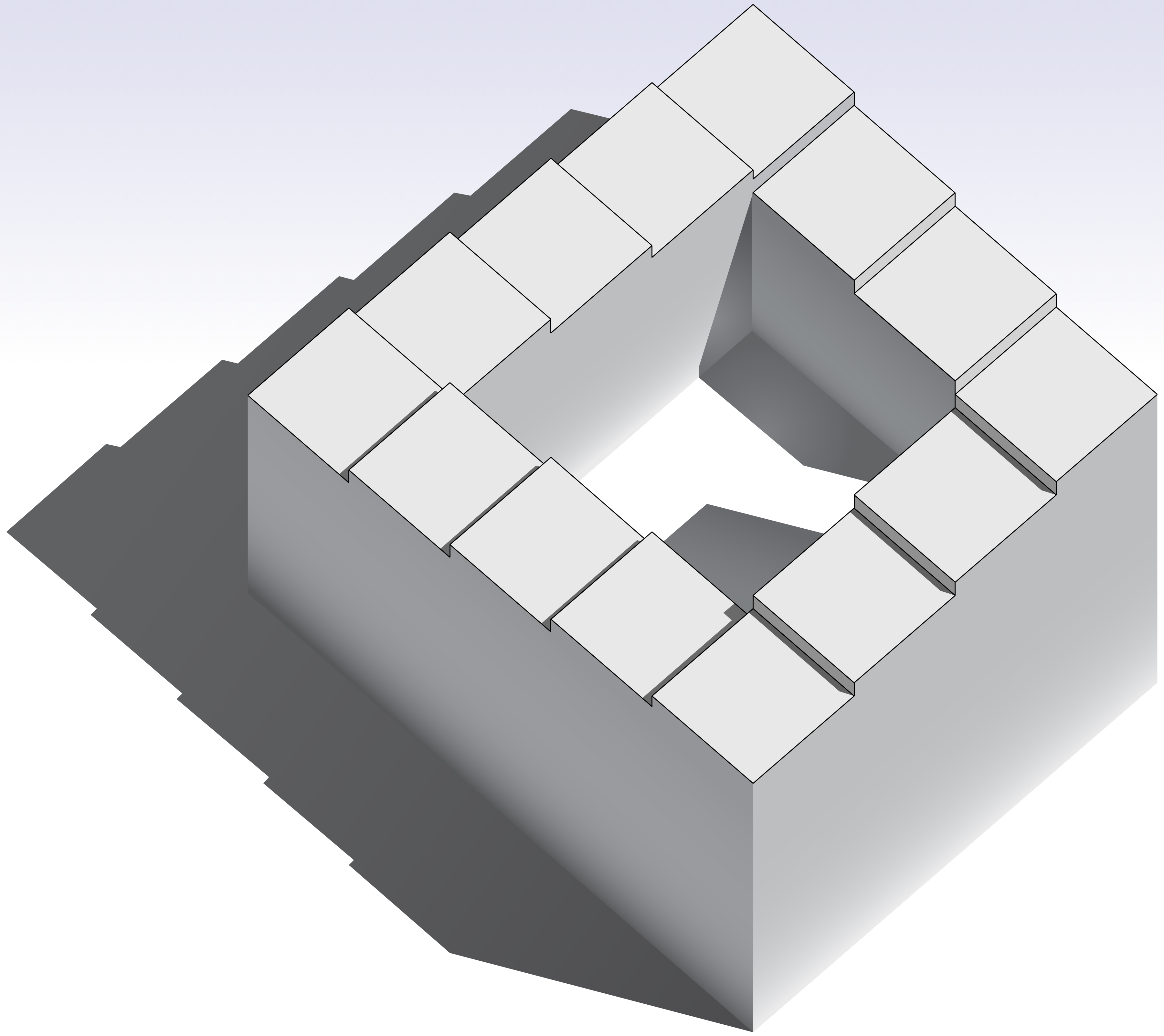
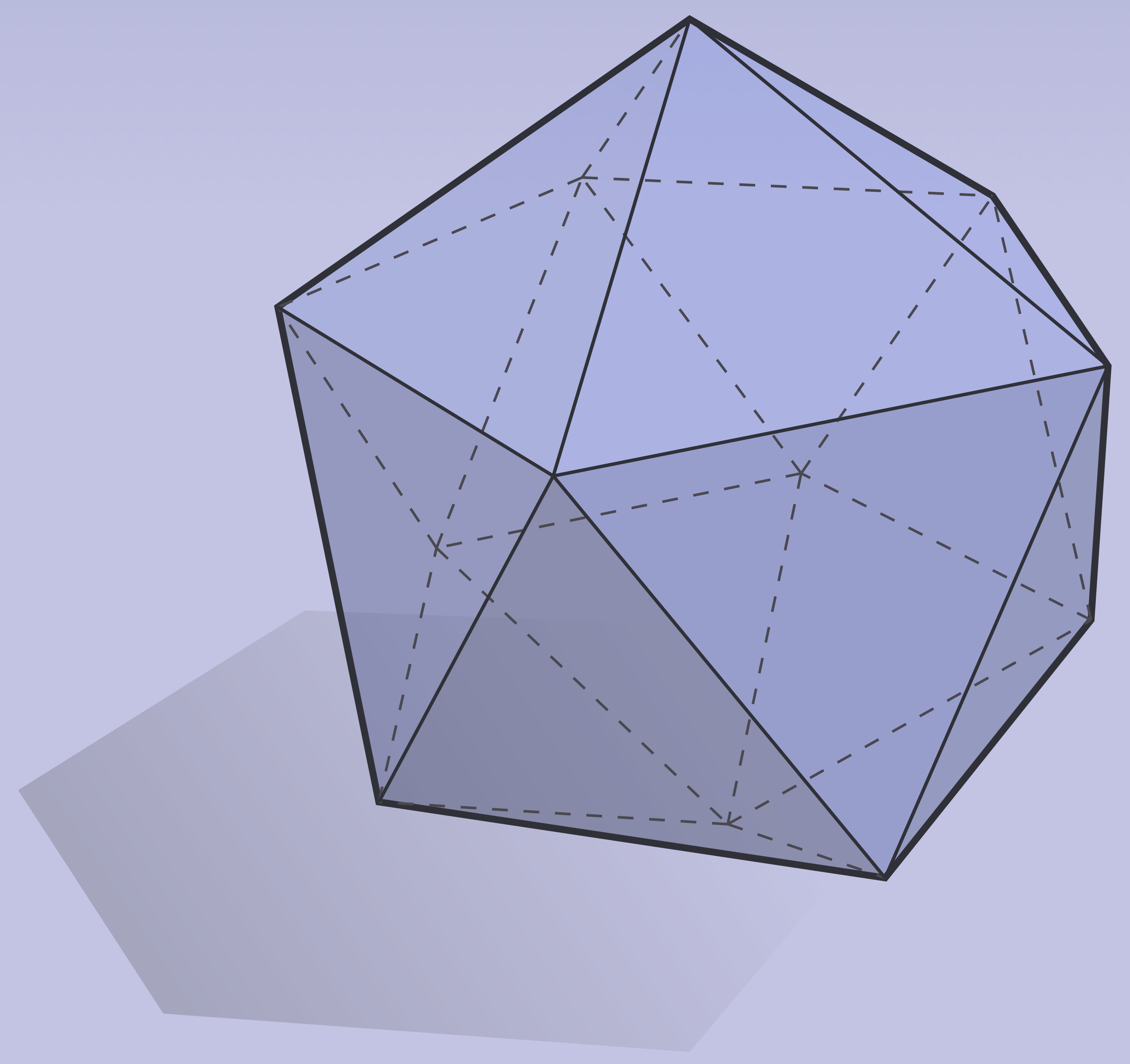


Discrete Differential Geometry: An Applied Introduction

Keenan Crane • CMU 15-458/858

LECTURE 8:
DISCRETE DIFFERENTIAL FORMS



DISCRETE DIFFERENTIAL GEOMETRY: AN APPLIED INTRODUCTION

Keenan Crane • CMU 15-458/858

Review—Exterior Calculus

- Last lecture we saw *exterior calculus* (differentiation & integration of forms)
- As a review, let's try *solving an equation* involving differential forms

Given: the 2-form $\omega := dx \wedge dy$ on $\mathbb{R}^2$

Find: a 1-form α such that $d\alpha = \omega$.

Well, *any* 1-form on $\mathbb{R}^2$ can be expressed as $\alpha = udx + vdy$ for some pair of coordinate functions $u, v : \mathbb{R}^2 \to \mathbb{R}$.

We therefore want to find u, v such that $du \wedge dx + dv \wedge dy = dx \wedge dy$.

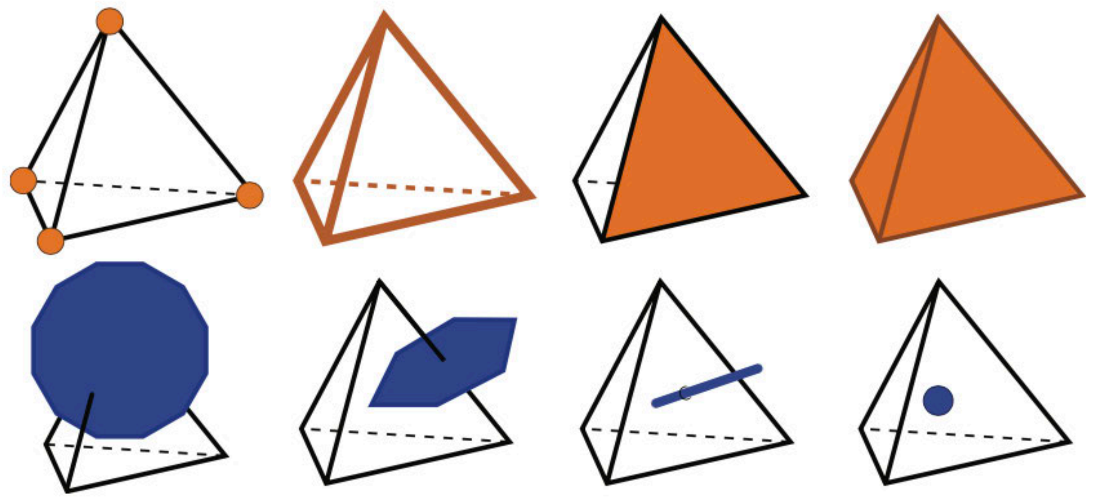
Recalling that $dx \wedge dy = -dy \wedge dx$, we must have $v = \frac{1}{2}x$ and $u = -\frac{1}{2}y$.

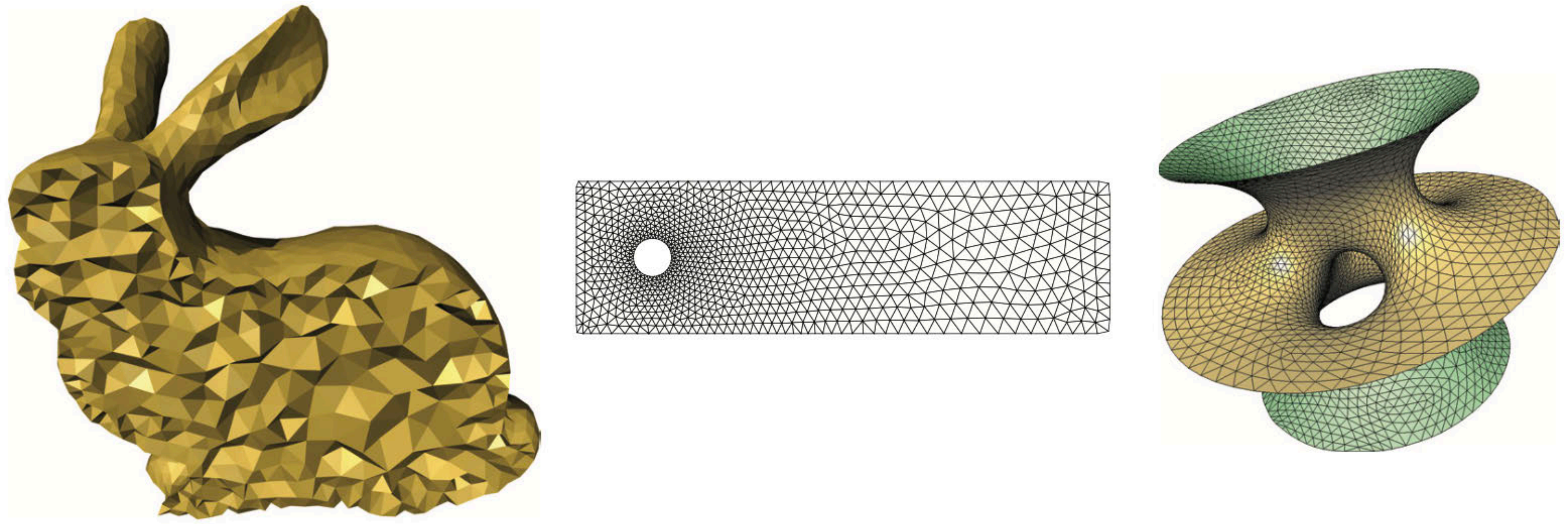
In other words, $\boxed{\alpha = \frac{1}{2}(xdy - ydx).}$

(…is that what you expected?)

Discrete Exterior Calculus—Motivation

- Solving even *very easy* differential equations by hand can be hard!
- If equations involve data, *forget* about solving them by hand!
- Instead, need way to approximate solutions via computation
- **Basic idea:**
  - replace domain with mesh
  - replace differential forms with values on mesh
  - replace differential operators with matrices





(pictures: Elcott et al, *"Stable, Circulation-Preserving, Simplicial Fluids"*)

Discrete Exterior Calculus—Basic Operations

- In smooth exterior calculus, we saw many operations (wedge product, Hodge star, exterior derivative, sharp, flat, …)
- For solving equations on meshes, the most basic operations are typically the **discrete exterior derivative** (d) and the **discrete Hodge star** ($\star$), which we'll ultimately encode as sparse matrices.

$$d\phi = \frac{\partial \phi}{\partial x^i} dx^i$$

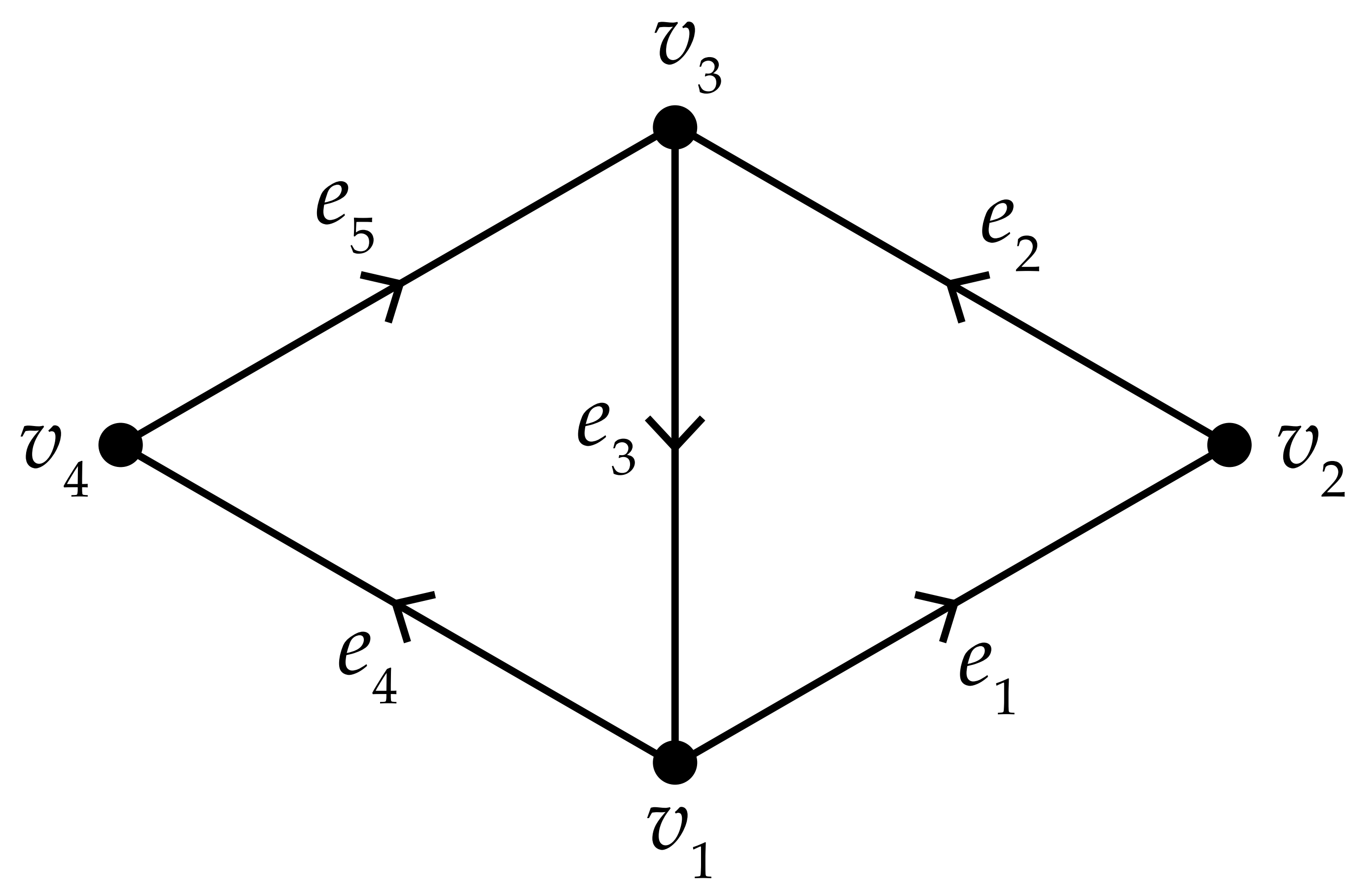
$$\downarrow$$

$$\begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{bmatrix}$$

$$\star(\alpha_1 dx^1 + \alpha_2 dx^2) = -\alpha_2 dx^1 + \alpha_1 dx^2$$

$$\downarrow$$

$$\begin{bmatrix} w_1 & 0 & 0 & 0 & 0 \\ 0 & w_2 & 0 & 0 & 0 \\ 0 & 0 & w_3 & 0 & 0 \\ 0 & 0 & 0 & w_4 & 0 \\ 0 & 0 & 0 & 0 & w_5 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix}$$



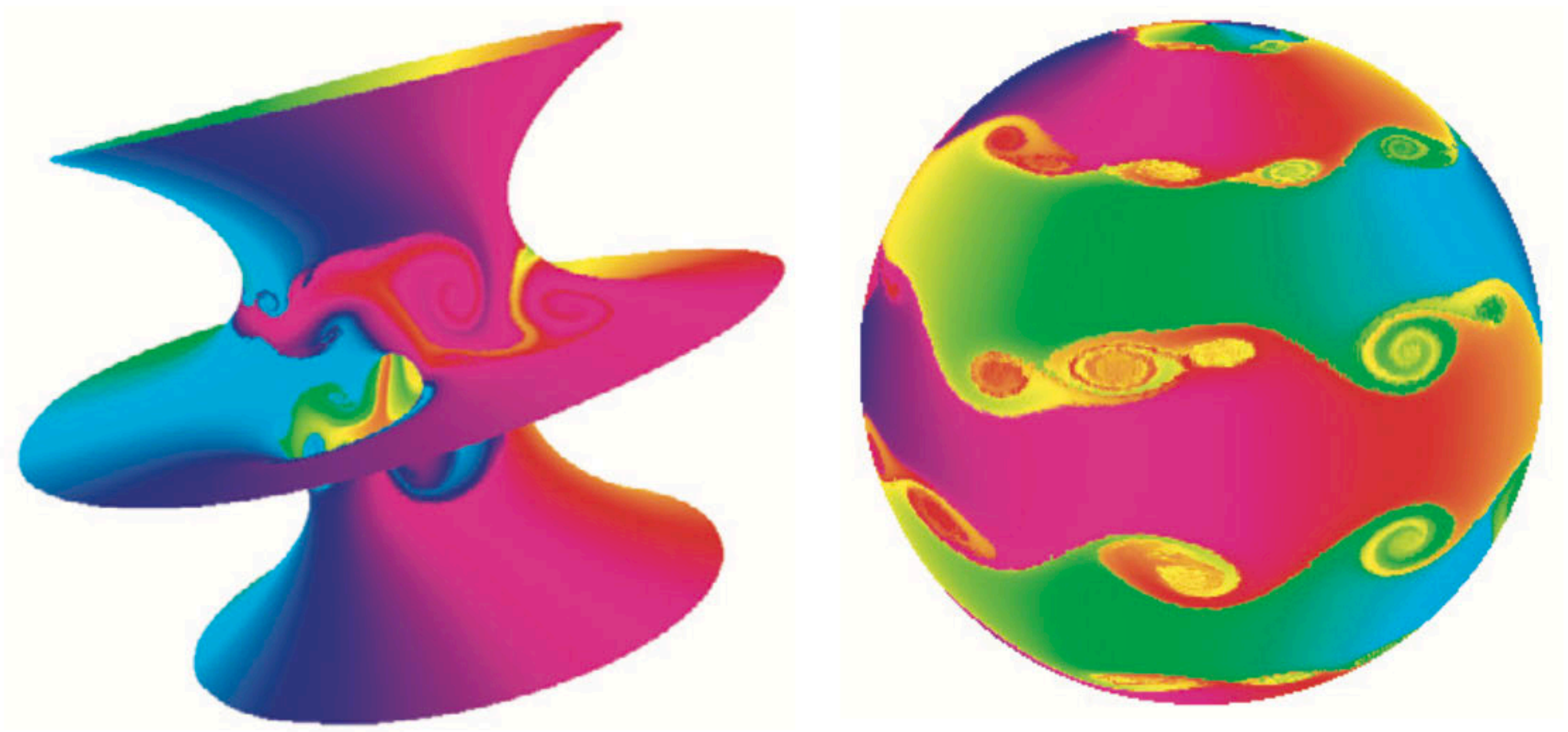
Composition of Operators

- By composing matrices, we can easily solve equations involving operators like those from vector calculus (grad, curl, div, Laplacian…) but in much greater generality (e.g., curved surfaces, k-forms…) and on complicated domains (meshes)

$$\text{grad} \longrightarrow d_0 \qquad \text{curl} \longrightarrow \star_2 d_1$$

$$\text{div} \longrightarrow \star_0^{-1} d_0^T \star_1$$

$$\Delta \longrightarrow \star_0^{-1} d_0^T \star_1 d_0$$

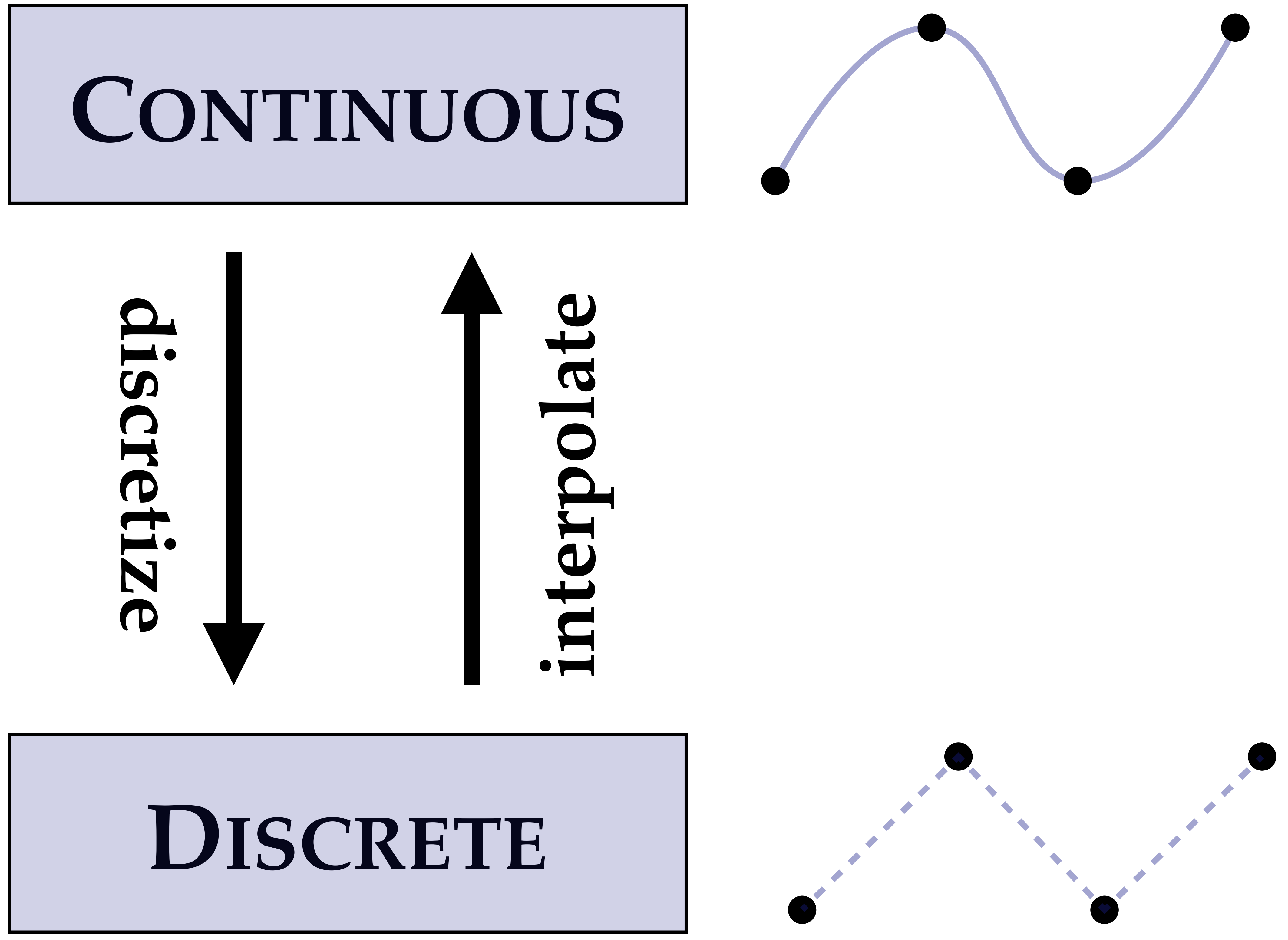


$$\Delta_k \longrightarrow d_{k-1} \star_{k-1}^{-1} d_{k-1}^T \star_k + \star_k^{-1} d_k^T \star_{k+1} d_k$$

Basic recipe: load a mesh, build a few basic matrices, solve a linear system.

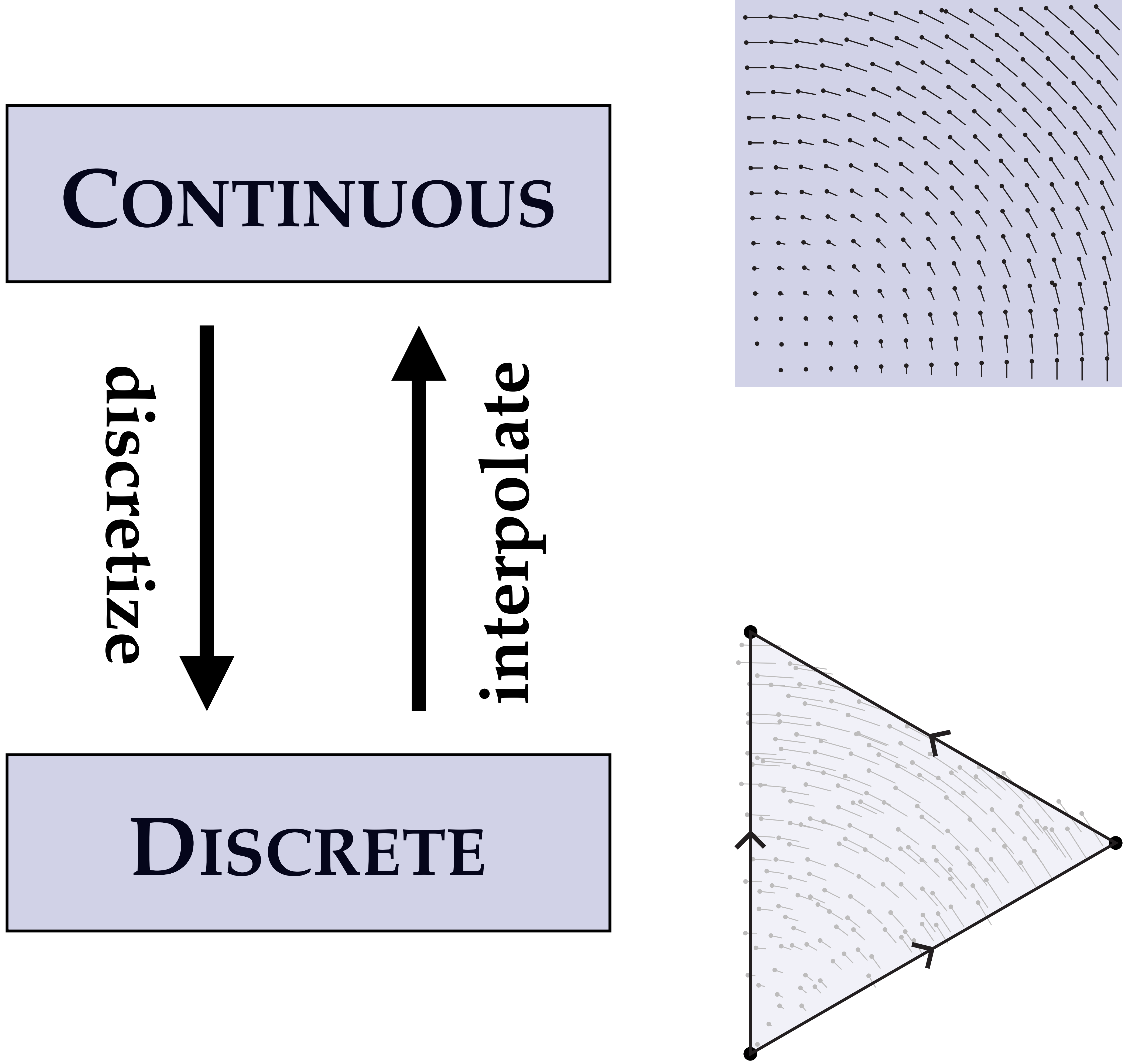
Discretization & Interpolation

- Two basic operations needed to translate between smooth & discrete quantities:
  - **Discretization** — given a continuous object, how do I turn it into a finite (or *discrete*) collection of measurements?
  - **Interpolation** — given a discrete object (representing a finite collection of measurements), how do I come up with a continuous object that agrees with (or *interpolates*) it?



Discretization & Interpolation — Differential Forms

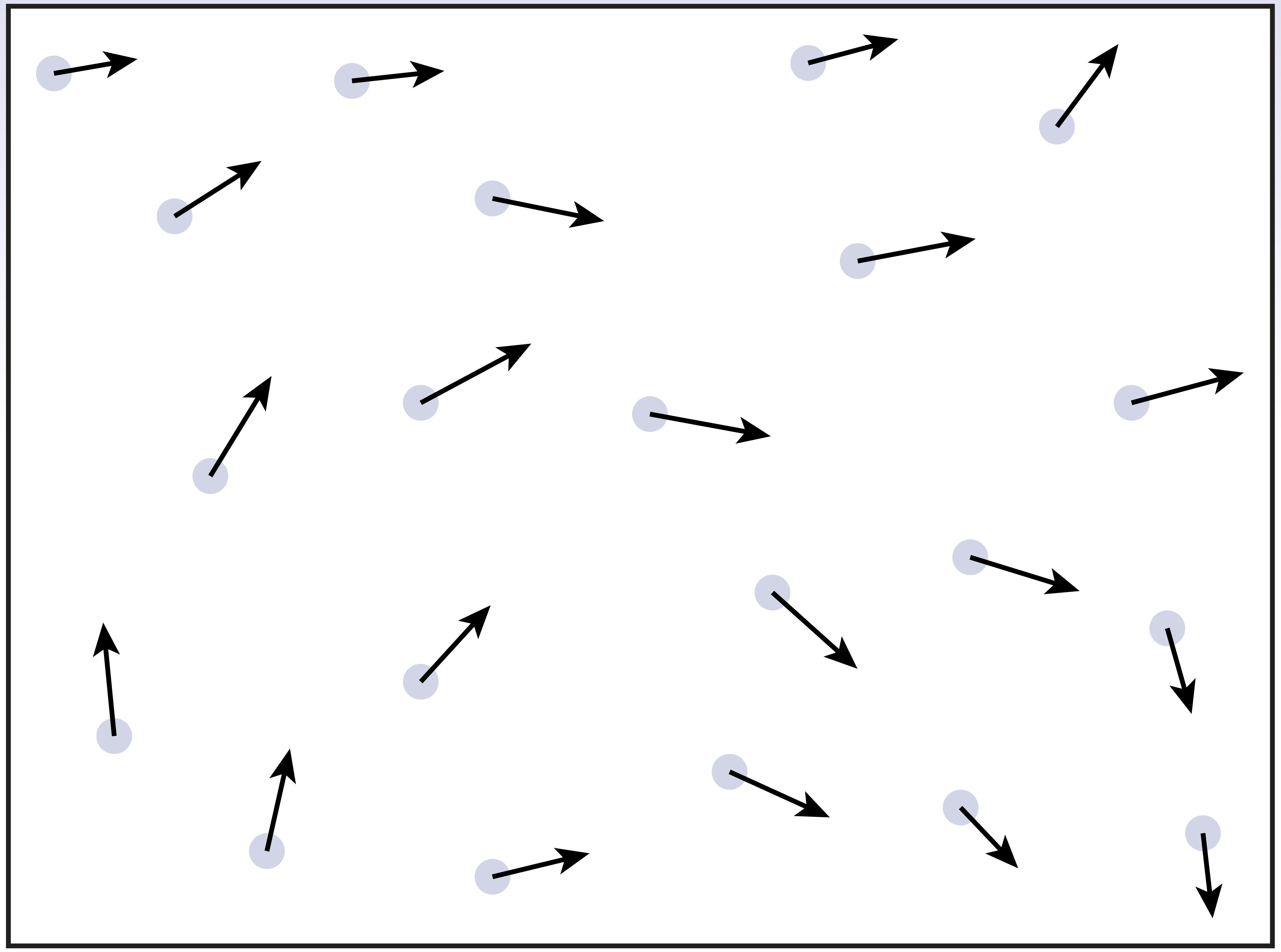
- In the particular case of a differential *k*-form:
  - **Discretization** happens via *integration* over oriented *k*-simplices (known as the *de Rham map*)
  - **Interpolation** is performed by taking linear combinations of continuous functions associated with *k*-simplices (known as *Whitney interpolation*)
- With these operations, becomes easy to translate some pretty sophisticated equations into algorithms!

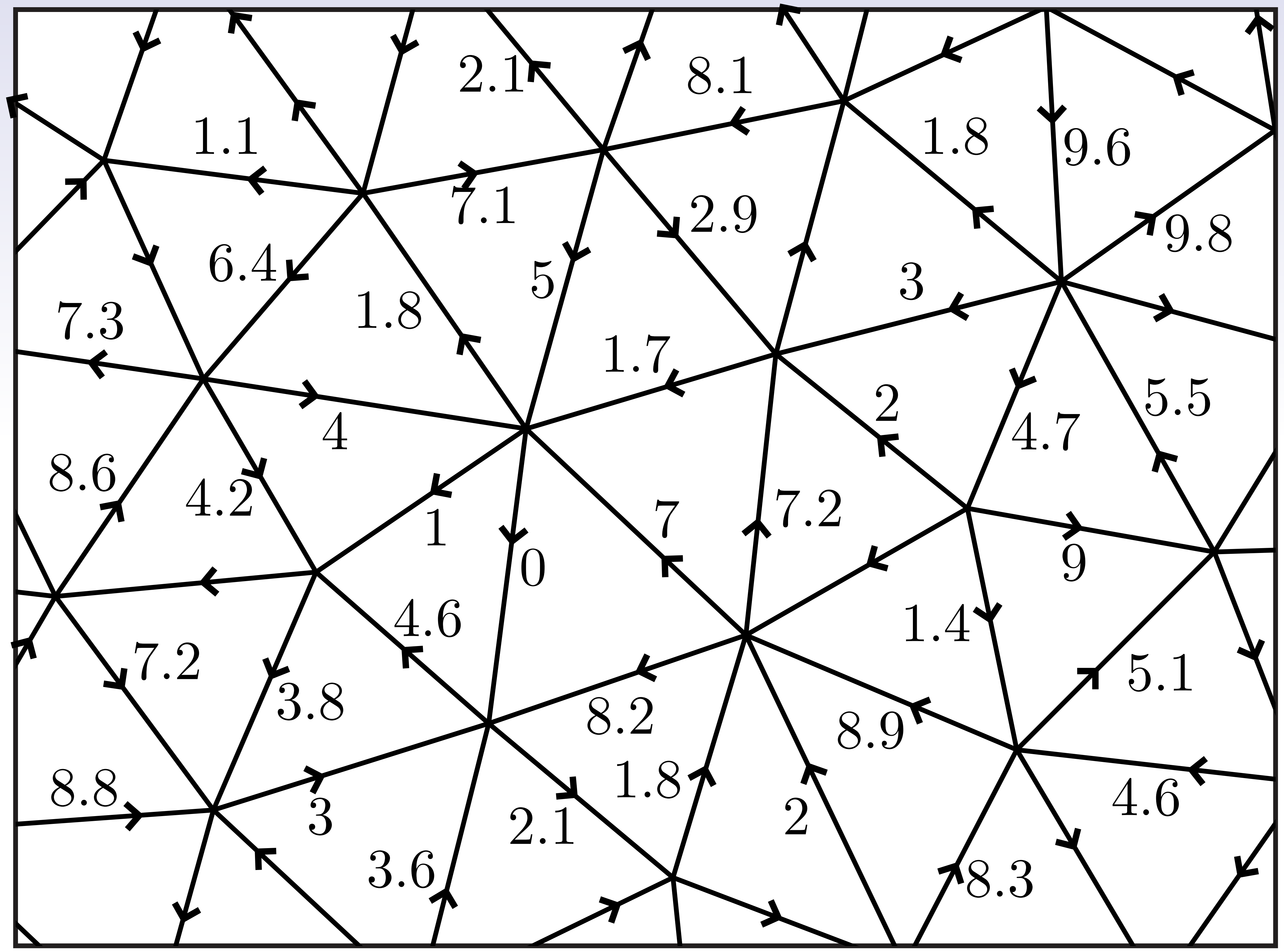


Discretization

Discretization—Basic Idea

Given a continuous differential form, how can we approximate it on a mesh?





Basic idea: integrate *k*-forms over *k*-simplices.

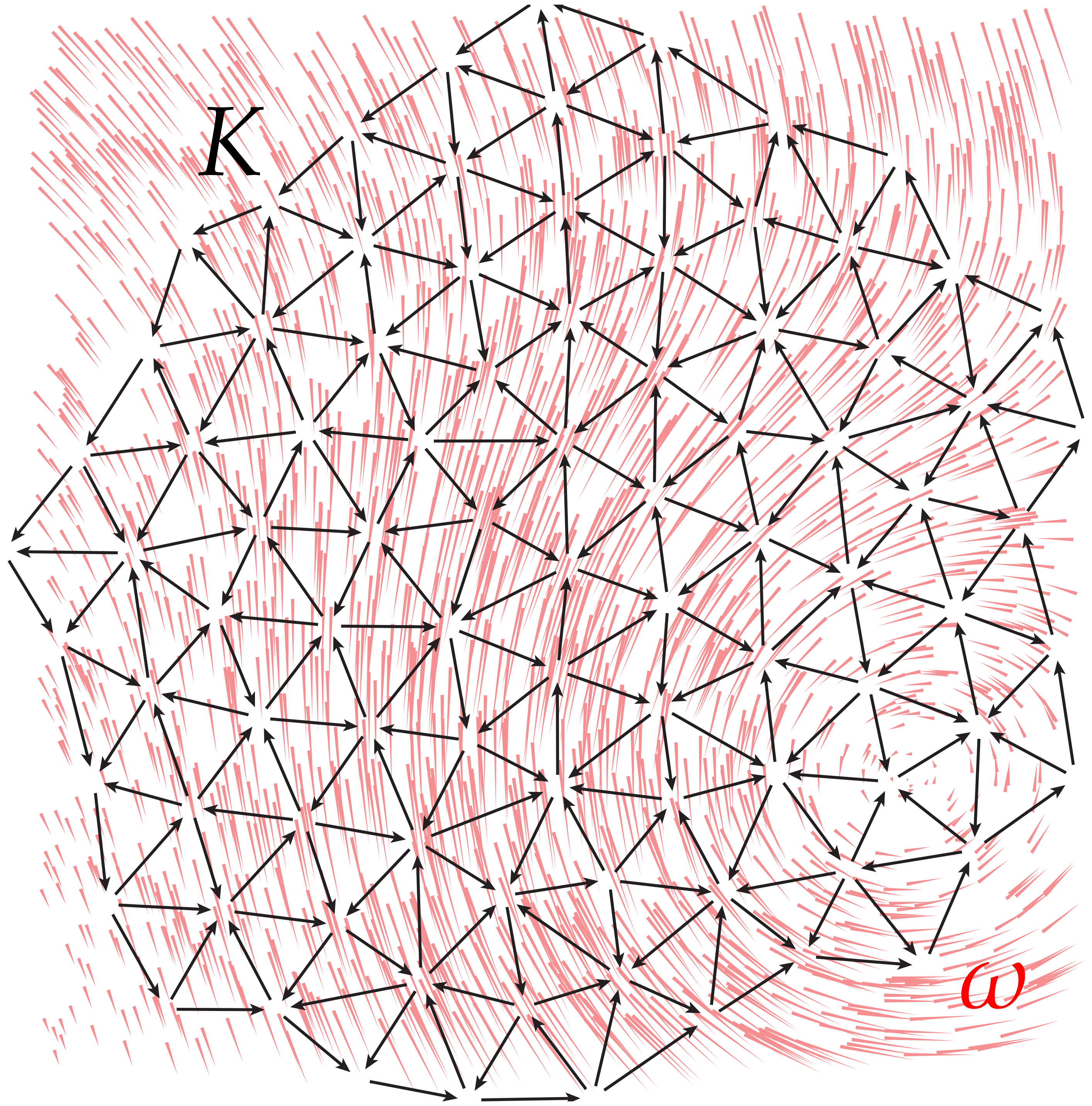
Doesn't tell us *everything* about the form… but enough to solve interesting equations!

Discretization of Forms (de Rham Map)

Let K be an oriented simplicial complex on $\mathbb{R}^n$, and let ω be a differential k-form on $\mathbb{R}^n$. For each simplex $\sigma \in K$, the corresponding value of the discrete k-form $\hat{\omega}$ is given by

$$\boxed{\hat{\omega}_\sigma := \int_\sigma \omega}$$

The map from continuous forms to discrete forms is called the *discretization map*, or sometimes the *de Rham map*.

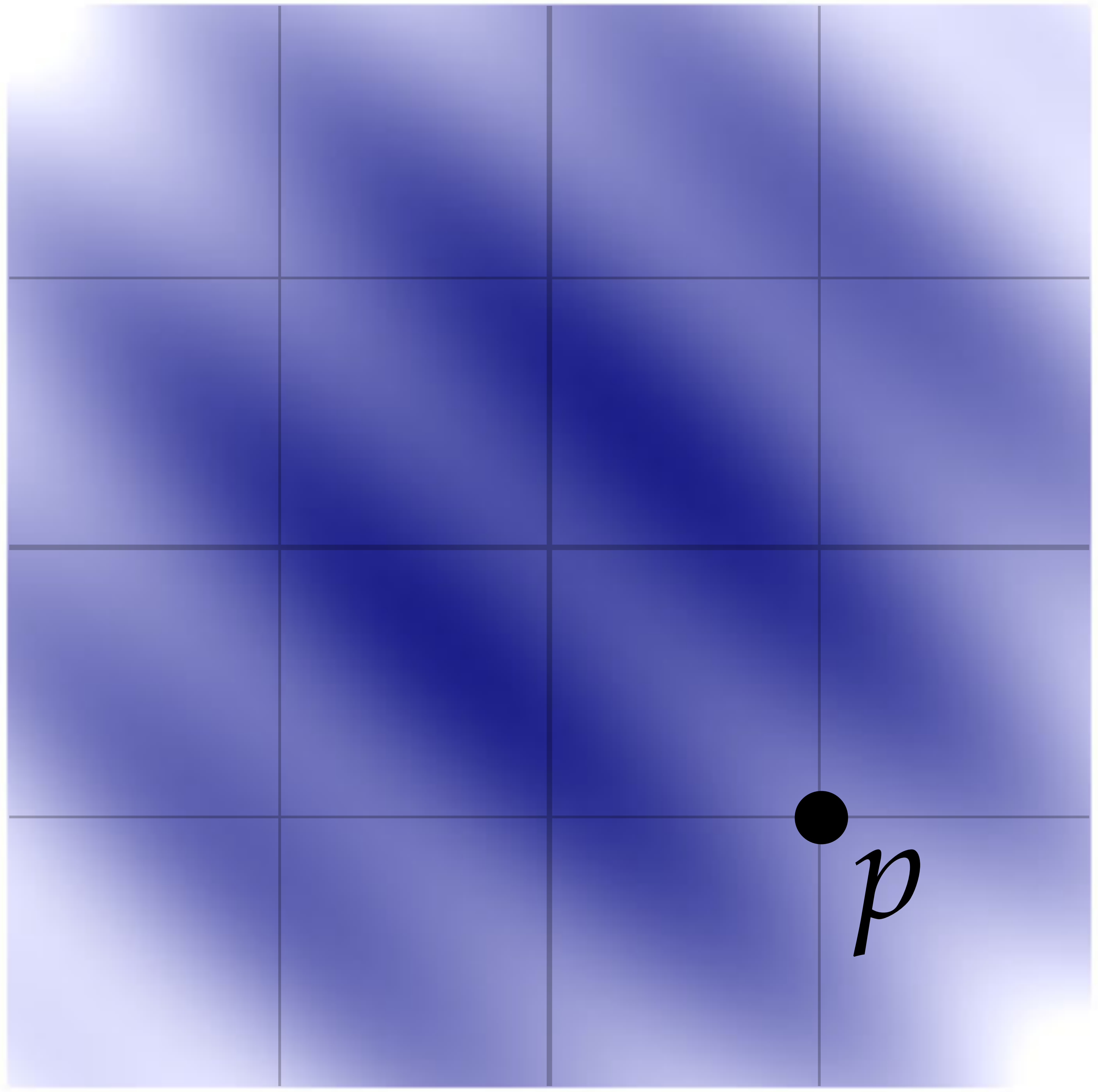


$$\mapsto \begin{bmatrix} 1.27 \\ 8.34 \\ -4.92 \\ 0.01 \\ \vdots \\ 2.96 \end{bmatrix}$$
$\hat{\omega}$

Key idea: *discretization* just means "integrate a k-form over k-simplices." Result is just a list of values.

Integrating a 0-form over Vertices

- Suppose we have a 0-form ϕ
- What does it mean to integrate it over a vertex v?
- Easy: just take the value of the function at the location p of the vertex!



Example:

$$\phi(x,y) := x^2 + y^2 + \cos(4(x+y))$$

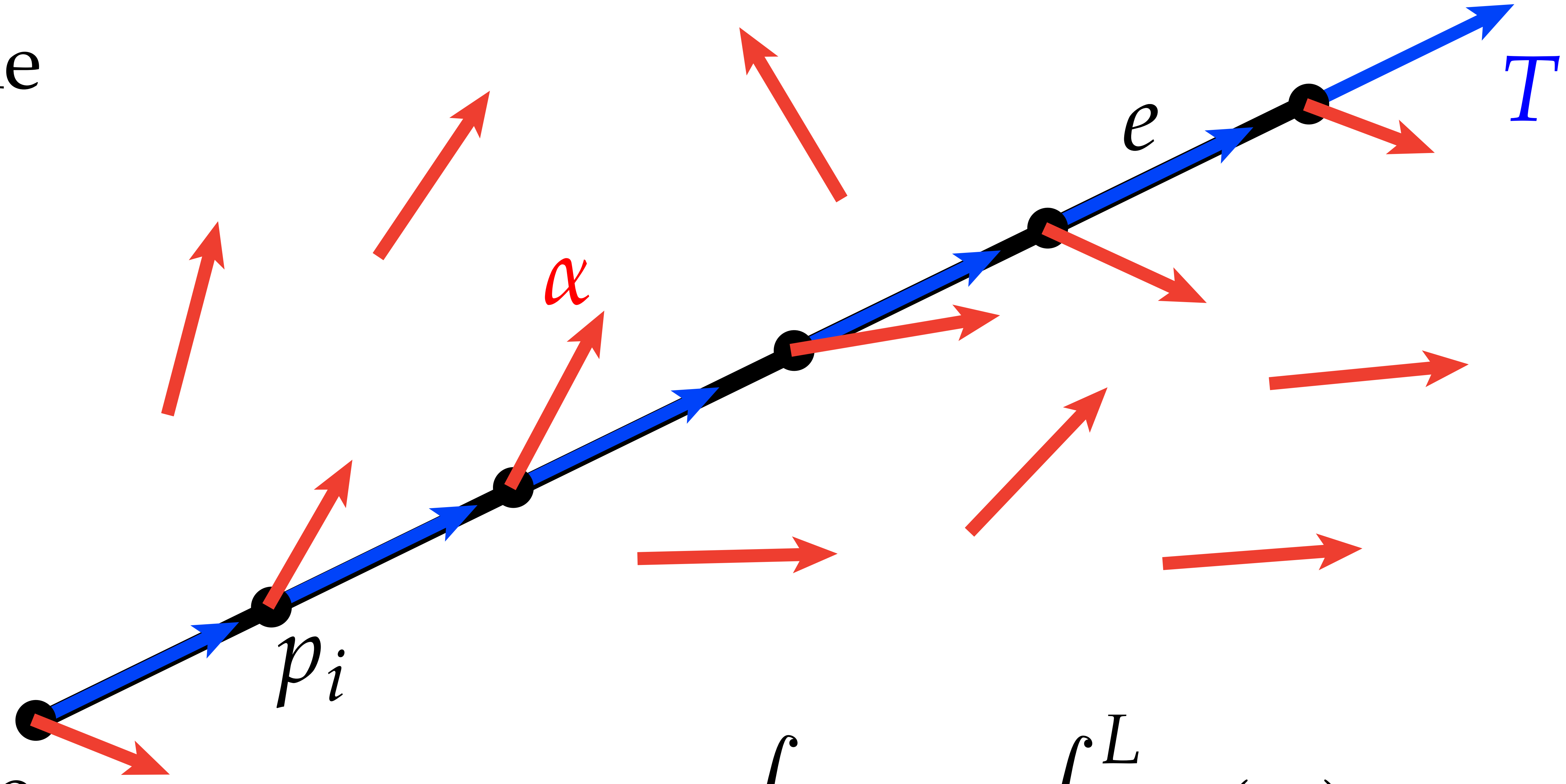
$$p = (1,-1)$$

$$\int_v \phi = \phi(p) = 1 + 1 + \cos(0) = 3$$

Key idea: integrating a 0-form at vertices of a mesh just "samples" the function

Integrating a 1-form over an Edge

- Suppose we have a 1-form α in the plane
- How do we integrate it over an edge e?
- **Basic recipe:**
  - Compute unit tangent T
  - Apply α to T, yielding function $\alpha(T)$
  - Integrate this scalar function over edge
- Result gives "total circulation"
- Can use *numerical quadrature* for tough integrals
  - In practice, rare to actually integrate!
  - More often, discrete 1-form values come from, *e.g.*, operations on discrete 0-form

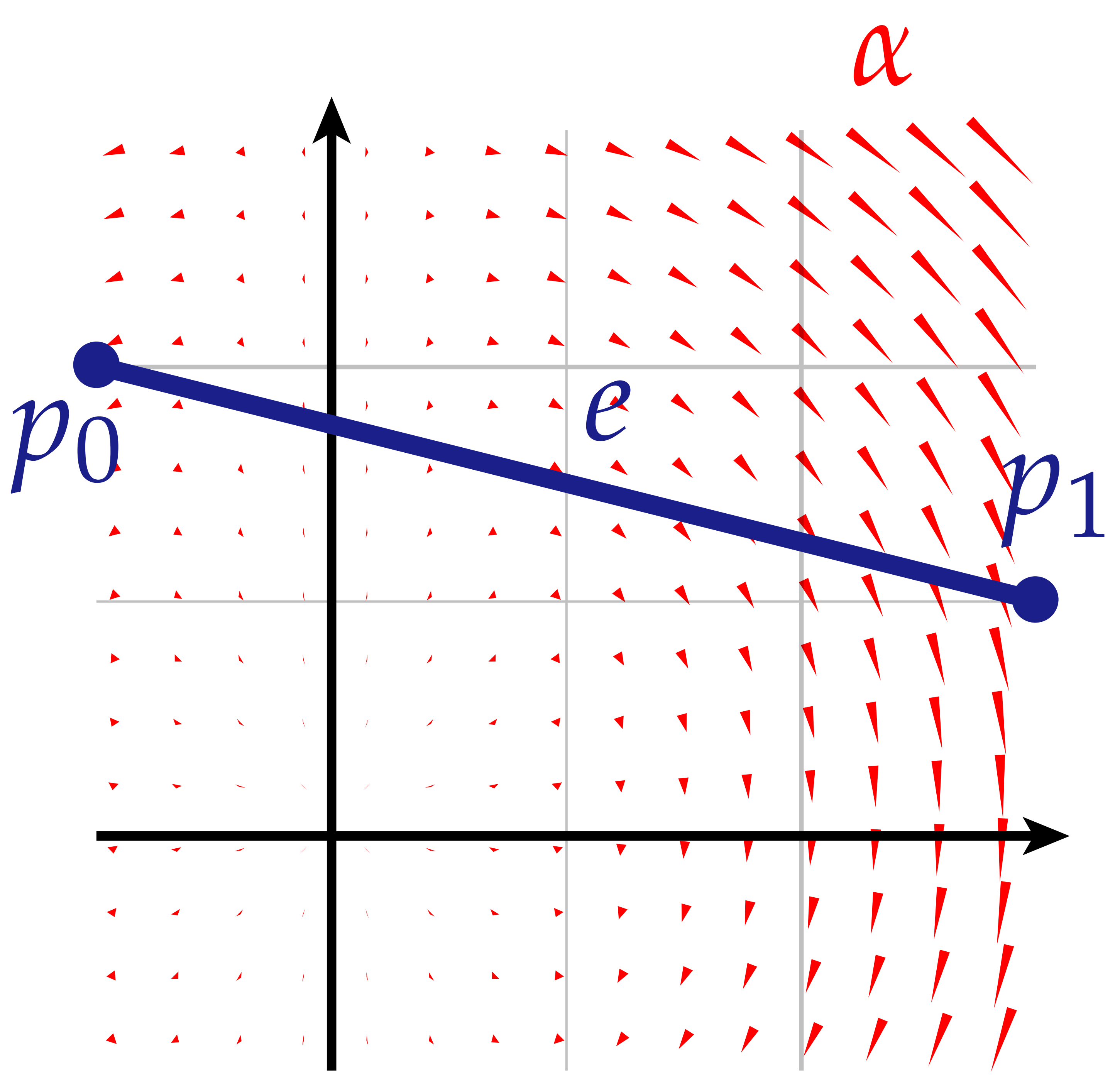


$$\hat{\alpha}_e := \int_e \alpha = \int_0^L \alpha(T)\, ds$$

$$\int_e \alpha \approx \text{length}(e) \left(\frac{1}{N} \sum_{i=1}^{N} \alpha_{p_i}(T) \right)$$

Integrating a 1-Form over an Edge—Example

In $\mathbb{R}^2$, consider a 1-form $\alpha := xydx - x^2dy$
and an edge e with endpoints $p_0 := (-1,2)$
$p_1 := (3,1)$



Q: What is $\int_e \alpha$?

A: Let's first compute the edge length L and unit tangent T:

$$L := |p_1 - p_0| = \sqrt{17} \qquad T := (p_1 - p_0)/L = (4,-1)/\sqrt{17}$$

Hence, $\alpha(T) = (4xy + x^2)/\sqrt{17}$.

An arc-length parameterization of the edge is given by

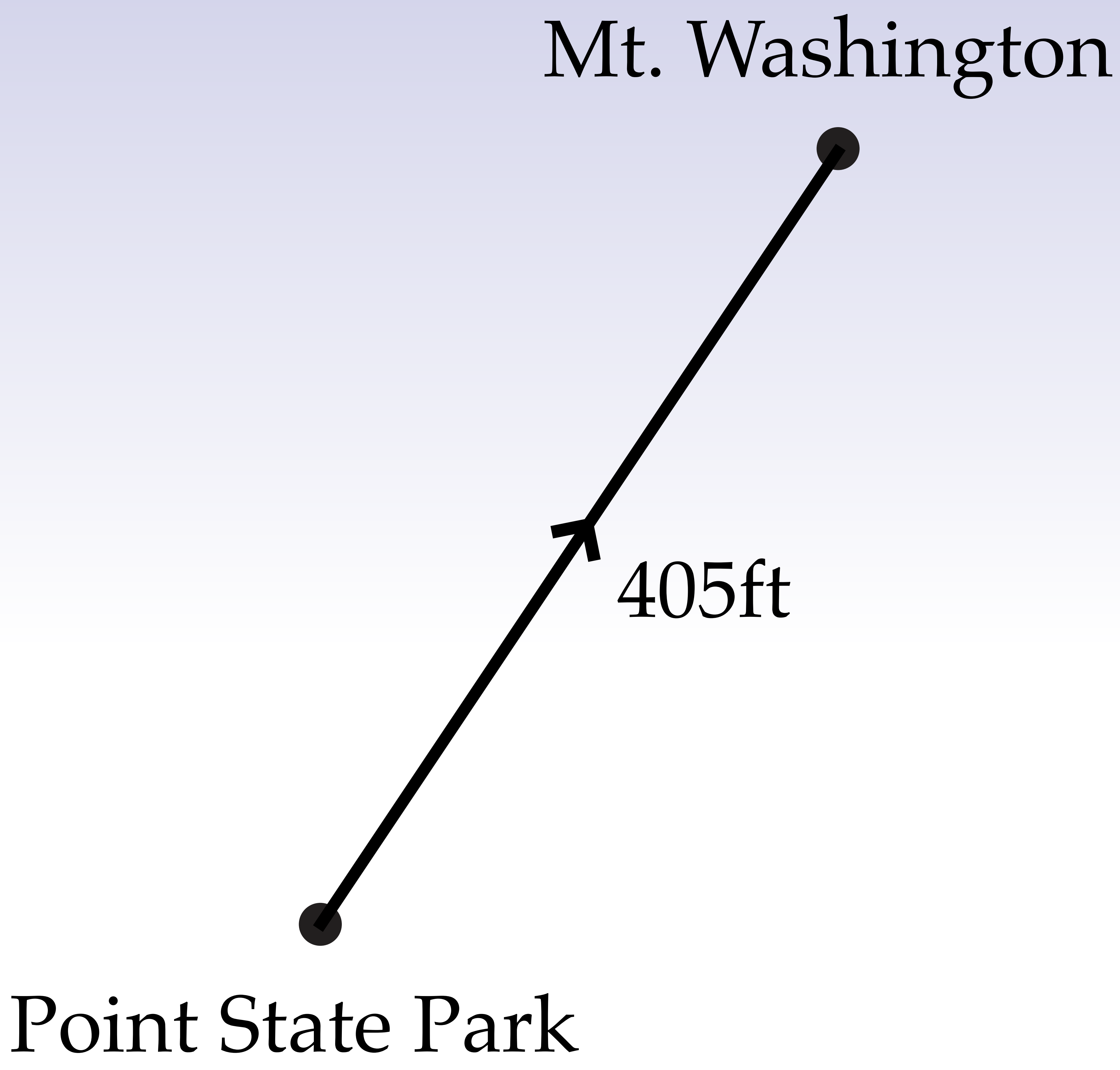
$$p(s) := p_0 + \tfrac{s}{L}(p_1 - p_0), \quad s \in [0, L]$$

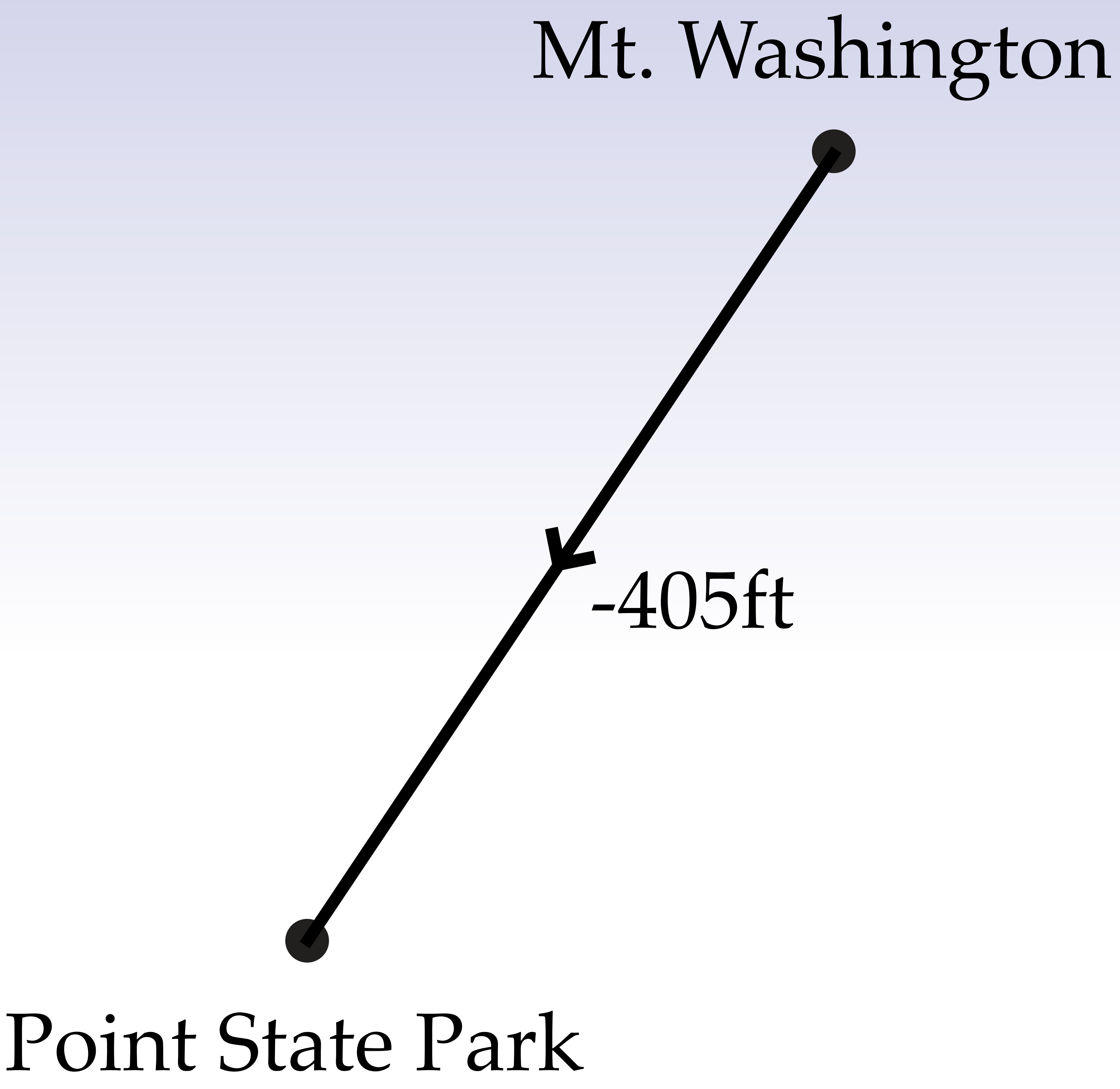
By plugging in all these expressions/values, our integral simplifies to

$$\int_0^L \alpha(T)_{p(s)}\, ds = \tfrac{7}{17} \int_0^L 4s - L\, ds = 7$$

…why not let $T := (p_0\text{-}p_1)/L$?

Orientation & Integration

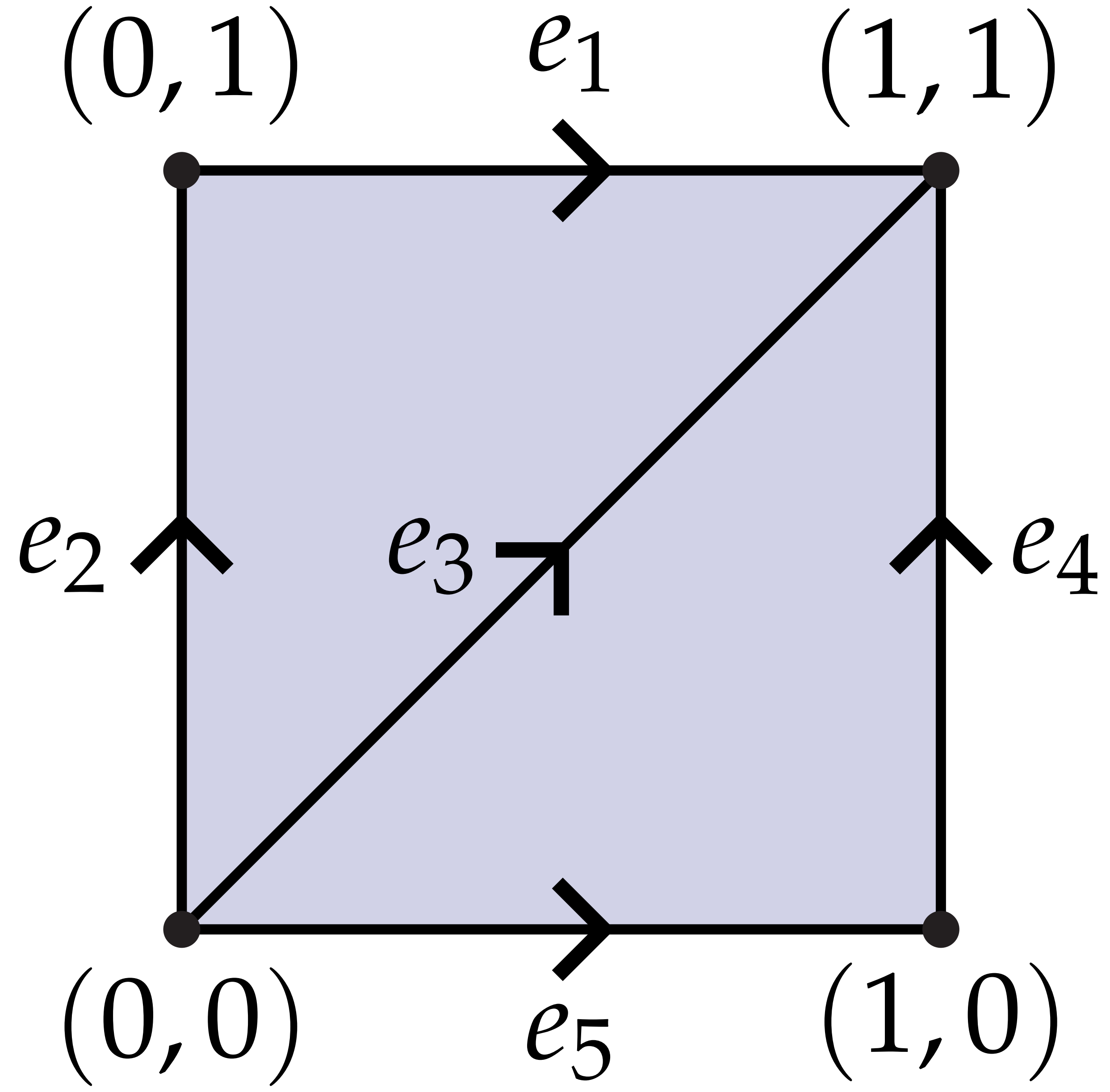




$$\int_a^b \frac{\partial \phi}{\partial x} dx = \phi(b) - \phi(a) = -(\phi(a) - \phi(b)) = -\int_b^a \frac{\partial \phi}{\partial x} dx$$

Discretizing a 1-form—Example

Example. Let M be the unit square $[0,1]^2$ with a complex K embedded as shown on the right. Using x, y to denote coordinates on M, the 1-form $\omega := 2dx$ is discretized by integrating over each edge:



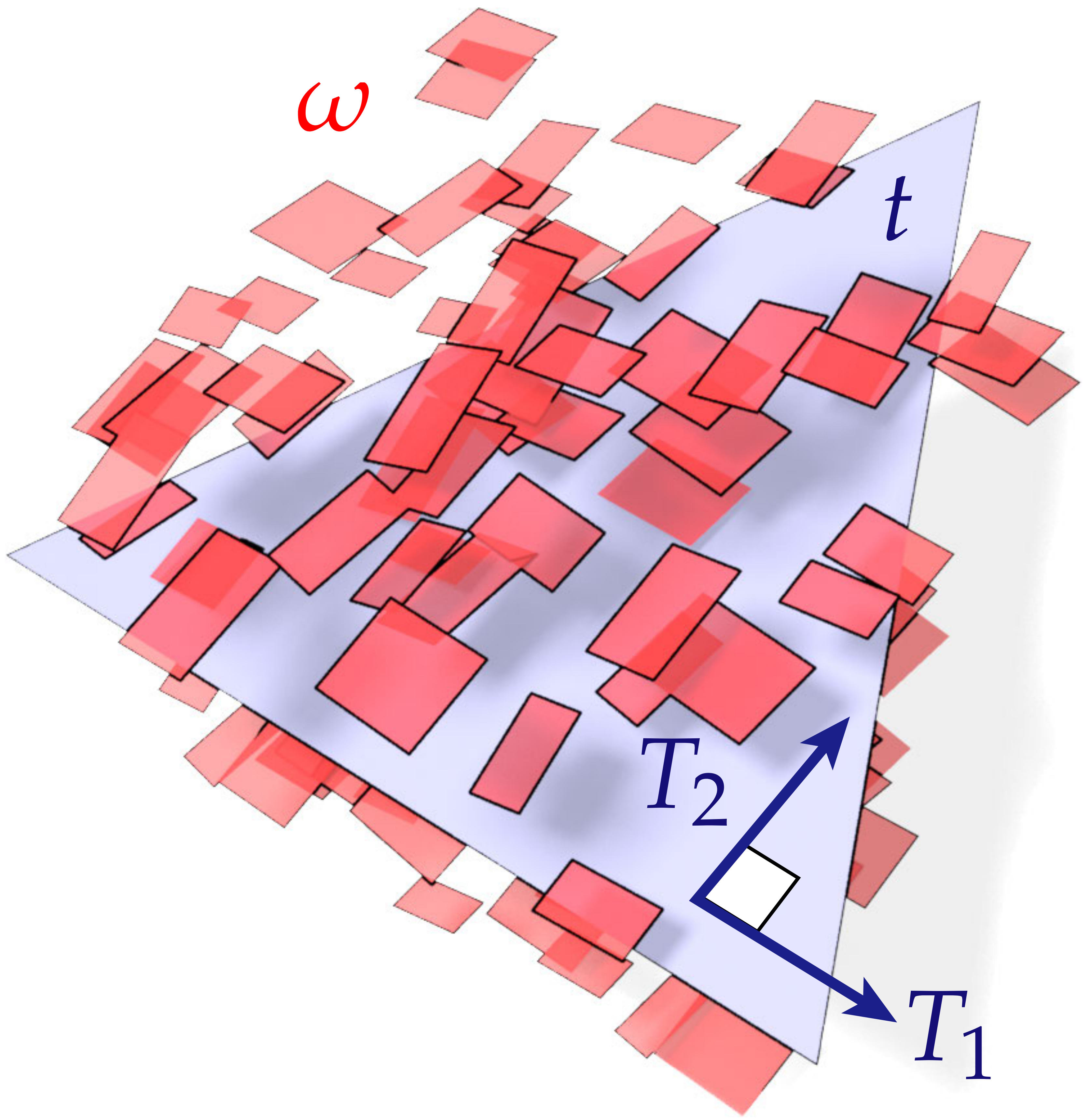
$$\begin{aligned}
\widehat{\omega}_1 &= \int_{e_1} \omega = \int_0^1 \omega\left(\tfrac{\partial}{\partial x}\right) d\ell = \int_0^1 2\, d\ell = 2. \\
\widehat{\omega}_2 &= \int_{e_2} \omega = \int_0^1 \omega\left(\tfrac{\partial}{\partial y}\right) d\ell = \int_0^1 0\, d\ell = 0. \\
\widehat{\omega}_3 &= \int_{e_3} \omega = \int_0^{\sqrt{2}} \omega\left(\tfrac{1}{\sqrt{2}}\left(\tfrac{\partial}{\partial x} + \tfrac{\partial}{\partial y}\right)\right) d\ell = \int_0^{\sqrt{2}} \tfrac{2}{\sqrt{2}}\, d\ell = 2. \\
\cdots &= \cdots
\end{aligned}$$

Question: Why does $\widehat{\omega}_1 = \widehat{\omega}_3$?

Integrating a 2-form Over a Triangle

- Suppose we have a 2-form ω in R^3
- How do we integrate it over a triangle t?
- Similar recipe to 1-form:
  - Compute orthonormal basis T_1, T_2 for triangle
  - Apply ω to T_1, T_2, yielding a function $\omega(T_1, T_2)$
  - Integrate this scalar function over triangle
- Value encodes how well triangle is "lined up" with 2-form on average, times area of triangle
- Again, rare to actually integrate explicitly!

Q: Here, what determines the *orientation* of t?

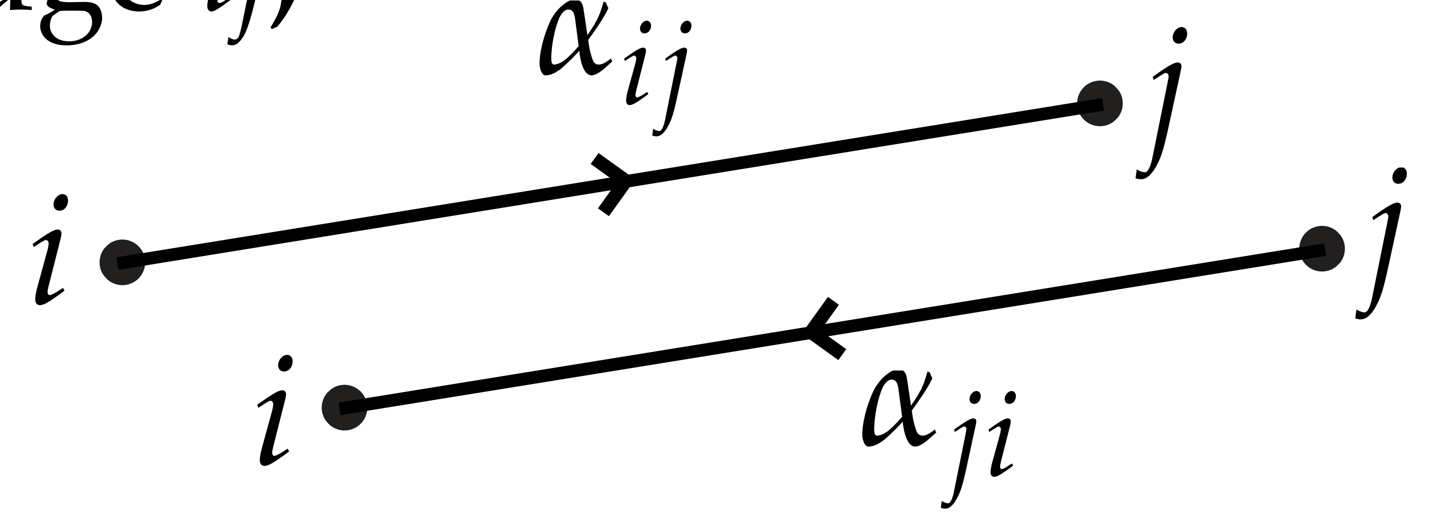


$$\int_t \omega \approx \text{area}(t) \left(\frac{1}{N} \sum_{i=1}^{N} \omega_{p_i}(T_1, T_2) \right)$$

Orientation and Integration

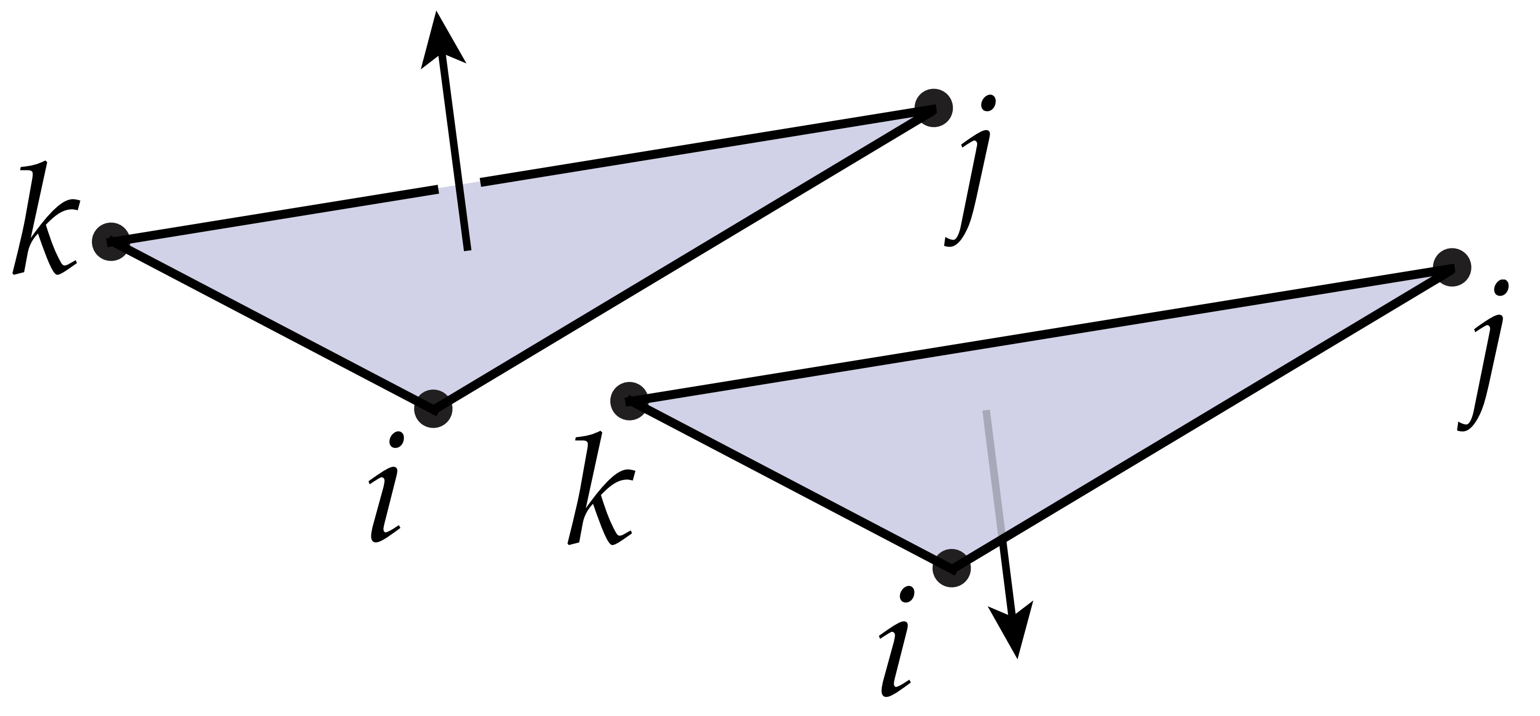
- In general, reversing the **orientation** of a simplex will reverse the **sign** of the integral.
- E.g., suppose we have a discrete 1-form α. Then for each edge *ij*,

$$\alpha_{ij} = -\alpha_{ji}$$



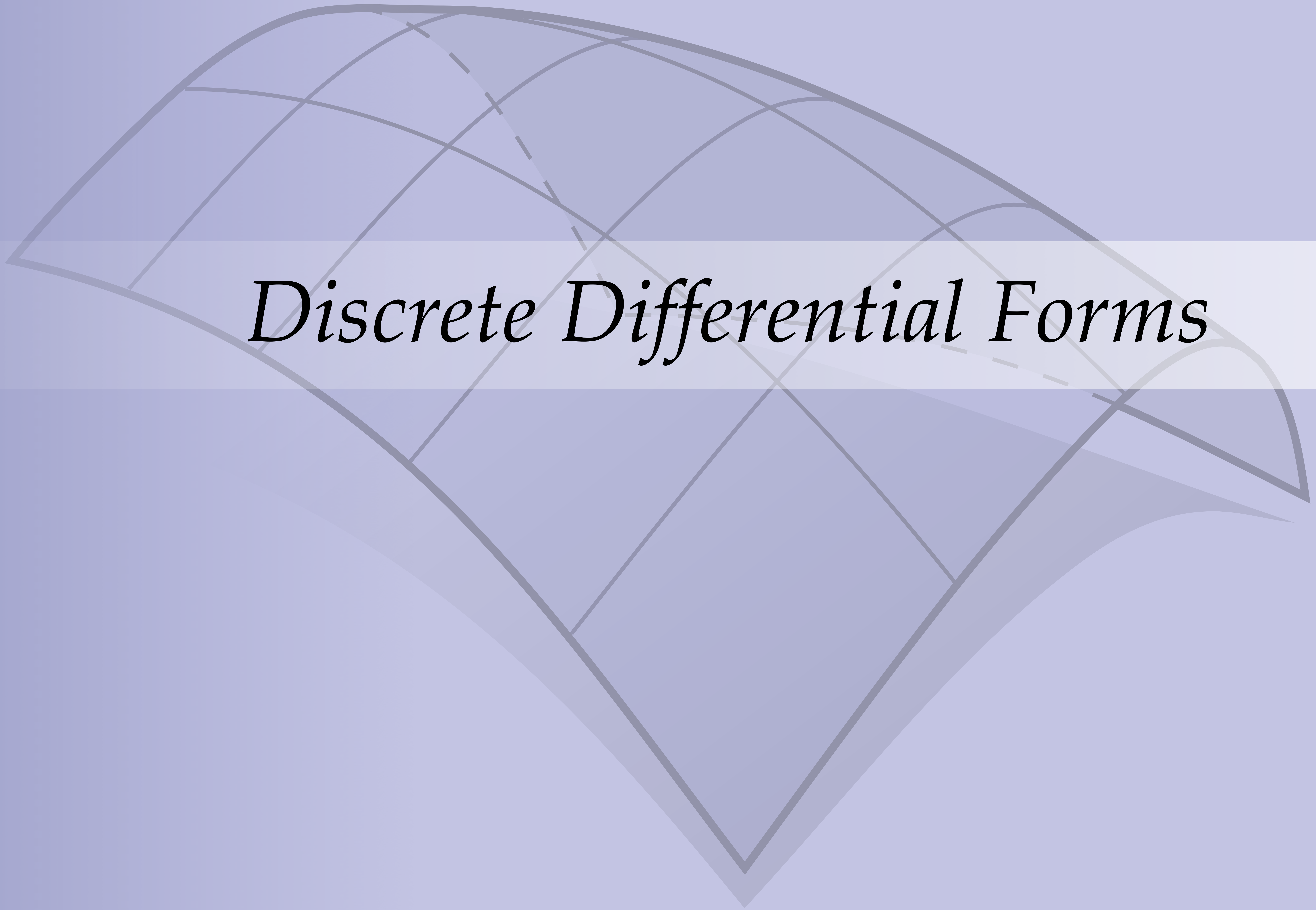
- **Q:** Suppose we have a 2-form β. What do you think the relationship is between…

$$\beta_{ijk} \;=\; \beta_{jki} \qquad\qquad \beta_{jik} = -\beta_{kij}$$



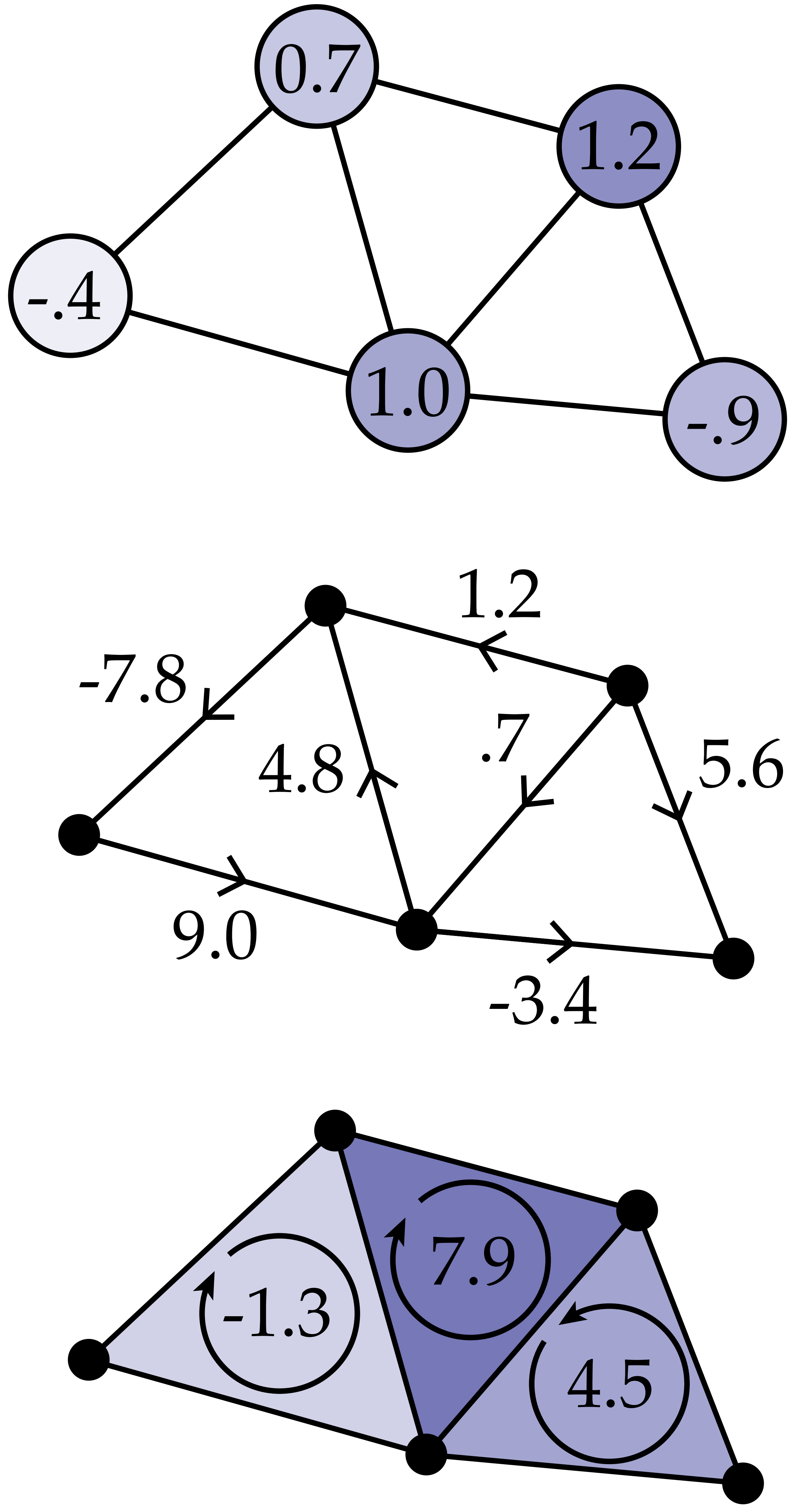
- **Q:** What's the rule in general?
- **A:** Discrete k-form values change sign under odd permutation. (Sound familiar? :-))

Discrete Differential Forms



Discrete Differential k-Form

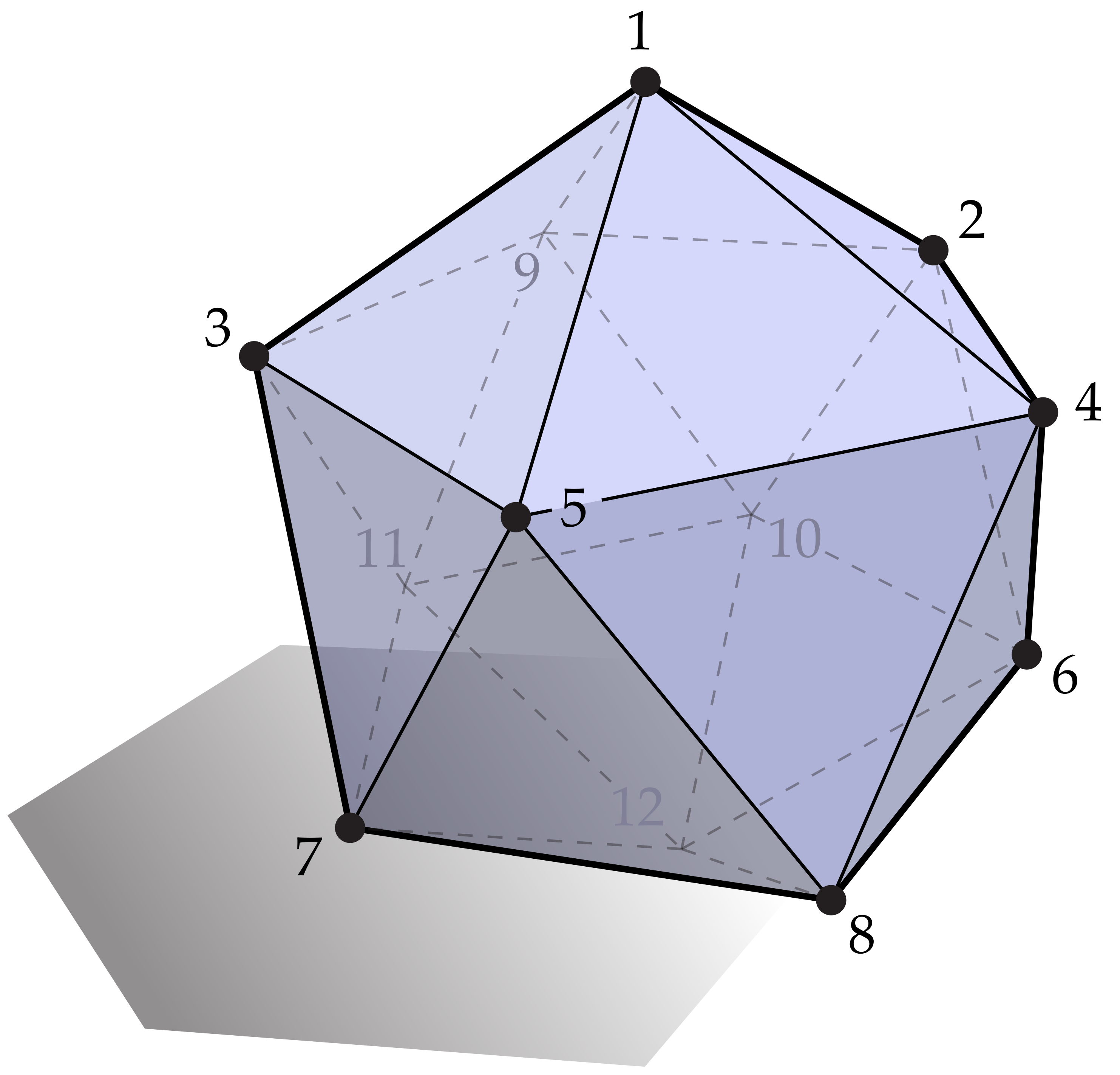
- Abstractly, a *discrete differential k-form* is just any assignment of a value to each oriented *k*-simplex.
- For instance, in 2D:
  - values at **vertices** encode a discrete **0-form**
  - values at **edges** encode a discrete **1-form**
  - values at **faces** encode a discrete **2-form**
- *Conceptually*, values represent integrated k-forms
- *In practice*, almost never comes from direct integration!
- More typically, values start at vertices (samples of some function); 1-forms, 2-forms, *etc.*, arise from applying operators like the (discrete) exterior derivative



Matrix Encoding of Discrete Differential k-Forms

- We can encode a discrete k-form as a column vector with one entry for every k-simplex.
- To do so, we need to first assign a unique *index* to each k-simplex
  - The order of these indices can be completely arbitrary
  - We just need some way to put elements of our mesh into correspondence with entries of the vector
- Simplest example: a discrete 0-form can be encoded as a vector with $|V|$ entries

$$\phi : V \to \mathbb{R}$$

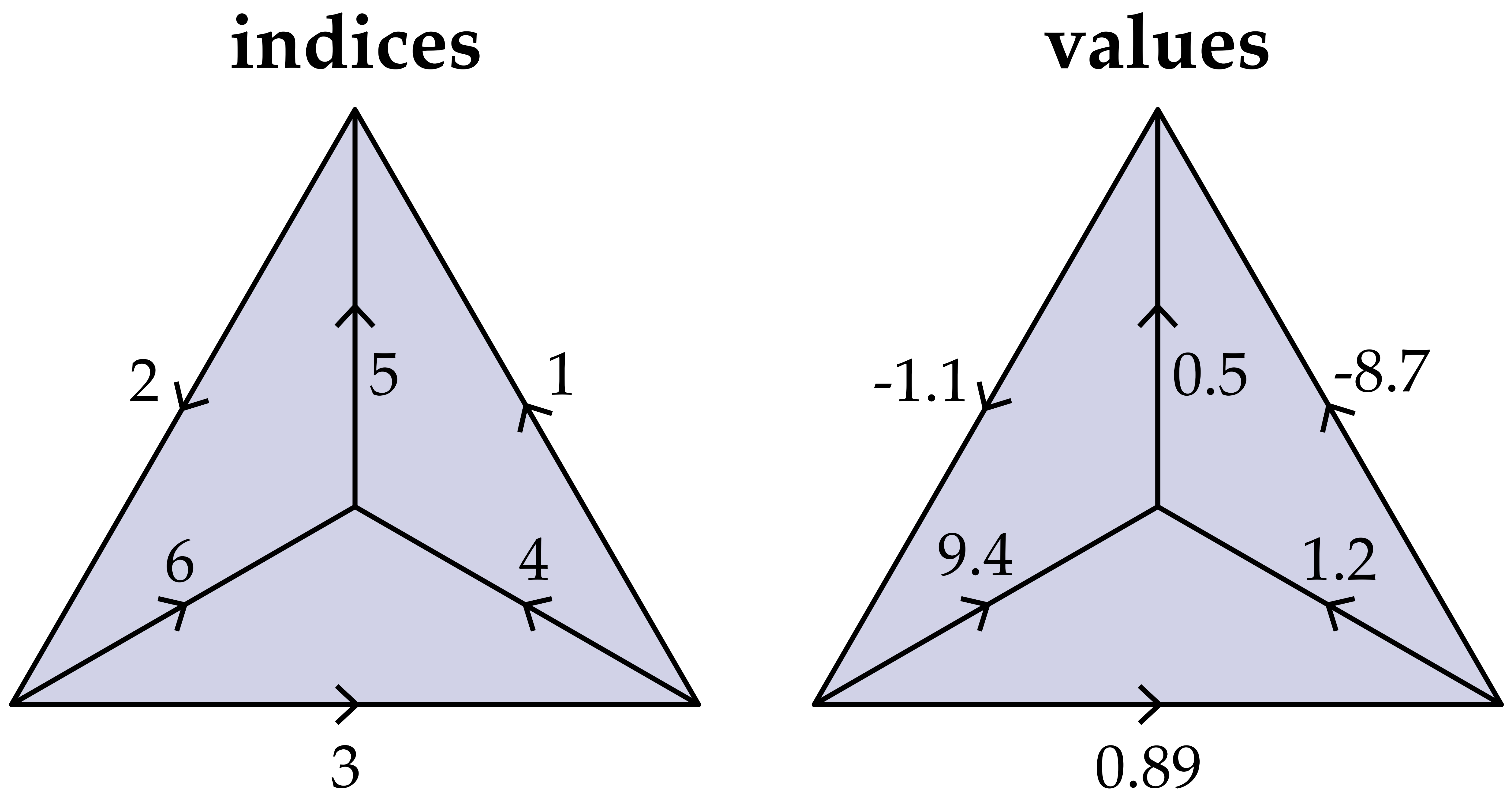


$$\phi = \left[\begin{array}{ccc} \phi_1 & \cdots & \phi_{|V|} \end{array} \right]$$

Careful: In code, indices often start from 0 rather than 1!

Matrix Encoding of Discrete Differential 1-Form

- A discrete differential 1-form is a value per edge of an oriented simplicial complex.
- To encode these values as a column vector, we must first assign a unique index to each edge of our complex.
- If we then have values on edges, we know how to assign them to entries of the vector encoding the discrete 1-form.

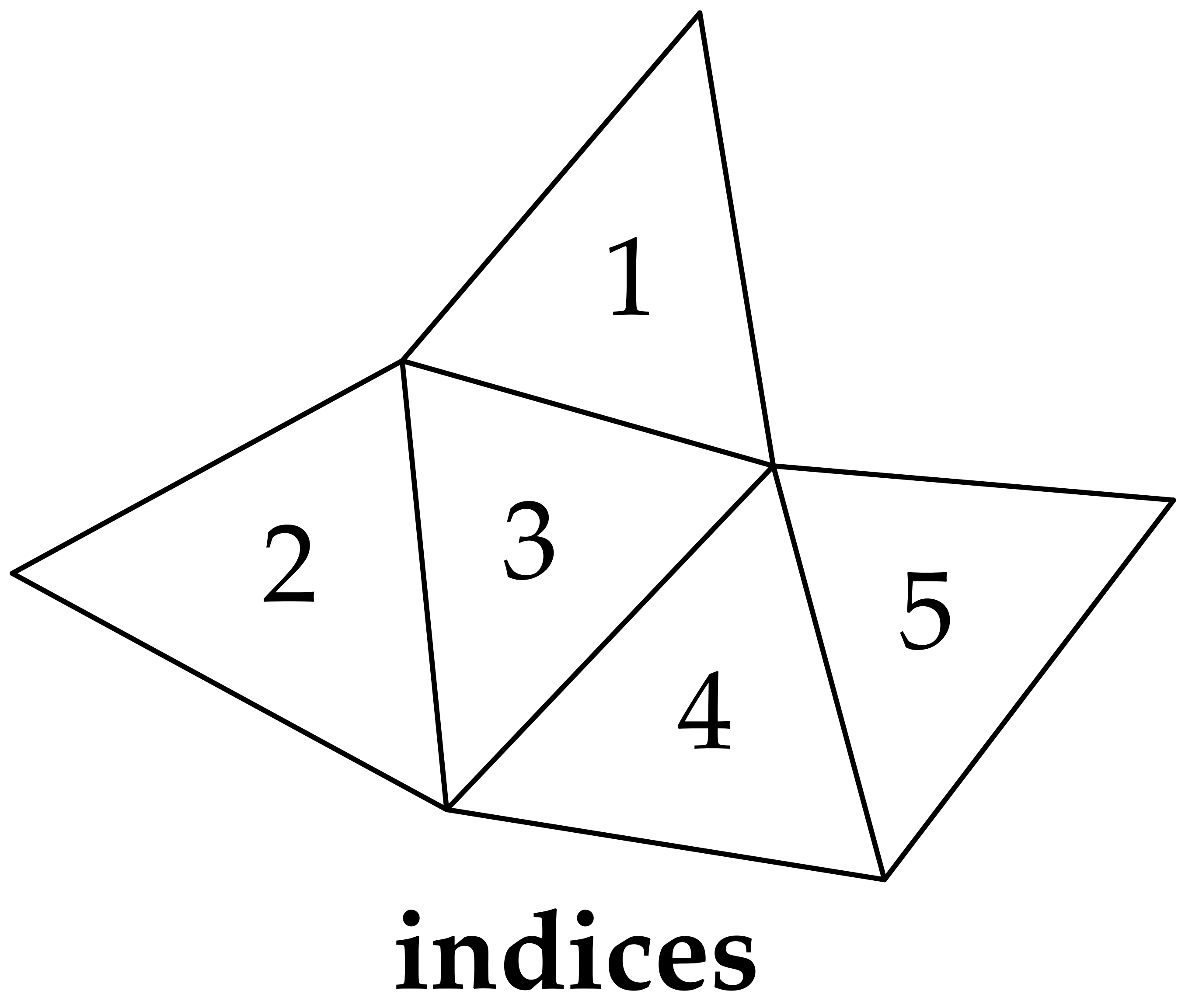


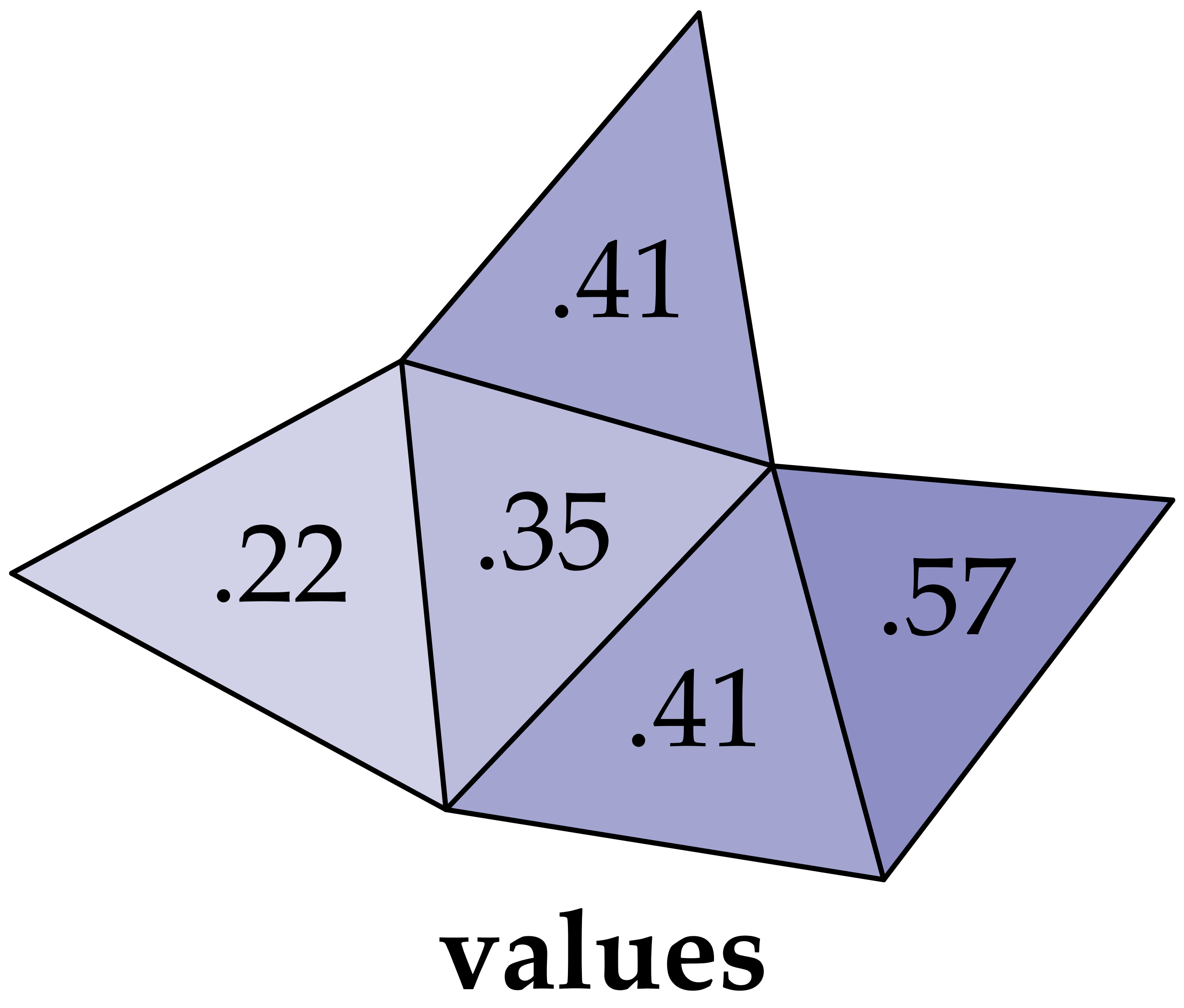
$$\alpha = \begin{bmatrix} -8.7 & -1.1 & 0.89 & 1.2 & 0.5 & 9.4 \end{bmatrix}^\mathsf{T}$$

Careful that if we ever change the orientation of an edge, we must also negate the value in our row vector!

Matrix Encoding of Discrete Differential 2-Form

- Same idea for encoding a discrete differential 2-form as a column vector
- Assign indices to each 2-simplex; now we know which values go in which entries



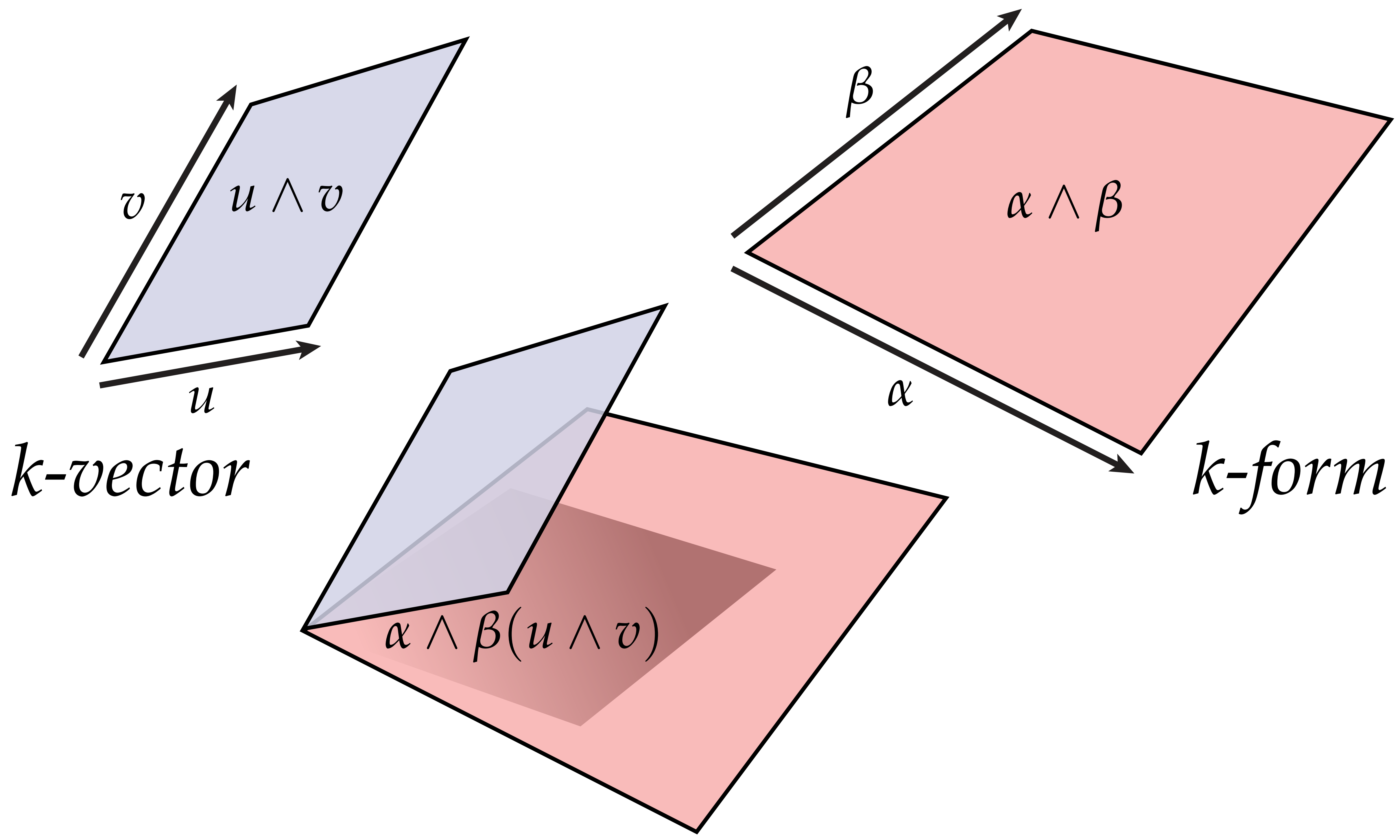


$$\omega = \left[\ .41 \quad .22 \quad .35 \quad .41 \quad .57\ \right]$$

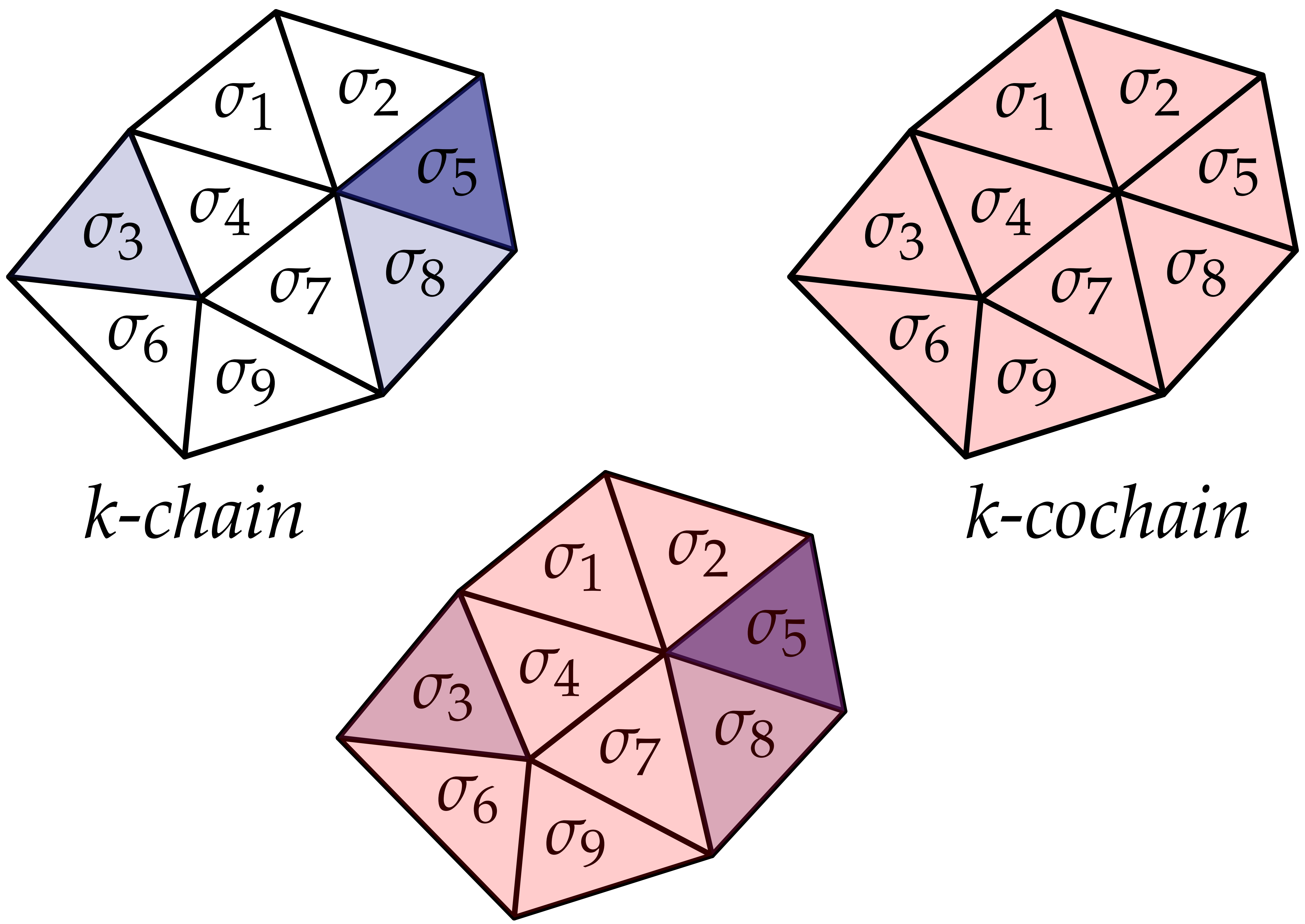
Chains & Cochains

In the discrete setting, duality between *"things that get measured"* (k-vectors) and *"things that measure"* (k-forms) is captured by notion of *chains* and *cochains*.

SMOOTH



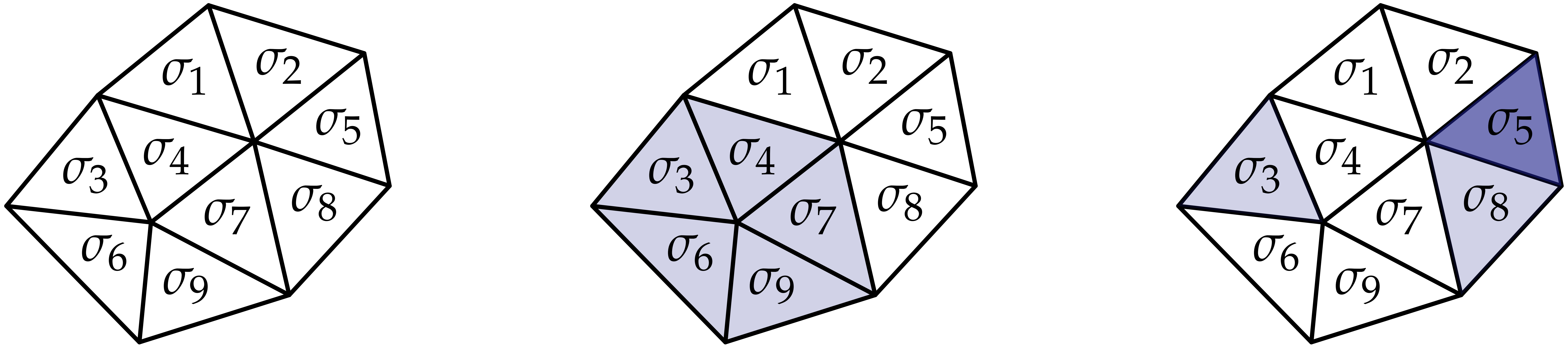
DISCRETE



Simplicial Chain

- Suppose we think of each k-simplex as its own basis vector
- Can specify some region of a mesh via a linear combination of simplices.

Example.



0 $\qquad$ $\sigma_3+\sigma_4+\sigma_6+\sigma_7+\sigma_9$ $\qquad$ $\sigma_3 + 3\sigma_5 + \sigma_8$

Q: What does it means when we have a coefficient other than 0 or 1? (Or *negative*?)

A: Roughly speaking, "n copies" of that simplex. (Or opposite *orientation*.)

(Formally: *chain group* C_k is the free abelian group generated by the k-simplices.)

Arithmetic on Simplicial Chains

$$c_1 = e_3 - e_{12} + e_{18} - e_{15} + e_6 - e_1$$

$$c_2 = e_{15} + e_{19} - e_{17} - e_8 - e_2 - e_6$$

$$c_1 + c_2 = e_3 - e_{12} + e_{18} - \cancel{e_{15}} + \cancel{e_6} - e_1 + \cancel{e_{15}} + e_{19} - e_{17} - e_8 - e_2 - \cancel{e_6}$$

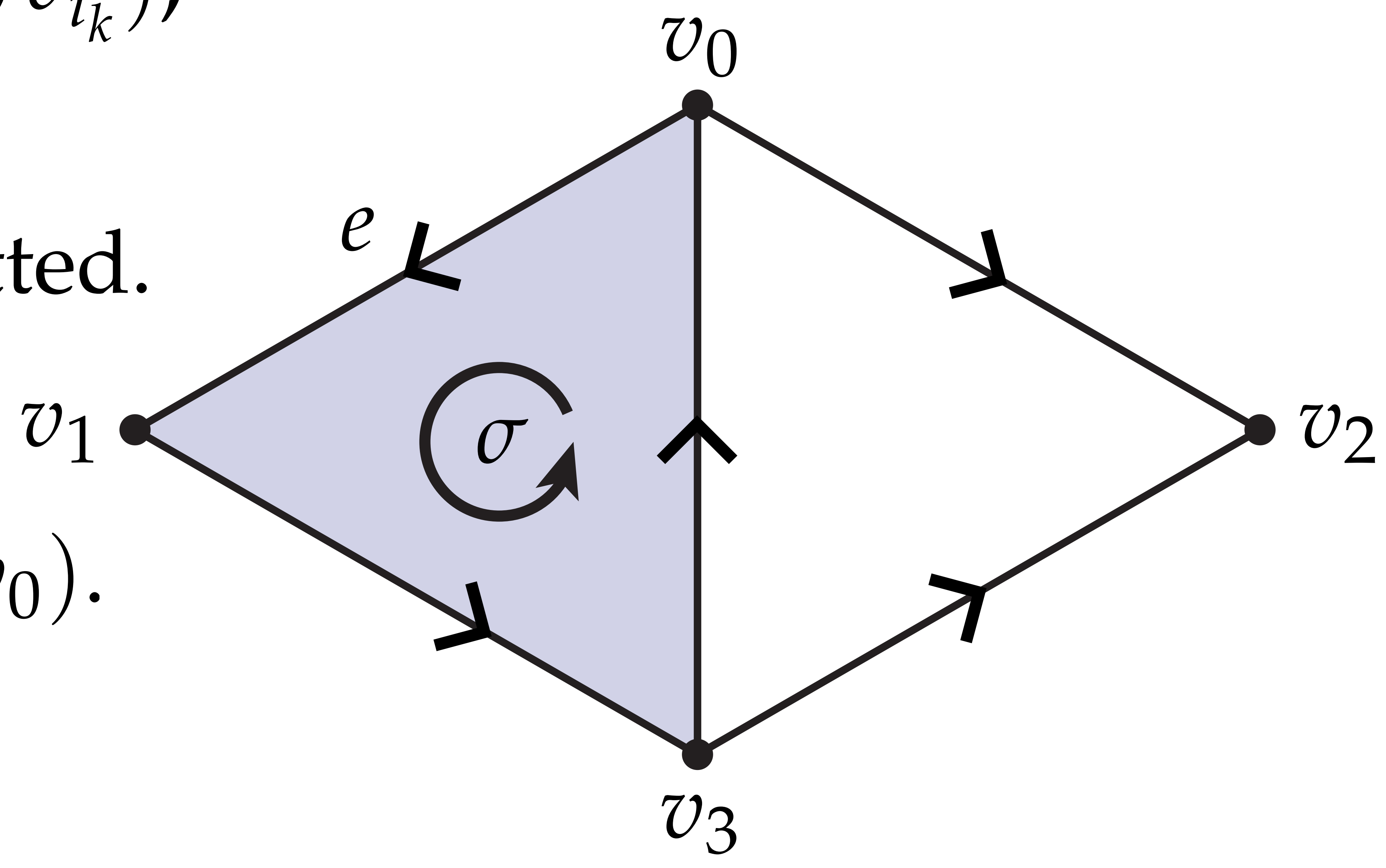
$$= e_3 - e_{12} + e_{18} - e_1 + e_{19} - e_{17} - e_8 - e_2 =: c_3$$

Boundary Operator on Simplices

Definition. Let $\sigma := (v_{i_0}, \ldots, v_{i_k})$ be an oriented k-simplex. Its *boundary* is the oriented $k-1$-chain

$$\partial\sigma := \sum_{p=0}^{k} (-1)^p (v_{i_0}, \ldots, \cancel{v_{i_p}}, \ldots, v_{i_k}),$$

where $\cancel{v_{i_p}}$ indicates that the pth vertex has been omitted.



Example. Consider the 2-simplex $\sigma := (v_0, v_1, v_3)$.
Its boundary is the 1-chain $(v_0, v_1) + (v_1, v_3) + (v_3, v_0)$.

Example. Consider the 1-simplex $e := (v_0, v_1)$.
Its boundary is the 0-chain $\partial e = v_1 - v_0$.

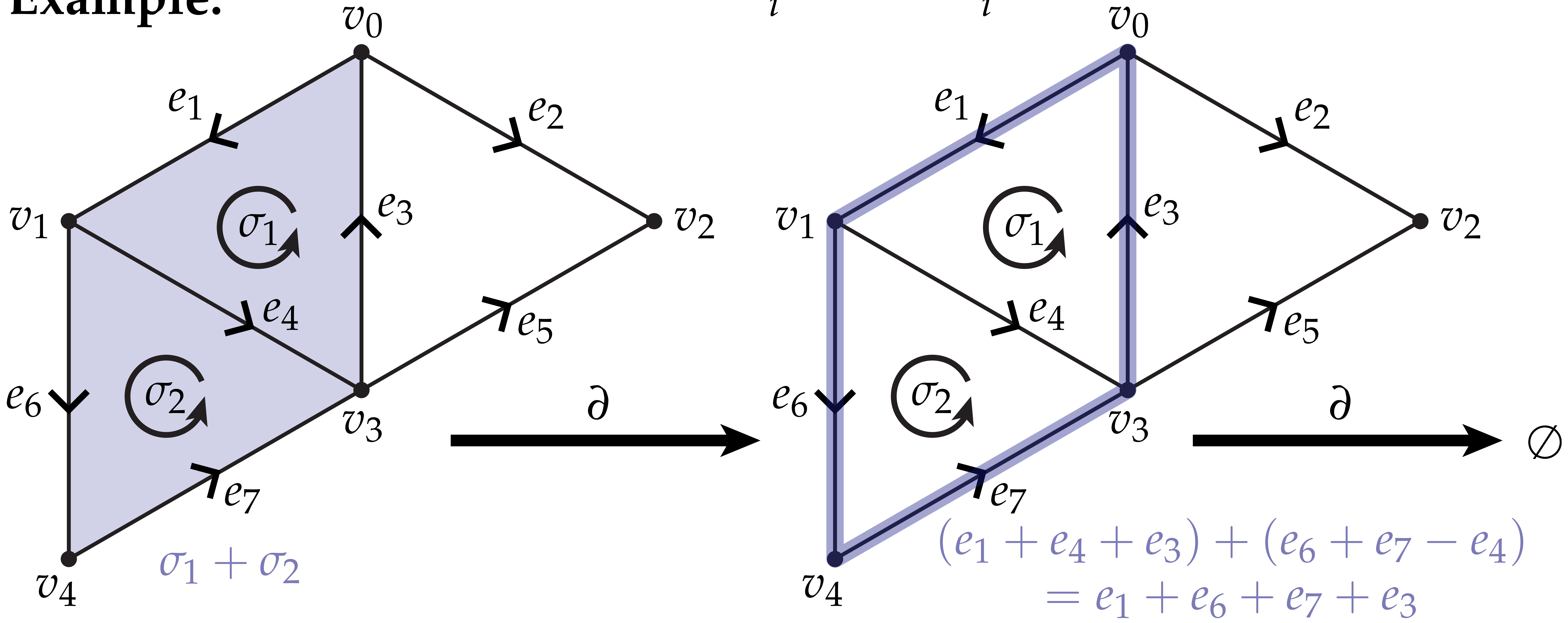
Example. Consider the 0-simplex (v_1).
Its boundary is the empty set.

Boundary Operator on Simplicial Chains

The boundary operator can be extended to any chain by linearity, *i.e.*,

$$\partial \sum_i c_i \sigma_i = \sum_i c_i \partial_i \sigma_i.$$

Example.

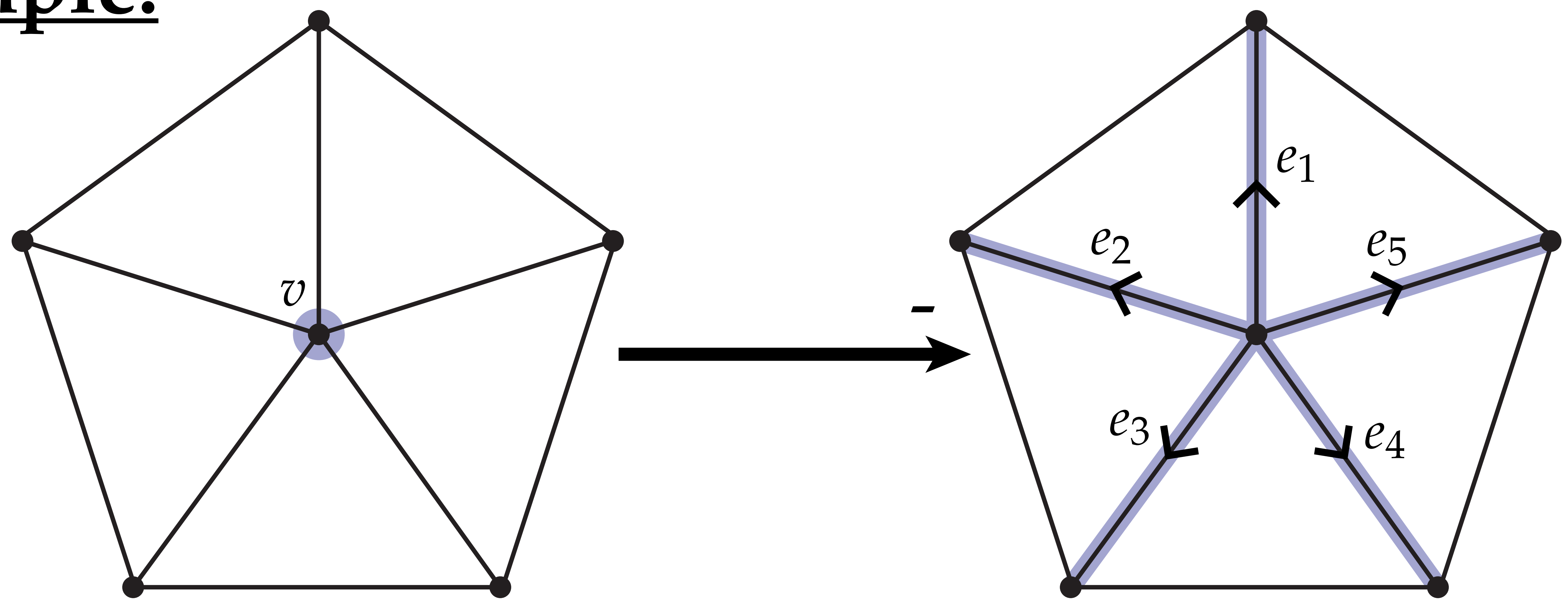


Note: boundary of boundary is *always* empty!

Coboundary Operator on Simplices

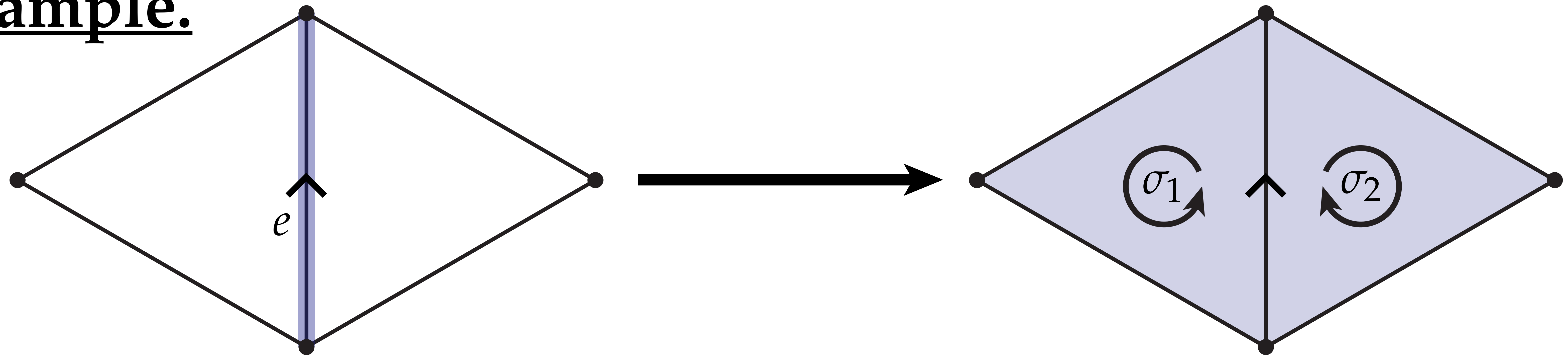
The *coboundary* of an oriented k-simplex σ is the collection of all oriented (k+1)-simplices that contain σ, and which have the same relative orientation.

Example.



(Orientation?)

Example.



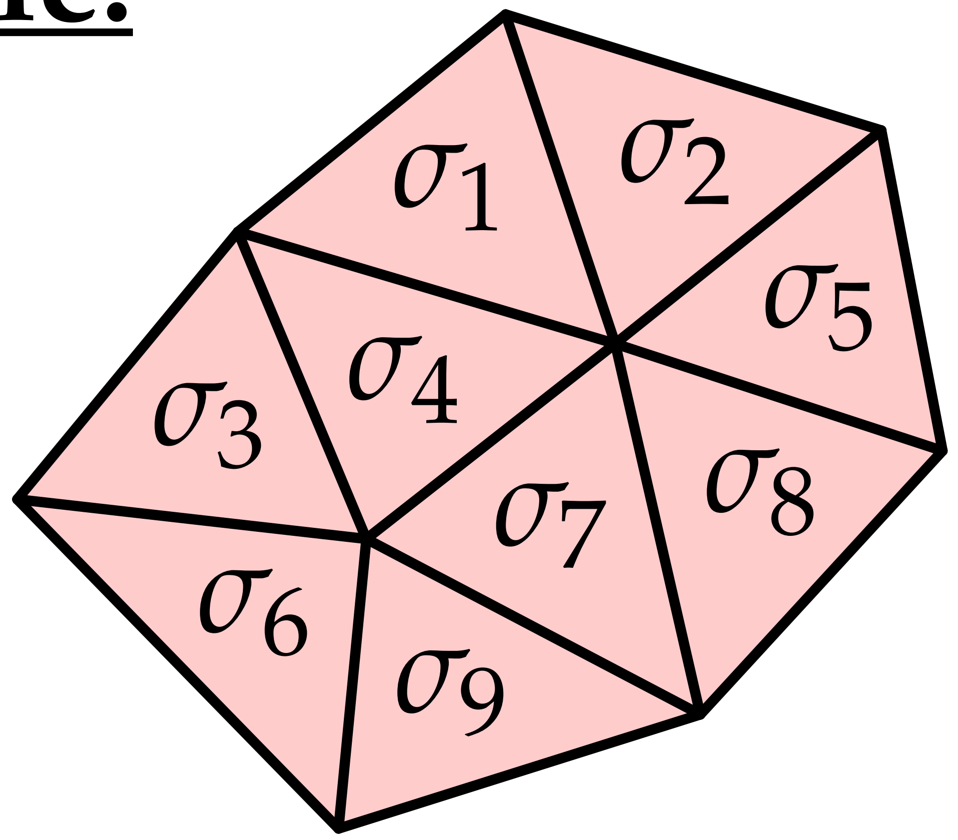
(**Analogy:** simplicial star)

Simplicial Cochain

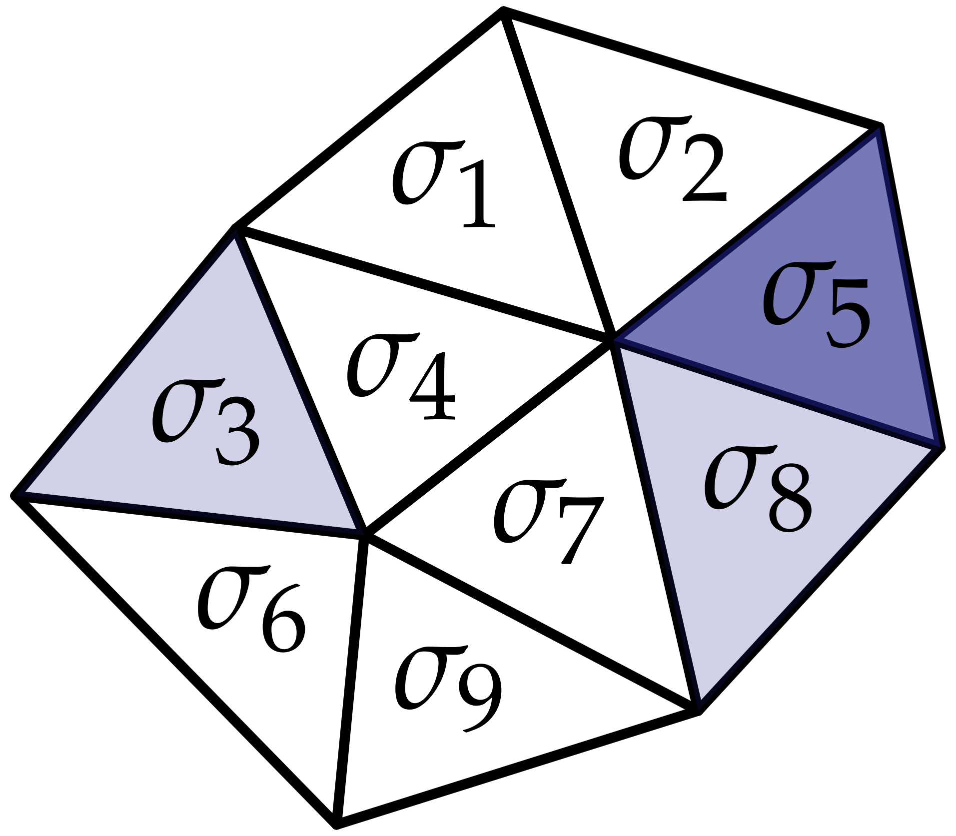
A *simplicial k-cochain* is basically any **linear** map from a simplicial *k*-chain to a number.

$$\alpha(c_1\sigma_1 + \cdots + c_n\sigma_n) = \sum_{i=1}^{n} \alpha_i c_i$$

Example.



$\forall i, \alpha(\sigma_i) = 1$



$\sigma_3 + 3\sigma_5 + \sigma_8$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 3 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = 1 + 3 + 1 = 5$$

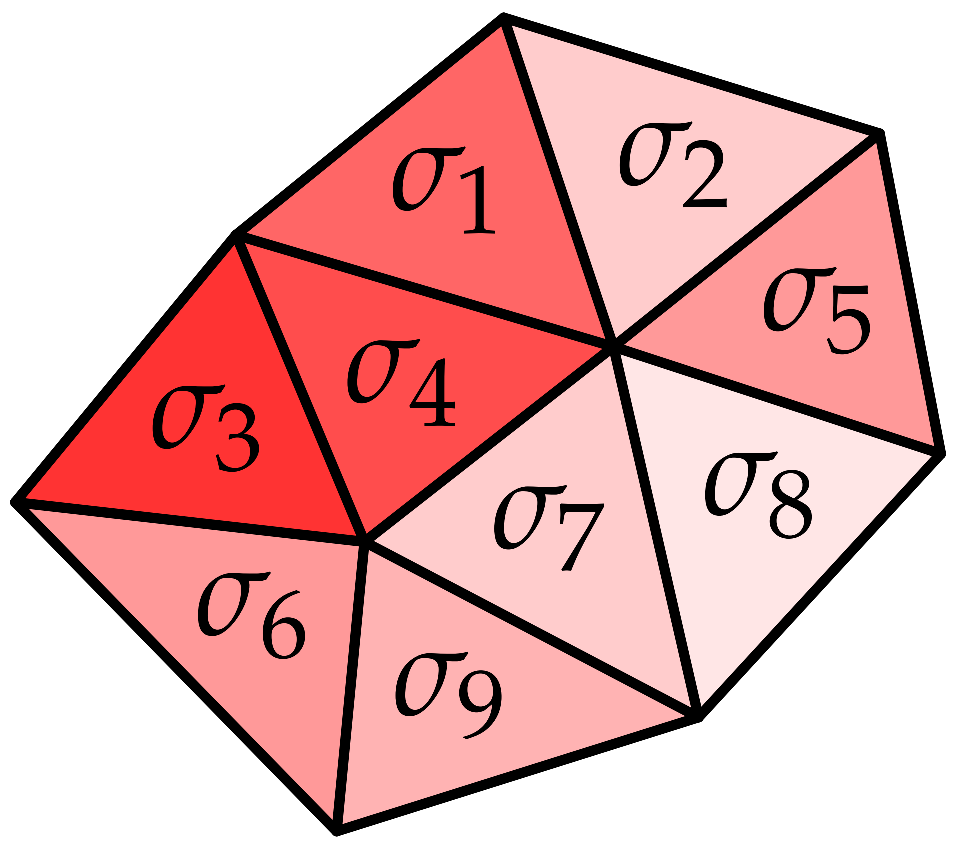
(Formally: *cochain group* is group of homomorphisms from cochains to reals.)

Simplicial Cochains & Discrete Differential Forms

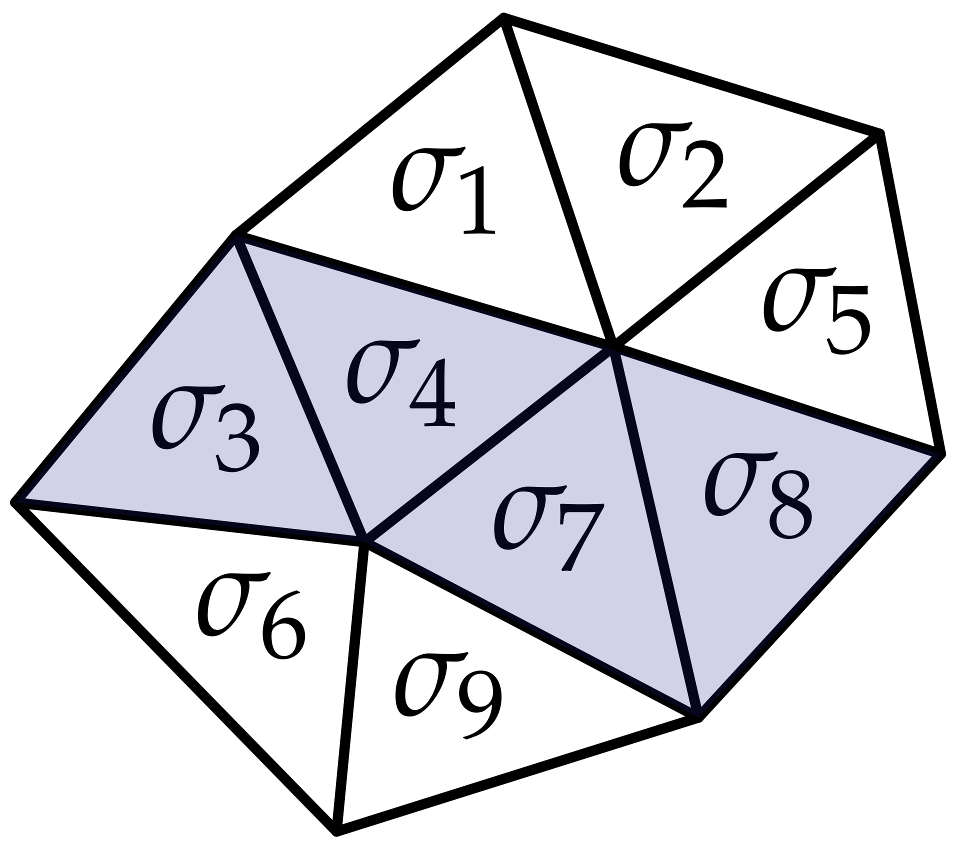
Suppose a simplicial *k*-cochain is given by the integrated values from a discrete *k*-form

Q: What does it mean (geometrically) when we apply it to a simplicial *k*-chain?

A: Our discrete *k*-form values come from integrating a smooth *k*-form over each *k*-simplex. So, we just get the integral over the region specified by the chain:



$$\hat{\alpha}_i := \int_\sigma \alpha$$



$$c = \sigma_3 + \sigma_4 + \sigma_7 + \sigma_8$$

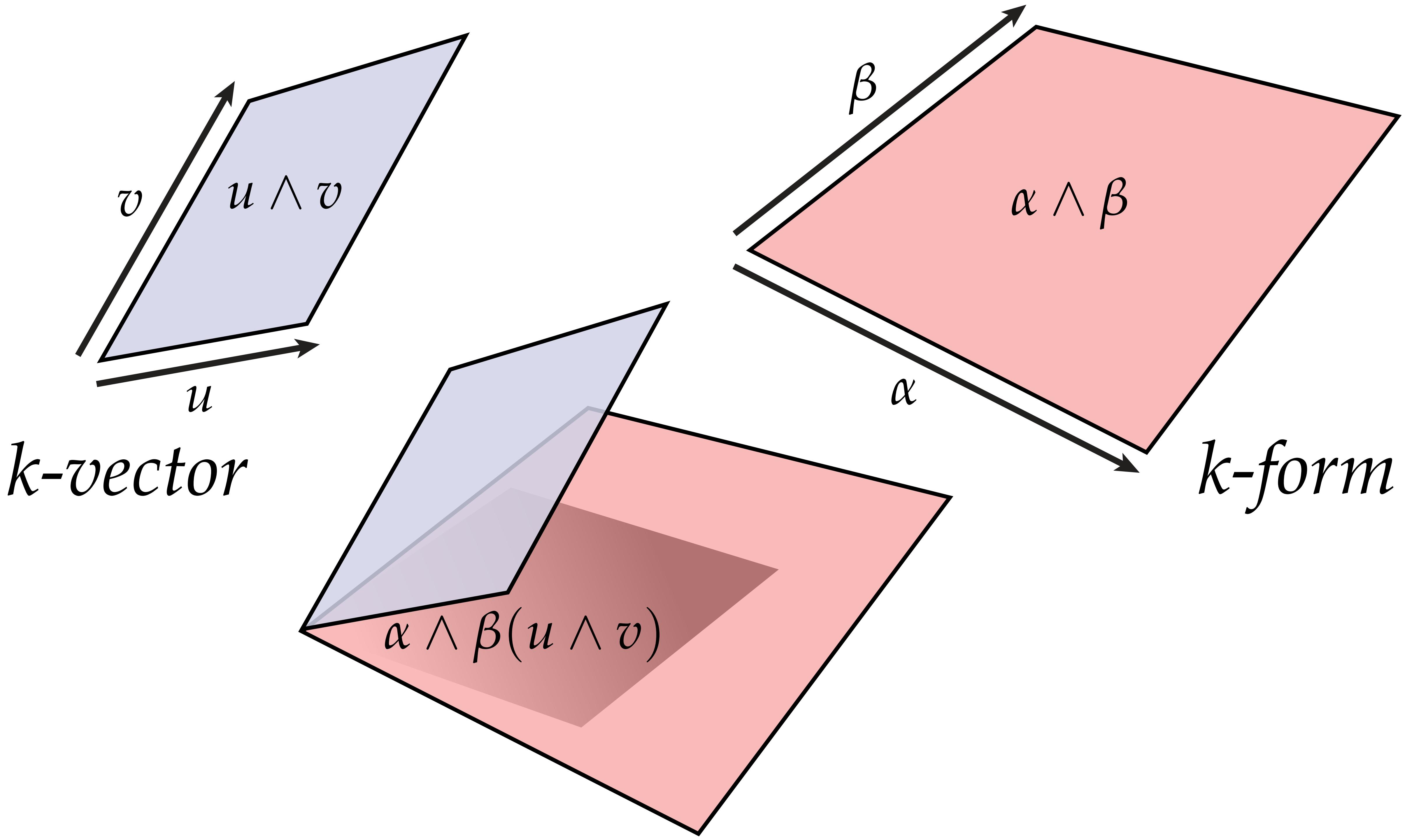
$$\hat{\alpha}(c) = \hat{\alpha}_3 + \hat{\alpha}_4 + \hat{\alpha}_7 + \hat{\alpha}_8$$

$$= \int_{\sigma_3 \cup \sigma_4 \cup \sigma_7 \cup \sigma_8} \alpha$$

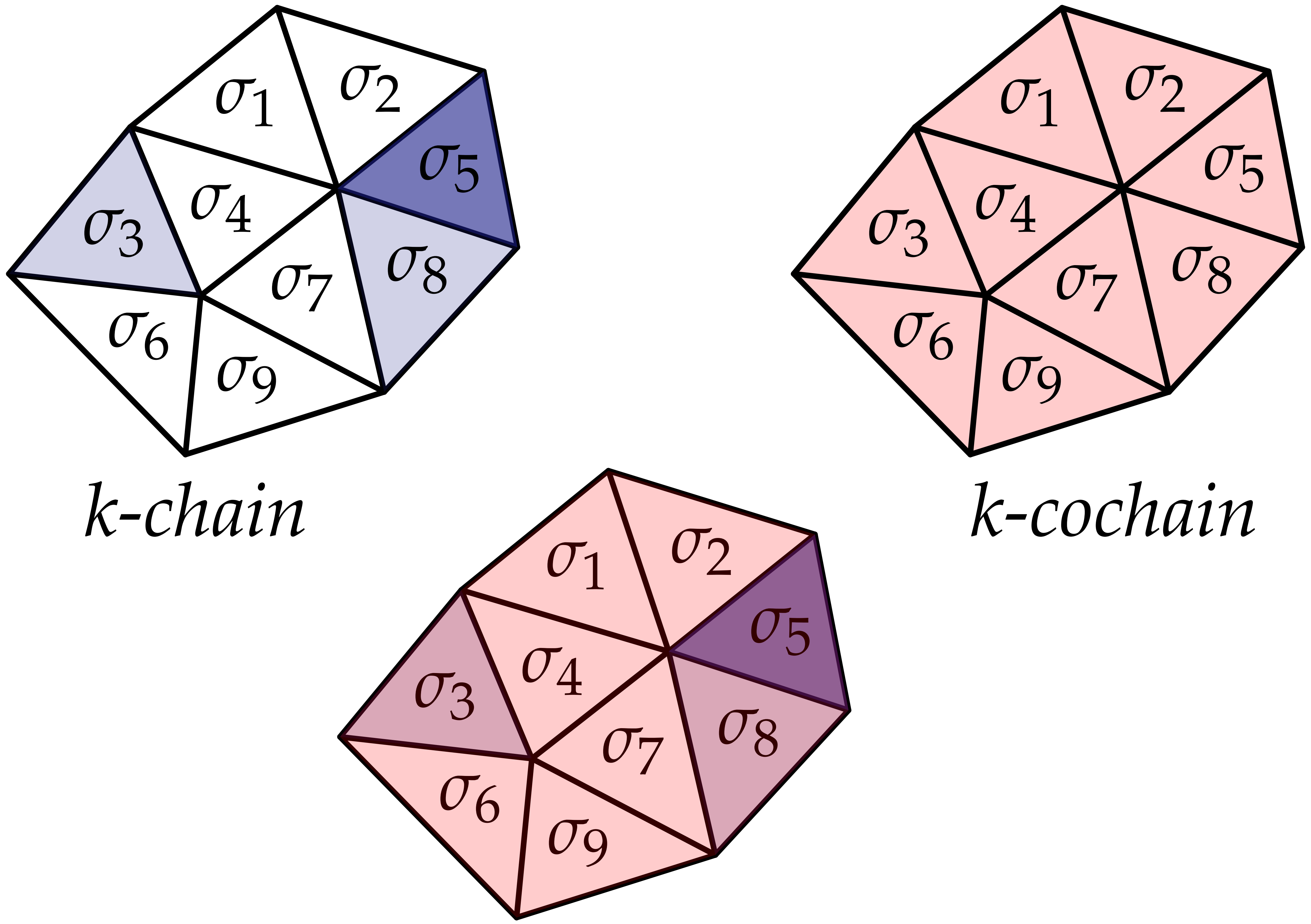
Discrete Differential Form

Definition. Let M be a manifold simplicial complex. A (primal) *discrete differential k-form* is a simplicial k-cochain on M. We will use Ω_k to denote the set of k-forms.

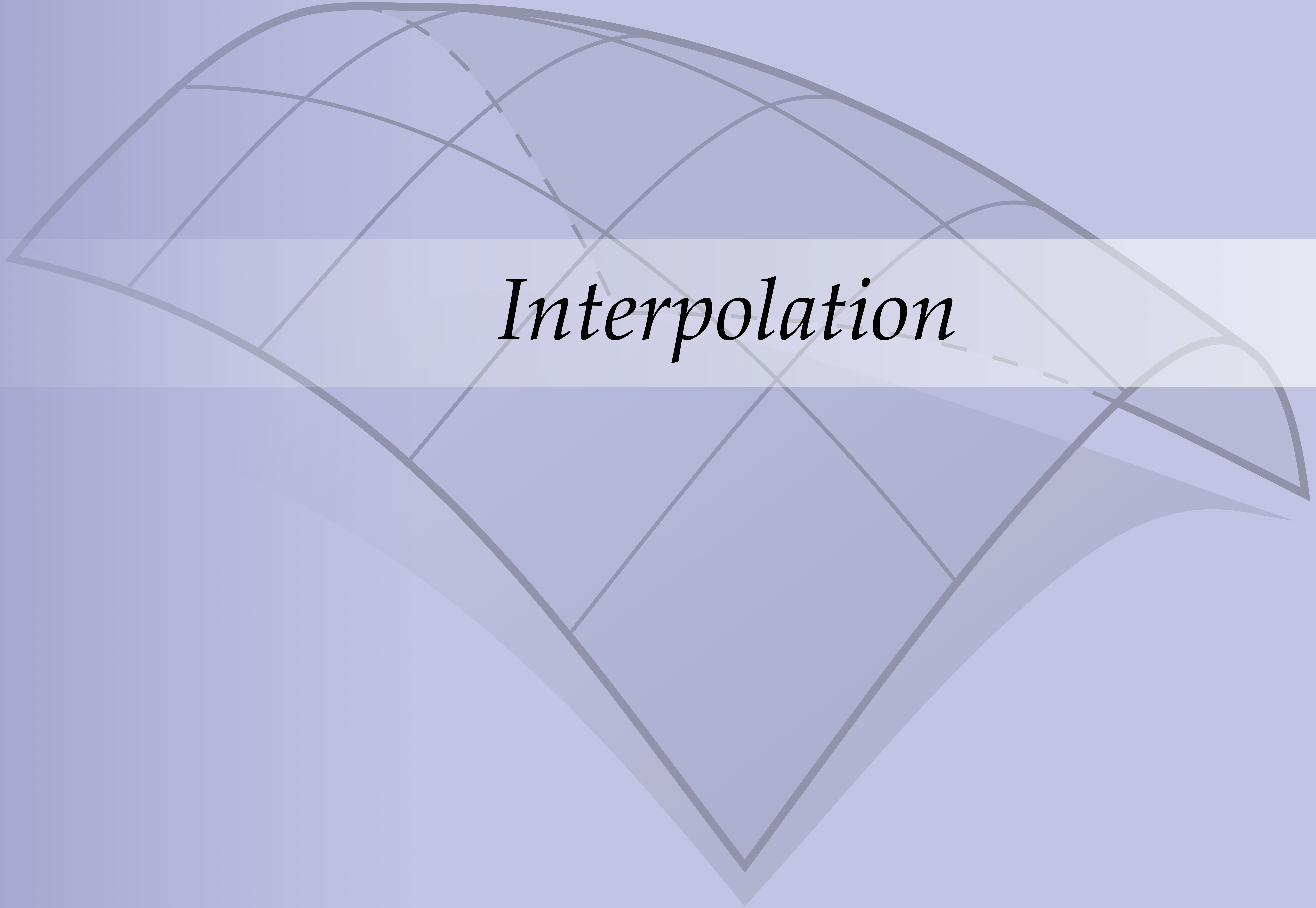
SMOOTH



DISCRETE



Interpolation



Interpolation—0-Forms

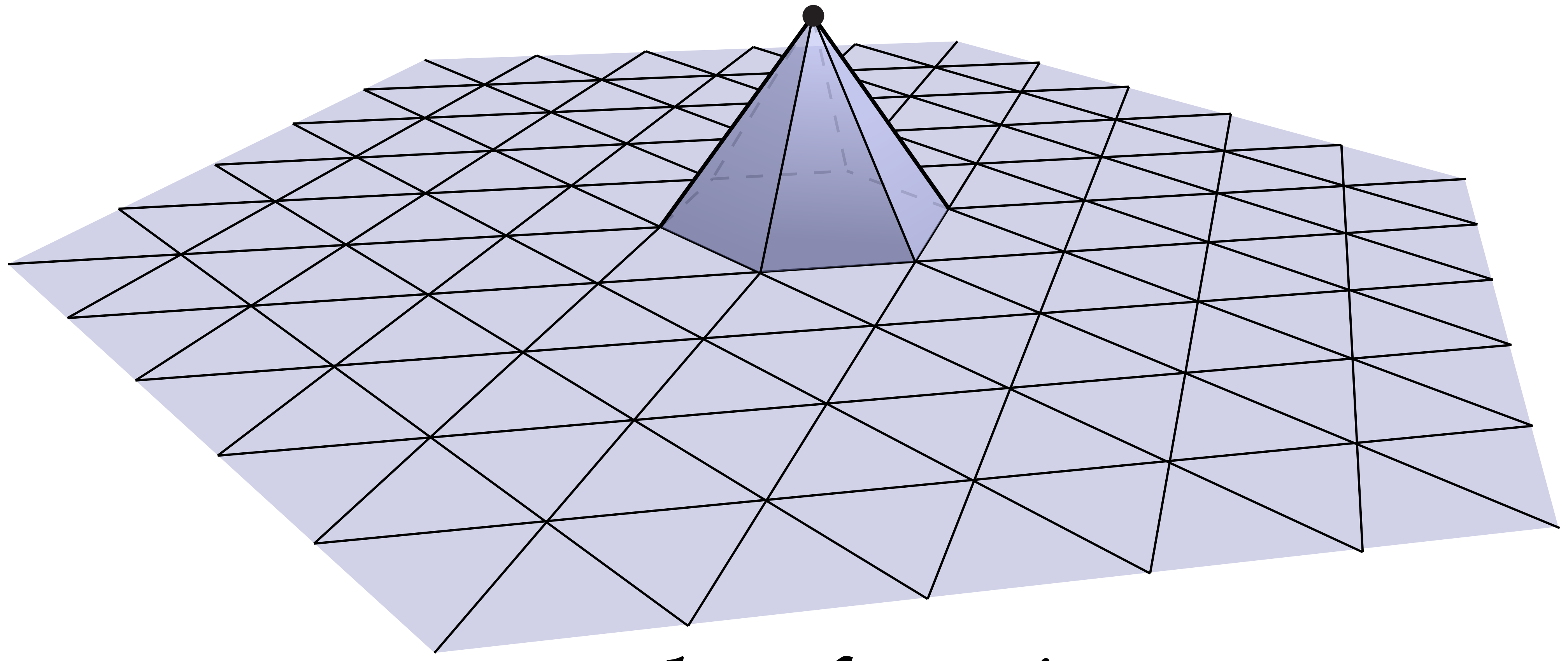
On any simplicial complex K, the *hat function* a.k.a. *Lagrange basis* ϕ_i is a real-valued function that is linear over each simplex and satisfies

$$\phi_i(v_j) = \delta_{ij},$$

for each vertex v_j, *i.e.*, it equals 1 at vertex i and 0 at vertex j. Given a (primal) discrete 0-form $u : V \to \mathbb{R}$, we can construct an *interpolating* 1-form via

$$\sum_i u_i \phi_i,$$

i.e., we simply weight the hat functions by values at vertices.

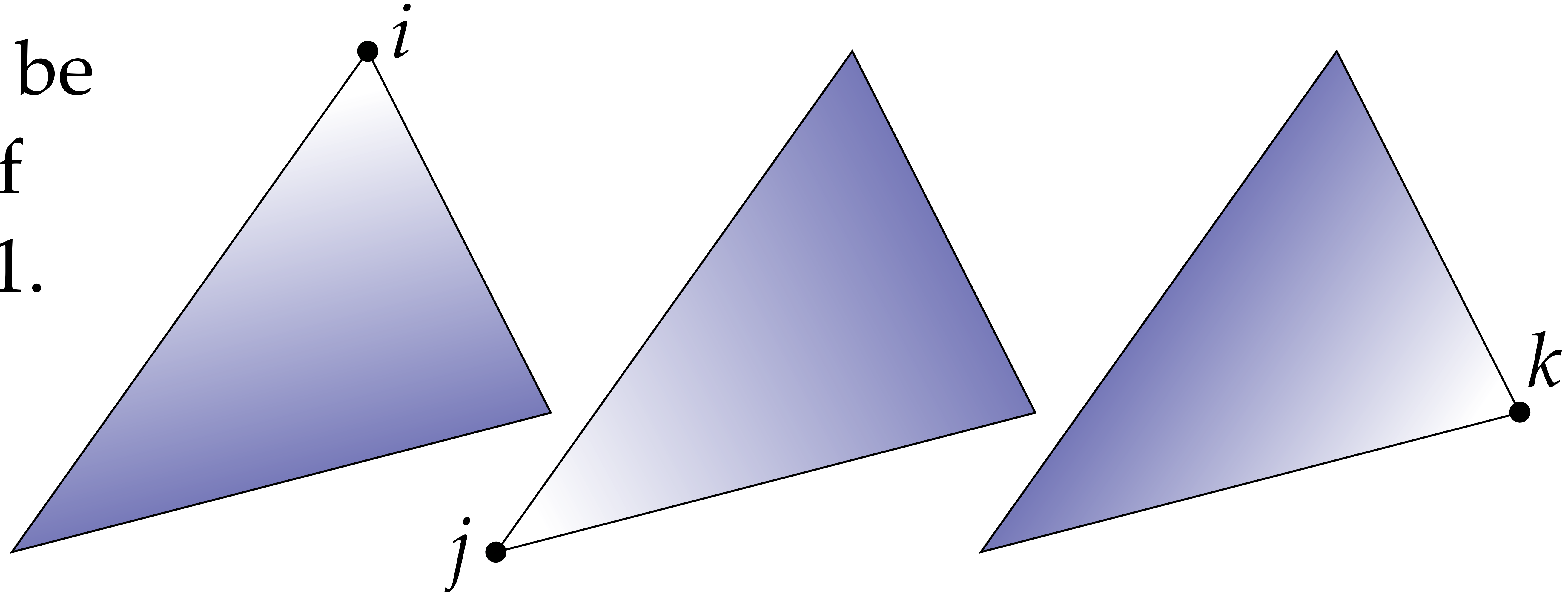


hat function ϕ_i

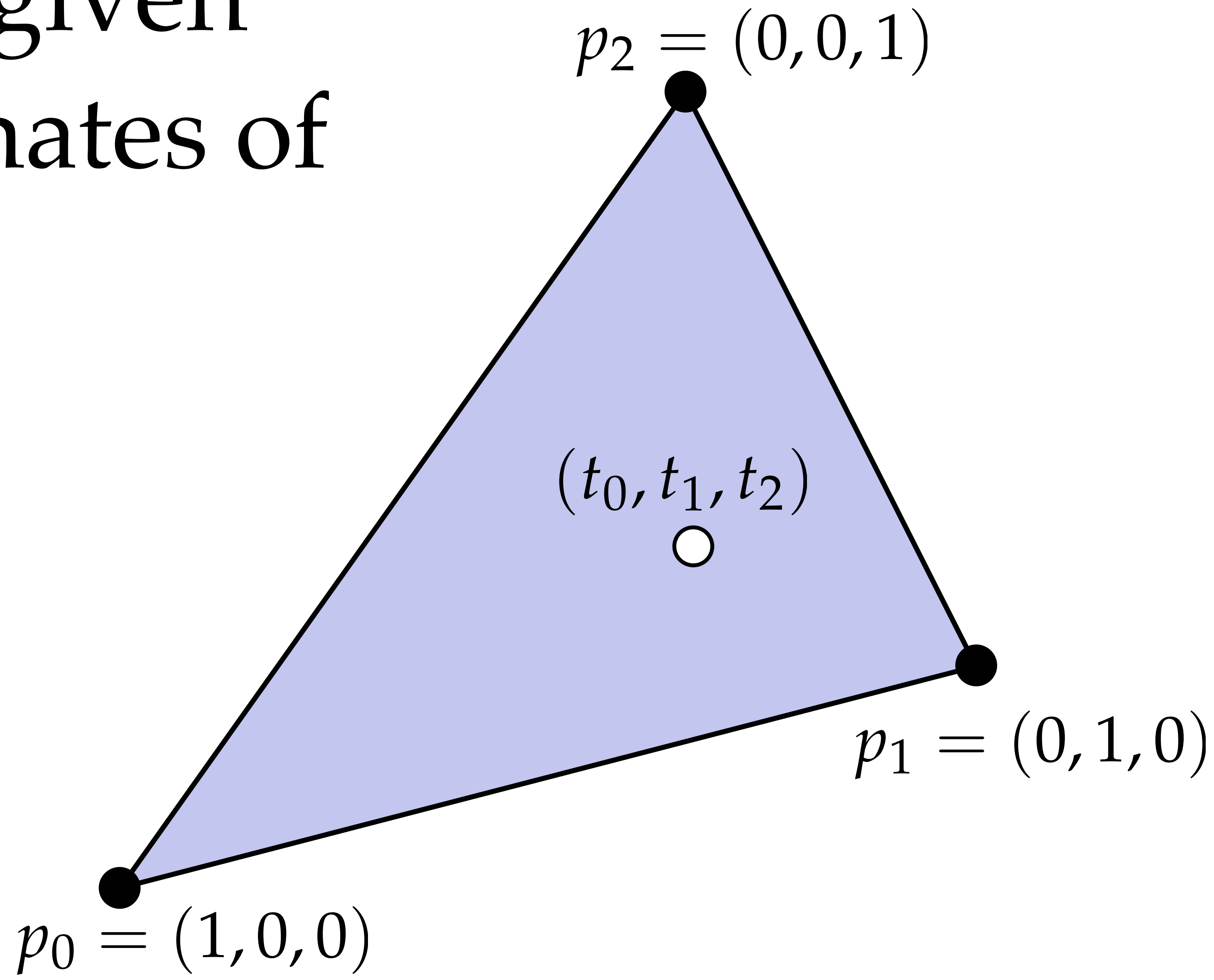
Note: result is a *continuous* 0-form.

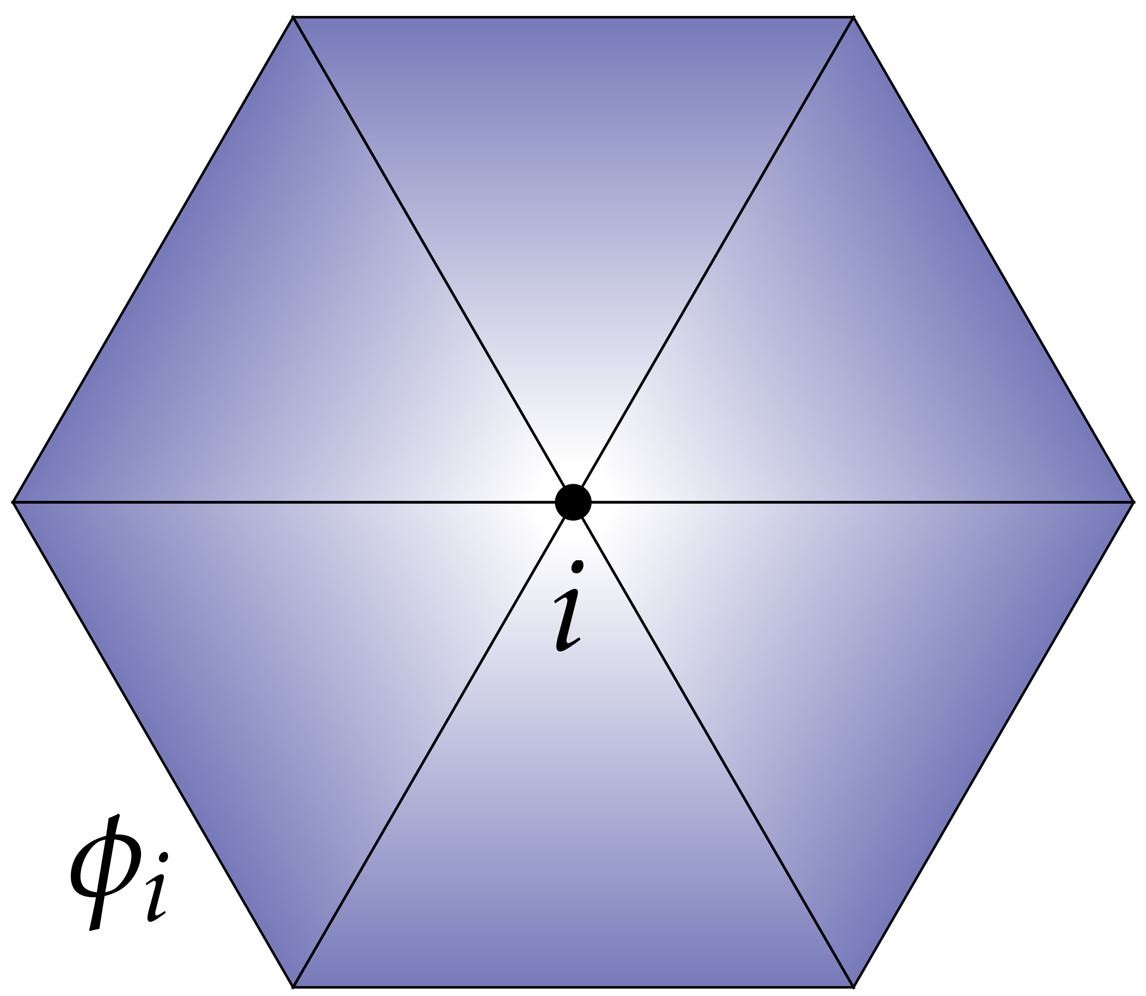
Barycentric Coordinates—Revisited

- Recall that any point in a k-simplex can be expressed as a weighted combination of the vertices, where the weights sum to 1.
- The weights t_i are called the *barycentric coordinates*.
- The Lagrange basis for a vertex i is given explicitly by the barycentric coordinates of i in each triangle containing i.



$$\sigma = \left\{ \sum_{i=0}^{k} t_i p_i \,\middle|\, \sum_{i=0}^{k} t_i = 1,\ t_i \geq 0\ \forall i \right\}$$





Interpolation—k-Forms (Whitney Map)

Definition. Let ϕ_i be the hat functions on a simplicial complex. The *Whitney 1-forms* are differential 1-forms associated with each oriented edge ij, given by

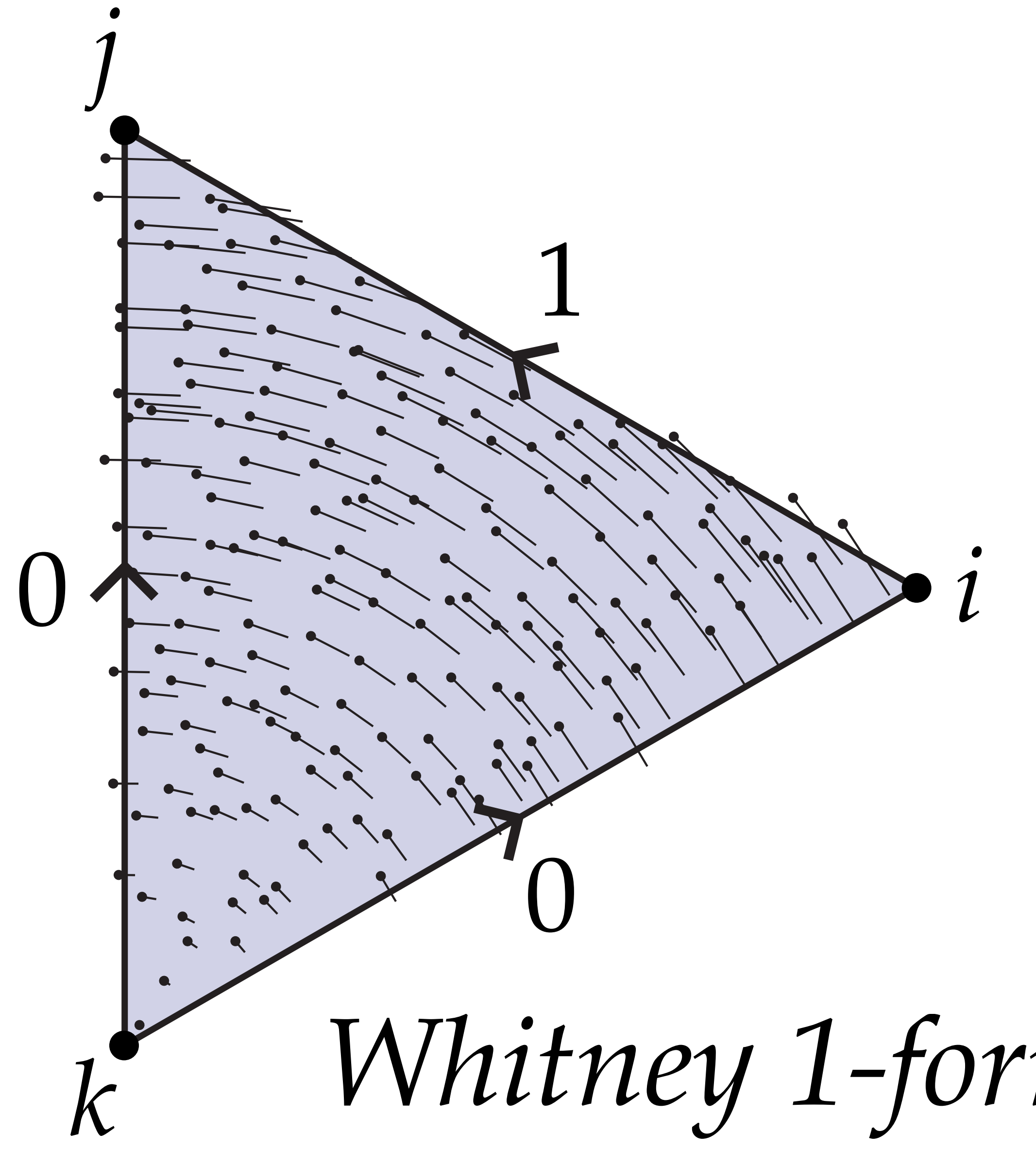
$$\phi_{ij} := \phi_i \, d\phi_j - \phi_j \, d\phi_i$$

(Note that $\phi_{ij} = -\phi_{ji}$). The Whitney 1-forms can be used to interpolate a discrete 1-form $\widehat{\omega}$ (value per edge) via

$$\sum_{ij} \widehat{\omega}_{ij} \phi_{ij}.$$

More generally, the *Whitney k-form* associated with an oriented k-simplex $(i_0, \ldots, i_k)$ is given by

$$\sum_{p=0}^{k} (-1)^p \phi_{i_p} d\phi_{i_0} \wedge \cdots \wedge \cancel{d\phi_{i_p}} \wedge \cdots \wedge d\phi_{i_k}$$



Whitney 1-form ϕ_{ij}

Discretization & Interpolation

- **<u>Fact:</u>** Suppose we have a discrete differential k-form. If we interpolate by Whitney bases, then discretize via the de Rham map (i.e., by integration), then we recover the exact same discrete k-form.

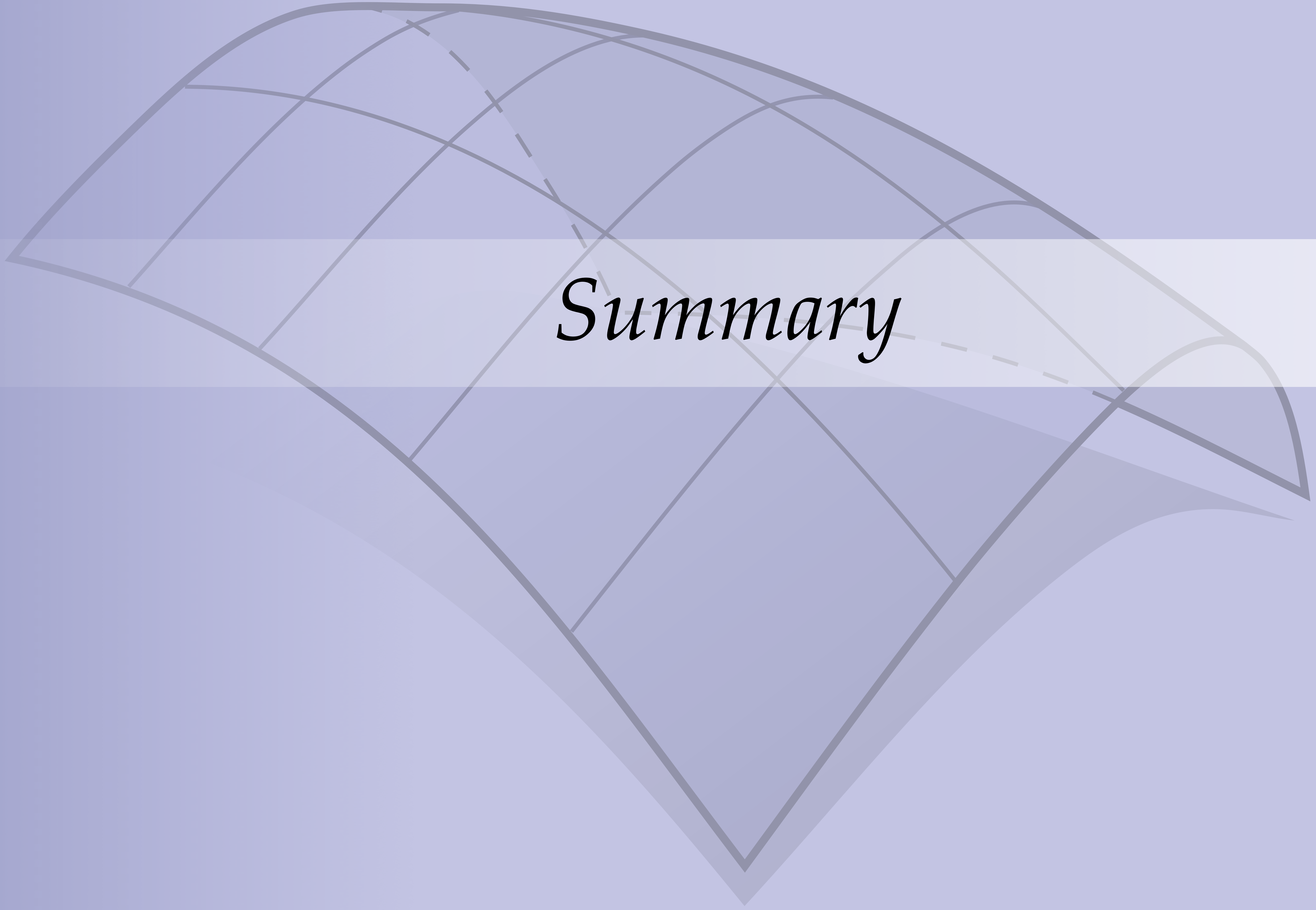
$$\Omega_k \quad \textit{(smooth differential k-forms)}$$

$$\textit{(discretize)} \; \int \downarrow \; \uparrow \; \phi \quad \textit{(interpolate)}$$

$$\widehat{\Omega}_k \quad \textit{(discrete differential k-forms)}$$

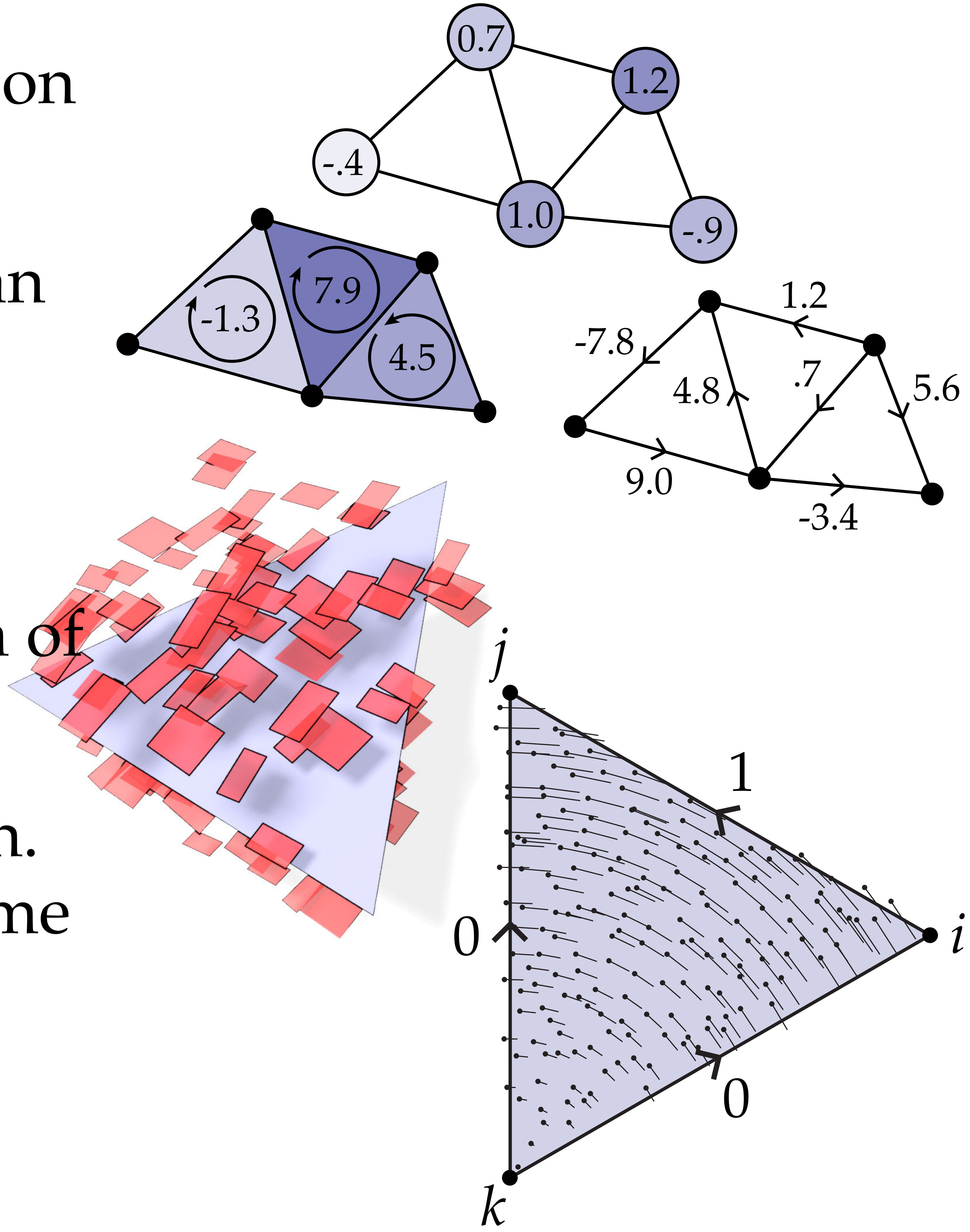
Q: What about the other direction? If we discretize a continuous k-form then interpolate, will we always recover the same continuous k-form?

Summary

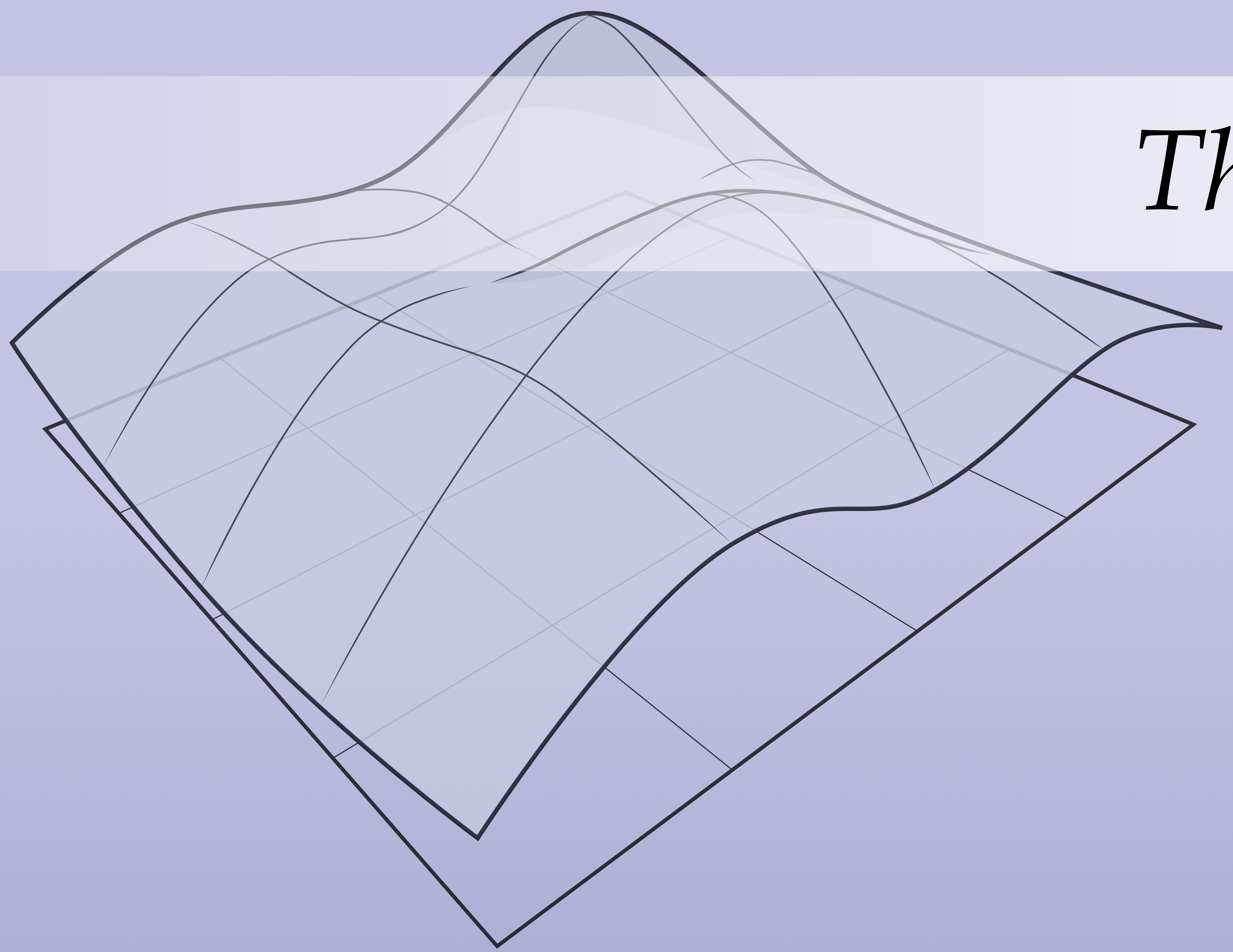


Discrete Differential Forms — Summary

- A *discrete differential k-form* amounts to a value stored on each oriented *k*-simplex
- **Discretization:** given a smooth differential k-form, can approximate by a discrete differential k-form by integrating over each k-simplex
- **Interpolation:** given a discrete differential k-form, construct a continuous one by taking a weighted sum of basis k-forms
- *In practice,* almost never comes from direct integration. More typically, values start at vertices (samples of some function); 1-forms, 2-forms, *etc.*, arise from applying operators like the (discrete) exterior derivative.
- Next lecture: develop these operators!



Thanks!



DISCRETE DIFFERENTIAL GEOMETRY

AN APPLIED INTRODUCTION