A Memory Swapper for Virtualized Environments

Nadav Amit, Dan Tsafrir, Assaf Schuster

Technion Computer Engineering

Contribution

1. Analysis of uncooperative swapping overheads
2. Swap Mapper to save memory-disk associations
3. Swap Preventer to avoid swap on reallocation
4. Significant improvement of common workloads

Virtual Machines Memory Overcommitment

Memory overcommitment allows the hypervisor to oversubscribe memory to virtual machines.

VM (2GB) VM (2GB) VM (2GB)   Host (4GB)
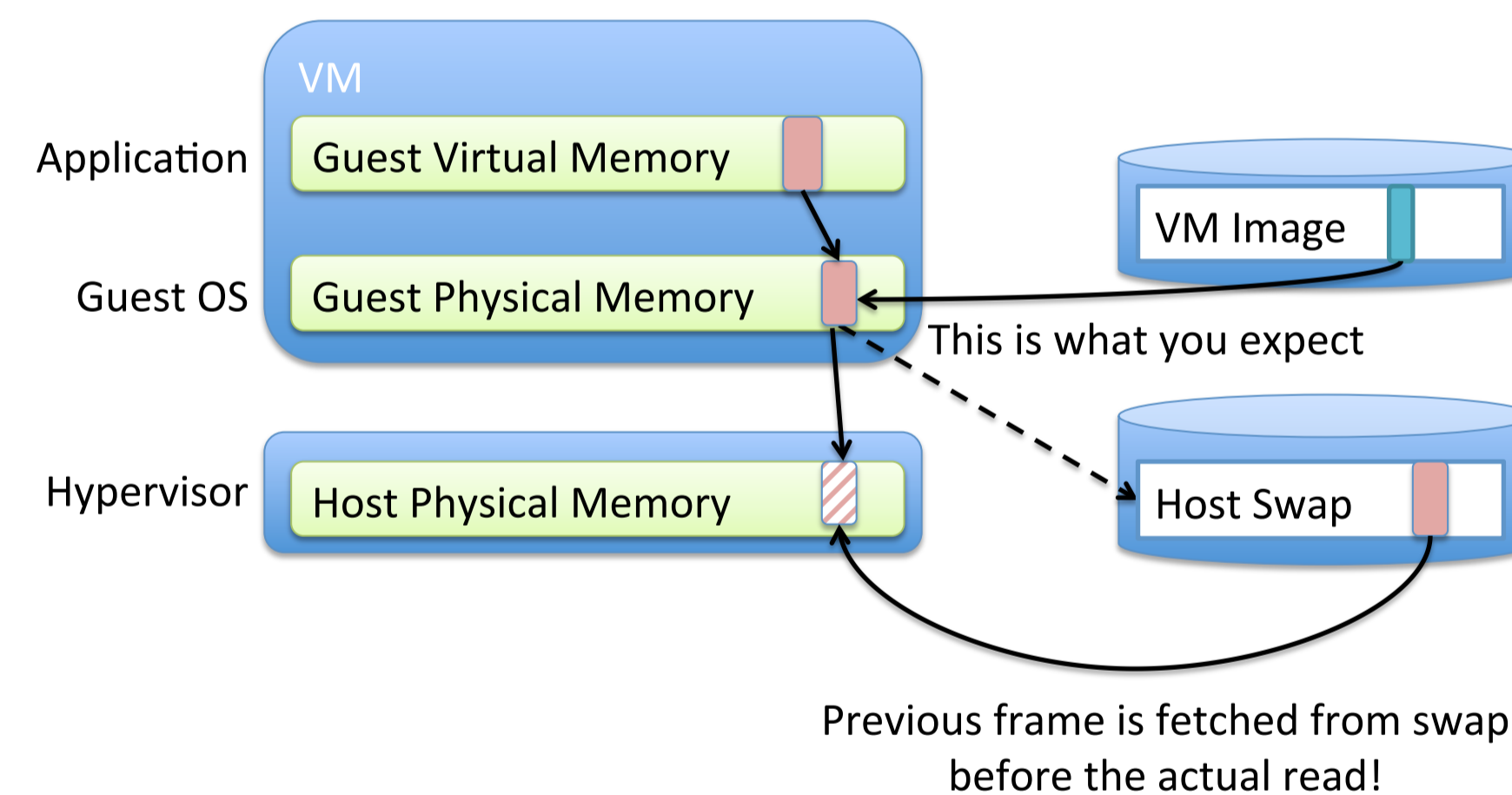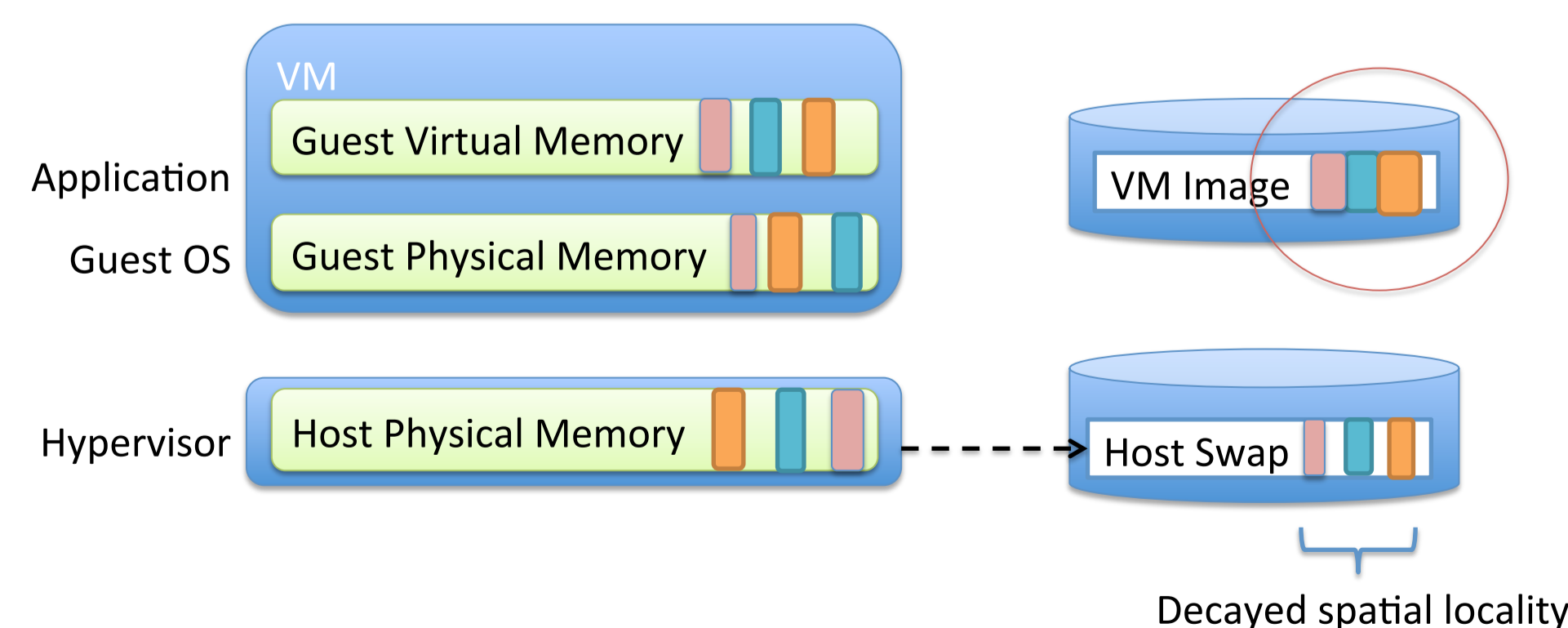Guest Memory Guest Memory Guest Memory   Host Memory

Two prevailing solutions:
Balloon – paravirtual, limited use, slow to respond
Host swapping – fully virtual, poor performance

Analysis: Uncooperative Swap Overheads

1. Reads of stale memory which is being overwritten. The case of a VM disk read is illustrated.

VM
Application — Guest Virtual Memory
Guest OS — Guest Physical Memory
Hypervisor — Host Physical Memory

VM Image
This is what you expect
Host Swap
Previous frame is fetched from swap before the actual read!

2. Memory backed by VM disk is swapped out, marked anonymous and loses spatial locality.

VM
Application — Guest Virtual Memory
Guest OS — Guest Physical Memory
Hypervisor — Host Physical Memory

VM Image
Host Swap
Decayed spatial locality

3. False anonymity of guest memory.
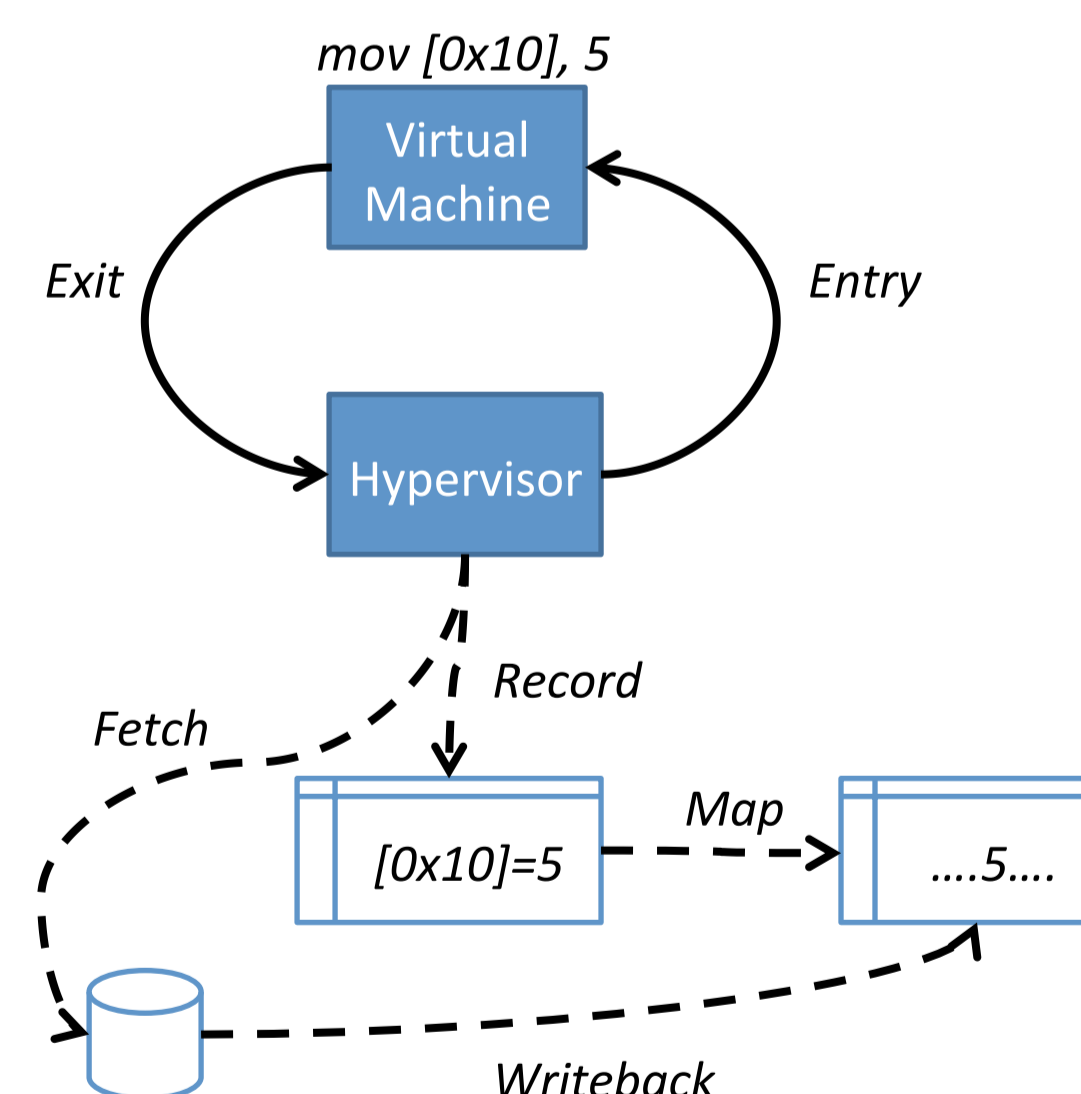OS PFRA prioritize page-cache evictions.

Efficient Host Swapping

Swap Mapper
- Clean pages which can be backed by the disk are mapped to disk blocks (new *mmap* variant).
- Mapped pages are discarded on eviction.
- Consistency is maintained on writes to blocks that back memory (unlike mmap).
- Prefetch only mapped memory.
- Use 4KB disk blocks.

False Read Preventer
- Hypervisors cause false swap reads when the guest OS reallocates page frames (e.g., zeroing pages, COW).
- VSWAPPER eliminates these reads by emulating write instructions to non present memory and writing the result into write buffers.
- Full write buffer pages can be mapped to VM.
- Disk reads can be saved.

mov [0x10], 5
Virtual Machine
Exit   Entry
Hypervisor
Fetch   Record
[0x10]=5   Map   ....5....
Writeback

Related Work

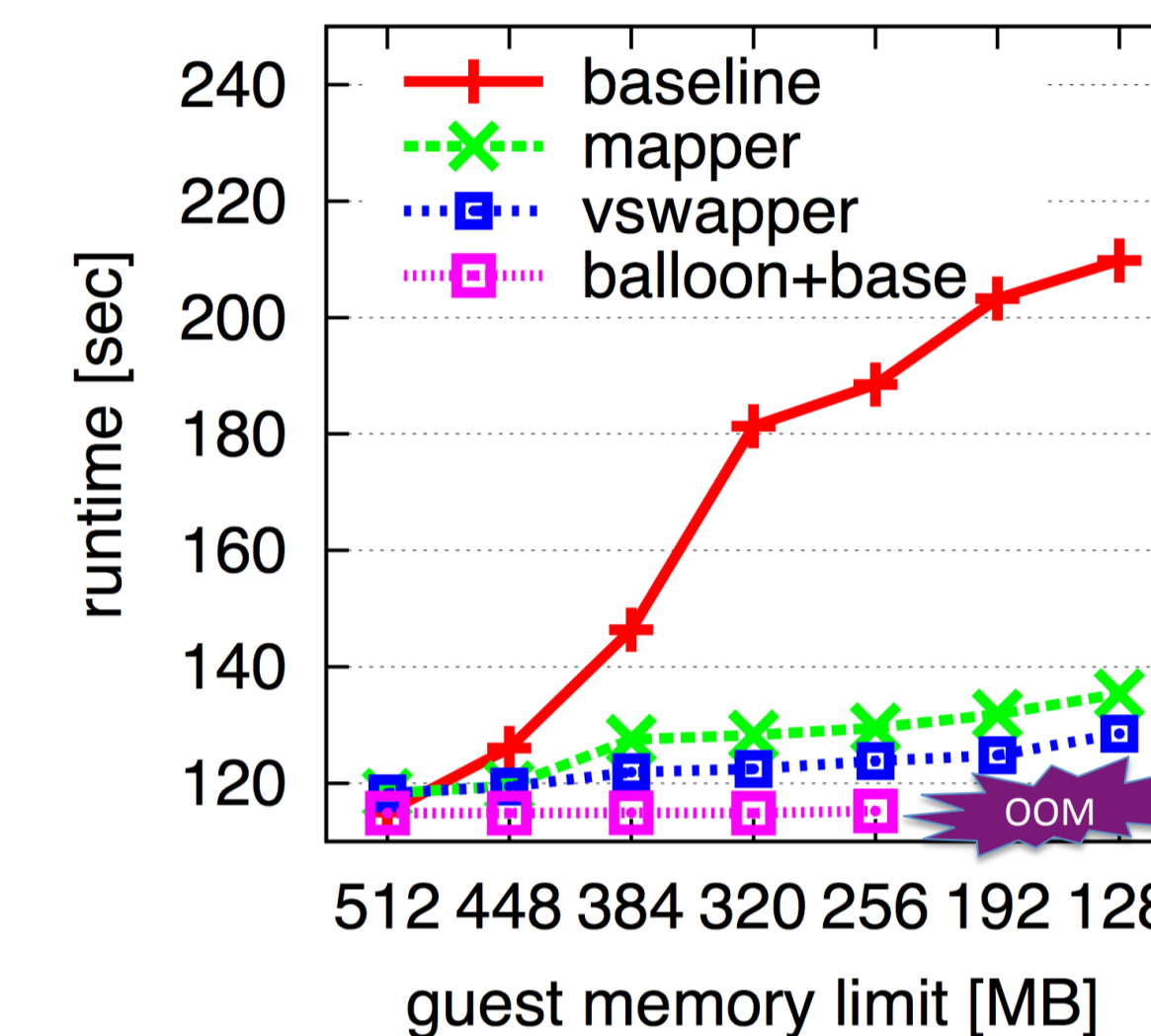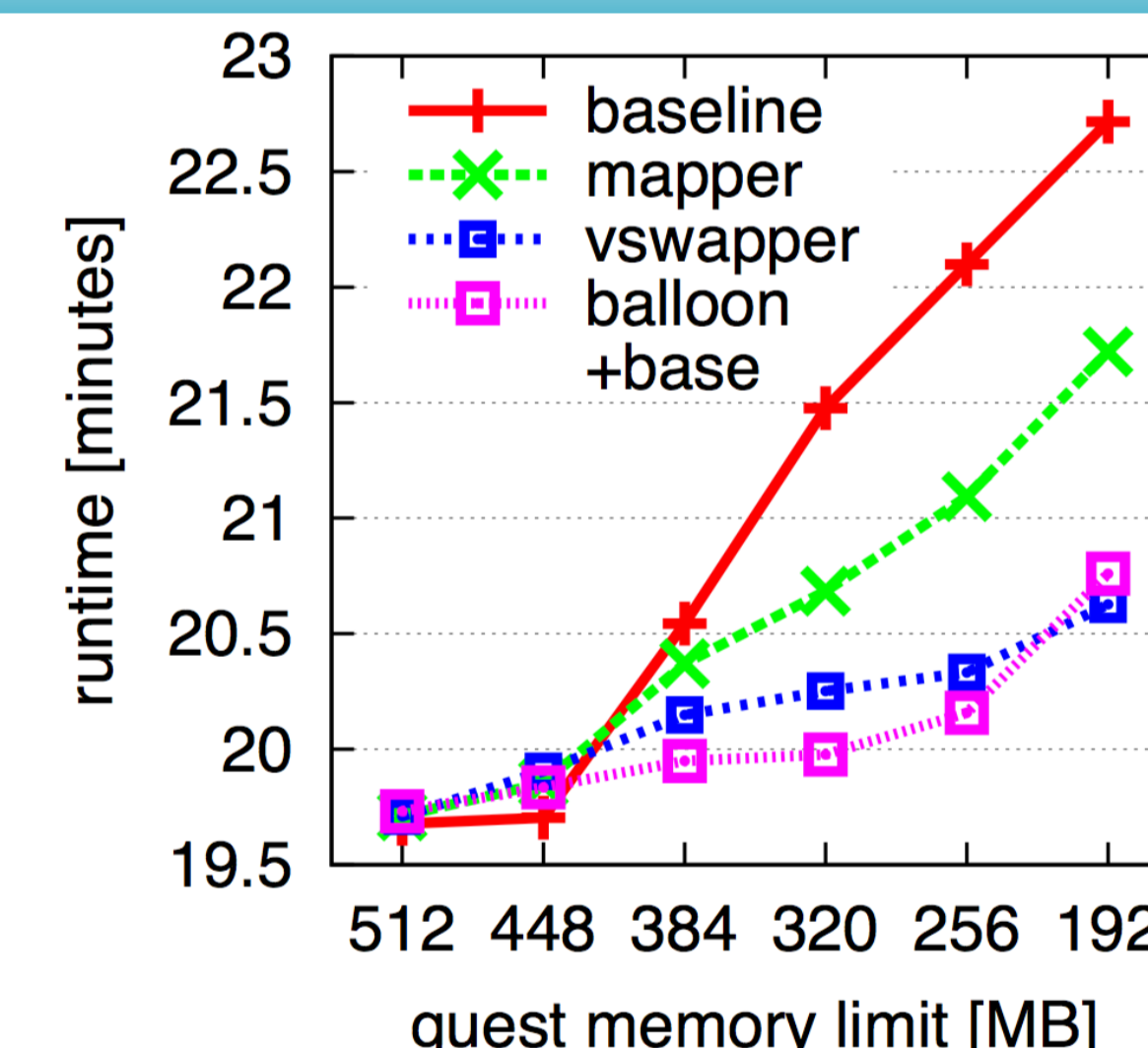Monitoring the buffer cache [Jones06]
Copy-on-write disks [Bugnion97]
VM memory access tracing [Lu07]
Asynchronous page faults [Useche12]

Static Results: KVM, 512 MB guest, 1 VCPU

PBZIP2 (Parallel BZIP2)

baseline
mapper
vswapper
balloon+base

runtime [sec] vs guest memory limit [MB]: 512 448 384 320 256 192 128
OOM

Avoiding stale reads and false anonymity improves performance by up to 1.63x.

Kernbench (Linux Kernel Compilation)

baseline
mapper
vswapper
balloon +base

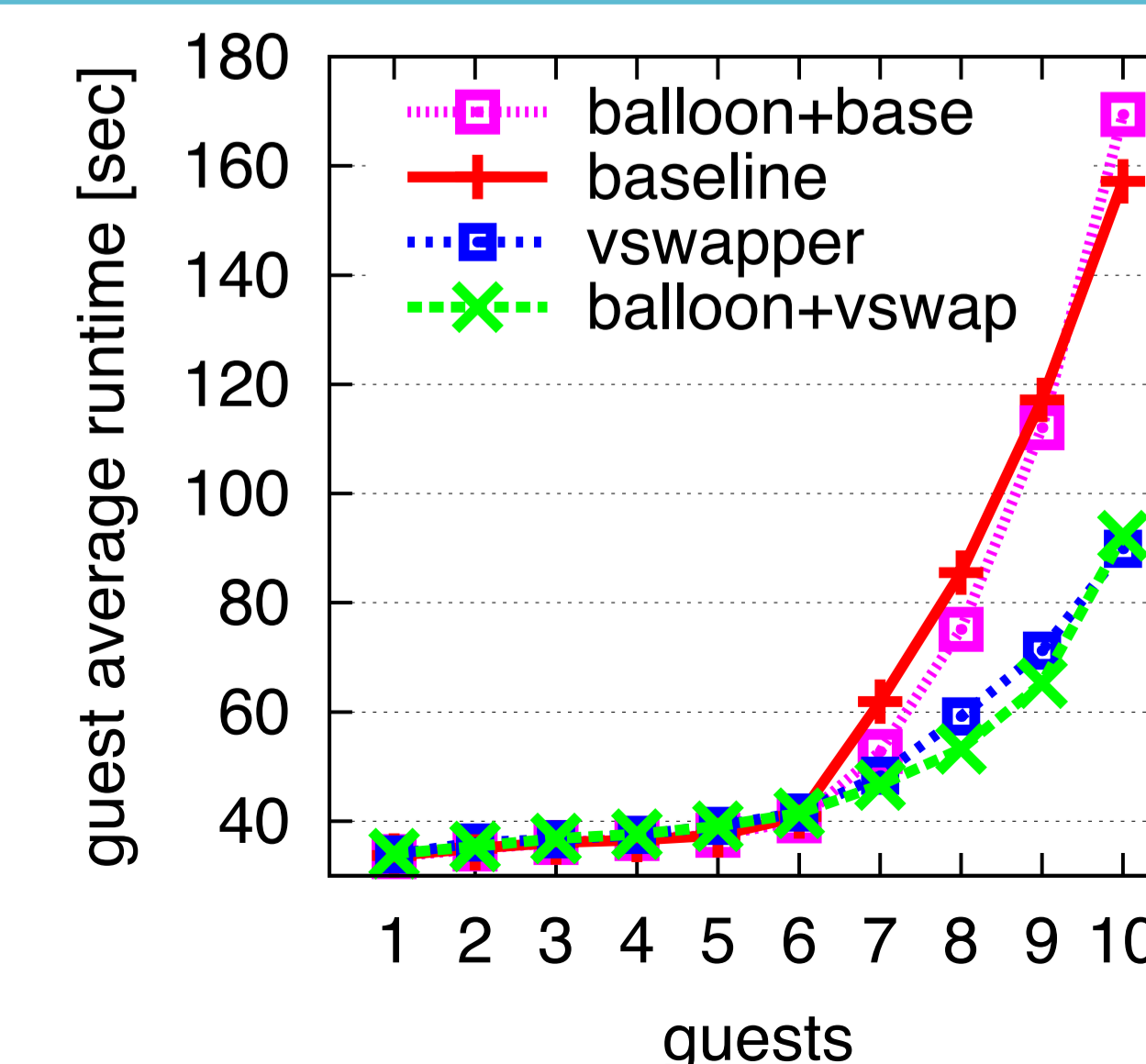runtime [minutes] vs guest memory limit [MB]: 512 448 384 320 256 192

Up to 1.1x performance improvement.

Swap Preventer saves thousands of swap operations.

Dynamic Results: 2GB guest – 8GB Host w/MoM

MapReduce with multiple VMs

balloon+base
baseline
vswapper
balloon+vswap

guest average runtime [sec] vs guests: 1 2 3 4 5 6 7 8 9 10

Balloon is less effective for dynamic workloads.

VSWAPPER delivers up to 2x improvement with ballooning