Dynamical System Modeling of a Micro Gas Turbine Engine

by

Chunmei Liu

B.S., Harbin Institute of Technology (1988)
M.S., Harbin Institute of Technology (1991)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2000

© Massachusetts Institute of Technology 2000. All rights reserved.

Author .....
Department of Aeronautics and Astronautics
May 12, 2000

Certified by .....
James D. Paduano
Principal Research Engineer
Thesis Supervisor

Accepted by .....
Nesbitt W. Hagood, IV
Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
SEP 07 2000
LIBRARIES
Aero

Dynamical System Modeling of a Micro Gas Turbine Engine

by

Chunmei Liu

Submitted to the Department of Aeronautics and Astronautics
on May 12, 2000, in partial fulfillment of the
requirements for the degree of
Master of Science

Abstract

Since 1995, MIT has been developing the technology for a micro gas turbine engine capable of producing tens of watts of power in a package less than one cubic centimeter in volume. The demo engine developed for this research has low and diabtic component performance and severe heat transfer from the turbine side to the compressor side. The goals of this thesis are developing a dynamical model and providing a simulation platform for predicting the microengine performance and control design, as well as giving an estimate of the microengine behavior under current design.

The thesis first analyzes and models the dynamical components of the microengine. Then a nonlinear model, a linearized model, and corresponding simulators are derived, which are valid for estimating both the steady state and transient behavior. Simulations are also performed to estimate the microengine performance, which include steady states, linear properties, transient behavior, and sensor options. A parameter study and investigation of the startup process are also performed.

Analysis and simulations show that there is the possibility of increasing turbine inlet temperature with decreasing fuel flow rate in some regions. Because of the severe heat transfer and this turbine inlet temperature trend, the microengine system behaves like a second-order system with low damping and poor linear properties. This increases the possibility of surge, over-temperature and over-speed. This also implies a potentially complex control system. The surge margin at the design point is large, but accelerating directly from minimum speed to 100% speed still causes surge.

Investigation of the sensor options shows that temperature sensors have relatively fast response time but give multiple estimates of the engine state. Pressure sensors have relatively slow response time but they change monotonically with the engine state. So the future choice of sensors may be some combinations of the two. For the purpose of feedback control, the system is observable from speed, temperature, or pressure measurements.

Parameter studies show that the engine performance doesn't change significantly with changes in either nozzle area or the coefficient relating heat flux to compressor efficiency. It does depend strongly on the coefficient relating heat flux to compressor pressure ratio. The value of the compressor peak efficiency affects the engine operation only when it is inside the range of the engine operation. Finally, parameter studies indicate that, to obtain improved transient behavior with less possibility of surge, over-temperature and over-speed, and to simplify the system analysis and design as well as the design and implementation of control laws, it is desirable to reduce the ratio of rotor mechanical inertia to thermal inertia, e.g. by slowing the thermal dynamics. This can in some cases decouple the dynamics of rotor acceleration and heat transfer.

Several methods were shown to improve the startup process: higher start speed, higher start spool temperature, and higher start fuel flow input. Simulations also show that the efficiency gradient affects the transient behavior of the engine significantly, thereby effecting the startup process.

Finally, the analysis and modeling methodologies presented in this thesis can be applied to other engines with severe heat transfer. The estimates of the engine performance can serve as a reference of similar engines as well.

Thesis Supervisor: James D. Paduano
Title: Principal Research Engineer

Acknowledgements

I would like to express my gratitude to Dr. Paduano for his invaluable guidance and direction throughout the course of this research, as well as his continuous encouragement and support.

I would also like to thank John Protz, who discussed nearly every piece of my research with me and gave precious suggestions; Yifang Gong, who was always there when I was stuck in the research. I would also like to give my special thanks to Prof. Epstein and Dr. Tan who made my research possible and gave me their guidance. In addition, I would like to acknowledge Dr. Stuart Jacobson for his help and willingness to answer all my questions.

I am also grateful to all those people who have helped me to make my stay here a pleasant one. In particular, I'll like to thank Yang-Sheng Tzeng and Shengfang Liao, as well as my officemates Kenneth Gordon, Brian Schuler, Sanith Wijesinghe, and Margarita Brito, who helped me a lot on my research, classes and the daily life; Lori Martinez, for her cheerfulness and friendliness; all other people in the microengine project and GTL, for the data, help and cooperations. I'm also thankful to many of my friends both here in the USA and there in China, who continue to give me a colorful life and are ready to give me a hand whenever I need one.

Last but not least, I'd like to thank my husband, my parents and my sisters, who are forever my sources of inspiration, encouragement, and support.

Contents

List of Figures

List of Tables

Nomenclature

Roman

a	Speed of sound; constant
A	Area; coefficient matrix
B	Coefficient matrix
C	Coefficient matrix
C_p	Specific heat at constant pressure
D	Coefficient matrix
e	Constant
f	Fuel-to-air ratio; function
h	Heat value of the fuel; convective heat transfer coefficient
J	Rotor inertia
k	Constant
L	Length
m	Mass
M	Mass; Mach number
N	Rotation speed
P	Total pressure
$Power$	Power
R	Gas constant; radius
Re	Renolds number
T	Total temperature
u	Speed
V	Volume
$Work$	Work
z	Constant

Greek

β	Angle
γ	Ratio of specific heats
η	Efficiency
π	Total-to-total pressure ratio
τ	Time constant; characteristic time; temperature ratio
ρ	Density
ω	Rotation speed

Superscripts

abs	Absolute value
no	Value with no heat flux
rel	Relative value

Subscripts

0	Operating point
2	Inlet of the compressor
3	Exit of the compressor
41	Inlet of the turbine
45	Exit of the turbine
acc	Acceleration
b	Combustor
c	Compressor
cm	Thermal inertia
$comp$	Compressor
des	Design point
ef	Efficiency
f	Fuel
gas	Gas dynamics
jg	Mechanical inertia
min	Minimum value
n	Nozzle
p	Percentage
pi	Pressure ratio
q	Heat flux related
rad	Radial component
s	Static quantity
t	Turbine; total quantity; temperature related
tan	Tangential component
$thermo$	heat transfer dynamics
$turb$	Turbine
v	Volume related
w	Spool

Full quantities

C_w	Specific heat of the rotor
e_{peak}	Peak efficiency the compressor can reach
$\dot{m}$	Mass flow rate
P_s	Static pressure
$\dot{Q}$	Heat flux
T_s	Static temperature

Chapter 1

Introduction

1.1 Background

Recent advances in the field of micro-fabrication have opened the possibility of building a micro-gas turbine engine. MIT is developing the technology for such engines. These are millimeter to centimeter-size heat engines fabricated with semiconductor industry micromachining techniques. As such, they are micro electro-mechanical systems (MEMS) devices. They contain all the main functional components of a conventional large-scale gas turbine engine. Preliminary studies show that these devices may ultimately be capable of producing 10-100W of power or 10-50 grams of thrust in less than a cubic centimeter. Applications include battery replacements, propulsion for small air vehicles, and a variety of blowers, compressors, and heat pumps.

Refractory structural ceramics, such as silicon nitride (Si_3N_4) and silicon carbide (SiC), have excellent mechanical, thermal, and chemical properties for gas turbine applications permitting uncooled operation up to the 1500-1700K combustor exit temperature range ([1]). While there is an ongoing effort to develop the needed SiC microfabrication technology, sufficient technology has not been demonstrated to date. On the other hand, most of the necessary technology has been demonstrated in silicon. Thus, for simplicity of construction and minimum technical risk as opposed to high power output and good fuel consumption, current efforts are focused on demonstrating a working micro gas turbine engine, the "demo-engine", which is made of silicon. Based on the analysis and design experience obtained from this demo engine, as well as the process development for more refractory materials, the final high temperature microengine will be built.

The baseline design of the demo engine (henceforth referred to as the "microengine") is shown in Fig. 1-1. Its basic geometry and parameters at the design point can be found in Fig. 1-2 ([9]). It has several characteristics which are different from conventional large-scale engines, some of which are listed below:

- Small scale.

- Low and diabatic component performance.

  The micro compressor and the micro turbine are a few millimeters in size. Hence, even at transonic tip Mach numbers, the Reynolds number is of the order of a few thousands only. This low Reynolds number, as well as the micofabrication constraints, lead to low component performance ([8]). In addition, the component performance is a function of the heat flux, as shown below.

- Heat transfer.

  Because of the low component performance, a combustor exit temperature greater than 1400K is needed for self-sustaining engine operation. But silicon must remain below about 950K to retain sufficient strength for the rotating structure ([1]). The approach of rejecting the heat the turbine absorbs into the compressor flow path was chosen to cool the turbine. This heat transfer from the turbine side to the compressor side has two negative effects. One effect is that the heat transfer dynamics become strongly coupled with the rotor acceleration dynamics (this effect will be discussed in detail in the following chapters). Another effect is the reduction of the component performance ([1]). Although the effect on turbine performance is relatively small, the effect on compressor pressure ratio and efficiency is significant.

Since these characteristics seldom exist in conventional large scale engines, they motivate us to investigate the performance and dynamic behavior of an engine with these characteristics. From the microengine project perspective, the results will enable prediction of the microengine performance. From a general perspective, the results can serve as a reference for future design and analysis of similar engines.

1.2 Technical Objectives and Development Approaches

The overall objectives of the thesis are to derive a model for the microengine which can describe the micriengine behaviors and to provide a simulation platform to estimate engine behavior. These objectives are pursued by performing the following steps:

- First the dynamics of the main engine processes are investigated and the important ones are determined.

- Then, component performance information is assembled based on the current available data.

- Based on the dynamics analysis and the component performance, a nonlinear model for the microengine is derived. This model is valid at both design and off-design points, and can describe transients as well.

- Linearized models are derived by linearization around steady operating points. These linearized models enable us to make use of the large body of results for linear systems to analyze the properties of the system, especially the transient properties. The linearized models will also be references for the development of control systems in the future.
- The simulators corresponding to the nonlinear model and linearized models are developed using a combination of C code and the Matlab/Simulink interface for dynamics and control.

Another objective of this thesis is to estimate the engine performance using the developed simulators. The performance characteristics that are of interest include:

- the running line and the steady states;
- the linear properties;
- the stability and the transient acceleration and deceleration processes;
- sensor options for measuring system behavior and for feedback control;
- the effect of the reduction of the compressor pressure ratio and efficiency to the engine performance, due to the heat addition into the compressor;
- the effect of the limit peak efficiency which the compressor can reach;
- the effect of the heat transfer dynamics;
- the effect of the nozzle area;
- the startup process.

1.3 Contributions and Organization of the Thesis

The main contributions of the thesis are:

1. obtaining the models for the engine with heat transfer;
2. providing a simulation platform for the overall engine behavior and future work;
3. finding out the main characteristics of the microengine performance, which include:
    - Due to the heat transfer dynamics, the system becomes a second-order system instead of a first-order system, which is one of the main differences from a conventional engine. The spool behaves like an energy storage device.
    - There is the possibility of increasing T_{41} with decreasing $\dot{m}_f$ in some regions, which is another main difference from a conventional engine.

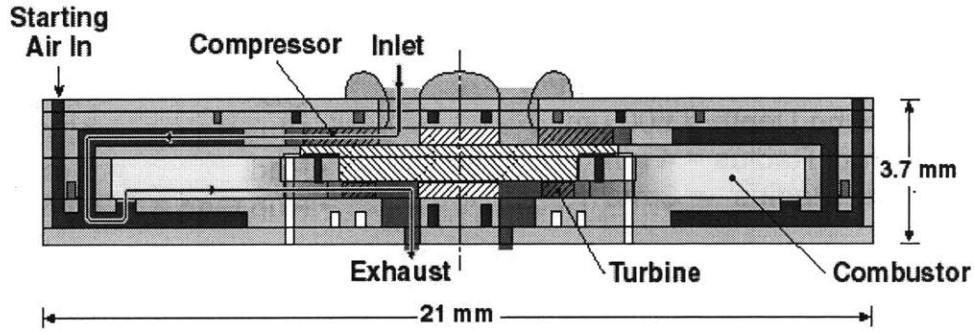
- The system behaves quite different at different operating points, and the linear properties are not good.
- The acceleration time constant is on the order of several hundred milliseconds.
- The lowest damping of the second-order system can reach 0.15, and the corresponding overshoot can be as large as 62%. Thus under some conditions it is possible to get over-temperature or over-speed unless precautions are taken.
- The surge margin is relatively large, but accelerating the engine from minimum speed to 100% speed will still cause surge. Therefore, an engine of this type would have to incorporate a control system to avoid fast acceleration.

4. investigating the sensor options. There are two aspects regarding the sensor choices. One aspect is for measuring the current engine operating point. Another aspect is for providing feedbacks for the future control system. Both are considered in this thesis.

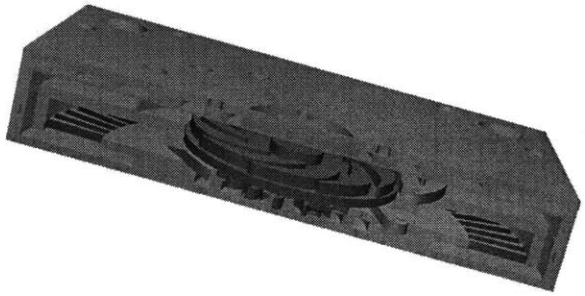
5. performing sensitivity analysis of the engine performance to certain elements. The elements include: the reduction of the compressor performance due to the heat addition, the limit peak efficiency the compressor can reach, the heat transfer dynamics, and the nozzle area.

6. simulating several procedures for the startup process, which include: start the engine at different speeds, different fuel inputs, and different spool temperatures. The effect of efficiency gradient of the compressor map on the startup process is simulated as well.

The organization of the thesis is as follows: chapter two analyzes the main dynamical elements in the microengine system, assembles the component characteristics, and derives the nonlinear and linearized models, as well as their corresponding simulators. Chapter three gives the simulation results for the microengine, which includes the steady states and transient behaviors of the microengine, the special properties of the microengine performance compared with conventional engines, and the sensor considerations. Chapter four investigates the effects of several elements on the engine performance, which include: the reduction of the compressor performance due to heat addition, the limit peak efficiency the compressor can reach, the heat transfer dynamics, and the nozzle area. These investigations are referred to as "parameter studies". Chapter four also investigates the startup process. Chapter five draws conclusions for the overall work done and gives some recommendations for future work.



11g Thrust
16g/hr H2 Fuel Burn
1600K (2420 F) Gas Temp.
1.2 million RPM rotor



3D Schematic Diagram

Figure 1-1: The MIT Demo Microengine

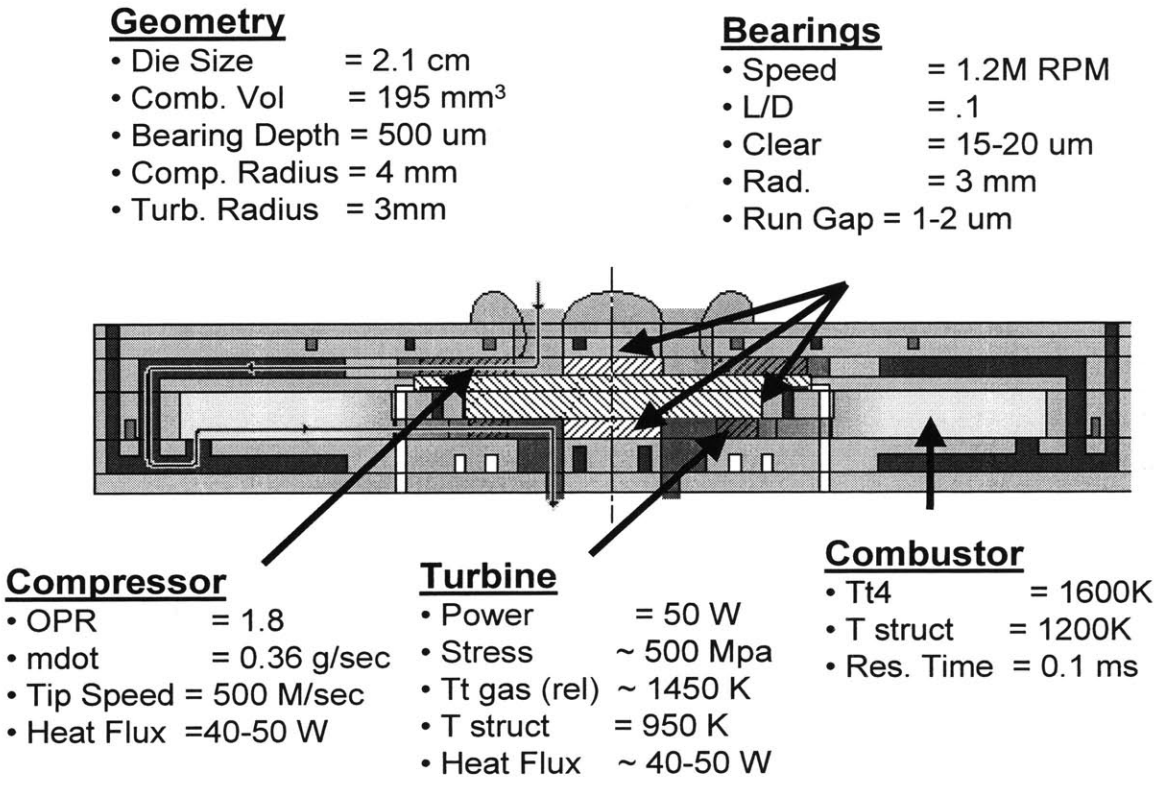


Figure 1-2: Demo Microengine Mechanical Layout

Chapter 2

Microengine Modeling and Simulator Development

This chapter derives dynamical models for the microengine, including both a nonlinear model and linearized models, as well as corresponding simulators. First the dynamics of the primary physical processes in the miroengine are investigated in order to determine which processes are important and should be incorporated in the model; then the nonlinear model and its corresponding simulators are developed; finally the linearized models are derived by linearization around the steady points.

2.1 Order of Magnitude Analysis

There are three main dynamical elements that are accounted for in this analysis: rotor acceleration, gas dynamics, and heat transfer. In addition, there is another dynamical element which may be important during the operation of the microengine: the dynamics of emptying of the fuel tank. This section estimates the time constants of these processes and determines the important dynamics, in order to arrive at a simple and effective model.

2.1.1 Rotor Acceleration Dynamics

The time constant for the rotor acceleration dynamics can be estimated using [7]:

$$\tau_{acc} = \frac{4\pi^2 J N^2 \frac{\gamma}{\gamma-1}}{C_{pc} T_2 \dot{m}_2 (\tau_c)_0},$$

where the subscript "0" refers to the current operating point, τ_{acc} is the approximate acceleration time constant, J is the rotor inertia, N is the rotation speed in *rps* (i.e., the units for $2\pi N$ are *rad/s*), γ is the ratio of specific heats, C_{pc} is the specific heat at constant pressure of the air at the

compressor, T_2 is the total temperature at the inlet of the compressor, $\dot{m}_2$ is the air mass flow at the inlet of the compressor, and τ_c is the temperature ratio of the compressor.

The parameters for the microengine at its design point, as well as the other parameter needed are (Fig. 1-2 and [9]):

$$2\pi N = 1.2 \times 10^6 rpm \approx 1.25 \times 10^5 rad/s,$$

$$\gamma = 1.4,$$

$$J = 9.3 \times 10^{-10} kgm^2,$$

$$C_{pc} = 1004.5 J/kgK,$$

$$T_2 = 300K,$$

$$\dot{m}_2 = 0.36 g/s = 0.36 \times 10^{-3} kg/s,$$

$$(\tau_c)_0 \approx 1.38.$$

Using these values, the time constant of the rotor acceleration can be estimated as:

$$\tau_{acc} \sim 0.34s. \tag{2.1}$$

2.1.2 Gas Dynamics

The engine is modeled as a Helmoltz resonator ([5]) to analyze the gas dynamics, as shown in Fig. 2-1. By applying mass conservation, momentum conservation, and assuming:

$$u_2 = u_3 \equiv u,$$

$$P_3 = P_{41} \equiv P,$$

$$\Delta P_T = \frac{1}{2} k_T u_{41}^2.$$

where the subscripts "2", "3", "41" and "45" refer to the inlet and exit of the compressor and the turbine, respectively, as shown in Fig. 2-1. u is the air speed, P is the pressure, $\Delta P_T \equiv P_{41} - P_{45}$ is the pressure drop across the turbine, and k_T is a constant.

The characteristic time of the gas dynamics can be computed ([5]) to be:

$$\tau_{gas} = \frac{1}{a_b} \sqrt{\frac{L_c V_b}{A_b}},$$

where τ_{gas} is the characteristic time of the gas dynamics, V_b is the volume of combustor, L_c is the

characteristic length of the duct of the compressor, a_b is the speed of sound in the combustor, and $\overline{A}_b$ is the average area of A_3 and A_{41}.

For microengine, according to Fig. 1-1, Fig. 1-2 and [9], the geometry is as:

$$L_c \approx 13mm,$$

$$\overline{A}_b \approx 3.8\pi mm^2,$$

$$V_b \approx 190mm^3.$$

Using these values, an estimate for the gas dynamics characteristic time is:

$$\tau_{gas} \sim 0.05s, \tag{2.2}$$

which is much smaller than the rotor acceleration time constant.

2.1.3 Heat Transfer Dynamics

This subsection estimates the heat transfer dynamics. Due to the relatively high thermal conductivity of silicon, the microengine has large heat flow from the turbine side to the compressor side (40W compared to the 40W of shaft power absorbed by the compressor) [10]. This heat flow not only decreases the compressor pressure ratio and efficiency and lowers the thermodynamic efficiency of the cycle, but also makes the dynamics of the engine more complex. This is because its dynamics lie in the same range as the rotor acceleration, which will be shown by estimating the time constant of the heat transfer dynamics.

In the heat flux path, the major sources and sinks are the fluid across the compressor and the turbine. Other minor sources and sinks include the blade tips, the seals and the journal bearing ([10]). Here for the first order estimation, these minor heat sources and sinks are neglected, and the following assumptions are made:

- uniform temperature of the whole spool T_w;
- bulk temperature and bulk heat transfer coefficients can be used;
- the bulk heat transfer coefficients are independent of the flow and the structural temperature;
- when calculating the heat flux from the flow at the turbine side into the rotor and the heat flux from the rotor into the flow at the compressor side, we can use characteristic bulk temperatures T_{turb} and T_{comp} to approximate the temperature of the flow on the turbine side and the compressor side, respectively.

Based on these assumptions, the heat flux from the flow at the turbine side to the rotor $\dot{Q}_t$ and the heat flux from the rotor to the flow at the compressor side $\dot{Q}_c$ can be represented as:

$$\dot{Q}_t = h_t A_t (T_{turb} - T_w),$$

$$\dot{Q}_c = h_c A_c (T_w - T_{comp}),$$

where A_t and A_c are the convection area at the turbine side and the compressor side, respectively, and h_t and h_c are the convective heat transfer coefficients at the turbine side and the compressor side, respectively.

Thus, the net heat flux flowing into the rotor is:

$$\dot{Q} = \dot{Q}_t - \dot{Q}_c = h_t A_t (T_{turb} - T_w) - h_c A_c (T_w - T_{comp}) \approx \overline{hA}(T - T_w), \tag{2.3}$$

where

$$\overline{hA} \equiv h_t A_t + h_c A_c,$$

and

$$T \equiv \frac{h_t A_t T_{turb} + h_c A_c T_{comp}}{h_t A_t + h_c A_c}.$$

Applying energy conservation for the spool yields the following:

$$\dot{Q} = C_w M_w \frac{dT_w}{dt}, \tag{2.4}$$

where C_w and M_w are the specific heat and the mass of the rotor, respectively. Combining equation 2.3 and equation 2.4, we arrive at:

$$C_w M_w \frac{dT_w}{dt} = \overline{hA}(T - T_w).$$

Therefore the time constant for the thermal dynamics is approximately:

$$\tau_{thermo} = \frac{C_w M_w}{\overline{hA}}. \tag{2.5}$$

For the microengine, the data is as follows [4, 9]:

$$C_w \sim 700 J/kgK \text{ (silicon property at } T \sim 1600K),$$

$$\rho_w \sim 2400 kg/m^3,$$

$$V_w \sim 10^{-8} m^3 (10 mm^3),$$

$$h \sim 1000 w/m^2 K,$$

$$A \sim 10^{-5} - 10^{-4} m^2.$$

Using these values the thermal time constant can be estimated as follows:

$$\tau_{thermo} \sim 0.17 - 1.68 s. \tag{2.6}$$

2.1.4 Dynamics of Emptying the Fuel Tank

In conventional engines, the dynamics of the fuel mass flow are neglected. In the microengine, because the engine is fed by a small pressurized fuel tank, the rate of change of the fuel tank pressure may be fast enough that we need to be concerned with its effects on the fuel mass flow rate. This subsection estimates these dynamics.

Assuming that the valve of the fuel tank works at the choked condition, the fuel mass flow can be computed as:

$$\dot{m} = \frac{P_t}{\sqrt{T_t}} A (1 + \frac{\gamma - 1}{2})^{-\frac{\gamma+1}{2(\gamma-1)}} \sqrt{\frac{\gamma}{R}},$$

where P_t, T_t are the total pressure and total temperature just ahead of the choked valve, A is the area of the valve, and $\dot{m}$ is the fuel mass flow.

Inside the tank, assuming the gas behaves as a perfect gas yields the following:

$$P_s = \rho R T_s = \frac{M}{V} R T_s,$$

where M is the mass of the fuel inside the tank, T_s is the temperature, and V is the volume of the tank.

Because the fuel has nearly zero velocity inside the tank, $P_t \approx P_s$ and $T_t \approx T_s$. Finally mass conservation for the fuel tank and the duct ahead of the valve can be written as:

$$\dot{m} = -\dot{M}.$$

Combining all of the above equations and doing some manipulations results in:

$$\dot{M} = -\dot{m} = -\sqrt{\gamma R T_s} (1 + \frac{\gamma - 1}{2})^{-\frac{\gamma+1}{2(\gamma-1)}} \frac{A}{V} M.$$

So the time constant for emptying the fuel tank is:

$$\tau_f = \frac{1}{\sqrt{\gamma R T_s}} (1 + \frac{\gamma - 1}{2})^{\frac{\gamma+1}{2(\gamma-1)}} \frac{V}{A}. \tag{2.7}$$

This time constant is a function of the geometry of the fuel tank and valve, as well as the chosen fuel (which determines γ). The small size of the fuel tank, characterized by its volume V, tends to

lead to a small time constant . If the time constant is small enough to be comparable to the engine main dynamics, like the rotor acceleration dynamics, the dynamics of the fuel mass flow cannot be neglected.

2.1.5 Summary of the Order-of-Magnitude Analysis of the Dynamics

The analysis in above subsections gives the following results:

$$\tau_{gas}(0.05s) \ll \tau_{acc}(0.34s),$$

$$\tau_{acc}(0.34s) \sim \tau_{thermo}(0.17 - 1.68s). \tag{2.8}$$

The much shorter characteristic time of the gas dynamics, given by the first equation, enables us to use a quasi-steady model of these process, i.e., gas dynamics can be computed as quasi-steady values.

The comparable time scale of rotor acceleration and heat transfer, given by the second equation, means that there can be strong coupling between rotor acceleration and heat transfer. This makes the system a second-order system, which is one of the main differences between the microengine and a conventional engine. Generally, in conventional engines the spool has relatively low conductivity and the thermal dynamics are very slow. Thus the thermal dynamics are usually ignored during most transients. The system is then first-order instead of second-order. The second-order nature of the microengine system makes the response of the engine more complex than a conventional engine, which will be obvious in the following sections of this chapter and in chapter 3. In addition, it may cause more overshoot. Considering the nonlinear character of the engine transient operation, it becomes important to do simulations to predict the behavior of the engine during operation, especially when spool acceleration and deceleration occur.

It is worth noting that the estimated value of τ_{acc} is on the order of 100 msecs, which is much longer than the originally expected milliseconds. Previously, τ_{acc} was expected to be much smaller because of the small inertia of the microengine. Actually after scaling down the engine, the engine net available torque is also very small. This leads to the not-so-small time scale for the microengine speed-up. The simulation results in chapter 3 are consistent with this estimate.

One remaining dynamical element in the propulsion system is the dynamics of the emptying of the fuel tank. Dynamical analysis shows that the time constant for these dynamics, denoted by τ_f, is a function of the geometry of the fuel tank. After the design of the fuel tank is completed, if τ_f is small enough, it may be important to account for changes in the fuel flow due to variations of the fuel tank pressure. Currently this issue is disregarded.

In summary, after first principle analysis of the order-of-magnitude of the dynamics, the following conclusions are drawn:

- Because gas dynamics are much faster than the rotor acceleration dynamics and the heat transfer dynamics, gas dynamics can be computed as quasi-steady values;
- Because the rotor acceleration dynamics and the heat transfer dynamics are on the same time scale, the microengine system is a second-order system instead of a first-order system, like conventional engines;
- In the future we may need to be concerned with another dynamical element: the dynamics for emptying the fuel tank.

2.2 Nonlinear Model and its Simulator

This section derives the nonlinear model and the corresponding simulator for the microengine which will be utilized to estimate the engine performance. First the functional components are modeled, then the dynamics of microengine operation are analyzed, which includes the gas dynamics, the rotor acceleration dynamics and the heat transfer dynamics. The results of these analyses give us the nonlinear model of the microengine. Finally the simulator is developed.

The nonlinear model and its simulator are adapted from a code written by Vincent [12] and Ballin [2], for conventional engines. Besides the different component performance, the following two issues must be taken into consideration during the adaption:

- The heat transfer dynamics are strongly coupled with the rotor acceleration dynamics;
- The compressor performance is a function of the heat addition.

The system diagram is shown in Fig. 2-8. In the diagram, besides the functional components of the microengine, three infinitesimal control volumes are modeled between the compressor and the combustor, the combustor and the turbine, and between the turbine and the nozzle, in order to obtain a description of the gas dynamics. The notations used in the following analysis are consistent with this diagram.

2.2.1 Component Characteristics

As in conventional engines, the microengine has the following four functional components: compressor, combustor, turbine and nozzle. This section describes the model for each of these components.

Compressor

The microengine compressor works at high Mach number and low Reynolds number. The compressor pressure ratio and efficiency are effected by the heat flux from the turbine side. These working

conditions are quite different from conventional engines. Currently no empirical maps for the microengine compressor are available. But there are two maps for the macro-compressor, which also works at high Mach number and low Reynolds number. One of them comes from the ARO review [9]. Another one is the experimental map for the macro-compressor [3]. Although neither of them take heat flux effects into account, an estimated relationship between the compressor pressure ratio and the heat flux, as well as an estimated relationship between the compressor efficiency and the heat flux, are given as follows [9]:

$$\frac{\pi_c}{\pi_c^{no}} = 1 - k_{qpi} \times \dot{Q}_c,$$
$$\frac{\eta_c}{\eta_c^{no}} = 1 - k_{qef} \times \dot{Q}_c, \tag{2.9}$$

where $\dot{Q}_c$ is the heat flux into the compressor, π_c and η_c are the pressure ratio and efficiency with heat flux $\dot{Q}_c$, π_c^{no} and η_c^{no} are the pressure ratio and efficiency when there is no heat flux, and $k_{qpi} = 10.1084 \times 10^{-3}$ and $k_{qef} = 2.0175 \times 10^{-3}$ are two constants.

Based on the two maps for the macro-compressor and equation 2.9, four compressor maps are derived which will be utilized in the simulations: a hypothetical map, an adiabatic experimental map, a diabatic experimental map, and a modified diabatic experimental map. The first two maps don't consider the effects of the heat flux; the last two do. The following describes these four maps in detail.

Hypothetical Map - the First Compressor Map

The hypothetical map is the first compressor map. It comes from the macro-compressor map from the ARO review [9]. There the 42% and 100% speed lines are given. The other speed lines are obtained by extrapolation and assuming:

$$\frac{\Delta P_c}{N_p^2} = f(\frac{\dot{m}_2}{N_p}),$$

where ΔP_c is the pressure rise across the compressor, $N_p = N/N_{design}$ is the percentage of the rotation speed, $\dot{m}_2$ is the mass flow at the inlet of the compressor, function $f(.)$ is taken as a quadratic function whose coefficients are obtained by fitting the given data of the 42% and 100% speed lines. The coefficients are computed off line. Also, a fixed efficiency of 0.48 is assumed. The map is shown in Fig. 2-2.

Because this map is well formularized, it is used to test the simulation code and do some analysis.

Adiabatic Experimental Map - the Second Compressor Map

The adiabatic experimental map is the experimental map for the macro-compressor provided by [3], without any modifications. It doesn't consider the effects of the heat addition on the compressor

performance. It is shown in Fig. 2-3.

The purpose of this map is to estimate the effects of heat transfer to the engine performance by comparing the results when using this map with the results when using the third map, which utilizes the same data but adds the estimated effects of heat flux.

Diabatic Experimental Map - the Third Compressor Map

The diabatic experimental map is the combination of the experimental map for the macro-compressor (the second map, provided by [3]) and equation 2.9. It takes the effects of the heat flux into account. Thus it is a 3D map with the mass flow, pressure ratio and heat flux as its three dimensions. In practice the experimental data for the macro-compressor is taken as the compressor performance with 40W heat flux. The resulting 3D map is shown in Fig. 2-4. Fig. 2-5 shows an approximate 2D representation of the 3D map, with the speed lines with no heat flux as dashed lines and the speed lines with 40W heat flux as solid lines. The efficiency contours are those with 40W heat flux. Here 40W is chosen since the heat flux into the compressor at the microengine design point is around 40W. Henceforth these line types are the default line types, as listed in Table 2.1.

Table 2.1: Default Line Types for Lines with Heat Flux and without Heat Flux

	line type
lines with heat flux	solid lines
lines with no heat flux	dashed lines

As mentioned before, the microengine compressor works at high Mach number and low Reynolds number, with effects of heat flux on the compressor pressure ratio and efficiency. The macro-compressors work at high Mach number and low Reynolds number, and equations 2.9 describe the effects of heat flux. Thus the third map, the diabatic experimental map, is currently the closest representation of the real microengine compressor performance.

Modified Diabatic Experimental Map - the Fourth Compressor Map

Although the diabatic experimental map is currently the best approximation of the real microengine compressor performance, it takes a long time to run a single simulation. Thus the fourth map, the modified diabatic experimental map, is created to speed up the simulations without losing the basic characteristics described by the third map.

The basic idea to reduce the computation time is to replace the look-up-table operations by analytical function computations. The analytical functions are obtained by assuming ([9], [6]):

- for the speed lines,

$$\frac{\Delta P_c}{(N_p)^2} = a(\frac{\dot{m}_2}{(N_p)^z} - \dot{m}_0)^2 + \Delta P_0 \tag{2.10}$$

for all speed lines;

- for efficiency,

$$\frac{\Delta P_c}{(N_p)^2} = (e_1 \frac{\dot{m}_2}{N_p} + e_2)\eta_c. \tag{2.11}$$

where $N_p = N/N_{design}$, ΔP_c is the pressure rise across the compressor, z is a function of Reynolds number, ΔP_0 and $\dot{m}_0$ corresponds to the highest point of 100% speed line, a, e_1 and e_2 are constant coefficients.

In practice for speed lines, the 100% speed experimental data is used to obtain the constants ΔP_0, $\dot{m}_0$ and a, and z is computed by fitting the 80% and 60% experimental data to equation 2.10.

For efficiency, because of the relatively big measurement error during the experiment, the experimental efficiencies of the 60% speed line are not used. Similar to the approach used for the speed lines, coefficients e_1 and e_2 are obtained by linear fitting of the experimental data at 100% and 80% speed to equation 2.11.

The final results for the constants are:

$$\dot{m}_0 = 0.5153,$$

$$\Delta P_0 = 1.0768,$$

$$z = 1.41,$$

$$a = -1.6346,$$

$$e_1 = -2.4364,$$

$$e_2 = 3.8995.$$

This final modified diabatic experimental map is shown in Fig. 2-6, where the default line types are used (see Table 2.1). This map was used during the parameter study and the startup process simulations, which will be described in detail in chapter 4. The simulation time for a single simulation when using this map is about 1/10 of the simulation time when using the third map, the diabatic experimental map.

Combustor

The combustor efficiency is taken as a fixed value ([9]):

$$\eta_b = 0.9.$$

Energy conservation in the combustor is the same as in conventional engines, which is:

$$\dot{m}_f\, h\eta_b = \dot{m}_{41}\, C_{pt} T_{41} - \dot{m}_3\, C_{pc} T_3, \tag{2.12}$$

where h is the heat value of the fuel, $\dot{m}_f$ is the fuel mass flow, η_b is the combustor efficiency, $\dot{m}_{41}$ and $\dot{m}_3$ are the gas mass flow at the exit and the inlet of the combustor, C_{pt} and C_{pc} are the specific heat at constant pressure of the gas at the exit and the inlet of the combustor, and T_{41} and T_3 are the total temperature at the exit and the inlet of the combustor.

Turbine

The turbine map is obtained by extrapolation of the data at the designed point [9]. It is shown in Fig. 2-7. Here the effect of heat flux on the turbine performance is ignored, since it is expected to be very small [9].

Nozzle

Assuming an ideally expanded nozzle, the Mach number at the throat of the nozzle is related to the pressure ratio across the nozzle as:

$$1/\pi_n = (1 + \frac{\gamma - 1}{2} M_n^2)^{\frac{\gamma}{\gamma - 1}}.$$

where π_n is the pressure ratio of the nozzle, M_n is the Mach number. Using mass conservation, the nozzle is modeled as:

$$\dot{m}_{45} = \frac{P_{45}}{\sqrt{T_{45}}} A_n (1 + \frac{\gamma - 1}{2} M_n^2)^{-\frac{\gamma+1}{2(\gamma-1)}} \sqrt{\frac{\gamma}{R}} M_n, \tag{2.13}$$

where $A_n = 5.06mm^2$ is the nozzle area. This value is computed by using the values of π_n, $\dot{m}_{45}$, P_{45} and T_{45} at the designed point.

2.2.2 Modeling of the Dynamics

As stated in section 2.1, there are three main dynamical elements in the operation of the micro-engine: the gas dynamics, the rotor acceleration dynamics, and the heat transfer dynamics. The gas dynamics can be computed as quasi-steady values. This section models these dynamics. The combination of these dynamics models and the component models derived in the last subsection result in the nonlinear model for the microengine.

Gas Dynamics

As shown in the system diagram Fig. 2-8, first mass conservation is applied to the compressor, the combustor, and the turbine which yields the following equations:

$$\dot{m}_3 = \dot{m}_2 - \dot{m}_{bleed},$$

$$\dot{m}_{41} = \dot{m}_3 + \dot{m}_f,$$

$$\dot{m}_{45} = \dot{m}_{41} + \dot{m}_{bleed}, \tag{2.14}$$

where $\dot{m}_3$ and $\dot{m}_2$ are the gas mass flow at the exit and inlet of the compressor, $\dot{m}_{45}$ and $\dot{m}_{41}$ are the gas mass flow at the exit and inlet of the turbine, $\dot{m}_{bleed}$ is the bleed part of the compressor mass flow which is injected back at the exit of the turbine, and $\dot{m}_f$ is the fuel mass flow.
Another way to obtain the mass flow into the combustor, at the turbine inlet, and at the nozzle inlet is by the pressure and the temperature, which is as follows:

$$\dot{m}_{3-comb} = \sqrt{\frac{P_3(P_3 - P_{41})}{k_{dpb} T_3}},$$

$$\dot{m}_{41-gt} = f_t\left(\frac{P_{41}}{\sqrt{P_{45}}}\right),$$

$$\dot{m}_{45-pt} = f_n(M_n)\frac{P_{45}}{\sqrt{T_{45}}}, \tag{2.15}$$

where $f_t(.)$ comes from the turbine map, and $f_n((M_n))$ comes from the nozzle model (equation 2.13).

Next mass conservation is applied to each of the three infinitesimal control volumes shown in Fig. 2-8. In general, for each of these control volume, the mass conservation is as follows:

$$\frac{dm}{dt} = \dot{m}_{in} - \dot{m}_{out}\ . \tag{2.16}$$

where m is the mass inside the control volume, $\dot{m}_{in}$ and $\dot{m}_{out}$ are the mass flow into the control volume and out of the control volume, respectively. Inside the control volume, assuming a perfect gas yields:

$$P_s = \rho R T_s = \frac{m}{V} R T_s.$$

where V is the volume of the control volume, P_s and T_s are the static pressure and temperature of the gas inside the control volume, respectively. Taking the time derivative and rearranging, results in the following relation:

$$\frac{dm}{dt} = \frac{1}{k_v T}\frac{dP}{dt},$$

where k_v is a constant related to the volume.
Plugging this relationship into equation 2.16 yields the following:

$$\frac{dP}{dt} = k_v T(\dot{m}_{in} - \dot{m}_{out}).$$

Applying this formula to the three control volumes gives:

$$\begin{aligned}
\frac{dP_3}{dt} &= k_{v3} T_3(\dot{m}_3 - \dot{m}_{3-comb}), \\
\frac{dP_{41}}{dt} &= k_{v41} T_{41}(\dot{m}_{41} - \dot{m}_{41-gt}), \\
\frac{dP_{45}}{dt} &= k_{v45} T_{45}(\dot{m}_{45} - \dot{m}_{45-pt}).
\end{aligned} \tag{2.17}$$

Equations 2.17 actually describes the gas dynamics.

As stated in section 2.1, gas dynamics can be computed as quasi-steady values. This means that the following equations hold:

$$\begin{aligned}
\frac{dP_3}{dt} &\equiv 0, \\
\frac{dP_{41}}{dt} &\equiv 0, \\
\frac{dP_{45}}{dt} &\equiv 0.
\end{aligned}$$

Plugging equation 2.17 into them results in:

$$\begin{aligned}
\dot{m}_3 &= \dot{m}_{3-comb}, \\
\dot{m}_{41} &= \dot{m}_{41-gt}, \\
\dot{m}_{45} &= \dot{m}_{45-pt}\ .
\end{aligned} \tag{2.18}$$

The set of equations 2.18 is one of the main sets of equations in the nonlinear model.

Rotor Acceleration Dynamics

In the microengine, the drop of enthalpy across the turbine provides power, the compressor absorbs power, and there is also some power loss due to friction, etc., which is called shaft off-take power. So, the rotor acceleration dynamics can be described by the following equation:

$$\frac{dN}{dt} = \frac{1}{JN}(Power_t - Power_c - Power_{shaft}), \tag{2.19}$$

where J is the rotor inertia, N is the rotation speed, $Power_t$ is the power provided by the turbine, $Power_c$ is the power absorbed by the compressor, and $Power_{shaft}$ is the shaft off-take power. These

three powers can be represented as:

$$Power_c = \dot{m}_2 \, C_{pc} T_2^{abs} (\pi_c^{(\gamma_c - 1)/\gamma_c} - 1)/\eta_c,$$

$$Power_t = \dot{m}_{41} \, C_{pt} T_{41}^{abs} (1 - \pi_t^{(\gamma_t - 1)/\gamma_t}) \eta_t,$$

$$Power_{shaft} = 13W \times (\frac{N}{N_{des}})^2,$$

The first two equations come from the definitions of the compressor and turbine efficiencies, and 13W is the shaft off-take power at design point [1].

Equation 2.19 is another main equation in the nonlinear model.

Heat Transfer Dynamics

This subsection models the heat transfer dynamics. Because of the relatively high thermal conductivity of silicon and the small structural scale, the rotor of the microengine has nearly uniform temperature. In the heat flux path, the major sources and sinks are the fluid across the compressor and the turbine. Here uniform rotor temperature T_w is assumed and the heat sources and sinks other than the fluid across the compressor and the turbine are ignored. When analyzing the heat transfer dynamics, the rotor is taken as the object and the rotor temperature is taken as the state. Then the heat flux into the object (the rotor) is the convective heat flux from the turbine flow, $\dot{Q}_t$, and the heat flux out of the object is the convective heat flux into the compressor flow, $\dot{Q}_c$. Note that because the rotor is rotating, the relative temperature should be used instead of the absolute temperature when calculating $\dot{Q}_t$ and $\dot{Q}_c$. Let's first find the product of the convective heat transfer coefficient and the convective area at the turbine side and the compressor side, $(hA)_t$ and $(hA)_c$; next derive the relationship between the relative temperature and the absolute temperature; then get the heat flux $\dot{Q}_t$ and $\dot{Q}_c$; and finally, obtain the heat transfer dynamics.

Finding $(hA)_t$ and $(hA)_c$

The convective heat transfer coefficient can be expressed as ([9]):

$$h = \frac{K}{L} 0.664 Re_L^{1/2} Pr^{1/3}.$$

Thus with nearly the same Prandtl number and fixed area ([9]), we have:

$$(hA)_t \propto h_{turb} \propto \dot{m}_{41}{}^{1/2},$$

$$(hA)_c \propto h_{comp} \propto \dot{m}_2{}^{1/2}.$$

That is:

$$(hA)_t = k_{hAt}\, \dot{m}_{41}^{1/2},$$

$$(hA)_c = k_{hAc}\, \dot{m}_2^{1/2}, \tag{2.20}$$

where k_{hAc} and k_{hAt} are two constants.

[9] gives us $(hA)_t$ and $(hA)_c$ at the design point, which are as follows:

$$(hA)_t = 0.0852 W/K,$$

$$(hA)_c = 0.0942 W/K,$$

The designed mass flows are:

$$\dot{m}_{41} = 0.342 g/sec,$$

$$\dot{m}_2 = 0.36 g/sec.$$

Then the two constants can be obtained:

$$k_{hAc} = 4.9648,$$

$$k_{hAt} = 4.6071. \tag{2.21}$$

Equation 2.20 and 2.21 give us the formula to compute $(hA)_t$ and $(hA)_c$.

Finding the Relationship between the Relative Temperature and the Absolute Temperature

The relationship between the total temperature and the static temperature are as follows:

$$T_t^{abs} = T_s + \frac{(V^{abs})^2}{2C_p},$$

$$T_t^{rel} = T_s + \frac{(V^{rel})^2}{2C_p}, \tag{2.22}$$

where superscript "abs" means absolute value, superscript "rel" means relative value, subscript "t" means total, subscript "s" means static, T is the temperature, V is the gas speed, and C_p is the specific heat at constant pressure.

The rotation has no radial component, and its tangential component is ωR (Here ω denotes the rotation speed, and R denotes the radius). Thus:

$$V_{rad}^{abs} = V_{rad}^{rel},$$

$$V_{tan}^{abs} = \omega R - V_{tan}^{rel}.$$

After some manipulations and noting that

$$V_{tan}^{rel} = V_{rad}^{rel} tan\beta,$$

$$V_{rad}^{rel} \propto \dot{m},$$

where β is an angle determined by the blade shape.
the relationship between the absolute speed and the relative speed can be expressed as:

$$(V^{abs})^2 - (V^{rel})^2 = (\omega R)^2(1 - k_t \frac{\dot{m}}{\omega R}), \tag{2.23}$$

where k_t is a constant.
Plugging equation 2.23 into equation 2.22 results in the following:

$$\Delta T = T_t^{abs} - T_t^{rel} = \frac{(\omega R)^2}{2C_p}((1 - k_t \frac{\dot{m}}{\omega R}).$$

Applying this formula to both the inlet and the outlet of the compressor and the turbine yields the following equations:

$$\Delta T_2 \equiv T_2^{abs} - T_2^{rel} = \frac{(\omega R_2)^2}{2C_{pc}}(1 - k_{t2} \frac{\dot{m}_2}{\omega R_2}),$$

$$\Delta T_3 \equiv T_3^{abs} - T_3^{rel} = \frac{(\omega R_3)^2}{2C_{pc}}(1 - k_{t3} \frac{\dot{m}_3}{\omega R_3}),$$

$$\Delta T_{41} \equiv T_{41}^{abs} - T_{41}^{rel} = \frac{(\omega R_{41})^2}{2C_{pt}}(1 - k_{t41} \frac{\dot{m}_{41}}{\omega R_{41}}),$$

$$\Delta T_{45} \equiv T_{45}^{abs} - T_{45}^{rel} = \frac{(\omega R_{45})^2}{2C_{pt}}(1 - k_{t45} \frac{\dot{m}_{45}}{\omega R_{45}}), \tag{2.24}$$

where R_2, R_3, R_{41} and R_{45} are the radius of the inlet and the outlet of the compressor and the turbine, respectively.

The differences of the absolute and relative temperatures at designed point at both the inlet and the outlet of the compressor and the turbine are given by [9], which are as follows:

$$\Delta T_2 = -62K,$$

$$\Delta T_3 = -31K,$$

$$\Delta T_{41} = 107K,$$

$$\Delta T_{45} = -11K.$$

The radius of the inlet and the outlet of the compressor and the turbine are ([9]):

$$R_2 = 2.00mm,$$

$$R_3 = 4.00mm,$$

$$R_{41} = 2.52mm,$$

$$R_{45} = 1.50mm.$$

Combining the above data with the other parameters at the design point, the constant k_t at the different stations can be obtained as:

$$k_{t2} = 1.0373e6$$

$$k_{t3} = 0.8704e6$$

$$k_{t41} = -0.8999e6$$

$$k_{t45} = 0.5111e6 \tag{2.25}$$

Equation 2.24 and 2.25 give us the relationship between the relative temperatures and the absolute temperatures at both the inlet and the outlet of the compressor and the turbine.

Heat Flux into the Compressor and Heat Flux out of the Turbine

Precise values for the convective heat flux $\dot{Q}_c$ and $\dot{Q}_t$ would be found by integrating the heat flux of each infinitesimal piece along the compressor and the turbine, respectively. Here for simplicity the "characteristic temperatures" T^{rel}_{comp} and T^{rel}_{turb} are used to calculate $\dot{Q}_c$ and $\dot{Q}_t$, i.e.:

$$\dot{Q}_c = (hA)_c(T_w - T^{rel}_{comp}),$$

$$\dot{Q}_t = (hA)_t(T^{rel}_{turb} - T_w). \tag{2.26}$$

The characteristic temperature T^{rel}_{comp} is taken as the average of the relative temperatures at the inlet and the outlet of the compressor, and the characteristic temperature T^{rel}_{turb} is taken as the average of the relative temperatures at the inlet and the outlet of the turbine, as follows:

$$T^{rel}_{comp} = (T^{rel}_2 + T^{rel}_3)/2,$$

$$T^{rel}_{turb} = (T^{rel}_{41} + T^{rel}_{45})/2. \tag{2.27}$$

Using the differences between the absolute temperatures and the relative temperatures to represent

equation 2.27 yields the following:

$$T_{comp}^{rel} = \frac{1}{2}[(T_2^{abs} + T_3^{abs}) - (\Delta T_2 + \Delta T_3)],$$

$$T_{turb}^{rel} = \frac{1}{2}[(T_{41}^{abs} + T_{45}^{abs}) - (\Delta T_{41} + \Delta T_{45})]. \tag{2.28}$$

Equations 2.26 and 2.28 show that $\dot{Q}_c$ and $\dot{Q}_t$ are functions of $(hA)_c$, $(hA)_t$ and absolute temperatures. Equations 2.20 shows that $(hA)_c$ and $(hA)_t$ are functions of mass flow and rotation speed. Equations 2.24 show that the differences between the absolute temperatures and the relative temperatures are also functions of mass flow and rotation speed. Thus $\dot{Q}_c$ and $\dot{Q}_t$ are functions of mass flow, absolute temperatures and rotation speed. In addition, compressor mass flow is a function of $\dot{Q}_c$ (the compressor map is a 3D map with $\dot{Q}_c$ as one of its dimensions). Absolute temperatures are also functions of $\dot{Q}_c$ and $\dot{Q}_t$, as follows:

$$\dot{m}_2\ C_{pc}(T_3^{abs} - T_2^{abs}) = W_{orkc} + \dot{Q}_c,$$

$$\dot{m}_{41}\ C_{pt}(T_{41}^{abs} - T_{45}^{abs}) = W_{orkt} + \dot{Q}_t\ . \tag{2.29}$$

where W_{orkt} and W_{orkc} are adiabatic work of the compressor and the turbine:

$$W_{orkc} = \dot{m}_2\ C_{pc} T_2^{abs} (\pi_c^{(\gamma_c - 1)/\gamma_c} - 1)/\eta_c,$$

$$W_{orkt} = \dot{m}_{41}\ C_{pt} T_{41}^{abs} (1 - \pi_t^{(\gamma_t - 1)/\gamma_t})\eta_t. \tag{2.30}$$

Thus these equations (equation 2.20, 2.24, 2.26, 2.28, 2.29, and 2.30), along with the compressor map, are highly coupled nonlinear equations. Solving these equations simultaneously can give us $\dot{Q}_c$ and $\dot{Q}_t$.

Heat Transfer Dynamics

Knowing $\dot{Q}_c$ and $\dot{Q}_t$, the heat transfer dynamics can be found. The net heat flux flowing into the rotor is:

$$\dot{Q} = \dot{Q}_t - \dot{Q}_c\ . \tag{2.31}$$

Applying energy conservation to the rotor results in:

$$\dot{Q} = C_w M_w \frac{dT_w}{dt}, \tag{2.32}$$

where C_w and M_w are the specific heat and the mass of the rotor, respectively.

Combining equation 2.31 and equation 2.32 yields the following:

$$\frac{dT_w}{dt} = \frac{\dot{Q}_t - \dot{Q}_c}{C_w M_w}. \tag{2.33}$$

This is the equation which describes the dynamics of the state T_w, i.e. the heat transfer dynamics.

2.2.3 Nonlinear Model Simulator

The above two subsections give us the nonlinear model, which is mainly described by equations 2.18, 2.19 and 2.33, combined with the component models and equations 2.14, 2.15, 2.20, 2.24, 2.26, 2.28, 2.29, and 2.30. This subsection describes the nonlinear model simulator.

The diagram of the simulator is shown in Fig. 2-9. As shown in the figure, the inputs are the fuel flow $\dot{m}_f$, the simulation parameters like the error tolerance, and the initial states of the engine. The outputs are the engine states at every time step. The integration of the rotor speed for the unbalanced powers and the rotor temperature for the unbalanced heat flux (correspond to equation 2.19 and 2.33) are solved in the "N and T_w update" block. Thus the "engine cycle" block is static; i.e., it does not include dynamical states. The task of the "engine cycle" block is to generate the quasi-static solutions for the gas dynamics (equations 2.18, 2.14 and 2.15) and the heat flux $\dot{Q}_c$ and $\dot{Q}_t$ (equations 2.20, 2.24, 2.26, 2.28, 2.29, and 2.30). Here a Lipschitz numerical approach is used to solve these highly coupled and nonlinear equations.

In practice, the simulator is accomplished by Simulink in MATLAB [11]. The "engine cycle" block is accomplished by an S-function [11] written in C. The code for the S-function is attached in the Appendix. Since the "engine-cycle" block is a little complicated, the following says a little more about the operations in this block.

"Engine Cycle" Block

As stated before, the quasi-static solutions for the gas dynamics (equations 2.18, 2.14 and 2.15) and the heat flux $\dot{Q}_c$ and $\dot{Q}_t$ (equations 2.20, 2.24, 2.26, 2.28, 2.29, and 2.30) are generated in the "engine cycle" block. The Lipschitz numerical approach is used to solve these highly coupled and nonlinear equations. The follows first describes how the Lipschitz numerical approach works, then applies it to the problem, and finally describes the steps in the "engine cycle" block.

Basically, the Lipschitz approach is a recursive solution procedure. In this approach one first sets initial values for the unknowns. The next estimates for the unknowns are modifications of the previous estimates. In general, assume the equation to be solved is:

$$x = f(x).$$

If the previous estimate is x_{n-1}, the next estimate is computed as follows:

$$x_n = x_{n-1} + k_{update}(f(x_{n-1}) - x_{n-1}),$$

where k_{update} is a constant which can be adjusted to get fast and stable convergence.

In our problem, it can be shown that when P_3, P_{41}, P_{45} are taken as the unknowns, this approach converges when applied to the gas dynamics, i.e. equation 2.18, equation 2.14 and equation 2.15. The modifications for P_3, P_{41}, P_{45} that should be applied at each iteration are computed as follows:

$$\Delta P_3 = \frac{P_{41} + \sqrt{{P_{41}}^2 + 4k_{dpb}T_3\,\dot{m}_{31}^2}}{2} - P_3,$$

$$\Delta P_{41} = P_3 - \frac{k_{dpb}T_3(\dot{m}_{41-gt} - \dot{m}_f)^2}{P_3} - P_{41},$$

$$\Delta P_{45} = \frac{\dot{m}_{45}\sqrt{T_{45}}}{f_n(M_n)} - P_{45}. \tag{2.34}$$

Once P_3, P_{41} and P_{45} are solved, all other variables, including the mass flows and temperatures, are solved simultaneously.

The Lipschitz approach is also used to solve equations 2.20, 2.24, 2.26, 2.28, 2.29, and 2.30, for the heat flux $\dot{Q}_c$ and $\dot{Q}_t$. Here the details are ignored.

Below are the steps in the "engine cycle" block:

1. Accept the inputs which includes $\dot{m}_f$, the simulation parameters, and the initial parameters of the engine, especially N, T_w and P_3, P_{41}, P_{45};

2. Apply Lipschitz approach to equations 2.20, 2.24, 2.26, 2.28, 2.29, and 2.30, and solve for $\dot{Q}_c$, at the same time obtain the corresponding $\dot{m}_3$, η_c, τ_c, and T_3;

3. Get T_{41} from the combustor model;

4. Similar to step 2, solve equations 2.20, 2.24, 2.26, 2.28, 2.29, and 2.30, for $\dot{Q}_t$, τ_t, η_t, and T_{45};

5. Plug in all the above data to the ΔP_3, ΔP_{41}, and ΔP_{45} (equations 2.34) and get the next estimate for P_3, P_{41}, P_{45};

6. Check if $|\Delta P_i - \Delta P_{iprev}| < error$, where $i = 3, 41, 45$ and $error$ is the error tolerance. If satisfied, this set of data is the quasi-steady data for the gas dynamics; otherwise go back to step 2;

7. Compute $\frac{dN}{dt}$ and $\frac{dT_w}{dt}$ by equation 2.19 and equation 2.33, which are ready to be integrated in the next block of Fig 2-9.

One thing that should be pointed out is that, in order to do some analysis and comparisons of the engine behavior with and without heat transfer, a nonlinear model for the microengine without

the heat transfer effects has also been developed. The difference between this model and the model derived in detail here is that the no heat transfer model doesn't have the heat transfer dynamics, and the compressor map is the first or second compressor map, which doesn't consider the heat flux effects to the compressor performance. Since it is a subset of the model derived in detail here, the details of this simpler model are ignored.

2.3 Linearized Model and its Simulator

This section derives the linearized model and the corresponding simulators for the microengine. First general aspects about the linearized models are presented; then the linearized model for the microengine are derived; finally the simulator is described.

2.3.1 General Aspects of the Linearized Models

The general assumption for linearization of a nonlinear system to make a linearized model that reasonably represents the real system is: the system behavior doesn't change too much in the neighborhood of the linearized equilibrium point. Because of the continuum behavior of many physical systems, this assumption holds for many practical systems. The linearized model is valid for approximate analysis and design of the real system within the neighborhood. The valid range of the neighborhood depends on the tolerable error.

The general form for linearized models is:

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

where x is the state vector, u is the control vector, y is the output, and A, B, C and D are coefficient matrices.

2.3.2 Linearized Model and Simulator for the Microengine

The linearized model for the microengine is derived around the steady operating points in order to allow use of the large body of results that exist for analyzing the properties of linear systems.

As stated before, the microengine is a second-order system. The two main dynamics are the rotor acceleration dynamics and the heat transfer dynamics. So the rotation speed and the spool temperature are chosen as the two states of the linearized model for the microengine. The control vector here is just one variable, which is the fuel mass flow $\dot{m}_f$. For the y vector, anything which are of interest can be included into the y vector, such as the pressures and the temperatures. Now the problem is to find the matrices A, B, C and D.

The general approach to get these matrices is to apply first-order multi-variable Taylor's series. Here no analytical form for the microengine system behavior is available. So a practical way is applied to obtain these matrices, based on the nonlinear model simulation. The idea is to use small changes to approximate the differentials, and the small changes are generated by perturbation inputs. The steps are as follows:

1. Choose the equilibrium point to be linearized;
2. Get the steady states x_0, y_0 and fuel flow $\dot{m}_{f_0}$ at the linearization point from the nonlinear model;
3. Choose these steady states to be the initial states. Input $\dot{m}_f = \dot{m}_{f_0} + \Delta\, \dot{m}_f$, where $\Delta\, \dot{m}_f$ is a small increase of $\dot{m}_{f_0}$. Run the simulation on the nonlinear model as described in section 2.2, without the integrations, and record the new x and y as x_+ and y_+;
4. Repeat step 3, but change the input to be $\dot{m}_f = \dot{m}_{f_0} - \Delta\, \dot{m}_f$. Record the new x and y as x_- and y_-;
5. Compute $\Delta x = x_+ - x_-$, $\Delta y = y_+ - y_-$;
6. Divide each row of Δx and Δy by $2\Delta\, \dot{m}_f$. The results are the corresponding row of matrices B and D;
7. Now change one of the initial states, let's say x_i, for nonlinear model simulation to be $x_i = x_{i0} + \Delta x_i$, and keep the others unchanged. Use $\dot{m}_{f_0}$ as the input. Run the nonlinear model again without integration, and record the new x and y as x_+ and y_+, as in step 3;
8. Repeat step 7, but change $x_i = x_{i0} - \Delta x_i$. Record the new x and y as x_- and y_-;
9. Compute $\Delta x = x_+ - x_-$, $\Delta y = y_+ - y_-$, as in step 5;
10. Divide each row of Δx and Δy by $2\Delta x_i$. The results are the corresponding row of matrices A and C;
11. Repeat step 7 to 10 for each x_i to get all of the elements of the matrices A and C.

The matrices A, B, C and D found here give us the model linearized around the point x_0, y_0. From the linearized assumption, the approximate behavior of the engine in the neighborhood of this point can be obtained using this linearized model. If the linear properties of the engine are good, this region can be very large, even the entire operation range. If the linear properties of the engine are poor, several linearized models around several points may be needed to approximate the system behavior. If the linear properties are very poor, the nonlinear model may have to be used to get the

system behavior. Then the large body of good properties of linear systems and analysis and design tools are no longer able to be used.

Simulink and MATLAB provide standard state-space model simulation tools [11]. They are taken as the simulator for the linearized models.

2.4 Summary

This chapter investigates the dynamics of the primary physical processes in the operation of the microengine, and determines which processes are important and should be incorporated in the model. The modeling procedures and the corresponding simulators are also described, including both the nonlinear model and the linearized model.

Analysis of the dynamics indicates that the rotor acceleration dynamics and the heat transfer dynamics have the same time scale. But the gas dynamics are much faster than these two. Thus the gas dynamics can be computed as quasi-steady values. Hence the microengine system is close to a second-order system, composed mainly of the rotor acceleration dynamics and the heat transfer dynamics. A conventional engine, on the other hand, is a first-order system, which contains only the rotor acceleration dynamics. Heat transfer thus plays a unique and important role in the dynamical behavior of the microengine.

Heat transfer not only affects the dynamical behavior of the microengine, it also affects the performance of the components, mainly the compressor. It reduces both the compressor pressure ratio and the efficiency. This is another aspect of the microengine that is different from a conventional engine.

The nonlinear model of the microengine consists of component models and dynamical models, described by a set of highly nonlinear, coupled equations. In order to do some analysis and comparisons, two kinds of nonlinear models will be used in the following chapters. One model considers the effects of heat transfer, while the other doesn't. Here the nonlinear model of the former kind is described in detail. The latter no heat transfer nonlinear model is a subset of the full model. Corresponding to these models there are four compressor maps: the hypothetical compressor map (no heat addition effects), the adiabatic experimental map, the diabatic experimental map, and the modified diabatic experimental map. The first two don't consider the effects of heat flux on compressor behavior; the last two do. Despite the differences between the various compressor maps, they all consider the special microengine compressor operating characteristics, that is, high Mach number and low Reynolds number.

The linearized model is obtained by applying small perturbation inputs to the nonlinear model. It will be used to analyze the behavior of the system, especially the transients. It will also be useful for future control design.

The simulators for both the nonlinear model and the linearized models are accomplished using Simulink in MATLAB.

The chapters that follow will give the simulation results based on both the nonlinear model and linearized models. These results give us a picture of how the engine works. They are also the basis for future control design. Hopefully they will also provide insights on how to modify the design of the engine in the future.

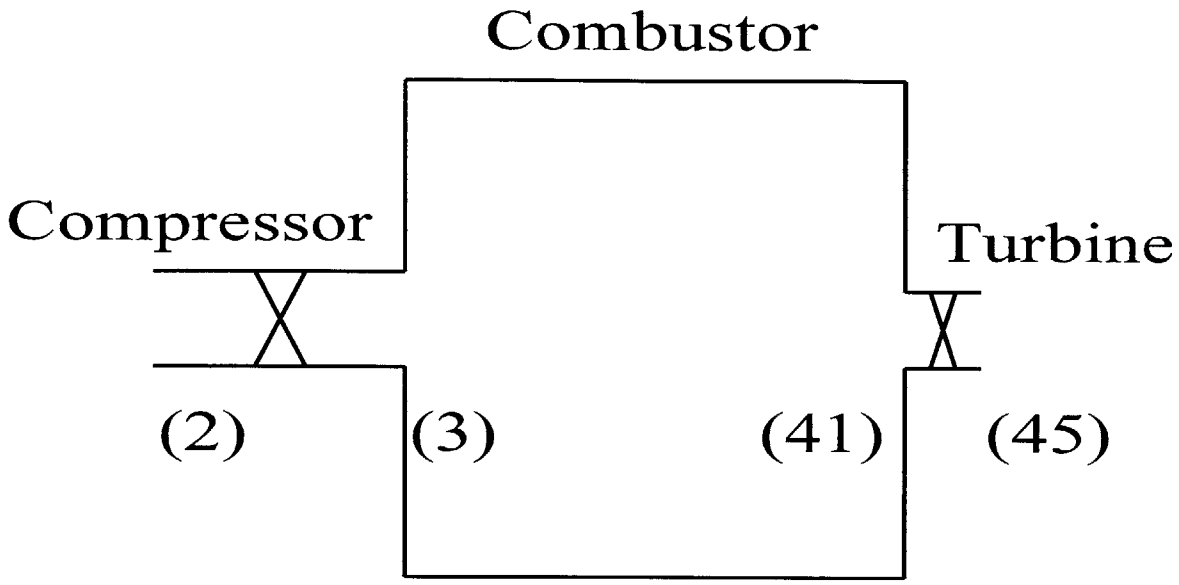


Figure 2-1: Helmotz Resonator Model for the Microengine

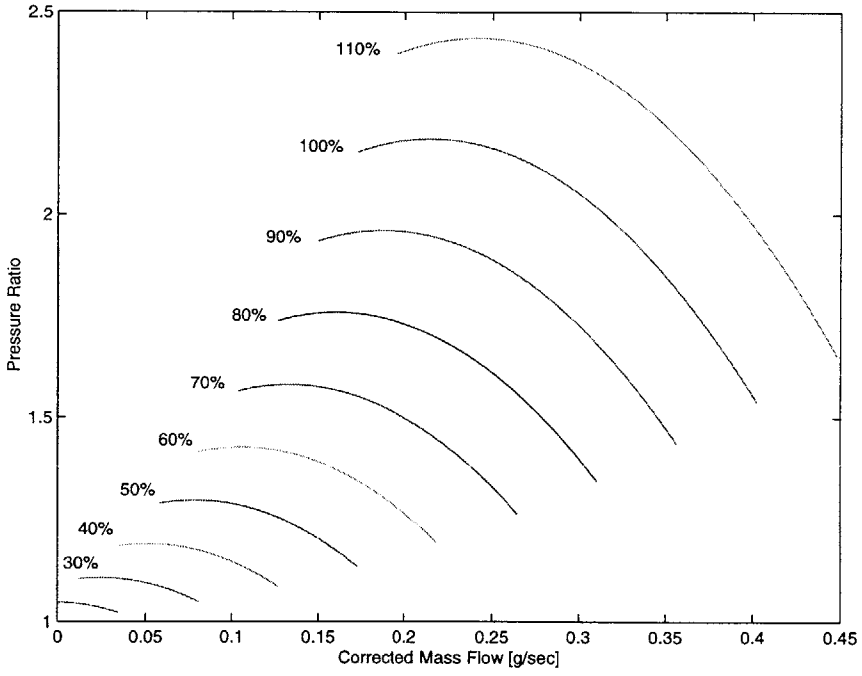


Figure 2-2: Hypothetical Compressor Map

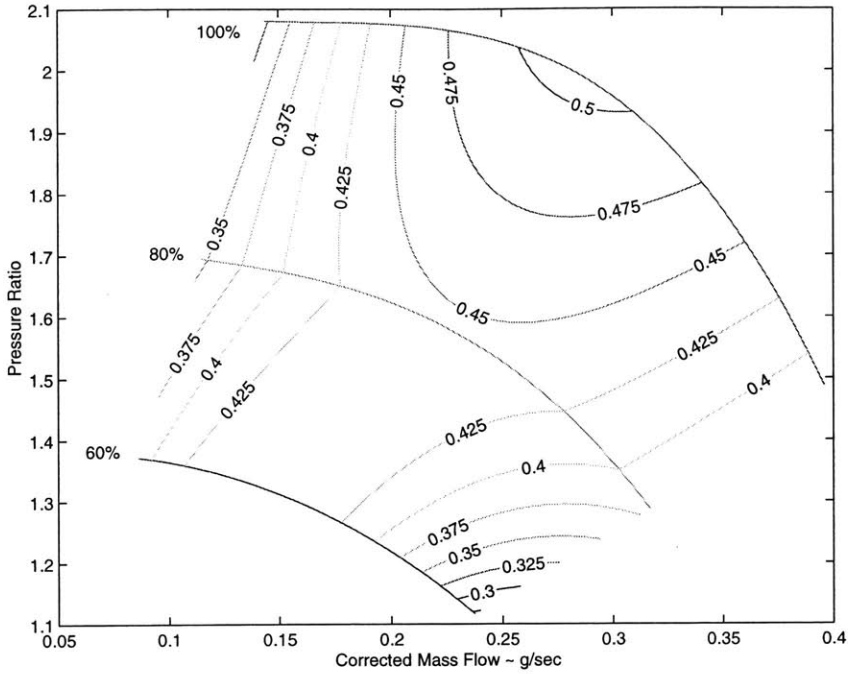


Figure 2-3: Adiabatic Experimental Compressor Map

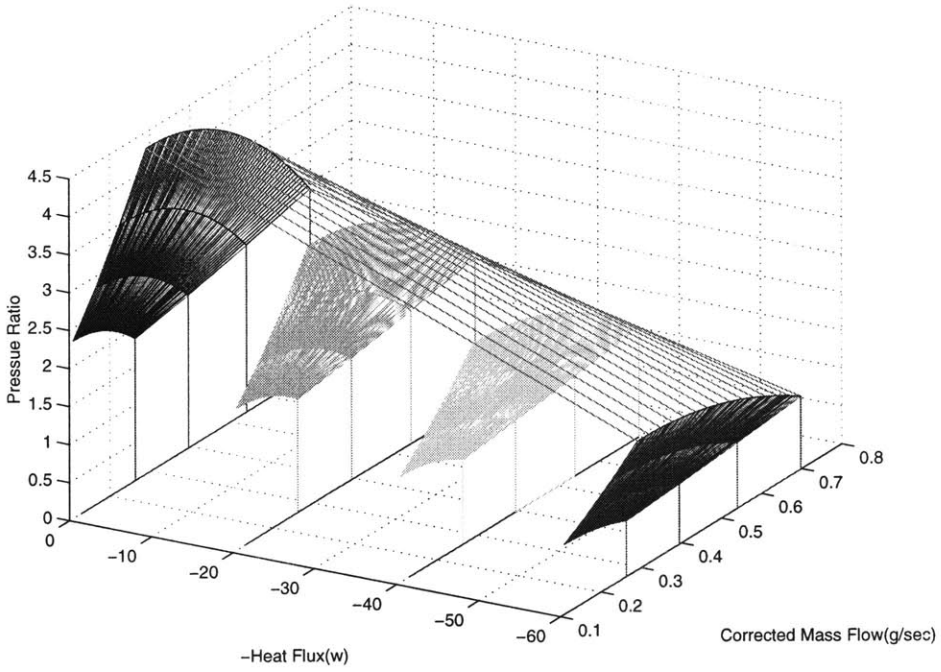


Figure 2-4: Diabatic Experimental Compressor Map - 3D

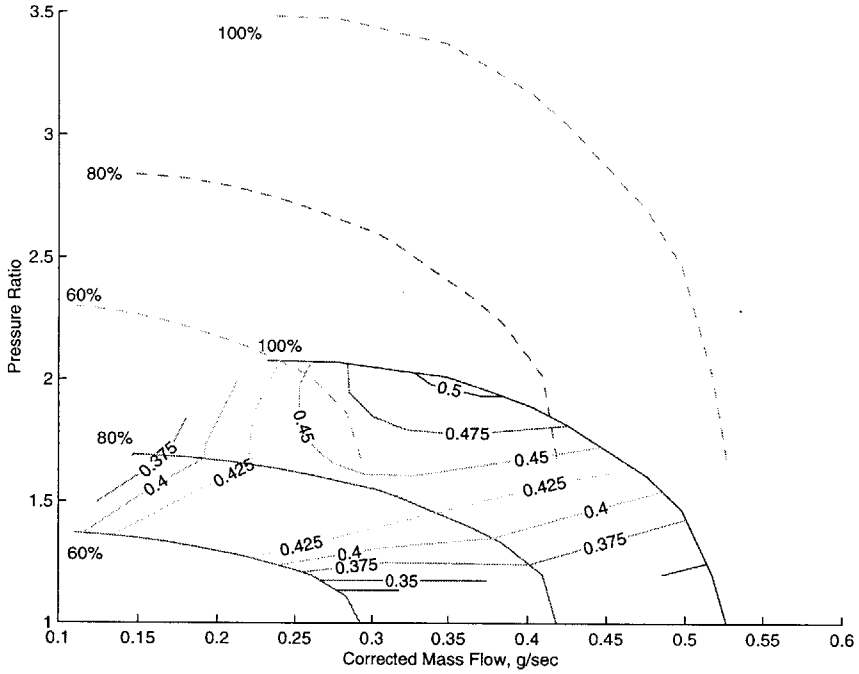


Figure 2-5: Diabatic Experimental Compressor Map - 2D Representation

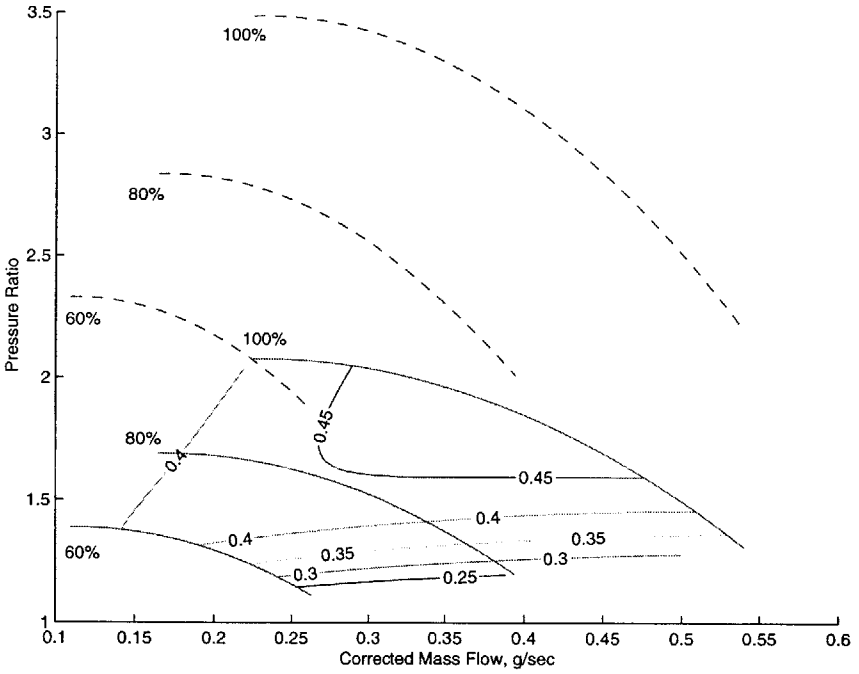


Figure 2-6: Modified Diabatic Experimental Compressor Map with Heat Flux - Used for Parameter Study

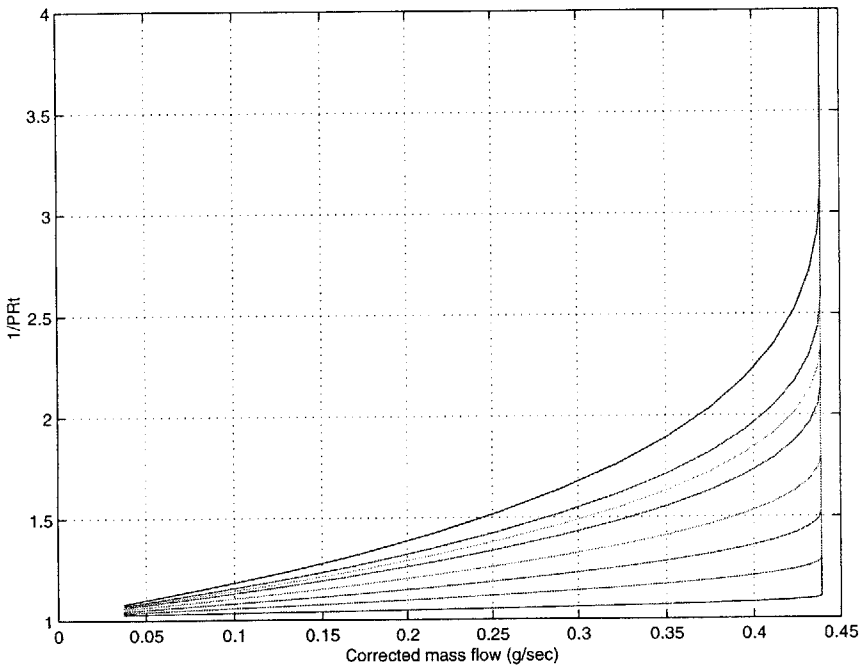


Figure 2-7: Turbine Map

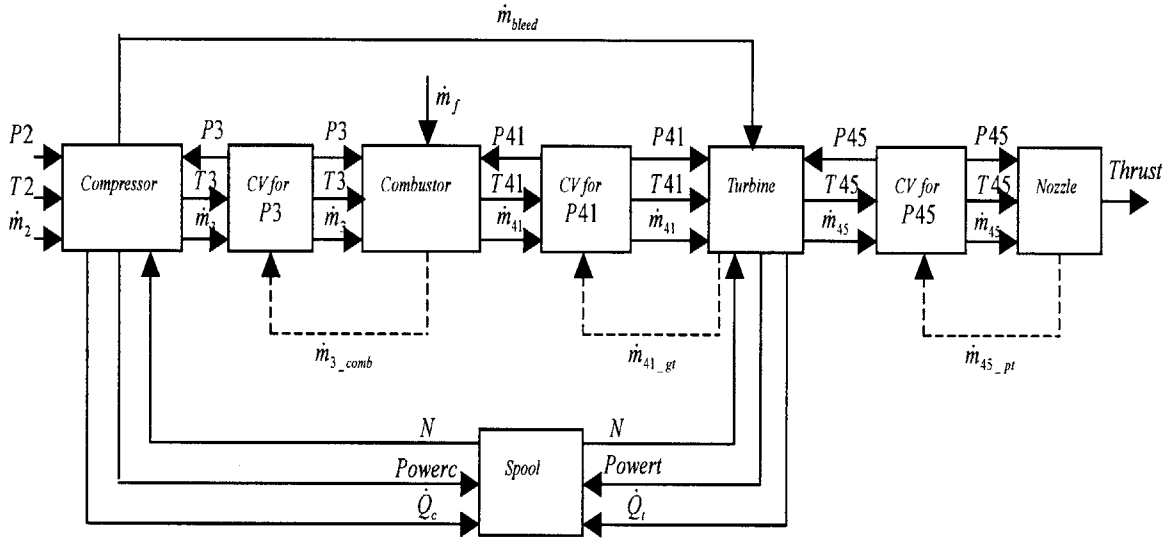


Figure 2-8: Engine Cycle Model Structure

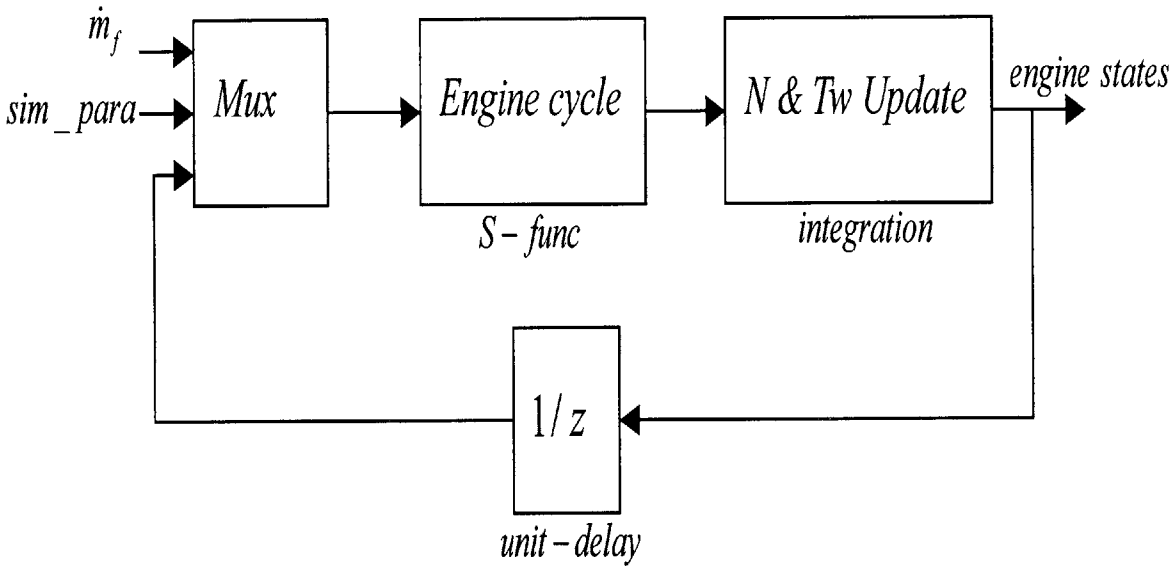


Figure 2-9: Simulation Signal Flow Diagram

(sim_para: simulation parameters, such as the error tolerance)

Chapter 3

Simulation Results

Based on the models and the simulators derived in Chapter 2, simulations are performed for the microengine behaviors. The behaviors simulated include: minimum achievable speed and steady states, linear properties, transient properties and sensor options. This chapter first describes the objects on which these simulations are performed, then gives the simulation results and explanations of the behavior, and finally summarizes the results and analysis. Simulations for the sensitivity analysis, as well as the startup process, are in Chapter 4.

3.1 Simulation Objects

The main objective of the simulations is to predict the performance of the microengine, which has the characteristics of small size, low and diabatic component performance, and severe heat transfer. Thus three simulation objects are chosen according to the different combinations of the compressor maps and the considerations of the heat transfer :

1. Object 1: using hypothetical compressor map (the first compressor map) and ignoring the heat transfer dynamics;
2. Object 2: using adiabatic experimental compressor map (the second compressor map) and ignoring the heat transfer dynamics;
3. Object 3: using diabatic experimental compressor map (the third compressor map) and considering the heat transfer dynamics.

The results of simulations on object 3 are the best to predict the microengine behavior, since its compressor map is currently the closest representation of the real microengine compressor performance, and it considers heat transfer dynamics as well. The purpose of simulating on object 2 is to estimate the heat transfer effects by comparing the results with the simulation results on

object 3, since the only difference between object 2 and object 3 is that object 2 ignores all the heat transfer effects, both its effects on the compressor performance and its dynamics. Because its compressor map is well formularized, object 1 is used to do some basic analysis. It is also used to further strengthen the comparisons between the simulation results of object 2 and 3.

The following sections present the simulation results on these three objects considering the microengine behaviors concerned. Throughout this chapter the approximate 2D representation of the 3D map, with both the speed lines with no heat flux and the speed lines with 40W heat flux on the map, is used for illustrating the simulation results on object 3.

3.2 Minimum Achievable Speed and Steady States - T_{41} Behavior with $\dot{m}_f$

The minimum achievable speed is obtained by gradually decreasing the input of the nonlinear model derived in chapter 2, the fuel mass flow, until the microengine cannot keep stable operation. For the three different objects, the minimum achievable speeds are all around 60%.

The steady states are obtained by gradually changing the input fuel mass flow and for each input recording the steady state parameters such as the speed, the pressures, the temperatures and other parameters concerned. They are characterized by the operating line on the compressor map, as well as the equilibrium maps with the fuel flow as the abscissa and the speed, the pressures, the temperatures and other parameters concerned as the ordinate. The operating lines are shown in Fig 3-1, and the equilibrium maps are shown in Fig 3-2, with a) for object 1, b) for object 2 and c) for object 3.

One interesting behavior shown in Fig. 3-2 a), b) and c) is that for different objects there is the possibility of increasing T_{41} with decreasing $\dot{m}_f$ in some regions, while in conventional engines, T_{41} usually decreases with $\dot{m}_f$ monotonously. This behavior is further discussed in the following subsections.

3.2.1 Mathematical and Physical Explanations of the T_{41} Behavior with $\dot{m}_f$

This unusual T_{41} behavior with $\dot{m}_f$ can be explained in both mathematical and physical sense.

To explain mathematically why T_{41} increases with decreasing $\dot{m}_f$ in some regions, first consider the combustor equation, which relates $\dot{m}_f$, $\dot{m}_3$, $\dot{m}_{41}$, T_{41} and T_3 together (equation 2.12):

$$\dot{m}_f \, h\eta_b = \dot{m}_{41} \, C_{pt} T_{41} - \dot{m}_3 \, C_{pc} T_3.$$

This equation can be rearranged to be:

$$T_{41} = \frac{1}{1+f}\frac{C_{pc}}{C_{pt}}T_3 + \frac{f}{1+f}\frac{h\eta_b}{C_{pt}}, \tag{3.1}$$

where $f \equiv \frac{\dot{m}_f}{\dot{m}_3} \ll 1$ is defined to be the fuel-to-air ratio.

Equation 3.1 means that both T_3 and the fuel-to-air ratio f (but not the fuel flow $\dot{m}_f$ itself) contribute to T_{41}. Here $f \ll 1$ and $1 + f \approx 1$. Because of the large value of h (for the microengine $h = 120e6$), usually the second term dominates T_{41}. So, it is the ratio of fuel-to-air mass flow that determines T_{41}, but not the fuel mass flow itself. With decreasing of $\dot{m}_f$, the engine rotation speed decreases. The compressor sucks in less air flow, i.e. $\dot{m}_3$ decreases. Usually in conventional engines, the percentage decrease of $\dot{m}_f$ is more than the percentage decrease of $\dot{m}_3$, thus f decreases with decreasing $\dot{m}_f$. Hence from equation 3.1, T_{41} decreases with decreasing $\dot{m}_f$. But for the microengine, the percentage decrease of $\dot{m}_f$ is less than the percentage decrease of $\dot{m}_3$ in some regions, thus f increases with decreasing of $\dot{m}_f$. Hence from equation 3.1, T_{41} increases with decreasing $\dot{m}_f$. This leads to the unusual T_{41} trends with $\dot{m}_f$.

A simple physical explanation can be used to expand on the above mathematical one. As mentioned before, decreasing fuel mass flow leads to decreasing of the rotation speed, which in turn leads to decreasing of air mass flow $\dot{m}_3$. For the microengine in some regions, the percentage decrease of $\dot{m}_f$ is less than the percentage decrease of $\dot{m}_3$. This means that the same unit of $\dot{m}_f$ needs to heat less air mass flow. Thus the temperature rise is higher, and T_{41} increases.

3.2.2 Simulation Verification of the T_{41} Behavior with $\dot{m}_f$

In order to support the above explanations, some data analysis is performed and some simulations are designed on object 1.

First the data analysis. Some of the simulation data at steady states for object 1 are listed in Table 3.1. The data in Table 3.1 shows that in the regions of increasing T_{41} with decreasing $\dot{m}_f$, the fuel-to-air mass flow ratio does increase with decreasing $\dot{m}_f$. This is consistent with the former explanations.

Table 3.1: Data Analysis for T_{41} Trends with $\dot{m}_f$

$\dot{m}_f$ (g/hr)	$N(\%)$	$\dot{m}_3$ (g/sec)	f	$T_3(K)$	$T_{41}(K)$
1.0304000e-002	5.7881616e+001	1.6295985e-001	1.7563972e-002	3.4737506e+002	1.5993400e+003
1.0672000e-002	5.9670445e+001	1.7191510e-001	1.7243653e-002	3.5073117e+002	1.5776015e+003
1.1040000e-002	6.1882773e+001	1.7969662e-001	1.7065800e-002	3.5426898e+002	1.5667239e+003
1.2880000e-002	7.2388389e+001	2.1769961e-001	1.6434470e-002	3.7258626e+002	1.5322049e+003
1.4720000e-002	8.2145307e+001	2.5322382e-001	1.6147331e-002	3.9108053e+002	1.5237712e+003
1.6560000e-002	9.1302271e+001	2.8683476e-001	1.6037108e-002	4.0968331e+002	1.5287737e+003
1.8400000e-002	9.9983023e+001	3.1834921e-001	1.6055046e-002	4.2791513e+002	1.5431897e+003
1.9320000e-002	1.0410450e+002	3.3322666e-001	1.6105154e-002	4.3653801e+002	1.5531501e+003

Next the simulations designed to verify the explanations. The idea is: for the same decreasing

rate of $\dot{m}_f$, if we can find a way to change the decreasing rate of $\dot{m}_3$, we can change the trends of the second term of equation 3.1, and thus the T_{41} trends. This idea can be accomplished by changing the power loss, the shaft-off-take power. If the shaft-off-take power is increased, for the same $\dot{m}_f$, the rotation speed is less, the compressor sucks in less air, and $\dot{m}_3$ is less. This results in different f, and thus T_{41}. The simulations are performed on objects 1 with $N \leq N_{des}$ in the following cases :

1. turbine map, $Power_{shaft} = 13W = constant$;
2. turbine map, $Power_{shaft} = 13W \times (\frac{N}{N_{des}})^2$ (original case);
3. turbine map, $Power_{shaft} = 0W$ (no loss);

   Simulations are also performed in which the turbine is changed to be a choked turbine:

4. choked turbine, $Power_{shaft} = 13W = constant$;
5. choked turbine, $Power_{shaft} = 13W \times (\frac{N}{N_{des}})^2$.

Obviously, for object 1 in cases 1, 2 and 3, the loss (shaft-off-take power) is highest in case 1, less in case 2, and least in case 3. From the above statement, the order of increasing rate of T_{41} with decreasing of $\dot{m}_f$ should be highest in case 1, middle in case 2, and least in case 3. For the choked turbine cases 4 and 5, the loss is more in case 4 and less in case 5. The order of increasing rate of T_{41} with decreasing of $\dot{m}_f$ should be more in case 4, and less in case 5. The results are shown in Fig 3-3, which is just the same as indicated above.

3.3 Linear Properties

3.3.1 Simulation Descriptions

Linearized models around steady operating points are derived for object 1, object 2 and object 3, respectively. The linear properties of these three objects are evaluated by comparing the steady states and transient responses of their linearized models against those generated by their nonlinear models.

For object 1, linearization is performed at three steady operating points: the steady operating points corresponding to 100%, 85% and 65% $\dot{m}_{f_{des}}$. A doublet fuel flow is the input for each linearized model and the nonlinear model for comparisons. The simulation results of the comparisons of the steady states are shown in Fig. 3-4 (a), and the time domain transients are shown in Fig. 3-5 (a). The comparisons of the transients on the compressor map are shown in Fig. 3-6 (a). Here and henceforth default line types are used for the comparisons, which are listed in Table 3.2.

For object 2, linearization is performed also at three steady operating points: the steady operating points corresponding to 100%, 85% and 70%$\dot{m}_{f_{des}}$. A doublet fuel flow is also chosen to be the input

Table 3.2: Default Line Types for Nonlinear Model and Linearized Model

	line type
responses of nonlinear model	solid lines
responses of linearized model	dashed lines
running line	dashdot lines

for each comparison. The simulation results of the comparisons of the steady states are shown in Fig. 3-4 (b), and the time domain transients are shown in Fig. 3-5 (b). The comparisons of the transients on the compressor map are shown in Fig. 3-6 (b). Note the default line types used here.

For object 3, a linearization is performed for steady points corresponding to 1 $\%\dot{m}_{f_{des}}$ input steps. Of course in reality this is impossible. Here the reason why we do this is just for analysis. A step fuel flow of 1 $\%\dot{m}_{f_{des}}$ is the input for each comparison. The simulation results of the comparisons of the steady states are shown in Fig. 3-4 (c), and the time domain transients are shown in Fig. 3-5 (c). The positions of the eigenvalues of these linearized models are shown in Fig. 3-7, with indications of the corresponding fuel flow for several points.

3.3.2 Simulation Results and Analysis

This subsection examines the linear properties of the three objects based on the simulation results.

Object 1 and object 2 ignore the effects of heat transfer. Thus they are close to first-order systems, as stated in section 2.1. Fig. 3-4 and Fig. 3-5 (a) and (b) show the comparisons of the steady states and time domain transients for their linearized models and nonlinear models, respectively. It can be seen from these figures that the linear response is nearly on the top of the nonlinear model response, for both the steady states and the transients. The linear properties of object 1 and object 2 are quite good. The comparisons of the transients on their compressor maps shown in Fig. 3-6 (a) and (b) gives the same results.

But for object 3, which considers the heat transfer effects and thus is close to a second-order system, it is not the case. Fig. 3-4 c) gives the comparisons of the steady states for the linearized models and nonlinear model. It shows that for the steady states, the linearized model response is nearly on the top of the nonlinear model response. This is because the linearization points are very close together(corresponding to 1 $\%\dot{m}_{f_{des}}$ steps), and thus the plot scale is too big to show the differences. Fig. 3-5 c) gives a better scale for the comparisons. It gives typical comparisons of the transients for the linearized models and nonlinear model. It shows the big steady state difference relatively to the small step input. It also shows that for the transients, both the damping and the natural frequency are quite different when one compares the linearized model and the nonlinear model. Thus the linear properties of object 3, which is the currently best approximation of the microengine, are very poor, especially considering that the step input so small (1% $\dot{m}_{f_{des}}$).

The same results can be obtained by investigating the positions of the eigenvalues for object 3 shown in Fig. 3-7. It shows that the eigenvalues change their positions very quickly with changes in fuel mass flow. This causes the damping and natural frequency to change very quickly, and thus the linear properties of the system are very different at different operating points. Hence the linear properties of the system are very poor.

Since the linear properties of the system when the heat transfer effects are ignored are very benign, as those of object 1 and object 2, we conclude that heat transfer is one of the causes for the poor linear properties. Thus one must either design a potentially complex control system, or reduce heat transfer to the compressor to improve the open-loop system properties.

Another cause of poor linear properties is the uncommon T_{41} behavior with $\dot{m}_f$. One may already notice that for object 1, the linearized model gives a relatively large steady state error for T_{41}, as shown in Fig. 3-4 (a). This is because of the unusual T_{41} trends with $\dot{m}_f$.

3.4 Transient Properties

The transient properties of the microengine determine how it should be accelerated and decelerated, and how it will behave in response to perturbations. Understanding them is essential to avoiding surge, over temperature and over speed. In the simulations performed the transient properties are presented in the following different ways: the time domain step responses, the damping and natural frequency, and acceleration from the minimum speed to 100% speed. This section describes the results.

3.4.1 Time Domain Step Response

Typical time domain responses for the three objects are shown in Fig 3-8 a), b) and c), respectively. The step responses shown in Fig 3-8 a) and b) are typical step responses of first-order systems, and the step responses shown in Fig 3-8 c) are typical step responses of second-order systems. Since object 1 and object 2 ignore the heat transfer dynamics, while object 3 takes it into account, these results are consistent with the dynamical analysis in section 2.1, which says that when the heat transfer dynamics are taken into account, they are in in the same range of the acceleration dynamics, and the system becomes a second-order system.

From these figures the time constants for object 1 and object 2 can be easily found to be around 0.3s. The characteristic time for object 3 can be found to be around 0.3s as well. These values are consistent with the dynamical analysis in section 2.1, which says that $\tau_{acc} \sim 0.34s$, $\tau_{thermo} \sim 0.17 - 1.68s$. They are much larger than milliseconds level, which was their expected level at the beginning of the microengine design. As stated in section 2.1, this is because of the small torque. As shown in Fig 3-8 a), b) and c), the turbine power and compressor power are both at the order

of 40W. Their difference is very small. As a comparison, the parameters of the microengine and the T700 engine ([12]) are listed in Table 3.3 and their acceleration time constants are estimated using Kerrebrock's formula ([7] and section 2.1). It can be seen that although the two engines are quite different in size, their time constants actually differ by only a factor of approximate 5. This is consistent with the simulation results.

Table 3.3: Comparisons of Acceleration Time Constants between Microengine and T700

	$N(rpm)$	$J(kgm^2)$	$\dot{m}_c\ (kg/s)$	τ_c	$\tau_{acc}(s)$
T700	44700	0.06	4.5	2.3	1.60
microengine	1.2×10^6	9.3×10^{-10}	0.36×10^{-3}	1.38	0.34
ratio	0.036	6.45×10^7	1.25×10^4	1.67	4.7

3.4.2 Damping and Natural Frequency

Since object 1 and object 2 are close to first-order systems, damping and natural frequency are not issues for them. For object 3, Fig. 3-7, which shows the positions of linearized model eigenvalues at different operating points, is a good picture for investigating the damping and natural frequency of the system. The damping and natural frequency vs fuel mass flow at different operating points are also shown in Fig. 3-2 (c).

Fig. 3-7 and Fig. 3-2 (c) show large variations of the damping at different operating points, from approximately 0.15 to 0.98. The overshoot thus can be as large as 62.0% (As a reference, the overshoot vs damping for second-order systems is shown in Fig. 3-9 (a)). Obviously, under some conditions it is possible to get over-temperature or over-speed unless precautions are taken.

Fig. 3-7 also shows large variations of the real part of the second-order system, from approximately 0.42 to 3.57. This means that the settling time can be as large as approximate 5.38 times the settling time at the design point, or as small as 0.63 times the settling time at the design point, within the speed range of 58% to 116% of design speed (As a reference, Fig. 3-9 (b) shows the variation of settling time vs the real part of the root for second-order systems).

Moreover, large variations of damping and natural frequency at different operating points, as shown in Fig. 3-7, mean that the transient properties at different operating points are quite different. This will make the future design of the fuel control potentially complex.

3.4.3 Acceleration from Minimum Speed to 100% Speed

The simulated acceleration from minimum speed to 100% speed for object 1, 2 and 3 are shown in Fig. 3-10 (a), (b) and c). Here in order to make the graph more clear, the efficiency contours are not plotted in Fig. 3-10 c).

Fig. 3-1 shows that object 1, object 2 and object 3 all have pretty big surge margins, whether or not the heat transfer effects are taken into account. For object 1 and object 2, when the heat transfer effects are ignored, accelerating the engine from minimum speed to 100% speed doesn't cause surge, as shown in Fig. 3-10 (a) and (b). But for object 3, when the heat transfer is taken into account, this acceleration leads to surge, as shown in Fig 3-10 (c), even though the surge margin is big. Here the no-heat-addition map, and the transients plotted on it, are taken as an equivalent 2D plot of the 3D behavior. It can be shown that proximity to the surge line is preserved by transforming to this representation. This behavior is consistent with the general behavior of a second-order system, when the damping is small and the overshoot is large, which happens in the microengine system, as indicated in the last subsection. Therefore, an engine of this type would have to incorporate a control system to avoid fast acceleration.

3.5 Sensor Options

Sensors will be used to determine the microengine operating point and to provide feedback for control purposes. A rotation speed sensor is the most direct sensor. It can give us directly the rotation speed, which is the most important parameter, and then the other parameters. But a rotation speed sensor may not be available for some microengines. Thus temperature and pressure sensors are of interest.

The trends of the temperatures and the pressures at different stations vs thrust are shown in Fig. 3-11, with a) for object 1, b) for object 2 and c) for object 3. These figures show the same trends of the temperatures and the pressures for different objects. Temperature sensors, although they have relatively fast response times, can lead to multiple thrust, and thus speed, estimates. This is true for T_3, T_{41}, and T_{45}. Physically this is because of the unusual trends of T_{41} with $\dot{m}_f$. Pressure sensors, on the other hand, although they have relative slow response time, change monotonically with thrust. So the future choice of sensors may be some combinations of the two. Also, some manipulations are necessary to get good estimates of the engine states.

According to the observability matrix of the linearized model, the approximate linear system is observable by any of the following parameters: rotation speed, T_3, T_{41}, T_{45}, P_3, P_{41}, and P_{45}. This makes the choice of sensors very flexible for feedback control purpose. Further study of the relative sensor sensitivities and response times for realistic sensors are needed.

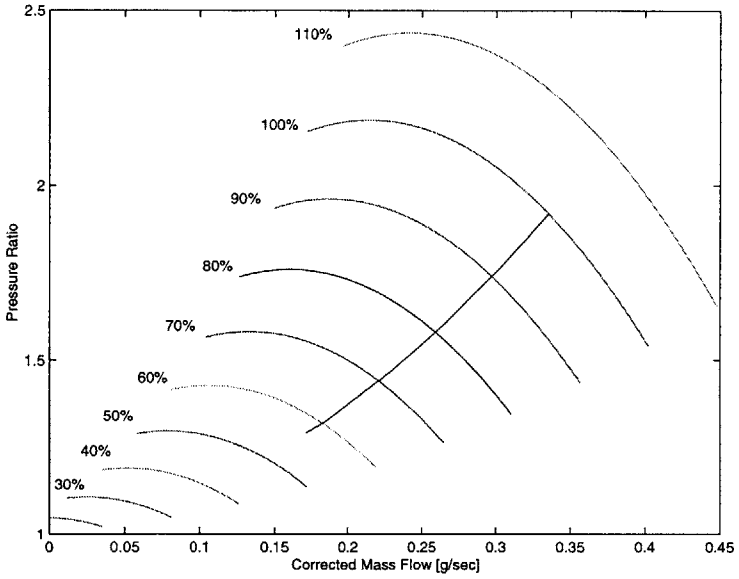
For object 3, one may wonder why there are multiple thrust values for same temperature T_3, while only one thrust value for one pressure P_3 value, given that P_2 and T_2 are constant. This is because of the heat transfer from the turbine side to the compressor side. Not only the pressure rise across the compressor contributes to the temperature rise, but also the heat transfer as well. The unusual trend of T_{41} with $\dot{m}_f$ leads to the possibility of an unusual trend of T_3 with $\dot{m}_f$ via heat

transfer. Summed with the pressure rise part, the results may be multiple trends, depending on the relative numbers of the two parts. Actually, when heat transfer is ignored as for object 1 and object 2 shown in Fig. 3-11 (a) and (b), the thrust does vary monotonically with respect to T_3.

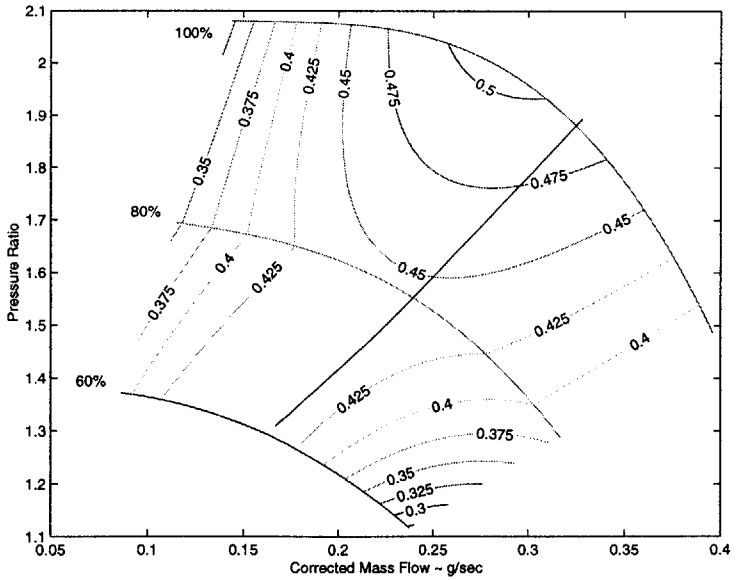
3.6 Summary

The simulation results show that because of the heat transfer dynamics, the microengine system behaves like a second-order system instead of a first-order system, as conventional engines. There is the possibility of increasing T_{41} with decreasing $\dot{m}_f$ in some regions, where the fuel-to-air mass flow ratio increases with decreasing fuel mass flow. Because of the severe heat transfer and the unusual trend of T_{41} with $\dot{m}_f$, the linear properties of the system are not good. Therefore one must either design a potentially complex control system, or reduce heat transfer to the compressor to improve the open-loop system properties. The time constant for acceleration of the microengine is at the level of several hundreds milliseconds. The surge margin at the design point is large, but because of the behavior of the second-order system and the low damping, the overshoot can be very large. This leads to surge when accelerating directly from minimum speed to 100% speed, which means that an engine of this type would have to incorporate a control system to avoid fast acceleration. The overshoot of the second-order system also increases the possibility of over temperature and over speed unless precautions are taken. If a speed sensor is not available in the microengine, temperature sensors and pressure sensors may be options. Although the temperature sensors have relatively fast response time, the same temperature gives multiple thrust, and thus speed, estimates. While the pressure sensors have relatively slow response time, their change with thrust is monotonic. So the future choice of sensors may be some combinations of the two. Also, some manipulations must be done to get good estimates of the engine states.

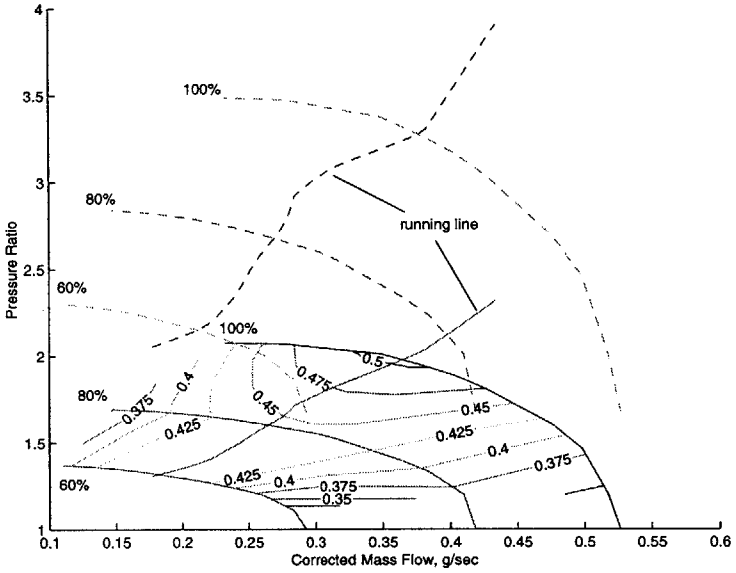
Most of the special characteristics of the microengine behavior arise from two unique characteristics. One characteristics is the heat transfer; another is the fact that T_{41} can increase with decreasing $\dot{m}_f$ in some regions. These two unique characteristics make the analysis, modeling, simulation, and the future control design for the microengine different from conventional engines, and give very interesting results. More on these issues will be seen in the following parameter studies and simulations of the startup process.



(a) running line - object 1



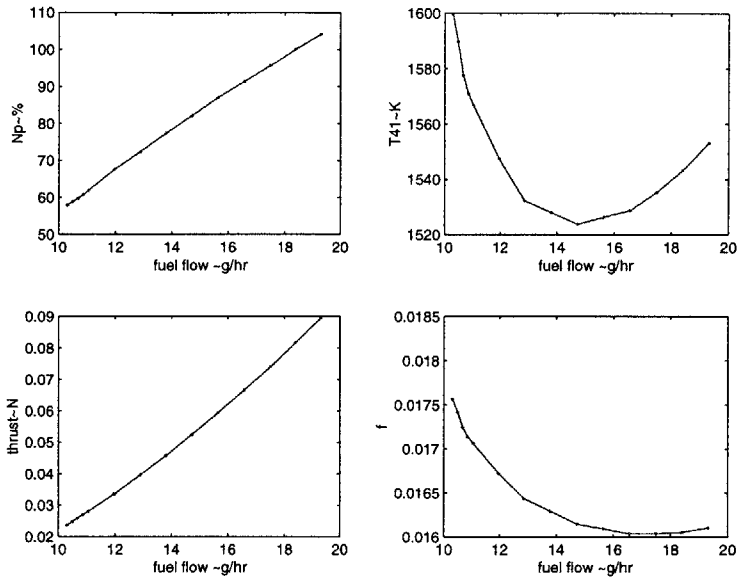
(b) running line - object 2



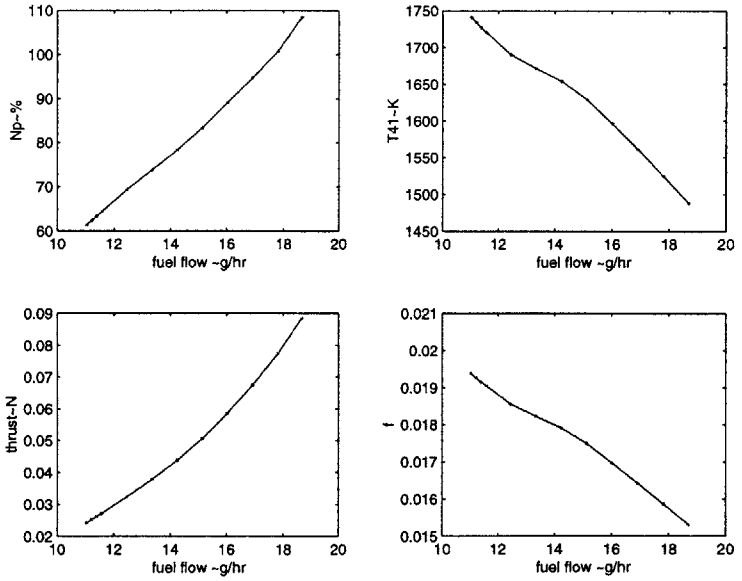
(c) running line - object 3

Figure 3-1: Running Line of Nonlinear Model

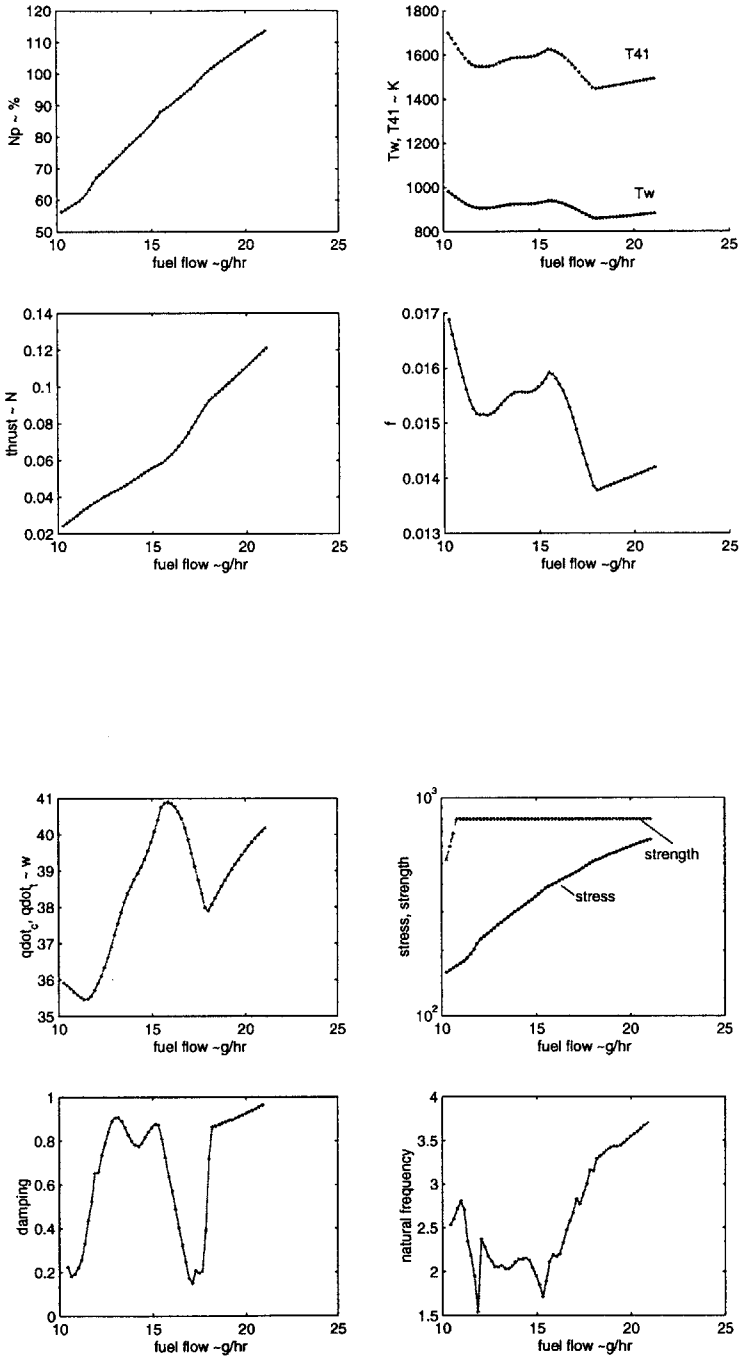
(see page 51 for descriptions of objects; see Table 2.1 for line types)



(a) steady states - object 1



(b) steady states - object 2



(c) steady states - object 3

Figure 3-2: Steady States of Nonlinear Model

(see page 51 for descriptions of objects)

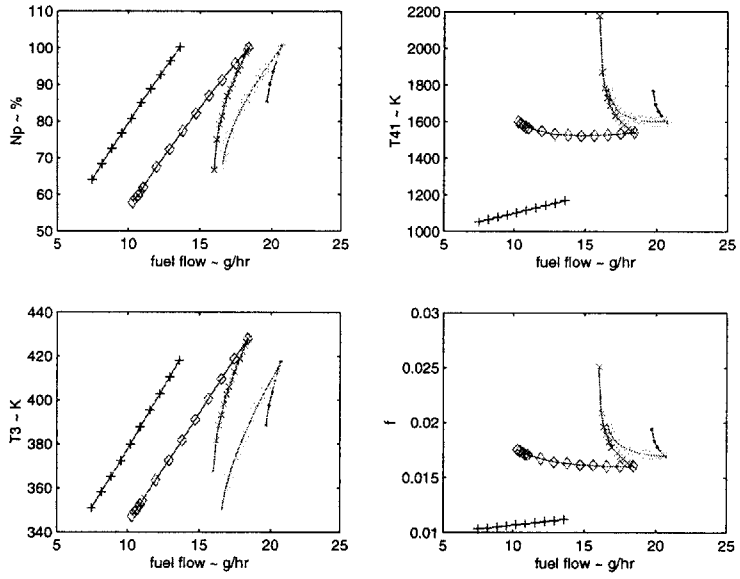
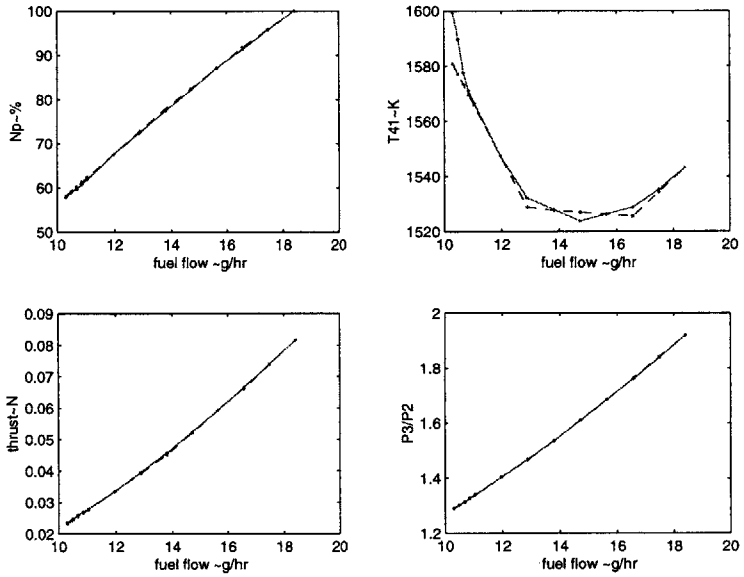
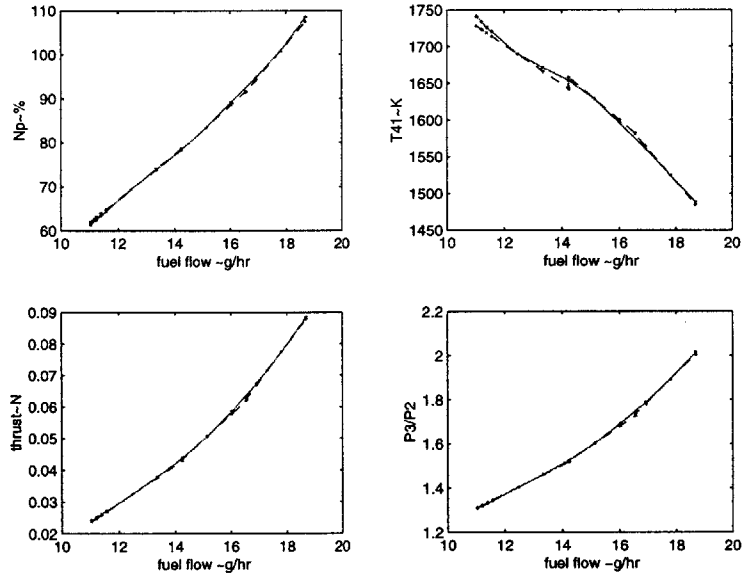


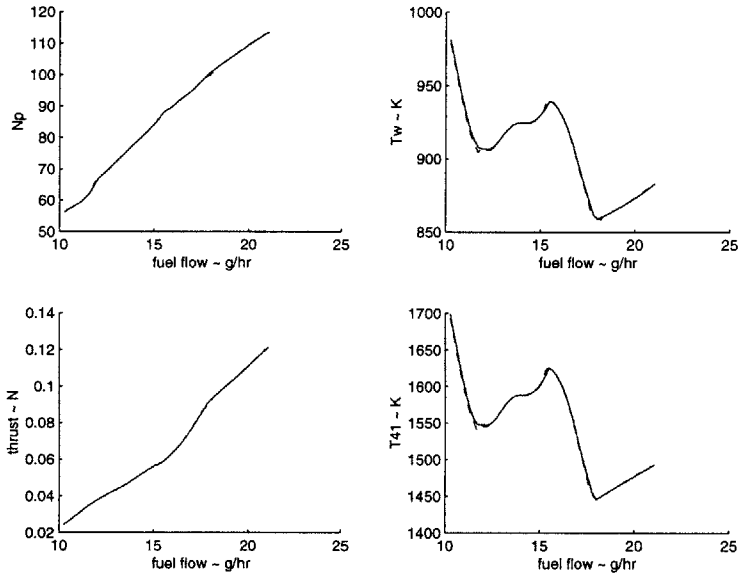
Figure 3-3: Steady States vs Fuel Flow for Different Shaft-off-take Power and Turbine Map
"×" - case 1; "◇" - case 2; "+" - case 3; "." - case 4; "○" - case 5.
(see page 54 for descriptions of cases)



(a) steady states comparisons - object 1



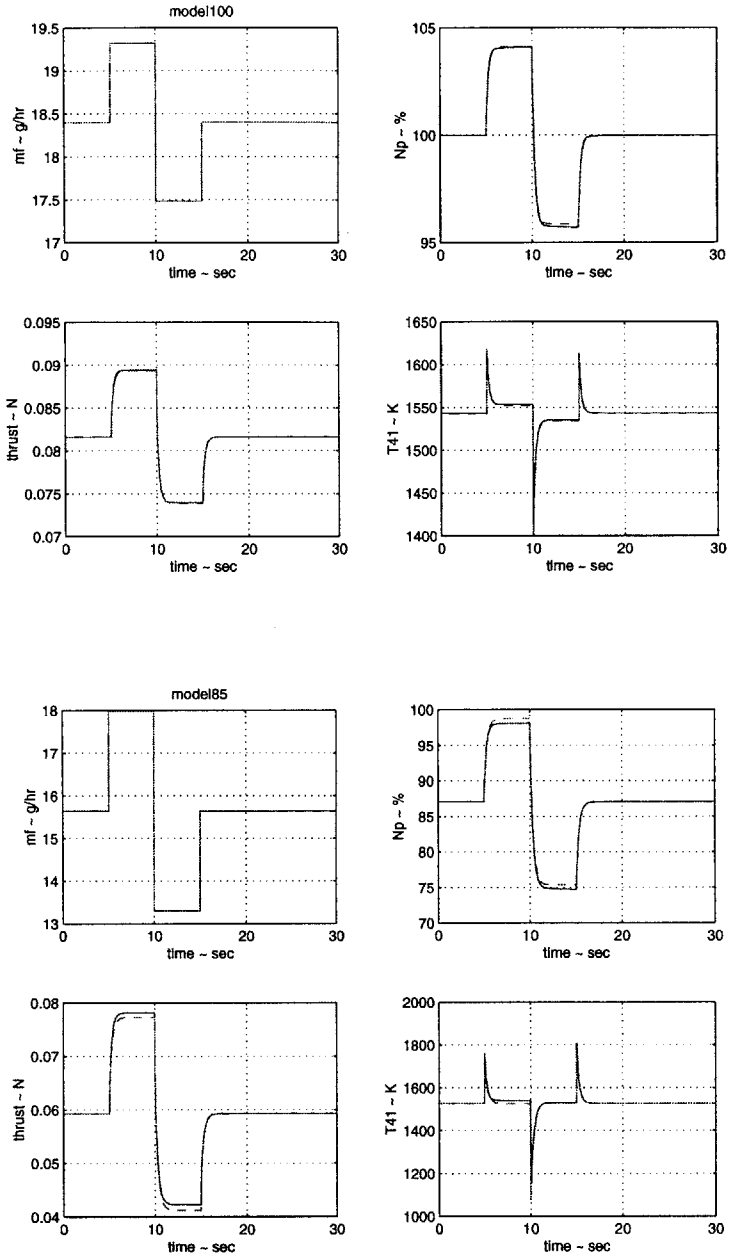
(b) steady states comparisons - object 2

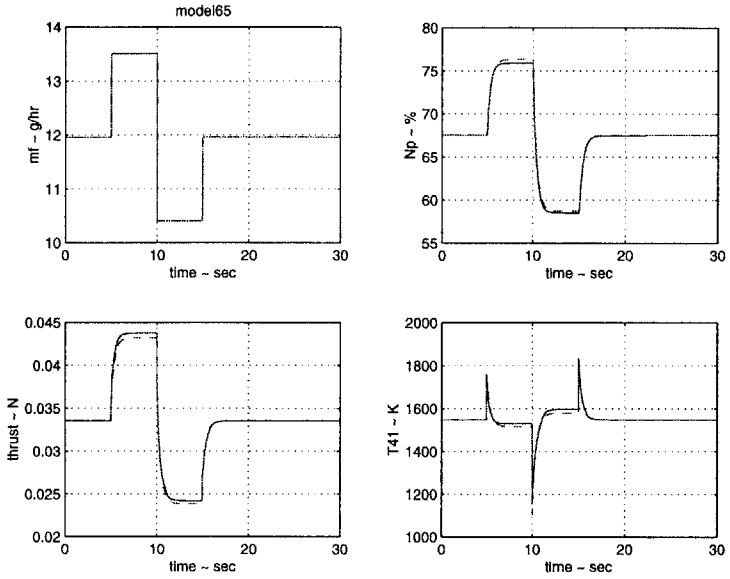


(c) steady states comparisons - object 3

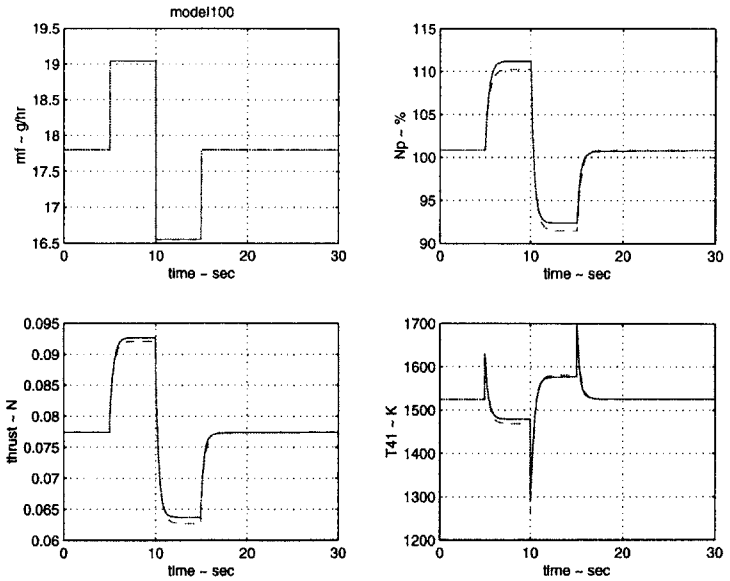
Figure 3-4: Steady States Comparisons between Linearized Model and Nonlinear Model

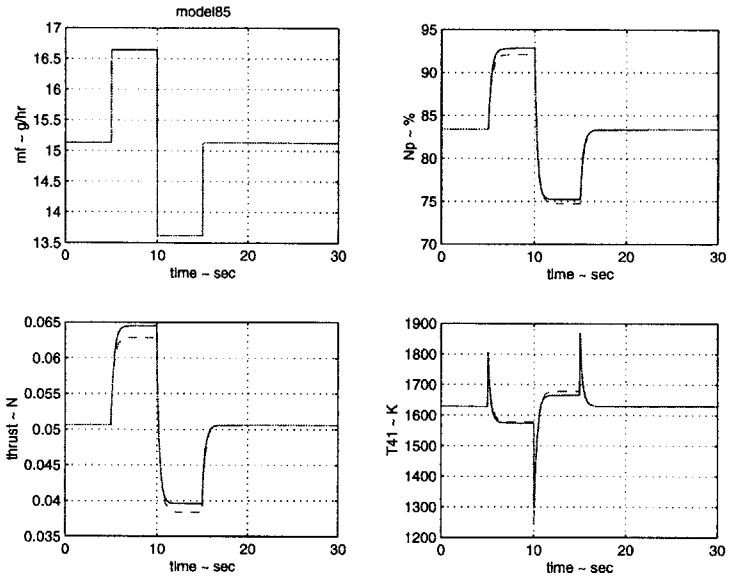
(see page 51 for descriptions of objects; see Table 3.2 for line types)

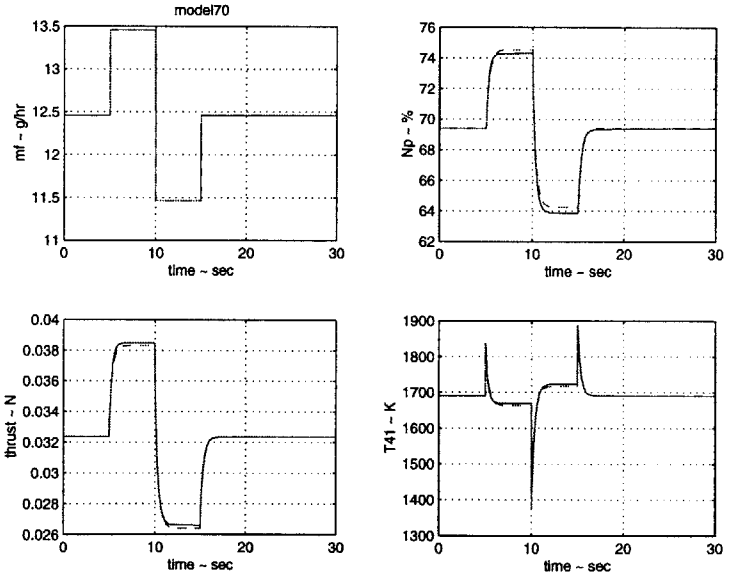

model100
mf ~ g/hr
Np ~ %
thrust ~ N
T41 ~ K
time ~ sec
model85
mf ~ g/hr
Np ~ %
thrust ~ N
T41 ~ K
time ~ sec



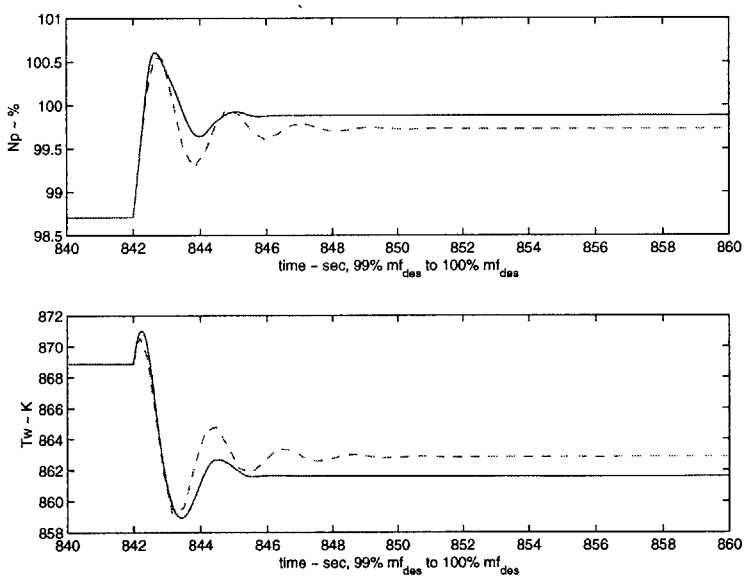
(a) time domain transients comparisons - object 1, linearized model around 3 operating points







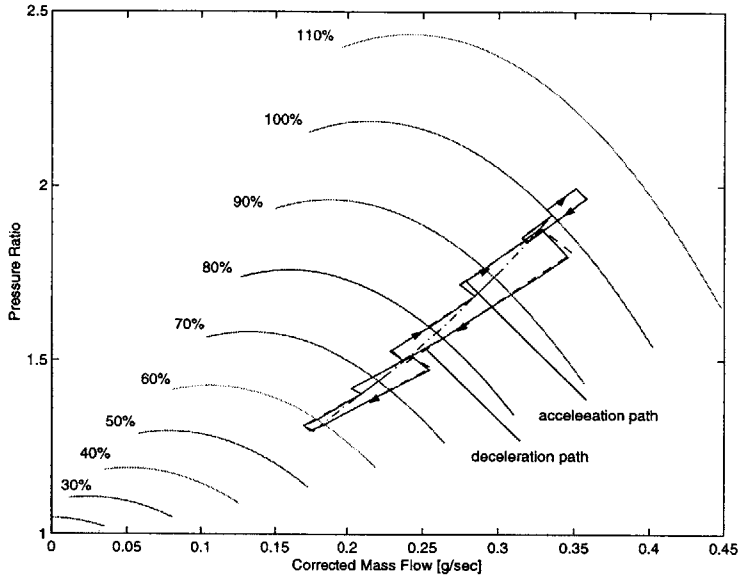
(b) time domain transients comparisons - object 2, linearized model around 3 operating points



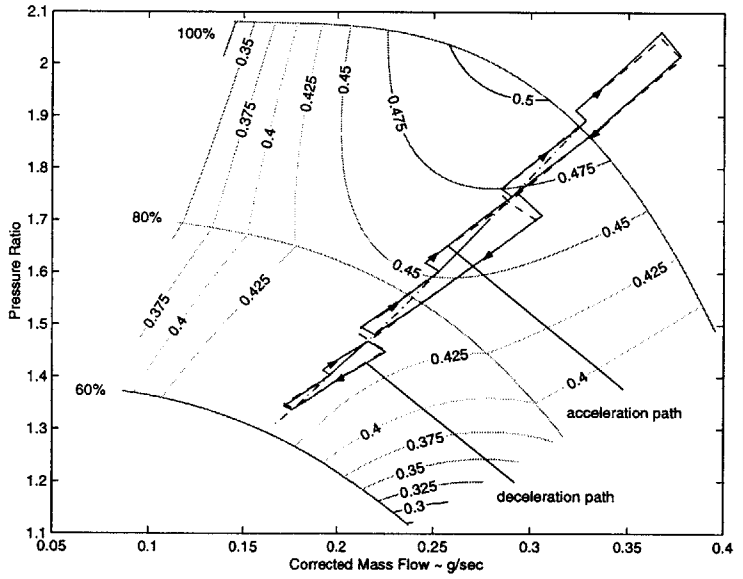
(c) time domain transients comparisons - object 3

Figure 3-5: Time Domain Transients Comparisons between Linearized Model and Nonlinear Model

(see page 51 for descriptions of objects; see Table 3.2 for line types)



(a) transients comparisons on compressor map - object 1



(b) transients comparisons on compressor map - object 2

Figure 3-6: Transients Comparisons on Compressor Map between Linearized Model and Nonlinear Model

(see page 51 for descriptions of objects; see Table 3.2 for line types)

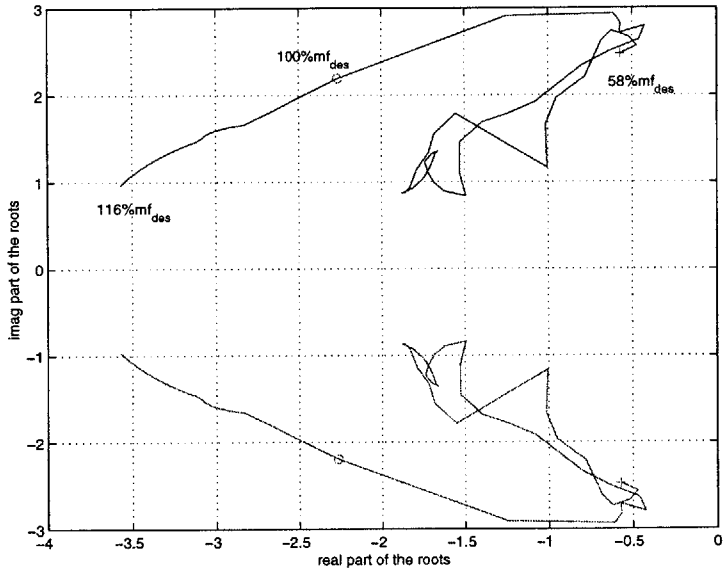
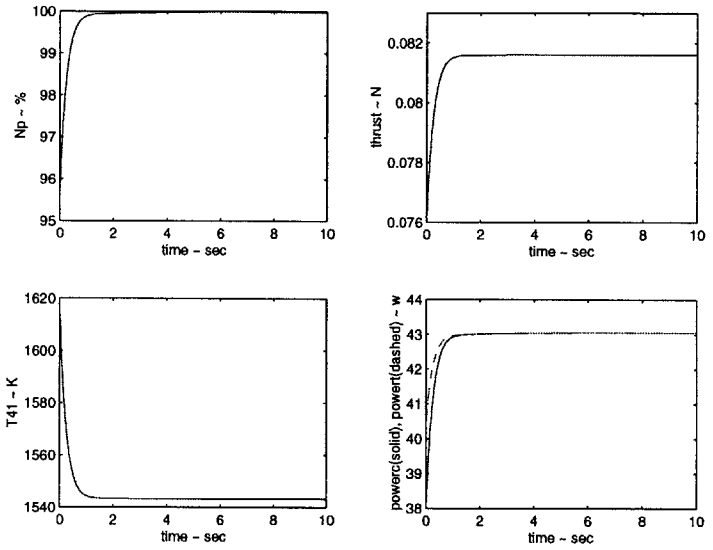
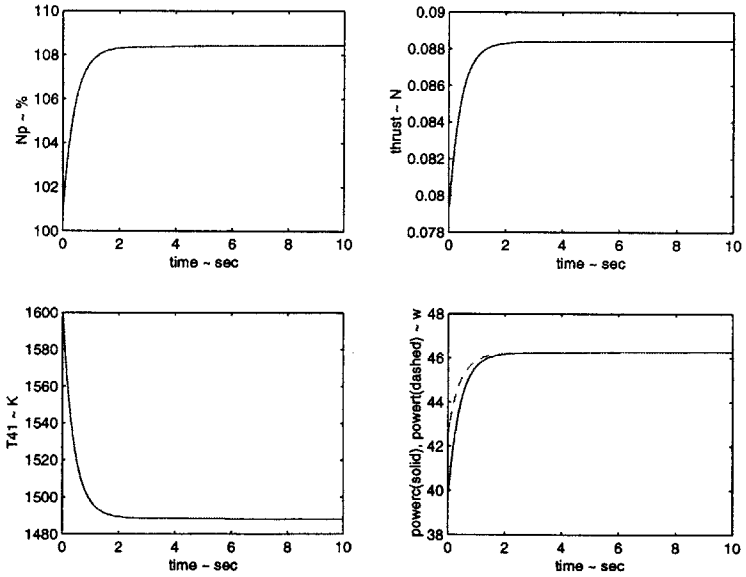


Figure 3-7: Root Distribution - Object 3

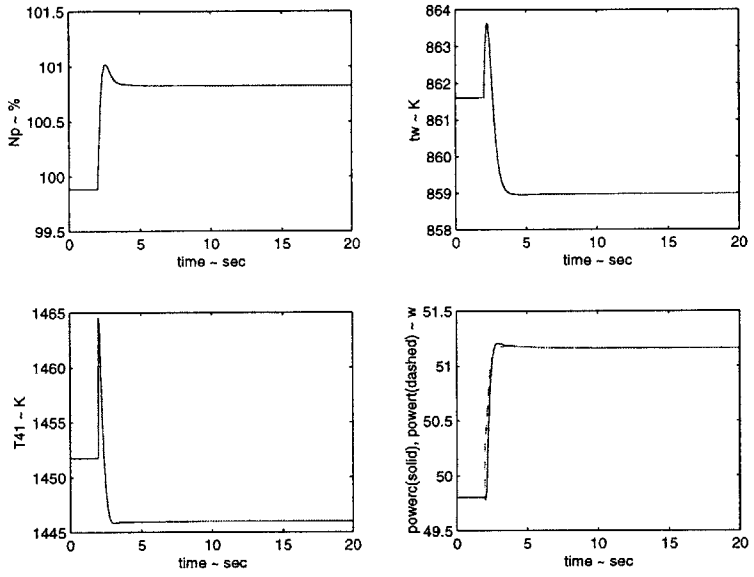
(see page 51 for descriptions of objects)



(a) typical transients - object 1



(b) typical transients - object 2



(c) typical transients - object 3

Figure 3-8: Typical Time Domain Step Response of Nonlinear Model

(see page 51 for descriptions of objects)

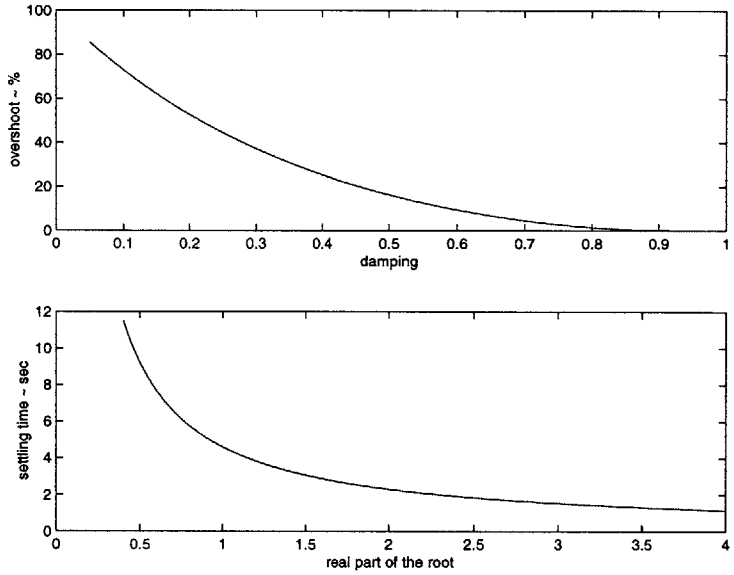
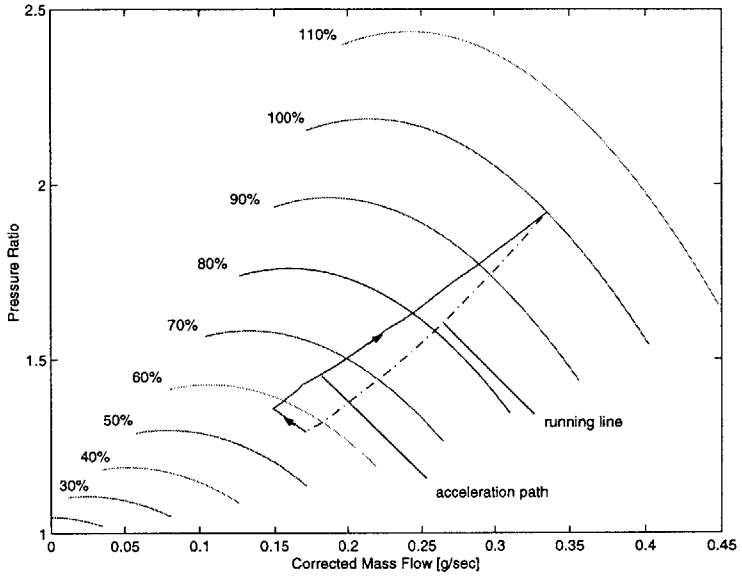
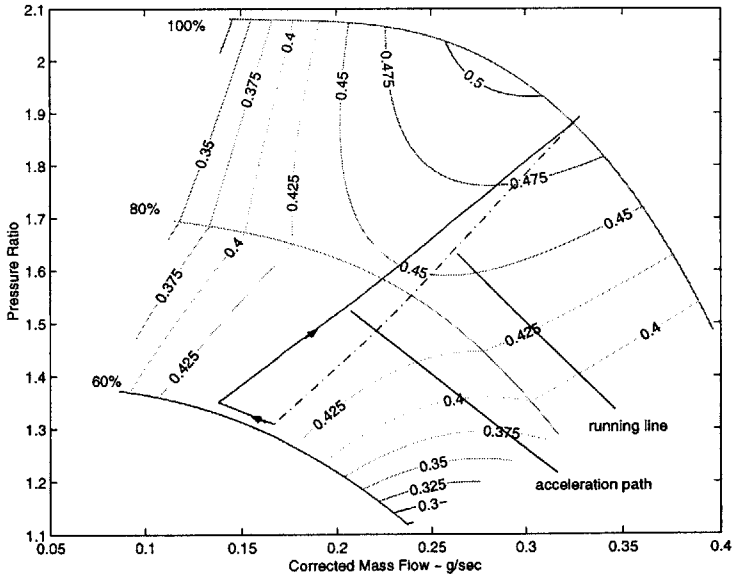


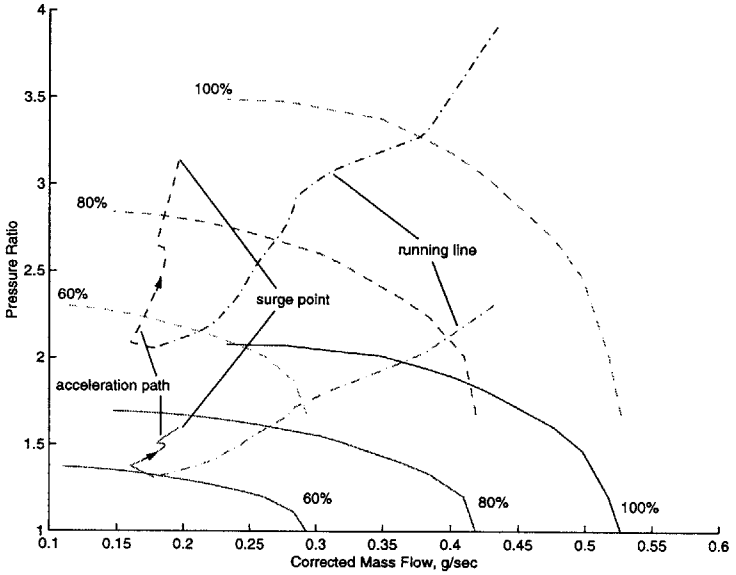
Figure 3-9: Characteristics of Second-order Systems



(a) from minimum speed to 100% speed - object 1



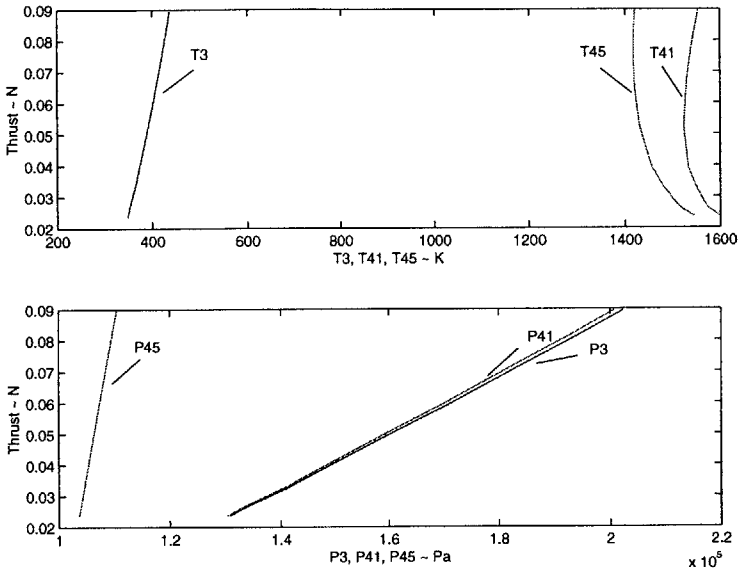
(b) from minimum speed to 100% speed - object 2



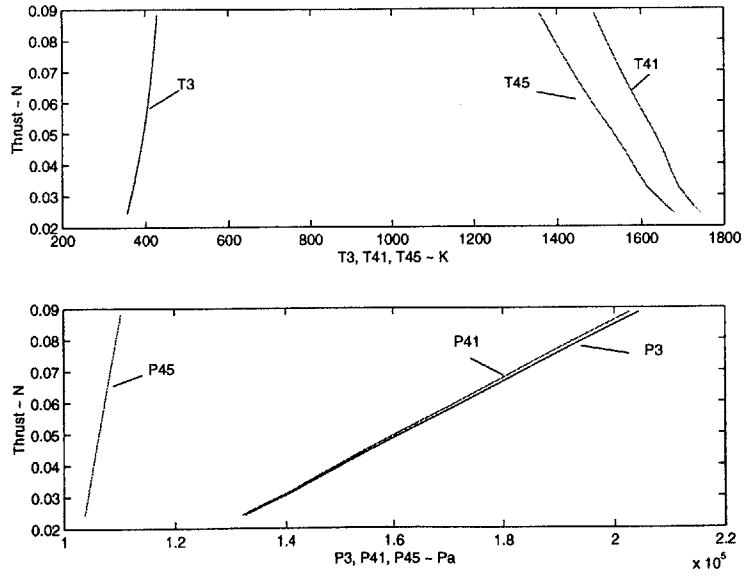
(c) from minimum speed to 100% speed - object 3

Figure 3-10: Acceleration from Minimum Speed to 100% Speed

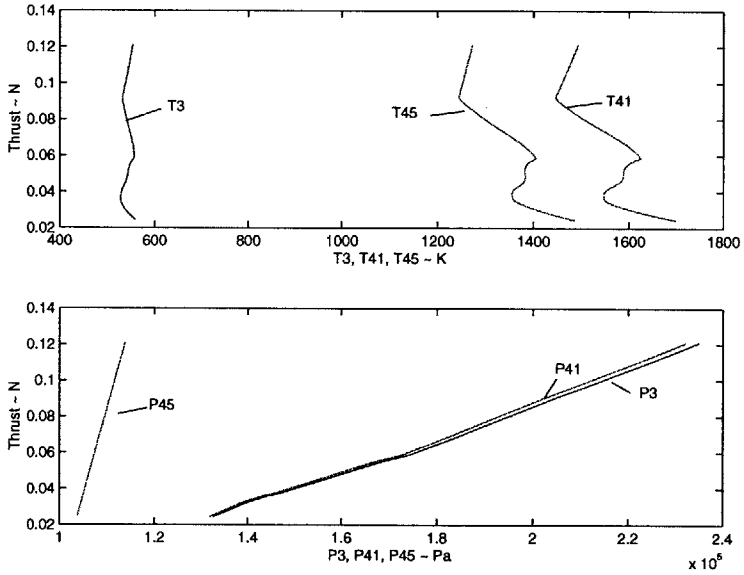
(see page 51 for descriptions of objects; see Table 3.2 and Table 2.1 for line types)



(a) Thrust Change with T and P - object 1



(b) Thrust Change with T and P - object 2



(c) Thrust Change with T and P - object 3

Figure 3-11: Thrust Change with T3, T41, T45 and P3, P41, P45

(see page 51 for descriptions of objects)

Chapter 4

Parameter Studies and Startup Process

This chapter considers the microengine behavior sensitivity to design parameters, as well as the startup process. Section 4.1 investigates the sensitivities, including the parameter descriptions and the simulation results. Section 4.2 describes the startup process and the corresponding results.

4.1 Parameter Studies

The purpose of the parameter studies is to predict the trends of the microengine behavior with parameter variations. These trends can help us to get an estimate of the possible engine behavior when the characteristics of the real engine components are not exactly the same as the modeled ones, which is always the case in reality. In addition, these trends can provide a reference for future modifications of the engine system and components.

4.1.1 Parameter Descriptions and Behaviors Studied

As stated before, the microengine has the characteristics of low and diabatic component performance and coupling between the rotor acceleration dynamics and the heat transfer dynamics. The parameters studied are thus chosen to be those that represent these characteristics of the microengine. The parameters and their values chosen are as follows:

1. $k_{qpi}, k_{qef} = 100\%, 50\%, 25\%, 0$ of their designed value, respectively: coefficients for compressor pressure ratio and efficiency drop due to heat addition, as in equation 2.9.

   Because of the high thermal conductivity of silicon and the integral design of the spool, the heat flux from the turbine side to the compressor side in the microengine is relatively high.

This heat flux has relatively strong effects on the compressor performance, which result in lower pressure ratio and efficiency, as described by equation 2.9. This is one of the unique characteristics of the microengine. The reason for choosing k_{qpi} and k_{qef} as parameters to study is that they indicate how much the heat flux can effect the engine performance via the compressor performance drop.

2. $e_{peak} = 0.65, 0.55, 0.50, 0.45$: peak efficiency the compressor can reach, where 0.55 is the designed value.

   The peak efficiency the current microengine compressor design can provide is about 0.55, which is relatively low compared with a conventional engine. This motivates us to investigate the effect of limiting the efficiency.

3. $k_{jg}, k_{cm} = 10, 1, 0.1$: the scaling factors of the spool mechanical inertia J_{jg} and thermal inertia J_{cm}, respectively, i.e., $k_{jg} \equiv \frac{J_{jg}}{J_{jgdes}}$ and $k_{cm} \equiv \frac{J_{cm}}{J_{cmdes}}$, where $J_{cm} = C_w M_w$ is the thermal inertia as in equation 2.33.

   As stated previously, there is coupling between the rotor acceleration dynamics and the heat transfer dynamics. This coupling makes the microengine a second-order system instead of a first-order system, which is different from a conventional engine. These two dynamics can be characterized by their respective inertias: rotor spool mechanical inertia J_{jg} and thermal inertia J_{cm}. Thus parameter sensitivity for these two inertias are studied to analyze the dynamic coupling.

4. $An = 0.85\times, 1\times, 3\times$, of its designed value: nozzle area.

   The reason for choosing An is to change the position of the running line. The number 0.85 moves the design point to the left on the 100% speed line, to about one half of the position between the current design point and the peak pressure point, in terms of the mass flow. The number 3 moves the design point to the right on the 100% speed line, to about one and a half times the position between the current design point and the peak pressure point.

All of the parameter studies are performed on the modified diabatic experimental map, the fourth compressor map, as stated in section 2.2, in order to reduce the computation times without losing the basic properties of the third compressor map, the best compressor map currently available. The behavior studied includes:

- minimum achievable speed;
- running line and steady states;
- transients.

4.1.2 Simulation Results for Parameter Studies

The simulation results for parameter studies are shown in Fig. 4-1 to Fig. 4-3, which indicate the running line, the steady states, and the locus of linearized system eigenvalues (root distribution), respectively. Here again the approximate 2D representation of the 3D map, with both the speed lines with no heat flux and the speed lines with 40W heat flux on the map, is used for illustrating the simulation results. The notation used in these figures is listed in Table 4.1. Because the time domain transients are similar to those shown in chapter 3, and because the root distribution, damping and natural frequency uniquely determine the transient behavior, the time domain transients are not shown here. There are several things that needed to be pointed out:

Table 4.1: Notation of Graphs for Parameter Study

notation	$\frac{k_{qpi}}{k_{qpides}}$	$\frac{k_{qef}}{k_{qefdes}}$	e_{peak}	k_{jg}	k_{cm}	$\frac{An}{An_{des}}$
o	1	1	0.55	1	1	1
$\diamond$	50%	50%	0.65	10	10	0.85
+	25%	25%	0.50	0.1	0.1	3
$\star$	0%	0%	0.45	-	-	-
index	(a)	(b)	(c)	(d)	(e)	(f)

dashed line	without $\dot{Q}$, T_w, strength
solid line	with $\dot{Q}$, T_{41}, stress

1. Fig. 4-1 to Fig. 4-3 (a) and (b) show the effects of variations of k_{qpi} and k_{qef}. Fig. 4-1 (a) shows that k_{qpi} variations move the running line quite a bit, while Fig. 4-1 (b) shows that k_{qef} variations have little effect on the position of the running line. Fig. 4-2 (a) and (b) and Fig. 4-3 (a) and (b) give the same results, showing that k_{qpi} variations cause strong changes in both steady states and transients, while k_{qef} variations cause little change to either of them. These results can be explained by the scale of k_{qpi} and k_{qef}. At the designed point, k_{qpides} is at the level of 0.01, while k_{qefdes} is at the level of 0.002. With around $40W$ typical heat flux, k_{qpi} will give a ratio between actual π_c and adiabatic π_c of 0.6 (a large change), while k_{qef} will give a ratio between actual η_c and adiabatic η_c of 0.92 (i.e. η_c remains almost the same) (equation 2.9). Therefore it is clear why the variations of the two coefficients gives us totally different effects on the engine performance.

2. Fig. 4-1 to Fig. 4-3 (c) show the effects of variations of e_{peak}. Fig. 4-1 (c) shows the different running lines when peak efficiency changes. Note that the adiabatic running line bifurcates towards the surge direction when it reaches the peak efficiency. Below the peak efficiency contour, all the running lines are on the top of each other. This is the reason why the adiabatic efficiency contours are plotted here. This is also the reason why the speed lines are plotted from 60% to 140% in this particular case, while in other cases the speed lines are plotted from

60% to 100%, the microengine operating range. In accordance with the bifurcations of the running lines, all the steady states and root distribution also bifurcate, as shown in Fig. 4-2 and Fig. 4-3 (c). The reason that the running line bifurcates toward the surge region is that an engine with lower efficiency requires higher pressure ratio to suck in the same amount of mass flow.

3. Fig. 4-1 to Fig. 4-3 (d) and (e) show the effects of variations of the scaling factors k_{jg} and k_{cm}, which are the same as the effects of variations of the two intertias J_{jg} and J_{cm}. Fig. 4-1 and Fig. 4-2 (d) and (e) shows that the running line and the steady states are the same with different k_{jg} and k_{cm}. This is reasonable, because the change of the inertias should change only the transients, but not the final states (steady states). Fig 4-3 (d) and (e) and the last two graphs of Fig. 4-2 (d) and (e) do show the large change of the distribution of roots, damping and natural frequency, which are the properties related to transient response. Actually when the root distributions are scaled, the root distribution when $k_{jg} = 0.1$ and when $k_{cm} = 10$ (i.e. $J_{jg}/J_{cm} = 0.1(J_{jg}/J_{cm})_{des}$ for both cases) have the same shape, and the roots distribution when $k_{jg} = 10$ and when $k_{cm} = 0.1$ (i.e. $J_{jg}/J_{cm} = 10(J_{jg}/J_{cm})_{des}$ for both cases) have the same shape (the scaled root distributions are not shown here, but the equivalent damping and natural frequency are shown in the last two graphs of Fig. 4-2 (e) and (d). One can find this relationship from these graphs, too). This is because the coupling of the two dynamical elements - rotor acceleration and heat transfer - depends on the ratio of the two inertias - mechanical inertia J_{jg} and thermal inertia J_{cm}, but not on its absolute values.

   Another note about the effects of the variations of the two inertias: when the thermal inertia J_{cm} is much less than the rotor inertia J_{jg} ($k_{jg} = 10$ or $k_{cm} = 0.1$ cases, i.e. $J_{jg}/J_{cm} = 10(J_{jg}/J_{cm})_{des}$), the system is still an oscillatory system (damping is less than 1); while when the thermal inertia J_{cm} is much more than the rotor inertia J_{jg} ($k_{jg} = 0.1$ or $k_{cm} = 10$ cases, i.e. $J_{jg}/J_{cm} = 0.1(J_{jg}/J_{cm})_{des}$), the roots of the system become two real roots at speeds higher than about 60%, which is the range of interest. Actually in this case the ratio of the two real roots is above 2, which means the thermal dynamics may be ignored and the system can be treated as a first-order system, after some modifications of the modeling process, as what is usually done for conventional engines. Thus, if one want to get better behavior with less possibility of surge, over-temperature and over-speed, and to simplify the system analysis and design as well as the design and implementation of control laws, it is desirable to change the ratio of the two inertias in the direction of increasing the thermal inertia, i.e. slowing the thermal dynamics. This is consistent with the common sense in conventional engines.

   One may wonder that the dynamic coupling should be determined by the ratio of the inertias, but that when the thermal inertia J_{cm} is much less than the mechanical inertia J_{jg}, the system

is still an oscillatory system, while when the thermal inertia J_{cm} is much more than the rotor inertia J_{jg}, the roots of the system become two real roots and we may be able to separate the two dynamics. Actually this is because the heat transfer not only has an effect on dynamic coupling with the rotor acceleration dynamics; it also effects the compressor pressure ratio and efficiency as well, which in turn affects the acceleration dynamics and the heat transfer dynamics.

4. Finally Fig. 4-1 to Fig. 4-3 (f) show the effects of variations of the nozzle area An. Fig. 4-1 (f) shows the different running lines when nozzle area takes the value of 85%, 100% and 300% of the designed value. It can be seen that the variations of the nozzle area don't change too much of the position of the running line, noting that 85% and 300% are not small variations. Fig. 4-2 (f) and Fig. 4-3 (f) shows that they do change a lot on the thrust, heat flux, spool temperature (which is of interest because of the associated stresses) and the transients. But the reason may just be the shift of the operating points. Considering the large variations of the nozzle area here, this should not be a big issue.

A simple summary of the parameter study is listed in Table 4.2.

Table 4.2: Summary of Results from Parameter Study

parameter		N_{min}	N w/ $\dot{m}_f$	thrust w/ $\dot{m}_f$	T_w w/ $\dot{m}_f$	damping	natural frequency
k_{qpi}	50%	64.80%	decrease	decrease	increase	decrease	complex
	25%	71.70%					
	0%	83.50%					
k_{qef}	50%	51.10%	close	close	close	close	close
	25%	51.20%					
	0%	51.20%					
e_{peak}	0.65	49.80%	increase	decrease	increase	complex	decrease
	0.50	49.80%					
	0.45	49.80%					
k_{cm}	10×	51.00%	same	same	same	decrease	increase
	0.1×	56.50%					
k_{jg}	10×	48.10%	same	same	same	increase	increase
	0.1×	51.00%					
An	0.85×	48.70%	close	mostly decrease	decrease	mostly decrease	increase
	3×	52.50%					

4.2 Startup Process

This section first describes the startup procedure and the simulations, then presents the simulation results for the startup process.

4.2.1 Startup Procedure

The startup procedure is as follows:

1. Inject the start air and spin up the engine;

2. Light the combustor;

3. Shut off the start air.

The simulations performed start after step 3, when the start air is shut off with the combustor lighting on.

Because the gas dynamics is much faster than the rotor acceleration and heat transfer (chapter 2, Order of Magnitude Analysis), the only two initial states needed to be considered in the process are the rotation speed and spool temperature, with the input of the system to be the fuel mass flow $\dot{m}_f$ (section 2.2.3).

Because of the existence of heat transfer in the real microengine, simulations of the startup process are performed only on the diabatic experimental compressor map and the modified diabatic experimental compressor map (the third and fourth compressor maps). The heat transfer dynamics are taken into considerations as well.

4.2.2 Startup Process on Diabatic Experimental Compressor Map

The simulation results of the startup process on diabatic experimental compressor map for initial states of $N = 100\%N_{des}$ and $T_w = 400K$ and the input to be the designed fuel mass flow at 100% speed are shown in Fig. 4-4 and Fig. 4-5. Fig. 4-4 shows the startup process on the compressor map (the default line types are as in Table 2.1), and Fig. 4-5 shows the startup process in time domain. It can be seen that at the beginning the rotation speed and the thrust drop, while the spool temperature rises. This means that the energy provided by the fuel is used to heat up the spool at the beginning, but not speed up the spool. After some transients the system reaches the steady states determined by the fuel flow setting. The spool thus behaves like an energy storage device.

Simulation is also performed for another set of initial states and input, where $N = 100\%N_{des}$, $\dot{m}_f$=$\dot{m}_{f_{des}}$ as before, and T_w is increased to $520K$. The graph is not shown here. The simulation results properties similar to the simulations with the previous initial states. Basically the settling time is almost the same, while the overshoot is about 10% less for rotation speed. This is because less energy is needed for heating up the spool to reach the steady temperature.

4.2.3 Startup Process on Modified Diabatic Experimental Compressor Map

Efficiency Gradient and Startup Process on Modified Diabatic Experimental Compressor Map

For the modified diabatic experimental compressor map, first the same initial states and inputs as those of the startup process on the diabatic experimental map, $N = 100\%N_{des}$, $T_w = 400K$ and the

input $\dot{m}_f$=$\dot{m}_{f_{des}}$, are chosen to begin the simulation. Unfortunately these states cannot start the engine properly. Some data analysis shows that it is because of the different efficiency distributions of the two maps. On the lower part of the modified diabatic experimental compressor map, the efficiency gradient is sharper than the original diabatic experimental map (see Fig. 2-6 and Fig. 2-5). During the startup process, when the transient point moves to this region, the system with modified diabatic experimental compressor map has a much lower damping and larger overshoot. This leads the transient point into the surge region, and the engine cannot start correctly.

In order to test this conjecture, two simulations are performed. One is performed when the original compressor efficiency is changed by setting a minimum efficiency. The purpose of this simulation is to discard the probability of failing because of low efficiency. The minimum efficiency is set to be 0.4. The simulation results are shown in Fig. 4-6 and Fig. 4-7. The figures show that after a certain time, there is no efficiency gradient, the damping of the system is very high, and the response is very flat. The startup process fails.

Another simulation is performed by changing the efficiency gradient. The change is as follows:

- when $\eta_c \geq 0.4$, keep the original value;

- when $\eta_c < 0.4$, reset η_c so that $\eta_c^{new} = 0.2 + 0.5\eta_c^{original}$.

This setting actually reduces the efficiency gradient to one-half of its original gradient when $\eta_c < 0.4$, and the efficiency is less than that used in the previous simulation where the minimum efficiency is set to be 0.4. Using the same initial states and fuel flow input, the simulation results are shown in Fig. 4-8 and Fig. 4-9. This time the response has some overshoot, but the overall startup process works well.

Thus we conclude that, for efficiency gradients that are too sharp, the response will oscillate too much and go to the surge region, and the startup process will fail. For efficiency gradient that are too flat, the response will be too flat and the start process will also fail. The efficiency gradient affects the damping, thus the startup process which depends highly on the transient properties of the system.

Adjust Parameters and Startup Process on Modified Diabatic Experimental Compressor Map

Because good experimental data for efficiency are not available, it is necessary to think about other possible ways to start the engine before it is done on the real engines. For this study several initial states and input fuel flow are chosen (shown in Table 4.3) and startup simulations are run for each. The results, together with the results from the test already discussed (setting minimum efficiency and changing efficiency gradient), are listed in Table 4.3. The detail graphs are not shown here.

Table 4.3: Summary of Startup Process

$N(\%)$	$T_w(K)$	$\dot{m}_f$ / $\dot{m}_{f_{des}}$ (%)	η_{cmin}	η_c gradient	succeed or not
100	400	100	-	-	NO
100	400	**130**	-	-	**YES**
100	**600**	100	-	-	**YES**
140	400	100	-	-	**YES**
100	400	100	**0.4**	-	NO
100	400	100	-	**1/2**	**YES**

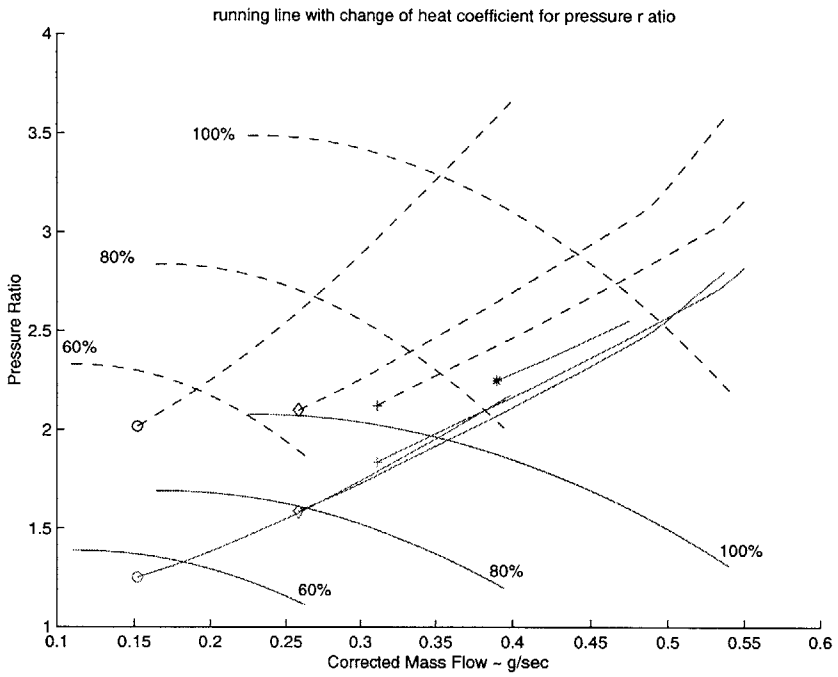
One thing needed to be pointed out is that, if the start fuel flow is set above the designed fuel flow, as the case in the third row of Table 4.3, the startup process ends at a higher rotation speed than the design point unless some control laws are applied to adjust it back to the designed value. Since the purpose of the startup study is to give possible ways to start the engine, the design of the control laws is not performed in the simulations.

Summary

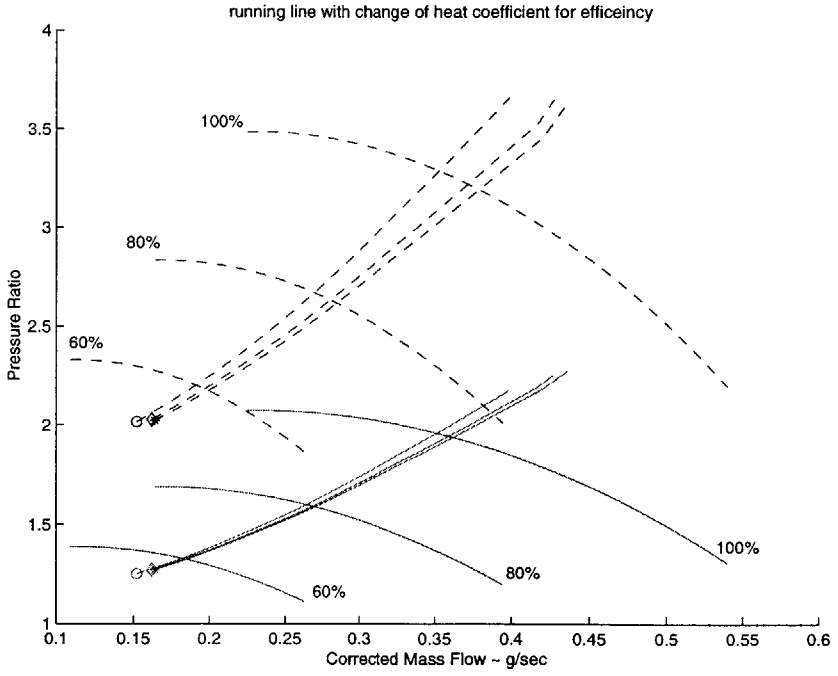
Above results show that there is a relationship between efficiency gradient and damping. In case some parameters of the real engine change, there are several options to make the startup process work:

1. shut off the start air at a higher rotation speed;
2. shut off the start air at a higher spool temperature (for example preheat the engine);
3. increase the start fuel flow (this needs to adjust the fuel flow back to the design value).

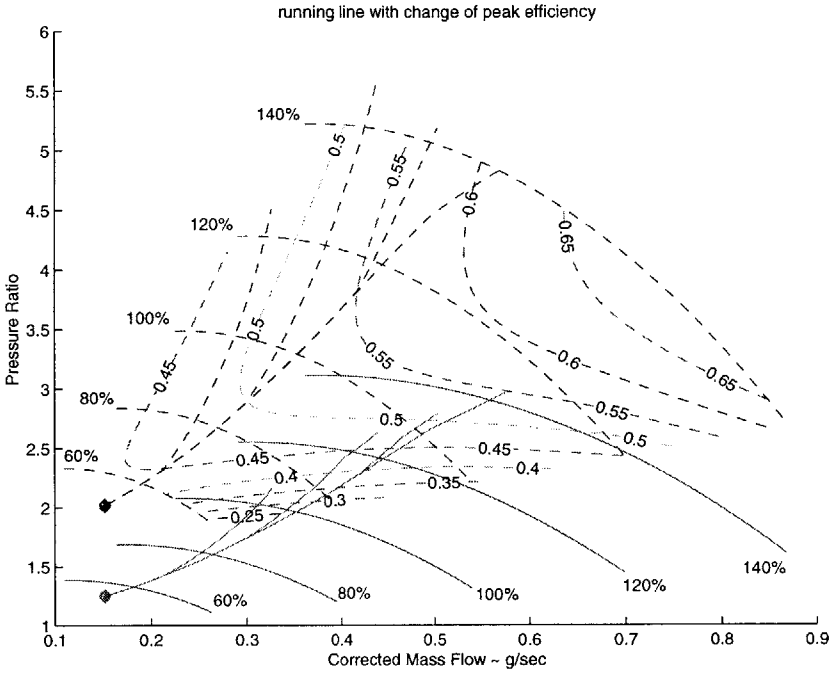
Here options 1 or 2 may be easier, depending on lighting the combustor at a higher rotation speed (option 1), or lighting the combustor at a lower rotation speed but prolonging the time between lighting the combustor and shutting off the start air (option 2), or lighting the combustor at a lower rotation speed but preheating the spool via some other approaches (option 2). The problem here is how to measure the rotation speed and spool temperature. The third option needs some adjusting laws for the fuel flow, which may not be preferred in the operation of engines. Because accurate efficiency data are not currently available, the startup process needs further study.



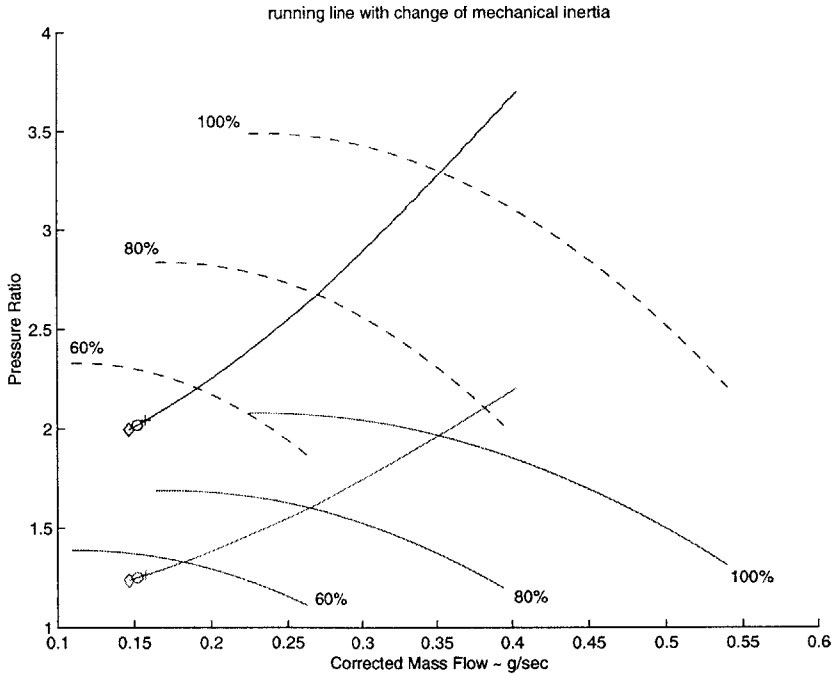
(a) Running Line - Change k_{qpi}



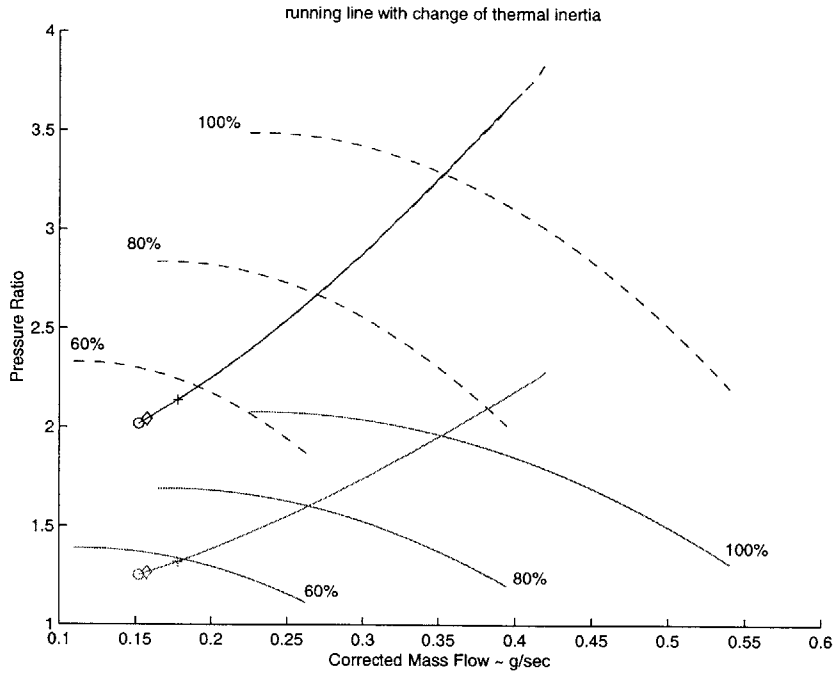
(b) Running Line - Change k_{qef}



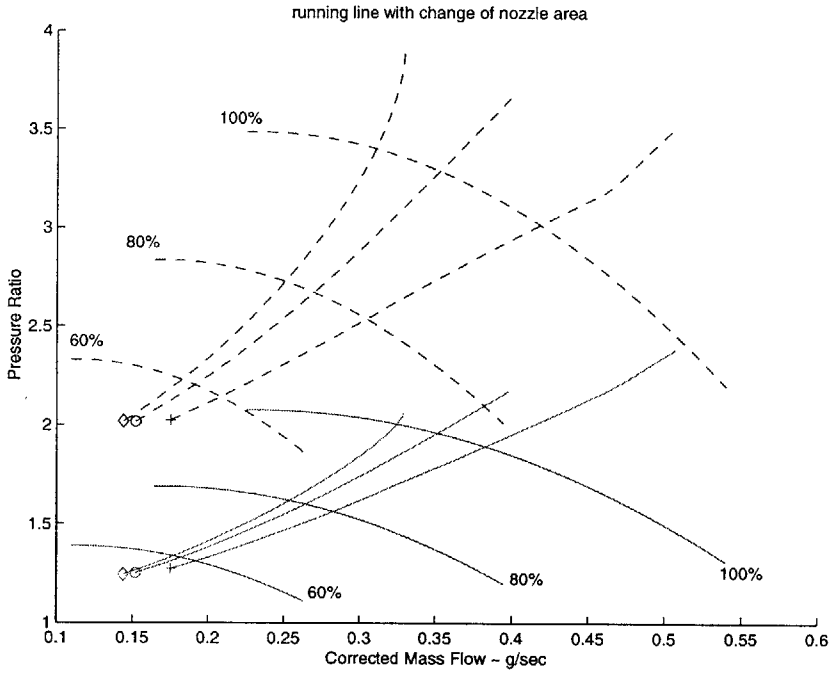
(c) Running Line - Change e_{peak}



(d) Running Line - Change k_{jg}



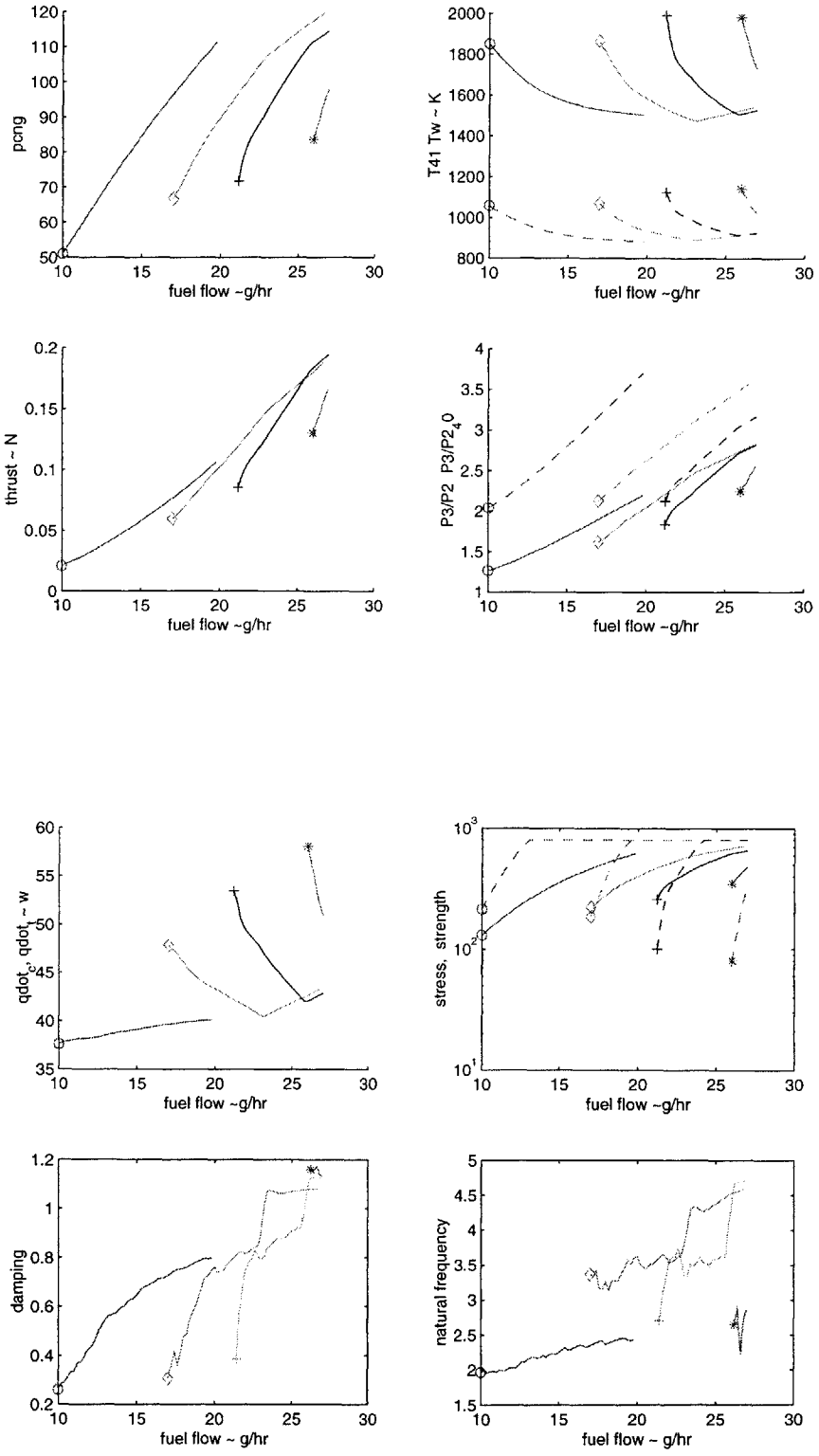
(e) Running Line - Change k_{cm}



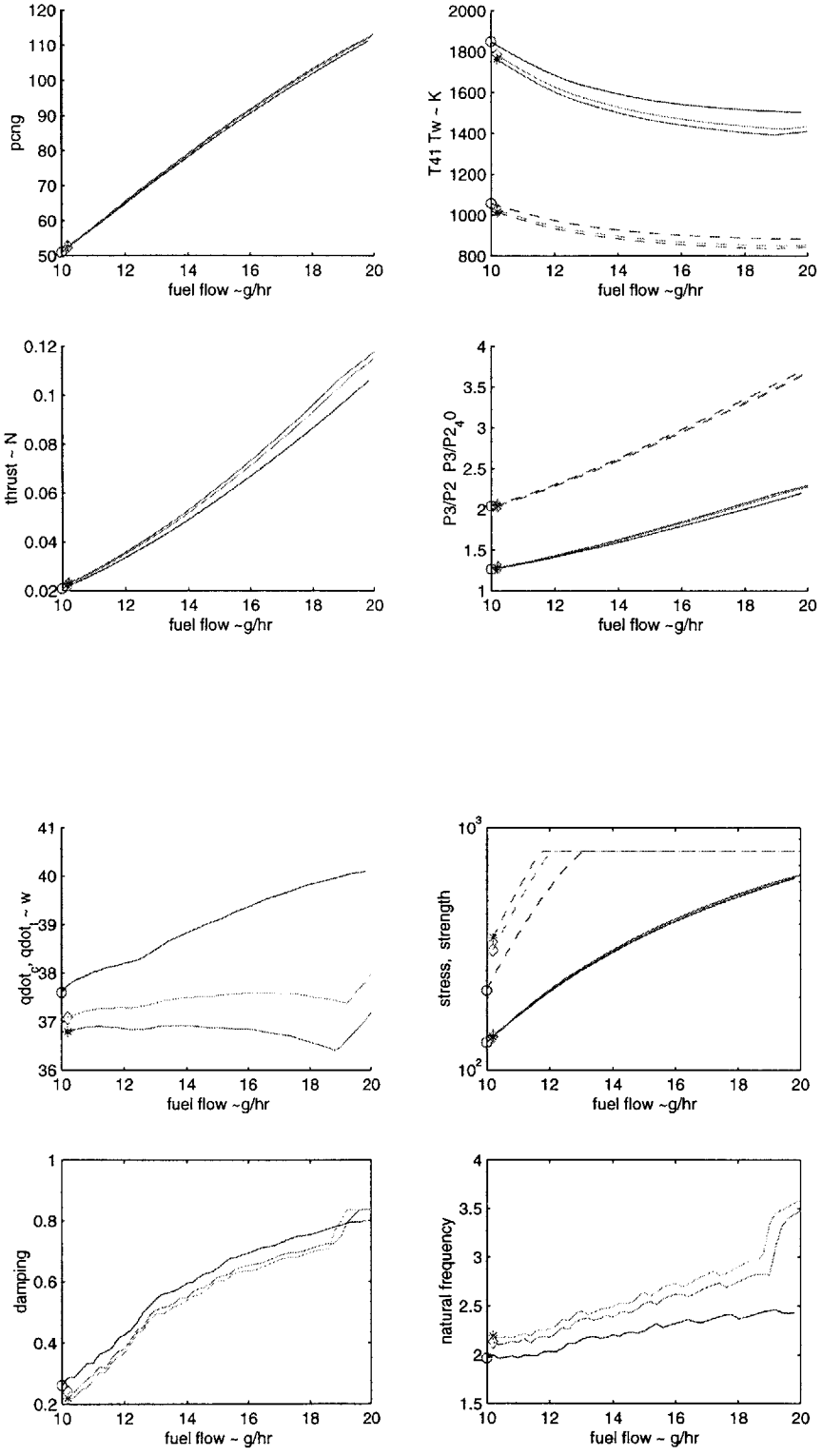
(f) Running Line - Change An

Figure 4-1: Running Line - Parameter Study

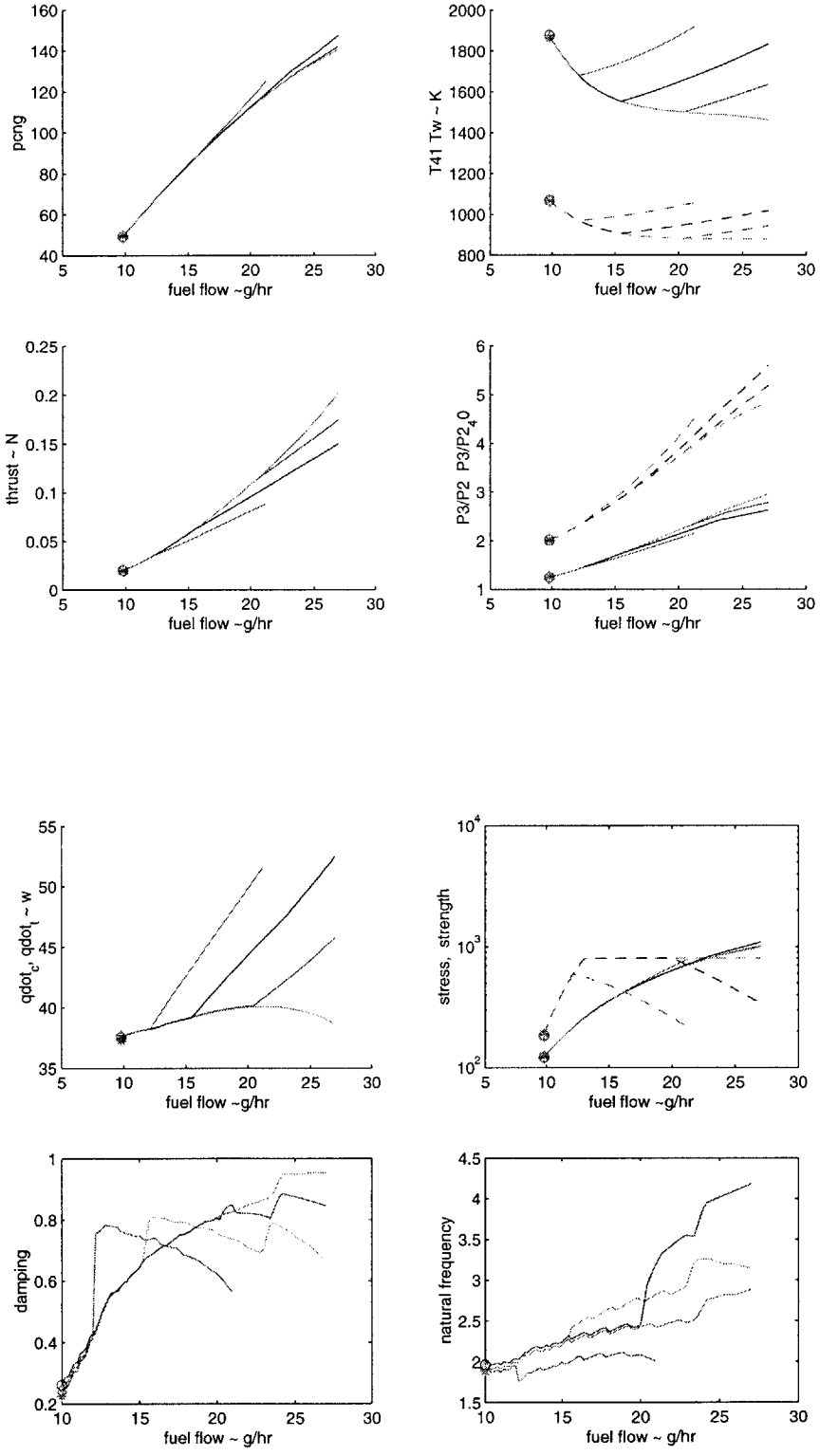
(see Table 4.1 for notations)



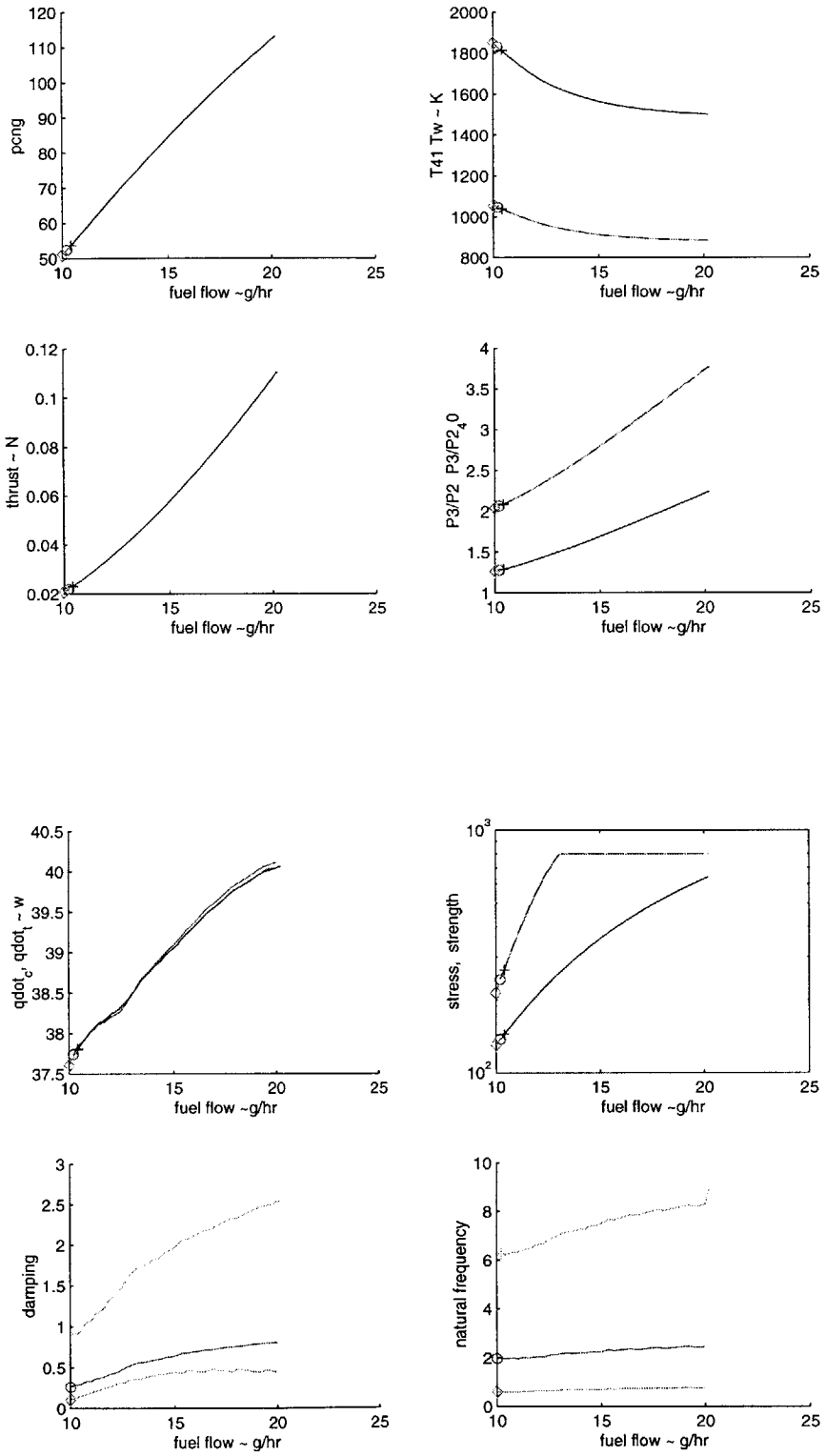
(a) Steady States - Change k_{qpi}



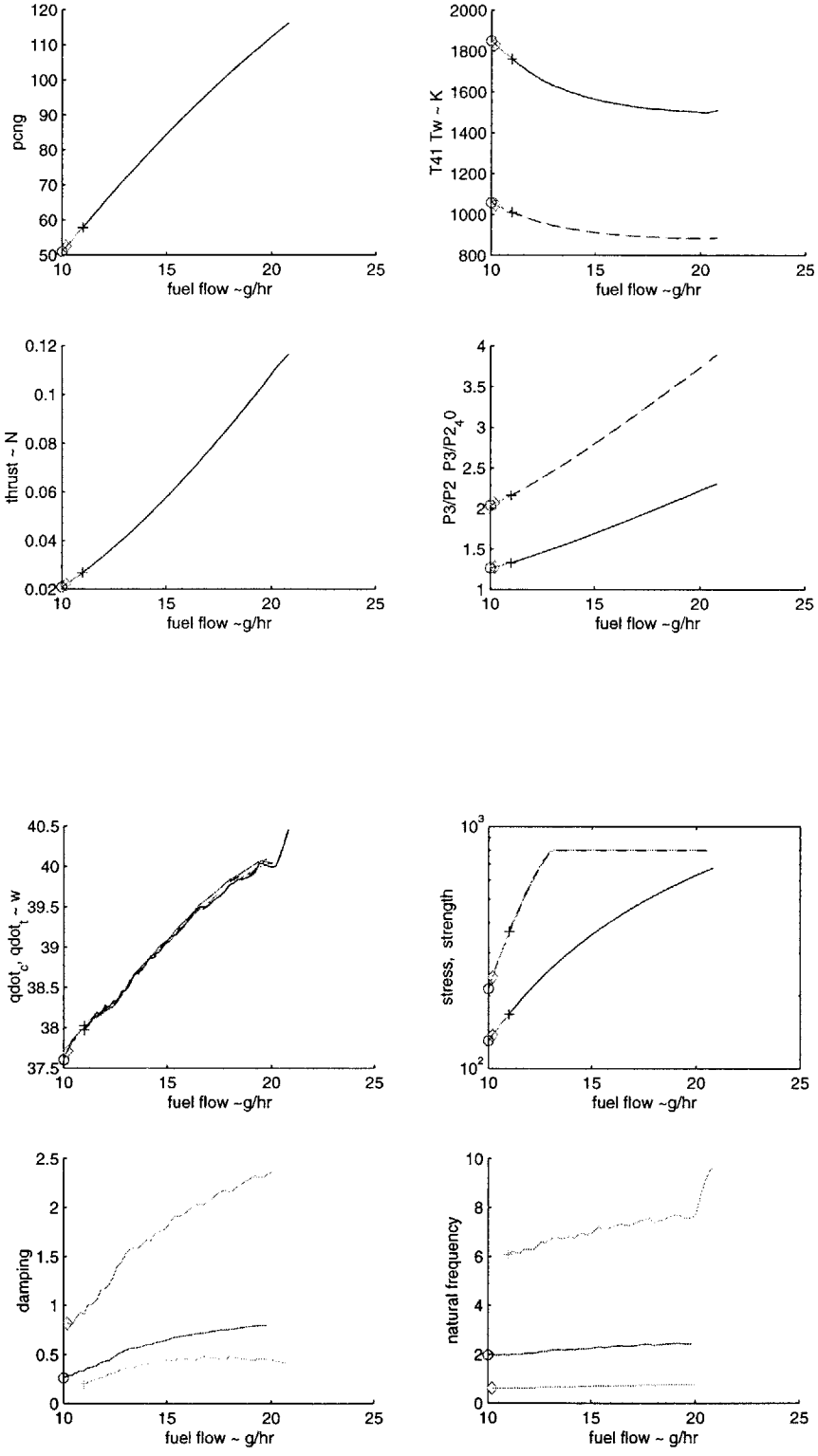
(b) Steady States - Change k_{qef}



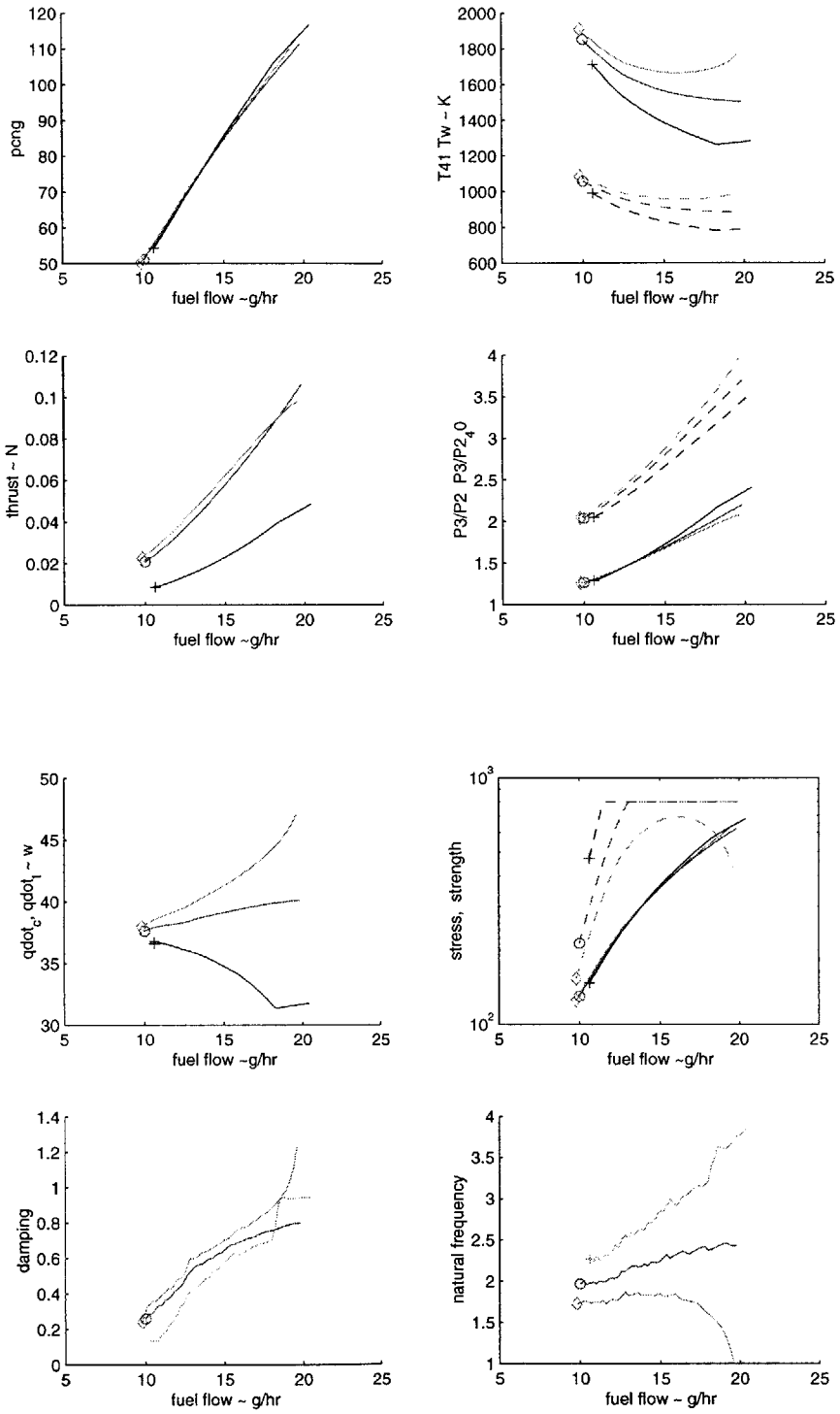
(c) Steady States - Change e_{peak}



(d) Steady States - Change k_{jg}



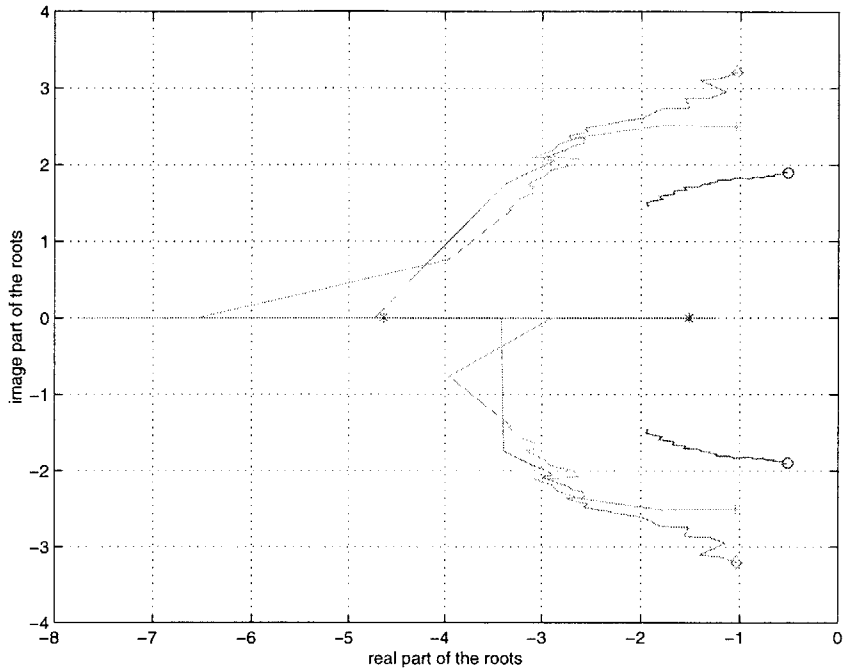
(e) Steady States - Change k_{cm}



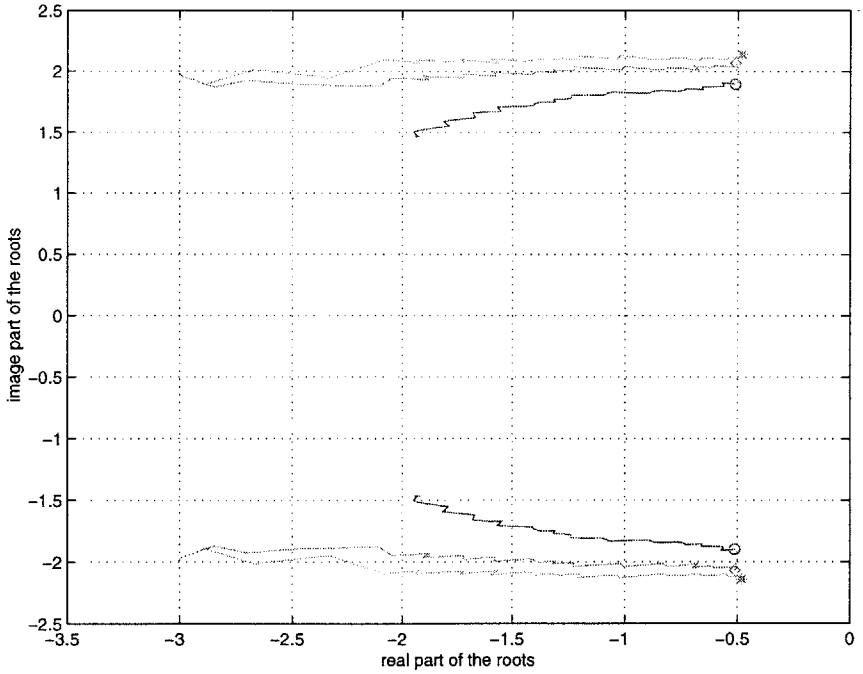
(f) Steady States - Change An

Figure 4-2: Steady States - Parameter Study

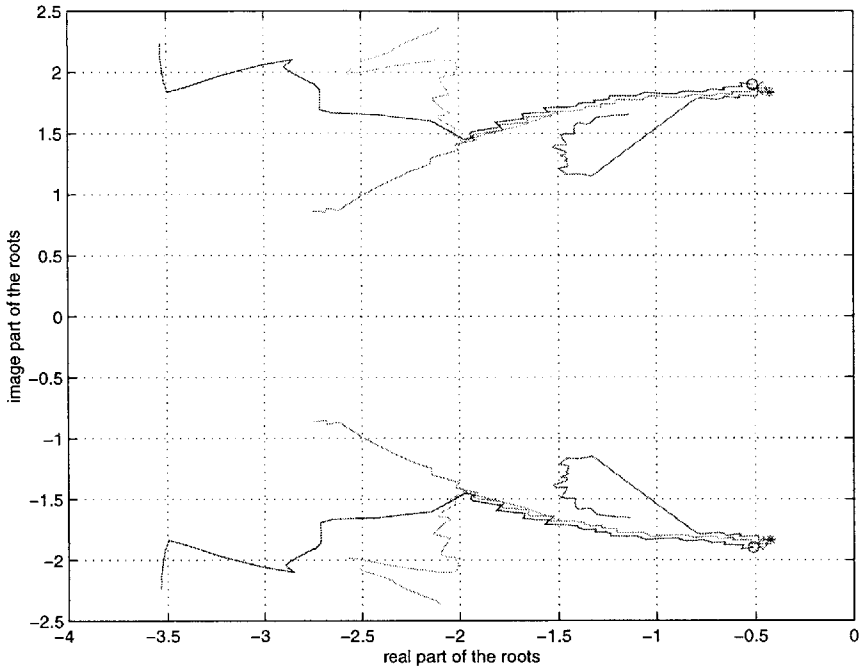
(see Table 4.1 for notations)



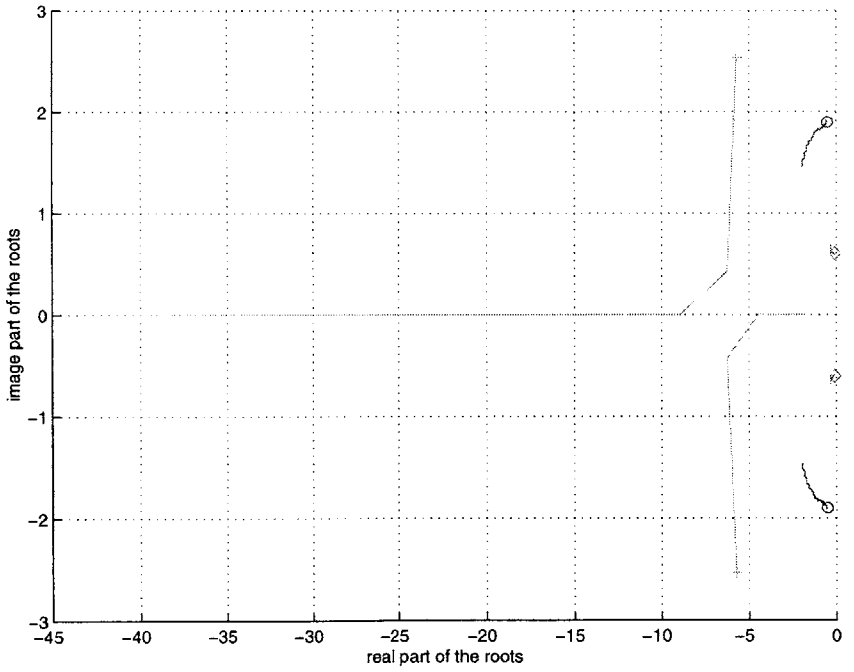
(a) Root Distribution - Change k_{qpi}



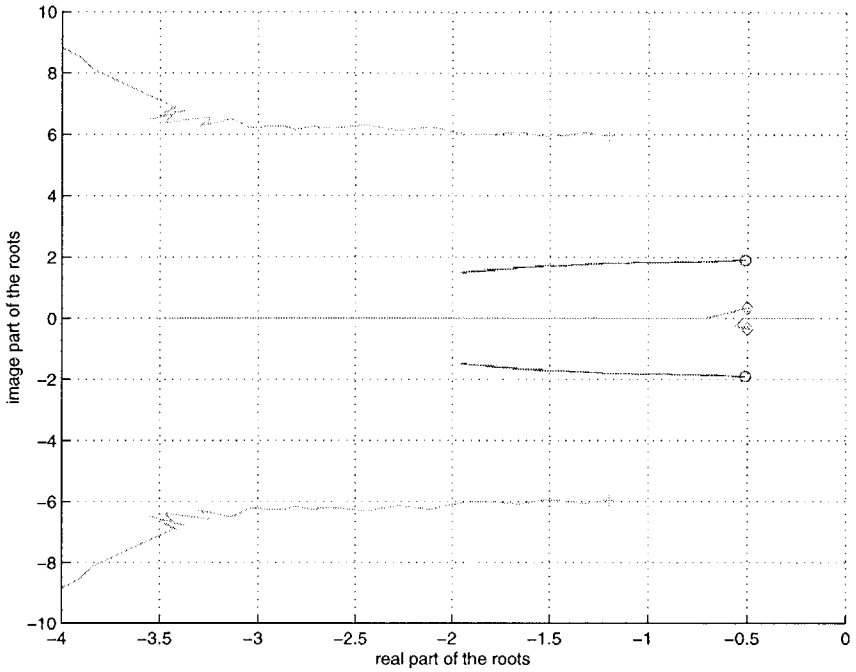
(b) Root Distribution - Change k_{qef}



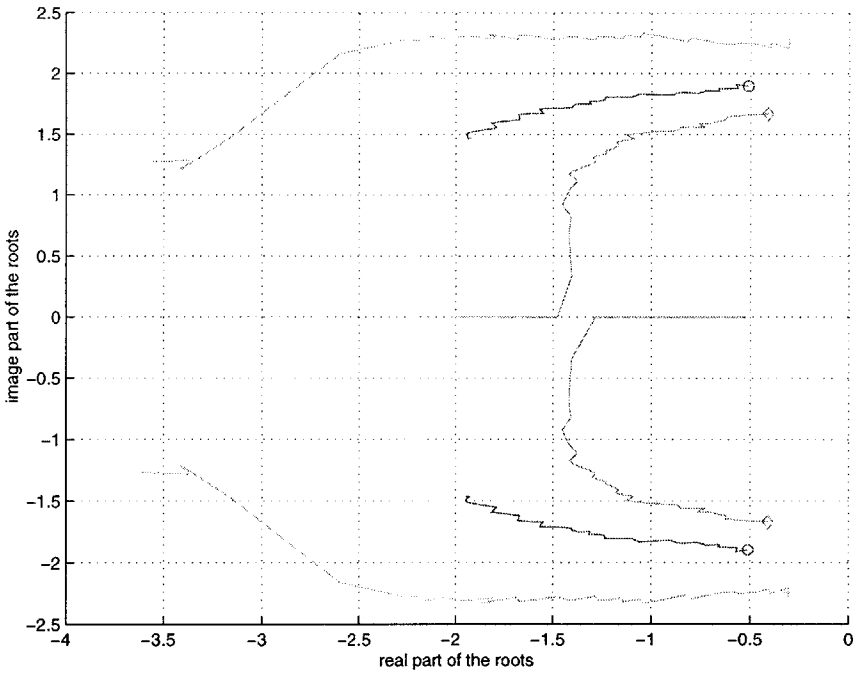
(c) Root Distribution - Change e_{peak}



(d) Root Distribution - Change k_{jg}



(e) Root Distribution - Change k_{cm}



(f) Root Distribution - Change An

Figure 4-3: Root Distribution - Parameter Study

(see Table 4.1 for notations)

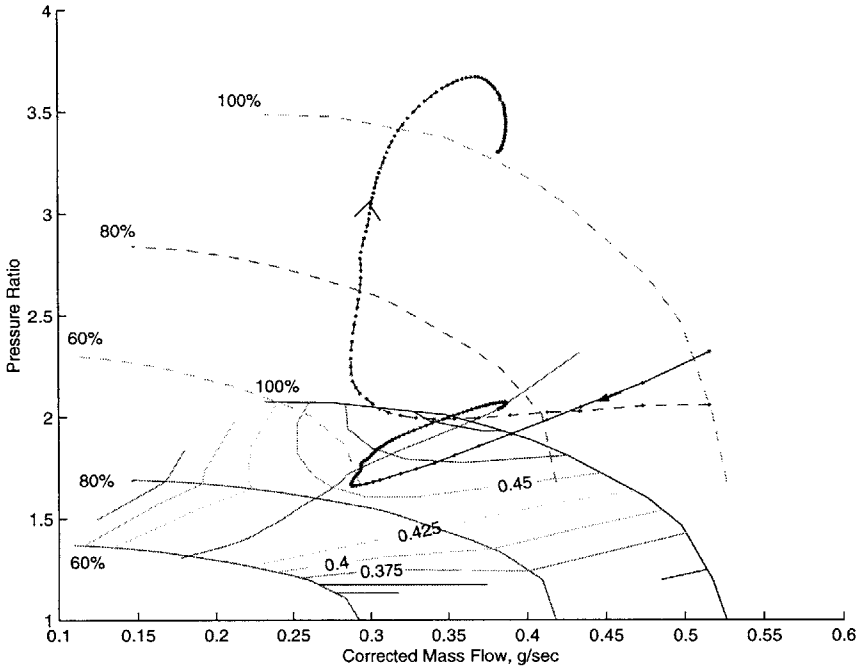


Figure 4-4: Startup Process on Compressor Map - Diabatic Experimental Compressor Map

(see Table 2.1 for line types)

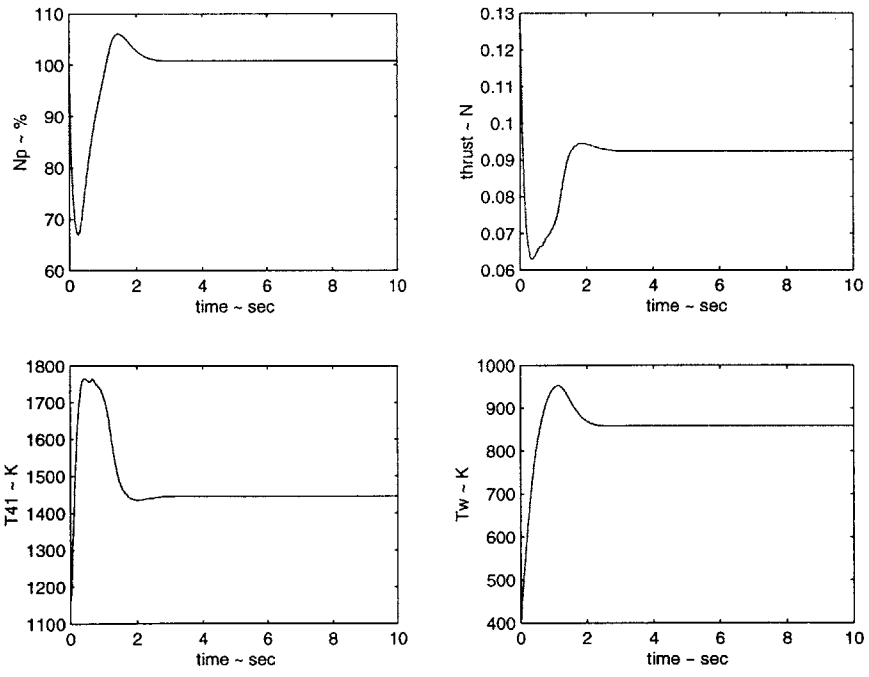


Figure 4-5: Time Domain Startup Process - Diabatic Experimental Compressor Map

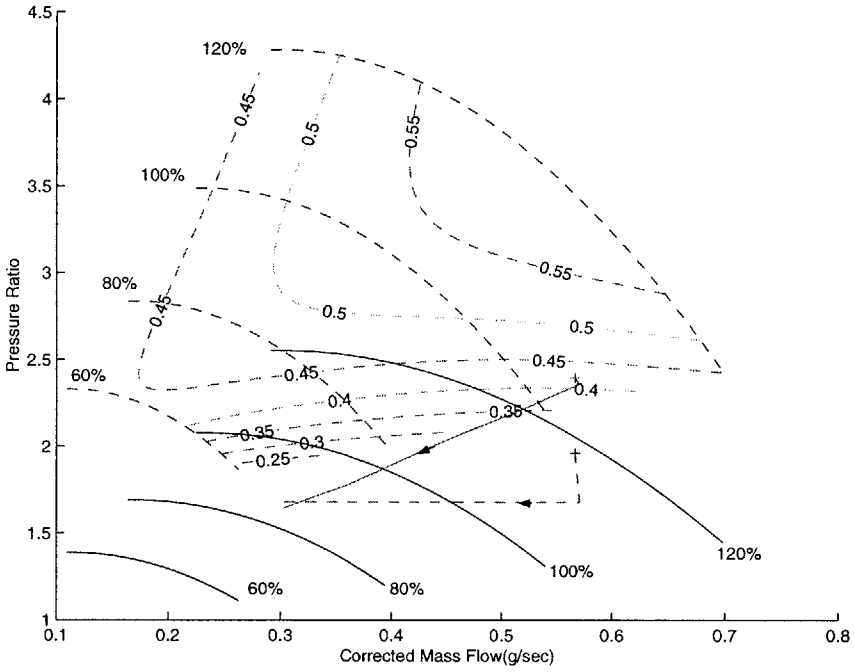


Figure 4-6: Startup Process on Compressor Map - Set Minimum Efficiency, on Modified Diabatic Experimental Compressor Map

(see Table 2.1 for line types)

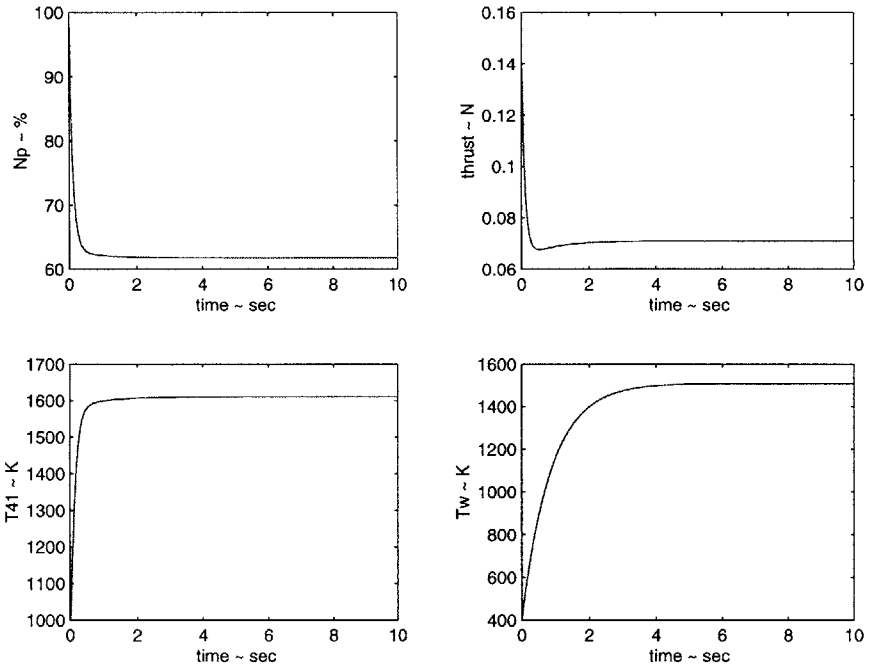


Figure 4-7: Time Domain Startup Process - Set Minimum Efficiency, on Modified Diabatic Experimental Compressor Map

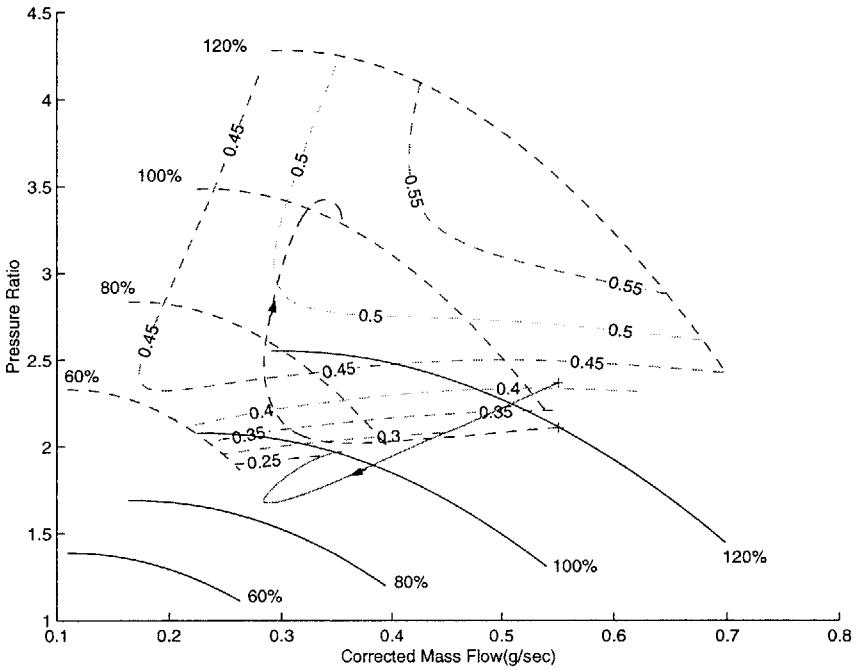


Figure 4-8: Startup Process on Compressor Map - Change Efficiency Gradient, on Modified Diabatic Experimental Compressor Map

(see Table 2.1 for line types)

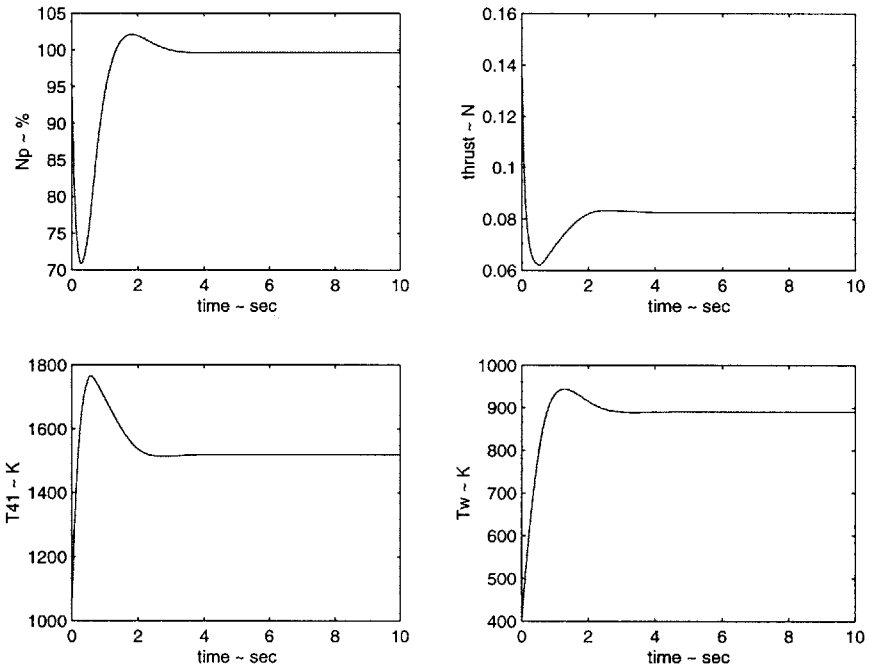


Figure 4-9: Time Domain Startup Process - Change Efficiency Gradient, on Modified Diabatic Experimental Compressor Map

Chapter 5

Conclusions and Future Work

5.1 Conclusions

This thesis develops microengine models, including both nonlinear models and linearized models. Based on these models, dynamic simulators for the microengine are developed. These simulators can be used to estimate the overall engine behavior, including both the steady state and transient behavior. They also provide platforms for further work, for instance to estimate the engine behavior when modifications are introduced to the engine design, to design the control law, etc. Simulation results for the microengine behavior under various situations using these simulators are also given.

Specifically, the following conclusions about the microengine can be drawn from the analysis, modeling and simulations:

- It is found by dynamics analysis that in the microengine, gas dynamics are much faster than the rotor dynamics and the heat transfer dynamics, and the rotor acceleration dynamics and the heat transfer dynamics are in the same range. On one hand, the much faster gas dynamics enable us to use quasi-steady models for the gas dynamics. Thus in the simulators, numerical approaches are applied to get the gas properties instead of integration, without compromising the properties of the engine behavior. On the other hand, the strong coupling between the rotor acceleration dynamics and the heat transfer dynamics makes the microengine system a second-order system, which is one of the main differences from a conventional engine. The second-order system makes the response of the engine more complex than a conventional engine. Considering the nonlinear character of engine transient operation, it becomes important to do simulations to predict the behavior of the engine during operation, especially when spool acceleration and deceleration occur, to avoid over-temperature, over-speed and surge.

  There are another dynamics which may need to be considered in the future. These are the dynamics of emptying the fuel tank. They are mainly dominated by the geometry of the fuel

tank. Currently these issues are disregarded.

- Simulations show that there is the possibility of increasing T_{41} with decreasing $\dot{m}_f$ in some regions, which is another main difference of the microengine from usual conventional engines. This is because for the microengine in some regions, the percentage decrease of $\dot{m}_f$ is less than the percentage decrease of $\dot{m}_3$.

- The linear properties of the microegine system are not good. This is evidenced by the comparisons of the responses of the nonlinear model and the linearized models, as well as the analysis of the locus of the eigenvalues at different operating points. One cause of the poor linear properties is the heat transfer; another cause is the unusual trend of T_{41} with decreasing $\dot{m}_f$. Therefore one must either design a potentially complex control system, or reduce heat transfer to the compressor to improve the open-loop system properties.

- Since the locus of linearized system eigenvalues shows large and fast variations of the damping and natural frequency, the microengine transient behavior is quite different at different operating points. In addition, the lowest damping can reach 0.15. The corresponding overshoot can be as large as 62%. Obviously, under some conditions it is possible to get over-temperature or over-speed unless precautions are taken. Moreover, the real part of the system eigenvalues varies from approximately 18.6% to 158.0% with respect to the real part at the design point. The corresponding settling time can be as large as approximate 5.38 times the settling time at the design point, or as small as 0.63 times the settling time at the design point, within the speed range of 58% to 116% of design speed. These large variations further complicate the analysis of the microengine behavior and the future design of the control system.

  On the other hand, the settling time is on the order of several hundred milliseconds. This is consistent with the previous dynamics analysis.

- The running line for the engine shows a relatively big surge margin (using the peak pressure ratio point as a reference). But accelerating the engine from minimum speed to 100% speed will still cause surge. This is consistent with the general behavior of second-order systems with small damping and large overshoot. Therefore, an engine of this type would have to incorporate a control system to avoid fast acceleration.

- Sensors will be used to determine the microengine operating point and to provide feedback for control purposes. For the first purpose, simulations show that although the temperature sensors have relatively fast response times, the same temperature can lead to multiple thrust, and thus speed, estimates. Pressure sensors, on the other hand, have relatively slow response time, but their change with thrust is monotonic. So the future choice of sensors may be some

combinations of the two. Also, some manipulations must be done to get good estimates of the engine states.

For feedback control purpose, the observability matrices of the linearized system model show that the system is observable by any one of the following parameters: rotation speed, T_3, T_{41}, T_{45}, P_3, P_{41}, and P_{45}. This makes the choice of sensors very flexible for feedback control purpose. Further study of the relative scale of the sensor response time and the system dynamics is needed.

- Parameter studies shows that the engine performance doesn't change significantly with changes in the coefficient which relates heat flux and compressor efficiency, but does change strongly with changes in the coefficient relating heat flux and compressor pressure ratio.

  Parameter studies also shows that changing the rotor inertia and thermal inertia doesn't effect the steady states, but strongly affect transients, as expected. This can in some cases even cause decoupling of the dynamics of rotor acceleration and heat transfer.

  Another results of parameter studies is the effect of the limited peak efficiency. The limited peak efficiency results in bifurcations of the running line, the steady states and the eigenvalue locus. The running line with lower peak efficiency bifurcates towards the surge margin when it achieves the peak efficiency contour. While the steady rotation speed doesn't change too much, the thrust is certainly lower for the same $\dot{m}_f$ when the peak efficiency is lower.

  It was also found in parameter studies that changing nozzle area doesn't cause large changes to either the steady states or the transient behaviors.

- The startup process was also investigated. There are several methods to make it works well: higher start speed, higher start spool temperature, and higher start fuel flow input. Simulations also show that changing the efficiency gradient changes the transient behavior of the engine significantly, thereby effecting the startup process. Because relatively accurate efficiency data are not currently available, the startup process needs some further study.

5.2 Recommendations of Future Work

The future work can be classified into two categories. One category is modeling and control. Another is for more general purpose.

For the purpose of modeling and control, the main future work will be in two areas. The first area is modifications of the current models when more accurate data is available, such as compressor data, turbine data, and fuel tank information. If there are no large changes to the current microengine design, the work to be performed will be mainly associated with more detailed study of the startup process.

The second area is control system design both for stabilizing the system at various operating points, and for ensuring smooth and stable acceleration, without over temperature and over speed. Simple scheduled control law will be first considered. If it doesn't work well, some control design based on the linearized models will be pursued. Because of the highly nonlinear properties of microengine, it is possible that the designed control laws based on the linearized models cannot satisfy the requirements. Then control design based on the nonlinear model will have to be performed.

For general purpose, the modeling and simulations show that most of the special characteristics of the microengine behavior arise from two unique characteristics. One characteristics is the heat transfer; another characteristics is the fact that T_{41} can increase with decreasing $\dot{m}_f$ in some regions. Let's consider a little more about these two issues.

First consider the heat transfer. This is one of the big differences between the microengine and a conventional engine. As indicated previously, the high thermal conductivity of silicon and the integral design of the spool lead to high and fast heat flux from the turbine side to the compressor side. It is this heat flux that can reduce the temperature of the turbine, thus allowing enough combustor exit temperature for self-sustained engine operation. But it is also this heat flux that leads to many undesirable microengine properties. In particular, the "fast" part of the heat flux makes the system become a second-order system, while the "high" part affects the compressor performance significantly: it lowers both the compressor pressure ratio and the compressor efficiency. This in turn affects the response of the whole microengine. In consequence, the microengine becomes a highly nonlinear system, which is close to a second-order system with poor linear properties, high possibility of surge, over-temperature and over-speed, and quite different behavior at different operating points. Obviously these properties are not desirable. They also imply the need for a potentially complex control system. Thus, finding a way to reduce the heat flux and at the same time allow enough combustor exit temperature for self-sustained engine operation would be highly desirable.

One possible way to do this may be introducing cooling air to the turbine blades. This will increase the thermal inertia of the spool. Simulations for this case are performed in chapter 4. As shown in the simulations and the dynamics analysis, once the thermal inertia is high enough compared with the spool mechanical inertia, the rotor acceleration dynamics and the heat transfer dynamics will be decoupled, and the effects of the heat flux to the compressor performance might be weakened. The system thus would become a first-order system. This could greatly improve the microengine open-loop characteristics, and simplify the analysis and the future design of the control system as well. Further studies could be performed on this subject.

Next consider the unusual trend of T_{41} with decreasing $\dot{m}_f$. There are two consequences of this trend. One is that it is one of the causes of the poor linear properties of the microengine. Another is that it leads to multiple thrust values corresponding to the same temperature, which makes the choice of the sensors for measuring the system more difficult. Thus finding a way to change this

unusual trend is also desirable. One possible way to do this may be modifying the compressor design to give relatively steep speed lines, which may lead to a relatively steep operating line. This steep operating line may result in less $\dot{m}_3$ drop with decreasing $\dot{m}_f$ (which is the cause of the unusual trend of T_{41} with decreasing $\dot{m}_f$), and thus eliminate the unusual trend of T_{41} with decreasing $\dot{m}_f$.

Appendix A

C Code for the S-function of the Engine Cycle Block

```
#include "mex.h"

/* C code S-function, engine_cycle block*/

static void engine_cycle (double *sim_par,double *ueng, double *yatmos,double
*yeng,double *rtn, double *addin, double *addout)
/*
------------ inputs -----------------------------------------

      *     simulation configuration parameters
      *     fuel flow at current time
      *     atmospheric parameters and airspeed for current time
      *     engine cycle parameters from last iteration or from trim

------------ outputs ----------------------------------------

      *     cycled parameters updated for current time step
      *     unbalanced torques for speed derivatives
      *     unbalanced heat flux for spool temperature derivatives

*/
{

#include "engine.h"
#include "heat1.h"
#include "heatroot_keep.h"

      double effc_t, fsurge;
      struct Meff meff;
      int i,j;

      double m2_45;
      double An, keffc, kpic, eff_peak, j_gt, CMs;

      double tmp1, tmp2, tmp3;
      double mach0, mach1, fmach;
      int imach, loop_mach;
      double kmach = 0.5;
      double Ac=1.2158/1000000; //from Yifang's CFD, mach=0.6 at design point
      int imach_max=1000;
      double mach_e=1e-4;


/* loop limit for finding qdot*/

      loop_inteval = 1000/step_inteval;
```

```
      loop_bisection = log(step_inteval/qdot_e)/log(2)+10;


/*    unpack the input vectors */

An   = addin[0];
keffc = addin[1];
kpic = addin[2];
eff_peak = addin[3];
j_gt = addin[4];
CMs  = addin[5];

i_sim_mode = (int)sim_par[0];
ii_max = (int)sim_par[1];
pdot_test = sim_par[2];
k_update = sim_par[3];


tw  = yeng[20];
qdotc = yeng[21];


p3  = yeng[0];
p41 = yeng[1];
p45 = yeng[2];
t41 = yeng[4];
pcng = yeng[9];
pcnp = yeng[10];


p3_no = p3/(1-kpic*qdotc);

/* fuel flow from the fuel pump is kg/hr, convert fuel flow to kg/sec */
wf = ueng[0]/3600.;

/*    unpack yatmos     */
alt   = yatmos[0];
delta_temp = yatmos[1];
Vrw = yatmos[2];

/* ------------------------------------------------------------- */
/*      environmental conditions          */
/* ------------------------------------------------------------- */

theta_amb_std = (1 - (6.875e-6 * alt));
delta_amb = pow(theta_amb_std, 5.256);
pamb = (delta_amb * pstd);
tamb = ((theta_amb_std * tstd) + delta_temp);
theta_amb = tamb/tstd;

/* Speed of sound and Mach number */

a = 1117.0 * sqrt(theta_amb);
mach = Vrw/a;

/*    total pressure at inlet */
p1 = pow((1.0 + 0.2*mach*mach), 3.5) * pamb;
```

```
t1 = (1.0 + 0.2*mach*mach)*tamb;

/*    static pressure at exhaust */
ps9 = pamb;

/* ------------------------------------------------------------- */
/*    inlet performance   */
/* ------------------------------------------------------------- */

p2 = p1;
t2 = t1;
h2 = kh2*t2;
del2 = p2/pstd;
thet2 = t2/tstd;
rtth2 = sqrt(thet2);

/* compute corrected speeds */

pcngc = pcng/rtth2;
pcnpc = pcnp/rtth2;

/* June 26, compute wr at different stage*/
ww = 0.01*pcng*ng_design*2*pi/60;
wr2 = ww*r2;
wr3 = ww*r3;
wr41 = ww*r41;
wr45 = ww*r45;
wr22 = wr2*wr2;
wr32 = wr3*wr3;
wr412 = wr41*wr41;
wr452 = wr45*wr45;


/*    Start iteration loop for gas dynamics   */

kount = 1;
do
      {

      p3_p = p3;
      p41_p = p41;
      p45_p = p45;

/* ------------------------------------------ */
/*       compressor                    */
/* ------------------------------------------ */

      ps3 = kps3*p3;
      ps3q2 = ps3/p2;


/*  July 5, solve for qdot_c */

      flag=0;
      qdot[0]=qdotc;
      loop_qdotc = 0;
      deltc = (pow(ps3q2,(gammac-1)/gammac) - 1)*t2;
```

```
      ps3q2_no = ps3q2/(1 - kpic*qdot[0]);
      ps3q2_t  = ps3q2_no*(1 - kpic_t*40);
      comp_map(ps3q2_t,pcng,&meff);
      wa2c = meff.x1;
      effc_t = meff.x2;
      fsurge = meff.x3;

      wa2 = wa2c*del2/rtth2;
      effc = effc_t*(1 - keffc*qdot[0])/(1 - keffc_t*40);

      delt2 = wr22/2/Cpc*(1-2*kt2*wa2/wr2);
      delt3 = wr32/2/Cpc*(1-2*kt3*wa2/wr3);
      delttmp = tw - t2 + (delt2+delt3)/2 ;

      fqdot[0] = khAc*sqrt(wa2)*(delttmp - deltc/effc/2 - qdot[0]/2/wa2/Cpc)
- qdot[0];

      i=0;
      j=0;

if (fabs(fqdot[0])>fqdot_e){

            do{

                  i++;

                  fqdot[1] = fqdot[0];
                   qdot[1]  = qdot[0];

                  if (fqdot[0]>0)         qdot[0] = qdot[0] + step_inteval;
                  else                    qdot[0] = qdot[0] - step_inteval;

                  ps3q2_no = ps3q2/(1 - kpic*qdot[0]);
                  ps3q2_t  = ps3q2_no*(1 - kpic_t*40);
                  comp_map(ps3q2_t,pcng,&meff);
                  wa2c = meff.x1;
                  effc_t = meff.x2;
                  fsurge = meff.x3;

                  wa2 = wa2c*del2/rtth2;
                  effc = effc_t*(1 - keffc*qdot[0])/(1 - keffc_t*40);

                  delt2 = wr22/2/Cpc*(1-2*kt2*wa2/wr2);
                  delt3 = wr32/2/Cpc*(1-2*kt3*wa2/wr3);
                  delttmp = tw - t2 + (delt2+delt3)/2 ;

                  fqdot[0] = khAc*sqrt(wa2)*(delttmp - deltc/2/effc -
qdot[0]/2/wa2/Cpc) - qdot[0];


                  if (fabs(fqdot[0])<fqdot_e)  {flag=1;break;}

                  if(i>loop_inteval)      {
                        printf("inteval is larger than 75w, qdot[0]=%lf",
qdot[0]);
                        break;
```

```
                }

        }while ((fqdot[0]*fqdot[1])>0);

    if (flag!=1){
        qdot[2] = qdot[0];

        do{
            j++;
            qdot[0] = (qdot[1]+qdot[2])/2;

            ps3q2_no = ps3q2/(1 - kpic*qdot[0]);
            ps3q2_t  = ps3q2_no*(1 - kpic_t*40);
            comp_map(ps3q2_t,pcng,&meff);
            wa2c = meff.x1;
            effc_t = meff.x2;
            fsurge = meff.x3;

            wa2 = wa2c*del2/rtth2;
            effc = effc_t*(1 - keffc*qdot[0])/(1 - keffc_t*40);

            delt2 = wr22/2/Cpc*(1-2*kt2*wa2/wr2);
            delt3 = wr32/2/Cpc*(1-2*kt3*wa2/wr3);
            delttmp = tw - t2 + (delt2+delt3)/2 ;

            fqdot[0] = khAc*sqrt(wa2)*(delttmp - deltc/2/effc -
qdot[0]/2/wa2/Cpc) - qdot[0];

            if (fabs(fqdot[0])<fqdot_e)  {flag=1;break;}

            if ((fqdot[0]*fqdot[1])>0)    qdot[1] = qdot[0];
            else                                qdot[2] = qdot[0];

            if(j>loop_bisection)    {
                printf("something is wrong with the code\n");
                flag=1;
                break;
            }

        }while (fabs(qdot[2]-qdot[1])>qdot_e);

        if (flag!=1)            qdot[0] = (qdot[1]+qdot[2])/2;

    }
}

    qdotc=qdot[0];
    fqdotc=fqdot[0];
    loop_qdotc=i+j;

    effc_no = effc_t/(1 - keffc_t*40);
    if (effc_no>eff_peak)  effc_no=eff_peak;
    if (effc_no<=eff_min)  effc_no=eff_min;
    effc = effc_no*(1 -keffc*qdotc);

    t3q2 = 1 + (pow(ps3q2,(gammac-1)/gammac) - 1)/effc;
    t3 = t3q2*t2 + qdotc/wa2/Cpc;
```

```
      h3 = kh3_1*t3 + kh3_2;

/*    seal pressurization and power turbine balance bleed */

      bl = 0;
      b2 = 0.05;
      wab24 = (bl + b2)*wa2;
      t24q2 = 1.15 + 0.039*ps3q2;
      t24 = t24q2*t2;
      h24 = 0.240*t24;
      wa3 = wa2 - wab24;

/* impeller tip leakage and turbine cooling bleed */

      b3 = 0;
      wa3bl =     wa2*(b3 + kb3);

/*    compressor discharge    */

      wa31 = wa3 - wa3bl;
      wa31_comb = sqrt((p3*(p3 - p41))/(kdpb*t3));

/* ------------------------------------------------------------- */
/*    combustor   */
/* ------------------------------------------------------------- */

/*    fuel to air ratio       */

      far41 = wf/wa31;

/* ------------------------------------------------------------- */
/*    gas turbine inlet */
/* ------------------------------------------------------------- */

      h41 = (h3 + effb*far41*hvf)/(1.0 + far41);
      t41 = kt41_1*h41 + kt41_2;

/*  Gas generator turbine enthalpy drop and flow */

      thta41 = kth41_1*t41 + kth41_2;
      rtth41 = sqrt(thta41);

      pr45q41 = p45/p41;

//May 3, use turbine map here to get mass flow

      del41 = p41/pstd;
      w41c = w41c_turb_map(pr45q41, pcng);
      w41_gt = w41c * del41/rtth41;

      coefp3 = 4.0*kdpb*t3*pow(wa31,2.0);
      coefp41 = t3*kdpb;
      w41 = wa31 + wf;

/*  June 26, solve for qdot_t*/

      delt41 = wr412/2/Cpt*(1-2*kt41*w41/wr41);
```

```
      delt45 = wr412/2/Cpt*(1-2*kt45*w41/wr41);
      delttmp = -tw + t41 - (delt41+delt45)/2;
      loop_qdott = 0;
      deltc = (1- pow(pr45q41,(gammat-1)/gammat)) * efft * t41;
      fqdott = khAt*sqrt(w41)*(delttmp - deltc/2);
      qdott = fqdott/(1 + khAt*sqrt(w41)/2/w41/Cpt);

      dh_gt = Cpt * t41 * (1 - pow(pr45q41,(gammat-1)/gammat)) * efft;

/* ------------------------------------------------------------- */
/*    Nozzle            */
/* ------------------------------------------------------------- */

      h45 = h41 - dh_gt - qdott/w41_gt;
      t45 = kt45_1*h45 + kt45_2;
      thta45 = kth45_1*t45 + kth45_2;

/*  total pressure at exhaust */

      p49qs9 = 1;
      p49 = p49qs9*ps9;
      ps9q45 = ps9/p45;

/* mass flow, volume 45 (time n) */

      w45 = w41 + kbl*wa3bl;
      coefp45 = w45*sqrt(thta45);

      m2_45 = (pow(ps9q45,-(gammat-1)/gammat)-1)/(gammat-1)*2;
      w45_pt = p45/sqrt(t45) * pow(1+(gammat-1)/2*m2_45,-
(gammat+1)/2/(gammat-1)) * sqrt(m2_45) * sqrt(gammat/R) * An ;
      del45 = p45/pstd;
      w45c = w45_pt/del45*sqrt(thta45);

      pr49q45 = p49/p45;
      dhptqth45 = 0;
      dh_pt = thta45*dhptqth45;
      h49 = h45 - dh_pt;
      t49 = kt49_1*h49 + kt49_2;

/* ------------------------------------------------------------- */
/*    mass flow errors at the three control volumes   */
/* ------------------------------------------------------------- */

      w3_err = wa31 - wa31_comb;
      w41_err = w41 - w41_gt;
      w45_err = w45 - w45_pt;

/* ------------------------------------------------------------- */
/* pressure dynamics at the three control volumes    */
/* ------------------------------------------------------------- */

      p3_dot = kv3*t3*w3_err;
      p41_dot = kv41*t41*w41_err;
      p45_dot = kv45*t45*w45_err;

/* ------------------------------------------------------------- */
```

```
/* update estimates of p3, p41, and p45 */
/* ------------------------------------------------------------ */

      deltp3 = (p41_p + sqrt( pow(p41_p, 2.0) + coefp3) )/2.0 - p3_p;
      deltp41 = p3_p -  coefp41*pow((w41c*p41_p/pstd/rtth41 - wf), 2.0)/p3_p
- p41_p;
      deltp45 = coefp45/(w45c/pstd) - p45_p;

      p3 = p3_p + k_update*deltp3*0.5;
      p41 = p41_p + k_update*deltp41*0.5;
      p45 = p45_p + k_update*deltp45*0.5;

/* exit iteration loop if all errors are less than pdot_test */

      pdot_error = fabs(p3_dot) + fabs(p41_dot) + fabs(p45_dot);

      if (pdot_error < pdot_test)
      {
            loop_count = 1.0*kount;
            kount = ii_max+1;
      }
      else
      {
            loop_count = 1.0*kount;
            kount++;
      }

}while (kount < ii_max);

/* ------------------------------------------------------------ */
/*    torque and heat flux calculations   */
/* ------------------------------------------------------------ */

k_torque = 60*cj/(2*pi);

/*    physical speeds */

ng =  0.01*pcng*ng_design;

/*    required compressor torque */

/* June 29, use t2, t41, pic, pit to compute torque, for heat flux*/
torque_c = (k_torque/ng) * wa2 * Cpc * t2 * (pow(p3/p2,(gammac-1)/gammac)-1)
/ effc;
torque_gt = (k_torque/ng) * (w41*Cpt*t41*(1-pow(p45/p41,(gammat-
1)/gammat))*efft - 13*pow(pcng*0.01,2));

/* horsepower calculation */

hpout = torque_pt*np/hprpm;
thrust = ps9 * (pow(p45/ps9,(gammat-1)/gammat)-1)*2*gammat/(gammat-1)*An;

/* Define yeng */

yeng[0] = p3;                  //printf("        p3 = %f\t\t", yeng[0]);
yeng[1] = p41;                 //printf("       p41 = %f\n",   yeng[1]);
yeng[2] = p45;                 //printf("       p45 = %f\t\t", yeng[2]);
```

```
yeng[3] = t3;                  //printf("        t3 = %f\n",   yeng[3]);
yeng[4] = t41;                 //printf("       t41 = %f\t\t", yeng[4]);
yeng[5] = t45;                 //printf("       t45 = %f\n",   yeng[5]);
yeng[6] = wa3;                 //printf("       wa3 = %f\t\t", yeng[6]);
yeng[7] = w41;                 //printf("       w41 = %f\n",   yeng[7]);
yeng[8] = far41;               //printf("       w45 = %f\t\t", yeng[8]);
yeng[9] = pcng;                //printf("      pcng = %f\n",   yeng[9]);
yeng[10] = wa2c;         //printf("      pcnp = %f\t\t", yeng[10]);
yeng[11] = w3_err;             //printf("    w3_err = %f\n",   yeng[11]);
yeng[12] = w41_err;            //printf("   w41_err = %f\t\t", yeng[12]);
yeng[13] = w45_err;            //printf("   w45_err = %f\n",   yeng[13]);
yeng[14] = torque_c;     //printf("  torque_c = %f\t\t", yeng[14]);
yeng[15] = torque_gt;    //printf(" torque_gt = %f\n",   yeng[15]);
yeng[16] = ng;                 //printf(" torque_pt = %f\t\t", yeng[16]);
yeng[17] = p3/p2;        //printf("    t41_ns = %f\n",   yeng[17]);
yeng[18] = thrust;             //printf("     hpout = %f\t\t", yeng[18]);
yeng[19] = loop_count;   //printf("loop_coutn = %f\n",   yeng[19]);

yeng[20] = tw;
yeng[21] = qdotc;
yeng[22] = qdott;
yeng[23] = ps3q2_no;;
yeng[24] = loop_qdotc;
yeng[25] = effc_no;
yeng[26] = effc;

rtn[0] = 60*100*(torque_gt - torque_c)/(2*pi*ng_design*j_gt);
rtn[1] = (qdott - qdotc)/CMs;

/* stress and strength calculation */

addout[0]=500*pcng*pcng*1e-4;    //stress
if (tw<=675)       addout[1]=2.9031;
else             addout[1]=6.3287 - 0.0051*tw;          //strength

/* Mach number at compressor */

tmp1 = wa2c / (p3/sqrt(t3)*Ac*sqrt(gammac/R));
tmp2 = (gammac-1)/2;
tmp3 = (gammac+1)/2/(gammac-1);
imach = 0;
mach0=0.5;
imach_max=1000;
kmach=0.5;
do{
      fmach = pow((1+tmp2*mach0*mach0),tmp3) * tmp1;
      mach1 = mach0 + kmach * (fmach-mach0);
      imach++;

      if(fabs(mach0-fmach) <= mach_e){
            loop_mach = imach;
            imach = imach_max+1;
      }
      else{
            loop_mach = imach;
            mach0=mach1;
      }
```

```
}while (imach < imach_max);

if (loop_mach >= imach_max)   printf("not solution for mach no");

addout[2]=mach0;          //mach no at compressor
addout[3]=loop_mach;      //loops for solving mach no
addout[4]=fsurge;

return ;

}
```

Bibliography

[1] *The MIT Microengine Project Annual Technical Report*, 1998.

[2] M.G. Ballin. A high fidelity real-time simulation of a small turboshaft engine. Technical report, NASA Technical Memorandum 100991, July 1988.

[3] C. Cadou and K. Khan. personal communication, 1999.

[4] K-S Chen. personal communication, 1999.

[5] E.M. Greitzer. The stability of pumping systems. *ASME Transactions, Journal of Fluids Engineering*, 103:193–242, 1981.

[6] S. Jacobson. personal communication, 1999.

[7] J. Kerrecrock. *Aircraft Engines and Gas Turbines.* the MIT Press, 1992.

[8] A. Mehra. Computational investigation and design of low reynolds number micro-turbomachinery. Master's thesis, Massachusetts Institute of Technology, 1997.

[9] J. Protz. personal communication, 1999.

[10] J. Protz. *The Design and Development of a Demonstration Micro Gas Turbine Engine.* PhD thesis, Massachusetts Institute of Technology, 2000.

[11] the Math Works Inc. *Simulink - Dynamic System Simulation for MATLAB.*

[12] J.H. Vincent. Real-time engine model, 1996.