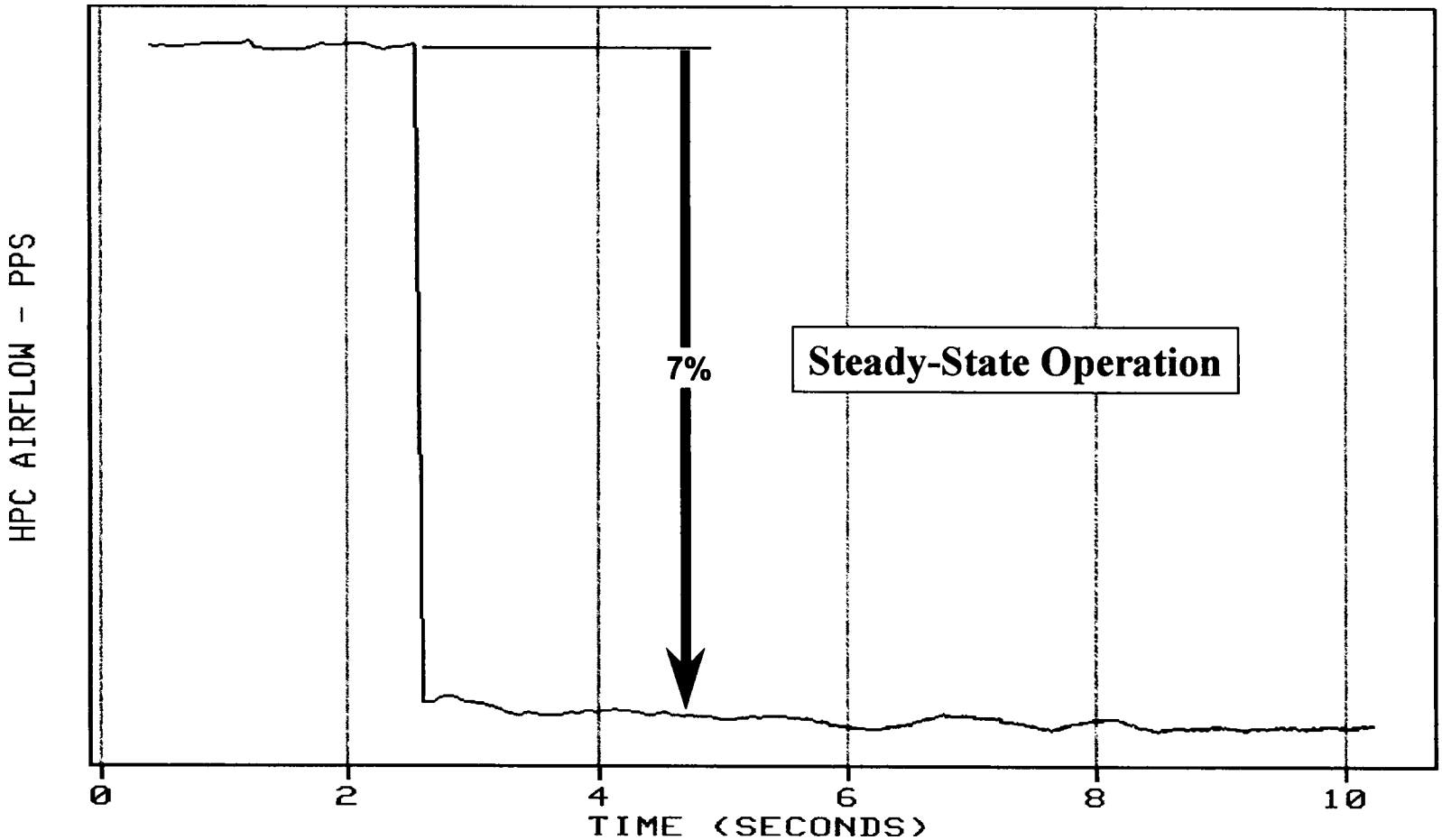


Fig. 7.21.1. Abrupt Change in Compressor Performance During Steady Engine Operation

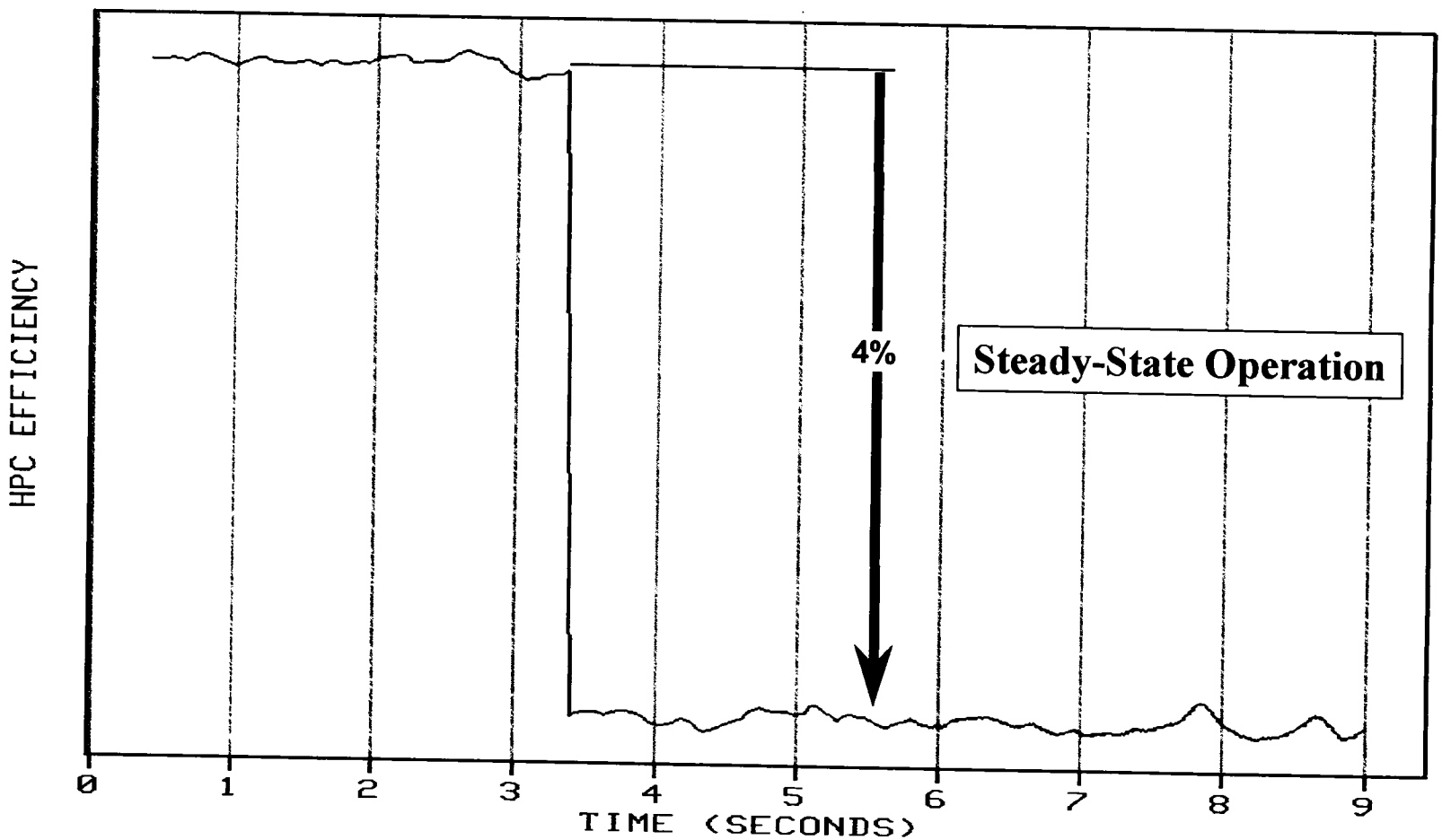


Fig. 7.21.2. Abrupt Change in Compressor Performance During Steady Engine Operation

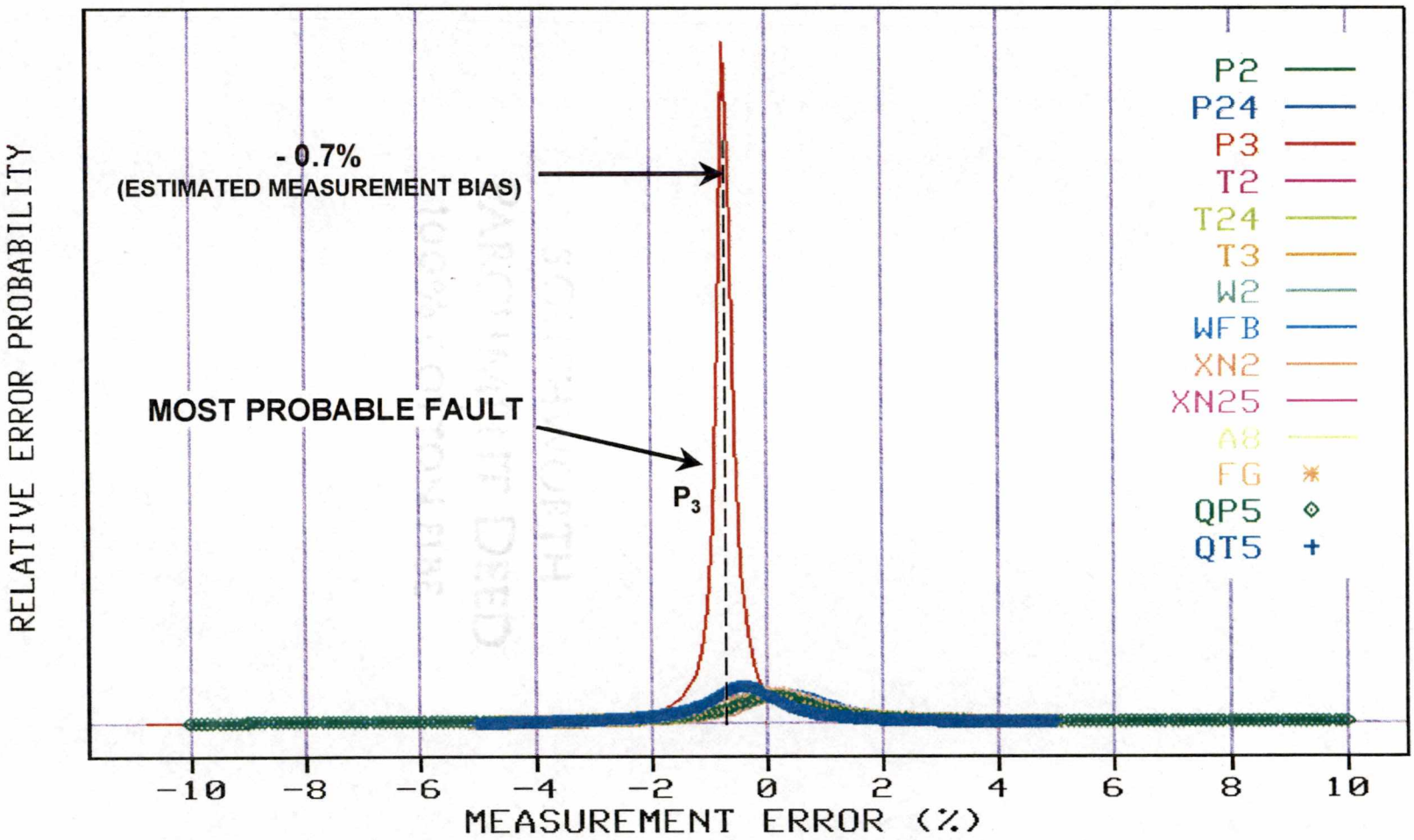


Fig. 7.22. Fault Identification and Estimate of Error

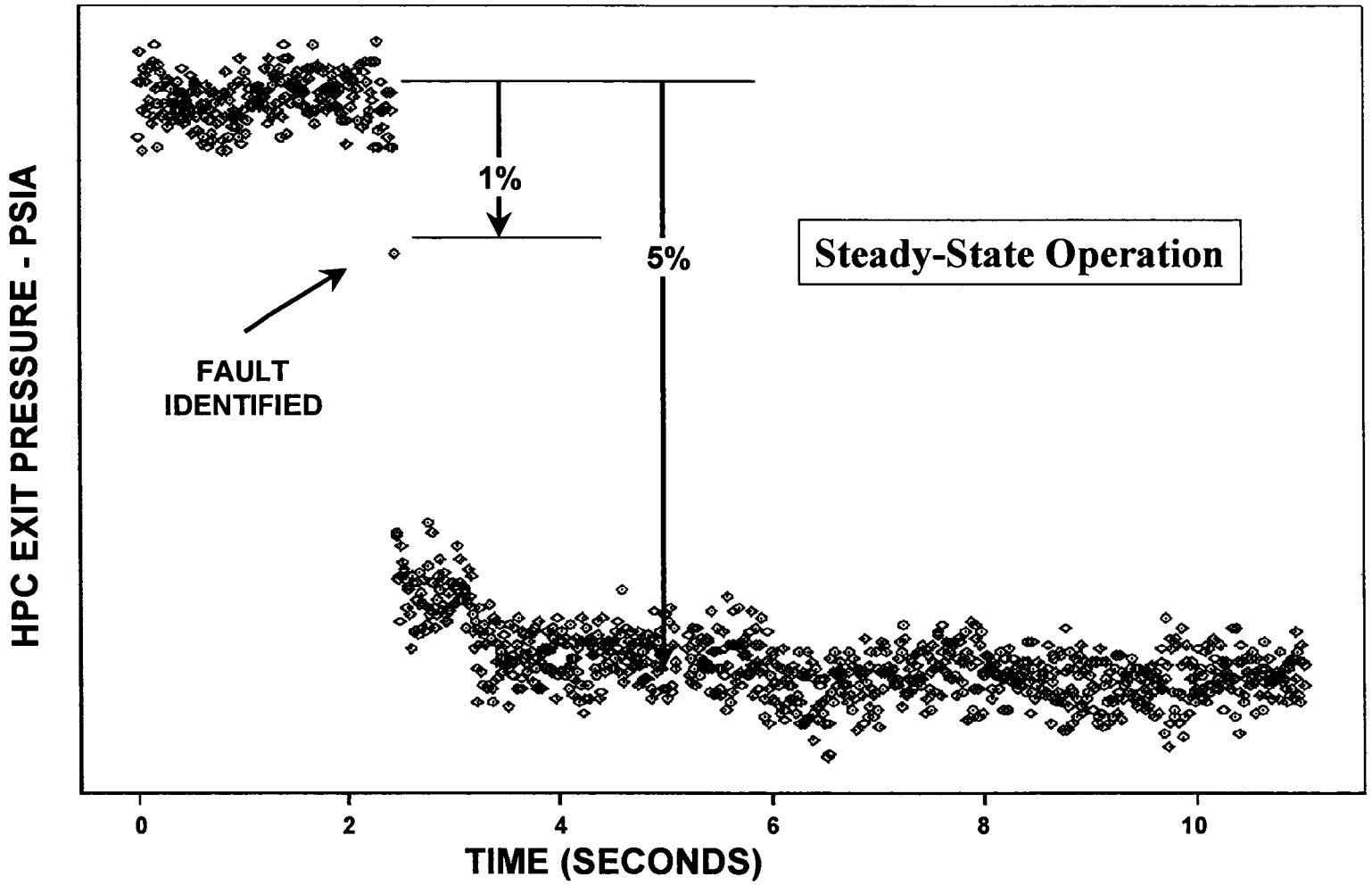


Fig. 7.23. Abrupt Change in Compressor Exit Pressure During Steady Engine Operation

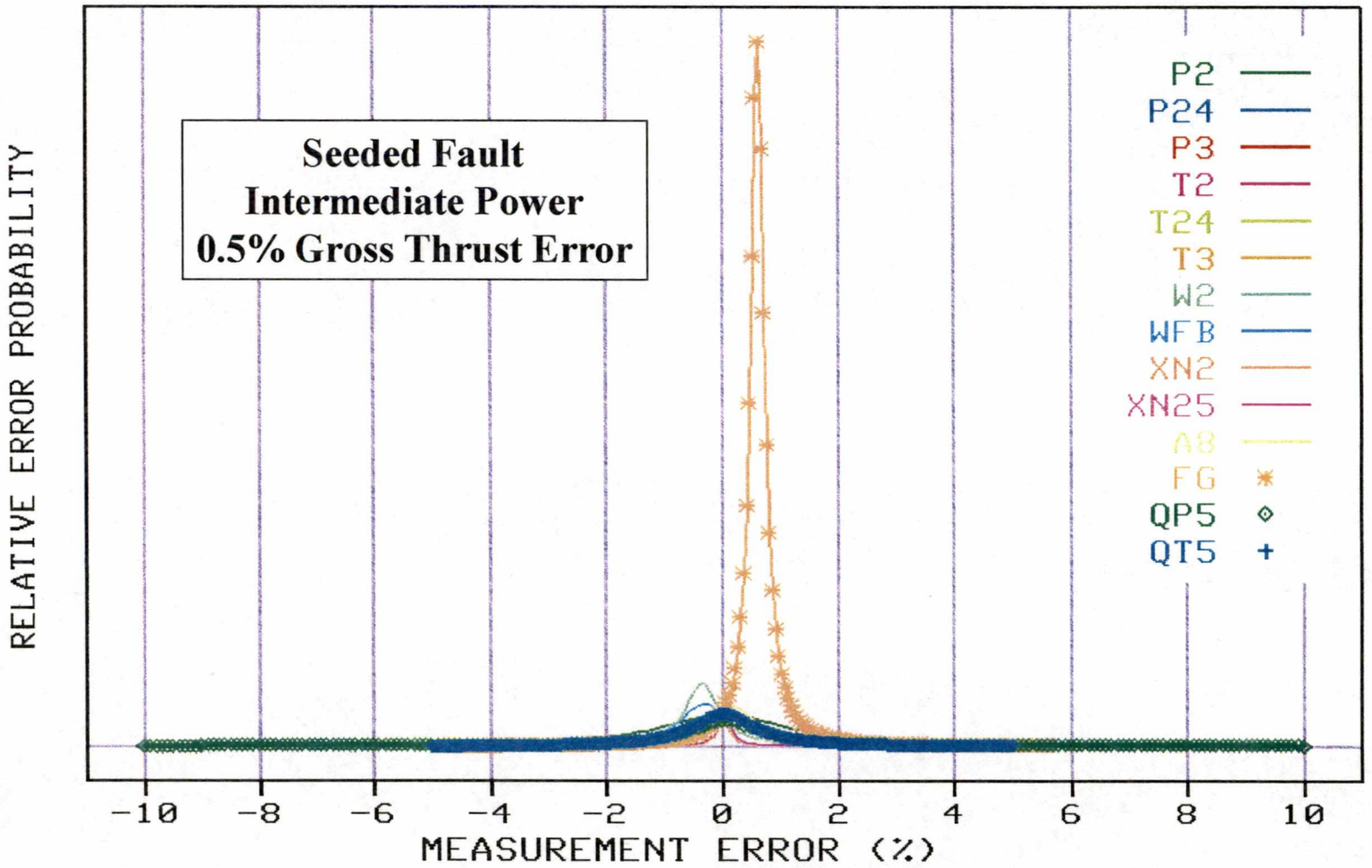


Fig. 7.24. Seeded Fault: 0.5% Gross Thrust Error

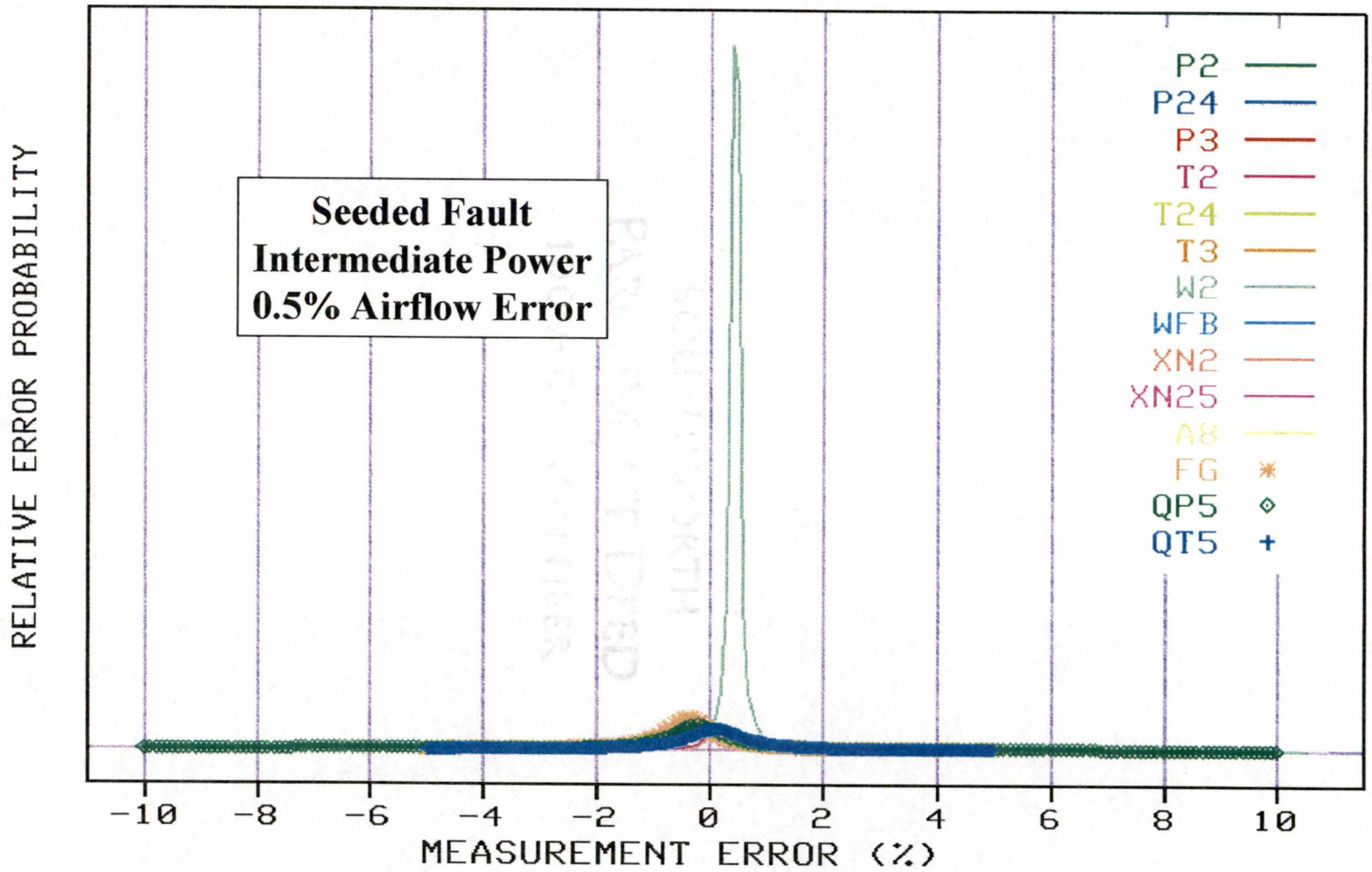


Fig. 7.25. Seeded Fault: 0.5% Airflow Error

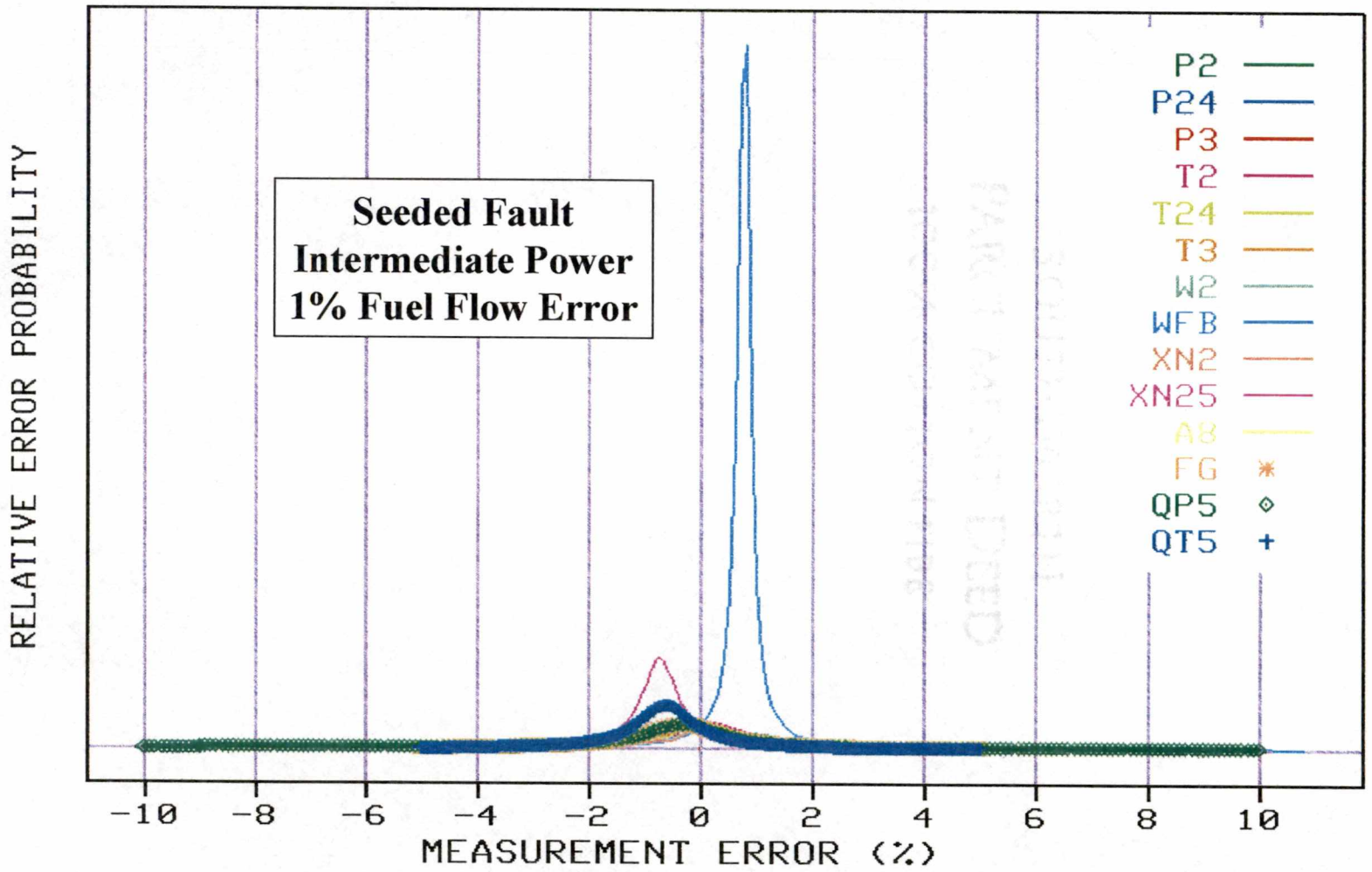


Fig. 7.26. Seeded Fault: 1.0% Fuel Flow Error

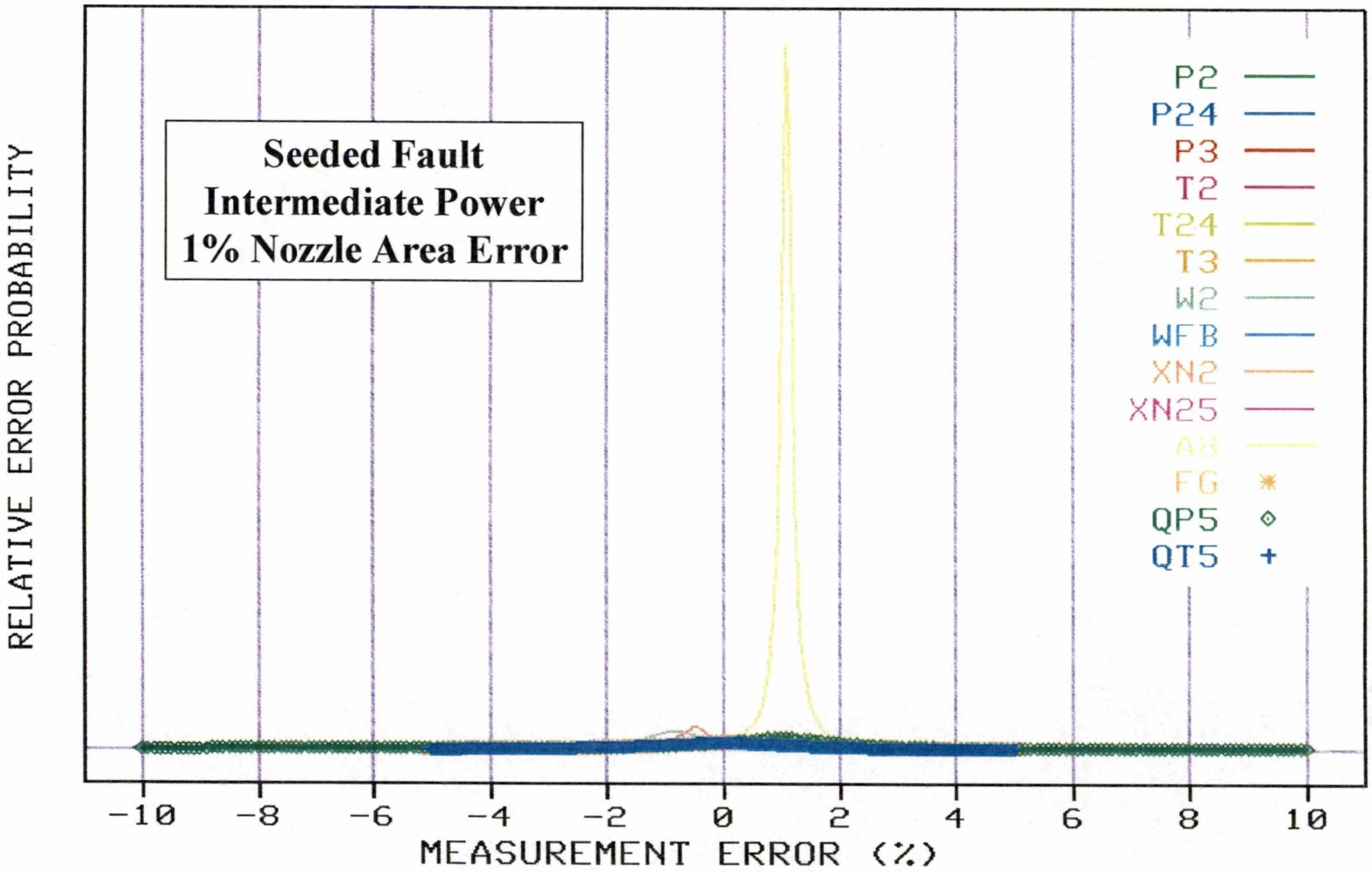


Fig. 7.27. Seeded Fault: 1.0% Nozzle Area Error

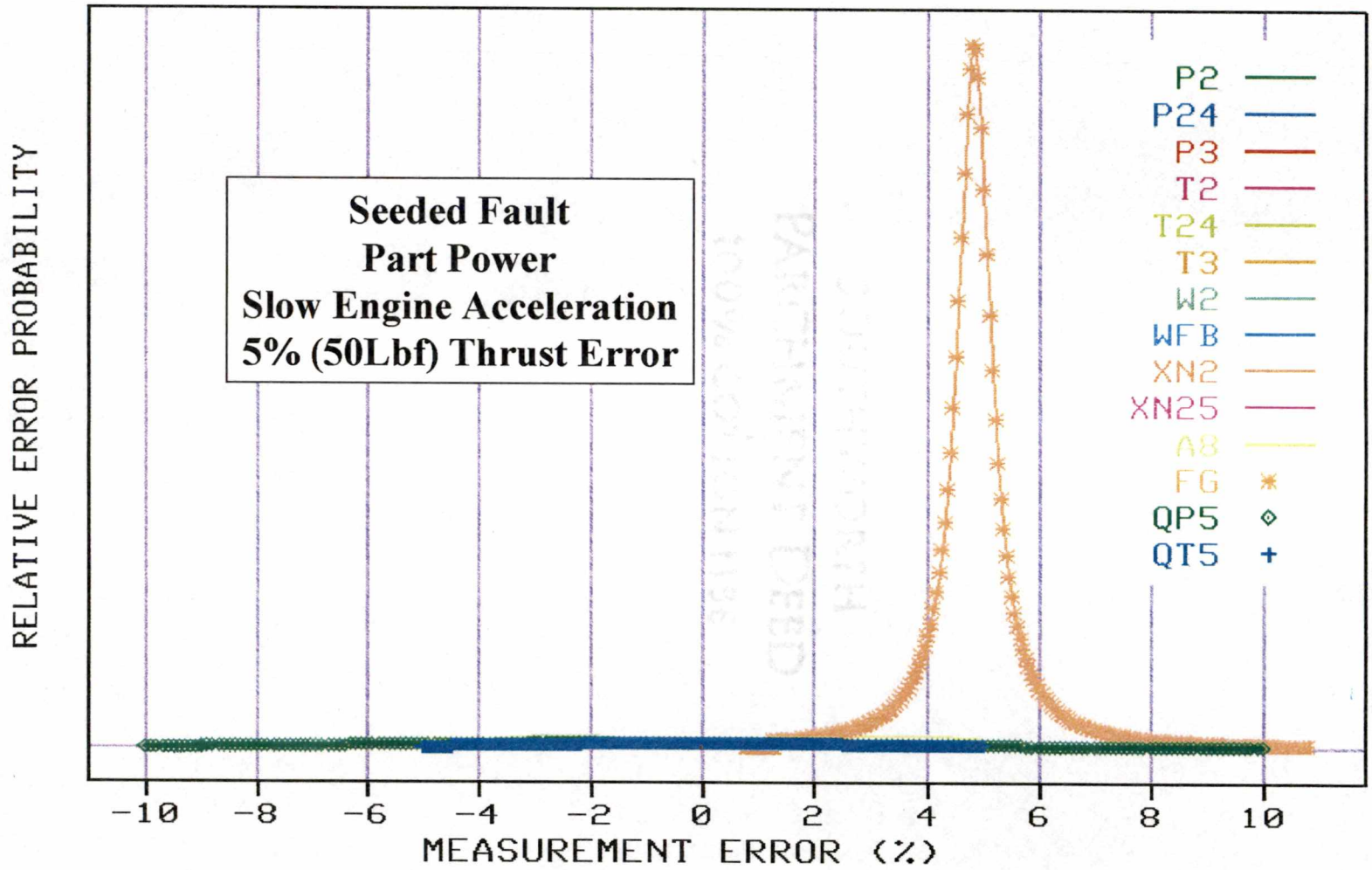


Fig. 7.28. Seeded Fault: 5% (50 Lbf) Thrust Error

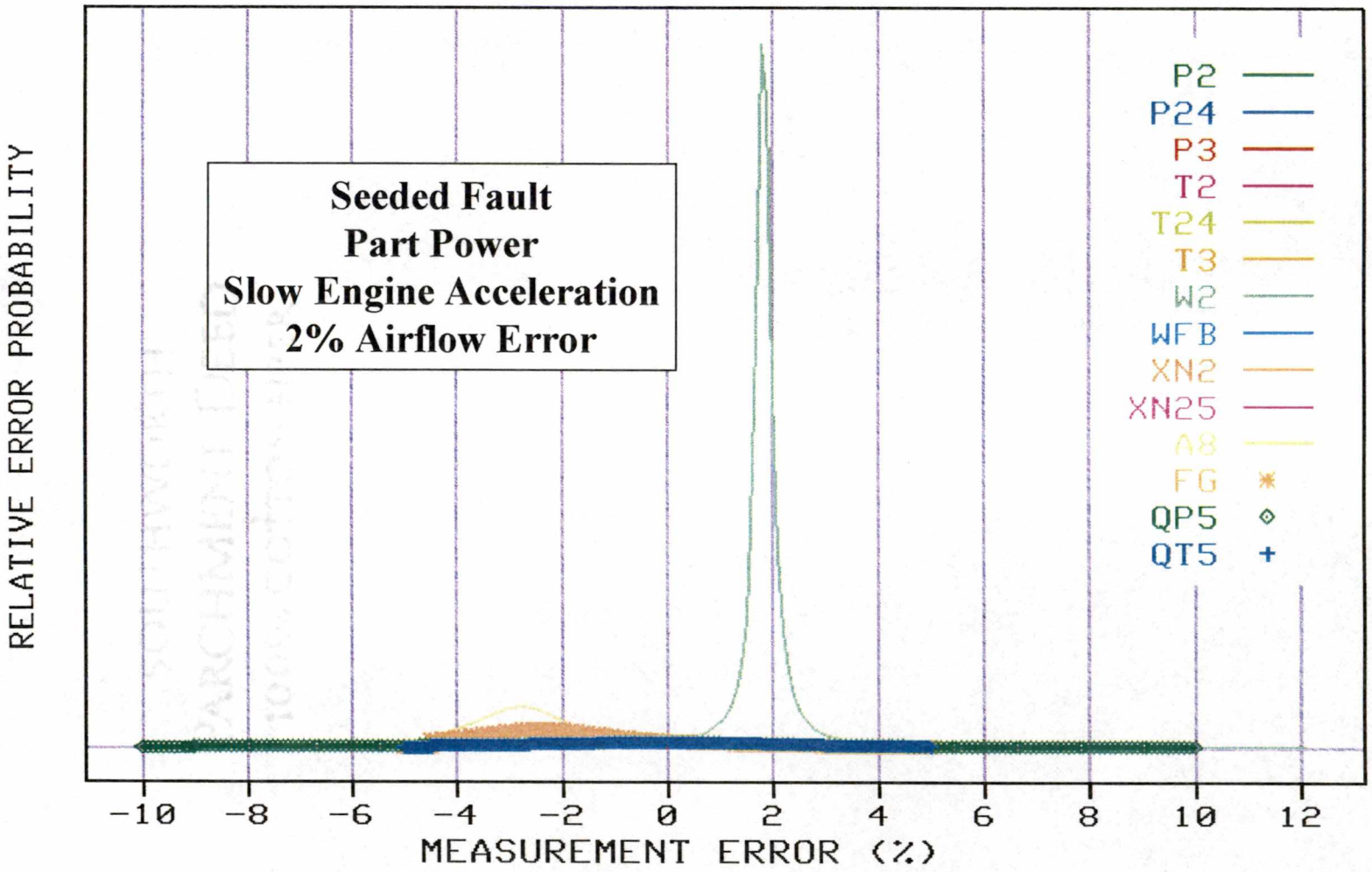


Fig. 7.29. Seeded Fault: 2% Airflow Error

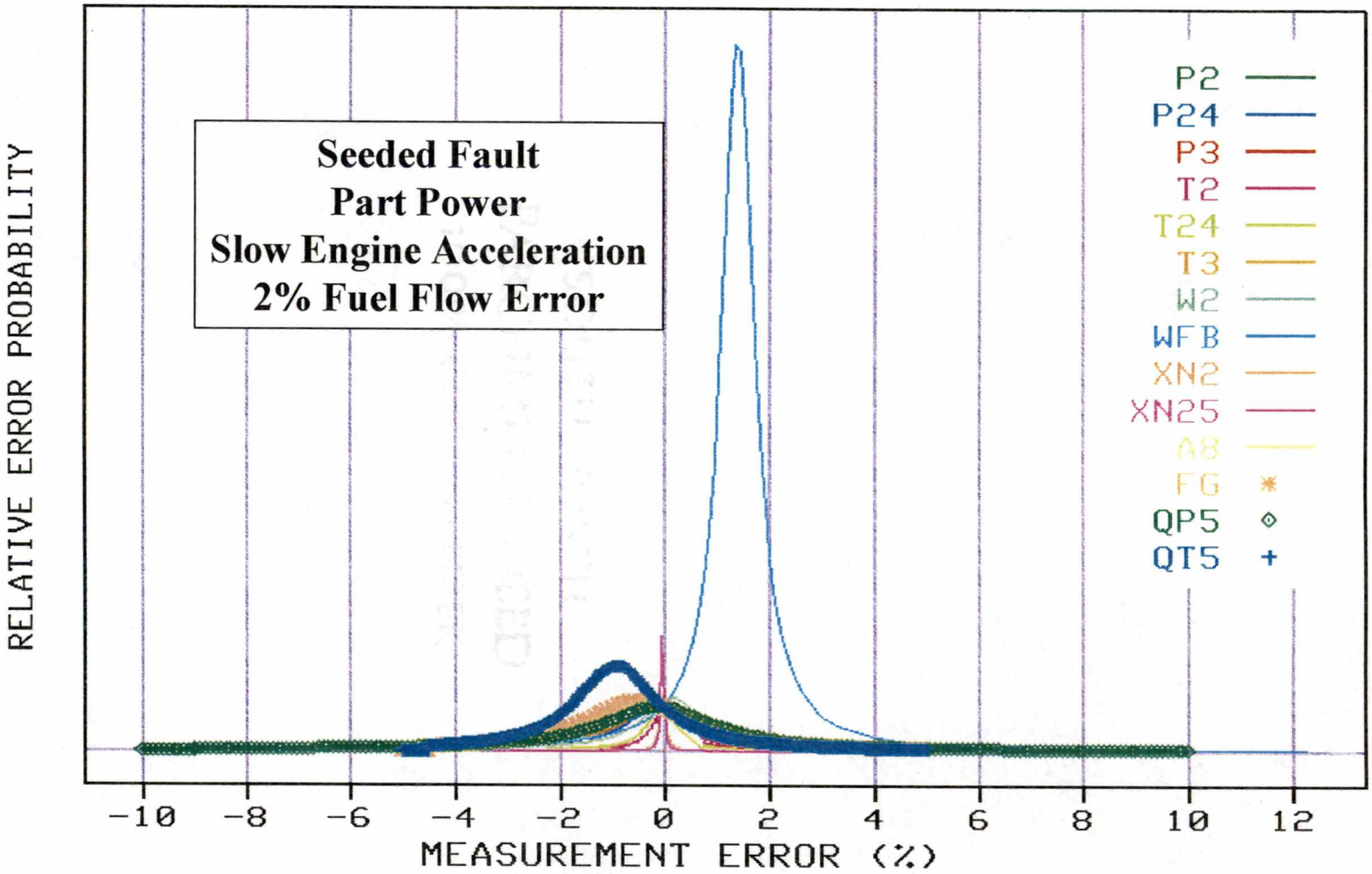


Fig. 7.30. Seeded Fault: 2% Fuel Flow Error

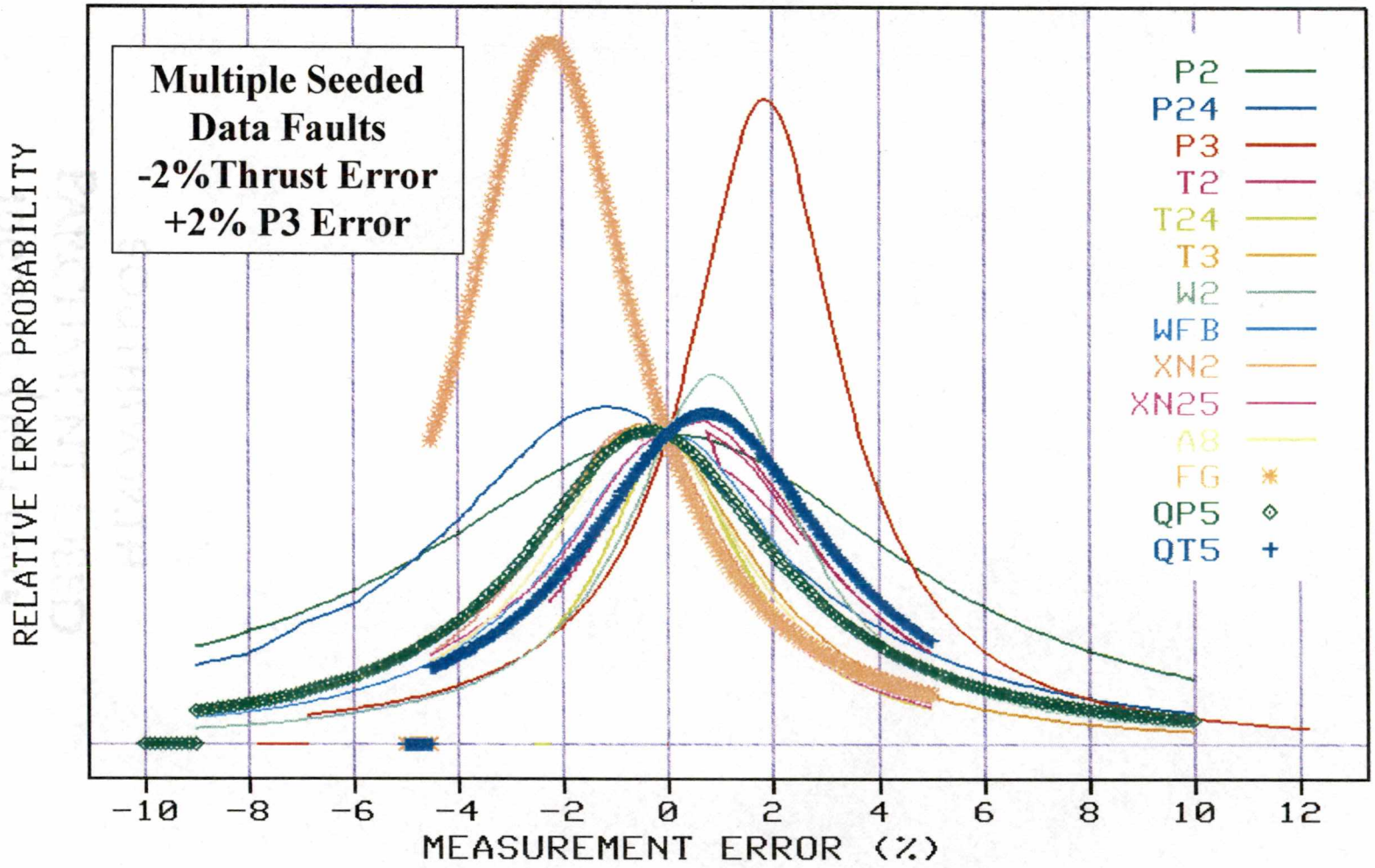


Fig. 7.31. Multiple Seeded Data Faults

APPENDIX B: REAL-TIME TURBINE ENGINE MODELS

SAE Aerospace Information Report (AIR) 4548 provides a review of real-time modeling methodologies for gas turbine engine performance. Overviews of the three real-time modeling methodologies for gas turbine engine performance provided in the AIR are repeated below.

Aerothermodynamic Models (from SAE AIR 4548)

The aerothermodynamic (aerothermo) real-time engine model can best be defined as a cycle model, which has been modified for transient simulations with constraints on consistency, size and computational speed. The extent of the modifications which are required will depend upon the cycle, the application and the processor. Models for multi-bypass engine cycles with compressible flow mixing planes are complex, which increases the difficulty of meeting the real-time model constraints. Ironically, the potential value of an aerothermo real-time model increases with the complexity of the cycle. The applications and processors which impose constraints upon the engine simulation may result in a series of real-time models for the same cycle. The application dictates how fast the model must execute and the processor determines how fast the model can execute. The size of the model is processor dependent.

The transient aerothermo model normally uses essentially the same representation as would a steady state cycle model. The time dependent aerodynamic equations of motion have been used for engine transients where the assumption of airflow continuity does not apply. The time step which is used with these equations must be consistent with the dimensions of the volume. Stall simulations use a time step on the order of 1 ms.

The transient model is the same as a steady state model running to fuel flow and variable geometry with two primary exceptions. Component torques acting upon the shaft are assumed to be unbalanced and together with moments of inertia define acceleration/deceleration (accel/decel) rates which are integrated in time to determine the transient characteristics of the engine. Also the transient model requires that appropriate heat soak representations be added in order to simulate the effect of heat transfer between the gas path and metal on the engine performance. Volume dynamics are included in transient models only for special cases, e.g., long ducts, large volumes, high frequency events such as stall and stall recovery.

Piece-Wise Linear Models (from SAE AIR 4548)

Piece-wise linear models, also referred to as state-space models, represent the majority of the engine's aerothermodynamic processes with a set of matrix equations that describe the variations about a set of base characteristics. The engine processes are

characterized as a whole rather than modeling individual components such as compressors, combustors and turbines. This type of model is generally derived from a non-linear aerothermodynamic model, although tuning of the model with engine data may also occur for certain applications. When engine control functions are included as part of an engine system model, these are simulated outside the aerothermodynamic piece-wise linear engine model.

The piece-wise linear modeling techniques extend classical linearized dynamic theory, which employs partial derivatives about a single operating point, by including multiple model points. Interpolation between these model points is used to define a base point which moves as engine power changes. Adjustments for flight condition and augmentation provide accurate steady state and transient characteristics of an engine operating between minimum and maximum power settings. The engine is modeled using three types of variables: states, inputs, and outputs. States represent energy storage parameters such as speed and metal temperature. Inputs are variables which perturb the systems, such as fuel flow and jet area. Outputs are additional parameters of interest other than the state and input variables, such as airflow and thrust. Dynamically, the engine is characterized by differential equations relating the time rate of change of the state variables to the state variables themselves and the input parameters. States are obtained transiently by calculating the derivatives and numerically integrating forward in time.

Transfer Function Models (from SAE AIR 4548)

Like piece-wise linear state-space models, the engine processes are not represented on a component level, but rather, are represented in terms of overall engine response. They are derived from non-linear aerothermo models and are normally driven by engine control functions simulated separately from the engine model.

In contrast, Transfer Function models represent the engine's aerothermo processes at steady-state with a set of base characteristics, and transiently with simple "leads" and "lags" applied to the base characteristics as they respond to control inputs. This architecture allows a reasonable amount of tunability to the steady-state performance levels and shapes, but is somewhat difficult to tune to specific transient events.

The gas generator model is segregated into 3 parts, the steady-state base lines, the subsystem offsets, and the transient characteristics.

The steady-state base lines are represented by non-linear functions which are the steady-state operating lines of the engine. These are usually curves of corrected output parameters vs. corrected fuel flow and flight Mach number. These curves would typically be generated using a detailed aerothermo model along a nominal flight path with nominal or fixed subsystem inputs. Additionally, the base lines can be derived from, or scaled, based on developmental or flight test results.

Subsystems (i.e., surge bleed, customer bleed, variable stators, etc.) are modeled via the application of offsets to the steady-state base lines. The offsets are functions of appropriate corrected input or output parameters (e.g., fuel flow or rotor speed) and flight Mach number. The offsets are obtained by observing the effect of each subsystem on the output parameters. The amount of offset used for a subsystem is dictated by the deviation of the control model command from the nominal or fixed subsystem input assumed in the steady-state base lines.

The transient characteristics of the engine are represented by transfer functions. A transfer function filters the change to an input variable and transfers it to an output variable to create its response. The transfer function consists of a ratio of two polynomials in the Laplace operator "s" which can be expressed as a ratio of factored roots. The denominator of a transfer function is called the "characteristic function" of the physical system and determines all the possible kinds of natural motion the system can have. The characteristic function is often referred to as the "lag" of the system. The numerator of the transfer function determines which part of the characteristic function will be dominant in the engine parameter response. The transfer function is implemented in the form of lead/lag pairs. The roots of the polynomials determine the lead and lag time constants. While a unique set of time constants can be generated for each engine parameter, in practice parameters will often share the same coefficients if their dynamic response characteristics are similar.

The coefficients (time constants) of the transfer functions vary as a function of a match parameter (e.g., corrected rotor speed). The coefficients are obtained from the partial derivative matrices of a state space model which are algebraically manipulated to generate an equivalent transfer function representation. The equivalent transfer function polynomials are determined by finding the set of coefficients which best approximate the frequency response of a parameter. In the state space representation the relationship between the states, inputs, and outputs is explicitly represented by the partial derivatives. In the transfer function representation, however, a parameter's response to all state and input perturbations is lumped into a lead/lag transfer function with a single input (usually fuel flow). The steady-state value of a parameter is scheduled as a function of fuel flow then passed through a set of lead/lag functions to determine the parameter's transient response.

APPENDIX C: ADVANCED TURBINE ENGINE SIMULATION TECHNIQUE (ATEST)

The information in this Appendix was extracted from a technical paper entitled "Advanced Turbine Engine Simulation Technique Development and Applications to Testing by M.A. Chappell and E.G. Blevins (Chappell, 1986).

The Advanced Turbine Engine Simulation Technique (ATEST) is a digital computer simulation system applicable to arbitrary turbine engine configurations. It is built using a modular concept and requires a conservative amount of computer time and execution time as compared to other engine simulation methods. ATEST is a one-dimensional component-matching simulation that is intended for aircraft gas turbine engine cycles, (e.g., single-spool turbojet, high-bypass turbofan, variable cycle engine, etc.) but it could also be used to simulate other physical systems, taking advantage of the modular structure. The current version of ATEST is capable of simulating steady-state and transient engine operations including idle-to-intermediate operations, windmilling, and starting.

General ATEST Development

It was recognized from the beginning of the ATEST development that certain aspects of simulating specific engine cycles could easily be generalized; for example, all of the new generation of tactical aircraft engines are dual-spool, mixed-flow, afterburning turbofan engines. It was also recognized that the characteristics which distinguish each engine from the others usually cannot be generalized. The distinguishing characteristics are primarily the unique component performance maps associated with each engine. Other characteristics, however, include component interactions (pressure losses, heat transfer, etc.), parasitic power management (horsepower extraction, compressor bleed air extraction, bearing friction losses, etc.), and non-standard component performance (bypass duct pressure loss). These other characteristics are typically documented by the manufacturer as analytical expressions in terms of engine cycle operating parameters. Since the expressions may be complex, it is desirable to use the precise expression. Subroutine ENGMOD primarily establishes the cycle flow path, thus determining the general cycle being simulated (dual-spool turbofan, turboshaft, variable cycle, etc.). In addition, the user-provided subroutine ENGMOD permits the user to include any engine specific characteristics, a feature that distinguishes ATEST from other generalized simulations.

BLOCK DATA contains all the component performance maps that are required for an ATEST cycle simulation along with appropriate values of program switches as

selected by the user. Output instruction information is also contained in BLOCK DATA. Definition of component performance maps is provided by BLOCK DATA (rather than reading the maps as input) in ATEST for two reasons. First, the size of the arrays containing component performance data can be established for each individual cycle as the component data are entered. The data arrays are sized to fit a specific application, thus tailoring the computer memory requirement to each application. (This is in contrast to current generalized programs that allocate a large fixed amount of memory for all applications.) Secondly, the computational overhead of loading the component data into the memory is encountered only as the program is compiled. This conserves execution time since the program can be executed repetitively from a single compilation of the program.

Component Subroutine Capabilities

A single subroutine was established for each type of component (compressors, turbines, ducts, etc.) as a result of the modular architecture. In order to preserve the generalized character of ATEST, each component subroutine must be generalized (capable of representing many variations of the same type component). To accomplish this subroutine generalization, a list of required capabilities was established for each component type. Each list attempted to encompass all variations for a particular component including physical size variations (e.g., large diameter fans or small diameter compressors), axial or centrifugal design, and cooling-air injection or extraction.

Consider the compressor subroutine as an example. The compressor subroutine must be capable of:

1. Determining the basic performance of any compressor from standard performance maps and adiabatic compression equations considering real-gas effects on thermodynamic properties.

2. Providing a generalized bleed capability.

3. Providing for engine specific operational effects (Reynolds effects, off-schedule variable geometry).

Item 1 provides a capability for a range of compressor configurations. This capability is particularly advantageous in dealing with the different ways of representing the performance of large diameter fans. The performance of these fans is typically segmented into hub and tip region performance, although an overall average performance map is also common. The segmented performance representations appear in a variety of forms. Both hub and tip performance may be represented with distinct maps (as if there were two compressors), or tip performance may be presented as a functional relationship of both hub performance and bypass ratio. The compressor subroutine within the ATEST

architecture is able to determine compressor performance for each of these large diameter configurations in addition to the more familiar smaller diameter configurations.

Item 2 provides a capability that manages multiple (up to a total of 10) interstage bleeds from any one or all of the compressor components. This is an important capability in modeling complex turbine cooling flows of modern turbine engine cycles. These bleeds are managed individually, and thermodynamic properties (temperature and pressure) are calculated for each. These bleeds can be routed anywhere within or outside of the cycle. Other generalized simulations (NEPCOMP, NNEP) provide only one interstage bleed extraction point per compressor.

Item 3 provides the capability of adjusting the compressor performance maps for the effects of off-schedule geometry, low Reynolds number and any other phenomena that might influence the nominal compressor performance. These operational effects are specific to each compressor and cannot be generalized. Therefore, these effects are calculated in a user-provided subroutine (CORECT) that is called from the compressor subroutine. Use of CORECT preserves the general applicability of the compressor subroutine in addition to permitting corrections to nominal compressor performance. Similar adjustments can be made to the turbine performance using the same technique. Current generalized simulations (NEPCOMP, NNEP, TURBOTRANS) do not provide this capability.

It should be noted that two turbine subroutines are provided as a result of following this procedure. Two subroutines are provided because of the lack of an industry standard format for turbine performance maps. It is not always possible to accurately convert one form of characteristic representation to the other; therefore, each turbine subroutine uses a different form.

Program Controls

An indication must be made in the ATEST, as in other generalized turbine engine models, of the engine parameters that are to be used as independent variables, i.e., variables that are manipulated by the equation-solving technique in an attempt to determine a solution. The equation-solving technique updates the values of each of these independent variables until a solution in determined. These are typically parameters such as shaft speeds, bypass ratios, and compressor ratios. An indication must also be made of the variables that are to be maintained at a prescribed value (dependent variables). These are typically parameters such as fuel flow and nozzle throat area. These variables are often referred to as "run-to" variables, the variables that the user specifies to prescribe an operating point. Other dependent variables that ensure mass, energy, and momentum conservation are automatically generated by ATEST. Together these parameters control the operation of the cycle simulation, and proper specification of these parameters is a critical factor in obtaining a solution in ATEST and all of the referenced programs.

APPENDIX D: MEASUREMENT UNCERTAINTY TABLES

REPRESENTATIVE STEADY-STATE TURBINE ENGINE TEST MEASUREMENT UNCERTAINTIES

PARAMETER	MEASUREMENT RANGE	BIAS B	PRECISION S	UNCERTAINTY[1] $U = \sqrt{B^2 + (2S)^2}$
Pressure System				
Aerodynamic				
In-Place				
Calibrated				
Absolute	0.5 to 5 psia			
(Vacuum Ref.)	1 to 10 psia	±0.007 psi	±0.0008 psi	±0.007 psi
	1 to 15 psia	±0.0075 psi	±0.0012 psi	±0.008 psi
		±0.008 psi	±0.0018 psi	±0.009 psi
Differential	1 to 29.2 psia	±0.01 psi	±0.0036 psi	±0.012 psi
(Ambient Ref.)	1.5 to 39 psia	±0.016 psi	±0.004 psi	±0.018 psi
	2.5 to 64 psia	±0.02 psi	±0.006 psi	±0.023 psi
	5 to 114 psia	±0.034 psi	±0.012 psi	±0.042 psi
	14.2 to 214 psia	±0.067 psi	±0.024 psi	±0.082 psi
	14.2 to 514 psia	±0.17 psi	±0.10 psi	±0.26 psi
	14.2 to 614 psia	±0.23 psi	±0.12 psi	±0.33 psi
Differential	±0.5 to 5 psi	±0.008 psi	±0.0008 psi	±0.008 psi
(Floating Ref.)	±1 to 10 psi	±0.013 psi	±0.0012 psi	±0.013 psi
	±1.5 to 15 psi	±0.019 psi	±0.0018 psi	±0.019 psi
Aerodynamic & Liquid Resistance Shunt Calibrated	1.5 to 3 psia	±0.021 psi	±0.0045 psi	±0.023 psi
	3 to 500 psia	±0.7% Rd	±0.15% Rd	±0.76 Rd
Temperature System				
(Thermocouple)	-300 to 700°F	±(0.38% Rd +0.8°F)	±0.6°F	±(0.38% Rd +1.4°F)
Cu/Con	-32 to 2300°F	±(0.38% Rd +0.8°F)	±0.6°F	±(0.38% Rd +1.4°F)
CA				
Speed System (F/A)	500 to 80K RPM	±0.1% Rd	±0.1% Rd	±0.14 % Rd
(Counter)		±0.001% Rd	±0.01% Rd	±0.02% Rd
Flow System	200 to 100K PPH	±0.4% Rd	±0.3% Rd	±0.7% Rd
Force System	5K to 50K lbf	±0.4% Rd	±0.15% Rd	±0.5% Rd
Geometry System	As Required	±0.4% Rd	±0.3% Rd	±0.7% Rd
Dimensional Xducer		±0.002 in	±0.002 in	±0.004 in

NOTE: The uncertainty values in percent reading (Rd) are applicable over 20% to 100% of the sensor range.

REPRESENTATIVE TURBINE ENGINE TEST MEASUREMENT UNCERTAINTIES
TRANSIENT

PARAMETER	MEASUREMENT RANGE	BIAS B	PRECISION S	UNCERTAINTY[1,2] $U = \sqrt{B^2 + (2S)^2}$	FREQUENCY RESPONSE
Pressure System					
Resistance Shunt	1.5 to 3.0 psia	±0.027 psi	±0.009 psi	±0.03 psi	0-3 Hz
Calibrated	3.0 to 500 psia	±0.9%	±0.3%	±1.1%	
Temperature System					
(Thermocouple)	-300 to 700°F	±(0.75% +1.6°F)	±1.2°F	±(0.75% Rd +3°F)	0-0.1 Hz
Cu/Con	32 to 2300°F	±(0.75% +1.6°F)	±1.2°F	±(0.75% Rd +3°F)	0-0.1 Hz
CA					
Speed System (F/A)	500 to 80K RPM	±0.2% Rd	±0.2% Rd	±0.45 % Rd	0-3 Hz
Flow System	200 to 100K PPH	±0.8% Rd	±0.6% Rd	±1.4% Rd	0-3 Hz
Force System	5K to 50K lb_f	±0.4% Rd	±0.3% Rd	±0.7% Rd	0-3 Hz
Geometry System					
Dimensional	As Required	±0.8% Rd	±0.6% Rd	±1.4% Rd	Response of
Xducer		±0.004 in	±0.004 in	±0.01 in	mechanical sys
Dimensional					
Tools					

NOTE:
1. All percentages (%) are percent of reading.
2. The uncertainty values in percent are applicable over 20 to 100% of the sensor full-scale range.

APPENDIX E: THERMOCOUPLE TIME RESPONSE

Velocity, conduction, and radiation effects must be evaluated to determine their contribution to the transient response of the thermocouple. When considering velocity effects, the heat transfer to the thermocouple is primarily dependent on the mass flow of the gas. The time response of a thermocouple is given by (Moffat, 1957).

$$-m_{TC}c_p\frac{dT_{TC}}{dt} = h_c A_{TC}\left(T - T_{TC}\right)$$

where

h_c is the average convective heat transfer coefficient at the junction

A_{TC} is the surface area of the thermocouple junction

T is the gas total temperature

T_{TC} is the temperature indicated by the thermocouple

t is time

m_{TC} is the mass of the thermocouple junction

c_p is the specific heat of the thermocouple junction

and the time constant τ is given by

$$\tau = m\frac{c_p}{h_c A}$$

The time constant is dependent on the average convective heat transfer coefficient. Therefore, the time constant is dependent on the mass flow. When the time constant is know at a reference temperature and pressure, the time constant under a new set of conditions is given by (Glawe et al, 1978):

$$\tau = \frac{\tau_o}{\sqrt{MaPs / Pso}} \frac{T_{TC}}{T_o}$$

where

τ_o is the reference time constant

T_o is the reference temperature

T_{TC} is the temperature indicated by the thermocouple

Ma is the gas stream Mach number

Ps is the gas stream static pressure

and Pso is the reference static pressure.

As would be expected, the orientation of the thermocouple to the flow direction and conduction effects also affect response time.

APPENDIX F: PROPAGATION OF ELEMENTAL MEASUREMENT UNCERTAINTIES

The information in this Appendix was extracted from AEDC's STANDARD DATA UNCERTAINTY CALCULATION PROGRAM USERS' MANUAL (AMU02788).

The proper method for propagating uncertainty values is to propagate the bias limits and precision estimates separately, then apply the uncertainty formula to the combined bias and precision terms. This is true whether measurement uncertainty is being determined from elemental errors or calculated uncertainty is being determined from measurement uncertainty.

Influence coefficients may be defined as the relative effect on a dependent calculated parameter of a variation in an independent measured parameter using an equation where the dependent parameter is calculated using the independent parameter. In general, if Q is some calculated parameter, Q = f(X,Y,Z) where X, Y, and Z are measured parameters, the influence coefficients may be found by the differential of Q:

$$\frac{dQ}{Q} = \frac{X}{Q}\frac{\partial Q}{\partial X}\frac{dX}{X} + \frac{Y}{Q}\frac{\partial Q}{\partial Y}\frac{dY}{Y} + \frac{Z}{Q}\frac{\partial Q}{\partial Z}\frac{dZ}{Z}$$

A variation in the measurement of X, dX/X, will cause a variation on Q, dQ/Q, of the magnitude

$$\left(\frac{X}{Q}\frac{\partial Q}{\partial x}\right).$$

The terms

$$\left(\frac{X}{Q}\frac{\partial Q}{\partial X}\right) \quad \left(\frac{Y}{Q}\frac{\partial Q}{\partial Y}\right) \quad \left(\frac{Z}{Q}\frac{\partial Q}{\partial Z}\right)$$

(Q ∂x) , (Q ∂y), and (Q ∂z) are the influence coefficients of X, Y and Z on Q. The influence coefficients may also be approximated by using a perturbation method on a digital computer. This is done by inducing a change in X and then computing the resultant change in Q while holding Y and Z constant. The influence coefficient of X on Q is then

$(\Delta Q/Q) / (\Delta X/X)$.

The process is then repeated for Y and Z to determine their influence coefficients.

APPENDIX G: CALCULATION OF AIR FLOW USING VENTURIS AND NOZZLES

Description	Calculation
Subsonic Venturi or Nozzle Flow	$W = \frac{\pi d^2}{4} \frac{C_w}{\sqrt{1-\beta^4}} Ya\, Fa \sqrt{2\rho_1 (P_1 - P_2)}$
Sonic Venturi Flow	$W = \frac{\pi d^2}{4} C_w \phi_i \left(\frac{\phi}{\phi_i}\right)\left(\frac{P_1}{\sqrt{T_1}}\right)$
Discharge Coefficient	C_w for nozzle = *f*(Reynolds No., β, pressure tap configuration) C_w for subsonic venturi =*f*(entrance cone config.) C_w for sonic venturi
Adiabatic Expansion Factor	$Ya = \left[P_r^{\frac{2}{\gamma}} \left(\frac{\gamma}{\gamma-1}\right)\left(\frac{1-P_r^{\frac{\gamma-1}{\gamma}}}{1-P_r}\right)\left(\frac{1-\beta^4}{1-\beta^4 P_r^{\frac{2}{\gamma}}}\right)\right]^{\frac{1}{2}}$
Thermal Expansion Factor	Fa = *f*(metal type, metal temperature)
Density (moist gas)	$\rho = 2.6991(1+SH)\frac{P-P_v}{TZ}$
Reynolds Number	$R_d = \frac{4W}{\pi d \mu}$
Absolute Viscosity	μ = *f*(gas temperature, gas type)
Compressibility Factor	Z = *f*(critical temperature and pressure)
Sonic-flow-function (Ideal gas)	$\phi_i \sqrt{\frac{\gamma(m_w)}{R}\left(\frac{1+\gamma}{2}\right)^{-\left(\frac{\gamma+1}{\gamma-1}\right)}}$
Ratio Real to Ideal Gas Sonic-flow-function	$\frac{\phi}{\phi_i}$ = *f*(inlet stagnation press & temp)
Specific Humidity	SH = 0.622 Pv/(P-Pv)
Vapour Pressure	P_v = *f*(wet and dry bulb temperatures)

APPENDIX H: FAN PERFORMANCE CHARACTERIZATION

Fan Efficiencies

The isentropic efficiency for a compression process is determined from the following relationship:

$$ETA = \frac{HFIS - HI}{HF - HI}$$

where HI denotes the total enthalpy based on total temperature and pressure at the initial condition, HF denotes the total enthalpy at the final condition, and HFIS represents the total enthalpy at the final condition resulting from an isentropic compression.

Fan efficiencies (overall, hub, and tip) are calculated as follows:

$$E2A = \frac{HF21AIS - H2A}{H21A - H2A}$$

$$E21 = \frac{HF21IS - H2}{H21 - H2}$$

$$E12 = \frac{HF13IS - H12}{H13 - H12}$$

where,

H2 and H12 are, respectively, the fan inlet hub and tip section total enthalpies; H21 and H13 are, respectively, the fan exit hub and tip section total enthalpies; and H2A and H21A are the average fan inlet and fan exit total enthalpies.

Fan Inlet Referred (Corrected) Mass Flow

Fan inlet referred (corrected) mass flow (W2AR) is calculated from the following relation:

$$W2AR = \frac{W2A\sqrt{TH2A}}{DEL2A}$$

where W2A is the total fan inlet mass flow, and TH2A and DEL2A are defined by

$$TH2A = \frac{T2A}{518.67}$$

and

$$DEL2A = \frac{P2A}{14.696}$$

Fan Referred (Corrected) Rotor Speed

Fan referred (corrected) rotor speed (XN2AR) is calculated as follows:

$$XN2AR = \frac{XN2}{\sqrt{TH2A}}$$

Fan Percent Corrected Rotor Speed

Fan percent corrected rotor speed (PCN2R) is calculated as follows:

$$PCN2R = \frac{XN2R}{XN2R_{Design}} \times 100$$

where $XN2R_{Design}$ is the design value of the fan corrected rotor speed.

Reynolds Number Index

Reynolds number index for air is calculated as follows, where the viscosity term utilizes Sutherland's relation.

$$RNI = \left[\frac{T + 198.72}{518.67 + 198.72}\right]\frac{\delta}{\theta^2}$$

$$\delta = \frac{P}{14.696}$$

$$\theta = \frac{T}{518.67}$$

APPENDIX I: HP COMPRESSOR PERFORMANCE CHARACTERIZATION

HP Compressor Efficiency

HP Compressor isentropic efficiency is determined from the compressor inlet and exit total enthalpies using the methodology previously presented in Appendix H to determine "FAN EFFICIENCIES."

HP Compressor Inlet Referred (Corrected) Mass Flow

HP Compressor inlet referred (corrected) mass flow (W24R) is calculated from the following relation:

$$W24R = \frac{W24\sqrt{TH24}}{DEL24}$$

where W24 is the compressor inlet mass flow, and TH24 and DEL24 are defined by

$$TH24 = \frac{T24}{518.67}$$

and

$$DEL24 = \frac{P24}{14.696}$$

HP Compressor Referred (Corrected) Rotor Speed

HP Compressor referred (corrected) rotor speed (XN24R) is calculated as follows:

$$XN24R = \frac{XN24}{\sqrt{TH24}}$$

HP Compressor Percent Corrected Rotor Speed

HP Compressor percent corrected rotor speed (PCN24R) is calculated as follows:

$$PCN24R = \frac{XN24R}{XN24R_{Design}} \times 100$$

where $XN24R_{Design}$ is the design value of the HP compressor corrected rotor speed.

Reynolds Number Index

Reynolds number index for air is calculated using the methodology presented in Appendix H, where the viscosity term utilizes Sutherland's relation.

Compressor Map Scaling

Multiple steps are required to scale the compressor. First, the map point must be selected. Then, The map is adjusted to operating conditions to account for variations in the Reynolds number index and rotor-to-case clearances. Additional adjustments may also be required for off-schedule guide vane operation. Then, the map is scaled to match the data. An OP-LINE Function is used to select the compressor map point. The OP-LINE relationship is derived from the test data being diagnosed. The OP-LINE function compensates for variations in the compressor operating line resulting from variations in the HP Turbine inlet nozzle area. The OP-LINE relationship is derived as a function of HP compressor airflow referred to engine inlet conditions and overall compression system (Fan and HP Compressor) pressure ratio.

Initially, the performance map scalars are set at values predetermined by performance analysis to best represent the baseline component performance. Map scalars are used in extensively in the fault identification process. They are used in the baseline engine model to generate pretest predictions, provide an initial indication of data validity, and provide a baseline for trending component degradation. Finally, these same map scalars are varied in real-time to balance the engine cycle, match test data, and provide the basis for the real-time fault identification process.

APPENDIX J: HP TURBINE PERFORMANCE CHARACTERIZATION

HP Turbine Efficiency

The isentropic efficiency for an expansion process is determined from the following relationship:

$$ETA = \frac{HF - HI}{HFIS - HI}$$

where HI and HF are as described above and HFIS represents the total enthalpy at the final condition obtainable from an isentropic expansion.

HP Turbine efficiency is calculated as follows:

$$E41 = \frac{H42 - H41}{H42IS - H41}$$

where H42 is HP turbine discharge Station 42 (prior to addition of turbine cooling flow not available for turbine work) total enthalpy, H41 is HP turbine inlet (HP turbine nozzle

discharge) total enthalpy, and H42IS is the HP turbine discharge Station 42 total enthalpy for an isentropic expansion.

HP Turbine Nozzle Discharge Mass Flow

HP Turbine nozzle discharge mass flow (known turbine nozzle discharge mass flow function), W41F is calculated from the following function:

$$W41F = W41FFX \frac{P41}{\sqrt{T41}} A41T$$

where W41FFX is the known value of HP turbine inlet mass flow function, P41 is the HP turbine inlet (turbine nozzle discharge) total pressure, T41 is the HP turbine inlet total temperature, and A41T is the thermally corrected HP turbine inlet flow area.

HP Turbine Inlet Referred (Corrected) Mass Flow

HP Turbine inlet referred (corrected) mass flow (W41R) is calculated from the following relation:

$$W41R = \frac{W41\sqrt{TH41}}{DEL41}$$

where W41R is the HP Turbine nozzle discharge mass flow, and TH41 and DEL41 are defined by

$$TH41 = \frac{T41}{518.67}$$

and

$$DEL41 = \frac{P41}{14.696}$$

HP Turbine Referred (Corrected) Rotor Speed

HP Turbine referred (corrected) rotor speed (XN24R41) is calculated as follows:

$$XN24R41 = \frac{XN24}{\sqrt{TH41}}$$

APPENDIX K: LP TURBINE PERFORMANCE CHARACTERIZATION

LP Turbine Efficiency

LP Turbine isentropic efficiency is determined from the LP Turbine inlet and exit total enthalpies using the methodology previously presented in Appendix J to determine "HP TURBINE EFFICIENCY."

LP Turbine Inlet Referred (Corrected) Mass Flow

LP Turbine inlet referred (corrected) mass flow (W44R) is calculated from the following relation:

$$W44R = \frac{W44\sqrt{TH44}}{DEL44}$$

where W44R is the LP Turbine inlet mass flow, and TH44 and DEL44 are defined by

$$TH44 = \frac{T44}{518.67}$$

and

$$DEL44 = \frac{P44}{14.696}$$

LP Turbine Referred (Corrected) Rotor Speed

LP Turbine referred (corrected) rotor speed (XN2R44) is calculated as follows:

$$XN2R44 = \frac{XN2}{\sqrt{TH44}}$$

LP Turbine Discharge Total Temperature Referred to Station 2A

LP turbine discharge total temperature referred to Station 2A, T5R2A, is calculated as follows:

$$T5R2A = \frac{T5}{TH2A}$$

APPENDIX L: OVERALL ENGINE PERFORMANCE CHARACTERIZATION

Free-stream total pressure (P0) is calculated as follows:

$$P0 = \frac{P1}{ERAM}$$

where P1 is the inlet/engine interface total pressure and ERAM is the inlet ram recovery.

Net thrust evaluated from measured scale force is calculated from the following relationships:

FN = FG - FRAM

where

$$FRAM = \frac{W1 \times V0}{CG}$$

and

FG = XMVIS + ADPIS + FS.

FG is the engine gross thrust (see Chapter 3), FRAM is the ram drag, W1 is the mass flow rate at the inlet/engine interface plane, V0 is the simulated free-stream velocity, XMVIS is the inlet momentum evaluated at the isolation seal plane, ADPIS is the area-pressure difference term evaluated at the isolation seal plane and FS is the measured scale force with appropriate test cell correction factors.

Engine performance parameters include referred gross thrust (FGR2A), referred net thrust (FNR2A), referred fuel flow (WFTR2A), and referred specific fuel consumption (SFCR2A), defined as follows:

$$FGR2A = \frac{FG}{DEL2A}$$

$$FNR2A = \frac{FN}{DEL2A}$$

$$WFTR2A = \frac{WFT}{\sqrt{TH2A}}$$

$$SFCR2A = \frac{SFC}{\sqrt{TH2A}}$$

where WFT, the total fuel flow, is the sum of the burner and augmentor fuel flows

$$WFT = WFB + WFAB$$

and

$$SFC = \frac{WFT}{FN}.$$

APPENDIX M: SOFTWARE/HARDWARE ENGINEERING CONSIDERATIONS

Work and analysis for the research reported herein was performed by Dr. Csaba Biegl of Vanderbilt University.

Dr. Csaba Biegl

Vanderbilt University

Department of Electrical Engineering

Nashville, TN

All current component-level engine models used at AEDC have been written in FORTRAN. Several reasons for the use of FORTRAN are discussed below:

- Efficiency: Although there are more modern programming languages available, FORTRAN is still one of the best languages for fast numerical computations. The lack of "modern" features (e.g., pointers) makes it relatively easy to write very good optimizing compilers for FORTRAN. Additionally, the emerging High Performance Parallel FORTRAN (HPF) standard promises to simplify creation of parallel FORTRAN programs — at least for certain kinds of parallel architectures and for certain types of problems.

- Historical Reasons: Most of the existing engine model software was developed on mainframes and later ported to personal computers and workstations. On the mainframe the natural development language was FORTRAN.

- Developers: Most engine models are written by aeromechanical engineers who are fluent FORTRAN programmers who want to concentrate on the aeromechanical aspects of the problem instead of software engineering issues.

- Modular Legacy Software: Most models have been developed using existing FORTRAN software libraries and a modular structure applicable to arbitrary engine configurations. The modular structure typically consists of a main program, engine-dependent subroutines and data, component routines, and utility routines. The component routines use generalized logic for determining engine component operating characteristics and for performing aerothermodynamic calculations. Utility routines perform vital tasks such as cycle balancing, data interpolation, and calculation of gas properties. Together, the customized FORTRAN library and the engine-specific data become the model of the given engine.

The programming style used in most existing models relies heavily on global data structures stored in FORTRAN common blocks. Moreover, most software was not intended either for parallel execution or for integration with a graphical user interface. In

addition, input and output of the models is typically done with formatted FORTRAN data files. Effective techniques for converting existing FORTRAN software for parallel operation are presented in the following sections.

The next issue after selecting the hardware platform was to select the methods and evaluate the tools that are available for converting existing software for parallel execution. Generating parallel software from FORTRAN input is a technique that has been used with success for some time. FORTRAN is a primitive language when compared to later languages (e.g., ADA, C, Pascal). This simplicity is, in fact, an advantage when creating parallelizing compilers. In modern languages which support pointers, various subtle interactions are possible between variables — for this reason it is much harder to write a correct and efficient parallel software generator for them.

A number of successful FORTRAN compilers for shared memory Multiple Instruction Stream/Multiple Data Stream (MIMD) or Single Instruction Stream/Multiple Data Stream (SIMD) architectures have been developed. One of the best examples is the vectorizing FORTRAN compiler for Cray super-computers which parallelizes on the instruction level. These compilers use techniques such as unrolling loops and assigning the iterations of the loop to different processors. These techniques work well in shared memory architectures, but in distributed systems their applicability is limited. In addition, shared memory parallel computers have limited scalability due to memory contention

problems; this effectively limits the possible speedup that can be achieved using these techniques.

Generating good parallel software for scaleable distributed parallel architectures is a more complex task. Simple instruction-level parallelizing does not work very well on these architectures due to communication overhead problems. For these reasons, parallelizing FORTRAN compilers for distributed architectures are just High Performance FORTRAN (HPF) starting to emerge. These compilers frequently rely on the extensions of the FORTRAN 90 language standard.

Two typical parallel software generators which claim FORTRAN support on distributed architectures are the SAGE (Distribute pC++, 1993) toolkit and the FORGE (1992) family of compilers. The SAGE toolkit is being developed at the Department of Computer Science, Indiana University, Bloomington, under a DARPA grant. Its primary aim is to implement a parallel version of the C++ language (pC++), but it supports C and FORTRAN as well. The program source is first translated by one of the available compiler front-end translators (pC++, C or FORTRAN) into an intermediate form. The output of this phase is a huge dependency file that contains the representation of all of the program's instructions and their interactions. This dependency output in itself is a valuable resource; various browsers can be used to analyze program structure, calling sequences, and data usage. The toolkit also provides software generators that can compile the dependency file for various shared memory and distributed parallel target

architectures. The supported distributed target environments include the Intel Paragon system and two widely accepted message passing standards for distributed computing: Parallel Virtual Machine (PVM, 1994) and the MPI (1994). The SAGE FORTRAN front-end currently processes only FORTRAN 90 input. Support for HPF extensions is being developed. The current implementation will not parallelize instructions operating on global data.

The FORGE compiler family is a commercial product offered by Advanced Parallel Research, Inc. It contains both interactive program dependency analyzers/browsers and batch mode parallelizing compilers. The interactive tools only support FORTRAN 90, while the batch mode compilers support both FORTRAN 90 and FORTRAN 77. The compilers generate FORTRAN 77 output for either shared memory or distributed architectures. The distributed architecture output contains calls to FORGE's own communication library. This library is implemented as a wrapper on top of commonly available communication interfaces such as PVM, EXPRESS, and TCP-IP. Most of the tools in the compiler family seem to be geared towards the FORTRAN 90 language and HPF, but some FORTRAN 77 parallelization support is provided as well.

In summary, both SAGE and FORGE generate better parallel software from FORTRAN 90 input with HPF directives than from FORTRAN 77 input. However, FORTRAN 77 programs relying extensively on global variables will probably never be parallelized by automatic tools to the same extent as HPF input. As an alternative to

existing parallel software generators, explicit algorithmic parallelization and an associated tookit are discussed in the following sections.

APPENDIX N: SCALABLE SOFTWARE PARADIGM AND MESSAGE-PASSING INTERFACE FOR THE TARGET PLATFORMS

The message passing interface employed to parallelize the model-based algorithms is based on the MPI (1994) standard. MPI is the emerging message passing standard for parallel applications; it is superseding older interfaces such as Parallel Virtual Machine (PVM). MPI is available from virtually all workstation-based HPC platform vendors and public domain versions are also available for installation on networked workstations.

The model-based fault identification algorithms have been run successfully on a number of hardware platforms including UNIX workstations and Windows based Personal Computers. Many of these platforms do not have MPI installed. In order to preserve a common code base an MPI interface layer was defined. The core of the application code does not use MPI calls directly, but it always goes through this interface layer. A "dummy" version of this interface code was also developed for platforms where MPI is not available. The interface code provides the following services:

Data propagation entries; these are wrappers around the MPI send and receive calls. They are provided for all data types.

Initialization and exit entries; these calls provide a more convenient interface to the calls MPI requires to start up a parallel application. The initialization entry returns the status variables based upon which the process can decide its role in the execution of the parallel application.

Event loop support entry; in the parallel version of both applications there is a main coordinator process that also handles the graphical user interface. This coordinating process has to react to messages from the other processes and to user interface events. This call implements a waiting mechanism with time out, which can handle both event types.

The advantage of this approach is that several alternative versions of the message passing layer can be developed. These can be substituted without any impact on the application code. Currently three versions are supported:

The "dummy" version which does not support parallel operation; its initialization code returns status variables to the application which will cause it to work in sequential mode.

The MPI version that implements the interface using MPI calls.

A TCP-IP based implementation of the interface. This interface has been used in earlier parallelization efforts.

The parallelization work described in this report is part of a longer-term effort to achieve near real-time operation of transient engine test data analysis software components. The model-based data validation system has been paralleled earlier using a low-level TCP-IP based approach. Comparing the experiences gained during the earlier and the current MPI-based efforts, the following observations can be made:

MPI provides a much more convenient way (the 'mpirun' utility) to execute the parallel application. The low-level TCP-IP based approach required the user to manually manage host assignments, ports, etc.

A well-configured installation of MPI uses (in theory) the best available communication media (shared memory, etc.) between the processes of a parallel application. No additional effort is required on the user's part when moving the application between different hardware platforms supporting MPI.

The current 1.x version of the MPI standard lacks a version of the 'MPI_probe' (or 'MPI_iprobe') facility with timeout. Such a facility would have made the implementation of the integrated graphical user interface and MPI event handling code more efficient.

APPENDIX 0: SIMULATION INTERFACE

Work and analysis for the research reported herein was performed by Dr. Csaba Biegl of Vanderbilt University.

Dr. Csaba Biegl

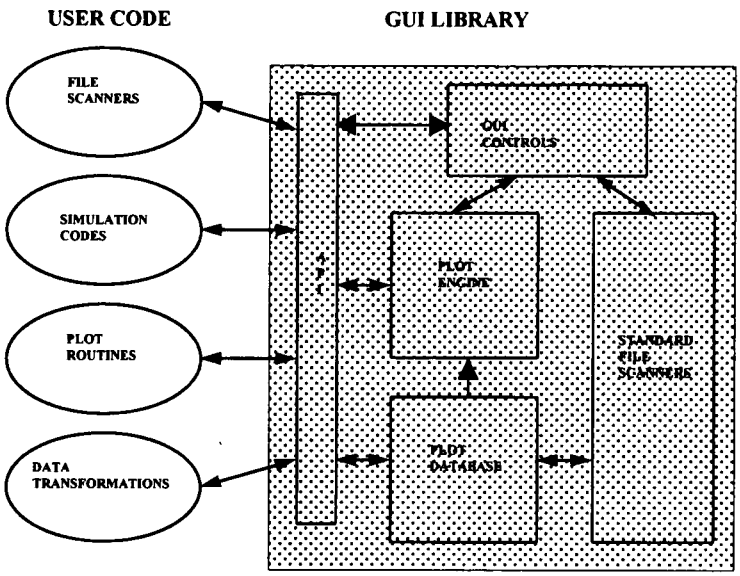
Vanderbilt University

Department of Electrical Engineering

Nashville, TN

In a related effort, a universal Graphical User Interface (GUI) library has been developed for use with turbine engine model software. The purpose of this GUI is to support the interactive execution of various simulations and provide data visualization capabilities. The main design goals of the GUI were universal applicability and user extensibility. The same user interface should support several different engine model types and their applications. This was achieved by providing a set of core functions for database services and basic plots and an application program interface (API) for implementation of customized functions. Most of the custom functions are implemented using user-defined FORTRAN software (e.g., engine models, user-defined plot formats,

and custom data file interfaces) which can be registered into the basic GUI framework. The main software components of the GUI design are shown below.



The main functional components of the GUI library include:

1. *The plot database.* This memory resident database holds records of parameters that can be plotted. The database is organized by data set and parameter names. Each data set can hold one or more named parameters. Each named parameter can have a stored one-dimensional data record of arbitrary length.

Data records are identified jointly by the parameter name and the data set name. When plotting data from the plot database, two records have to be identified: one for the Y-axis, and one for the X-axis. As a convenience feature, the GUI maintains a default X-axis parameter name for each data set (commonly this is the time variable) and applies it whenever a parameter is selected from the given data set for plotting. This can then be

overridden if a parameter versus parameter plot is required. The GUI allows any pairing of data records for the X and Y-axes (even from different data sets), it is up to the user to select data record pairs which make sense to plot against each other.

Besides the default X-axis variable, data sets have a few other default parameters, including:

- Default plot title for any plot window that uses data from the data set.

- Data set size limit. This can be used to control the growth of the database.

- Run-time plot size limit. This limit is effective only when a built-in application code is executing. To speed up plot updates it is possible to restrict the number of data points from the stored data records that are used to generate plots. Once the application finishes the execution the plots are redrawn with all available points.

There are two ways to enter data into the plot database: One is to scan output files generated by external programs. The GUI currently supports several common data file formats, or custom scanners can be written for application-specific data file formats. The other way is to enter data at run-time from an integrated (e.g. linked in) model using the API calls.

2. *The plot engine* is used to display data from the plot database. The plot engine is based on the GNUPLOT plot program. (Note: despite its name, GNUPLOT does not fall under the GNU software distribution rules.) All features of the stand-alone GNUPLOT program are available in the version embedded into the GUI, but the embedded version also provides additional ways to generate both 2D and 3D plots. 2D plots can be generated several ways:

- By selecting data records from the plot database for the X and Y-axes.
- By applying transformation functions to data records selected from the plot database for the X and Y-axes.
- By generating plot data from user supplied subroutines (these are called programmed plots).

In most cases several 2D data record pairs can be displayed in a single plot window. On the other hand, the 3D plot facility allows only on plot per window. Additionally, 3D data plots can only be generated using the programmed plot facility.

On the user interface screen the plot windows are organized as pages. Each plot page has a user-defined layout with 1 to 12 plots. The GUI supports up to 1000 plot pages, giving up to 12,000 user configurable plot windows. The plot engine offers several customization options for each individual plot window. These options include the color,

style (line, point, etc.), coordinate limits and titles for each plot window. Hardcopies can also be generated from the plots on a variety of supported printers.

3. *The standard file scanners* can input data from various file formats into the plot database. Supported formats include:

   - Exported (ASCII) versions of spreadsheets. Such files contain tabulated data, either in vertical (first row contains the parameter names, with the corresponding values in successive rows) or horizontal (first column of every row is the parameter name, additional columns contain the values) formats.

   - Raw data (numbers only) in ASCII and binary (IEEE 32 and 64 bit floating point) formats. Such data requires separate user input to supply the parameter names under which the read data is stored.

4. *The graphical user interface controls* allow the user to configure plots, start and control the execution of the interfaced application code and to configure many aspects of the GUI application's operation.

5. *The application program interface* (API) provides the entry points into the GUI using which user code can customize the look of the application and interface to the other components. The API is provided for C or C++ programs and for FORTRAN

programs compiled with the F2C translator, a 'standard' UNIX f77 compiler, or the Microsoft Developer Studio® FORTRAN tools. FORTRAN entry points are implemented as wrappers around the corresponding C entry points to adapt the interface to the FORTRAN subroutine calling conventions.

The graphical interface library has been implemented for a variety of platforms including Intel-based PC-s running DOS (full screen) or 32-bit Windows applications (WIN32S, Windows™ 95, and Windows NT) and several UNIX workstations with the X window environment (Linux, SGI, Sun, DEC Alpha and HP). The universal interface was written in C++ using common interface elements (e.g., buttons and scrollbars) which were adapted from public domain toolkits. Some new widgets were developed because none of the commercially available "portable" toolkits cover all of the targeted plat-forms (DOS full-screen graphics, Windows, and the X Window system) -- at least not without unacceptable restrictions (like limiting the compiler choices under DOS and Windows or requiring a Motif license for workstations).

VITA

Don Malloy attended Grant Community High School in Fox Lake, Illinois, obtained his high undergraduate degree in Mechanical Engineering from the University of Illinois, and received a Masters degree in Mechanical Engineering from Memphis State University. Mr. Malloy joined Sverdrup Technology, Inc., technical services contractor for the Air Force's Arnold Engineering Development Center as a test and evaluation engineer in 1986 where he helped develop, qualify, and troubleshoot operational problems with turbine engine propulsion systems for the Advanced Tactical Fighter (ATF). Recently his work has focused on the development of analysis and evaluation capabilities to evaluate, monitor, and improve performance of advanced aeropropulsion systems. His efforts linking computer utilization in all phases of test led to his current doctoral research work coupling control and information technologies with thermodynamics, fluid dynamics and propulsion, numerical modeling, and instrumentation and measurement technologies to evaluate, monitor, and improve system performance of advanced aeropropulsion systems.

Technical society participation in organizations as diverse as AIAA and SAE and his licensing and participation as a Registered Professional Engineer reflect his diversity and concerns not only with the technical aspects of engineering, but also with the professional, ethical, economic, legal, political, and social aspects of engineering. Mr.

Malloy is a past member of the AIAA Air Breathing Propulsion Technical Committee and recently served as AIAA/SAE/ASME Technical Program Chair for the AIAA/ASME/SAE/ASEE Joint Propulsion Conference. Mr. Malloy has organized numerous technical sessions, is an active member of the SAE E-32 Engine Condition Monitoring Committee, and currently serves as Chairperson of the SAE Engine Condition Monitoring Technologies Committee.