Building a System to Perform Fluid Dynamics Simulations and Experiments

OU Supercomputing Symposium
Oct. 6, 2010

Willy Duffle, Chris Kiser, Grant Armstrong, Andrew Baker, Andrew Henderson, Alex Macharia, Lamar Williams, and Evan Lemley

Department of Engineering and Physics
University of Central Oklahoma

Dimitrios Papavassiliou, Henry Neeman, Ed O'Rear
University of Oklahoma
Generate Junction Geometry

Junction code (C++) → SolidWorks VBA

Simulations

Generate Mesh
Junction Code → GAMBIT

Simulate Flow
Junction Code → FLUENT

Analyze Results
Junction Code → tsv

Experiments

Milliscale

Microscale

Graph showing data plot
Overview

Background
Simulations
Flow Through Porous Media (FTPM)
Flow in Junctions/Microjunctions
Microelbows
Entrance Length in Microtubes
Reynolds Number Dependence tees and wyes

Experimental
Millijunctions & MicroJunctions
Current and Future Directions
Bio-Scaffolds
Renal Artery Aneurysm
Microtubes
Porous Media

Highly porous magnesian limestone. (www.dawntnicholson.org.uk)

Microfluidic Devices

Microfluidic Valve Structure. (http://www.cchem.berkeley.edu/sjmggrp/people/boris/boris.htm)
Artificial Porous Media

Packed Beds, Gas and Liquid Filters

Sphere sizes $\mu$m to cm

Hold-up for chemical reaction, thermal processing, or filtering
Basics of Porous Media

Low Speed Flow – Darcy’s Law

\[ \frac{\phi}{\kappa} = \frac{\mu}{\kappa} u \]

\[ p = \text{pressure} \quad x = \text{position} \]
\[ \mu = \text{viscosity} \quad \kappa = \text{permeability} \]
\[ u = \text{filtration velocity} \]

High Speed Flow – Forchheimer’s Law

\[ \frac{\phi}{\kappa} = \frac{\mu}{\kappa} u + \rho \beta u^2 \]

\[ \rho = \text{density} \]
\[ \beta = \text{Forchheimer's Coefficient} \]

Packed Beds – Ergun’s Equation (empirical)

\[ \frac{-\Delta p}{\Delta L} = \frac{150}{d_p^2} \frac{\mu}{\phi^2} (1-\phi)^2 u + \frac{175}{d_p \phi^3} \rho (1-\phi) u^2 \]

\[ \phi = \text{porosity} \]
\[ d_p = \text{mean sphere diameter} \]
Flow Through Porous Media

Collaborative Effort with Dimitrios Papavassiliou and Henry Neeman from OU (began Fall 2004)
Simulation of Flow of Fluids through Porous Media

Flow Network Analysis

Design and Analysis of networks depends on knowledge of flow and energy losses in arbitrary branches. No systematic studies to generalize these bifurcations.
Porous Network Simulator
(Collaboration with Univ. of Oklahoma)

3D Monte Carlo networks from normal, beta, or empirical distribution (pore size pdf)
Coordination Number (1, 2, 3)
number of pores entering and leaving a junction $\theta \pm 90^\circ$

Projection on the xy plane of a 3D network that has 200 entry points at x=0, porosity equal to 10% and a range of $\pm 60^\circ$ relative to the x axis and $\pm 30^\circ$ relative to the y axis.
FTPM Results

Beta vs. k for phi from 0.02 to 0.26 glass beads

Tests from 26JUL06 and 28MAR06

- FTPM Runs 26JUL06
- beta = 6.092e-6*k^-1
- FTPM Runs 28MAR06
- Ergun’s Equation
FTPM Results

Forchheimer Coefficient versus Permeability Comparison of Simulation to Empirical Results

- **Sandstone Data of Jones**
- **Simulation Results (Porosity 7.3% - 24.8%)**
Literature

T's and Y's – limited configurations and most are for turbulent flow


Laminar loss coefficients and elbows, reductions, contractions – much larger loss coefficients than turbulent case – strong dependence on Reynold's number.


Importance of roughness at microscale.
**Problem Description**

Stagnation Loss Coefficient

\[ K_2 = \left[ \frac{p_1}{\rho} + \frac{u_1^2}{2} \right] - \left[ \frac{p_2}{\rho} + \frac{u_2^2}{2} \right] \]

Parameters: \( \frac{d_2}{d_1}, \frac{d_3}{d_1}, \theta_2 \) and \( \theta_3 \)

\( f_2 \) (this sets \( f_3 \)) – (why? other literature and possibility of simulations where this is unknown initially)
Generate Junction Geometry
Junction code (C++) → SolidWorks VBA

Simulations
Generate Mesh
Junction Code → GAMBIT

Simulate Flow
Junction Code → FLUENT

Analyze Results
Junction Code → tsv

Experiments
Milliscale
Microscale

Graph showing data analysis


Custom Code was written to:

(a) create GAMBIT journal files that instantiate the desired geometry based on existing 2D geometries.
(b) create a script that loads journal files into GAMBIT and meshes
(d) create all necessary preprocessing files for FLUENT.
(e) create post-processing files for FLUENT results and to tabulate results for a complete set of runs
Solution Methodology

2D Geometry Generalization

\[ L_1 = 5 \ d_{\text{max}} \]
\[ L_2 \text{ and } L_3 = 10d_{\text{max}} \]

If \( d_2 > d_{\text{avg}} \),
then \( r_2 = 3d_2 \);
else \( r_2 = 2d_2 \)

If \( d_3 > d_{\text{avg}} \),
then \( r_3 = 3d_3 \);
else \( r_3 = 2d_3 \)

\[ r_4 = \frac{d_{\text{max}}}{2} \]

Generalized Geometry
larger and smaller outlet ducts; 2Dim. - 3Dim are underway
avoid sharp edges; 5 – 90 degrees for angles
3D Geometry – Mark I
Simulation Parameters

- \( \text{Re}_D \) (Reynolds Number = proportional to speed) was maintained at constant value at the inlet duct
- \( d_1 \) was 30 microns. The fluid was chosen to be liquid water at 20˚C. The inlet flow velocity, \( u_1 \), was set to 0.5 m/s giving a Reynolds number of 15
- \( \text{Le}_D = 0.06 \text{Re}_D \) D --- gives 0.9 D for \( \text{Re}_D = 15 \)
- FLUENT output files contain surface averaged static pressure and magnitude of flow velocity at duct cross-sections at the duct inlets and exits.
Duct Inlet/Outlets

Duct inlet and exit sections considered to be where geometry of duct is the same as the downstream portion for outlets and upstream portion for inlet.
Mesh Automation

Mesh was set to 1/4 of smallest duct
Tetrahedral Mesh
Large number of tests to assess ability to generalize the mesh (1/4 factor determined in this manner)
Some testing to verify no change in results with change in mesh size.
Inlet was specified as velocity inlet
Outlets were outflow boundaries – allowed specification of flow fraction
Numerical Methods

Finite Volume solution of integral Navier Stokes Steady-State 3D Implicit SIMPLE for pressure velocity coupling 1st order upwind scheme of momentum discretization Max number of iterations Convergence criterion = 0.1%
Parameter Values

d_2/d_1 and d_3/d_1 = 0.5, 1.0, 1.5
f_2 = 0.1, 0.3, 0.5, 0.7, 0.9
θ_2 and θ_3 = 5°, 25°, 45°, 65°, 85°

600 runs attempted – 475 completed (*geometry issues on remainder*)

Suite of C++ procedures to create geometries, input files, read and collate results
Create GAMBIT script to create geometries
Create input files for GAMBIT and FLUENT
Read results files for static pressures and velocities averaged over surfaces in and out of junctions.
Fluent Result

\( f_2 = 0.1, \ \frac{d_2}{d_1} = 0.5, \ \theta_2 = 5^\circ, \ \frac{d_3}{d_1} = 0.5, \ \text{and} \ \theta_3 = 45^\circ \)
$f_2 = 0.1, \theta_2=45^\circ, \theta_3=45^\circ, \frac{d_2}{d_1}=0.5, \frac{d_3}{d_1}=1.5.$

$K_2 = 5.47$
\[ f_2 = 0.3, \theta_2=65^\circ, \theta_3=45^\circ, \frac{d_2}{d_1}=0.5, \frac{d_3}{d_1}=1.5. \]

\[ K_2 = 11.6 \]
\[ f_2 = 0.5, \ \theta_2=65^\circ, \ \theta_3=45^\circ, \ \frac{d_2}{d_1}=0.5, \ \frac{d_3}{d_1}=1.5. \]

\[ K_2 = 18.4 \]
f_2 = 0.7, \theta_2=65^\circ, \theta_3=45^\circ, d_2/d_1=0.5, d_3/d_1=1.5.

K_2 = 25.7
**2D vs. 3D Differences**

For \( d_2/d_1 = 0.5, \ d_3/d_1 = 0.5, \ \theta_3 = 5\) degrees:

\[
\begin{bmatrix}
    k_2^{\text{3D}} \\
    k_2^{\text{2D}}
\end{bmatrix}
\text{average} = 1.55 \quad (\text{std. dev.}) = 0.45
\]

For \( d_2/d_1 = 0.5, \ d_3/d_1 = 1.5, \ \theta_3 = 45\) degrees:

\[
\begin{align*}
  k_2 &= \frac{\Delta p/\rho}{\frac{u_1^2}{2}} - \frac{u_2^2}{u_1^2} + 1 \\
\end{align*}
\]
Generate Junction Geometry
Junction code (C++) → SolidWorks VBA

Simulations

Generate Mesh
Junction Code → GAMBIT

Simulate Flow
Junction Code → FLUENT

Analyze Results
Junction Code → tsv

Experiments

Milliscale

Microscale

Analysis Graph
3D Mark II Junction

Junction Without Surface Fills

Junction With Surface Fills
Initial Simulations – 3D Mark II

Small batch
Comparable results to original 3D runs

Reynolds number of 14 inlet diameter of 30 microns
Pipe Mesh Testing

Database Interface

Pipe Results Database

- Flow Parameters
  - Reynolds
    - 25, 50, 100, 200, 500, 1000
  - Diameters: 3
  - Lengths: 30, 50, 150, 300, 450, 600

- Edgemesh Parameters
  - Type: successive bottleneck
  - Ratio: 0.8
  - Spacing Type: 0
  - Spacing: 10, 20, 50, 100, 30, 40

- Face mesh Parameters
  - Type: cooper
  - Spacing Type: 1
  - Spacing: 2

- Volume mesh Parameters
  - Type: cooper
  - Spacing Type: 1
  - Spacing: 2

Valid Runs

- Reynolds: 500, 3, 150
- Edge Params: bottleneck 0.0 0.80
- Face Params: map 0.90
- Volume Params: cooper 1.2

Get Data

Plot Preview

Rakes (Reynolds = 500)
Pipe Mesh Testing

Flow-aligned hex core
Varied axial and radial spacing
MySQL results database
Automated
Results database easy to setup for junction runs
Pipe Mesh Testing

“Bell Shaped” Axial Spacing
Pipe Mesh Testing

Uniform Axial Spacing
Entrance Length in Microtubes

0.50*L

Uniform Inlet Profile

Fully Developed Paraboloid Profile

Circumferential Spacing
Entrance Length Results
Entrance Length Results
Loss Coefficients in Microelbows

\[ K = \frac{\Delta p}{\frac{1}{2} \rho V^2} \]
Microelbow Loss Coefficients

\[ K = \frac{\Delta p}{\frac{1}{2} \rho V^2} \]
Tee and Wye Reynolds Number Dependence
Loss Coefficient Versus $Re = \text{proportional to speed}$
Millijunction Experiments
Junction and Measurements
Here is the Solidworks model as well as the physical realization of our efforts to create a useable bifurcation.

Upstream from the head tank and downstream from the junction are our new, highly sensitive needle valves used to adjust the head tank flow and the flow fraction through the bifurcation.

The flow straightener we designed has a replaceable core made of glass micropipettes that can be removed completely for unrestricted flow.
Collecting Data

Sensor
- Submersible
- Internal Instrumentation Op-Amp

Filter
- 750Hz Low Pass
- Reduces Noise from 10mV to 2mV

ADC
- 12 bit
- 11 Channel

Basic Stamp
- Easy to program
- Affordable

Computer
- Python
- Real Time Plotting with GNU Plot

```python
if __name__ == '__main__':
    serial = connect_serial()
    fileSave = csv.writer(open("data.csv", "wb"))
    while True:
        data = serial.readline().split()
        fileSave.writerow(data)
        plot(data)
```
Basic Stamp
Passive Filter
Pressure Sensors
Flow Sensors
Ruby Interface
Microscale Junction Experiments

Stereo Lithography Technique

Diameter = 0.75 mm
Results - Simulations/Milliscale/Microscale

![Graph showing results of simulations and experiments for different values of Re and k2.](image-url)
Using Network Simulations to Understand Non-Darcy Flow

Objectives

- Develop an algorithm to create, mesh, and perform CFD on simplified models of real porous media networks.
- Equal number of entry and exit pores (no splits)
- $90^\circ$ elbows only
- No overlap of pores within media
- Compare results of CFD to FTPM and empirical data in literature.
- Modify algorithm to allow for complex models of real porous media.
- Splits with arbitrary angles
- Overlaps
Using Network Simulations to Understand Non-Darcy Flow

Implementation of Objectives

Use custom codes to extract FTPM networks from code
Implementation of Objectives

Use Solidworks to create network designs

Phase 1

- 90° elbows
- No overlapping

Phase 2

- Arbitrary angles
- Possible overlaps
Using Network Simulations to Understand Non-Darcy Flow

Implementation of Objectives

Use Gambit to mesh networks

Close up of 90° elbow

Close up of arbitrary junction
Implementation of Objectives

Use Fluent (CFD) to obtain $\kappa$ and $\beta$ from pressure and velocity data
Bio-Scaffold Testing
Renal Artery Aneurysm Experiments
PDMS Microtubes
Acknowledgments

The Office of Research and Grants at the University of Central Oklahoma is acknowledged for support of this research. The Donors of The Petroleum Research Fund, administered by the American Chemical Society, are also be acknowledged for support of this research through grant PRF# 47193-B9. National Science Foundation EPSCoR Research Opportunity Award Program
Flow in microfluidic flow networks and flow in porous networks are of interest in many engineering applications. Applications include porous media, micro-power generation, biomedical, computer chips, chemical separation processes, micro-valves, micro-pumps, and micro-flow sensors.


Flow in these applications is usually laminar