The paper [1] experiments Block/Random and Private/Shared tests on IBM Power 6 and Sun UltraSPARC T2 Plus with STREAM Copy Benchmark. For Block/Random test, the cores access the long arrays in two different ways: in block-wise manner or randomly. Data localities become maximized in block-wise manner while they become minimized in random access. For Private/Shared test, they created a different version of the STREAM Copy benchmark operating on short vectors each of which is smaller than a cache line. For Private-test, a thread exclusively accesses a short vector with no L1 false sharing while the threads on different cores share the short vectors for Shared-test, which cause to maximize L1 false sharing.

IBM Power 6 consists of 2 cores and 2 threads and SMT per core are supported by hardware. A core has its own on-chip L1, L2 and shared off-chip L3 caches. Sun UltraSPARC T2 Plus is composed of 8 cores with hardware support for 8 threads per core and fine-grained thread scheduling and each core has its own L1 on-chip and a shared on-chip L2 cache.

Based on the above workload characteristics and system classes, we roughly but accordingly set up workload and a model that characterizes the system class. Then we explore performance behaviors of the systems with our theoretical framework.

The Figure 1-4 compares the results from our theoretical framework and the ones given in [1]. The model attempts to capture the overall system behaviors, rather than matching data values, which require detailed data about the workload and system parameter setups, which we don't have access to. For Block-test, the throughput increases for Power 6 and T2 stop at 4 threads while for Random-test, memories of both systems become bottlenecks resulting in no performance improvement with the increase of the number of threads on both systems. For Private/Shared-test, both systems reach their maximum speed with a few threads in Private-test but with more threads in Shared-test. This results show that multithreading is more effective to compensate for bad core-to-core cache locality.
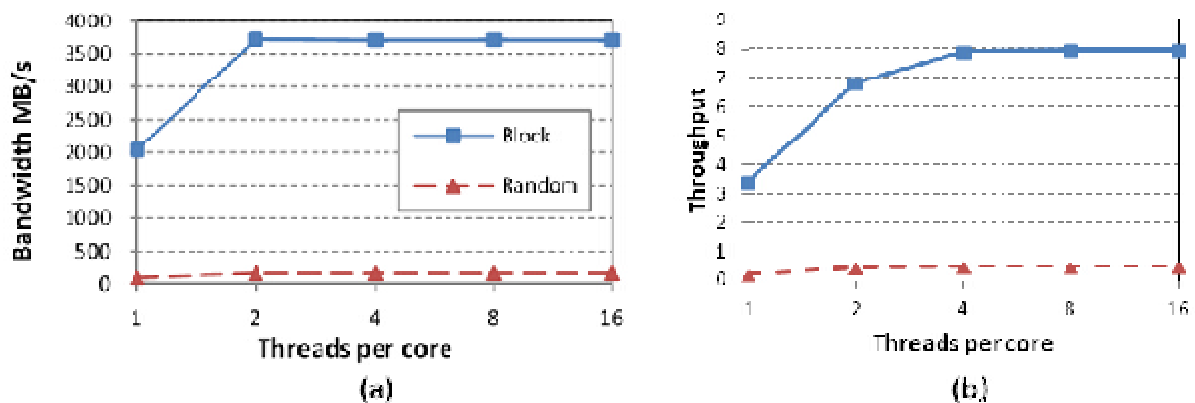


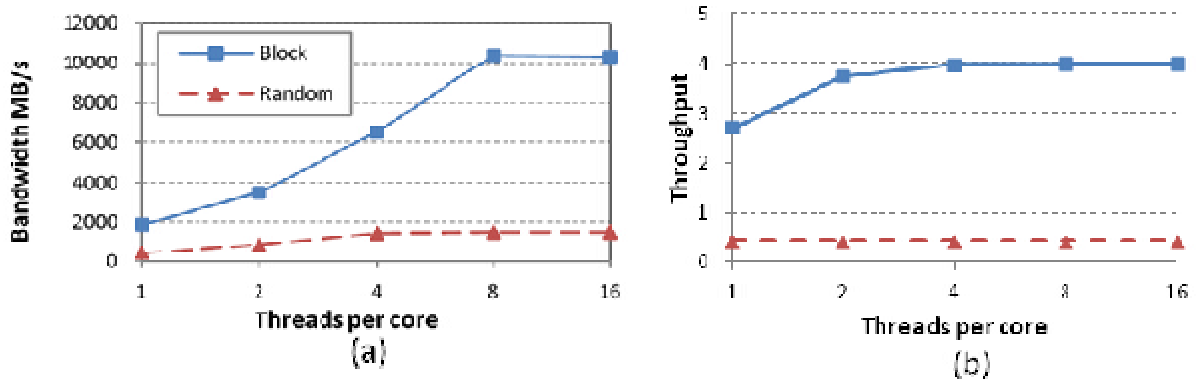Figure 1. Power 6 for Block/Random-test: (a) results in [1] and (b) from our framework

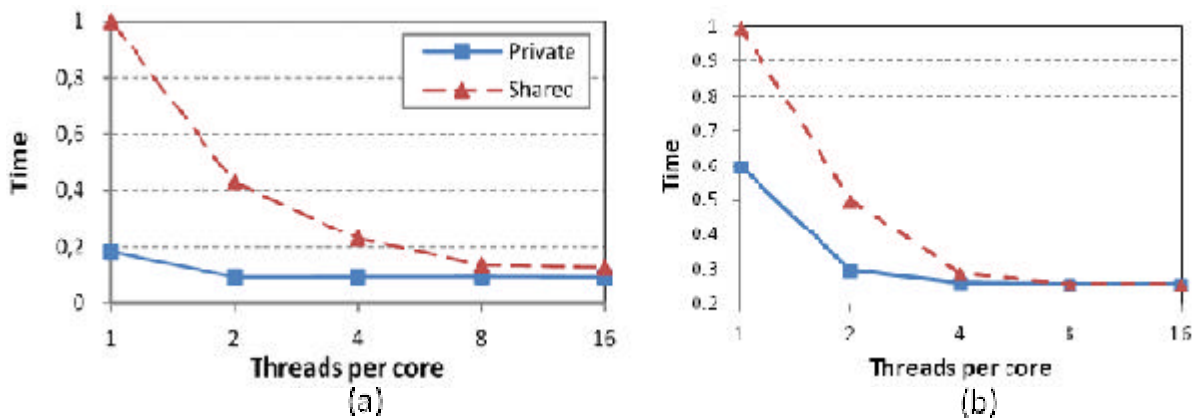Figure 2. T2 for Block/Random-test: (a) results in [1] and (b) from our framework



Figure 3. Power 6 for Private/Shared-test: (a) results in [1] and (b) from our framework
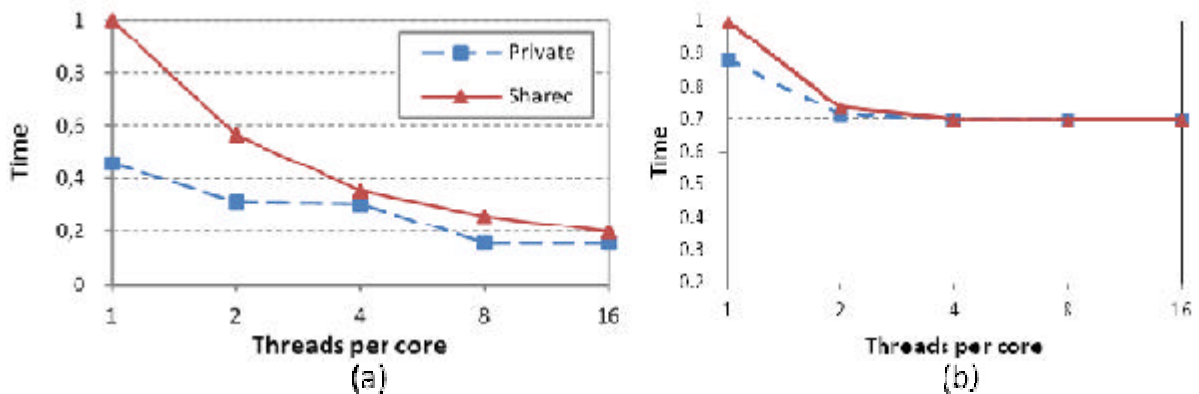


Figure 4. T2 for Private/Shared-test: (a) results in [1] and (b) from our framework

[1] E. Barjrovic and E. Mehofer. *Experimental Study of Multithreading to Improve Memory Hierarchy Performance of Multi-core Processors for Scientific Applications.* In international Conference on Complex, Intelligent and Software Intensive Systems, pages 645-650, Mar. 2009 645-650