Computers and Geosciences 119 (2018) 49–67

Contents lists available at ScienceDirect

Computers and Geosciences

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

Research paper

Multi-objective unstructured triangular mesh generation for use in hydrological and land surface models

Christopher B. Marsh[a,b,*], Raymond J. Spiteri[a,c], John W. Pomeroy[a,b], Howard S. Wheater[a,b]

[a] Center for Hydrology, Dept. Geography, University of Saskatchewan, Canada
[b] Global Institute for Water Security (GIWS), University of Saskatchewan, Canada
[c] Numerical Simulation Lab, Dept. Computer Science, University of Saskatchewan, Canada

ARTICLE INFO

Keywords:
Hydrology
Mesh generation
Modelling
Unstructured grid
Multi-objective refinement

ABSTRACT

Unstructured triangular meshes are an efficient and effective landscape representation that are suitable for use in distributed hydrological and land surface models. Their variable spatial resolution provides similar spatial performance to high-resolution structured grids while using only a fraction of the number of elements. Many existing triangulation methods either sacrifice triangle quality to introduce variable resolution or maintain well-formed uniform meshes at the expense of variable triangle resolution. They are also generally constructed to only fulfil topographic constraints. However, distributed hydrological and land surface models require triangles of varying resolution to provide landscape representations that accurately represent the spatial heterogeneity of driving meteorology, physical parameters and process operation in the simulation domain. As such, mesh generators need to constrain the unstructured mesh to not only topography but to other important surface and sub-surface features. This work presents novel multi-objective unstructured mesh generation software that allows mesh generation to be constrained to an arbitrary number of important features while maintaining a variable spatial resolution. Triangle quality is supported as well as a smooth gradation from small to large triangles. Including these additional constraints results in a better representation of spatial heterogeneity than from classic topography-only constraints.

1. Introduction

Distributed hydrological and land surface models aggregate the surface and sub-surface into internally homogenous control volumes (Vrugt et al., 2008). These control volumes are used to discretize the mass and energy conservation equations or to apply point-scale models. Correct selection of these control volumes has profound implications for the numerical stability of the discretized equations (Berger and Colella, 1989; Hagen et al., 2000; Parrish and Hagen, 2007; Caviedes-Voullième et al., 2012). Cold regions are characterized by seasonal snowcover and snowfall; here, snow-landscape interactions and energy flux considerations further complicate the selection of control volumes. In these regions, landscape heterogeneity such as vegetation, slope, aspect, and elevation are often critical controls on important processes such as blowing snow (Pomeroy et al., 1997; Essery et al., 1999; Mott et al., 2008), vegetation interactions (Pomeroy et al., 1998; Gelfan et al., 2004; Ménard et al., 2014), snowmelt (Essery and Pomeroy, 2004; Dornes et al., 2008a; Grünewald et al., 2010; Marsh et al., 2012; Debeer and Pomeroy, 2017), and runoff dynamics (Carey and Woo, 2001). Surface heterogeneity is also critical for land-atmosphere interactions (Foken, 2008; Husain et al., 2016). The commonly used fixed-resolution control volume, e.g., raster approach, often has substantial computation burdens (Vivoni et al., 2004; Caviedes-Voullième et al., 2012), as well as high uncertainty when applied to areas of interest for water resources such as mountain watersheds. There is a motivation for a discretization that balances surface heterogeneity, numerical requirements, and a reduction in computational elements for use with hydrological and land-surface models.

Triangular meshes represent the topography via a set of irregularly sized, non-overlapping connected triangles (Chang, 2008). Meshes with variable sized and shaped elements are *unstructured*. Areas of high spatial variability can have a greater density of small triangles than areas that are more homogeneous, providing a more efficient terrain representation than the raster format (Shewchuk, 1996) by reducing computational elements in models by up to 90% (Ivanov et al., 2004). Efficiency increases of this magnitude make distributed modelling approaches more feasible and less uncertain due to reduced parameter sets, initial conditions, and wall-clock time (e.g., Ivanov et al., 2004; Kumar et al., 2009a,b). Due to the widespread availability of raster data, unstructured meshes for hydrology are typically derived from raster digital elevation

* Corresponding author. Center for Hydrology, Dept. Geography, University of Saskatchewan, Canada.
E-mail address: chris.marsh@usask.ca (C.B. Marsh).

https://doi.org/10.1016/j.cageo.2018.06.009
Received 12 September 2017; Received in revised form 30 April 2018; Accepted 25 June 2018
Available online 27 June 2018
0098-3004/

models (DEMs). Because these meshes act as an approximation to landscape variability, care must be taken during creation, and constraints on triangle shape, size, and error to the underlying raster(s) should be included to ensure suitability for hydrological modelling (Caviedes-Voullième et al., 2012; Bilskie and Hagen, 2013).

Two common methods for mesh creation exist: point selection and domain constraints. There are five popular selection algorithms: Heuristic, Hierarchy, Skeleton, and Filter (also known as Very Important Points, VIP) (Lee, 1991; El-Shimy et al., 2005; Chang, 2008), and ArcGIS tools. These share the trait that possible stopping criteria be either a pre-set number of points to be selected or pre-set differences in elevation between the selected point and neighbouring raster cells (Lee, 1991). Importantly, these methods make no guarantees concerning triangle quality. Therefore, long skinny triangles can be created with poor gradations from small to large triangles. These triangles are generally unsuitable for use as a discretization mesh in numerical applications. An example of poor point selection is shown in Fig. 1, where the selected points (green dots) duplicate the structured mesh corners, doubling the number of elements (black lines are triangle edges). In areas of generally flat topography, such as plains or broad valley bottoms, constraining meshes only to topography fails to capture the spatial heterogeneity of hydrologically important characteristics. Alternatively, inner and outer domain boundaries such as basin delineation, streams, and lakes are defined and triangles are inserted to cover the area defined by these boundaries. Triangular mesh generation using this technique is generally done via constrained Delaunay triangulation (Ruppert, 1995; Shewchuk, 2002). Strong guarantees on triangle shape and inner angles ensures suitability for use as a discretization mesh for numerical applications.

In this paper, a multi-objective meshing tool, *Mesher*, is presented. Based on an existing, high-quality implementation of constrained Delaunay triangulation, its novel contribution is in how triangles are chosen for refining. Mesher uses various objective functions to measure triangle error with the underlying primary raster as well as constraining to non-topographic discrete and classified data (e.g., land cover, soils). This permits variably sized triangles throughout the domain, allows for guarantees about triangle quality and shape, and ensures that spatial heterogeneity in secondary features is represented. Specifically, this meshing software is optimized for use in hydrological and land surface models that mix many point-scale and non-PDE (partial differential equation) distributed algorithms along with PDE discretizations. Due to this mixing of methods, meshes are generated considering only the landscape, e.g., elevation, vegetation, and soil, and not the discretization of physical processes such as Hagen et al. (2002) or Parrish and Hagen (2007) who consider numerical error in the mesh generation. This meshing tool is quantitatively tested against an existing mesh generation method, and a surface heterogeneity measure is used to quantify whether important landscape characteristics are well approximated.

2. Meshing algorithm

2.1. Overview

The core meshing algorithm is built upon the constrained Delaunay meshing algorithm of J. Shewchuk (2002), as implemented in the Computational Geometry Algorithms Library (CGAL; Rineau (2016)). In brief, Delaunay meshes constrain triangle inner angles, edge lengths, number of total triangles, and the gradation from small to large triangles in the domain (Shewchuk, 2002). Delaunay meshes have been used with success for a coupled representation of surface-sub-surface processes (Qu and Duffy, 2007) and for shallow water flow equations (Hagen et al., 2001, 2002; Kumar et al., 2009a). Due to the importance of including sub-mesh scale vertical features (Bilskie et al., 2015) as well representing rivers and streams, these constraint features may be included. Boundary and inner feature constraints are defined via planar straight-line graphs (PSLGs). The pre- and post-processing steps, as well as the multi-objective refinement algorithm, are detailed below.

2.2. Details

Outlined in Algorithm 1, the meshing algorithm uses the extent of the DEM to bound the meshing area. All optional secondary input parameters (e.g., vegetation and soils) are converted to the DEM's coordinate system and are clipped to the DEM's spatial extent, allowing mismatched raster resolutions and extents in these data. The data/nodata region of the DEM is used to generate an (optionally simplified) outer PSLG. The *z*-value of the triangle vertices (v_z) are assigned a value from the DEM. However if the PSLG is simplified it may result in a vertex laying outside the original raster extent. These invalid vertices have their *z*-value interpolated from neighbour vertices. More novel DEM to mesh interpolation techniques such as Bilskie and Hagen (2013) could be included if required. These pre-processing steps are done in Python. The core meshing algorithm is written in C++11. All geospatial manipulation is done via the Geospatial Data Abstraction Library(GDAL) (GDAL Development Team, 2016).

```
input  : A digital elevation map DEM
input  : An optional set of secondary constraints parameters P
Extent ← Extent(DEM)
nodata ← NoDataValue(DEM)
Projection ← GetProjection(DEM)
foreach p in P do
    ClipExtent(p, Extent)
    NoDataValue(p)← nodata
    Project(p, Projection)
end
PSLG ← Polygonize(DEM)
mesh ← Mesher(PSLG, DEM, P)
foreach v in Mesh verticies do
    if v outside Extent then
        Interpolate v_z from connected verticies
    end
end
```