World Academy of Science, Engineering and Technology
International Journal of Computer and Systems Engineering
Vol:9, No:03, 2015

Software Transactional Memory in a Dynamic Programming Language at Virtual Machine Level

Authors : Szu-Kai Hsu, Po-Ching Lin

Abstract : As more and more multi-core processors emerge, traditional sequential programming paradigm no longer suffice. Yet only few modern dynamic programming languages can leverage such advantage. Ruby, for example, despite its wide adoption, only includes threads as a simple parallel primitive. The global virtual machine lock of official Ruby runtime makes it impossible to exploit full parallelism. Though various alternative Ruby implementations do eliminate the global virtual machine lock, they only provide developers dated locking mechanism for data synchronization. However, traditional locking mechanism error-prone by nature. Software Transactional Memory is one of the promising alternatives among others. This paper introduces a new virtual machine: GobiesVM to provide a native software transactional memory based solution for dynamic programming languages to exploit parallelism. We also proposed a simplified variation of Transactional Locking II algorithm. The empirical results of our experiments show that support of STM at virtual machine level enables developers to write straightforward code without compromising parallelism or sacrificing thread safety. Existing source code only requires minimal or even none modi cation, which allows developers to easily switch their legacy codebase to a parallel environment. The performance evaluations of GobiesVM also indicate the difference between sequential and parallel execution is significant.

Keywords : global interpreter lock, ruby, software transactional memory, virtual machine
Conference Title : ICPP 2015 : International Conference on Parallel Processing
Conference Location : Paris, France
Conference Dates : March 30-31, 2015