Gasimo: A Global Address Space Simulation Model
(Poster Abstract)

Worawan Marumgsith
Department of Computer Science,
Faculty of Science and Technology,
Thammasat University, Pathum Thani, 12121 Thailand
+66 2986 9157, +66 813 444 027
wmrs@cs.tu.ac.th

Roland N. Ibbett
Edinburgh Parallel Computing Centre,
University of Edinburgh,
Edinburgh EH9 3JZ UK
+44 131 650 5030
R.N.Ibbett@ed.ac.uk

ABSTRACT
The partitioned global address space (PGAS) programming model
has gained attention as a robust model suitable for a diversity of
emerging concurrent architectures. PGAS offers more scalability
over the former distributed shared memory system (DSM) by
supporting asynchronous execution based on message passing.
Combining asynchronous communication with the facility to
make the location of data transparent, applications written in
PGAS languages have to trade off the benefits of concurrent
architectures with the overhead caused by accessing distant
memories.

Here we present an effective simulation model to reflect the cost
of distant memory accesses on a PGAS system. The model, called
Gasimo, simulates a generic PGAS execution environment on top
of a cluster of homogeneous dual-core machines. Gasimo is a
parallel extension of a particular DSM simulator, called
DSiMCluster, which has been implemented on top of a discrete
event simulation (DES) engine known as HASE.

Categories and Subject Descriptors
I.6.5 [Simulation and Modeling]: Model Development –
Modeling methodologies; I.6.8 [Simulation and Modeling]:
Types of Simulation – Parallel; C.1.2 [Processor Architectures]:
Multiple Data Stream Architectures (Multiprocessors) – Multiple-
instruction-stream, multiple-data-stream processors (MIMD).

General Terms
Design, Performance, Experimentation, Verification

Keywords
Parallel discrete-event simulation (PDES), Multi-core, OpenMP,
Partitioned Global address space (PGAS), Simulation Model

1. INTRODUCTION
The emergence of concurrent architectures such as clusters of
symmetric multiprocessors, heterogeneous accelerators, large
core-count integrated machines, and multithreaded multi-core
machines have created a demand for a suitable, robust
programming model such as a partitioned global address space
(PGAS). Similar to the concept of distributed shared memory
systems (DSM), PGAS allows applications to access data in a
logically shared address space by abstracting away the distinction
of physical memory location. PGAS offers more scalability over
the DSM by supporting asynchronous execution using message
passing. Combining asynchronous communication with the
facility to make the location of data transparent, applications
written in PGAS languages have to trade off the benefits of
concurrent architectures with the overhead caused by accessing
distant memories [1]. This creates a requirement for an efficient
tool to analyse such performance tradeoffs.

Current research attempts to simulate a diversity of concurrent
architectures such as a simulation of clusters of symmetric
multiprocessors [2], simulation of heterogeneous accelerators [3],
simulation of large core-count integrated machines [4] and
simulation of multithreaded multi-core machines [5]. However, a
model which could reflect the performance tradeoffs of the PGAS
system is not yet available.

Here we propose Gasimo, a simulation model of a PGAS system
on top of a cluster of homogeneous dual-core machines. The
proposed model is a parallel extension of a particular DSM
simulator, DSiMCluster, which has been implemented on top of a
discrete event simulation (DES) engine known as HASE.

Figure 1 The Gasimo 4x4 Dual-Core Cluster Model

2. THE GASIMO MODEL
Gasimo is a parallel extension of a DSM Simulator called
DSiMCluster [2] built on top of HASE1, a legacy DES simulation
framework. The model has been developed on Windows 7 with
Intel Software Tools2 and integrated with the HASE-III
environment as shown in Figure 1.

Permission to make digital or hard copies of all or part of this work for
personal or classroom use is granted without fee provided that copies are
not made or distributed for profit or commercial advantage and that
copies bear this notice and the full citation on the first page. To copy
otherwise, or republish, to post on servers or to redistribute to lists,
requires prior specific permission and/or a fee.

2.1 HASE Framework and DSiMCluster

The Gasimo model is built on the DES simulation framework named HASE, a Hierarchical computer Architecture design and Simulation Environment. HASE provides some mechanisms to support the following four steps in modeling a simulation: (1) model design, (2) construction of a simulation executable, (3) experimental control to set parameters and run a simulation and (4) tracing and post-mortem animation. Several research and teaching models have been implemented using HASE.

DSiMCluster is a reconfigurable model emulating a system with multiple shared caches with hierarchical coherence such as clusters of distributed shared memory systems. The simulator comprises an on-the-fly verification and has been shown to give a correct reflection of memory characteristics [6]. DSiMCluster is a sequential simulation model that runs parallel workloads by multithread interleaving to emulate a multithreaded runtime environment.

2.2 Parallel extension in Gasimo

Despite the extensibility of the DSiMCluster, its sequential implementation does not exploit the parallelism of recent multi-core host machine. To address this limitation, we implement Gasimo by parallelising DSiMCluster in three parts. First, the main search loops in DSiMCluster library routines including the instruction set emulation module, cache controller and translation look-aside buffer have been parallelized using OpenMP, a standard shared memory programming model. Second, the behavior file implementing the processor entity has been modified to include OpenMP constructs in order to simulate a homogeneous dual-core processor. Two uni-processors are composed in a compound entity. Each compound entity is implemented as two parallel threads working in different functions (using OpenMP Sections). This is to emulate multiple instruction streams, multiple data stream (MIMD) execution. Third, the description and layout of the system entities include the new coupled processor entities. Figure 2 depicts the steps of the Gasimo extension and its integration into the HASE framework.

3. Gasimo 4x4 Dual-Core Cluster Model

We have tested the correctness of the Gasimo model by running a test program (LU) written in MPI and OpenMP, using 4 processes each of which creates 8 threads. Gasimo has been configured to represent a cluster of symmetric multiprocessor (SMP) machines consisting of 32 cores (see Table 1). Each SMP is an eight-core machine made up of four dual-core processors sharing the same physical memory. Four SMPs networked together on a single bus and were made into a single system by using the page-based global address space technique. Our preliminary test results show that the model can produce the correct output.

<table>
<thead>
<tr>
<th>Table 1. Gasimo configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component</td>
</tr>
<tr>
<td>Processing Elements</td>
</tr>
<tr>
<td>Cache</td>
</tr>
<tr>
<td>Main Memory</td>
</tr>
<tr>
<td>Virtual Shared Memory</td>
</tr>
<tr>
<td>Cluster</td>
</tr>
<tr>
<td>Interconnection</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

4. FUTURE WORKS

We are verifying and evaluating the model against measurement results. After verification, we plan to carry out memory analysis experiments on Gasimo using legacy benchmarks.

5. REFERENCES


