School of STEM, Computing & Software Systems, UWB

CSS 534 Assignment 5

Yangde Li

December 11, 2018

Contents

0.1 Introduction 2
0.2 Parallelism Techniques 2
0.3 Performance Analysis 2
0.4 Programmability Analysis 2
0.5 Execution Snapshots or Demo with Graphical Results 2

0.1 Introduction

This report covers an assignment on implementing Ant Colony Optimization with MpiJava.

This is a compulsory exercise in the course CSS 534 Parallel Programming, given by the school of computing & software systems at the University of Washington, Bothell in 2018.

0.2 Parallelism Techniques

In this implementation with MpiJava, we mainly focus on two types of parallelism techniques: **Task** and **Data** decomposition. For the task decomposition, we split the ants to different computing nodes, thus to accelerate whole computing speed. In every optimization loop, the worker nodes send back the delta pheromone generated by the ants in this iteration to master, then master updates the whole pheromone matrix and scatter it to all workers. Another parallel technique is to make every worker run a complete ACO process and update the pheromone matrix using mean value, this is a way to increase the ants searching space and increase the possibility of finding the best path.

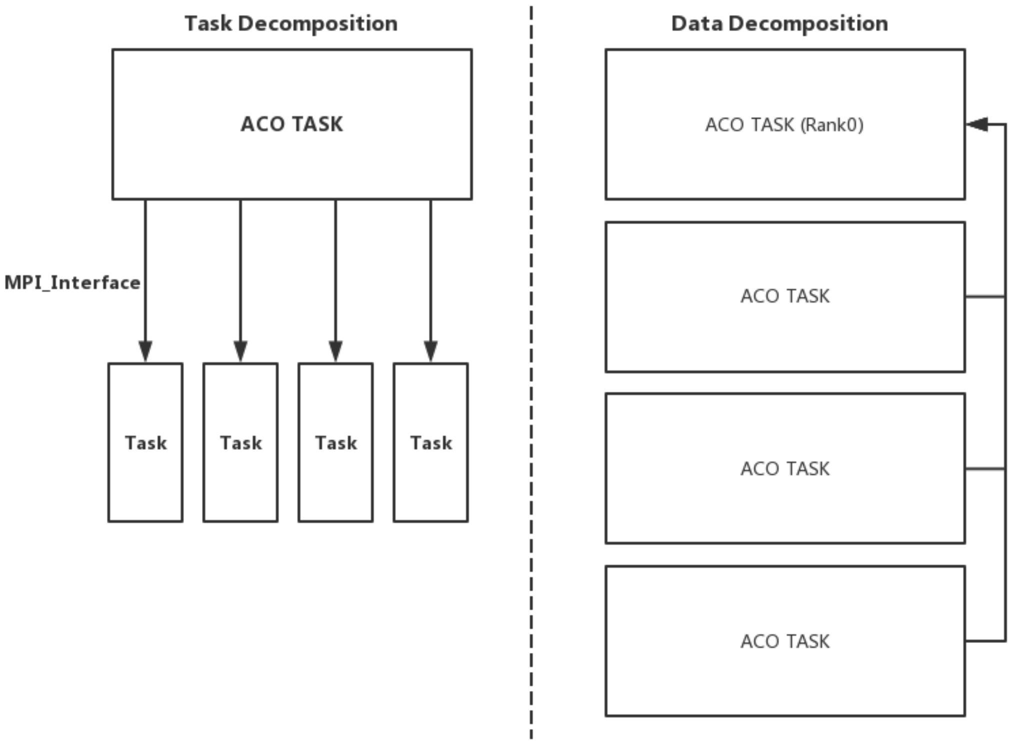

Figure 1: Two types of Parallel Pattern

0.3 Performance Analysis

For task parallelization, computing task is split into different nodes and the exchanged data is rather small so the performance improvement is obvious. However, the Data Decomposition duplicates the whole data set to run in every node, so the performance improvement should be little.

0.4 Programmability Analysis

0.5 Execution Snapshots or Demo with Graphical Results

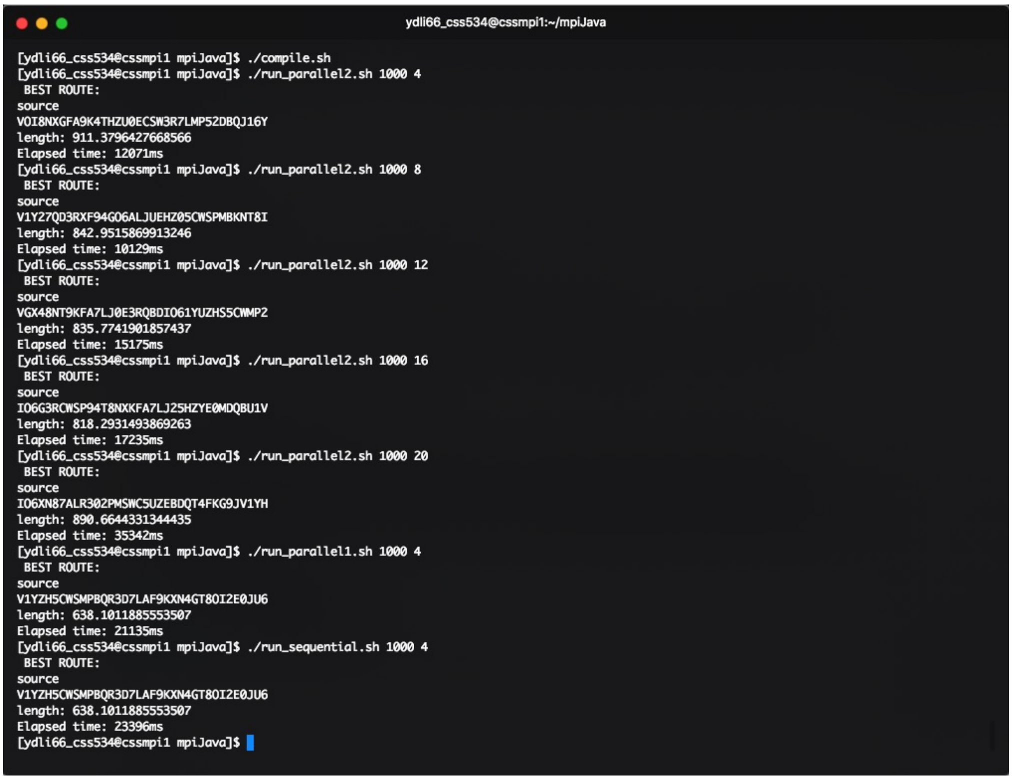

Figure 2: Execution Snapshots