Otto-von-Guericke-Universität Magdeburg
Universitätsrechenzentrum

Programmierung von 5832 Cores

Wie programmiert man auf der SC5832?

- natürlich parallel
- verteilter Speicher erfordert Message Passing Interface (MPI)
- Multi Threading Programme nun nutzlos
- keine automatische Parallelisierung (PGCC fehlt der Welt noch)
- Programmierer / Forscher müssen sich selbst um Datenfluss kümmern oder auf fertige parallele Bibliotheken zurückgreifen (passt nicht immer)

Das klingt erstmal nicht so erfreulich ...

Aber wieviel bringt mir der eventuelle Mehraufwand?

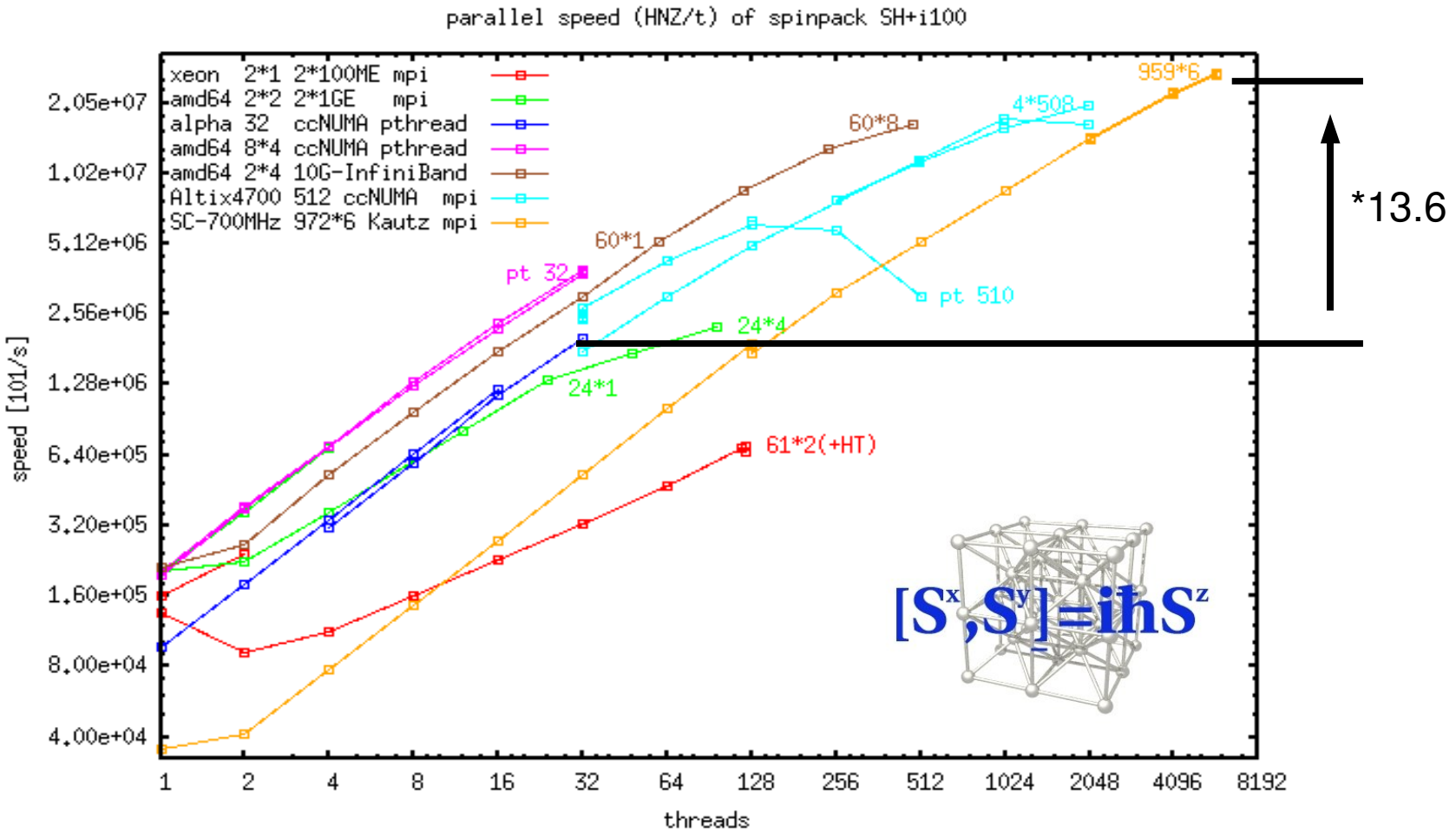
Erwartung Verbesserung:

	HP GS1280	--->	SC5832
Peak:	74 GFLOP	*110	8.2 TFLOP
RAM:	128 GB	*30	3.9 TB
MPI-BW:	4 GB/s	*75	300 GB/s

Erwartung Verbesserung:

	HP GS1280	--->	SC5832
Remote-Mem:	32/250ns	*3	5832/15us
Disk:	500 MB/s	*2	1 GB/s
1/pOverhead:	1/32	*1/180	1/5832

Reale Verbesserung:



oft: realer Speedup < erwarteter Speedup

	HP GS1280	--->	SC5832
Peak:	74 GFLOP	*110	8.2 TFLOP
MPI-BW:	4 GB/s	*75	300 GB/s
Spinpack:		*13.6	

+ größere Modelle, nur doppelter Stromverbrauch, gleicher Preis = OK

trotzdem:

Warum nicht 75 mal schneller? Das wäre doch besser!

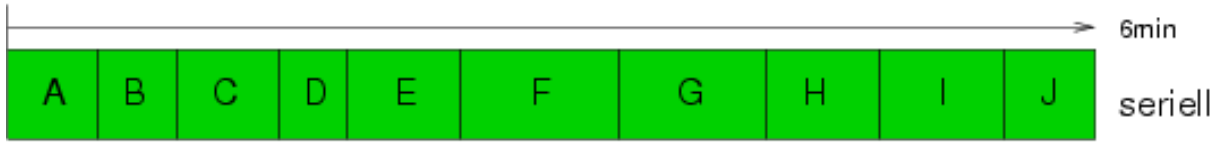
Warum nicht 75 mal schneller? Das wäre doch besser!

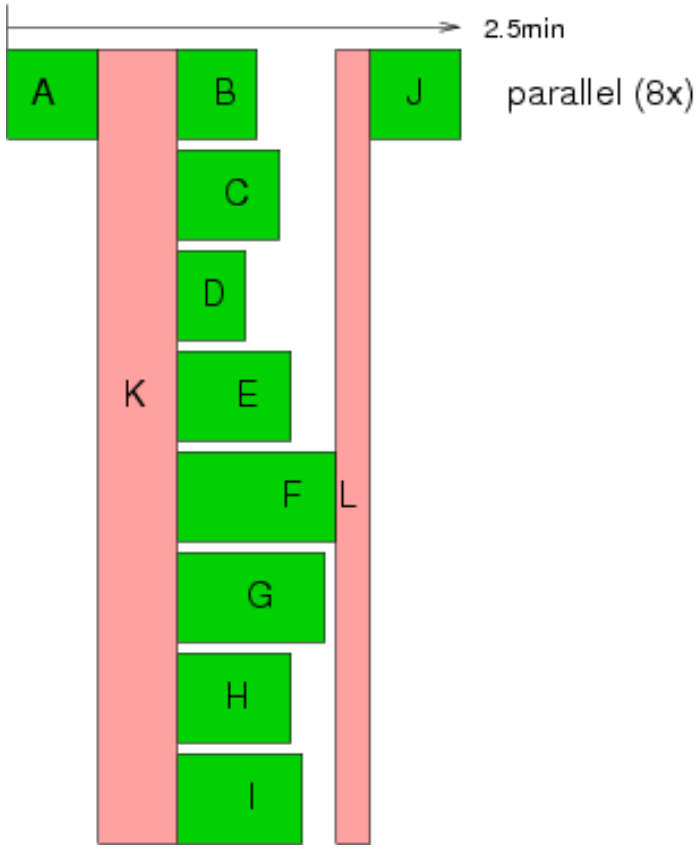
- Parallelisierung nun sehr wichtig!
- nicht alle Algorithmen lassen sich einfach parallelisieren
- Parallelisierung ist aufwendig (kostet Zeit und Nerven)
- Parallelisierungscode erhöht Programmkomplexität
- komplexere Fehler und Fehlersuche (Mehraufwand Debugging)
- Tuning verschlechtert meist Lesbarkeit des Codes (kostet Z+N)
- (Parallel-)Programmierer oder Erfahrung fehlen

-> deshalb realer Code nie perfekt angepasst, sondern Kompromiss mit weiterem Optimierungspotential

- zum Trost: CPU Tuning ist unwichtiger!

ein Beispiel: ...

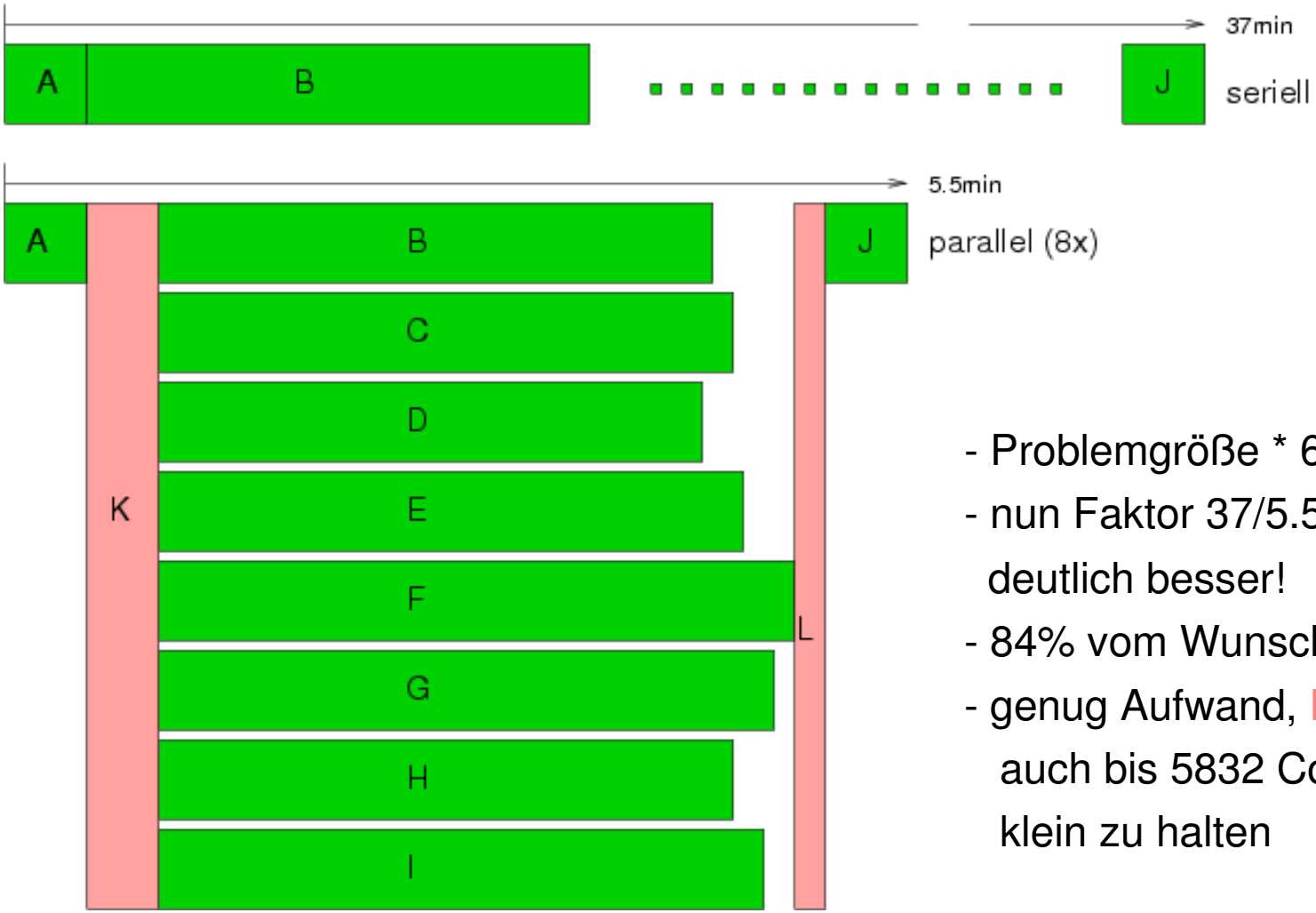




Ein Beispiel:

- A,J: sequentieller Code
- B...I: Ungleichverteilung
- K,L: Datentransfer + Mehraufwand

- Erwartung Faktor 8 :-)
- real nur Faktor 6/2.5 = **2.4** schneller :-(
- einfachste Lösung: größere Probleme
  mit (B,C,D,..,I) >> (A,J,K,L)



- Problemgröße * 6
- nun Faktor 37/5.5 = **6.7**
  deutlich besser!
- 84% vom Wunschfaktor 8
- genug Aufwand, K+L
  auch bis 5832 Cores
  klein zu halten

Zusammenfassung:

- Abschied vom SMP (nicht ganz: Meggie = 256 GB + 32 Cores)
- mehr Möglichkeiten und höherere Herausforderungen mit der SC5832
- 5832 Cores brauchen große Modelle

- gewohnte Linuxumgebung + MPI + Tuningtools + parallele Bibliotheken
- Tuning-Unterstützung auch aus dem URZ
- Austausch HPC-Erfahrungen mit anderen Unis

Die ersten Nutzer sind ja schon da und ...
die SC5832 läuft schon ...
... und läuft ... und läuft ... und läuft ...

... natürlich schneller ...