Introduction to MSC.EASY5 with Emphasis on Fluid Power Systems

Thermal/Hydraulics Modeling and Simulation

HC Introductory Class Notes
MSC.EASY5™

Goals and Content

• Goals
  - Learn how to use the features and tools within MSC.EASY5
  - Learn the basic steps in building a model in MSC.EASY5
  - Appreciate MSC.EASY5 as a set of tools to solve hydraulics engineering problems
  - Look for an MSC.EASY5 tool or feature to help with an unusual problem
  - Work with MSC.EASY5, not around it

• What class is not about:
  - How to design valves and hydraulic systems
  - Control analysis/design, although some is inevitable
  - Advanced instruction in the general features of MSC.EASY5

• It will:
  - Make you aware of the capabilities of MSC.EASY5
  - Teach you how to use MSC.EASY5 to model hydraulic systems
  - Help you learn some fundamentals that are usually not well understood
Outline of Course Content

- Overview of MSC.EASY5
- Modeling an Open Loop Oil Cooling System  
  - Learn and practice basic MSC.EASY5 skills  
  - Obtain initial operating points, steady state
- Merging and Splitting Flows
- Modeling a Closed Loop Oil Cooling System  
  - Try different linear analyses  
  - Learn about numerical simulation
- Writing Code in MSC.EASY5
- Building Models of Complex Valves  
  - Mass dynamics and friction  
  - Actuators
- Linking External Functions into MSC.EASY5
MSC.EASY5™

Overview

- MSC.EASY5 is an engineering tool for analyzing complex systems
  
  *Can be Electrical, Pneumatic, Hydraulic, Mechanical,...*
  
  *Used for “intermediate” level of detail modeling and analysis*
    - More detailed than discrete event or steady state tools
    - Less detailed than finite element tools

  *Models use nonlinear, discontinuous algebraic, differential, and difference equations*

- Models can be built in different ways
  
  *Use MSC.EASY5 general purpose blocks (integrators, saturation, sums,...)*

  *Use MSC.EASY5 libraries for specific application areas*
    - Environmental control
    - Thermal-hydraulic
    - Drive train
    - Vapor cycle
    - Electric drive

  *Write your own equations in Fortran components*

  *Build your own application libraries*
Overview
Analysis Options

Types of Analysis:

• Steady State
  — Find the values the plant would settle out to after an initial transient

• Simulation – time response
  — How does the plant respond to a command or a disturbance

• Model Linearization
  — Determine the stability of the system
  — For control system design
  — Also for understanding system

• Frequency response between any to points in model

• Root locus, Stability Margins, Eigenvalue Sensitivity, Power Spectral Density

• Matrix Algebra Tool
  — Controls Design
  — Data Analysis before or after other analyses

Use the MSC.EASY5 Plotter to visualize results
Other Tools

- 100% GUI from start to finish
- Model dynamic systems
  - Differential equations (continuous)
  - Difference equations (discrete)
  - Algebraic equations
  - Differential algebraic equations (DAE)
- Suite of tools for linear/non-linear analysis
- Complete plotting package

MSC.EASY5
Graphical Modeling
Analysis & Plotting

Matrix Algebra Tool
Data Analysis & Calculator

"Extensions"
Link to external programs

Multi-Body Dynamics
  - DADS by LMS
  - ADAMS by MSC
  - Pro/Mechanica by PTC
Finite State Machines
NASTRAN/LS-DYNA
Others
MSC.EASY5’s Integration With Other Applications Enables Complete Virtual Prototyping

- MATLAB®
- SIMULINK®
- MATRIXx®
- SystemBuild®
- Ricardo Engine Simulator
- STATEMATE®
- Finite State Tools
- Multi-Body Dynamics Tools
- ADAMS®
- DADS®
- Pro/ENGINEER®
- CATIA®
- Pro/MECHANICA®
- NASTRAN/ANSYS®
- LS-DYNA
- Beacon®
- Auto-Code Generator Tools
- Structural Tools
Example/Demo

• Simple example to show process from start to finish
  – Model Building
  – Simulation
  – Analysis
Overview
Model Building Process

1st: Define system dynamics.

2nd: Convert to Engineering Block Diagram.

3rd: Translate to MSC.EASY5 Block Diagram.

Lastly: Build MSC.EASY5 model on workstation.
How to Represent Your Model

• Describe systems using:
  – Transfer functions
  – Physical devices
  – Tables

• Systems are modeled/represented as a schematic block diagram

• MSC.EASY5 translates block diagram into:
  – Differential equations
  – Difference equations
  – Algebraic equations

• Nonlinear System:

  \[ \dot{x} = f(x, u, t) \]
  \[ y = g(x, u, t) \]

  where:

  \[ x = \text{State Vector} \]
  \[ u = \text{Input Vector} \]
  \[ t = \text{Time} \]
  \[ y = \text{Output Vector} \]
Schematic Blocks are Components of Your System

MSC.EASY5™

Model definition is a four-step process

Step 1: Drag required components into “schematic pad”.
Schematic Connections Are Physical Connections

MSC.EASY5™

Step 2: Establish data flows by connecting component outputs to inputs.

![Diagram showing schematic connections and physical connections]

- **Outputs**
- **Inputs**
- **Pending Connection**

**Transfer Function**

\[
\frac{Z_1 s + Z_0}{s^2 + P_1 s + P_0}
\]

**Servo valve test data**

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (12)
Step 3: Create “executable” form of model by initiating model build process.

- Converts schematic to equations
- Sorts all algebraic equations
- Generates source code

Source code then compiled and linked to analysis routines.
Define Parameters, IC’s and Tables

Step 4: Define model data for each component.

Parametric Data

Initial Condition Data
Tables $= f(\text{time})$ or $f(\text{independent vars.})$

**Empirical data approach**

Measured data converted to table-lookup format
Set Up and Run Analyses

- Analyses are set up and run using fill-in-the-blank data forms.
- For example, a simulation is executed as follows:
  1) Enter “Title” (optional).
  2) Enter “Start Time”.
  3) Enter “Stop Time”.
  4) Enter “Time Increment” value.
  5) Identify outputs to plot.
  6) Select integration method.
  7) Select “Execute”.

MSC.EASY5™ Intro/Thermal Hydraulic Systems Class- Chart (16)
Execute Analysis

After execution, analysis data is automatically displayed in printed or plotted format.
MSC.EASY5 Summary

• Modeling
  – Large, complex systems.
  – Nonlinear and discontinuous dynamics.
  – Allows for “systems” approach - schematic diagrams versus block diagrams.
  – Hybrid systems (analog + digital) easily modeled and analyzed.
  – Predefined library of components.
  – User-defined libraries and components.

• Analysis
  – Linear and nonlinear analysis performed on same model.
  – 100% graphical-user-interface to setup and execute analyses.
  – Complete plotting package.

• Open Architecture
  – Provides access to a broad set of software tools.

• Customer Support and Service
  – Technical support has received high rating from customers.
  – HOTLINE: 1-800-426-1443 ; email: easy5.support@mscsoftware.com
  – Web: http://www.mscsoftware.com/support/prod_support/easy5/
Introduction to MSC.EASY5 with Emphasis on Modeling and Simulation of Fluid Power Systems

Getting Started With MSC.EASY5
• Focus
  – Getting familiar with MSC.EASY5
  – Learning the basics - mouse, GUI, menus

• Outline
  – Modeling fundamentals and methods
  – EASY5’s graphical-user-interface (GUI)
  – MSC.EASY5 fundamentals
  – Build and simulate a simple hydraulic model
To start MSC.EASY5:
- Unix: easy5x
- Win32 (EASY5 Dos or Korn shell): easy5x
- Win32: Double-click on MSC.EASY5 Desktop icon
- Win32: Start > Programs > MSCSoftware> MSC.EASY5 2004> MSC.EASY5>

Opening Menu:
- Enter directory name.
- Select directory.
- Enter new model name.
- Select existing model.

Enter a new model name: OpenLoop (When the model is saved, a text file called OpenLoop.#.ezmf is created. This is the model file.)
Modeling Tutorial
MSC.EASY5 GUI

• MSC.EASY5 Window

- Version window
- Model name window
- Title line
- Submodel path line

- Main menu
- Schematic pad
- Schematic scroll bar
- Message line
- Message log

• Mouse Usage

- CLICK-L “select”
- CLICK-R “center schematic”
- CLICK-R (HOLD) “pop-up menu”
- CLICK-C (both buttons) “examine”
Modeling Tutorial
Add Component

MSC.EASY5™

• First component to add: **AF - Analytical Function Generator**

• Adding components is a simple process
  – First select [Add] pushbutton (select with CLICK-L) to open “Add Components” window.
  – Select desired library/group/component => drag-and-drop component into mode.

![Diagram of component selection and drop process]

1. Select Group.
2. Select Component.
3. Move mouse and component across schematic.
4. Press left mouse button to drop component in place.
Add remaining components

The first component has been added. Now add remaining components in the order shown.

<table>
<thead>
<tr>
<th>Library:</th>
<th>Group:</th>
<th>Component:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interactive simulation (IS)</td>
<td>Interactive Components</td>
<td>TI: Simulated and accumulated CPU times</td>
</tr>
<tr>
<td>Thermal Hydraulic (HC)</td>
<td>Miscellaneous</td>
<td>FP: Global Fluid Properties</td>
</tr>
<tr>
<td>Thermal Hydraulic (HC)</td>
<td>Boundary Conditions</td>
<td>TN: Constant Pressure Source</td>
</tr>
<tr>
<td>Thermal Hydraulic (HC)</td>
<td>Pumps</td>
<td>PD: Positive Displacement Pump</td>
</tr>
<tr>
<td>Thermal Hydraulic (HC)</td>
<td>Miscellaneous</td>
<td>FI: Filter</td>
</tr>
<tr>
<td>Thermal Hydraulic (HC)</td>
<td>Pipes</td>
<td>PI: Pipe</td>
</tr>
<tr>
<td>Thermal Hydraulic (HC)</td>
<td>Valves</td>
<td>VM: Metering Valve</td>
</tr>
<tr>
<td>Thermal Hydraulic (HC)</td>
<td>Boundary Conditions</td>
<td>TN2: Constant Pressure Source</td>
</tr>
</tbody>
</table>

![Diagram of components and symbols]
Connect Components

MSC.EASY5™

• Connecting components is a simple process:
  1. Select the “from” component
  2. Select the “to” component

• There are three types of connections:
  • *Default* Connection
    • Connect TN to PD (this connects the tank exit port to the pump inlet port).
    • Connection line is automatically drawn.
  • *Port* Connection
    • Connect PD to FI [from PD Out to FI In]
  • *Custom (Manual)* Connection
    • Manually connect AF to VM
    • Select AF, then select VM
    • In the connection data table, select S_Out_AF (signal output), then select ARE_VM (area of orifice opening)

• Finish connecting HC components to each other
  • Connect FI to PI, PI to VM, and VM to TN2
Port Connections

- A component port is a collection of variables (both inputs and outputs) grouped to represent a physical connection.

- A port connection between a port on one component to a port on a second component consists of:
  - A connection of each output variable of either port to the input of the other port with the same variable identifier (ignoring the port number).

- Ports are named (e.g., “In”, “Drain”, “Pos”)
  - Port name is preceded and followed by an underscore character.
  - In Port conventionally is main input (or fluid “inlet”).
  - Out Port conventionally is main output (or fluid “outlet”).
  - Others are assigned as needed.

- Connection scheme
  - Conventional names (W for flow rate, etc.) and ports make automatic connection possible.
  - Also can check for complete connection – i.e. won’t misconnect $W = \text{flow rate}$ to $W = \text{shaft speed}$ because rest of port names don’t match.
Complex Connections

MSC.EASY5™

- Connections between the GP components are usually a single output to a single input.
  - Example: Connection from S_In_AF to ARE_VM

- MSC.EASY5 is able to use a single schematic connection line to represent a physical association between two components of a physical system.
  - Examples:
    - *Hydraulic fluid flowing from a valve into a heat exchanger*
    - *Electric power flowing from a transmission line to a transformer*
    - *Mechanical power flowing from a drive shaft to a differential*

- Modeling these associations requires multi-variable, bi-directional information passing between the component models.

Example (simplified HC connection, double-click on any HC connection to see actual):

![Diagram showing connections between a Valve Component and a Heat Exchanger](image)

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (27)
Storage and Resistive Boundary Conditions

**Storage**
- Inlet or Exit
- W
- P

**Resistive**
- Inlet or Exit
- P
- W

- In general, like boundary conditions cannot be connected together.
- Components with both inlet and exit storage boundary conditions are labeled
- Components with both inlet and exit resistive boundary conditions are labeled
- Components with inlet storage and exit resistive boundary conditions are labeled
- Icon “pins” give you clues
  - Storage exit
  - Resistive exit
  - Storage inlet
  - Resistive inlet

MSC.EASY5™ Intro/Thermal Hydraulic Systems Class- Chart (28)
Open Loop Hydraulic System
MSC.EASY5™

Modeling Tutorial
Selecting Menu Items

Review menu selection techniques

• Using the mouse
  • Select menu item with a single CLICK-L or HOLD-L (hold left mouse button).

• Using the keyboard
  • To access main menu use <meta-key> then enter any underlined menu character.
    • Example: To open File menu, enter <meta key>F.
    • Then, to select menu options: use underlined character, or, arrow keys
      Example: To select Save from File menu, enter S.

• Using accelerator keys
  • Enter: <Ctrl>[accelerator key]
    Example: To select Save from File menu, enter <Ctrl>S.

Save model using any of the above methods.

• Model version number increases each time model is saved: openloop(1)
• Multiple versions are saved so you can return to old versions.
Modeling Tutorial
Manipulating the Schematic

• Schematic operations
  • Zoom In/ Zoom Out
    • Use *right* mouse button to “*part*” schematic.
  • Use scroll bars to pan schematic (see next chart).
  • Review View menu options.
    – View Entire Schematic.
    – Zoom To Defined Area.
  • Use Schematic Menu with CLICK-R and HOLD.

• Move components
  • Move a single component use drag-and-drop method.
    – HOLD-L on component and drag, RELEASE-L in new location.
  • Move a group of components.
    – Draw selection box.
    – Select [Move] button.
    – Then drag-and-drop.
MSC.EASY5™

Modeling Tutorial
Using Scroll Bars

• Scroll one line up/down
  • Select scroll bar arrows.

• Scroll one screen
  • Select scroll bar above/below scroll box.

• Scroll to any position
  • Drag the scroll box to desired position.

• Scroll continuously
  • Select and hold scroll arrows.
Modeling Tutorial
Other MSC.EASY5 Functions

• Copy components (single/group)

• Delete components

• Create Submodel
  • “Examine” submodel.
  • Move components in and out of submodels.
  • Expand submodel.

• Restore original OpenLoop model
  • Select: File => Open New or Existing Model.
  • Select the highest version of the OpenLoop model.
Component inputs/outputs are defined in the Component Data Table (CDT). To examine and edit a CDT, double-click on the icon. Examine the AF component:

- **Component Name/library**
- **Component Title**
- **Inputs**: defines input parameters and connections.
- **States**: defines state data, initial conditions, error controls.
- **Variables**: displays outputs that can be connected and/or printed/plotted.
- **Documentation**: click on “Info” button to see documentation.

"States" defines state data, initial conditions, error controls.
"Variables" displays outputs that can be connected and/or printed/plotted.
"Inputs" defines input parameters and connections.
Click on “Info” button to see documentation.
Modeling Tutorial
Edit Component Data Table

MSC.EASY5™

• Define input parameters
  • Define Function Code COD=Cosine (select from list of choices)
  • Enter numerical parameters
    – Select the current value (0.99999) and enter number<Return>

• Define output variable name
  • MSC.EASY5 defines outputs with default names. AF component output name:
    “S_Out_AF”
  • You can change MSC.EASY5’s name to your own “user-defined” name.
    – Select the “S_Out_AF” with a double-click of the left mouse button (this highlights the name).
    – Type in your own name; enter: OrificeArea.
Fluid Properties Component

Global Fluid Properties Component:

- Required when using any HC or HB components.

- Defines:
  - **Units** (metric or English).
  - **Type of fluid(s)** in system.
  - **Ambient temperatures**.
  - Percentage of **entrained air**.
  - **Viscosity derating** (can be used to simulate multi-vis fluids).
  - Whether **temperature considerations** are required in the calculation (a great simulation time saver if TCX=0).
## Define Model Data

Define the model data as given in the following table:

<table>
<thead>
<tr>
<th>Component</th>
<th>Input/Output</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>TN</td>
<td>PTK</td>
<td>1.0</td>
<td>Const. upstream pressure</td>
</tr>
<tr>
<td></td>
<td>TFK</td>
<td>30.0</td>
<td>Will default to TC in FP if not set (0.99999)</td>
</tr>
<tr>
<td></td>
<td>LP_In</td>
<td>1</td>
<td>Loop number</td>
</tr>
<tr>
<td>PD</td>
<td>DPR</td>
<td>5</td>
<td>Displacement per rev</td>
</tr>
<tr>
<td></td>
<td>RPM</td>
<td>1500</td>
<td>Pump RPM</td>
</tr>
<tr>
<td></td>
<td>EO_Eff</td>
<td>0.93</td>
<td>Pump efficiency (0-1)</td>
</tr>
<tr>
<td></td>
<td>EV_Veff</td>
<td>0.95</td>
<td>Volumetric efficiency</td>
</tr>
<tr>
<td>FI</td>
<td>QRT</td>
<td>50</td>
<td>Nominal flow rate</td>
</tr>
<tr>
<td></td>
<td>DPR</td>
<td>10</td>
<td>Nominal pressure</td>
</tr>
<tr>
<td></td>
<td>PF</td>
<td>5.0</td>
<td>Upstream pressure</td>
</tr>
<tr>
<td>PI</td>
<td>DH</td>
<td>1.0</td>
<td>(change name to) PipeDiameter</td>
</tr>
<tr>
<td></td>
<td>LEN</td>
<td>600</td>
<td>Extra volume</td>
</tr>
<tr>
<td></td>
<td>XVO</td>
<td>10</td>
<td>Upstream pressure</td>
</tr>
<tr>
<td></td>
<td>PF</td>
<td>5.0</td>
<td></td>
</tr>
</tbody>
</table>

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (37)
Define the model data as given in the following table:

<table>
<thead>
<tr>
<th>Component</th>
<th>Input/Output</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>VM</td>
<td>VOL, PF</td>
<td>50, 5.0</td>
<td>Upstream pressure</td>
</tr>
<tr>
<td>TN2</td>
<td>PTK, TFK</td>
<td>4.8, 30</td>
<td>Const. downstream P Will default to TC in FP if unset</td>
</tr>
<tr>
<td>FP</td>
<td>FC, TC, TCX</td>
<td>4, 45, 1.0</td>
<td>Use hydraulic fluid Ambient Temperature Temp. Consideration Flag</td>
</tr>
<tr>
<td>AF</td>
<td>COD, C1, C2, C3, C4, C5</td>
<td>Cosine 0.05, Bias 0.04, Amplitude 0.314, Phase 0</td>
<td>Function code</td>
</tr>
</tbody>
</table>

MSC.EASY5™

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (38)
Final Open Loop Hydraulic System

MSC.EASY5™

Untitled
• MSC.EASY5 uses “compiled” versus “interpretive” language.
  • Feature has advantages/disadvantages.
  • MSC.EASY5 is a code generator; Fortran and/or C code is generated.

• Must create the “executable” model.

• Select menus: Build => Create Executable

“Create Executable” is a background process.
  • You can continue using MSC.EASY5.
  • Setup analyses while model is building.
  • When finished, message bar displays: “Executable has been created.”
Hydraulic Loop Source File

MSC.EASY5™

- The source code for this model should look like:

- Connections are made by substituting in the name of the output of the ‘from’ component for the name of the input in the code of the ‘to’ component

- Notice that the source code has no values for parameters, initial conditions, tables, etc.
• Built simple open-loop hydraulic model.
• What’s next?
The open-loop model will be used to demonstrate:
  • Steady-state analysis and operating points
  • Simulation
  • Integration methods
  • Analysis tools and methodology
Analyzing the Open Loop

Hydraulic Model:

Steady State and Transient Analysis
Calculate Initial Conditions

Select ‘Analysis/Miscellaneous/Initial Condition Calculation’.

**STATES**

<table>
<thead>
<tr>
<th>AFS_VM</th>
<th>PF_FI</th>
<th>PF_PI</th>
<th>PF_VM</th>
<th>SSS_FI</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.000</td>
<td>5.0000</td>
<td>5.0000</td>
<td>5.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>SSS_PI</td>
<td>SSS_VM</td>
<td>SWP_PD</td>
<td>SWQ_PI</td>
<td>SWQ_VM</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
</tr>
<tr>
<td>TFL_VM</td>
<td>TF_Drain_PD</td>
<td>TF_Out_FI</td>
<td>TF_Out_PD</td>
<td>TF_Out_PI</td>
</tr>
<tr>
<td>50.000</td>
<td>45.000</td>
<td>55.000</td>
<td>55.000</td>
<td>55.556</td>
</tr>
<tr>
<td>TW_PI</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**RATES**

<table>
<thead>
<tr>
<th>AFS_VM</th>
<th>PF_FI</th>
<th>PF_PI</th>
<th>PF_VM</th>
<th>SSS_FI</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4.0000</td>
<td>155.90</td>
<td>-2.12096E-12</td>
<td>-11445.</td>
<td></td>
</tr>
<tr>
<td>SSS_PI</td>
<td>SSS_VM</td>
<td>SWP_PD</td>
<td>SWQ_PI</td>
<td>SWQ_VM</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>2.0000</td>
</tr>
<tr>
<td>TFL_VM</td>
<td>TF_Drain_PD</td>
<td>TF_Out_FI</td>
<td>TF_Out_PD</td>
<td>TF_Out_PI</td>
</tr>
<tr>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-29.081</td>
<td>0.0000</td>
</tr>
<tr>
<td>TW_PI</td>
<td></td>
<td></td>
<td></td>
<td>0.0000</td>
</tr>
</tbody>
</table>

**VARIABLES**

<table>
<thead>
<tr>
<th>AF_VM</th>
<th>DP_DrFlow_PD=</th>
<th>DP_Eff_PD=</th>
<th>DP_VEff_PD=</th>
<th>DS_DrFlow_PD=</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1000</td>
<td>4.0000</td>
<td>4.0000</td>
<td>4.0000</td>
<td>5.0000</td>
</tr>
<tr>
<td>DS_Eff_PD</td>
<td>DS_VEFF_PD=</td>
<td>FRC_PI=</td>
<td>HTC_PI=</td>
<td>LP_Drain_PD=</td>
</tr>
<tr>
<td>5.0000</td>
<td>5.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>LP_Out_FI</td>
<td>LP_Out_PD=</td>
<td>LP_Out_PI=</td>
<td>LP_Out_TN=</td>
<td>LP_Out_TN2=</td>
</tr>
<tr>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>LP_Out_VM</td>
<td>MEF_PD=</td>
<td>OEF_PD=</td>
<td>OrificeArea=</td>
<td>PD_In_FI=</td>
</tr>
<tr>
<td>0.97895</td>
<td>0.930000</td>
<td>9.000000E-02</td>
<td>155.90</td>
<td></td>
</tr>
<tr>
<td>PD_In_PI</td>
<td>PD_In_TN=</td>
<td>PD_In_TN2=</td>
<td>PD_In_VM=</td>
<td>PD_Out_TN=</td>
</tr>
<tr>
<td>-2.12096E-12</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-11445.</td>
<td>0.0000</td>
</tr>
<tr>
<td>PD_Out_TN2</td>
<td>PWR_PD=</td>
<td>P_In_FI=</td>
<td>P_In_PI=</td>
<td>P_In_TN=</td>
</tr>
<tr>
<td>0.0000</td>
<td>51.075</td>
<td>5.0000</td>
<td>5.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>P_In_TN2</td>
<td>P_In_VM=</td>
<td>P_Out_FI=</td>
<td>P_Out_PI=</td>
<td>Q1_PD=</td>
</tr>
<tr>
<td>4.8000</td>
<td>5.0000</td>
<td>1.0000</td>
<td>4.8000</td>
<td>6.9861</td>
</tr>
<tr>
<td>QF_PI</td>
<td>Q_Drain_PD=</td>
<td>Q_Out_FI=</td>
<td>Q_Out_PI=</td>
<td>Q_Out_VM=</td>
</tr>
<tr>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>7.1250</td>
<td>2.5579</td>
</tr>
<tr>
<td>Q_Out_VM</td>
<td>REV_PI=</td>
<td>SS_DrFlow_PD=</td>
<td>SS_Eff_PD=</td>
<td>SS_VEFF_PD=</td>
</tr>
<tr>
<td>2.5579</td>
<td>1.0000000E-12</td>
<td>1500.0</td>
<td>1500.0</td>
<td>1500.0</td>
</tr>
<tr>
<td>TF_Out_TN</td>
<td>TF_Out_TN2=</td>
<td>TF_Out_VM=</td>
<td>TQ_Shaf_PD=</td>
<td>TR_In_FI=</td>
</tr>
<tr>
<td>30.000</td>
<td>30.000</td>
<td>50.011</td>
<td>-0.32516</td>
<td>55.000</td>
</tr>
<tr>
<td>TR_In_PD</td>
<td>TR_In_PI=</td>
<td>TR_In_TN=</td>
<td>TR_In_TN2=</td>
<td>TR_In_VM=</td>
</tr>
<tr>
<td>30.000</td>
<td>55.556</td>
<td>30.000</td>
<td>30.000</td>
<td>50.000</td>
</tr>
<tr>
<td>VEF_PD</td>
<td>VE_Ripple_PD=</td>
<td>W_Drain_PD=</td>
<td>W_In_PD=</td>
<td>W_Out_FI=</td>
</tr>
<tr>
<td>0.095000</td>
<td>157.08</td>
<td>0.0000</td>
<td>5.8110</td>
<td>0.0000</td>
</tr>
<tr>
<td>W_Out_PD</td>
<td>W_Out_PI=</td>
<td>W_Out_VM=</td>
<td>W_Out_VM=</td>
<td>2.0948</td>
</tr>
</tbody>
</table>

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (44)
Operating Point
What Is An Operating Point?

- The “operating point” is the value of the system’s states; it is defined by the state vector $X$.
  - Linear system: Dynamics are *invariant* with operating point (but you still may want the simulation to start at a specific operating point).
  - Nonlinear system: Dynamics depend on operating point (example: variable ‘gain’ in valve).

- Note on the previous slide that some of the states have large derivatives. If we were to run a simulation from this operating point, we would see a sharp artificial transient at the start.

- If you don’t define an operating point that is within the valid operating region of your model:
  - Simulations may never start.
  - Simulation run-times may be excessive.
  - Linear analyses may produce the wrong answer.
  - Steady-state analyses may never converge.
• If you don’t define an operating point, MSC.EASY5 GP components use default values of zero for all states. The HC library uses defaults of 1 for pressure states and 40 for temperature states, regardless of units.
Operating Point
Methods To Define Operating Points

• Different ways operating point can be defined.
  • Enter values by hand directly into component data tables.
  • Execute a simulation and save the resulting end-point.
  • Calculate a steady-state operating point using MSC.EASY5’s Steady-State analysis.
  • Calculate an initial condition dynamically using special FORTRAN code and the reserved word flag ICCALC.
Operating Point
Define Operating Point Directly In CDT

- Operating point data must be defined for each component in a model that calculates one or more output states, and is entered in the respective component data tables.

Component Data Table

Initial Condition Data
Define Operating Point From Simulation or Steady-state

• Once an operating point has been defined/calculated, they can be “saved” in named files as follows:
  • During Simulation and Steady-State analyses, using the “Save...” field in the analysis forms. (The text file name for the operating point is modelname.ID.ic, or OpenLoop.final_op.ic in this example).

• When calculated directly in user code, an operating point is loaded into the initial condition vector during an Initial Conditions calculation (by using the CALC-XIC command).
Operating Point
Save/Restore Operating Point

• You can save and restore the operating point to a file.

• To save an operating point, select Options => Save Operating Point.
  • Writes all state values from component data files to a file for future use.
  • Useful for saving nominal values.

• To restore an operating point, select Options => Restore Operating Point.
  • Copies in all state values into the component data forms.
  • This overwrites state initial condition values in component data forms.

• In all analysis data forms, an Initial Operating Point field can be used, which uses the referenced operating point file only for the respective analysis.

Use of this field loads the operating point only for the respective analysis.
Initial Operating Point and Temporary Settings data (we’ll talk about this later) are loaded after the nominal model data when an MSC.EASY5 analysis is executed, thereby over-riding the nominal values.

```
COMMAND INITIAL CONDITIONS
TFL_VM = 50,AFS_VM = 10, SWQ_VM = 1, SSS_VM = 1, PF_VM = 5
COMMAND SWP_PD = 1, TF_Out_PD = 55, TF_Drain_PD = 55,
COMMAND PF_PI = 5, TF_Out_PI = 55, SSS_PI = 1
COMMAND ERROR, CONTROLS,TFL_VM = 1e-06,AFS_VM = 1e-06,
COMMAND SWQ_VM = 1e-06, SSS_VM = 1e-06
COMMAND TF_VM = 1e-06, SWP_PD = 1e-06, TF_Out_PD = 1e-06,
COMMAND PF_PI = 1e-06, TF_Out_PI = 1e-06, SSS_PI = 1e-06
COMMAND INT, CONTROL,TFL_VM = 1, AFS_VM = 1, SWQ_VM = 1, SSS_VM = 1,
COMMAND TF_Out_PD = 1, TF_Drain_PD = 1, PF_PI = 1, TF_Out_PI = 1, SSS_PI = 1,
COMMAND PF_VM = 1, SWP_PD = 1
COMMAND INITIAL TIME = 0
COMMAND KEYCODE = 1
COMMAND INITIAL CONDITIONS
COMMAND TFL_VM = 50
COMMAND AFS_VM = 10
COMMAND SWQ_VM = 1
COMMAND SSS_VM = 1
COMMAND PF_VM = 5
COMMAND SWP_PD = 1
COMMAND TF_Out_PD = 55
COMMAND TF_Drain_PD = 55
COMMAND PF_PI = 5
COMMAND TF_Out_PI = 55
COMMAND SSS_PI = 1
COMMAND PF_PI = 5
COMMAND TF_Out_PI = 55, SSS_PI = 1
COMMAND SSS_PI = 1
COMMAND SWQ_PI = 1
COMMAND ...END OF DATA FROM AUXILIARY FILE OPENLOOP.BENCHMARK.EZIC
COMMAND TITLE UNTITLED
COMMAND PRINT CONTROL = 3
COMMAND INITIAL TIME = 0
COMMAND * - TEMPORARY SETTINGS FILE 'Openloop.test.ezty' ---- *
COMMAND PARAMETER VALUES, VOL_FI = 1000
COMMAND CALC XIC
```
It would be nice to have a robust method of calculating an initial operating point that makes sense physically.

**MSC.EASY5 Steady State Analysis:**

- Algebraically computes an operating point at which all time derivatives are zero.
  - This corresponds to the assumption that most plants are normally “at rest.”
- Iterative solution based on Newton’s method.
  - Iteration begins with specified initial conditions.
  - Default is 100 iterations.
  - Uses error controls to help manage iteration step sizes.
- Computes eigenvalues after last iteration.
- Almost always converges:
  - If initial conditions are not too far off.
  - If the values make physical sense.
  - If the values don’t violate assumptions of components (reverse flow not allowed, etc).
  - If the operating point is not too near a discontinuity (steam/water boundary, etc).
• Steady-state operating points can be calculated using MSC.EASY5’s Steady-State Analysis.

Analysis locates the state vector $X_{ss}$ where:

$$\dot{x} = F(X_{ss}, t_0) = 0$$
Steady State Analysis
Steady State Method

- During a Steady-State analysis, MSC.EASY5 makes use of a Newton-Raphson iterative method to locate the nearest steady-state point.
  - The Steady-State analysis starts its search from the currently defined operating point.
  - Any states that cause a singularity in the Jacobian are automatically frozen.
Steady State Analysis
Not All Systems have a Steady State

• Examples:
  – Rotating machines (shaft velocity never goes to zero, so position never has a zero derivative).
  – Airplanes (forward velocity better not go to zero).
  – Open loop systems (integrator output not fed back).

• MSC.EASY5 will detect most of these:
  – Freeze the state (hold the initial condition)
  – Try the steady state solution.
  – May freeze part of system if initial guess is in dead zone or other flat spot.
  – Looks like an open loop.

• Try to pick initial guess to bias system out of dead zones.

• If states get frozen, save result, perform new analysis starting from saved result.
Steady State Analysis
Steady State Doesn’t Always Work!

• Steady-State analyses may fail to converge to a solution because:
  • A steady-state point is too far away from the initial operating point and the algorithm fails.
  • No steady-state point(s) exists.
  • Nonlinear models have not been properly modeled with switch states and the algorithm fails.
  • The “maximum number of iterations” variable is exceeded.
  • The problem is too complex and the algorithm simply fails.

• Steady-State analyses may fail to locate the correct (the one you want) steady-state point because the algorithms converges to a steady-state point closer to its starting point.

• Steady-State analysis may locate a steady-state point, but the steady-state point is unstable.
  • To guard against this, MSC.EASY5 also calculates the eigenvalues at the respective point and they should be checked to verify stability.
Steady State Analysis
Finding a Steady State for the Open Loop Model

MSC.EASY5™

• Select “Analysis/Steady State”.

• Click “Yes” after “Save Final Operating Point” and enter “SteadyState” as the operating point file name.

• Press the “Execute and Close” button at the bottom of the page.

• Review the results when the results window appears.
Steady State Analysis
Results of Steady State Calculation

MSC.EASY5™

### STEADY STATE ANALYSIS CONVERGED WITH AN RMS ERROR OF 6.504901E-05 IN 14 ITERATIONS TO THE VALUES LISTED BELOW.

<table>
<thead>
<tr>
<th>States</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>AFS_VM</td>
<td>5.0000</td>
<td>PF_PI</td>
<td>8.7431</td>
<td>PF_PI</td>
<td>7.3184</td>
<td>PF_VM</td>
</tr>
<tr>
<td>SSS_PI</td>
<td>10.0000</td>
<td>SSS_VM</td>
<td>1000.0</td>
<td>SWQ_PD</td>
<td>1.0000</td>
<td>SWQ_PI</td>
</tr>
<tr>
<td>TF_LI_VM</td>
<td>30.134</td>
<td>TF_Drain_P0</td>
<td>45.000</td>
<td>TF_Out_P0</td>
<td>30.101</td>
<td>TF_Out_P0</td>
</tr>
<tr>
<td>TW_PI</td>
<td>45.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rates</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>AFS_VM</td>
<td>0.0000</td>
<td>PF_PI</td>
<td>3.84877E-10</td>
<td>PF_PI</td>
<td>2.81609E-08</td>
<td>PF_VM</td>
</tr>
<tr>
<td>SSS_PI</td>
<td>10.0000</td>
<td>SSS_VM</td>
<td>1000.0</td>
<td>SWQ_PD</td>
<td>1.0000</td>
<td>SWQ_PI</td>
</tr>
<tr>
<td>TF_LI_VM</td>
<td>4.76976E-10</td>
<td>TF_Drain_P0</td>
<td>0.0000</td>
<td>TF_Out_P0</td>
<td>1.0000E-13</td>
<td>TF_Out_P0</td>
</tr>
<tr>
<td>TW_PI</td>
<td>0.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variables</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>AF_VM</td>
<td>9.000000E-02</td>
<td>DP_DrFlow_P0</td>
<td>7.7431</td>
<td>DP_EFF_P0</td>
<td>7.7431</td>
<td>DP_DrFlow_P0</td>
</tr>
<tr>
<td>DS_Eff_P0</td>
<td>5.0000</td>
<td>DS_Eff_P0</td>
<td>5.0000</td>
<td>FRC_PI</td>
<td>0.10000</td>
<td>HMT_P0</td>
</tr>
<tr>
<td>LP_LI_P0</td>
<td>1.0000</td>
<td>LP_LI_P0</td>
<td>1.0000</td>
<td>LP_Out_P0</td>
<td>1.0000</td>
<td>LP_Out_P0</td>
</tr>
<tr>
<td>LP_Out_P0</td>
<td>1.0000</td>
<td>LP_Out_P0</td>
<td>1.0000</td>
<td>LP_Out_P0</td>
<td>1.0000</td>
<td>LP_Out_P0</td>
</tr>
<tr>
<td>PD_IN_P0</td>
<td>2.81609E-08</td>
<td>PD_IN_TN</td>
<td>0.0000</td>
<td>PD_IN_TN</td>
<td>0.0000</td>
<td>PD_IN_P0</td>
</tr>
<tr>
<td>PD_OUT_P0</td>
<td>0.0000</td>
<td>PD_OUT_P0</td>
<td>0.0000</td>
<td>PD_OUT_P0</td>
<td>0.0000</td>
<td>PD_OUT_P0</td>
</tr>
<tr>
<td>P_IN_TN</td>
<td>4.8000</td>
<td>P_IN_VM</td>
<td>6.7431</td>
<td>P_OUT_TN</td>
<td>1.0000</td>
<td>P_OUT_TN</td>
</tr>
<tr>
<td>QI_P0</td>
<td>0.0000</td>
<td>QI_Drain_P0</td>
<td>0.0000</td>
<td>QI_OUT_P0</td>
<td>7.1245</td>
<td>QI_Drain_P0</td>
</tr>
<tr>
<td>QI_OUT_P0</td>
<td>7.1245</td>
<td>QI_OUT_P0</td>
<td>7.1245</td>
<td>QI_OUT_P0</td>
<td>7.1245</td>
<td>QI_OUT_P0</td>
</tr>
<tr>
<td>TQ_Shift_P0</td>
<td>30.0000</td>
<td>TQ_Shift_P0</td>
<td>30.0000</td>
<td>TQ_Shift_P0</td>
<td>30.0000</td>
<td>TQ_Shift_P0</td>
</tr>
<tr>
<td>TR_IN_TN</td>
<td>30.0000</td>
<td>TR_IN_TN</td>
<td>30.0000</td>
<td>TR_IN_TN</td>
<td>30.0000</td>
<td>TR_IN_TN</td>
</tr>
<tr>
<td>VEF_P0</td>
<td>9.05000</td>
<td>VEF_P0</td>
<td>197.00</td>
<td>VEF_P0</td>
<td>197.00</td>
<td>VEF_P0</td>
</tr>
<tr>
<td>W_OUT_P0</td>
<td>5.9283</td>
<td>W_OUT_P0</td>
<td>5.9283</td>
<td>W_OUT_P0</td>
<td>5.9283</td>
<td>W_OUT_P0</td>
</tr>
</tbody>
</table>

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (58)
MSC.EASY5™

Steady State Analysis
Steady-State Scan

• MSC.EASY5 unique feature: Steady-State Scan Analysis
  – Scan (vary) a parameter - calculate Steady-State - plot results

  Example: What affect does the diameter of the pipe (input name: DH_PI) have on system’s steady-state?

  – Steady-State Scan will vary the parameter and calculate a steady state at each value – always at the given time.

  – This is not a simulation!

  – You can even vary a state and observe the effects if you freeze it first (more on freezing states later).

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (59)
Steady State Analysis
Steady-State Scan

Set-up and run the Steady-State Scan Analysis:

1. Set “Mode” to Scan.
2. Pick DH_PI from schematic as scan parameter.
3. Vary DH_PI from .01 to 1.0
4. Plot 100 points.

Make sure this is set this way.
Steady State Analysis
Specifying Plot Variables

MSC.EASY5™

• Pick the desired variables to plot from the schematic - or
• Use the Show Name List option.
Steady State Analysis
Results of DH_PI Scan

P_In_VM (pressure in pipe), Q_Out_VM (metering valve flow rate), P_In_FI (Pressure developed by pump at the filter inlet) as a function of DH_PI.

Use Locator and Tag functions to find pressure at a specific area.
Plot Locate Feature

Center Click-H to use plot “locate” feature to display exact numerical values.
• Any portion of plotted data can be “zoomed” with the mouse for a closer look.
Running a Simulation

- Simulation Analysis Data Form works just like Steady State form:
  - Select Analysis > Simulation.
  - Must specify:
    - Start time and stop time.
    - Time increment (This has different meaning with different integration methods).
    - Integration method.
  - Plotted Output:
    Frequency of plot points - plot increment.
    Names of variables to be plotted.
    Default plot format.
  - Printed Output:
    Frequency of print points - print increment.
    Print options.
Settings for Open Loop System Simulation

MSC.EASY5™

Open the Simulation Data Form and enter the following:

Title = Open Loop Hydraulic System Model
Start Time = 0 ; Stop Time = 100;  Time Increment = 0.1;  Int. Method = BCS Gear
Set the Initial Operating Point file to: SteadyState (created in previous exercise)
Settings for Open Loop System Simulation

Need to define which data to plot.

- Set “Plot Results” to “Yes” and press the “Show/Edit Plot Variables”.
- Seltec [Show Name List] From Name List window, select following data:
  1. ‘P_Out_TN’ ‘P_In_FI’ ‘P_In_PI’ ‘P_Out_TN2’ [click ‘Yes’ for overplot]
  2. ‘Q_Out_PD’ ‘Q_Out_FI’ ‘Q_Out_PI’ ‘Q_Out_VM’ [click ‘Yes’ for overplot]
  4. ‘AF_VM’
  5. ‘TF_Out_VM’ ‘PF_VM’ ‘OrificeArea’ ‘Q_Out_VM’
  6. ‘PD_In_VM’
Modeling Tutorial
Plot Simulation Data

MSC.EASY5™

- Output data automatically plotted as “displays”.
  - A display is defined to be a single page of information containing 1 to 4 plots.
  - Plots can include single plots, overlaid plots, plots vs. time, or other indep. Variables.

Plotter Window

Plot Selection Window
Simulation Results

MSC.EASY5

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (69)
Introduction to MSC.EASY5 with Emphasis on Modeling and Simulation of Fluid Power Systems

Analysis Methodology & Tools
MSC.EASY5 Analysis Methodology

Introduction

- MSC.EASY5 provides many different methods for setting up model data and analysis data.

- “Analysis Settings Files” - used to set-up and save analysis data.

- “Temporary Settings Files” - used to modify model data.

- MSC.EASY5 Plotter - powerful graphical plotting tool.
  
  - Default plots - setup and formatted by MSC.EASY5 for quick plots.

  - Custom plots - user-defined plot formats.

  - Comparison plots - automatically cross-plot results from 2 or more analyses.

  - Export capability - plot can be exported via a Word Metafile format that can be exported directly into Microsoft applications.

- Multiple Analyses - used to setup and run a sequential number of analyses, all in a single run (we may use this in a later exercise).
MSC.EASY5 Analysis Methodology
Analysis Settings Files

- Analysis Settings File can be saved to different file names by selecting the [Save] push button.
  - Analysis name is printed in the top window frame:
    Simulation Data Form - Analysis File: *simulation*
  - Default name: “simulation” for the Simulation Data Form

- You should use different names for different analyses to help organize and save output data
  - Analysis name is used to name all of MSC.EASY5’s output files.

![Select an Analysis file dialog box](image)

Model name: act.Euler.rpd  
Analysis Name: act.Gear.apl  
Plot file extension  
Listing file extension
EASY5 Analysis Methodology
Temporary Settings Files

• Partial sets of parametric and/or operating point data can be saved for re-use during future analyses. Temporary Settings Files data do not alter model data!
MSC.EASY5 Analysis Methodology
Temporary Settings are Applied to Analyses

Multiple Temp. Settings Files may be entered.
MSC.EASY5 Analysis Methodology
Create Temporary Settings Files

• Create Temporary Settings File named Pump1000 for different pump speeds.

• Create Temporary Settings File named CosAmp to change maximum opening.
Apply Temporary Settings Files to Simulation

MSC.EASY5™

- Re-run the simulation and apply Temporary Settings Files.
- Temporary Settings data is only applied to analysis — does not modify model data.

Link Temporary Settings Files to simulation
MSC.EASY5 Analysis Methodology

Analysis Input File

• Analysis Input File is created by GUI whenever an analysis is performed.

• It contains:
  • Parameter values
  • Initial conditions
  • Error controls
  • Integration controls (whether state is frozen)
  • Analysis commands

• It is executed sequentially- parameter values at the end take precedence over upstream values.

• References
  • MSC.EASY5 Reference Manual, Appendix A.
  • MSC.EASY5 Technical Note - Accessed from MSC.EASY5 Help menu.
Introduction to MSC.EASY5 with Emphasis on Modeling and Simulation of Fluid Power Systems

Modeling Manifolds:
Splits and Merges
Now, suppose that the OpenLoop model will be used to supply fluid to a series of drilled holes that serve as lubrication ports.

The resistance to fluid flow through the drilled holes of the lubrication ports can be modeled with orifices.

The fluid is then collected after flowing through the holes.
Modeling Splits and Merges

The MSC.EASY5 HC Library has a number of components in the Splits and Merges group to assist in building this kind of fluid network. There are some guidelines that will help in selecting the proper components:

1. Decide where there are unique pressures that need to be modeled:
2. Pick the correct split and merge:
Modeling Splits and Merges

3. Need more ports?
   - Use an RS to add another port to the S3 split without adding another pressure state.
   - Cascade JU merges to get more junction inlet ports.
Modeling Splits and Merges

• Use the OD orifices to model the four holes.
• Reorient (HOLD-R, select Choose Alternate Icon) the splits and merges so that the appearance is correct:
Set up four different types of resistances (see the Info page for the CD component).

OD = Sharp-edged orifice
OD2=Sharp-edged jet
OD3=Thick (tube) orifice in conduit
OD4=Discharge defined by lab test data

<table>
<thead>
<tr>
<th>Component</th>
<th>Input/Output</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>OD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DH</td>
<td>0.1</td>
<td>Hydraulic diameter</td>
</tr>
<tr>
<td></td>
<td>CD</td>
<td>0.61</td>
<td>Fixed, sharp edge restriction</td>
</tr>
<tr>
<td>OD2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DH</td>
<td>0.1</td>
<td>Hydraulic diameter</td>
</tr>
<tr>
<td></td>
<td>CD</td>
<td>0</td>
<td>Cd calculated</td>
</tr>
<tr>
<td></td>
<td>AUP</td>
<td>1.0</td>
<td>Upstream area</td>
</tr>
<tr>
<td>OD3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DH</td>
<td>0.1</td>
<td>Hydraulic diameter</td>
</tr>
<tr>
<td></td>
<td>CD</td>
<td>0</td>
<td>Cd calculated</td>
</tr>
<tr>
<td></td>
<td>AUP</td>
<td>1.0</td>
<td>Upstream area</td>
</tr>
<tr>
<td></td>
<td>ADN</td>
<td>0.1</td>
<td>Downstream area</td>
</tr>
<tr>
<td></td>
<td>L</td>
<td>1.0</td>
<td>Length of hole</td>
</tr>
<tr>
<td>OD4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DH</td>
<td>0.1</td>
<td>Hydraulic diameter</td>
</tr>
<tr>
<td></td>
<td>CD</td>
<td>0</td>
<td>Cd Calculated</td>
</tr>
<tr>
<td></td>
<td>CDT</td>
<td>Set</td>
<td>Set table of Cd vs Reynolds No.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(see next page)</td>
</tr>
</tbody>
</table>
Define Data in Tables

MSC.EASY5™

• Create a data table for the CDT input in OD4:
  1. Click once on the ‘Table of 1 var’ next to CDT
  2. Enter a table size of 5, click ‘Open’
  3. Enter the values given below:

<table>
<thead>
<tr>
<th>Re (Reynolds Number)</th>
<th>Cd (Discharge coefficient)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0.4</td>
</tr>
<tr>
<td>500</td>
<td>0.65</td>
</tr>
<tr>
<td>1000</td>
<td>0.75</td>
</tr>
<tr>
<td>10,000</td>
<td>0.8</td>
</tr>
<tr>
<td>100,000</td>
<td>0.9</td>
</tr>
</tbody>
</table>

4. Click ‘OK’
• Create an Executable
• Execute a Steady-State Analysis

• This time we want to see only the values of volumetric flow and discharge coefficient in the four sections of the lubrication subsystem.

• Use

This menu works the same way as the Plot Specification form. Pick the variables from the schematic:

Results:

<table>
<thead>
<tr>
<th>TIME</th>
<th>Q_Out_OD</th>
<th>Q_Out_OD2</th>
<th>Q_Out_OD3</th>
<th>Q_Out_OD4</th>
<th>CD0_OD</th>
<th>CD0_OD2</th>
<th>CD0_OD3</th>
<th>CD0_OD4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.</td>
<td>1.5152</td>
<td>1.5181</td>
<td>2.2180</td>
<td>1.8722</td>
<td>0.61000</td>
<td>0.61115</td>
<td>0.83290</td>
<td>0.75371</td>
</tr>
</tbody>
</table>
Schematic Formatting

- Create a submodel for the lubrication network.
- Change the title of the submodel.
- Edit the icon and draw your own, or import another icon.
- Hide the bi-directional arrowheads.
- Change the OrificeArea connection to a dotted red line.

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (87)
Building a Closed Loop

Hydraulic Model
Model 2: Closed Loop Hydraulic System

- Save your ‘OpenLoop’ model as ‘ClosedLoop’.
- Delete components TN, TN2 and the lubrication network from your model.
- Add components (HC) VR, JU, HF, RV, AS, (GP) GB and LA.
- Arrange as shown below:

![Diagram of Closed Loop Hydraulic System]

MSC.EASY5™

MSC SOFTWARE
SIMULATING REALITY

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (89)
• Make the default and ported connections as shown below and to the right.
• Make the manual connections as shown below to the right.

Ported Connections:
RV Out1->PD In
HF Out->RV In1
VR Out->PI In
VM Exit -> JU In1
VR Drain -> JU In2

Default Connections:
JU -> HF
FI -> VR
PI -> VM
AS -> HF
GB -> LA

Manual Connections:
PF_VM -> S_Feedback_GB
S_Out.LA -> DPR_PD
Selection of Components for Modeling

Components may be:

- *Adiabatic* (no heat transfer from fluid to ambient) or
- *Non-adiabatic* (heat transfer from fluid to ambient possible).

If heat gain/loss from the environment is not a major factor in the component, select the adiabatic version if available, or set heat loss coefficients to zero.

Rule No.1 for modeling

REDUCE COMPLEXITY (number of states) whenever possible.
HC Heat Exchangers

HC Library Heat Exchangers are separated into two components:
Define Model Data

Define the model data as given in the following table:

<table>
<thead>
<tr>
<th>Component</th>
<th>Input/Output</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>VR</td>
<td>PC</td>
<td>100</td>
<td>Crack Pressure</td>
</tr>
<tr>
<td></td>
<td>PFO</td>
<td>105</td>
<td>Full Open Pressure</td>
</tr>
<tr>
<td></td>
<td>AMX</td>
<td>0.05</td>
<td>Max Relief Area</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>0.25</td>
<td>Response time constant</td>
</tr>
<tr>
<td>HF</td>
<td>IND</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AHT</td>
<td>3,300</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AXF</td>
<td>5.655</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LWF</td>
<td>35.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MTW</td>
<td>4553.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VXT</td>
<td>393</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PF</td>
<td>5.0</td>
<td></td>
</tr>
</tbody>
</table>
Define Model Data

Define the model data as given in the following table:

<table>
<thead>
<tr>
<th>Component</th>
<th>Input</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>RV</td>
<td>VLQ</td>
<td>5000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>P_PressureIn</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HI</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HO</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MTW</td>
<td>14520</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AHT</td>
<td>10800</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AMN</td>
<td>1000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VLI</td>
<td>5000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VMX</td>
<td>6500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TAM</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>AS</td>
<td>IND</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TA_AirIn</td>
<td>25.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WA_AirIn</td>
<td>10.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PA_AirOut</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TYP</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AHT</td>
<td>10000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VOL</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DH</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AF</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PA_AirIn</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>
Define Model Data

Define the model data as given in the following table:

<table>
<thead>
<tr>
<th>Component</th>
<th>Input</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>GB</td>
<td>REF</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GKP</td>
<td>7.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GKF</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GKI</td>
<td>0.7</td>
<td></td>
</tr>
<tr>
<td>LA</td>
<td>GAI</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>.05</td>
<td></td>
</tr>
</tbody>
</table>
(Please wait here for the rest of the class)

• What is the Matrix Editor?
  • Interactive graphics-based editor used to define and edit vectors, matrices and tables.
  • Similar to a spreadsheet program.
  • Reference: *MSC.EASY5 User’s Guide*, Chapter 11, “Matrix Editor”.

• Demo
  • Data input methods and the data dialog.
  • Format & display - change field width, font size.
  • Resize vector, matrices, tables.
  • Delete multiple rows/columns.
  • Save and restore data.
    – MSC.EASY5 formatted data
    – MATLAB(TM) formatted data
MSC.EASY5™

- EASY5 provides “table-driven” components
  - Are used to drive your model with “real-world” data
  - Include algorithms which use linear interpolation
  - Include two types:
    - tables as function of time
    - tables as function of variables

- Time-dependent tables
  - Components T1 (1 table)
  - TA (2 tables)
  - TB (4 tables)

- Variable-dependent tables are multi-dimensional
  - FU 1-dimensional table
  - FV 2-dimensional table
  - FW 3-dimensional table

- Note: Table look-up algorithms may also be called from within FORTRAN and Macro code.
  - Example: Call 2-d table look-up: CALL FV(TABLEX, VAROUT, VAR1, VAR2, FLG1, FLG2).
  - See MSC.EASY5 Users Guide, Chapter 9, “Example of Using Data Tables in FORTRAN”.

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (97)
Data Tables

2-D and 3-D Tables - DEMO

MSC.EASY5™

• DEMO: how to load external data into a table using the Matrix Editor.
  • Reference: *MSC.EASY5 User's Guide*, “Creating External Data Files For Loading Data Into the Matrix Editor”.

• 2-D tables:
  • Use FV component.
  • 2 independent variables (S_Ind1, S_Ind2).

• 3-D tables:
  • Use FW component.
  • 3 independent variables (S_Ind1, S_Ind2, S_Ind3).

• Higher dimensioned tables
  • Special components (non-graphical) can be used to model 4-D, 5-D, 6-D, and n-D tables up to 9 dimensions.
Define Model Data

**MSC.EASY5™**

- **Create a data table for the HTC input in HF:**
  1. Click once on the ‘Table of 1 var’ next to HTC
  2. Enter a table size of 5, click ‘Open’
  3. Enter the values given below:

<table>
<thead>
<tr>
<th>W (mass flow)</th>
<th>Heat Transfer Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>1,050</td>
</tr>
<tr>
<td>20</td>
<td>1,900</td>
</tr>
<tr>
<td>50</td>
<td>3,925</td>
</tr>
<tr>
<td>100</td>
<td>4,500</td>
</tr>
</tbody>
</table>

4. Click ‘OK’

- **Create a data table for the HTC input in AS:**
  1. Click once on the ‘Table of 1 var’ next to HTC
  2. Enter a table size of 5, click ‘Open’
  3. Enter the values given below:

<table>
<thead>
<tr>
<th>W (mass flow)</th>
<th>Heat Transfer Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
</tr>
<tr>
<td>20</td>
<td>125</td>
</tr>
<tr>
<td>50</td>
<td>200</td>
</tr>
<tr>
<td>100</td>
<td>500</td>
</tr>
</tbody>
</table>

4. Click ‘OK’
MSC.EASY5™ Verify Table Data

• Click ‘Build->Create Executable’
• Verify that data entered into AS and HF tables is correct.
• Perform Plot Tables Analysis:
  • Select Analysis => Miscellaneous => Plot Tables
  • Select All option
  • Select Execute & Close push-button
• Table automatically plotted.
Steady State Analysis

- Click on ‘Analysis->Steady State
- Run a Steady State analysis; save the Final Operating Point as ‘SteadyState’
Steady State Analysis Listing File

MSC.EASY5™

/****/ STEADY STATE ANALYSIS /****/

A MAXIMUM OF 100 ITERATIONS CAN BE USED

UNTITLED

CASE NO. 2

4-NOV-2003 16:28:04

*** WARNING *** STEADY STATE FAILED TO CONVERGE IN 100 ITERATIONS.
THE FOLLOWING VALUES ARE THE RESULTS OF THE LAST ITERATION AND HAVE AN RMS ERROR OF 7212.06

Here are some suggestions to help the steady state analysis to converge:

* Try setting the SSV parameter in the FP component to another value. (See HC User’s Guide)
* Increase the number of iterations.
* Set pressure initial conditions to values that might be closer to the final operating point.
* You may want to try to solve for pressures first by setting TCX in the FP component to zero, and saving the converged operating point. Then, use this operating point as the initial one, and set TCX back to one. Consider that if you do not need a solution for temperatures, leave TCX at zero. (See ‘two-step method’ in HC User’s guide tutorial)

*** WARNING *** A fluid temperature is less than -53. degrees C, and will be limited during property evaluation.

TIME = 0.

STATES

<table>
<thead>
<tr>
<th>STATE</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFS_VM</td>
<td>9.000000</td>
</tr>
<tr>
<td>PA_AirIn_AS</td>
<td>1.012216</td>
</tr>
<tr>
<td>PF_VM</td>
<td>20.000000</td>
</tr>
<tr>
<td>SSS_JU</td>
<td>100.0000</td>
</tr>
<tr>
<td>SWP_PD</td>
<td>-1.000000</td>
</tr>
<tr>
<td>SWV_RV</td>
<td>0.</td>
</tr>
<tr>
<td>TFL_VM</td>
<td>-52.76950</td>
</tr>
<tr>
<td>TF_Out_JU</td>
<td>-52.17400</td>
</tr>
<tr>
<td>TW_PI</td>
<td>30.000000</td>
</tr>
</tbody>
</table>

EPI_GB = -0.1380829
MC_AS = 0.
PF_FI = 67.82921
PF_HF = 10.48058
PF_PI = 68.02741
PF_VI = 100.0000
PSS_HF = 10.00000
SMM_HF = 1000.000
SSQ_PI = 1.000000
SSQ_VM = 1.000000
S_Out_LA = -0.6954143
TFL_HF = -43.99243
TF_Out_PI = -55.05277
TF_Out_PI = -52.76950
TW_Heat_HF = 27.45601

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (102)
Troubleshooting

• If it doesn’t converge
  Check the rates:
    - If rates are small, increase the number of iterations and try again.
    - If it doesn’t converge after 50 or 100 iterations, it probably won’t.
  Choose a better set of initial conditions:
    - Closer to the operating point.
    - In a smoother area of the state space.

• May converge to the wrong operating point
  Use a set of initial conditions closer to the desired operating point.

• Special Component Features for Steady State
  Many Thermal/Hydraulic library components have special code to help the Steady State solver.
Two Step Steady State Solution

Try increasing the number of iterations to 300; re-run the analysis.

Two Step Method:
- Create a Temporary Settings file called ‘NoTemp’
- Add the variable TCX_FP to ‘NoTemp’
- Set TCX_FP to 0.0 (this freezes all temperature states) in ‘NoTemp’

Re-run the Steady State analysis including ‘NoTemp’.
First Step Steady State Solution

STEADY STATE ANALYSIS CONVERGED WITH AN RMS ERROR OF 2.136867E-05 IN 13 ITERATIONS TO THE VALUES LISTED BELOW.

TIME = 0.  CASE NO. 2

STATES

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFS_VM</td>
<td>9.0000000</td>
</tr>
<tr>
<td>PA_AirIn_AS</td>
<td>1.012216</td>
</tr>
<tr>
<td>PF_VM</td>
<td>20.000000</td>
</tr>
<tr>
<td>SSS_JU</td>
<td>100.0000</td>
</tr>
<tr>
<td>SWP_PD</td>
<td>1.000000</td>
</tr>
<tr>
<td>SWV_RV</td>
<td>0.000000</td>
</tr>
<tr>
<td>TFI_VM</td>
<td>30.000000</td>
</tr>
<tr>
<td>TF_Out_JU</td>
<td>30.000000</td>
</tr>
<tr>
<td>TW_PI</td>
<td>30.000000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFS_VR</td>
<td>5.6692651E-18</td>
</tr>
<tr>
<td>PF_FI</td>
<td>26.04931</td>
</tr>
<tr>
<td>SMCA_S</td>
<td>0.000000</td>
</tr>
<tr>
<td>SSS_PI</td>
<td>10.000000</td>
</tr>
<tr>
<td>SWQ_HF</td>
<td>1.000000</td>
</tr>
<tr>
<td>SWQ_VM</td>
<td>0.000000</td>
</tr>
<tr>
<td>TF_Drain_PD</td>
<td>30.000000</td>
</tr>
<tr>
<td>TF_Out1_RV</td>
<td>30.000000</td>
</tr>
<tr>
<td>TW_RV</td>
<td>30.000000</td>
</tr>
<tr>
<td>VLQ_RV</td>
<td>5000.0000</td>
</tr>
</tbody>
</table>

RATES

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFS_VM</td>
<td>0.0000000</td>
</tr>
<tr>
<td>PA_AirIn_AS</td>
<td>-3.9773171E-11</td>
</tr>
<tr>
<td>PF_VM</td>
<td>-2.1073987E-05</td>
</tr>
<tr>
<td>SSS_JU</td>
<td>100.0000</td>
</tr>
<tr>
<td>SWP_PD</td>
<td>1.000000</td>
</tr>
<tr>
<td>SWV_RV</td>
<td>0.000000</td>
</tr>
<tr>
<td>TFI_VM</td>
<td>0.000000</td>
</tr>
<tr>
<td>TF_Out_JU</td>
<td>0.000000</td>
</tr>
<tr>
<td>TW_PI</td>
<td>0.000000</td>
</tr>
<tr>
<td>ERI_GB</td>
<td>-2.2677060E-17</td>
</tr>
<tr>
<td>PF_HF</td>
<td>-1.4802974E-09</td>
</tr>
<tr>
<td>SSS_PI</td>
<td>0.000000</td>
</tr>
<tr>
<td>SSS_RV</td>
<td>10.000000</td>
</tr>
<tr>
<td>SWQ_HF</td>
<td>1.000000</td>
</tr>
<tr>
<td>S_S_out_LA</td>
<td>3.5527137E-14</td>
</tr>
<tr>
<td>TF_Out1_RV</td>
<td>0.000000</td>
</tr>
<tr>
<td>TW_Out_PI</td>
<td>0.000000</td>
</tr>
<tr>
<td>VLQ_RV</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (105)
Two Step Steady State Solution

- Re-run the Steady State analysis:
  - Remove the NoTemp settings file
  - Use ‘SteadyState’ as the Initial Operating Point
  - Save the final operating point as ‘SteadyStateT’
## Two Step Steady State Solution

STEADY STATE ANALYSIS CONVERGED WITH AN RMS ERROR OF 3.980712E-05 IN 13 ITERATIONS TO THE VALUES LISTED BELOW.

<table>
<thead>
<tr>
<th>TIME = 0.</th>
<th>CASE NO. 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATES</td>
<td></td>
</tr>
<tr>
<td><strong>AFS_VM</strong> = 9.000000</td>
<td><strong>AFS_VR</strong> = -4.6811975E-24</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RATES</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AFS_VM</strong> = 0.</td>
<td><strong>AFS_VR</strong> = 1.8724790E-23</td>
</tr>
</tbody>
</table>
Always Find a Valid Operating Point Before Simulating

MSC.EASY5™

• HC models present special challenges for finding steady-state.

• Highly nonlinear coupled differential equations in P and T.

• If energy (temperature variation) is not a consideration, freeze all temperature states using the TCX parameter in the FP component.

• Increase number of iterations.

• Use the “two-step” method.

• Set the default fluid temperature (TC) to be close to the final operating temperature.

• Starting with any (even partial) stable operating point will make it easier to find another valid operating point.
Other Important Points Concerning Steady State

- Realize that “steady state” may have no meaning for some moving parts, such as actuators or free integrators.
- Initially freeze positions of all moving masses such as actuators, valve spools, etc. to reasonable values.
- Make rough calculations for initial guesses for pressures temperatures, positions.
- Set pressures to cause initial flow in correct direction.
- In some cases, valves may shut during steady state, causing unexpected results:
Update Model with Steady-State Operating Point

MSC.EASY5™

• You should get the steady-state results copied into your model so that your model is at steady state. Two methods;
  
  # 1- Open each component, and type-in each state’s I.C. (initial condition) value obtained from the steady state.
  
  # 2- Use MSC.EASY5’s “Restore Operating Point” feature.

• Obviously, method #2 is preferred. To do this:
  
  • Select Options > Restore Operating Point.
  
  • Select “SteadyStateT” from pick list.

• This will copy in from the “SteadyStateT” file, the values of the states into the “IC Value” field of each component.
A Brief Overview of Hydraulics Theory
• Governing equations for fluid flow are represented as ordinary differential equations rather than partial differential equations.

• Fluid flow is considered one-dimensional; but this is still a relatively vigorous treatment that includes:
  – Transient energy effects
  – Fluid compressibility
  – No flow or flow reversal possibilities
  – Recognizes onset of cavitation
Governing Equations

• Conservation of Mass

• Conservation of Energy

• Conservation of Momentum

• Flow/Pressure Drop Correlations for Pipes and Orifices

• Pipe Friction Factors as a Function of Reynolds Number
Conservation of Mass

\[
\frac{d}{dt}(\rho V) = w_{in} - w_{out} \quad \text{or} \quad \rho \frac{dV}{dt} + V \frac{d\rho}{dt} = w_{in} - w_{out}
\]

\[
\frac{d\rho}{dt} = \left( \frac{\partial \rho}{\partial P} \right) \frac{dP}{dt} + \left( \frac{\partial \rho}{\partial T} \right) \frac{dT}{dt}
\]

*fluid property terms*

\[
\frac{dP}{dt} = \dot{P} = \frac{1}{V} \frac{\partial P}{\partial \rho} (w_{in} - w_{out} - \rho \dot{V}) - \frac{\partial P}{\partial \rho} \cdot \frac{\partial \rho}{\partial T} \cdot \dot{T}
\]

If energy transfer is ignored,

\[
\frac{dP}{dt} = P = \frac{1}{V} \frac{\partial P}{\partial \rho} (w_{in} - w_{out} - \rho \dot{V})
\]

or the more familiar

\[
\dot{P} = \frac{\beta}{V} (Q_{in} - Q_{out} - \dot{V})
\]
Conservation of Energy

\[
\frac{d}{dt}(\rho Vu) \approx \frac{d}{dt}(\rho Vh) = \sum (wh_i) - \sum (wh_j) + Q_f
\]

Enthalpy and internal energy rates are practically the same for liquids. (For gases, a rigorous formulation uses the definition of enthalpy, \( h = u - P/r \).)

Since enthalpy is a function of temperature only, it can be replaced with temperature:

\[
h_2 = \left( \frac{\partial h_2}{\partial T} \right) \dot{T}_2 = c_p \dot{T}_2
\]

\[
Q_f = \frac{\Delta P w}{\rho} + HA \Delta T
\]

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (115)
Conservation of Momentum, Transient Form

\[
\dot{w}_2 = \frac{A(P_1 - P_2)}{L} - \frac{w_2 \cdot |w_2|}{2\rho AD_h} \cdot f + \frac{\left( w_{1M}^2 - (w_2 \cdot |w_2|) - w_{R2m}^2 \right)}{\rho AL}
\]

- **pressure force**
- **shear force**
- **convective velocity**

\[
w_{1M} = \max (w_1, 0)
\]

\[
w_{R2m} = \min (w_{R2}, 0)
\]

\[f = \text{friction factor} = g(\text{Re}, d)\]

**Note that** \(w_2\) **is a state variable**
Orifice Flow

Turbulent

\[ w_{\text{turb}} = C_d \cdot \frac{\pi D_h^2}{4} \sqrt{2\rho |\Delta P|} \cdot \text{sgn}(\Delta P) \]

Laminar

\[ w_{\text{lam}} = \frac{\pi \rho D_h^3 C_d^2 (\Delta P)}{2\mu \text{Re}_T} \]

\( \text{Re}_T \) is the Reynolds number where transition occurs.
Switch State Representation of Orifice Flow Regimes

MSC.EASY5™

MSC SOFTWARE
SIMULATING REALITY

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (118)
Pipe Model

- Use correct type of pipe
  - Adiabatic or non-adiabatic
  - Storage/Resistive or Resistive
  - Momentum transfer ignored or considered
  - Flexible walls
  - Roughness modeled or smooth walls:

\[
f = \frac{2 \rho D_h h}{(X_L + L) (A/ \bar{w})^2} \cdot \Delta P
\]
Linear Analyses
The solution of the equation \( \dot{x} = ax \) is \( x = ce^{at} \), since by the "chain rule" we have
\[
\frac{d}{dt}(ce^{at}) = \frac{d}{dt}(ce^{at}) \frac{d}{dt} = ce^{at}a = cae^{at} = ace^{at}
\]

Since the simple linear equation has solutions of the form \( x = ce^{at} \), setting \( t = 0 \), we have
\[
x(0) = ce^{a0} = ce^0 = c1 = c
\]
If we write \( x_0 = x(0) \) for our initial condition at time 0, the solutions look like
\[
x = x_0 e^{at}
\]
From our knowledge of how exponents work, here are the two types of transient response possible:

The parameter \( a \) is called the **characteristic value** (or eigenvalue) of the equation \( \dot{x} = ax \).
Characteristic Values of a Matrix

MSC.EASY5™

For the systems of linear differential equations \( \dot{x} = Ax \)
the polynomial equation \( \text{det}(sI - A) = 0 \)
(s is a scalar unknown) is called the characteristic equation of \( A \).

Case 1:
\[
A = \begin{bmatrix} \alpha & 0 \\ 0 & b \end{bmatrix}
\quad \text{det}(sI_2 - A) = \text{det} \begin{bmatrix} s - \alpha & 0 \\ 0 & s - b \end{bmatrix} = (s - \alpha)(s - b) = 0
\]
The roots are \( s = \alpha \) and \( s = b \).

Case 2:
\[
A = \begin{bmatrix} \alpha & -b \\ b & a \end{bmatrix}
\quad \text{det}(sI_2 - A) = \text{det} \begin{bmatrix} s - \alpha & b \\ -b & s - a \end{bmatrix} = (s - \alpha)^2 + b^2 = s^2 - 2\alpha s + \alpha^2 + b^2 = 0
\]

Using the quadratic formula, the roots are
\[
s = \frac{2\alpha \pm \sqrt{4\alpha^2 - 4\alpha^2 - 4b^2}}{2} = a \pm bi
\]
The roots of the characteristic equation of a matrix are called the **eigenvalues** of the matrix.

For an $n \times n$ matrix $A$ with eigenvalues $s_1, s_2, s_3, \ldots, s_n$ (possibly not all distinct), there is a change of variables transformation $x = Ty$ so that

$$
\bar{A} = T^{-1}AT = \begin{bmatrix}
B_1 & 0 & \ldots & 0 \\
0 & B_2 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & \ldots & 0 & B_k
\end{bmatrix}
$$

where each $B_j$ is either a real eigenvalue of $A$ or a $2 \times 2$ matrix of the form

$$
B_j = \begin{bmatrix}
\alpha + \beta i & -\beta - \alpha i \\
\beta + \alpha i & \alpha - \beta i
\end{bmatrix}
$$

where $(\alpha + \beta i, \alpha - \beta i)$ are a conjugate pair of complex eigenvalues of $A$.

- The eigenvalues of $A$ are called the **modes** of the linear system $\dot{x} = Ax$.
- The matrix $T$ is called the **modal matrix** of the system.
- The columns of $T$ are called the **eigenvectors** of the system.
Why Care about Eigenvalues?

If $T$ is the modal matrix of the linear system $\dot{x} = Ax$, then the linear system

$$\dot{y} = T^{-1}ATy = \bar{A}y$$

is a set of uncoupled 1st and 2nd order linear differential equations.

The first order systems are lags, the second order systems are damped oscillators.

• Each real eigenvalue determines the rate of growth (for positive eigenvalue) or decay (negative eigenvalues) of one of the lags.

• Each complex eigenvalue pair determines both the frequency of oscillation (complex part) and the rate of growth (for positive real part) or decay (negative real part) for one of the damped oscillators.

• Each eigenvector (since it is a column in the change of variables matrix) determines which of the original physical states contribute to the $j^{th}$ real eigenvalue or complex eigenvalue pair.
MSC.EASY5™

Linear Model Generation

• Select Analysis -> Linear -> Linear Model Generation

  • Basic Form - No inputs or outputs
    – Calculates Stability Matrix [A] and eigenvalues

  • Complete Form - Specify Inputs/Outputs
    – A, B, C, and D matrices
    – Calculates eigenvalues, eigenvectors
### Linear Model Results - Eigenvalues

#### MSC.EASY5™

- **Eigenvalues:** dynamic modes of your system

#### 20 SYSTEM EIGENVALUES

<table>
<thead>
<tr>
<th>MODE</th>
<th>REAL</th>
<th>IMAGINARY</th>
<th>NATURAL FREQ.</th>
<th>DAMPING RATIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.089249E-10</td>
<td>0.</td>
<td>5.089249E-10</td>
<td>-1.00000</td>
</tr>
<tr>
<td>2</td>
<td>0.</td>
<td>0.</td>
<td>0.</td>
<td>0.</td>
</tr>
<tr>
<td>3</td>
<td>-2.250740E-03</td>
<td>0.</td>
<td>2.250740E-03</td>
<td>1.00000</td>
</tr>
<tr>
<td>4</td>
<td>-7.802131E-03</td>
<td>0.</td>
<td>7.802131E-03</td>
<td>1.00000</td>
</tr>
<tr>
<td>5</td>
<td>-9.881227E-02</td>
<td>0.</td>
<td>9.881227E-02</td>
<td>1.00000</td>
</tr>
<tr>
<td>6</td>
<td>0.163147</td>
<td>0.</td>
<td>0.163147</td>
<td>1.00000</td>
</tr>
<tr>
<td>7</td>
<td>-0.802358</td>
<td>0.</td>
<td>0.802358</td>
<td>1.00000</td>
</tr>
<tr>
<td>8</td>
<td>-1.97807</td>
<td>0.</td>
<td>1.97807</td>
<td>1.00000</td>
</tr>
<tr>
<td>9</td>
<td>-3.20201</td>
<td>0.750384</td>
<td>3.28875</td>
<td>0.973622</td>
</tr>
<tr>
<td>10</td>
<td>-4.00000</td>
<td>0.</td>
<td>4.00000</td>
<td>1.00000</td>
</tr>
<tr>
<td>11</td>
<td>-4.00000</td>
<td>0.</td>
<td>4.00000</td>
<td>1.00000</td>
</tr>
<tr>
<td>12</td>
<td>-4.12044</td>
<td>0.417302</td>
<td>4.14151</td>
<td>0.994911</td>
</tr>
<tr>
<td>13</td>
<td>-84.7072</td>
<td>561.130</td>
<td>567.487</td>
<td>0.149267</td>
</tr>
<tr>
<td>14</td>
<td>-12764.0</td>
<td>0.</td>
<td>12764.0</td>
<td>1.00000</td>
</tr>
<tr>
<td>15</td>
<td>-17688.1</td>
<td>0.</td>
<td>17688.1</td>
<td>1.00000</td>
</tr>
<tr>
<td>16</td>
<td>-33983.2</td>
<td>0.</td>
<td>33983.2</td>
<td>1.00000</td>
</tr>
<tr>
<td>17</td>
<td>-117018.</td>
<td>0.</td>
<td>117018.</td>
<td>1.00000</td>
</tr>
</tbody>
</table>
Transfer Function Analysis
Sinusoidal Response

- Select Analysis -> Linear -> Transfer Function
- Define where to "inject" Transfer Function input signal (TF Input) and where to "pick off" Transfer Function output signal (TF Output)

1. First, select data field
2. Select GB block from block diagram
3. Select name from pick menu

Or, type in name directly into field
Frequency Response in Bode Form

- **Gain margin** – factor by which gain must change to make the system unstable
  - Measure at phase of $-180^\circ$

- **Phase margin** – factor by which phase must shift to make the system unstable
  - Measure at gain of 0 dB and from phase $= -180^\circ$
MSC.EASY5 ‘Root Locus’ is Very General

• Textbook definition:

\[ \frac{G(s)}{H(s)} \] vary a separable gain

• MSC.EASY5 definition:

\[ f(K) \] vary any model parameter
Root Locus on Feedback Gain

MSC.EASY5™

• Select Analysis -> Linear -> Root Locus

• Define the parameter to vary; in this example, vary GKPGB (the proportional gain on the controller) from 0 to 500
Root Locus Plot - Upper Half of Plane
Linear Analysis: Predictions About Step Response

• Stable (Eigenvalues in Left Half Plane)
  • Linear model
  • Root Locus

• Exponential Rise (with oscillation)
  • Eigenvalues

• High Damping
  • Eigenvalues
  • Root Locus
  • Frequency Response
Confirm Prediction With Simulation

MSC.EASY5™

• Set up simulation with \textit{Stop Time} = 40.0, \textit{Time Inc.} = 0.1, \textit{Integr.} BCS-Gear.

• Define plots.

\begin{verbatim}
MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (133)

MSC.EASY5™

• Set up simulation with \textit{Stop Time} = 40.0, \textit{Time Inc.} = 0.1, \textit{Integr.} BCS-Gear.

• Define plots.

\begin{verbatim}
• Set up printed output.
• Print Results: Selected.
• Select: Show/Edit Print Variables.
• Enter names as shown.
• Set \textit{Print Rate} to 10
\end{verbatim}

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (133)
\end{verbatim}
Plot Simulation Results

MSC.EASY5™

System Temperatures

System Pressures

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (134)
Analysis Output File Contains Printed Output/Run Info

MSC.EASY5™

- Analyze Data Output Listing File - select: Analysis => Display Analysis Output Listing

<table>
<thead>
<tr>
<th>TIME</th>
<th>P_In_VM</th>
<th>S_Out_LA</th>
<th>W_Out_VM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>20.000</td>
<td>15.149</td>
<td>17.842</td>
</tr>
<tr>
<td>0.10</td>
<td>20.000</td>
<td>15.149</td>
<td>17.841</td>
</tr>
<tr>
<td>0.20</td>
<td>20.000</td>
<td>15.146</td>
<td>17.838</td>
</tr>
<tr>
<td>0.30</td>
<td>20.000</td>
<td>15.140</td>
<td>17.831</td>
</tr>
<tr>
<td>0.40</td>
<td>20.001</td>
<td>15.130</td>
<td>17.818</td>
</tr>
<tr>
<td>0.50</td>
<td>20.001</td>
<td>15.114</td>
<td>17.800</td>
</tr>
<tr>
<td>0.60</td>
<td>20.002</td>
<td>15.093</td>
<td>17.775</td>
</tr>
<tr>
<td>0.70</td>
<td>20.003</td>
<td>15.066</td>
<td>17.743</td>
</tr>
<tr>
<td>0.80</td>
<td>20.004</td>
<td>15.032</td>
<td>17.704</td>
</tr>
<tr>
<td>0.90</td>
<td>20.005</td>
<td>14.993</td>
<td>17.658</td>
</tr>
<tr>
<td>1.00</td>
<td>20.006</td>
<td>14.948</td>
<td>17.605</td>
</tr>
<tr>
<td>1.10</td>
<td>20.008</td>
<td>14.896</td>
<td>17.544</td>
</tr>
<tr>
<td>1.20</td>
<td>20.009</td>
<td>14.838</td>
<td>17.476</td>
</tr>
<tr>
<td>1.30</td>
<td>20.011</td>
<td>14.774</td>
<td>17.400</td>
</tr>
<tr>
<td>1.40</td>
<td>20.013</td>
<td>14.704</td>
<td>17.317</td>
</tr>
<tr>
<td>1.50</td>
<td>20.015</td>
<td>14.627</td>
<td>17.228</td>
</tr>
<tr>
<td>1.60</td>
<td>20.018</td>
<td>14.545</td>
<td>17.131</td>
</tr>
<tr>
<td>1.70</td>
<td>20.020</td>
<td>14.457</td>
<td>17.027</td>
</tr>
<tr>
<td>1.80</td>
<td>20.023</td>
<td>14.363</td>
<td>16.916</td>
</tr>
<tr>
<td>1.90</td>
<td>20.025</td>
<td>14.263</td>
<td>16.798</td>
</tr>
<tr>
<td>2.00</td>
<td>20.028</td>
<td>14.158</td>
<td>16.674</td>
</tr>
<tr>
<td>2.10</td>
<td>20.031</td>
<td>14.046</td>
<td>16.543</td>
</tr>
<tr>
<td>2.20</td>
<td>20.034</td>
<td>13.930</td>
<td>16.406</td>
</tr>
<tr>
<td>2.30</td>
<td>20.037</td>
<td>13.808</td>
<td>16.262</td>
</tr>
<tr>
<td>2.40</td>
<td>20.041</td>
<td>13.681</td>
<td>16.112</td>
</tr>
<tr>
<td>2.50</td>
<td>20.044</td>
<td>13.548</td>
<td>15.956</td>
</tr>
<tr>
<td>2.60</td>
<td>20.048</td>
<td>13.411</td>
<td>15.794</td>
</tr>
<tr>
<td>2.70</td>
<td>20.051</td>
<td>13.269</td>
<td>15.627</td>
</tr>
<tr>
<td>2.80</td>
<td>20.055</td>
<td>13.122</td>
<td>15.454</td>
</tr>
<tr>
<td>2.90</td>
<td>20.059</td>
<td>12.970</td>
<td>15.275</td>
</tr>
<tr>
<td>3.00</td>
<td>20.063</td>
<td>12.814</td>
<td>15.091</td>
</tr>
</tbody>
</table>
• Plotter Windows - output and spec window

• Default Plots - “.ezrpd” files (raw plot data)

• Custom Plots - “.ezrpl” files (layout)
  • “.ezrpd” versus “.ezrpl” files
  • Edit a “display” layout
  • Display Spec Form - features, functionality, usage
  • Combine data from different runs

• Comparison Plots

• Plotter window menus

• How to print plot data
Plot Simulation Results

The relief valve is not active in the model. Try lowering the relief valve cracking pressure to just below the maximum pressure of the system.

Here’s a little test of your knowledge so far:

• Create a new temporary settings file.
• Select the two parameters PC (cracking pressure) and PFO (fully open pressure) from the VR relief valve.
• Set these pressures to 13 and 15 bar respectively.
• Rerun the analysis and compare the plots. Do the results look different?
• Does the valve open? How can you tell?
  Hint: if the pressure doesn’t change, why is that?

• Now, use the default settings for the relief valve, but change the set pressure to 90 bar. Can you get to steady state? Why or why not?
Simulation and Integration Tutorial
Integration Cycle

MSC.EASY5™

• MSC.EASY5 makes use of central integrator to solve the differential equations.

• In general, code calculates only rates, integrators are responsible for setting state values and time. That means you cannot directly set state or time values.

• Fixed- and variable-step integrators are available.
Fixed-Step Integration Method

MSC.EASY5™

• Fixed-step integrators use the same time step for the entire simulation
  • In many systems, this is a severe limitation resulting in excessive run times.
  • User is responsible for setting the integration step size.

Integration Time Increment = constant

- EULER
- HUEN
- FSRK
Variable Step Integration Method

- Variable-step integrators adjust their step size as a function of system derivatives and eigenvalues.
- Simulation run-times are often several orders of magnitude lower compared to fixed-step solutions.

Integration Time Increment = f(x)

Integration Points

- VSRK
- ADAMS
- BCS-GEAR
Requirement for Variable Step Integration

MSC.EASY5™

• In order for variable-step integrator to work they must be able, at any point, to back up in time.

• Therefore, your model must be explicit, where:

\[ \dot{x} = f(u,x,t) \]

where:

- \( x \) is the state vector
- \( u \) are all model inputs
- \( t \) is ‘time’

• That is, the model implementation shall be such that for a given vector of state variables, model inputs (parameters), and a value of time, repeated calls to SUBROUTINE EQMO shall always return the same numerical value for the derivative vector.
Effect of “Time Increment” on Integrators

What value do you set “Time Increment” (TINC) to?

Fixed-Step Integration methods
- TINC: sets integrator step size ΔT (affects integration stability and accuracy), and defines the data output rate.
- Rule of thumb: TINC = 5 times smaller than smallest time period
- Example: highest mode (largest eigenvalue)= 20 rps

\[
T_{\text{period}} = \frac{1}{2 \pi \text{ (rad/cycle)}} = 0.314 \text{ secs}
\]

\[
TINC = T_{\text{period}}/5 = 0.06 \text{ secs}
\]

Variable-Step Integration methods
- TINC does not set the integrator time step; only used to define data output
- TINC affects the maximum ΔT allowed; example, BCS Gear method ΔT= 10*TINC
- Setting TINC to a small value slows down variable-step integration method
Relationship of Time Increment, Print & Plot Variables

• Time increment, plot rate and print rate are related as follows:

Time Increment: \( T_{\text{INC}} \)

Plot Rate Multiplier: \( P_{\text{LOT}} \) \( \times 2 \)

Print Rate Multiplier: \( P_{\text{INT}} \) \( \times 2 \)

0  \hspace{1cm} \text{TIME}
DEMO: Multiple Analysis

- Multiple Analysis is a way to stack many analyses:

- Explain use of “Update I.C./Operating Point” analysis type.
- Review use of Insert and Delete buttons and the “#” column to edit a multiple analysis form.
- Review how Analysis Files and Temporary Settings Files can be opened with Double-CLICK.

NOTE: data from temp. Settings files is **cumulative** - use “Undo temp setting” to remove the data settings.
Interactive Simulation

MSC.EASY5™

• Interactive Simulation: interactively change model data during simulation and view/plot results.

• See *MSC.EASY5 User’s Guide*, “Interactive Simulation”.

• Example of running IS:

View output with SC widget.

SL widget - move the slider to change the feedback gain.
Interactive Simulation
Interactive Simulation Method

• Components contained in IS - Interactive Simulation Library.

• Add IS blocks to your model.

• Edit component data and enter user-defined names (optional).

• Create Executable and run simulation.

• Launches interactive simulation as a background process:
  • IS “widgets” will pop-up - you interact with these widgets to change parameters and view data
  • This is a background process -- you can exit MSC.EASY5 and the interactive simulation will continue!
Connections:
S_Out_TG -> REF_GB
LA -> DR
S_Output_SL -> PC_VR
P_In_VM -> S_In1_SC
Interactive Simulation

Edit IS Component Data Tables

DR - Digital Readout & SL - Slider Block

Common Data:

TAU: Update rate (in seconds) - how often input is sampled.

INI or ISV: initial value

ACT: activation flag active(=1);
     inactive(=0)
Interactive Simulation
Edit IS Component Data Tables

SC - Strip Chart Block:

LEN = x axis length; time axis “traveling window”

If X is left at default value, then X axis is by default “TIME”.
TG - Toggle Switch:

![TG - Component Data Table (CDT)](image)

### Configuration

<table>
<thead>
<tr>
<th>Input Name</th>
<th>Value/Type</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Inputs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TAU_TG</td>
<td>.01</td>
<td>Output signal update interval</td>
<td>sec</td>
</tr>
<tr>
<td>S_Val1_TG</td>
<td>30.0</td>
<td>Value of toggle button 1</td>
<td></td>
</tr>
<tr>
<td>S_Val2_TG</td>
<td>40.0</td>
<td>Value of toggle button 2</td>
<td></td>
</tr>
<tr>
<td>S_Val3_TG</td>
<td>50.0</td>
<td>Value of toggle button 3</td>
<td></td>
</tr>
<tr>
<td>S_Val4_TG</td>
<td>99999</td>
<td>Value of toggle button 4</td>
<td></td>
</tr>
<tr>
<td>S_Val5_TG</td>
<td>99999</td>
<td>Value of toggle button 5</td>
<td></td>
</tr>
<tr>
<td>S_Val6_TG</td>
<td>99999</td>
<td>Value of toggle button 6</td>
<td></td>
</tr>
<tr>
<td>S_Val7_TG</td>
<td>99999</td>
<td>Value of toggle button 7</td>
<td></td>
</tr>
<tr>
<td>S_Val8_TG</td>
<td>99999</td>
<td>Value of toggle button 8</td>
<td></td>
</tr>
<tr>
<td>INI_TG</td>
<td>2</td>
<td>Initial toggle position</td>
<td></td>
</tr>
<tr>
<td>FMT_TG</td>
<td>12.05</td>
<td>Format spec: width, precision</td>
<td>used when DF=0,2</td>
</tr>
</tbody>
</table>

Secondary Inputs

<table>
<thead>
<tr>
<th>Input Name</th>
<th>Value/Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT_TG</td>
<td>Active (1)</td>
<td>Activation flag: 1=active; 0=inactive</td>
</tr>
<tr>
<td>AB_TG</td>
<td>Continue (0)</td>
<td>Auto break flag (0 = continue)</td>
</tr>
<tr>
<td>DF_TG</td>
<td>N + V (0)</td>
<td>Display flag (0: N+V, 1:N, 2:V)</td>
</tr>
</tbody>
</table>
**Interactive Simulation**

**Essential IS Component Data**

<table>
<thead>
<tr>
<th>Component</th>
<th>Input/Output</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>TG</td>
<td>TAU</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S_Val1</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S_Val2</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S_Val3</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td></td>
<td>INI</td>
<td>2</td>
<td>Port number of initial value (S_Val2)</td>
</tr>
<tr>
<td>SC</td>
<td>TAU</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LEN</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>DR</td>
<td>TAU</td>
<td>0.1</td>
<td></td>
</tr>
<tr>
<td>SL</td>
<td>TAU</td>
<td>0.1</td>
<td>Initial value</td>
</tr>
<tr>
<td></td>
<td>ISV</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAX</td>
<td>150</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MIN</td>
<td>25</td>
<td></td>
</tr>
</tbody>
</table>

**Simulation Settings**

- Set Stop Time to a large number (1e6).
- Set Plot Results and Print Results to “No” (otherwise, a large amount of data may be generated).
- Set Save Final Operating Point to “No.”

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (152)
MSC.EASY5™

Widgets:

• “CONTROL BOX” widget used to kill job.

• Interact with widgets - move, resize, etc.

• In sliders: grab & move sliders or type in data.

• Can “break” or stop interactive sim by selecting [Break] -- then “step” through sim.

• To continue, select [Continue].

• In SC widget you can print current view.
MSC.EASY5 is designed to easily answer “what if?” questions, so…

What if a large leak suddenly opened in the connector between the heat exchanger and the fluid reservoir?

• Break the connection between the heat exchanger and the reservoir

• Add a split with an orifice.

• Connect the diameter of the split to a toggle button (TG) component (making sure to set the default diameter to zero).

• Add a strip chart to plot the volume of the reservoir.

• Find a new steady state.

How long does it take for the system to fail when the connector blows out?
Adding Fortran or C Code Components
Adding a Fortran Component

- Now replace the air side heat exchanger table data with a FORTRAN component.
  - Delete the AS Air Side Heat Exchanger
  - Add FORTRAN model to schematic.
  - Open component data table.

Open component with Click-C or double-click.
 MSC.EASY5™

• First, define all input names.

  - Select Edit -> Add an Input...

  Enter up to 28 character alphanumeric name.

  Description and units are optional.

• Add more inputs:
  T_AIR_IN (temperature at inlet)
  WALL_TEMP (wall temperature)
  UAW (air/wall heat transfer coefficient)
Define FO Component Outputs

MSC.EASY5™

• Define Output Variables.
  – Select Edit -> Add a Variable...
  – Add more output names:
    T_AIR_OUT (temp. of exit air)
    HEATFLOW (heat transfer rate to fluid-side HX)
    NTU (heat transfer measure)
Set Parameter Values in Fortran Component

MSC.EASY5™

This input will be connected.

MSC.EASY5 Intro/Thermal Hydraulic Systems Class-Chart (160)
MSC.EASY5™

Input Fortran Code

• Define FORTRAN* code.
  • Enter code.

```fortran
* Enter Fortran code here...
declarations, real*8 DELTA_T, SPEC_HT
*
SPEC_HT = 1.005d3
NTU = UAW/(W_AIR_IN/60.0d0*SPEC_HT)
DELTA_T = WALL_TEMP - T_AIR_IN
T_AIR_OUT = WALL_TEMP - DELTA_T * EXP(-NTU)
HEATFLOW = W_AIR_IN * SPEC_HT * (T_AIR_IN - T_AIR_OUT)/60.0d0
```

*remember FORTRAN? It’s still a powerful way to quickly enter formulas and process data in your model.
Connect Heat Exchanger (HF) to FORTRAN component.
Connect Fortran Component

- Connect FORTRAN component to HF block.

**Connection Data Table**

<table>
<thead>
<tr>
<th>FO Source Ports</th>
<th>HF Target Ports</th>
<th>Connection Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>HeatFlow</td>
<td>IND_HF</td>
<td>HeatFlow -&gt; GW_Host_HF</td>
</tr>
<tr>
<td>T_AIR_OUT</td>
<td>INF_HF</td>
<td></td>
</tr>
<tr>
<td>NTU</td>
<td>G_W_Host_HF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TR_IN_HF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>W_IN_HF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>P_IN_HF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MID_IN_HF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TR_IN_HF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AHT_HF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AXF_HF</td>
<td></td>
</tr>
</tbody>
</table>

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (163)
• Re-do steady state.
• Run a simulation; plot the results.
Introduction to MSC.EASY5 with Emphasis on Modeling and Simulation of Fluid Power Systems

MSC.EASY5 Architecture

Files and Options
MSC.EASY5™

• Large number of files are associated with a model, and the “create executable” process.

• Files marked with an “*” are files you must keep; the others can be easily re-built.
Analysis Files

MSC.EASY5™

- Do not delete the data form files; others may be deleted and easily re-built.

Analysis Type xx suffix
- Simulation si
- Transfer Function tf
- Root Locus rl
- Steady-State ss
- Linear Model Generation lm
- Function Scan fs
- Plot Tables pt
- Eigenvalue Sensitivity es
- Stability Margins sm
- Multiple Analysis ma
- Initial Conditions cx
- Single Call sc

MSC.EASY5 Background Program

Set-up & "Save" Analysis Data Forms

open and fill-out data forms

"Execute" Analysis


build EASY5 analysis file

Print analysis results

Plot analysis results

model_name.runid.ezxx *

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (168)
EASY5 Options

For MSC.EASY5 “options” enter: easy5x -help

Options:
- easy5x -v prints the version of MSC.EASY5 –
  … or use Help->About…
- easy5x -vars prints variables you can set to define different MSC.EASY5 options
- easy5x -p file runs the plotter and plots the data in ‘file’
- easy5x -notes prints the Release Notes … or use Help->Release Notes
- easy5x -hotline displays the hotline numbers for technical support

Help->Licensed Features
  lists important license information about your license and has
  license debugging tools (or use easy5x –license)

Help->Install Demo Files
  install demonstration models for the various MSC.EASY5 libraries
  in your folder

and many more …
MSC.EASY5™

Thermal/Hydraulics Modeling and Simulation With MSC.EASY5

Building a 2-Stage Piloted Servo Valve
Example: Flows and forces on a poppet valve

- Inlet port
- Relief port
- Outlet port
- Spring force
- Damping forces
- Flow forces
- Pressure force
Plan for Design of 2-Stage Valve

- Build pilot spool valve test model.
- Build main spool model from basic blocks.
- Compare and adjust response to a known servo valve transfer function given by the manufacturer.
- Add pressure forces to main spool.
- Connect pilot spool to main spool.
- Investigate effects of breakaway (stiction) friction on main spool.
Model 3: Pilot Valve

MSC.EASY5™

1. Open a new model in MSC.EASY5 and name it ‘PilotSpool’
2. Add components FP, SQ, LA, GN, VS, and another GN
3. Arrange on your schematic as shown:
Make Component Connections

**MSC.EASY5™**

Make the default connections:

- SQ->LA
- LA->GN

Make the manual connections:

- S_Out_GN -> I_VS
- A_VS -> S_In_GN2
Parameters for MSC.EASY5 Servo Valves

MSC.EASY5™

Manufacturer’s specifications for servovalve:

2nd order Natural frequency: 35 Hz
2nd order damping: 0.8
Valve rating: 5 lpm at 200 bar pressure drop at full open.
Valve solenoid current input: -50mA to +50mA

QRT specifies orifice opening instead of KV if known.
1) 5 lpm at...
2) 200 bar delta-P...
3) For a 50 mA current

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (175)
**Add Data to the Pilot Valve Model**

Define the rest of the model data as given in the following table:

<table>
<thead>
<tr>
<th>Component</th>
<th>Input/Output</th>
<th>Value</th>
<th>User-defined Name</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQ</td>
<td>C1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>C2</td>
<td>-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>T1</td>
<td>1</td>
<td></td>
<td>Generate a 1 Hz symmetric square wave</td>
</tr>
<tr>
<td></td>
<td>T2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LA</td>
<td>GAI</td>
<td>1</td>
<td></td>
<td>Smooth output to realistic waveform</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>0.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GN</td>
<td>K</td>
<td>50</td>
<td></td>
<td>Amplify input to +/- 50 mA</td>
</tr>
<tr>
<td>GN2</td>
<td>K</td>
<td>0.01</td>
<td>Pilot_area</td>
<td>Convert area from mm$^2$ to cm$^2$</td>
</tr>
<tr>
<td>VS</td>
<td>P_Supply_VS</td>
<td>200</td>
<td></td>
<td>Bar</td>
</tr>
<tr>
<td></td>
<td>P_Return_VS</td>
<td>1.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>P_Out1_VS</td>
<td>1.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>P_Out2_VS</td>
<td>1.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FP</td>
<td>FC</td>
<td>1</td>
<td></td>
<td>SAE 30W Oil</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>40</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Completed Pilot Test Model

MSC.EASY5™

Verify that the servo valve produces the correct flow:

1) Perform a steady state analysis with the valve centered, saving the operating point.

*Hint:* Think about how you want the valve to start. Use a temporary settings file to freeze the output of the lag to zero to force a null input command to the valve.
Check the Pilot Valve Performance

2) Verify volumetric flows and that the valve works:

- Simulate from 0 to 4 seconds
- Use the initial operating point just calculated.
- Use a time step of .01.
- Plot output volumetric flows from the Out1 and Out2 ports.
Begin Model of Main Spool Valve

**Requirements:**
- Unpiloted valve spool has same second-order response characteristics as the piloting valve:
  - $w = 35$ Hz, damping = 0.8
- Valve spool weighs 1 kg.
- The valve has centering springs.
- The rated flow is 50 lpm @ 200 bar.

**Strategy:**
- Find appropriate damping factor to give correct response.
- Find input pressure force necessary to move spool to limits.
Model Mass Dynamics of Main Spool

MSC.EASY5™

This time, use a first-principles mass model to model the valve, rather than as a linear approximation. - Save the PilotValve model. - Delete the VS component.

Add the following components and arrange your schematic like this:
Connect Main Spool Model

- Connect the SS (Spring Stop) components to the FS (Force Summer).
- Connect the FS component to the PM (mass) component.
- Now, make the following custom connections:
  
  \[
  \begin{align*}
  S_{Out\_GN} & \rightarrow FC_{AltMass\_PM} \\
  AX_{Mass\_PM} & \rightarrow S_{In\_GN2} \\
  S_{Out\_GN2} & \rightarrow S_{Position\_In\_V4}
  \end{align*}
  \]

- Change the value of \( K_{\_GN2} \) from 0.01 to 100

Save your model as “MainSpool”
The ATB table governs how the valve area changes as a function of spool position. In this valve, the areas are symmetric, so only ½ the profile (0-100%) is necessary.

The maximum flow area is either input directly...

...or, calculated as in the VS component.

Change supply pressure.
For a second-order mass/spring system, the solution of the differential equation
\[ F = m\ddot{x} + kx \]
Yields the expression for the natural frequency
\[ \omega = \sqrt{\frac{k}{m}} \]
In our case, \( k = (35 \times 2\pi)^2 \times 1 = 483.6 \text{ N/cm} \)

Make sure the left spring has SGN = -1, and the right spring SGN = +1
Find Correct Damping

MSC.EASY5™

- Use Root Locus analysis to identify the correct damping

- Root Locus parameter is the viscous damping in the PM component: CHP_PM

- Try Start Value=1, End Value=10, 20 points. You can use successively finer values.

- Identify the locus traced by the oscillatory eigenvalue.

- Use the plotter locator and zoom features to hone in on a more exact value.

- Insert the correct value for damping into CHP, and set the spool mass limits on motion to –1 to 1 cm.
Suppose instead of having a known parameter and an unknown variable or state, you have a known variable or state (e.g. flow rate through a valve) and an unknown parameter (e.g. valve opening or flow conductance parameter).

**Fundamental construction:**
- Add a temporary integral controller component GI to your model (in the gp library).
- Connect the variable or state you wish had a particular value to the S1 input of the GI component.
- Set the value of the GI parameter REF to the desired value of the variable or state.
- Connect the S2 output of the GI controller up to the input parameter you don't know.
- Create a new executable. Run a Steady State Analysis.

**Why it works:**
The equation for the GI block is $\frac{dS_{Out}}{dt} = K(\text{ref} - \text{var})$, where var = input connected to GI.

At steady state (derivatives = 0) we must have $K(\text{ref} - \text{var}) = 0$ or $\text{var} = \text{ref}$. 
Using Integral Control to Identify Parameters

MSC.EASY5™

In this case, we want to identify the external force that will just move the mass to its limit of 1 cm.

Note: Before executing, use a temporary settings file to set the initial value of the state in the LA block to 0.

The answer here is trivial, but this is a powerful technique for identification of parameter values. Multiple GI blocks may be used.

**STeady State Analysis Converged With An RMS Error Of 0. In 18 Iterations To The Values Listed Below.**

<table>
<thead>
<tr>
<th>TIME = 0.</th>
<th>CONVERGED WITH AN RMS ERROR OF 0. IN 18 ITERATIONS TO THE VALUES LISTED BELOW.</th>
</tr>
</thead>
</table>

**States**

- **AX_Force_PM** = 1.000000
- **DVP_PM** = 0.
- **SWC_SQ** = 1.000000
- **SWP_PM** = 1.000000
- **S_Out_GI** = 483.6000
- **S_Out_LA** = 1.000000
- **VPC_PM** = 0.
- **SWR_V4** (1) = 1.000000
- **SWR_V4** (2) = 1.000000
- **SWS_V4** (1) = 1.000000
- **SWS_V4** (2) = 1.000000
- **SWT_SQ** = 0.
Simulate Main Spool to Check Results

• Remove the GI block and enter the correct gain in the GN block.

• Rebuild the executable and execute a steady state analysis, saving the operating point.
  – Remember to use a temporary settings file to freeze the value of the lag state S_Out to zero to get the starting position to be null (centered).
  – Check the eigenvalues. Are the frequency and damping of the oscillatory eigenvalue pair correct?

• Now, execute a simulation as with the PilotSpool model.

• Do the flows meet expectations? If not, can you diagnose the problem?
Validation by Simulation
Add Piloting Pressure Forces to the Model

MSC.EASY5™

There are two convenience components under Forces (CD and CV) that calculate pressure forces. Both determine:

- Force = pressure x area
- Volume = position x area
- Volume rate = velocity x area

Add two variable volume components (VX) and set the qualifiers as shown.
Then add a CD component.

The VX components represent the piloting volumes at each end of the spool that change with spool position. They will output a pressure, which will be converted to force by the CD component.
Add Piloting Pressure Forces to the Model

MSC.EASY5™

• When connecting the VX components to the CD component, choose the proper port. Positive means as position increases, volume increases, while connecting to the Negative port will cause the volume to decrease as position increases.
Add Piloting Pressure Forces to the Model

• Connect the CD component to the force summer.
Add Piloting Pressure Forces to the Model

MSC.EASY5™

You can tell which input connects to which chamber here.

Add parameters to the CD component.

MSC.EASY5 Intro/Thermal Hydraulic Systems Class- Chart (192)
Combine PilotSpool and MainSpool

• Copy in the PilotSpool model previously created:

Delete these (why?)

Delete the extra FP component.

Make these connections as shown

MSC.EASY5™ Intro/Thermal Hydraulic Systems Class- Chart (193)
Add Actuator and Opposing Spring Load

- Add AC Actuator and SF Spring Force components.

Extend chamber
Make connections to the actuator from the main spool as shown.
Complete Two-Stage Valve Model
• Note definitions of volumes VOE and VOR. These must be set so they are consistent with actuator areas APE and APR, and with the limits of travel XMX and XMN.

Tip: always leave at least a little volume when the piston is contacting either limit.
Add Parameters to Actuator Return Spring

MSC.EASY5™

External force on the actuator is defined as positive in the -x direction (causing the actuator to retract).

- Check the equation for output force FC to see why you must use a negative spring constant and preload.

- Always carefully check the signs of forces and motion and make sure they are consistent.
Steady State Problems with Valves and Actuators

• Build the executable.

• Try a steady state analysis, using a temporary settings file to freeze the input of the pilot valve (S_Out_LA) to zero, as we did before.

• What happens? Why?

  SINGULAR MATRIX ENCOUNTERED ON ITERATION 1. STATE PV_VXL WILL BE FROZEN AT 1.00000
  SINGULAR MATRIX ENCOUNTERED ON ITERATION 1. STATE PV_VXR WILL BE FROZEN AT 1.00000

• What is the significance of this message? How can we fix this?

• Check the pressures in the actuator, and the actuator position. Are they reasonable? How can we do better?

  Hint: where is the position of the spool and could it be in a dead zone?
MSC.EASY5™

- Simulate, starting with the calculated operating point, and plot
  - Actuator position
  - Actuator pressures
  - Volumetric Flow to the Supply and Return ports of the main spool.
Investigate Effect of Friction on Main Spool

MSC.EASY5™

Components in MSC.EASY5 hydraulic libraries that model masses often contain inputs to model coulomb and breakaway friction.

- **Coulomb friction** is a constant force opposing motion.
- **Breakaway friction** does not allow motion until the force on the mass exceeds the breakaway force.

+ Breakaway friction must also have a nonzero coulomb friction value.
+ Coulomb friction cannot be more than breakaway friction.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>Dynamic coulomb friction force mag</th>
<th>N (lbf)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCF_PM</td>
<td>0</td>
<td>Breakaway friction force magnitude</td>
<td>N (lbf)</td>
</tr>
<tr>
<td>PBF_PM</td>
<td>0</td>
<td>Linear damping coef</td>
<td>N−sec/cm(lbf−s/in)</td>
</tr>
<tr>
<td>CHP_PM</td>
<td>3.51</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Plot force on the spool mass and add breakaway friction just below the maximum value of the force. See how this changes:
  - The performance of the actuator.
  - The motion of the spool.
Investigate Effect of Friction on Main Spool

Use a multiple analysis to run cases comparing the effects of various friction types to the baseline case.

- Set up an analysis form with two plots – AX_mass_PM and AX_Piston_AC.
- Make two temporary settings files as shown.
Friction Changes Response of Actuator
Extra Topic:
Writing and Linking Code with MSC.EASY5
Introduction

MSC.EASY5™

• FO component used to enter Fortran code or call to C functions.

• CC component used to enter C code.

• MSC.EASY5 FORTRAN is a pseudo-FORTRAN.
  • Conventional FORTRAN used.
  • Special MSC.EASY5 commands used.

• Additional MSC.EASY5 FORTRAN commands are included.
  • Declaration statements used to declare special MSC.EASY5 commands.
  • Declare a variable to be a state.
  • MSC.EASY5 reserved words - cannot be used by user.
  • Reserved unit numbers.
  • MSC.EASY5 matrix notation included.

• User has access to MSC.EASY5 subroutines used in GP library.

• For complete info, see the MSC.EASY5 User’s Guide, Chapter 10.
Fortran Component Data Table

• Which variables do I add to the Component Data Table (CDT)?

• Input Parameters:
  • Inputs that will be connected to outputs of other components.
  • Parameters whose value are defined by user - may vary from run-to-run.

• Output Variables/States:
  • All states must be defined in CDT.
  • Any variable that needs to be connected to other components.
  • Any variable to be printed or plotted.

• All inputs/outputs in CDT are defined as Real Double Precision!

• All variables not in CDT are called “local variables”.
Local Variables

• Local variables: any variable not defined in CDT.
  • Typically used for temporary storage of data.

• Rules:
  • Variable name is any legal Fortran name (except MSC.EASY5 Reserved Words).
  • Fortran rules apply:
    – Size: compiler dependent (up to 255 character length?)
    – A-H, O-Z: real double precision; I-N: integer
  • Cannot print or plot local variables.
Other Considerations

MSC.EASY5™

• MSC.EASY5 uses DOUBLE PRECISION arithmetic:
  • All arguments passed (function or subroutine calls) between EQMO and user-defined routines must be double precision.
  • All literal number references must be double-precision references, e.g., “1.0D0”, not “1.0”.

• Reserved words (partial list):
  • “zero” = 0.0D0
  • “one” = 1.0D0
  • “pi” = 3.14159265358979D0
  • “rpd” = radians/degree conversion constant
  • “dpr” = degrees/radian conversion constant
  • “grm” = SI gravity constant (m/sec-sec)
  • “gre” = English gravity constant (ft/sec-sec)
  • “ezxlg” = 1.0E+36
  • “ezxsm” = 1.0E-14
Adding Non-Executable FORTRAN Statements

MSC.EASY5™

• Non-Executable Statements: DIMENSION, COMMON, DATA, INTEGER, etc.

• To include non-executable statements, use the MSC.EASY5 command:
  – DECLARATIONS, non-executable statement

• The comma following DECLARATIONS must appear.

• Supply a separate declaration command for each non-executable statement.
  For example, suppose that DUMMY is to be dimensioned (3,12), XARRAY
dimensioned to (10,10), and LOGI and ANSW are to be declared logical.

  The appropriate declarations are:

  • DECLARATIONS, DIMENSION DUMMY (3,12), XARRAY(10,10)
  • DECLARATIONS, LOGICAL LOGI, ANSW
Declaring States

MSC.EASY5™

- MSC.EASY5 allows user to declare variables as states in a FORTRAN block.
- Special declaration statements are used to define a state as follows.

  - Continuous State Declaration Statement
    - DERIVATIVE OF, state name = mathematical expression
      - For example: DERIVATIVE OF, VOLTAGE = AMPRATE * INDUCT

  - Delay State Declaration Statement
    - NEXT VALUE OF, state name = mathematical expression
      (You must also define TAU)

  - Sample-and-Hold State Declaration Statement
    - VALUE OF, state name = mathematical expression
      (You must also define TAU)
User-Defined External Code Exercise

MSC.EASY5™

• Rewrite the Heat Exchanger Fortran block in the Closed Loop Hydraulic Model so it calculates the exit temperature and heat transfer in a subroutine.

• Code:

  SPEC_HT = 1.005d3
  NTU = UAW/(W_AIR_IN/60.0d0*SPEC_HT)
  DELTA_T = WALL_TEMP - T_AIR_IN
  T_AIR_OUT = WALL_TEMP - DELTA_T * EXP(-NTU)
  HEATFLOW = W_AIR_IN * SPEC_HT * (T_AIR_IN - T_AIR_OUT)/60.0d0

• First, create an external subroutine with above code (you can export it from the Fortran block).

• Second, modify the Fortran component to call the subroutine.
• Create an external subroutine to model the heat exchange block.
  – name the file: my_heat_exch.f

** Source code:

```fortran
** my_heat_exch
* subroutine my_heat_exch(uaw, w, tin, twall, tout, ntu, q)
  *
  * uaw - wall to air conductance, (W/C)
  * w   - air mass flow rate, (kg/min)
  * tin - temperature of inlet air, (C)
  * twall - wall temperature (C)
  * tout - temperature of exit air, (C)
  * ntu - NTU's
  * q    - heat transfer rate, (W)

implicit none

real*8 uaw, tin, tout, twall, w, ntu, q
real*8 delta_t, spec_ht, wsec

* parameter(spec_ht = 1.005d3)

wsec = w/60.0d0
ntu = uaw/max(wsec*spec_ht, 1.d-10)

delta_t = twall - tin
tout = twall - delta_t * exp(-ntu)
q = wsec * spec_ht * (tin - tout)

return
end
```
External Code
Call External Routine from MSC.EASY5 Model

User supplied routines (in object form) can be called from components.

```
* Enter Fortran code here...
3
   call my_heat_exch(UAW, W_AIR_IN, T_AIR_IN, WALL_TEMP, T_AIR_OUT, NTU, HEATFLOW)
```
External Code
Linking External Routines

• Before creating the executable you must compile and link all external routines (subroutines, functions).

• External routines must be compiled before linking.
  • Enter: ‘easy5x -fc file_name.f’ (uses MSC.EASY5 predefined compiler options).
  • Output file name: file_name.o {Unix}, or file_name.obj {Windows}
  • A list of recommended compiler options is given in MSC.EASY5 Reference Manual, “Fortran Compilation Options”

• To link user-supplied external routines, select Link External Object from the Build menu.
Advanced MSC.EASY5 Programming Techniques

Forced Explicit Typing

• What is “explicit typing”?
  • Explicit (or “strong”) typing requires all variables names to be explicitly typed - that is, defined.

• Why use explicit typing?
  • Good programming practice - convention used by most languages, such as C.
  • Helps you avoid making programming errors - most common miss-typing: “zero” (0) and letter “oh” (O), or one (1) and letter “el” (l)

• How is explicit typing used in MSC.EASY5?
  • To turn this option on, select Build > Force Explicit Typing
  • When creating executable model, the compiler will check to make sure all variables are declared.
  • In MSC.EASY5 you must “declare” all local variables.

• Example: Assume TV1 and IVAL are local variables
  • DECLARATIONS, REAL*8 TV1
  • DECLARATIONS, INTEGER IVAL
  • X_out= (Xvar * TV1)/IVAL
Advanced MSC.EASY5 Programming Techniques

Initialization

MSC.EASY5™

• Use INCALL and ICCALC to initialize model parameters and states.

• INCALL
  • Flag used to indicate initial call to model EQMO - used to optimize code.
  • INCALL=2  first call to EQMO during first analysis run
  • INCALL=1 first call to EQMO during second and subsequent runs (used for multiple analyses)
  • INCALL= 0 second call and all subsequent calls to EQMO

• Example:
  • IF (INCALL .GT. 0) THEN
  •   GRAV=9.8d0
  •   END IF

• ICCALC
  • Special flag used to setup and calculate the initial conditions - only executed if a “Calculate Initial Conditions” analysis is run.
  • This will over-ride the state IC settings in the component data table.
  • ICCALC=1 only once during calculation of initial conditions; =0 all other times.

• Example:
  • IF (PRESS .EQ. ZERO .AND. ICCALC .EQ. 1) PRESS=33.45d0
Advanced MSC.EASY5 Programming Techniques
 Initialization (contin)

MSC.EASY5™

- **BEGIN/END INITIALIZATION** command
  - Used to take a block of code and move it to top of EQMO.
  - Convenient way to initialize constants, initialize parameters, check data, etc.

- **Example:**
  
  BEGIN INITIALIZATION
  
  C Check if VEL(1)=0, if so, exit. VEL(1) is used in the denominator
  
  IF (INCALL .GT. 0 .AND. vel_RB(1) .EQ. ZERO) then
    WRITE(IWRITE,+++70)
    call ezexit(4,0)
  ENDIF
  
  END INITIALIZATION

- The above block of code will get moved to the top of EQMO as the first executable lines, before the model equations begin.
• **IWRITE**: variable equal to the Fortran unit number used to write data to the Analysis Output Listing file.

• You can use this in place of the WRITE statement unit number to write output to the MSC.EASY5 output file.

• Example:

```fortran
C If DIAGNST flag is = 1, and pressure has exceeded acceptable limit,
C then print out diagnostic message & data
IF ((DIAGNST .EQ. one) .AND. (PRESS .GT. 1.5E3)) THEN
  write(IWRITE,*) '*** WARNING: PRESS has exceeded limit! ***'
  write(IWRITE,*) 'Time= ',TIME, 'Pressure= ',PRESS
ENDIF
```
MSC.EASY5™

Data Report Interval

• ITINC= flag used to determine data “time increment”
  – ITINC=1 at every “time increment”; 0 otherwise.
  – Used to ensure that data is output is at a valid reporting interval.
  – Used to streamline/optimize code to avoid unneeded computations.

• Example:
  
  IF (ITINC .EQ. 1) THEN
    COUNT = COUNT + 1
    WRITE(*,*) 'The time is: ', TIME, '*** COUNT is: ', COUNT
  ENDIF

  During simulation using variable step integration methods, model is called multiple times during a single integration step, and TIME marches forward/backwards at different intervals if using variable step integration method. You don’t wish to write out data at all integration steps, or up a counter.

• Example:
  – Add above code without the ITINC test to a model and run simulation using different integration methods: Euler, 4th-step RK, and BCS gear.
  – Run again but add the ITINC test.
  – Compare the difference. What happens? Why are results different?
• IDIAG: flag used to turn “print integration diagnostics” on/off
  – Simulation data form provides the user a means of printing integration diagnostics.
  – Problem: it prints this for the entire simulation - massive data.
  – User may only wish to view integration data over a small period of time - this can be done using the IDIAG flag to turn the print on the off.
  – Recommend using 2 input parameters to set the on/off time as follows:

• Example:

  * DIAG_ON= Time at which to turn diagnostic on
  * DIAGOFF= Time at which to turn diagnostic off
  if (TIME .ge. DIAG_ON .AND. ITINC .eq. 1) IDIAG= 1
  if (TIME .gt. DIAGOFF .AND. ITINC .eq. 1) IDIAG= 0

• This code is built in to the II (Integration Information) block in the GP Library.
MSC.EASY5™

Advanced MSC.EASY5 Programming Techniques
Terminating a Simulation

• ISTOP flag used to terminate a simulation run
  – ISTOP=1 terminates simulation; =0 otherwise
  – Note that this does not immediately “kill” the simulation; the simulation
    is allowed to complete current integration step and complete going through
    the end of EQMO, then, the simulation is terminated, plot/print files saved
    and opened files are closed.

Example:

```fortran
IF (PRESS .GT. PLIMIT) THEN
  write(IWRITE,*),'*** FATAL ERROR: PRESS has exceeded limit! ***'
  write(IWRITE,*),'Time= ',TIME,'Pressure= ',PRESS
  write(IWRITE,*),'*** Simulation will terminate! ***'
  ISTOP= 1
ENDIF
```

Example:

– Add code to write-out fatal error message, and use ISTOP to terminate
  simulation.

– Run simulation -- look at output data file.
• **iezfcx**: function that returns state index number
  — States and data are defined by 4 vectors all of same length \((n)\) and in same order:
    – \(XDOT(n)=\) state rate vector
    – \(X(n)=\) state vector
    – \(INX(n)=\) state frozen/unfrozen vector \(\{0=\text{unfrozen, } 1=\text{frozen}\}\)
    – \(XIC(n)=\) state initial condition vector
  — Use function call to iezfcx to determine the state vector index number - this can be used then to freeze/unfreeze states, extract IC values, etc.
  — Call:  `iezfcx('state name')`

• **Example: freeze state called voltage**

  * at first call to model, determine state vector number for state
  * named ‘voltage’ then freeze the state
    if \((\text{INCALL} .\text{eq. } 2)\) then
      \(KSTATE=\text{iezfcx('voltage')}\)
      \(\text{INX}(KSTATE)=0\)
    endif
Advanced MSC.EASY5 Programming Techniques
Reference Analysis Type

MSC.EASY5™

• INST: flag used to indicate which analysis is being executed
  – See the Reference Manual, “Reserved Words” for list of all flags
  – Most commonly used: simulation: INST=26 ; steady state: INST= 31

Example:

* This code takes an output variable and transforms it into a state
* using a lag to break an implicit loop.
* This code unfreezes the lag state if a simulation performed,
* otherwise, it’s frozen for all other analyses

* INPUTS:
  * statex= lag state
  * inputx= input var
  * cx = time constant

* OUTPUTS:
  * statex = lagged state output

* at first call to model, determine state vector number KSTATE
  if (INCALL .eq. 2) KSTATE=iezfcx('statex')

    statex_derivative=(inputx - statex)/tcx

DERIVATIVE OF, statex=statex_derivative
* if NOT doing a steady state, then freeze state
  if (INST .EQ.26) then
    INX(KSTATE) = 0
  else
    INX(KSTATE)=1
endif
• ez01234: subroutine used to determine type of variable
  — Used to determine if a given variable is an input parameter, table, output
  — State or output variable
  — Call: call ez01234(vname, ntype)
    – vname: variable name
    – ntype: return integer
      0= vname not found; 1=state; 2=variable; 3=parameter; 4=table

Example: determine if parameter “maxlimit” is connected;
if not connected, set to an initial value

    IF (INCALL .GT. 0) then
      call ez01234(maxlimit,Ntype)
    * if ntype=3, then name is an unconnected input parameter
      if (Ntype .EQ. 3) maxlimit=1.0e6
    ENDIF
Advanced MSC.EASY5 Programming Techniques
Calling MSC.EASY5 Table Routines

• MSC.EASY5 has built-in routines for table lookup - can be called by user
  – See MSC.EASY5 User Guide, Chapter 11, “Example of Using Data Tables in Fortran”
  – User can define an input as a 1-, 2-, 3-dimensional table; but you need routine to
    “drive” the table to extract data

• Subroutines:
  – 1-Dimensional: CALL FU (TN, DV, IDV, AN)
  – 2-Dimensional: CALL FV (TN, DV, IDV1, IDV2, AN1, AN2)
  – 1-Dimensional: CALL FW(TN, DV, IDV1, IDV2, IDV3, AN1, AN2, AN3)
    where: TN=table name; DV=dependent variable name (output);
    IDV=independent var. name(input); AN= extrapolation flag {0d0=no, 1.0d0=yes}

Example:
  * 2-D Table named TabAero
  * Indep. Vars: Mach, alpha
  * Dep. Var: Cbalpha
    CALL FV(TabAero, Cbalpha, Mach, alpha, 1.0D0, 1.0D0)
MSC.EASY5 Class Wrap-up

• What Have We Done?
  • Mechanics of Screens, Menus, ...
  • Basics of Model Building using the Thermal Hydraulic library
  • Basics of Simulation and Other Analyses

• What Haven’t We Done?
  • Sampled-Data Systems
  • Building Library Components (including Ports)
  • Details of Switch State Use
  • Modeling Logic, Time Delays

• What Should You Do Next?
  • Use MSC.EASY5
  • Keep in Touch - call the hotline (1-800-426-1443)
  • Email to easy5.support@mscsoftware.com
  • Surf our web pages:
    (http://www.mscsoftware.com/support/prod_support/easy5/ )