University of Glasgow

EARTH4072 – Igneous Geology

Introduction to Computational Geosciences

WKSHP 3 | Introduction to Comp Modelling I

Dr. Tobias Keller

Tobias.Keller@glasgow.ac.uk

@magmamatters

WORLD CHANGING GLASGOW

Intro Comp Geosci | Programme

Week	WKSHP I	WKSHP II	WKSHP III	WKSHP IV
20/10/2020	First Steps Coding	Comp Data Analysis	Comp Modelling I	Comp Modelling II

Comp GeoSci | Intended Learning Outcomes

Introduction to Computational Modelling I

- understand why and how scientist use modelling
- understand role of conceptual, analytical, numerical models
- understand model dimensionality (0-D, 1-D, 2-D, 3-D)
- understand the fundamentals of discretisation
- understand the fundamentals of model verification
- program and use a 0-D box model of crustal heat transport

```
% update constitutive relations
txx = eta .* exx + chi .* txxo;       % x-normal stress
tzz = eta .* ezz + chi .* tzzo;       % z-normal stress
txz = etac.* exz + chic.* txzo;       % xz-shear stress

p   = - zeta .* Div_V + xi .* po;     % compaction pressure
p([1 end],:) = p([end-1 2],:);        % periodic boundaries
p(:,[1 end]) = p(:,[end-1 2]);

w   = - (K(1:end-1,:).*K(2:end,:)).^0.5 .* (diff(P,1,1)./h + 1);
w(:,[1 end]) = w(:,[end-1 2]);

u   = - (K(:,1:end-1).*K(:,2:end)).^0.5 .* (diff(P,1,2)./h);
u([1 end],:) = u([end-1 2],:);

% update z-reference velocity
Div_tz = diff(tzz(:,2:end-1),1,1)./h + diff(txz,1,2)./h;

res_W(:,2:end-1) = - Div_tz + diff(P(:,2:end-1),1,1)./h + diff(p(:,

res_W([1 end],:) = [sum(res_W([1 end],:),1)./2;sum(res_W([1 end],:)
res_W(:,[1 end]) = res_W(:,[end-1 2]);

W = Wi - alpha.*res_W.*dtW + beta.*(Wi-Wii);

% update x-reference velocity
Div_tx  = diff(txx(2:end-1,:),1,2)./h + diff(txz,1,1)./h;

res_U(2:end-1,:) = - Div_tx + diff(P(2:end-1,:),1,2)./h + diff(p(2:

res_U([1 end],:) = res_U([end-1 2],:);
res_U(:,[1 end]) = [sum(res_U(:,[1 end]),2)./2,sum(res_U(:,[1 end])

U = Ui - alpha.*res_U.*dtU + beta.*(Ui-Uii);

% update reference pressure
Div_V(2:end-1,2:end-1) = diff(U(2:end-1,:),1,2)./h + diff(W(:,2:end
Div_v(2:end-1,2:end-1) = diff(u(2:end-1,:),1,2)./h + diff(w(:,2:end

res_P = Div_V + Div_v;

res_P([1,end],:) = res_P([end-1,2],:);
res_P(:,[1,end]) = res_P(:,[end-1,2]);

P = Pi - alpha.*res_P.*dtP + beta.*(Pi-Pii);

% update liquid evolution equation (enforce min/max limits on f)

flxdiv_fromm;  % upwind-biased advection/compaction term for liquid

res_f = (f-fo)./dt - (theta.*Div_fV + (1-theta).*Div_fVo);

res_f([1,end],:) = res_f([end-1,2],:);
res_f(:,[1,end]) = res_f(:,[end-1,2]);

if ~mod(step,nop); res_f = res_f - mean(res_f(:)); end

f = fi - alpha.*res_f.*dt/50;
f = max(0.001/f0,min(0.999/f0, f ));

% check and report convergence every nup iterations
if ~mod(it,nup); report; end
```

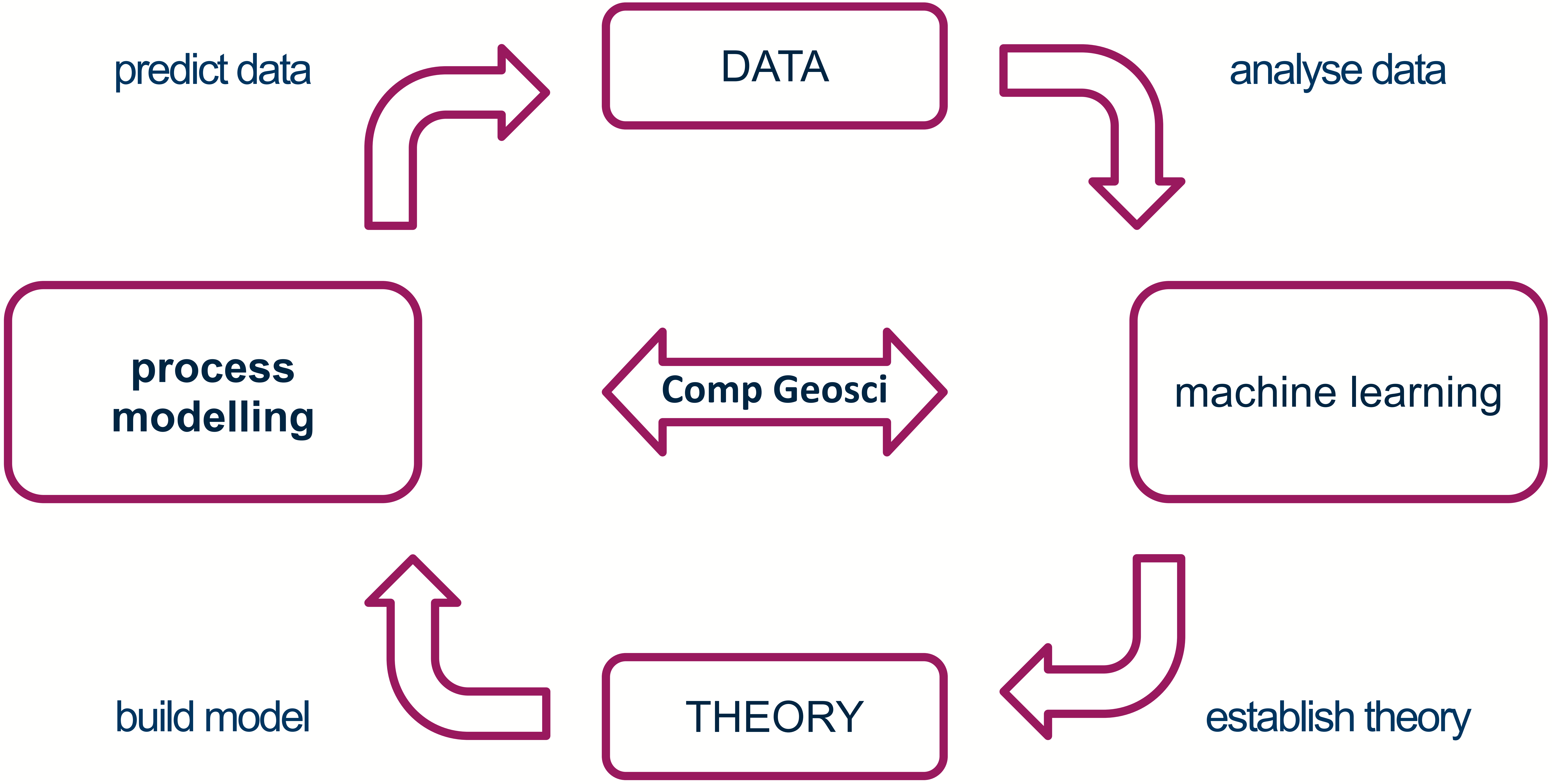
What is a model?

a simplified representation of a complex reality

Models in Geosciences

- global circulation models
- mantle convection models
- plate reconstruction models
- geochemical evolution models
- mantle tomography models
- …

Comp Modelling | Basics



Comp Modelling | Basics

Definition

A simplified representation of a natural process or system aimed at interpreting, understanding, and predicting data.

"Universal Law" of Modelling

All models are wrong – some are useful.

"Verification and validation of numerical models of natural systems is impossible. This is because natural systems are never closed and because model results are always non-unique. Models can be confirmed by the demonstration of agreement between observation and prediction, but confirmation is inherently partial. … The primary value of models is heuristic."

Oreskes et al., *Science*, 1994

Comp Modelling | Process Modelling

Process-driven

- start with fundamental laws, build model to predict data

Hypothesis-based

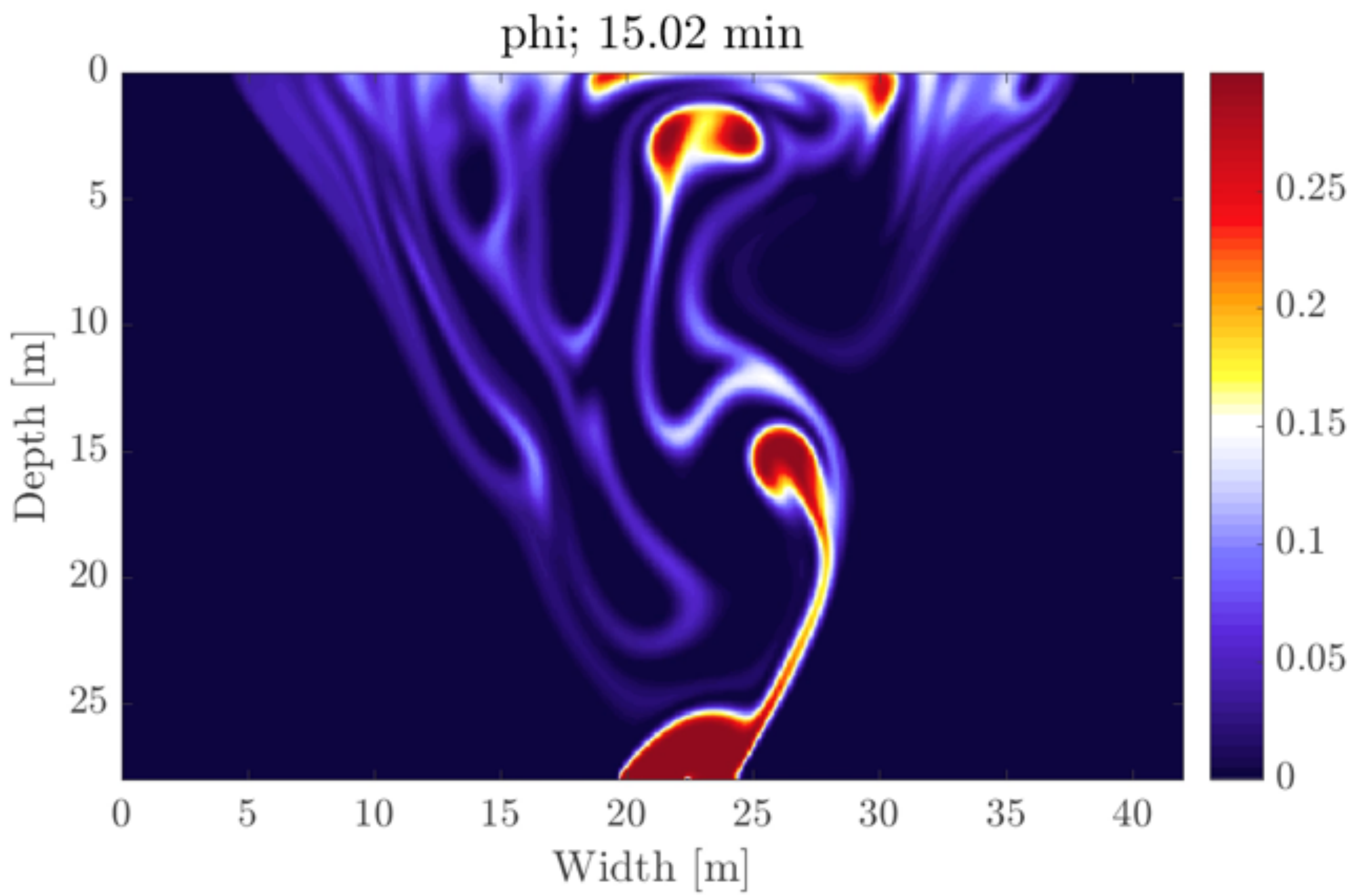
- testing hypothesis more important than fitting data

Uncertainty

- theory incomplete, models idealised, parameters poorly constrained

Examples

- lava flow simulation, rock deformation model, climate model



Lava lake convection at Mt Erebus, Antarctica (Birnbaum et al., EPSL, 2019)

Comp Modelling | Terminology

perceptual	simplified understanding based on our perception
conceptual	simplified representation using basic concepts
analogue	analogue system built under controlled conditions
computational	mathematical model based on computer code
analytical	exact mathematical model with continuous variables
numerical	approximate math. model with discrete variables

Comp Modelling | Terminology

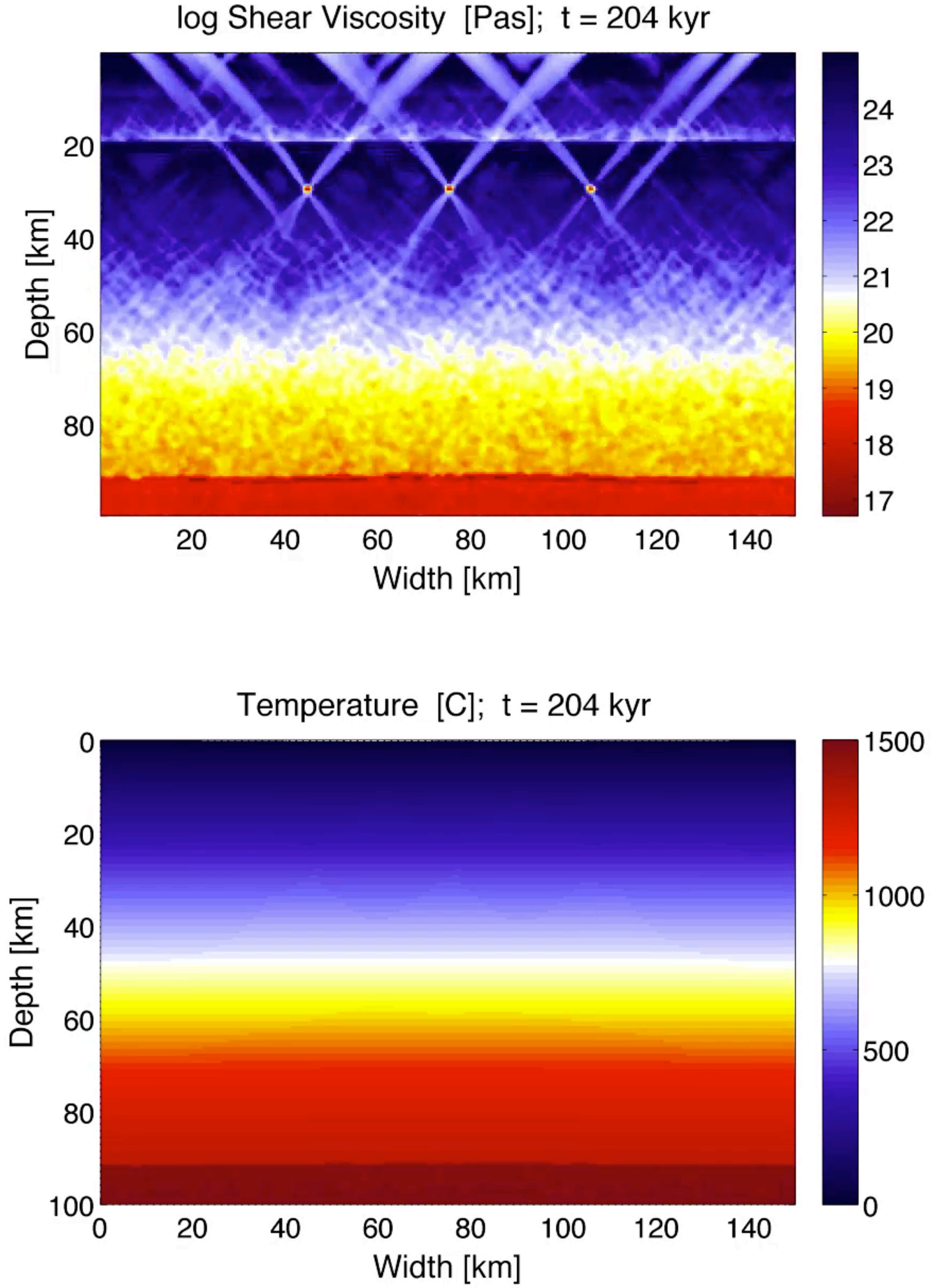
variable	physical quantities used to calculate model solution (e.g., temperature, space, time)
known	independent variable we have control of (e.g., spatial coordinates, time)
unknown	dependent variable we wish to calculate (e.g., temperature diffusing away from a hot intrusion)
parameter	properties specified by the user to adjust the model outcome (e.g. heat diffusivity of rock)
initial condition	initial value at start of time-evolution model (e.g. initial temperature of an intrusion)
boundary cond.	conditions at model domain boundaries (e.g. constant temperature value, constant heat flux)

Comp Modelling | Example Application

Magma Intrusion into Crust

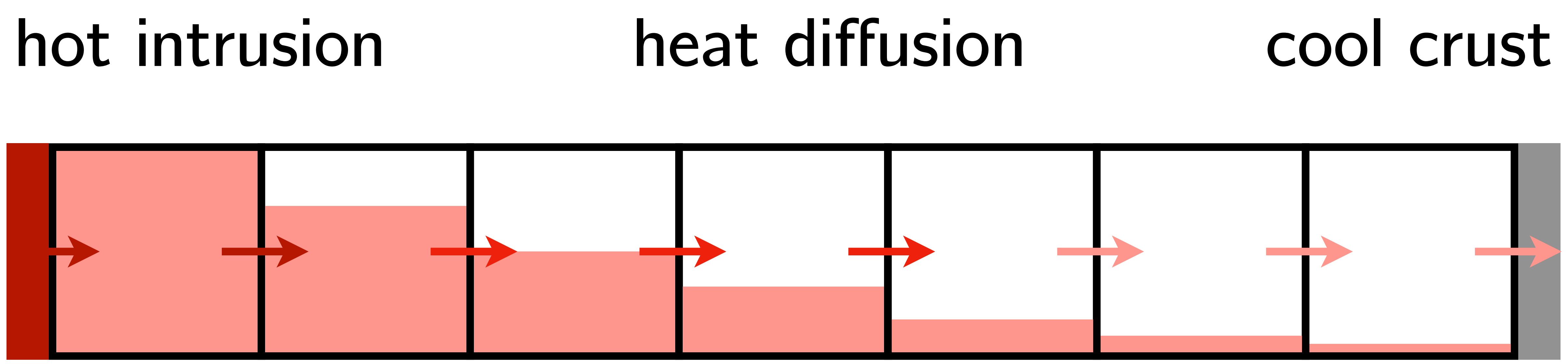
- hot magma rises into cool crust
- magma-filled tensile fractures: dyke
- heat transfer from magma to crust

en.wikipedia.org/wiki/Dike_(geology)

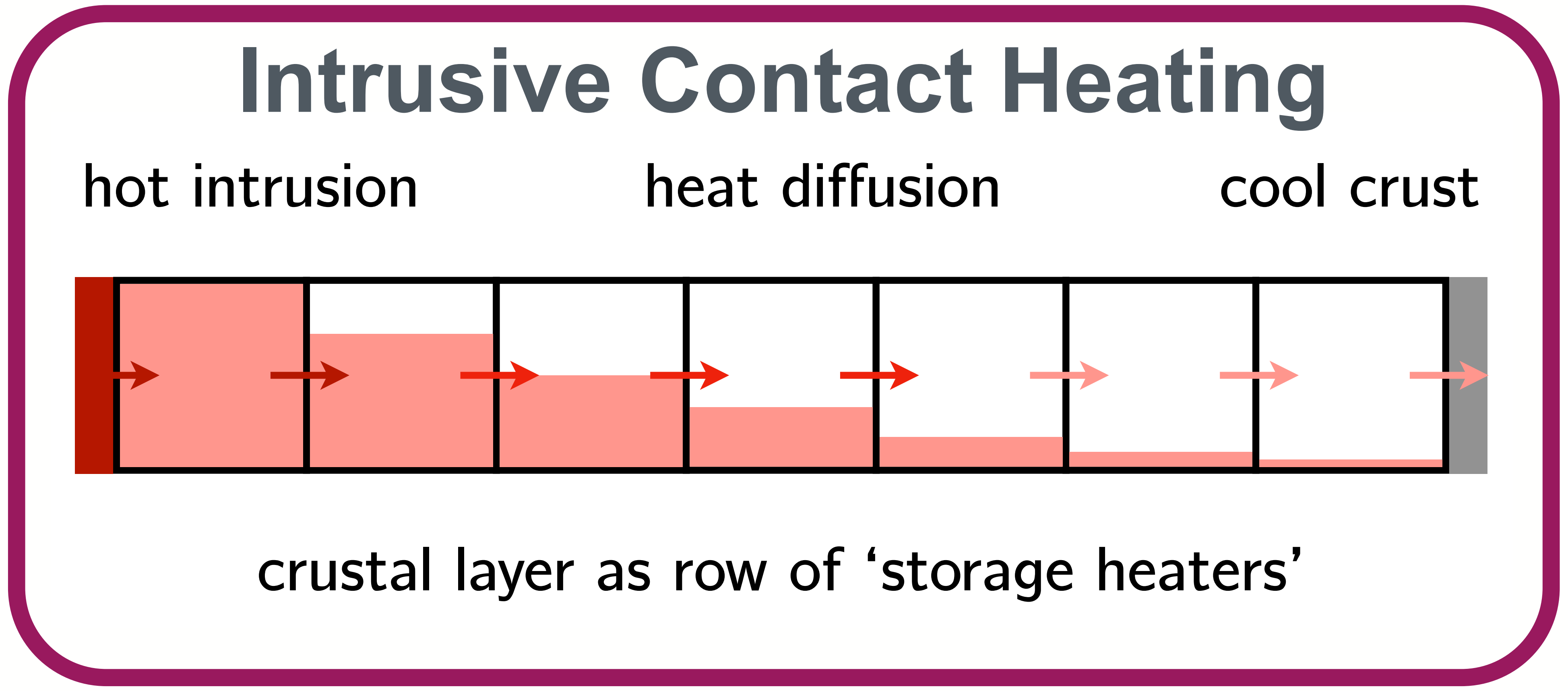


2-D model of magma ascent, heat transfer

Conceptual Model



Comp Modelling | Concept to Model



Conceptual model

- conceptual understanding of a natural process

Fundamental principles

- identify what fundamental principles apply (e.g., conservation of energy, production of entropy)

Governing equations

- find mathematical form based on fundamental principles, simplifying assumptions

rate of temperature change

$$\frac{\partial T(\mathbf{x}, t)}{\partial t} = -\nabla \cdot \mathbf{q}(\mathbf{x})$$

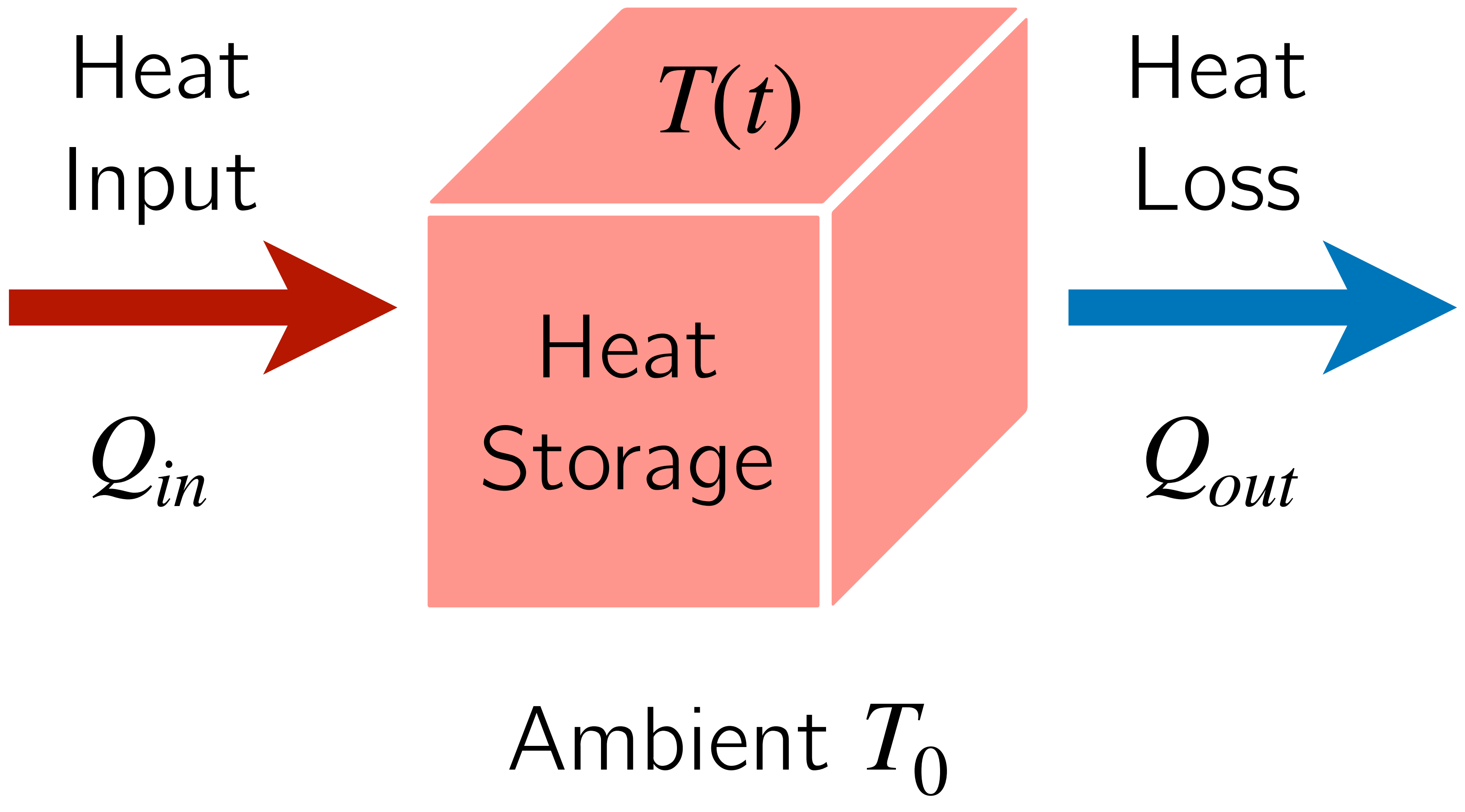
balance between heat in-, outflow

Comp Modelling | Model to Solution

Dimensionality

- how many spatial dimensions and/or time to represent model?
- 0-dimensional model: only time-dependence, no spatial variability

Storage Heater Model



rate of temperature change

$$\frac{dT(t)}{dt} = Q_{in} - Q_{out}$$

balance between heat in-, outflow

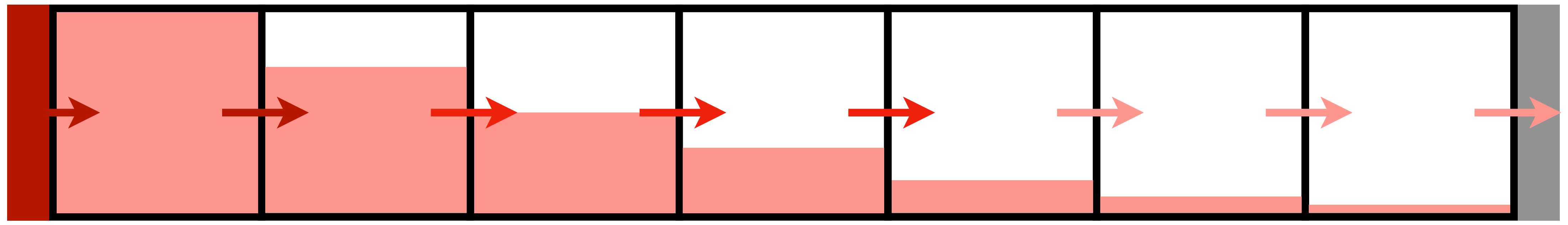
Comp Modelling | Model to Solution

Dimensionality

- decide how many spatial dimensions and/or time to represent model in
- 1-dimensional model, time-dependence, one spatial dimension

Intrusive Contact Heating Model

hot intrusion heat diffusion cool crust



crustal layer as row of ‘storage heaters’

change in temperature through time

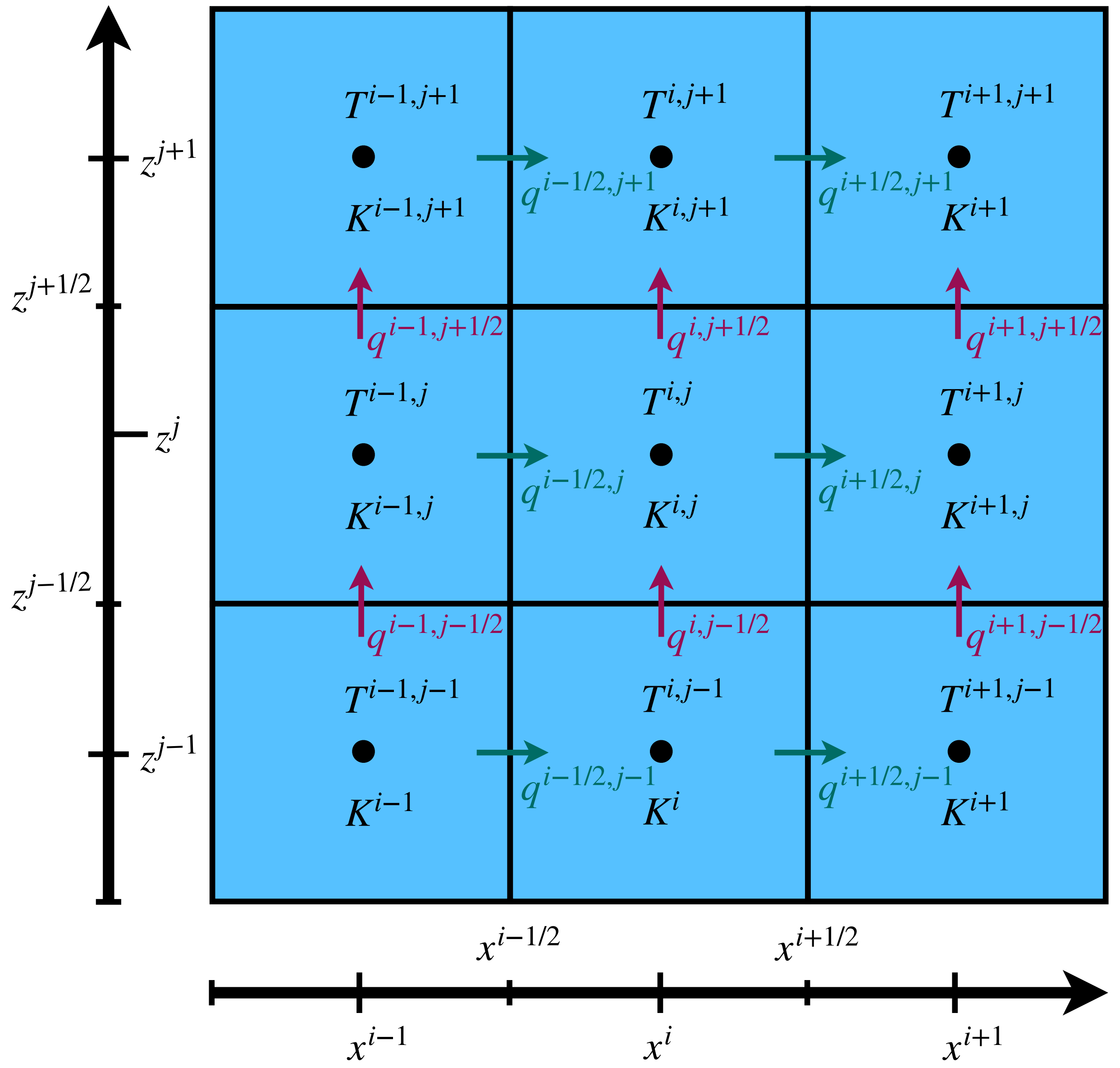
$$\frac{\partial T(t,x)}{\partial t} = -\frac{\partial q(x)}{\partial x}$$

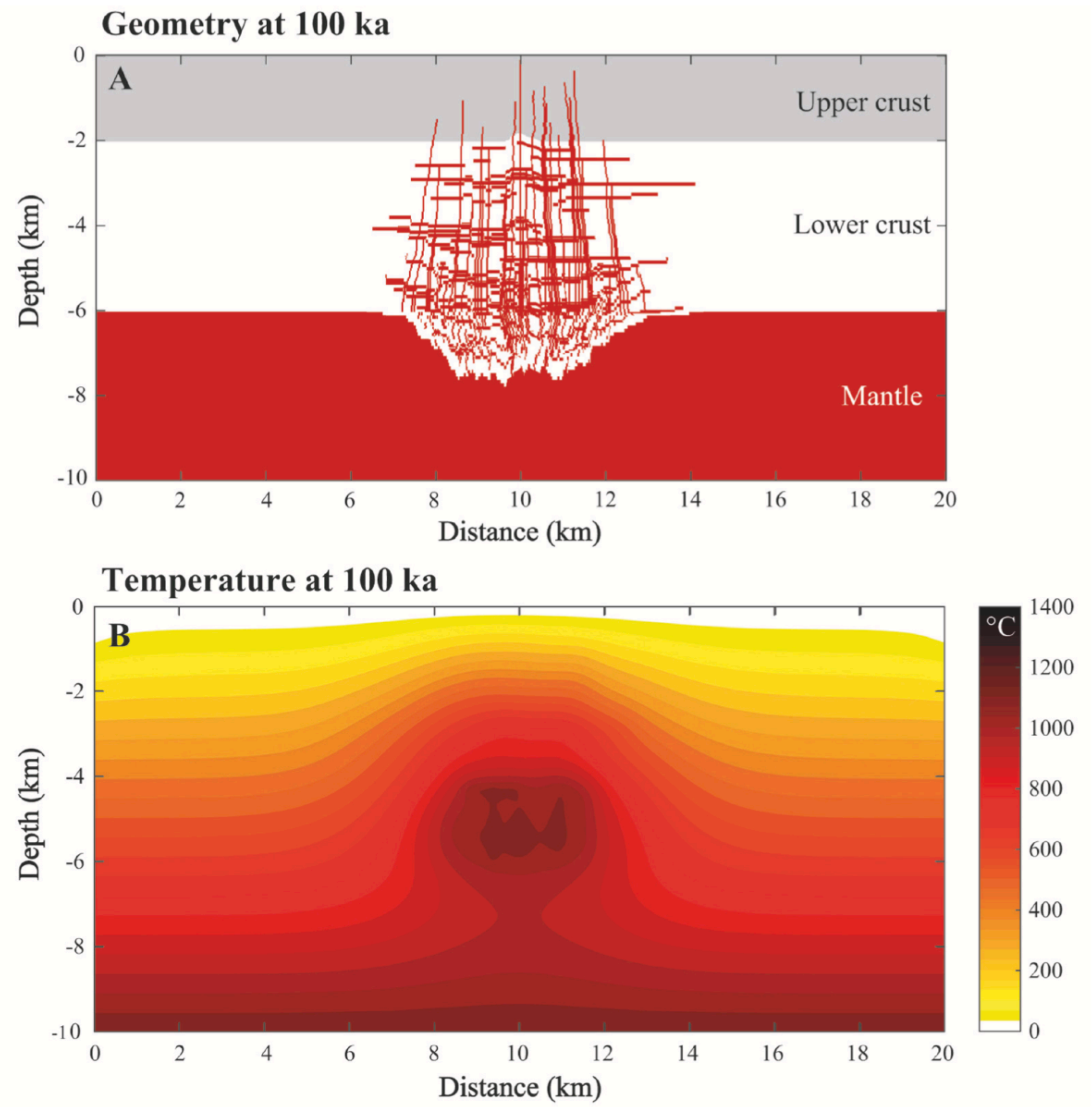
change in heat flux through x-direction

Comp Modelling | Model to Solution

Dimensionality

- same procedure to expand from one- to two- and three-dimensional model
- higher dimensionality => closer to natural system => higher complexity => higher cost!

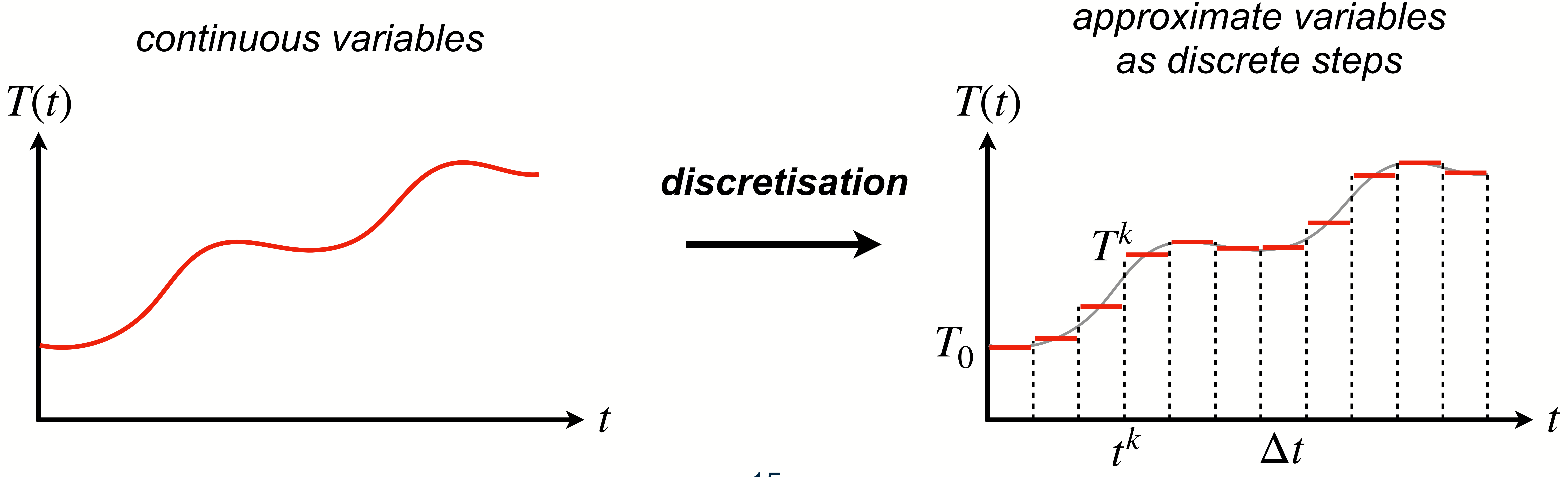




Comp Modelling | Model to Solution

Discretisation (0-D box model)

- approximate continuous variables as discrete values on discrete points in time
- discrete time: $t^k \ (k = 1,...,m)$, discrete temperatures: $T^k = T(t = t^k)$



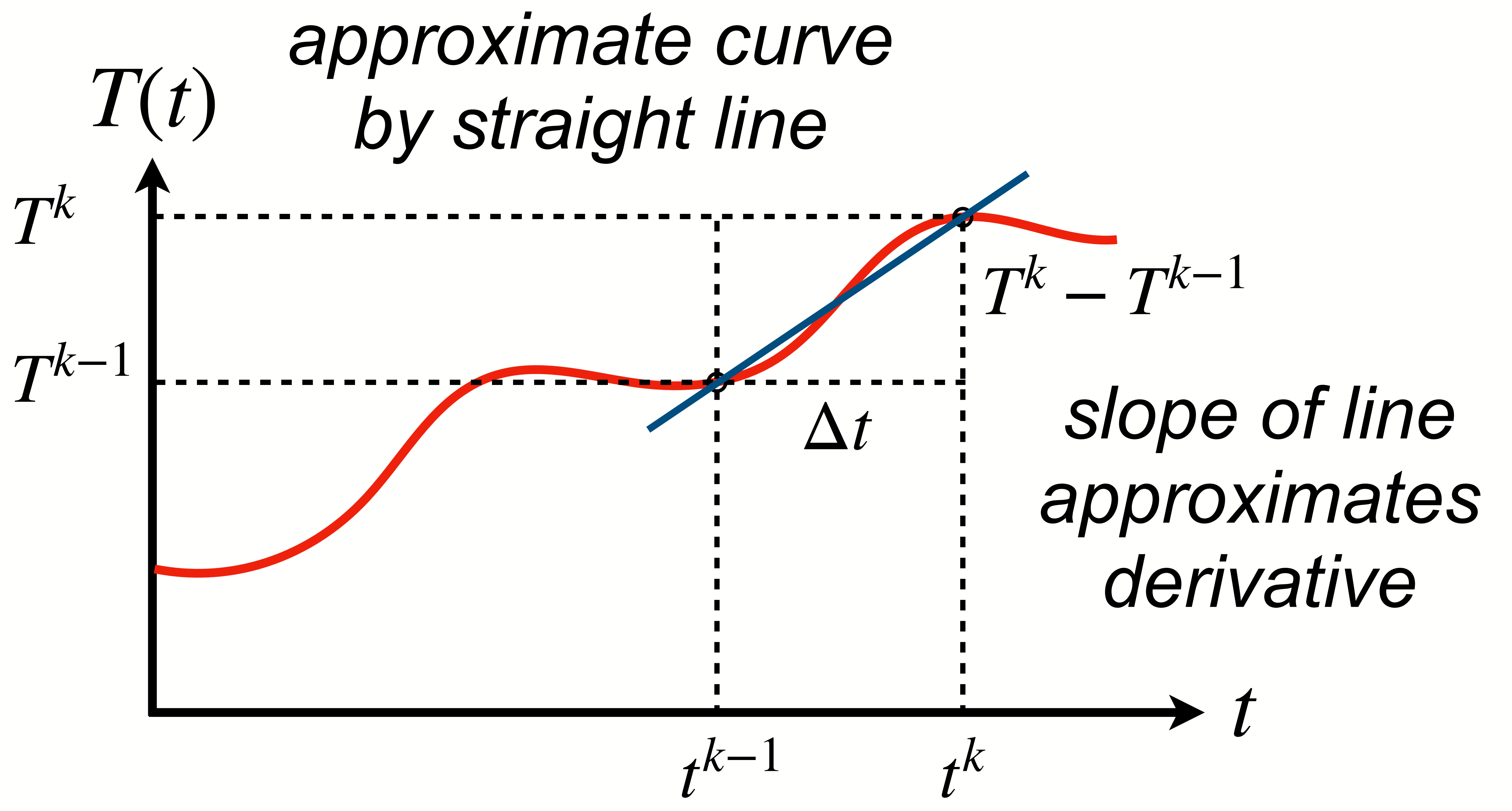
Comp Modelling | Model to Solution

Discretisation (0-D box model)

- approximate continuous rate of change as discrete steps of change
- approximate *infinitesimal derivative* by *finite difference*

finite difference approximation

$$\frac{dT}{dt} \approx \frac{T^k - T^{k-1}}{\Delta t}$$



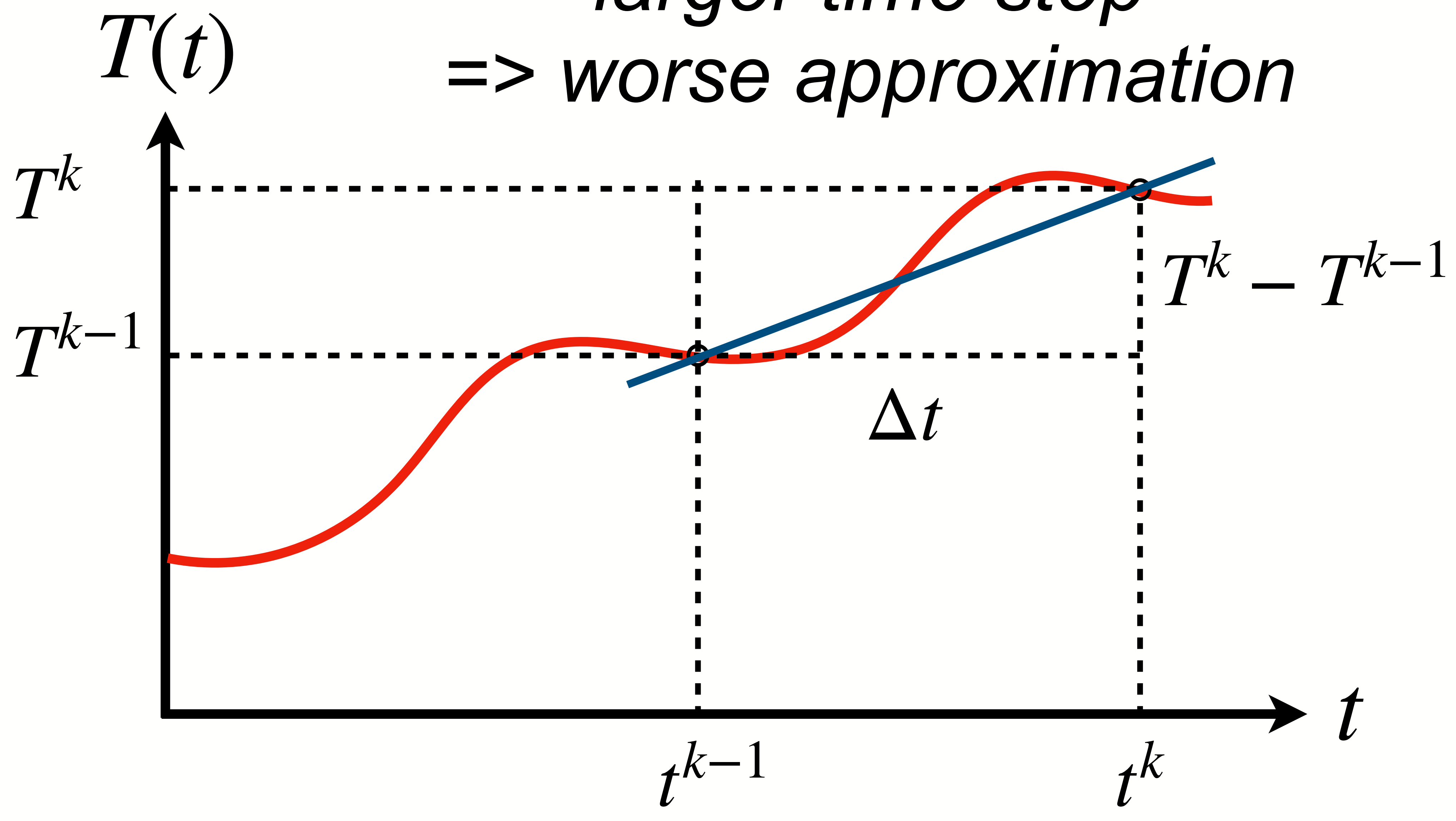
Discretisation (0-D box model)

- approximate continuous rate of change as discrete steps of change
- approximate *infinitesimal derivative* by *finite difference*

finite difference approximation

$$\frac{dT}{dt} \approx \frac{T^k - T^{k-1}}{\Delta t}$$

larger time step => worse approximation



Comp Modelling | Model to Solution

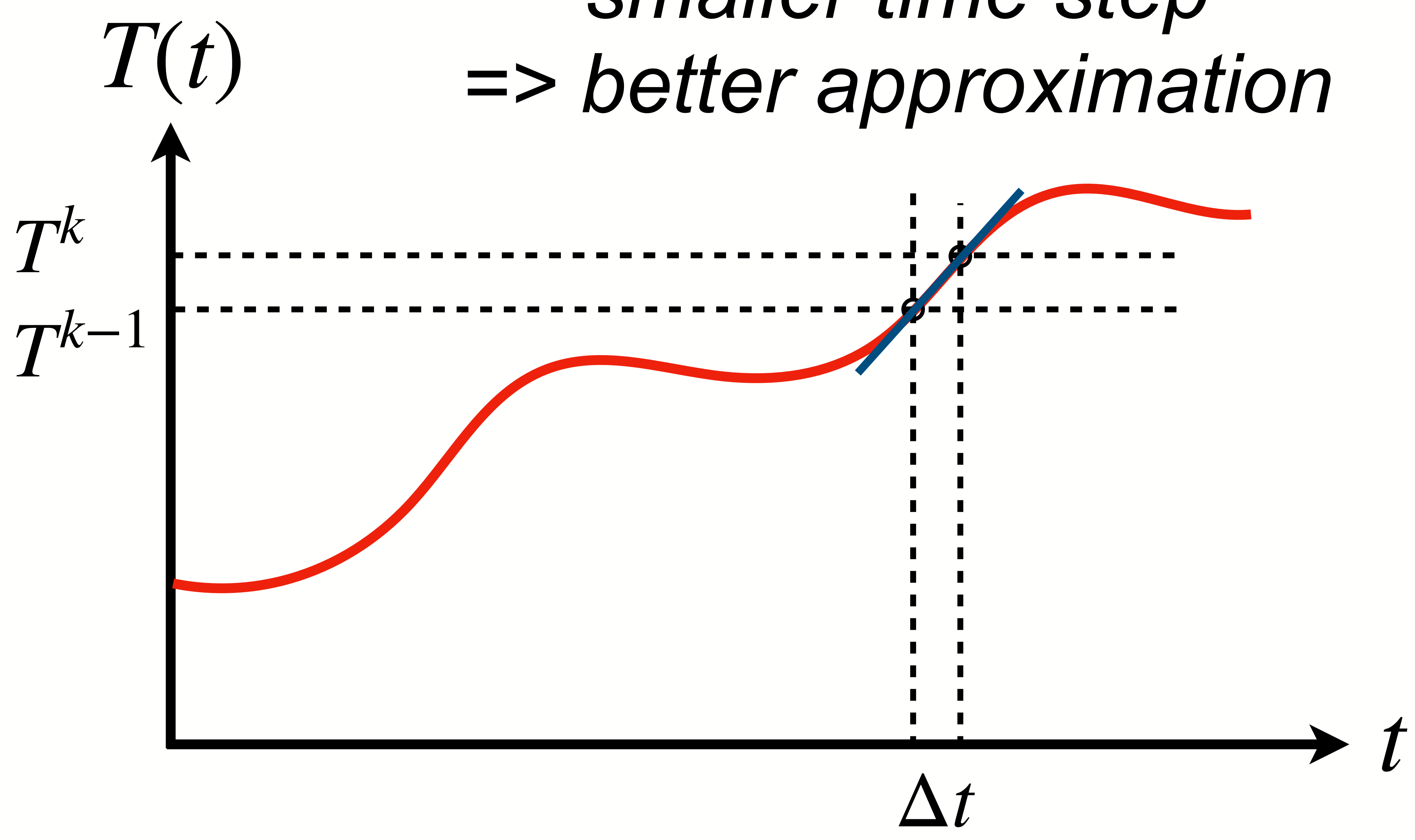
Discretisation (0-D box model)

- approximate continuous rate of change as discrete steps of change
- approximate *infinitesimal derivative* by *finite difference*

finite difference approximation

$$\frac{dT}{dt} \approx \frac{T^k - T^{k-1}}{\Delta t}$$

smaller time step => better approximation



University of Glasgow

Comp Modelling | Model to Solution

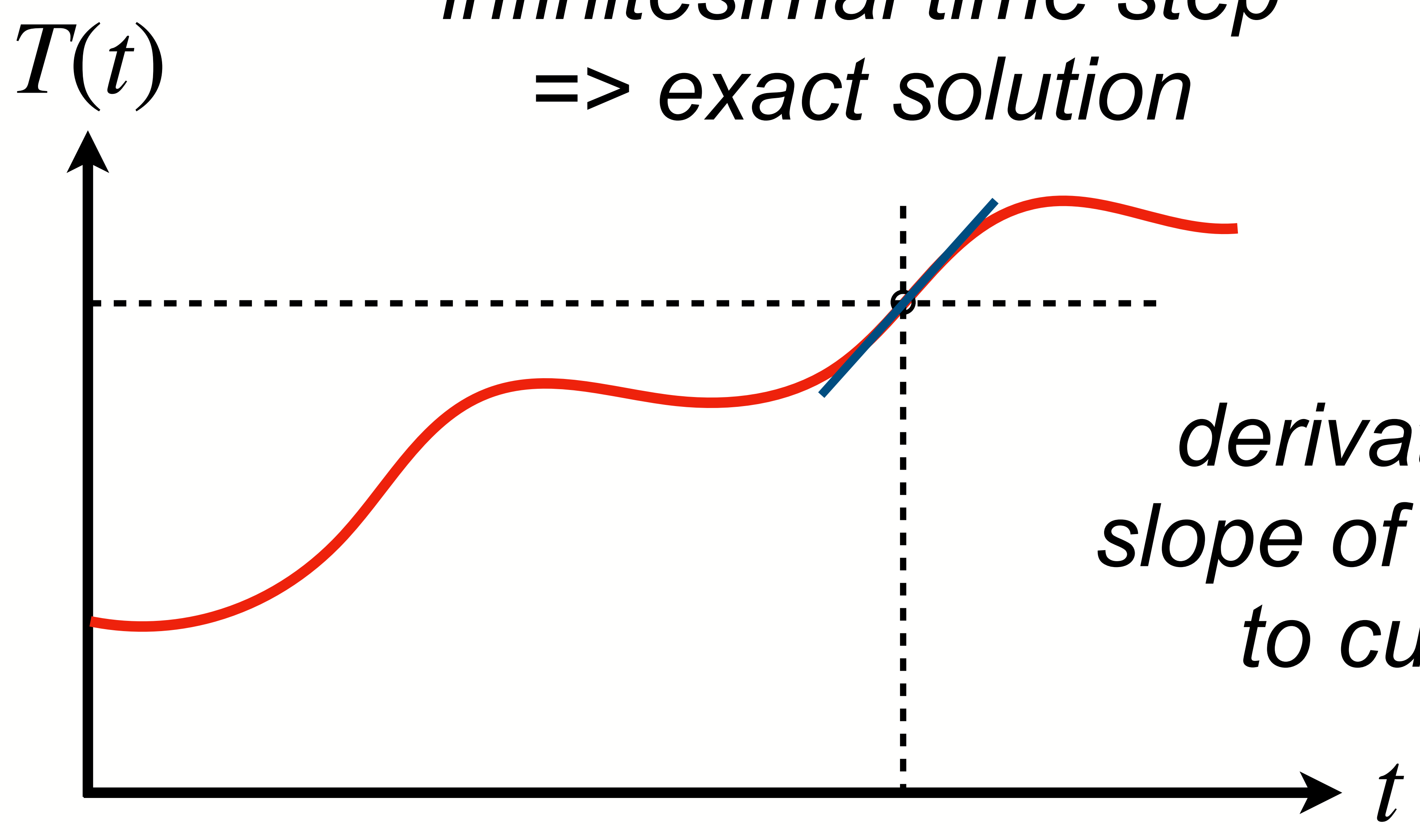
Discretisation (0-D box model)

- approximate continuous rate of change as discrete steps of change
- approximate *infinitesimal derivative* by *finite difference*

finite difference approximation

$$\frac{T^k - T^{k-1}}{\Delta t} = \frac{dT}{dt} \quad \text{for } \Delta t \to 0$$

infinitesimal time step => exact solution



derivative is slope of tangent to curve!

Comp Modelling | Model to Solution

Discretisation (0-D box model)

- rearrange discrete equation into *explicit scheme*
- *unknown* variable on left, *known* variables on right

discretised governing equation

$$\frac{T^k - T^{k-1}}{\Delta t} = Q_{in} - Q_{out}$$

explicit scheme

parameters

$$T^k = T^{k-1} + \left(Q_{in} - Q_{out}\right) \Delta t$$

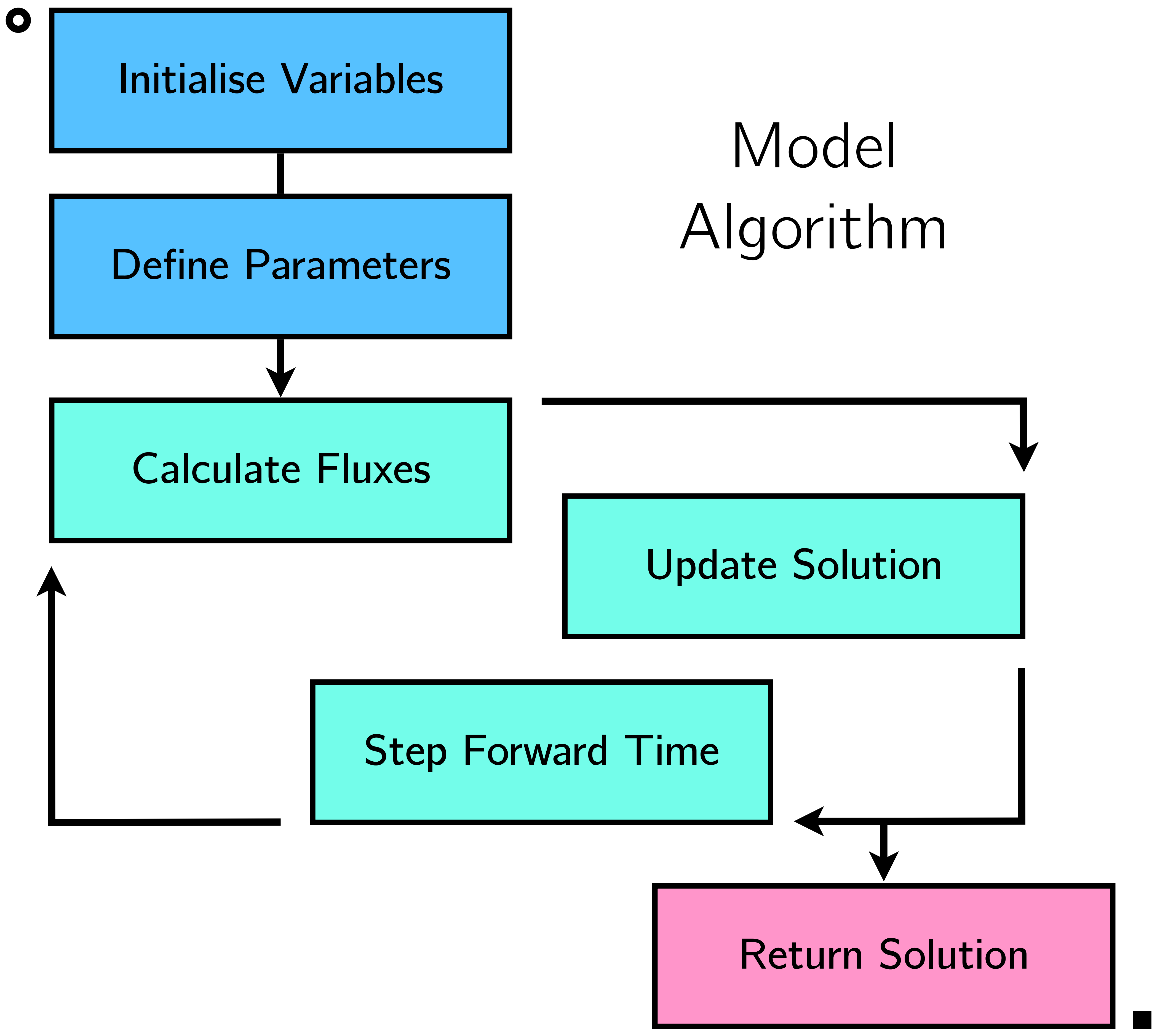
unknown

knowns

Comp Modelling | Model to Solution

Numerical Implementation

- program algorithm to calculate solution step by step
- check if solution is robust, no numerical artefacts
- verify code to ensure accurate numerical solution



Comp Modelling | Terminology

verification

- *did we build the model right?*
- does the numerical model produce the correct solution to the model equation?

validation

- *did we build the right model?*
- e.g., does the model usefully approximate the natural system?

calibration

- what choice of parameters produces the best model fit to available dataset?

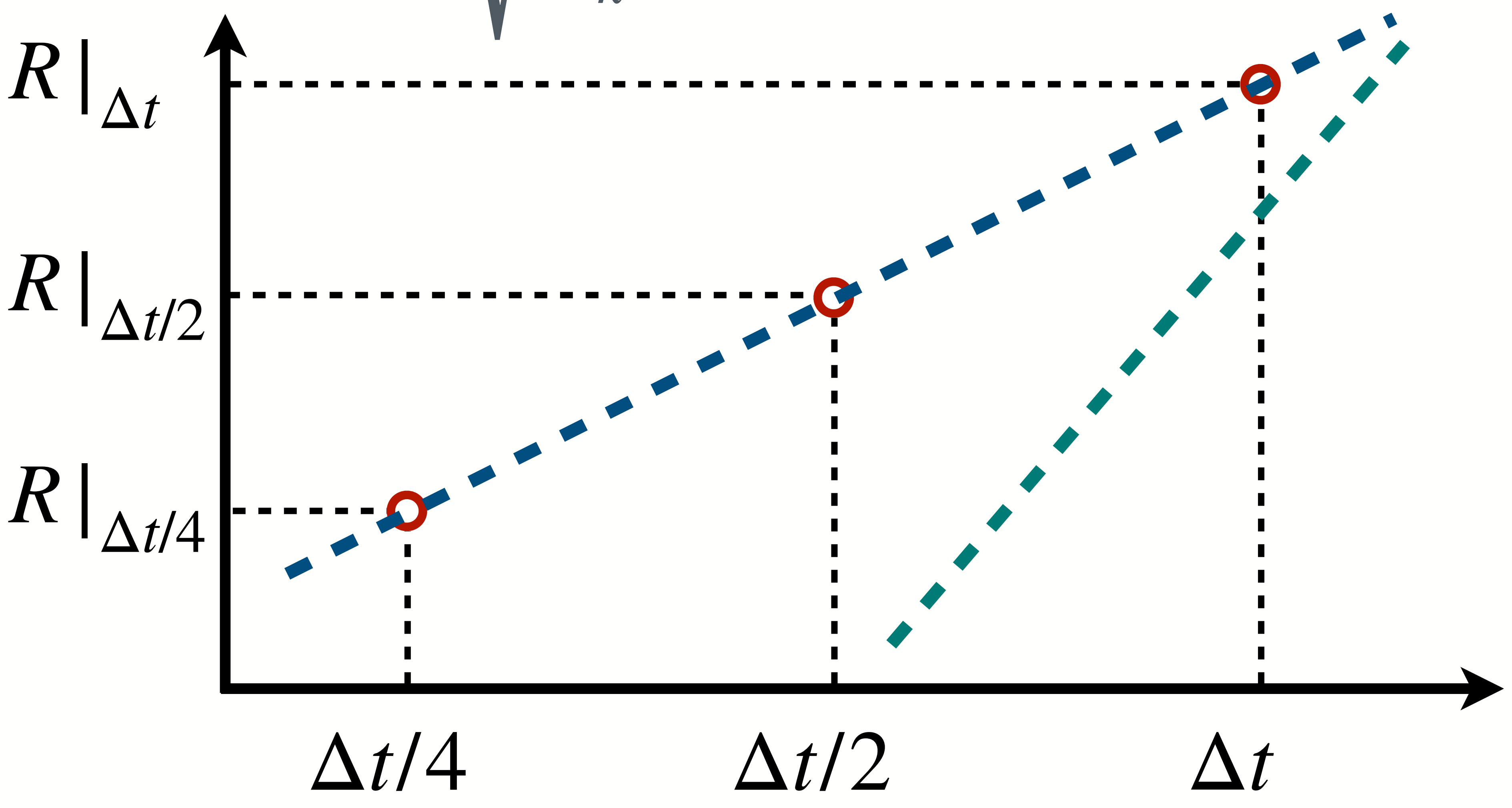
Comp Modelling | Benchmarking

$$T(t) = T_0 + (T_\infty - T_0)\left[1 - \exp\left(-t/t_{eq}\right)\right]$$

Model Verification

- if possible, compare numerical approximation to exact analytical solution
- perform resolution test: reduce step size, measure error at each level
- model convergence: if model correct, error reduces proportionally to discrete step size
- numerical analysis predicts convergence rate:
  - ***first-order***
    error reduces same factor as step size
  - ***second-order***
    error reduces square as of step size
  - etc.

$$R = \sqrt{\sum_k (T^k_{\mathrm{num}} - T(t^k))/N_k}$$



Activity | Introduction to Numerical Modelling with Python

Build simple box model of heat transfer

- hot magma intrusion heats crustal wall rock
- simplified representation as storage heater
- heat input vs. heat output rate

Get started with basic numerical modelling

- complete prepared model code
- test different rate parameters
- verify your code, compare to analytical solution
- build second-order accurate version

Padlet task

- share an example of computational modelling in geosciences