

# An $O(1)$ Time Complexity Software Barrier

Liqun Cheng and John B. Carter  
School of Computing  
University of Utah  
50 South Central Campus Drive, Room 3190  
Salt Lake City, UT 84112  
{*legion, retrac*}@cs.utah.edu

## Abstract

*As network latency rapidly approaches thousands of processor cycles and multiprocessors systems become larger and larger, the primary factor in determining a barrier algorithm's performance is the number of serialized network latencies it requires. All existing barrier algorithms require at least  $O(\log N)$  round trip message latencies to perform a single barrier operation on an  $N$ -node shared memory multiprocessor. In addition, existing barrier algorithms are not well tuned in terms of how they interact with modern shared memory systems, which leads to an excessive number of message exchanges to signal barrier completion.*

*The contributions of this paper are threefold. First, we identify and quantitatively analyze the performance deficiencies of conventional barrier implementations when they are executed on real (non-idealized) hardware. Second, we propose a queue-based barrier algorithm that has effectively  $O(1)$  time complexity as measured in round trip message latencies. Third, by exploiting a hardware write-update (PUT) mechanism for signaling, we demonstrate how carefully matching the barrier implementation to the way that modern shared memory systems operate can improve performance dramatically. The resulting optimized algorithm only costs one round trip message latency to perform a barrier operation across  $N$  processors. Using a cycle-accurate execution-driven simulator of a future-generation SGI multiprocessor, we show that the proposed queue-based barrier outperforms conventional barrier implementations based on load-linked/store-conditional instructions by a factor of 5.43 (on 4 processors) to 93.96 (on 256 processors).*

**Keywords:** shared memory multiprocessors, synchronization, barriers, write update, coherence protocols.