Comparing Spark with MapReduce: 
Glowworm Swarm Optimization 
Applied to Multimodal Functions

Goutham Miryala, North Dakota State University, Fargo, USA 
Simone A. Ludwig, Department of Computer Science, North Dakota State University, Fargo, USA

ABSTRACT

Glowworm swarm optimization (GSO) is one of the optimization techniques, which need to be parallelized in order to evaluate large problems with high-dimensional function spaces. There are various issues involved in the parallelization of any algorithm such as efficient communication among nodes in a cluster, load balancing, automatic node failure recovery, and scalability of nodes at runtime. In this article, the authors have implemented the GSO algorithm with the Apache Spark framework. Even though we need to address how to distribute the data in the cluster to improve the efficiency of algorithm, the Spark framework is designed in such a way that one does not need to deal with any actual underlying parallelization details. For the experimentation, two multimodal benchmark functions were used to evaluate the Spark-GSO algorithm with various sizes of dimensionality. The authors evaluate the optimization results of the two evaluation functions as well as they will compare the Spark results with the ones obtained using a previously implemented MapReduce-based GSO algorithm.

KEYWORDS

Apache Spark, Glowworm Swarm Optimization, Hadoop, MapReduce

1. INTRODUCTION

Optimization is a process whereby the best possible solution to a particular problem is being sought. Thinking in terms of function optimization, one is usually interested in finding the minimum or maximum value of a particular function. There are many different kinds of optimization algorithms and one subgroup belongs to the stochastic algorithms, namely nature-inspired algorithms. Nature-inspired algorithms are algorithms that are inspired by natural processes such as evolution or swarming behavior. Within the nature-inspired algorithm category there are two, namely evolutionary algorithm and swarm intelligence approaches (Engelbrecht, 2007). Evolutionary algorithms take inspirations of processes in nature such as natural selection, crossover and mutation. Examples of evolutionary algorithms are genetic algorithm, evolutionary strategy, genetic programming, etc. (Eiben, 1994). 

Swarm intelligence methods’ core concept is that interacting swarm members are exchanging information to achieve a particular goal, which then is used to find the optimum value of a particular problem. Particle swarm optimization is one of the swarm intelligence methodologies whereby the concept of finding food sources based on the birds’ current movement, the flocks’ best food source ever found, and an individual bird in the flock experiencing the best food source is used. In ant colony optimization, the actions of ants performed for example during the process of finding the shortest path is applied (Stützle, 2009). In particular, the ants’ indirect communication ability such as secreting

DOI: 10.4018/IJSIR.2018070101

Copyright © 2018, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
pheromone on various paths is used in the algorithm. Another example of a swarm optimization algorithm is the bee colony optimization algorithm (Wong, Low, & Chong, 2008). In bee colony optimization, the concept of using local and global searching honeybees to build honeybee colonies is applied.

Yet another swarm intelligence algorithm, inspired by the characteristics shown by glowworms, called Glowworm Swarm Optimization (GSO) (Krishnanand & Ghose, 2009a), has been introduced. To mimic goals such as attracting a mate during the breeding season, the glowworms govern the emission of light. Application areas of GSO are for example, hazard sensing in ubiquitous environments (Krishnanand & Ghose, 2008), robotics and portable sensor networks (Krishnanand & Ghose, 2005), and data clustering (Aljarah & Ludwig, 2013b). All these applications can make use of the GSO algorithm due to its simplicity and small number of parameters that are required for tuning (Krishnanand & Ghose, 2009a) (Krishnanand & Ghose, 2008) (Krishnanand & Ghose, 2005).

GSO has been applied to many different application areas, however, one area that GSO is well suited for are multi-modal function optimization (Barrera & Coello, 2009). The aim of multi-modal function optimization is to find all maxima (peaks) or minima (valleys) given some constraints. However, one problem multi-modal function optimization suffers from is the relatively long runtime of the algorithm when searching for the peaks/valleys. In particular, the running time of the algorithm is significantly increased when the peak count is increased for higher dimensional spaces. The count of swarm members is also increased in order to have a better chance to find all the peaks or valleys in higher dimensional spaces. One approach to speed up the processing is to divide the search into several groups as to carry out the optimization process. Thus, a parallelized solution is required to achieve the optimization task in a feasible amount of time.

One solution that has been implemented using the MapReduce framework was proposed in (Aljarah & Ludwig, 2013b). The benefit of the MapReduce framework is that no knowledge of parallel programming is required to implement the parallelization. A conceptual understanding of MapReduce allows one to program a parallel implementation. Besides that, the MapReduce framework comes with built-in fault-tolerance, load balancing, and data locality.

The latest parallelization technology is Spark (Apache Spark™, 2017). The Spark development was initiated by the UC Berkeley RAD lab that started as a research project in 2009. The main goal behind the development is to provide iterative in-memory computing support for MapReduce. Companies like Yahoo, Databricks and Intel are major contributors to the Spark development (Matei Zaharia, Holden Karau, Andy Konwinski, 2015). Spark is used for data intensive applications that companies such as Facebook work with (Apache Spark @Scale, 2017).

Apache Spark can handle large sets of data and scale well by increasing the number of nodes at runtime. Apache Spark applications can be developed in various programming languages including Java, Python, Scala and R. A Spark application in general is mainly operated by a driver program that controls parallel operations on a cluster. A Resilient Distributed Dataset (RDD) is the main abstraction provided by Spark, which is a collection of elements that can be partitioned and distributed over a cluster of nodes for parallel processing. RDD provides reusability through caching capabilities and node failure recovery.

In this paper, a parallel GSO algorithm is proposed using Apache Spark. The following contributions are presented implementing parallel glowworm swarm optimization on Spark (Spark-GSO):

- Apache Spark concepts have been successfully applied to Glowworm Swarm Optimization to enable parallelization.
- Higher dimensional multimodal functions have been evaluated in Spark-GSO and compared with the MR-GSO algorithm.
CSMA: Context-Based, Service-Oriented Modeling and Analysis Method for Modern Enterprise Applications
[www.igi-global.com/article/csma-context-based-service-oriented/43008?camid=4v1a](www.igi-global.com/article/csma-context-based-service-oriented/43008?camid=4v1a)

Microwave Circuit Design
Gabriel Cormier and Tyler Ross (2013). *Swarm Intelligence for Electric and Electronic Engineering* (pp. 18-39).
[www.igi-global.com/chapter/microwave-circuit-design/72821?camid=4v1a](www.igi-global.com/chapter/microwave-circuit-design/72821?camid=4v1a)