

CCGRID 2003 Poster:

Design and Implementation of a Grid-Enabled BSP

Tong Weiqin
wqtong@mail.shu.edu.cn

Ding Jingbo
dingjingbo@etang.com

Cai Lizhi
llfclz@163.com

School of Computer Engineering and Science, Shanghai Univ.

Shanghai 20072, China

1 What's BSP

BSP is an abbreviation of Bulk Synchronous Parallel. The BSP model is a generalization of the widely researched PRAM model and was initially proposed by L.G. Valiant as a Bridging Model for Parallel Computation.

A BSP computer consists of the following:

- a set of processor-memory pairs
- a communications network that delivers messages point-to-point
- a mechanism for the efficient synchronisation of all, or a subset, of the processors.

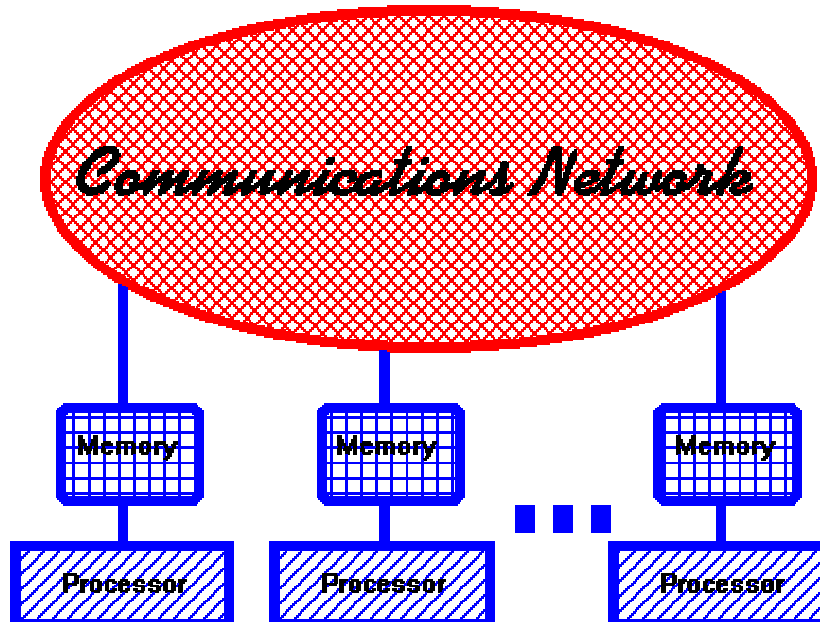


Fig. 1. The BSP Computer

A BSP program consists of sequential supersteps. Each superstep consists of a computation in each processor, using only locally-held values; a global message transmission from each processor to any set of the others; and a barrier synchronization.

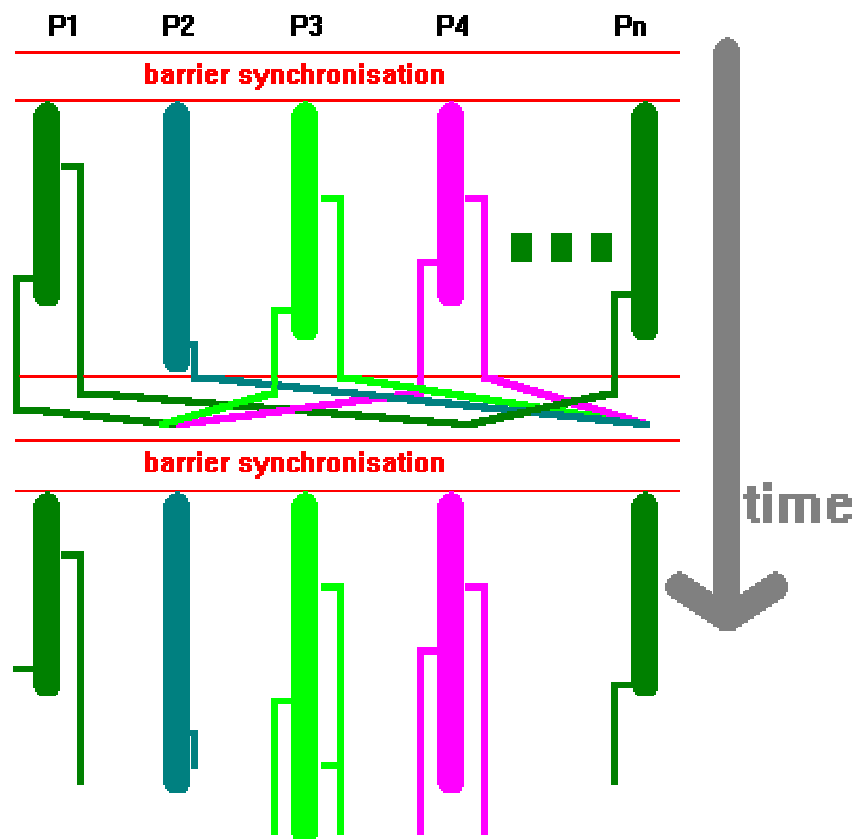


Fig. 2. Superstep of BSP Program

2 What's BSP-G

In a word, BSP-G is an implementation of BSP model that allows user to run a BSP program on computing grid. It uses services provided by the Globus Toolkit for authentication, authorization, resource allocation, executable staging, and I/O, as well as for process creation, monitoring, and control.

The goal of BSP-G is to provide a dynamic computing Virtual Organization (VO) to perform parallel BSP application on Grid Environment.

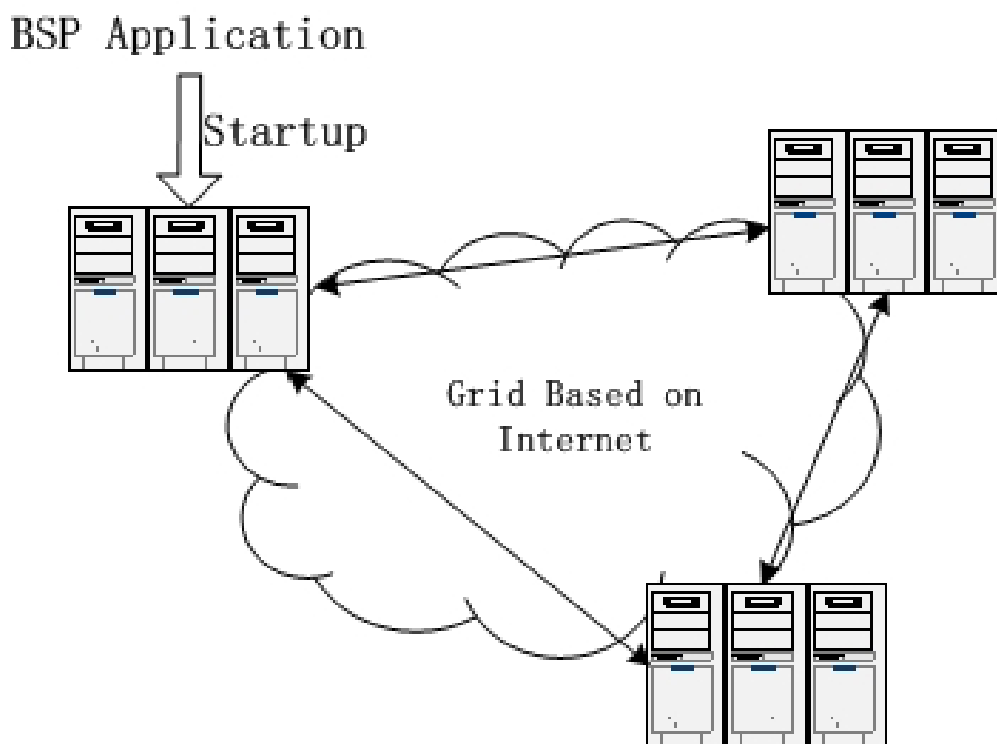


Fig. 3. Computing VO on Grid

3 The Need of Developing BSP-G

Tools and environment of the computing Grid that hide or enable the heterogeneity of different computing resources provide a lot of protocols and service for users to exploit the computing Grid. It is desirable to build various popular programming libraries such as PVM, MPI and BSPlib for portability of the programs on computing Grid. BSP model is widely accepted because of tremendous advantages such as prediction of performance, avoidance of deadlock. So it's necessary to do research on BSP-G.

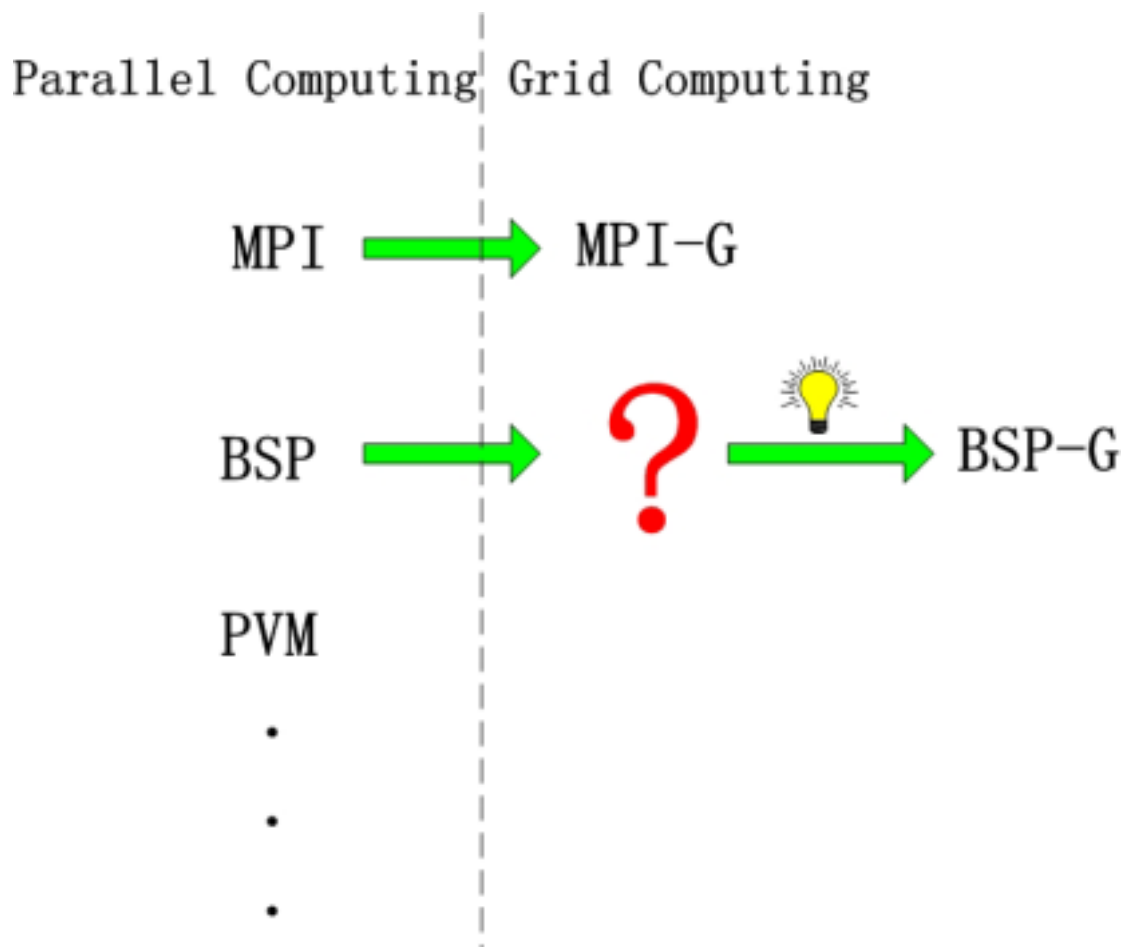


Fig. 4. Idea of Developing BSP-G

4 Architecture of BSP-G

BSP-G implements the core BSPlib according to the BSP standard proposed by BSP worldwide using Globus Toolkit services to support efficient and transparent execution in heterogeneous Grid environments.

BSP-G is based on Grid Services to implement the core library.

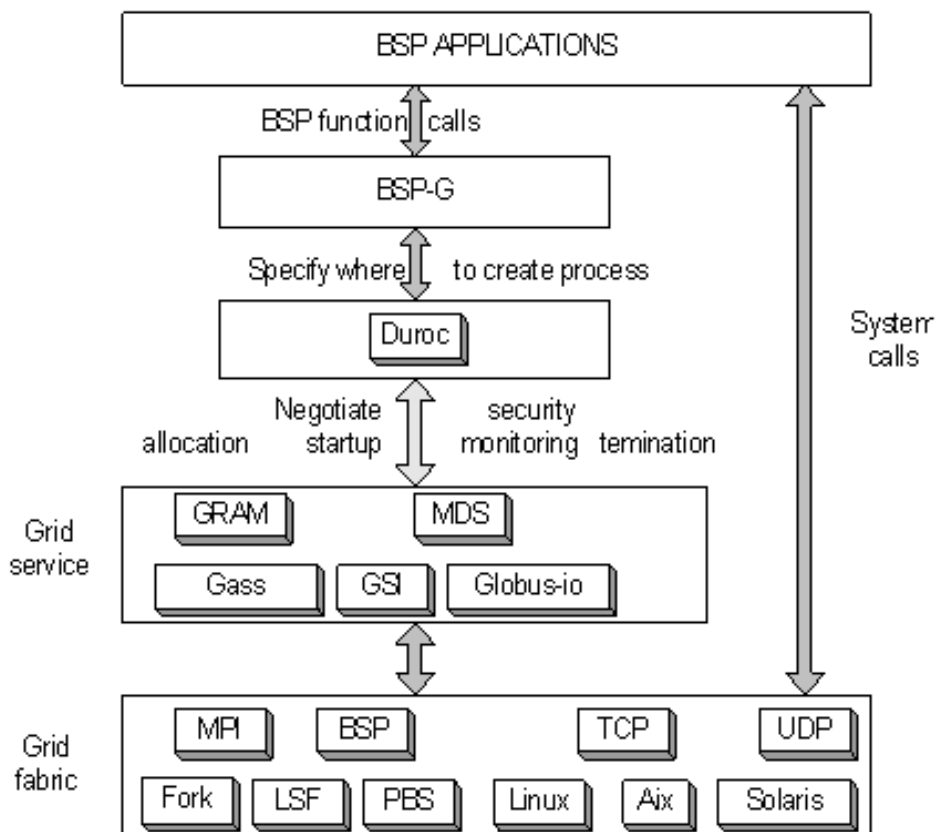


Fig. 5. BSP-G Architecture

5 Experiment Result

To quantify the performance of the BSP-G library, we have designed some small experiments on a pairs of PC. Each PC has dual Pentium III 550MHz processors with 128Mbyte RAM connected via 100Mbps Ethernet, running Redhat Linux 7.2 SMP version. The BSP-G was built using a nonthreaded, no-debug Globus toolkit 2.0. The BSP library we use for reference is Pub 7.0.

The BSP model simplifies a parallel machine by three components, a set of processors, an interconnection network, and a barrier synchronizer, which are parameterized as $\{p, s, g, l\}$. Parameter p represents the number of processors; s is the rate at which computation can be performed by each processor. The parameter g is the permeability of the communications system to continuous traffic between uniformly random destinations. l is the time required for the barrier synchronization. The time required for a superstep is given by:

$$\text{Time for superstep } i = \max_i(s * w_i) + \max_i(h_i * g) + l$$

Where h_i is the size of the h -relation realized in this superstep by process i .

These parameters are obtained with the BSP benchmark program in the package of BSPEDUpack provided by Rob Bisseling [18], a library of numerical algorithms written in C language according to the BSP model, using the BSPlib standard library. The benchmark program tries to expose

the worst behavior of a system. The performance is expected to be better in real applications.

shows that the l parameter of BSP-G library is very close to that of PUB library. Because the l parameter is much larger than the g parameter, and also much larger than the ghi in most cases, the cost of network latency is determined by the l parameter. Except the g parameter, the total cost of BSP-G in a superstep is very close to that of PUB library.

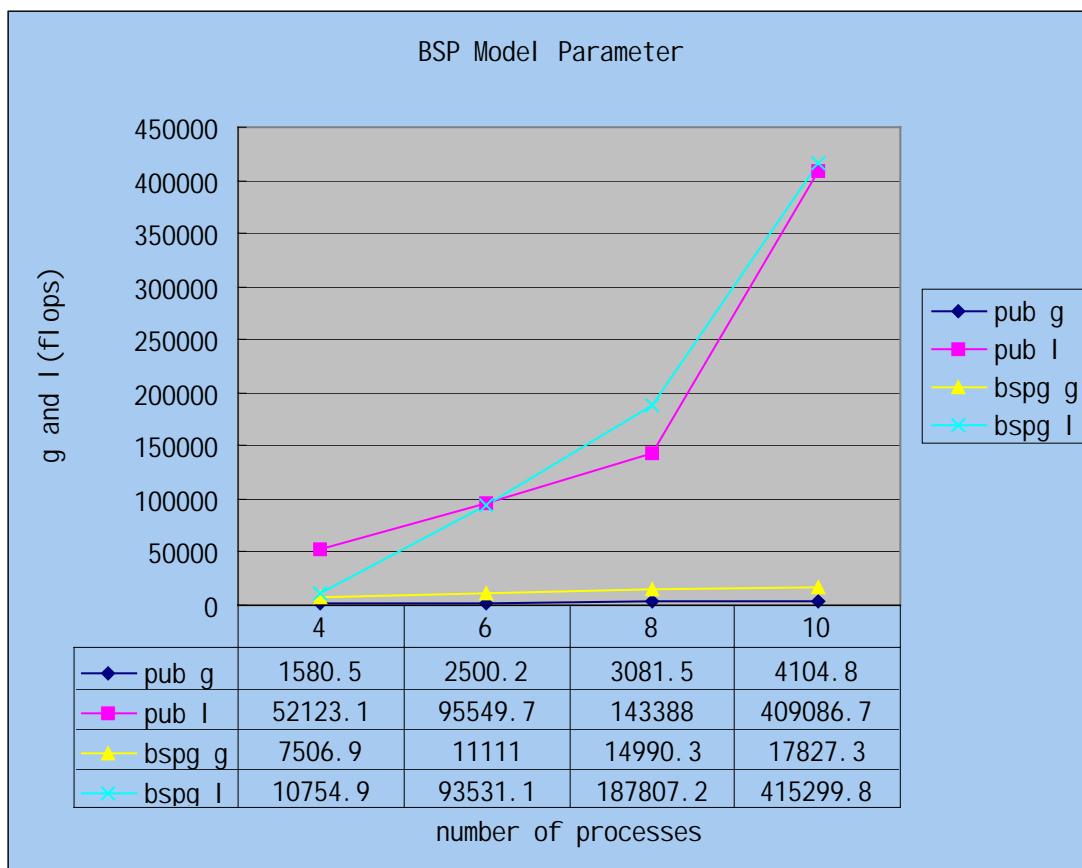


Fig. 6. Experiment Result of BSP-G and PUB

6 Trace of OGSA

To trace the OGSA (Open Grid Service Architecture) proposal, we are now developing a new implementation of BSPlib. It is based on OGSA. In this new architecture, we deem local computing as a service. Communication is also implemented by corresponding Grid Service.

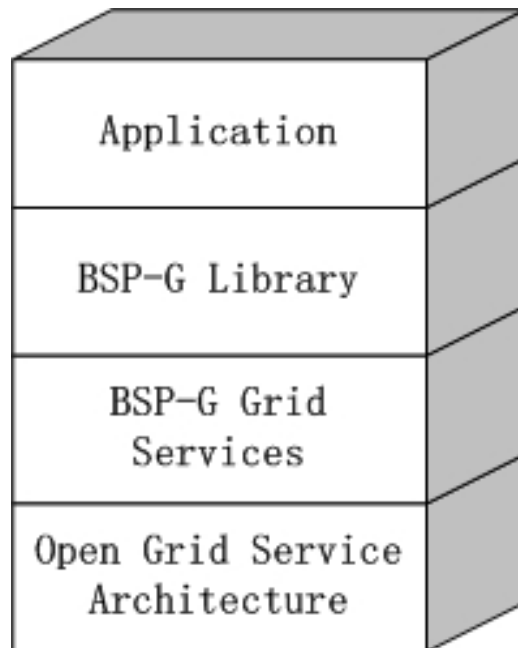


Fig. 7. BSP-G based on OGSA

7 Conclusion and Future Work

We have presented an implementation of Grid-enabled BSP called BSP-G. BSP-G exploits the services provided by Globus toolkit 2.0 to enable and hide the heterogeneity of different computation resource. Users can run a BSP program on the Grid but do not need to care about the detail the resource. The benchmark shows the performance of our implementation is very close to that of PUB library.

We also give much concern on the trace of OGSA proposal. New version of our BSP-G will come out soon.