Computers & Graphics 37 (2013) 596–607

Contents lists available at SciVerse ScienceDirect

Computers & Graphics

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

SMI 2013

Efficient computation of constrained parameterizations on parallel platforms ☆

Theodoros Athanasiadis [a], Georgios Zioupos [b], Ioannis Fudos [a,*]

[a] Department of Computer Science, University of Ioannina, GR45110 Ioannina, Greece
[b] Department of Mathematics, Aristotle University of Thessaloniki, GR54124 Thessaloniki, Greece

ARTICLE INFO

Article history:
Received 18 March 2013
Received in revised form
29 May 2013
Accepted 31 May 2013
Available online 13 June 2013

Keywords:
Non linear solver
Constrained parameterization
Mesh parameterization
GPU
Massively parallel platforms
Tesselation

ABSTRACT

Constrained isometric planar parameterizations are central to a broad spectrum of applications. In this work, we present a non linear solver developed on OpenCL that is efficiently parallelizable on modern massively parallel architectures. We establish how parameterization relates to mesh smoothing and show how to efficiently and robustly solve the planar mesh parameterization problem with constraints. Furthermore, we demonstrate the applicability of our approach to real-time cut-and-paste editing and interactive mesh deformation.

1. Introduction

The purpose of mesh parameterization is to obtain a piecewise linear map, associating each face of the mesh with a surface patch on the parameterization zdomain. The parameterization domain is the surface that the mesh is parameterized on. Since the geometric shape of the parameterization surface will typically be different than the shape of the original mesh, angle and area distortion is introduced. Maps that minimize the angular distortion are called *conformal*, maps that minimize area distortion are called *authalic*, and maps that minimize distance distortion are called *isometric*. In this work, we deal with constrained *isometric* planar parameterizations. These maps are central to a broad spectrum of applications such as texture mapping, mesh completion, morphing and deformation transfer.

An important goal of parameterization is to obtain bijective (invertible) maps. The bijectivity of the map guarantees that there is no triangle flipping or overlapping. This is an important guarantee for certain applications, especially in the presence of user defined constraints on the vertices. On a planar parameterization domain a map may exhibit local or global bijectivity. Local bijectivity is achieved when there are no local triangle flips in the local neighborhoods of the mesh, whereas global bijectivity is achieved when there is no global mesh overlapping. Generally, global bijectivity is harder to achieve. Nevertheless, for most applications local bijectivity is sufficient.

The existing planar parameterization methods can be classified into two categories (for an extensive survey see [1]) : (i) methods that solve only linear systems, for example [2–4] and (ii) methods that use some kind of non-linear optimization. Typical methods of the former category, especially the earlier ones, have no guarantee for local or global bijectivity and usually offer inferior results as compared to the latter. Nevertheless, they are usually very fast and can be useful even as an initial solution for non-linear approaches. For example, in [5] although the energy minimized is non linear, a linear system is solved to obtain an initial parameterization of the mesh on the plane.

Amongst the latter category, several methods use some form of constrained or unconstrained non-linear optimization. These methods either reformulate the problem (resulting in non linearity) [6,7] or directly minimize a non linear energy term [5,8]. An indicative example is the work of [6] where the parametrization problem is reformulated in terms of angles subject to a set of constraints that ensure planarity and triangle validity of the final parameterization. Another example is the work of [7] where the authors use a set of vertices of the mesh called *cone singularities* to

☆This research has been co-financed by the European Union (European Social Fund—ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF)—Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

* Corresponding author. Tel.: +30 26510 08805; fax: +30 26510 08886.
E-mail addresses: fudos@cs.uoi.gr, ifudos@otenet.gr (I. Fudos).

0097-8493/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cag.2013.05.023

absorb the Gaussian curvature so as to compute conformal parameterizations of meshes. This idea was further extended in [9] where the authors first determine automatically the location and the target curvatures of the singularities. They then proceed by solving a discrete Poisson equation on the mesh vertices to compute edge lengths and compute the final embedding using a linear least squares methodology based on the computed edge lengths. A related work is [10] based also on *cone singularities* where a non linear solver is used to minimize the corresponding metric and compute the final parameterization.

For practical applications there is usually an additional requirement to accommodate user defined or automatically imposed constraints on the vertices of the parameterization. Generally, these constraints can be categorized into two groups: *soft constraints* that are approximately satisfied in the least squares sense and *hard constraints* that are precisely satisfied. Methods based on energy minimization can support soft constraints by adding a quadratic term to the energy function that measures the distance between the vertices and the desired location. Nevertheless, for linear approaches the additional term usually breaks the guarantees for bijectivity even for parameterizations on convex domains [1]. Hard constraints are even more difficult to support. Some methods can be extended to enforce hard constraints by the use of *Lagrange* multipliers [3]. However, such methods do not guarantee parameterization bijectivity.

In a nutshell, many previous approaches employ non linear solvers for constrained or unconstrained non linear optimization targeted to conformal parameterizations ([6,7,10,9]). Others use fast linear solvers (e.g. [4]) to obtain isometric parameterizations but fail to support constraints and local bijectivity. In this work, we deal with the problem of computing an isometric bijective planar parameterization of a mesh, subject to hard constraints. Additionally, soft constraints can be trivially supported due to the formulation of the problem. More specifically, this paper makes the following technical contributions:

- Establishes the relation between mesh smoothing and parameterization techniques and derives a simplified formulation for the *isometric* parameterization problem.
- Presents an efficient parallel implementation of a non-linear solver along with a number of heuristics that speed up substantially the parallel realization on modern hardware.
- Presents an iterative topological untangling process that solves efficiently the constrained parameterization problem.
- Demonstrates the applicability of the parallel solver on realizing the feature cut-and-paste design paradigm.

The rest of the paper is organized as follows. Section 2 offers theoretical background for mesh smoothing and establishes how it is related to parameterization. Section 3 describes the core of our constrained parallel solver for isometric parameterizations. Section 4 presents an application of our solver on cut-and-paste design. Finally, Section 5 offers conclusions.

2. Isometric parameterization

2.1. Mesh smoothing preliminaries

Before explaining the connection between the parameterization and the smoothing problem, we define three element types: (i) the *physical* element which is obtained through a mapping, possibly with area and angle distortion, of an element of the original mesh on the parameterization space, (ii) the *reference* element which is constructed by placing one node at the origin and the other nodes at unit lengths along the cartesian axes, and (iii) the *ideal* element which depends on the desired properties of the final mesh (see [11,12]).

Furthermore, we define two affine mappings. The first mapping from the *reference* element x_r to the *ideal* element x_i is defined as

$$x_i = \mathbf{W} x_r \tag{1}$$

where matrix $\mathbf{W}$ is the edge matrix of the *ideal* element. The second mapping from the *reference* element x_r to the *physical* element x is defined as

$$x = \mathbf{A} x_r + x_0 \tag{2}$$

where matrix $\mathbf{A}$ is the edge (Jacobian) matrix of the *physical* element and x_0 is the vector with the coordinates of the first vertex. The matrix $\mathbf{A}$ holds information about the volume, the area, and the orientation of the *physical* element while x_0 controls its translation.

Based on the above definitions the shape matrix from the *ideal* to the *physical* element was defined in [11] as

$$\mathbf{S} = \mathbf{A}\mathbf{W}^{-1} \tag{3}$$

and the associated barrier *shape* quality metric $(\eta_{shape}) : \mathbb{R}^{n\times n} \rightarrow \mathbb{R}$ as

$$\eta_{shape} = \frac{\|\mathbf{S}\|_{\mathbf{F}}^2}{n\,\det(\mathbf{S})^{2/n}} \tag{4}$$

where for surface and volume meshes n is 2 and 3, respectively. The above metric can be used in an optimization process as an objective function to minimize over the vertices to obtain an optimal mesh. This quality metric assumes that each element has positive and non-zero determinants and consequently non-zero local area or volume. Furthermore, the barrier form is used to enforce positive Jacobian determinants to prevent folding. The mappings are depicted in Fig. 1.

2.2. Shape matrix construction for conformal parameterization

As noted in the previous section the definition of the *ideal* element depends on the desired properties of the final mesh.

Therefore to preserve the angles of a triangle of the original mesh, we define on the parameterization space an *ideal* triangle $\Delta v_0 v_1 v_2$ with the same angles. Moreover for reasons that will become apparent, we define its similar triangle $\Delta v'_0 v'_1 v'_2$ with base $\|v'_0 v'_1\| = 1$ and $v'_0 = v_0$ (see Fig. 1) where

$$\frac{\|v'_0 v'_1\|}{\|v_0 v_1\|} = \lambda, \quad \lambda > 0 \tag{5}$$

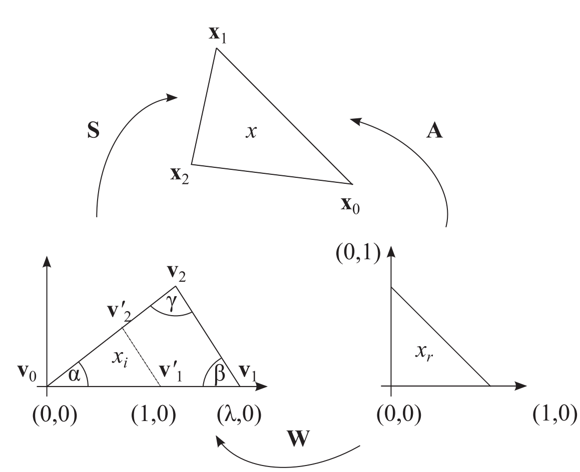

Fig. 1. Ideal triangle $\Delta v_0 v_1 v_2$ and its similar triangle $\Delta v'_0 v'_1 v'_2$ on $\mathbb{R}^2$ along with the corresponding mappings.