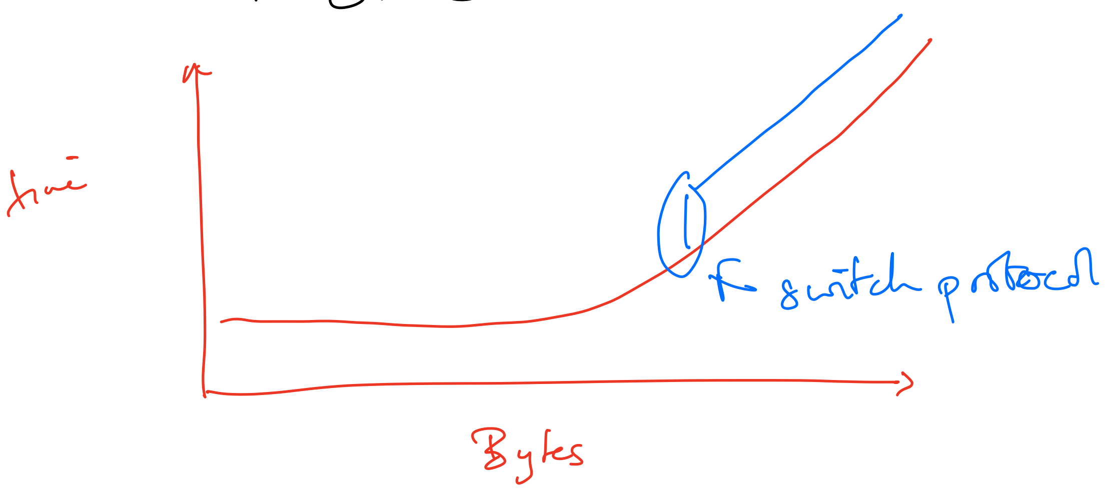
PSC II

Exploring network properties:

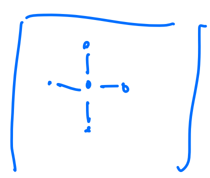
pingpong → code.



Why does parallel computing work at all?

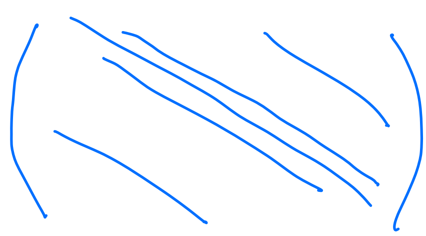
→ Sparsity!

Explicit $\partial_t u - \nabla^2 u = f$



$$u_i^{n+1} = u_i^n + \Delta t \left(f_i + u_i^n \right)$$

banded diagonal.



But equation couples points a long way away ⇒ many steps.

Implicit

– Multigrid components.

"Brings far away pairs close together".

→ Restriction operator $\quad (\ \ldots\ ) := R$

sparse

→ Prolongation operator $\quad (\ \ldots\ ) := R^T$

sparse.

→ Smoother on the levels

→ has same sparsity as an explicit update.

$$u_f^{n+1} = u_f^n + P \, S \, R \, e_f$$

$S e_c$ = just applies

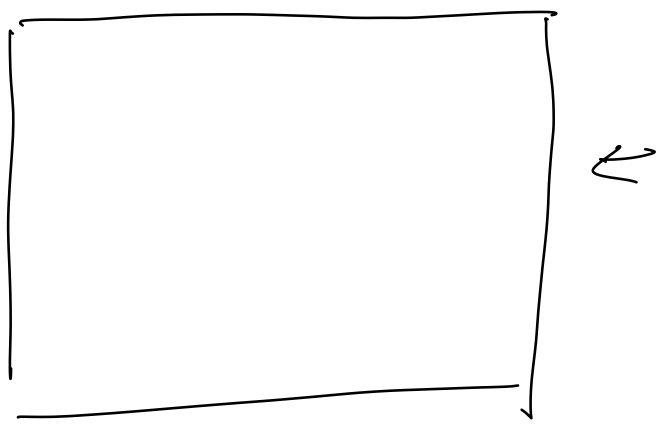
Geometric coarsening means we get a log factor.

✓ Can solve on $O(10^7)$ cores with $O(10^{16})$ dofs.

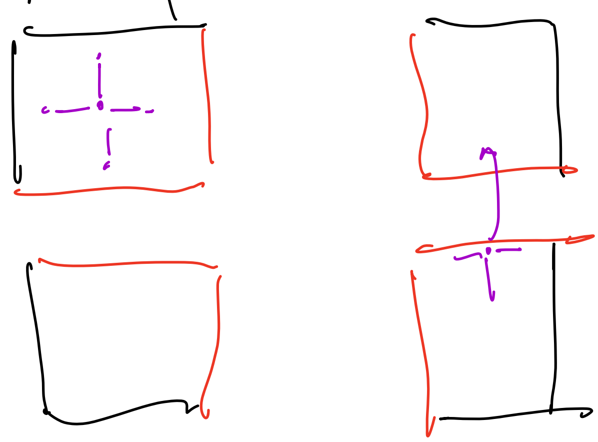
How to exploit this in our datastructures?

FD grids: only local overlap.

$\Longrightarrow$ Domain decomposition



Chop into N pieces

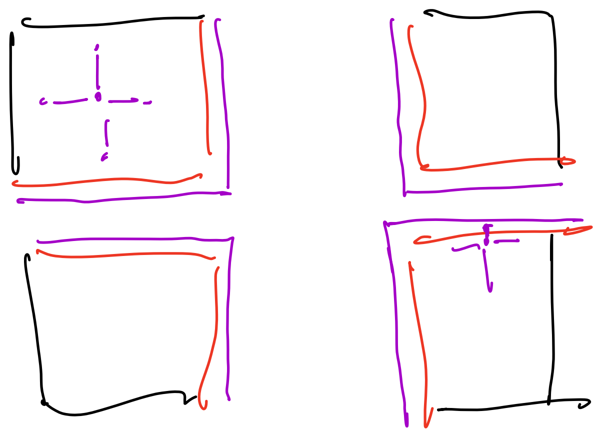


Could send/receive message for each remote part.

Suppose we need n such messages

$\Rightarrow$ model says $T_m = n(\alpha + \beta 1)$

Instead bulk synchronous
→ add a "halo" or "ghost" region
to our local domain.



Communication cost is now

$$T_{m}^{halo} = \alpha + n\beta$$

For $n \gg 1$ $\quad n\alpha > \alpha$.

All processes collectively
decide to do update.

→ communication round to
fill in gaps.

→ local compute.

Next week: some time on

scaling limits for PDE solvers

Scaling Limits for PDE-Based Simulation

⇒ will link from website.