

Discrete Inverse Spectral Problem - Progress Summary

Feng Ling

August, 2017

1 The Problem

The question in exploration is the 2D discrete inverse spectral problem, namely that can one obtain the metric of a 2D discrete surface from its discrete Laplace-Beltrami spectrum? To make this problem simpler, we assume the surface have the topology of \mathbb{S}^2 , then its metric would be conformally equivalent to that of a discrete sphere. So we only need to obtain a conformal map that recovers the surface from the sphere.

2 Methods

We approach this problem by naïvely optimizing the ‘energy’ $E(\mathbf{c})$ that represents the squared difference between the discrete spectrum of the current shape $\boldsymbol{\lambda}$, and that of a target shape, $\boldsymbol{\lambda}_T$, where \mathbf{c} is the diagonal conformal factors of the shape in comparison to the round sphere. *i.e.* Let $E(\mathbf{L}(\mathbf{c})) = \frac{1}{2}|\boldsymbol{\lambda} - \boldsymbol{\lambda}_T|^2$, where \mathbf{L} is our choice of the Laplace-Beltrami operator of the current surface, and we will compute its gradient $\nabla E(\mathbf{L}(\mathbf{c})) = |\boldsymbol{\lambda} - \boldsymbol{\lambda}_T| \frac{\partial \boldsymbol{\lambda}}{\partial \mathbf{L}} \frac{\partial \mathbf{L}}{\partial \mathbf{c}}$ and do simple gradient descent (or BFGS).

In order to measure success, we first use the modified conformal Mean Curvature Flow (cMCF) of the original mesh to obtain a set of conformal factors to which we compare our optimized factors. Then we also try to use another gradient based method to re-embed our optimized conformal factors into a mesh and visually inspect its differences with the the original target mesh.

2.1 Simplicial basis

If we use triangle shape functions (linear FEM basis functions) and triangular mesh to initialize our sphere, then its LB spectrum can be estimated by the standard weighted *cotan operator*. There are known results about the accuracy of the eigenvalues of this discrete operator with respect to that of the smooth surface, at least in the fine mesh limit. Its spectrum will be called the piecewise linear (PL) spectrum.

2.2 Spherical harmonic basis

We can also use spherical harmonics (SH) as our basis functions to represent our sphere. This way, there is no dependence on which specific mesh we start with as the final target shape will simply be the sum of the corresponding harmonics with the desired magnitudes. through Wigner 3j symbols

Without giving any proofs, it is intuition to believe that eigenvalues of different harmonics should roughly correspond to features of varying spatial frequencies as only higher and higher degress harmonics have finer and finer ‘bumps.’

2.3 Regularization

We can also append a bi-Laplacian regularization term to our energy E to (hopefully) overcome high-frequency discretization noises. This is reasonable because assuming the change in metric is small between each time step, we want to minimize fluctuations of the current Laplace-Beltrami operator, hence its Laplacian (approximated by itself) should be 0 or very small.

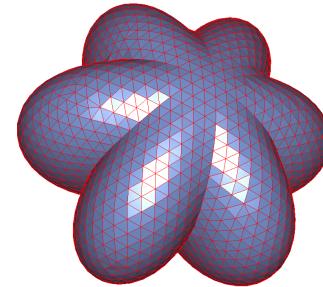
3 Experiments and Results

3.1 target shapes

We have tested our methods on the following shapes,

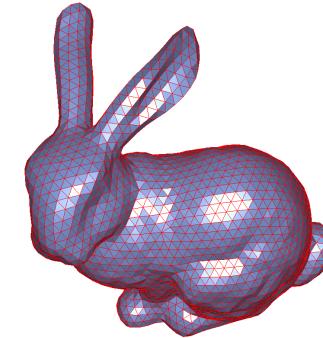
1. “star-shaped” surfaces

These are surfaces obtained from a spherical mesh flowed along its mesh normals proportional to some scalar spherical harmonic functions ($|Y_3^3|$ on the right).



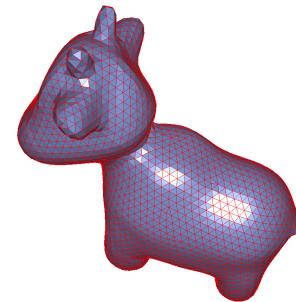
2. classic bunny

This is the archetypal mesh used by computer graphics literature. While it has many crucial features interesting for robust testing, its ears require rather large conformal factors that might not work too well with naïve gradient descent type methods.



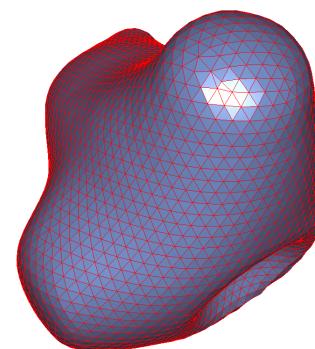
3. spot the cow

This is a tamer version of the bunny as it has slightly rounder protrusions due to its cartoon-ish looks. We also tested some procedures on a sequence of “melted” versions (meshes that underwent cMCF before it becomes a round sphere).



4. the blob

To avoid large changes in conformal factors yet maintain some asymmetry and generality, this blob is used for testing in spherical harmonic basis problems as we want to be able to test performance of our method against frequency truncations.



3.2 parameter nomenclature

In order of appearance:

1. numeig - the total number of eigenvalues being optimized
2. pert - a scalar proportional to how distorted the target shape is comparing to a round sphere.
3. numa - the total number of a
4. maxL - the maximal degree used by spherical harmonic basis approximation

3.3 Simplicial basis inverse problem

3.3.1 raw

1. Stability against feature ‘smoothness’ of the target mesh

To guarantee higher success rates, the star-shaped surfaces (from $|Y_3^2|$) are used. We vary the craziness of the target mesh by supplying a scaling perturbation constant $p \in \mathbb{R}^{\geq 0}$ where $p = 0$ is unperturbed sphere and $p \rightarrow \infty$ limits to the actual spherical harmonic shape with singular points.

2. Stability against the number of optimized eigenvalues

Fixing the target surface to be the star-shaped surfaces (from $|Y_3^3|$) with a perturbation constant of 0.5,

3.3.2 regularized

To attack the problem for more complicated target shapes such as the bunny, we used a

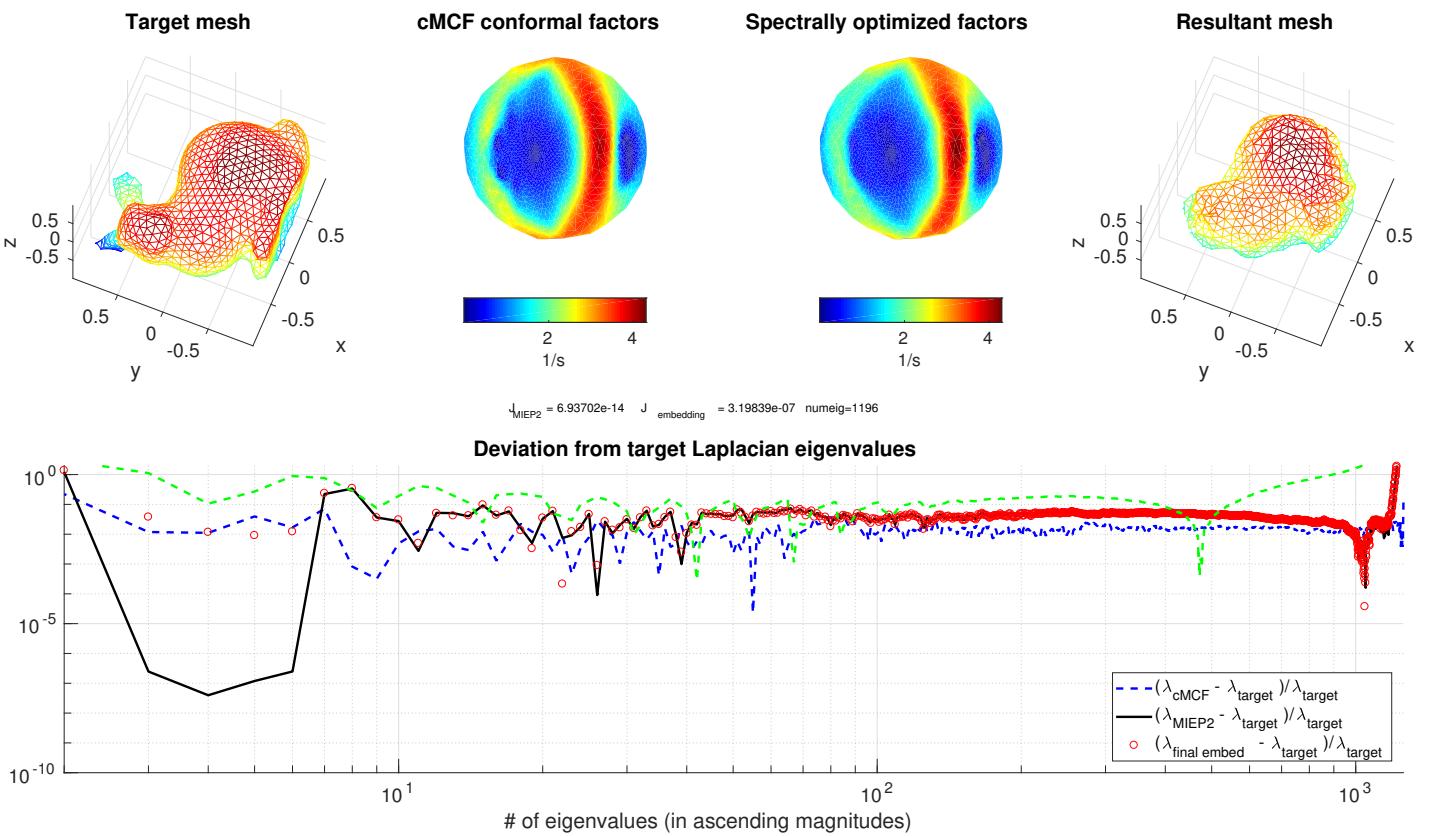


Figure 1: $a=1196$, $r=0.05$,
note that the inverse conformal factors are plotted on the hammer projections.

3.3.3 recursive fit

To demonstrate that our optimized conformal factor indeed encode the desired geometric information and even a very simple mesh reconstruction method can yield some visual results despite the large conformal factors at the ears and feet, we passed a resultant mesh from a regularized fit through the algorithm one more time to obtain a new resultant mesh more resemblant to the target bunny as below.

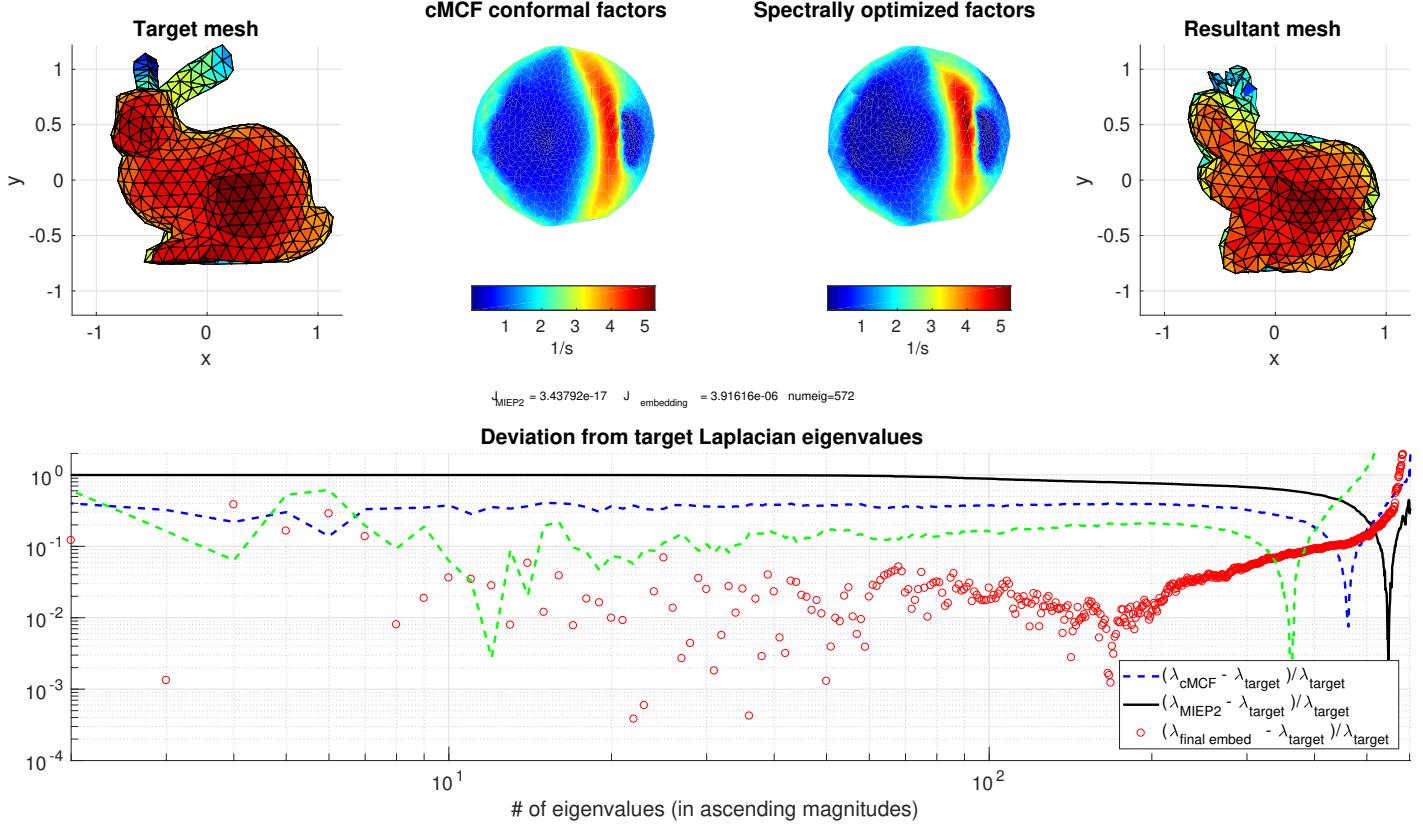


Figure 2: $a=572$, $r=0.1$,
note that the inverse conformal factors are plotted on the hammer projections.

An interesting result is that when using our naïve mesh reconstruction, the size of conformal factors can force protrusion of the ears but local errors make them curl pathologically. Also in some cases the features became inverted during the reconstruction (*e.g.* the tail).

3.3.4 Discussion

Since our simplicial basis are attached to some given mesh, we have to work with a PL embedding of the sphere that is the result of the cMCF bunny mesh. This is clearly undesirable as in real life we would not have access to a correctly adaptively refined spherical mesh to begin our gradient descent. To attack this problem, we believe either: 1) we develop a adaptive scheme where we start with a coarse mesh and gradually refine the mesh where the fitted conformal factor is getting large [adaptive mesh], 2) or we have to run structured optimizations that start with lower frequencies and move on to higher frequencies later [recursive optimization].

3.4 Spherical harmonics basis forward problem

Before diving into the inverse problem, we need to verify that the LB spectrum can be well represented by a truncated spectrum in spherical harmonic basis.

In Figure ?? below we demonstrate some convergence of the spectrum from above to some value versus the convergence from below when looking at PL spectrum.

convergence w.r.t. maximal degree and w.r.t. the number of free SH basis coefficients

3.5 Spherical harmonics basis inverse problem

3.5.1 raw

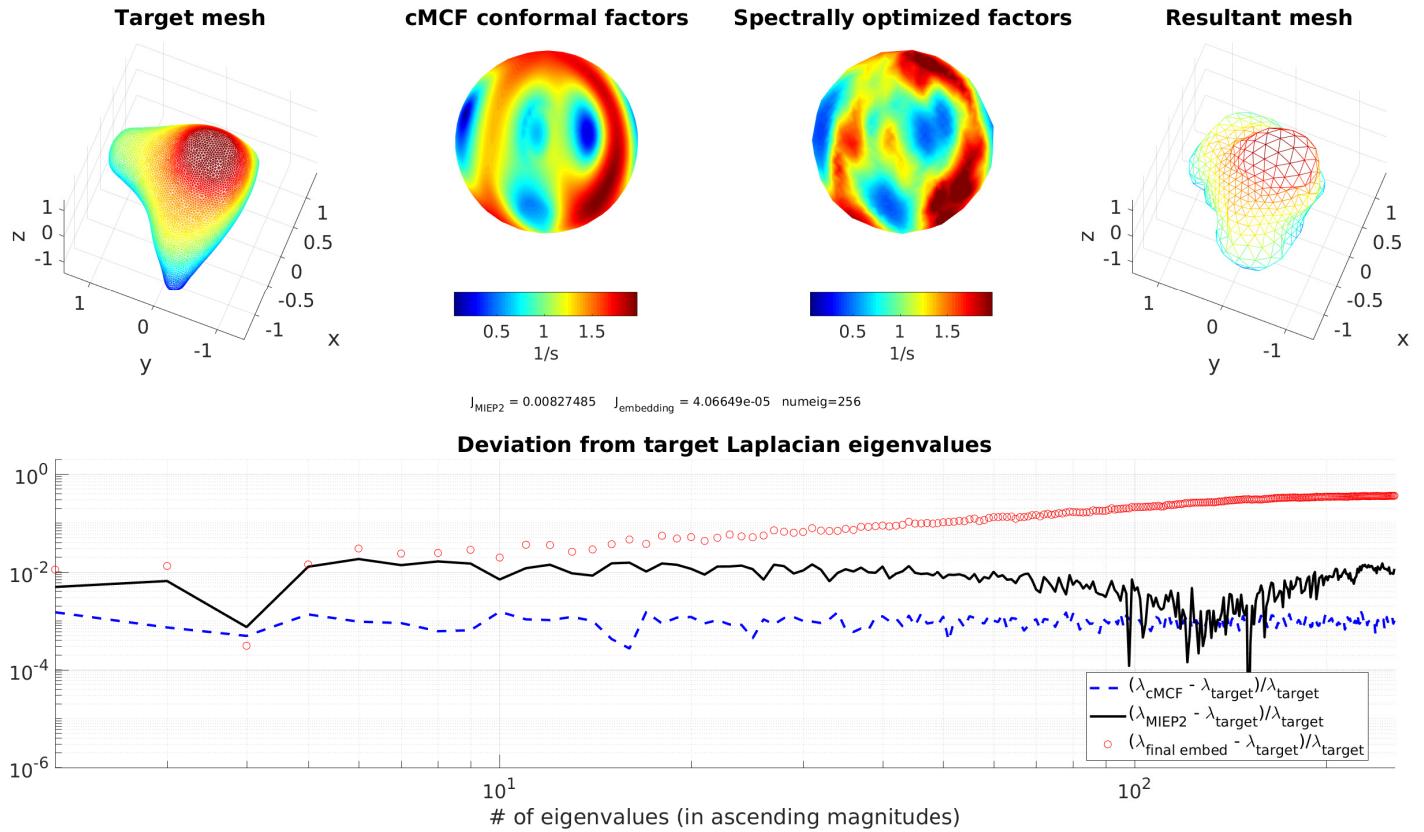


Figure 3: $a=256$, $L=30\times 30$

3.5.2 regularized

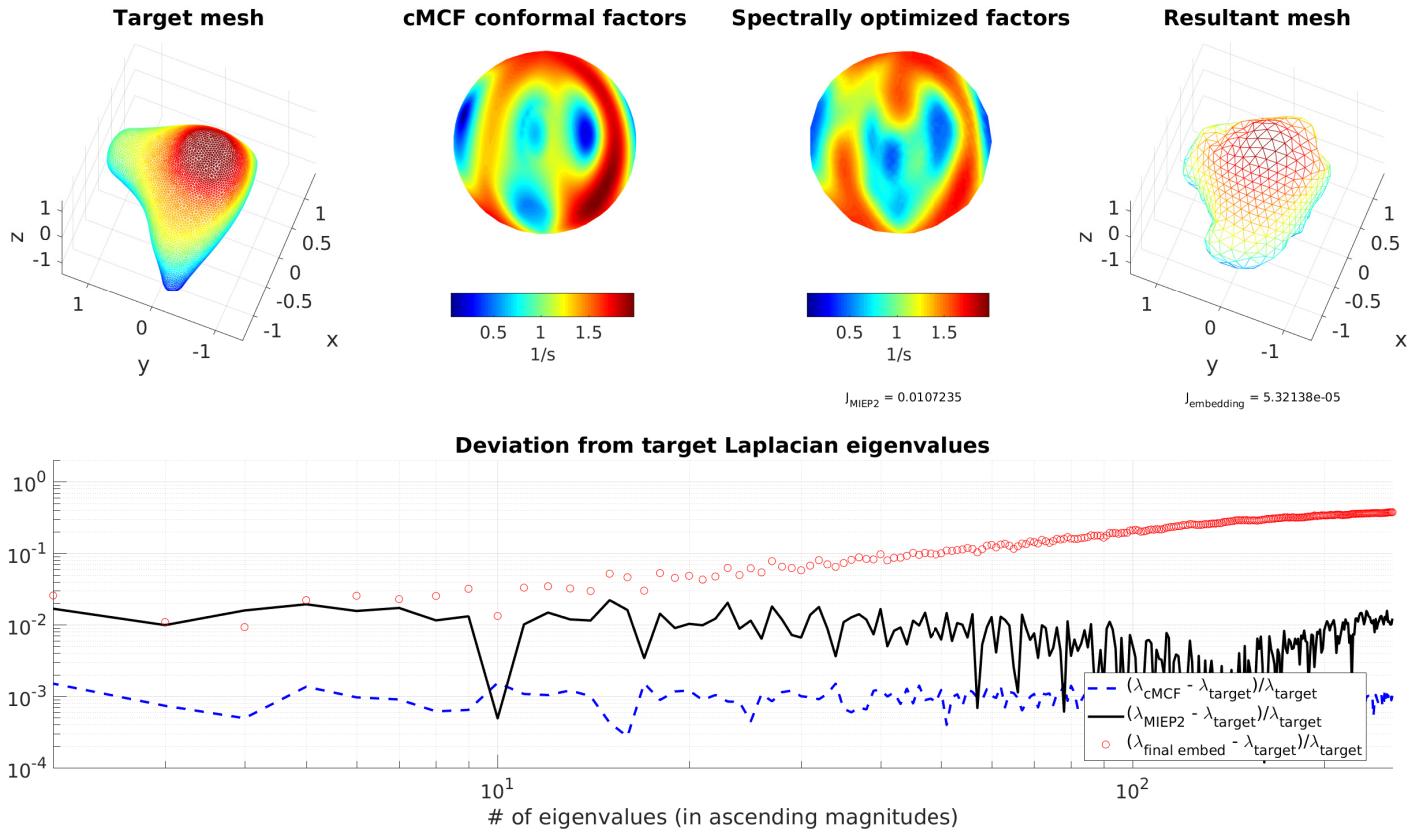


Figure 4: $a=256$, $L=30\times 30$, $R=1e-6$

4 A To-Do list

4.1 Möbius balancing routine

For reference, Prof. Keenan Crane have a write-up and working C codes in his geometry processing toolset. The current MATLAB implementation is experiencing (too much?) oscillatory behavior possibly due to some sort of silly bug.

4.2 Structured optimization

Again assuming the above equivalence, it might stand to reason that we shall first optimize for lower frequency features ($a_{l,m}$ with small l ?) and then move on to refining the higher frequency ones.

Related to this is that there is no indication how many coefficient should be used.

4.3 Precomputing more 3j symbols

maybe it will be worthwhile to utilize some state-of-the-art method to calculate as many 3j symbols as one's RAM can handle to eliminate this uncertainty for the experiments.

Reference: <http://pubs.siam.org/doi/abs/10.1137/S1064827503422932>

4.4 Topology

We also would like to calculate the topology of the surface from the spectrum beforehand. In the smooth setting, we can infer these from the **asymptotics** of heat trace from the spectrum. In the discrete case, it is not immediate how to extract these information. A reference is found to mention successfully doing this through some sort of curve fitting. Or is this a suitable task for machine learning? And after knowing a topology with $g \geq 1$, can be also use the corresponding harmonic basis functions?

Reference: Martin Reuter, Franz-Erich Wolter, Niklas Peinecke, “Laplace-Beltrami spectra as ‘Shape-DNA’ of surfaces and solids”

<http://reuter.mit.edu/blue/papers/reuter-shapeDNA06/reuter-shapeDNA06.pdf>

4.5 Adaptive remeshing for the simplicial inverse problem

Since it works pretty well in the simplicial setting when we regularize and use enough eigenvalues (up to embedding issues), the only obstacle barring us from full victory is the apparent need to assume MCF starting mesh. The idea to improve here is to remesh at each step according to the current conformal factor solution. Practically speaking, it might preserve more sanity to try these in a better development environment (so no more MATLAB non OOP shenanigans, at least not with the built-in BFGS optimization algorithm).

4.6 Embedding problem

Visually and numerically, it seems that we can obtain spectrally optimized conformal factors that matches the desired cMCF ones. However, the (again) naïve way of re-inflating a sphere using a scalar conformal factor field seems to less than efficient, as in we need very fine meshes and the naïve algorithm scales with number of meshes used squared. But this should be a problem that others have and will continue to tackle even on their own.