The New Costs of Physical Memory Fragmentation

DIMES '24

Alexander Halbuer[1], Illia Ostapyshyn[1], Lukas Steiner[2], Lars Wrenger[1], Matthias Jung[3], Christian Dietrich[4], Daniel Lohmann[1]

[1]Leibniz Universität Hannover
[2]Rheinland-Pfälzische Technische Universität Kaiserslautern-Landau
[3]Universität Würzburg and Fraunhofer IESE
[4]Technische Universität Braunschweig

November 03, 2024

Physical Memory

- Motivation
    - One third of TCO and power consumption
    - Average utilization is at 75%

Physical Memory

- Motivation
  - One third of TCO and power consumption
  - Average utilization is at 75%
- Beliefs
  - Only used memory is good memory (cache ‘em all)
  - All frames are equal (no external fragmentation)

Physical Memory

- Motivation
  - One third of TCO and power consumption
  - Average utilization is at 75%
- Beliefs
  - Only used memory is good memory (cache ‘em all)
  - All frames are equal (no external fragmentation)
- This is no longer the reality!
  - Huge/giant pages
  - Redistribute between VMs/via CXL
  - or power down unused memory

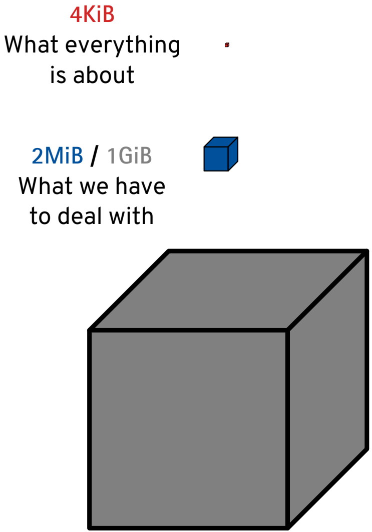
Physical Memory

4KiB
What everything
is about

- Motivation
  - One third of TCO and power consumption
  - Average utilization is at 75%
- Beliefs
  - Only used memory is good memory (cache 'em all)
  - All frames are equal (no external fragmentation)
- This is no longer the reality!
  - Huge/giant pages
  - Redistribute between VMs/via CXL
  - or power down unused memory

Physical Memory

- Motivation
  - One third of TCO and power consumption
  - Average utilization is at 75%
- Beliefs
  - Only used memory is good memory (cache 'em all)
  - All frames are equal (no external fragmentation)
- This is no longer the reality!
  - Huge/giant pages
  - Redistribute between VMs/via CXL
  - or power down unused memory



4KiB
What everything is about

2MiB / 1GiB
What we have to deal with

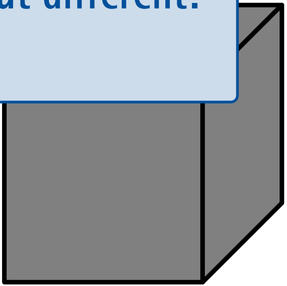
Physical Memory

- Motivation
  - One third of TCO and power consumption
  - Average utilization is at 75%
- B
- This is no longer the reality!
  - Huge/giant pages
  - Redistribute between VMs/via CXL
  - or power down unused memory

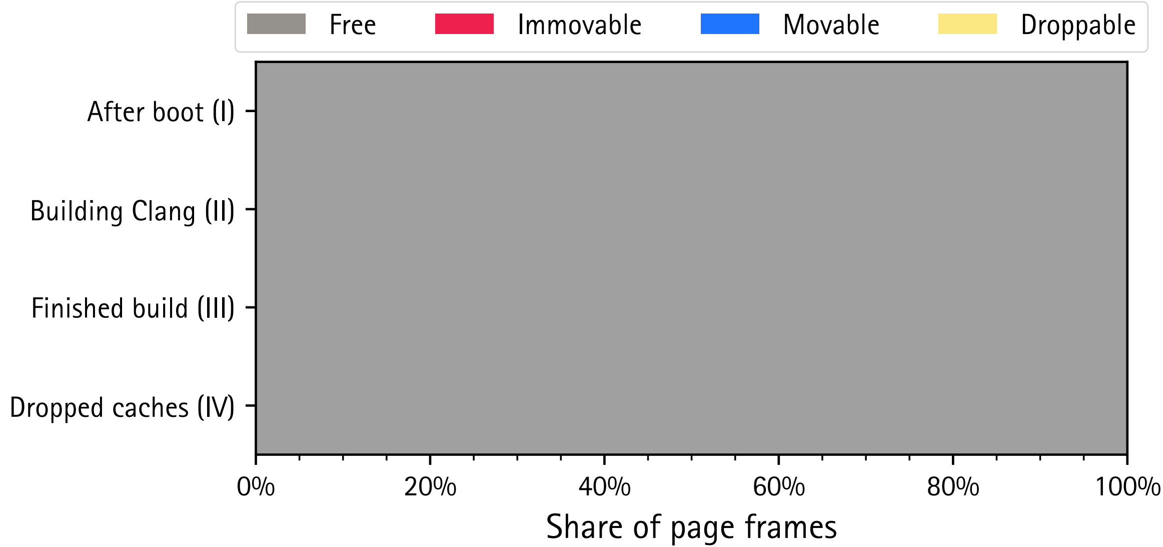
⇒ The new fragmentation problem is real but different:
We must think bigger!

4KiB
What everything
is about

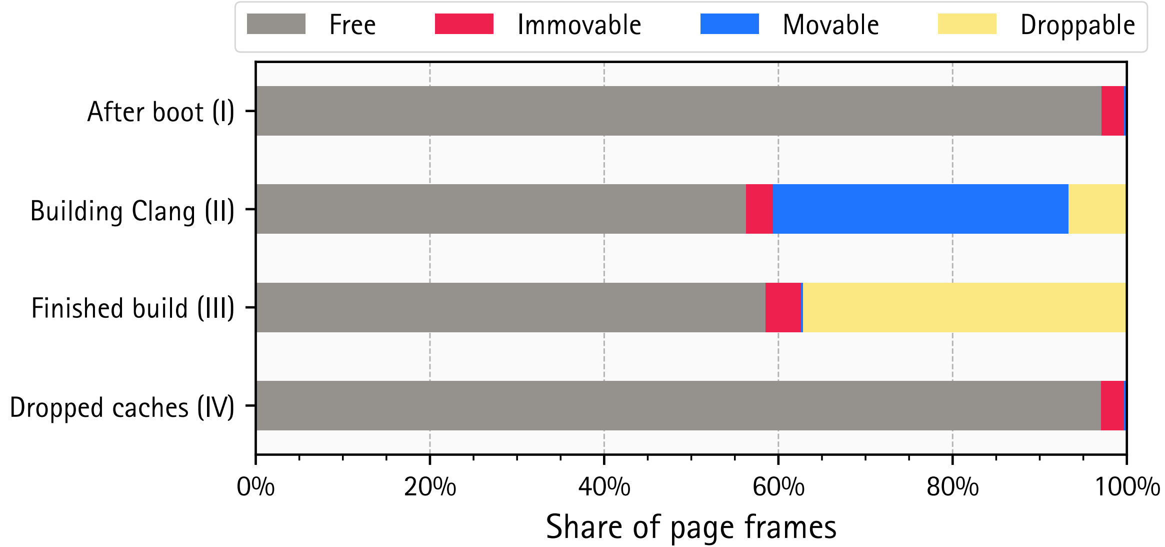
2MiB / 1GiB
What we have



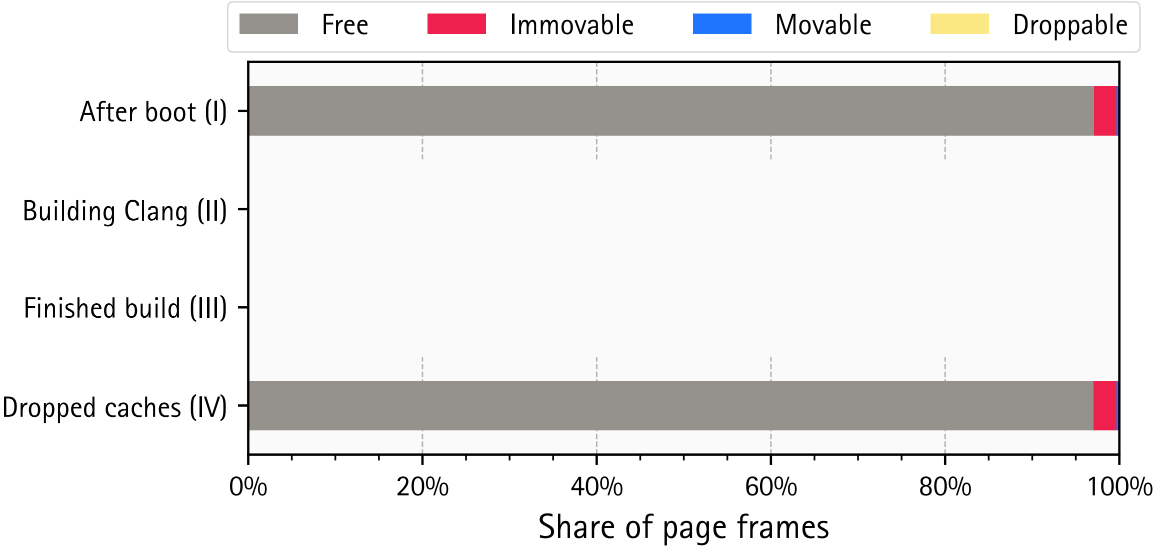
Case Study: Building Clang, 12 CPUs, 16 GiB



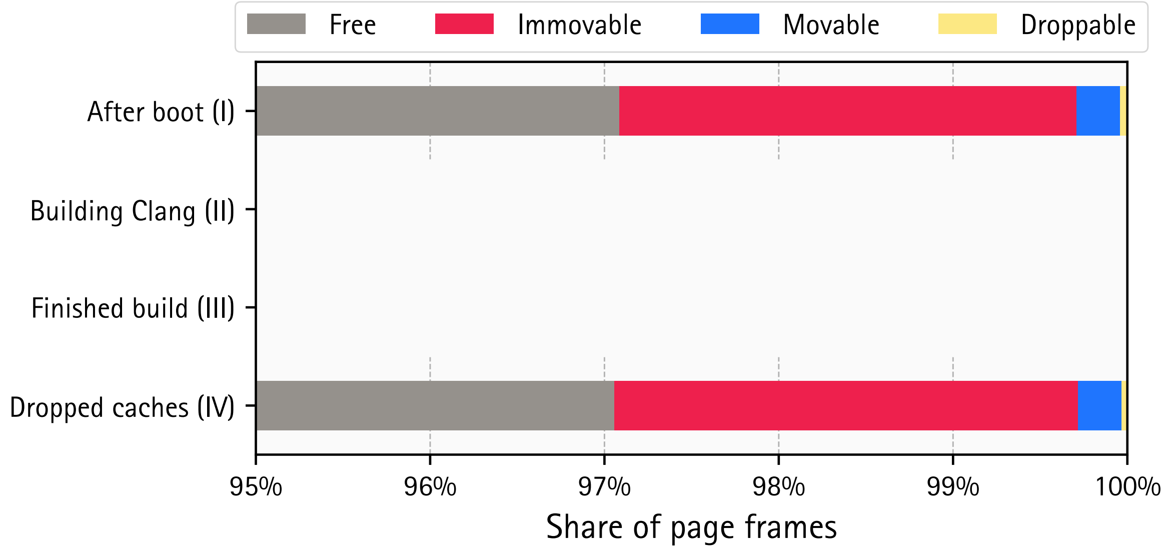
Case Study: Building Clang, 12 CPUs, 16 GiB



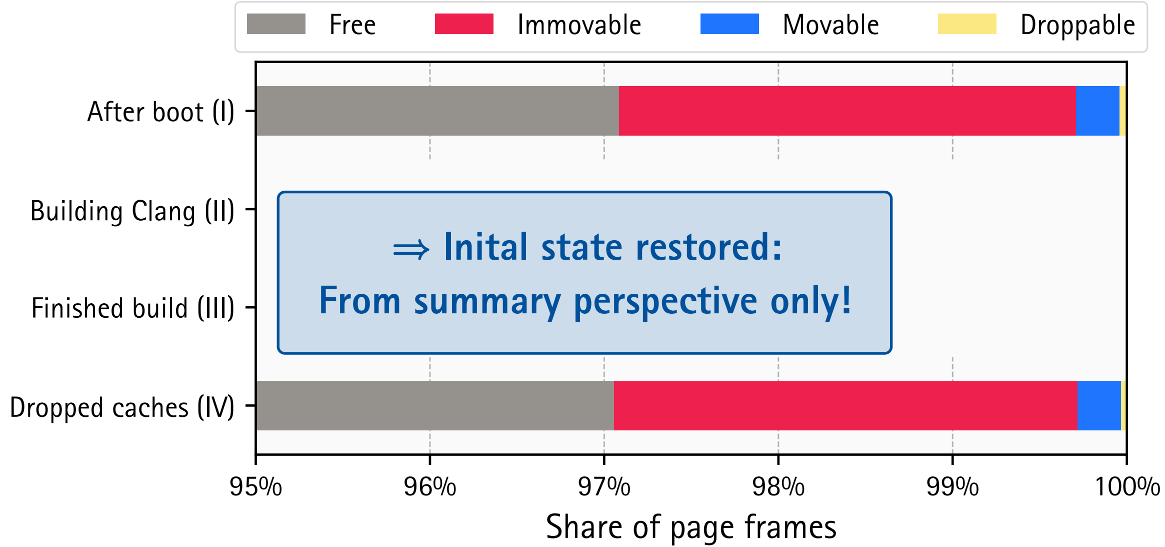
Case Study: Building Clang, 12 CPUs, 16 GiB



Case Study: Building Clang, 12 CPUs, 16 GiB



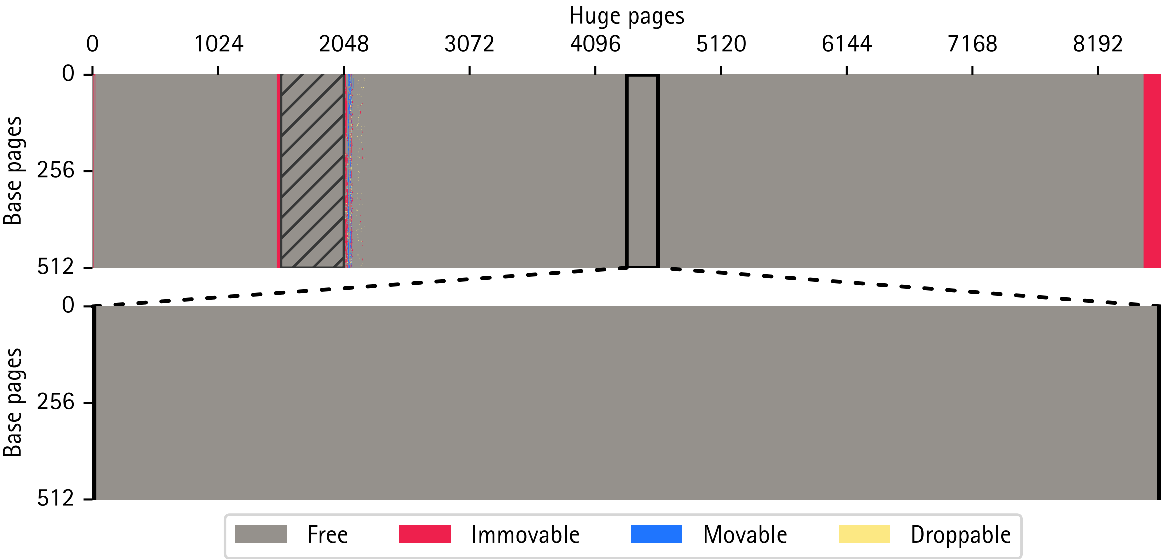
Case Study: Building Clang, 12 CPUs, 16 GiB



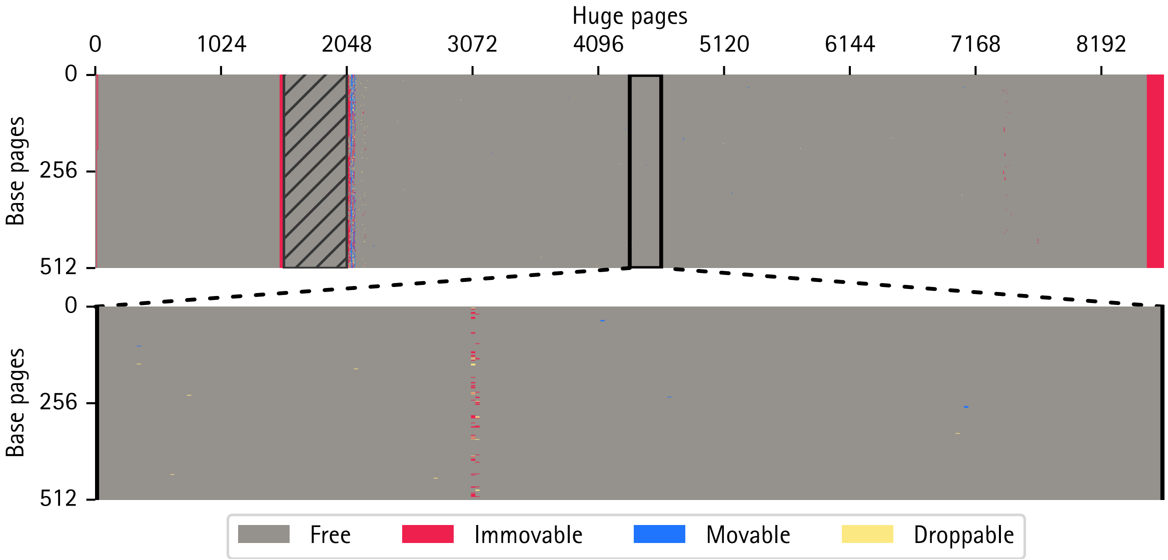
⇒ Inital state restored:

From summary perspective only!

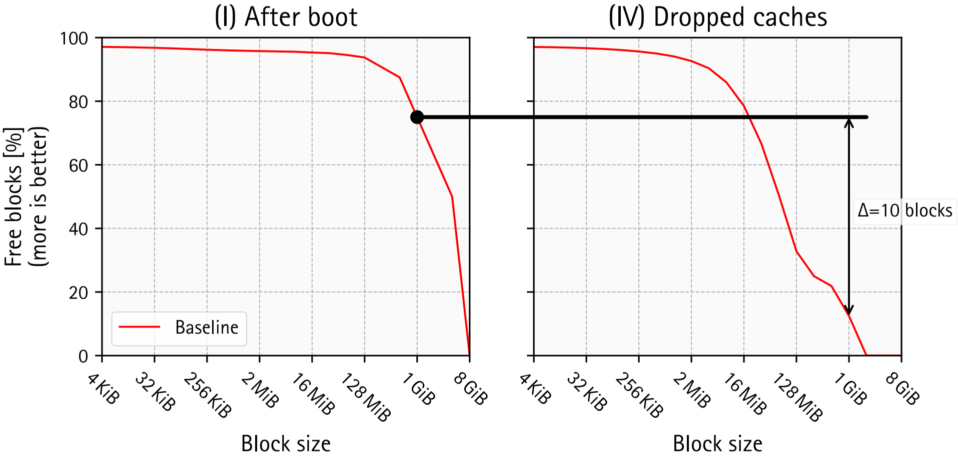
Case Study: Insights – (I) After boot



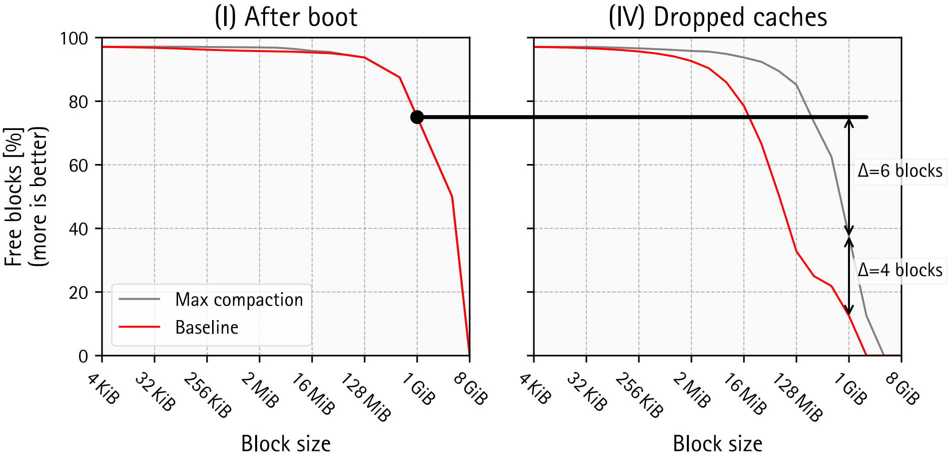
Case Study: Insights - (IV) Dropped caches



Case Study: Analysis

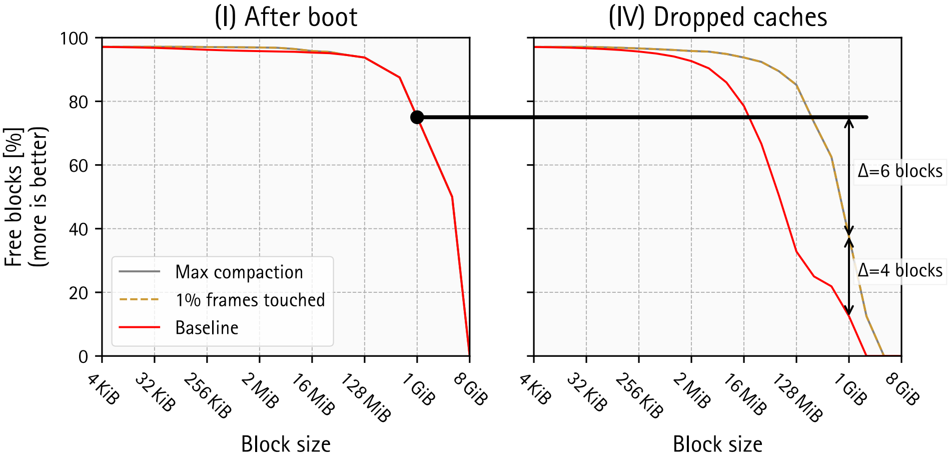


Case Study: Looking for a Solution



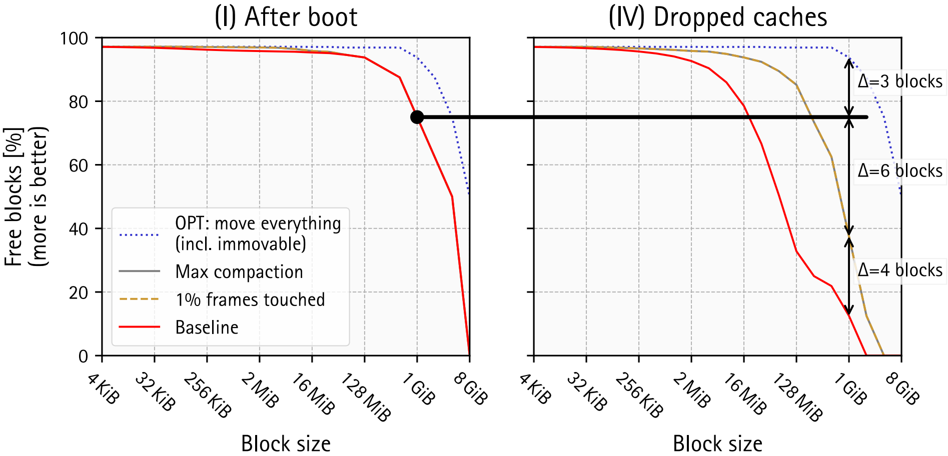
Case Study: Looking for a Solution

Cost function: $\#\mathrm{TOUCHes} = \#\mathrm{DROPs} + 2 \times \#\mathrm{MOVEs}$



Case Study: Looking for a Solution

Cost function: $\#\mathrm{TOUCHes} = \#\mathrm{DROPs} + 2 \times \#\mathrm{MOVEs}$

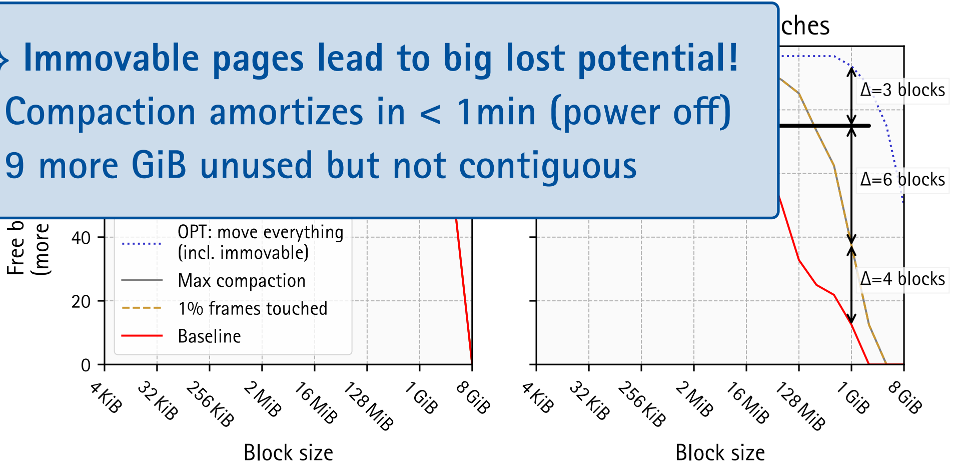


Case Study: Looking for a Solution

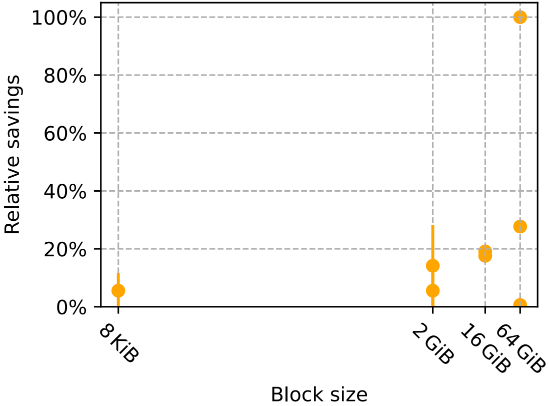
Cost function: $\#\text{TOUCHes} = \#\text{DROPs} + 2 \times \#\text{MOVEs}$

⇒ **Immovable pages lead to big lost potential!**

- Compaction amortizes in < 1min (power off)
- 9 more GiB unused but not contiguous



DRAM Power Saving



Example system

512 GiB DDR5

8 channels

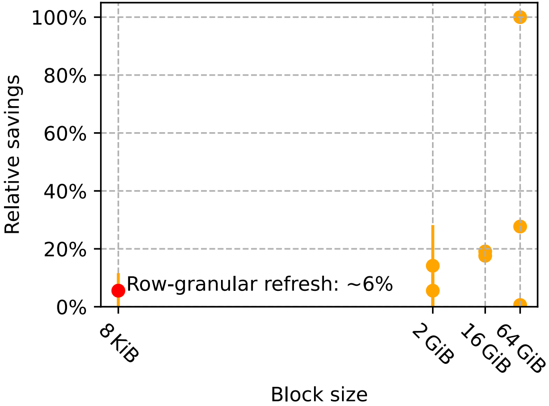
4 ranks per channels

8×2 GiB chips per rank

32 banks per rank

65 536 rows per bank

DRAM Power Saving



Example system

512 GiB DDR5

8 channels

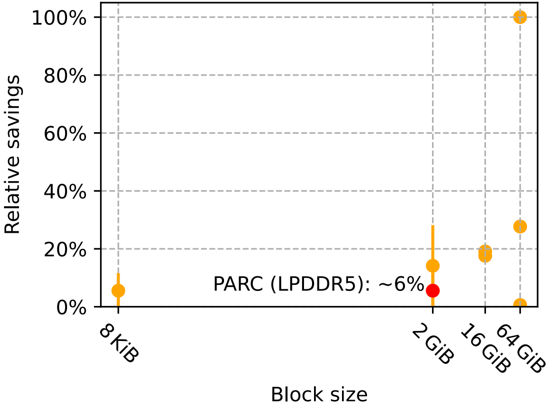
4 ranks per channels

8×2 GiB chips per rank

32 banks per rank

65 536 rows per bank

DRAM Power Saving



Example system

512 GiB DDR5

8 channels

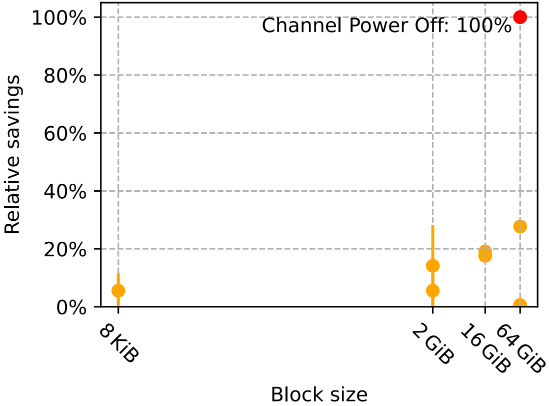
4 ranks per channels

8×2 GiB chips per rank

32 banks per rank

65 536 rows per bank

DRAM Power Saving



Example system

512 GiB DDR5

8 channels

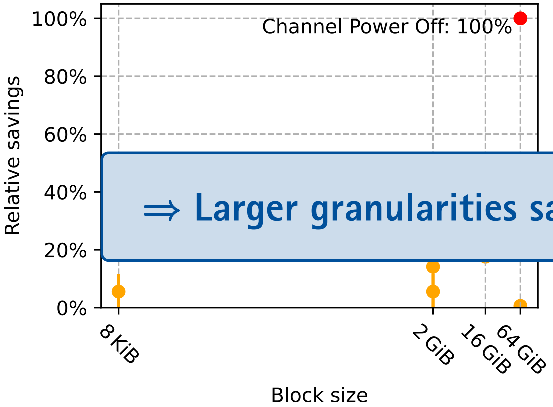
4 ranks per channels

8×2 GiB chips per rank

32 banks per rank

65 536 rows per bank

DRAM Power Saving



⇒ Larger granularities save more power

Example system

512 GiB DDR5

8 channels

4 ranks per channels

per rank

ank

65 536 rows per bank

Conclusion

- Linux is bad at managing physical memory
  - No fragmentation avoidance above (2 MiB) huge page size
  - Page cache unconditionally fills all available memory
  - Immovable pages make compaction impossible
- Using memory comes at a cost
  - Huge/giant page availability
  - Redistribute or turn off unused memory
  - Dynamic cloud pricing models

https://sra.uni-hannover.de/p/dram-dimes24