A Gappy Simulation

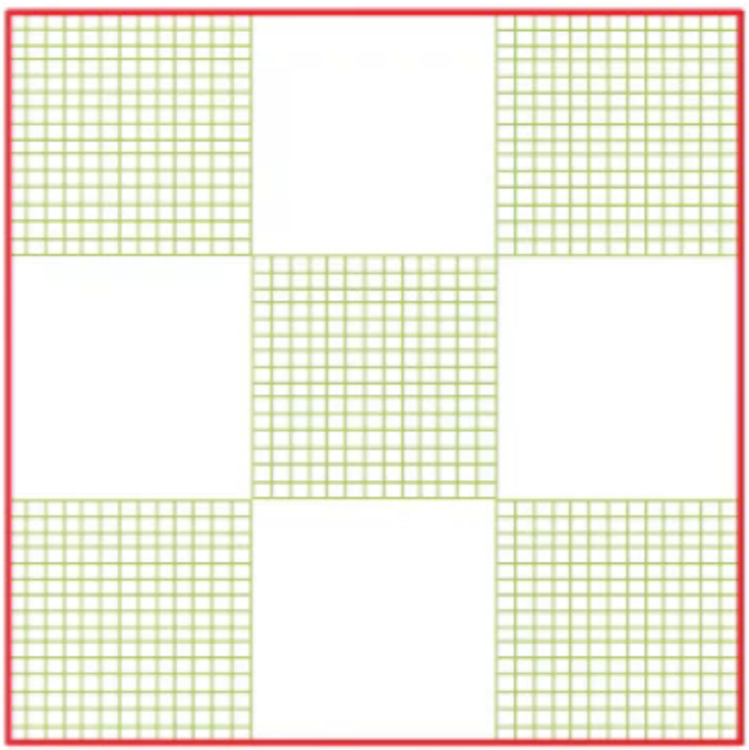
with a Multi-fidelity Information Fusion

Seungjoon Lee

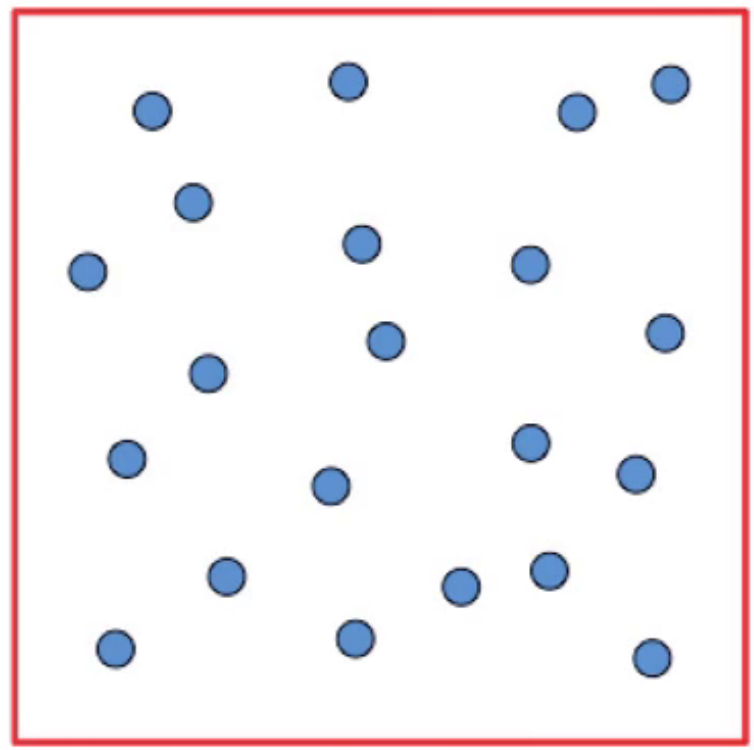
Division of Applied Mathematics
Brown University

Gappy Simulation

- Solve only selected subdomains independently with partial information.
- Partial information – coarse but global information.



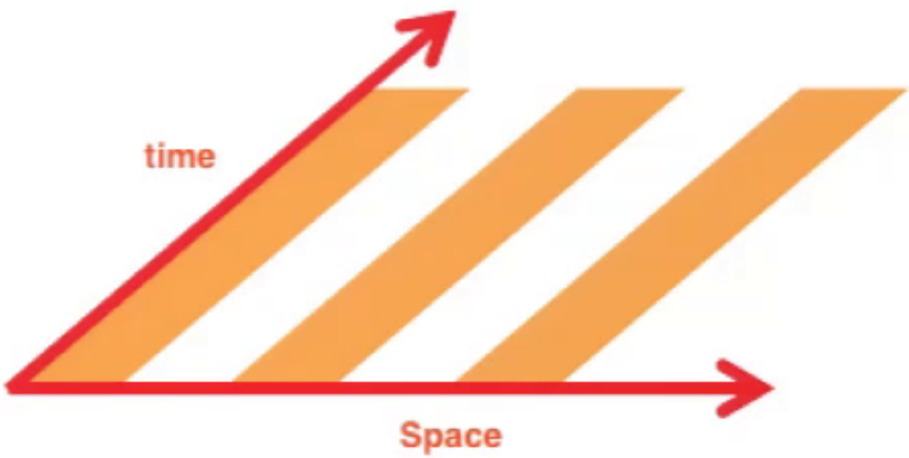
Local/refine information



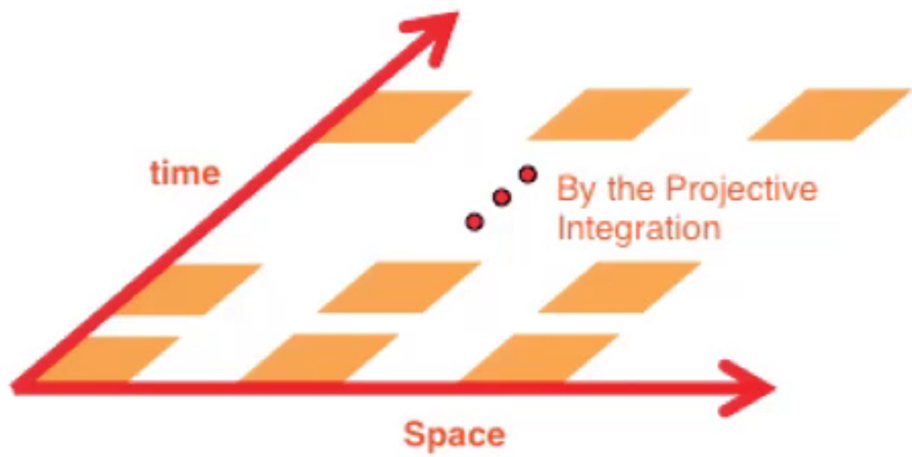
Global/coarse information

Gappy Simulation

- Spatio-gappy simualtion - gaps in space only (gap-tooth algorithm).
- Saptio-temporal gappy simulation –gaps in space and time (patch dynamics).



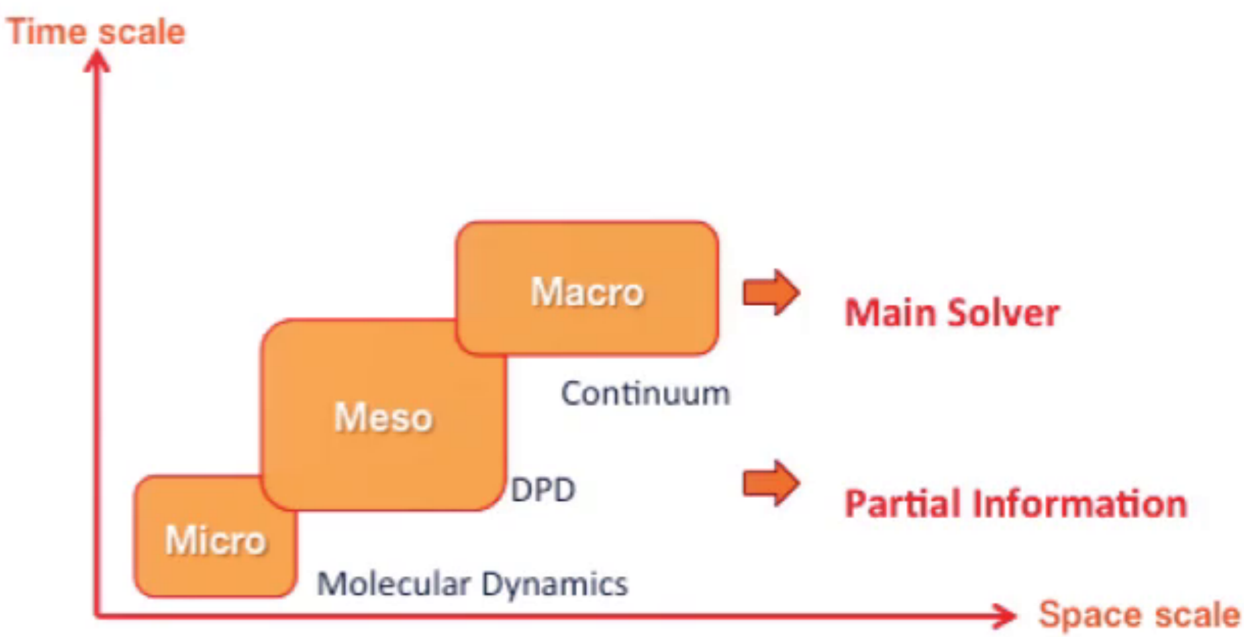
Solution space of spatio-gappy simulation

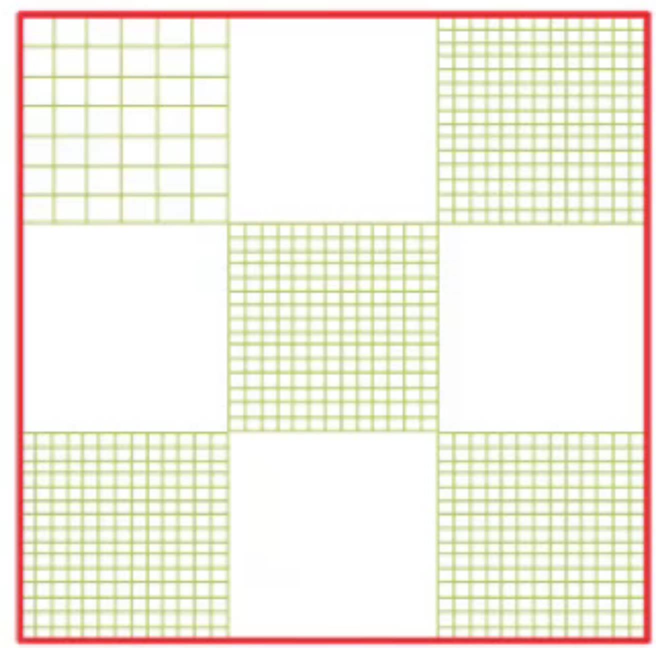


Solution space of spatio-temporal gappy simulation

Motivations

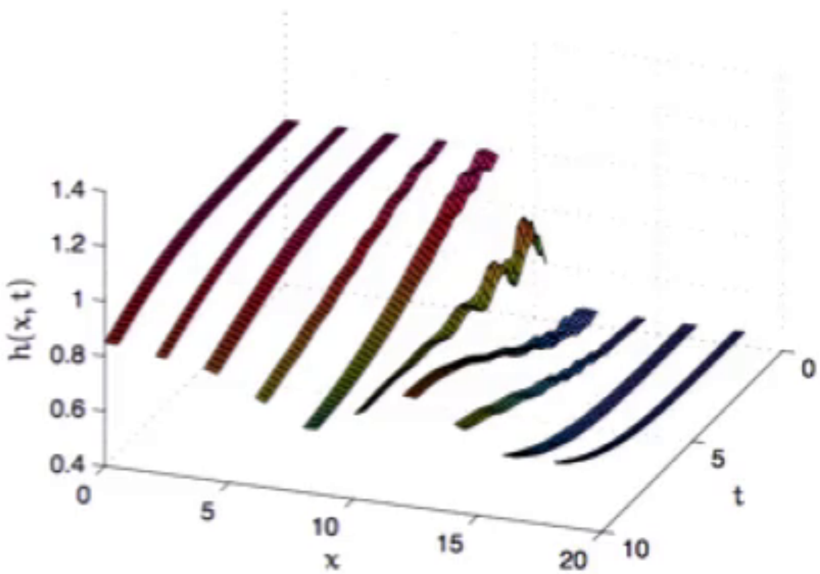
- ***Basic Idea***: If our solution is smooth enough, we can solve only subdomains.
- Computational cost reduction.
- Adaptively refinement.
- Easy-to-combine multi-fidelity/multi-scale problem.
- Machine learning based information fusion.
- Expand the efficient framework from ***dynamical systems*** to ***continuum system***.



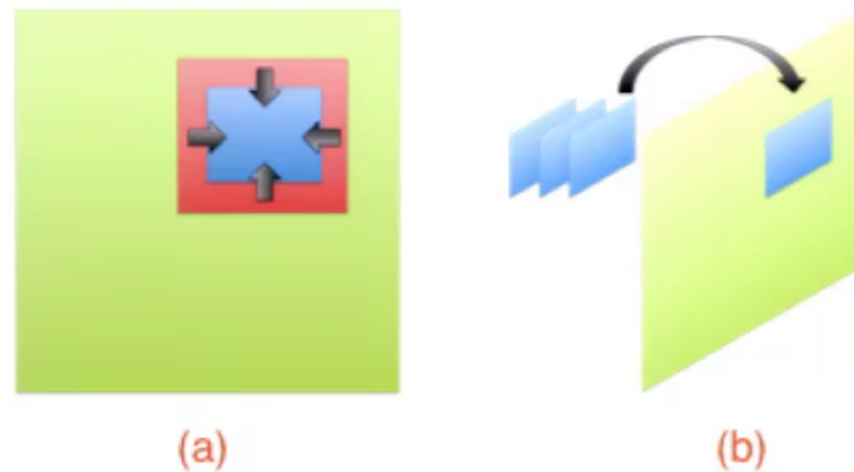


General Algorithm

- Start from a "gap-tooth" algorithm for micro simulators.
- Combine with estimation theories.



Indicative gap-tooth simulation of a dam-break shows the water depth h(x, t) [1)]

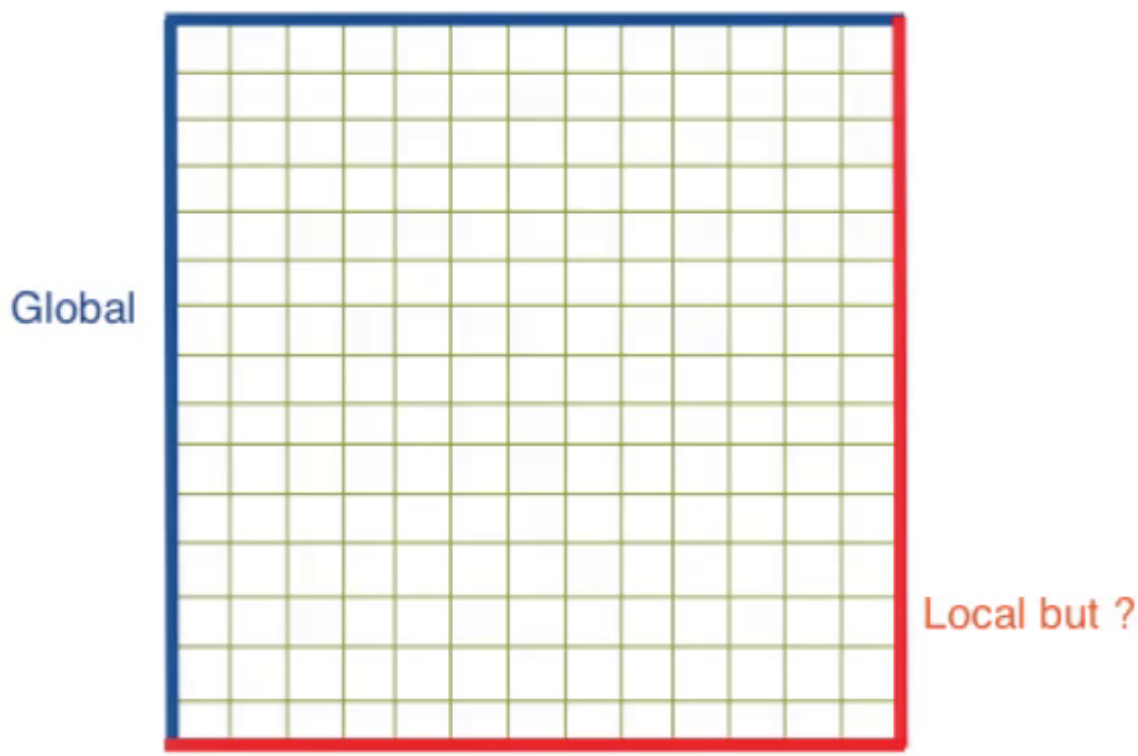


Estimation theories: (a) spatio-estimation (b) temporal estimation [2)]

1) M. Cao *et al.* Multiscale modelling couples patches of nonlinear wave-like simulations (2014).
2) S. Lee *et al.* Resilient Algorithms for Reconstructing and Simulating Gappy Flow Fields in CFD (2015).

A Key Issue in a Gappy Simulation

- Imposing boundary condition at local boundary by multi-fidelity information fusion.



Multi-fidelity information fusion: CoKriging

- CoKriging with two different information – local/refine and global/coarse.
- Estimate x_b by the linear combination of these two information.

$$\mathbf{x} = \{x \in \Omega_P\} \quad \text{and} \quad \mathbf{y} = \{y \in \Omega_L\}$$

Where Ω_L is a set of partial domains (local/refine)

Ω_P is a set of partial information (global/coarse)

$$\hat{u}(\mathbf{x_b}) = \lambda_\mathbf{1}^\mathbf{T}\mathbf{x} + \lambda_\mathbf{2}^\mathbf{T}\mathbf{y}$$

$$\arg\min_{\lambda\mathbf{1},\lambda\mathbf{2}} E[(\hat{u}(\mathbf{x_b}) - u(\mathbf{x_b}))^2]$$

Subject to

$$\sum \lambda_i^1 = 1 \quad \text{and} \quad \sum \lambda_i^2 = 0$$

Multi-fidelity information fusion: CoKriging

- By matrix form,

$$\begin{bmatrix} C_{11} & C_{12} & \mathbf{1} & \mathbf{0} \\ C_{21} & C_{22} & \mathbf{0} & \mathbf{1} \\ \mathbf{1}^{\mathsf{T}} & \mathbf{0}^{\mathsf{T}} & 0 & 0 \\ \mathbf{0}^{\mathsf{T}} & \mathbf{1}^{\mathsf{T}} & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda^{\mathbf{1}} \\ \lambda^{\mathbf{2}} \\ \mu^{\mathbf{1}} \\ \mu^{\mathbf{2}} \end{bmatrix} = \begin{bmatrix} \mathbf{c_1}(\mathbf{x_b}) \\ \mathbf{c_2}(\mathbf{x_b}) \\ 1 \\ 0 \end{bmatrix}$$

Where C_{11} covariance matrix between Ω_P

C_{12} covariance matrix between Ω_P and Ω_L

C_{22} covariance matrix between Ω_L

c_1 covariance vector between x_b and Ω_p

c_1 covariance vector between x_b and Ω_L

Multi-fidelity information fusion: CoKriging

- Different type of correlation kernels like below.
- Hyper-parameters are calculated by "least-squared" or "Maximum likelihood".

Power	$\gamma(\mathbf{h}) = Ah^c$
Exponential	$\gamma(\mathbf{h}) = A\left[1 - e^{-h/B}\right]$
Spherical	$\gamma(\mathbf{h}) = A\left[1.5(h/B) - 0.5(h/B)^3\right]$
Gaussian	$\gamma(\mathbf{h}) = A\left[1 - e^{-(h/B)^2)}\right]$
Sine wave	$\gamma(\mathbf{h}) = A\left[1 - (B/h)\sin(h/B)\right]$

Multi-fidelity information fusion: CoKriging

- Dirichlet/Neumann/Robin boundary condition.

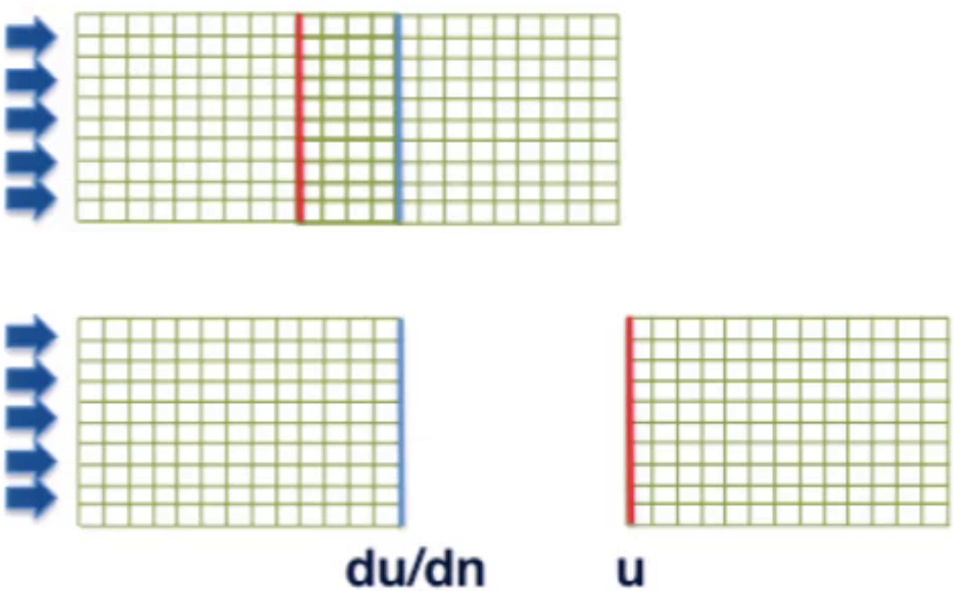
Dirichlet
$$u_d(x_b) = \sum_{i=1}^{n1} \lambda_i^1 u(x_i) + \sum_{i=1}^{n2} \lambda_i^2 u(y_i)$$

Neumann
$$u_n(x_b) = \frac{u_d(x_b + dx) - u_d(x_b - dx)}{2dx}$$

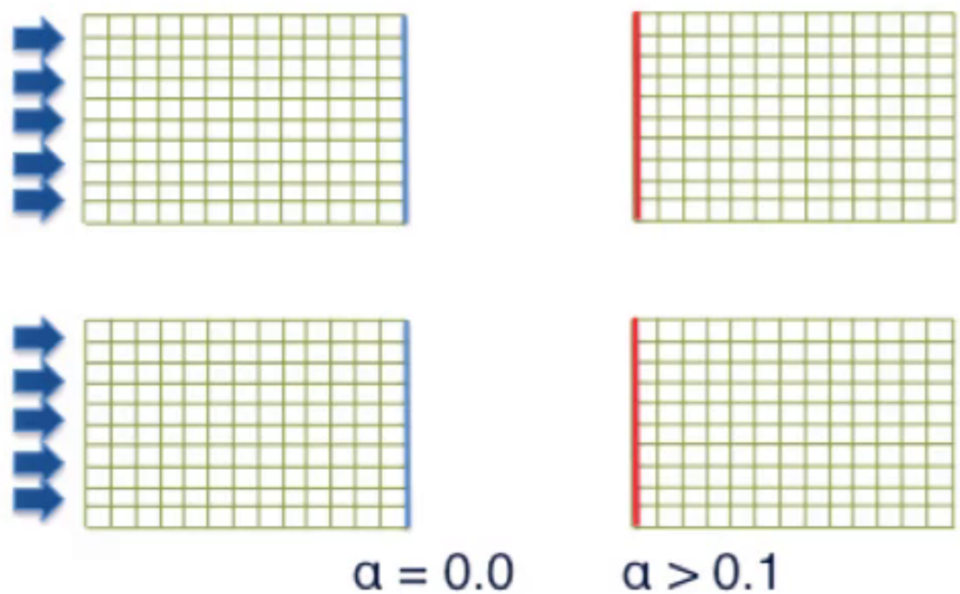
Robin
$$u_r(x_b) = \alpha u_d(x_b) + (1 - \alpha) u_n(x_b)$$

Boundary condition

- Use “Robin” boundary condition. $u_r = \alpha u_d + (1-\alpha) u_n$
- Same analogy of Inter-Patch Conditions in overlapped domain decomposition.



Inter-Patch Condition [1)]

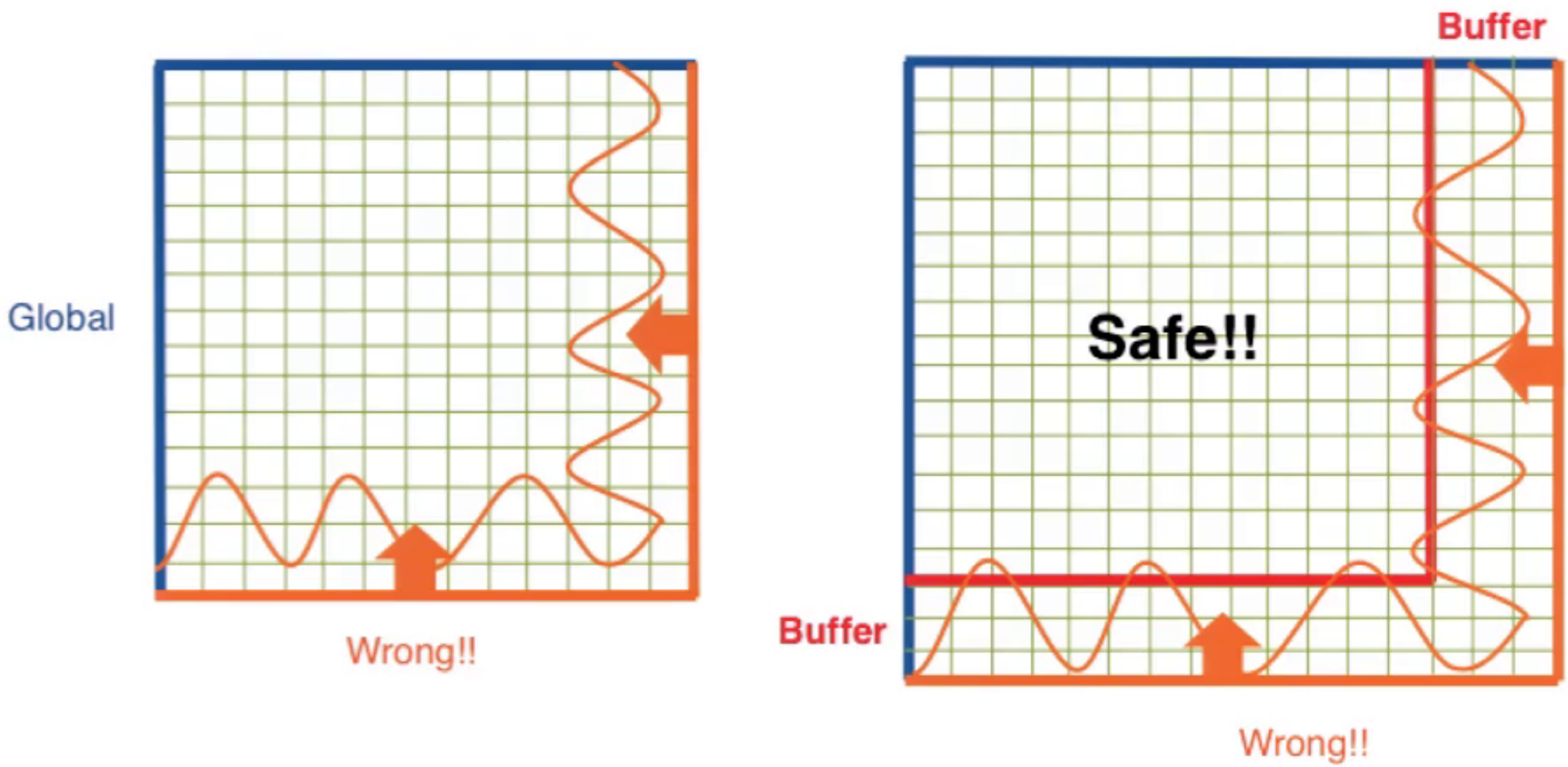


Gappy Simulation

1) L. Grinberg *et. al.* A new domain decomposition method with overlapping patches for ultrascale simulations: Application to biological flows (2010).

Buffer

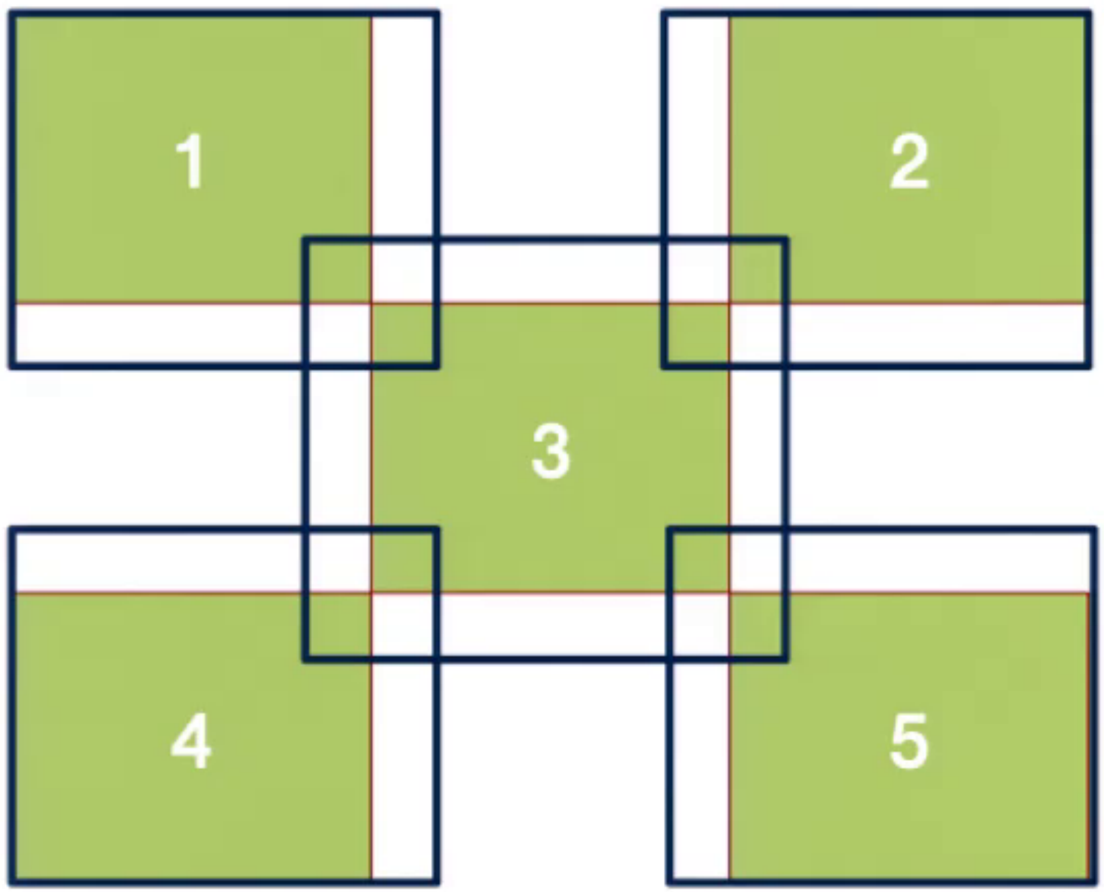
- Avoid penetrating the negative effect from wrong boundary values.



G. Samaey *et. al.* Patch dynamics with buffers for homogenization problems (2006).

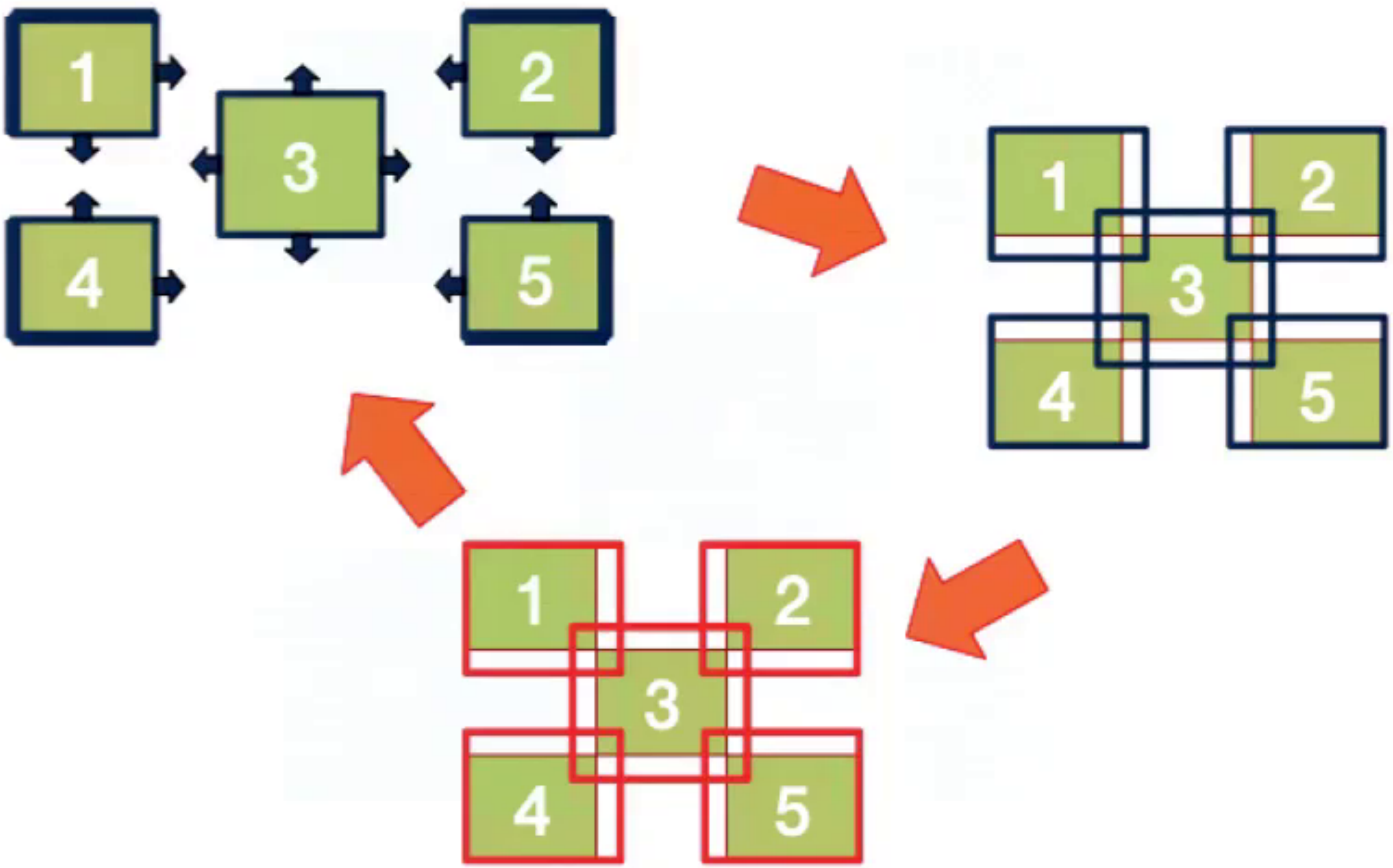
Flow Chart

3. Choose “buffer” size.



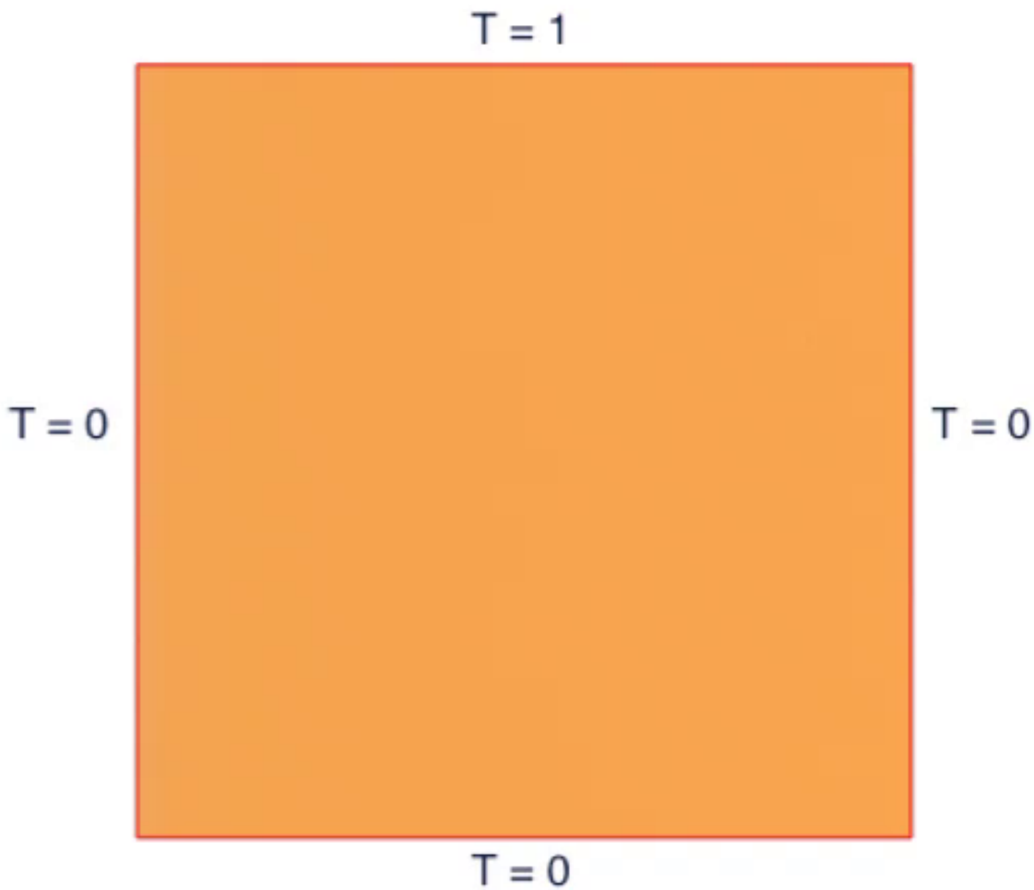
Flow Chart

Repeat step 5-7 until simulation end.



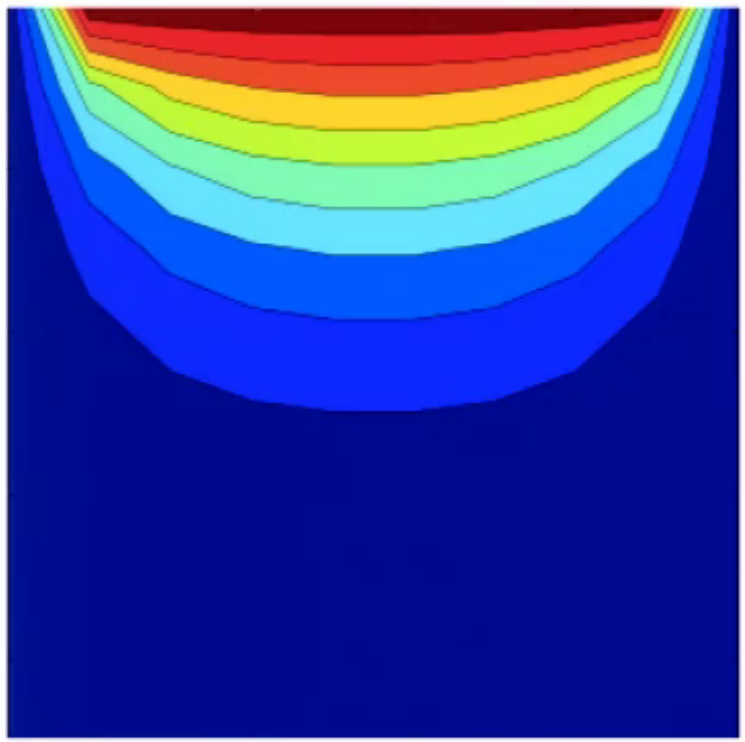
2D Heat Equation

$$\frac{\partial T}{\partial t} = \kappa \Delta T$$



2D Heat Equation

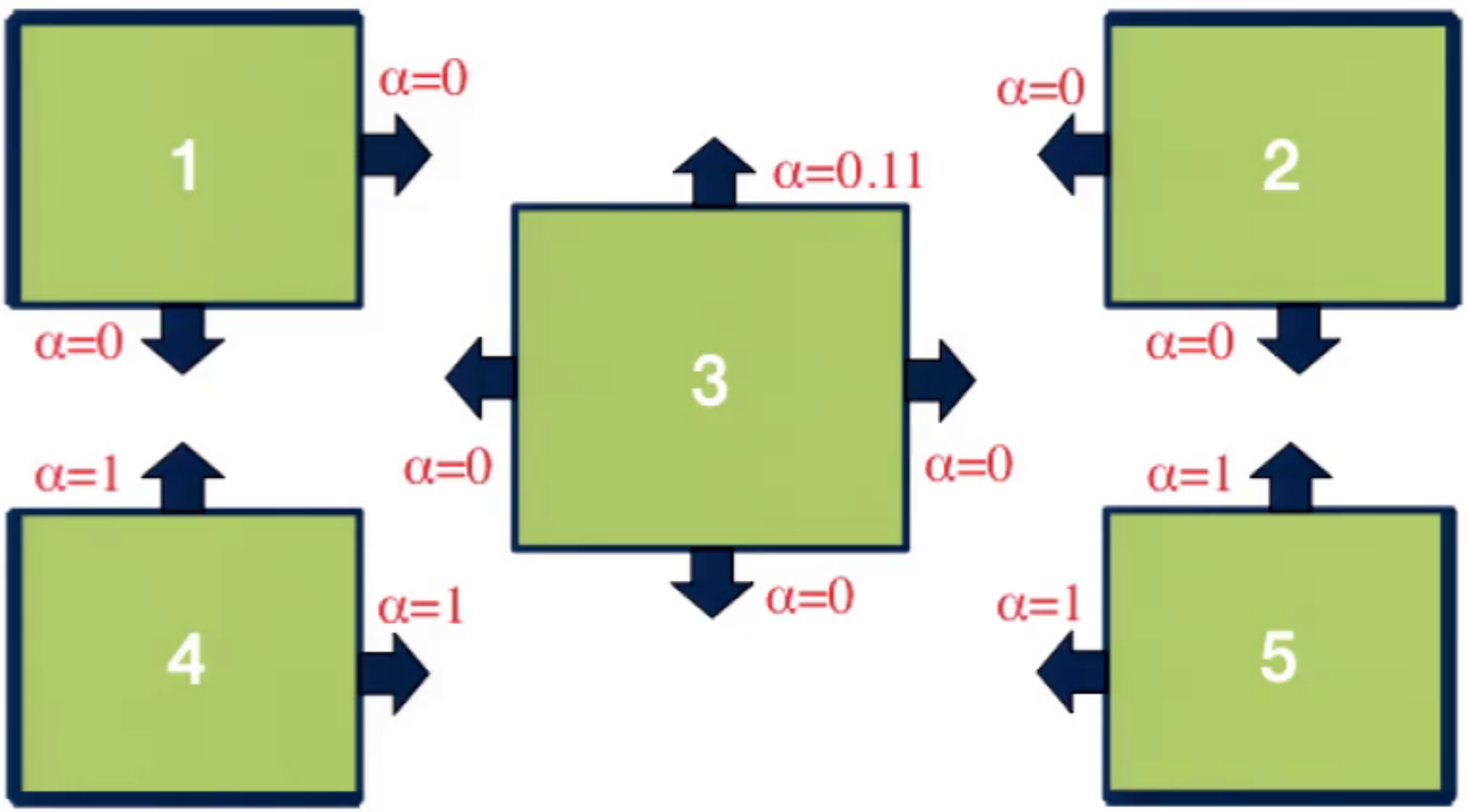
- Partial information comes from a coarse grid.



10 x 10 coarse grid

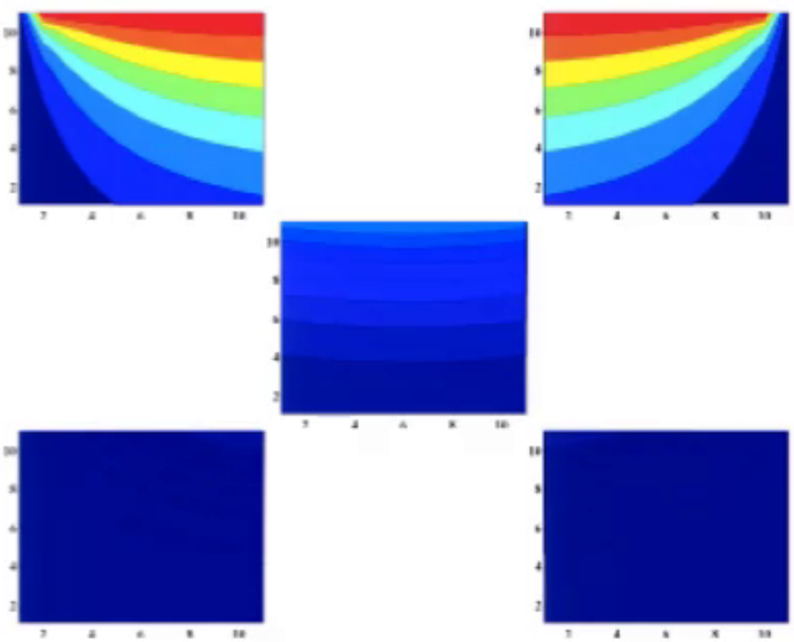
2D Heat Equation

- Boundary condition – Inter-Patch Condition.
- Robin boundary condition - $U_r = \alpha\, u_d + (1-\alpha)\, u_n$.

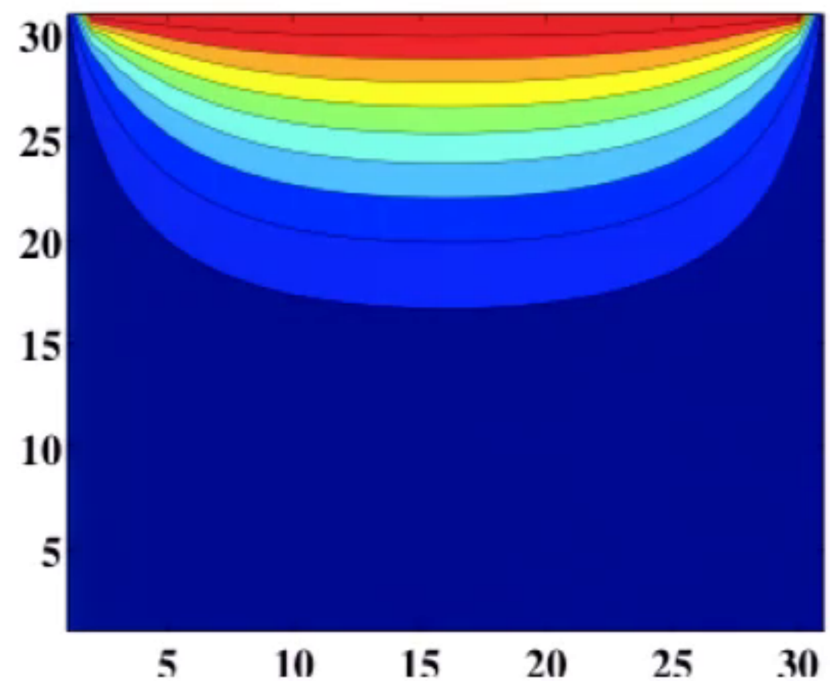


2D Heat Equation

- Partial information comes from a coarse grid.



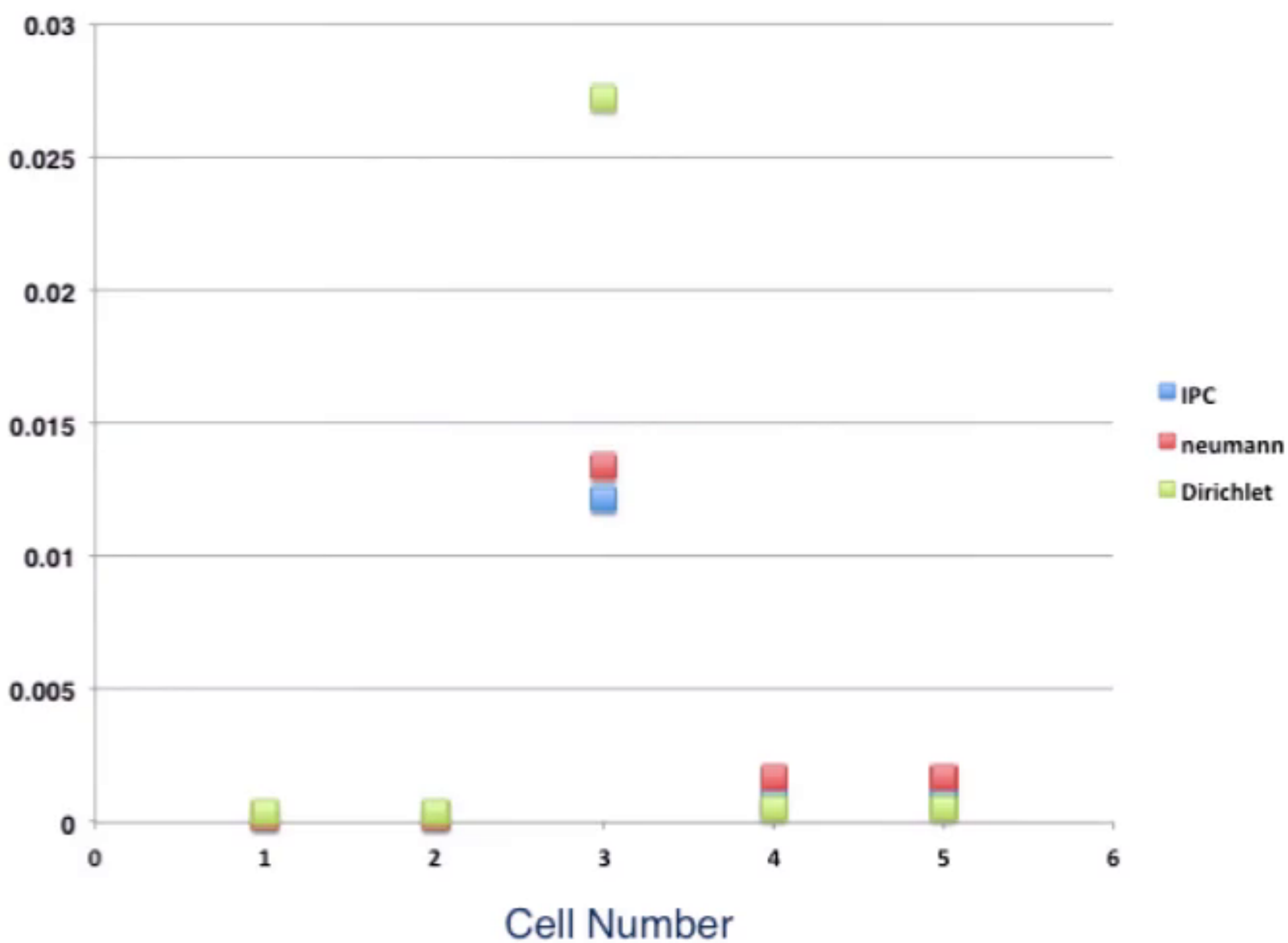
Gappy Simulation

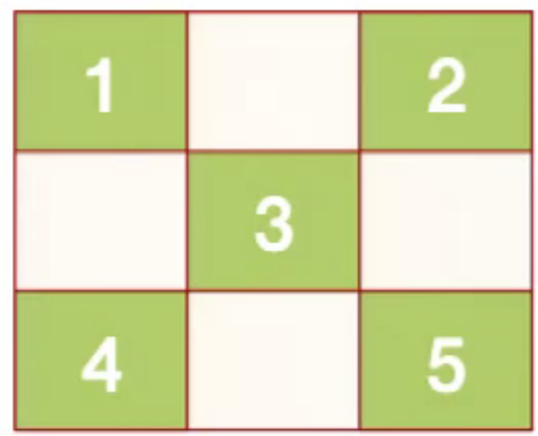


Original solution

2D Heat Equation

- RMS Error for each cells.
- Cell 3 has the biggest error due to absence of global(exact) boundary.



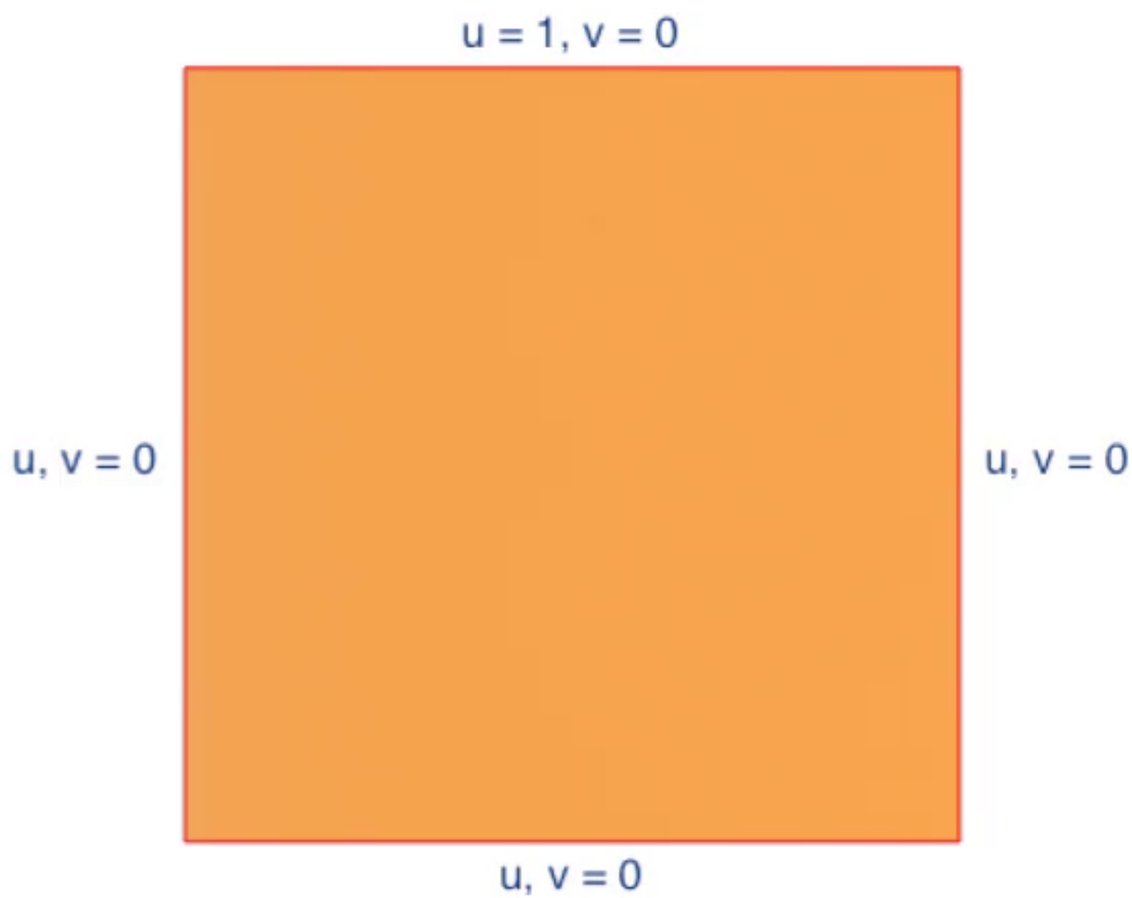


Cell index

2D Lid-Driven Cavity Flow

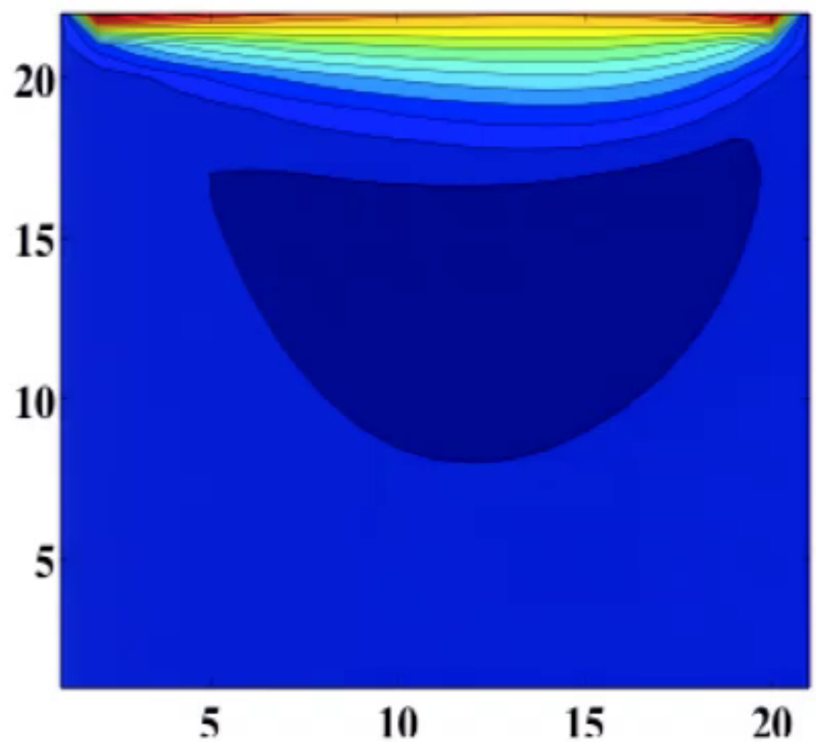
$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla p + \nu \Delta \mathbf{v} + f$$

$$\nabla \cdot \mathbf{v} = 0$$



2D Lid-Driven Cavity Flow

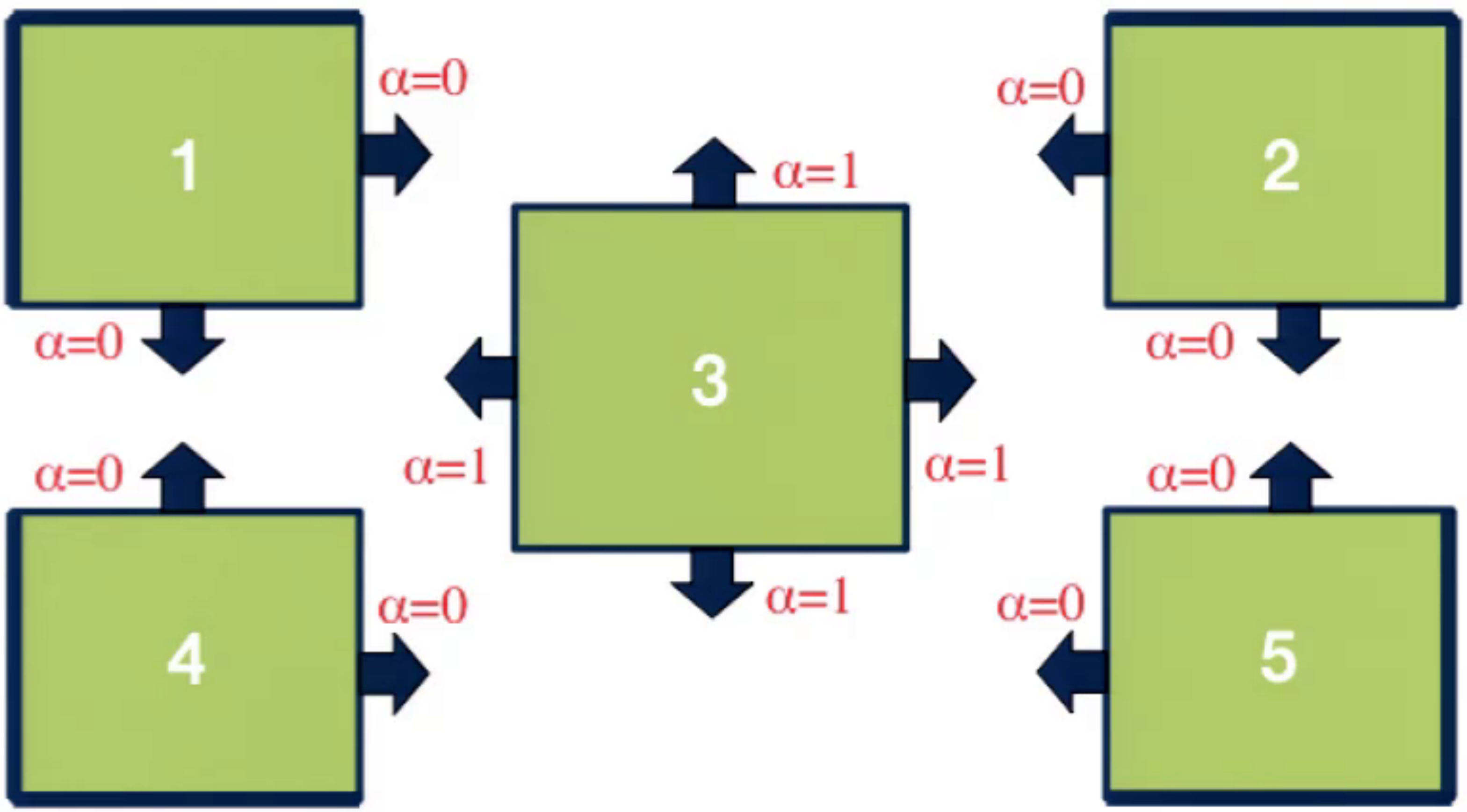
- Partial information comes from a coarse grid.



21 x 22 coarse grid

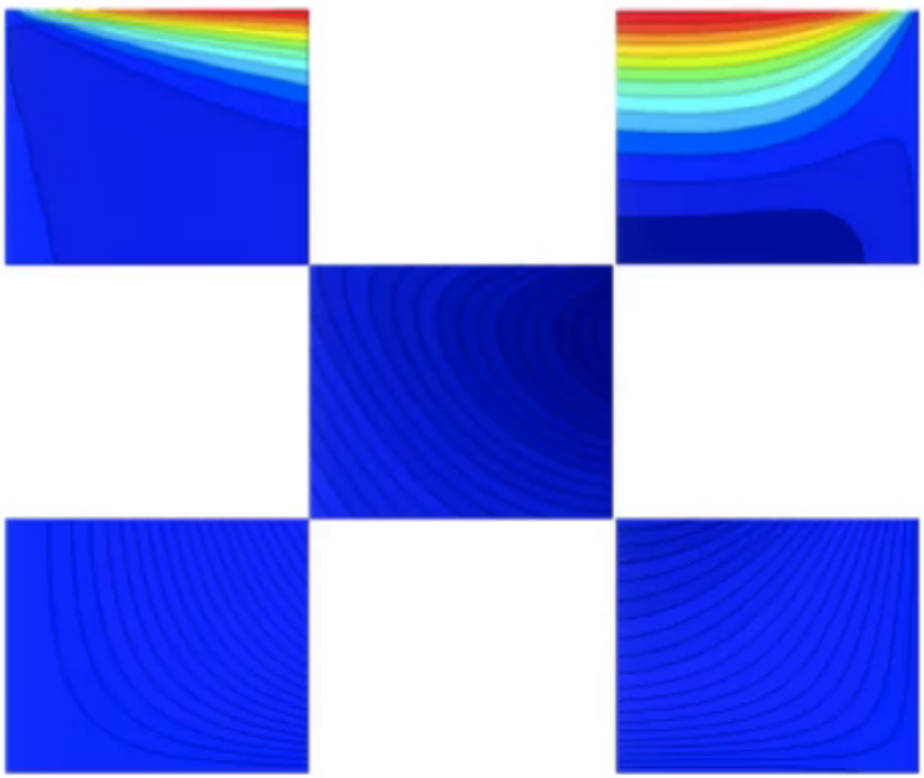
2D Lid-Driven Cavity Flow

- Boundary condition – Inter-Patch Condition.
- Robin boundary condition - $U_r = \alpha\, u_d + (1-\alpha)\, u_n$.

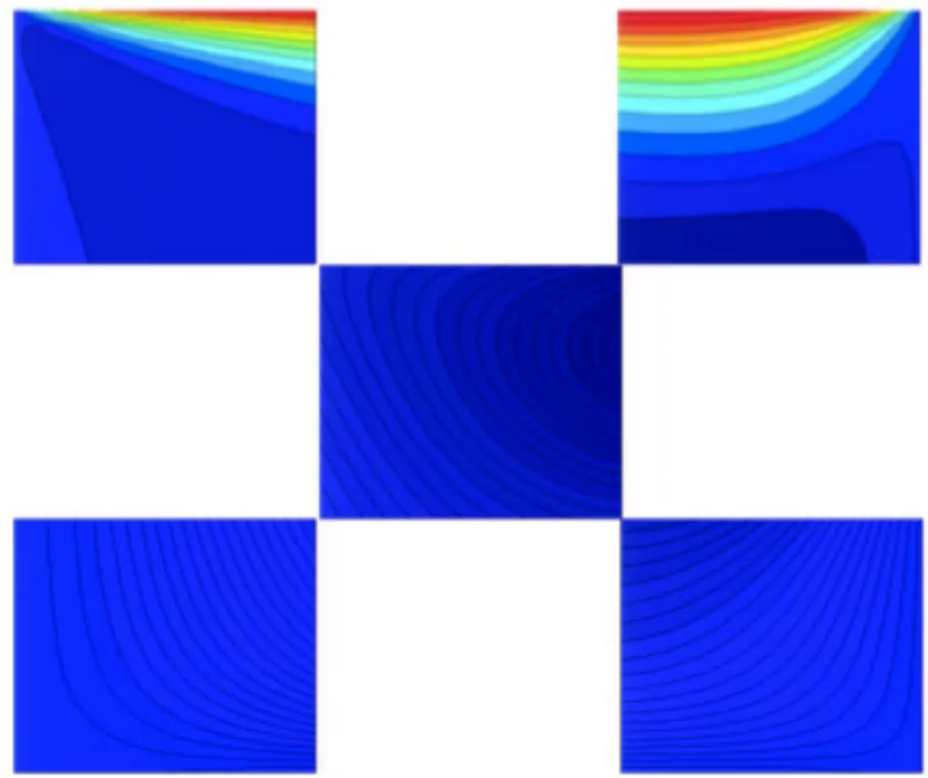


2D Lid-Driven Cavity Flow

- Partial information comes from a coarse grid.



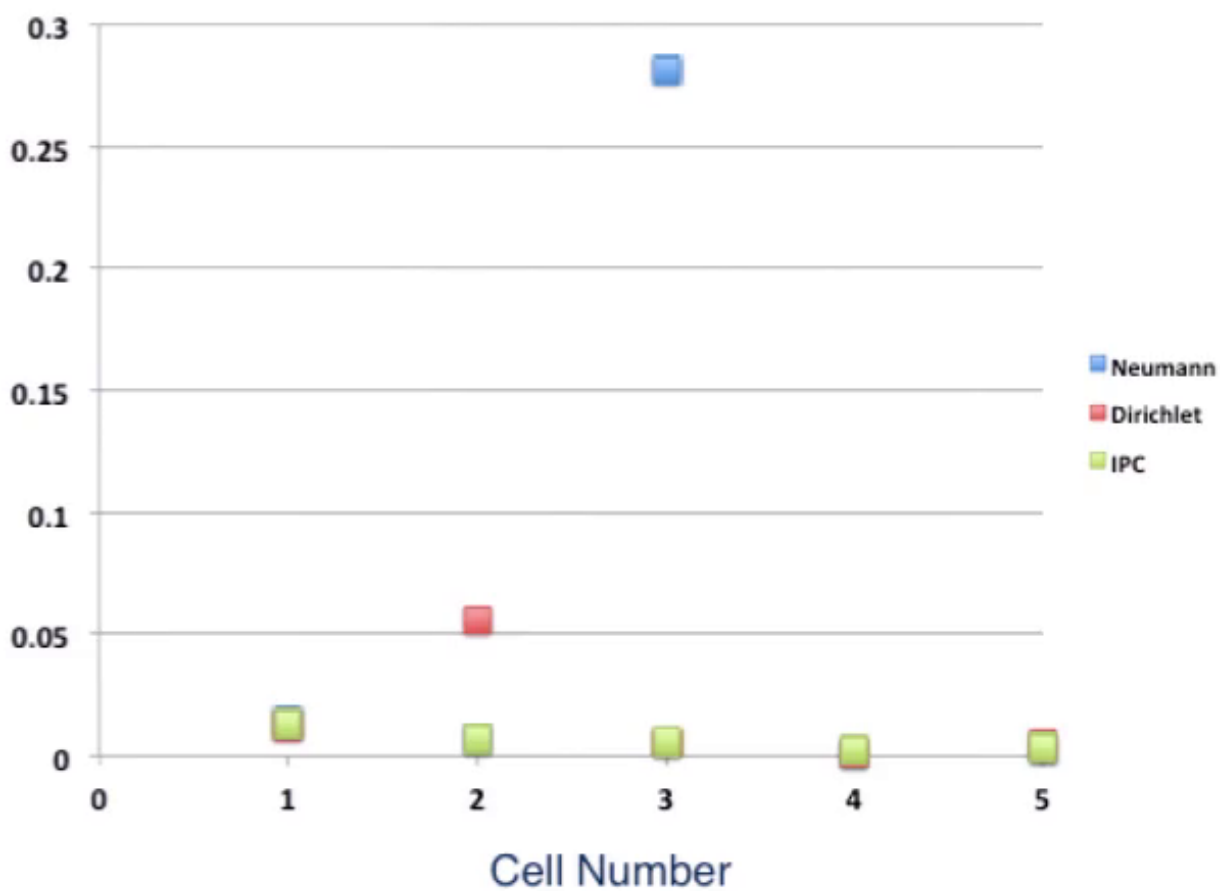
Gappy Simulation

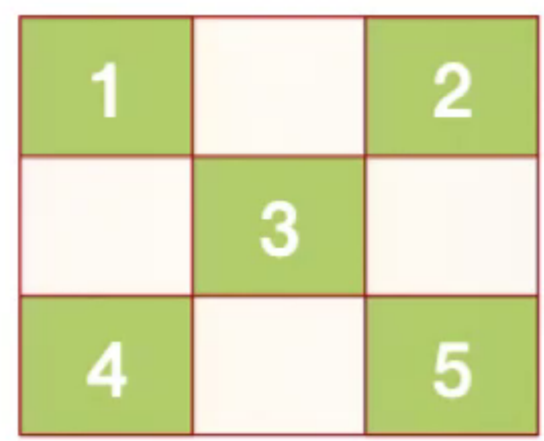


Original solution

2D Lid-Driven Cavity Flow

- RMS Error – Inter-Patch condition is the best but need to find optimal choice.

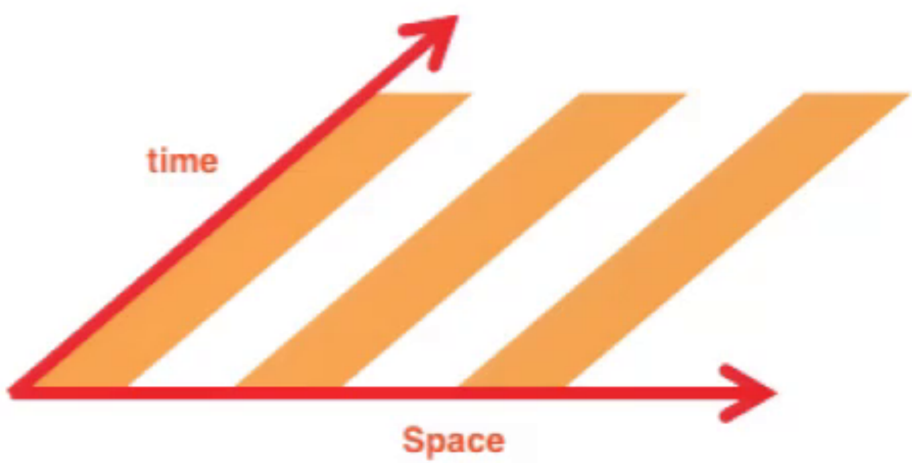




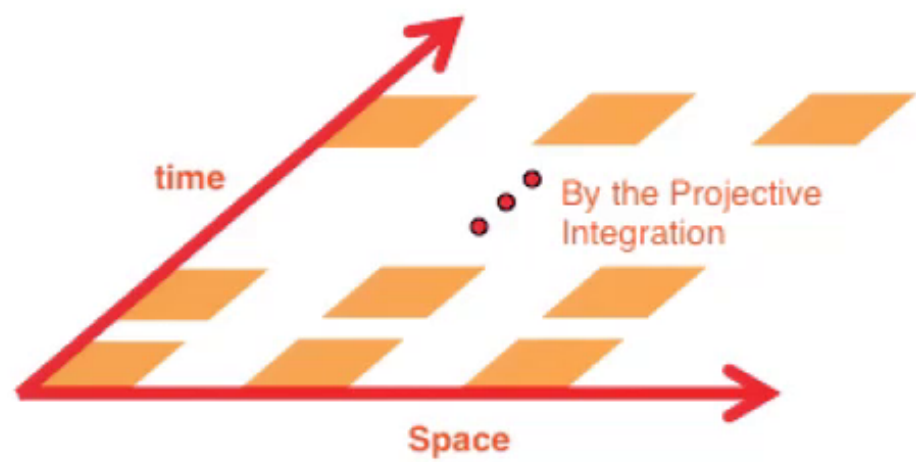
Cell index

A Spatio-temporal Gappy Simulation

- Spatio-gappy simualtion - gaps in space only (gap-tooth algorithm).
- Saptio-temporal gappy simulation –gaps in space and time (patch dynamics).



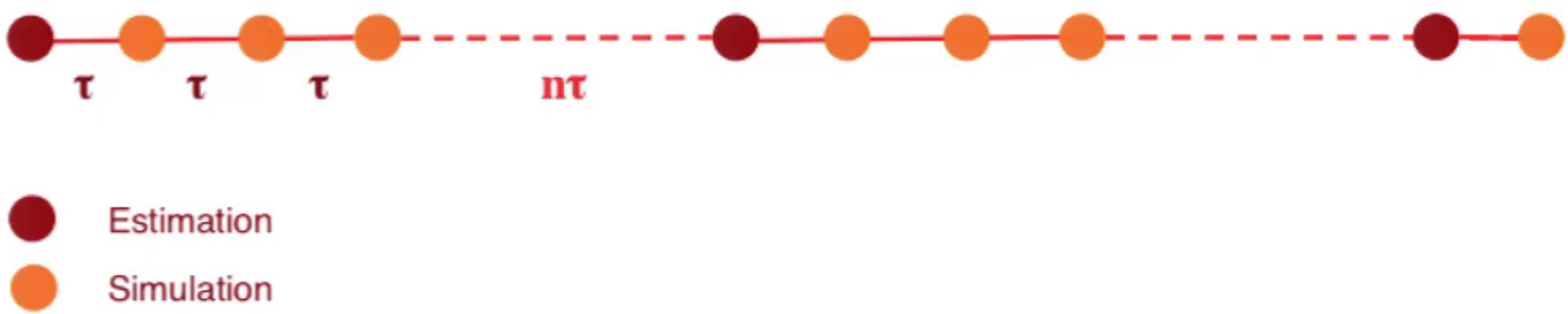
Solution space of spatio-gappy simulation



Solution space of spatio-temporal gappy simulation

Equation-free Projective Integration

- Save a snapshot every τ and after third snapshot is saved, project (estimate) "**nτ**" by POD-based projective integration.



Equation-free Projective integration

- General algorithm for Equation-free projective integration.

$$u(t,x) = \sum_k a_k(t)\phi_k(x).$$

(restriction) $$a(t_i) = \mathcal{P}u(t_i,x) = \{(u(t_i,x),\phi_k(x)), \forall k\} \quad i = 1,\ldots,n.$$

(estimation) $$a(t^*) = \left[\frac{d_i + d_{i+1} - 2\Delta_i}{(t_{i+1}-t_i)^2}\right](t^*-t_i)^3 + \left[\frac{-2d_i - d_{i+1} + 3\Delta_i}{t_{i+1}-t_i}\right](t^*-t_i)^2 + d_i(t^*-t_i) + a(t_i).$$

where $\Delta_i = (a(t_{i+1}) - a(t_i))/(t_{i+1}-t_i)$ and $d_i = a(t_i)'$

(lifting) $$u(t^*,x) = \mathcal{L}a(t^*) = \sum_k a_k(t^*)\phi_k(x).$$

S. Sirisup *et. al.* Equation-free/Galerkin-free POD-assisted computation of incompressible flows (2005).

Summary/Future Work

- Introduced **Efficient** framework of a gappy simulation in PDEs from the gap-tooth/patch dynamics algorithm in ***Dynamical Systems***.
- Introduced a multi-fidelity information fusion - Machine learning based algorithm by coKriging.
- Example of a spatio-gappy simulation – Heat equation, Lid-driven cavity flow.
- Example of a spatio-temporal gappy simulation – Heat equation by the projective integration.

- Expand multi-scale/multi-source information fusion.