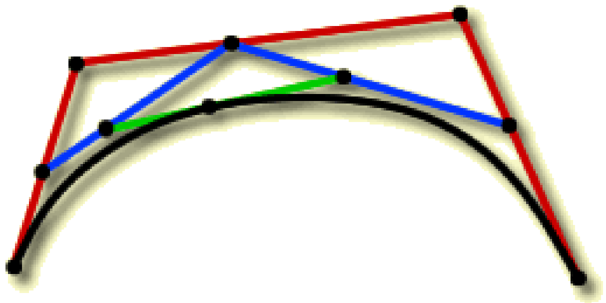
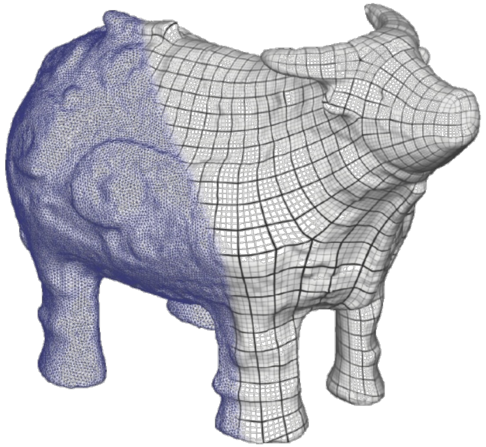
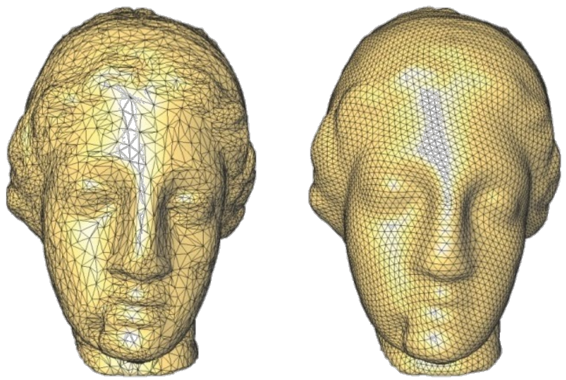
CS348a: Geometric Modeling and Processing







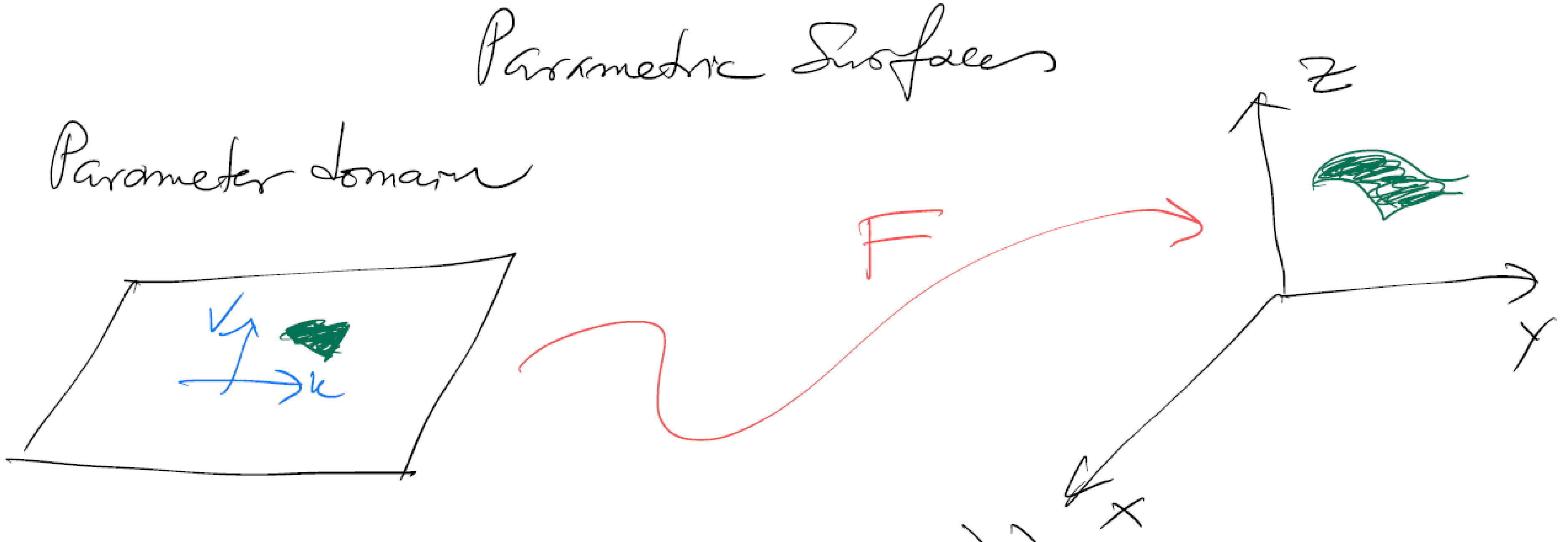
Leonidas Guibas
Computer Science Department
Stanford University

Leonidas Guibas Laboratory

Geometric Computing

Last Time:
Parametric Surfaces,
Spline Surfaces

Parametric Surfaces



$$F(u,v) = (X(u,v), Y(u,v), Z(u,v))$$

Polynomials

$$X(u,v) = 2uv + 3uv^2 + 7u^2$$

$$Y =$$

$$Z =$$

Tensor Product Surfaces

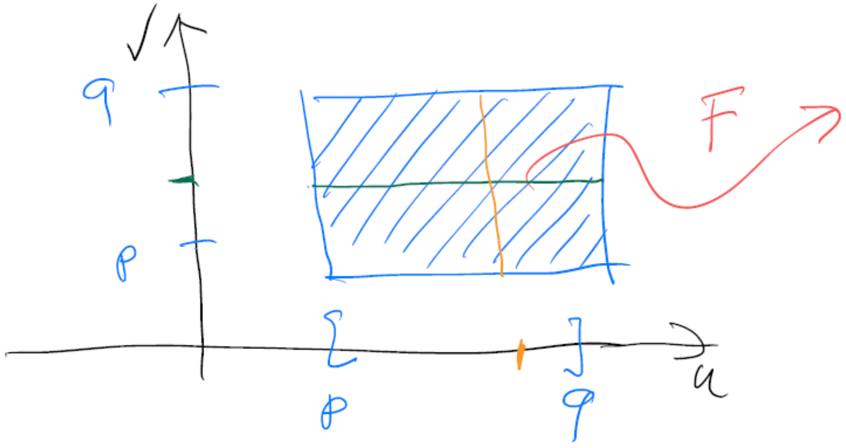
Curves of Curves

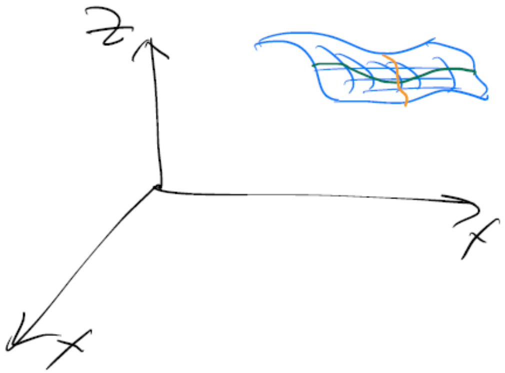
$$X(u,v) = 2uv + 3uv^2 + 7u^2$$

$(2,2)$ $\quad u \to u_1, u_2 \quad v \to v_1, v_2$

Tensor Product Surfaces

$$x\big((u_1,u_2);(v_1,v_2)\big) = 2\left(\frac{u_1+u_2}{2}\right)\left(\frac{v_1+v_2}{2}\right) + 3\left(\frac{u_1+u_2}{2}\right)v_1 v_2 + 7u_1u_2$$





Bézier Control Points: Rectangular Grid

Control Points of

$F([p,q],[r,s])$

$pp \quad pq \quad qq \qquad rr \quad rs \quad ss$

$d=2$

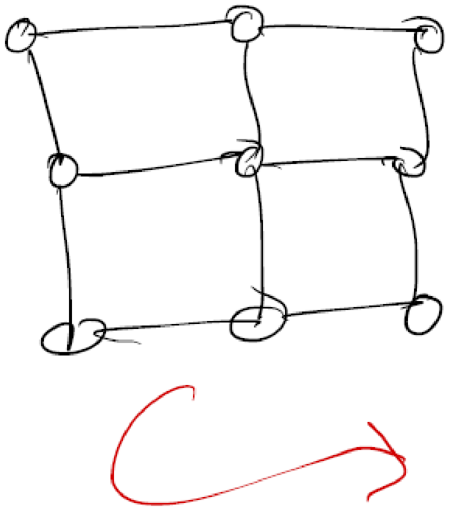
Bi-quadratic

9 control points

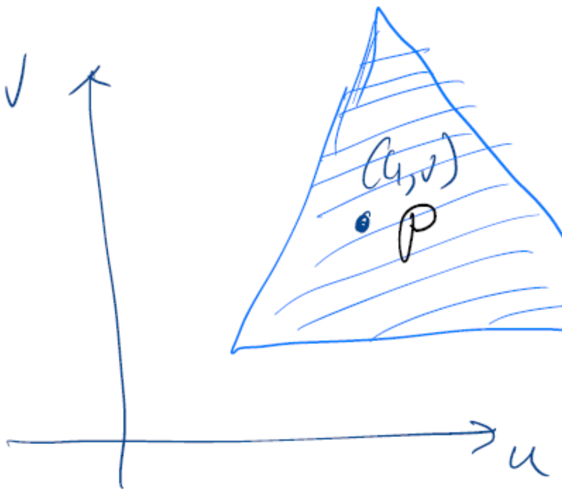
$f(pp;rr) \quad f(pq,rr) \quad f(qq,rr)$

$f(pp;rs) \quad f(pq,rs) \quad f(qq,rs)$

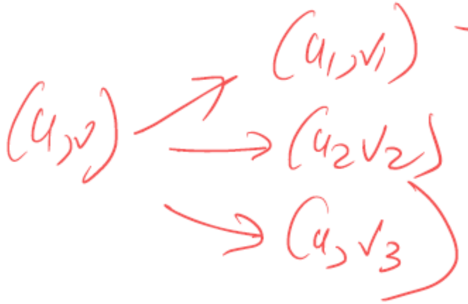
$f(pp;rr) \quad f(pq;rs) \quad f(qq,ss)$



Triangular Patch / Total Degree Surfaces



$$X(u,v) = 2uv + \underbrace{3uv^2}_{3} + 7u^2$$



Total degree Surfaces

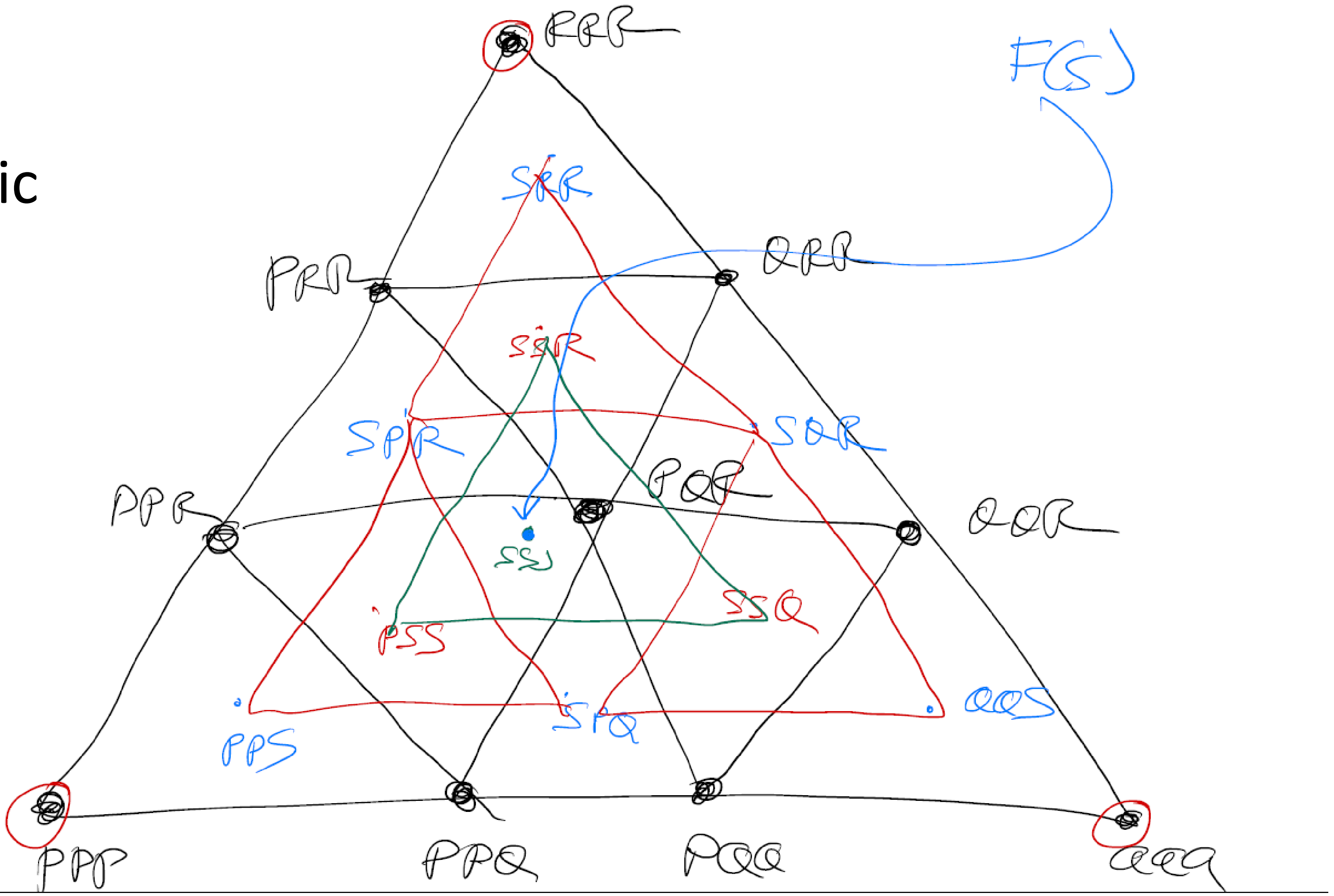
$$uv = \frac{1}{6}\left(u_1v_2 + u_1v_3 + u_2v_1 + u_2v_3 + u_3v_1 + u_3v_2\right)$$

$$uv^2 = \frac{1}{3}\left(u_1v_2v_3 + u_2v_1v_3 + u_3v_1v_2\right)$$

$$u^2 = \frac{1}{3}\left(u_1u_2 + u_2u_3 + u_3u_1\right)$$

Triangular Bézier Control Net

Cubic



C^1 Continuity

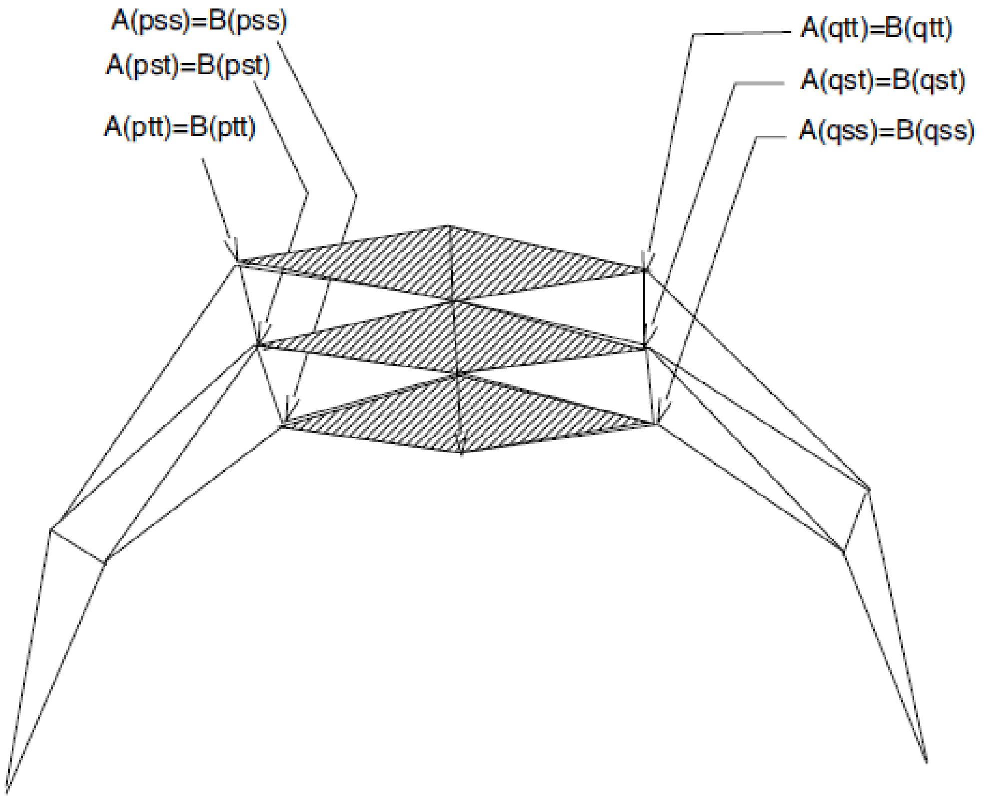


Figure 5: The Bezier points of two cubic surfaces that join with C^1 continuity

C^2 Continuity

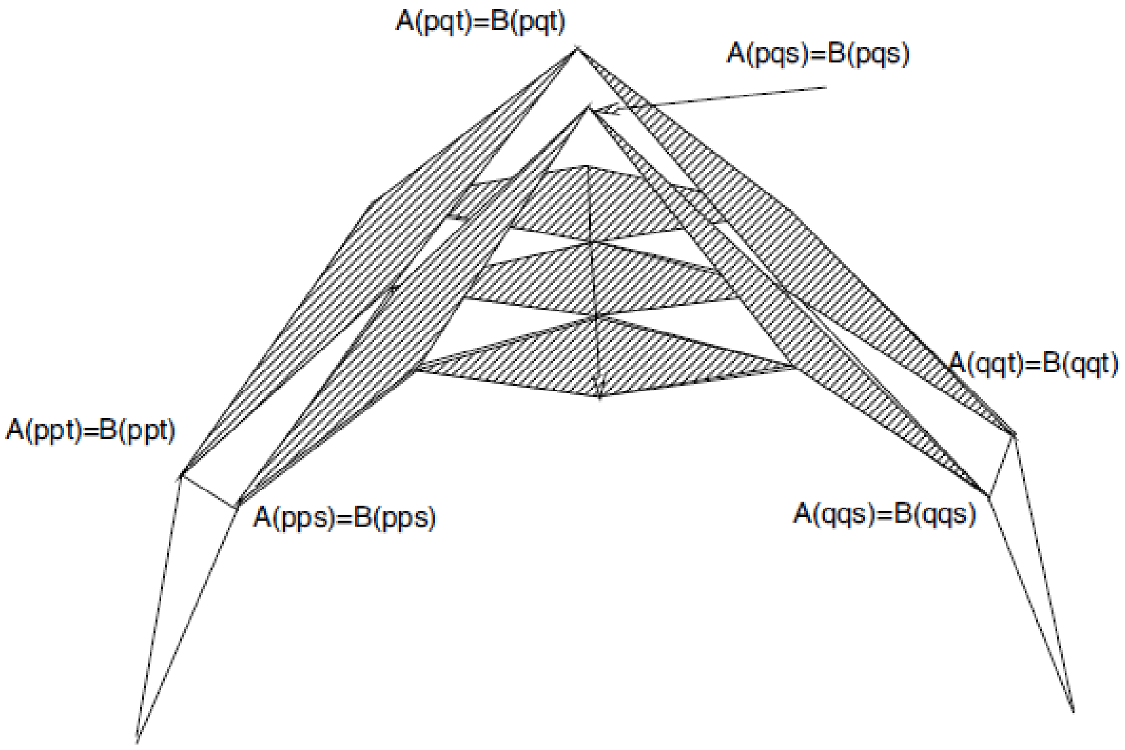
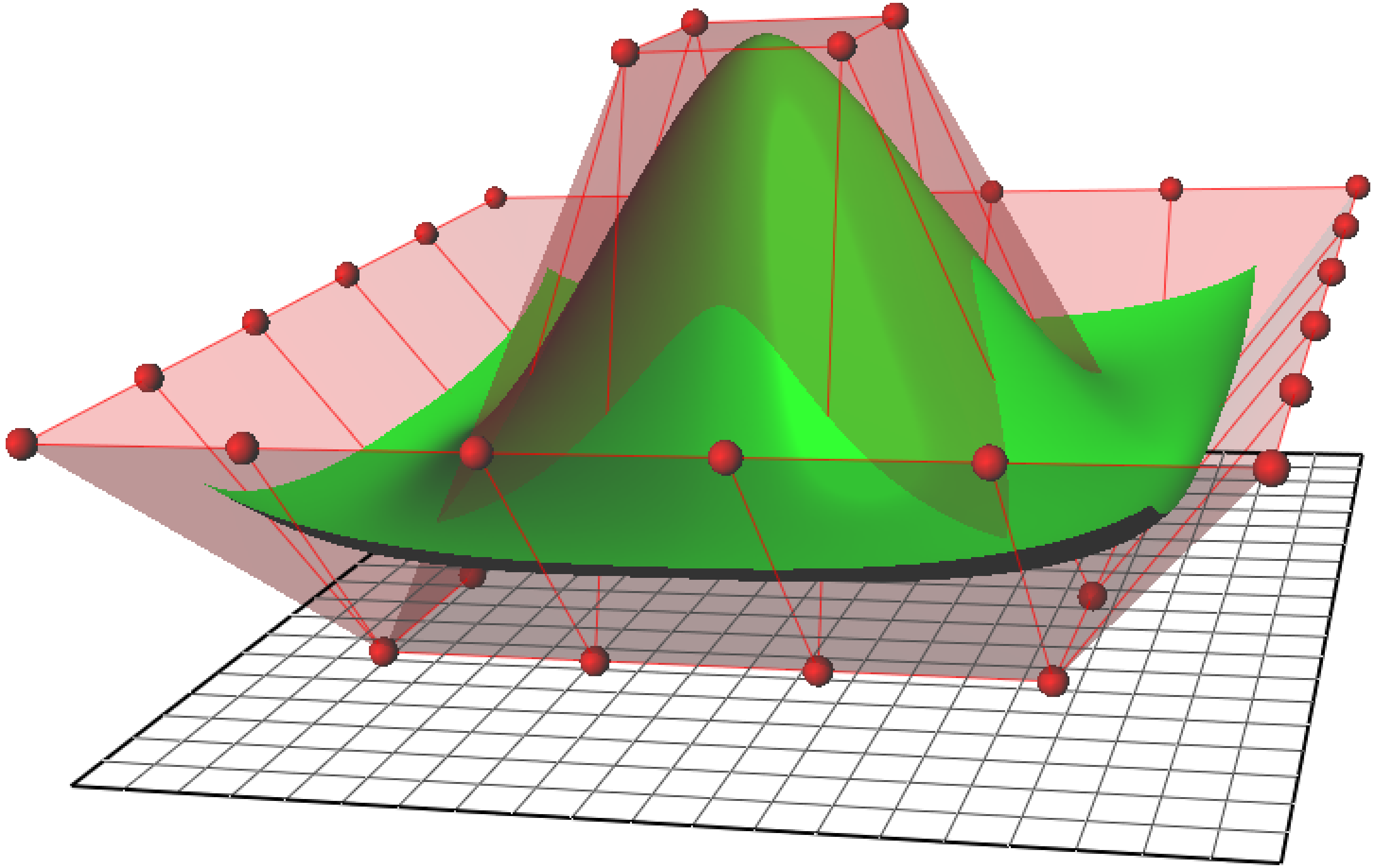
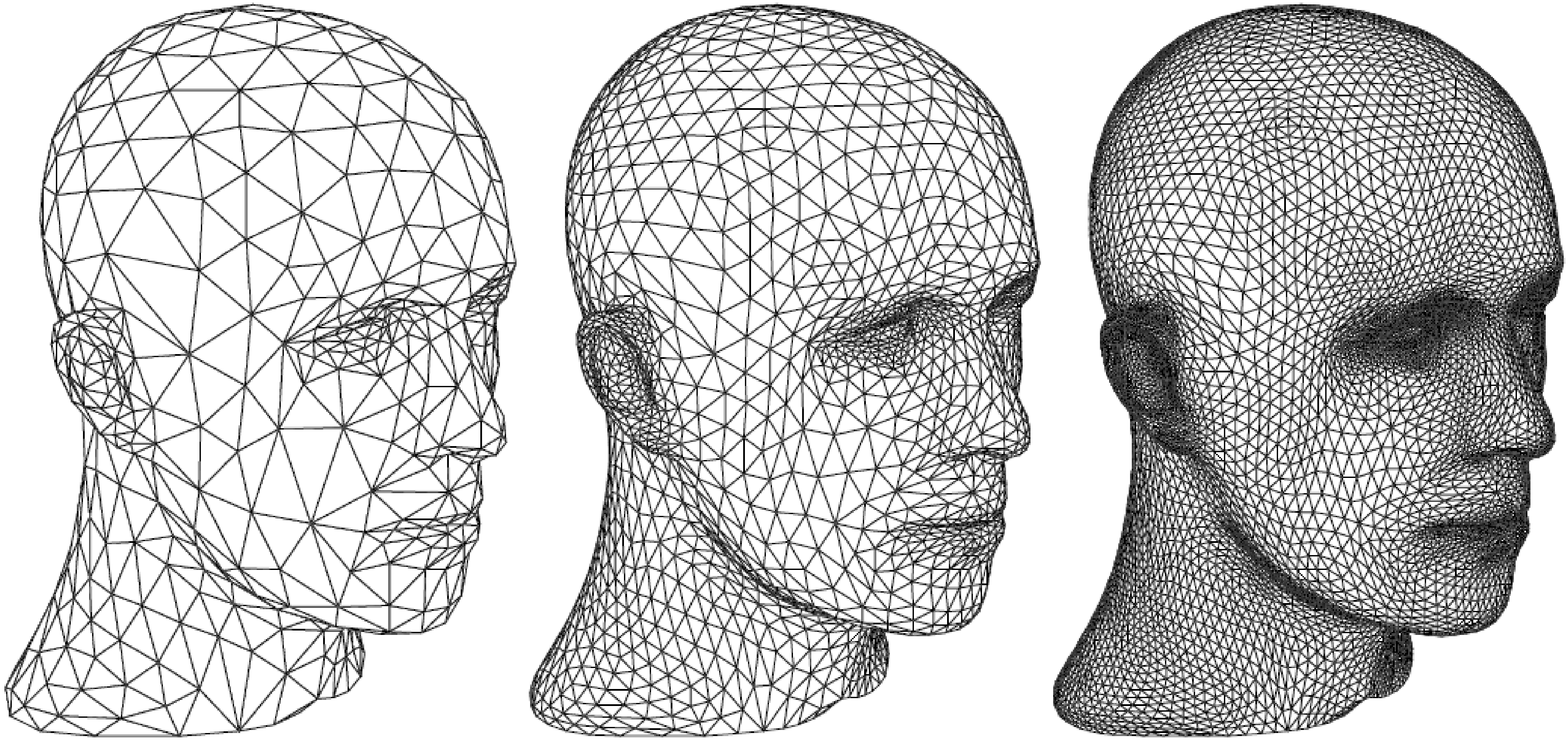


Figure 6: The Bezier points of two cubic surfaces that join with C^2 continuity

NURBS – Non-Uniform Rational B-Splines

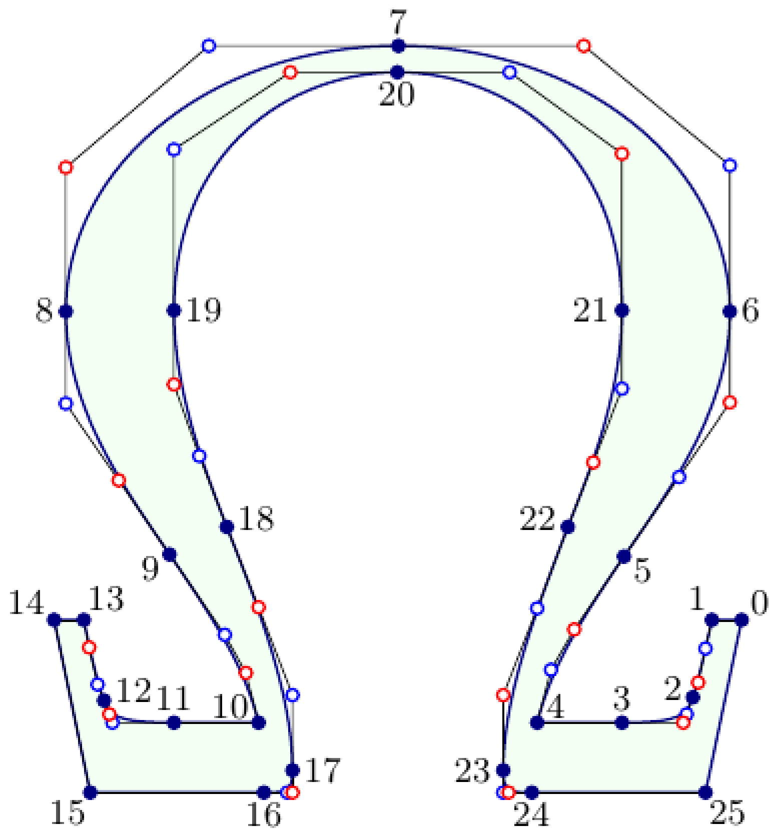


Subdivision Surfaces



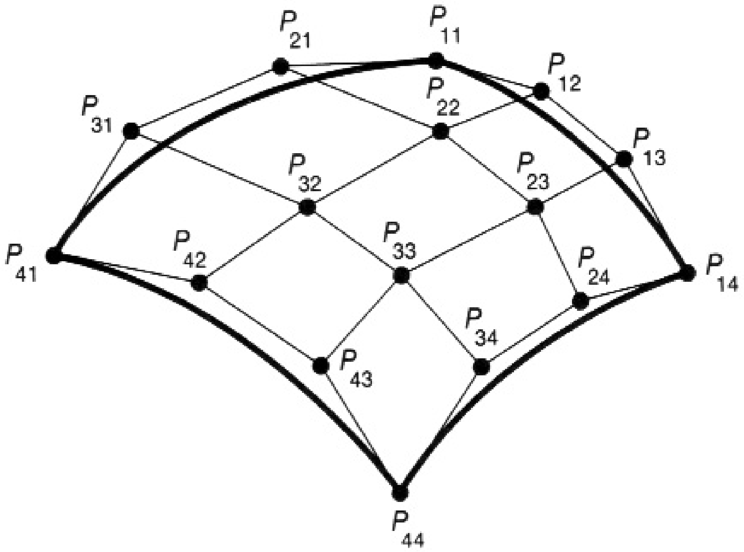
Today:
Stitching Patches Together, Quad-Edge Data Structure

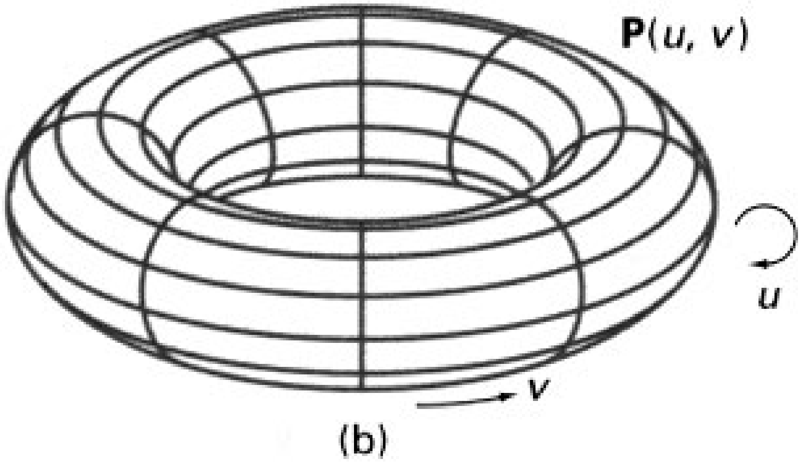
Modeling 2D Shapes with Spline Curves

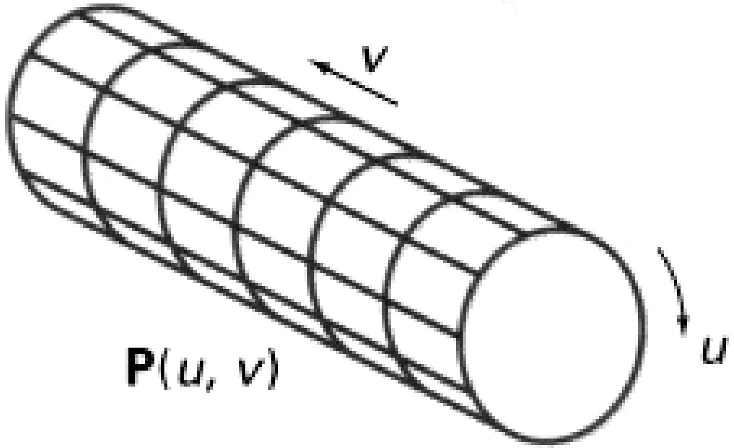


The fonts we use …

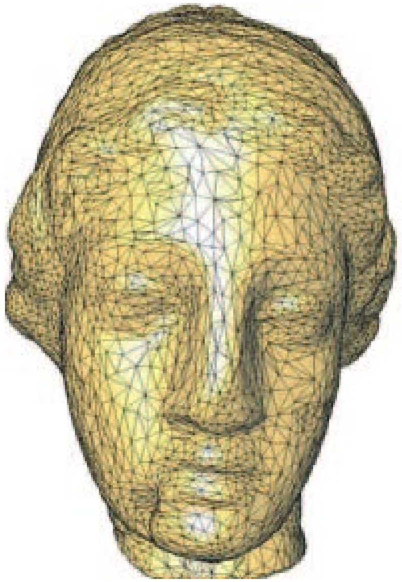
Tensor Product B-Spline Topology Limitations

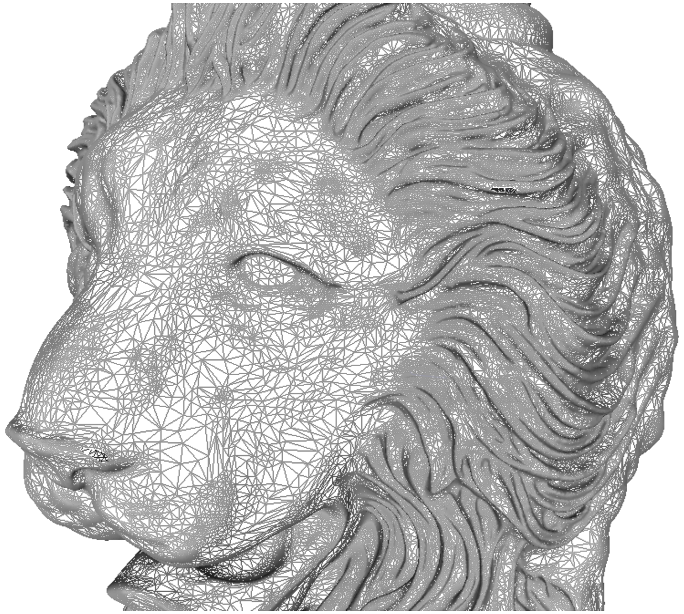


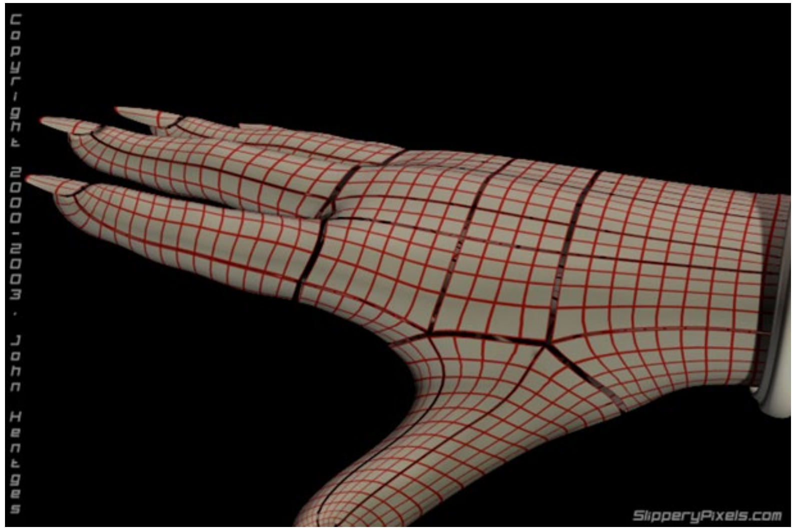




Meshes of Patches Can Be Irregular

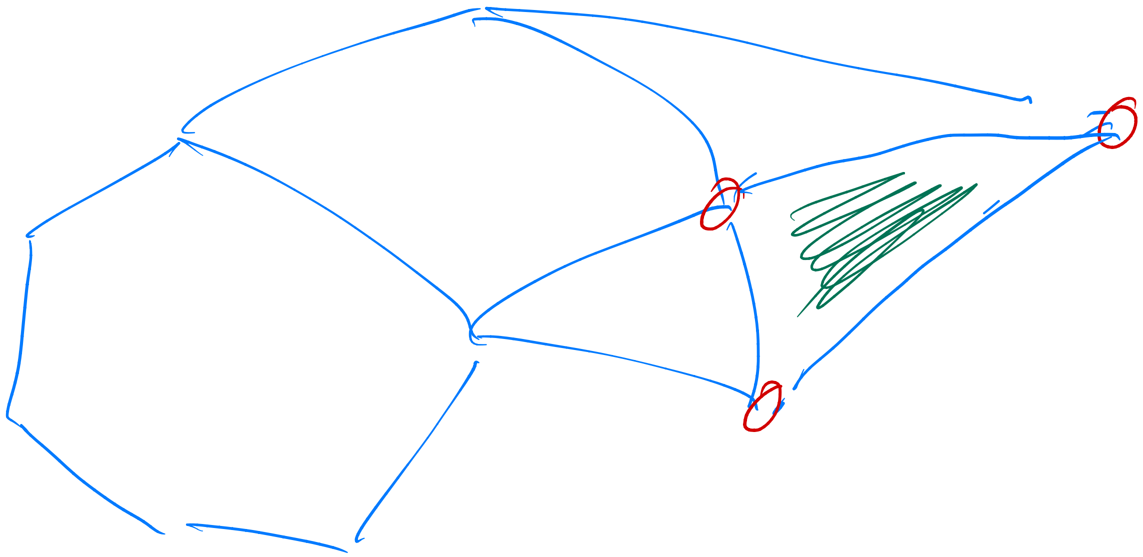






Whiteboard

Networks of Patches & How They Are Joined

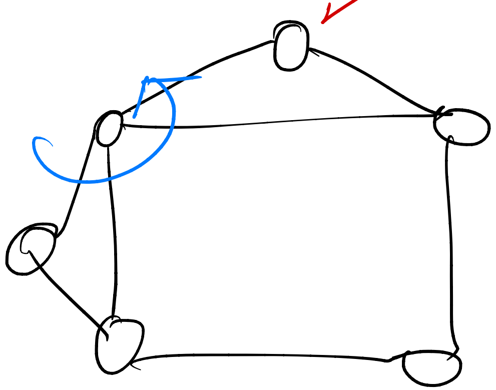


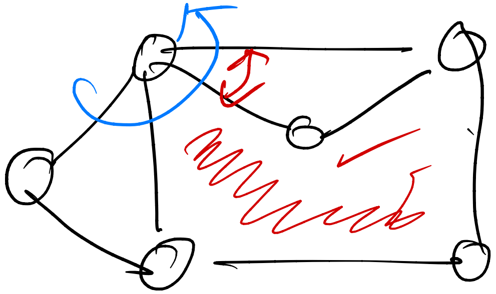
0-d vertices
1-d edges
2-d faces

Embedded Graph

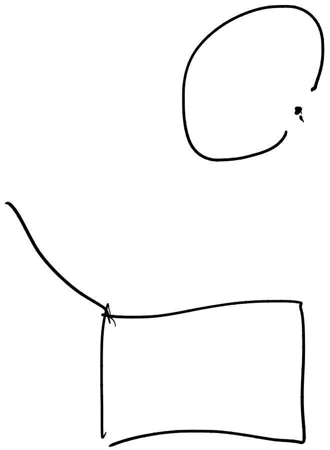
graph embedded in a 2-D manifold

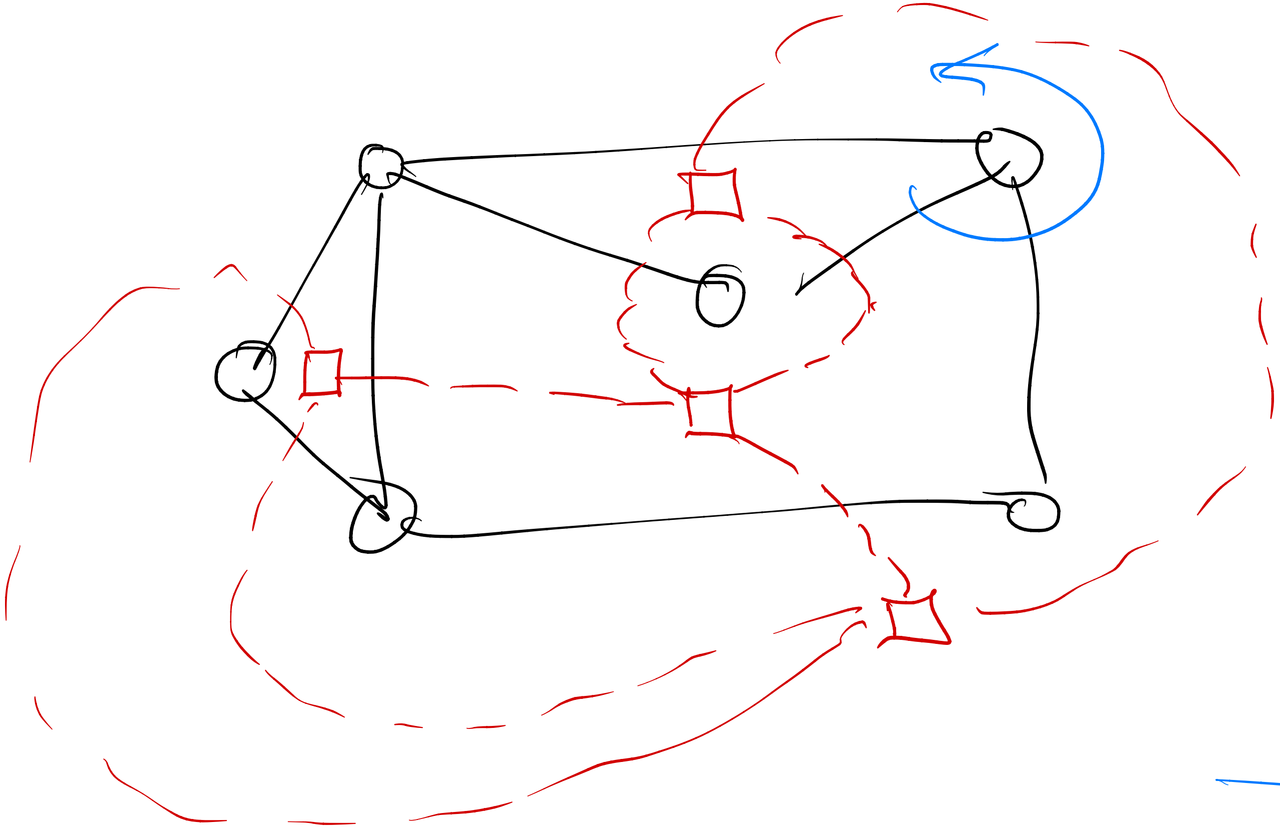
Embedded Graphs





Embedded Graphs have Duals





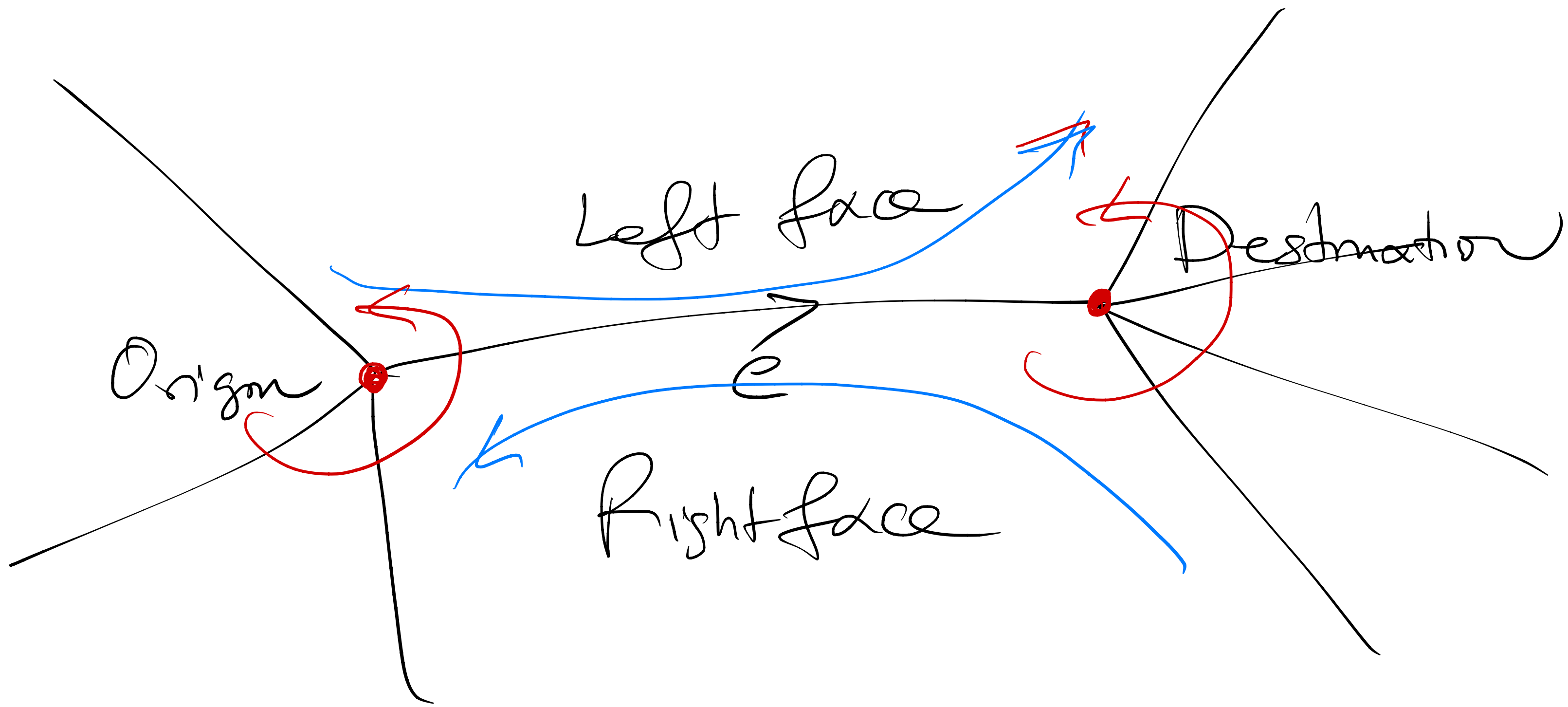
Dual Graph

Vertex ↔ Face

Edges ↔ Edges

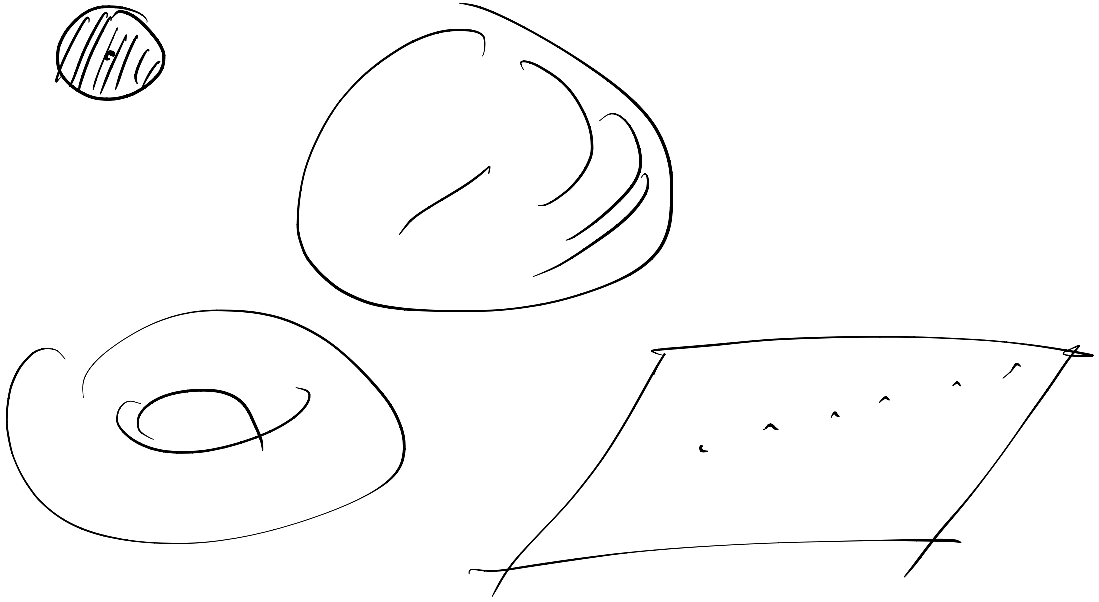
Faces ↔ Vertex

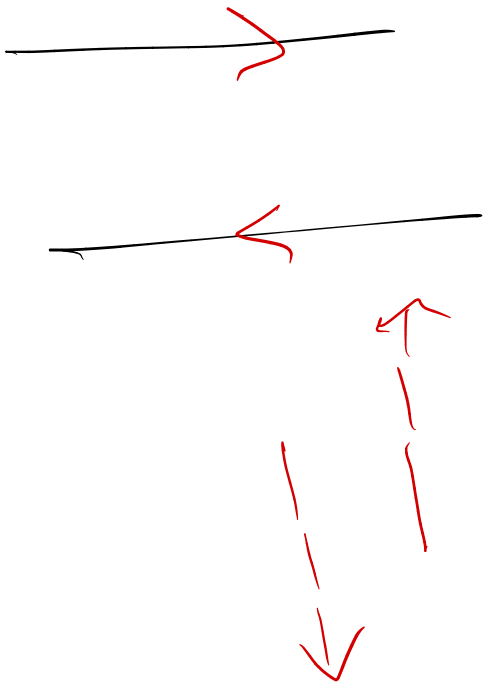
Edge-Centric Point of View

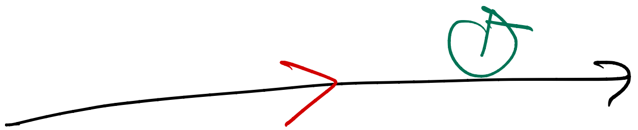


2d manifolds

- bounded
- compact
- closed
- orientable

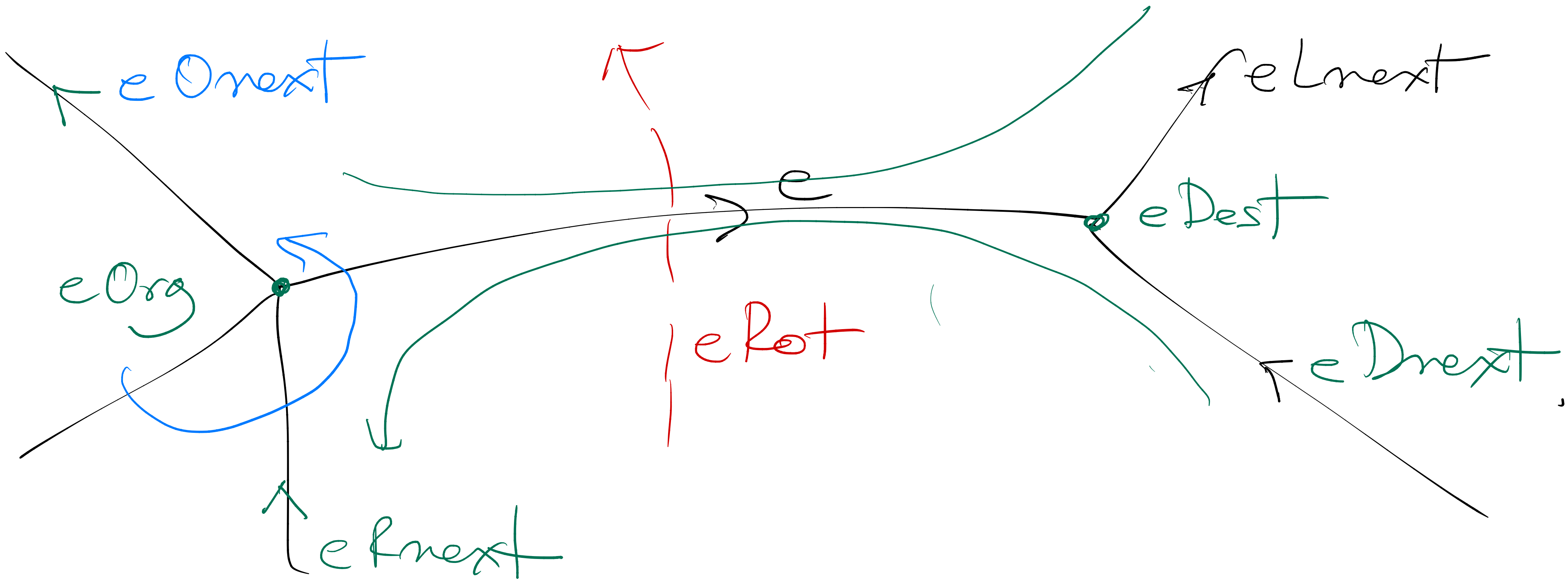






Edge Algebra

$e \mapsto e\text{Rot}$



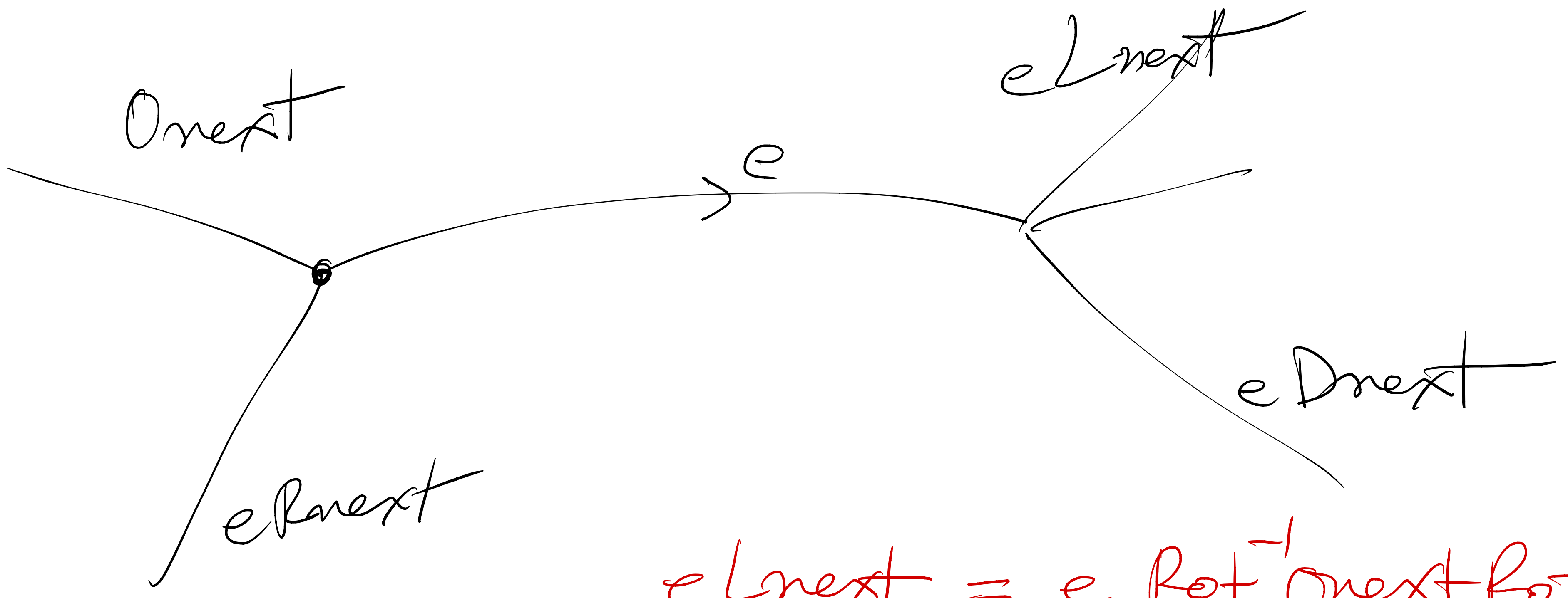
e

$eRot$

$eRot^2 = eSym$

$eRot^3 = eRotSym$

$eRot^4 = e$

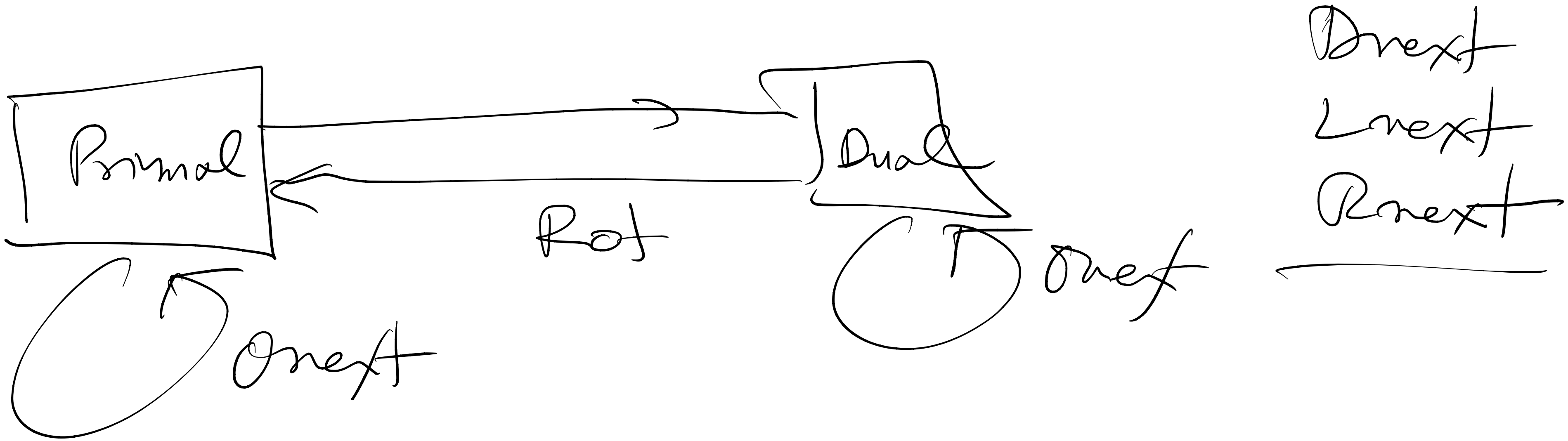


$$e\,Lnext = e\,Rot^{-1}\,Onext\,Rot$$

$$e\,Rnext = e\,Rot\,Onext\,Rot^{-1}$$

$$e\,Dnext = e\,Rot^{2}\,Onext\,Rot^{2}$$

Edge Algebra



2D

Rot Onext

$e\,Rot^4 = e$

$e\,Rot\,Onext\,Rot\,Onext = e$

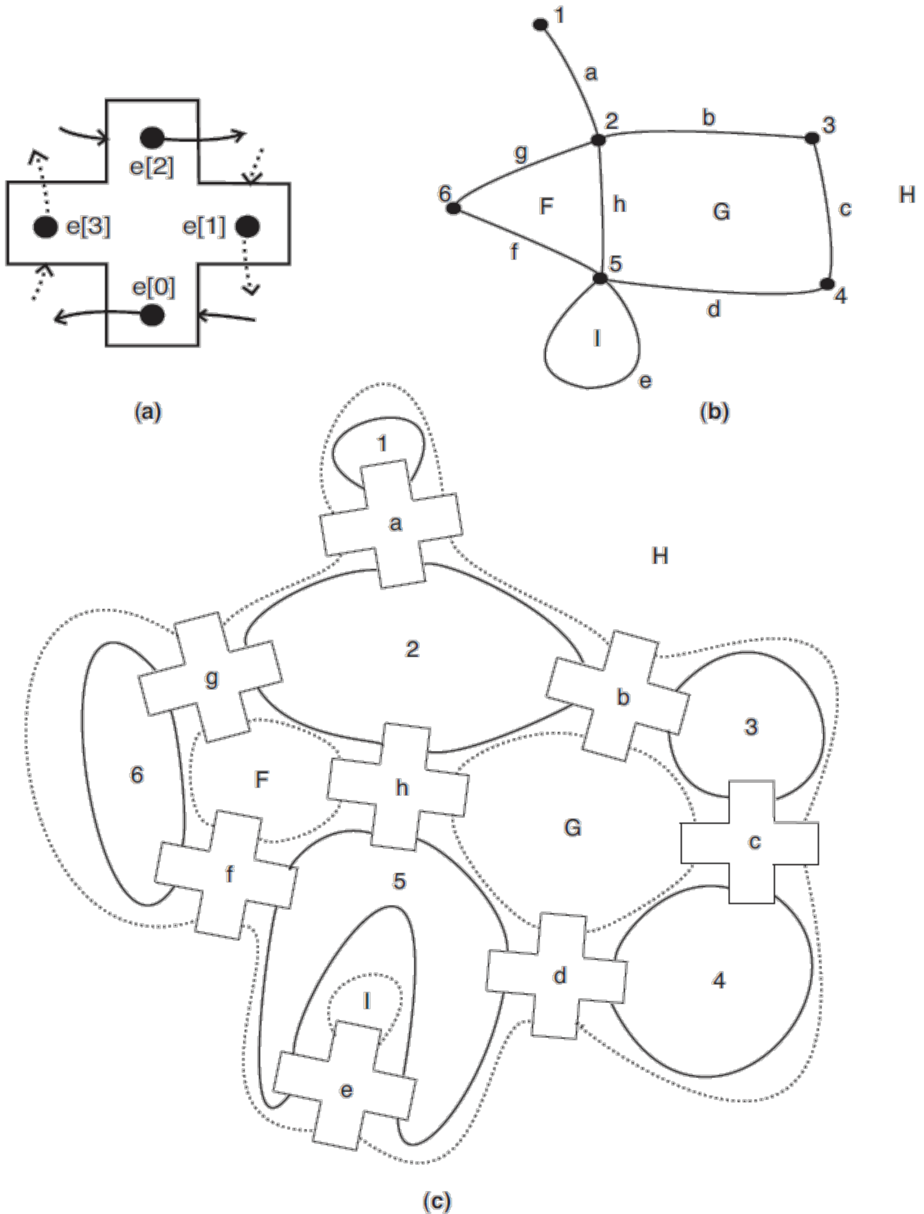
$e\,Rot^2 \neq e$

e Primal ~~iff~~ e ~~Rot~~ dual

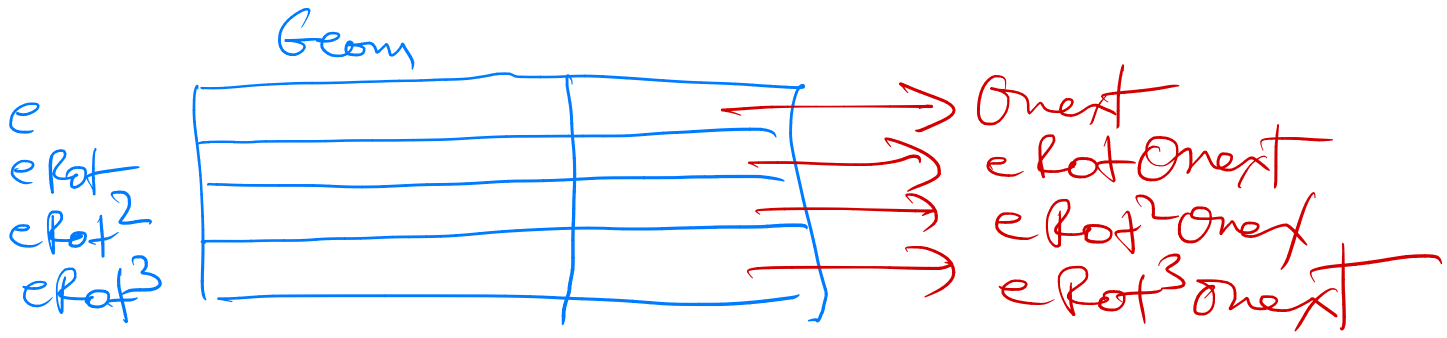
e Primal iff e Onext Primal

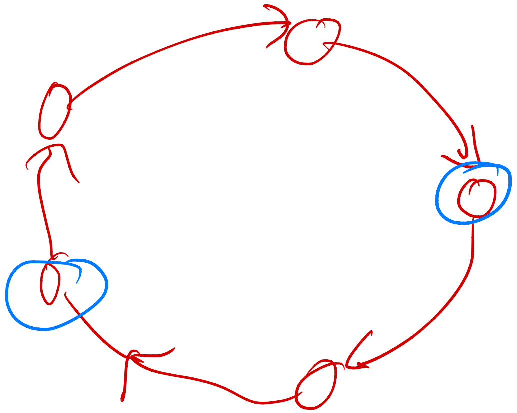
The Quad-Edge Data Structure

The Quad-Edge Data Structure

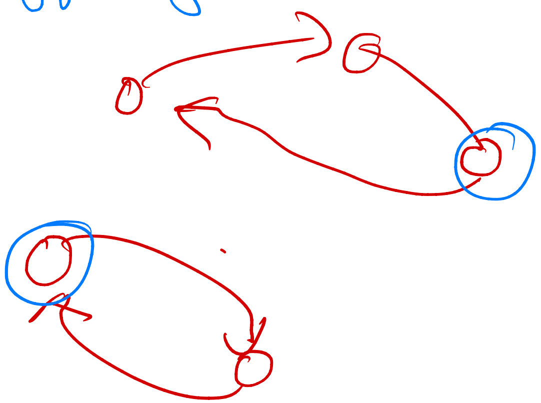


Edge Records





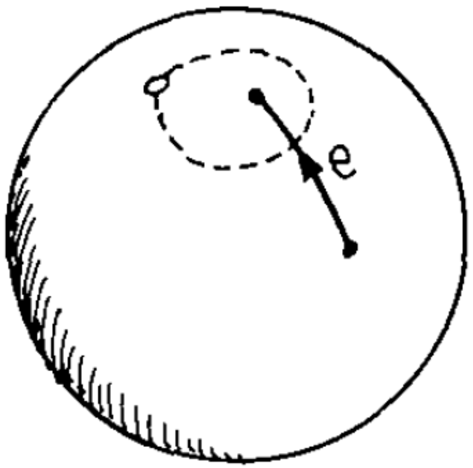
Pointer Swapping

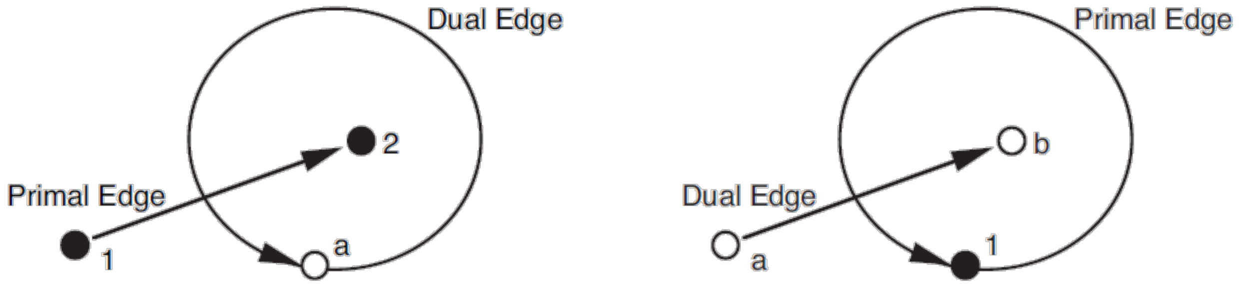


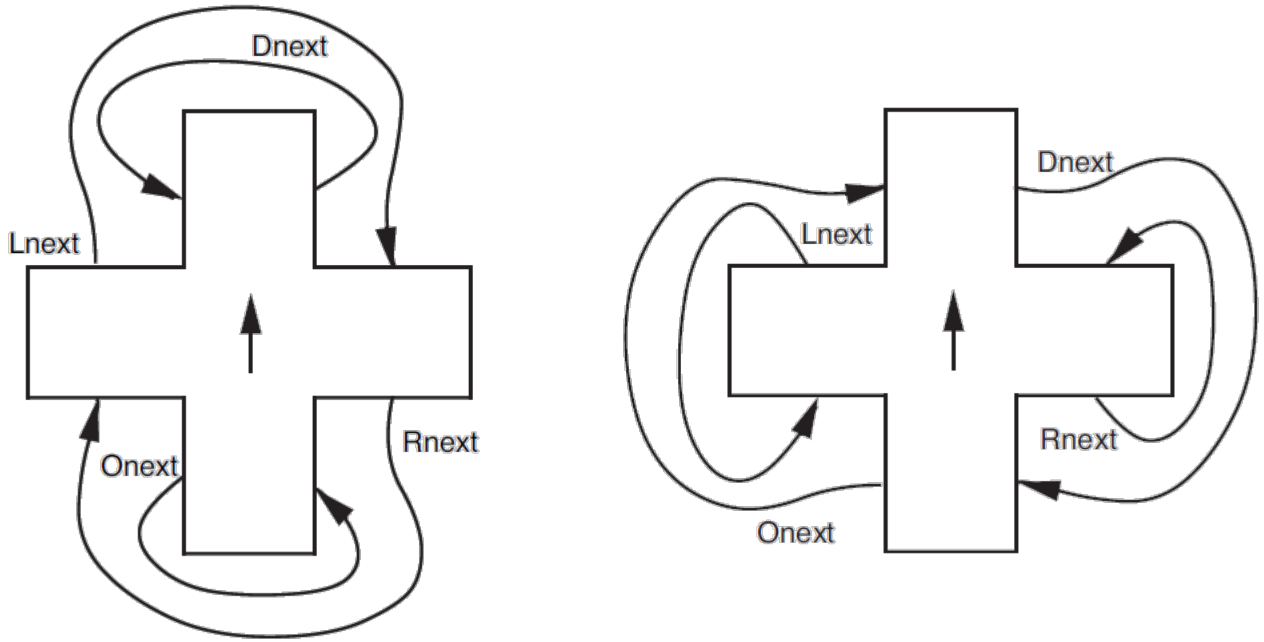
Quad-Edge
Modification Operators

MakeEdge

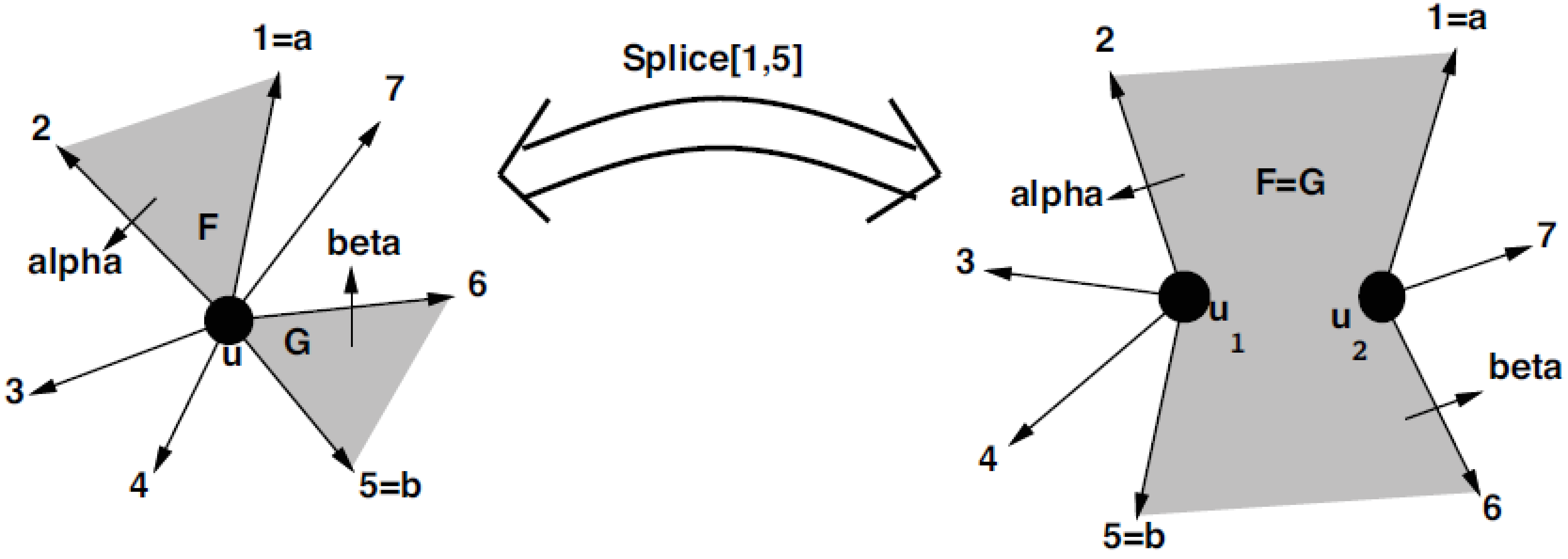
MakeEdge[]







Splice, Case I



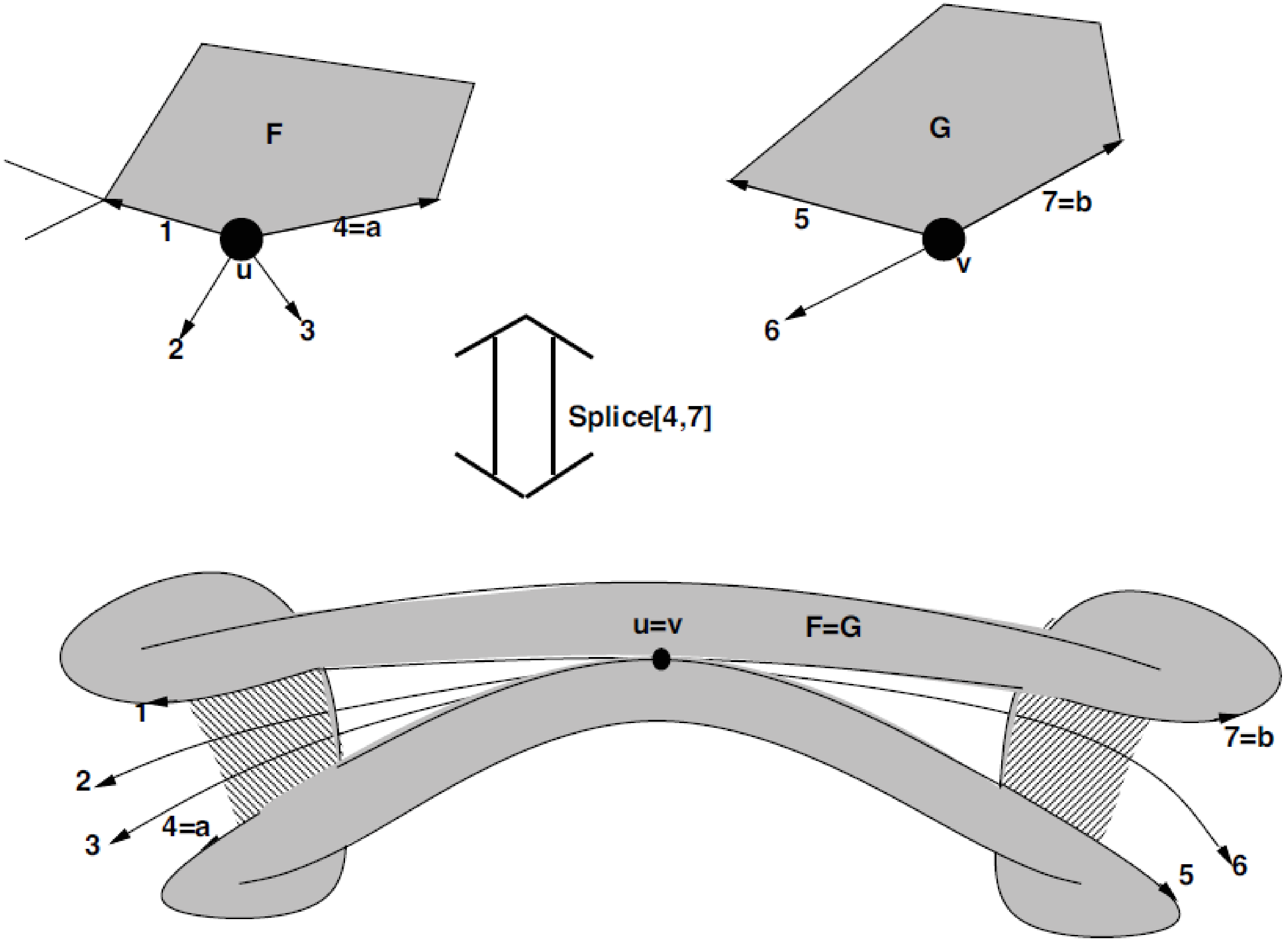
α = a Onext Rot
β= b Onext Rot
Swap[a Onext, b Onext]
Swap[α Onext, β Onext]

Splice[a, b]

Splice, Case II

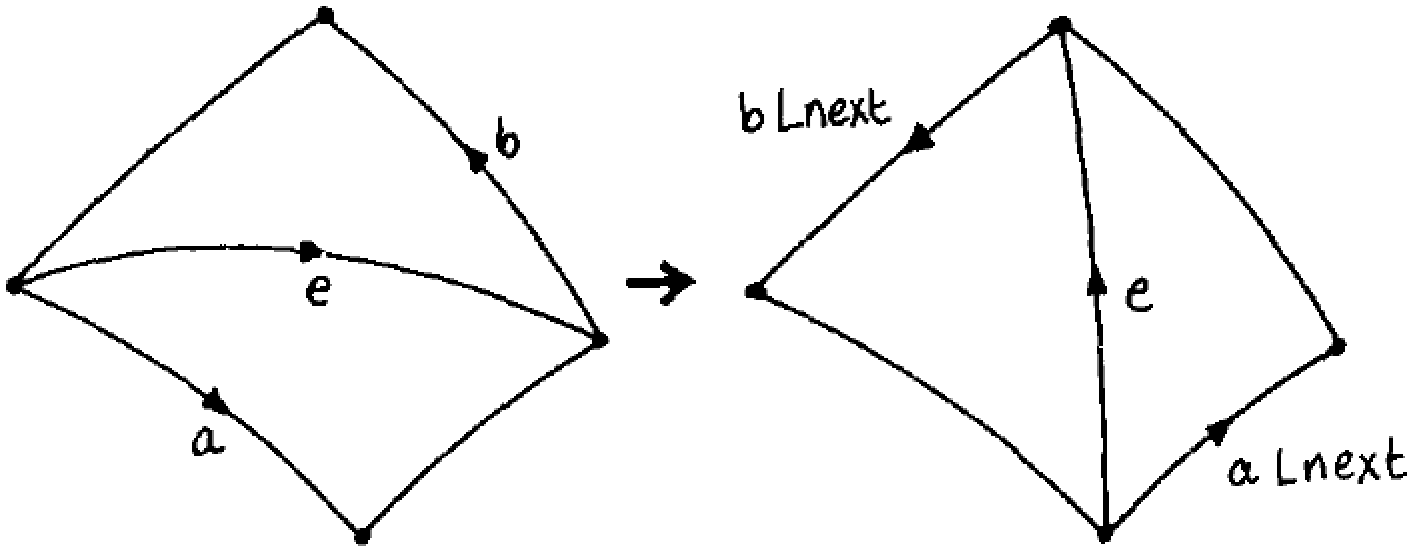
Splice[a, b]

α = a Onext Rot
β= b Onext Rot
Swap[a Onext, b Onext]
Swap[α Onext, β Onext]



Edge Swapping

Fig. 14. The effect of Swap [*e*].




```
PROCEDURE Swap[e]:
  a ← e.Oprev;
  b ← e.Sym.Oprev;
  Splice[e, a]; Splice[e.Sym, b];
  Splice[e, a.Lnext]; Splice[e.Sym, b.Lnext];
  e.Org ← a.Dest; e.Dest ← b.Dest
END Swap.
```

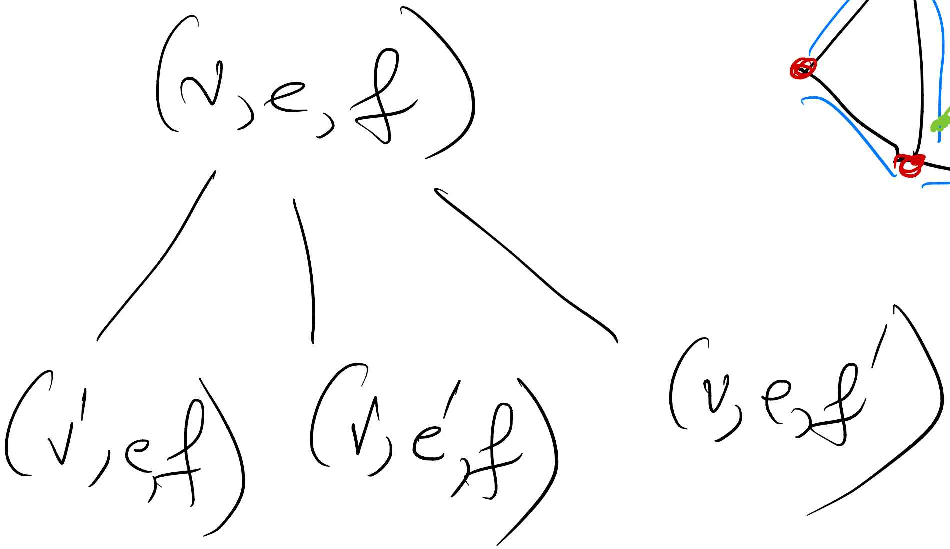
Quad-Edge

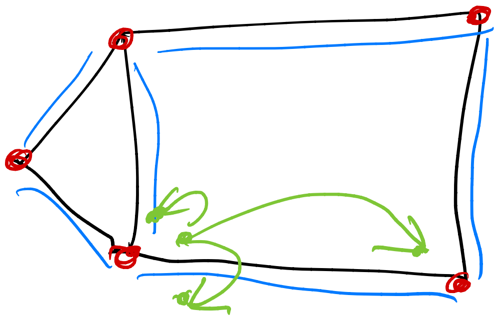
Half-Edge

Ordering Info

Cell-tuple Data Structure

2D

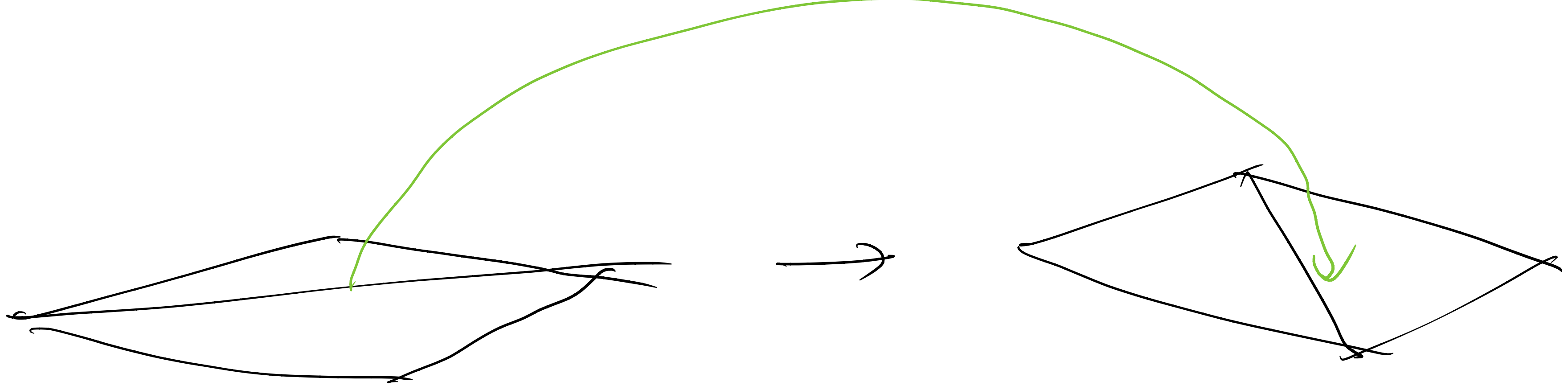




(v, e, f)

v is a vertex of e

e is an edge of f



That’s All