MSPM Changelog – records all changes made to the MSPM code by Matthias from Steven’s final version (April 2021) until…

Contents

[Matrix.m 2](#_Toc108118423)

[Function discretize 2](#_Toc108118424)

[Fin Connected Channels 2](#_Toc108118425)

[Fixing of Fin Enhanced Surface HX 2](#_Toc108118426)

[Comparison of the discretized HX types 4](#_Toc108118427)

[Sensor.m 5](#_Toc108118428)

[General 5](#_Toc108118429)

[Function getData 5](#_Toc108118430)

[Model.m 5](#_Toc108118431)

[General 5](#_Toc108118432)

[function discretize 5](#_Toc108118433)

[function show 5](#_Toc108118434)

[Known Issues 5](#_Toc108118435)

[Recursion depth when saving 5](#_Toc108118436)

[Error when solving certain gas node geometries (loops?) 7](#_Toc108118437)

# Matrix.m

## Function discretize

### Fin Connected Channels

* Perhaps missing: Solid-solid faces between skin nodes of current and previous gas node
* The way this HX type is discretized, the fins are only connected to source channel on one side. Equivalent to fin enhanced surface connected directly to source.
  + 🡪 should model several source channels. Will likely not produce reliable results as-is.
* Skin Nodes are created when discretizing, but are subsequently deleted (‘Node pairs collapsed’) if skin thickness is very small (0.8mm in this case)
  + A picture containing text, sky, screenshot

    Description automatically generated (manually plotted skin nodes during discretization)

### Fixing of Fin Enhanced Surface HX

Feb 07: Reviewed and fixed most of Matrix discretization code. Discretized successfully. Fixed all errors to be able to solve but no changes to discretization logic.

Chart

Description automatically generated Chart, line chart

Description automatically generated

* (close up of outer radius of HX, const temperature body as source on right)
* Faces seem correct in X direction only: Source to 1st fin node and gas nodes (on correct side), fins to gas, fin nodes to each other
* **Faces in Y direction irregular.** Solid nodes not connected in Y. Far top solid nodes connected to far bottom ones, as suspected by faulty code using ‘NY’ for internal faces **🡪 FIX**
* Gas nodes very thin?
* Fin nodes oddly stacked
* Solving time: ~10 hours for one setpoint!

**Feb 08**: Fixed the creation of solid faces. Removed ‘Modify node volume’ section of discretization code as I don’t know its purpose. The mesh looks to me like it should be with this section disabled.

Chart

Description automatically generated Chart, diagram

Description automatically generated Chart, line chart, scatter chart

Description automatically generated

* **Left**: Solid faces, **Middle**: Mix (solid-gas) faces, **Right**: Gas faces

### Comparison of the discretized HX types

**Fin Enhanced Surface**

Graphical user interface, text

Description automatically generated

Solving time ~ 16min / setpoint

**Fin Connected Channels:**

(collapsed node pairs are the HX skin nodes that were observed to disappear because of their small thickness)

Text, letter

Description automatically generated

# Sensor.m

## General

* Added velocity (U) and Reynolds number (Re) to options for sensor variables
* Re plots validated with hand calculation, values make sense
* U sensor plots show 0 everywhere, seems that U is not used by nodes
  + U may not be velocity, but internal energy in a solid node. Therefore plot shows zero for gas nodes
* Added 'Interpreter','none' to titles and labels for display of underscores

## Function getData

Fixed turbulence sensor error because of typo (‘Turb’🡪’turb’)

# Model.m

## General

* Changed default ‘AnimationLength\_s’ to 10 seconds
* Added Boolean properties for display of node centers, bounds and faces by type in GUI: showFacesGas, showFacesSolid , showFacesMix , showFacesLeak , showFacesMatrixTransition, showFacesEnvironment, showNodeBounds , showNodesSVGN ,showNodesVVGN ,showNodesSAGN ,showNodesSN ,showNodesEN
  + Did not add these to ‘showOptions’ yet

## function discretize

* Inverted the signs assigned to ‘Sim.ToEnvironmentSolid’ and ‘Sim.ToEnvironmentGas’ to correct the sign of the 'statistics.ToEnvironment' output. It was observed to have a sign opposite to the expected. This change is in line with the signs assigned to ‘ToSource’ and ‘ToSink’.
  + 🡪 The positive flow through a face is defined as from node 1 to node 2.

## function show

* Added code for Node Bounds (‘showNodeBounds‘): Obtains node xmin,xmax,ymin,ymax and displays node outlines as boxes in GUI based on settings selected in GUI
* Expanded functionality of ‘showInterConnections’ and ‘showNodes’ to display node centers and connections (faces) in different colors and based on their type, as selected in new GUI options. Colors and circle sizes adjustable in code.
  + Explored several options for displaying node bounds, e.g. using ‘rectangle’. However, plotting each node as separate GUI object makes the process slow as MATLAB is much faster at vectorized operations. Thus, all nodes are plotted as a single ‘line’ object which is much faster. Can plot multiple lines as one object by adding ‘NaN’ values in the data given to ‘line’.

# Known Issues

## Recursion depth when saving

* When using newer MATLAB version (I used 2020b Update 6) the model file gets unusable when it is saved. Following errors appear. It seems that the new versions don’t want to save any objects that have an infinite recursion depth, as is the case with the Nodes, Faces, Bodies of an MSPM model. This means that an object contains references to itself and you can get into an infinite loop when opening it in the workspace, e.g.   
  Model.Groups.Nodes(1, 1).Body.Group.Nodes(1, 1) and so on

Warning: While saving an object of class 'Node':

Recursion limit exceeded when saving instance of class to a MAT-file. This is because the class definition contains an

error.

> In Model/saveME (line 4164)

In SimulationInterfaceV5>saveModel (line 1066)

In SimulationInterfaceV5>save\_Callback (line 956)

In gui\_mainfcn (line 95)

In SimulationInterfaceV5 (line 42)

In matlab.graphics.internal.figfile.FigFile/read>@(hObject,eventdata)SimulationInterfaceV5('save\_Callback',hObject,eventdata,guidata(hObject))

Warning: While saving an object of class 'Face':

Recursion limit exceeded when saving instance of class to a MAT-file. This is because the class definition contains an

error.

> In Model/saveME (line 4164)

In SimulationInterfaceV5>saveModel (line 1066)

In SimulationInterfaceV5>save\_Callback (line 956)

In gui\_mainfcn (line 95)

In SimulationInterfaceV5 (line 42)

In matlab.graphics.internal.figfile.FigFile/read>@(hObject,eventdata)SimulationInterfaceV5('save\_Callback',hObject,eventdata,guidata(hObject))

Warning: While saving an object of class 'Body':

Recursion limit exceeded when saving instance of class to a MAT-file. This is because the class definition contains an

error.

> In Model/saveME (line 4164)

In SimulationInterfaceV5>saveModel (line 1066)

In SimulationInterfaceV5>save\_Callback (line 956)

In gui\_mainfcn (line 95)

In SimulationInterfaceV5 (line 42)

In matlab.graphics.internal.figfile.FigFile/read>@(hObject,eventdata)SimulationInterfaceV5('save\_Callback',hObject,eventdata,guidata(hObject))

Warning: While saving an object of class 'Mesher':

Recursion limit exceeded when saving instance of class to a MAT-file. This is because the class definition contains an

error.

> In Model/saveME (line 4164)

In SimulationInterfaceV5>saveModel (line 1066)

In SimulationInterfaceV5>save\_Callback (line 956)

In gui\_mainfcn (line 95)

In SimulationInterfaceV5 (line 42)

In matlab.graphics.internal.figfile.FigFile/read>@(hObject,eventdata)SimulationInterfaceV5('save\_Callback',hObject,eventdata,guidata(hObject))

## Error when solving certain gas node geometries (loops?)

Array indices must be positive integers or logical values.

Error in Simulation/KValue (line 2463)

K(i) = ((1-ME.Fc\_W(fc))\*ME.Fc\_fFunc\_l{fc}(ME.Fc\_RE(fc)) + ...



* This is about calculating friction for the gas faces. All variables involved have length = N(gas faces).
* Reason: the functions used here (‘ME. Fc\_fFunc\_l’, ‘ME. Fc\_fFunc\_t’) are empty for some indexes. The variables are NaN for some or all indexes.
* Occurs when solving model with **the last of the following** geometries of appendix gap gas nodes:

Chart

Description automatically generated with medium confidence

Appendix gap

HX inlet

HX inlet

Appendix gap

**Example A**

EP1 model (Steven): Works. 0 loops of gas nodes because there is no face between the appendix gap nodes and the HX inlet nodes. Also the appendix gap is sealed by a lip so no flow can pass through it.

(number of loops shown in command line when discretizing)

This is how the gas nodes should be connected IMO. Gas can always flow between appendix gap and the pink expansion / compression space nodes.

Graphical user interface

Description automatically generated

**Example B**

Raphael model, regular configuration: Appendix gap not sealed, 1 node. Works.

Problem: appendix gap node only connected to HX inlet nodes, but not to expansion and compression space nodes.

🡪 No flow can pass appendix gap unless Displacer is at TDC or BDC (shown here) and the faces between appendix gap and HX inlet nodes are active.

Chart

Description automatically generated with medium confidence

**Example C**

Raphael, appendix gap node split into small nodes (top and bottom) that form faces with expansion / compression space, and large node between them.

3 loops. Works only with fix to ‘Simulation. solve\_loops’. Runtime much longer than previous model.

Problem: The large Appendix gap center node is still connected to the HX inlet nodes, thus there are 3 loops.

Diagram

Description automatically generated with medium confidence

**Example D**

Raphael, appendix gap node split (as previous), now with displacer seal (red circle).

2 loops. **Does not work.** This is the configuration that leads to the error described above.

Problem: The split appendix gap nodes are needed to make the face between exp/comp space and the appendix gap, but they also introduce the 2 loops that seem to cause the error. In the EP1 model (shown 1st) these loops don’t occur with the same configuration of nodes.

Graphical user interface

Description automatically generated with medium confidence

**Solution: With DP seal, but app gap nodes not split. 0 loops.**

# Notes on model usage

## Influence of the appendix gap and gas node ‘loops’

* The appendix gap, its gas nodes, and the faces that connect them are critical and influence the model result and can cause errors when solving (as seen in section ‘Known Issues’).
* Example for difference in indicated work between two models that differ only in the appendix gap nodes: Model 1 has simple, single appendix gap node. Model 2 has split appendix gap node (like Example C in previous section)
* The number of loops (displayed in command window when discretizing) significantly influences the solving time. With 0 loops (e.g. with appendix gap sealed) the model solves much faster than with loops.