Cache Pirating: Measuring the Performance Impact of Cache Sharing

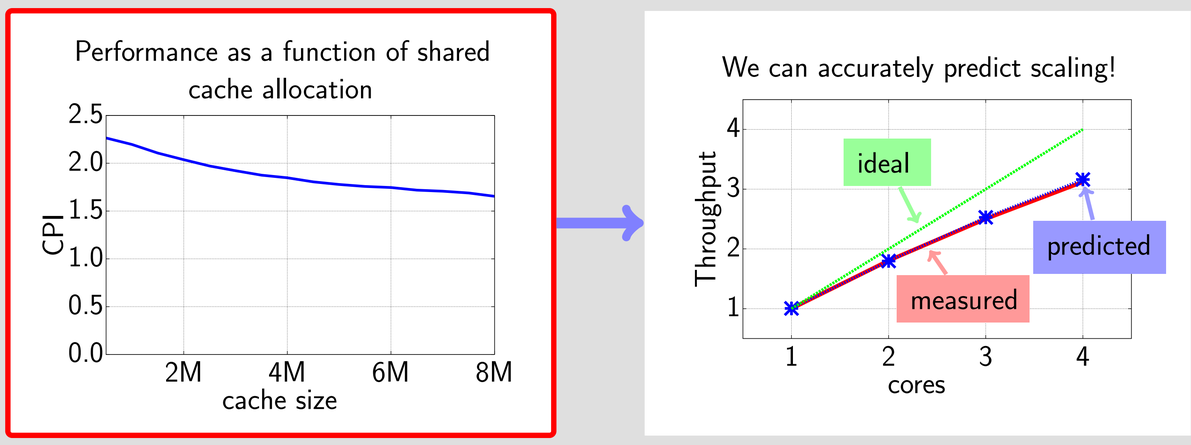
UPMARC Uppsala Programming for Multicore Architectures Research Center

Nikos Nikoleris, David Eklov, David Black-Schaffer and Erik Hagersten

UPPSALA UNIVERSITET

Shared Cache Matters

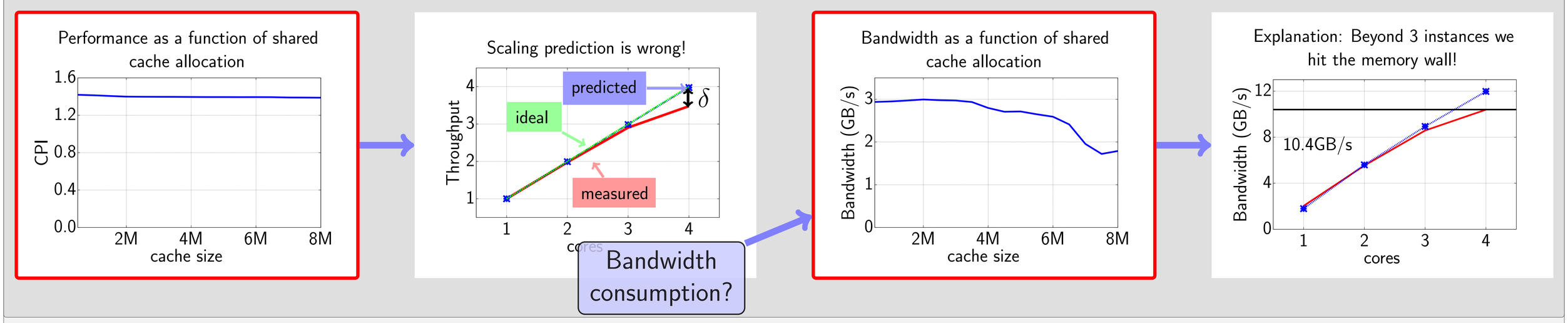
1. OMNeT++



We run up to four instances of the same application on a quad core Nehalem with 8*MB* of shared cache

- Each instance takes equal portion of the cache.
- Naively expect: throughput to scale perfectly.
- Measured data: show that shared caches and bandwidth have a great impact on performance (not surprising).
- With Cache Pirating we can accurately predict the performance impact due to cache sharing on real hardware with 5.5% overhead.

2. LBM



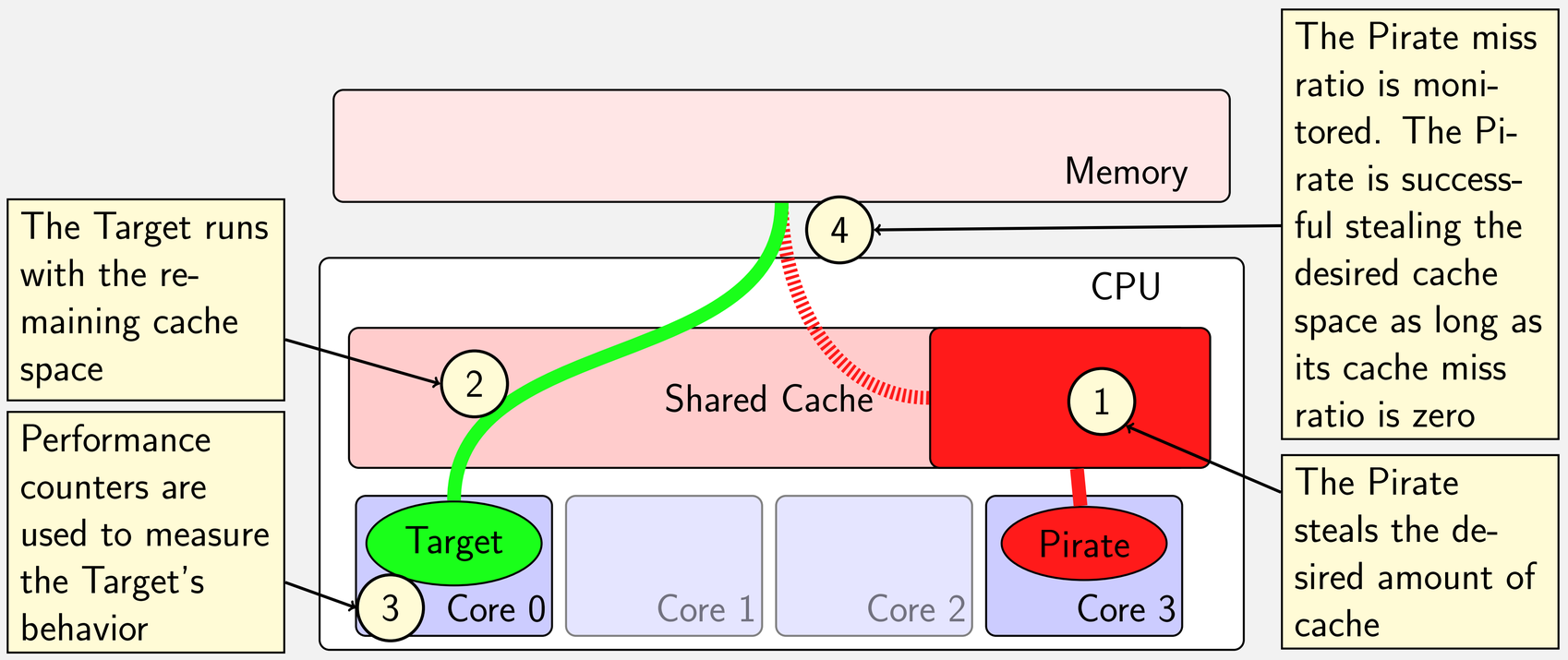
Overview

Cache Pirating is a *low overhead method* for measuring application performance and off-chip bandwidth as a function of the available *shared cache* capacity on real hardware.

The Target application is co-run with a Pirate application that steals shared cache. The Pirates steals the desired amount of cache to control how much the Target gets.

Cache Pirating uses performance counters to measure the Target's behavior. Any performance metric available on the hardware can be measured.

Cache Pirating



Application Insights from Cache Pirate Data