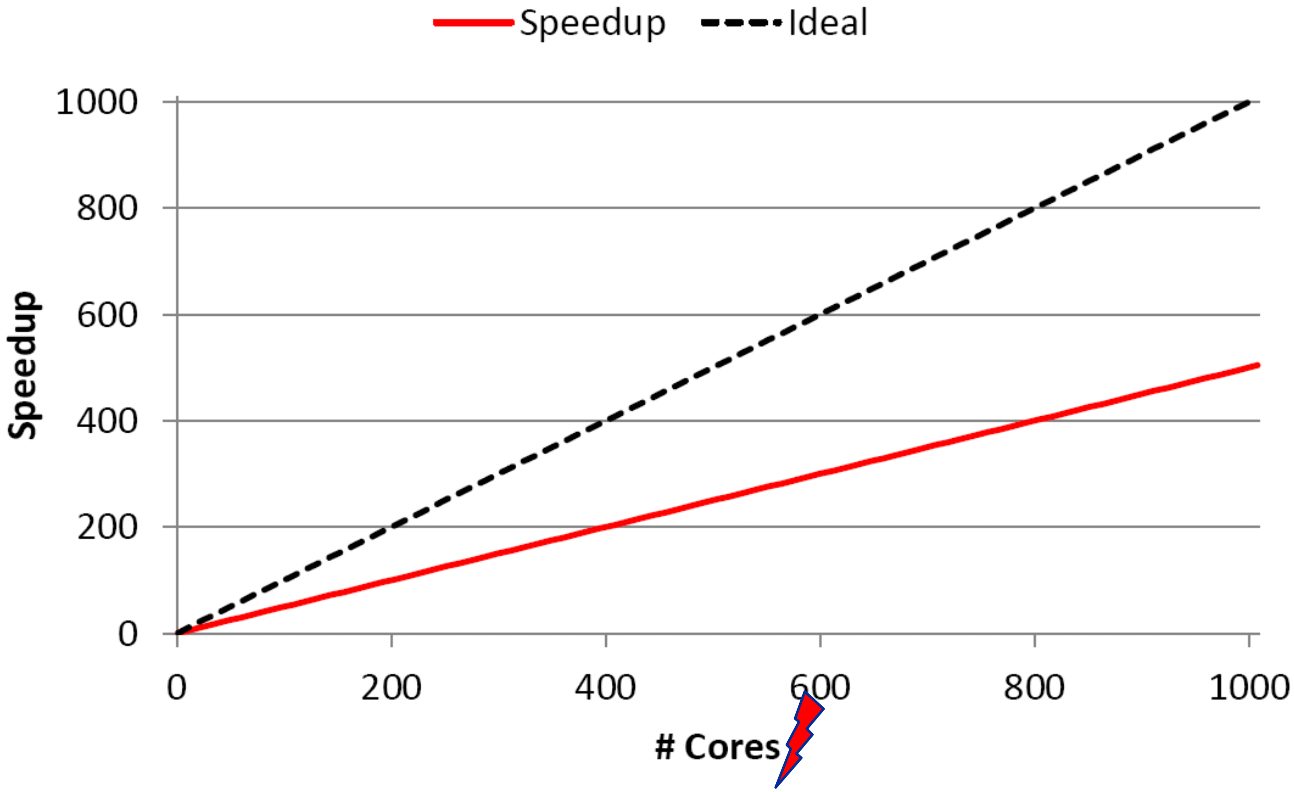
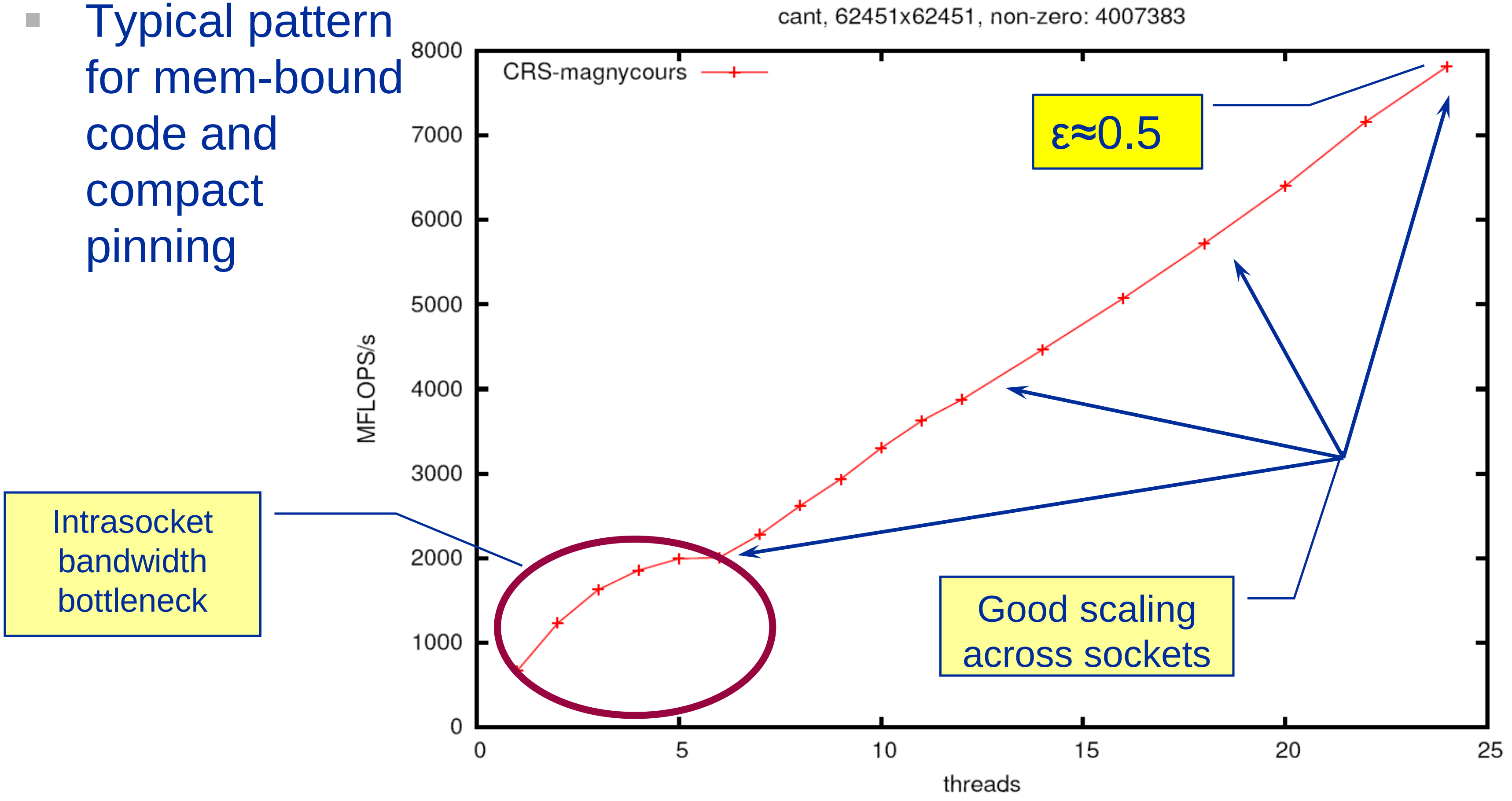
Assignment 11 – Task 1

- Natural question: What is the scaling behavior inside one multicore socket?
- Plotting speedup vs. cores is not useful in most cases since the intra-socket scaling may be limited by other factors than the inter-node scaling
- Think "scaling baseline"!



Scaling baselines

- Example: sparse matrix-vector multiply on a 4-socket AMD system (6 cores/socket)
- Typical pattern for mem-bound code and compact pinning



Assignment 11 – Task 2

- IBM BlueGene scalability for a parallel application
- Assumptions
  - One BG processor is μ times slower than a standard processor. Execution time for serial code is $\mu(s+p)=\mu$
  - Simple communication model: overhead = kN
  - We compare a BG machine with a system having standard processors and the same network characteristics. On this machine, execution time for serial code is $s+p=1$
  - Speedup for BG (assuming strong scaling w/ communication):

$$S_{BG}(N)=\frac{\mu}{\mu(s+(1-s)/N)+kN}=\frac{1}{s+(1-s)/N+kN/\mu}$$

  - Whereas for the standard computer we get

$$S_{std}(N)=\frac{1}{s+(1-s)/N+kN}=S_{BG}(N)\Big|_{\mu=1}$$

Slow Computing

- Number of processors N_s for which speedup is at maximum:

$$\frac{\partial}{\partial N} S_{BG}(N) = 0 \;\Rightarrow\; N_{s,BG} = \sqrt{\frac{\mu(1-s)}{k}}$$

- Speedup at N_s:

$$S_{BG}\left(N_{s,BG}\right) = \frac{1}{s + 2\sqrt{\frac{k(1-s)}{\mu}}}$$

- Special case for standard processor: μ=1

$$S_{std}\left(N_{s,std}\right) = \frac{1}{s + 2\sqrt{k(1-s)}}$$

μ>1 means higher max speedup, i.e. “better scalability”

- Comparison of max performance BG vs. standard:

$$\frac{P_{BG}\left(N_{s,BG}\right)}{P_{std}\left(N_{s,std}\right)} = \frac{S_{BG}\left(N_{s,BG}\right)}{\mu \cdot S_{std}\left(N_{s,std}\right)} = \frac{s+2\sqrt{k(1-s)}}{\mu s+s\sqrt{\mu k(1-s)}}$$

This is always ≤1 and goes to 0 as μ gets large

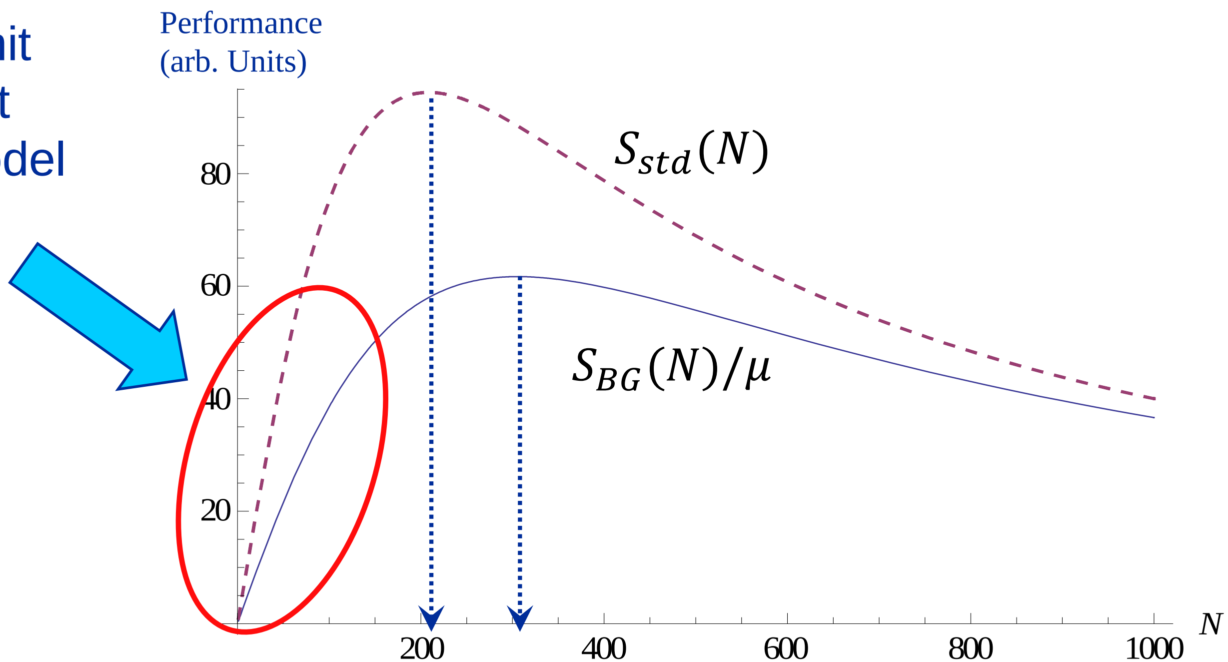
Slow Computing

- How much more "iron" do we need to get max performance on BG than on the standard machine?

$$\frac{N_{s,BG}}{N_{s,std}} = \sqrt{\mu}$$

Independent of *k* and *s* !

Look at small-N limit (*s* negligible): What communication model will make BG win?



Slow Computing: What is a favorable communication model for BG?

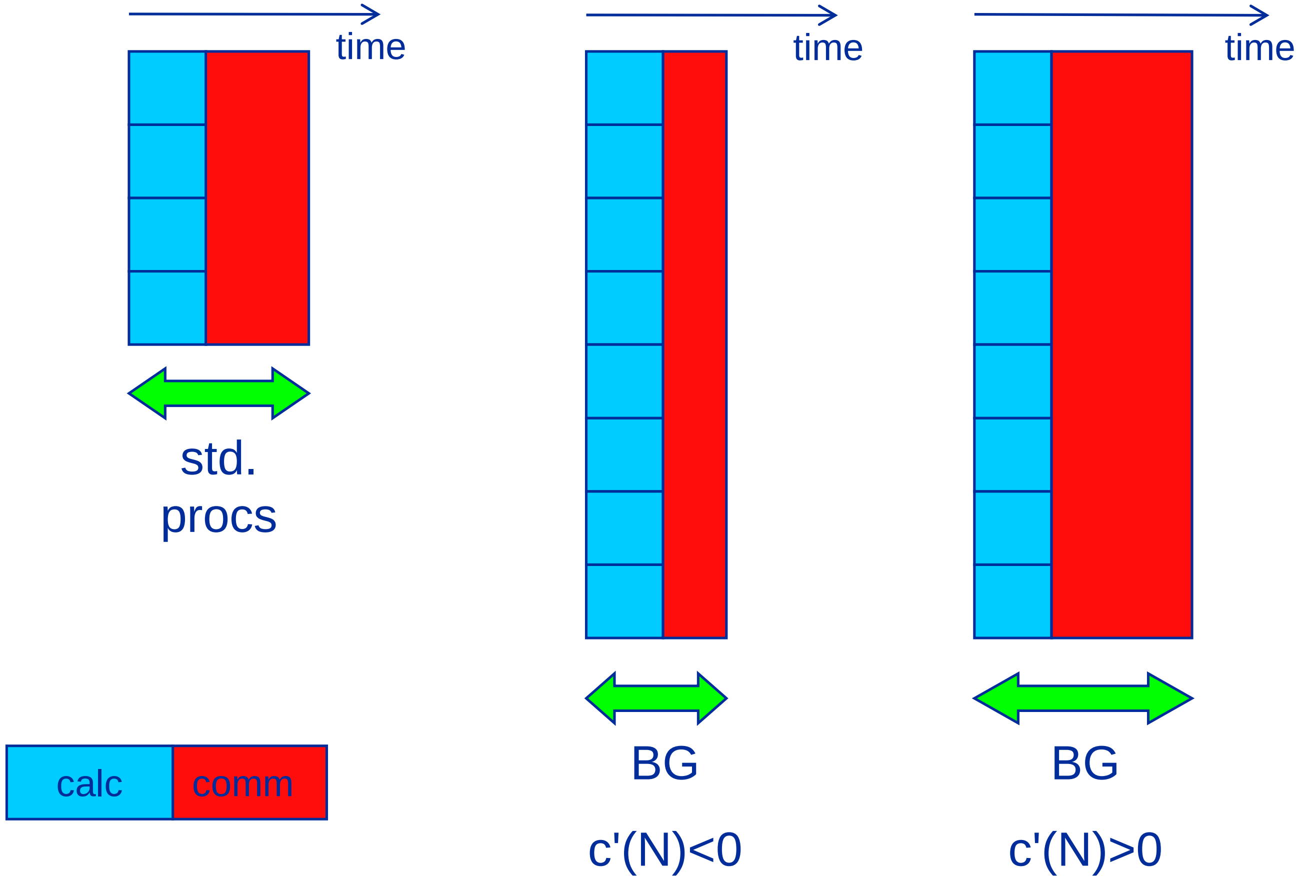
- Look at execution times for N std. procs versus μN BG procs, and N not too large:

$$\frac{T_{std}(N)}{T_{BG}(\mu N)} = \frac{s + p/N + c(N)}{\mu(s + p/(\mu N))} \xrightarrow{s \ll p/N} \frac{p/N + c(N)}{p/N + c(\mu N)}$$

- This is >1 only if $c(\mu N) < c(N)$, i.e., if communication overhead goes down with N.
  - Is there a plausible explanation for this? → see next slide
- Example: d-dimensional domain decomposition with halo exchange, non-blocking network, no overlap between communication and computation: $c(N) = \lambda + kN^{(1-d)/d}$
  - 3D: $c(N) = \lambda + kN^{-2/3}$ → check!
  - 2D: $c(N) = \lambda + kN^{-1/2}$ → check!
  - 1D: $c(N) = \lambda + k$ → fail!

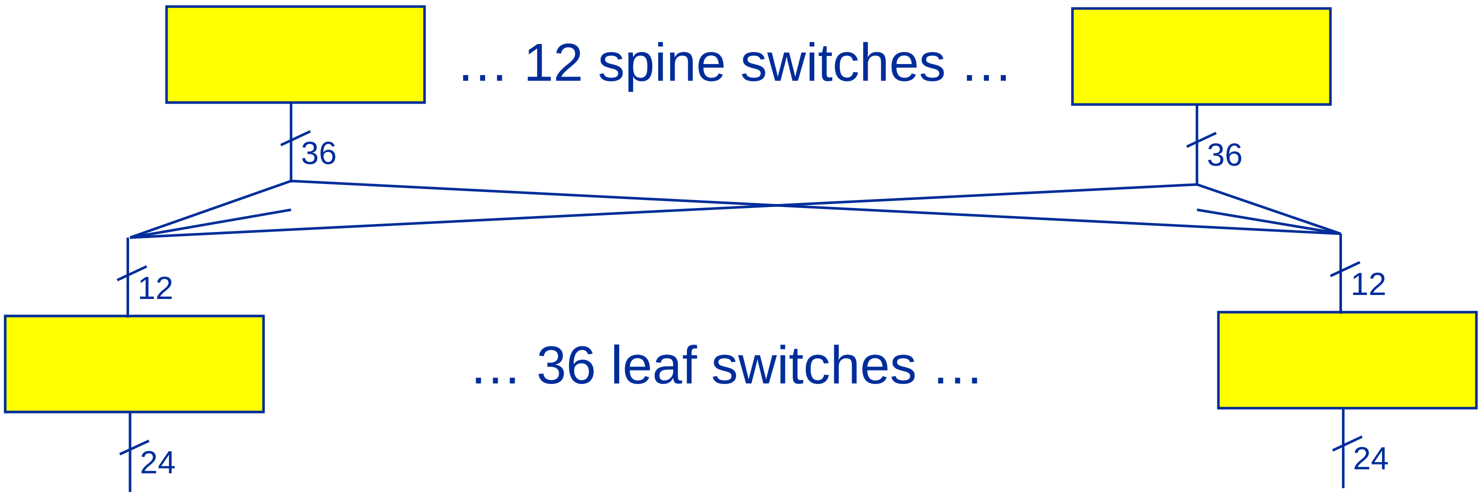
Slow Computing: What is a favorable communication model for BG?

Example: μ=2, s=0, strong scaling



Assignment 11 – Task 3
QDR InfiniBand 2:1 oversubscribed fat tree

- QDR IB has 36-port switch elements
  → 2:1 oversubscription means every leaf switch uses 24 ports to connect to nodes and the remaining 12 ports to connect into the spine
- Every leaf switch needs at least one wire to every spine switch
  → 12 spine switches
- Every spine switch has 36 connections, one per leaf switch
  → 36 leaf switches and 24*36=864 node ports



Assignment 11 – Task 3
QDR InfiniBand 3:2 oversubscribed fat tree

→ 3:2 oversubscription means unequal distribution to spine level due to static routing.

- Example:
  - 20 Port QDR-IB Switch: 3:2 oversubscription
  - 12 Nodes connected to spine by 8 uplinks.

  → 8 of the nodes will have to share 4 uplinks which effectively cuts the bandwidth in half for them