Equation-Based Modeling

COMSOL Conference 2017 Singapore

Pratyush Sharma

$$\rho \frac{\partial^2 \underline{u}}{\partial t^2} = \nabla \cdot \underline{\underline{S}} + F_v$$

$$\rho C_\rho \frac{\partial T}{\partial t} + \rho C_\rho \underline{u} \cdot \nabla T + \nabla \cdot (-\kappa \nabla T) = Q$$

A Look Under the Hood

$$\rho(\mathbf{u}\cdot\nabla)\mathbf{u} = \nabla\cdot\left(-p\mathbf{I} + \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I}\right) + \mathbf{F}$$

$$\nabla\cdot(\rho\mathbf{u}) = 0$$

$$\left[\frac{-\hbar^2}{2m}\nabla^2 + V\right]\Psi = i\hbar\frac{\partial}{\partial t}\Psi$$

$$\rho C_p\frac{\partial T}{\partial t} + \rho C_p\mathbf{u}\cdot\nabla T = \nabla\cdot(k\nabla T)$$

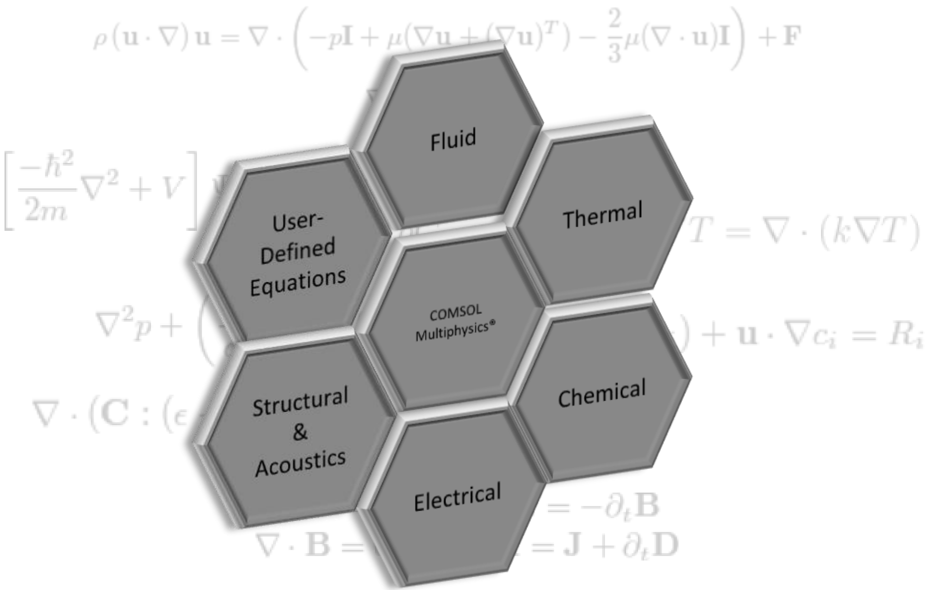
$$\nabla^2 p + \left(\frac{\omega^2}{c_s^2}\right)\frac{p}{\rho} = 0$$

$$\nabla\cdot(-D_i\nabla c_i) + \mathbf{u}\cdot\nabla c_i = R_i$$

$$\nabla\cdot(\mathbf{C}:(\epsilon - \epsilon_{\mathbf{0}} - \epsilon_{\mathbf{th}}) + \sigma_{\mathbf{0}}) = \mathbf{F}$$

$$\begin{aligned} \nabla\cdot\mathbf{D} &= \rho & \nabla\times\mathbf{E} &= -\partial_t\mathbf{B} \\ \nabla\cdot\mathbf{B} &= 0 & \nabla\times\mathbf{H} &= \mathbf{J} + \partial_t\mathbf{D} \end{aligned}$$

A Look Under the Hood



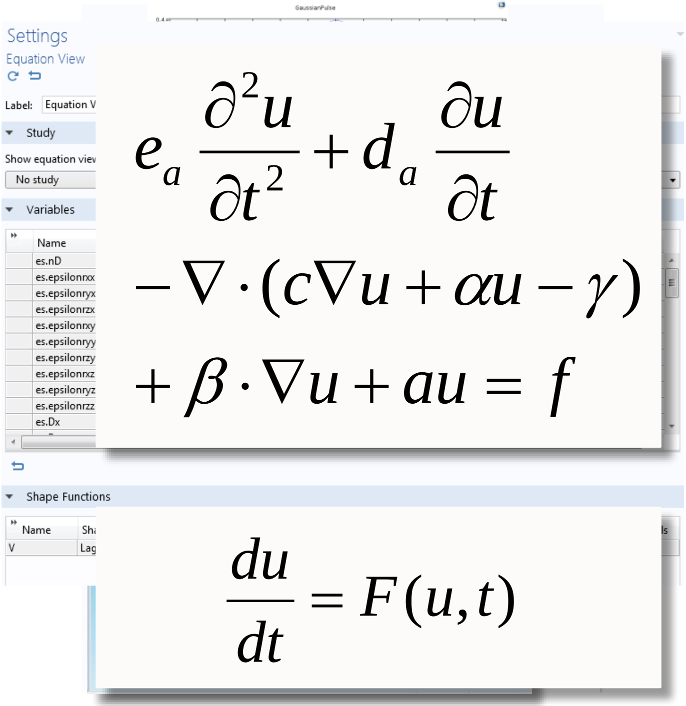
Modeling Approaches

- Insert predefined physics
- Coupling physics (manual or automatic)

- Enter user defined functions. All expressions can depend on all the variables introduced (analytical, interpolation from file …)

- Add or change terms of the equations set in the physics (and multiphysics) nodes

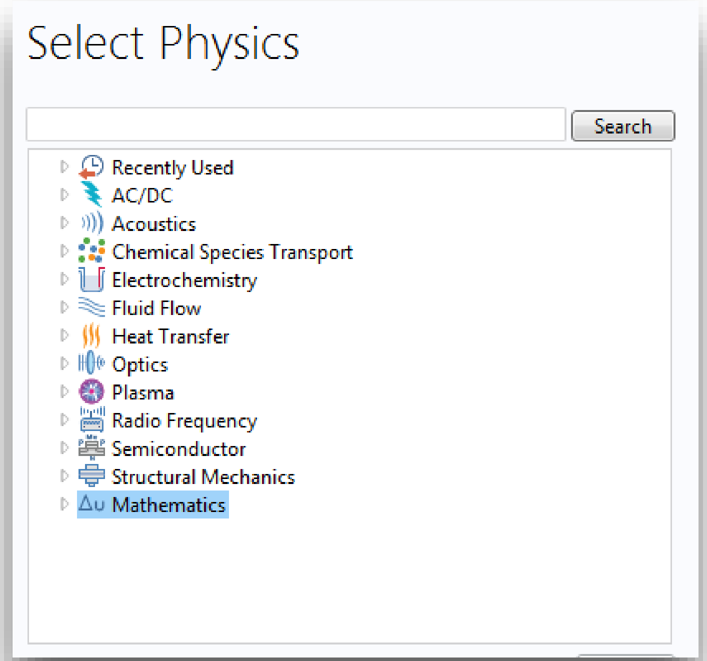
- Insert your own equations with equation forms (and make it automatic with Physics Builder)



$$e_a \frac{\partial^2 u}{\partial t^2} + d_a \frac{\partial u}{\partial t} - \nabla \cdot (c \nabla u + \alpha u - \gamma) + \beta \cdot \nabla u + a u = f$$

$$\frac{du}{dt} = F(u,t)$$

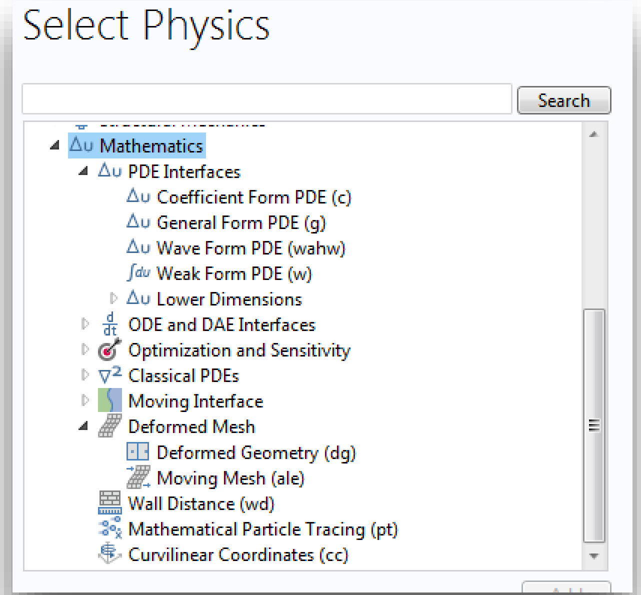
The Mathematics Interface



Select Physics

Search

- Recently Used
- AC/DC
- Acoustics
- Chemical Species Transport
- Electrochemistry
- Fluid Flow
- Heat Transfer
- Optics
- Plasma
- Radio Frequency
- Semiconductor
- Structural Mechanics
- Mathematics



Select Physics

Search

- Mathematics
  - PDE Interfaces
    - Coefficient Form PDE (c)
    - General Form PDE (g)
    - Wave Form PDE (wahw)
    - Weak Form PDE (w)
    - Lower Dimensions
  - ODE and DAE Interfaces
  - Optimization and Sensitivity
  - Classical PDEs
  - Moving Interface
  - Deformed Mesh
    - Deformed Geometry (dg)
    - Moving Mesh (ale)
  - Wall Distance (wd)
  - Mathematical Particle Tracing (pt)
  - Curvilinear Coordinates (cc)

Equation-Based Modeling

- What?
  - ODE & DAE interfaces, PDE interfaces, boundary conditions
- Why?
  - Lumped parameter systems, continuous systems
- How?
  - Demo: population dynamics, thermal curing
- Could I get myself in trouble?
  - Verification & validation
- Hands on : Coupling PDE with global and distributed ODE

What PDE, ODE, and DAE stand for?

- Partial differential equations (PDE)
  - Three main interfaces (form)
  - Need boundary conditions
- Ordinary differential equations (ODEs)
  - Global (space indipendent)
  - Distributed (space dependent)
  - Distributed, but not continuous
  - (Always) time dependent
- Algebraic equations
  - As for ODE, global or distributed
  - Do not contain time (and often not even the space)

$$d_a \frac{\partial u}{\partial t} - \nabla \cdot (c \nabla u) = f$$

Mechanics, CFD, EM…. .

$$\frac{du}{dt} = F(u,t)$$

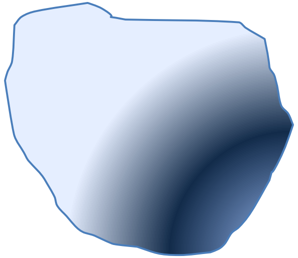
Electrical circuits, cinematism..

$$u^2 + 1 = 0$$

Part I: Lumped Parameter Systems

- Global ODEs
- Global Algebraic Equation

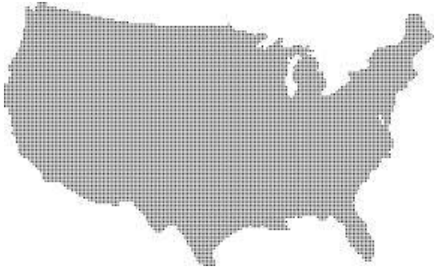
Lumped Parameter Systems



Contaminant concentration
$c(x, y, z, t)$

Contaminant amount
$C(t)$

Population dynamics



Population dynamics

Lumped Parameter Systems

- Balance laws

Rate of change of some quantity = amount entering - leaving + production - consumption

Example 1 : Dynamics

$$m\frac{d^2u}{dt^2} = -ku - c\frac{du}{dt}$$

Example 2 : Prey-predator system

$$\frac{du}{dt} = a_1 u\left(1 - {}^{u}/_{k}\right) - b_1 uv$$

$$\frac{dv}{dt} = -a_2 v + b_2 uv$$

$$m\frac{d^2u}{dt^2} + ku + c\frac{du}{dt} = 0 \quad \frac{du}{dt} - a_1 u\left(1 - {}^{u}/_{k}\right) + b_1 uv = 0 \quad \frac{dv}{dt} + a_2 v - b_2 uv = 0$$

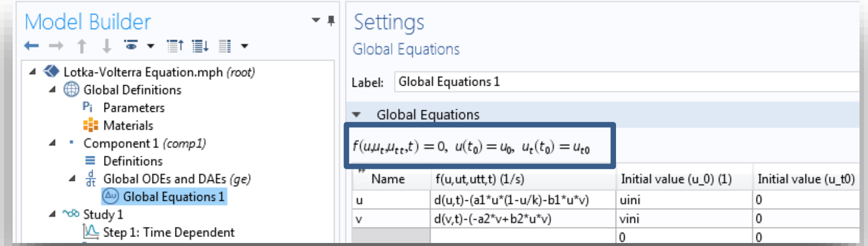
Global ODEs

- Example: Lotka-Volterra equations

$$\frac{du}{dt} = a_1 u(1 - u/k) - b_1 uv$$

$$\frac{dv}{dt} = -a_2 v + b_2 uv$$

$$u(0) = u_{ini}, v(0) = v_{ini}$$



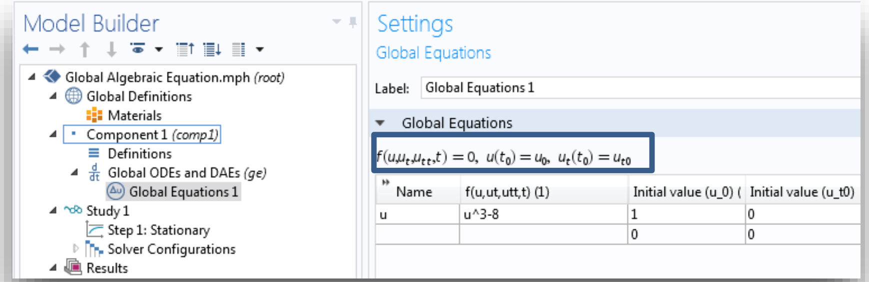
Label: Global Equations 1

$f(u,u_t,u_{tt},t) = 0,\ u(t_0) = u_0,\ u_t(t_0) = u_{t0}$

Name	f(u,ut,utt,t) (1/s)	Initial value (u_0) (1)	Initial value (u_t0)
u	d(u,t)-(a1*u*(1-u/k)-b1*u*v)	uini	0
v	d(v,t)-(-a2*v+b2*u*v)	vini	0
		0	0

Global Algebraic Equations

- Same template as Global ODEs
- What are the initial values for?
    - Stationary solvers



Label: Global Equations 1

Global Equations

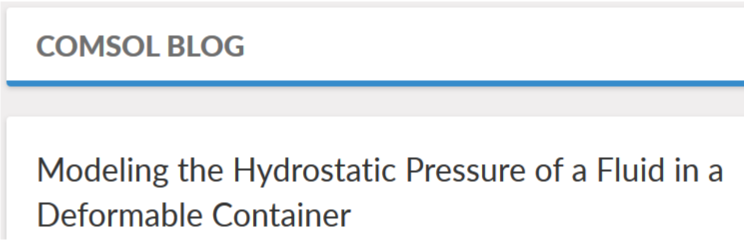
$f(u,u_t,u_{tt},t)=0,\ u(t_0)=u_0,\ u_t(t_0)=u_{t0}$

Name	f(u,ut,utt,t) (1)	Initial value (u_0) (	Initial value (u_t0)
u	u^3-8	1	0
		0	0

https://www.comsol.com/blogs/modeling-hydrostatic-pressure-fluid-deformable-container/

Physics + Global Algebraic Equations

- You can add extra degrees of freedom to a physics interface



COMSOL BLOG

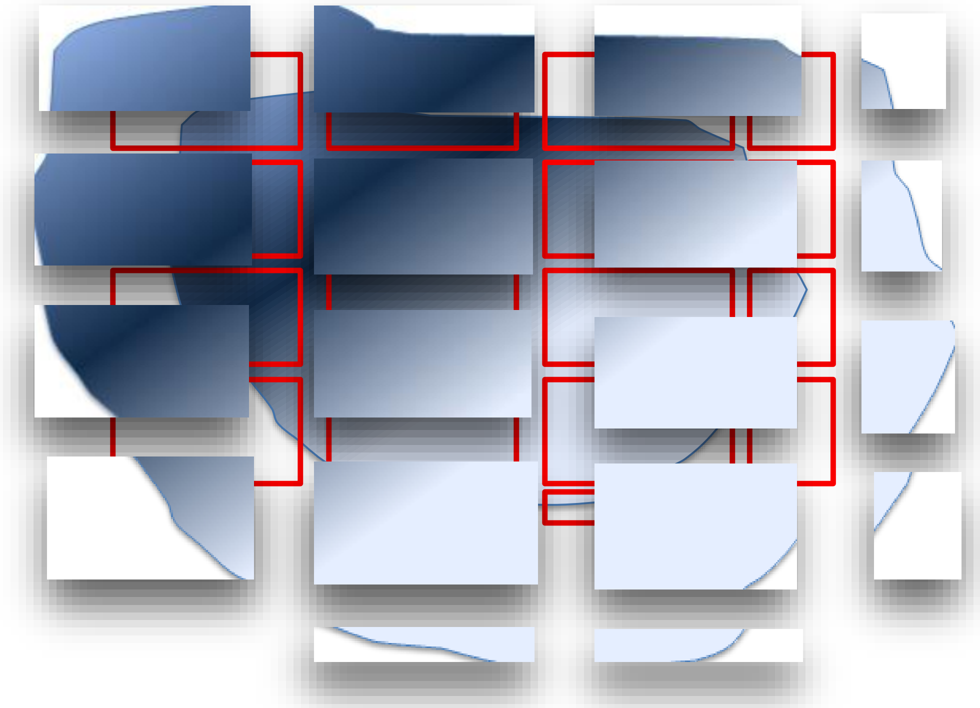
Modeling the Hydrostatic Pressure of a Fluid in a Deformable Container

https://www.comsol.com/blogs/modeling-hydrostatic-pressure-fluid-deformable-container/

Part II: Continuous Systems Without Spatial Interaction

- Domain ODEs and DAEs
- Boundary ODEs and DAEs
- Edge ODEs and DAEs
- Point ODEs and DAEs

Domain ODEs



Domain ODEs

- Problem parameters are spatially variable
  - Initial conditions $u(x, 0) = u_{ini}(x)$,…
  - Carrying capacity is spatially variable $k = k(x)$
  - Prey-predator interactions have different outcomes based on location
    - $b_1 = b_1(x), b_2 = b_2(x)$
  - Climate effect? $a_1 = a_1\ (x, T)$
- Every point evolves independent of neighbors
  - No spatial derivatives in the equation!
  - No boundary conditions needed

Domain ODEs in Physics

- Material evolution

$$\frac{\partial\alpha}{\partial t} = Ae^{\left(-{E_a}/{RT}\right)}(1-\alpha)^n$$

$$\frac{\partial\alpha}{\partial t} = Ae^{\left(-{E_a}/{RT(x)}\right)}(1-\alpha)^n$$

- Reaction kinetics
  - Built-in if you use the Chemical Reaction Engineering Module

ODE or 1D PDE?

$$\frac{d^2u}{dt^2} - au + g(t) = 0$$

$$\frac{d^2u}{dx^2} - au + g(x) = 0$$

IVP = ODE

BVP = PDE

Spatial derivative ⇒ use PDE interfaces

Domain Algebraic Equations

- Solve $u^3 = p(x, t, ..)$ for u
- Interface is the same as Domain ODE

https://www.comsol.com/blogs/solving-algebraic-field-equations/

Example: Ideal/Non-ideal gas law

- Assume ***u****=(u,v,w)* and *p* given by Navier-Stokes
- Want to solve Convection-Conduction in gas:

$$-\nabla\cdot(k\nabla T)+\rho C\,\mathrm{u}\cdot\nabla T=0$$

- Ideal gas: ρ given by

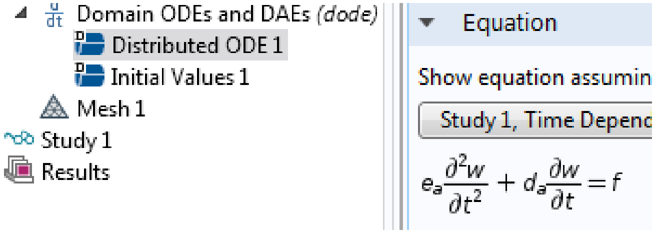
$$\rho=\frac{pM}{RT}$$

- Easy - analytical

- non-ideal gas law (needed for high molecular weight at very high pressures): ρ solution of $A(p+B\rho^2)(1-C\rho)-D\rho=0$
- Difficult – implicit equation
- How to proceed?

Example: Non-ideal gas law

- How to solve: $A(p+B\rho^2)(1-C\rho)-D\rho=0$
- Third order equation in
- Pressure *p* is function of space
- So: this is an algebraic equation at each point in space. Solution



Domain ODEs and DAEs *(dode)*
Distributed ODE 1
Initial Values 1
Mesh 1
Study 1
Results

Equation
Show equation assumin
Study 1, Time Depend

$e_a\frac{\partial^2 w}{\partial t^2}+d_a\frac{\partial w}{\partial t}=f$

$$e_a=d_a=0,\ f=A(p+B\rho^2)(1-C\rho)-D\rho$$

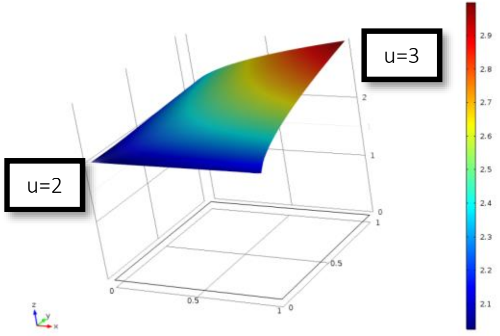
http://www.comsol.com/blogs/solving-algebraic-field-equations/

Distributed Algebraic Equation

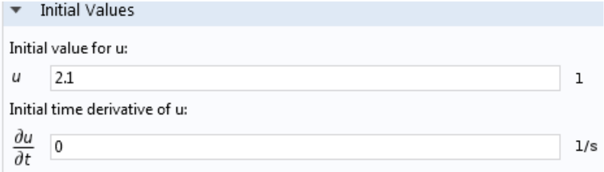
- What about non-linear equations with multiple solutions?
- Which solution do you get?
- For simplicity, consider the equation (u-2)^2-p=0, where p is a constant
- The solution you get will depend on the Initial Guess given by the PDE Physics Interface

- If we let p=x*y and let our modeling region be the unit square, then at (x,y)=(0,0) we should get the unique solution u=2 but at (x,y)=(1,1) we get 1 or 3 depending on our starting guess. See next slide.

Distributed Algebraic Equation



f (u-2)^2-p

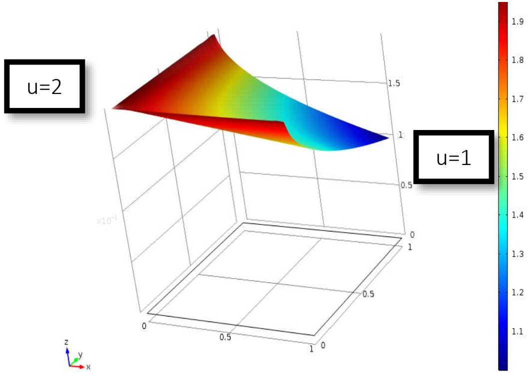


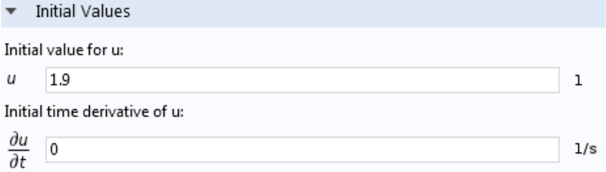
Initial value for u:

u 2.1 1

Initial time derivative of u:

$\frac{\partial u}{\partial t}$ 0 1/s





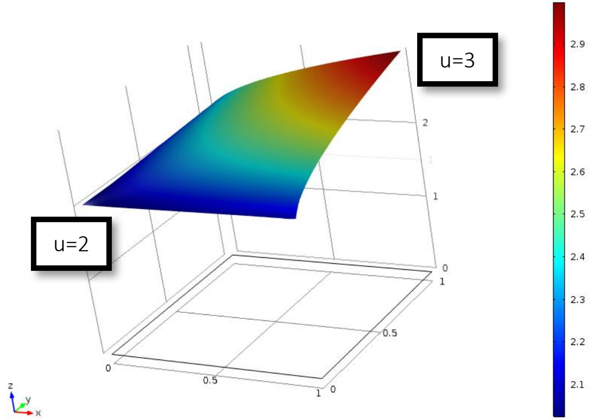
Initial value for u:

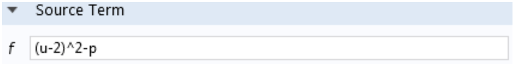
u 1.9 1

Initial time derivative of u:

$\frac{\partial u}{\partial t}$ 0 1/s

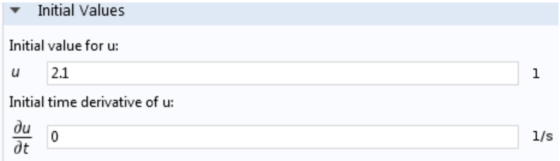
Distributed Algebraic Equation





Source Term

f (u-2)^2-p



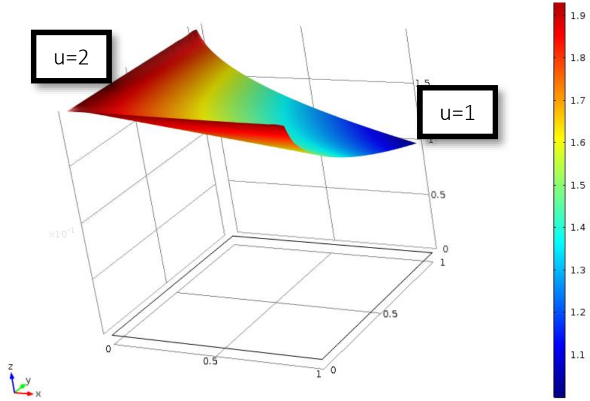
Initial Values

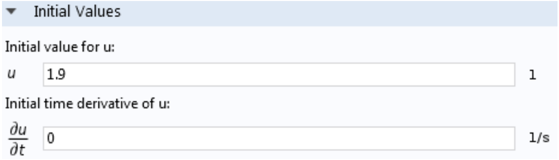
Initial value for u:

u 2.1 1

Initial time derivative of u:

$\frac{\partial u}{\partial t}$ 0 1/s





Initial Values

Initial value for u:

u 1.9 1

Initial time derivative of u:

$\frac{\partial u}{\partial t}$ 0 1/s

Part III: Continuous Systems with Spatial Interaction

Partial Differential Equations

Prey-Predator System with Migration

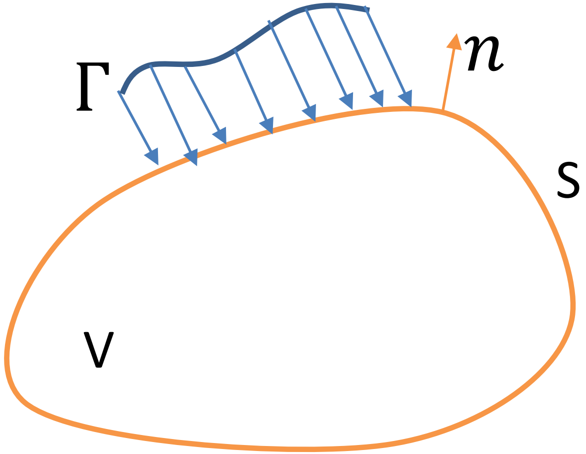
- PDEs!
- What if the second species does not migrate?
  - Domain ODE or PDE with $c_2 = 0$?
- Can we have a convective term?
- What happens for negative c_1 or c_2?

$$\frac{\partial u}{\partial t} = \nabla \cdot (c_1 \nabla u) + a_1 u(1 - u/k) - b_1 uv$$

$$\frac{\partial v}{\partial t} = \nabla \cdot (c_2 \nabla v) - a_2 v + b_2 uv$$

Balance Laws: Integral Formulation

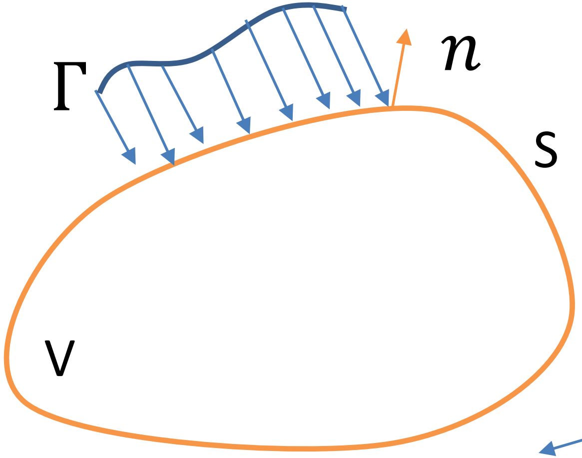
- Balance laws for a continuous system



Rate of change of some quantity = amount entering or leaving through the boundary + production or consumption inside

$$\frac{d}{dt}\int_V \phi dV = -\int_S \vec{\Gamma}\cdot\vec{n}dS + \int_V f dV$$

Balance Laws: Differential Equations



$$\frac{d}{dt}\int_V \phi dV = -\int_S \vec{\Gamma}\cdot\vec{n}dS + \int_V f dV$$

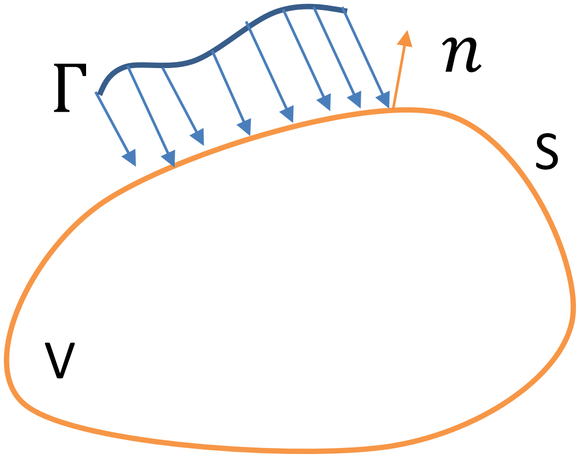
$$\int_V \frac{\partial\phi}{\partial t}dV = -\int_S \vec{\Gamma}\cdot\vec{n}dS + \int_V f dV$$

Divergence theorem

$$\int_V \frac{\partial\phi}{\partial t}dV = -\int_V div(\vec{\Gamma})\, dV + \int_V f dV$$

$$\int_V [\frac{\partial\phi}{\partial t} + div\left(\vec{\Gamma}\right) - f]dV = 0$$

Balance Laws: Differential Equations



$$\int_V [\frac{\partial \phi}{\partial t} + div\left(\vec{\Gamma}\right) - f]dV = 0$$

Localization argument $$\frac{\partial \phi}{\partial t} + div\left(\vec{\Gamma}\right) = f$$

General Form PDE

$$\frac{\partial \phi}{\partial t} + div\left(\vec{\Gamma}\right) = f$$

Usually $\phi = e\frac{\partial u}{\partial t} + du$

Template

$$\boxed{e\frac{\partial^2 u}{\partial t^2} + d\frac{\partial u}{\partial t} + div\left(\vec{\Gamma}\right) = f}$$

HT $\phi = \rho c_p T$

$$\rho c_p \frac{\partial T}{\partial t} + div\left(\vec{\Gamma}\right) = f$$

Constitutive Assumptions

$$e\frac{\partial^2 u}{\partial t^2} + d\frac{\partial u}{\partial t} + div\left(\vec{\Gamma}\right) = f$$

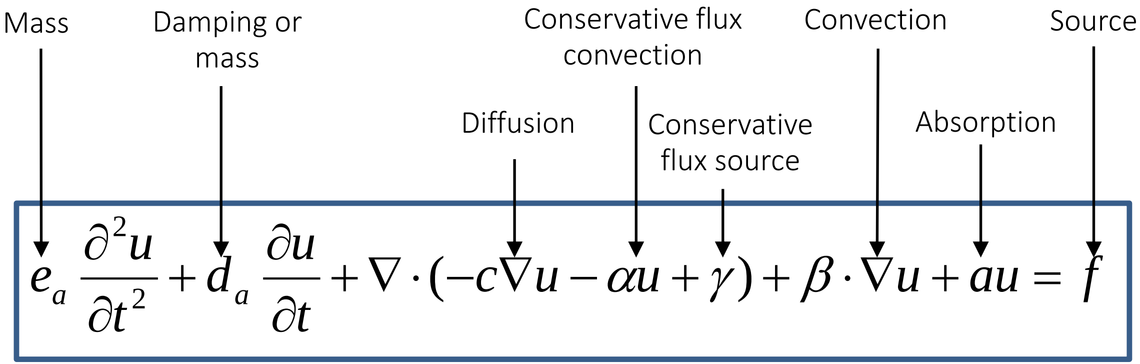
- Take the usual form for the flux

$$\vec{\Gamma} = -c\nabla u - \alpha u + \gamma$$

You specify c, α, γ

Coefficient Form PDE Template

- Specify units for independent variable and source



Mass

Damping or mass

Diffusion

Conservative flux convection

Conservative flux source

Convection

Absorption

Source

$$e_a \frac{\partial^2 u}{\partial t^2} + d_a \frac{\partial u}{\partial t} + \nabla \cdot (-c\nabla u - \alpha u + \gamma) + \beta \cdot \nabla u + au = f$$

Coefficient Form PDE

	Acoustics	Chemistry	Black-Scholes	Fischer's Ecologic Model
$\boldsymbol{u}$	Pressure	Concentration	Cost of option	Population
$\mathbf{e_a}$	$1/\rho c^2$			
$\boldsymbol{d}_{\mathbf{a}}$		1	1	1
$\boldsymbol{c}$	$1/\rho$	Diffusion coef.	$-\frac{1}{2}\sigma^2 x^2$	Dispersal rate
$\boldsymbol{\gamma}$	q_d/ρ			
$\boldsymbol{\beta}$		Velocity	$rx - \sigma^2 x$	
$\boldsymbol{a}$			$-r$	$r(^u/_K - 1)$
$\boldsymbol{f}$	Q_m	Reaction rate		

Helmholtz Equation, Coefficient Form PDE

$$-\nabla\cdot(c\nabla u)-k^2u=g$$

$$\cancel{e_a}\frac{\partial^2u}{\partial t^2}+\cancel{d_a}\frac{\partial u}{\partial t}-\nabla\cdot(c\nabla u+\cancel{\alpha}u-\cancel{\gamma})+\cancel{\beta}\cdot\nabla u+au=f$$

Coefficient matching

$$a=-\kappa^2$$
$$f=g$$

Option 1

$$a=0$$
$$f=g+\kappa^2u$$

Option 2

Helmholtz Equation, General Form PDE

$$-\nabla\cdot(c\nabla u)-k^2u=g$$

$$e\cancel{\frac{\partial^2 u}{\partial t^2}}+d\cancel{\frac{\partial u}{\partial t}}+div\left(\vec{\Gamma}\right)=f$$

Match terms

$$\vec{\Gamma}=-c\nabla u$$

$$f=g+k^2u$$

Black-Scholes Equation Coefficient Form PDE

$$\frac{\partial u}{\partial t} + \frac{1}{2}\sigma^2 x^2 \frac{\partial^2 u}{\partial x^2} + rx\frac{\partial u}{\partial x} - ru = 0$$

$$\cancel{e_a}\frac{\partial^2 u}{\partial t^2} + d_a\frac{\partial u}{\partial t} + \nabla\cdot(-c\nabla u - \cancel{\alpha u} + \cancel{\gamma}) + \beta\cdot\nabla u + au = \cancel{f}$$

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{1}{2}\sigma^2 x^2 \frac{\partial u}{\partial x}\right) + [rx - \sigma^2 x]\frac{\partial u}{\partial x} - ru = 0$$

Coefficient matching

$$d_a = 1, c = -\frac{1}{2}\sigma^2 x^2$$

$$\beta = rx - \sigma^2 x, a = -r$$

Alternative

$$d_a = 1, c = -\frac{1}{2}\sigma^2 x^2$$

$$\beta = 0, a = -r$$

$$f = (-rx + \sigma^2 x)\frac{\partial u}{\partial x}$$

https://www.comsol.com/model/the-black-scholes-equation-82

Weak Form PDE

- NO TEMPLATE!
- Extreme flexibility!

$$C\frac{\partial u}{\partial t} + \nabla \cdot (-\kappa \nabla u) - Q = 0$$

$$\int \left[C\frac{\partial u}{\partial t} w + \kappa \nabla w \cdot \nabla u \right] d\Omega = \int Q w d\Omega + \int w q_n dS \ \forall w$$

https://www.comsol.com/blogs/implementing-the-weak-form-in-comsol-multiphysics/

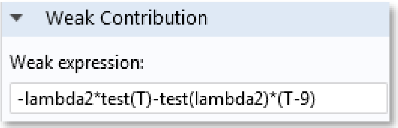
COMSOL

PDE in Weak formulation

- Weak = Integral ; based on variational formulation (conservation law)
- Most of COMSOL (and other tools') physics use such a formulation since most versatile
- It is the base of finite element (but also used within other schemes)
- Understanding how it works is the way to master what COMSOL does under the hood

$$\nabla \cdot \mathbf{\Gamma} = F$$

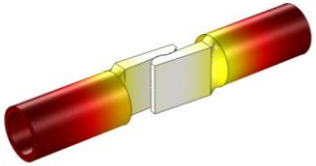
$$\int_{\Omega} \nu \, \nabla \cdot \mathbf{\Gamma} \, dV = \int_{\Omega} \nu F \, dV$$



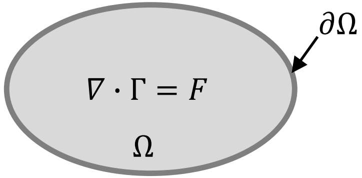
Weak Contribution

Weak expression:

-lambda2*test(T)-test(lambda2)*(T-9)



Weak Form, Stationary



- General form: $\nabla \cdot \Gamma = F$
- Multiply by test function v

  and integrate: $\int_\Omega v\nabla \cdot \Gamma dA = \int_\Omega vF dA$
- Perform integration by parts

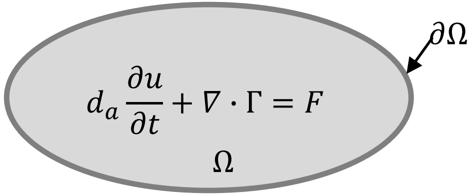
  on left-hand side: $\int_{\partial\Omega}(v\Gamma \cdot n)ds - \int_\Omega (\nabla v \cdot \Gamma)dA = \int_\Omega vF\, dA$
- Rearrange: $0 = \int_\Omega (\nabla v \cdot \Gamma + vF)\, dA + \int_{\partial\Omega}(-v\Gamma \cdot n)ds$
- Remember:
  - For Poisson's eq: $\Gamma = [u_x\ u_y]$, F = 1, R = u - 0

    (u constrained to 0 on boundaries)
  - Subdomain integral above is entered in the "weak" field: `-test(ux)*ux - test(uy)*uy + test(u)*F`
  - On the boundary, set constraint expression: u

Weak Form, Time Dependent

- Same development as stationary but start from:

$$d_a \frac{\partial u}{\partial t} + \nabla \cdot \Gamma = F$$



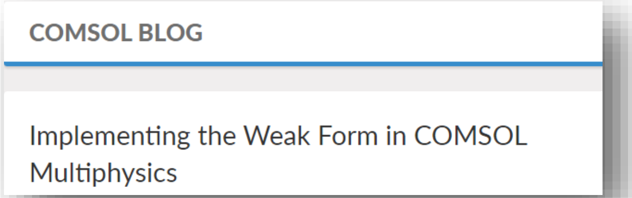
- And arrive at

$$0 = \int_\Omega \left(\nabla v \cdot \Gamma + vF - d_a v \frac{\partial u}{\partial t} \right) dA + \int_{\partial\Omega} (-v\Gamma \cdot n) ds$$

- Subdomain integral in the weak field:

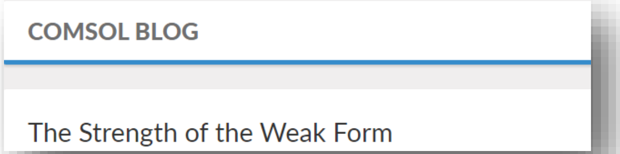
```
-test(ux)*ux - test(uy)*uy + test(u)*F - da*test(u)* ut
```

Using the Weak Form PDE Interface


COMSOL BLOG

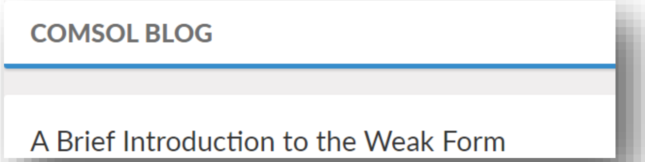
Implementing the Weak Form in COMSOL Multiphysics

https://www.comsol.com/blogs/implementing-the-weak-form-in-comsol-multiphysics/


COMSOL BLOG

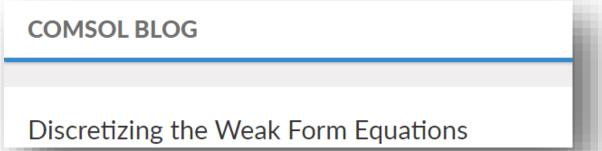
The Strength of the Weak Form

https://www.comsol.com/blogs/strength-weak-form/


COMSOL BLOG

A Brief Introduction to the Weak Form

https://www.comsol.co.in/blogs/brief-introduction-weak-form/


COMSOL BLOG

Discretizing the Weak Form Equations

https://www.comsol.co.in/blogs/discretizing-the-weak-form-equations/

Derivatives

Solution field: **`u`**

Spatial 1st derivatives: **`ux, uy, uz`**

Spatial 2nd derivatives: **`uxx, uxy, …, uyz, uzz`**

Time derivatives: **`ut, utt`**

Mixed derivatives: **`uxt, uytt`**

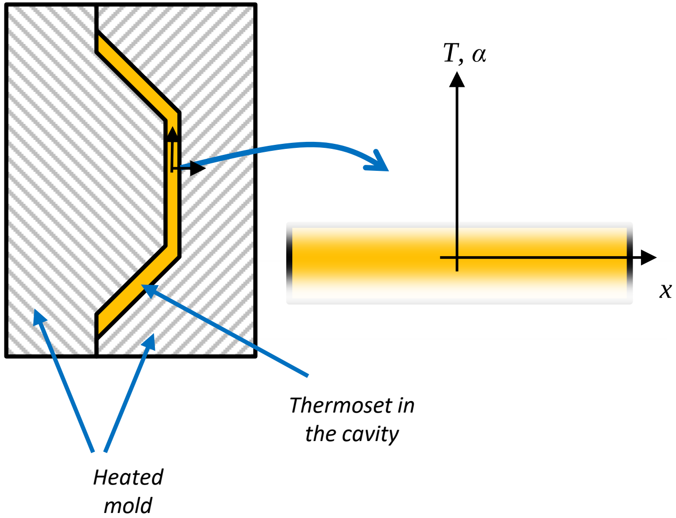
Derivatives tangent to surfaces: **`uTx, uTy, uTz`**

Derivatives of quantities other than the primary dependent variable: **`d(q,t),d(q,x)`**

More: “Differentiation Operators” in the *COMSOL Multiphysics Reference Manual*

Integrated Demo: Thermal Curing Physics

Image by Joe Haupt — Own work. Licensed under [CC BY-SA 2.0](), via [Wikimedia Commons]().



Thermal Curing: Mathematical Model

$$\rho C_p \frac{\partial T}{\partial t} + \nabla \cdot (-\kappa \nabla T) = -\rho H_r \frac{\partial \alpha}{\partial t}$$

$$\frac{\partial \alpha}{\partial t} = A e^{(-E_a/RT)} (1-\alpha)^n$$

- Initial conditions: room temperature, zero curing
- Boundary conditions: heat flux of $10\ \mathrm{kW/m^2}$
- Step 1: Pick appropriate Mathematics interfaces
  - PDE + Domain ODE
- Step 2: Fit the equations into the templates

Thermal Curing: Choosing the Interface

$$\rho C_p \frac{\partial T}{\partial t} + \nabla \cdot (-\kappa \nabla T) = -\rho H_r \frac{\partial \alpha}{\partial t}$$

$$e_a \frac{\partial^2 u}{\partial t^2} + d_a \frac{\partial u}{\partial t} - \nabla \cdot (c \nabla u + \alpha u - \gamma) + \beta \cdot \nabla u + a u = f$$

$$d_a = \rho C_p, c = \kappa, f = -\rho H_r \frac{\partial \alpha}{\partial t}$$

$$\left.\begin{matrix} \\ \\ \end{matrix}\right\}$$ PDE

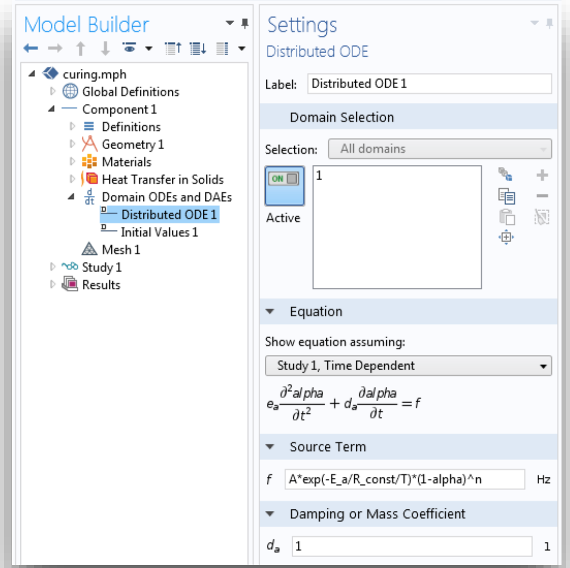
$$\frac{\partial \alpha}{\partial t} = A e^{(-E_a/RT)} (1 - \alpha)^n$$

$$e_a \frac{\partial^2 \alpha}{\partial t^2} + d_a \frac{\partial \alpha}{\partial t} = f$$

$$d_a = 1, f = A e^{(-E_a/RT)} (1 - \alpha)^n$$

$$\left.\begin{matrix} \\ \\ \end{matrix}\right\}$$ Domain ODE

Thermal Curing: Heat Transfer Interface



Label: Distributed ODE 1

Selection: All domains

1

Show equation assuming: Study 1, Time Dependent

$$e_a\frac{\partial^2 alpha}{\partial t^2}+d_a\frac{\partial alpha}{\partial t}=f$$

f A*exp(-E_a/R_const/T)*(1-alpha)^n Hz

d_a 1 1

https://www.comsol.com/blogs/modeling-the-thermal-curing-process/

Part IV: Boundaries and Interfaces

The World Versus Your World!

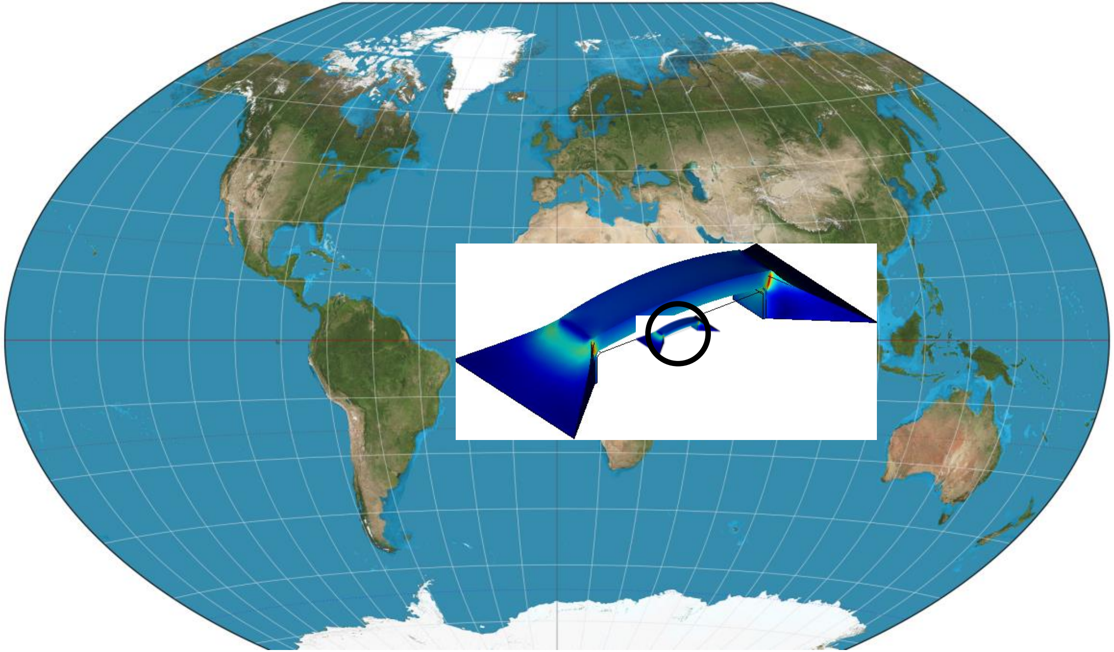
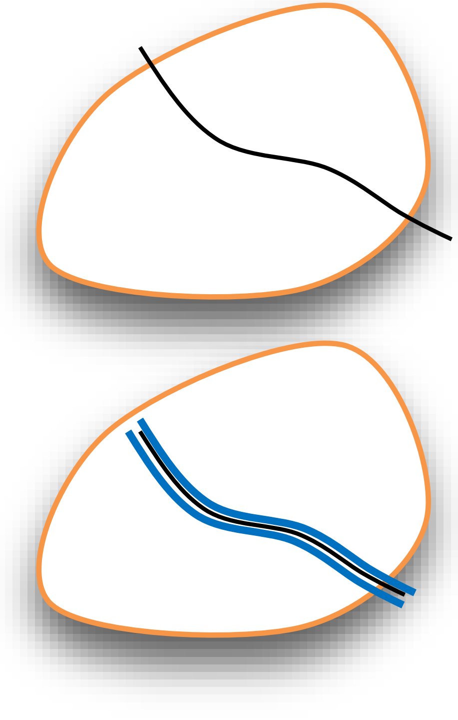


Image by Strebe — Own work. Licensed under CC BY-SA 3.0, via Wikimedia Commons.

Jump and Boundary Conditions



$$\int_V \frac{\partial\phi}{\partial t}dV = -\int_S \vec{\Gamma}\cdot\vec{n}\,dS + \int_V f\,dV$$

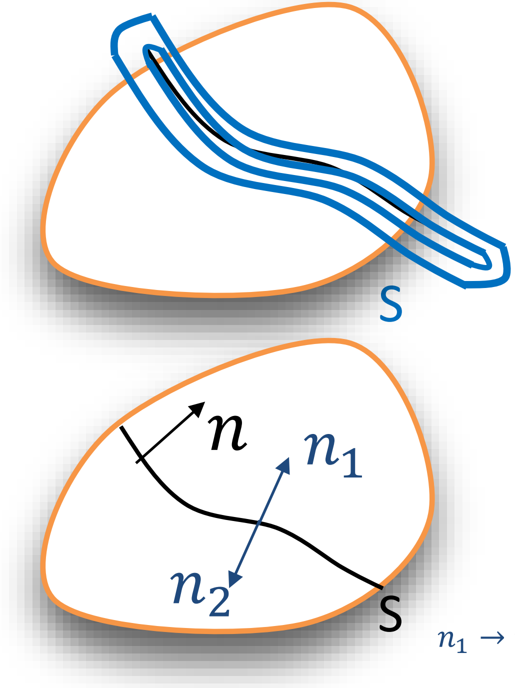
$$\Big\Downarrow$$

Can’t do this!

$$\int_V \frac{\partial\phi}{\partial t}dV = -\int_V div(\vec{\Gamma})\,dV + \int_V f\,dV$$

1. Stay with the integral equation
2. Focus

Jump and Boundary Conditions



$n_1 \to n, n_2 \to -n$

$$\cancel{\int_V \frac{\partial \phi}{\partial t} dV} = -\int_S \vec{\Gamma}\cdot\vec{n}\,dS + \cancel{\int_V f\,dV}$$

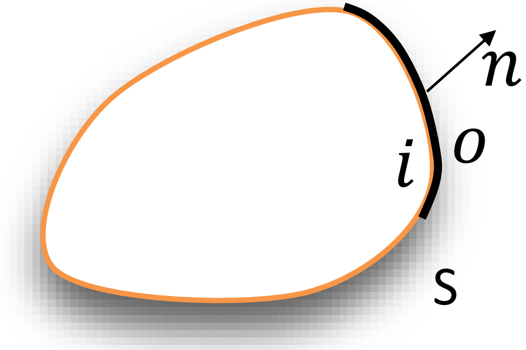
$$V \to 0, S \neq 0$$

$$\int_S \vec{\Gamma}\cdot\vec{n}\,dS = 0$$

$$\int_S (\vec{\Gamma}_1 - \vec{\Gamma}_2)\cdot\vec{n}\,dS = 0$$

$$[\![\vec{\Gamma}]\!]\cdot\vec{n}=0$$

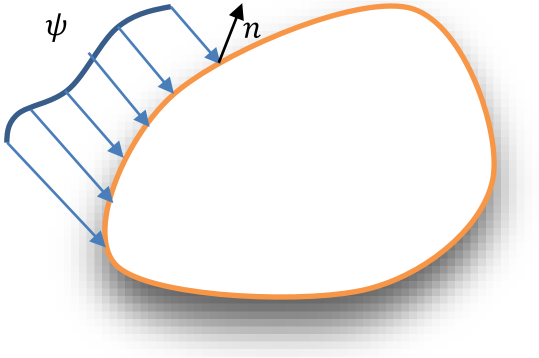
Jump and Boundary Conditions



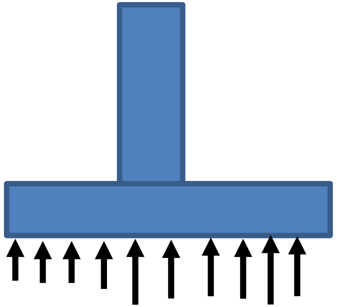
$$\underbrace{-\vec{\Gamma}_i \cdot \vec{n}}_{\text{Inward flux}} = -\vec{\Gamma}_o \cdot \vec{n}$$

- Think about what is outside
- We have NOT considered surface production here

Boundary Conditions 1: Flux



$$\underbrace{-\vec{\Gamma}_i \cdot \vec{n}}_{\text{Inward flux}} = \psi$$



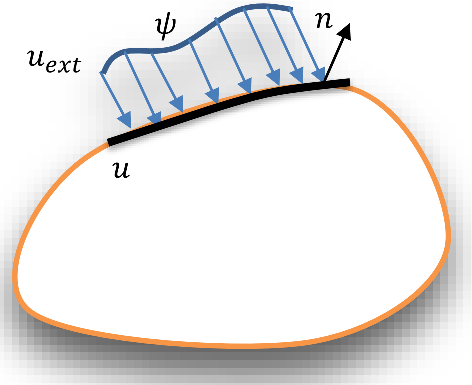
- Example: Heat Transfer in Solids

$$-(-\kappa \nabla T) \cdot \vec{n} = q_0$$

- Natural (Neumann) boundary conditions

Boundary Conditions: Mixed

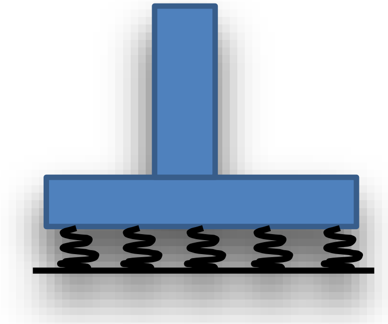
- A constitutive assumption about the outside



$$-\vec{\Gamma}_i \cdot \vec{n} = \psi$$

$$\psi = \boxed{\frac{\kappa_o}{L_{ext}}}(u_{ext} - u)$$

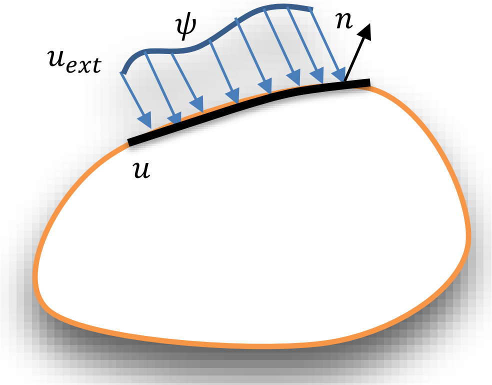
$$-(-\kappa \nabla T) \cdot \vec{n} = \boxed{h}(T_{ext} - T)$$



- Example: Heat Transfer in Solids

- Mixed (Robin) boundary conditions

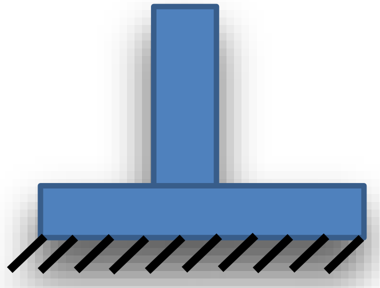
Boundary Conditions: Extremes?



$$-\vec{\Gamma}_i \cdot \vec{n} = \psi$$

$$-(-\kappa\nabla u) \cdot \vec{n} = \frac{\kappa_o}{L_{ext}}(u_{ext} - u)$$

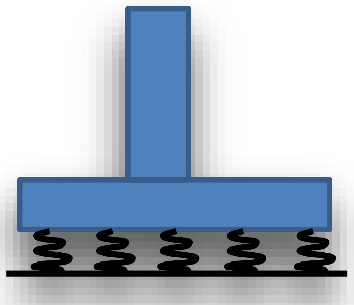
$$(\frac{\kappa}{\kappa_o} L_{ext}\nabla u) \cdot \vec{n} = (u_{ext} - u)$$

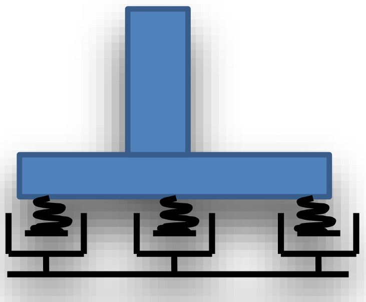


$$\kappa_o \gg \kappa \ \Rightarrow u = u_{ext}$$

- Temperature, voltage, displacement
- Dirichlet boundary conditions

More on Boundary Conditions





- Built-in spring foundation
- DIY boundary condition

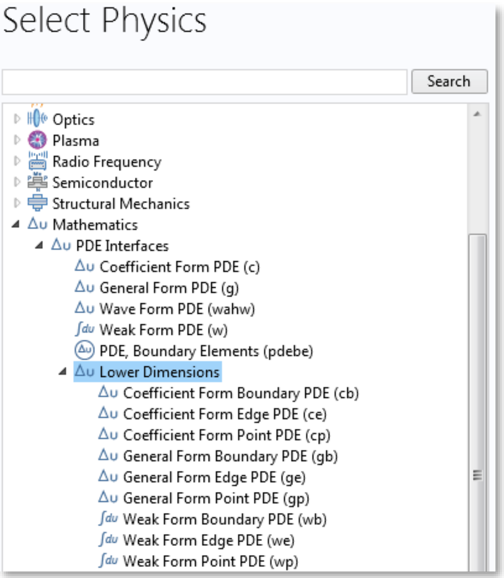
$$f = f(u, \dot{u})$$

https://www.comsol.com/blogs/how-to-make-boundary-conditions-conditional-in-your-simulation/
https://www.comsol.com/blogs/modeling-natural-and-forced-convection-in-comsol-multiphysics/

Part V: Surface Phenomena

- Boundary PDEs
- Edge PDEs

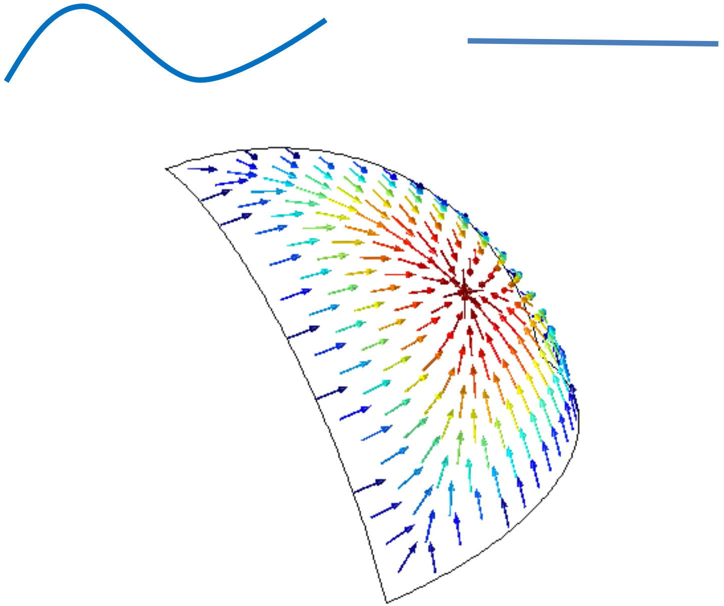
Lower Dimension PDE Interfaces



Select Physics

Search

- Optics
- Plasma
- Radio Frequency
- Semiconductor
- Structural Mechanics
- Mathematics
  - PDE Interfaces
    - Coefficient Form PDE (c)
    - General Form PDE (g)
    - Wave Form PDE (wahw)
    - Weak Form PDE (w)
    - PDE, Boundary Elements (pdebe)
    - Lower Dimensions
      - Coefficient Form Boundary PDE (cb)
      - Coefficient Form Edge PDE (ce)
      - Coefficient Form Point PDE (cp)
      - General Form Boundary PDE (gb)
      - General Form Edge PDE (ge)
      - General Form Point PDE (gp)
      - Weak Form Boundary PDE (wb)
      - Weak Form Edge PDE (we)
      - Weak Form Point PDE (wp)



Built-In Interfaces for Lower Dimensional Physics

- Fluid Flow
  - *Pipe Flow (pfl)*
  - *Water Hammer (whtd)*
  - *Thin-Film Flow, Shell (tffs)*
- Heat Transfer
  - *Heat Transfer in Pipes (htp)*
  - *Heat Transfer in Thin Shell (htsh)*
  - *Heat Transfer in Thin Films (htsh)*
  - *Heat Transfer in Fractures (htsh)*
- Structural Mechanics
  - *Shell (shell)*
  - *Membrane (mbrn)*
  - *Beam (beam)*
  - *Truss (truss)*
- AC/DC
  - *Electric Currents, Shell (ecs)*
- RF
  - *Transmission Line (tl)*
- Chemical and Reaction Engineering
  - *Surface Reactions (sr)*
- Electrochemistry
  - *Electrode, Shell (els)*

Part VI: Miscellaneous

- PDEs in axisymmetric components
- Integrodifferential equations
- Nonlocal interactions
- Verification and validation
- Stabilization

PDEs in Axisymmetric Components

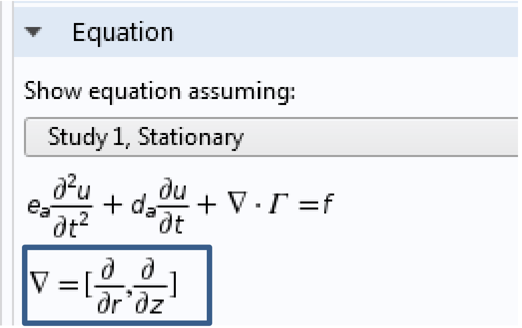
- In the PDE interfaces, differential operators do not have tensorial meanings

$$\nabla \cdot \Gamma = Q$$

Axisymmetry

$$\frac{1}{r}\frac{\partial (r\Gamma_r)}{\partial r} + \frac{\partial \Gamma_\phi}{\partial \phi} + \frac{\partial \Gamma_z}{\partial z} = Q$$

$$\frac{\partial \Gamma_r}{\partial r} + \frac{\partial \Gamma_z}{\partial z} + \frac{\Gamma_r}{r} = Q$$



Equation

Show equation assuming:

Study 1, Stationary

$$e_a\frac{\partial^2 u}{\partial t^2} + d_a\frac{\partial u}{\partial t} + \nabla \cdot \Gamma = f$$

$$\boxed{\nabla = [\frac{\partial}{\partial r},\frac{\partial}{\partial z}]}$$

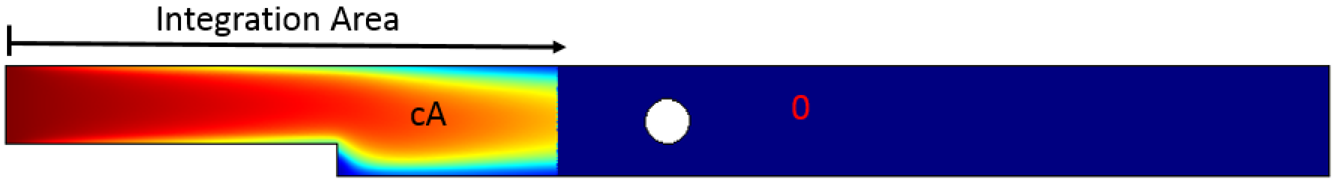
$$\frac{\partial \Gamma_r}{\partial r} + \frac{\partial \Gamma_z}{\partial z} = f$$

- The source term is your friend!

COMSOL Blog Post

- Guidelines for Equation-Based Modeling in Axisymmetric Components
  - https://www.comsol.com/blogs/guidelines-for-equation-based-modeling-in-axisymmetric-components/

Integrodifferential Equations



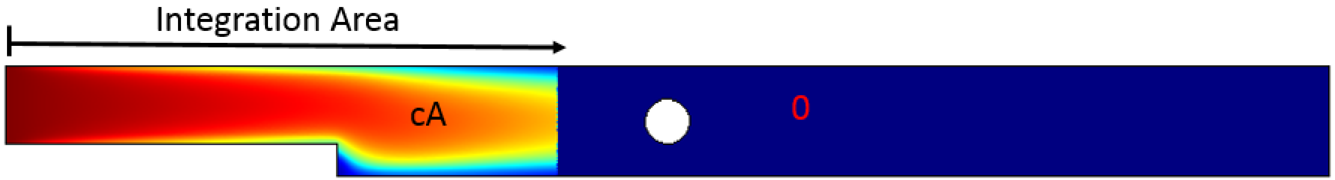
$$I = \int_0^L c_A dx$$

Integration Coupling Operator

$$I(s) = \int_0^s c_A dx$$

?

Variable Limits of Integration



$$I(s) = \int_0^s c_A dx = \int_0^L k(s,x) c_A dx$$

Integration Coupling operator!

$$I(s) = \int_0^s c_A dx = \int_0^L k(dest(x), x) c_A dx$$

Solving Integrodifferential Equations

$$\rho C_p \frac{\partial T}{\partial x} + \frac{\partial}{\partial x}\left(-\kappa \frac{\partial T}{\partial x}\right) = aT^4 - b\int_0^L K(x,s)\,T(s)^4 ds$$

Source 1: *a*T^4*

Source 2: -b*intop1(K(dest(x),x)*T^4)

https://www.comsol.com/blogs/integrals-with-moving-limits-and-solving-integro-differential-equations/

Nonlocal Interactions: Component Coupling Operators

Purpose:

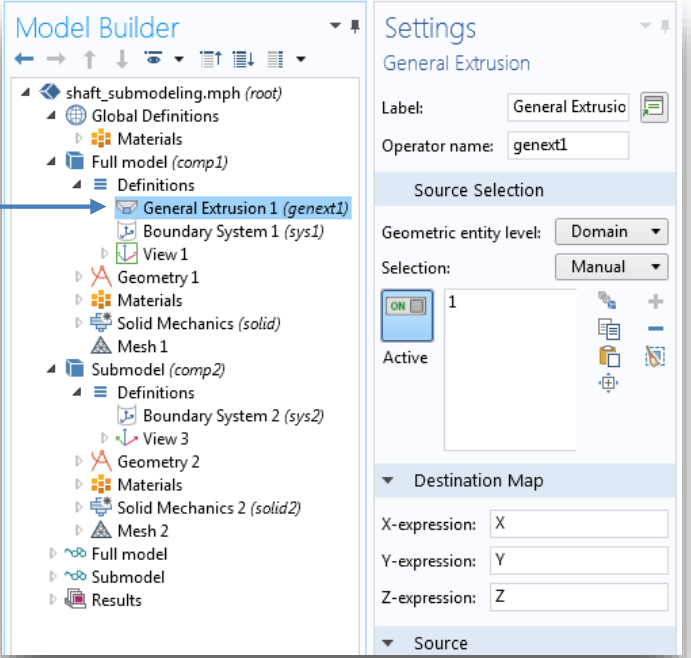
- Pass data from one part of a component to another
- Pass data between different components

$$q_d(x_d) = f(q_s(x_s))$$

$$\boxed{T: x_d} \to x_s$$

Usage:

- Define operator in the source component/entity
- Use in the destination component/entity



Model Builder

- shaft_submodeling.mph *(root)*
  - Global Definitions
    - Materials
  - Full model *(comp1)*
    - Definitions
      - General Extrusion 1 *(genext1)*
      - Boundary System 1 *(sys1)*
      - View 1
    - Geometry 1
    - Materials
    - Solid Mechanics *(solid)*
    - Mesh 1
  - Submodel *(comp2)*
    - Definitions
      - Boundary System 2 *(sys2)*
      - View 3
    - Geometry 2
    - Materials
    - Solid Mechanics 2 *(solid2)*
    - Mesh 2
  - Full model
  - Submodel
  - Results

Settings

General Extrusion

Label: General Extrusio

Operator name: genext1

Source Selection

Geometric entity level: Domain

Selection: Manual

1

Active

Destination Map

X-expression: X

Y-expression: Y

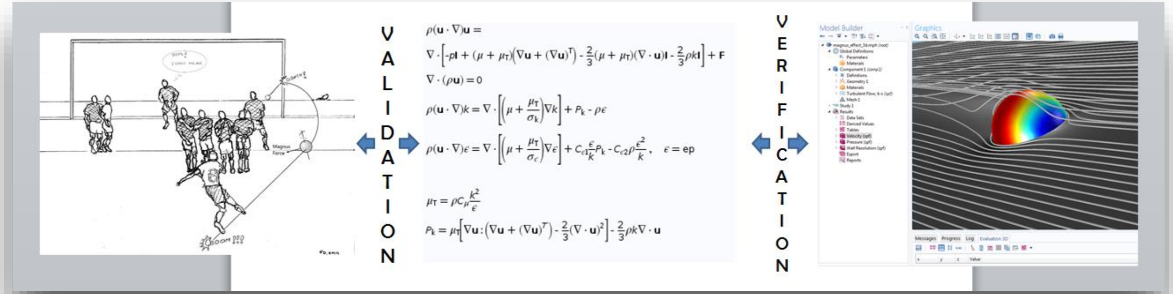
Z-expression: Z

Source

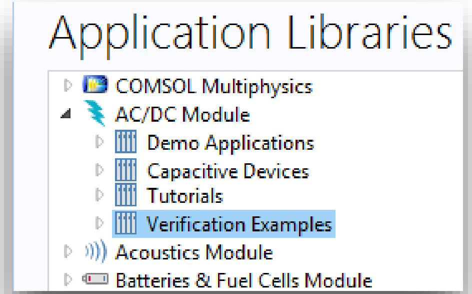
https://www.comsol.com/blogs/part-2-mapping-variables-with-general-extrusion-operators/

Verification & Validation

- Exact solutions
- Benchmarks
- Analogous modules in COMSOL Multiphysics®



$$\rho(\mathbf{u}\cdot\nabla)\mathbf{u} =$$
$$\nabla\cdot\left[-p\mathbf{I} + (\mu+\mu_T)\left(\nabla\mathbf{u}+(\nabla\mathbf{u})^T\right) - \frac{2}{3}(\mu+\mu_T)(\nabla\cdot\mathbf{u})\mathbf{I} - \frac{2}{3}\rho k\mathbf{I}\right] + \mathbf{F}$$
$$\nabla\cdot(\rho\mathbf{u}) = 0$$
$$\rho(\mathbf{u}\cdot\nabla)k = \nabla\cdot\left[\left(\mu+\frac{\mu_T}{\sigma_k}\right)\nabla k\right] + P_k - \rho\epsilon$$
$$\rho(\mathbf{u}\cdot\nabla)\epsilon = \nabla\cdot\left[\left(\mu+\frac{\mu_T}{\sigma_\epsilon}\right)\nabla\epsilon\right] + C_{\epsilon 1}\frac{\epsilon}{k}P_k - C_{\epsilon 2}\rho\frac{\epsilon^2}{k}, \quad \epsilon = \mathrm{ep}$$
$$\mu_T = \rho C_\mu \frac{k^2}{\epsilon}$$
$$P_k = \mu_T\left[\nabla\mathbf{u}:\left(\nabla\mathbf{u}+(\nabla\mathbf{u})^T\right) - \frac{2}{3}(\nabla\cdot\mathbf{u})^2\right] - \frac{2}{3}\rho k\nabla\cdot\mathbf{u}$$



Application Libraries

- COMSOL Multiphysics
- AC/DC Module
  - Demo Applications
  - Capacitive Devices
  - Tutorials
  - Verification Examples
- Acoustics Module
- Batteries & Fuel Cells Module

Verification

Method of manufactured solutions:

1. Assume a solution
2. Plug in PDE to get source term

$$\nabla\cdot(-c\nabla u-\alpha u+\gamma)+au=\boxed{f}$$

3. Find initial & boundary conditions
4. Compute with IC, BC, and source term
5. Compare assumed and computed solution

https://www.comsol.com/blogs/verify-simulations-with-the-method-of-manufactured-solutions/

Stabilization

- Convection dominated transport problems are numerically unstable
- Sophisticated techniques implemented in physics interfaces

$$u_t + b \cdot \nabla u = f$$

$$u_t + b \cdot \nabla u + \nabla \cdot (-c \nabla u) = f$$

- A simple stabilization technique for convective transport problems

https://www.comsol.com/blogs/understanding-stabilization-methods/

Hands-on #1

3D, time dependent
Use General form PDE

Computing integrals over time and space
(Adding ODE, global or distributed)

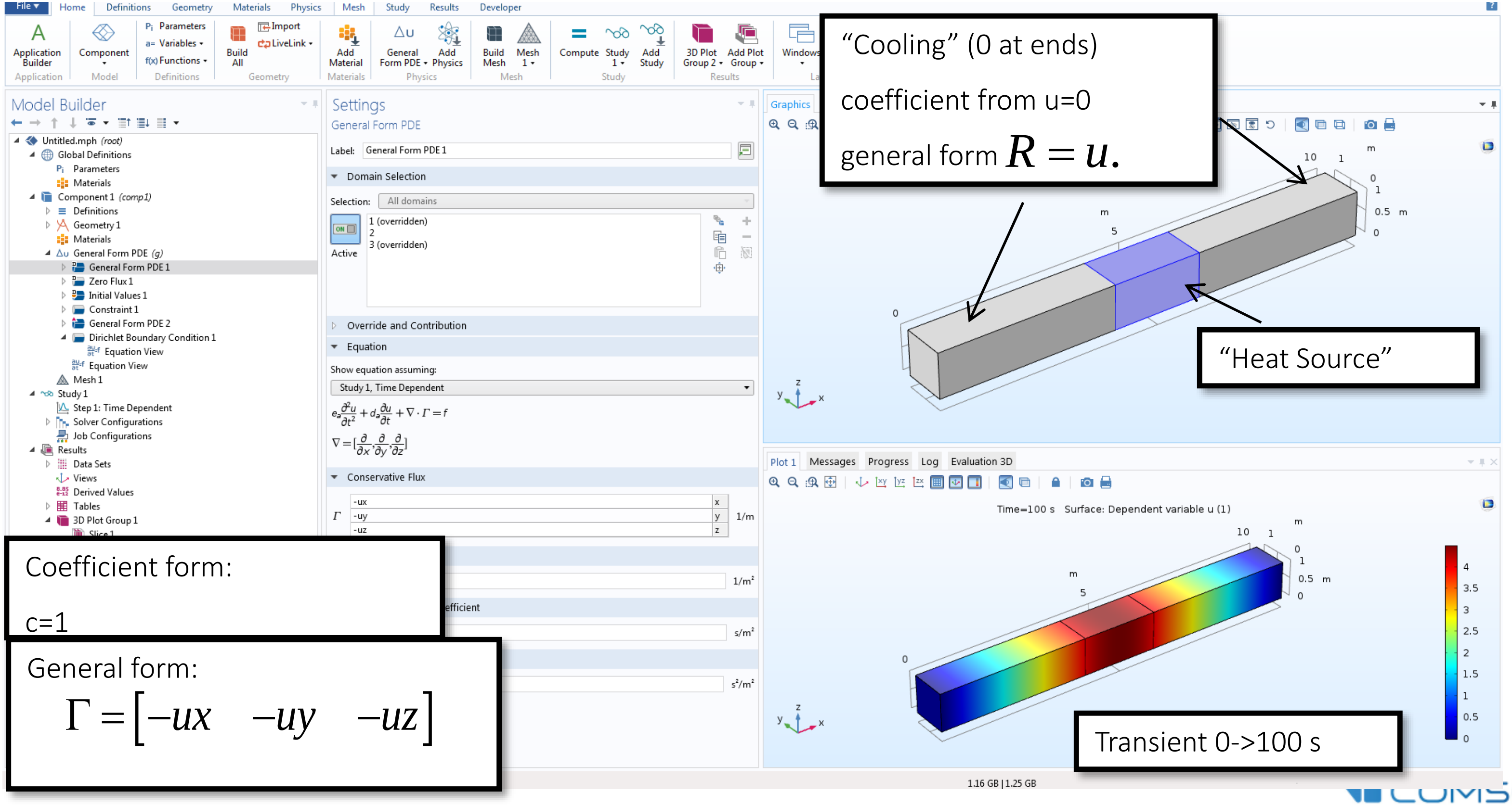
General Form – A more compact formulation

- Inside domain

$$e_a \frac{\partial^2 u}{\partial t^2} + d_a \frac{\partial u}{\partial t} + \nabla \cdot \Gamma = F$$

- On domain boundary

$$\left.\begin{array}{c} -\mathbf{n} \cdot \Gamma = G + \left(\dfrac{\partial R}{\partial u} \right)^T \mu \\ 0 = R \end{array}\right\}$$



General Form PDE

Label: General Form PDE 1

Domain Selection

Selection: All domains

1 (overridden)
2
3 (overridden)

Active

Override and Contribution

Equation

Show equation assuming: Study 1, Time Dependent

$$e_a\frac{\partial^2 u}{\partial t^2}+d_a\frac{\partial u}{\partial t}+\nabla\cdot\Gamma=f$$

$$\nabla=[\frac{\partial}{\partial x},\frac{\partial}{\partial y},\frac{\partial}{\partial z}]$$

Conservative Flux

Γ: -ux (x), -uy (y), -uz (z) 1/m

“Cooling” (0 at ends)

coefficient from u=0

general form $R = u.$

“Heat Source”

Coefficient form:

c=1

General form:

$$\Gamma = \begin{bmatrix}-ux & -uy & -uz\end{bmatrix}$$

Transient 0->100 s

Transient Diffusion Equation + ODE

$$d_a \frac{\partial u}{\partial t} - \nabla \cdot (c\nabla u + \alpha u - \gamma) + \beta \cdot \nabla u + au = f$$

What if we wish to measure the global accumulation of “heat” over time?

$$U = \iiint_V u \, dV \quad \text{volume integral of solution}$$

$$w = \int_t U dt \qquad \text{time integral of volume integral}$$

Adding a ODE
Transient Diffusion Equation + ODE

$$d_a \frac{\partial u}{\partial t} - \nabla \cdot (c \nabla u + \alpha u - \gamma) + \beta \cdot \nabla u + au = f$$

What if we wish to measure the global accumulation of “heat” over time?

$$U = \iiint_V u \, dV$$

$$w = \int_{t=[t_0, t_1]} U dt \Leftrightarrow \frac{dw}{dt} = U$$

$$wt - U = 0$$

=> This is a **Global ODE** in the global **state variable** *w*

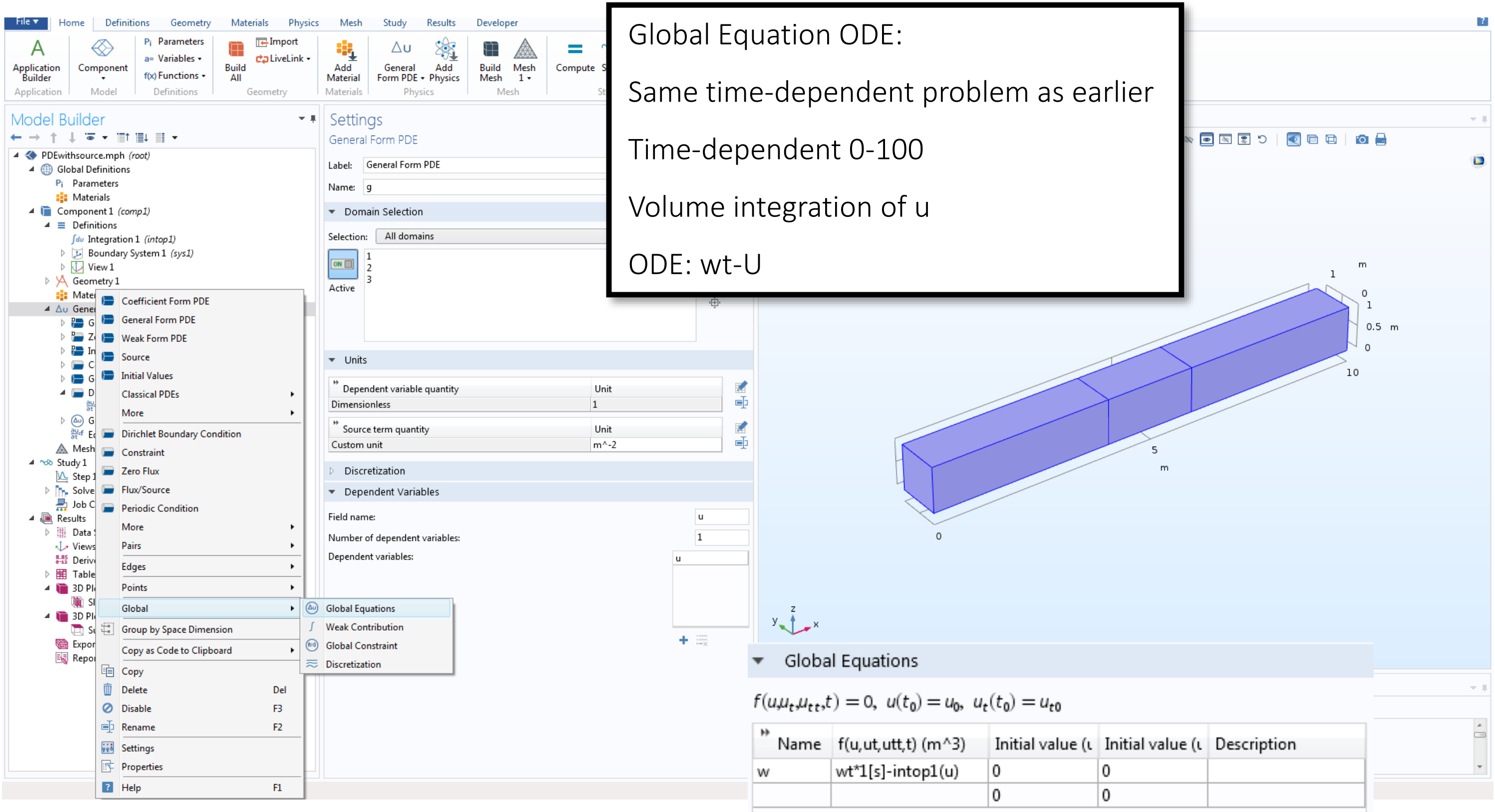
Global Equation ODE:

Same time-dependent problem as earlier

Time-dependent 0-100

Volume integration of u

ODE: wt-U



Settings

General Form PDE

Label: General Form PDE

Name: g

Domain Selection

Selection: All domains

1
2
3

Active

Units

Dependent variable quantity	Unit
Dimensionless	1

Source term quantity	Unit
Custom unit	m^-2

Discretization

Dependent Variables

Field name: u

Number of dependent variables: 1

Dependent variables: u

Global Equations

$$f(u,u_t,u_{tt},t)=0,\ \ u(t_0)=u_0,\ \ u_t(t_0)=u_{t0}$$

Name	f(u,ut,utt,t) (m^3)	Initial value (u	Initial value (u	Description
w	wt*1[s]-intop1(u)	0	0	
		0	0	

PDEs + Distributed ODEs

(what if the ODE is depending on space)
(comment on continuity of the solution)

Transient Diffusion Equation + Distributed ODE

What if we get “*damage*” from local accumulation of “heat”.

Example of real application: bioheating

$$P(x,y,z)=\int_t u(x,y,z)dt$$

integral of solution

local time

We want to visualize the *P*-field to assess local damage.

Let’s assume damage happens where P>20.

$$\boxed{\Leftrightarrow \frac{dP}{dt}=u \quad \text{at each point in space}}$$

Transient Diffusion Equation + Distributed ODE

$$\frac{dP}{dt} = u \quad \text{local time integral of solution}$$

But this can be seen as a PDE with no spatial derivatives =

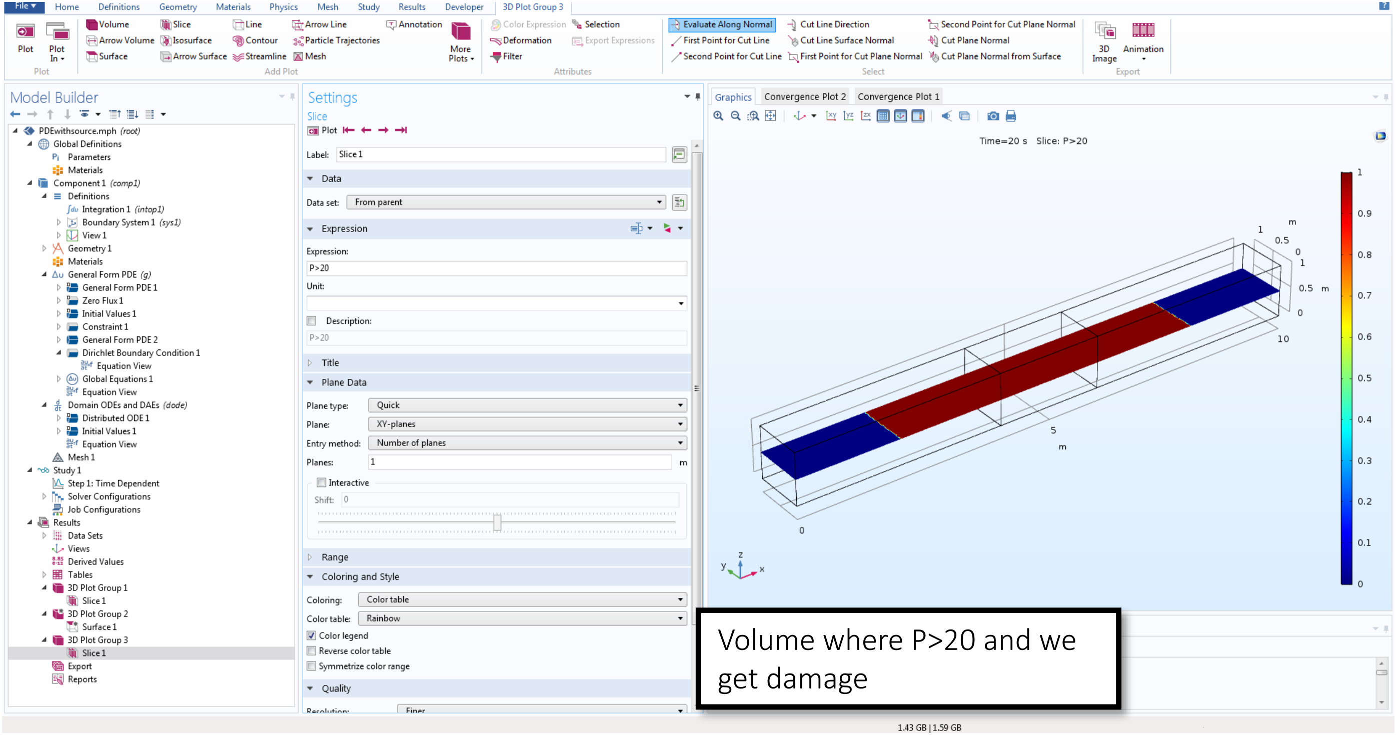
= *Distributed ODE*

Use coefficient form with unknown field P, c = 0, f = u, da=1

Let all other coefficients be zero

Or use new Domain ODEs and DAEs interface

Use of logical operators



Label: Slice 1

Data set: From parent

Expression: P>20

Unit:

☐ Description: P>20

Plane type: Quick

Plane: XY-planes

Entry method: Number of planes

Planes: 1

☐ Interactive

Shift: 0

Coloring: Color table

Color table: Rainbow

☑ Color legend

☐ Reverse color table

☐ Symmetrize color range

Resolution: Finer

Time=20 s Slice: P>20

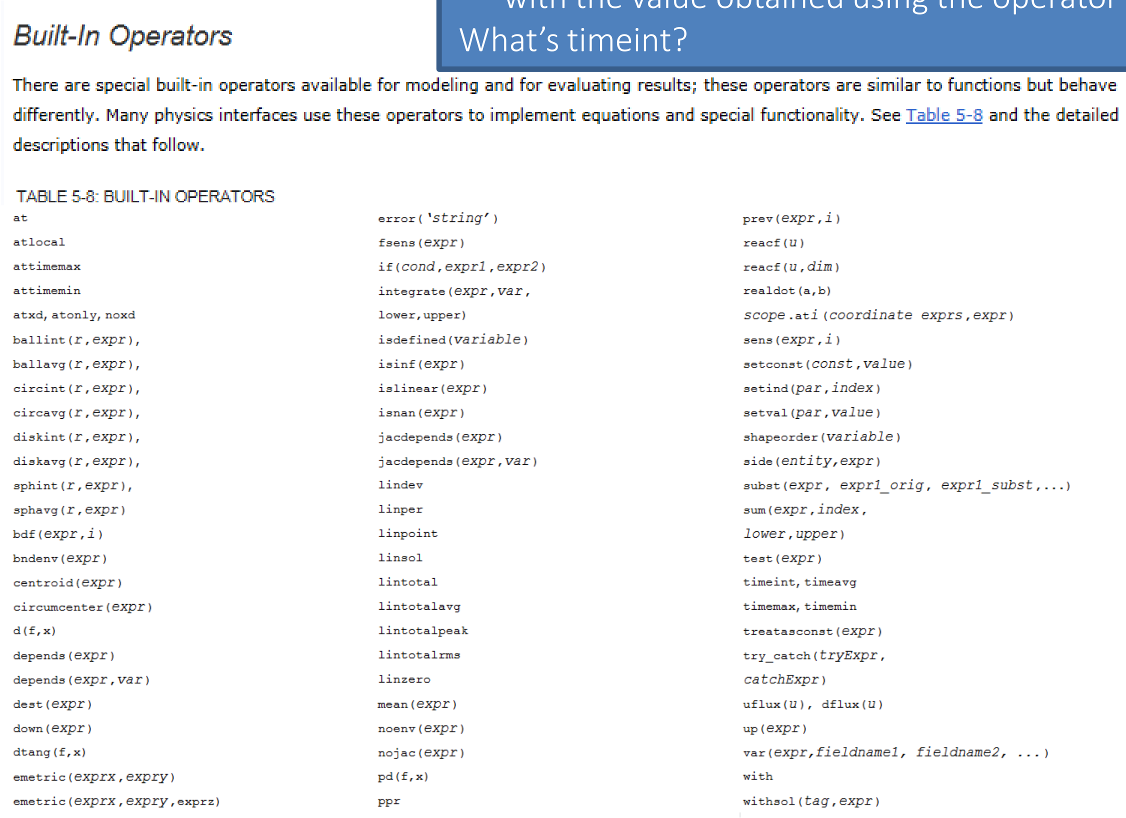
Volume where P>20 and we get damage

Questions?

Let’s compare

- derived values
- with the value obtained using the operator “timeint”

What’s timeint?



Built-In Operators

There are special built-in operators available for modeling and for evaluating results; these operators are similar to functions but behave differently. Many physics interfaces use these operators to implement equations and special functionality. See Table 5-8 and the detailed descriptions that follow.

TABLE 5-8: BUILT-IN OPERATORS

at
atlocal
attimemax
attimemin
atxd, atonly, noxd
ballint(*r*,*expr*),
ballavg(*r*,*expr*),
circint(*r*,*expr*),
circavg(*r*,*expr*),
diskint(*r*,*expr*),
diskavg(*r*,*expr*),
sphint(*r*,*expr*),
sphavg(*r*,*expr*)
bdf(*expr*,*i*)
bndenv(*expr*)
centroid(*expr*)
circumcenter(*expr*)
d(f,x)
depends(*expr*)
depends(*expr*,*var*)
dest(*expr*)
down(*expr*)
dtang(f,x)
emetric(*exprx*,*expry*)
emetric(*exprx*,*expry*,exprz)

error('*string*')
fsens(*expr*)
if(*cond*,*expr1*,*expr2*)
integrate(*expr*,*var*, lower,upper)
isdefined(*variable*)
isinf(*expr*)
islinear(*expr*)
isnan(*expr*)
jacdepends(*expr*)
jacdepends(*expr*,*var*)
lindev
linper
linpoint
linsol
lintotal
lintotalavg
lintotalpeak
lintotalrms
linzero
mean(*expr*)
noenv(*expr*)
nojac(*expr*)
pd(f,x)
ppr

prev(*expr*,*i*)
reacf(*u*)
reacf(*u*,*dim*)
realdot(a,b)
scope.ati(*coordinate exprs*,*expr*)
sens(*expr*,*i*)
setconst(*const*,*value*)
setind(*par*,*index*)
setval(*par*,*value*)
shapeorder(*variable*)
side(*entity*,*expr*)
subst(*expr*, *expr1_orig*, *expr1_subst*,...)
sum(*expr*,*index*, *lower*,*upper*)
test(*expr*)
timeint, timeavg
timemax, timemin
treatasconst(*expr*)
try_catch(*tryExpr*, *catchExpr*)
uflux(*u*), dflux(*u*)
up(*expr*)
var(*expr*,*fieldname1*, *fieldname2*, ...)
with
withsol(*tag*,*expr*)