Overall Project Title: **Center for Programming Models for Scalable Parallel Computing**
Coordinating PI: **Rusty Lusk, ANL**
Subproject Title: **Future Programming Models**
PI: **Guang R. Gao**
Reporting Period: **Final Report**

Executive Summary (Final):

The mission of the pmodel center project is to develop software technology to support scalable parallel programming models for terascale systems. The goal of the specific UD subproject is in the context developing an efficient and robust methodology and tools for HPC programming. More specifically, the focus is on developing new programming models which facilitate programmers in porting their application onto parallel high performance computing systems.

During the course of the research in the past 5 years, the landscape of microprocessor chip architecture has witnessed a fundamental change – the emergence of multi-core/many-core chip architecture appear to become the mainstream technology and will have a major impact to for future generation parallel machines. The programming model for shared-address space machines is becoming critical to such multi-core architectures. Our research highlight is the in-depth study of proposed fine-grain parallelism/multithreading support on such future generation multi-core architectures. Our research has demonstrated the significant impact such fine-grain multithreading model can have on the productivity of parallel programming models and their efficient implementation.

Research Areas

The research activities of the first 4 years are already summarized in their individual annual reports.
Here we will be focused on the 5th year of 2005-2006, the Delaware team has continued on the following areas:

1. Continue to conduct and finish in-depth performance studies to demonstrate the performance scalability and portability of a fine-grain multithreaded program execution model (such as the EARTH model) across multiple parallel execution platforms.
2. Continue and finish the design of an experimental infrastructure/platform for development and evaluation of research in future programming models.
3. Continue and to finish the study of parallel programming construct that can simplify the use of fine-grained synchronization, while delivering scalable parallelism by using a weak memory consistency model.

4. Continue and finish the development of Open64 compiler framework and use it as our research for mapping high-level programming language constructs and demonstrate the performance improve on selected target architecture platforms.

Highlight of Achievements

The research achievement of the first 4 years is already summarized in their individual annual reports.
Here we will be focused on the research achievement 5th year of 2005-2006. As a highlight, the Delaware team has accomplished in the following areas:

For development and evaluation of future programming models, we are first looking into an industry de facto standard programming model for shared memory systems based on the emerging class of multi-core chip architectures – OpenMP. Based on our work on TiNy Threads (TNT), a thread virtual machine that, as a case study, efficiently maps the thread model directly to the Cyclops-64 (C64) ISA features, we ported and optimized an open source OpenMP (and our extensions) implementation to the C64 chip architecture. We have developed a mapping strategy that explores the opportunities to optimize OpenMP programs on a state-of-the-art multi-core chip architecture such as C64. Specifically, we consider the following three areas: (1) a memory aware runtime library that places frequently used data structures in scratchpad memory to ensure fast access; (2) a unique spin lock algorithm for shared memory synchronization based on in-memory atomic instructions and native support for thread level execution; (3) a fast barrier that directly uses C64 hardware support for collective synchronization.

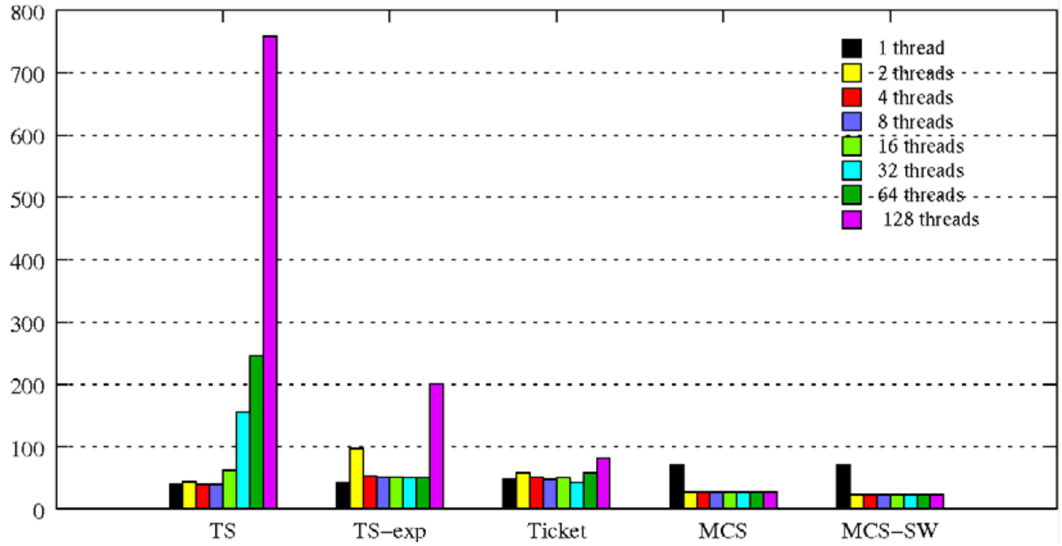

Figure 1: Overhead (cycles) of Spin Lock Algorithms

For example, we customized the well-known MCS spin lock algorithm to a new extension called MCS-SW to C64 using the C64 efficient sleep/wakeup support. Instead of spinning, a thread waiting for a lock goes to sleep after it adds itself into the linked-list based waiting queue. When the lock owner releases the lock, it wakes up its successor by sending a wakeup signal. MCS-SW also uses less memory space than the original implementation. As shown in Figure 1, the new MCS-SW algorithm shows several cycles lower overhead than the Original MCS for more than one thread. Additionally, MCS-SW executes much less instructions per lock/release pair. Thus it consumes less power. In summary, MCS-SW is a more efficient spin lock algorithm for C64 in all three aspects of interest: time, memory, and power.

All three optimizations together result in an 80% overhead reduction for language constructs in OpenMP (example in Figure 2). We believe that such a significant reduction in the cost of managing parallelism makes OpenMP more amenable for writing parallel programs on the C64 platform.

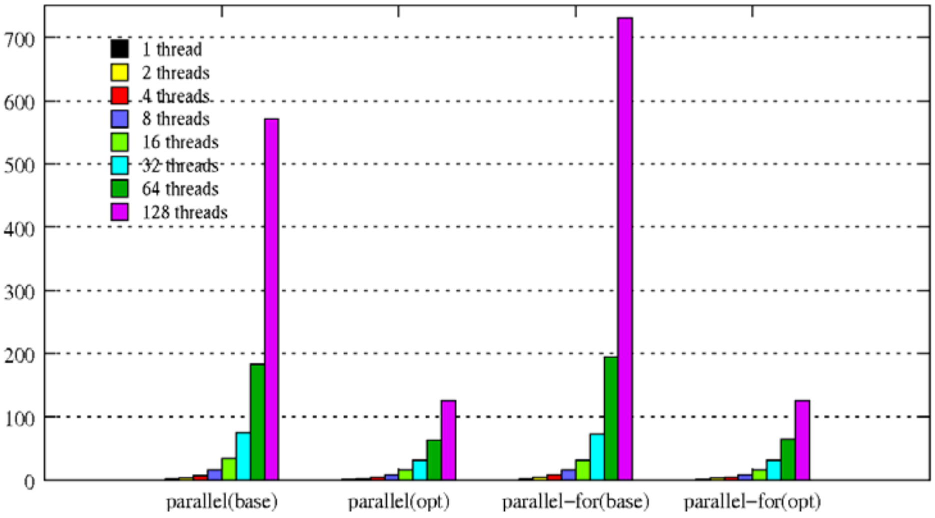

Figure 2: Overhead (us) of OpenMP parallel and parallel for (base: direct porting; opt: optimized)

Moreover, to increase our understanding of the behavior and performance characteristics of OpenMP programs on multi-core architectures, we have studied the performance of basic OpenMP language constructs on a multi-core chip architecture such as the C64 architecture. Compared with previous work on conventional SMP systems, the overhead of OpenMP language constructs on C64 is at least one order of magnitude lower. For example, when 128 threads are executing a barrier concurrently with our OpenMP implementation on C64, the overhead is only 983 cycles (see Figure 3). In comparison, a barrier performed with tens of threads normally takes tens of thousands of cycles on a conventional commodity SMP system.

The impact of this work on future programming models has been discussed in Prof. Gao's Keynote address at the International OpenMP Workshop (IWOMP) held in Reims, France, June 12-13, 2006.

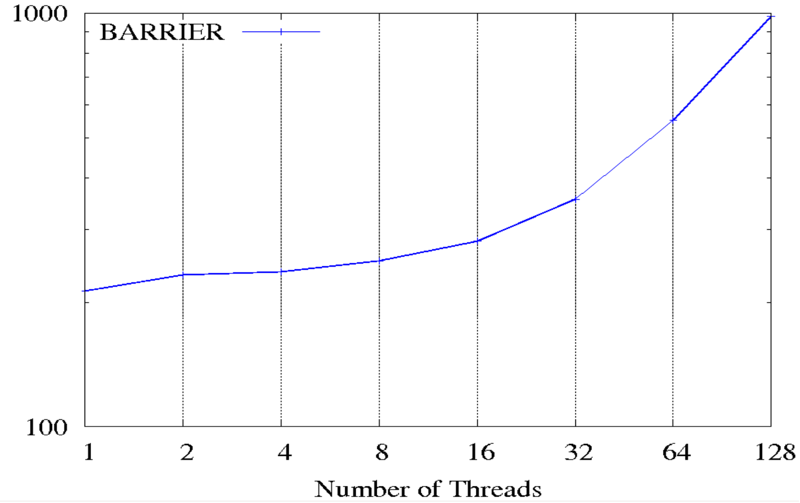

Figure 3: Overhead (cycles) of barrier

References and publications (2005-2006)

[CuvilloEtal05] Juan B. del Cuvillo, Weirong Zhu, Ziang Hu and Guang R. Gao, TiNy Threads: a Thread Virtual Machine for the Cyclops64 Cellular Architecture, 5th Workshop on Massively Parallel Processing (WMPP05), April 4-8, 2005 in Denver, Colorado.

[CuvilloEtal06.1] Juan del Cuvillo, Weirong Zhu, Ziang Hu and Guang R. Gao, Toward a Software Infrastructure for the Cyclops-64 Cellular Architecture, the 20th International Symposium on High Performance Computing Systems and Applications (HPCS2006), St. John's, Newfoundland and Labrador, Canada May 14-17, 2006

[CuvilloEtal06.2] Juan del Cuvillo, Weirong Zhu and Guang R. Gao, Landing OpenMP on Cyclops-64: An Efficient Mapping of OpenMP to a many-core System-on-a-chip, the 3rd ACM International Conference on Computing Frontiers (CF2006), Ischia, Italy May 2-5, 2006.

[Gao] Guang R. Gao, Landing OpenMP on Multi-Core Chips - Challenges and Opportunities, keynote talk, the 2nd International Workshop on OpenMP (IWOMP2006), Reims, France, June 12-15, 2006 [zhuEtAl06] Weirong Zhu, Juan del Cuvillo, and Guang R. Gao, Performance Characteristics of OpenMP Language Constructs on a Many-core-on-a-chip Architecture, the 2nd International Workshop on OpenMP (IWOMP2006), Reims, France, June 12-15, 2006

References and Publications (2002-2005)

In the context of the reported work, we have included some publications from previous years as well.

Refereed Journal Publications

[YangEtA05]Hongbo Yang, R. Govindarajan, Guang R. Gao, Ziang Hu, Improving Power Efficiency with Compiler-Assisted Cache Replacement, Will appears in Journal of Embedded Computing, 2005

[ZhuEtA04]Weirong Zhu, Yanwei Niu, Jizhu Lu, Chuan Shen, and Guang R. Gao, A Cluster-Based Solution for High Performance Hmmpfam Using EARTH Execution Model, International Journal on High Performance Computing and Networking, Accepted to appear in Issue No. 3, September 2004.

[ThulasiramanEtA04] Parimala Thulasiraman, Ashfaq A. Khokhar, Gerd Heber, Guang R. Gao, A Fine-Grain Load Adaptive Algorithm of the 2D Discrete Wavelet Transform for Multithreaded Architectures, Journal of Parallel and Distributed Computing (JPDC), Vol.64, No.1, Pages: 68-78, January 2004.

[RamasswamyEtA03] Govind. Ramasswamy, Hongbo Yang, José N Amaral, Chihong Zhang and Guang R. Gao, Minimum Register Instruction Sequencing to Reduce Register Spills in Out-of-Order Issue Superscalar Architectures, in IEEE Transactions on Computers, Vol. 52, No. 1, Pages: 4-20, January 2003.

[ThulasiramanEtA03] Parimala Thulasiraman, Kevin B. Theobald, Ashfaq A. Khokhar and Guang R. Gao, Efficient Multithreaded Algorithms for the Fast Fourier Transform, The Journal of Supercomputing, Vol. 26, Pages: 43-58, 2003.

[TremblayEtA03] Guy Tremblay, Christopher J. Morrone,José N Amaral and Guang R.Gao, Implementation of the EARTH programming model on SMP clusters: a multi-threaded language and runtime system , Concurrency and Computation: Practice and Experience, Vol. 15, No. 9, Pages: 821-844, August 2003.

Refereed Conference Publications

[RongDouGao]Hongbo Rong Alban Douillet Guang R. Gao, Register Allocation for Software Pipelined Multi-dimensional Loops, ACM Symposium 2005 Conference on Programming Language Design and Implementation (PLDI05), Chicago, Illinois, USA, June, 2005

[CuvilloEtA05]Juan Cuvillo, Weirong Zhu, Ziang Hu, Guang R. Gao, TiNy Threads: a Thread Virtual Machine for the Cyclops64 Cellular Architecture, The 19th International Parallel & Distributed Processing System, April 3-8, 2005 in Denver, Colorado

[ZhuEtA05]Weirong Zhu, Yanwei Niu and Guang R. Gao, Performance Portability on EARTH: A Case Study across Several Parallel Architectures, The 4th International Workshop on Performance Modeling, Evaluation, and Optimization of Parallel and Distributed Systems(PMEO-PDS'05), conjuncted with IPDPS 2005, April 4 – 8, 2005 in Denver, Colorado, USA

[ZhangEtA05]Yuan Zhang, Weirong Zhu, Fei Chen, Ziang Hu, Guang R. Gao, Sequential Consistency Revisit:The Sufficient Condition and Method to Reason The Consistency Model of a Multiprocessor-On-A-Chip Architecture, The Twenty-Third IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN 2005) Innsbruck, Austria, February 15 . 17, 2005

[NiuHuGao04]Yanwei Niu, Ziang Hu and, Guang R. Gao, Parallel Reconstruction for Parallel Imaging SPACE RIP on Cellular Computer Architecture, The 16th IASTED International Conference on PARALLEL AND DISTRIBUTED COMPUTING AND SYSTEMS (PDCS 2004), Cambridge, MA, USA, November 9-11, 2004.

[StoutchininGao04]Arthur Stoutchinin and Guang R. Gao, If-Conversion in SSA Form, Euro-Par 2004, Pisa, Italy, Aug. 31~ Sept. 3, 2004

[ChenTheGao04]Fei Chen, Kevin B. Theobald, and Guang R. Gao. Implementing Parallel Conjugate Gradient on the EARTH Multithreaded Architecture, 2004 IEEE International Conference on Cluster Computing (CLUSTER 2004), San Diego, CA, September, 2004

[RongEtA04.1] Hongbo Rong, Zhizhong Tang, R.Govindarajan, Alban Douillet, Guang R.Gao, Single-Dimension Software Pipelining for Multi-Dimensional Loops, 2004 International Symposium on Code Generation and Optimization with Special Emphasis on Feedback-Directed and Runtime Optimization (CGO-2004), in Palo Alto, California, March 20-24, 2004.

[RongEtA04.2] Hongbo Rong, Alban Douillet, R. Govindarajan, Guang R.Gao, Code Generation for Single-Dimension Software Pipelining of Multi-Dimensional Loops, 2004 International Symposium on Code Generation and Optimization with Special Emphasis on Feedback-Directed and Runtime Optimization (CGO-2004), in Palo Alto, California, March 20-24, 2004.

[SakaneEtA03] Hirofumi Sakane, Levent Yakay, Vishal Karna, Clement Leung and Guang R. Gao, DIMES: An Iterative Emulation Platform for Multiprocessor-System-on-Chip Designs, IEEE International Conference on Field-Programmable Technology (FTP'03), Tokyo, Japan, December 15-17, 2003.

[HuXiaGao03] Ziang Hu, Yan Xie, Guang R. Gao, Code size oriented memory allocation for temporary variables, Fifth Workshop on Media and Streaming Processors (MSP-5/MICRO-36), San Diego, California, December 1, 2003.

[ZhuEtA03] Weirong Zhu, Yanwei Niu, Jizhu Lu, Chuan Shen and Guang R. Gao, A Cluster-Based Solution for High Performance Hmmpfam Using EARTH Execution Model, Fifth IEEE International Conference on Cluster Computing (Custer2003), Hong Kong, P.R. China, December, 2003.

[HuEtA03] Ziang Hu, Yuan Zhang, Yan Xie, Hongbo Yang, Guang. R. Gao, Weirong Zhu and Haiping Wu, Code Size Reduction with Global Code Motion, Workshop on Compilers and Tools for Constrained Embedded Systems (CTCES/CASES) 2003, San Jose, California, Oct. 29, 2003.

[CuvilloEtA03] Juan del Cuvillo, Xinmin Tian, Guang Gao and Millind Girkar, Performance Study of a Whole Genome Comparison Tool on a Hyper-Threading Multiprocessor, Fifth International Symposium on High Performance Computing, Tokyo, Japan, October 20-22, 2003.

[MarquezGao03] Andres Marquez and Guang R. Gao, CARE: Overview of an Adaptive Multithreaded Architecture, Fifth International Symposium on High Performance Computing, Tokyo, Japan, October 20-22, 2003.

[YangEtA03] Hongbo Yang, Govind. Ramasswamy, Guang R. Gao and Ziang Hu, Compiler-Assisted Cache Replacement: Problem Formulation and Performance Evaluation, 16th International Workshop on Languages and Compilers for Parallel Computing(LCPC'03), College Station, Texas, October, 2003

Technical Reports

[CuvilloEtA04] Juan Cuvillo, Ziang Hu, Weirong Zhu, Fei Chen, Guang R. Gao , Toward a Software Infrastructure for the Cyclops64 Cellular Architecture, CAPSL Technical Memo 055, Department of Electrical and Computer Engineering, University of Delaware, April 26, 2004

[SarkarGao04] Vivek Sarkar and Guang R. Gao, Analyzable Atomic Sections: Integrating Fine-Grained Synchronization and Weak Consistency Models for Scalable Parallelism, CAPSL Technical Memo 052, Department of Electrical and Computer Engineering, University of Delaware, February 09, 2004