



Development of Gas Turbine Performance Models using a Generic Simulation Tool

A. Alexiou

K. Mathioudakis

**Laboratory of Thermal Turbomachines
National Technical University of Athens**





Development of Gas Turbine Performance Models using a Generic Simulation Tool

§ Object-Oriented Gas Turbine Performance Modelling

§ Features of Simulation Tool

§ Modelling Methods

§ Application Example

§ Engine Dynamics

§ Frequency Response

§ Adaptation to Test Data

§ Conclusions



Object Oriented Gas Turbine Performance Modelling

Object-Oriented Programming (OOP) Features

§ Encapsulation

§ Inheritance

§ Abstraction

§ Polymorphism

§ Aggregation

OOP Advantages

§ Supports flexible and modular design

§ Facilitates code re-use

§ Makes code evolution and maintenance easier

§ Provides user-friendly interface



Object Oriented Gas Turbine Performance Modelling

GasTurb

Predefined gas turbine configurations

GSP

Only the developer can create new components

NPSS / Onyx

Restricted availability

Matlab-Simulink

Not fully OO

Use a commercially available general purpose OO simulation tool to model gas turbine components from which to build any engine configuration using a flexible and user-friendly interface



Development of Gas Turbine Performance Models using a Generic Simulation Tool

§ Object-Oriented Gas Turbine Performance Modelling

§ Features of Simulation Tool

§ Modelling Methods

§ Application Example

§ Engine Dynamics

§ Frequency Response

§ Adaptation to Test Data

§ Conclusions



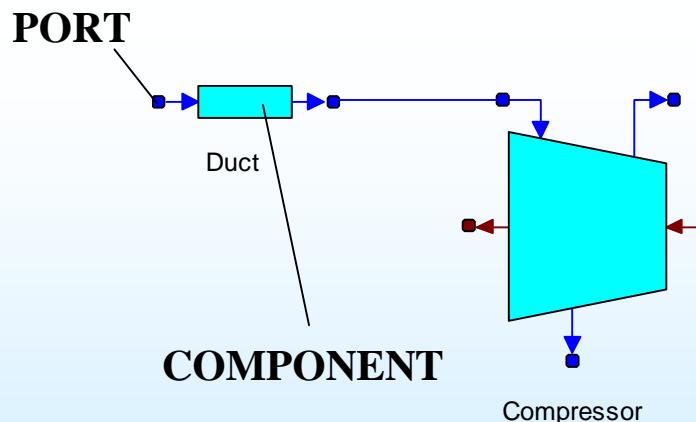
Features of Simulation Tool

- § Software **COMPONENT** = mathematical description of engine component
- § Components are joined together through their **PORTS**
- § PORTS define the set of variables transferred between connections

Component Code

```
COMPONENT Duct IS_A GasChannel
  DATA
    REAL K = 0      "Loss Factor"
    REAL q = 0.     "Heat flux (W)"
  CONTINUOUS
    H_out = H_in + q / g_in.W
    g_out.P = g_in.P * (1.0 - K * M_inl**2)
  END COMPONENT
```

Component Icon

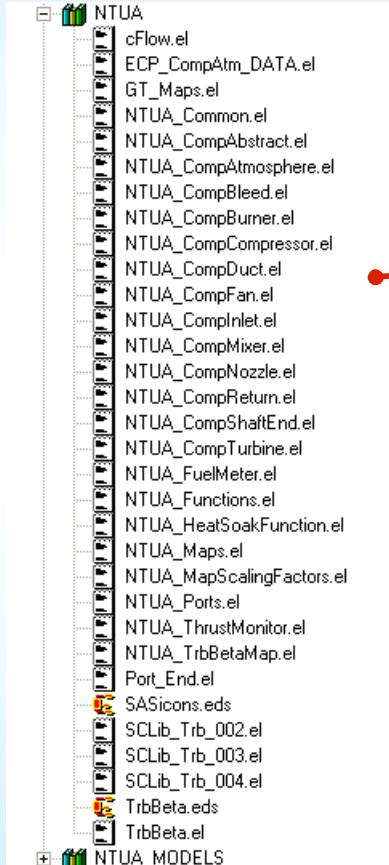




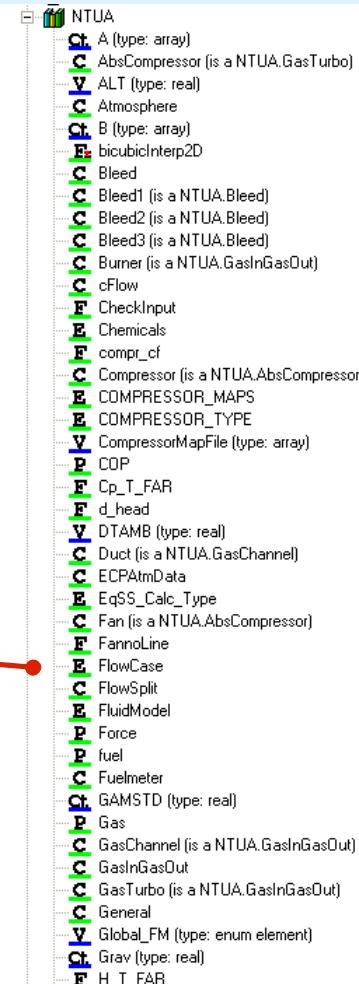
Features of Simulation Tool

COMPONENTS are stored in a **LIBRARY**

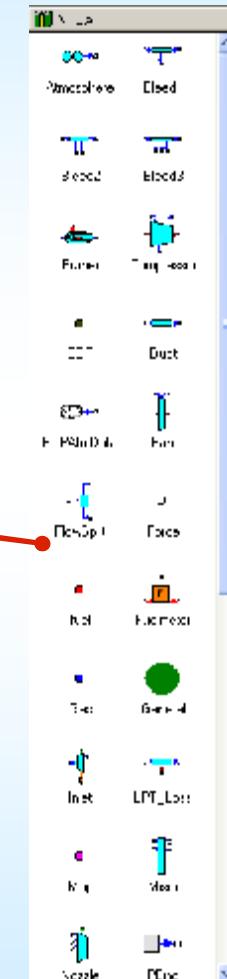
Library files



Library elements



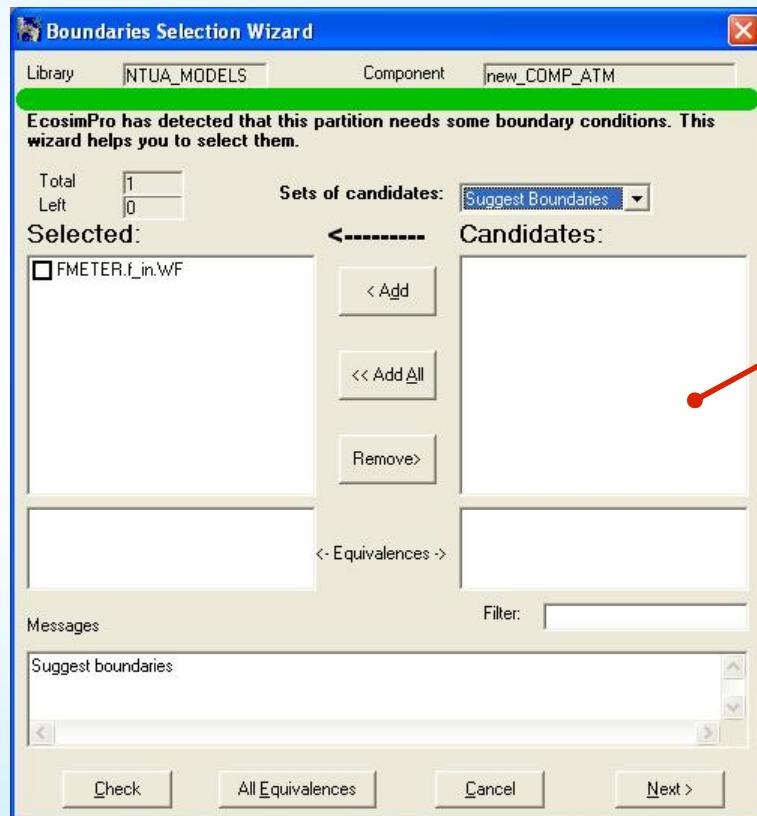
Library icons



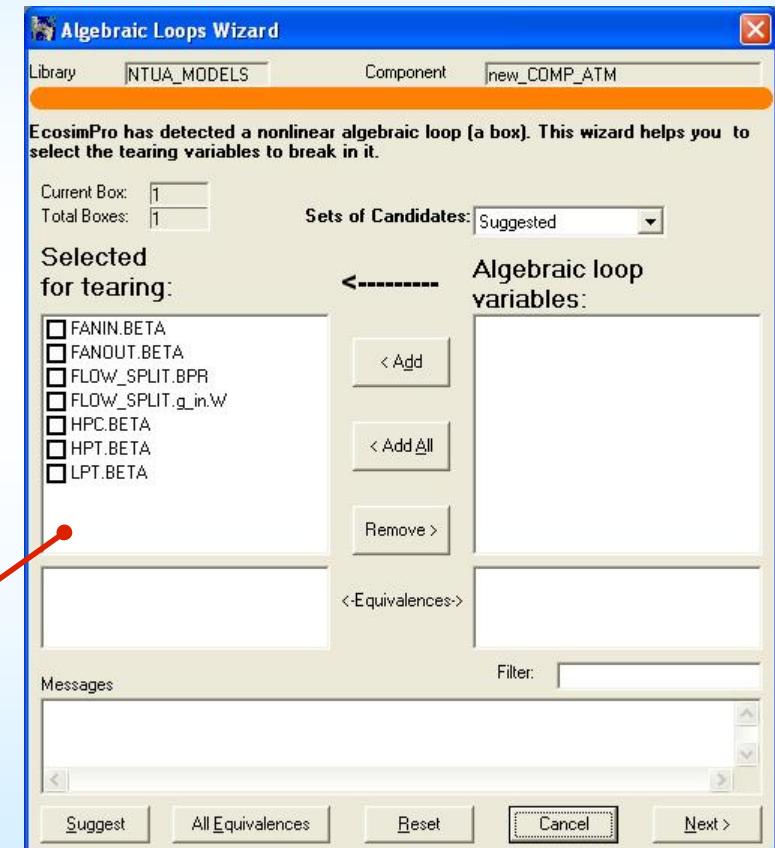


Features of Simulation Tool

A PARTITION is created to define the mathematical model



Selecting
Boundary
Conditions



Selecting
Algebraic
Variables



Features of Simulation Tool

```
EXPERIMENT ACCEL ON TURBOFAN_OBIDICOTE_MODEL.TESTS

DECLS
  TABLE_1D fuel_law

INIT  -- set initial values for variables
      -- Dynamic variables
  FANIN.XN = 1740
  HPC.XN = 9785
      -- Algebraic variables
  FANIN.BETA = 0.48
  FANOUT.BETA = 0.69
  FLOW_SPLIT.BPR = 6.0
  FLOW_SPLIT.g_in.W = 150
  HPC.BETA = 0.52
  HPT.BETA = 0.89
  LPT.BETA = 0.29

BOUNDS -- set expressions for boundary variables: v = f(t,...)
  NTUA__ALT = 0
  ATM.DTAMB = 0
  NTUA__RH = 0
  ATM.XM = 0

BODY
  REPORT_TABLE(***)
    EcoReadTable1D("C:\EcosimPro\USER_LIBS\NTUA_MODELS\EXP"
    FMETER.f_in.WF = 1.0 * timeTableInterp(TIME, fuel_law)

STEADY()
  INTEG_TO(9,0.02)
END EXPERIMENT
```

Different EXPERIMENTS can be made for a PARTITION

Initial Values for Dynamic & Algebraic Variables

Boundary Condition Values

Reporting

Steady State Calculation

Integration



Main Window of Simulation Tool

The screenshot shows the EcosimPro 3.3.16 software interface. The window title is "EcoStudio for EcosimPro 3.3.16 [C:\EcosimPro\USER_1\BSV\DEFAULT_LIB\WORK\aircraftGear.e]". The menu bar includes File, Edit, View, Library, Experiment, Window, Help. The left pane is labeled "Workspace Area" and contains a tree view of project files under "STANDARD":

- CONTROL
- DEFAULT_LIB
 - aircraftGear.el
 - bouncingBall.el
 - circuit.eds
 - circuit.el
 - circuit.igr
 - equation.el
 - freezer.el
 - lorentz.el
 - waveFunctions.eds
 - waveFunctions.el
 - waveFunctions.igr
 - whenExample.el
 - zoneExample.el
- ECP
- ELECTRIC
- GAS_TURBINE
- MATH
- NTUA
- NTUA_MODELS
- SCLIB
- THERMAL

The right pane is labeled "Editing Area" and displays the code for the "aircraftGear" component:

```
COMPONENT aircraftGear "Zone Test (aircraftGear)"

DATA
    REAL m1 = 2042.6      "aircraft mass (Kg)"
    REAL m2 = 660.6 "carriage mass (Kg)"
    REAL m3 = 291.8 "squeezor mass (Kg)"
    REAL k1 = 6170 "spring coefficient 1 (N/m)"
    REAL k2 = 34310 "spring coefficient 2 (N/m)"
    REAL h = 38.1 "horizontal length of cable (m)"

DECLS
    TABLE_1D tab= { 0, 9.14, 18.29, 36.58, 45.72, 54.86, 64.01, 73.15, 82.30, 85.95, 89.61, 93.27, 95.10, 98.76 },
                  { 398.56, 191.39, 76.56, 248.80, 248.80, 315.79, 397.13, 511.96, 765.55, 1004.78, 1339.71, 1961.72, 2392.
    REAL y1, y2, y3 "displacements (m)"
    REAL x "horizontal distance aircraft arrester cable (m)"
    REAL fk1, fk2 "cable tensions (N)"
    REAL sintheta "sin of angle of cable under tension ()"
    REAL fdrag "drag force (N)"
    REAL temp "drag coefficient (N*s**2/m**2)"

CONTINUOUS
    ... Geometry
    y1 = sqrt(x**2 + h**2) - h
    sintheta = x/(h+y1)

    ... Calculates drag coefficient from the table
    temp = linearInterp1D(tab,y3)

    ... drag force
    fdrag = temp * y3**2

    ... cable tensions
    fk1 = ZONE (y1 > 2 * y2) k1 * (y1 - 2 * y2)
    OTHERS 0
    fk2 = ZONE (y2 > y3) k2 * (y2 - y3)
    OTHERS 0

    ... acceleration of masses
```

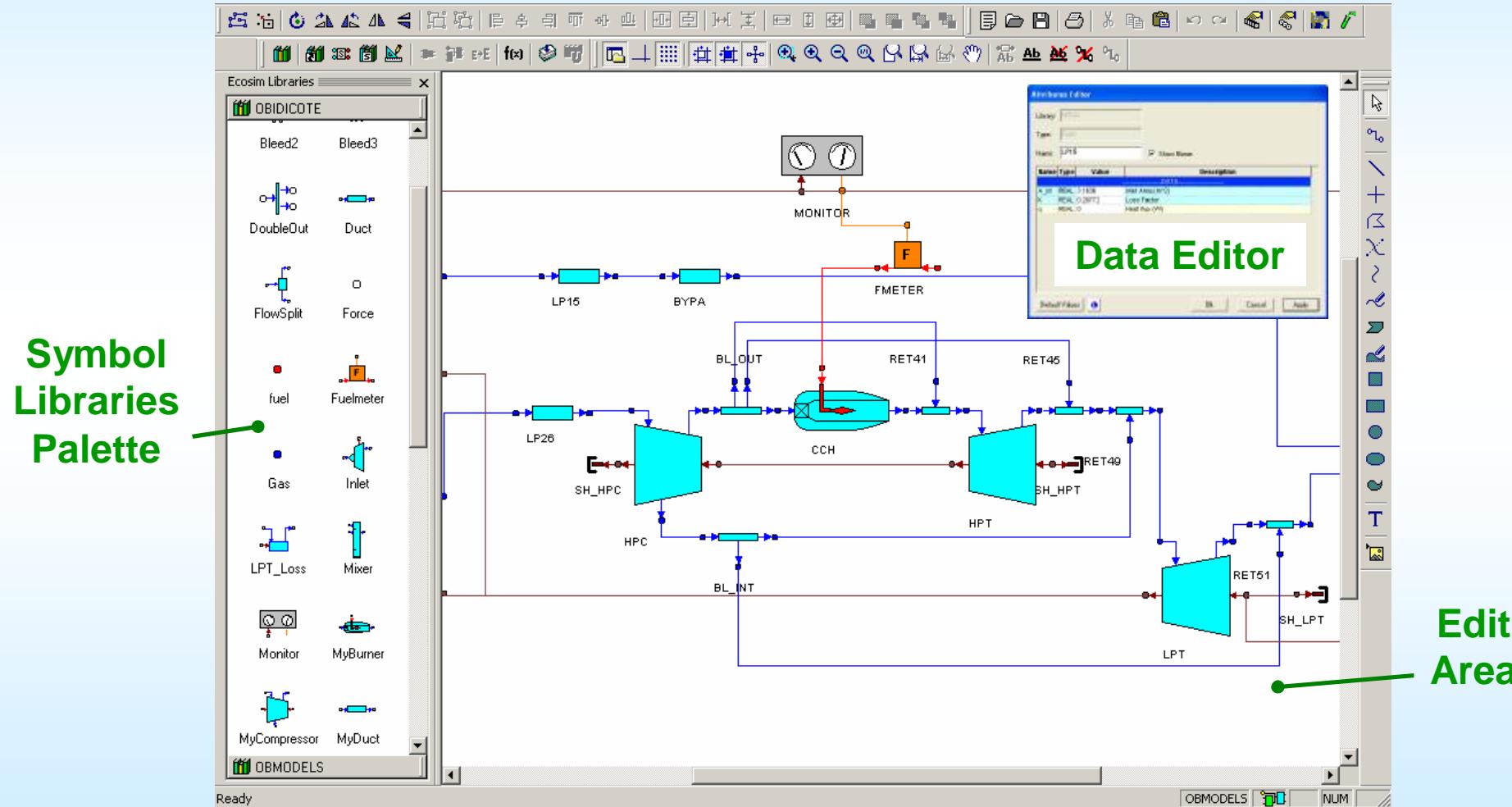
The bottom status bar says "EcosimPro Ready".

Output
Area

Editing
Area

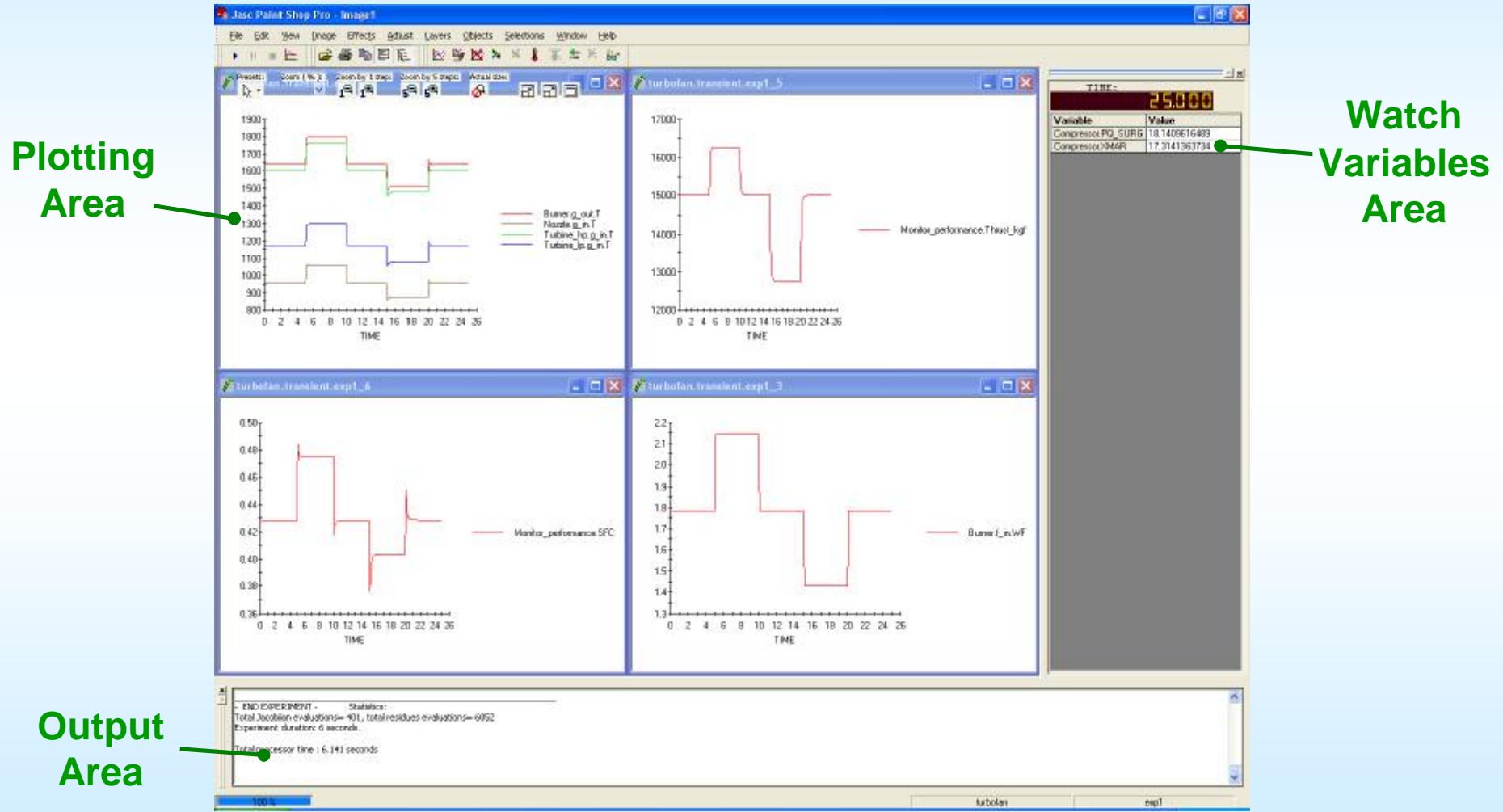


Graphical Component Creation





Graphical Output from Simulation





Development of Gas Turbine Performance Models using a Generic Simulation Tool

§ Object-Oriented Gas Turbine Performance Modelling

§ Features of Simulation Tool

§ Modelling Methods

§ Application Example

§ Engine Dynamics

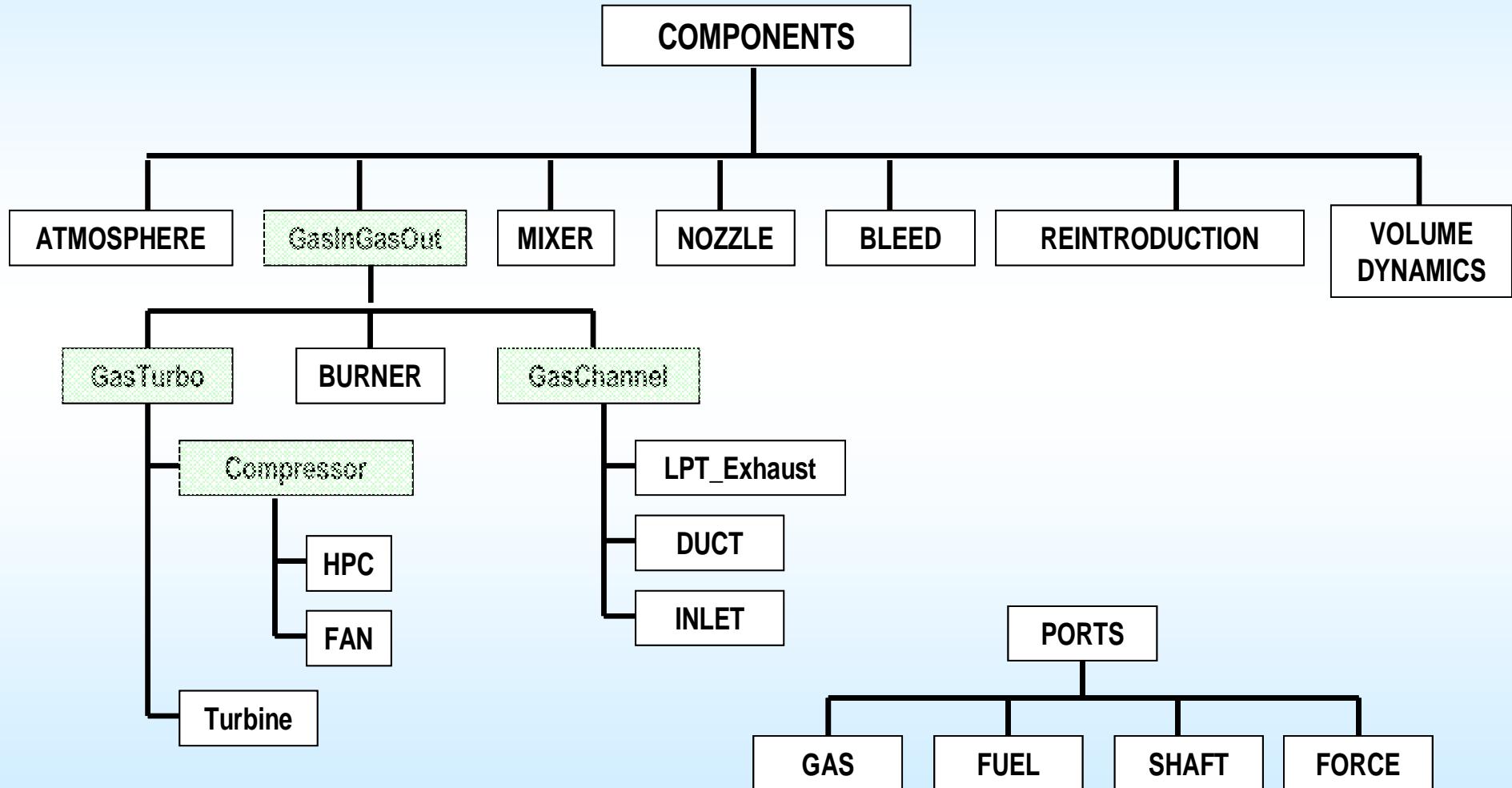
§ Frequency Response

§ Adaptation to Test Data

§ Conclusions



Modelling Methods: Component Hierarchy





Modelling Methods: Dynamic Modelling

$$\Delta PW = J * XN * \frac{dXN}{dt} * \left(\frac{2\pi}{60} \right)^2$$

Shaft Dynamics

$$h = 0.0201 * C * Re^{0.8} * K / L$$

$$c_p * M * \frac{dT_m}{dt} = h * A_s * (T_g - T_m)$$

Heat Soakage

- HP compressor (blades and casing)
- Combustion (casing)
- HP turbine (blades and casing)

$$\frac{d\rho}{dt} = \frac{W_{in} - W_{out}}{\text{Volume}}$$

$$\frac{dw}{dt} = \frac{1}{L} * [(W * V + p * A)_{in} - (W * V + p * A)_{out} + F_{body}]$$

$$\frac{d}{dt}(\rho * H - p) = \frac{1}{\text{Volume}} * [(W * H)_{in} - (W * H)_{out} + PW + Q]$$

Gas Dynamics:

- Conservation of Mass
- Conservation of Momentum
- Conservation of Energy



Development of Gas Turbine Performance Models using a Generic Simulation Tool

§ Object-Oriented Gas Turbine Performance Modelling

§ Features of Simulation Tool

§ Modelling Methods

§ Application Example

§ Engine Dynamics

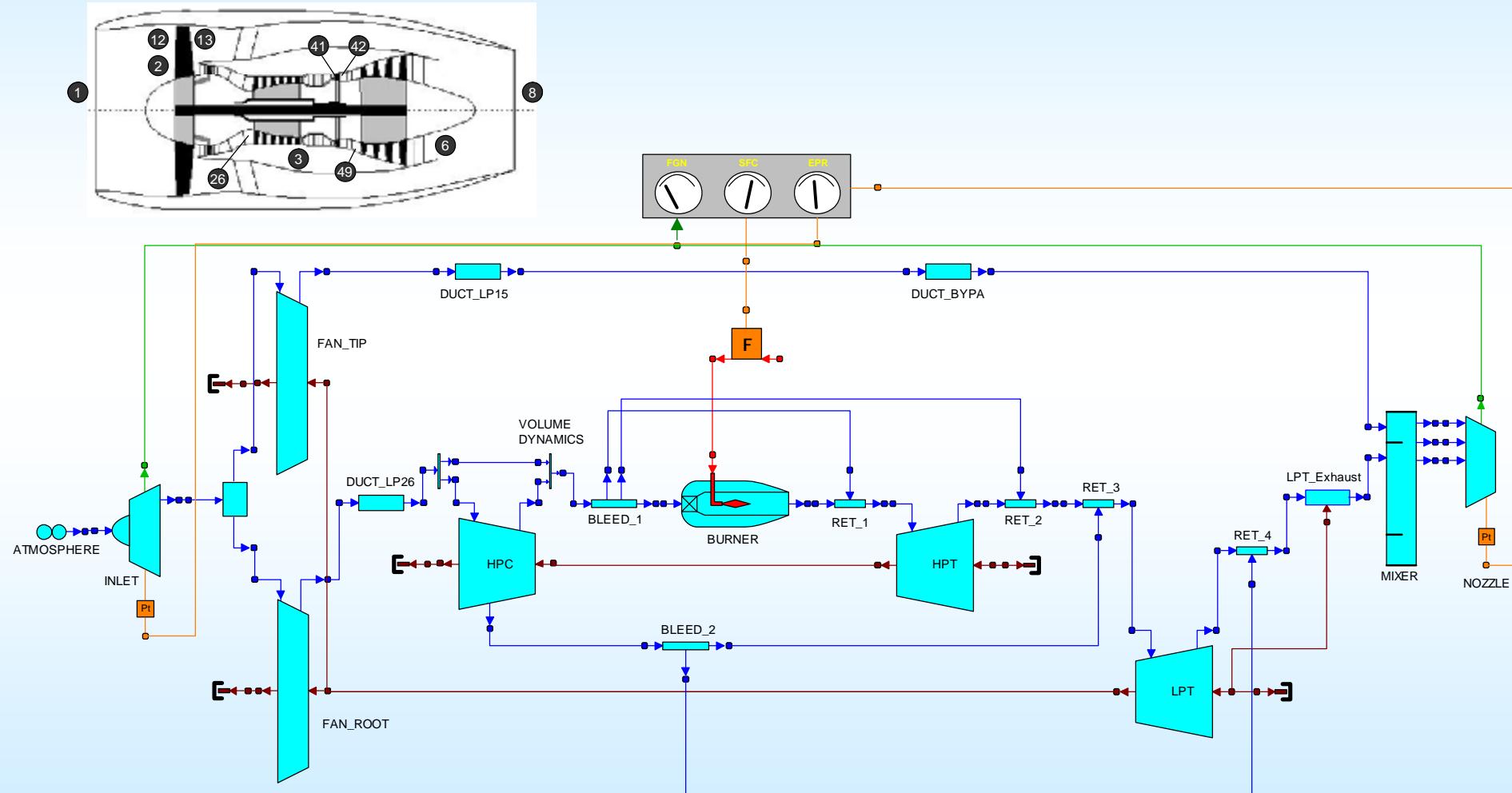
§ Frequency Response

§ Adaptation to Test Data

§ Conclusions

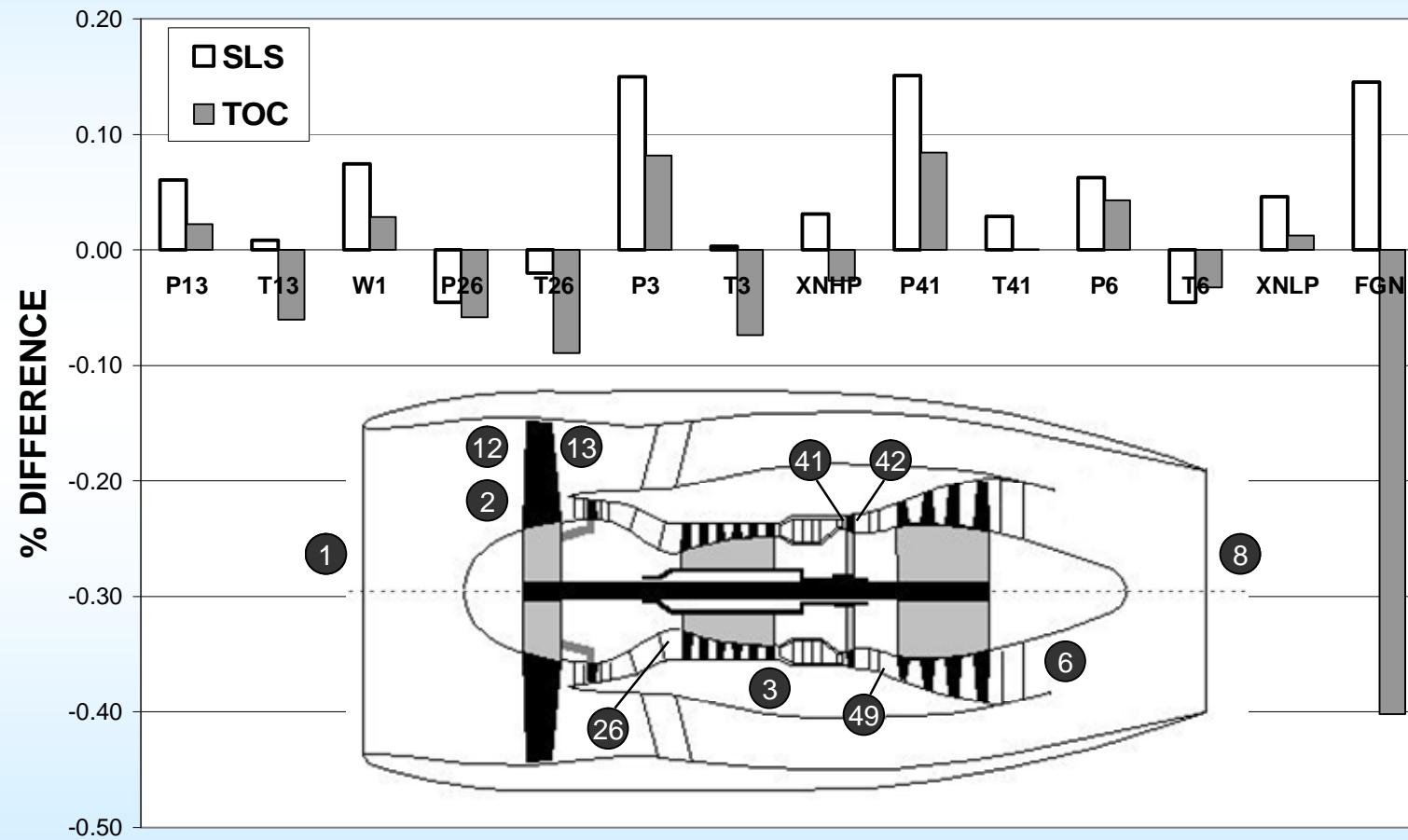


Specific Application Case: Turbofan Model



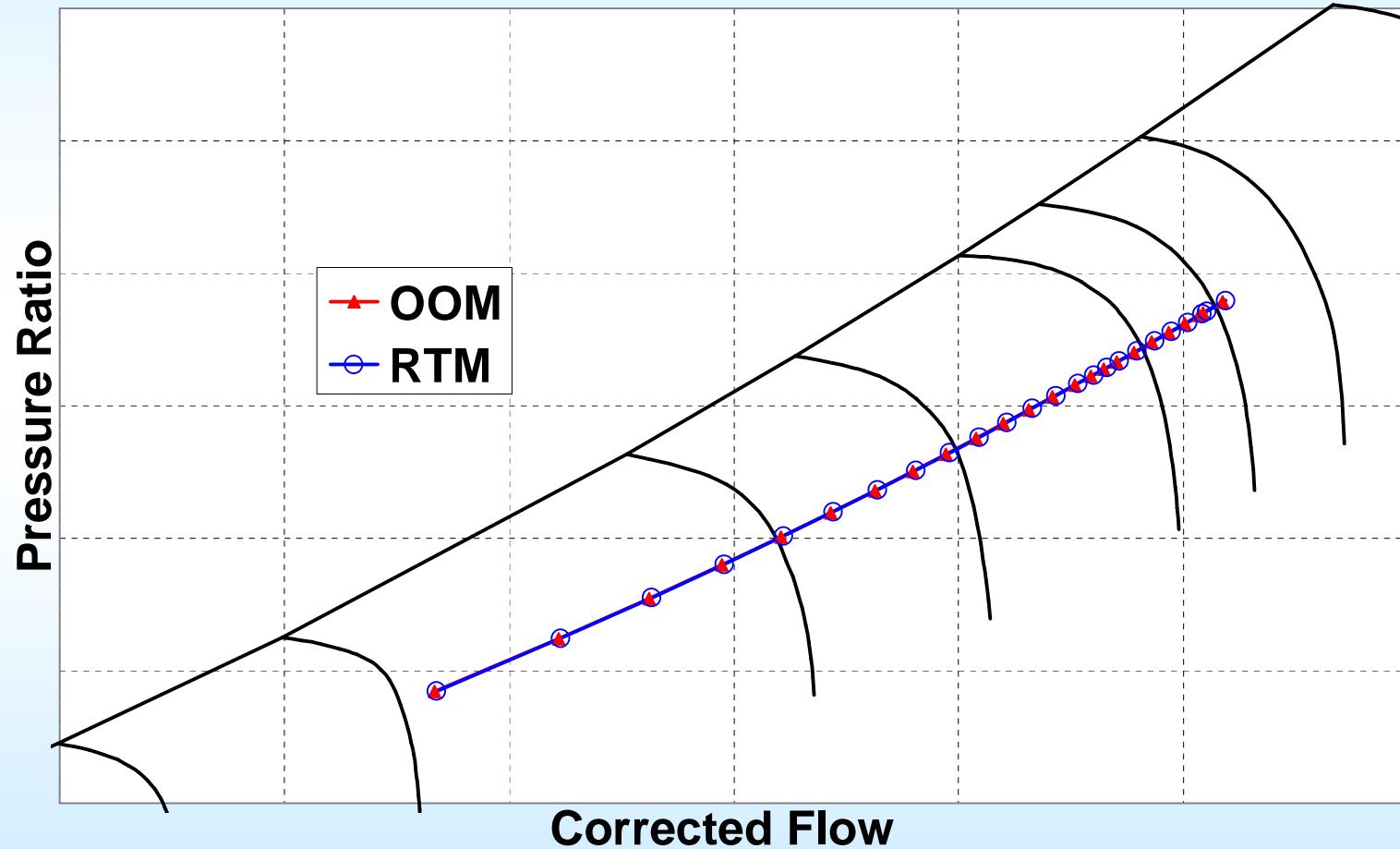


Model Comparison for SLS and TOC Cases



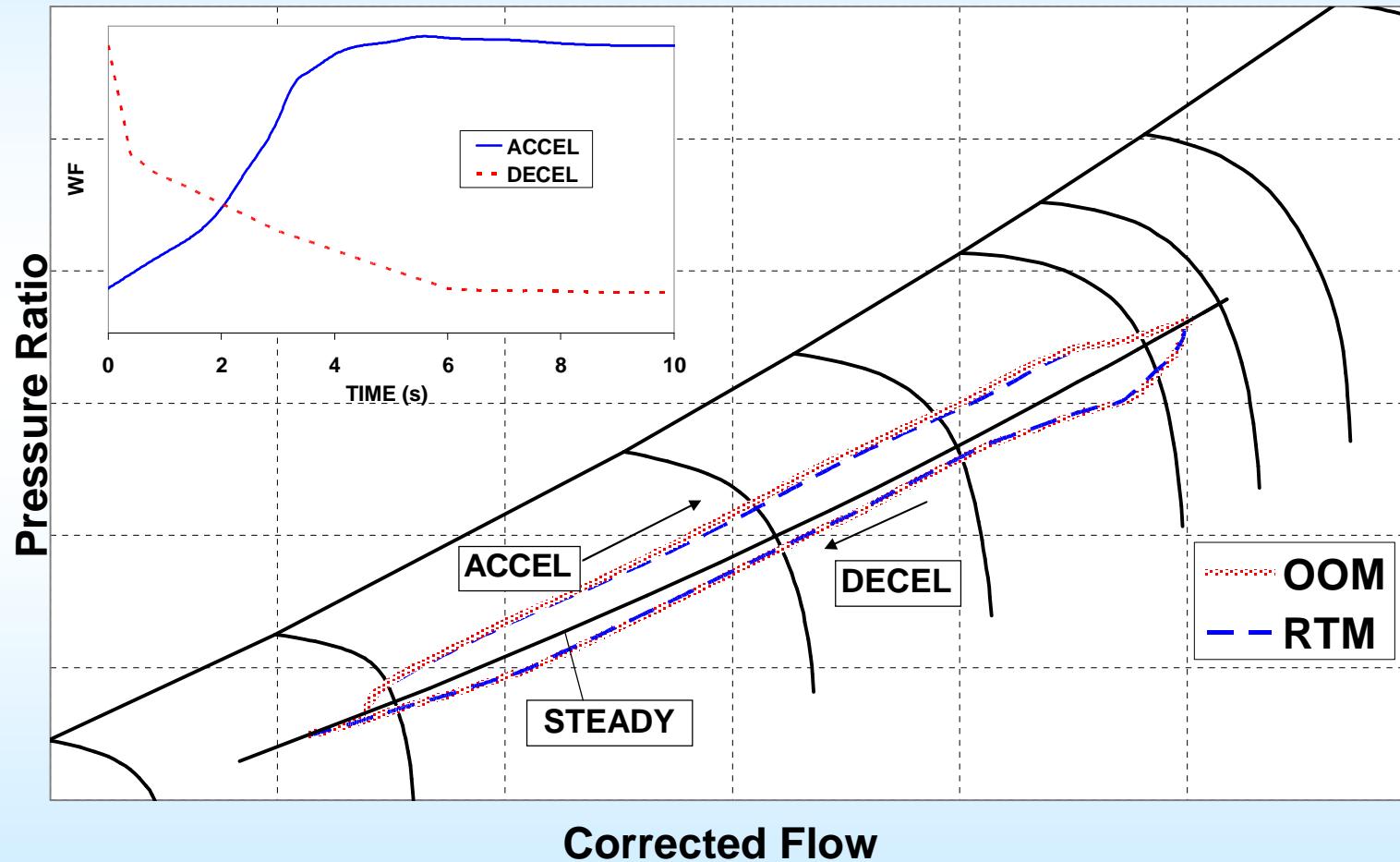


Model Comparison for Steady State Cases at SLS





Model Comparison for Transient Cases





Development of Gas Turbine Performance Models using a Generic Simulation Tool

§ Object-Oriented Gas Turbine Performance Modelling

§ Features of Simulation Tool

§ Modelling Methods

§ Application Example

§ Engine Dynamics

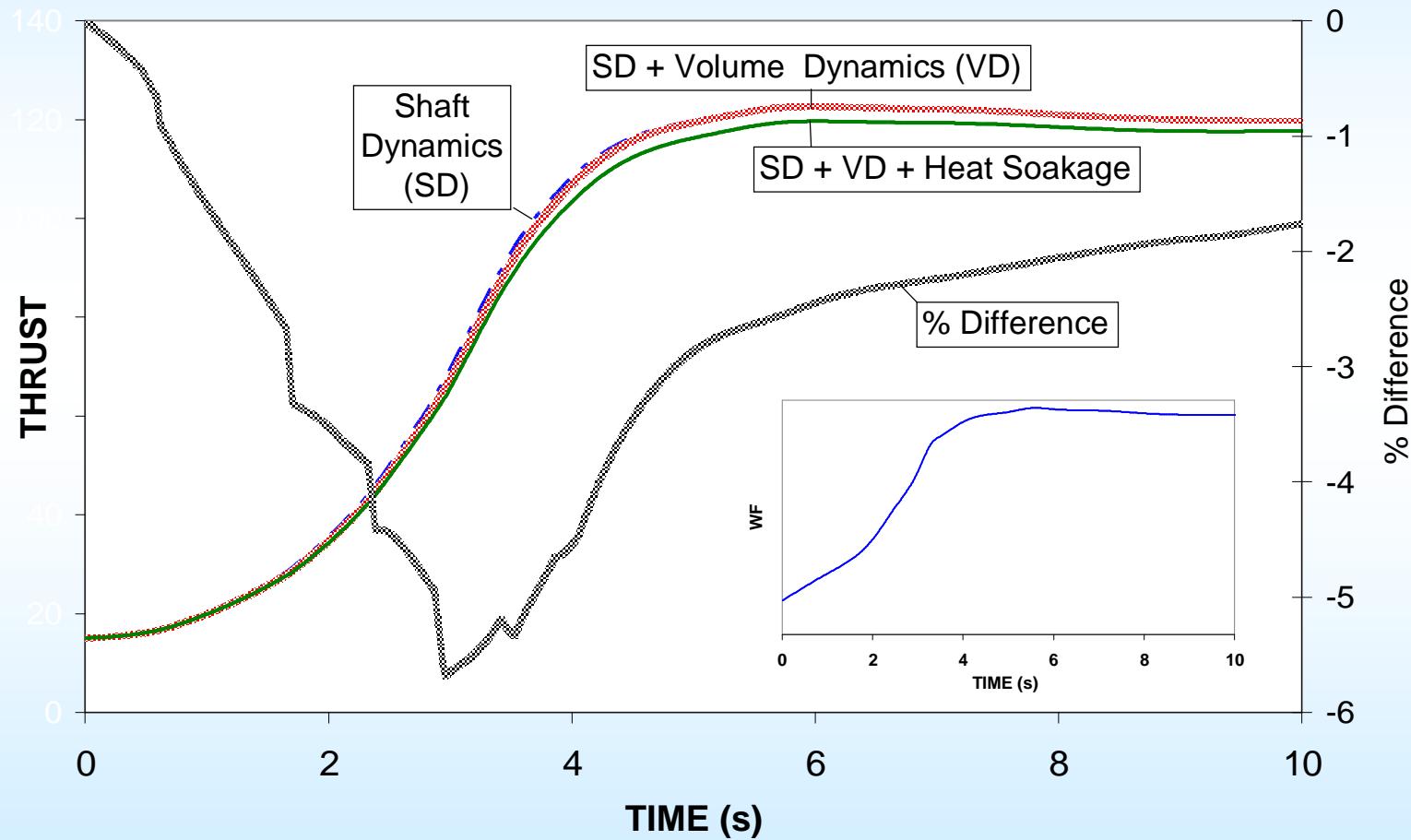
§ Frequency Response

§ Adaptation to Test Data

§ Conclusions

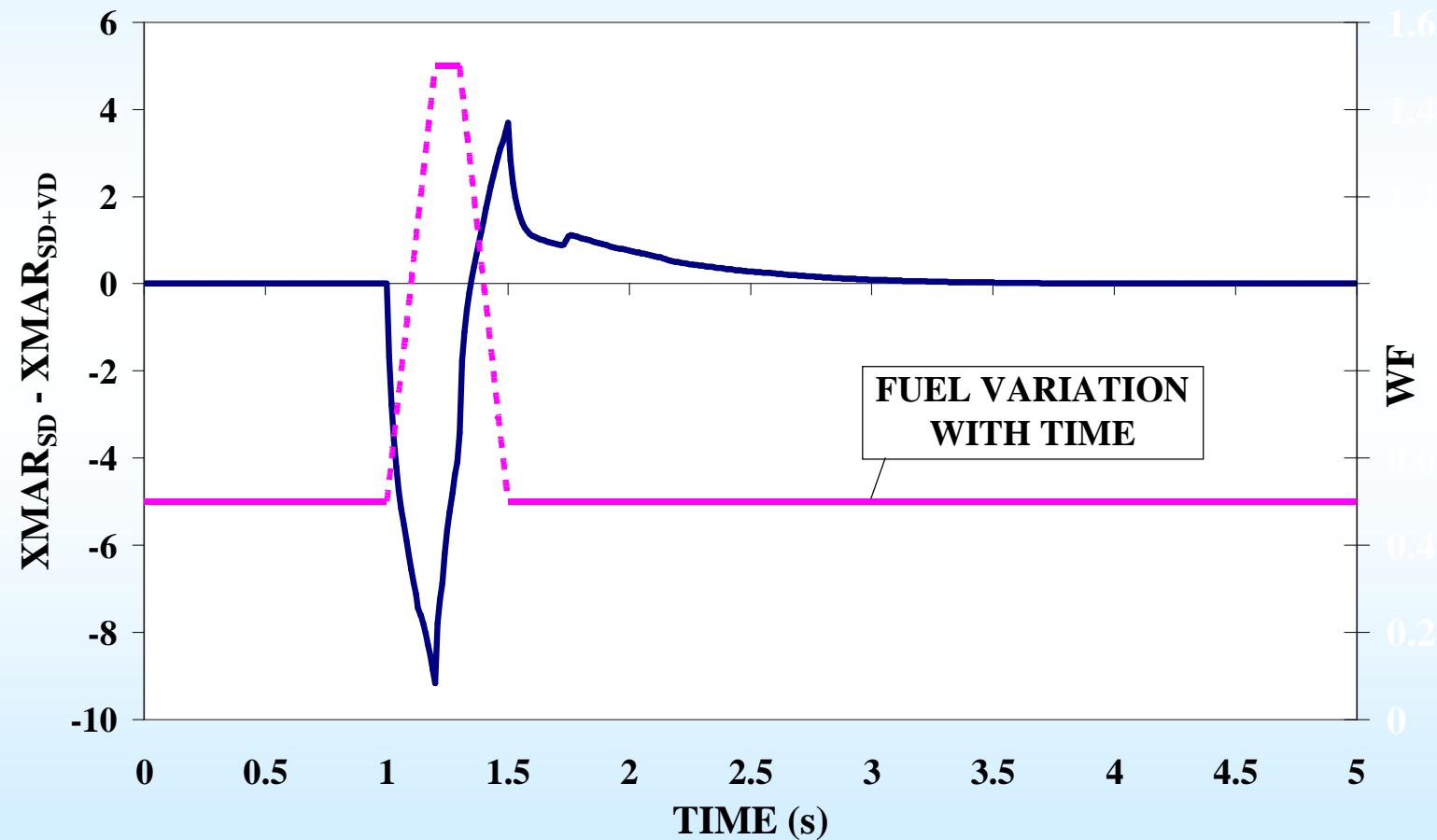


Engine Dynamics: Effect on Thrust





Gas Dynamics Effects on Compressor Surge Margin during a Fuel Spike





Development of Gas Turbine Performance Models using a Generic Simulation Tool

§ Object-Oriented Gas Turbine Performance Modelling

§ Features of Simulation Tool

§ Modelling Methods

§ Application Example

§ Engine Dynamics

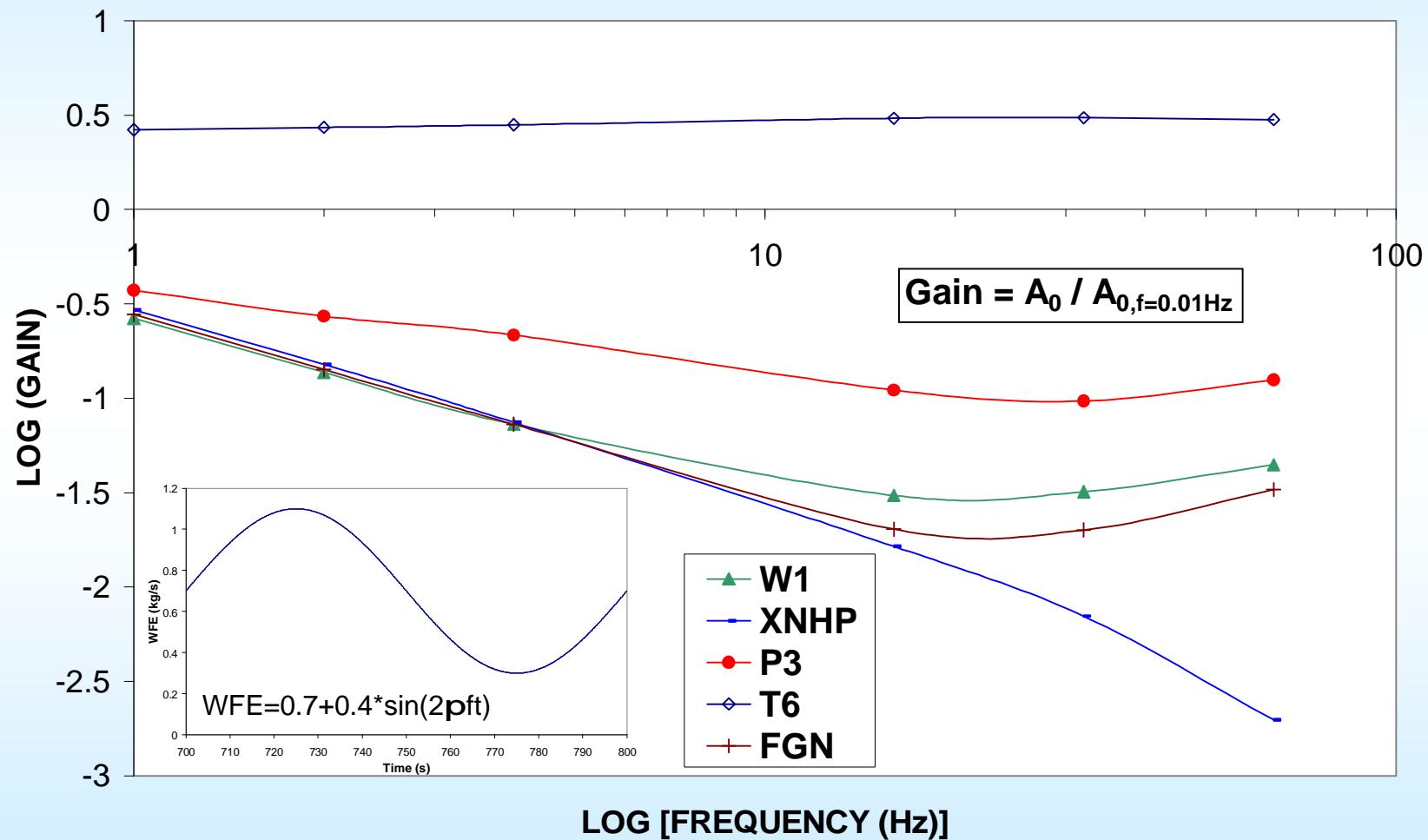
§ Frequency Response

§ Adaptation to Test Data

§ Conclusions



Gain Variation with Frequency for Sinusoidal Fuel Input





Development of Gas Turbine Performance Models using a Generic Simulation Tool

§ Object-Oriented Gas Turbine Performance Modelling

§ Features of Simulation Tool

§ Modelling Methods

§ Application Example

§ Engine Dynamics

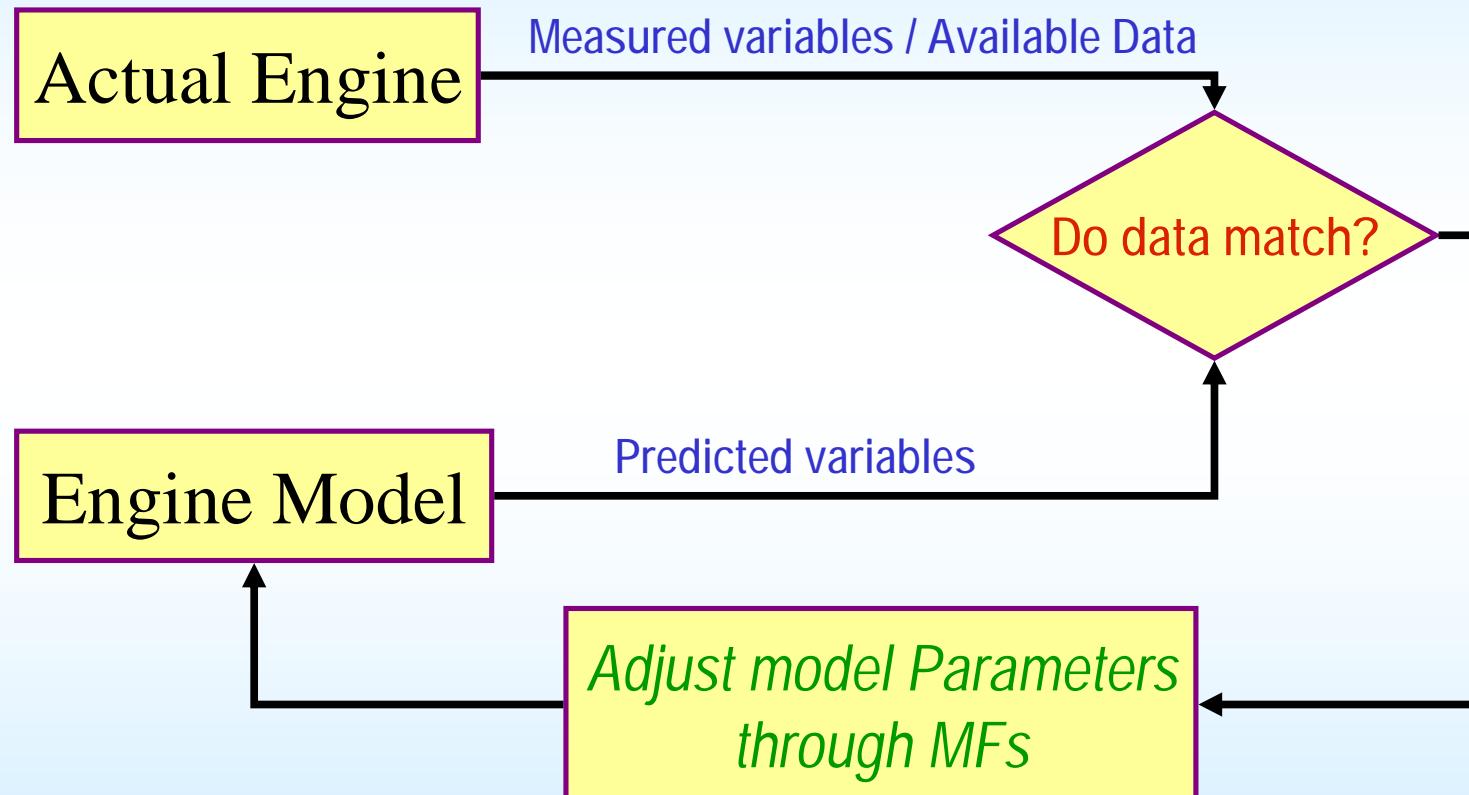
§ Frequency Response

§ Adaptation to Test Data

§ Conclusions

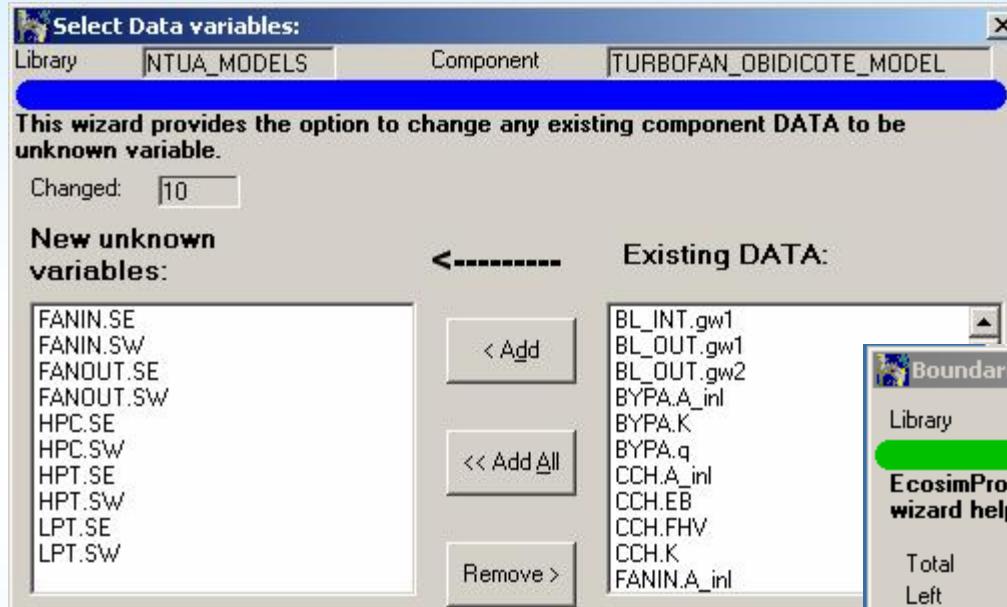


Adaptive Performance Modeling: Principle

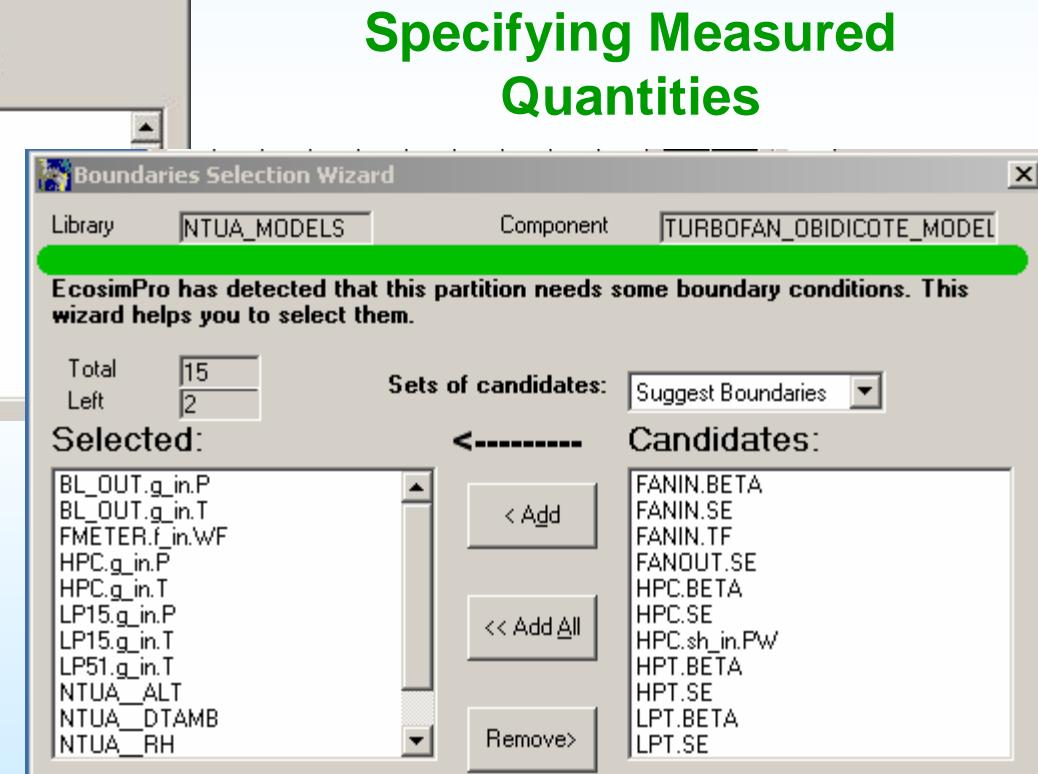




Adaptation to Test Data: Implementation in OOP



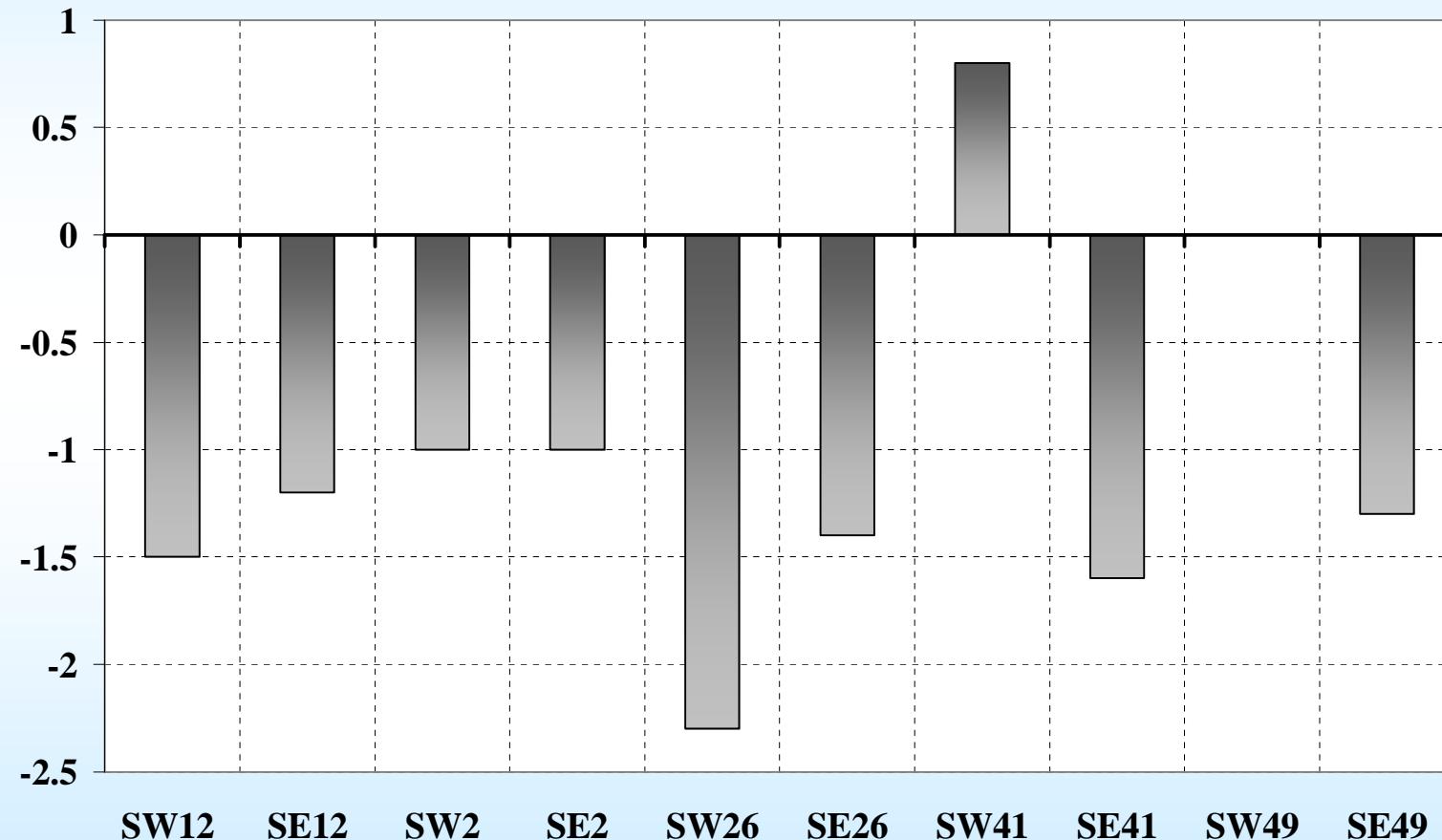
Specifying Adaptation Factors



Specifying Measured Quantities



Sample Results: Estimated Values of Adaptation Factors from Measurements Simulating a Deteriorated Engine





Conclusions

- § An Object-Oriented simulation environment was used to create a re-usable library of gas turbine components
- § A model of a typical civil two-spool turbofan engine was developed by connecting the appropriate components from the library through an advanced graphical user interface
- § Both steady state and transient simulation results agreed well with those produced by an industry-accepted model
- § The ease of incorporating dynamic effects into the model and implementing adaptation factors (for matching a model to given measurements) was demonstrated
- § The approach presented allows for quick implementation of new models and rapid analysis of results