Unconstrained Spherical Parameterization

Ilja Friedel Peter Schröder Mathieu Desbrun
Caltech *

Introduction

We present a simple technique for easing the computation of spherical parameterizations by a simple modification of traditional planar parameterization methods: our spherical energies differ from the usual planar quadratic energies only by multiplication by a simple factor based on the inverse distance of each triangle from the sphere center, such as the following:

$$E_{Tutte} = d_{min}^{-2} \cdot ((x_A - x_B)^2 + (x_B - x_C)^2 + (x_A - x_C)^2)$$

$$E_{Dirichlet} = d_{min}^{-2} \cdot (\cot(\alpha) \cdot (x_B - x_C)^2 + \cot(\beta) \cdot (x_A - x_C)^2 + \cot(\gamma) \cdot (x_A - x_B)^2)$$

$$E_{Area} = d_{min}^{-2} \cdot Area^2 \cdot InputArea^{-1}$$

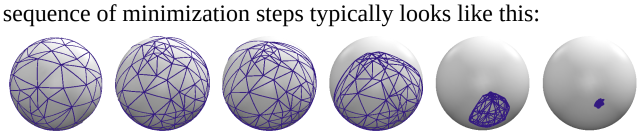
The main motivation for scaling the planar energy by d_{min}^{-2} is to obtain an *upper bound* of the spherical integrals. Intuitively this can be done by measuring the energy of each triangle *after* transforming it into the tangent space of the sphere. As we show next, *this extra term removes the usual need for repeated spherical reprojections or for unnatural point constraints.*

Analysis

In practice the factor d_{min}^{-2} is close to 1. Why is then such a minor correction necessary? Let us examine what happens if we solve the classic flat energies

$$E_{spring} = w_{AB} \cdot (x_A - x_B)^2 + w_{BC} \cdot (x_B - x_C)^2 + w_{AC} \cdot (x_A - x_C)^2$$

as discussed in [Floater and Hormann 2005] on the sphere. A sequence of minimization steps typically looks like this:



As the iterations proceed in the solver a triangle starts growing. Finally it slips over the "equator", eventually shrinking the entire mesh to a point. During this process, the energy is *reduced in each iteration step*, finally reaching its minimum at zero. This failure can be consistently observed. Our conclusion is that the spherical spring energy has no minimum at the expected configuration. Instead the minimizer moves down a continuous slope leading to a collapsed configuration. We cannot fully explain this with Möbius transformations, which are invariants of the continuous setting, but do not leave the discretized energy constant. One common fix for avoiding the complete collapse consists of constraining three or more points. In practice the number of points varies with the input mesh and selected vertices. Each additional point fixed also introduces extra distortion. For these reasons it is desirable to construct a method that does not require any additional constraints.

We decided to analyze the situation for various energies, like $E_{Dirichlet}$ and E_{Area} [Floater and Hormann 2005]. The planar energies *always underestimate* the integrals over spherical triangles. This is easy to see for energies based on areas: the area of a planar triangle cutting through the sphere is always smaller than the area of the corresponding spherical triangle. This is problematic,

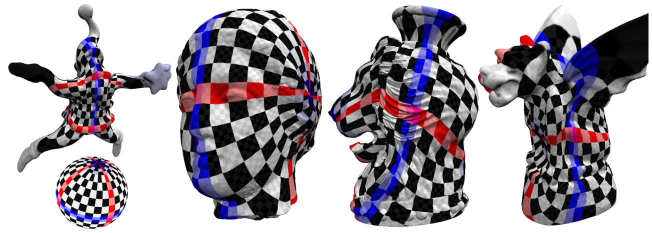
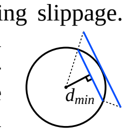
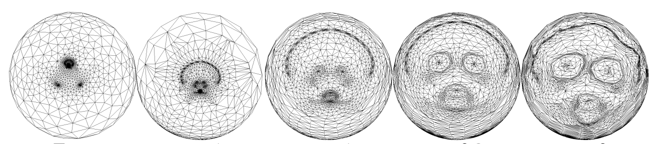


Figure 1: *We computed parameterizations for several large (70k … 400k triangles) meshes using the combined energy $E_{combined}$ with weightings $(1,1)$. No conditions were enforced during the solve, nevertheless the parameterizations are fold-free.*

because the error increases disproportionately with triangle size. Used in a minimization process this is a recipe for disaster: the minimizer can find a way to decrease the energy by increasing the size of the triangle with the largest error, creating slippage. One way to avoid this situation is to design spherical energies that are accurate for small triangles but otherwise *always overestimate* the continuous energy. We show that this can be achieved by using the central projection (or gnomonic map) which projects each flat triangle outwards until it is essentially tangential[1], as done by the division with d_{min}. A further consequence of this construction is the creation of infinite energy barriers for hemispherical triangles.



Many planar energies, like MIPS and stretch, try to trade off angle and area distortion. They can be written as combinations of Dirichlet and area energies. For simplicity we have experimented with weighted averages of these energies $E_{combined} = a \cdot E_{Dirichlet} + b \cdot E_{Area}$ and examples are shown below.



Tutte	$a=1$, $b=0$	$a=1$, $b=1$	$a=0.2$, $b=1$	$a=0$, $b=1$

Results

We have implemented the modified energies presented here. There is no need to write custom solvers; we use TAO [Benson et al.], which provides implementations of standard Newton and trust-region methods. There is no need to define constraints or reprojection during the minimization. The energies are defined in Maple and are automatically differentiated and translated to C++. Run times are within a few minutes for a 70k triangle mesh like igea.

Acknowledgements The Lion-vase model is courtesy of SensAble Technologies, Inc.; all other models Cyberware, Inc.; Viewpoint and H. Hoppe.

References

Benson, S. J., McInnes, L. C., Moré, J., and Sarich, J. TAO user manual. Tech. rep. http://www.mcs.anl.gov/tao.

Floater, M. S., and Hormann, K. 2005. Surface parameterization: a tutorial and survey. In *Advances in Multiresolution for Geometric Modelling*.

*e-mail: {ilja | ps | mathieu}@cs.caltech.edu

[1] There is one technicality here as obtuse triangles are only projected until they touch in exactly one point, which is the center of the longest edge.