Multiphase flows in Fluent

using Volume of Fluid

Willem van Engen

Volume of Fluid

- interacting, non-penetrating phases
- extra computation variable:

  volume fraction

$$\alpha_w = 0 \quad \text{cell contains just air}$$

$$\alpha_w = 1 \quad \text{cell contains just water}$$

$$0 < \alpha_w < 1 \quad \text{cell contains interface}$$

Multiphase workflow

- Enable multiphase model
- Define materials
- Set initial phase
  - Areas from mesh
  - Marking rectangular/circular areas
  - Using custom field function

Multiphase post-processing

- Contours of volume fraction
- Phase-specific velocity vectors
- Reporting flux/force/flow rate for phase

Multiphase example

Contours of Velocity Magnitude (mixture) (m/s) (Time=2.3600e+00)

May 3, 2006

FLUENT 6.2 (2d, segregated, vof, lam, unsteady)

Automation in Fluent & Gambit

with journals and scheme

Willem van Engen

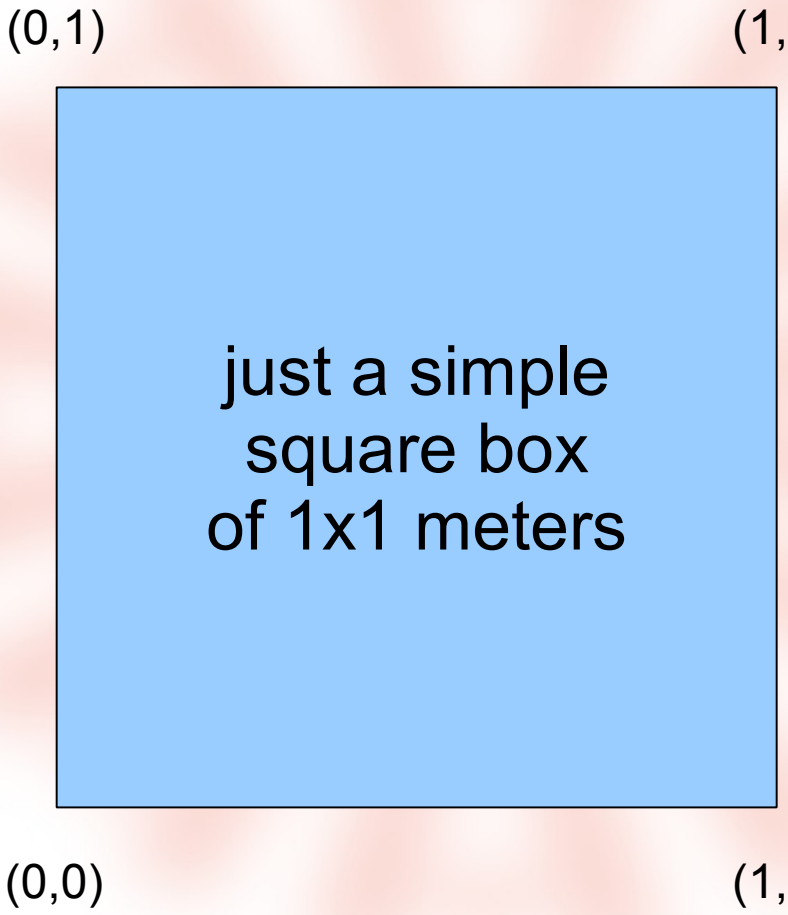
Automation facilities

- Journals (Gambit&Fluent)
  - record, replay & edit
- Scheme programs (Gambit)
  - full programming language

Gambit journals

- Just a list of commands from GUI
- Variables ($a=1) and statements
- Automatically saved (*.jou)
- Easily saved/run from GUI/command-line

Gambit journals – an example (1)



Gambit journals – an example (2)

```
/ Declare variables
$gridsize = 0.01

/ Create geometry
vertex create coordinates 0 0 0
vertex create coordinates 1 0 0
vertex create coordinates 1 1 0
vertex create coordinates 0 1 0
edge create straight \
     "vert.1" "vert.4" \
     "vert.3" "vert.2"
edge create straight \
     "vert.2" "vert.1"
face create wireframe \
     "edge.1" "edge.2" \
     "edge.3" "edge.4" real
```

```
/ Create mesh
face mesh "face.1" map size $gridsize

/ Specify boundary conditions
physics create "bottom" btype "WALL" \
        edge "edge.4"
physics create "sides" btype "WALL" \
        edge "edge.1" "edge.3"
physics create "top" btype "WALL" \
        edge "edge.2"

/ Export mesh!
export fluent5 "output.msh" nozval
```

Fluent journals

- Save chain of actions for exact replay
- Gets ugly soon
- Unsuitable for more general use
- Easily created/run from GUI/command-line

Fluent scheme programming

- Scheme (lisp) programming language
- Access to both GUI and text commands
- Hard to learn, sparse documentation
- Possible to hook into GUI

Fluent scheme programming - uses

- Set window title based on options
- Setup a new simulation for a mesh with one command
- Create a movie with overlay plots

Fluent scheme programming - example

```
; This is a comment. I love comments.

(define timestep 0.01)
(define iterations (/ 60 timestep))
(define outputmovie "coolthing.avi")
(define meshfile "model/thesetup.msh")
(define title (format #f "~a; timestep ~a"
                                    outputmovie timestep))

(load "initial-setup.scm")
(load "create-movie.scm")

(ti-menu-load-string (format #f
              "/solve/dual-time-iterate ~d 50" iterations))

(movie-finish)
(exit)
```

For this presentation, a list of links and scheme files,

see my website

http://willem.engen.nl/uni/fluent/