Applied Mathematics and Computation 231 (2014) 544–559

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Supporting adaptive and irregular parallelism for non-linear numerical optimization

P.E. Hadjidoukas [a,*], C. Voglis [b], V.V. Dimakopoulos [b], I.E. Lagaris [b], D.G. Papageorgiou [c]

[a] Chair of Computational Science, ETH Zurich, Zurich CH-8092, Switzerland
[b] Department of Computer Science, University of Ioannina, GR-45110 Ioannina, Greece
[c] Department of Materials Science and Engineering, University of Ioannina, GR-45110 Ioannina, Greece

ARTICLE INFO

Keywords:
Irregular and multilevel parallelism
Adaptive task parallelism
Multicore clusters
Message passing
Numerical differentiation
Numerical optimization
Protein conformation

ABSTRACT

This paper presents an infrastructure for high performance numerical optimization on clusters of multicore systems. Building on a runtime system which implements a programming and execution environment for irregular and adaptive task-based parallelism, we extract and exploit the parallelism of a Multistart optimization strategy at multiple levels, which include second order derivative calculations for Newton-based local optimization. The runtime system can support a dynamically changing hierarchical execution graph, without any assumptions on the levels of parallelization. This enables the optimization practitioners to implement, transparently, even more complicated schemes. We discuss parallelization details and task distribution schemes for managing nested and dynamic parallelism. In addition, we apply our framework to a real-world application case that concerns the protein conformation problem. Finally, we report performance results for all the components of our system on a multicore cluster.

1. Introduction

Numerical optimization is an indispensable tool that has been widely applied on many scientific problems. An important feature of optimization problems is their long execution time which is attributed to the high computational demands and the possibly multiple local minima of the objective function to minimize. There are several applications where the time for a single function call is substantial. Parallelization can drastically reduce the required processing time to find a solution. The inherent parallelism of these problems can be found at various levels, including function or gradient evaluations, linear algebra calculations and the optimization algorithms themselves. Global optimization algorithms that can take advantage of parallel and distributed architectures are particularly suitable for solving problems with high computational requirements. The emerging multi-core architectures provide a cost-effective solution for high-performance optimization.

Although many parallel local and global optimization algorithms were proposed in the last decades [1–6], only a handful of actual systems exist. One of the most widely used scientific software programs, MATLAB, presented its first parallel optimization solution in 2009 [7]. In the pioneer work of [8] an interval global optimization method is implemented using dynamic load balancing. In [9] the authors present ParaGlobSol, a parallel global optimization package written in Fortran 90/95 with MPI to perform inter process communication and a dynamic load balancing scheme. PGO [10] is a general parallel computing based on the Genetic Algorithm. In PGO, the parallel (and heterogeneous) computing framework is organized as a global master–slave system using a central database management system for storing all the data during optimization

[*] Corresponding author.
E-mail addresses: phadjido@mavt.ethz.gr (P.E. Hadjidoukas), voglis@cs.uoi.gr (C. Voglis), dimako@cs.uoi.gr (V.V. Dimakopoulos), lagaris@cs.uoi.gr (I.E. Lagaris), dpapageo@cc.uoi.gr (D.G. Papageorgiou).

0096-3003/$ - see front matter © 2014 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.amc.2013.12.092

progress. Oriented in interoperability, the MHGrid platform [11] exploits meta-heuristics based search methods and Grid computing to enable the transparent sharing of heterogeneous and dynamic resources offering a versatile Global optimization framework. MANGO [12] is a middleware that involves the development of an extensible and flexible multiagent platform, in which autonomous agents can solve global optimization problems in cooperation. PaGMO [13] is a recently released open source multi-threaded software that offers a plethora of local and global optimization codes exploiting modern multi-core architectures. Finally in [2,14] the authors present VTDIRECT95 a parallel implementation of the DIRECT algorithm, using a three-level hierarchical parallel scheme.

In this work we study the parallelization of Multistart [15] method which is a standard and widely used scheme for dealing with global optimization problems. According to this method, a local optimization procedure is applied to a number of randomly selected points. The Multistart method forms the basis for many successful global optimization methods and any parallelization study on it can be easily extended to more elaborate optimization schemes. For local optimization we have chosen the Newton method which is a general and powerful method for multidimensional non-linear optimization that makes use of first and second derivatives of the objective function. This choice, in turn, introduces further computational complexity as derivative estimation via finite differentiation requires a number of function evaluations. In many practical situations analytical calculation of second order derivatives may be prohibitive.

Task-based parallelism, as expressed by the master-worker programming model, can be an effective approach for a cluster-aware implementation of global optimization methods such as Multistart. Function evaluations are mapped to tasks and assigned to workers. The dynamic load balancing of the model further enhances its suitability. A naive implementation of the model, however, cannot meet all the requirements that a parallel global optimization method (Multistart) imposes. First, the large expected number of spawned tasks affect scalability as the single master becomes a bottleneck. Secondly, the exploitation of nested parallelism requires advanced runtime techniques, able to facilitate programming and provide efficient management of processing elements. Ideally, a parallel implementation must target both shared and distributed memory machines, without the burden of explicit message passing for the programmer. Additionally, it is important, from a performance point of view, to have a hardware-independent solution that transparently uses multi-threading to fully exploit the physically shared memory of SMP/multi-core systems.

We first present our Tasking library for Clusters (TORC), a novel general-purpose runtime environment for programming and executing irregular and adaptive task-parallel applications on multi-core SMPS and clusters of such machines. Building on TORC, we design a standalone Parallel Numerical Differentiation Library (PNDL) that provides routines for first and second order derivative approximation. For the latter we extract two levels of parallelism and study several task distribution schemes.

We also present the parallelization of a Newton-based Multistart method using both TORC and PNDL to execute concurrently multiple local optimizations and gradient/Hessian calculations. In contrast to previous approaches, we manage to express and exploit parallelism at all possible levels in a straightforward and seamless manner using a single runtime framework. In addition, the task distribution and stealing mechanisms of TORC provide efficient load balancing without the need for explicit partitioning of processors.

In contrast to other infrastructure, and with the exception of VTDIRECT95, our proposal is the only one that supports hierarchical and multi-level task parallelism. What differentiates our approach from VTDIRECT95 implementation, is that our hierarchical execution graph is changing dynamically unlike the static three level scheme applied in [14]. In addition, our system is platform-agnostic supporting transparently both shared and distributed memory architectures.

By integrating a molecular modeling software package [16] with our system, we are able to apply our framework to a real application case that deals with the protein conformation problem, that is the problem of determining the three dimensional structure of a protein. It is a fundamental problem in biophysical sciences with applications in drug design and in genetic information decoding.

We present an experimental evaluation on a dedicated homogeneous multicore cluster, providing results at both intranode and cluster-wide levels using a single application executable. The performance results with synthetic benchmarks and applications demonstrate the efficiency of our system.

Summarizing, the contributions of this paper are:

- a runtime library for task-based computations of multicore clusters, which allows for extraction and exploitation of dynamic, hierarchical and multilevel parallelism in global optimization algorithms,
- a novel high-performance implementation of the Multistart method using the above infrastructure, over different architectures, which allows multiple numerical derivative computations to be deployed concurrently,
- a highly efficient application of the optimization algorithm involved in the protein folding problem.

The rest of this paper is organized as follows: Section 2 gives an overview of the non-linear global optimization problem. Section 3 discusses the inherent parallelism structure of Multistart and introduces the organization of our software infrastructure. Section 4 outlines the programming interface and some implementation details of TORC. Section 5 focuses on PNDL and Multistart, while Section 6 presents the protein conformation problem. Experimental evaluation is reported in Section 7. We conclude with a discussion in Section 8.