energie atomique • energies alternatives

An evaluation of Linux Kernel Samepage Merging on simulations

Sébastien Valat

CEA, DAM, DIF F-91340 Arpajon, France

PhD supervisor : William Jalby

CEA,DAM advisor : Marc Pérache

Plan

energie atomique • energies alternatives

- **Introduction of KSM**
- **KSM configuration**
- **Testing KSM on HERA**
- **Conclusion**

energie atomique • energies alternatives

KSM : Kernel Samepage Merging

KSM : Kernel Samepage Merging

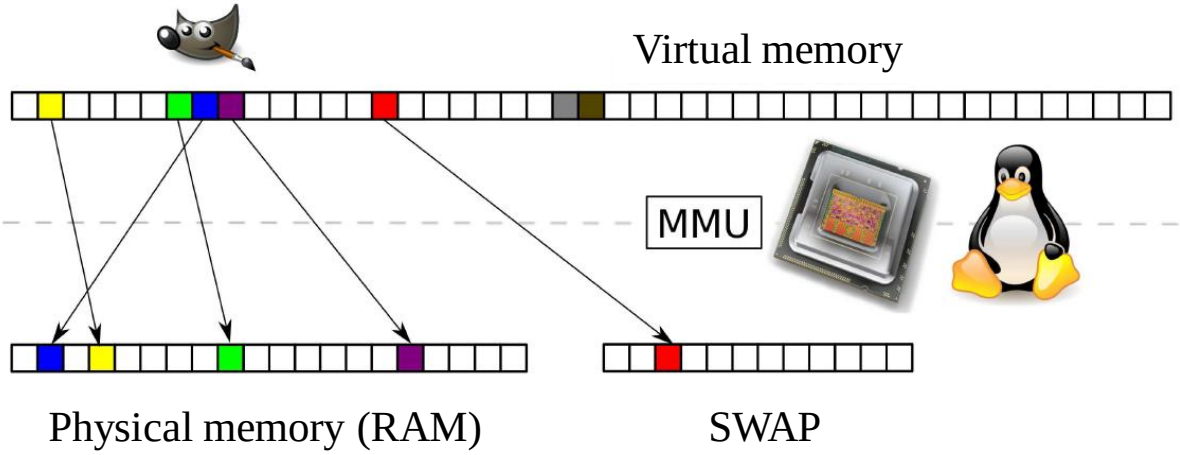
- **Introduced in Linux kernel by version 2.6.32**
- **Goal : reduce memory usage in multi-VM environnement.**
- **Base idea : use the page mapping to merge identical physical pages.**
- **Dynamically detect similar pages with a scaning daemon.**
- **Implemented at kernel level**

KSM : Kernel Samepage Merging

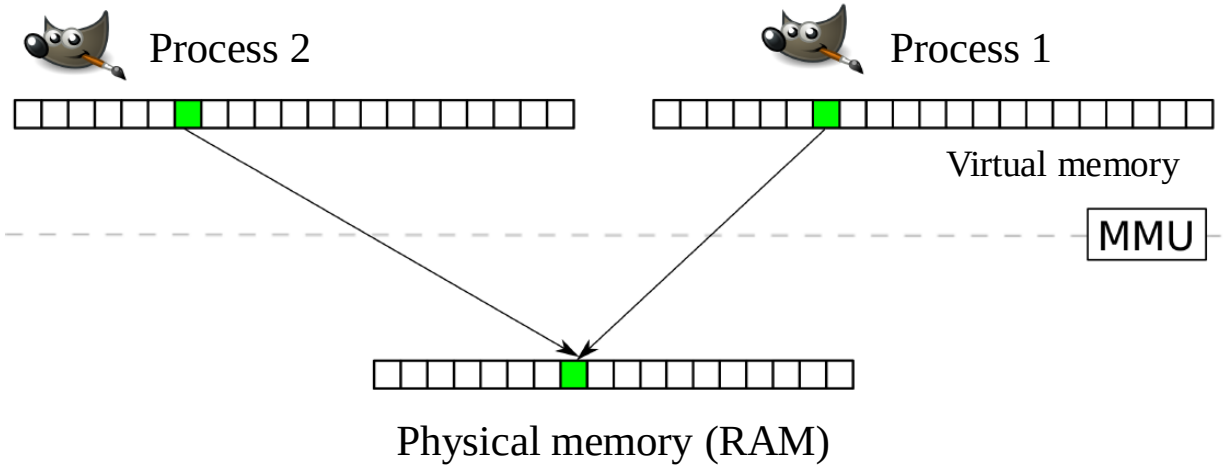
energie atomique • energies alternatives

- **Efficient for VM with same of similar operating systems.**
  - Same binary files
  - Same resources
  - Zeroed pages
- **Transparent for the user.**
- **Price : CPU usage for the scanning daemon**
- **Generic implementation : target standard process, not only VM**

- **Virtual memory :**

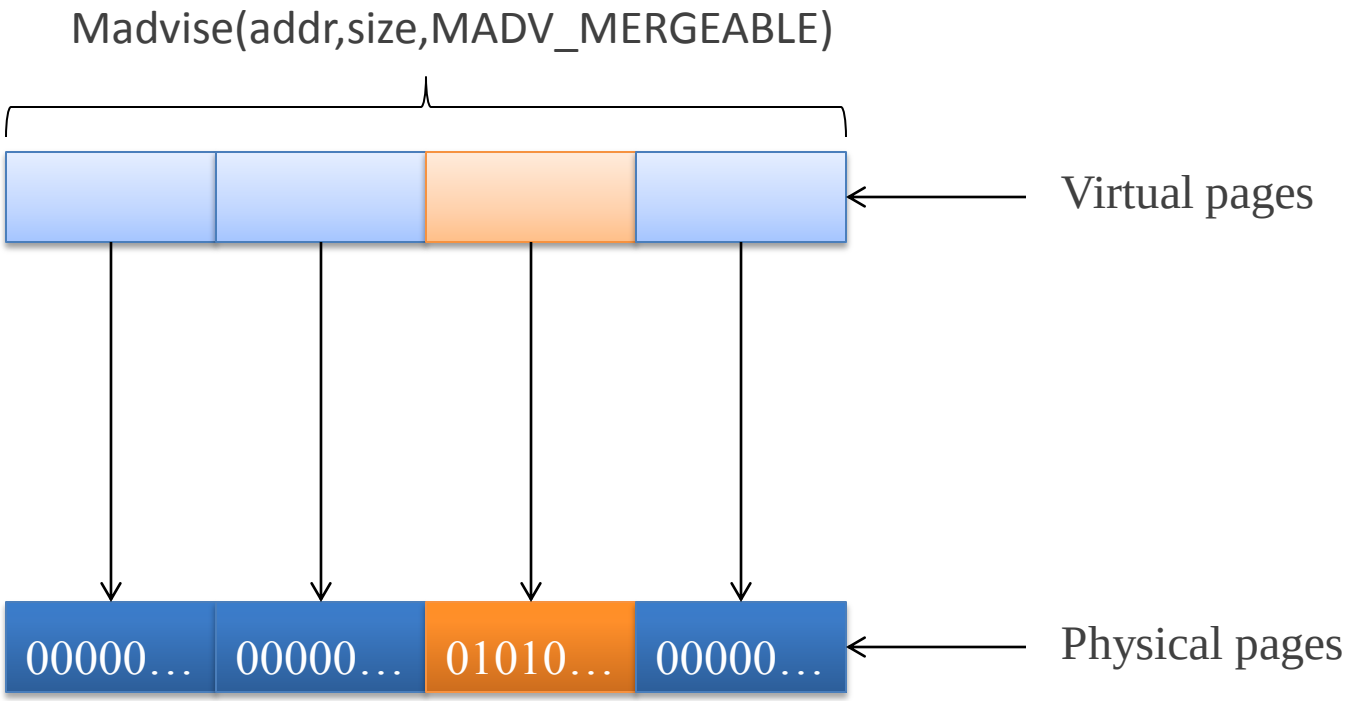


- **Shared memory :**



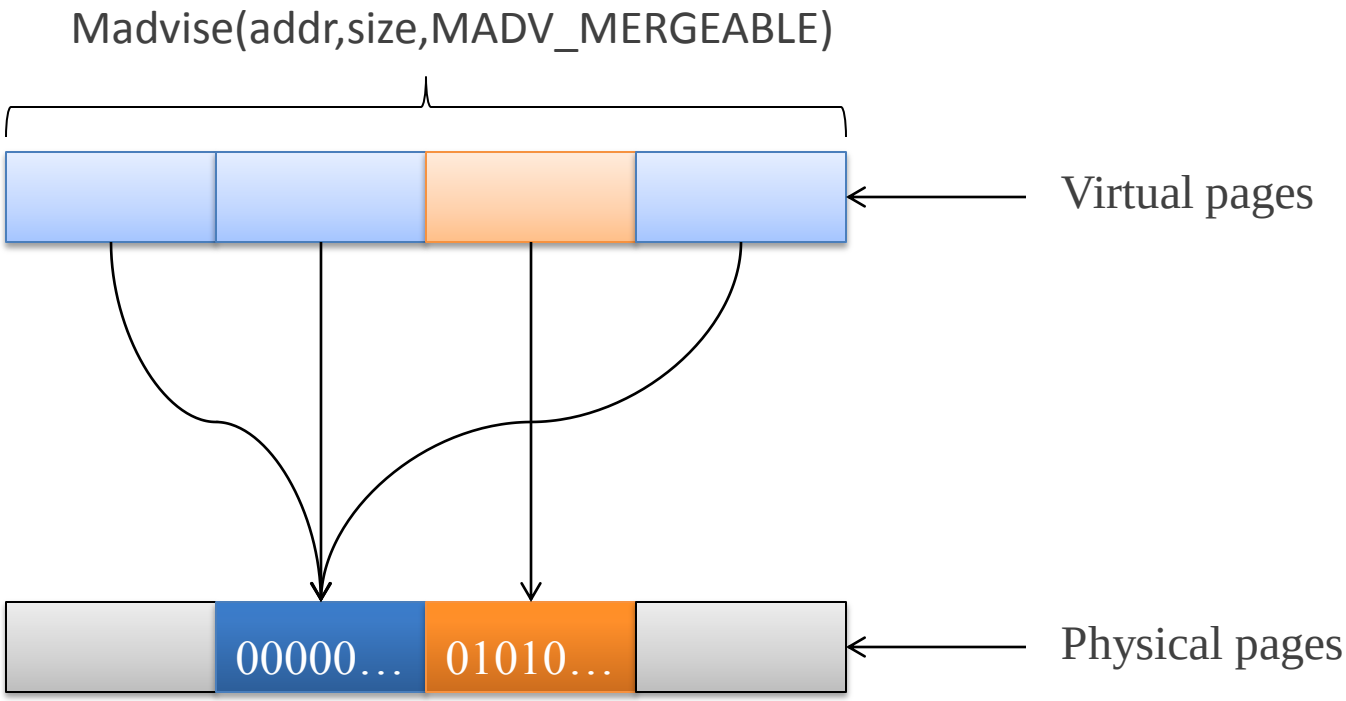
How to use it

- **The user only need to mark the targeted segments.**
- **Marked with system call : *madvise()***



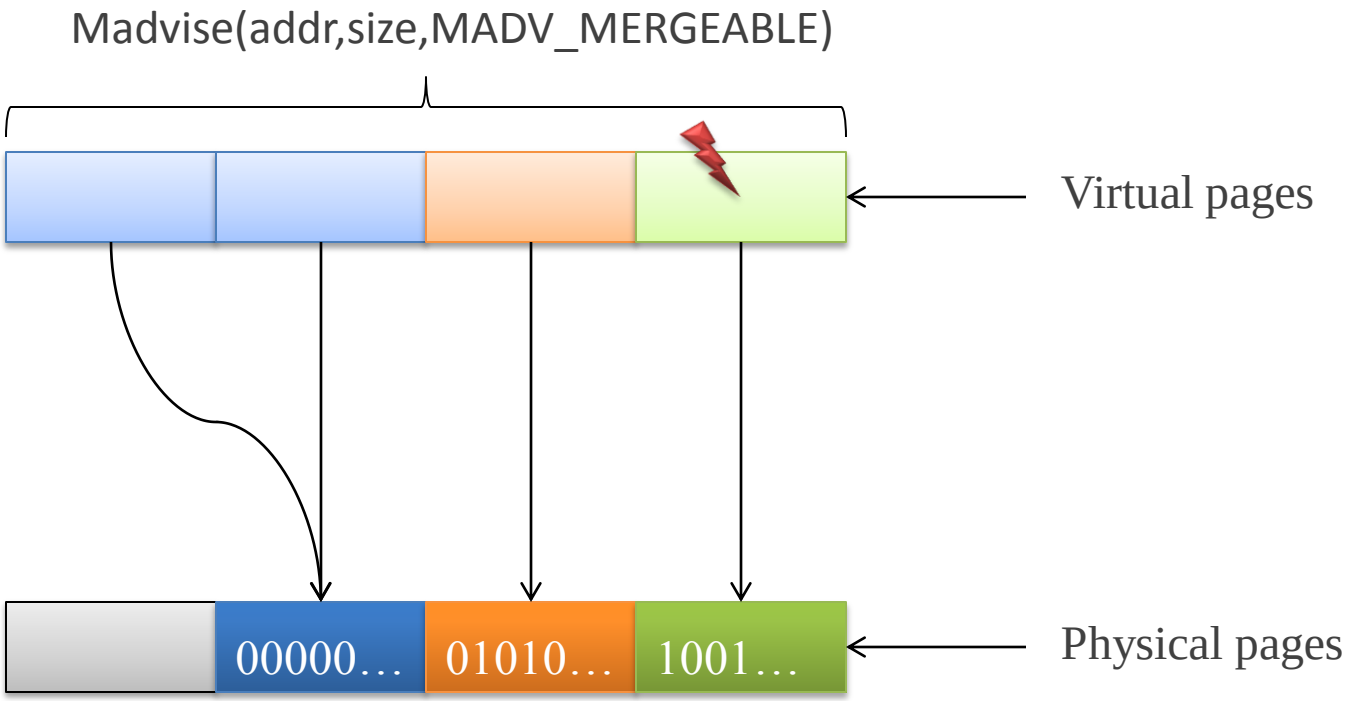
Page merging

- **The daemon (ksmd) scan the pages**
- **Identical physical pages are transparenly merged**



Modifying pages

- **Usage of *Copy On Write* mechanisms**
- **All steps are transparent for the user**



KSM parameters

- **Configuration files : /sys/kernel/mm/ksm/***
- **Parameters :**

Name	Default value	For our tests
Run	0	1
Sleep_millisecs	20	20
Pages_to_scan	100	2000

- **Access limits :**
  - Root to configure and enable/disable KSM
  - Everybody to use the *madvise()* call.
- **Better to select candidate regions**

KSMTuned

energie atomique • energies alternatives

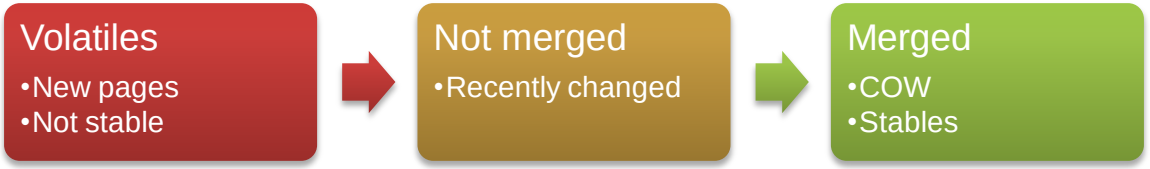
- **A daemon to automatically setup KSM parameters**

- **Permit to :**
  - Automatically enable/disable KSM on memory threshold
  - Automatically increase/decrease *pages_to_scan* parameter depending of the number of pages in queue

- **Available on :**
  - Debian
  - Fedora
  - Redhat

Some internals

- **Three page class :**



- **Two red-black trees :**
  - Stable (merged pages, sorted)
  - Not stable (page ordering not fixed)

- **Tree search by comparing pages bits**

- **Checksum only used to detect unstable, but not for page search.**

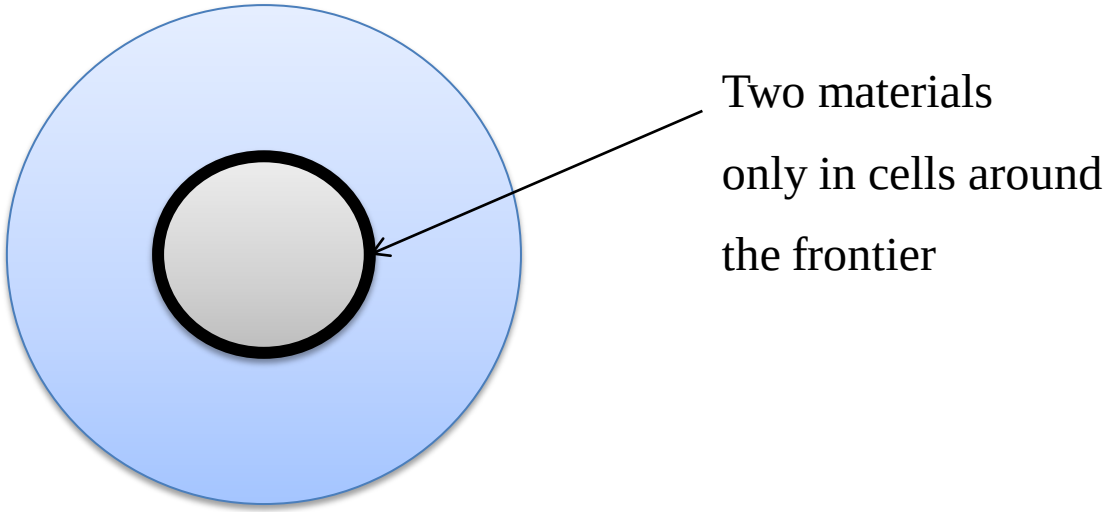
energie atomique • energies alternatives

Evaluation on HERA

Interest

- **HERA is a 3D multi-physics, mutli-material AMR platform**

- **Multi-materials imply :**
  - Require a cells entry for each materials (density)
  - Many location contain only one material
  - It imply many 0 in memory representation



Current solution : using chunks

- **Group blocs of 0 by using indirection at code level**
- **Done by overloading C++ operator []**
  - « Transparent » for the solver
- **Impact :**
  - Memory reduction of roughly 25%
  - Add indirections and loss of compiler optimizations
  - Can double the runtime
- **Is KSM an alternative ?**

energie atomique • energies alternatives

How to measure KSM memory ?

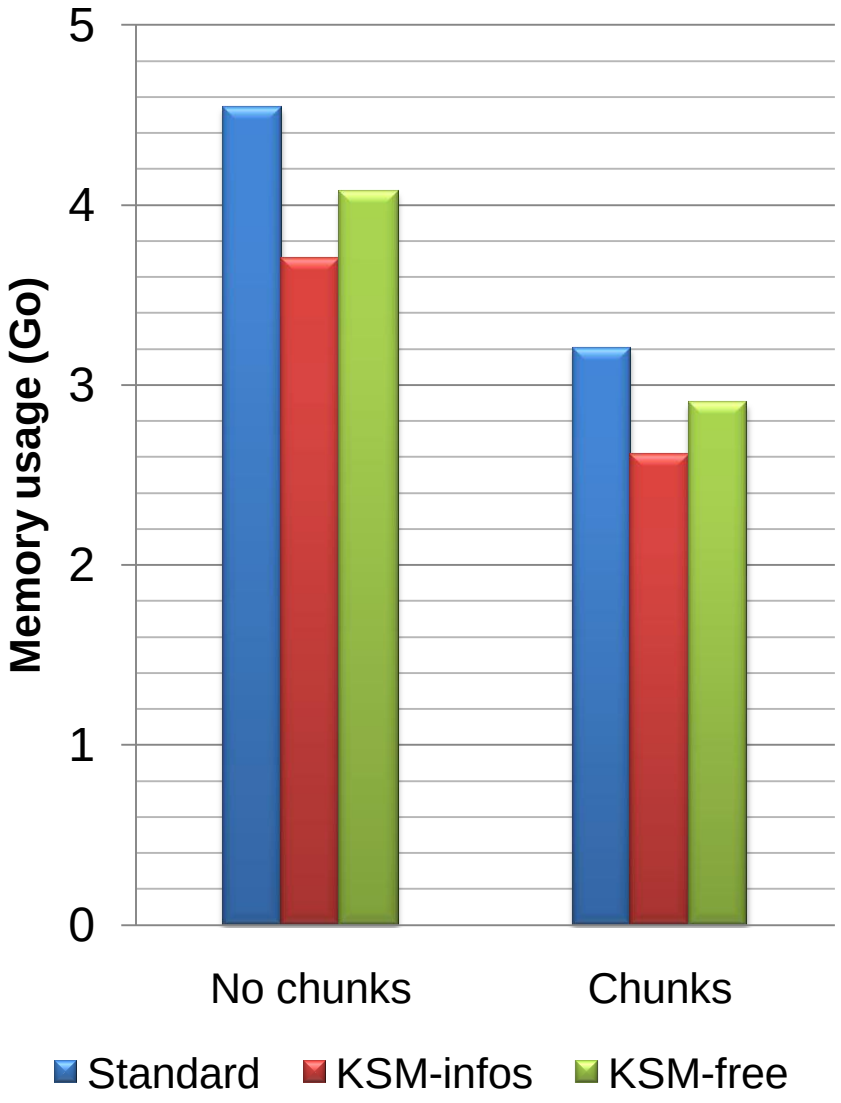
How to measure ?

- **Memory measurement, not easy….**
  - How to evaluate merges between MPI process
  - Evaluator significance
- **Possibilities :**
  - ~~RSS of the current process~~ *(Resident Segment Size)*

    => *not updated by KSM*
  - Total free memory
  - KSM provide some stats in /sys/kernel/mm/ksm/*
    - Number of volatile pages (new or not stable)
    - Number of unshared pages
    - Number of shared pages

How to observe the KSM memory usage ?

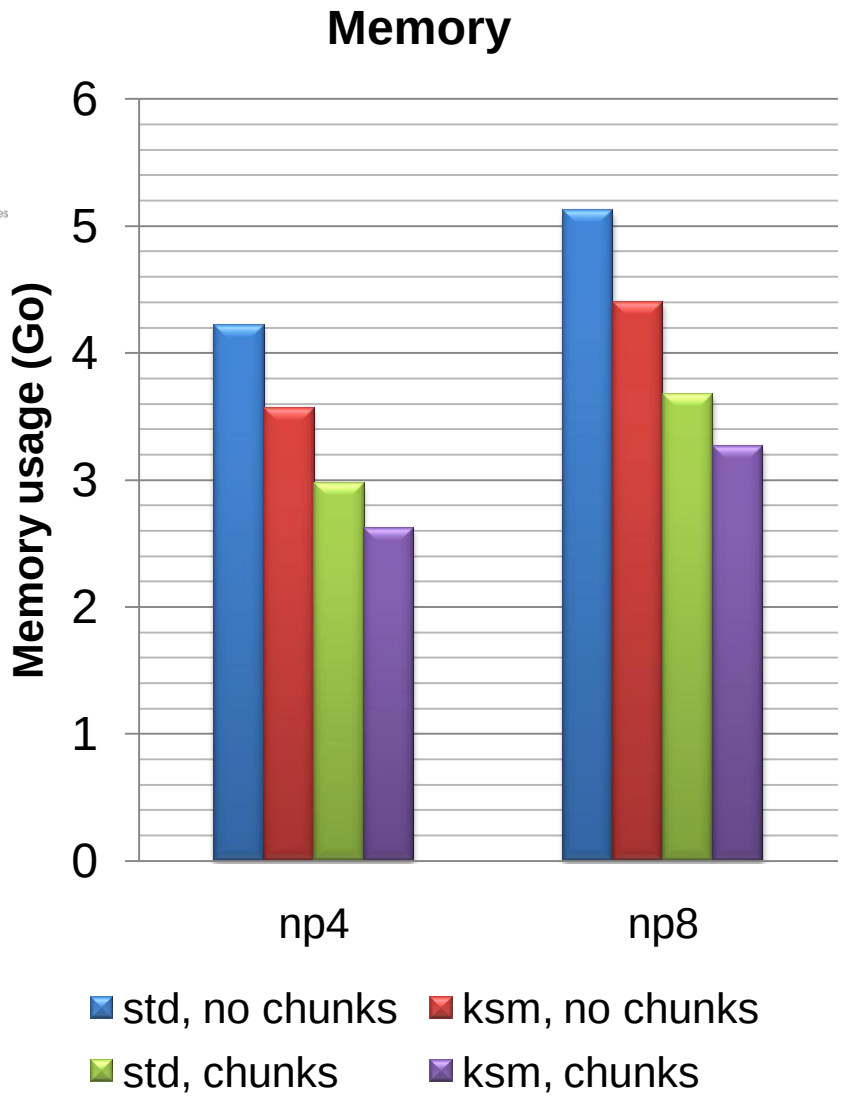
- RSS not updated
  - What about SLURM ?
- KSM infos are biased and overestimate gains
- Free not limited to the process

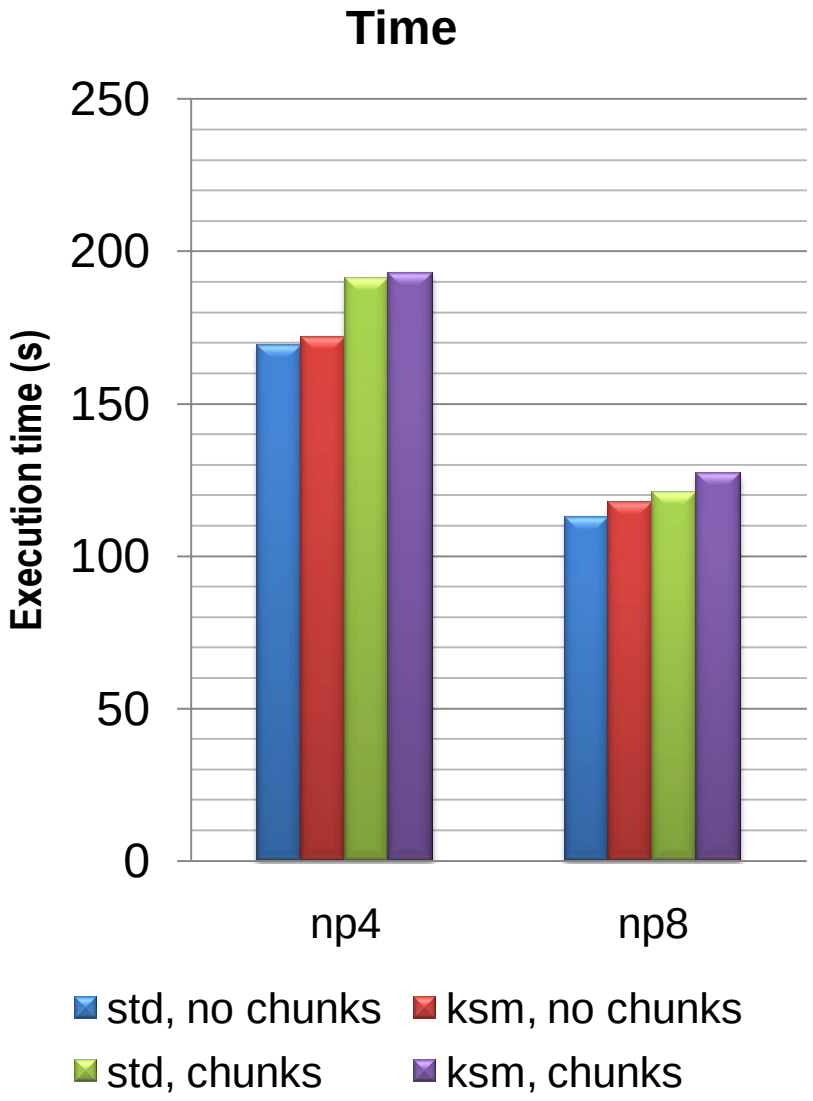


energie atomique • energies alternatives

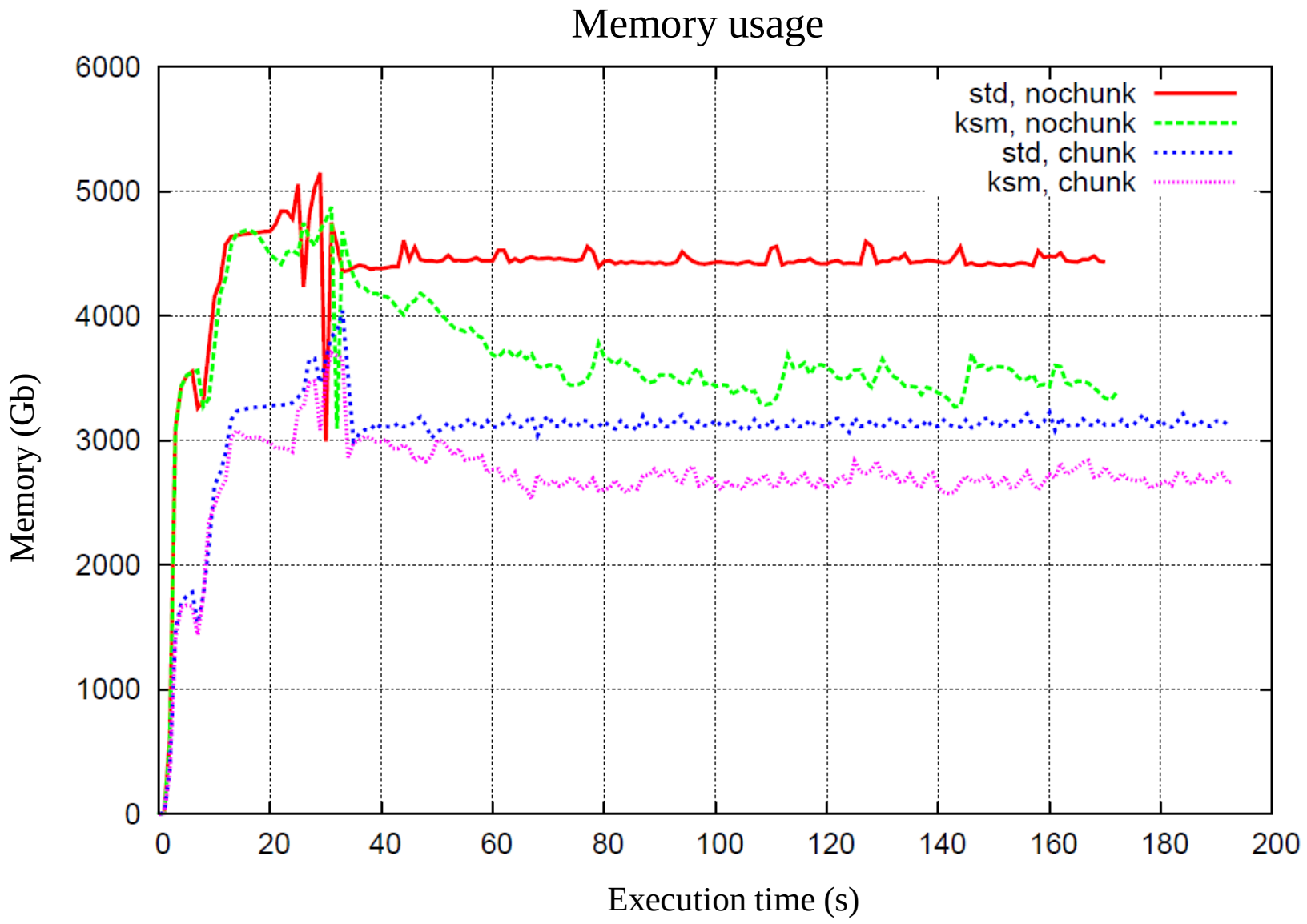
Results with HERA

HERA - 4 or 8 MPI process – 6 materials

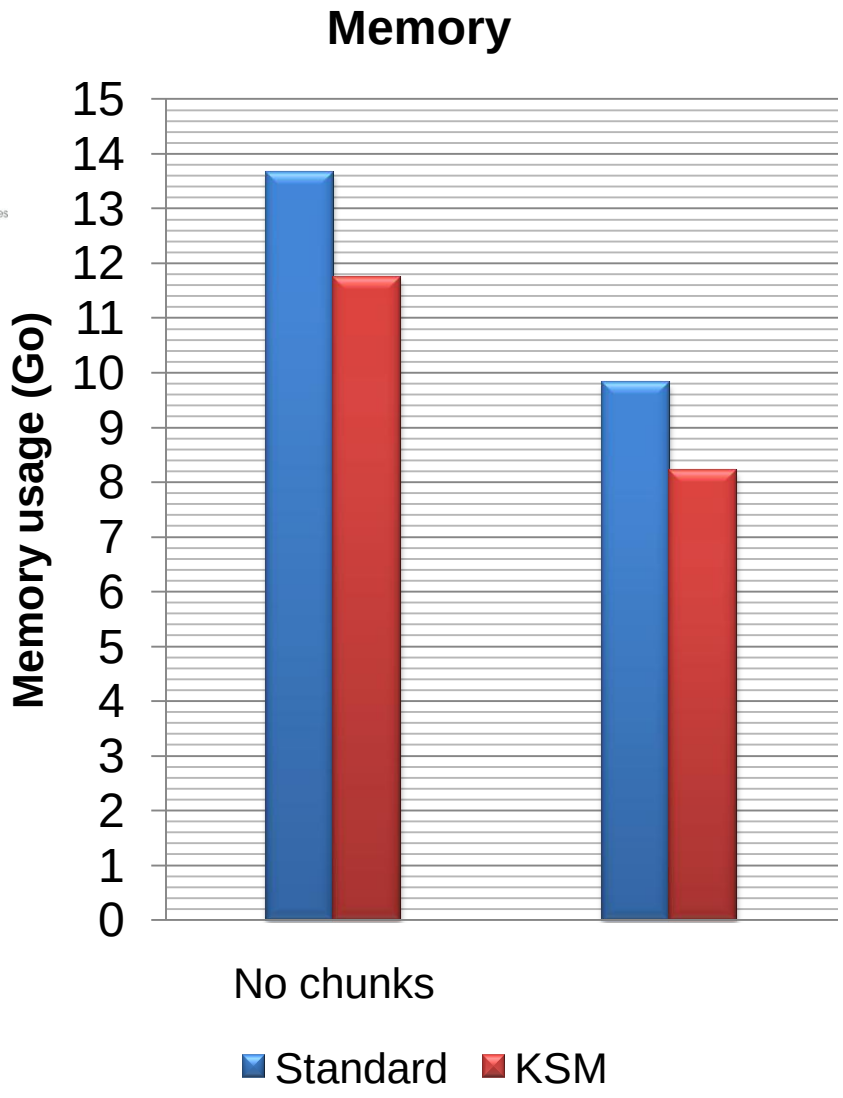


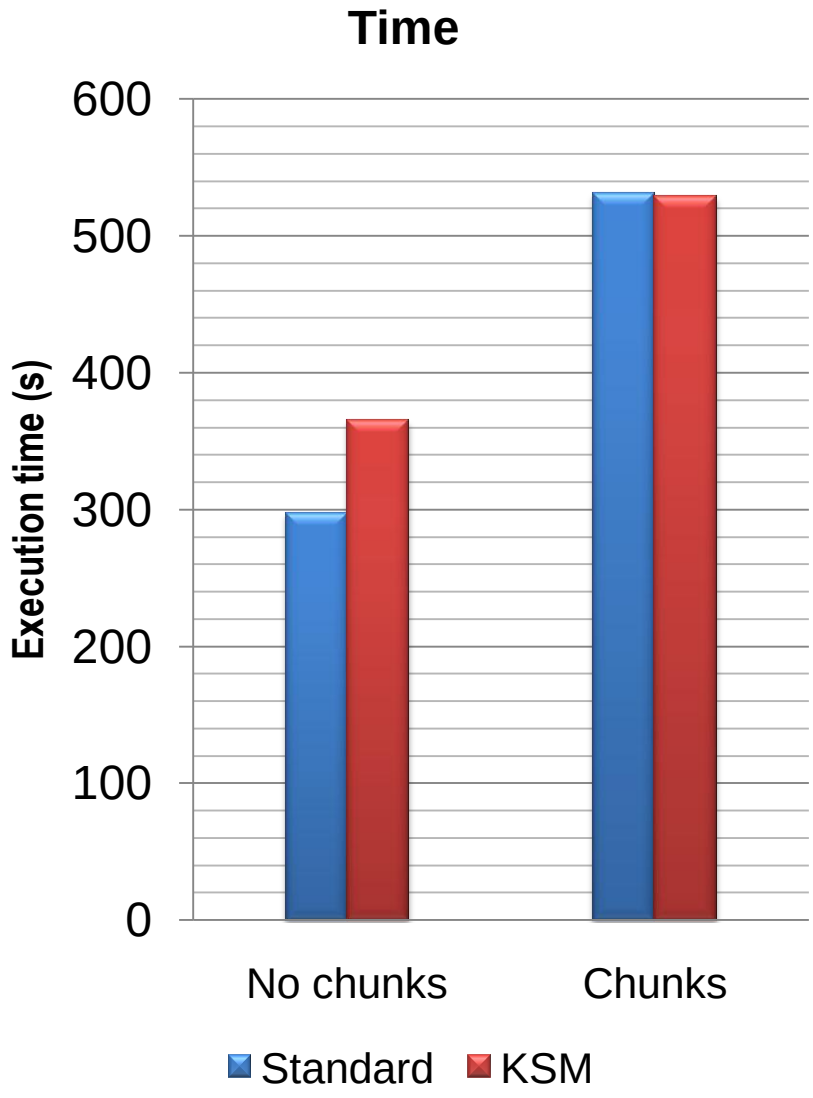


Memory usage over time (8 process)

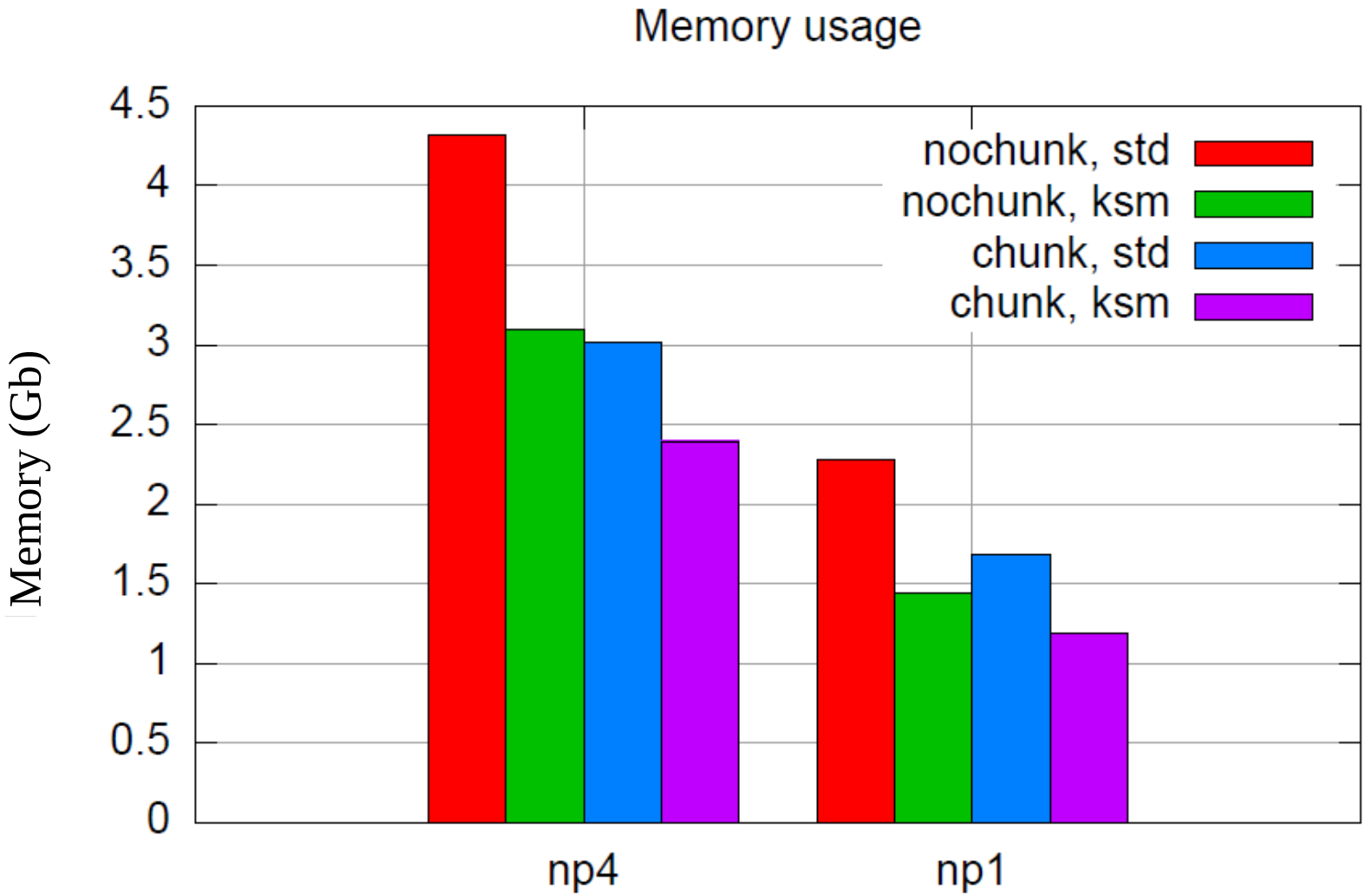


HERA - 8 MPI process – bigger case



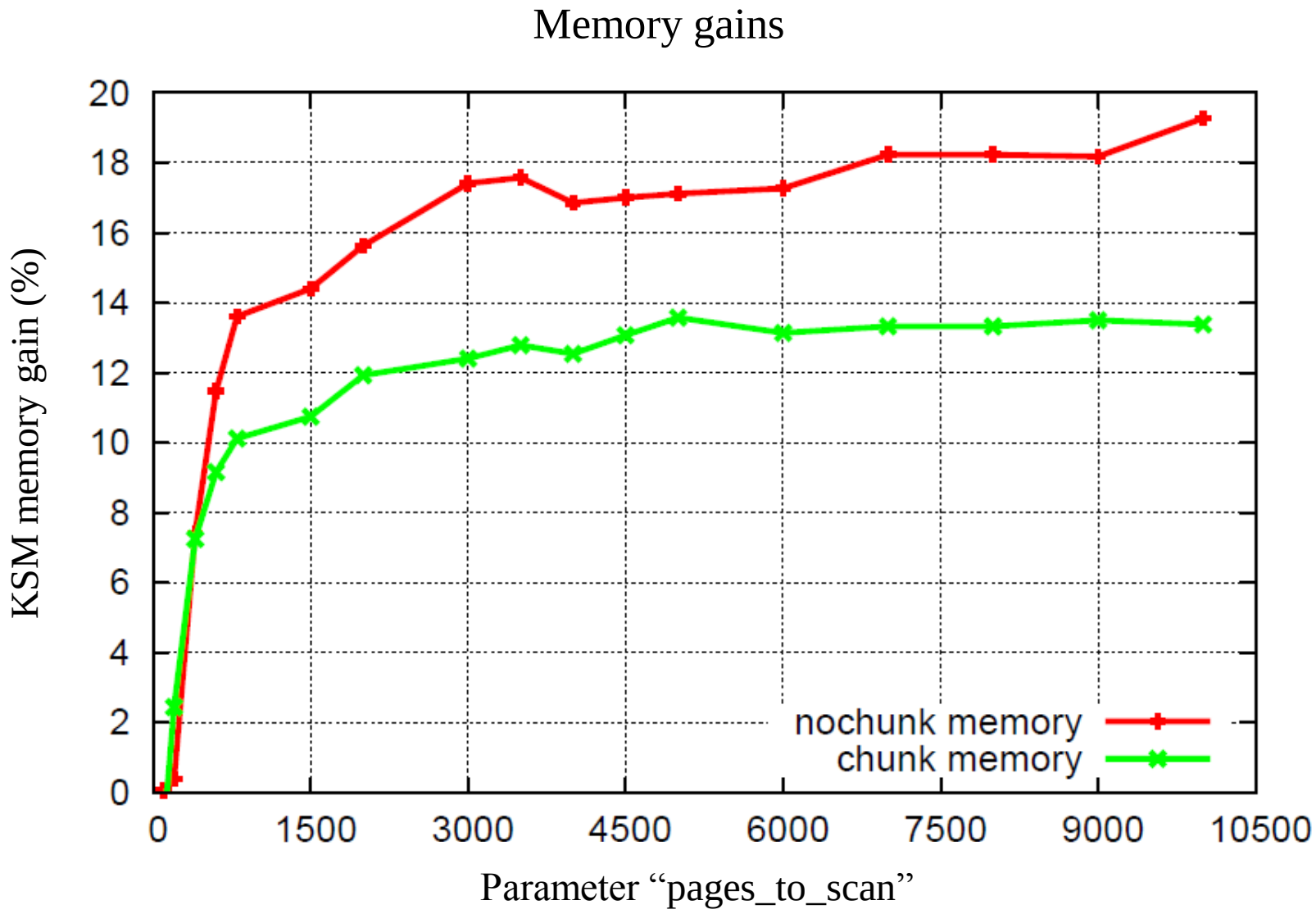


On slower Hera (-finstrument-function)

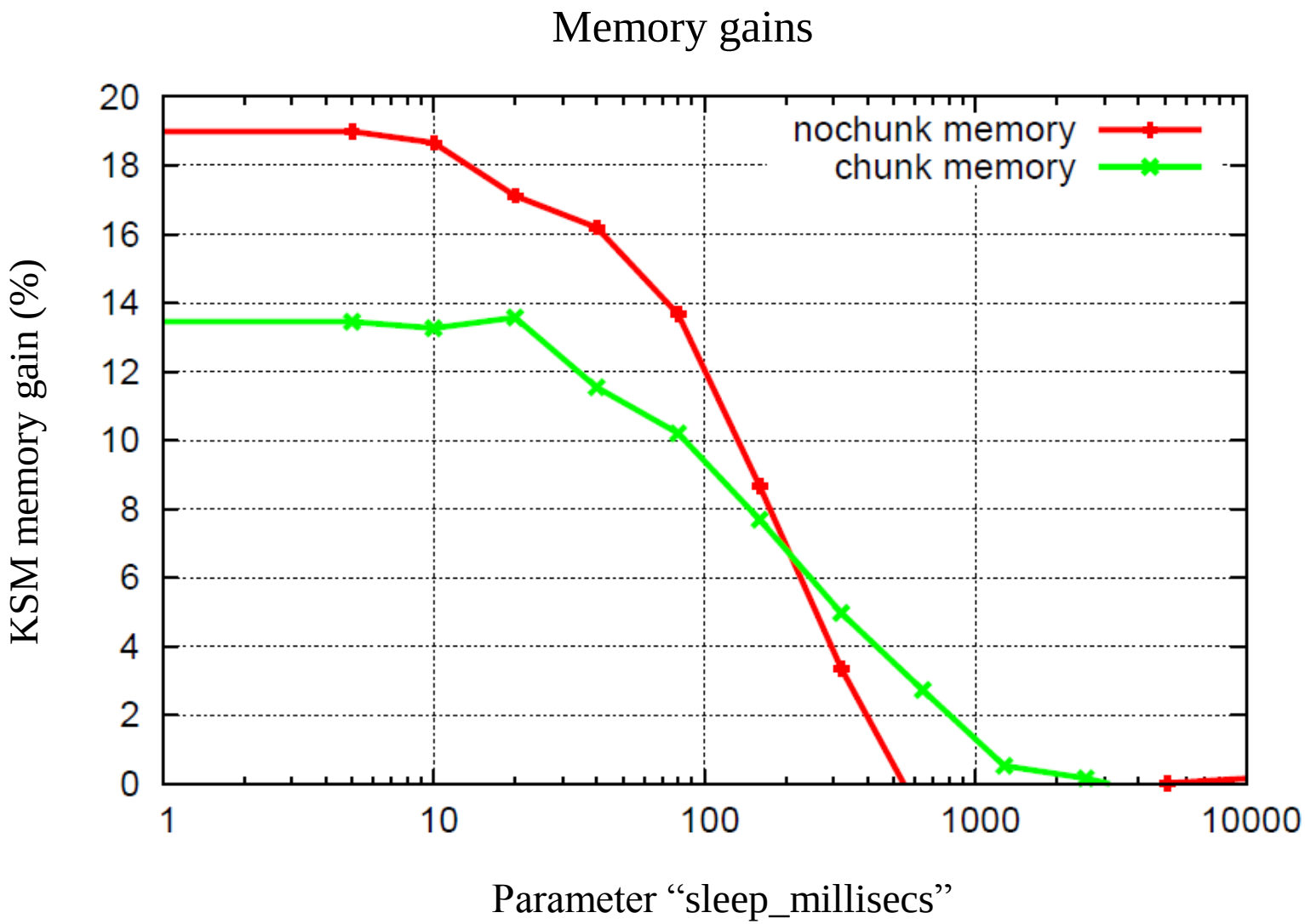


On slower Hera (-finstrument-function)

Parameter page_to_scan, sleep=20



Parameter sleep, page_to_scan = 5000



energie atomique • energies alternatives

Conclusions

Limitation of current implementation

energie atomique • energies alternatives

- **Hardware limitations :**
  - Require large similar segments (multiple of 4Kb)
- **Implementation limitations :**
  - Non support of NUMA architecture
  - Only one scanning thread
  - Parameters are global and limited to root
  - RSS isn’t updated by KSM
  - No way to do scan on demand or control KSM from process

Conclusion

energie atomique • energies alternatives

- **Initially designed for VM but usable for applications**
- **KSM can reduce memory usage of some applications**
  - ~12-16% on Hera with or without chunks
  - Max performance degradation observed 23%, but mostly around of a few percent.
- **Provide a transparent way to implement indirections**
- **Get gains even with 4Kb limitation.**
- **Interesting, but need some improvement for usage in HPC**

energie atomique • energies alternatives

THANKS

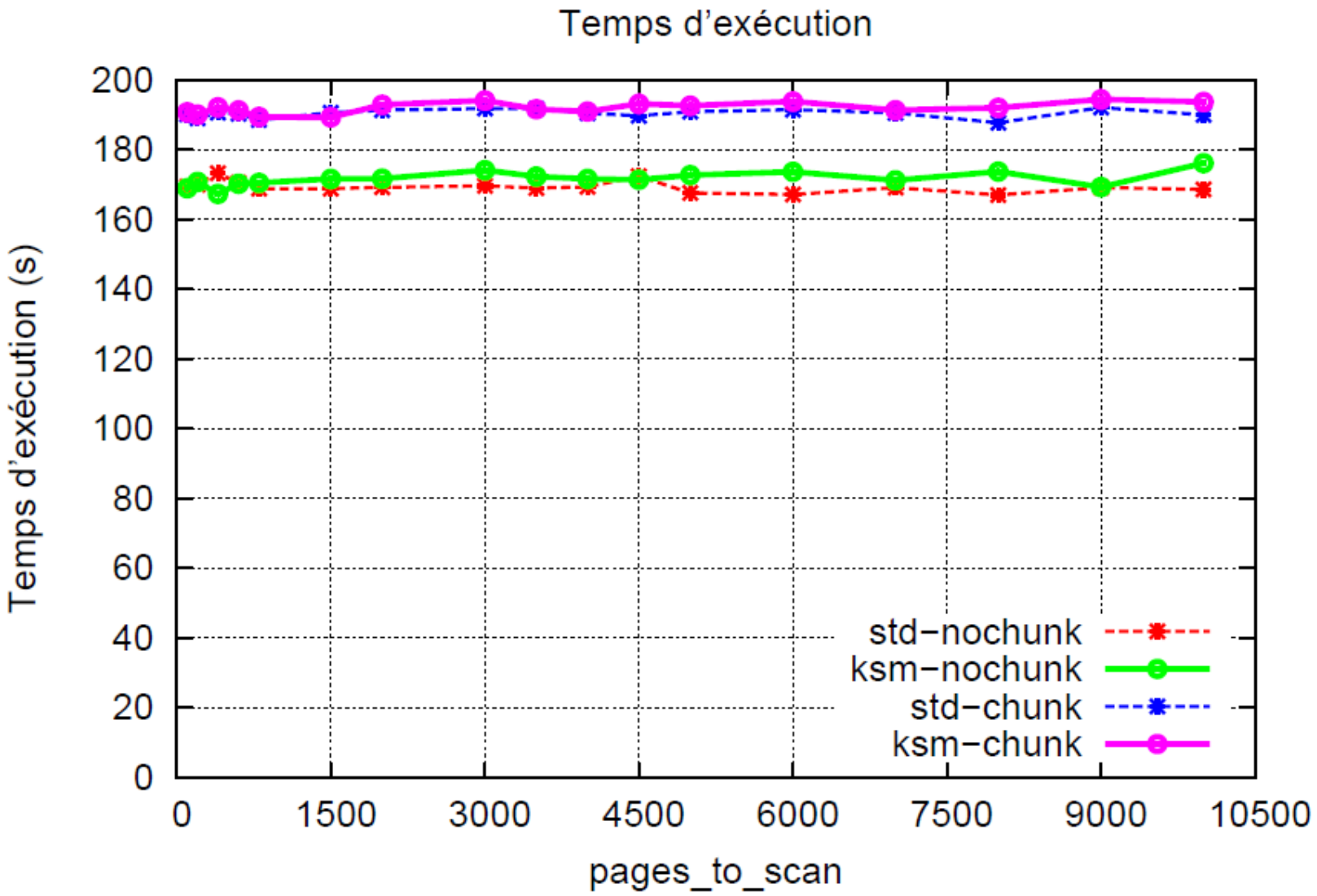
References

- **[1] Andrea Arcangeli, Izik Eidus, and Chris Wright. Increasing memory density by using KSM. In Proceedings of the Linux Symposium, pages 19–28, July 2009.**

energie atomique • energies alternatives

BACKUP

BACKUP


Temps d'exécution
Temps d'exécution (s)
200
180
160
140
120
100
80
60
40
20
0
0
1500
3000
4500
6000
7500
9000
10500
pages_to_scan
std−nochunk
ksm−nochunk
std−chunk
ksm−chunk

energie atomique • energies alternatives