Available online at www.sciencedirect.com
ScienceDirect
Mathematics and Computers in Simulation 137 (2017) 424–430
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original articles

Parallelizing drainage network algorithm using free software: Octave as a solution

J.F. Reinoso-Gordo[a], M.J. Ibáñez[b], R. Romero-Zaliz[c,*]

[a] Department of Architectonic and Engineering Graphic Expression, University of Granada, Spain
[b] Department of Applied Mathematics, University of Granada, Spain
[c] Department of Computer Science and Artificial Intelligence, University of Granada, Spain

Received 27 October 2015; received in revised form 21 June 2016; accepted 16 September 2016
Available online 29 September 2016

Abstract

Drainage network is a product, normally derived from a DEM (digital elevation model), widely used in environmental and civil engineering, and particularly in hydrology. The computation requirements increase exponentially as the size DEM increases, limiting the applications when a fast analysis is necessary. This is specially noticeably when working with multiple flow direction (MFD) drainage networks. Nowadays, some solutions have been explored, but focusing on Graphical Processing Units (GPU) technology. We propose a CPU-based approach which has the advantage that uses free software such as Octave and MPI wrap for it. Our parallelized algorithm not only improved the time computation but also allows adaptive behavior to different cluster settings.

Keywords: Drainage network; DEM; Flow accumulation; Parallel algorithm; MPI

1. Introduction

The drainage networks and the watershed are essential features derived from digital terrain models (DTMs), which can be defined as a grid structure or as a TIN (Triangulated Irregular net). The grid is the more frequent structure where each cell has a value representing a height. The cell size range is very variable, for example 60 m in the Landsat Satellite panchromatic band until a few centimeters derived from LiDAR (Light Detection and Ranging) data (see Fig. 1 which shows an orthophoto and its corresponding LiDAR data shaded). The reader can notice how precise the LiDAR data is: trees, houses and roads are perfectly identifiable. So, it is easy to understand that in order to get the drainage network for a functional watershed, the number of cells composing such a watershed can reach rapidly several millions, which means the computation resources have to be really powerful.

The application of drainage network and the watershed host are countless in areas such as Cartography [12], Civil Engineering, Hydrology [16], Environmental Science [8], Ecology, Limnology, Urban Planning, Agriculture and so

* Corresponding author.
E-mail addresses: jreinoso@ugr.es (J.F. Reinoso-Gordo), mibanez@ugr.es (M.J. Ibáñez), rocio@decsai.ugr.es (R. Romero-Zaliz).

http://dx.doi.org/10.1016/j.matcom.2016.09.004
0378-4754/

Fig. 1. Orthophoto (left) and LiDAR shaded DEM (8 points per square meter) (right).

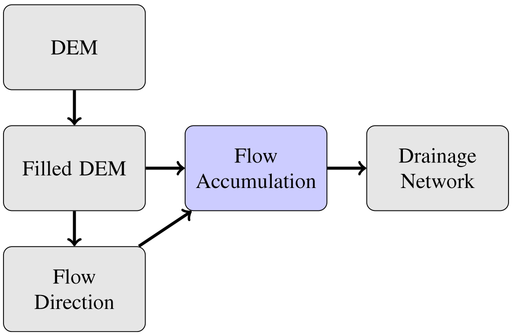

Fig. 2. Workflow in the drainage network algorithm.

on. One instant application is the determination of areas where a flood risk exists, so the decision makers are able to forbid the urban construction in such areas. Another applications may be to decide the place where to build a dam, or planning agricultural policies in order to avoid erosion and the consequent desertification for a particular region.

In the last 30 years, researchers have developed several algorithms to derive drainage network [13,11,5,2,19,3]. These traditional serial algorithms fall insufficient in order to compute drainage networks in a decent time. Therefore, nowadays parallelization becomes an important issue showing as the best option to compute drainage networks. Some solutions have been explored, but focusing on Graphical Processing Units (GPU) technology [14,15].

The proposed parallelized version of the flow accumulation for Multiple Flow Direction (MFD) uses Octave and Message Passing Interface (MPI) package `mpi` to reduce computing time. The MPI version is flexible and can be easily customized for different computer configurations.

2. Methods

In order to implement a parallel version of the flow accumulation algorithm for drainage networks, we will focus on the O'Callaghan and Mark (1984) D8 algorithm [13,10,7], because it is the most suitable algorithm for parallelization. Particularly O'Callaghan is suitable because the existence of several "for" loops which are time consuming and they can be improved by parallelization. Other algorithms need a more complicated partition [4,1].

2.1. Workflow

We propose a flow accumulation parallel algorithm for MFD which is included in the workflow showed in Fig. 2 and it is briefly explained below: