Haken's Algorithm

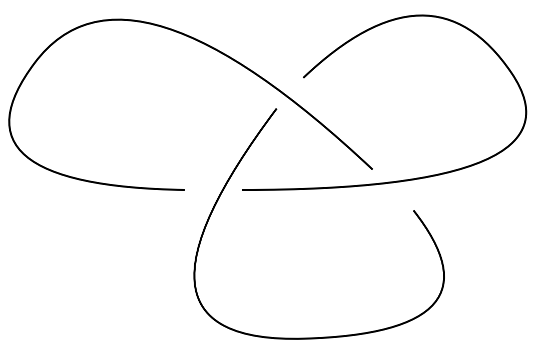
The first successful algorithm. Took about 50 years to find.
(Dehn 1910 to Haken 1961)

Theorem (Haken) There is an algorithmic procedure to

1. Recognize the Unknot
2. Classify knots
3. Compute the genus of a knot
4. Determine if a link is split

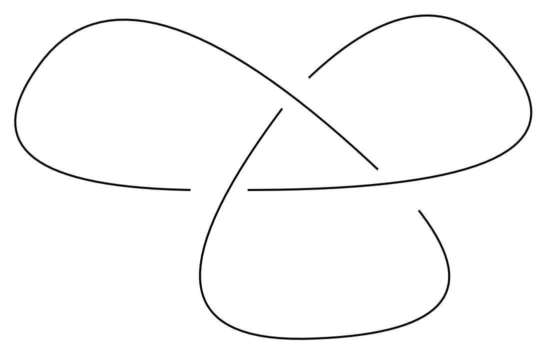
Haken's algorithm for unknotting searches for a disk in 3-space whose boundary is the knot. It is based on 3-manifold theory.

Idea behind Haken's Algorithm



Is this the unknot?

Idea behind Haken's Algorithm



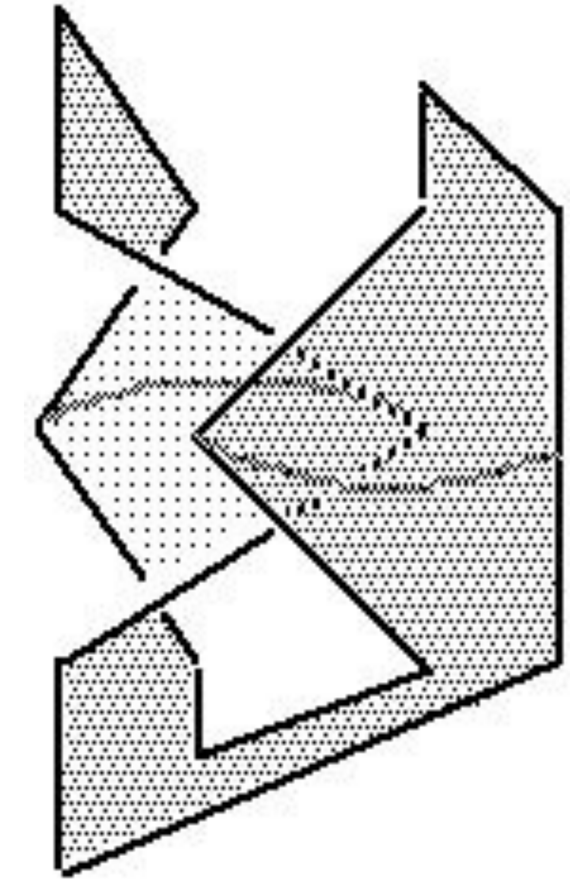
Is this the unknot?

Haken's algorithm is based on the following fact:
A knot is trivial if and only if it is the boundary of a disk in R^3.

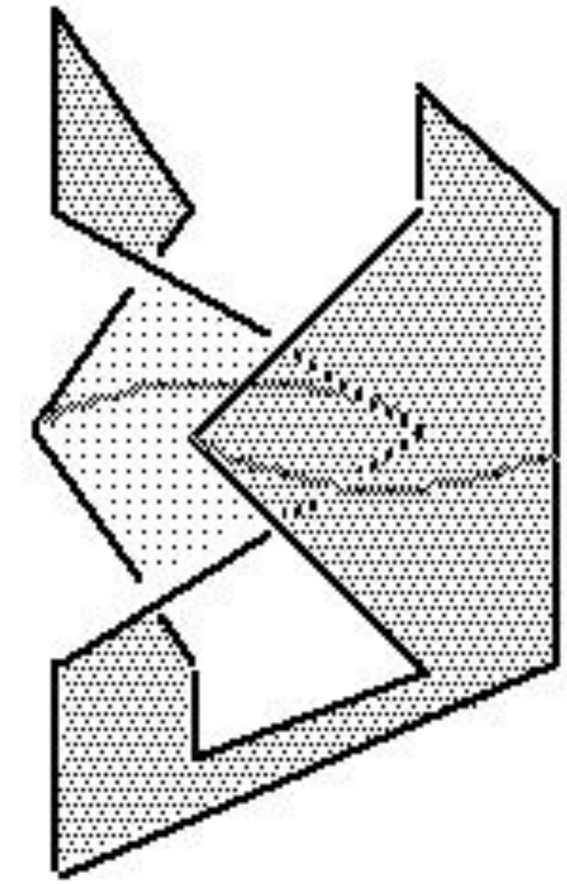
The algorithm is inherently based on 3-manifold topology (as opposed to diagrams or combinatorics).

Idea behind Haken's Algorithm

A knot is trivial if and only if it is the boundary of a disk in R^3.
So look for a disk spanning the knot.

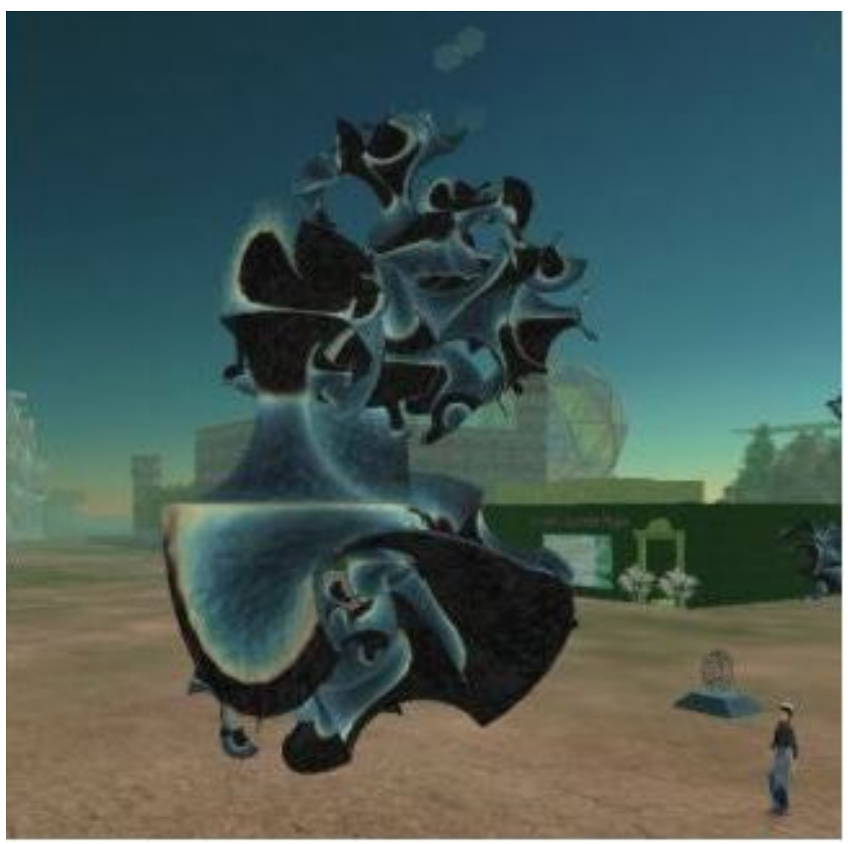


Idea behind Haken's Algorithm



A knot is trivial if and only if it is the boundary of a disk in R^3.
So look for a disk spanning the knot.

But some disks in R^3 are very complicated and hard to describe.

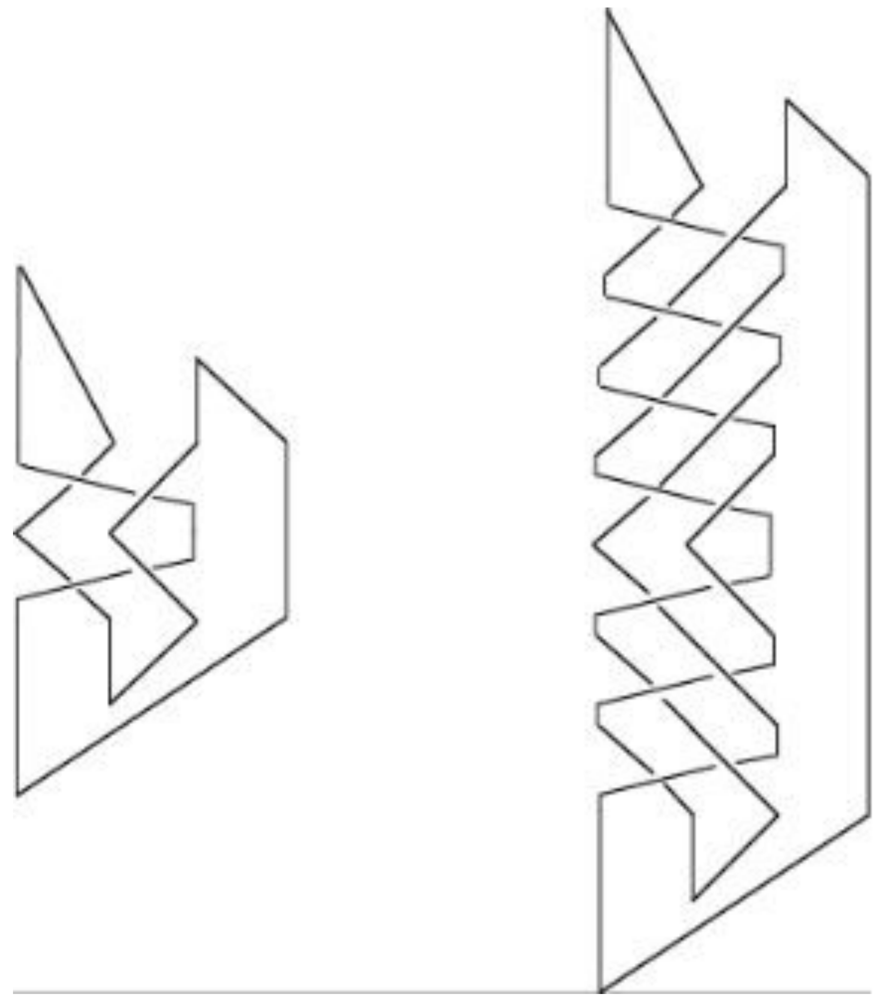


Question: Is there always a "simple" disk spanning an unknot K?

Complicated disks cannot be avoided

Theorem (H-Snoeyink-Thurston 2002)
There is a sequence of unknotted polygonal curves K_n such that K_n has less than 11n edges and any triangulation of a disk with boundary K_n contains at least 2^n triangular faces.

K_1



K_3

The number of triangles in any spanning disk for these unknots grows exponentially with the number of edges.

Surfaces in Topology

Many problems in 3-dimensional topology reduce to understanding surfaces in a 3-manifold.

Examples

A curve in a manifold is unknotted if it is the boundary of an **embedded disk**.

A 3-manifold with infinite fundamental group admits a hyperbolic structure if and only if it contains no **essential spheres or tori**.

Question

Can we systematically search for surfaces such as essential disks, spheres or tori?

Can we determine whether such surfaces exist?

Surfaces in Topology

Many problems in 3-dimensional topology reduce to understanding surfaces in a 3-manifold.

Examples

- A curve in a manifold is unknotted if it is the boundary of an **embedded disk**.
- A 3-manifold with infinite fundamental group admits a hyperbolic structure if and only if it contains no **essential spheres or tori**.

Question

Can we systematically search for surfaces such as essential disks, spheres or tori? Can we determine whether such surfaces exist?

Answer

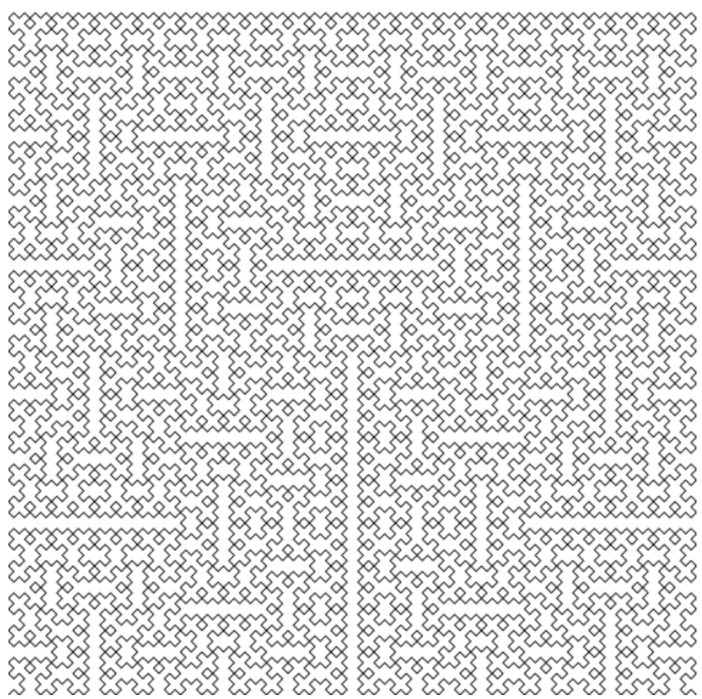
Generally yes, by using the theory of *Normal Surfaces*.

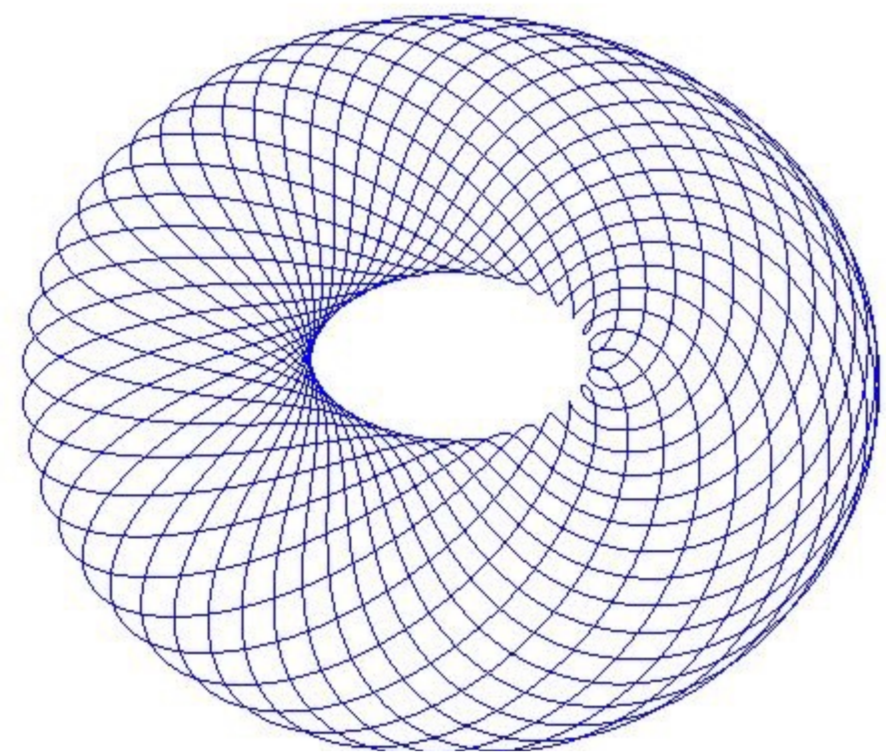
The algorithma to solve our problem have two basic steps:

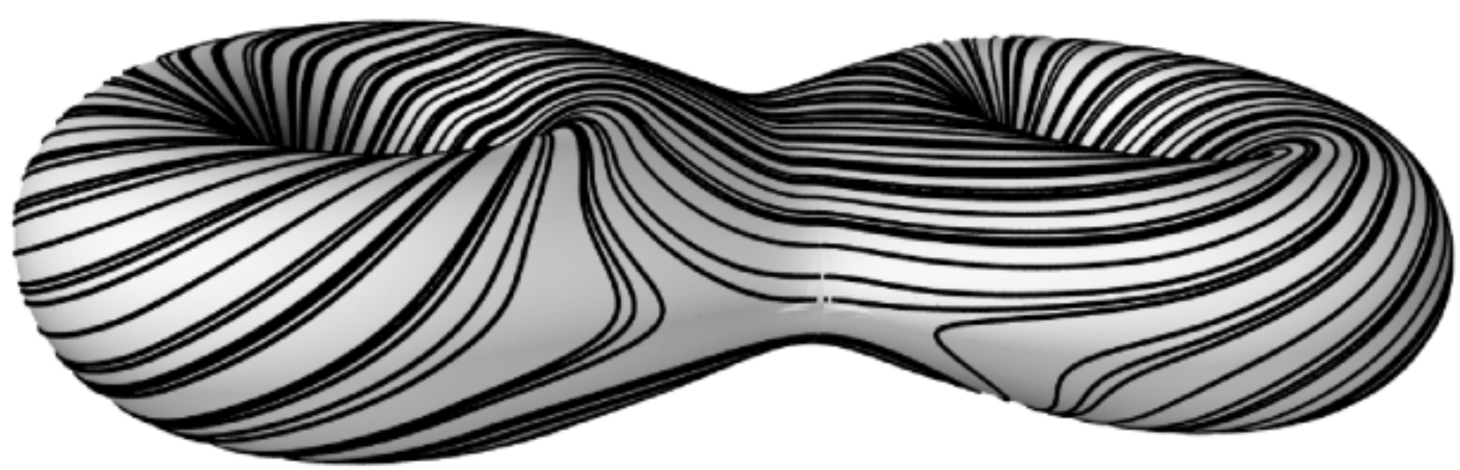
- Normalization
- Fundamentalization

Warmup - Curves in surfaces

Curves on a surface can be complicated and hard to describe.





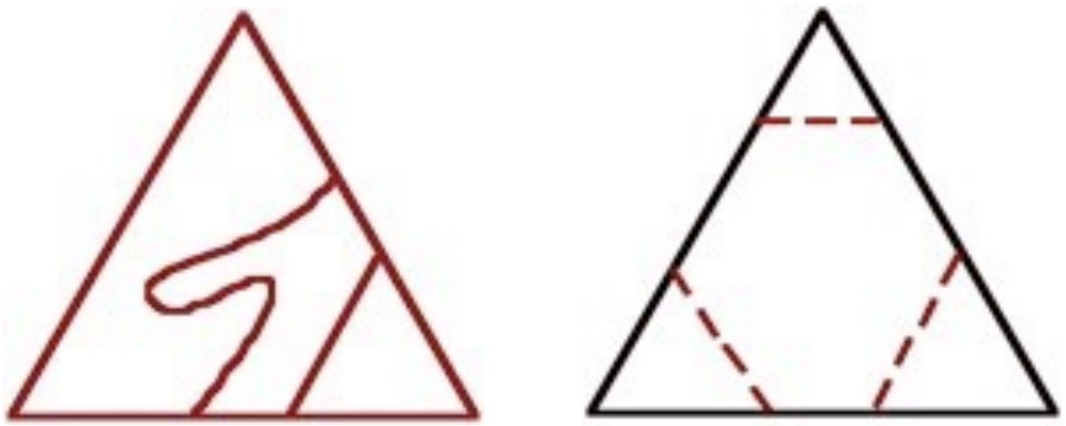


D. Calegari

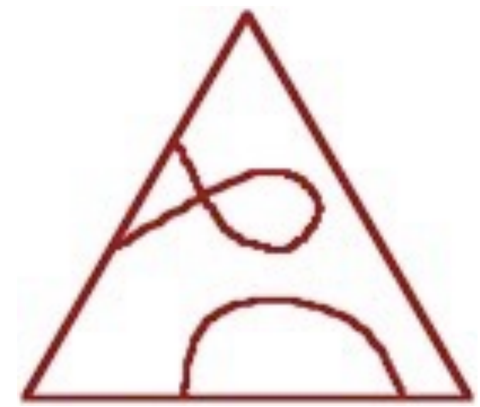
Normal curves give efficient descriptions of curves on a surface.
Exponentially more compact than listing the successive vertices of a polygon or a triangulation.

Normal curves in surfaces

Definition: An arc in a triangle is *elementary* or *normal* if it is embedded and its two endpoints lie on distinct edges of the triangle.

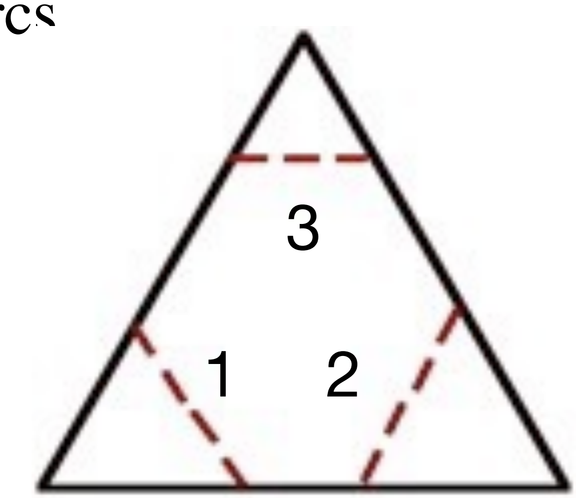


Elementary arcs

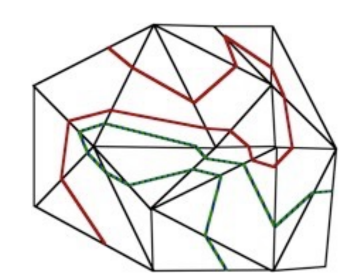


Non-elementary arcs

There are three types of elementary arcs in a triangle (up to an isotopy preserving the edges of the triangle.)



Definition: A curve on a triangulated surface is ***normal*** if it intersects each triangle in a disjoint union of elementary arcs.



The idea of normal curves and surfaces originated with Kneser (1930).

Normal curves in surfaces

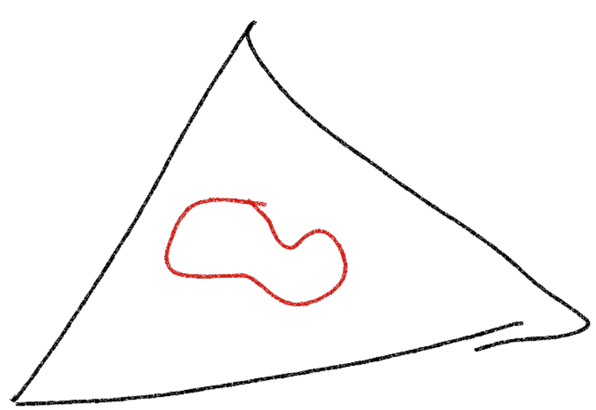
Fix a triangulation on a surface.
Then take any curve.
The curve can be made normal. We call this **normalization**.

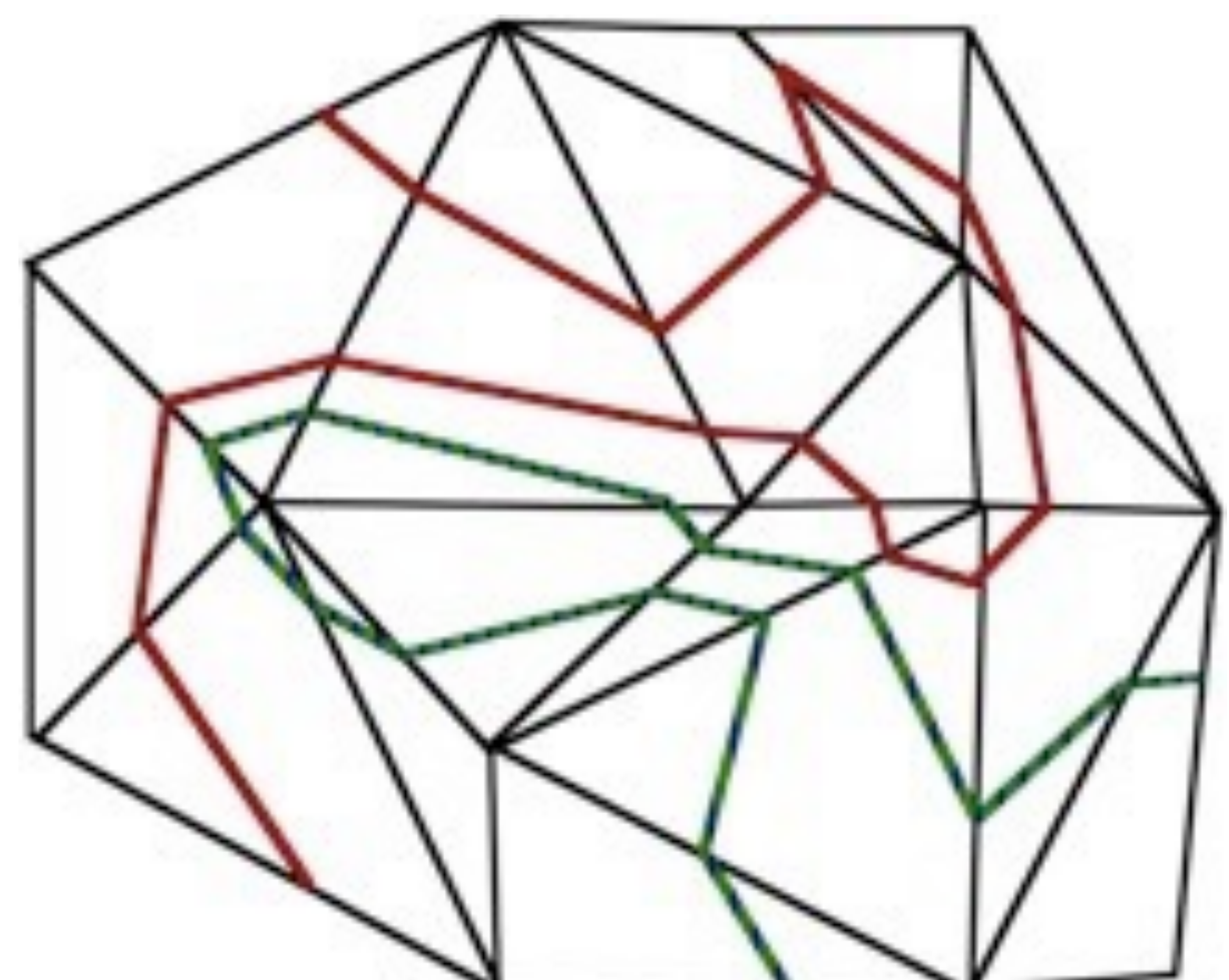
Theorem An embedded curve on a triangulated surface can be isotoped so that each component is either

1. normal

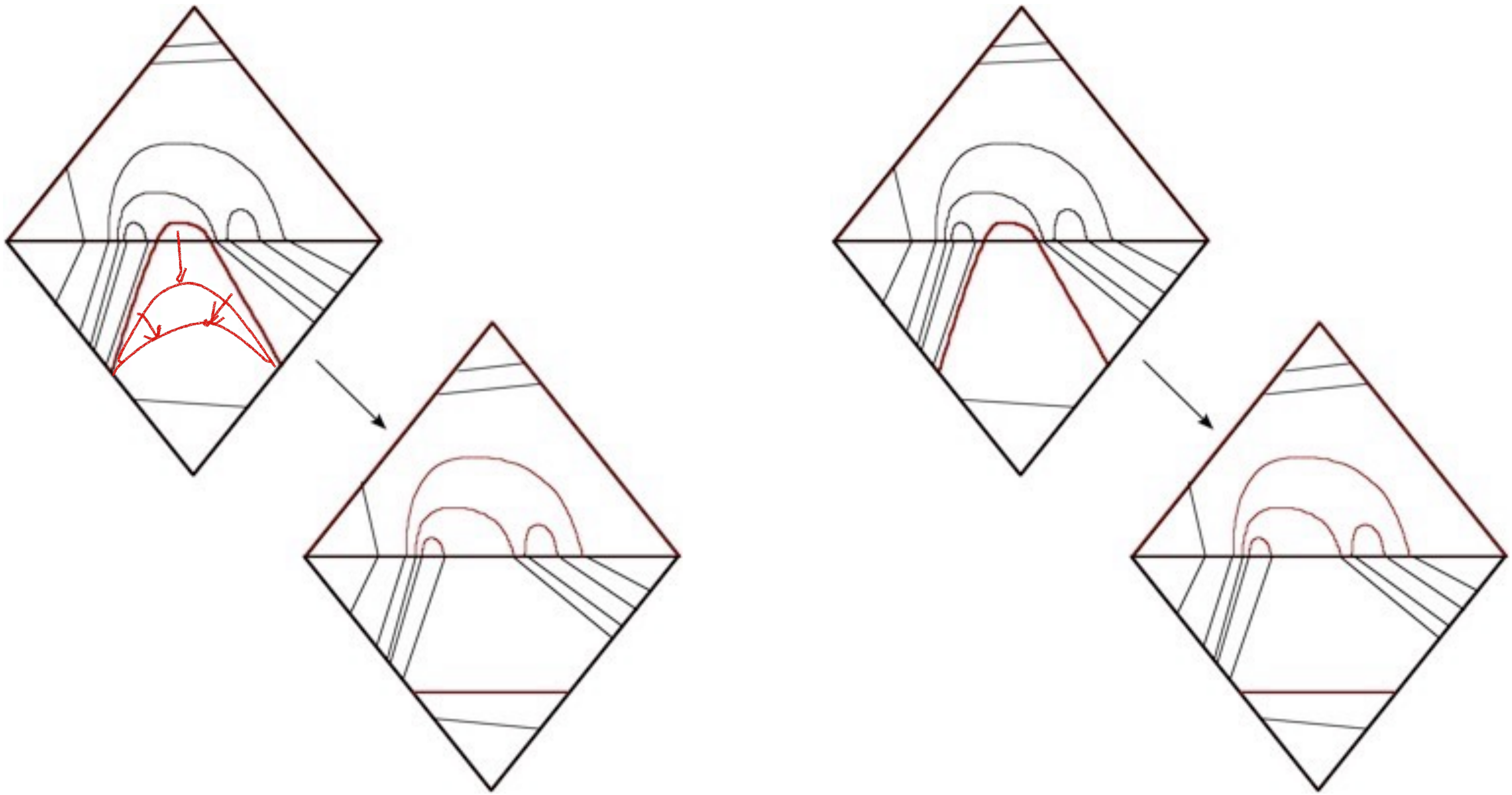
   or

2. contained in the interior of a triangle.





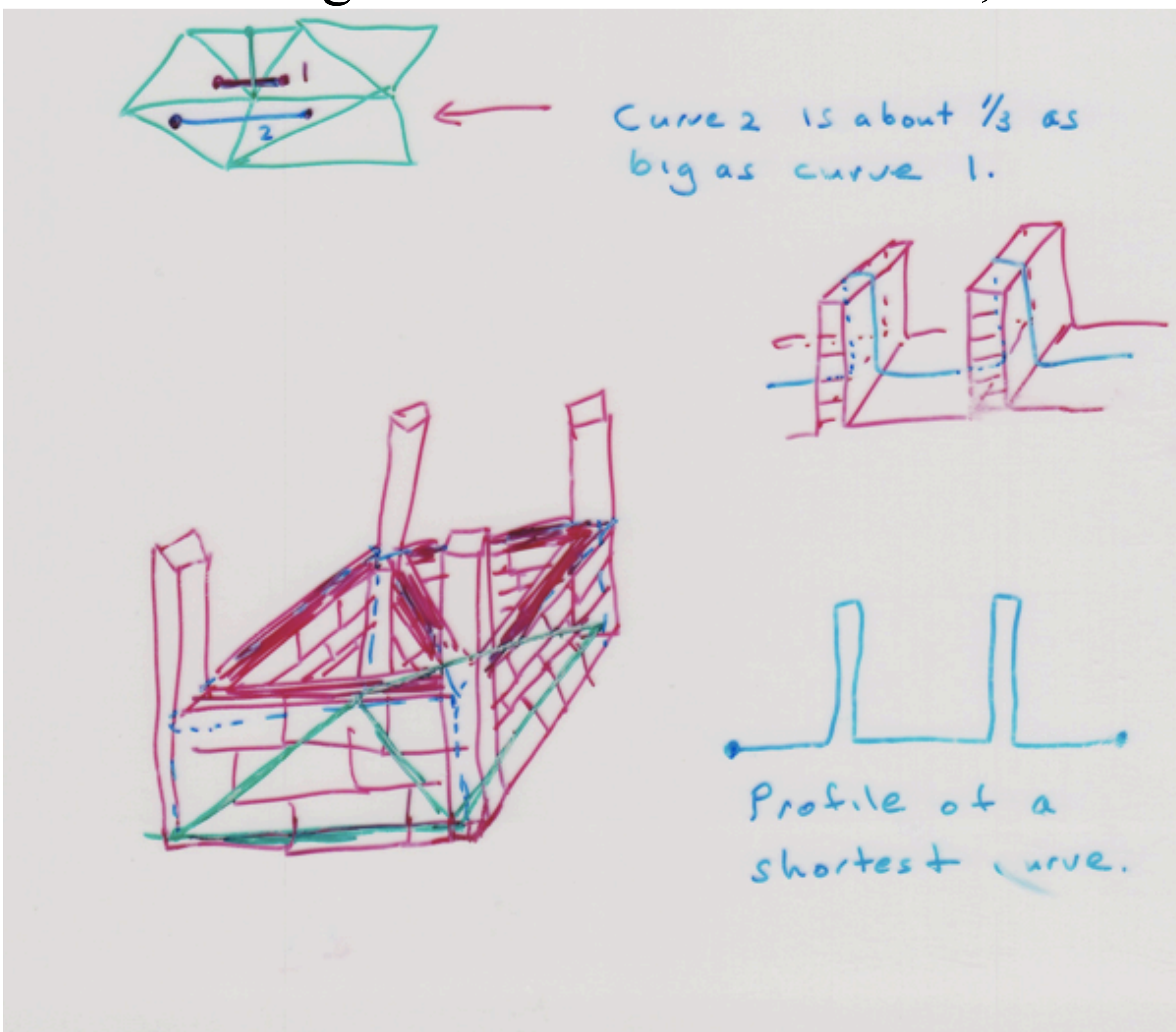
Normalizing a curve on a surface



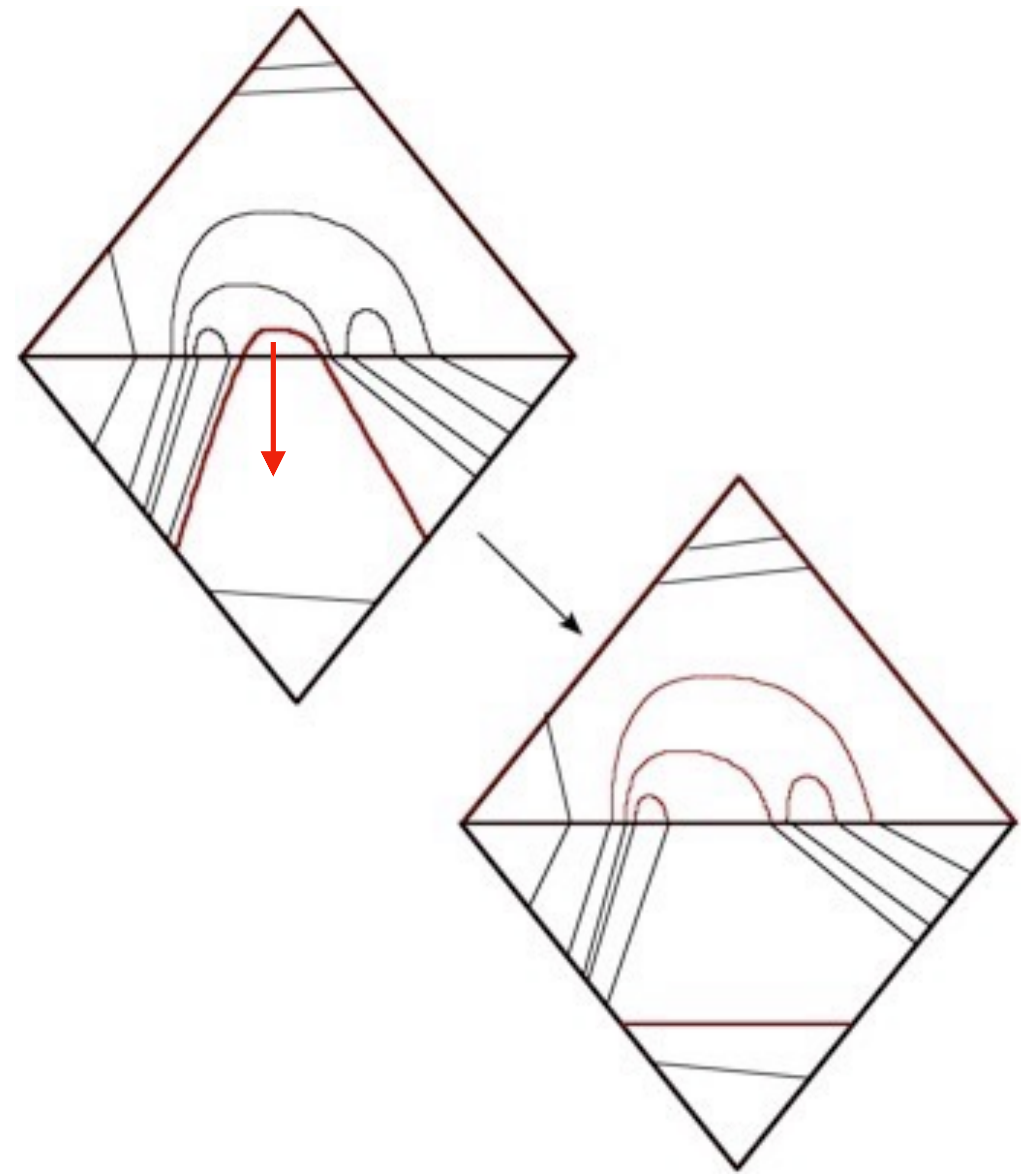
The *weight* of a curve is the number of times the curve meets an edge.
Normalizing decreases the weight of a curve.

Normalizing a curve on a surface

The *weight* of a curve is the number of times the curve meets an edge. We can think of weight as giving a crude measure of length, with a surface metric that is concentrated along the edges of a triangulation. As we traverse a curve on the surface, weight counts how many times we climb over walls. Normal curves climb over walls a minimal number of times. There are even higher towers at the vertices, which are avoided.

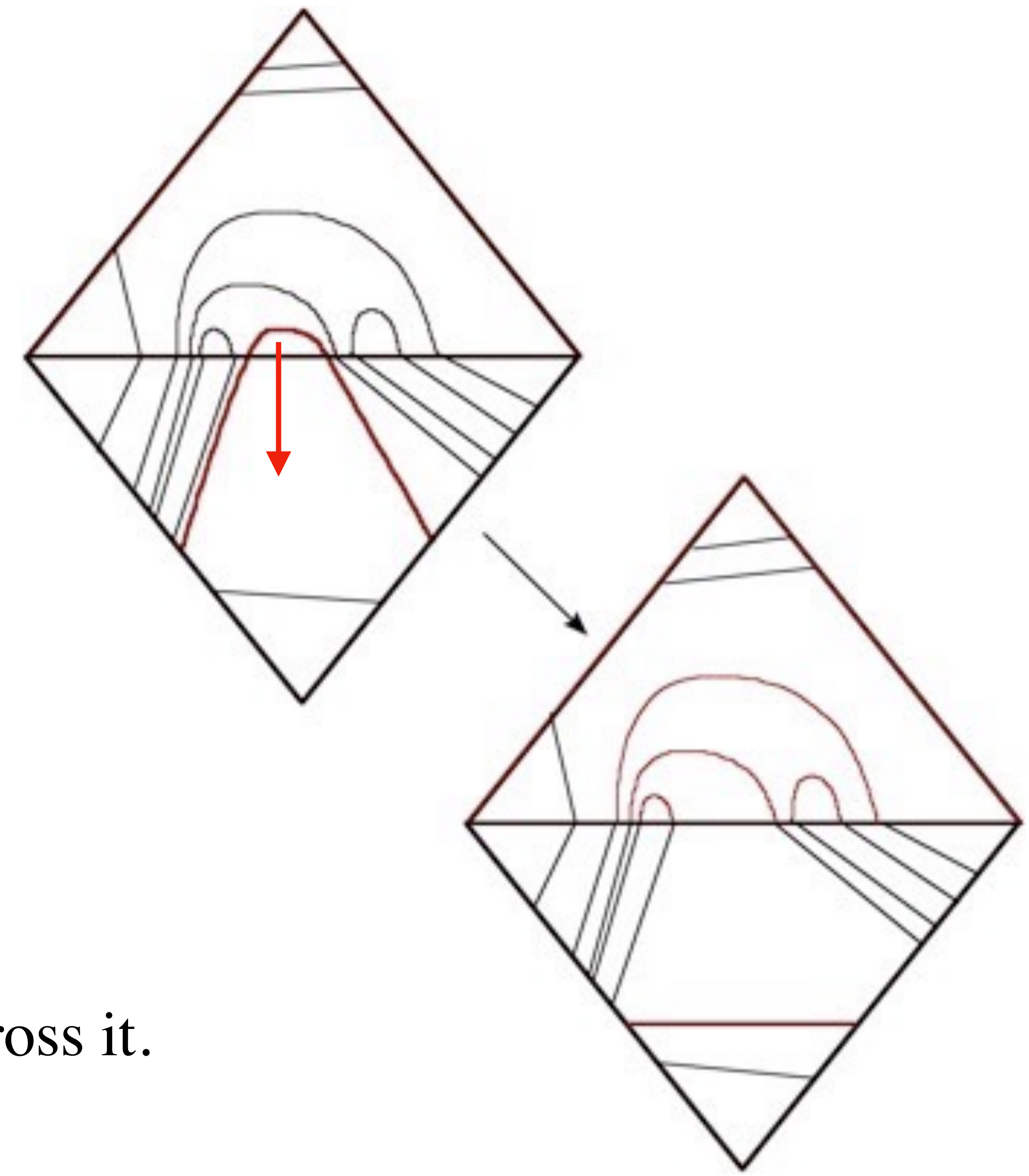


Normalizing a curve on a surface



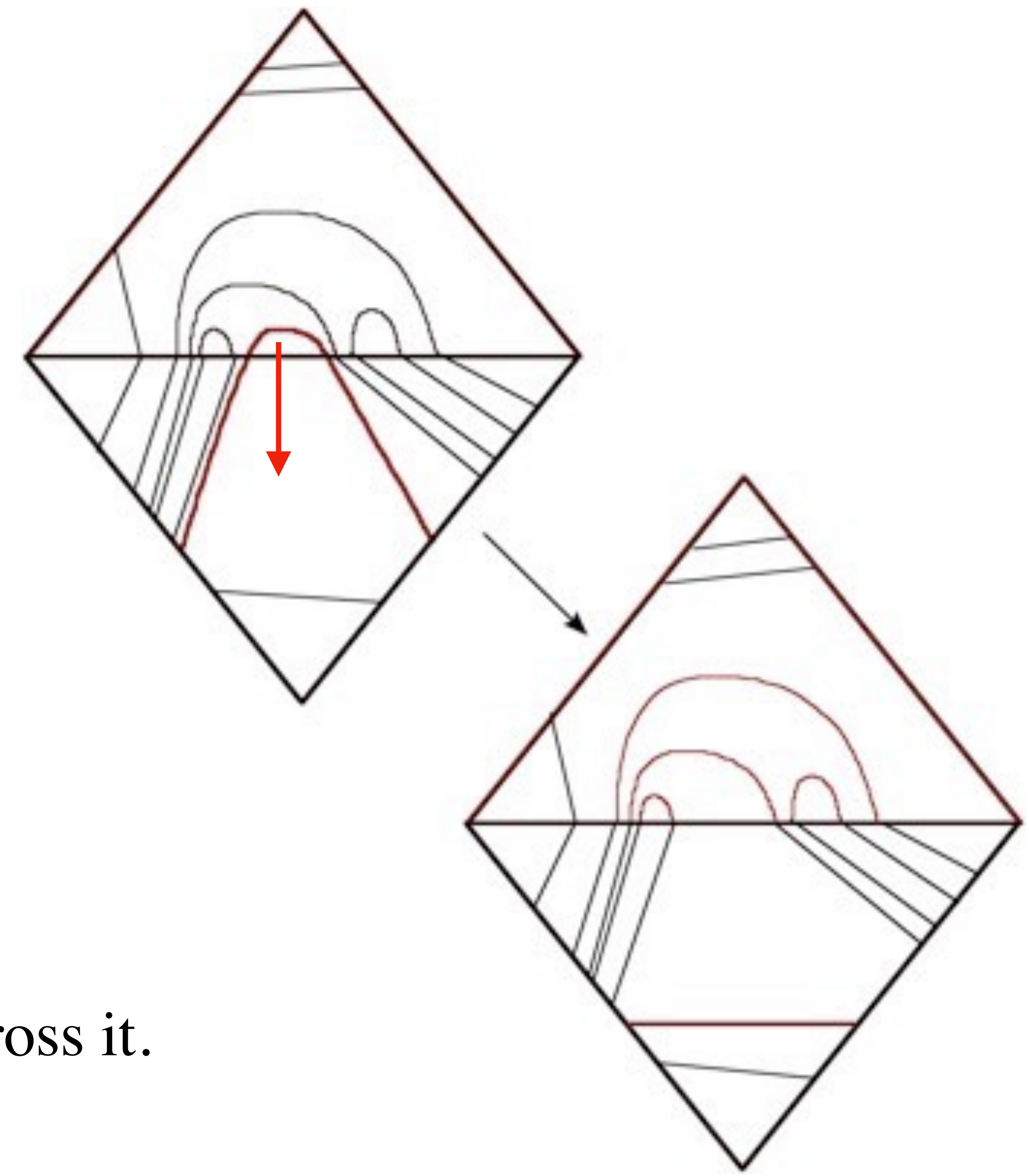
Find a triangle where the curve is not normal.

Normalizing a curve on a surface



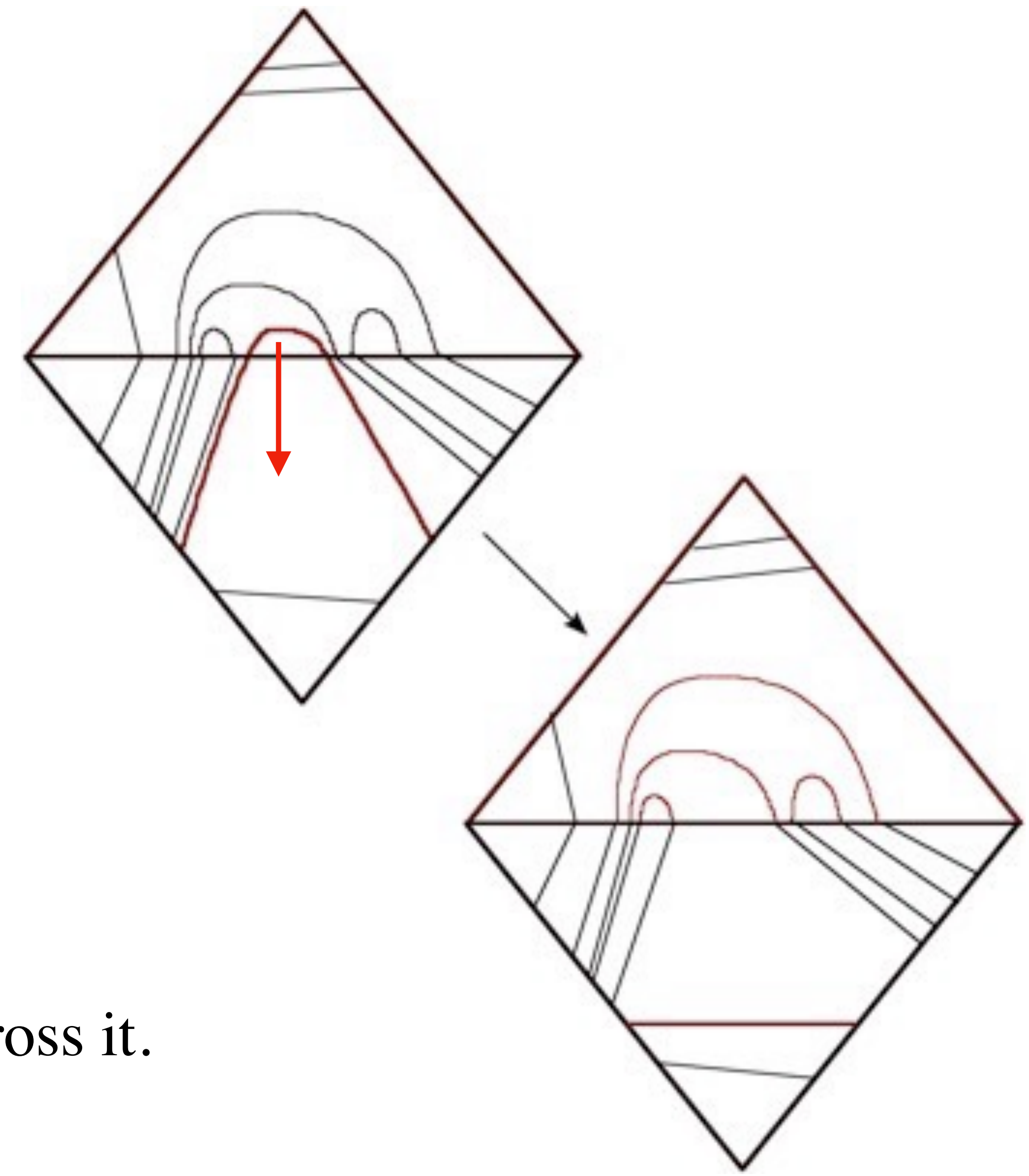
Find a triangle where the curve is not normal.
Find an outermost disk and isotop the curve across it.
This move keeps the curve embedded.

Normalizing a curve on a surface



Find a triangle where the curve is not normal.
Find an outermost disk and isotop the curve across it.
This move keeps the curve embedded.
This move reduces the weight of the curve by two.
Induction completes the proof, giving an isotopic normal curve.
The argument also works for disconnected curves, keeping them disjoint as they normalize.

Normalizing a curve on a surface



Find a triangle where the curve is not normal.
Find an outermost disk and isotop the curve across it.
This move keeps the curve embedded.
This move reduces the weight of the curve by two.
Induction completes the proof, giving an isotopic normal curve.
The argument also works for disconnected curves, keeping them disjoint as they normalize.

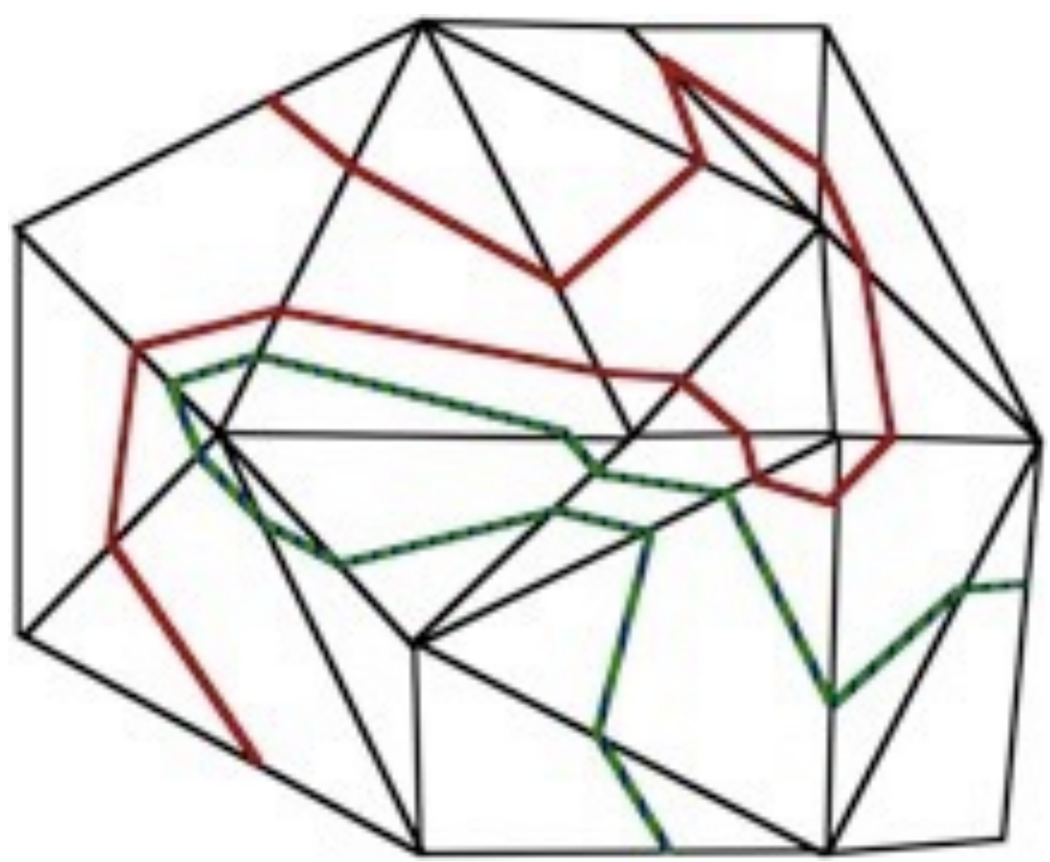
Lemma A curve of least weight in its isotopy class either lies inside a triangle or is normal.

Theorem An embedded curve on a triangulated surface can be isotoped so that each component is normal or contained in the interior of a triangle.

Normalization of curves on a surface

Fix a triangulation on a surface.
Then take any curve on the surface, connected or not.
The curve can be made normal.

Theorem An embedded curve on a triangulated surface can be isotoped so that each component is normal or contained in the interior of a triangle.



Haken's contribution: From Topology to Algebra

In the mid 1950's, Haken contributed a breakthrough idea.

The normal surfaces of topology and geometry are in one-to-one correspondence with certain algebraic objects.

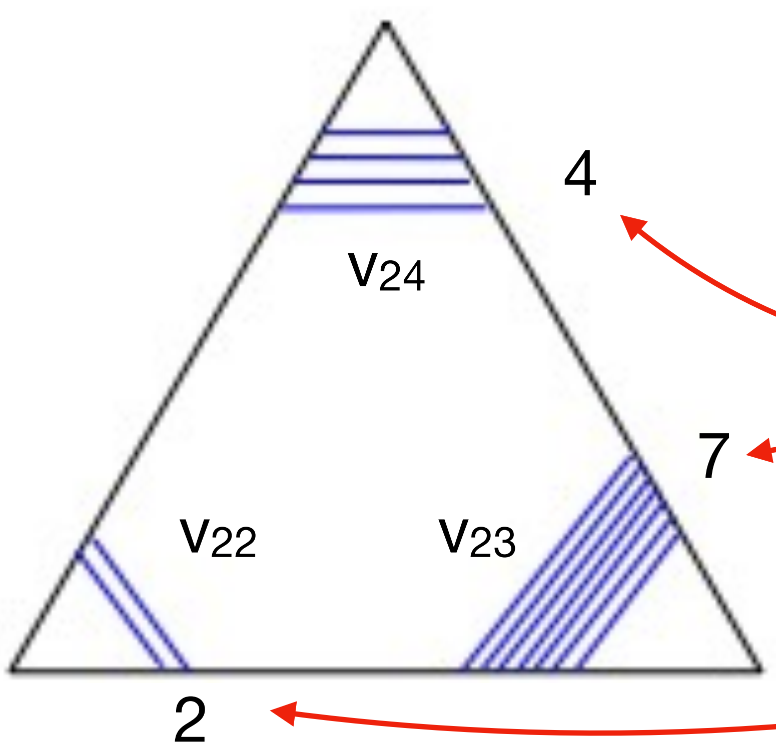
This correspondence opens topological problems to extremely efficient algorithmic analysis.

Normal curves and integer vectors

Haken's Idea:

Normal curves are in one-to-one correspondence with certain integer vectors.

Normal curves connect topology and algebra.



A normal curve determines a vector with integer entries, three for each 2-simplex.

$$\mathbf{V} = \langle v_1,\ v_2,\ v_3,\ v_4,\ v_5, \ldots v_{22},\ v_{23},\ v_{24}, \ldots\ v_{3t} \rangle$$

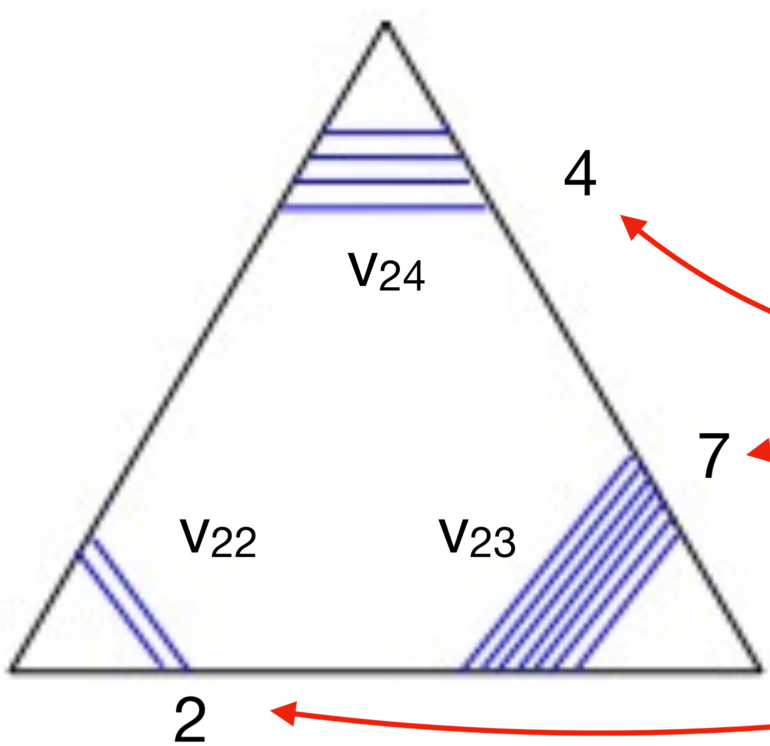
$$\mathbf{V} = \langle \ldots, 2, 7, 4, \ldots \rangle$$

Normal curves and integer vectors

Haken's Idea:

Normal curves are in one-to-one correspondence with certain integer vectors.

Normal curves connect topology and algebra.



A normal curve determines a vector with integer entries, three for each 2-simplex.

$\mathbf{V} = \langle v_1,\ v_2,\ v_3,\ v_4,\ v_5, \ldots v_{22},\ v_{23},\ v_{24}, \ldots\ v_{3t} \rangle$

$\mathbf{V} = \langle \ldots., 2, 7, 4, \ldots \rangle$

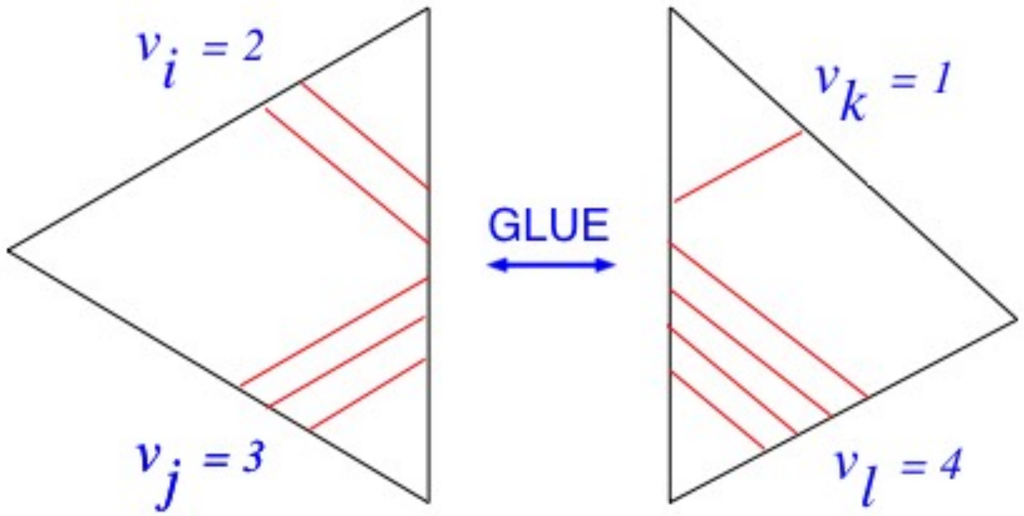
But - not all integer vectors give normal curves.

Which ones do?

Normal curves and integer vectors

Not all integer vectors give normal curves. Which ones do?

The curves need to match up along adjacent triangles.

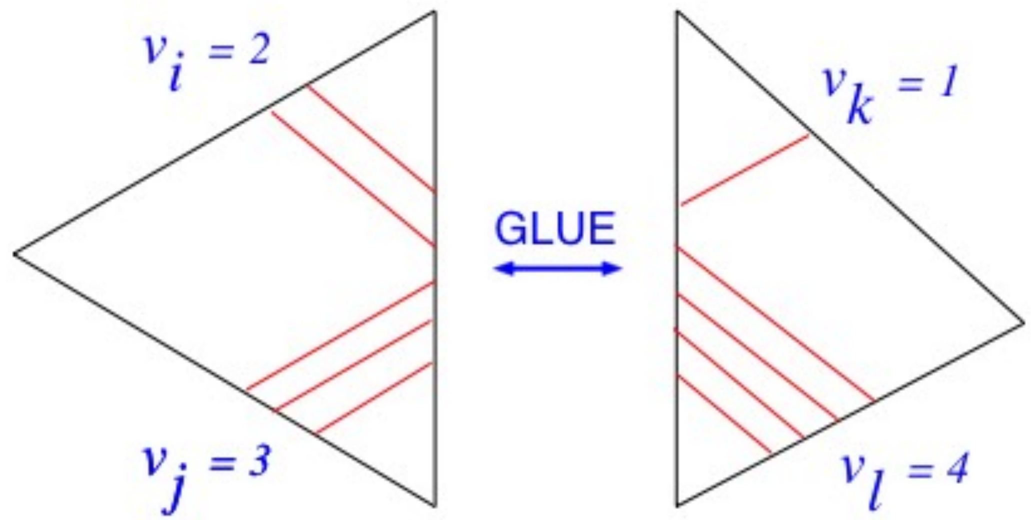


$$v_i + v_j = v_k + v_l$$

Normal curves and integer vectors

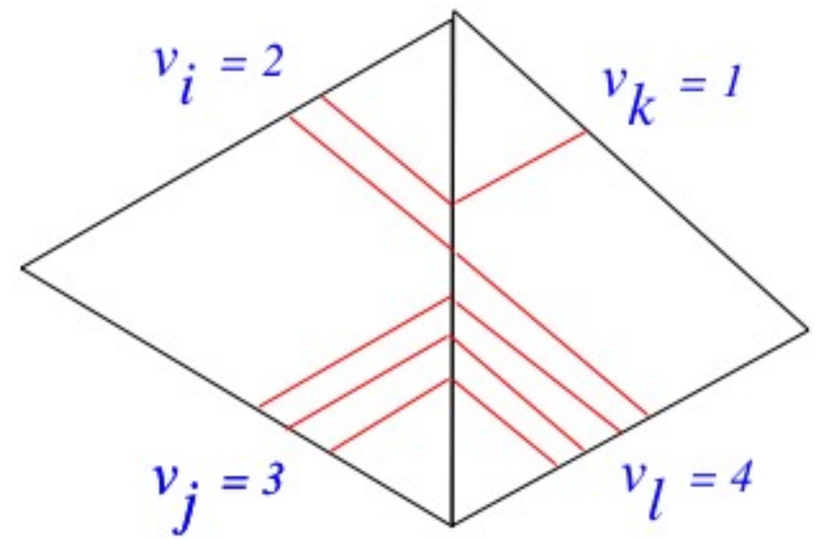
Not all integer vectors give normal curves. Which ones do?

The curves need to match up along adjacent triangles.



$v_i + v_j = v_k + v_l$

2 + 3 = 1 + 4

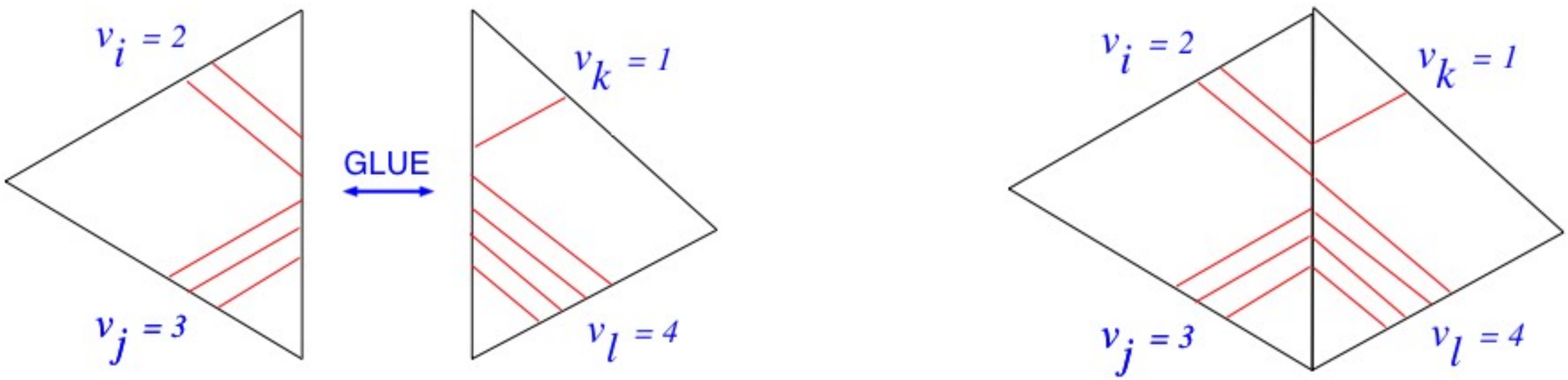


$v_i + v_j = v_k + v_l$

Normal curves and integer vectors

Not all integer vectors give normal curves. Which ones do?

The curves need to match up along adjacent triangles.



$v_i + v_j = v_k + v_l$ **2 + 3 = 1 + 4** $v_i + v_j = v_k + v_l$

The curves match up along adjacent triangles if the vector coordinates v_i satisfy equations of the form

$3t$ variables in $\mathbf{Z}_+$	v_i
$3t/2$ equations	$v_i + v_j = v_k + v_l$
$3t$ inequalities	$v_i \geq 0$

These are called the **Matching Equations**.

Their integer solutions determine a curve.

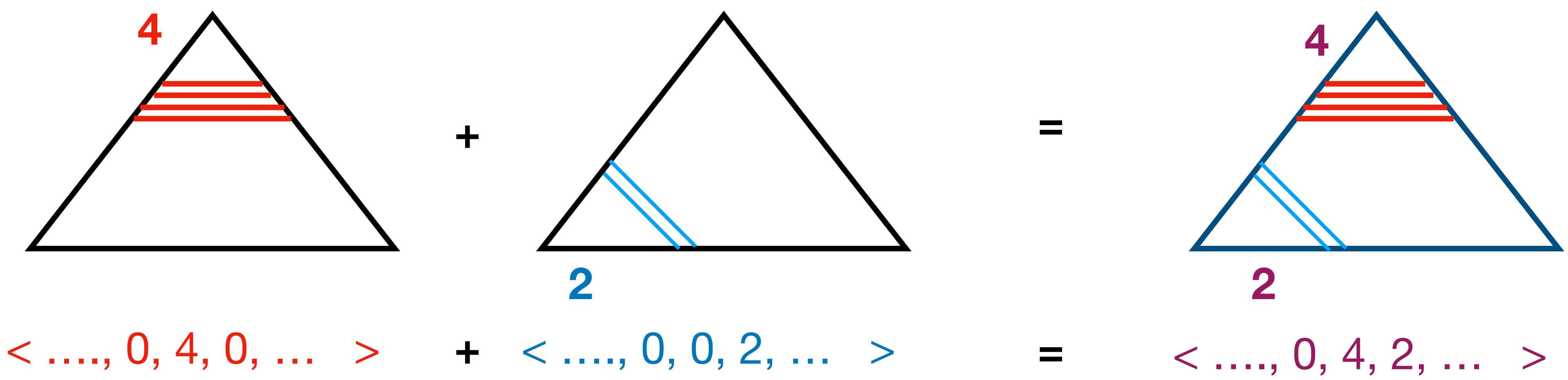
Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding “addition” of curves

Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding “addition” of curves

1. If two normal curves are disjoint, take their geometric sum to be their disjoint union. This curve addition agrees with vector addition.

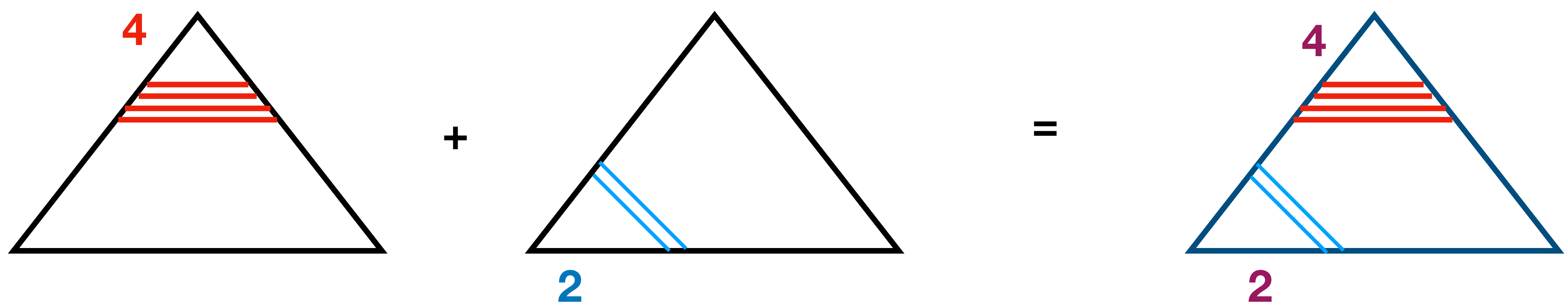


Haken sum for disjoint curves

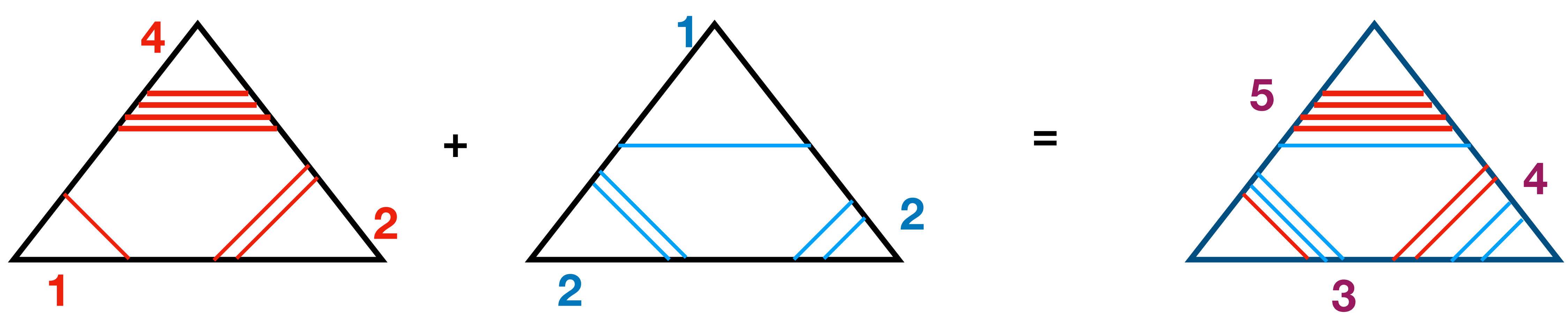
Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding “addition” of curves

1. If two normal curves are disjoint, take their geometric sum to be their disjoint union. This curve addition agrees with vector addition.



< …., 0, 4, 0, … > + < …., 0, 0, 2, … > = < …., 0, 4, 2, … >



< …., 2, 4, 1, … > + < …., 2, 1, 2, … > = < …., 4, 5, 3, … >

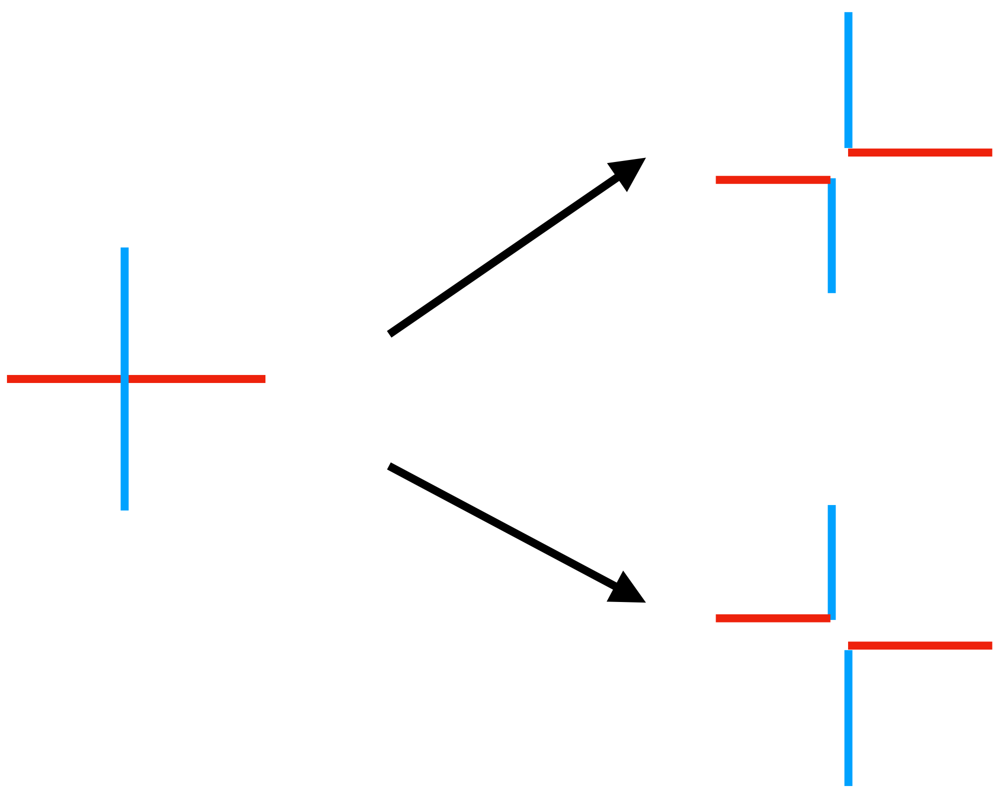
Haken sum for disjoint curves

Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding "addition" of curves

2. If two normal curves intersect, form their "normal sum" or "Haken sum". This curve addition agrees with vector addition.

Curve addition for intersecting curves is done using "cut and paste."

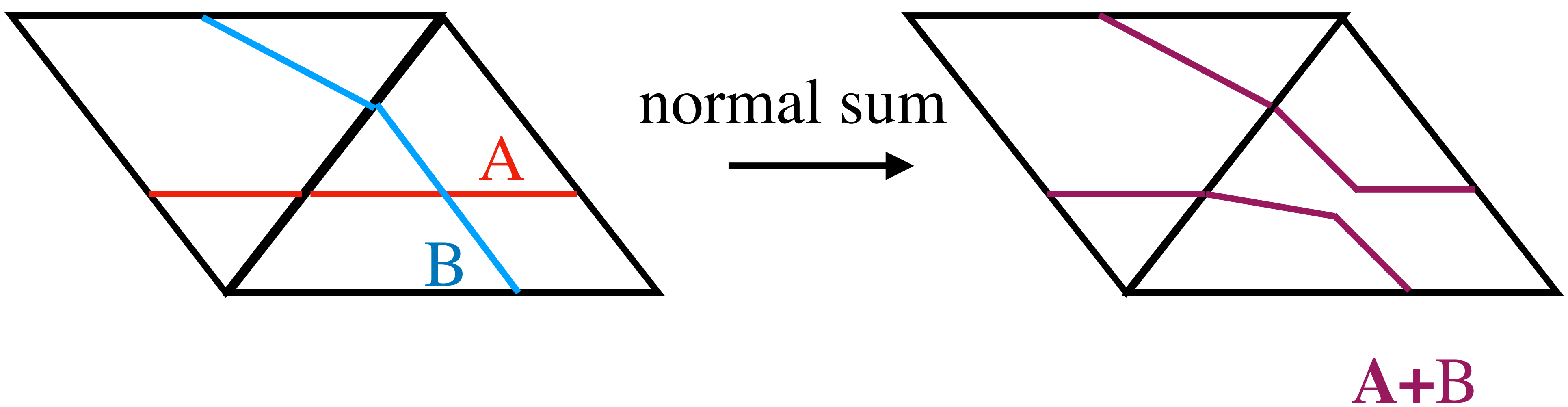


There are two choices for resolving a crossing. But only one results in normal curves. That is the regular sum that we will use.
Note: The orientation of the curve plays no role.

Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding "addition" of curves

2. If two normal curves intersect, form their "normal sum" or "Haken sum". This curve addition agrees with vector addition.

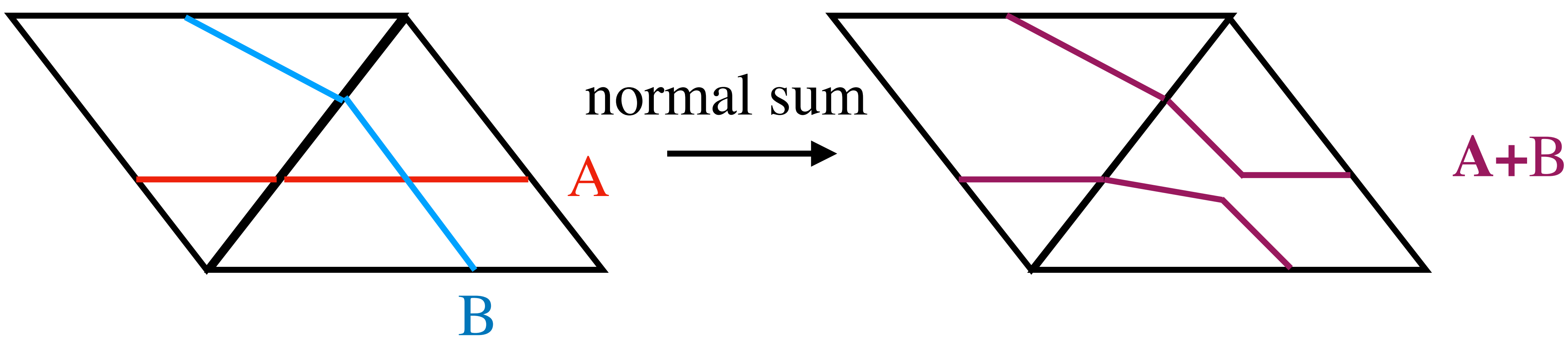


This is also called "regular" sum
When we cut and paste curves in this way, normal curves A and B give rise to a new normal curve A+B.

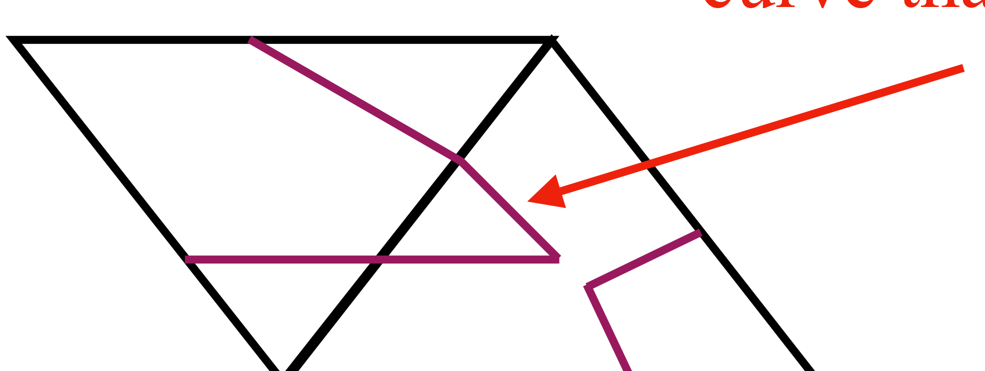
Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding "addition" of curves

2. If two normal curves intersect, form their "normal sum" or "Haken sum". This curve addition agrees with vector addition.



This irregular sum does not give a normal curve. This cut and paste choice leads to a curve that hits an edge twice.

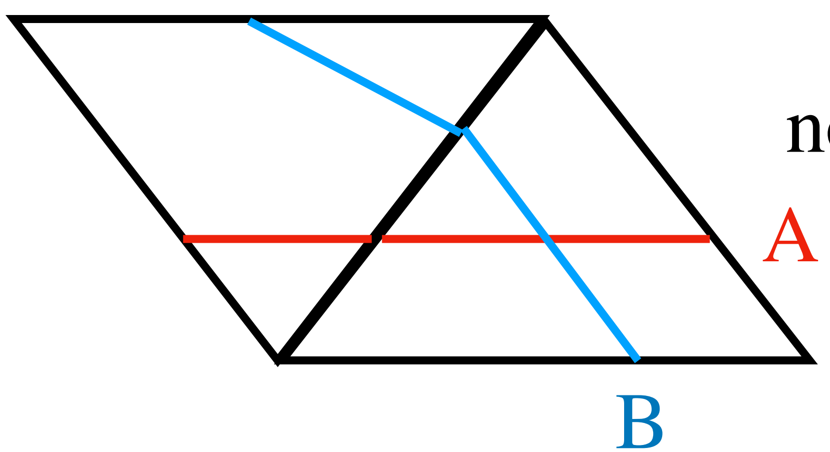


When adding curves algebraically, we don't cut and paste this way. But if we do, we can reduce weight!

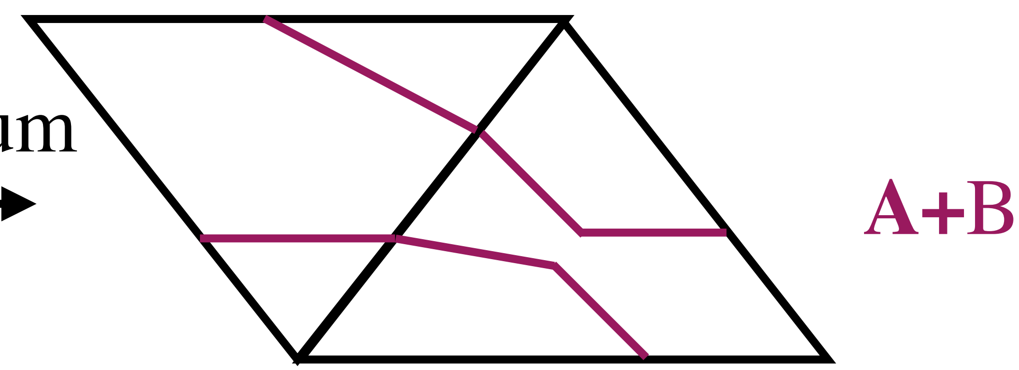
Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding "addition" of curves

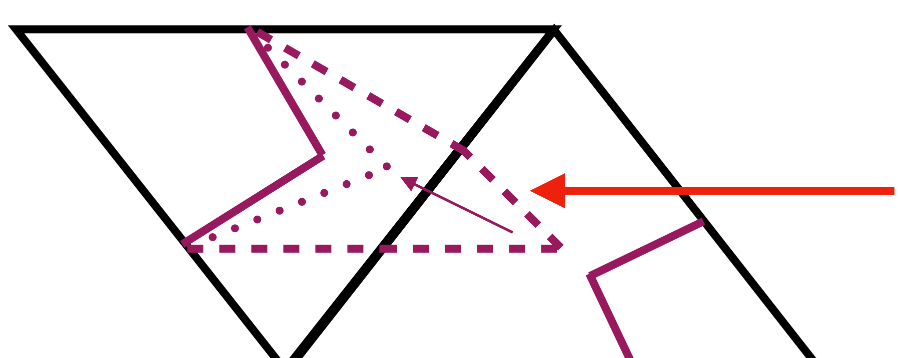
2. If two normal curves intersect, form their "normal sum" or "Haken sum". This curve addition agrees with vector addition.



normal sum



irregular sum - does not give a normal curve,
does not correspond to algebra.

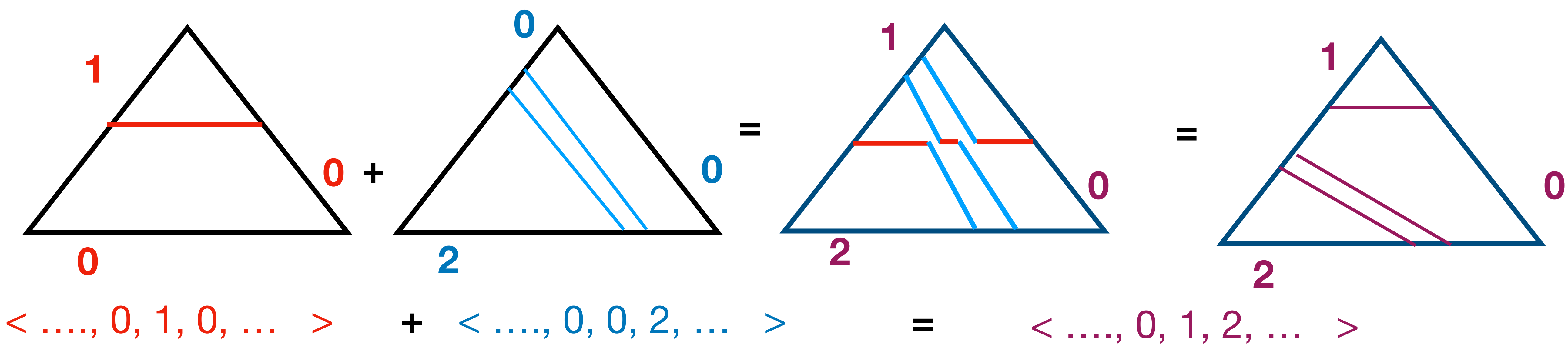


When adding curves algebraically, we don't cut and paste this way. But if we do, we can reduce weight!

Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding "addition" of curves

2. If two normal curves intersect, form their "normal sum" or "Haken sum". This curve addition agrees with vector addition.

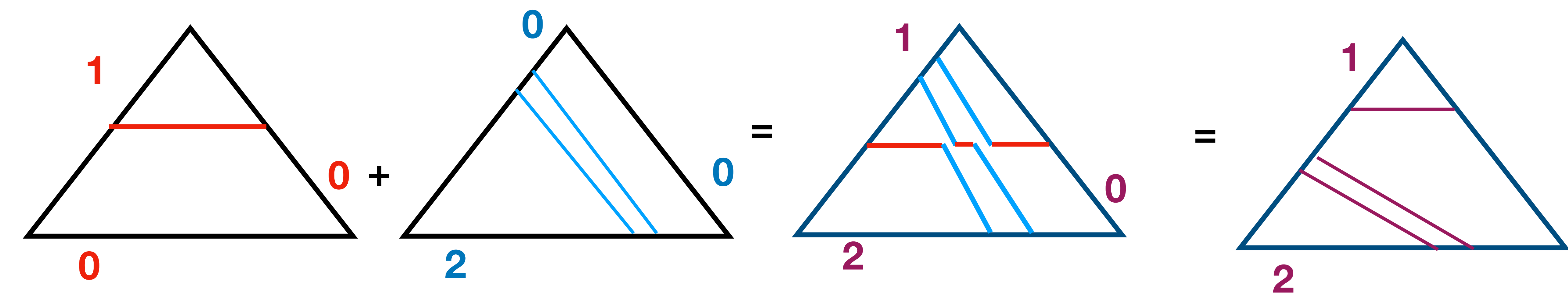


Haken sum for intersecting curves

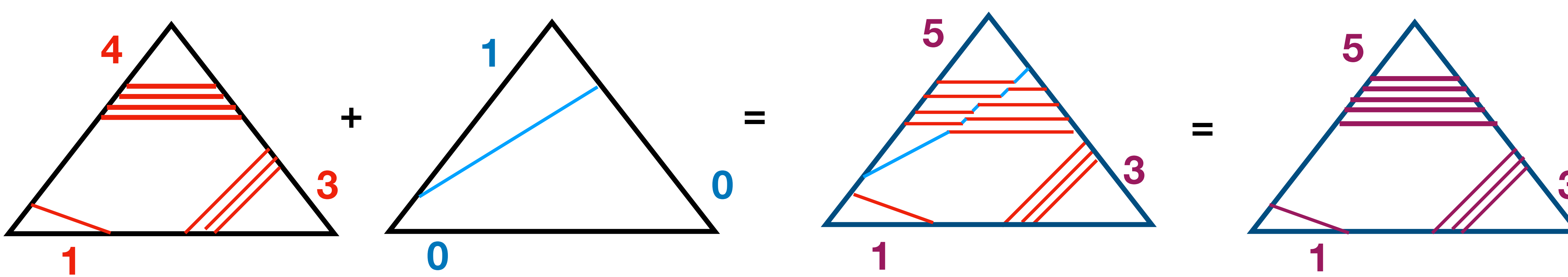
Adding Curves and Adding Vectors

Adding Curves: We can add normal vectors using vector addition. An amazing fact is that there is a natural corresponding “addition” of curves

2. If two normal curves intersect, form their “normal sum” or “Haken sum”. This curve addition agrees with vector addition.



< …., 0, 1, 0, … > + < …., 0, 0, 2, … > = < …., 0, 1, 2, … >



< …., 3, 4, 1, … > + < …., 0, 1, 0, … > = < …., 3, 5, 1, … >

Haken sum for intersecting curves

Matching Equations

We have reduced the problem of finding normal curves to the problem of finding solutions to the matching equations.
A topological problem can be translated into an algebraic problem.

Sample Topology Problem:

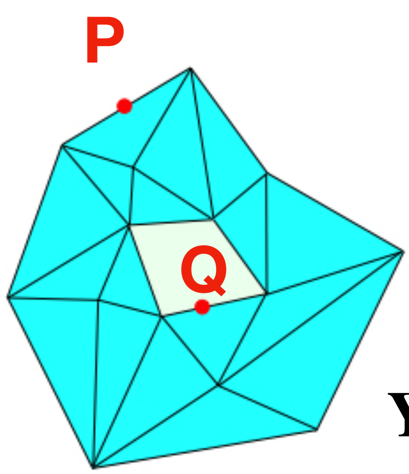
UNKNOTTING in Dimension 2.
Instance: A pair of points $P \cup Q$ in a surface F.
Question: Is the 1-sphere $P \cup Q$ unknotted?
(i.e. is $P \cup Q$ the boundary of a 1-disk)

This is equivalent to asking if P and Q lie in the same component of F.

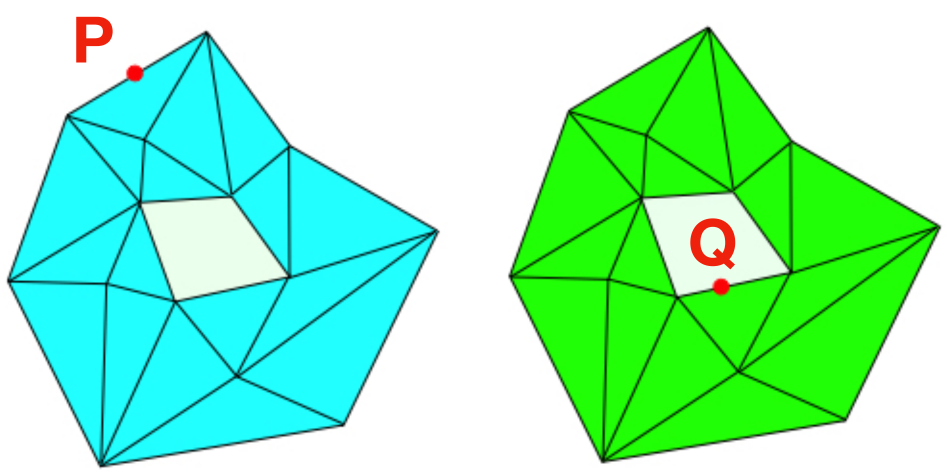
Matching Equations

We have reduced the problem of finding normal curves to the problem of finding solutions to the matching equations.
A topological problem has been translated into an algebraic problem.

Sample Problem: Is the 1-sphere P ∪ Q unknotted? (i.e the boundary of a 1-disk)
or: Is there a curve in *F* connecting boundary points P and Q?



Yes

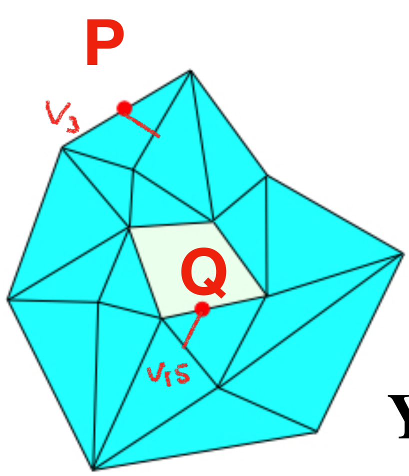


No

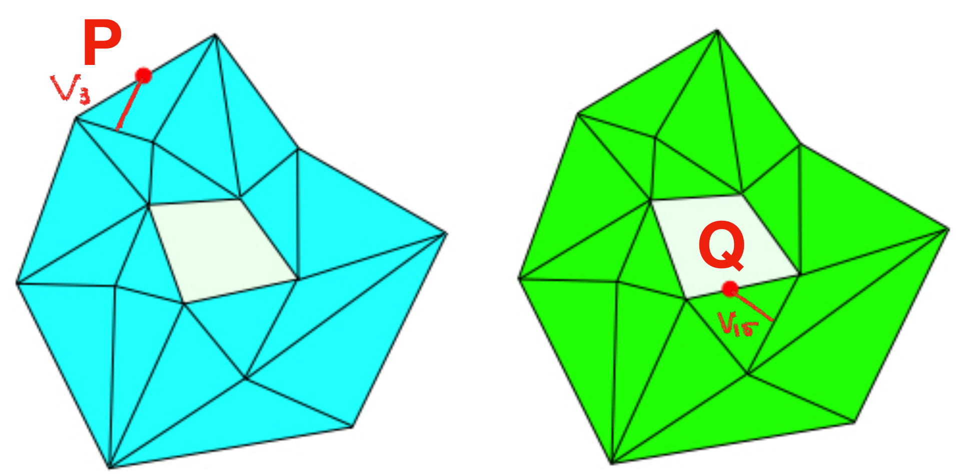
Matching Equations

We have reduced the problem of finding normal curves to the problem of finding solutions to the matching equations.
A topological problem has been translated into an algebraic problem.

Sample Problem: Is the 1-sphere $P \cup Q$ unknotted? (i.e the boundary of a 1-disk)
or: Is there a curve in *F* connecting boundary points P and Q?



Yes



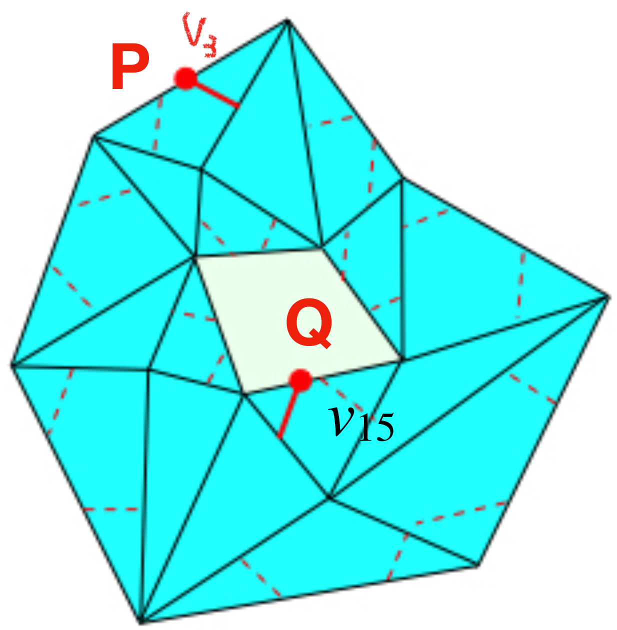
No

Equivalent Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations and with $v_3 = 1$ and $v_{15} = 1$, and all other variables meeting the boundary equal to zero?

$3t$ variables in $\mathbf{Z}_+$	v_i
$3t/2$ equations	$v_i + v_j = v_k + v_l$
$3t$ inequalities	$v_i \geq 0$

Normal Curves

Equivalent Algebraic Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations, with $v_3 = 1$ and $v_{15} = 1$, and with $v_k = 0$ for all other arcs meeting the boundary?

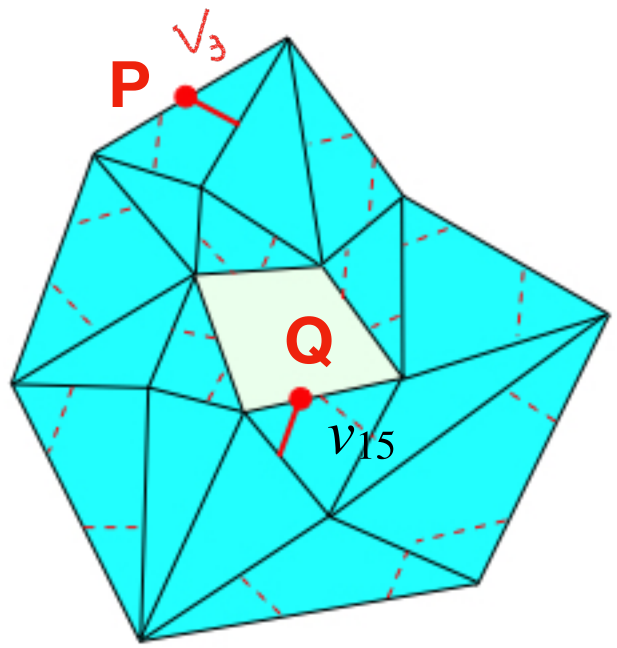


We are not allowed to use any of the dashed red arc types in constructing a normal curve. from P to Q. Set them equal to 0.

We do use the red arc types near P and near Q. Set them equal to 1.

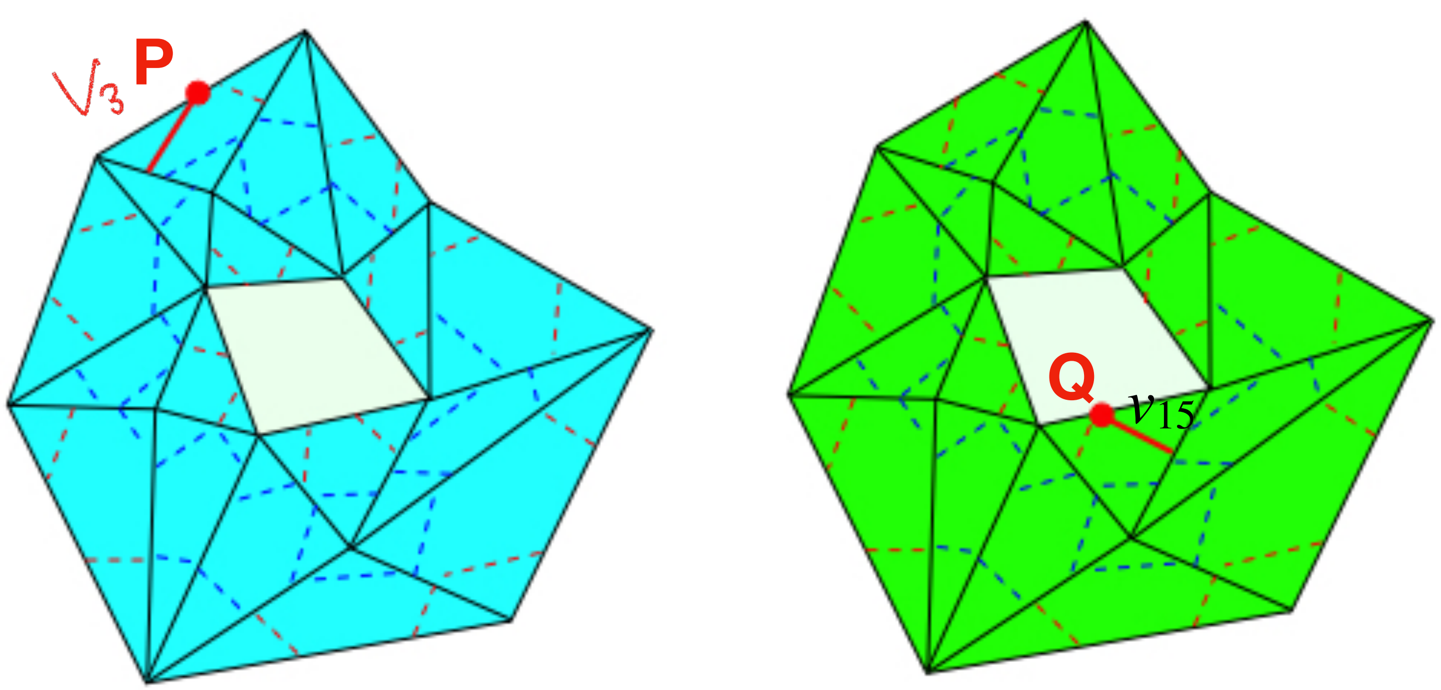
Normal Curves

Equivalent Algebraic Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations, with $v_3 = 1$ and $v_{15} = 1$, and with $v_k = 0$ for all other arcs meeting the boundary?



We are not allowed to use any of the dashed red arc types in constructing a normal curve. from P to Q. Set them equal to 0.

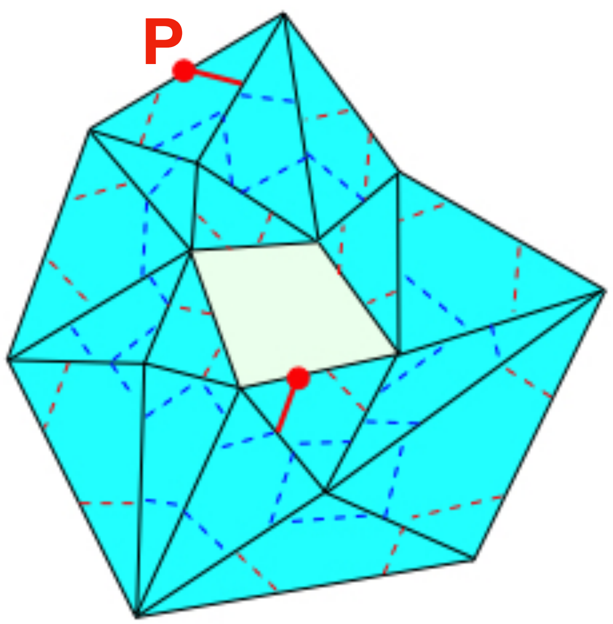
We do use the red arc types near P and near Q. Set them equal to 1.



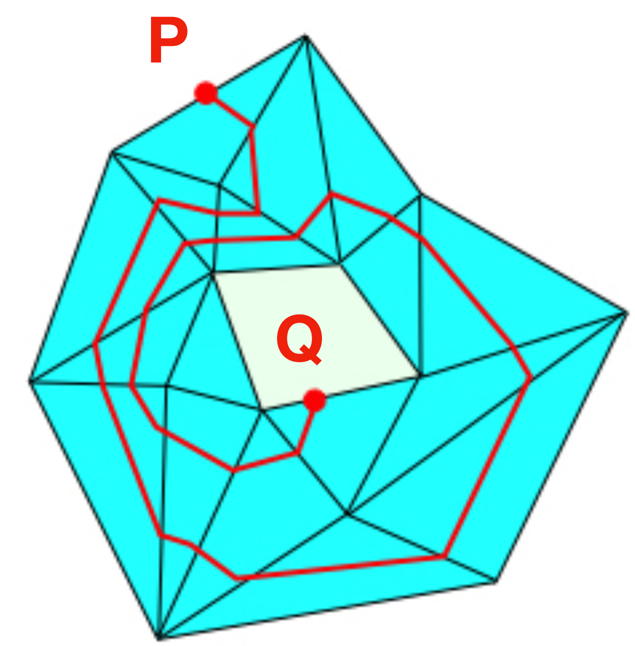
The dashed blue arc types can be used.

Normal Curves

Equivalent Algebraic Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations, with $v_3 = 1$ and $v_{15} = 1$, and with $v_k = 0$ for all other arcs meeting the boundary?

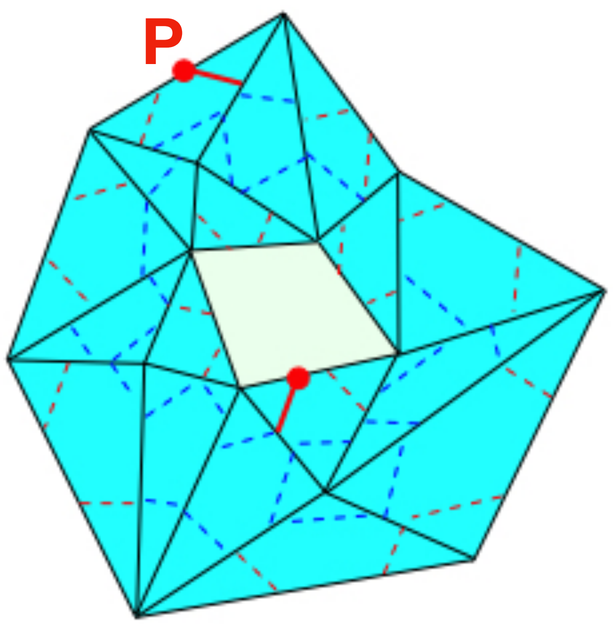


Yes, there is a solution.

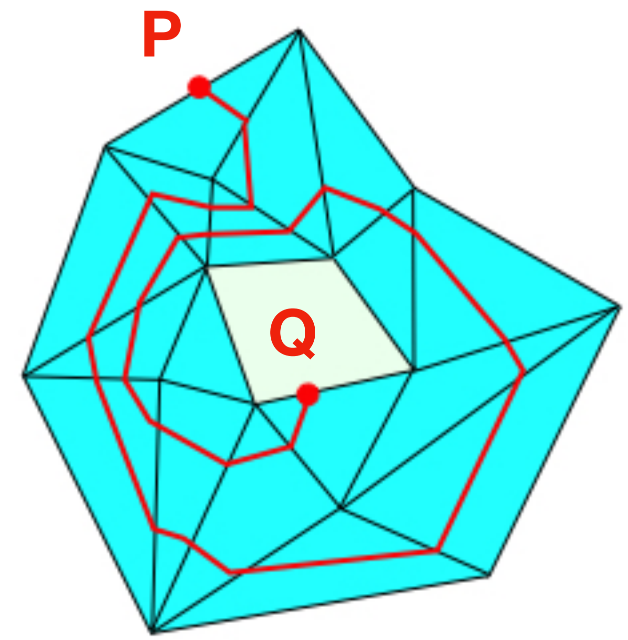


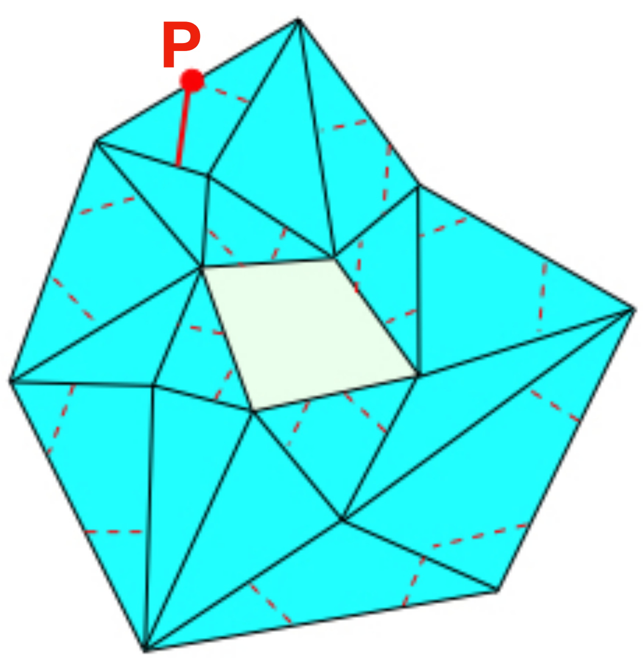
Normal Curves

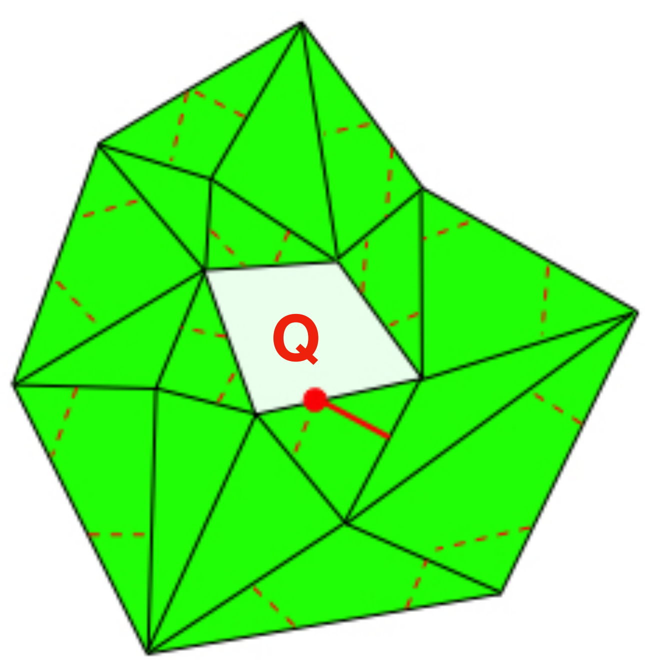
Equivalent Algebraic Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations, with $v_3 = 1$ and $v_{15} = 1$, and with $v_k = 0$ for all other arcs meeting the boundary?



Yes



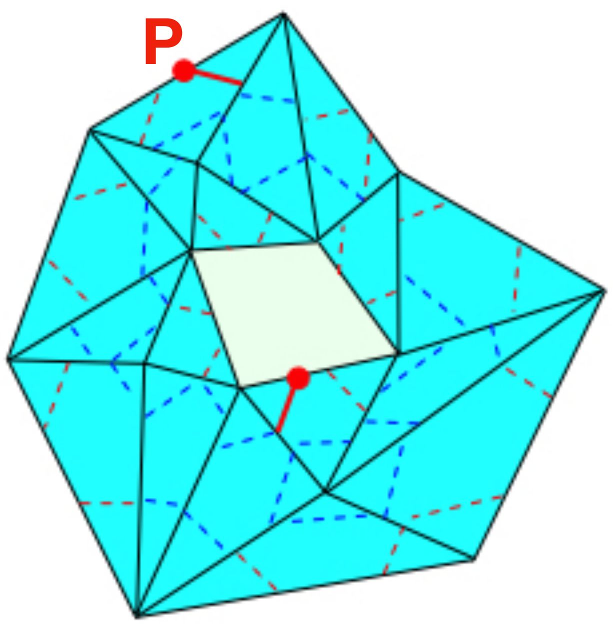




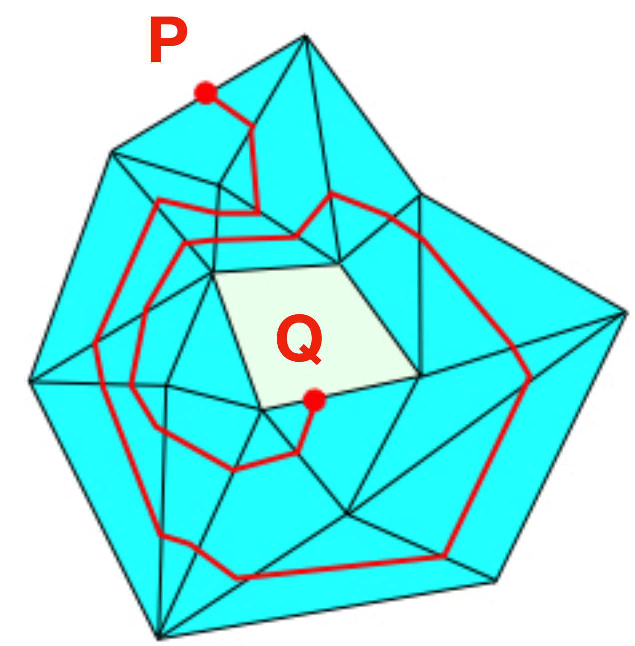
No

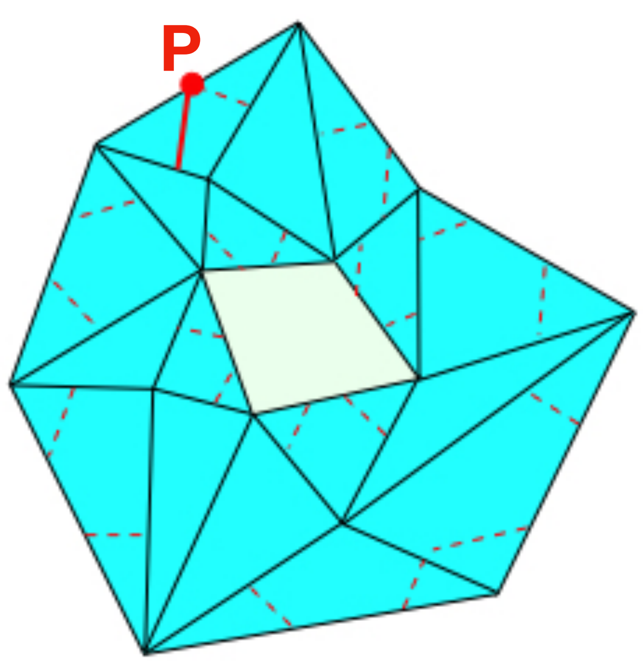
Normal Curves

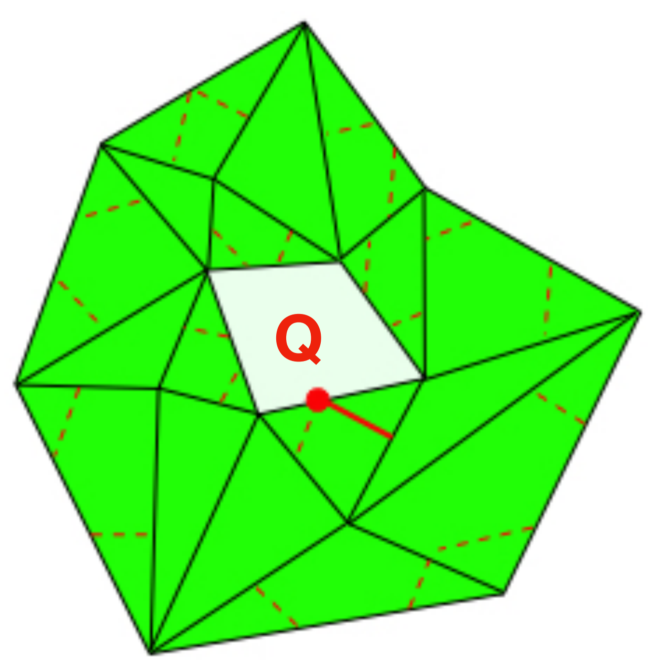
Equivalent Algebraic Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations, with $v_3 = 1$ and $v_{15} = 1$, and with $v_k = 0$ for all other arcs meeting the boundary?



Yes







No

We can answer such questions by an algorithmic procedure, using **Integer Linear Programming**

Integer Linear Programming

A procedure going back to Hilbert constructs a finite set of solutions, called Fundamental solutions, to problems of integer linear programming.
All solutions are sums of a finite number of these “Hilbert basis” solutions.

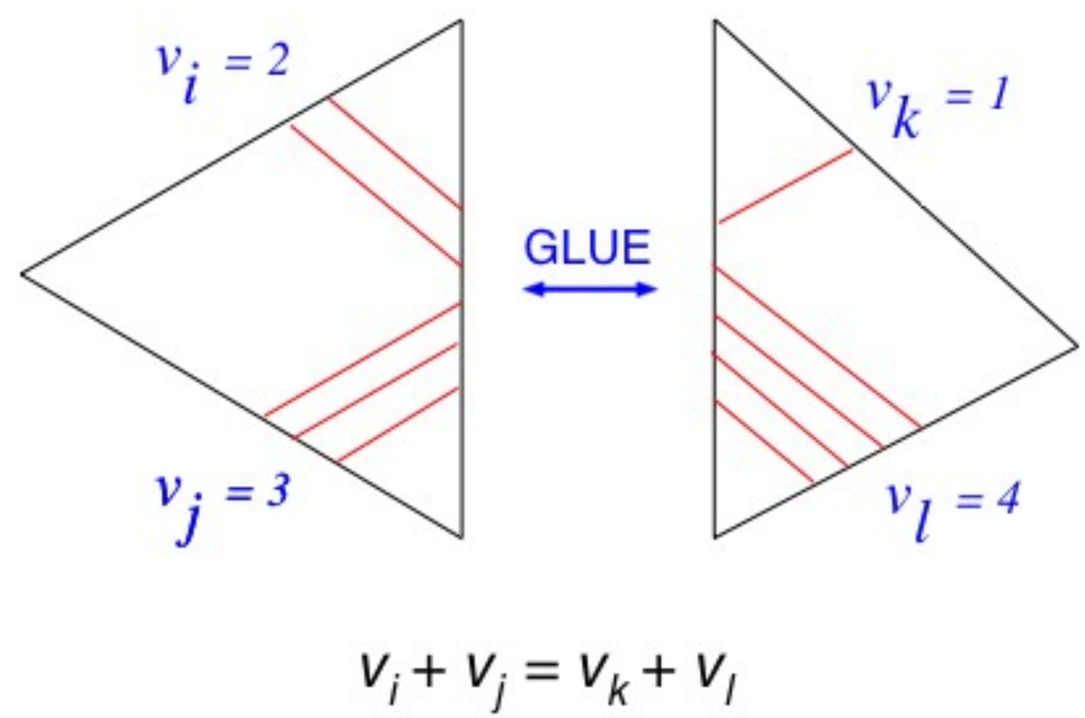
Consider a collection of linear equations and inequalities in $\mathbf{R}^d$.
For normal curves on a surface with t triangles, $d=3t$, and these equations are

$$v_i + v_j - v_k - v_l = 0$$

for all v_i, v_j, v_k, v_l, sharing a common edge.
Also require that each v_i is an integer and $v_i \geq 0$.

These equations define a linear subspace of $\mathbf{R}^d$.

What can we say about the solutions?

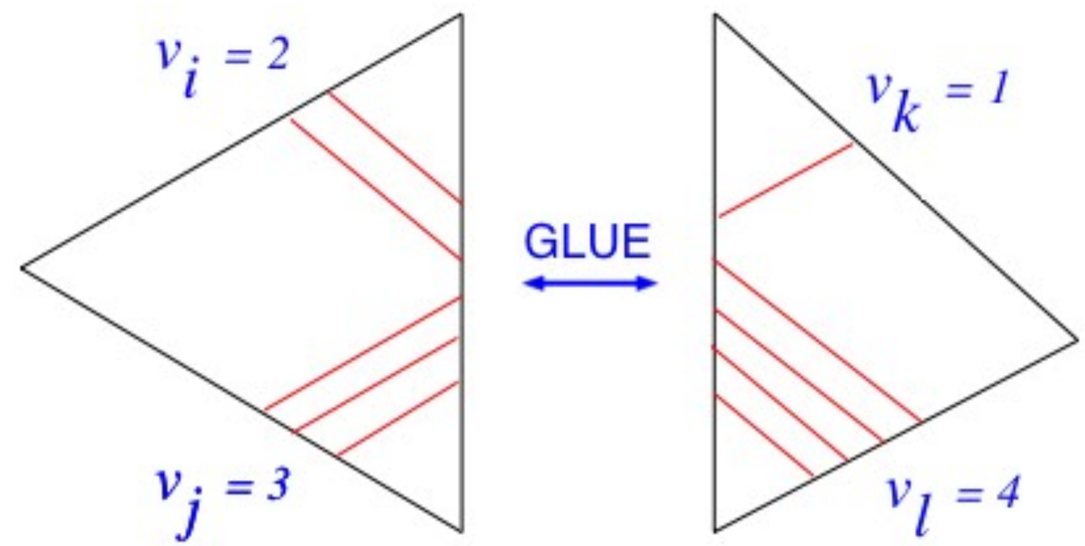


Integer Linear Programming - A Quick Introduction

A procedure going back to Hilbert constructs a finite set of solutions, called Fundamental solutions, to problems of integer linear programming.
All solutions are sums of a finite number of these "Hilbert basis" solutions.

Consider a collection of linear equations and inequalities in $\mathbf{R}^d$.
For normal curves on a surface with t triangles, $d=3t$, and these equations are

$$v_i + v_j - v_k - v_l = 0$$



$$v_i + v_j = v_k + v_l$$

for all $v_i, v_j, v_k, v_l,$ sharing a common edge.
Also require that each v_i is an integer and $v_i \geq 0$.

These equations define a linear subspace of $\mathbf{R}^d$.

What can we say about the solutions?

Theorem.
There are finitely many "fundamental" solutions $\mathbf{F_1}, \ldots, \mathbf{F_k}$ such that any integer solution $\mathbf{w}$ is an integer linear combination

$$\mathbf{w} = n_1\mathbf{F_1} + n_2\mathbf{F_2} + \ldots + n_k\mathbf{F_k}$$

for some integers $n_1, n_2, \ldots, n_k \geq 0$. These fundamental solutions can be constructed explicitly.

Integer Linear Programming

Example: Take $d=3$, so looking at vectors in $\mathbf{R}^3$, satisfying

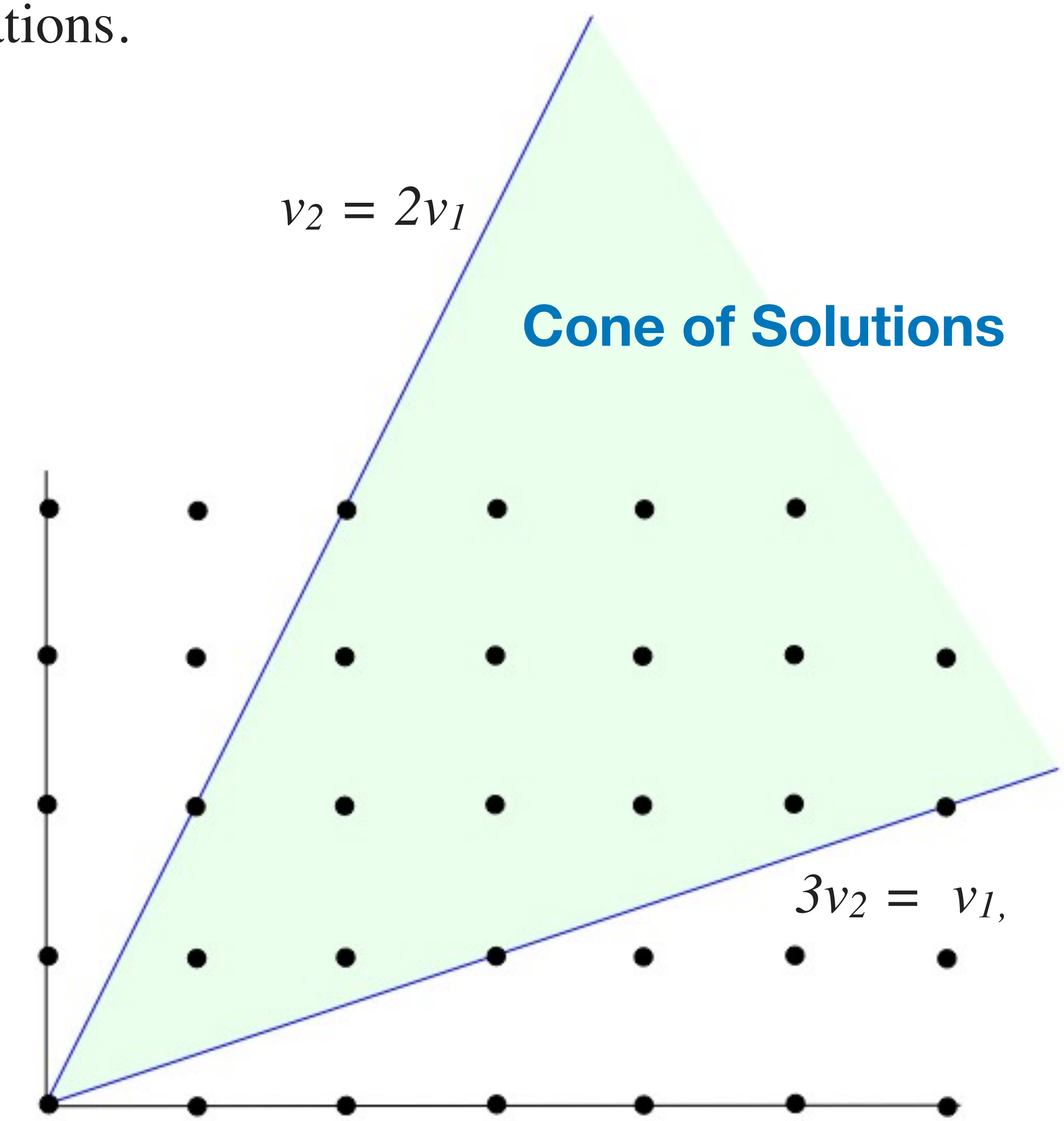
Linear equations: $v_3 = 0$

Inequalities $v_1 \geq 0, \quad 2v_1 - v_2 \geq 0, \quad 3v_2 \geq v_1$

Find all integer vectors (v_1, v_2, v_3) satisfying these equations.

Solution: Take any integer vector (v_1, v_2, v_3) with $v_3 = 0, v_2 \leq 2v_1, 3v_2 \geq v_1.$

How can we construct all the lattice points inside this cone?



Integer Linear Programming

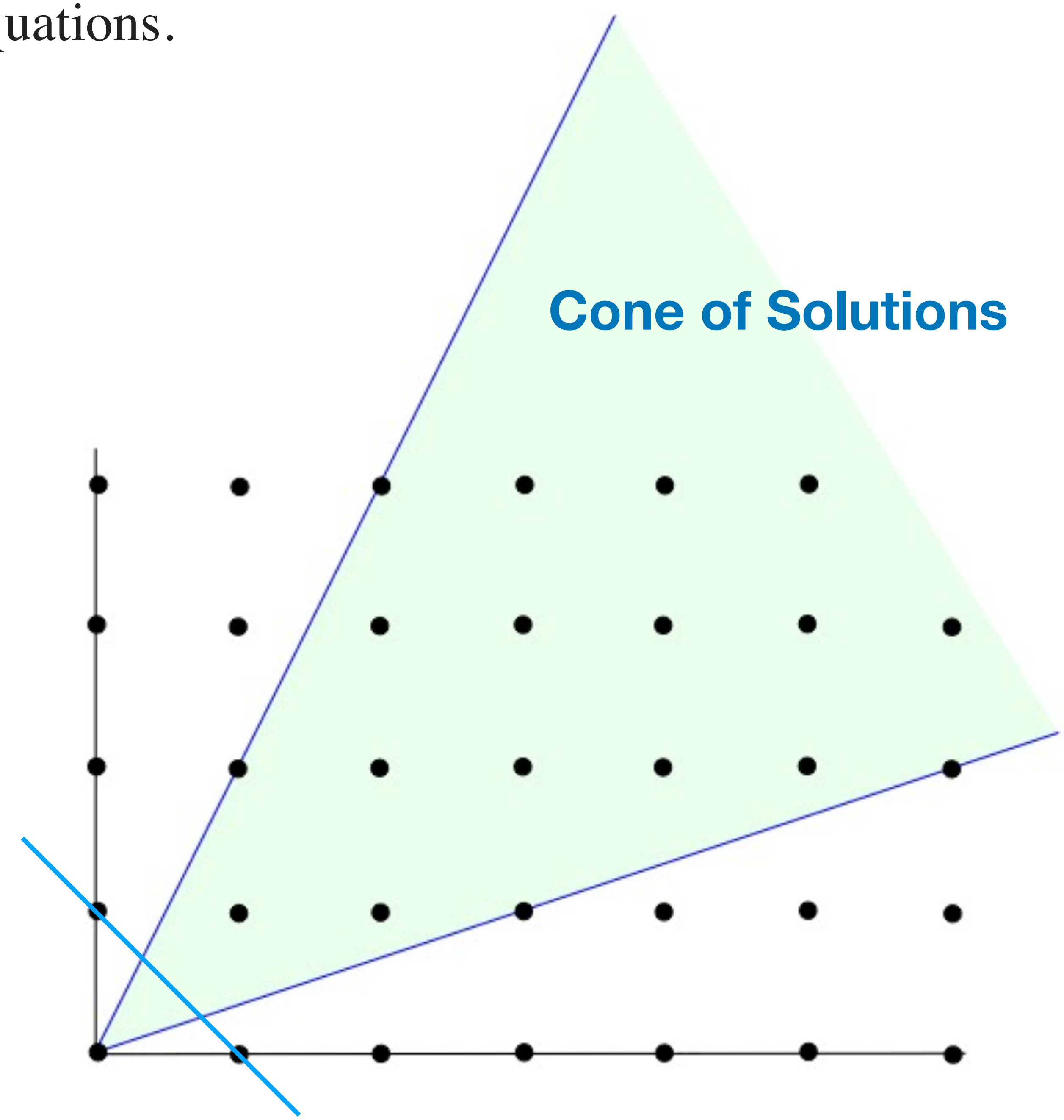
Example: Take $d=3$, so looking in **R**^{3}.
Linear equations: $v_3 = 0$
Inequalities $v_1 \geq 0$, $2v_1 - v_2 \geq 0$, $3v_2 \geq v_1$

Find all integer vectors (v_1, v_2, v_3) satisfying these equations.

Solution: Take any integer vector (v_1, v_2, v_3) with $v_3 = 0$, $v_2 \leq 2v_1$, $3v_2 \geq v_1$.

How can we construct all the lattice points inside this cone?

1. Projectivize: Add equation
$v_1 + v_2 + v_3 = 1$



Integer Linear Programming

Example: Take $d=3$, so looking in R^3.

Linear equations: $v_3 = 0$

Inequalities $v_1 \geq 0$, $2v_1 - v_2 \geq 0$, $3v_2 \geq v_1$

Find all integer vectors (v_1, v_2, v_3) satisfying these equations.

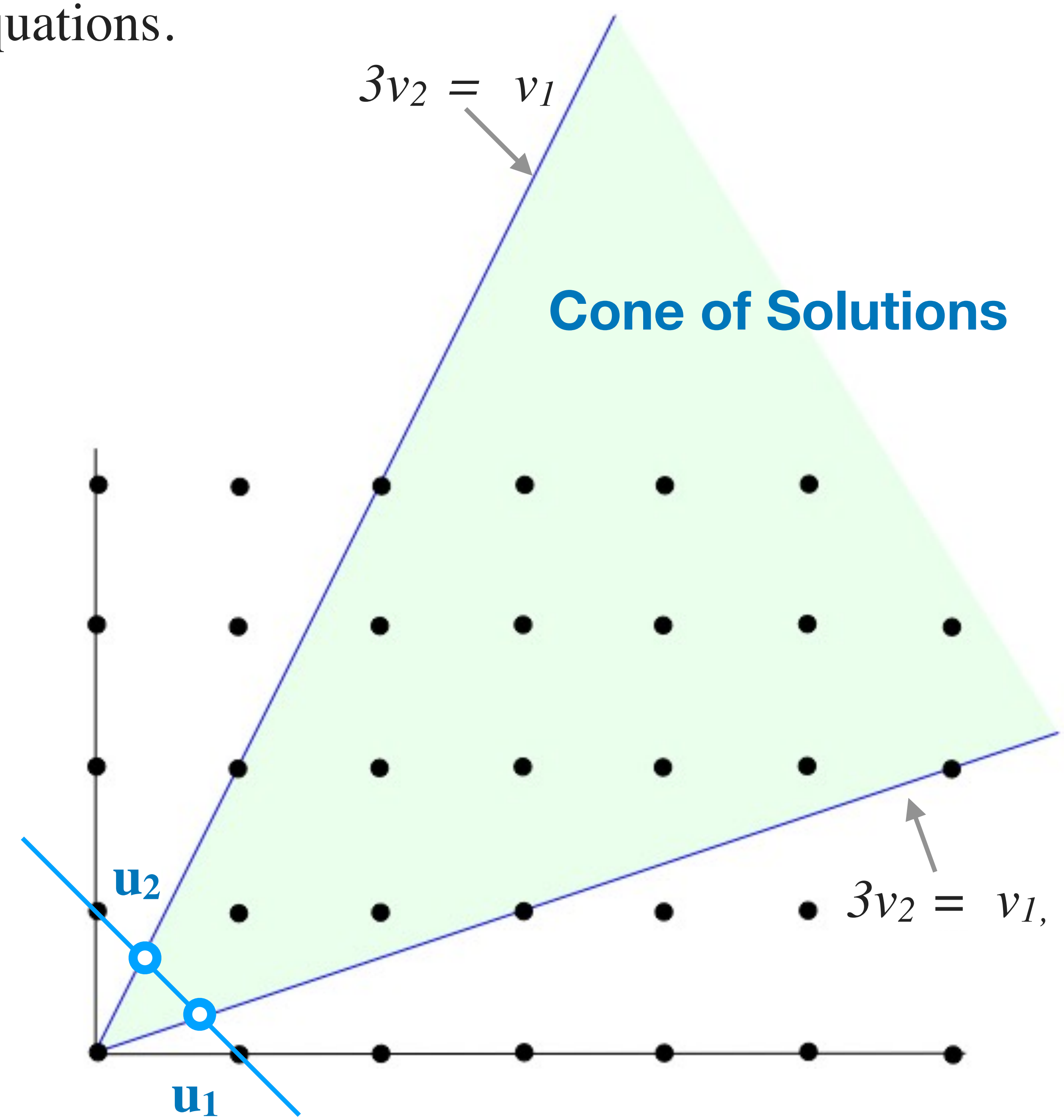
Solution: Take any integer vector (v_1, v_2, v_3) with $v_3 = 0$, $v_2 \leq 2v_1$, $3v_2 \geq v_1$.

How can we construct all the lattice points inside this cone?

1. Projectivize: Add equation $v_1 + v_2 + v_3 = 1$

2. Replace inequality $3v_2 \geq v_1$ with equality $3v_2 = v_1$ to find rational vertex solution $\mathbf{u_2}$ (and $\mathbf{u_1}$).

$\mathbf{u_1} = < 3/4, 1/4, 0 >$, $\mathbf{u_2} = < 1/3, 2/3, 0 >$



Integer Linear Programming

Example: Take $d=3$, so looking in R^3.
Linear equations: $v_3 = 0$
Inequalities $v_1 \geq 0,\ 2v_1 - v_2 \geq 0,\ 3v_2 \geq v_1$

Find all integer vectors (v_1, v_2, v_3) satisfying these equations.

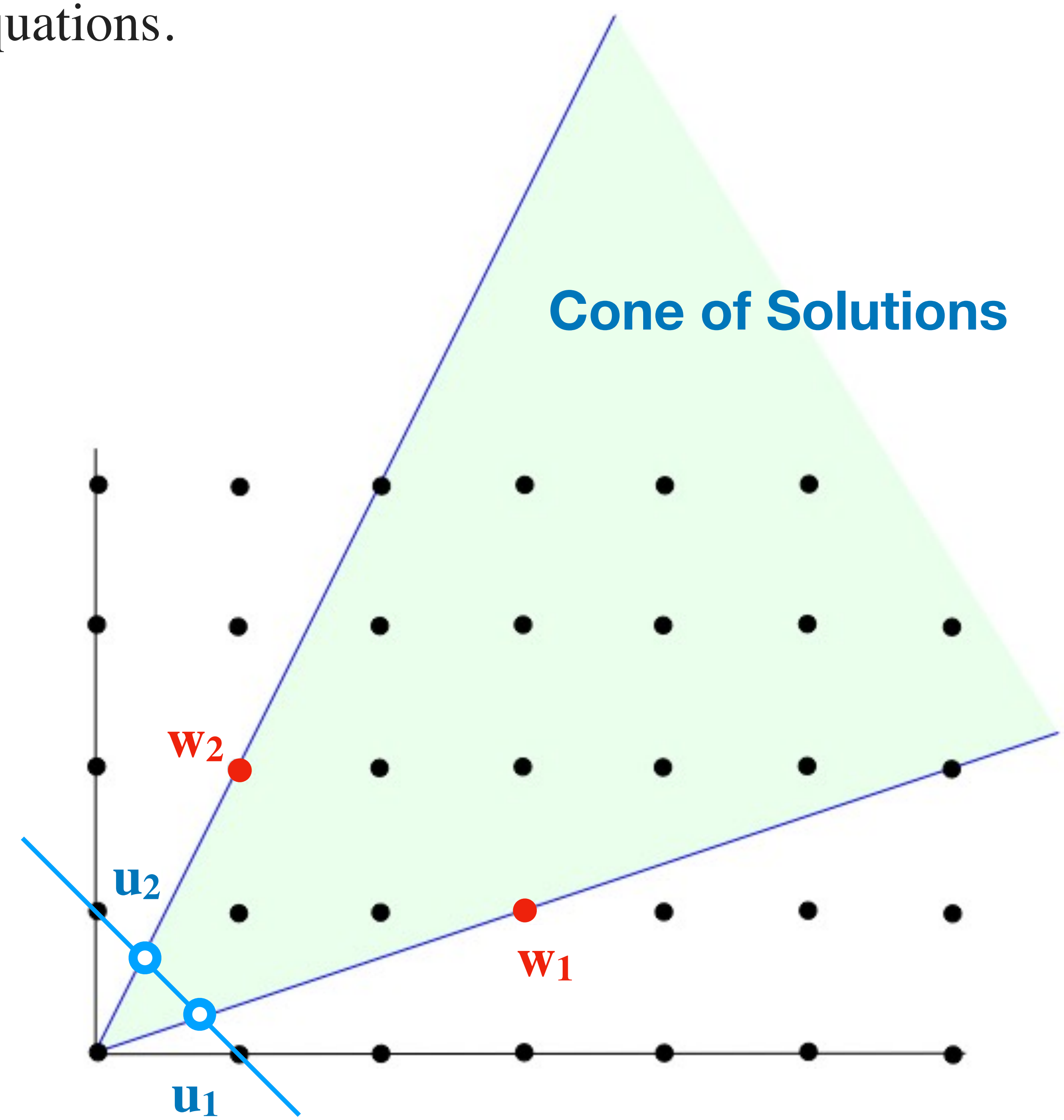
Solution: Take any integer vector (v_1, v_2, v_3) with $v_3 = 0,\ v_2 \leq 2v_1,\ 3v_2 \geq v_1$.

How can we construct all the lattice points inside this cone?

1. Projectivize: Add equation $v_1 + v_2 + v_3 = 1$

2. Replace inequality $3v_2 \geq v_1$ with equality $3v_2 = v_1$ to find rational vertex solution $\mathbf{u_2}$ (and $\mathbf{u_1}$).
$\mathbf{u_1} = < 3/4, 1/4, 0 >$, $\mathbf{u_2} = < 1/3, 2/3, 0 >$

3. Clear denominators to get integer vertex solutions $\mathbf{w_1}$ and $\mathbf{w_2}$.
$\mathbf{w_1} = < 3, 1, 0 >$, $\mathbf{w_2} = < 1, 2, 0 >$



Integer Linear Programming

Claim: Every solution in the cone is a finite integer linear combination of finitely many *fundamental* integral solutions. (The Hilbert Basis).

Note: Not all solutions are integer linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$.

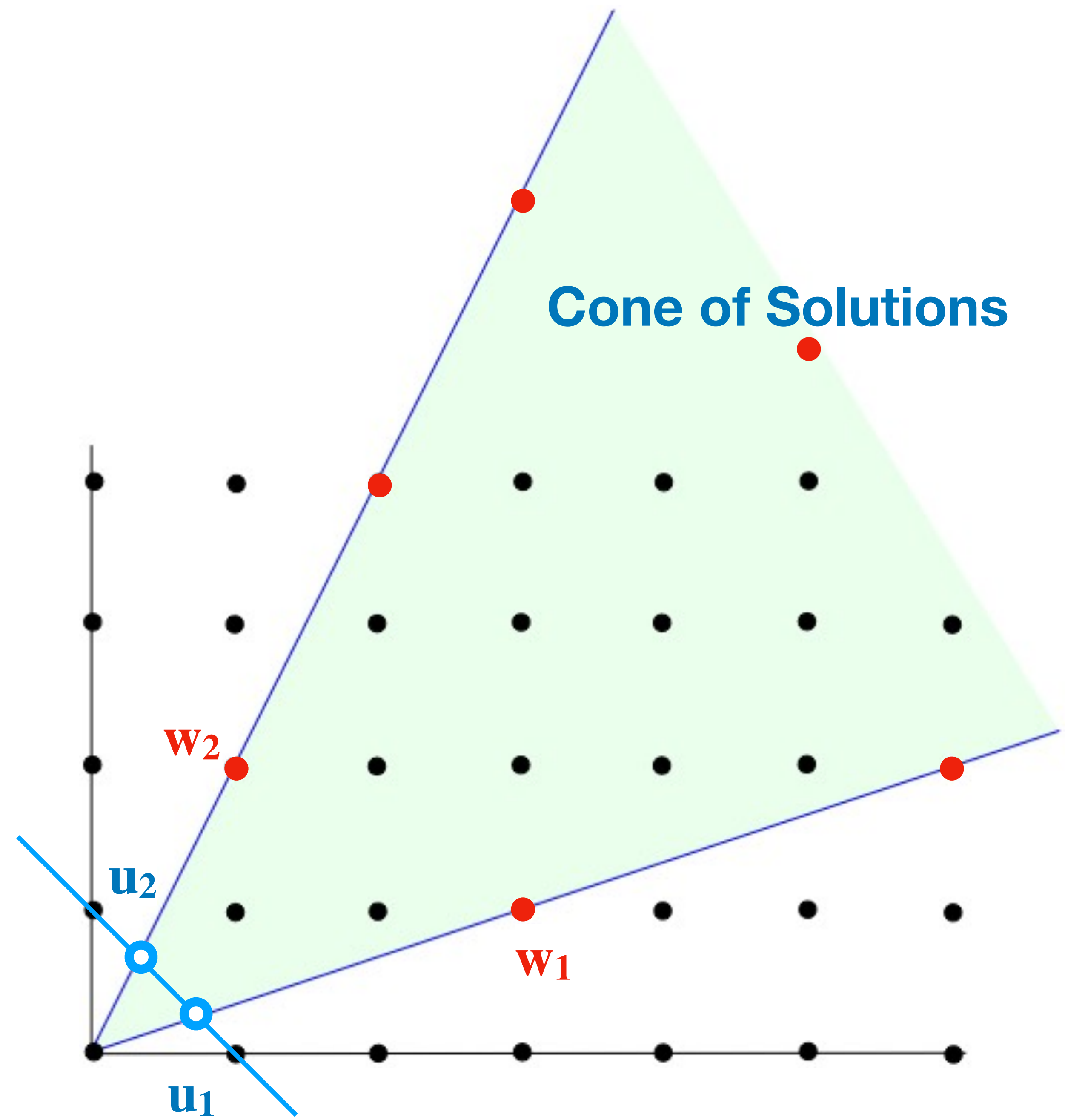
1. Projectivize: Add equation $v_1 + v_2 + v_3 = 1$

2. Replace inequality $3v_2 \geq v_1$ with equality $3v_2 = v_1$ to find rational vertex solution $\mathbf{u_2}$ (and $\mathbf{u_1}$).

$\mathbf{u_1} = < 3/4, 1/4, 0 >$, $\mathbf{u_2} = < 1/3, 2/3, 0 >$

3. Clear denominators to get integer vertex solutions $\mathbf{w_1}$ and $\mathbf{w_2}$.

$\mathbf{w_1} = < 3, 1, 0 >$, $\mathbf{w_2} = < 1, 2, 0 >$



Integer Linear Programming

Claim: Every solution in the cone is an integer linear combination of finitely many *fundamental* integral solutions. (The Hilbert Basis).

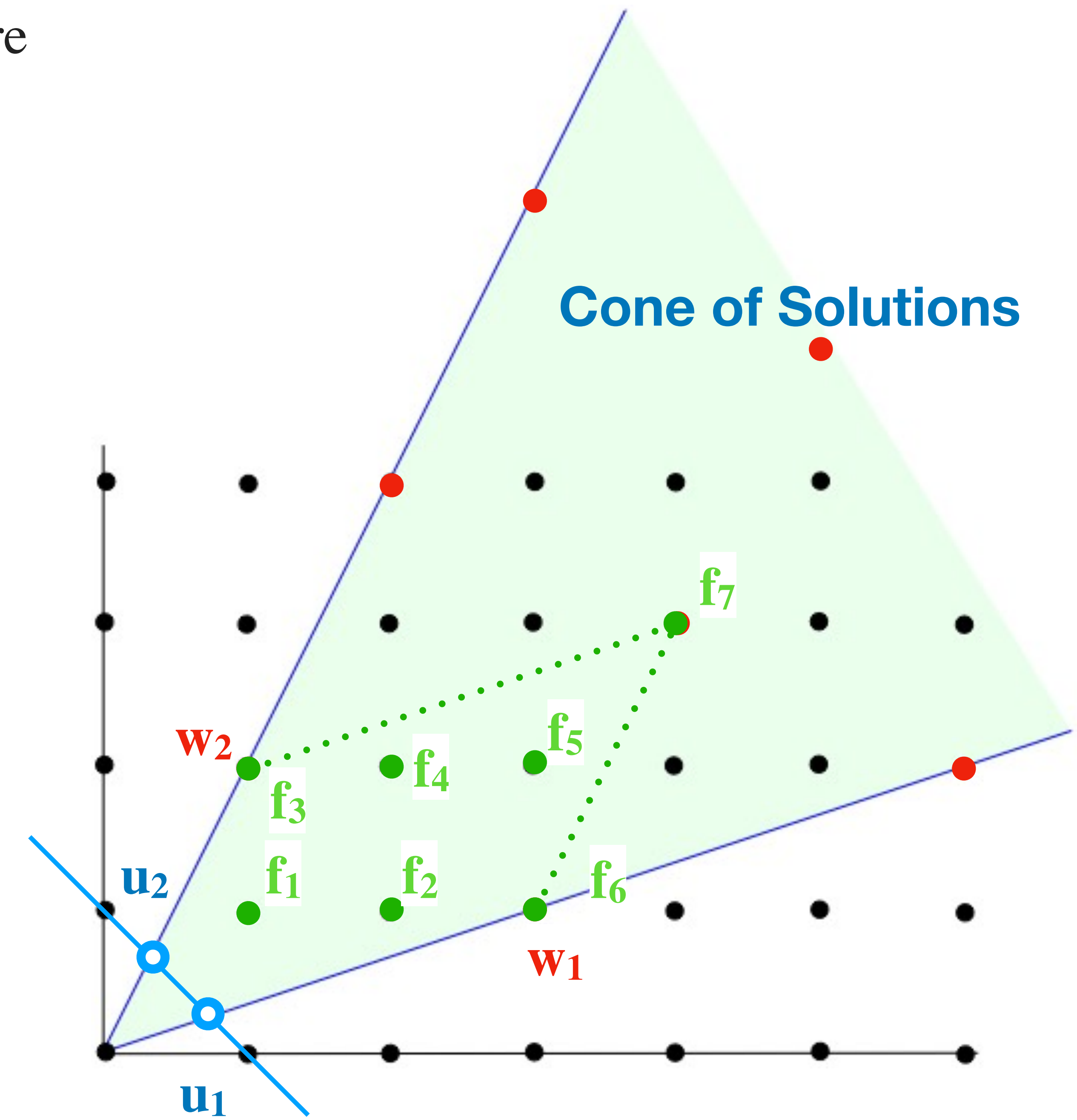
Note: Not all solutions are integer linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$.

All solutions are *rational* linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$.

All solutions are *integer* linear combinations of $\mathbf{f_i}$.

The $\mathbf{f_i}$ are the finitely many lattice solutions that are rational linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$

$\mathbf{f_i} = q_1\mathbf{w_1} + q_2\mathbf{w_2}$ with $0 \leq q_i \leq 1$.



Integer Linear Programming

Claim: Every solution in the cone is an integer linear combination of finitely many *fundamental* integral solutions. (The Hilbert Basis).

Note: Not all solutions are integer linear combinations of $\mathbf{w}_1$ and $\mathbf{w}_2$.
All solutions are *rational* linear combinations of $\mathbf{w}_1$ and $\mathbf{w}_2$.

All solutions are integer linear combinations of $\mathbf{f}_i$.
The $\mathbf{f}_i$ are the finitely many lattice solutions that are rational linear combinations of $\mathbf{w}_1$ and $\mathbf{w}_2$

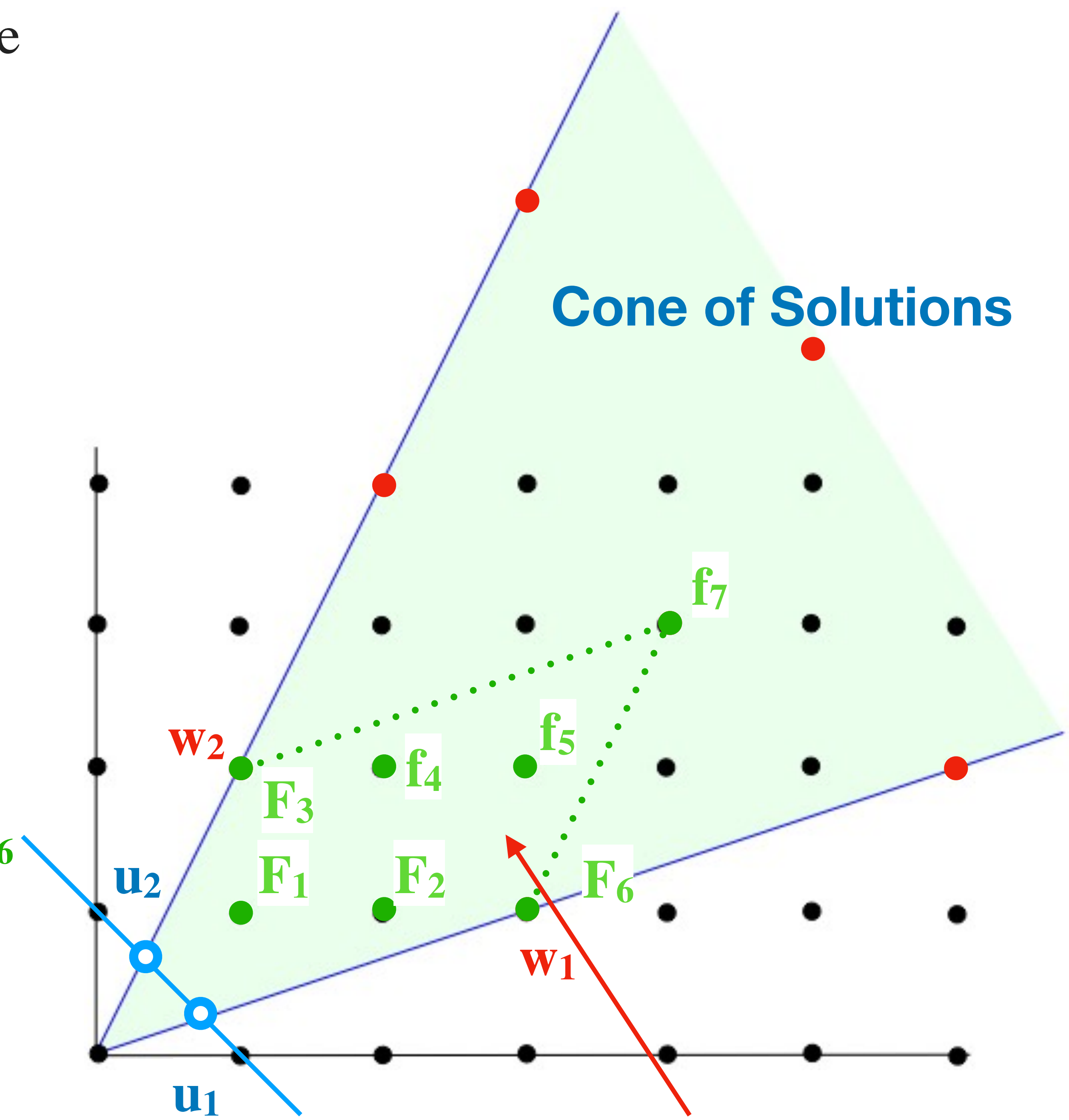
$\mathbf{f}_i = q_1\mathbf{w}_1 + q_2\mathbf{w}_2$ with $0 \le q_i \le 1$.

A *fundamental* solution $\mathbf{F}_i$ is not a sum of two other solutions.

Claim: *Any* integer solution in the cone is an integer linear combination of fundamental solutions.

There are four fundamental solutions: $\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3, \mathbf{F}_6$

$\mathbf{w}_1 = \mathbf{F}_6 = \langle 3, 1, 0 \rangle$, $\mathbf{w}_2 = \mathbf{F}_3 = \langle 1, 2, 0 \rangle$



Fundamental solutions lie within this parallelepiped.

Integer Linear Programming

Claim: Every solution in the cone is an integer linear combination of finitely many *fundamental* integral solutions. (The Hilbert Basis).

Note: Not all solutions are integer linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$.
All solutions are *rational* linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$.

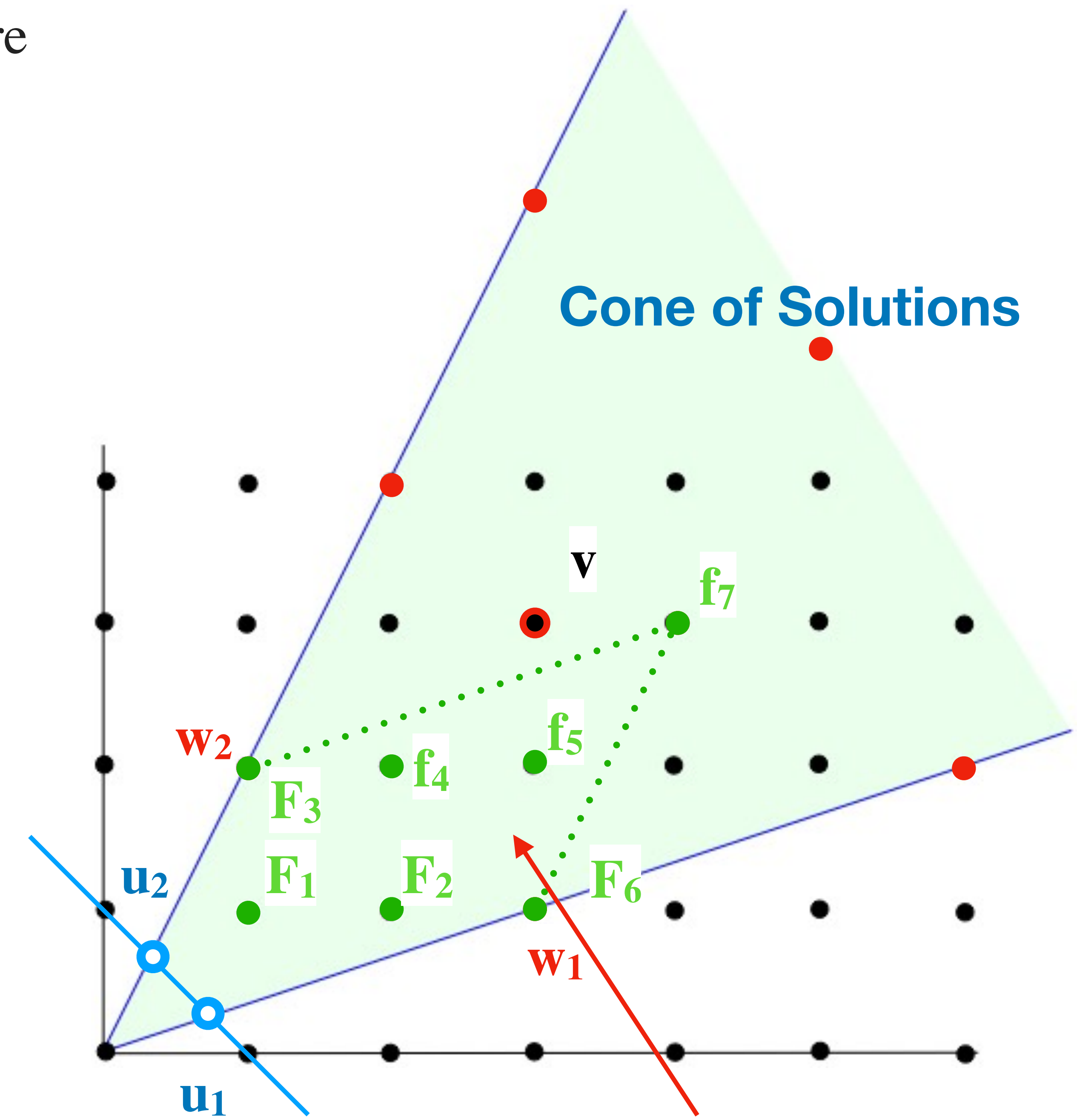
All solutions are integer linear combinations of $\mathbf{f_i}$.
The $\mathbf{f_i}$ are the finitely many lattice solutions that are rational linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$

$\mathbf{f_i} = q_1\mathbf{w_1} + q_2\mathbf{w_2}$ with $0 \leq q_i \leq 1$.

A *fundamental* solution $\mathbf{F_i}$ is not a sum of two other solutions.

Claim: *Any* integer solution in the cone is an integer linear combination of fundamental solutions.

Example: ● $\mathbf{v}$



Fundamental solutions lie within this parallelepiped.

Integer Linear Programming

Claim: Every solution in the cone is an integer linear combination of finitely many *fundamental* integral solutions. (The Hilbert Basis).

Note: Not all solutions are integer linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$.
All solutions are *rational* linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$.

All solutions are integer linear combinations of $\mathbf{f_i}$.
The $\mathbf{f_i}$ are the finitely many lattice solutions that are rational linear combinations of $\mathbf{w_1}$ and $\mathbf{w_2}$

$\mathbf{f_i} = q_1\mathbf{w_1} + q_2\mathbf{w_2}$ with $0 \le q_i \le 1$.

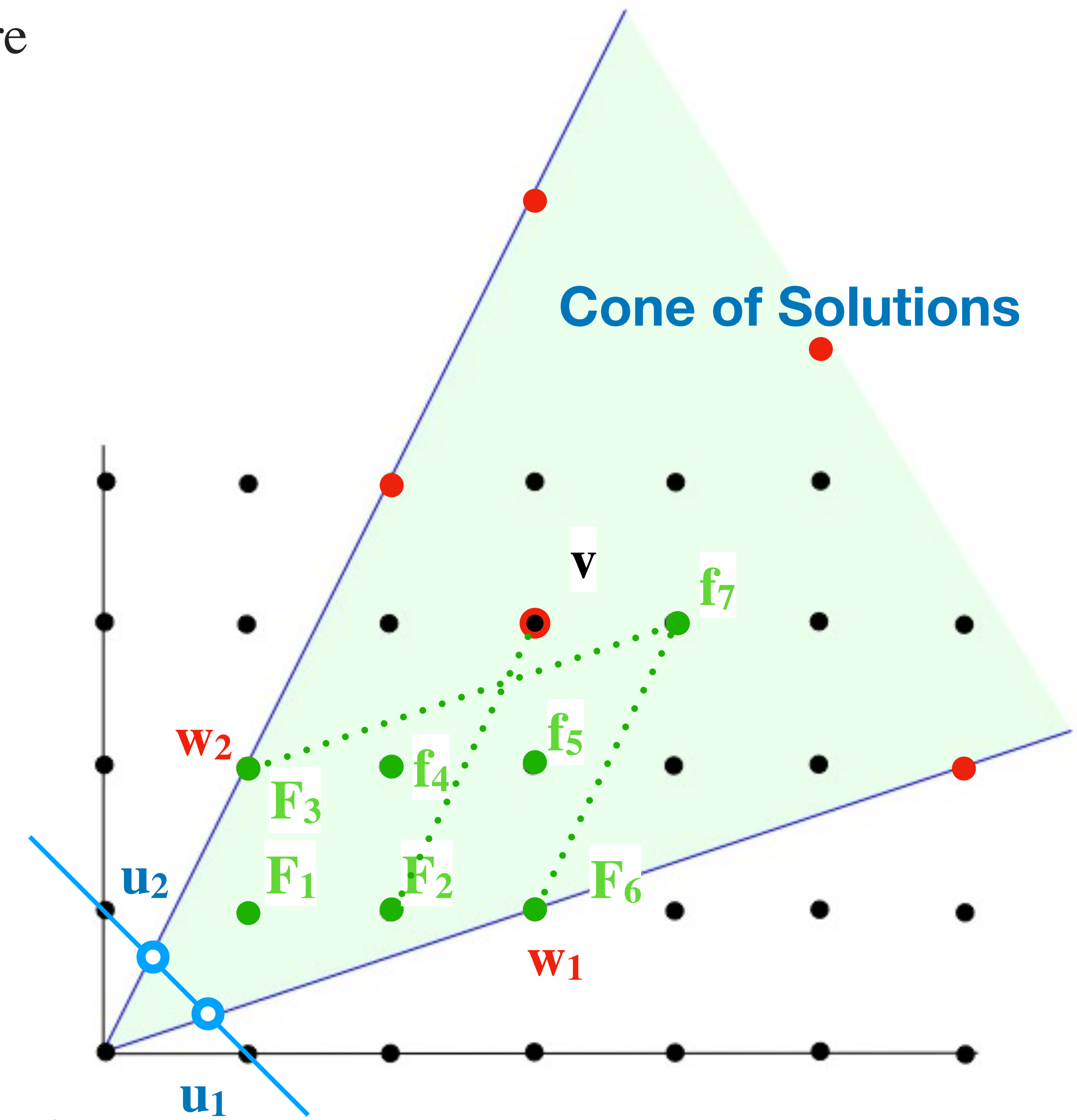
A *fundamental* solution $\mathbf{F_i}$ is not a sum of two other solutions.

Claim: *Any* integer solution in the cone is an integer linear combination of fundamental solutions.

$$\mathbf{v} - \mathbf{F_3} = \mathbf{F_2}$$

Example: $\mathbf{v}$

$$\mathbf{v} = \mathbf{F_2} + \mathbf{F_3}$$



Non-fundamental solutions are sums of smaller solutions.

Summary - Integer Linear Programming

Take a collection of linear equations in $\mathbf{R}^d$ with integer coefficients, such as $v_i + v_j - v_k - v_l = 0$, and inequalities, such as $v_i \geq 0$, that describe a pointed cone in $\mathbf{R}^d$.

A solution vector $\mathbf{F}$ in $\mathbf{Z}^\mathbf{d}$ that cannot be written as a sum of two non-zero normal vectors in $\mathbf{Z}^\mathbf{d}$, $\mathbf{F} \neq \mathbf{A} + \mathbf{B}$, is called *fundamental*.

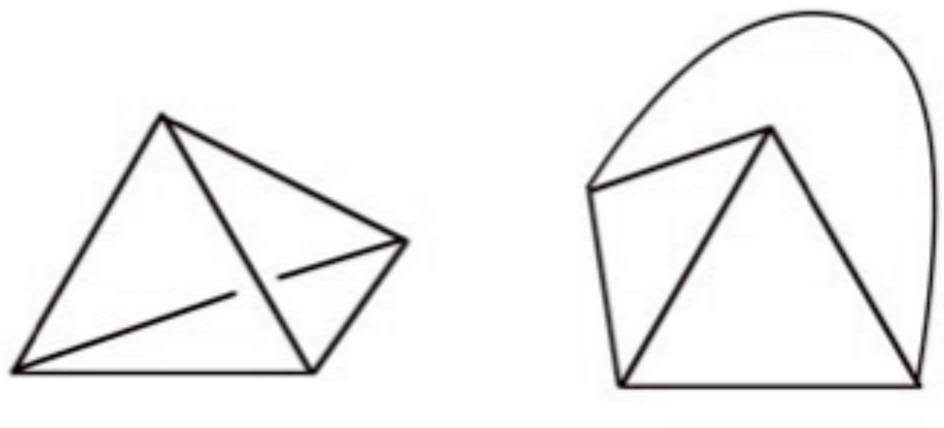
Theorem.
There are finitely many fundamental solutions $\mathbf{F_1}, ..., \mathbf{F_k}$ such that any integer solution $\mathbf{w}$ is an integer linear combination of these solutions:

$$\mathbf{w} = n_1\mathbf{F_1} + n_2\mathbf{F_2} + \dots + n_k\mathbf{F_k}$$

for integers $n_1, n_2, \dots, n_k \geq 0$.
Moreover there is a procedure to construct these solutions.

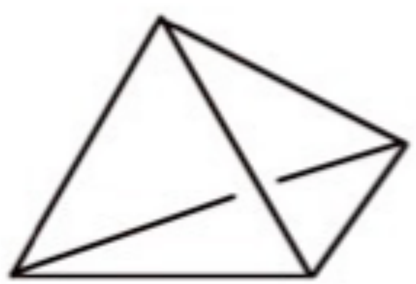
We will revisit this construction when we look at computational complexity.

Simple example - Fundamental Normal curves

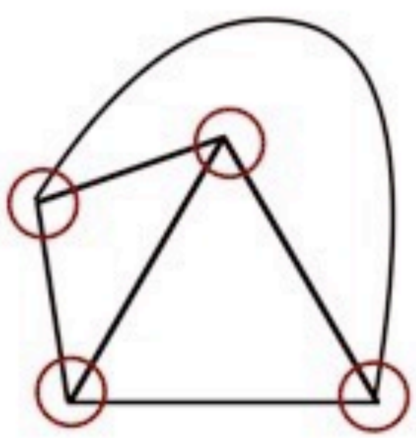


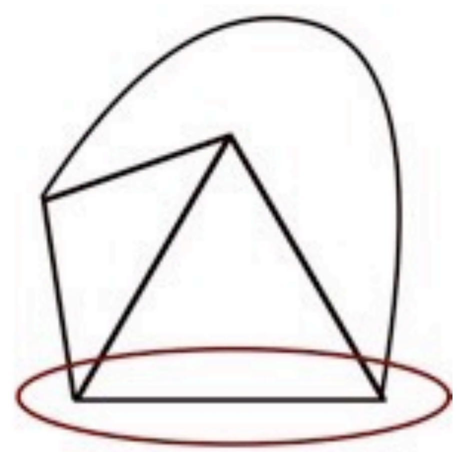
A triangulation of S^2 with 4 triangles.
There are many normal curves.
Seven are fundamental.
Which ones?

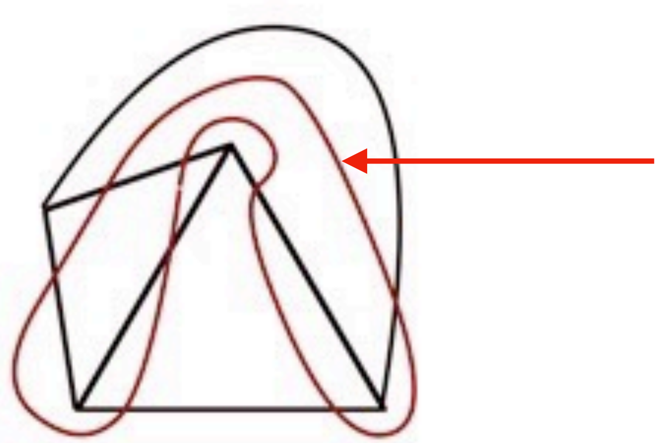
Simplest example



A triangulation of S^2 with 4 triangles.
There are many normal curves.
Seven are fundamental.
Which ones?







Exercise: This normal curve is not fundamental. It is a sum of two fundamental curves. Which ones?

Curves and Normal Vectors

We have shown that

1. There is a 1-1 correspondence between normal curves and normal vectors.

2. All normal vectors are a sum of a finite number of fundamental normal vectors, and this finite "Hilbert Basis" can be constructed algorithmically.

Next we show:

3. If a solution exists for a problem we are interested in (such as an unknotting disk or a splitting 2-sphere) then a solution exists among the fundamental surfaces.

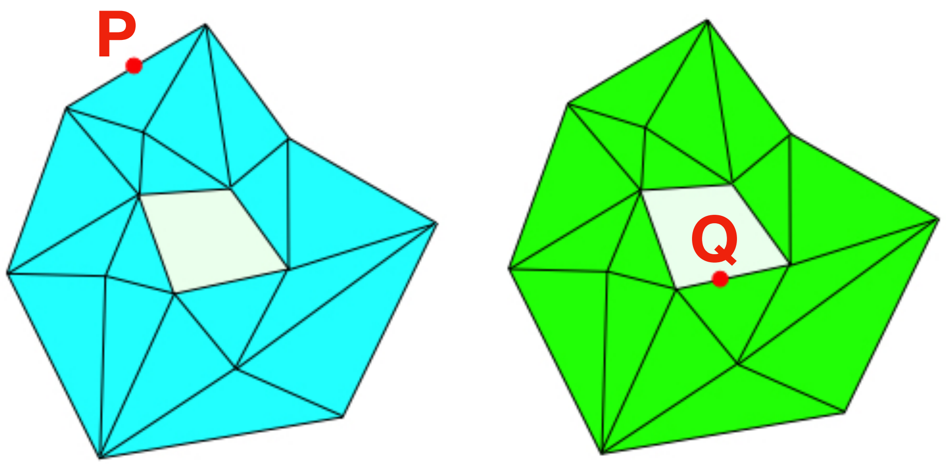
Following Haken, we then obtain an algorithm with the following steps:

1. Compute the fundamental solutions to the integer linear equations determined b our problem.
2. Check if any of this finite collection of surfaces solves our problem.

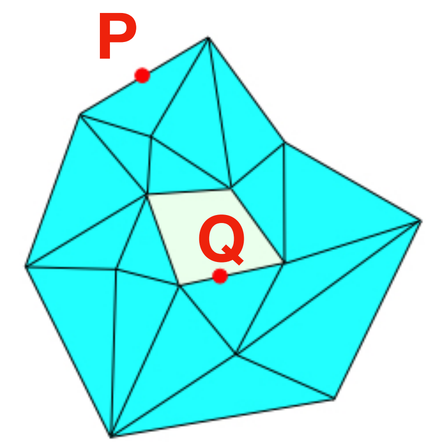
This applies to a large number of 3-manifold algorithms.

We look at an example algorithm for curves on a surface:

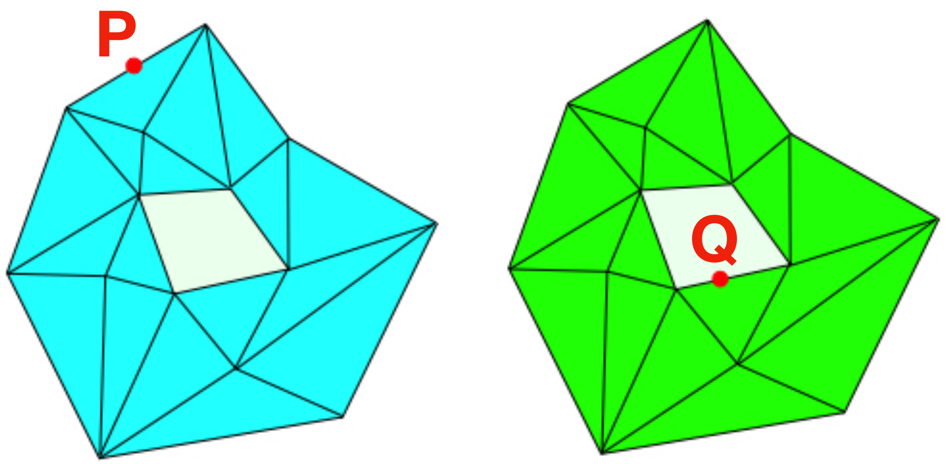
Example - A Normal Curve Algorithm



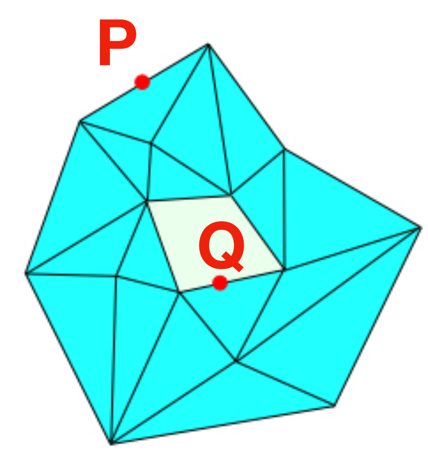
Sample Question:
Is there a curve in F connecting boundary points P and Q?



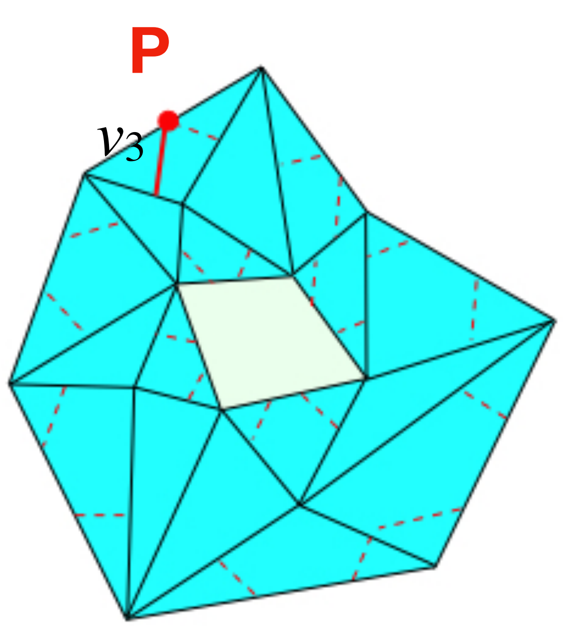
A Normal Curve Algorithm

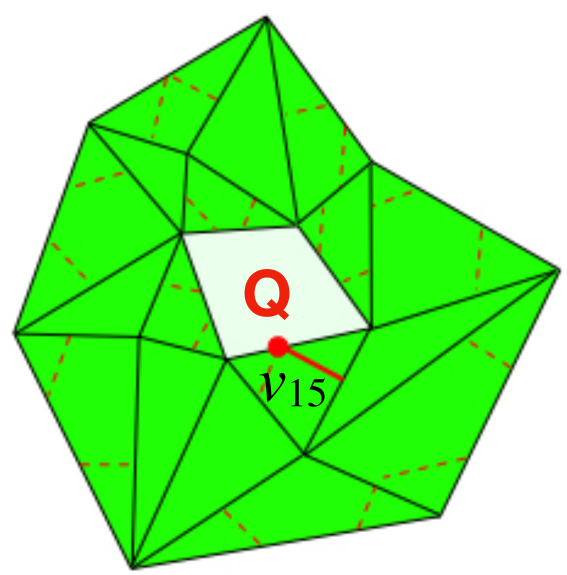


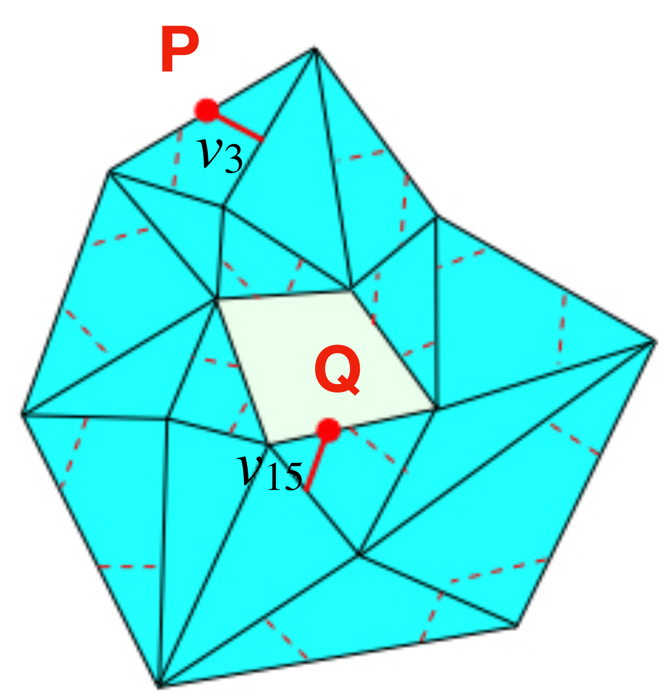
Sample Question:
Is there a curve in *F* connecting boundary points *P* and *Q*?



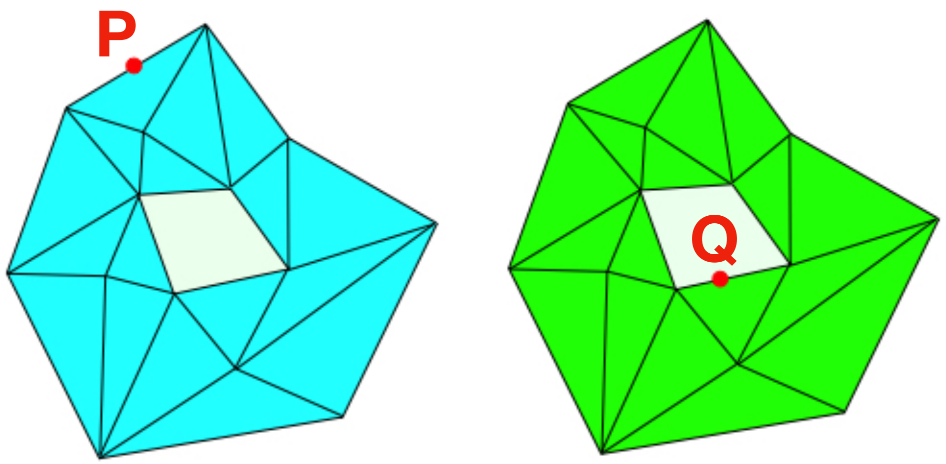
Equivalent Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations with $v_3 = 1$ and $v_{15} = 1$, and with all other variables that meet the boundary equal to zero?



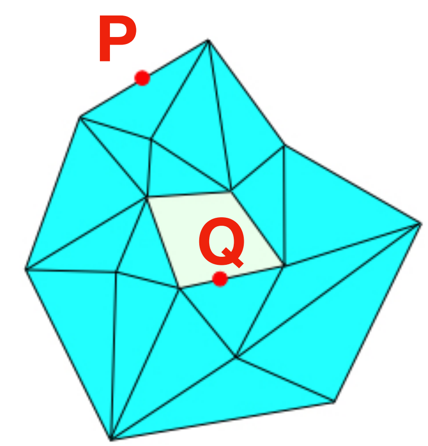




A Normal Curve Algorithm



Sample Question:
Is there a curve in *F* connecting boundary points *P* and *Q*?

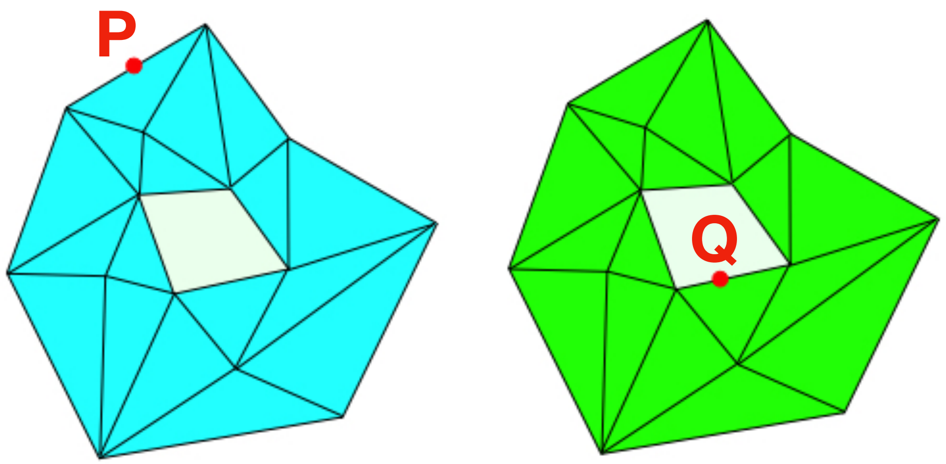


Equivalent Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations with $v_3 = 1$ and $v_{15} = 1$, and with all other variables that meet the boundary equal to zero?

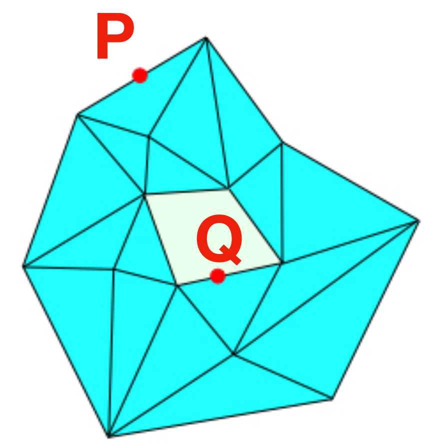
An Algorithm

1. Construct all Fundamental Solutions (There are finitely many and they can be constructed in finite time. We will see how long this takes later.)
2. Check whether any of the Fundamental Solutions satisfies $v_3 = 1$ and $v_{15} = 1$ and all other variables meeting the boundary equal zero.
3. If yes, output "YES". Otherwise output "NO".

A Normal Curve Algorithm



Sample Question: Is there a curve in F connecting boundary points P and Q?



Equivalent Question: Is there an integer vector in $\mathbf{Z}_+^{3t}$ satisfying the matching equations with $v_3 = 1$ and $v_{15} = 1$, and with all other variables that meet the boundary equal to zero?

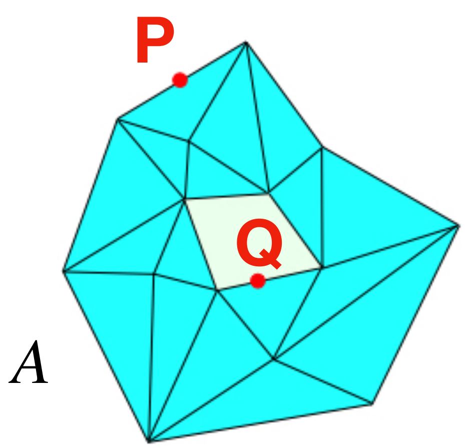
An Algorithm

1. Construct all Fundamental Solutions (There are finitely many and they can be constructed in finite time. We will see how long this takes later.)
2. Check whether any of the Fundamental Solutions satisfies $v_3 = 1$ and $v_{15} = 1$ and all other variables meeting the boundary equal zero.
3. If yes, output "YES". Otherwise output "NO".

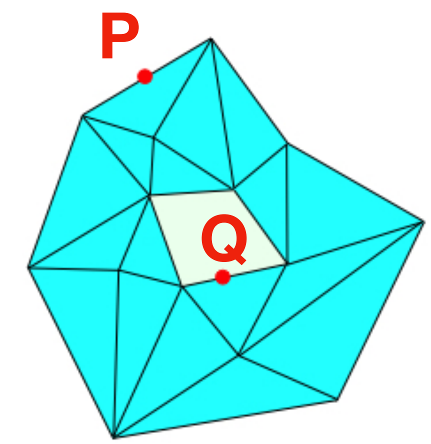
To show the algorithm does what it claims

Need to check: If there is a curve connecting P and Q then there is a *Fundamental Normal Curve* with the same property.

A Normal Curve Algorithm



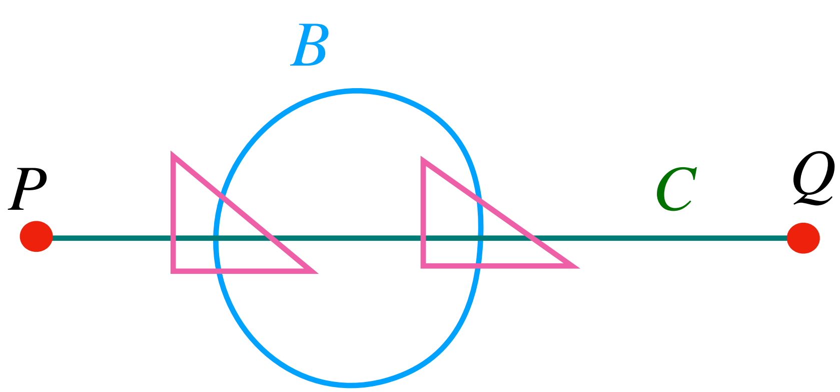
Sample Question:
Is there a curve in *F* connecting boundary points *P* and *Q*?

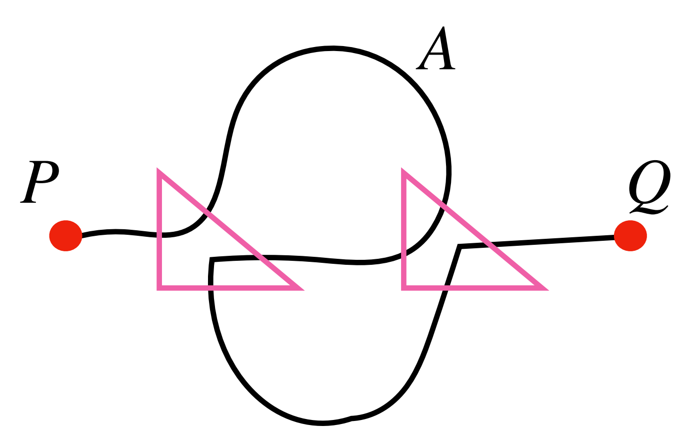


To show the algorithm does what it claims

If there is a curve connecting *P* and *Q* then there is a *Fundamental* Normal Curve with the same property.

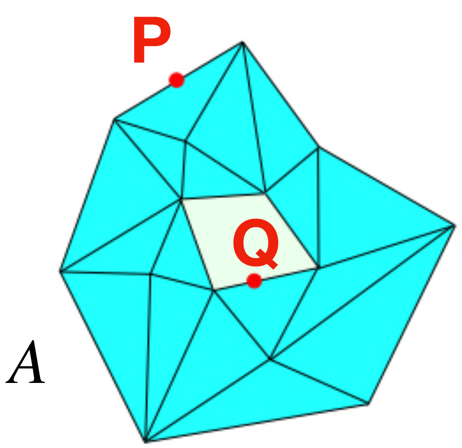
Proof. Normalization gives a normal curve *A* with endpoints *P* and *Q*. We show that we can find such a normal curve A that is fundamental.
Consider the integer vector **A** associated to this curve. Pick **A** to have smallest weight among all normal curves from *P* to *Q*.



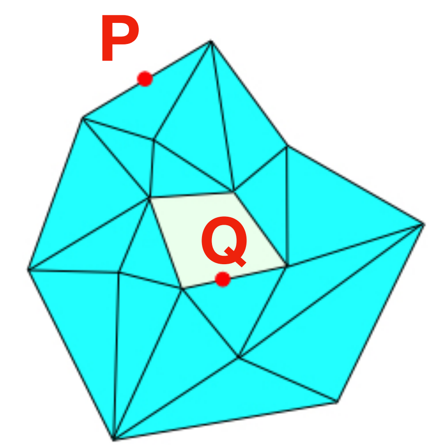


Claim. **A** is fundamental.
Suppose for contradiction that **A** = **B** + **C**. Then one of **B** or **C** has boundary at the points *P* and *Q*. Moreover wt(**A**) = wt(**B**) + wt(**C**), so each of **B** and **C** has smaller weight than **A**. This contradicts the assumption that **A** has smallest weight.

A Normal Curve Algorithm



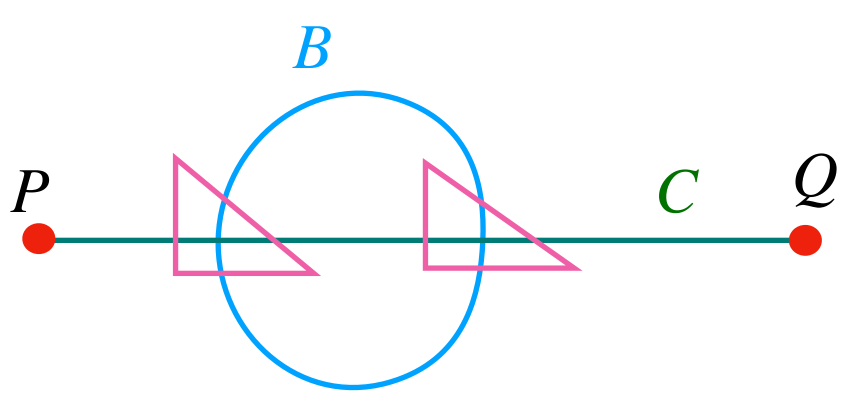
Sample Question:
Is there a curve in F connecting boundary points P and Q?

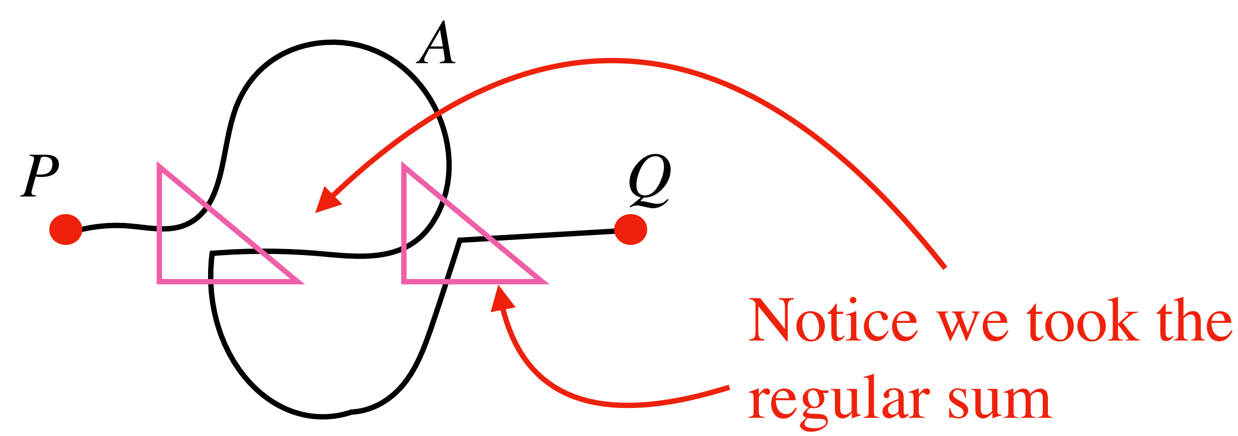


To show the algorithm does what it claims

If there is a curve connecting P and Q then there is a *Fundamental* Normal Curve with the same property.

Proof. Normalization gives a normal curve A with endpoints P and Q. We show that we can find such a normal curve A that is fundamental.
Consider the integer vector **A** associated to this curve. Pick **A** to have smallest weight among all normal curves from P to Q.

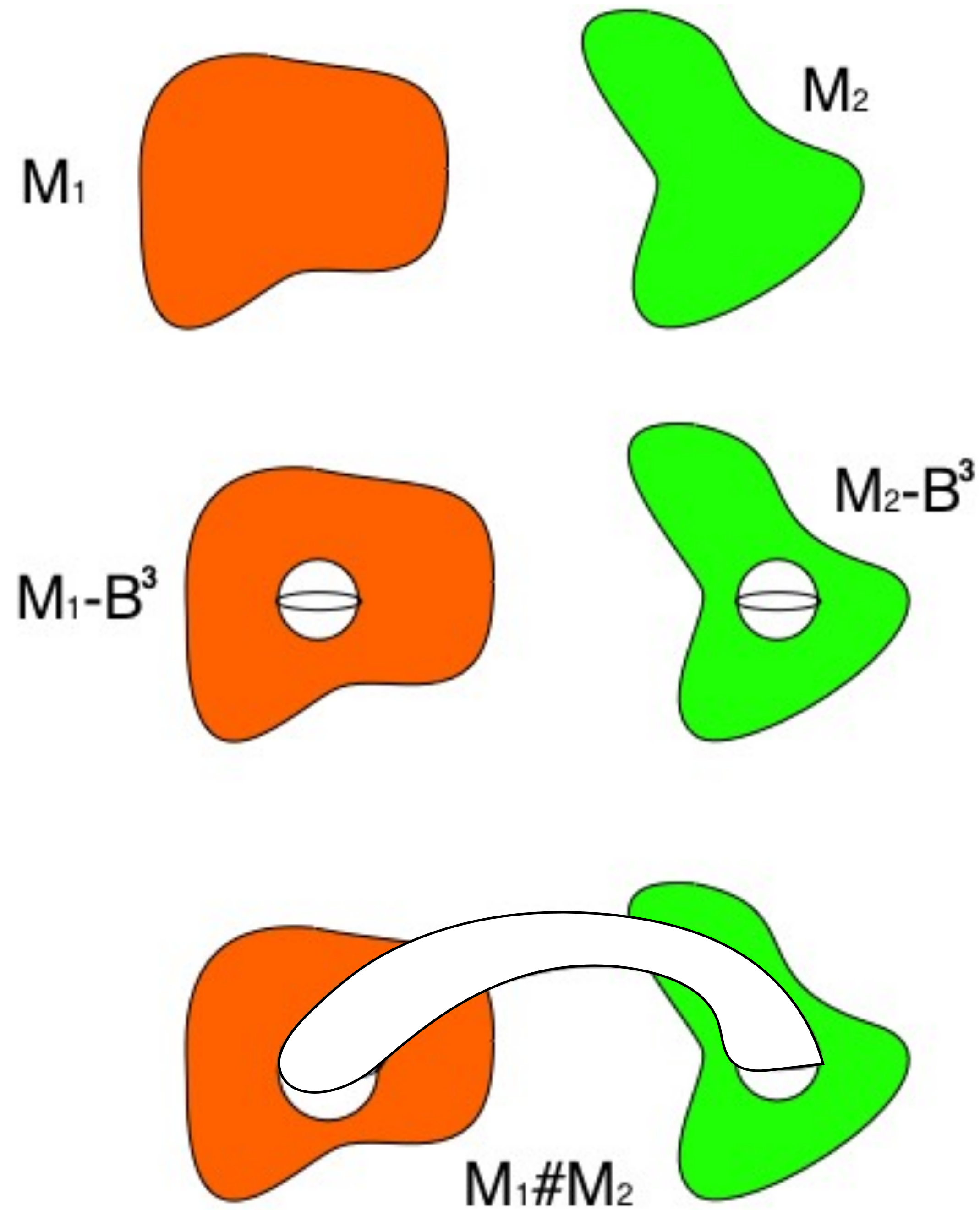




Claim. **A** is fundamental.
Suppose for contradiction that **A** = **B** + **C**. Then one of **B** or **C** has boundary at the points P and Q. Moreover wt(**A**) = wt(**B**) + wt(**C**)**,** so each of **B** and **C** has smaller weight than **A**. This contradicts the assumption that **A** has smallest weight.

From Curves to Surfaces

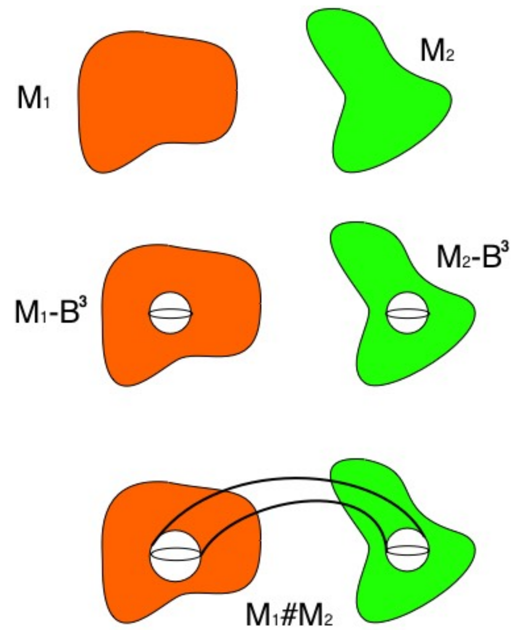
Certain surface types arise in the process of cutting a 3-manifold into simpler pieces.



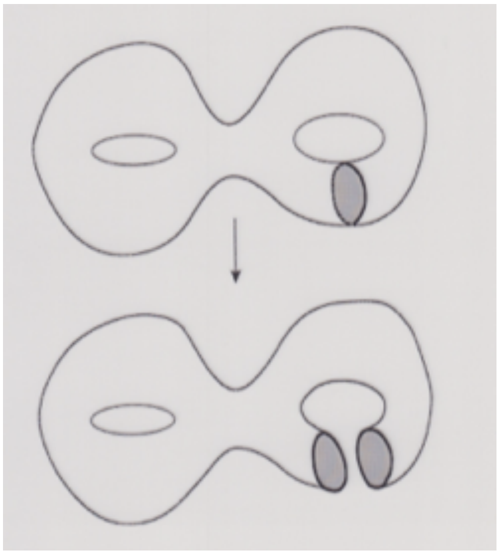
Cutting along **2-spheres** gives 3-manifolds that are *prime*.

Surfaces in 3-Manifolds

Certain surface types arise in the process of cutting a 3-manifold into simpler pieces.



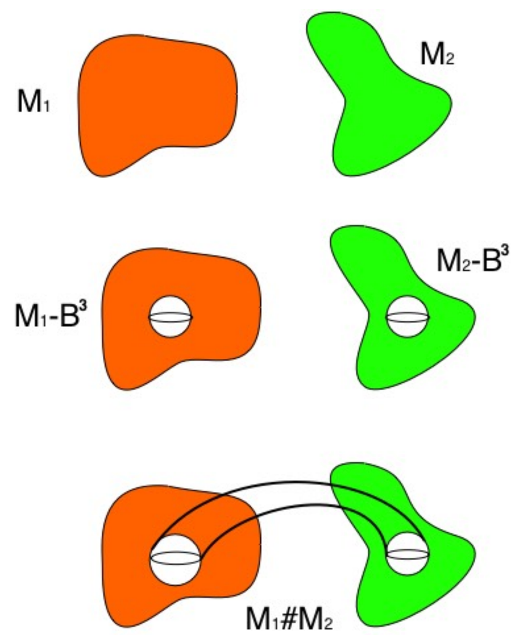
Cutting along **spheres** gives 3-manifolds that are prime.



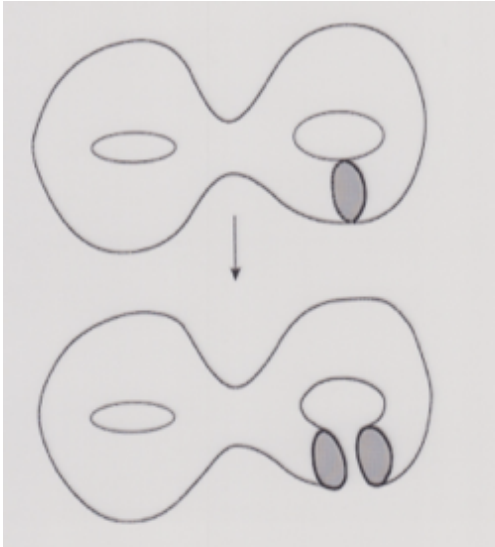
Cutting along **disks** gives pieces with simpler boundary.

Surfaces in 3-Manifolds

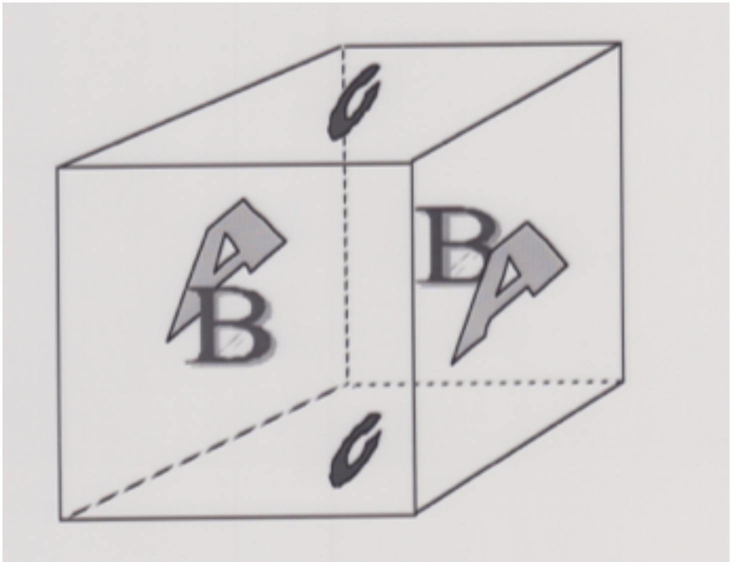
Certain surface types arise in the process of cutting a 3-manifold into simpler pieces.



Cutting along **2-spheres** gives 3-manifolds that are prime.



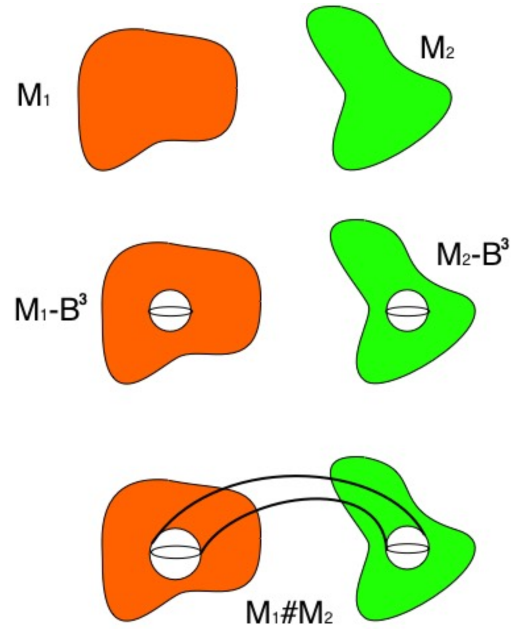
Cutting along **disks** gives pieces with simpler boundary.



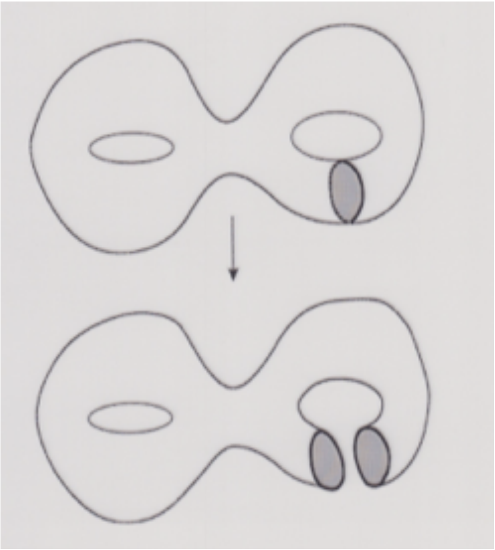
Cutting along **incompressible surfaces** eventually cuts a manifold into 3-balls.

Surfaces in 3-Manifolds

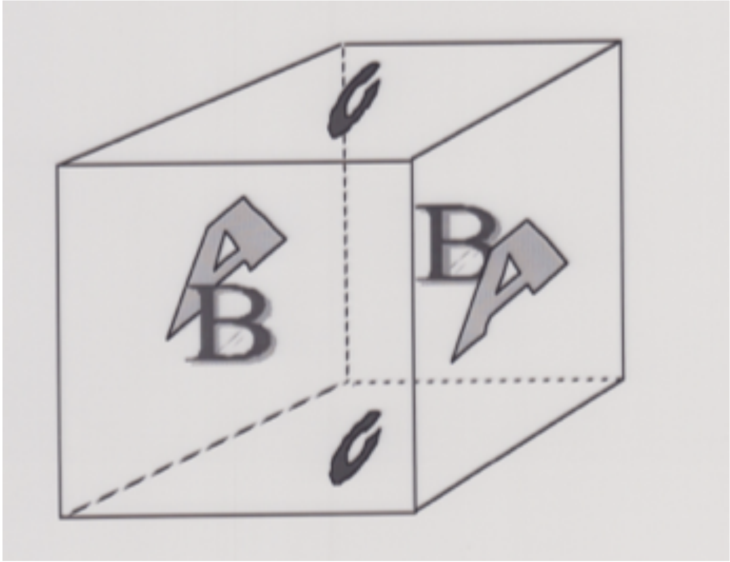
Certain surface types arise in the process of cutting a 3-manifold into simpler pieces.



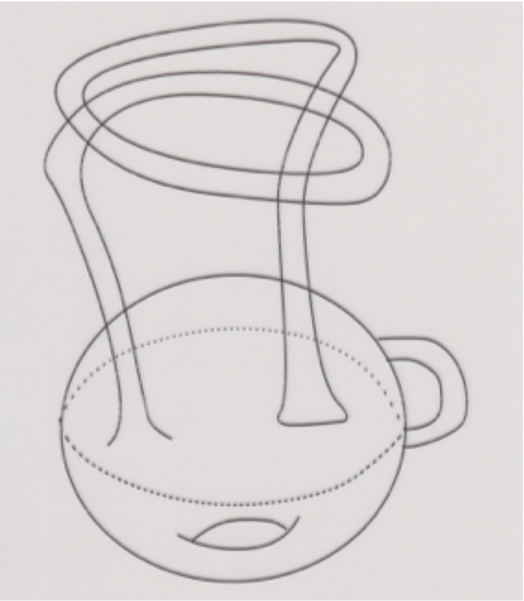
Cutting along **spheres** gives 3-manifolds that are prime.



Cutting along **compressing disks** gives pieces with simpler boundary.



Cutting along **incompressible surfaces** eventually cuts a manifold into 3-balls.



Don't do this!

Cutting along **compressible surfaces** can make a 3-manifold more complicated.

Normal Surfaces in 3-Manifolds

Surfaces can be complicated.

Normal surfaces give an efficient method to describe surfaces in 3-manifolds.

As with curves, the surfaces that most interest topologists can be deformed into normal surfaces.

Normal Surfaces

Definition. Take a manifold *M* with a fixed triangulation.
A *normal surface* is a surface that intersects each tetrahedron in a finite collection of disjoint triangles and quadrilaterals