DLB: A Framework to improve load balance of hybrid applications and maximize resource utilization

Marta Garcia-Gasulla
marta.garcia@bsc.es

Victor Lopez
victor.lopez@bsc.es

Jesús Labarta
jesus-labarta@bsc.es

Raul Sirvent
raul.sirvent@bsc.es

Julita Corbalan*
juli@ac.upc.edu

Barcelona Supercomputing Center
Barcelona, Spain

ABSTRACT

We present DLB: Dynamic Load Balancing Library. DLB is a framework to improve the efficient use of the computational resources of a computational node. The DLB framework includes a load balancing algorithm LeWI (Lend When Idle). The main idea of LeWI is to use the computational resources assigned to a process. Or thread when it is idle, to speed up another process running on the same node that it is still doing computation.

CCS CONCEPTS

• **Computing methodologies** → **Parallel programming languages**; *Distributed programming languages*; **Parallel programming languages**;

KEYWORDS

Load Balance, hybrid, HPC, parallel computing, MPI, OpenMP

1 INTRODUCTION

DLB offers a dynamic solution to load balance MPI applications. DLB is applied at runtime and does not need previous information to solve load imbalance, for this reason it can deal with load imbalances coming from any source. Within DLB is implemented a load balancing algorithm: LeWI (Lend When Idle)[1][2]. LeWI is based on the idea that the computational resources of an MPI process waiting in an MPI blocking call are idle. Therefore, these resources can be used to speed up the execution of another process running on the same node.

To achieve this, we will use the malleability of OpenMP to change the number of threads. DLB will intercept the MPI calls and change the number of OpenMP threads when necessary. When a process reaches a blocking MPI call, it will lend its CPUs to the system, and another MPI process running on the same node will be able to use those CPUs. When the MPI process finishes the blocking call, it will retrieve its resources from the borrowing process.

The DLB framework also includes a module called DROM (Dynamic Resource Ownership Manager). The DROM module offers an interface for resource managers (job managers). The provided interface allows changing the resources assigned to a process. With this module we address to main scenarios. On one hand, allow a more efficient use of resources when load imbalance occurs across nodes, ie. Start applications on unused resources. On the other hand, allow to reassign resources from a long running application to run a high priority and short application, i.e. Start a visualization in the resources of a simulation.

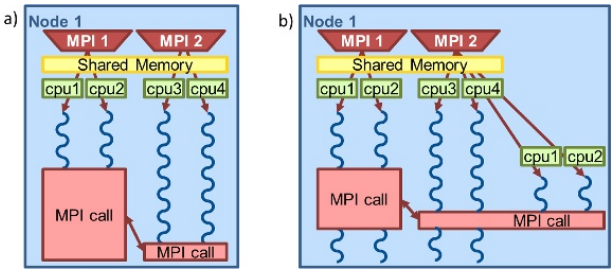

Figure 1: (a) Unbalanced App. (b) Balanced App. with LeWI

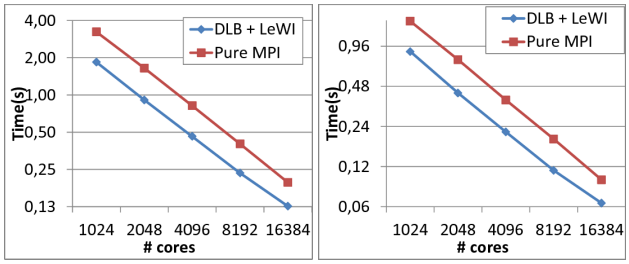

Figure 2: DLB scalability in MN3 upt to 1024 nodes

We will present the performance evaluation of DLB and LeWI when applied at a simulation code for computational mechanics: Alya. We will also demonstrate that LeWI can scale up to thousands of cores showing scalability tests performed up to 16.000 cores (1024 nodes) of Marenostrum3.

ACKNOWLEDGMENTS

Work supported by the Spanish Government (grant SEV2015-0493), by the Spanish Ministry of Science and Innovation (contract TIN2015-65316), by Generalitat de Catalunya (contract 2014-SGR-1051), by the Intel-BSC Exascale Lab project and from the European Union's Horizon 2020 under grant agreement No 720270 (HBP SGA1).

REFERENCES

[1] M. Garcia, J. Corbalan, and J. Labarta. 2009. LeWI: A Runtime Balancing Algorithm for Nested Parallelism. In *2009 International Conference on Parallel Processing*. 526–533. https://doi.org/10.1109/ICPP.2009.56

[2] Marta Garcia, Jesus Labarta, and Julita Corbalan. 2014. Hints to improve automatic load balancing with LeWI for hybrid applications. *J. Parallel and Distrib. Comput.* 74, 9 (2014), 2781 – 2794. https://doi.org/10.1016/j.jpdc.2014.05.004

*Also with Universitat Politecnica de Catalunya.

womENcourage'2017, September 2017, Barcelona, Spain
2017. ACM ISBN none.
https://doi.org/none