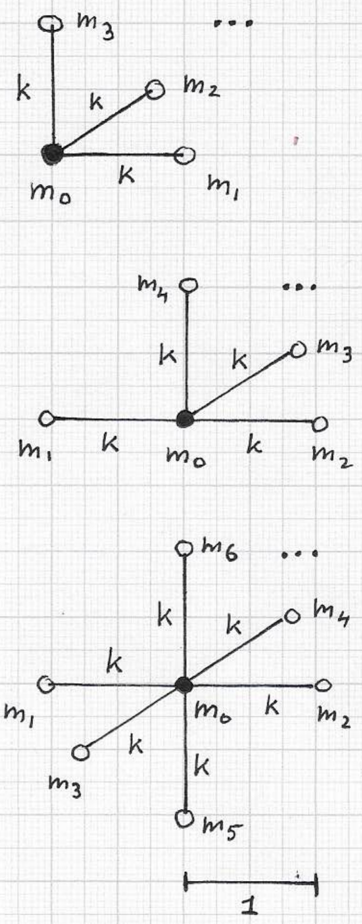
■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions: Considering the discussion of mass-spring system models and the differential equations resulting from such models, it is possible to view the voxels defining a segment/object, together with the edges connecting the centers of neighbor voxel pairs, as mass points with the connecting edges as springs. A simple and sufficient model can use the voxel-specific masses m_i as masses for the respective mass points $\mathbf{x}_i$ and a single, global spring constant k for all springs – which, in the case of a voxels all of dimension $1 \times 1 \times 1$, have constant length 1. Further, for pure feature computation purposes it suffices to only consider the homogeneous equation type "$m\ddot{\mathbf{x}} + k\mathbf{x} = \mathbf{0}$" for all voxels. Since a voxel can have at most six face neighbors, at most six springs and six voxel neighbor masses must be considered when determining the i^{th} homogeneous equation of voxel i with mass m_i. As a consequence of the fact that no mass point can have a valence/degree larger than six, the matrix representing the system of homogeneous equations is sparse and its bandwidth can be minimized via the Cuthill–McKee algorithm.



"Building blocks" of a simple mass-spring model for a segment consisting of a set of voxel with associated masses. A specific voxel 0 can have between 0 and 6 neighbor voxels; three possible local neighborhoods are shown. When assembling all equations in matrix form, one obtains an eigenvalue problem of dimension N – where N is the number of voxels. A "small number" of eigenvalues should define a "good spectral feature."

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

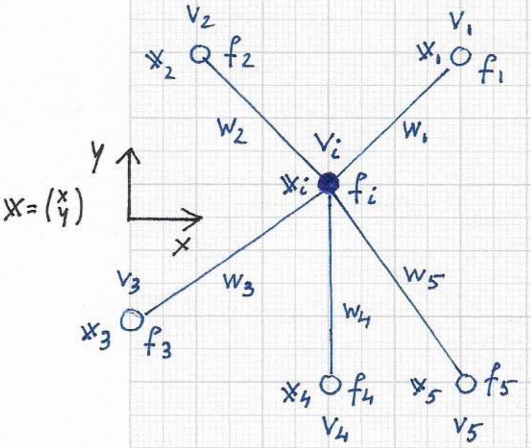
• Laplacian eigenfunctions: The discussed mass–spring model approach for computing a feature (vector), a spectral signature, based on the eigenvalues of the resulting eigenvalue problem, could become computationally expensive for "large" voxel segments. Nevertheless, it is likely that a relatively small number of eigenvalues ("the 10 largest ones") suffices for classification purposes. Highly efficient eigenvalue computation tools exist and can be used.

NEVERTHELESS, SINCE OUR CLASSIFICATION PROBLEM CONCERNS DISCRETE OBJECTS – SETS OF VOXELS WITH ASSOCIATED MASSES AND IMPLIED VOXEL NEIGHBORHOODS – AN ALTERNATIVE METHOD CAN BE DEVISED BASED ON THE **DISCRETE LAPLACE OPERATOR** AND THE **GRAPH LAPLACIAN**.

Graph Laplacian approaches are used with success for the computation of eigenvalues resulting from 2D surface meshes embedded in 3D space, for purposes like shape signature definition, shape retrieval and shape segmentation. Our application is concerned with solid object signature computation for 3D volumetric meshes embedded in **4D** $(x, y, z, mass(x,y,z))^T$–space.

$$\begin{bmatrix} & & \\ & M & \\ & & \end{bmatrix} \begin{bmatrix} \\ e \\ \\ \end{bmatrix} = \lambda \begin{bmatrix} \\ e \\ \\ \end{bmatrix}$$

General nature of eigen-vector/-value computational problem; M is a band-diagonal matrix with its dimensions defined by the number of a segment's voxels. Most likely, a small number of eigenvalues define a segment's signature, its "fingerprint."

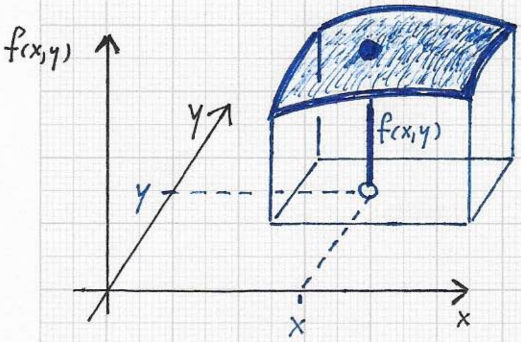


Local data needed to define discrete Laplacian value for vertex v_i of a graph with nodes in the plane with associated function values; edges between v_i and v_j, $j=1...5$, have weights w_j. Laplacian

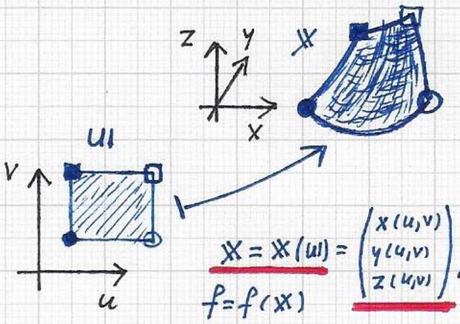
$$\Delta(v_i) = \sum_{j=1}^{5} w_j \, | f_j - f_i | .$$

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions:



"Standard" Laplacian operator is applied to a bivariate function $f(x,y)$. The surface $(x, y, f(x,y))^T$ is a special parametric surface: it represents the graph of the bivariate function $f(x,y)$.



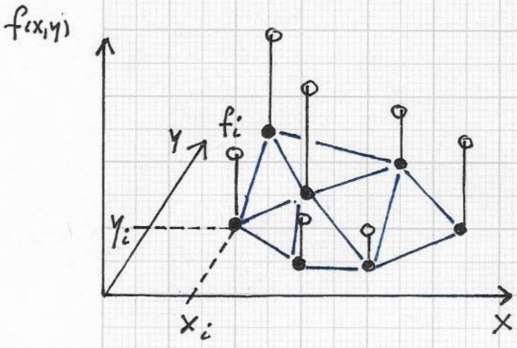
$$\mathbf{x} = \mathbf{x}(\mathbf{u}) = \begin{pmatrix} x(u,v) \\ y(u,v) \\ z(u,v) \end{pmatrix}, \quad f = f(\mathbf{x})$$

More general Laplace-Beltrami operator is applied to a bivariate function $f = f(\mathbf{x}(\mathbf{u}))$, for example. In this case, the function f can still be expressed in term of two local coordinates at a point $\mathbf{x}_0$ on the parametric surface – by using the local coordinate system at $\mathbf{x}_0$, defined by the tangent plane and normal at point $\mathbf{x}$. Thus, the "standard" Laplacian operator Δ can locally applied to the function f when expressed in local coordinates.

Before discussing the potential use of the discrete Laplace (and Laplace-Beltrami) operator for computing feature data for a 3D volumetric object/segment (= set of voxels with masses), it is helpful to review the Laplace operator Δ and its application to bivariate functions f – with domains that are planar regions in the 2D xy-plane or curved 2D surface manifold regions embedded in 3D xyz-space. Further, when making the transition from a continuous 2D domain to a discrete and gridded 2D domain representation for the function f, one must keep in mind that the discretization step makes it necessary to use specific finite difference formula based approximations of differential properties of f at certain grid vertices. The grid – IF ONE USES A GRID! – can consist of only triangles, only quadrilaterals, a combination of these two cell types, or simply of unit squares (if possible). Grid, function values at vertices and geometry define the values of the Laplacian Δ.

OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions:



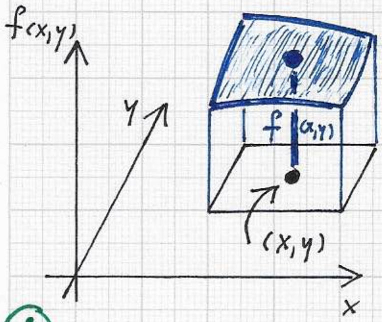
Simple discretization of bivariate function $f = f(x,y)$ by xy-values (x_i, y_i) with function values f_i. For example, a triangulation can be used to define a grid in the xy-plane. With the use of local finite difference formulas one can estimate values of Δf at the grid vertices and then solve the resulting eigenvalue problem $-\Delta f = \lambda f$.

Here: One "associates one eigenfunction with each vertex" in the grid. Since this grid has seven vertices, seven eigenfunctions will result, i.e., a seven-dimensional eigenvector per vertex. These seven eigenvectors/-functions define a quasi-hierarchical basis for the function f given in its discrete form.

Keeping in mind that the Laplacian operator Δ when applied to a bivariate function $f(x,y)$ is $\Delta f(x,y) = \nabla \cdot \nabla f = \text{div grad } f = \frac{\partial^2}{\partial x^2} f + \frac{\partial^2}{\partial y^2} f$, one must ask: (1) How can one compute the Laplacian of f when f is only known at a finite number of xy-values and a certain grid type is used? (2) As stated above, we are interested in the eigenvectors/eigenfunctions/eigenmodes defined by the general eigenvalue problem $-\Delta f = \lambda f$; how can we interpret the eigenvalues/eigenfunctions and use them (ultimately) for the computation of feature data for "our segments" to be classified? To simplify and describe the eigenfunction computation at a high level, one can think of the resulting eigenfunctions as "hierarchical basis functions" of lower and higher frequencies. Combinations of some of these eigenfunctions permit the reconstruction/approximation of f at increasing level of precision with an increasing number of eigenfunctions. In other words, the eigenvalues themselves define a SPECTRUM of the given function f being analyzed. In our application $f \approx$ mass, and a "mass spectrum" of a segment should be a very good segment feature. !

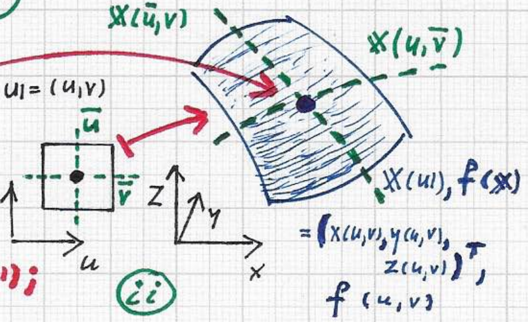
■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions:

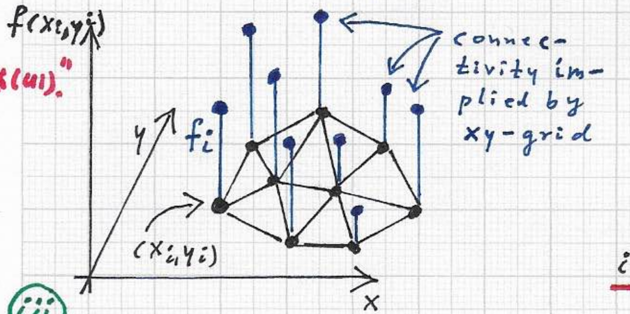


(i)

● must use curvi-linear coordinates to compute div grad $f(\mathbb{u})$; "grad f is a vector field on $\mathbb{X}(\mathbb{u})$."

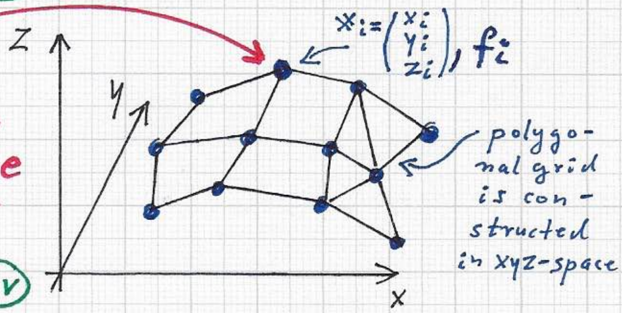


(ii)



(iii)

● must define local coordinate system to estimate Δf



(iv)

Basic settings for use of Laplace / Laplace-Beltrami operator: i) continuous, functional; ii) continuous, parametric; iii) discrete, functional; and iv) discrete, non-functional

The illustrations (left) show the main "data representations" one must consider when computing eigenfunctions and eigenvalues via the Δ operator applied to a function, i.e., a bivariate function in the shown prototypical cases. On a high level, one must solve the same eigenvalue problem in different settings:

$$\Delta f = -\lambda f \, .$$

i) In the continuous and "functional" setting, the function to be analyzed is a bivariate function $f(x,y)$, defined over a continuum domain in the xy-plane. The Δ operator is the Laplacian $\Delta f = \frac{\partial^2}{\partial x^2} f + \frac{\partial^2}{\partial y^2} f$.

ii) In the continuous and "parametric" setting, the function to be analyzed is a bivariate function $f(\mathbb{X}(\mathbb{u}))$, i.e., a FUNCTION OF A POINT $\mathbb{X} = (x,y,z)^T = (x(u,v), y(u,v), z(u,v))^T$ ON A PARAMETRIC SURFACE (= 2D MANIFOLD DOMAIN IN 3D SPACE), MAPPING A PARAMETER TUPLE (u,v) TO $\mathbb{X}(\mathbb{u})$. Thus, $f = \ldots = f(u,v)$. THE MORE GENERAL LAPLACE-BELTRAMI OPERATOR DESCRIBES THIS SITUATION.

iii), iv) These cases are the discrete versions of i), ii).