Delaunay Meshes for Space-Time Applications: Open Problems

David M. Williams
Pennsylvania State University

Mathijs Wintraecken
Centre Inria d'Université Côte d'Azur

May 19, 2026

Presentation Overview

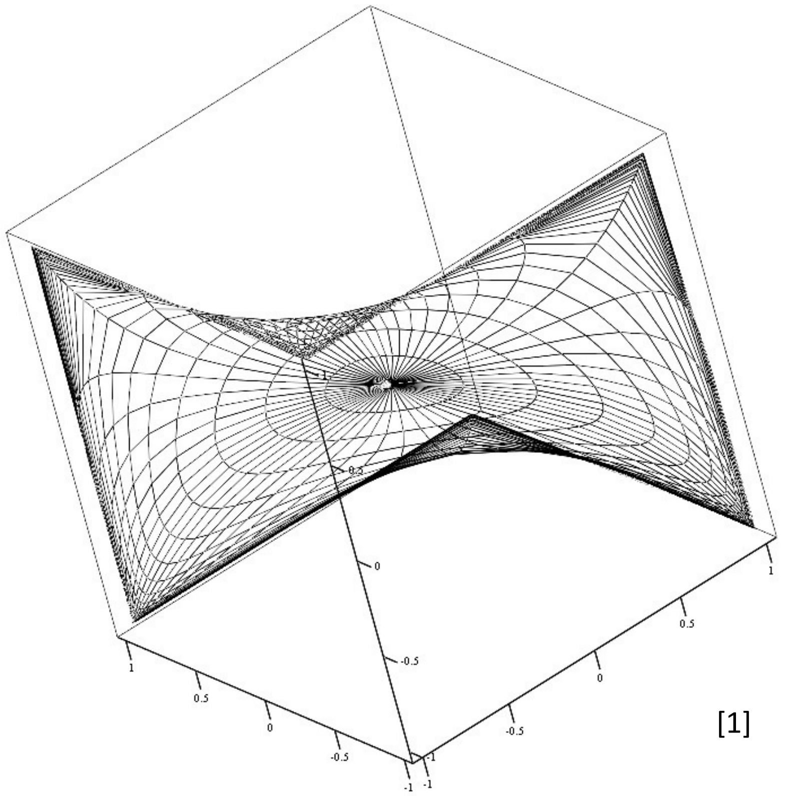
- **Outline:**
  - Motivation
  - The Delaunay criterion and Euclidean spaces
  - The Delaunay criterion and non-Euclidean spaces
  - Anisotropic Delaunay meshes
  - Open problems and solution strategies

Motivation

Space-Time Methods

- **Objective:** develop space-time finite element methods for solving fluid-structure interaction (FSI) problems on adapted, unstructured, four-dimensional meshes.

- **Challenge:** we must operate in four dimensions
  - 4D CAD
  - 4D hypersurface meshes
  - 4D hypervolume meshes
  - 4D mesh adaptivity
  - 4D finite element basis functions

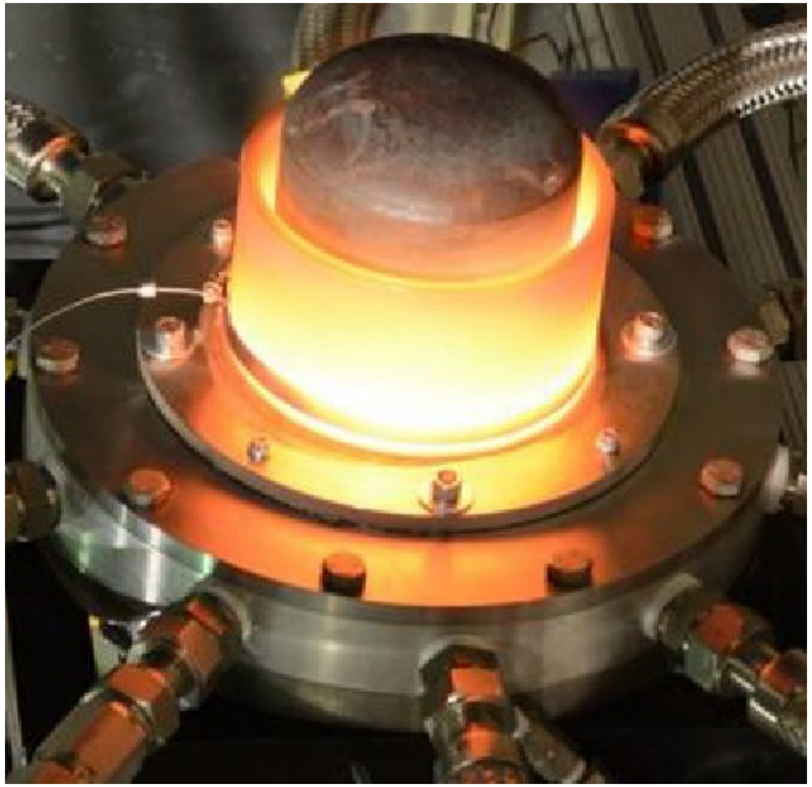


[1]

Space-Time Methods

- **Benefits**
  - Space-time mesh adaptivity
  - High-order accuracy for moving shockwaves
  - High-fidelity boundary motion
  - Unsteady adjoint
  - Space-time multigrid

- **Applications:**
  - RDEs and RDAs
  - Conventional jet engines



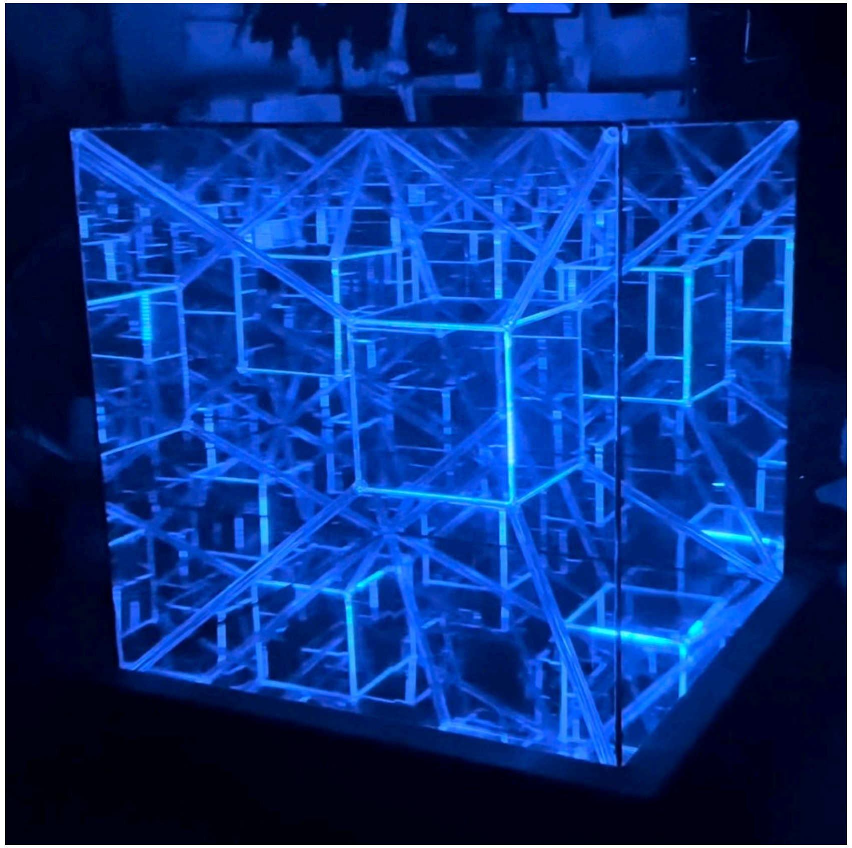
[2]

Meshes for Space-Time Applications

- **Question:** why use Delaunay meshes for four-dimensional space-time applications?

- **Answer 1:** we want optimal interpolation and approximation for the numerical solutions of space- and time-dependent partial differential equations

- Contrary to popular belief, Delaunay meshes possess optimality properties which extend into higher dimensions (greater than two)



[3]

Meshes for Space-Time Applications

- **Question:** why use Delaunay meshes for four-dimensional space-time applications?

- **Answer 2:** we want to avoid ad-hoc procedures which are often used to generate meshes in each dimension. Frequently, these procedures are "case-by-case" and require significant modifications each time one changes the number of dimensions

- The Bowyer-Watson algorithm allows us to rapidly and automatically generate a Delaunay mesh in any number of dimensions

[3]

Delaunay Optimality in 2 Dimensions

- Optimality for elliptic boundary-value problems

- **Interpolation:** Delaunay triangulations minimize the gradient interpolation error relative to any other triangulation of the same set of points
- **Approximation:** Delaunay triangulations minimize the error of a finite element solution in the energy norm
- **Degree:** piecewise-linear functions

Rippa, Schiff, “Minimum energy triangulations for elliptic problems,” *Computer Methods in Applied Mechanics and Engineering*, 1990

Rippa, “Minimal roughness property of the Delaunay triangulation,” *Computer Methods in Applied Mechanics and Engineering*, 1990

Delaunay Optimality in *d* Dimensions

- Quasi-optimality for elliptic boundary-value problems

- **Interpolation:** protected Delaunay triangulations yield the best-known gradient interpolation error relative to any other triangulation of the same set of points
- **Approximation:** protected Delaunay triangulations yield the best-known error of the finite element solution in the energy norm
- **Degree:** piecewise high-order polynomial functions

Williams and Wintraecken, "Error estimates for the interpolation and approximation of gradients and vector fields on protected Delaunay meshes in R^d," *arXiv preprint arXiv: 2505.18987*, 2025

Delaunay Optimality in *d* Dimensions

- Optimality for interpolating C^1-continuous functions with pointwise-bounded second derivatives

- **Interpolation:** Delaunay triangulations minimize the worst-case pointwise interpolation error relative to any other triangulation of the same set of points
- **Degree:** piecewise-linear functions

Rajan, "Optimality of the Delaunay triangulation in R^d," *Discrete & Computational Geometry*, 1994

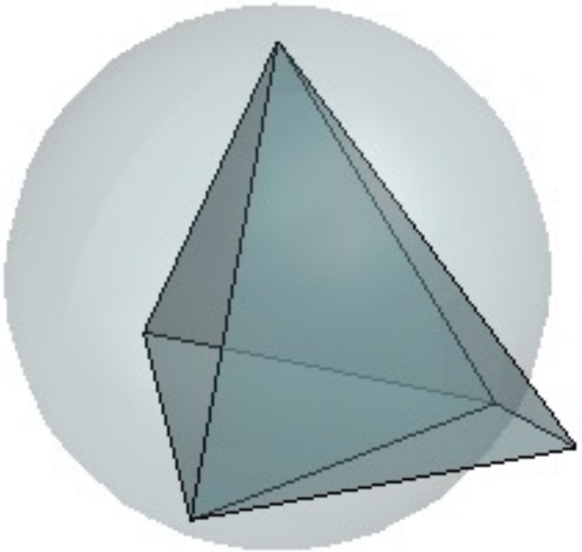
Waldron, "The error in linear interpolation at the vertices of a simplex," *SIAM Journal on Numerical Analysis*, 1998

The Delaunay Criterion and Euclidean Spaces

PennState
College of Engineering

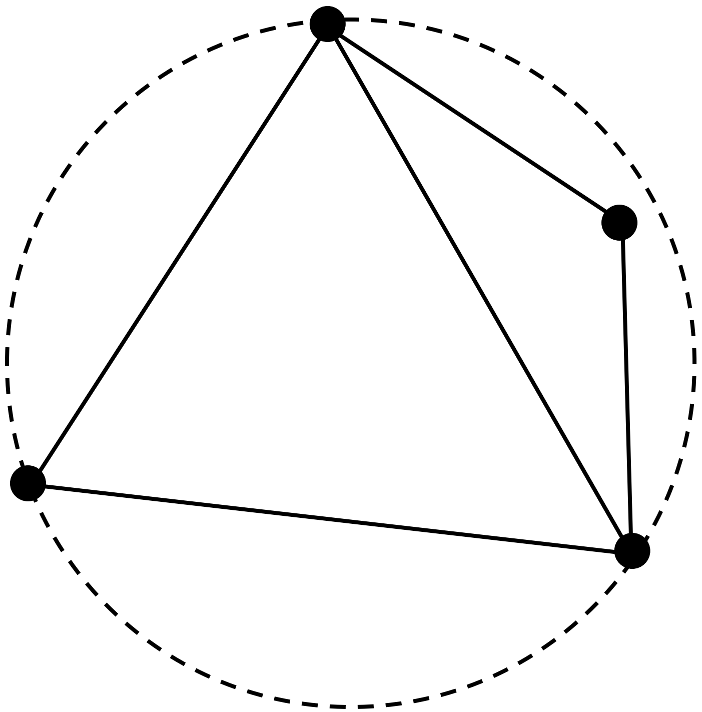
Delaunay Meshes in Euclidean Space

- **Delaunay criterion in 3D:** “the circumsphere of each tetrahedron must not contain any points other than the vertices of the tetrahedron itself”
- The Delaunay criterion implicitly depends on the notion of the Euclidean distance between points
- The distance between the center of the circumsphere and a (non-vertex) point must exceed the radius of the circumsphere
- In Euclidean space this is easy to quantify

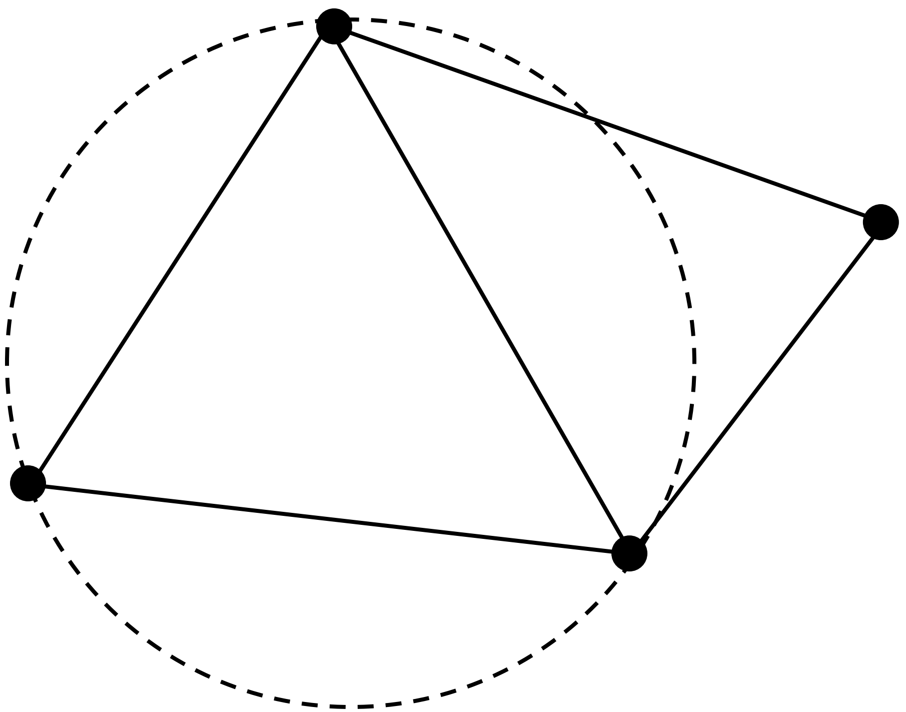


[4]

Delaunay Meshes in Euclidean Space



non-Delaunay



Delaunay

Delaunay Meshes in Euclidean Space

- Consider two points in the Euclidean space $\mathbb{E}^3$

$$\mathbf{x}_c = [x_c, y_c, z_c]^T \qquad \mathbf{x}_p = [x_p, y_p, z_p]^T$$

- The left point is the center of the circumsphere
- The right point is a generic, non-vertex point
- **Key idea:** both points are "space" points with spatial coordinates *x*, *y*, *z*

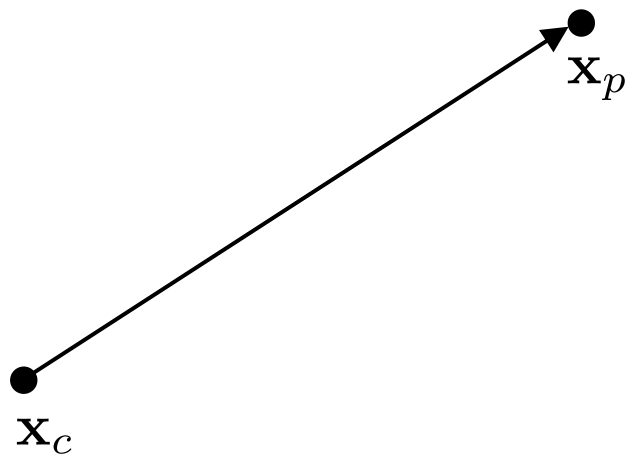
Delaunay Meshes in Euclidean Space

- The inner product in Euclidean space $\mathbb{E}^3$ is well defined

$$(\mathbf{x}_p - \mathbf{x}_c)^T (\mathbf{x}_p - \mathbf{x}_c) \quad \Rightarrow \quad \text{length}^2$$

- Dimensionally consistent inner products
- The squared distance and the power are also well defined

$$\mathcal{P} = (\mathbf{x}_p - \mathbf{x}_c)^T (\mathbf{x}_p - \mathbf{x}_c) - R^2$$

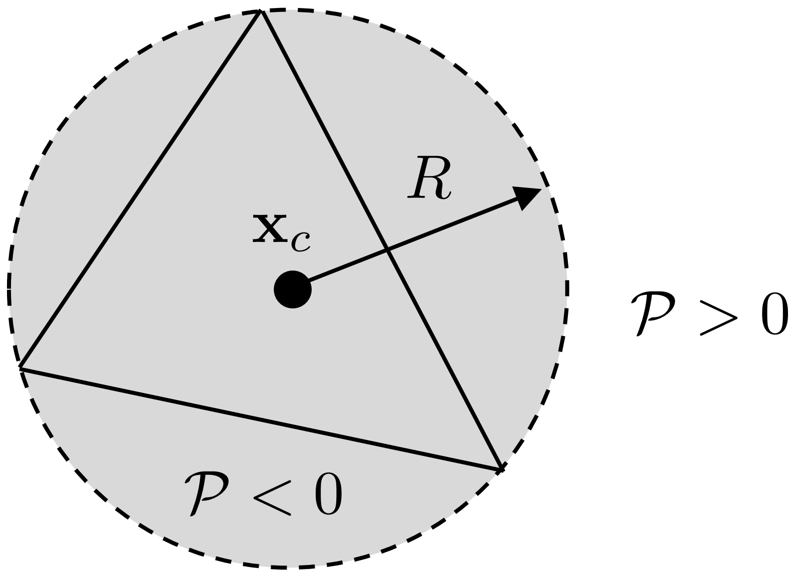


Delaunay Meshes in Euclidean Space

- The Delaunay criterion simply insists that all non-vertex points of each tetrahedron have positive powers

$$\mathcal{P} = \left(\mathbf{x}_p - \mathbf{x}_c\right)^T \left(\mathbf{x}_p - \mathbf{x}_c\right) - R^2 > 0$$

- Point inside circumsphere $\quad \mathcal{P} < 0$
- Point on circumsphere $\quad \mathcal{P} = 0$
- Point outside circumsphere $\quad \mathcal{P} > 0$



- **Key idea:** the Delaunay criterion is well defined when an inner product for the space is also well defined

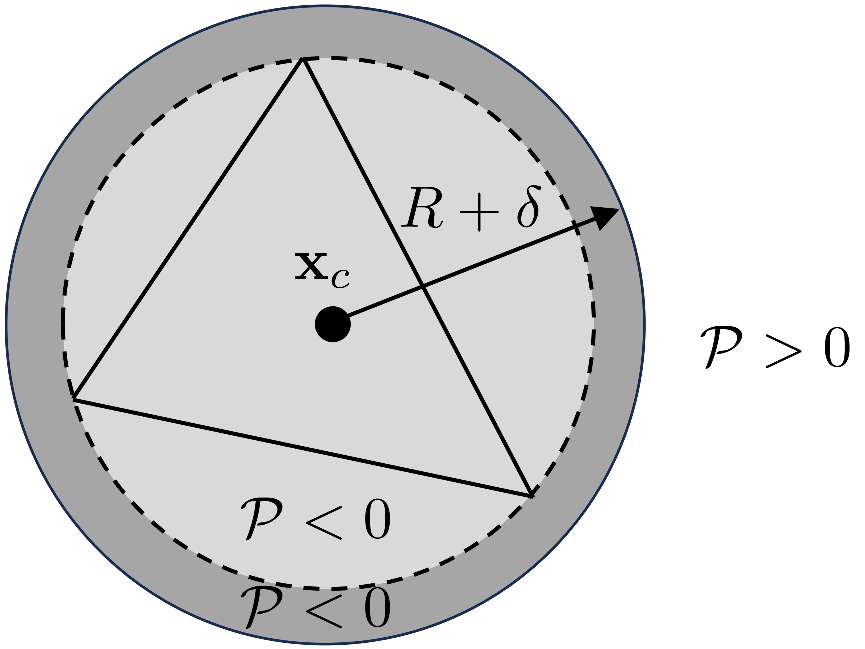
Protected Delaunay Meshes in Euclidean Space

- Consider the extended circumsphere with radius $R + \delta$

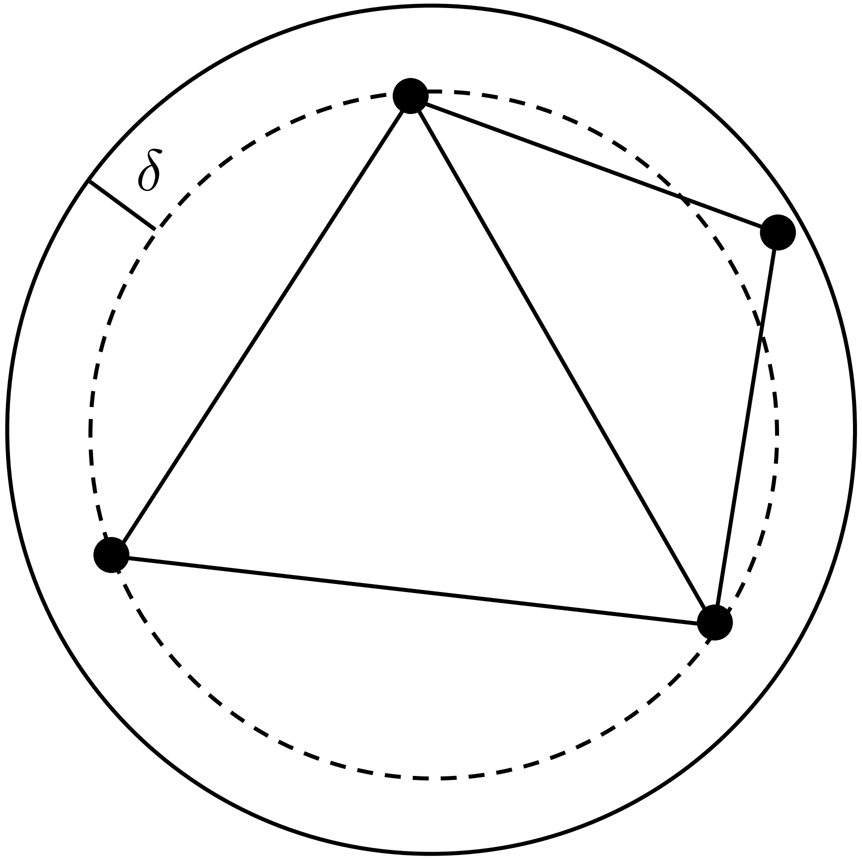
- The protected Delaunay criterion insists that all non-vertex points of each tetrahedron have positive powers with respect to the extended circumsphere

$$\mathcal{P} = (\mathbf{x}_p - \mathbf{x}_c)^T (\mathbf{x}_p - \mathbf{x}_c) - (R + \delta)^2 > 0$$

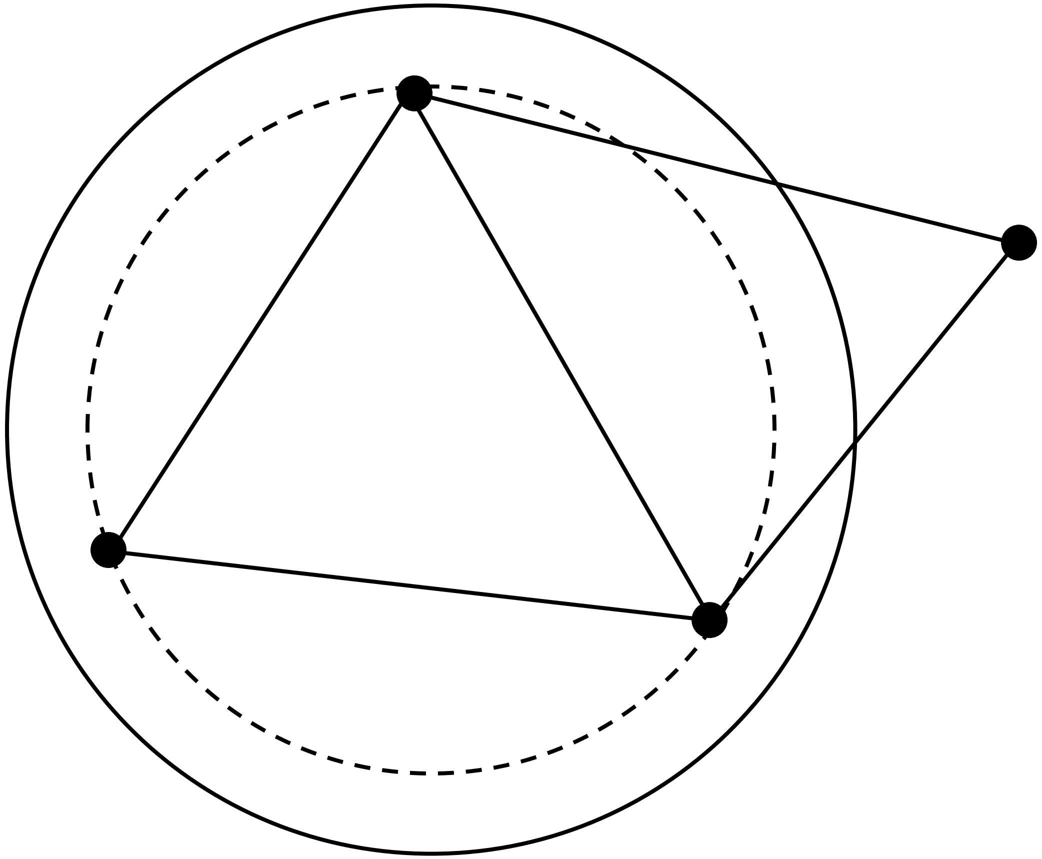
- Here the quantity δ is called the protection



Protected Delaunay Meshes in Euclidean Space



Delaunay



Protected Delaunay

Protected Delaunay Meshes in Euclidean Space

- **Importance:** having non-zero protection prevents the formation of slivers. There is a direct relationship between large values of protection, and large guaranteed minimum thicknesses of the tetrahedra

$$\delta > 0$$

- A sliver is a very flat tetrahedron with nearly zero relative altitude with respect to one of its vertices

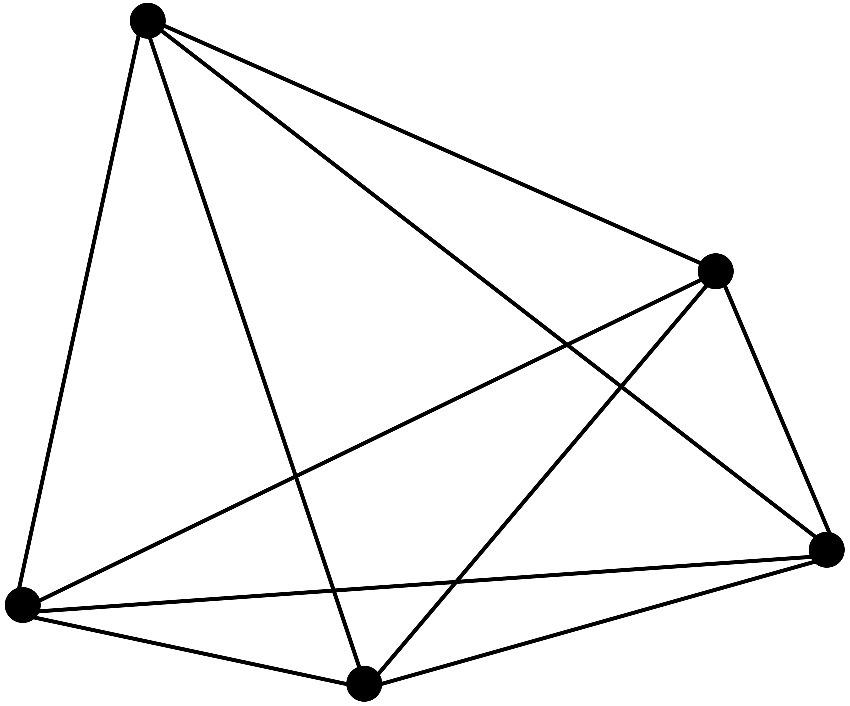
- **Key idea:** the protected Delaunay criterion is well defined when an inner product for the space is also well defined

The Delaunay Criterion and non-Euclidean Spaces

PennState College of Engineering

Delaunay Meshes in Riemannian Space

- **Delaunay criterion in 4D:** "the circumhypersphere of each pentatope must not contain any points other than the vertices of the pentatope itself"
- The Delaunay criterion implicitly depends on the notion of the Euclidean distance between points
- **Problem:** The Euclidean distance is NOT well defined



Delaunay Meshes in Riemannian Space

- Consider two points in the non-Euclidean space $\mathbb{E}^3 \times \mathbb{R}$

$$\mathbf{x}_c = [x_c, y_c, z_c, t_c]^T \qquad\qquad \mathbf{x}_p = [x_p, y_p, z_p, t_p]^T$$

- The left point is the center of the circumsphere
- The right point is a generic, non-vertex point
- **Key idea:** both points are "space-time" points with spatial coordinates *x*, *y*, *z* and time coordinate *t*

Delaunay Meshes in Riemannian Space

- The Euclidean inner product is not well defined

$$\left(\mathbf{x}_p - \mathbf{x}_c\right)^T \left(\mathbf{x}_p - \mathbf{x}_c\right) \quad \Rightarrow \quad \text{length}^2 + \text{time}^2$$

- Dimensionally inconsistent inner products
- All downstream quantities, such as the squared distance, power, and Delaunay criterion are also ill defined
- This is practically okay, but it interferes with optimality arguments which require dimensional consistency

Delaunay Meshes in Riemannian Space

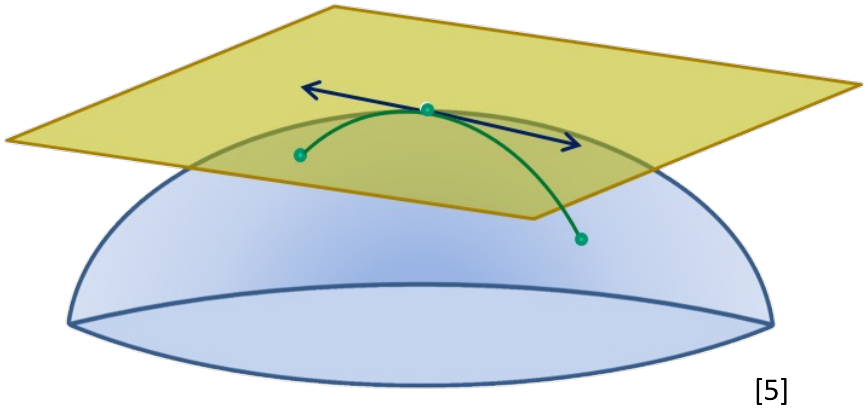
- **Problem:** The Euclidean distance is NOT well defined
- **Solution:** Define a Riemannian distance with respect to a metric field

- Metric field: a symmetric positive-definite (SPD) matrix function which is defined at every point in the space-time domain

$$\mathbf{M}(x, y, z, t)$$

- Separates points

$$\mathbf{x}^T\mathbf{M}(x, y, z, t)\mathbf{x} > 0, \qquad \mathbf{x} \neq \mathbf{0}$$



[5]

Delaunay Meshes in Riemannian Space

- Enforces dimensional consistency

$$\mathbf{M}(x,y,z,t) = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{12} & m_{22} & m_{23} & m_{24} \\ m_{13} & m_{23} & m_{33} & m_{34} \\ m_{14} & m_{24} & m_{34} & m_{44} \end{bmatrix}$$

- Dimensionless: $m_{11}, m_{12}, m_{13}, m_{22}, m_{23}, m_{33}$
- Dimensions of *length/time*: m_{14}, m_{24}, m_{34}
- Dimensions of *length^2/time^2*: m_{44}

Spatial anisotropy

Characteristic speeds

Characteristic speeds squared

Delaunay Meshes in Riemannian Space

- The Riemannian inner product IS well defined

$$(\mathbf{x}_p - \mathbf{x}_c)^T \mathbf{M} (\mathbf{x}_p - \mathbf{x}_c) \quad \Rightarrow \quad \text{length}^2$$

- Dimensionally consistent inner products
- The squared distance and the power are also well defined

$$\mathcal{P}_M = (\mathbf{x}_p - \mathbf{x}_c)^T \mathbf{M} (\mathbf{x}_p - \mathbf{x}_c) - R_M^2$$

- The “anisotropic Delaunay criterion” is now well defined

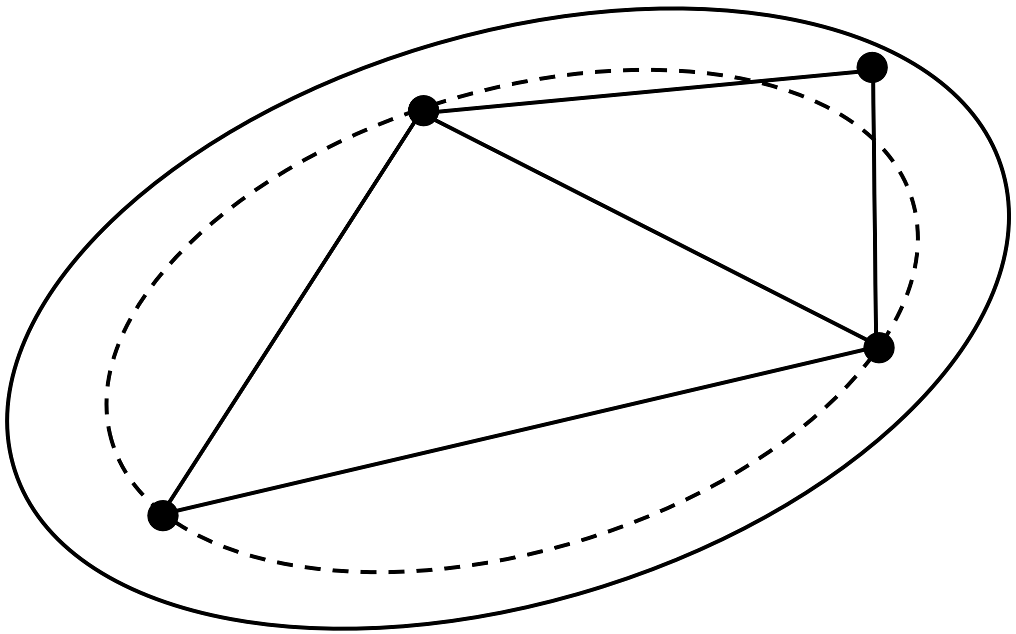
Protected Delaunay Meshes in Riemannian Space

- The "protected anisotropic Delaunay criterion" is also well defined

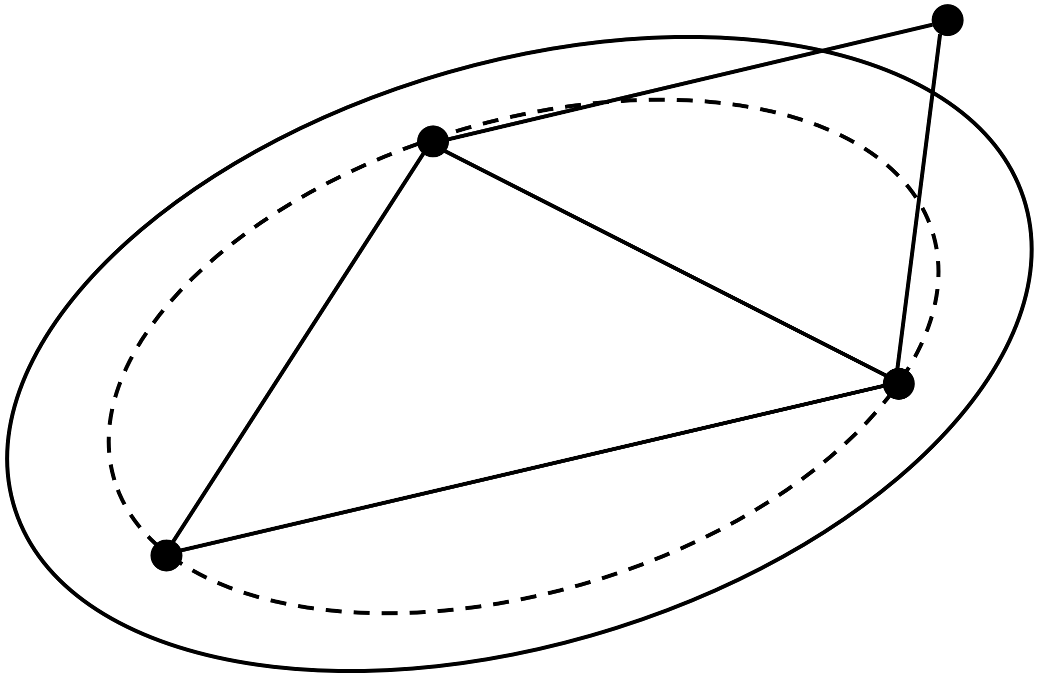
$$\mathcal{P}_M = \left(\mathbf{x}_p - \mathbf{x}_c\right)^T \mathbf{M} \left(\mathbf{x}_p - \mathbf{x}_c\right) - \left(R_M + \delta_M\right)^2$$

- Protection is now enforced relative to a distorted version of the circumhypersphere, where the distortion is specified by the metric field

Protected Delaunay Meshes in Riemannian Space



Anisotropic Delaunay



Protected Anisotropic Delaunay

Delaunay Meshes in Riemannian Space

- Enforces dimensional consistency [More precise construction]

$$\mathbf{M}(x,y,z,t) = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{12} & m_{22} & m_{23} & m_{24} \\ m_{13} & m_{23} & m_{33} & m_{34} \\ m_{14} & m_{24} & m_{34} & m_{44} \end{bmatrix}$$

- Dimensions of *1/length²*: $m_{11}, m_{12}, m_{13}, m_{22}, m_{23}, m_{33}$
- Dimensions of *1/length*time*: m_{14}, m_{24}, m_{34}
- Dimensions of *1/time²*: m_{44}

$$(\mathbf{x}_p - \mathbf{x}_c)^T \mathbf{M} (\mathbf{x}_p - \mathbf{x}_c) \quad \Rightarrow \quad \text{dimensionless}$$

Delaunay Meshes in Riemannian Space

- How do we obtain a metric like this?
- Compute the Hessian of a scalar field $\phi(x, y, z, t)$

$$\mathbf{H}(x,y,z,t) = \begin{bmatrix} \frac{\partial^2 \phi}{\partial x^2} & \frac{\partial^2 \phi}{\partial x \partial y} & \frac{\partial^2 \phi}{\partial x \partial z} & \frac{\partial^2 \phi}{\partial x \partial t} \\ \frac{\partial^2 \phi}{\partial x \partial y} & \frac{\partial^2 \phi}{\partial y^2} & \frac{\partial^2 \phi}{\partial y \partial z} & \frac{\partial^2 \phi}{\partial y \partial t} \\ \frac{\partial^2 \phi}{\partial x \partial z} & \frac{\partial^2 \phi}{\partial y \partial z} & \frac{\partial^2 \phi}{\partial z^2} & \frac{\partial^2 \phi}{\partial z \partial t} \\ \frac{\partial^2 \phi}{\partial x \partial t} & \frac{\partial^2 \phi}{\partial y \partial t} & \frac{\partial^2 \phi}{\partial z \partial t} & \frac{\partial^2 \phi}{\partial t^2} \end{bmatrix}$$

- The Hessian approximates the truncation error for a second-order method!
- Anisotropy is dictated by the Hessian which acts as an error indicator

Delaunay Meshes in Riemannian Space

- **Problem:** the Hessian is symmetric, but not guaranteed to be positive definite
- We need a field of SPD matrices!

- **Solution:** we set the metric field equal to the matrix absolute value of the Hessian field

$$\mathbf{H} = \mathbf{P}\mathbf{D}\mathbf{P}^{-1} = \mathbf{P}\begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 \\ 0 & 0 & 0 & \lambda_4 \end{bmatrix}\mathbf{P}^{-1}$$

$$|\mathbf{H}| = \mathbf{P}\,|\mathbf{D}|\,\mathbf{P}^{-1} = \mathbf{P}\begin{bmatrix} |\lambda_1|+\varepsilon & 0 & 0 & 0 \\ 0 & |\lambda_2|+\varepsilon & 0 & 0 \\ 0 & 0 & |\lambda_3|+\varepsilon & 0 \\ 0 & 0 & 0 & |\lambda_4|+\varepsilon \end{bmatrix}\mathbf{P}^{-1}$$

$$\varepsilon > 0$$

Delaunay Meshes in Riemannian Space

- The Riemannian inner product is well defined if $\mathbf{M} = |\mathbf{H}|$

$$(\mathbf{x}_p - \mathbf{x}_c)^T \, \mathbf{M} \, (\mathbf{x}_p - \mathbf{x}_c) \quad \Rightarrow \quad (\text{units of } \phi)^2$$

Delaunay Meshes in Riemannian Space

- The anisotropic Delaunay criterion, and the associated predicates are defined with respect to a metric field

- **Problem:** Where should we evaluate the metric field when we compute distances or Delaunay predicates?

$$\mathbf{M}(x, y, z, t)$$

$$\boxed{\left(\mathbf{x}_p - \mathbf{x}_c\right)^T \mathbf{M} \left(\mathbf{x}_p - \mathbf{x}_c\right) = \left(\mathbf{x}_p - \mathbf{x}_c\right)^T \mathbf{M}(\mathbf{x}_q) \left(\mathbf{x}_p - \mathbf{x}_c\right)}$$

Anisotropic Delaunay Meshes

Practical Application: Bowyer-Watson Algorithm

1. Create a collection of points which will appear as vertices in the mesh
2. Create a *d*-dimensional bounding box which contains all the points
3. Subdivide the bounding box into a collection of Delaunay *d*-simplices
4. Insert the points one at a time into the Delaunay mesh
   - Check to see if the point to be inserted lies inside the circumhypersphere of any existing *d*-simplices
   - Mark all such simplices as “bad simplices”
   - Create a cavity by identifying the non-shared facets of the bad simplices
   - Remove any invisible facets from the cavity
   - Connect the facets of the cavity to the newly inserted point to create new simplices
   - Remove all bad simplices

Practical Application: Bowyer-Watson Algorithm

5. Repeat step 4 until all points have been inserted
6. Remove all simplices which contain vertices that belong to the original bounding box

- **Key idea:** evaluating if a point lies inside of the circumhypersphere of a *d*-simplex requires two predicates:
  - $\textit{orientation}_M$ predicate
  - $\textit{inhypersphere}_M$ predicate

- Here, the natural choice is to evaluate the input metric *M* at the point being inserted

$$\mathbf{M} = \mathbf{M}(\mathbf{x}_q)$$

Borouchaki-George Approach

- The modified Bowyer-Watson algorithm with metric-weighted predicates is essentially the Borouchaki-George approach

Borouchaki, George, Hecht, Laug, and Saltel, “Delaunay mesh generation governed by metric specifications. Part I. Algorithms,” *Finite Elements in Analysis and Design,* 1997

Borouchaki, George, and Mohammadi, “Delaunay mesh generation governed by metric specifications Part II. Applications,” *Finite Elements in Analysis and Design,* 1997

- The metric field is evaluated at the point being inserted and/or at the vertices of the simplex which is being assessed by the $inhypersphere_M$ predicate

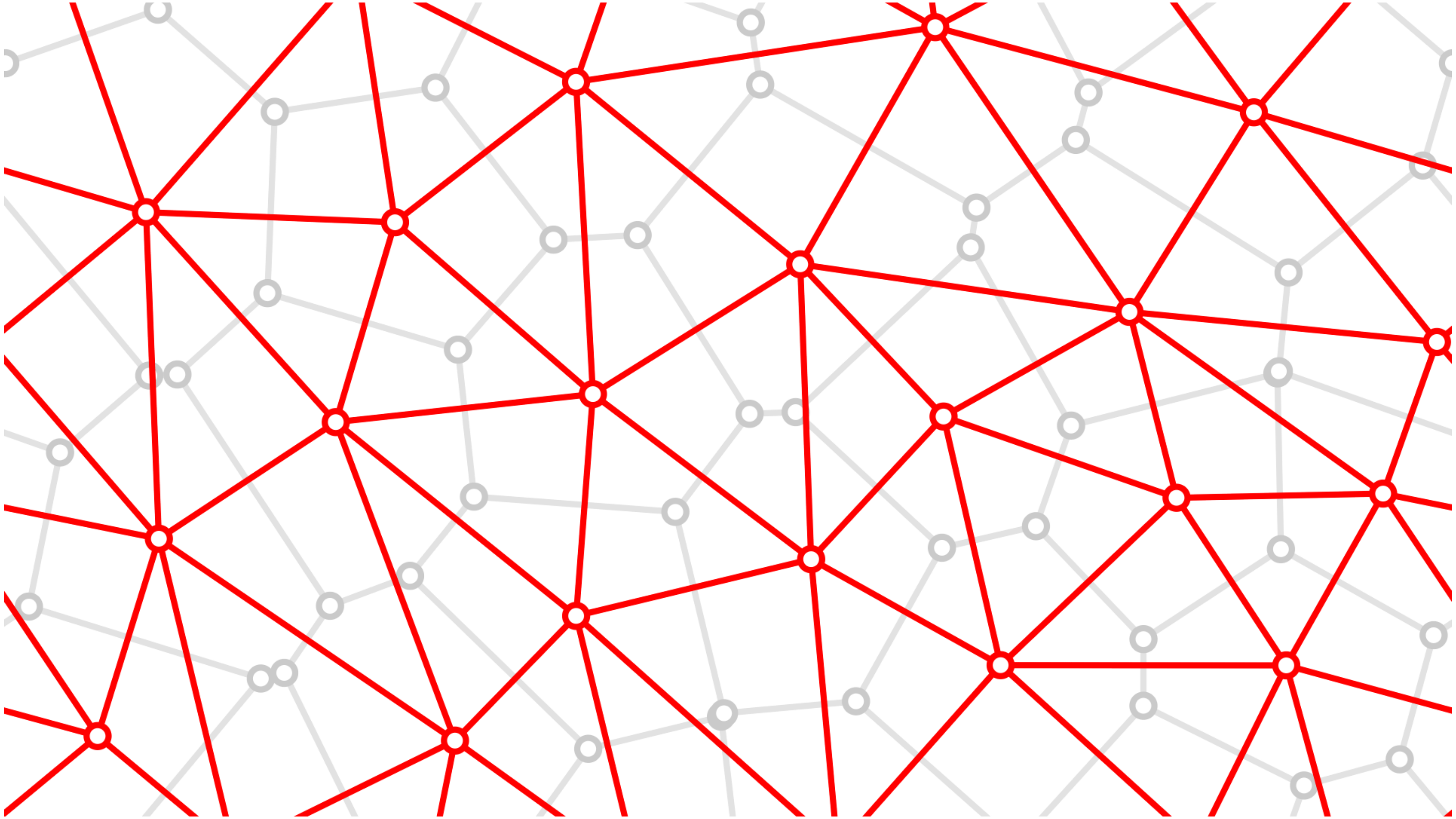
Relationship with Labelle-Shewchuk Approach

- **Key insight:** The Borouchaki-George approach is related to the well-known Labelle-Shewchuk approach

Labelle and Shewchuk, “Anisotropic Voronoi diagrams and guaranteed-quality anisotropic mesh generation,” *Proceedings of the Nineteenth Annual Symposium on Computational Geometry*, 2003

- The Borouchaki-George approach uses circumhyperspheres
- The Labelle-Shewchuk approach uses Voronoi cells
- The computational cost of both approaches is low, as no geodesic calculations are required

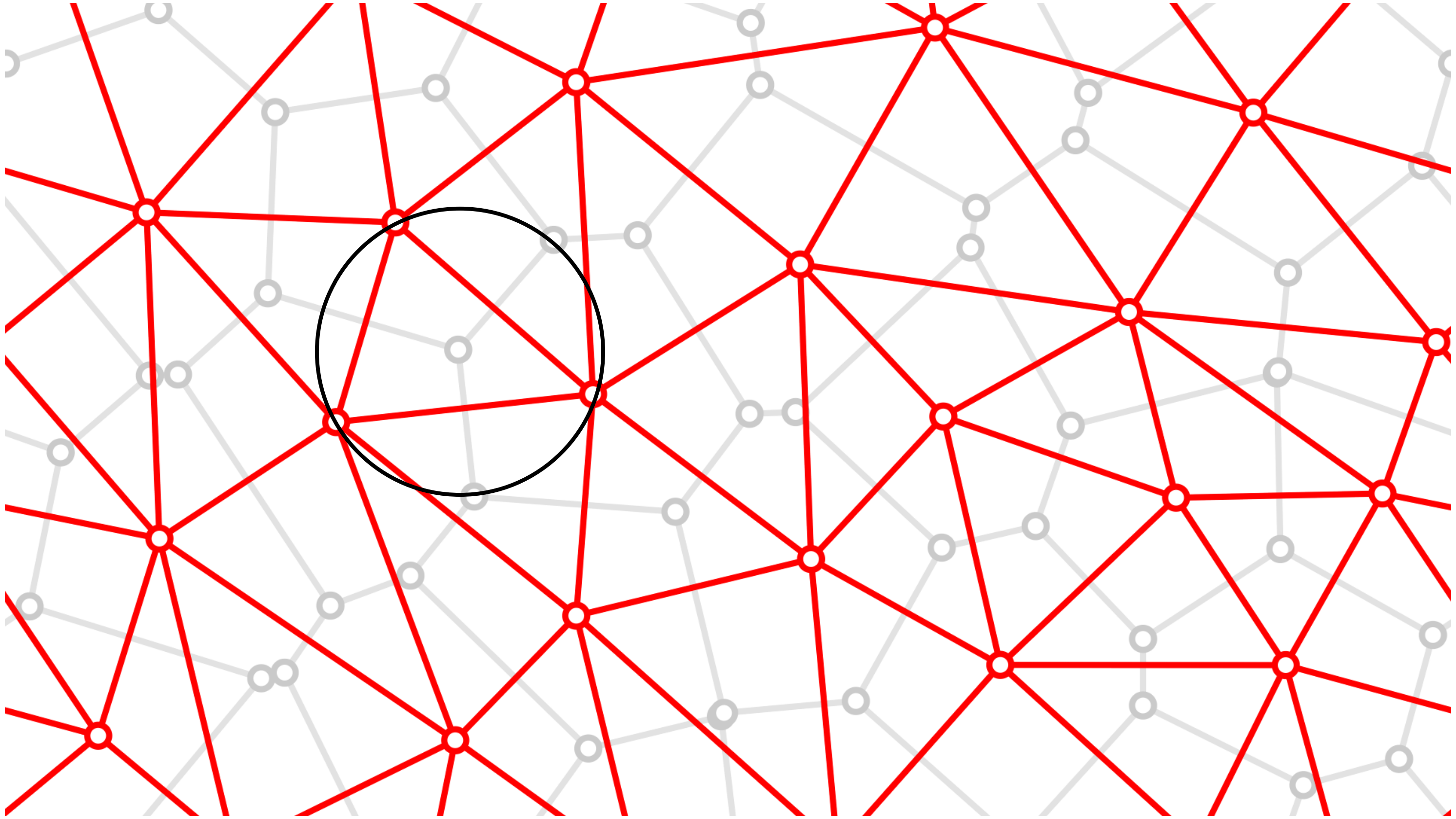
Duality of Voronoi Diagrams and Delaunay Triangulations



[6]

Duality of Voronoi Diagrams and Delaunay Triangulations

- **Key idea:** the vertices of anisotropic Voronoi cells coincide with the centers of hyperellipsoids associated with each simplex in the anisotropic Delaunay mesh



Relationship with Labelle-Shewchuk Approach

- **Key insight:** the Labelle-Shewchuk approach based on anisotropic Voronoi diagrams is preferred to the Borouchaki-George approach based on circumhypersphere tests

- **Reasoning:** the mathematical properties of anisotropic Voronoi diagrams are easier to understand and assess relative to the mathematical properties of anisotropic circumhyperspheres (hyperellipsoids)

  - Anisotropic Voronoi cells are effectively dual polytopal proxies for hyperellipsoids
  - Undesirable intersections between anisotropic Voronoi cells are easier to characterize and prevent

Labelle-Shewchuk Approach: Challenges

- Unfortunately, the Labelle-Shewchuk approach is not guaranteed to produce valid meshes in all cases …
- The conditions under which this approach will work have been sketched:

Chapter 5 of: Rouxel-Labbé “Anisotropic mesh generation,” PhD Thesis, Université Côte d'Azur (2016)

- Rigorous results and associated algorithms are still pending

Star Consistency

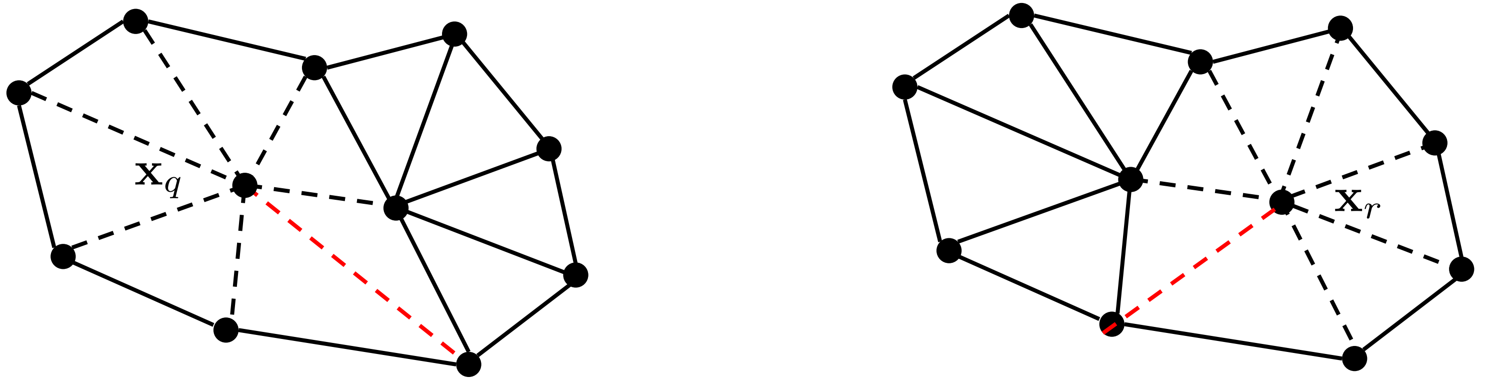
- **Key Problem:** if the metric field changes rapidly in a small region, then the metric field at adjacent inserted points *q* and *r* may be very different. In turn, the anisotropic Delaunay <u>stars</u> of these points may be inconsistent, i.e. they may not contain the same anisotropic elements

$$\mathbf{M}(\mathbf{x}_q) \not\approx \mathbf{M}(\mathbf{x}_r)$$

- **Star:** the collection of elements in a Delaunay mesh which contain a particular point

Star Consistency

- **Star inconsistency example:**



- Adjacent points *q* and *r* may disagree about whether a particular anisotropic Delaunay element can be formed, as they will be computing the Delaunay criterion with respect to different metrics

Impacts and Objectives

- **Practical Impacts:**
  - The anisotropic Delaunay criterion will not be enforced in a consistent fashion
  - The connectivity of the mesh will depend on the order in which points were inserted
  - We lose any mathematical guarantees of optimality

- **Long-Term Objectives:**
  - Construct a valid anisotropic Voronoi diagram using the approach of Labelle and Shewchuk
  - Extract an anisotropic Delaunay triangulation from the anisotropic Voronoi diagram using duality

Open Problems and Solution Strategies

Problem Statement

- **Informally:**
  - Determine conditions under which the anisotropic Delaunay criterion can be properly enforced--- maintaining consistent stars between adjacent points
  - Construct procedure for inserting points in regions where the metric field varies rapidly

$$\mathbf{M}(\mathbf{x}_q) \sim \mathbf{M}(\mathbf{x}_r)$$

- **Formally:**
  - <u>Determine conditions under which the Labelle-Shewchuk Voronoi diagrams and the well-studied Riemannian Voronoi diagrams are combinatorially equivalent</u>

Background

- Riemannian Voronoi diagrams have already been investigated
- Mathematical guarantees and conditions regarding the existence of anisotropic Delaunay triangulations have already been obtained:

> Boissonnat, Rouxel-Labbé, Wintraecken, "Anisotropic triangulations via discrete Riemannian Voronoi diagrams," *SIAM Journal on Computing*, 2019

- Constructing meshes directly using this approach is difficult; it requires the computation of approximate geodesics in metric space

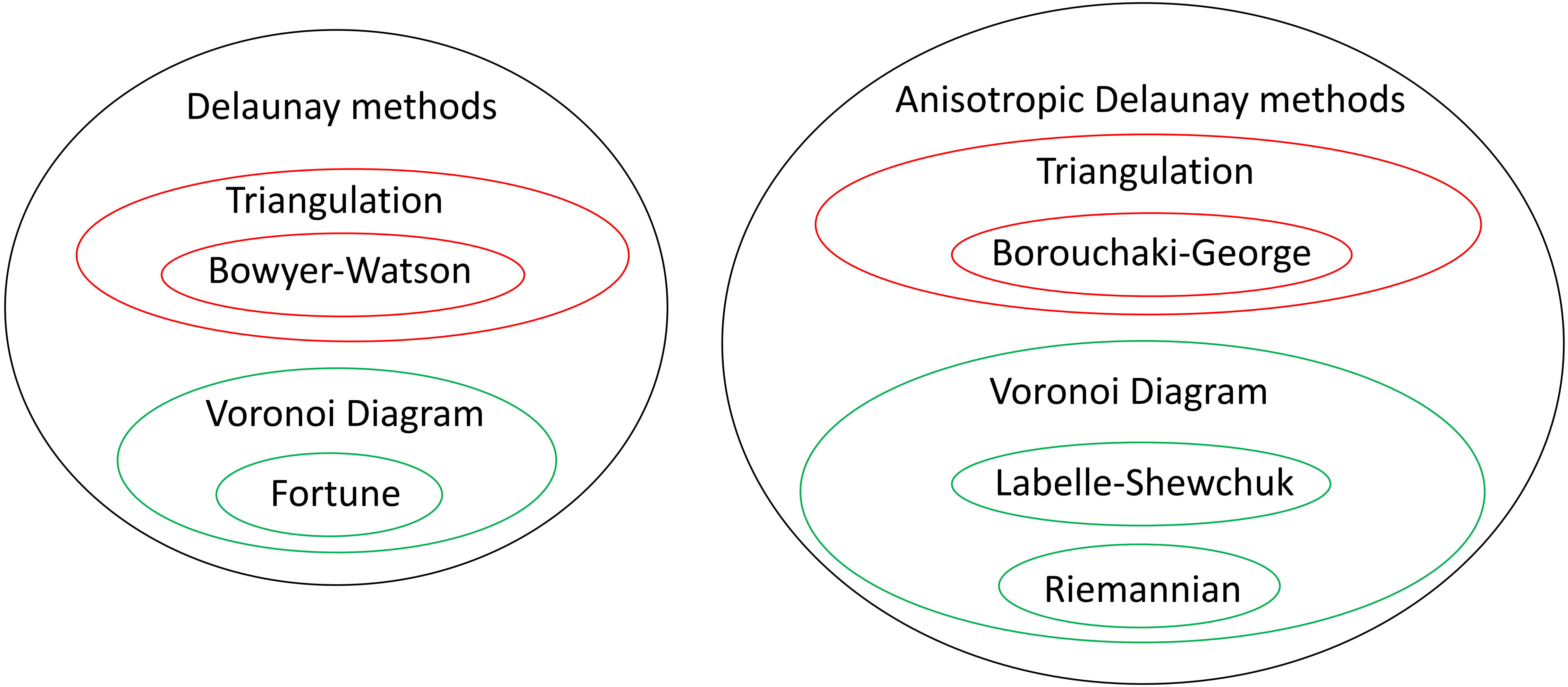
Background

- Approximate geodesic computations are expensive, and involve many evaluations of the metric field in relatively small regions of space
- Some clever approaches exist however:

Campen, Heistermann, and Kobbelt, "Practical anisotropic geodesy," *In Proceedings of the Eleventh Eurographics/ACMSIGGRAPH symposium on Geometry Processing*, 2013

- We hope to bypass this entirely by constructing a Labelle-Shewchuk approach with rigorous theoretical guarantees

Landscape of Delaunay Methods



Important Proofs

Triangulation

Proof of Validity: We can recover a valid mesh of non-degenerate, non-overlapping elements from the Labelle-Shewchuk Voronoi diagram.

Labelle-Shewchuk

Proof of Equivalence: We can recover the exact Riemannian Voronoi diagram from the Labelle-Shewchuk diagram. Demonstrate combinatorial equivalence of Voronoi diagrams.

Riemannian

Subsequent Problems

- **Problem 1:** Construct a computationally tractable algorithm for increasing the protection of anisotropic Delaunay meshes. Derive mathematical guarantees on the amount of protection provided.

- **Problem 2:** Identify the particular class of space-time problems for which the protected anisotropic Delaunay meshes are optimal.

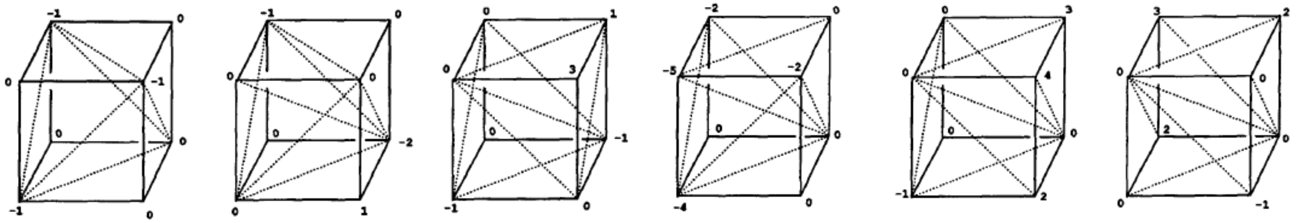
- **Problem 3:** Determine if the flip graph (or extended flip graph) in four dimensions is connected for a finite number of points. In other words, can we transform any mesh of a finite set of points into any other mesh of the same points through basic or extended flip operations?

The Flip Graph Problem

- Previous work:
  - The flip graph and extended flip graph are guaranteed to be connected in 2D
  - Connectedness of the flip graph and extended flip graph in 3D and 4D is unknown
  - The flip graph is not connected in 5D (Santos 2000, 2005)

- **Key result:** The flip graph of the tesseract (4-cube) is connected

Pournin, "The flip-graph of the 4-dimensional cube is connected," *Discrete & Computational Geometry*, 2013



[7]

Summary

- **Objective:** establish a strong mathematical framework for generating optimal, Delaunay space-time meshes. In turn, this requires mathematical results governing the construction of protected, anisotropic Delaunay meshes for finite sets of points

- **Key problems:**
  a) Establishing combinatorial equivalence of Voronoi Diagrams
  b) Establishing computationally tractable algorithms for increasing protection
  c) Identifying classes of space-time problems for which meshes are optimal
  d) Establishing connectedness of the flip graph

Questions

Figure References

1. Mewat, Haryana, https://altexploit.wordpress.com/tag/riemannian-geometry
2. Rankin, Brent, “Performance, applications, and analysis of rotating detonation engine technologies”, PrePrint, 2015
3. Nicky Alice, https://www.nickyalice.com/artforpurchase/tesseract-hypercube-infinity-mirror-sculpture-nicky-alice
4. MathWorks, https://www.mathworks.com/help/matlab/math/delaunay-triangulation.html
5. Marco Congedo, https://marco-congedo.github.io/PosDefManifold.jl/dev/introToRiemannianGeometry/
6. Freya Holmer, https://x.com/FreyaHolmer/status/1252636033654685697?lang=bg
7. Jesus A. De Loera, “Nonregular triangulations of products of simplices,” *Discrete & Computational Geometry*, 1996