The character table for E_8

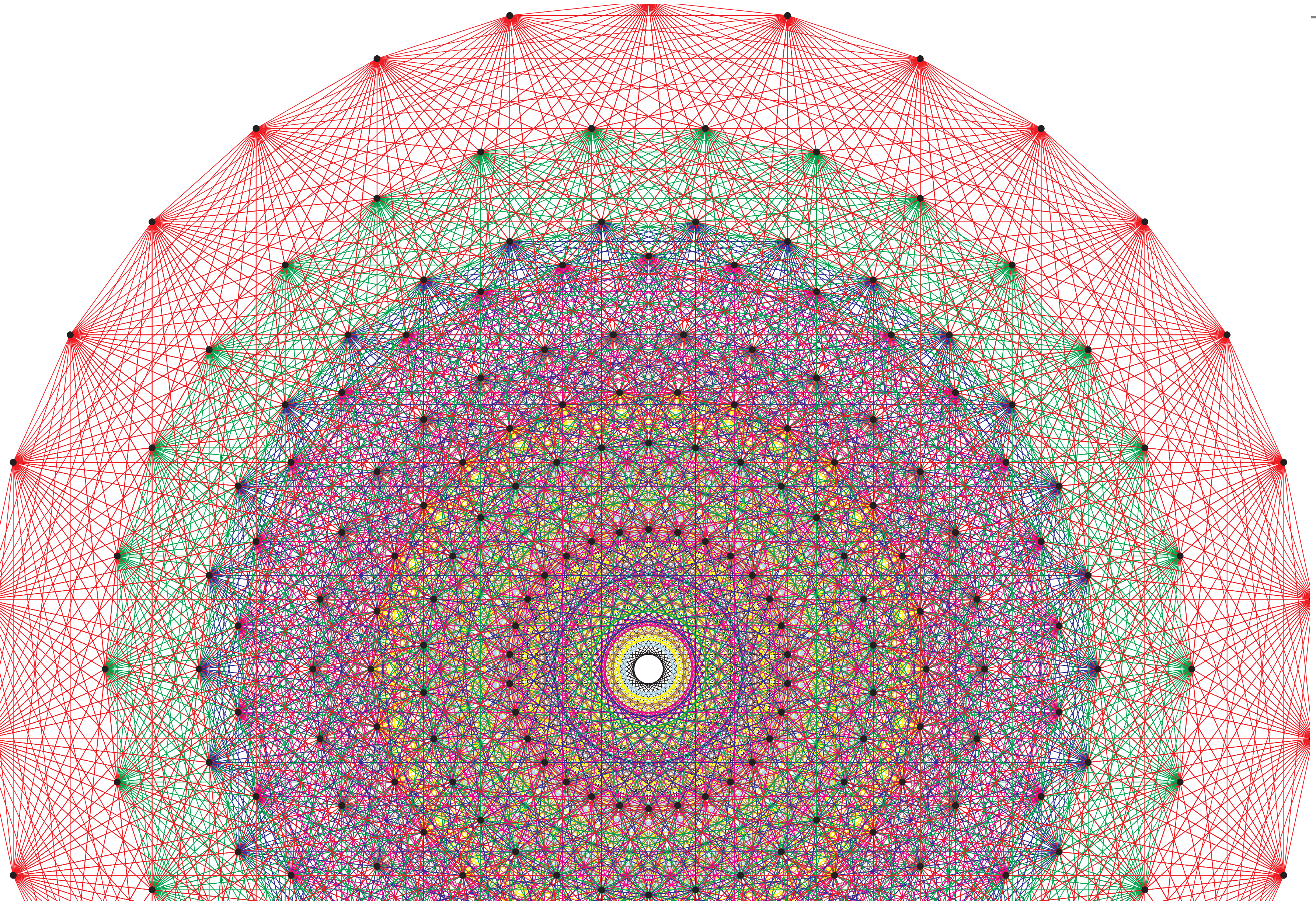
or

how we wrote down
a 453060×453060 ***matrix***
and found happiness

David Vogan

Department of Mathematics, MIT

Root system of E_8



The Atlas members:

Jeffrey Adams
Dan Barbasch
Birne Binegar
Bill Casselman
Dan Ciubotaru
Fokko du Cloux
Scott Crofts
Tatiana Howard
Marc van Leeuwen
Alfred Noel
Alessandra Pantano
Annegret Paul
Siddhartha Sahi
Susana Salamanca
John Stembridge
Peter Trapa
David Vogan
Wai-Ling Yee
Jiu-Kang Yu

American Institute of Mathematics `www.aimath.org`
National Science Foundation `www.nsf.gov`

`www.liegroups.org`

The Atlas members:

The story in code:

At 9 a.m. on January 8, 2007, a computer finished writing sixty gigabytes of files: Kazhdan-Lusztig polynomials for the split real group $G(\mathbb{R})$ of type E_8. Their values at 1 are coefficients in irreducible characters of $G(\mathbb{R})$. The biggest coefficient was 11,808,808, in

$$152q^{22} + 3472q^{21} + 38791q^{20} + 293021q^{19}$$
$$+ 1370892q^{18} + 4067059q^{17} + 7964012q^{16} + 11159003q^{15}$$
$$+ 11808808q^{14} + 9859915q^{13} + 6778956q^{12} + 3964369q^{11}$$
$$+ 2015441q^{10} + 906567q^{9} + 363611q^{8} + 129820q^{7}$$
$$+ 41239q^{6} + 11426q^{5} + 2677q^{4} + 492q^{3} + 61q^{2} + 3q$$

Its value at 1 is 60,779,787.

Questions you might want to ask:

Questions you might want to ask:

- Mathematicians don't look at single examples (in public). Why E_8?

Questions you might want to ask:

- Mathematicians don't look at single examples (in public). Why E_8?
- What *is* E_8 anyway?

Questions you might want to ask:

- Mathematicians don't look at single examples (in public). Why E_8?
- What *is* E_8 anyway?
- What's a character table?

Questions you might want to ask:

- Mathematicians don't look at single examples (in public). Why E_8?
- What *is* E_8 anyway?
- What's a character table?
- Sixty gigabytes? Which byte do I care about?

Questions you might want to ask:

- Mathematicians don't look at single examples (in public). Why E_8?
- What *is* E_8 anyway?
- What's a character table?
- Sixty gigabytes? Which byte do I care about?

Questions you might want to ask:

- Mathematicians don't look at single examples (in public). Why E_8?
- What *is* E_8 anyway?
- What's a character table?
- Sixty gigabytes? Which byte do I care about?

Excellent questions. Since it's my talk, I get to rephrase them a little.

Questions I *want* you to ask:

Questions I *want* you to ask:

- **What's a Lie group?**

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.
- **Which one is E_8?**

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.
- **Which one is E_8?**
  - The one for the icosahedron.

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.
- **Which one is E_8?**
  - The one for the icosahedron.
- **What's a group representation?**

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.
- **Which one is E_8?**
  - The one for the icosahedron.
- **What's a group representation?**
  - A way to change under symmetry.

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.
- **Which one is E_8?**
  - The one for the icosahedron.
- **What's a group representation?**
  - A way to change under symmetry.
- **What's a character table?**

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.
- **Which one is E_8?**
  - The one for the icosahedron.
- **What's a group representation?**
  - A way to change under symmetry.
- **What's a character table?**
  - A description of all the representations.

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.
- **Which one is E_8?**
  - The one for the icosahedron.
- **What's a group representation?**
  - A way to change under symmetry.
- **What's a character table?**
  - A description of all the representations.
- **How do you write a character table?**

Questions I *want* you to ask:

- **What's a Lie group?**
  - A continuous family of symmetries.
- **How many Lie groups are there?**
  - One for every regular polyhedron.
- **Which one is E_8?**
  - The one for the icosahedron.
- **What's a group representation?**
  - A way to change under symmetry.
- **What's a character table?**
  - A description of all the representations.
- **How do you write a character table?**
  - RTFM (by Weyl, Harish-Chandra, Kazhdan/Lusztig).

Our Contribution

Our Contribution

- **So what did you guys do exactly?**

Our Contribution

- **So what did you guys do exactly?**
  - We read TFM.

Our Contribution

- **So what did you guys do exactly?**
  - We read TFM.

Here are longer versions of those answers.

What's a Lie group?

A continuous family of symmetries.

What's a Lie group?

A continuous family of symmetries.

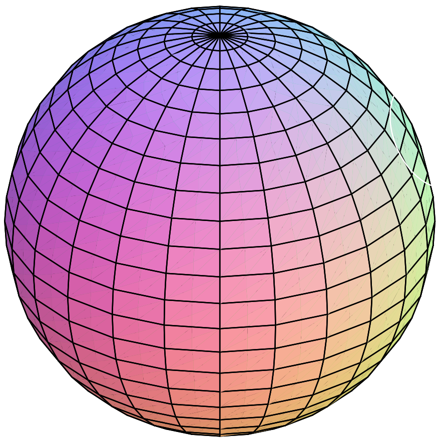
Example. Rotations of the sphere

What's a Lie group?

A continuous family of symmetries.

Example. Rotations of the sphere

To make a rotation of a two-dimensional sphere, pick

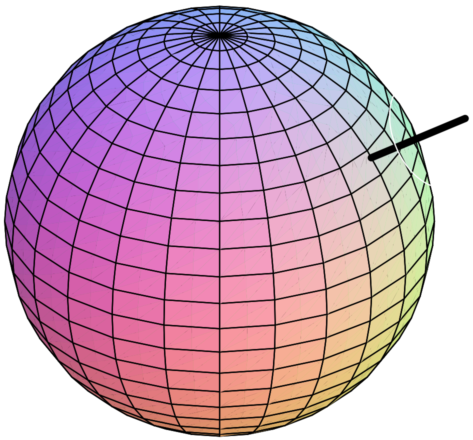


What's a Lie group?

A continuous family of symmetries.

Example. Rotations of the sphere

To make a rotation of a two-dimensional sphere, pick



- axis of rotation

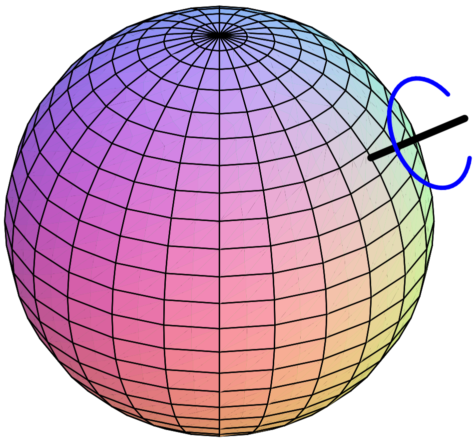
  (2-diml choice: point on sphere)

What's a Lie group?

A continuous family of symmetries.

Example. Rotations of the sphere

To make a rotation of a two-dimensional sphere, pick



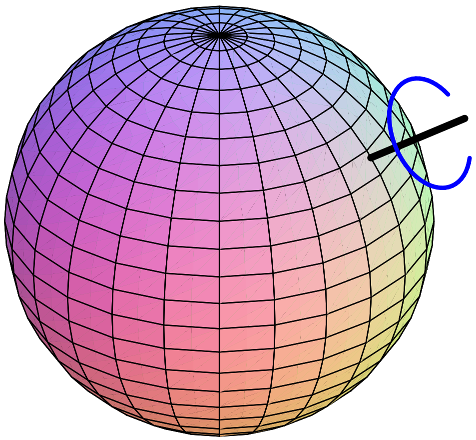
- axis of rotation
  (2-diml choice: point on sphere)
- angle of rotation
  (1-diml choice: $0°$–$360°$)

What's a Lie group?

A continuous family of symmetries.

Example. Rotations of the sphere

To make a rotation of a two-dimensional sphere, pick



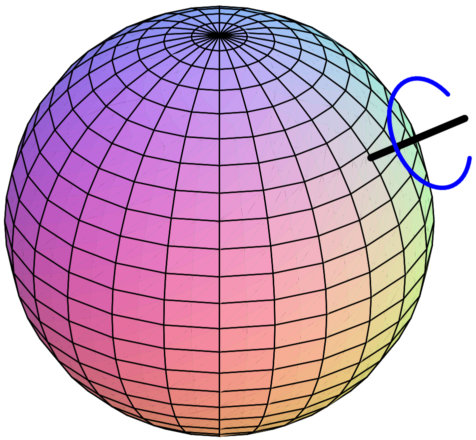
- axis of rotation
  (2-diml choice: point on sphere)
- angle of rotation
  (1-diml choice: $0°$–$360°$)

Altogether that's three dimensions of choices. Rotations of the sphere make a three-dimensional Lie group.

What's a Lie group?

A continuous family of symmetries.

Example. Rotations of the sphere
To make a rotation of a two-dimensional sphere, pick



- axis of rotation
  (2-diml choice: point on sphere)
- angle of rotation
  (1-diml choice: $0°$–$360°$)

Altogether that's three dimensions of choices. Rotations of the sphere make a three-dimensional Lie group.

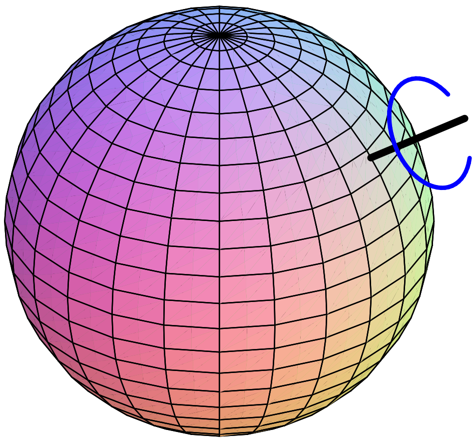
Representations of this group $\leftrightsquigarrow$ periodic table.

What's a Lie group?

A continuous family of symmetries.

Example. Rotations of the sphere

To make a rotation of a two-dimensional sphere, pick



- axis of rotation
  (2-diml choice: point on sphere)
- angle of rotation
  (1-diml choice: $0°$–$360°$)

Altogether that's three dimensions of choices. Rotations of the sphere make a three-dimensional Lie group.

Representations of this group $\leftrightsquigarrow$ periodic table.

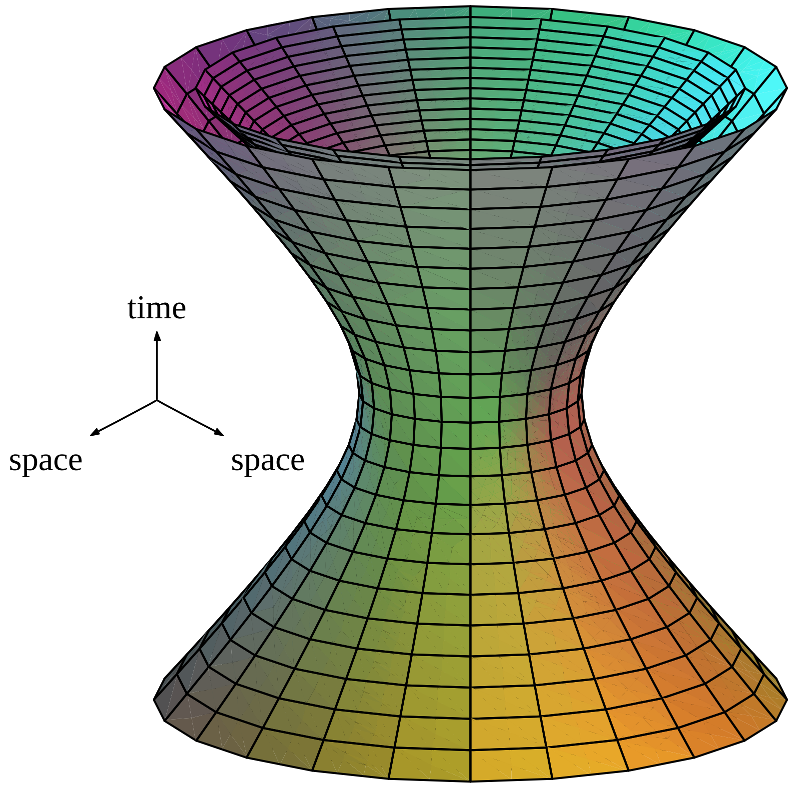
Other groups $\leftrightsquigarrow$ other geometries, other physics. . .

The Lorentz group

Special relativity concerns a different geometry. . .

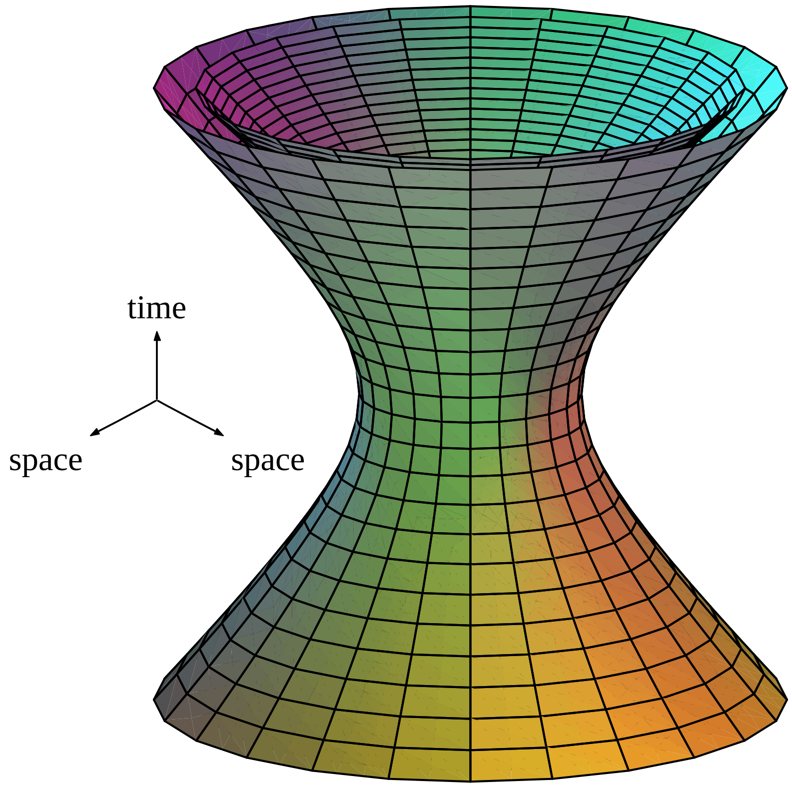
The Lorentz group

Special relativity concerns a different geometry. . .



The Lorentz group

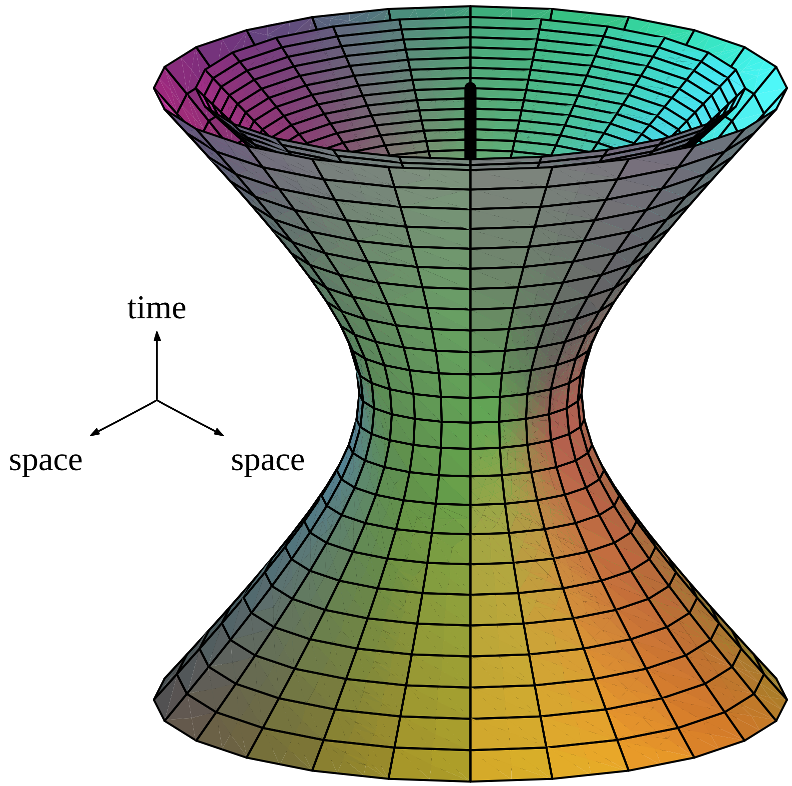
Special relativity concerns a different geometry. . .



Two essentially different kinds of symmetry:

The Lorentz group

Special relativity concerns a different geometry. . .

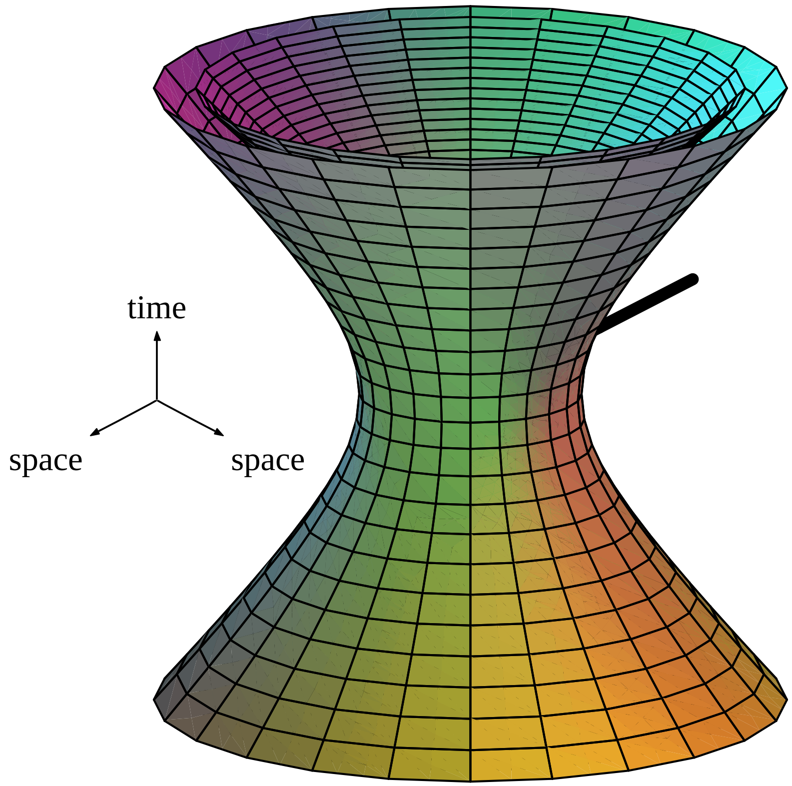


Two essentially different kinds of symmetry:

rotation around time-like vector

The Lorentz group

Special relativity concerns a different geometry. . .



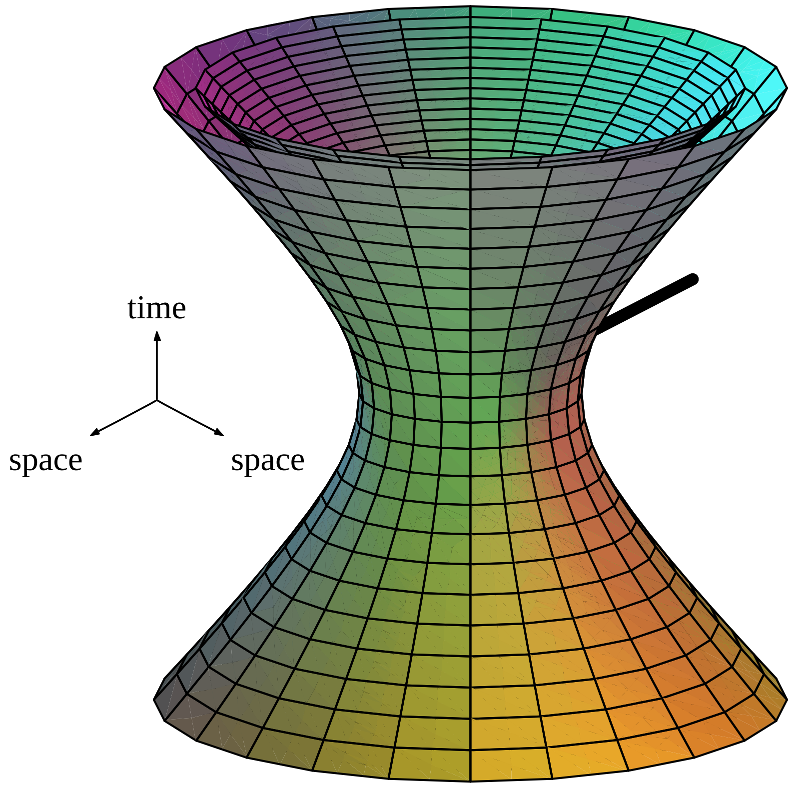
Two essentially different kinds of symmetry:

rotation around time-like vector

Lorentz boost around space-like vector

The Lorentz group

Special relativity concerns a different geometry. . .



Two essentially different kinds of symmetry:

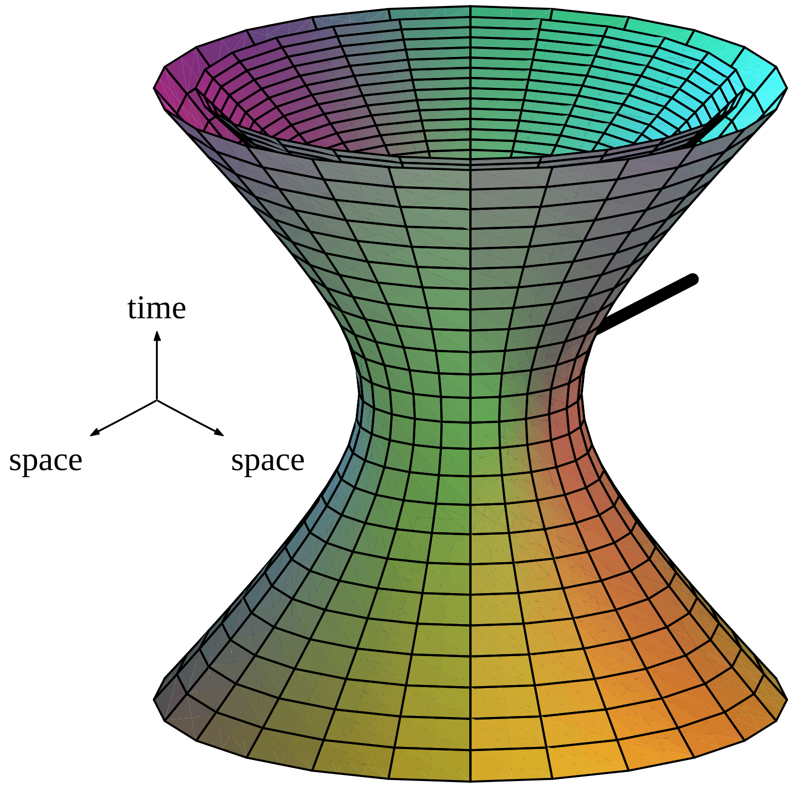
rotation around time-like vector

Lorentz boost around space-like vector

The Lorentz group is another three-dimensional group: a noncompact form of the rotation group.

The Lorentz group

Special relativity concerns a different geometry. . .



Two essentially different kinds of symmetry:

rotation around time-like vector

Lorentz boost around space-like vector

The Lorentz group is another three-dimensional group: a noncompact form of the rotation group.

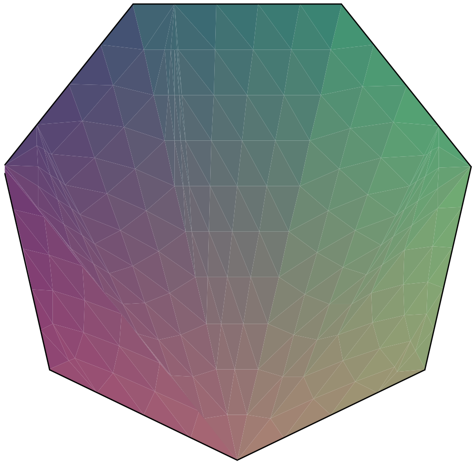
Representations $\leftrightsquigarrow$ relativistic physics.

How many Lie groups are there?

One for every regular polyhedron.

How many Lie groups are there?

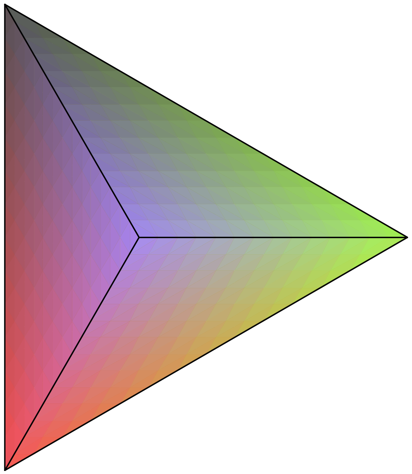
One for every regular polyhedron.



- 2D polygons: classical groups.

How many Lie groups are there?

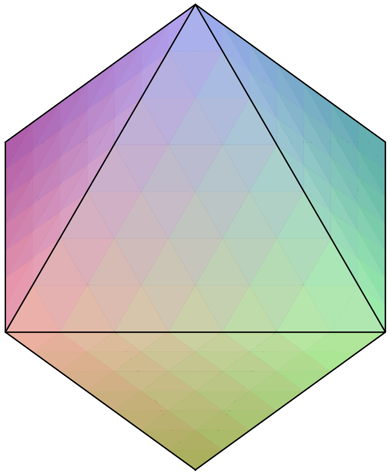
One for every regular polyhedron.



- 2D polygons: classical groups.
- Tetrahedron: E_6, dimension 78.

How many Lie groups are there?

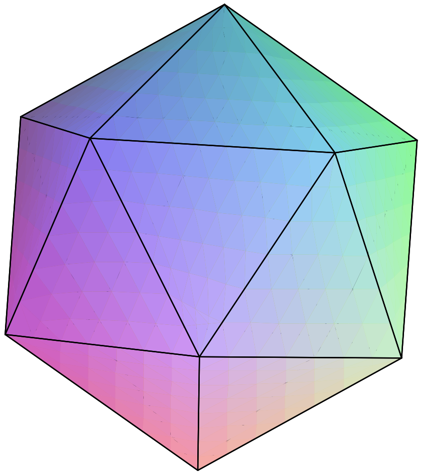
One for every regular polyhedron.



- 2D polygons: classical groups.
- Tetrahedron: E_6, dimension 78.
- Octahedron: E_7, dimension 133.

How many Lie groups are there?

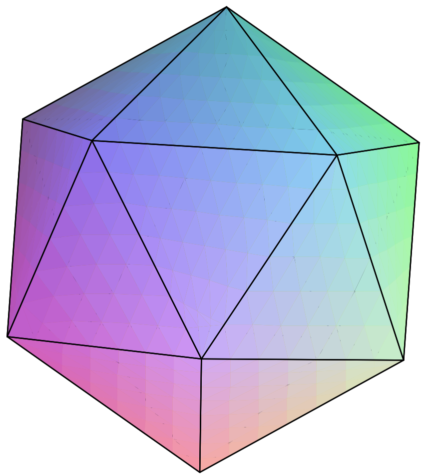
One for every regular polyhedron.



- 2D polygons: classical groups.
- Tetrahedron: E_6, dimension 78.
- Octahedron: E_7, dimension 133.
- Icosahedron: E_8, dimension 248.

How many Lie groups are there?

One for every regular polyhedron.

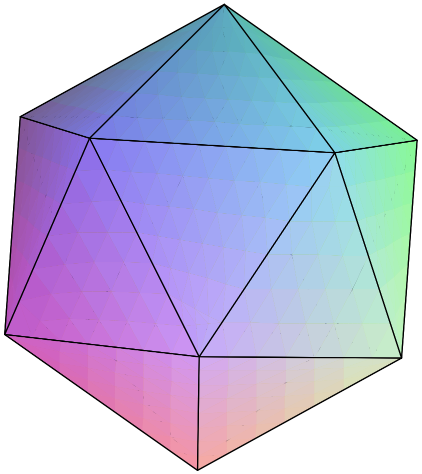


- 2D polygons: classical groups.
- Tetrahedron: E_6, dimension 78.
- Octahedron: E_7, dimension 133.
- Icosahedron: E_8, dimension 248.

Actually it's quite a bit more complicated.

How many Lie groups are there?

One for every regular polyhedron.



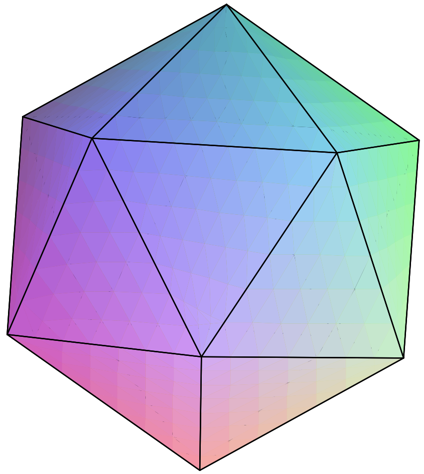
- 2D polygons: classical groups.
- Tetrahedron: E_6, dimension 78.
- Octahedron: E_7, dimension 133.
- Icosahedron: E_8, dimension 248.

Actually it's quite a bit more complicated.

- Several Lie groups for each regular polyhedron.

How many Lie groups are there?

One for every regular polyhedron.



- 2D polygons: classical groups.
- Tetrahedron: E_6, dimension 78.
- Octahedron: E_7, dimension 133.
- Icosahedron: E_8, dimension 248.

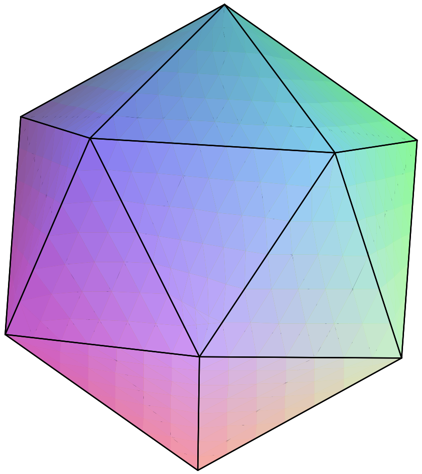
Actually it's quite a bit more complicated.

- Several Lie groups for each regular polyhedron.

  Rotation group and Lorentz group both correspond to 1-gon.

How many Lie groups are there?

One for every regular polyhedron.



- 2D polygons: classical groups.
- Tetrahedron: E_6, dimension 78.
- Octahedron: E_7, dimension 133.
- Icosahedron: E_8, dimension 248.

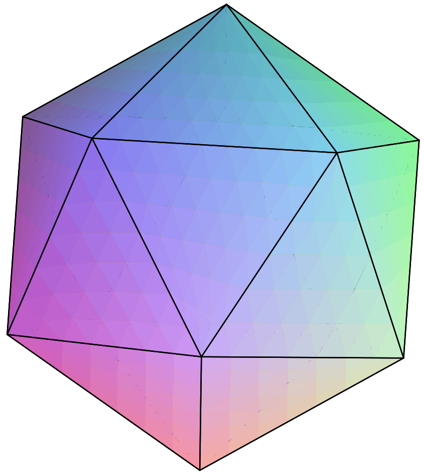
Actually it's quite a bit more complicated.

- Several Lie groups for each regular polyhedron.

  Rotation group and Lorentz group both correspond to 1-gon.
- Get only simple Lie groups in this way.

How many Lie groups are there?

One for every regular polyhedron.



- 2D polygons: classical groups.
- Tetrahedron: E_6, dimension 78.
- Octahedron: E_7, dimension 133.
- Icosahedron: E_8, dimension 248.

Actually it's quite a bit more complicated.

- Several Lie groups for each regular polyhedron.

  Rotation group and Lorentz group both correspond to 1-gon.
- Get only simple Lie groups in this way.
- Building general Lie groups from simple is hard.

Which one is E_8?

The one for the icosahedron.

Which one is E_8?

The one for the icosahedron.

There are three different groups called E_8, each one 248-dimensional and wonderfully complicated.

Which one is E_8?

The one for the icosahedron.

There are three different groups called E_8, each one 248-dimensional and wonderfully complicated.

- Compact E_8. Characters computed by Weyl in 1925.

Which one is E_8?

The one for the icosahedron.

There are three different groups called E_8, each one 248-dimensional and wonderfully complicated.

- Compact E_8. Characters computed by Weyl in 1925.

  In atlas shorthand, encoded by $\begin{pmatrix} 1 \end{pmatrix}$.

Which one is E_8?

The one for the icosahedron.

There are three different groups called E_8, each one 248-dimensional and wonderfully complicated.

- Compact E_8. Characters computed by Weyl in 1925.

  In atlas shorthand, encoded by $\begin{pmatrix}1\end{pmatrix}$.

  (Which hides deep and wonderful work by Weyl.)

Which one is E_8?

The one for the icosahedron.

There are three different groups called E_8, each one 248-dimensional and wonderfully complicated.

- Compact E_8. Characters computed by Weyl in 1925.

  In atlas shorthand, encoded by $\begin{pmatrix} 1 \end{pmatrix}$.

  (Which hides deep and wonderful work by Weyl.)
- Quaternionic E_8. Characters computed in 2005.

Which one is E_8?

The one for the icosahedron.

There are three different groups called E_8, each one 248-dimensional and wonderfully complicated.

- Compact E_8. Characters computed by Weyl in 1925.

  In atlas shorthand, encoded by $\begin{pmatrix} 1 \end{pmatrix}$.

  (Which hides deep and wonderful work by Weyl.)

- Quaternionic E_8. Characters computed in 2005.

  In atlas shorthand, a 73410×73410 matrix. One entry:

$$3q^{13}+30q^{12}+190q^{11}+682q^{10}+1547q^{9}+2364q^{8}+2545q^{7}$$
$$+2031q^{6}+1237q^{5}+585q^{4}+216q^{3}+60q^{2}+11q+1$$

Which one is E_8?

The one for the icosahedron.

There are three different groups called E_8, each one 248-dimensional and wonderfully complicated.

- Compact E_8. Characters computed by Weyl in 1925.

  In atlas shorthand, encoded by $\begin{pmatrix} 1 \end{pmatrix}$.

  (Which hides deep and wonderful work by Weyl.)
- Quaternionic E_8. Characters computed in 2005.

  In atlas shorthand, a 73410×73410 matrix. One entry:

  $$3q^{13}+30q^{12}+190q^{11}+682q^{10}+1547q^{9}+2364q^{8}+2545q^{7}$$
  $$+2031q^{6}+1237q^{5}+585q^{4}+216q^{3}+60q^{2}+11q+1$$
- Split E_8. This is the tough one.

What's a group representation?

A way to change under symmetry.

What’s a group representation?

A way to change under symmetry.

This time what we do is actually *less* complicated.

What's a group representation?

A way to change under symmetry.

This time what we do is actually *less* complicated.

We look for irreducible representations: simplest possible ways to change under symmetry.

What's a group representation?

A way to change under symmetry.

This time what we do is actually *less* complicated.

We look for irreducible representations: simplest possible ways to change under symmetry.

Irreducible representations are like atoms in chemistry. Knowing the atoms doesn't tell you all the molecules you can build from those atoms.

What's a group representation?

A way to change under symmetry.

This time what we do is actually *less* complicated.

We look for irreducible representations: simplest possible ways to change under symmetry.

Irreducible representations are like atoms in chemistry. Knowing the atoms doesn't tell you all the molecules you can build from those atoms.

But knowing the atoms is a good place to start.

What's a group representation?

A way to change under symmetry.

This time what we do is actually *less* complicated.

We look for irreducible representations: simplest possible ways to change under symmetry.

Irreducible representations are like atoms in chemistry. Knowing the atoms doesn't tell you all the molecules you can build from those atoms.

But knowing the atoms is a good place to start.

First Lie group is 1-dimensional: symmetry in time.

Repns of time symmetry

Repns of time symmetry

Means all possible ways to change in time: hard.

Repns of time symmetry

Means all possible ways to change in time: hard.

Irreducible repns are simplest ways to change. . .

Repns of time symmetry

Means all possible ways to change in time: hard.

Irreducible repns are simplest ways to change. . .

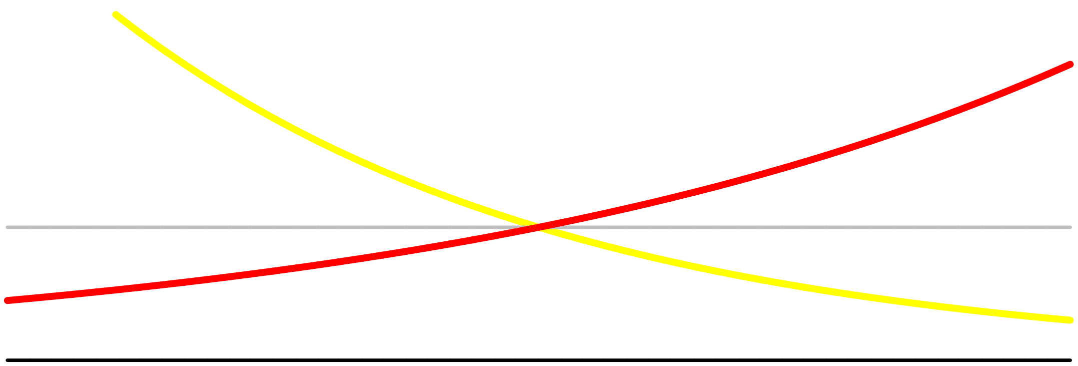
- No change: trivial representation.

Repns of time symmetry

Means all possible ways to change in time: hard.

Irreducible repns are simplest ways to change. . .

- No change: trivial representation.
- Exponential growth or decay.

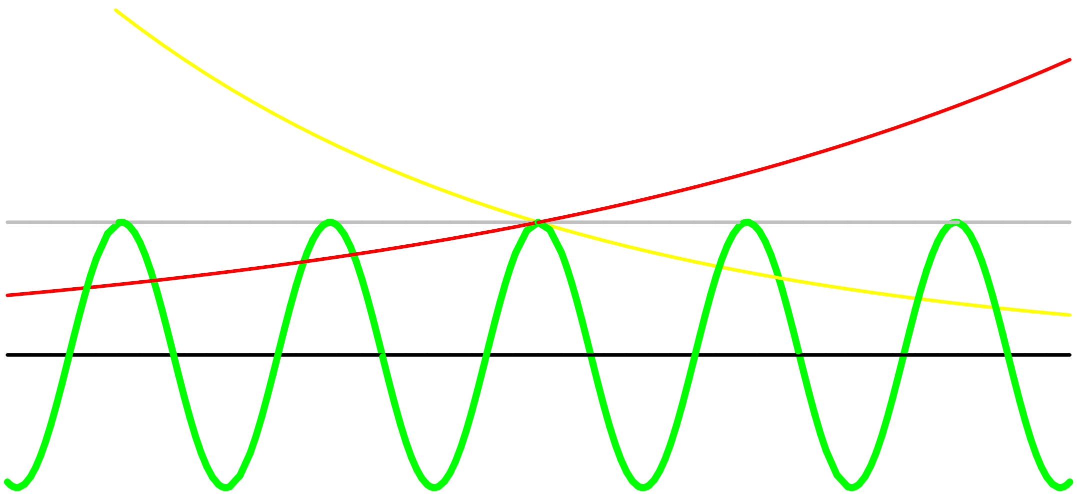


Repns of time symmetry

Means all possible ways to change in time: hard.

Irreducible repns are simplest ways to change. . .

- No change: trivial representation.
- Exponential growth or decay.
- Oscillation.

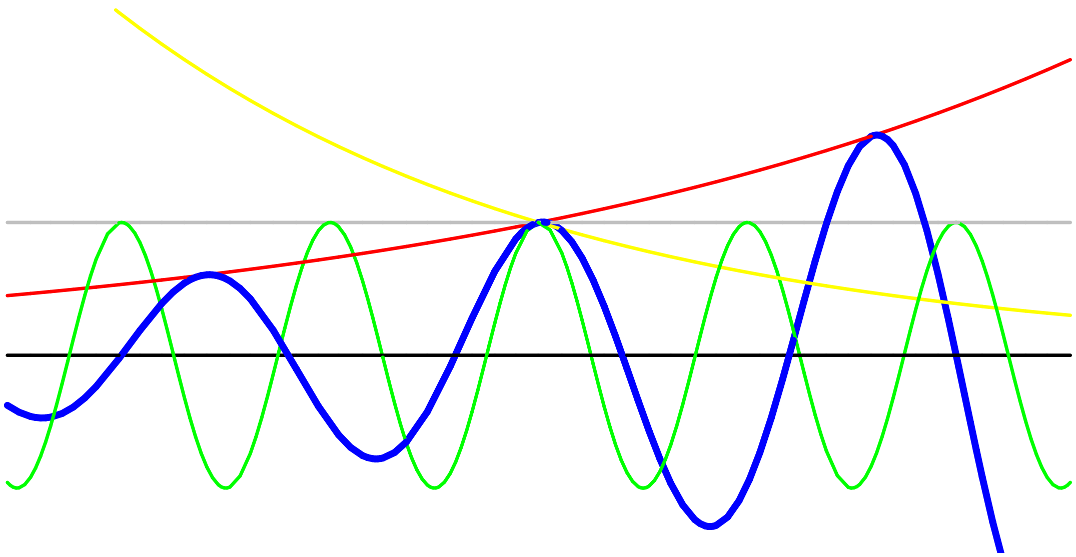


Repns of time symmetry

Means all possible ways to change in time: hard.

Irreducible repns are simplest ways to change. . .

- No change: trivial representation.
- Exponential growth or decay.
- Oscillation.
- Exponentially growing or decaying oscillation.

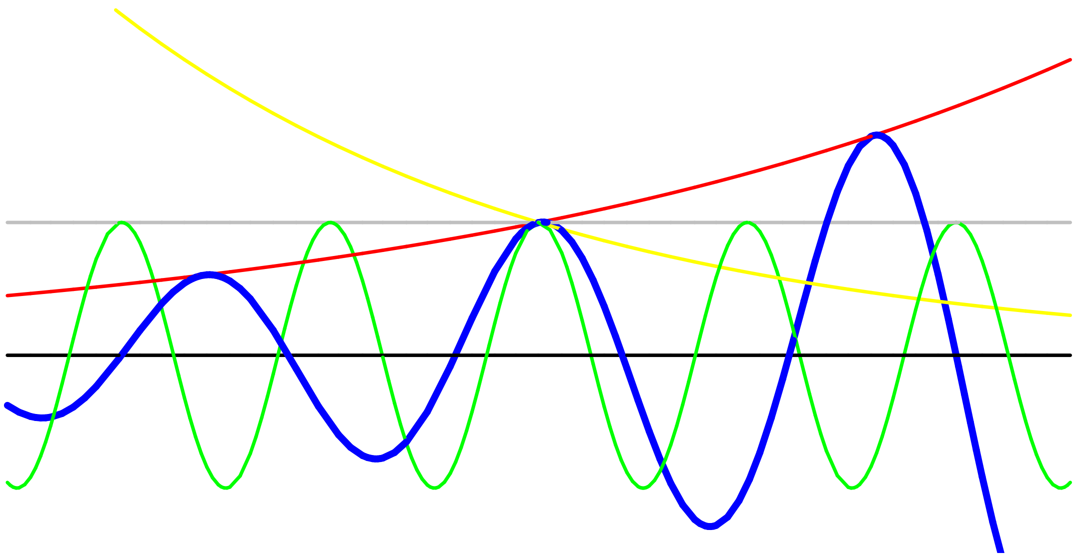


Repns of time symmetry

Means all possible ways to change in time: hard.

Irreducible repns are simplest ways to change. . .

- No change: trivial representation.
- Exponential growth or decay.
- Oscillation.
- Exponentially growing or decaying oscillation.



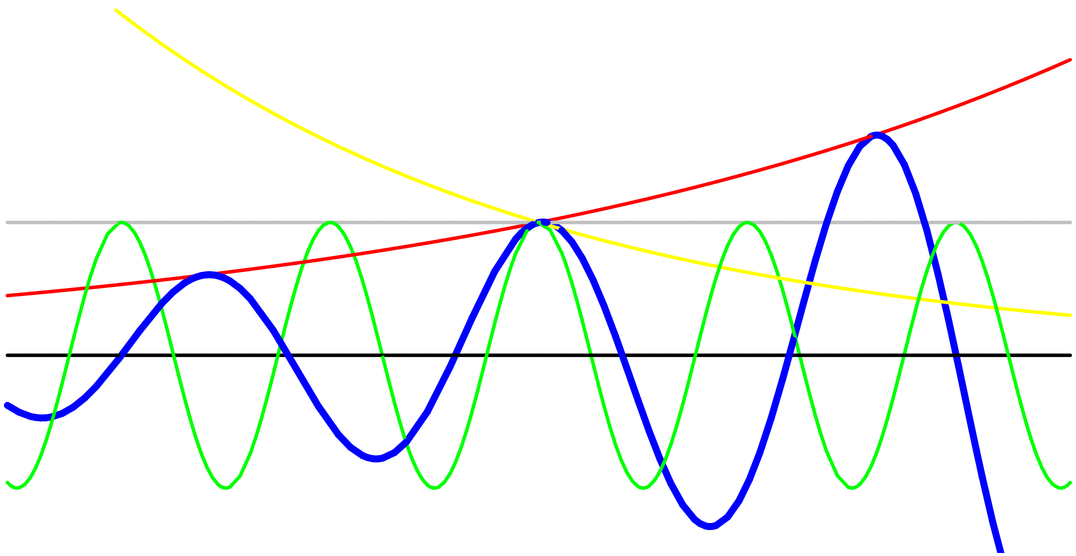
That's all the irreducible representations for time symmetry. Given by two real numbers: growth rate, frequency.

Repns of time symmetry

Means all possible ways to change in time: hard.

Irreducible repns are simplest ways to change. . .

- No change: trivial representation.
- Exponential growth or decay.
- Oscillation.
- Exponentially growing or decaying oscillation.



That's all the irreducible representations for time symmetry. Given by two real numbers: growth rate, frequency.

$$\frac{df}{dt} = z \cdot f$$

Repns of compact time symmetry

Repns of compact time symmetry

Time symmetry is *not* the easiest Lie group. Simplest is time symmetries repeating after unit time.

Repns of compact time symmetry

Time symmetry is *not* the easiest Lie group. Simplest is time symmetries repeating after unit time.

Technical term is compact.

Repns of compact time symmetry

Time symmetry is *not* the easiest Lie group. Simplest is time symmetries repeating after unit time.

Technical term is compact.

Irreducible representations are simplest kinds of change repeating after unit time. Examples:

Repns of compact time symmetry

Time symmetry is *not* the easiest Lie group. Simplest is time symmetries repeating after unit time.

Technical term is compact.

Irreducible representations are simplest kinds of change repeating after unit time. Examples:

- No change: trivial representation.

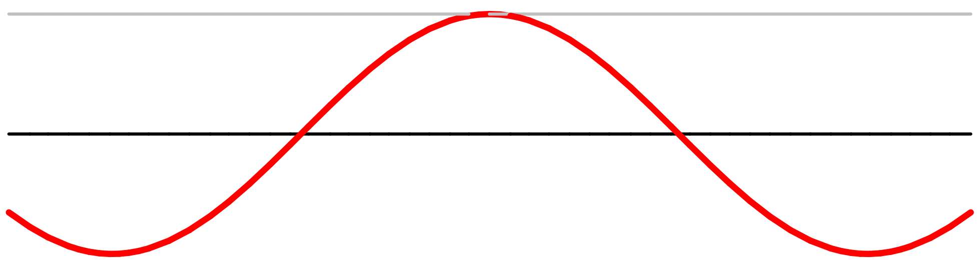
Repns of compact time symmetry

Time symmetry is *not* the easiest Lie group. Simplest is time symmetries repeating after unit time.

Technical term is compact.

Irreducible representations are simplest kinds of change repeating after unit time. Examples:

- No change: trivial representation.
- Oscillation with frequency $F = 1$



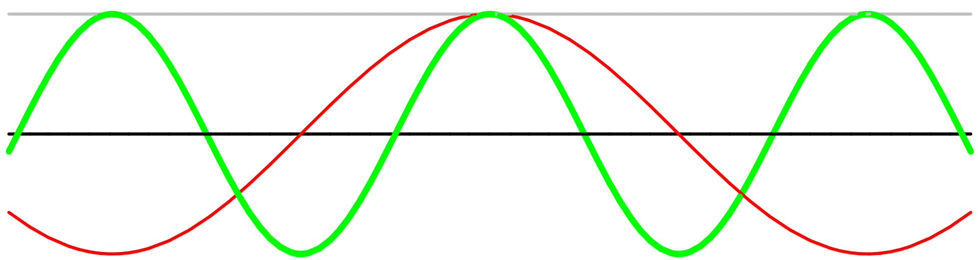
Repns of compact time symmetry

Time symmetry is *not* the easiest Lie group. Simplest is time symmetries repeating after unit time.

Technical term is compact.

Irreducible representations are simplest kinds of change repeating after unit time. Examples:

- No change: trivial representation.
- Oscillation with frequency $F = 1$ or 2



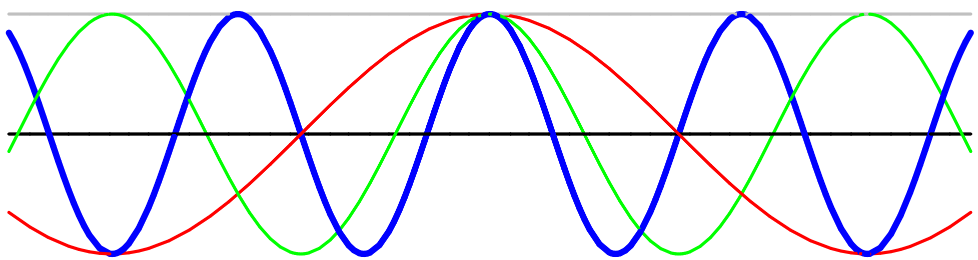
Repns of compact time symmetry

Time symmetry is *not* the easiest Lie group. Simplest is time symmetries repeating after unit time.

Technical term is compact.

Irreducible representations are simplest kinds of change repeating after unit time. Examples:

- No change: trivial representation.
- Oscillation with frequency $F = 1$ or 2 or $3 \ldots$



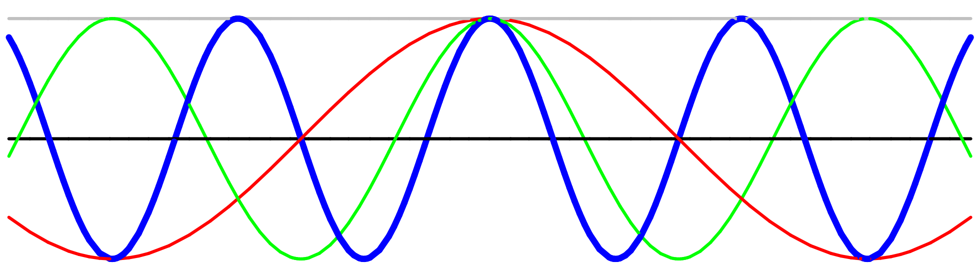
Repns of compact time symmetry

Time symmetry is *not* the easiest Lie group. Simplest is time symmetries repeating after unit time.

Technical term is compact.

Irreducible representations are simplest kinds of change repeating after unit time. Examples:

- No change: trivial representation.
- Oscillation with frequency $F = 1$ or 2 or $3 \ldots$



That's all the irreducible repns for compact time symmetry. Given by one integer: frequency.

Repns of rotation group

Repns of rotation group

Next simplest Lie group is rotations of the sphere.

Repns of rotation group

Next simplest Lie group is rotations of the sphere.

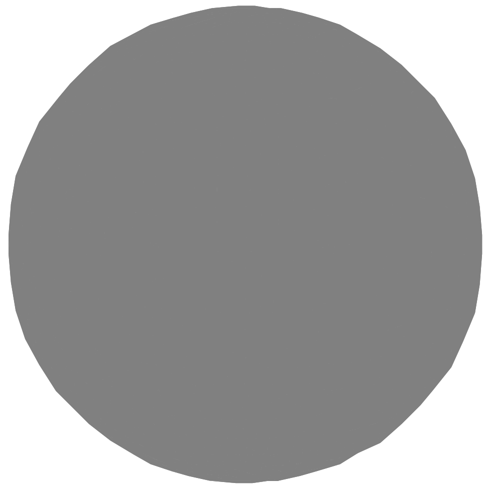
Irreducible representations of rotation group are simplest ways to change with rotation. Examples:

Repns of rotation group

Next simplest Lie group is rotations of the sphere.

Irreducible representations of rotation group are simplest ways to change with rotation. Examples:

- No change: trivial repn.

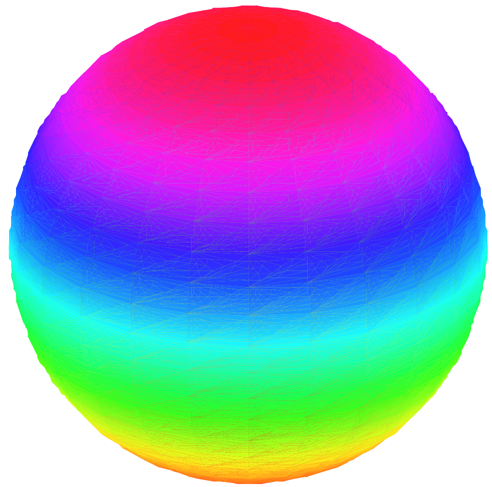


Repns of rotation group

Next simplest Lie group is rotations of the sphere.

Irreducible representations of rotation group are simplest ways to change with rotation. Examples:

- No change: trivial repn.
- Oscillation with freq $F = 1$.



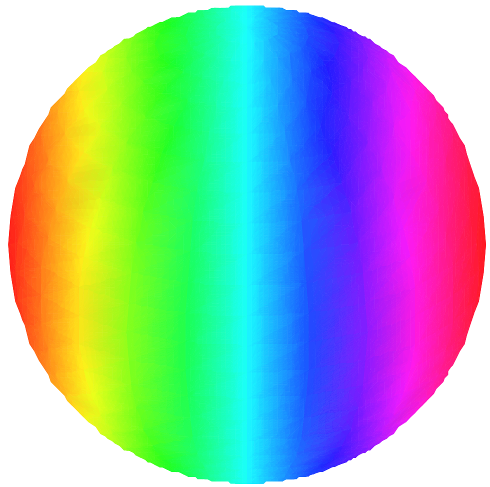
Repns of rotation group

Next simplest Lie group is rotations of the sphere.

Irreducible representations of rotation group are simplest ways to change with rotation. Examples:

- No change: trivial repn.
- Oscillation with freq $F = 1$.

  Different from compact time symmetry: need also direction to oscillate (up/down, left/right, in/out).



Repns of rotation group

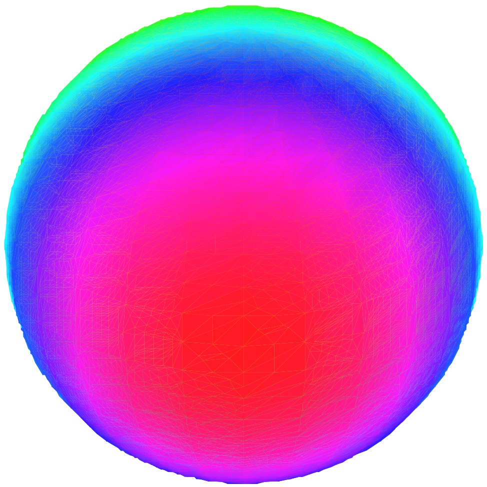
Next simplest Lie group is rotations of the sphere.

Irreducible representations of rotation group are simplest ways to change with rotation. Examples:

- No change: trivial repn.
- Oscillation with freq $F = 1$.

  Different from compact time symmetry: need also direction to oscillate (up/down, left/right, in/out).

  This repn has dimension 3.



Repns of rotation group

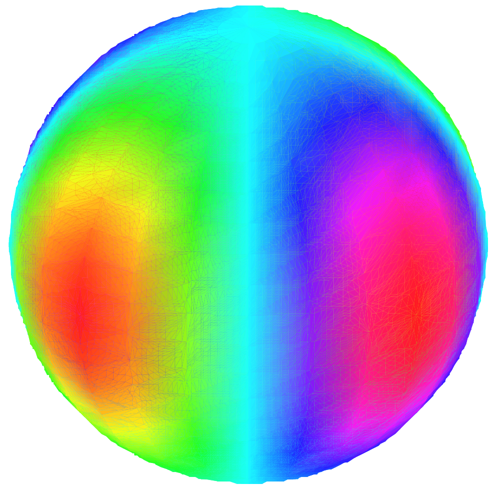
Next simplest Lie group is rotations of the sphere.

Irreducible representations of rotation group are simplest ways to change with rotation. Examples:

- No change: trivial repn.
- Oscillation with freq $F = 1$.

  Different from compact time symmetry: need also direction to oscillate (up/down, left/right, in/out).

  This repn has dimension 3.



- Oscillation freq $F = 2$ or $3 \ldots$

Repns of rotation group

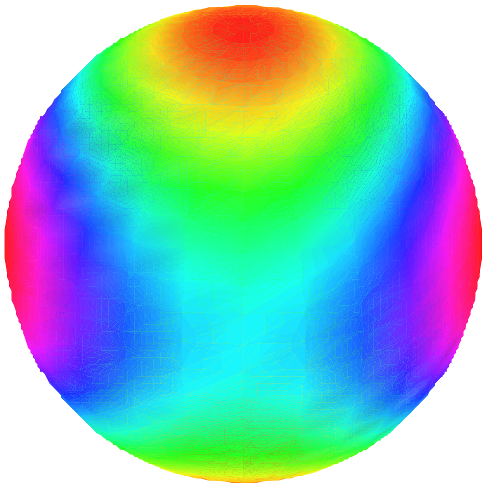
Next simplest Lie group is rotations of the sphere.

Irreducible representations of rotation group are simplest ways to change with rotation. Examples:

- No change: trivial repn.
- Oscillation with freq $F = 1$.

  Different from compact time symmetry: need also direction to oscillate (up/down, left/right, in/out).

  This repn has dimension 3.



- Oscillation freq $F = 2$ or $3 \ldots$

  This repn has dimension $2F + 1$.

Repns of rotation group

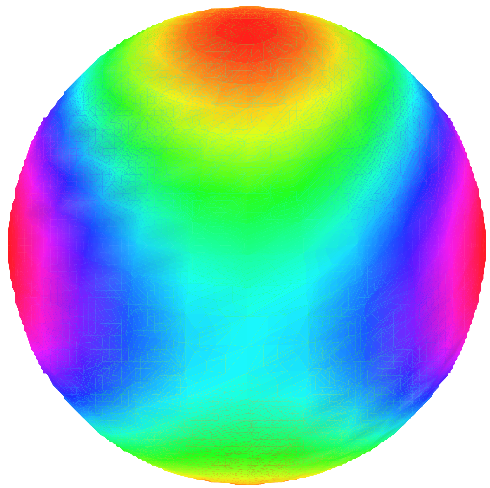
Next simplest Lie group is rotations of the sphere.

Irreducible representations of rotation group are simplest ways to change with rotation. Examples:

- No change: trivial repn.
- Oscillation with freq $F = 1$.

  Different from compact time symmetry: need also direction to oscillate (up/down, left/right, in/out).

  This repn has dimension 3.



- Oscillation freq $F = 2$ or $3 \ldots$

  This repn has dimension $2F + 1$.

That's all irreducible representations for the rotation group. Given by one integer F: frequency.

Repns of Lorentz group

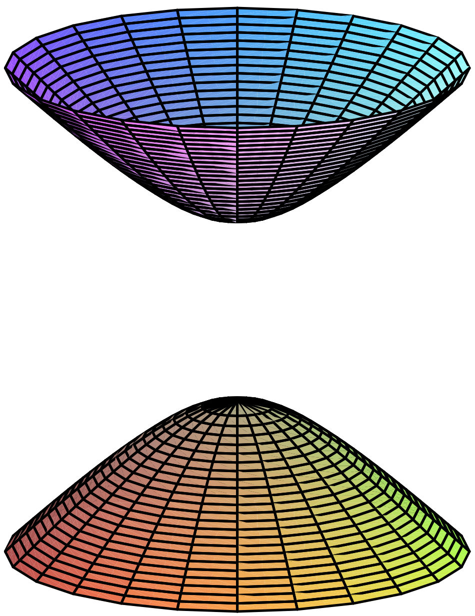
Repns of Lorentz group

Representations of Lorentz group are ways to change under relativistic symmetry. Two families. . .

Repns of Lorentz group

Representations of Lorentz group are ways to change under relativistic symmetry. Two families. . .

- Discrete series with frequency $F = \pm 1$ or ± 2 or. . . .

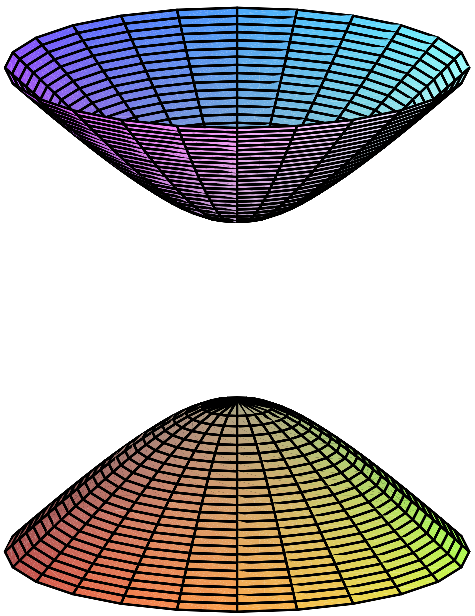


Repns of Lorentz group

Representations of Lorentz group are ways to change under relativistic symmetry. Two families. . .

- Discrete series with frequency $F = \pm 1$ or ± 2 or. . . .

$\leftrightsquigarrow$ holomorphic functions on hyperboloid of two sheets.



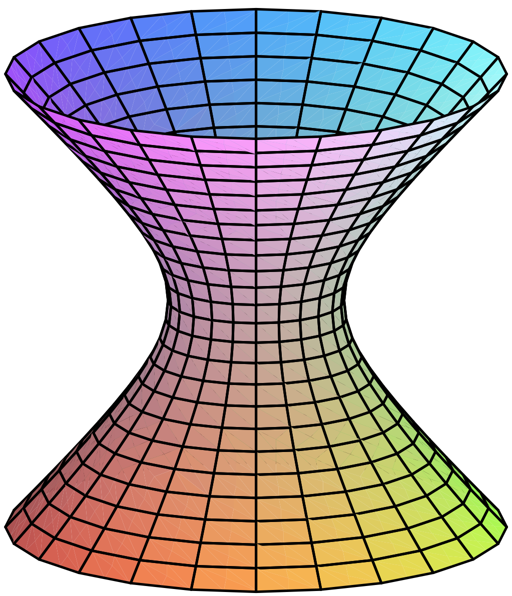
Repns of Lorentz group

Representations of Lorentz group are ways to change under relativistic symmetry. Two families...

- Discrete series with frequency $F = \pm 1$ or ± 2 or....

$\leftrightsquigarrow$ holomorphic functions on hyperboloid of two sheets.

- Principal series with growth rate $z =$ complex number.



Repns of Lorentz group

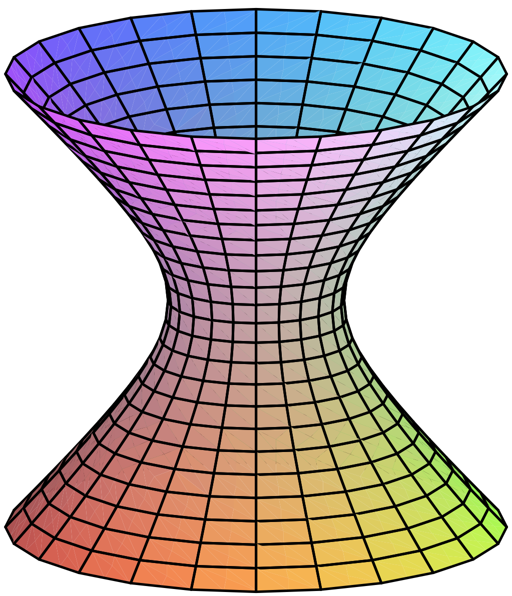
Representations of Lorentz group are ways to change under relativistic symmetry. Two families...

- Discrete series with frequency $F = \pm 1$ or ± 2 or....

$\leftrightsquigarrow$ holomorphic functions on hyperboloid of two sheets.

- Principal series with growth rate $z =$ complex number.

$\leftrightsquigarrow$ functions of homogeneity degree z on hyperboloid of one sheet.



Repns of Lorentz group

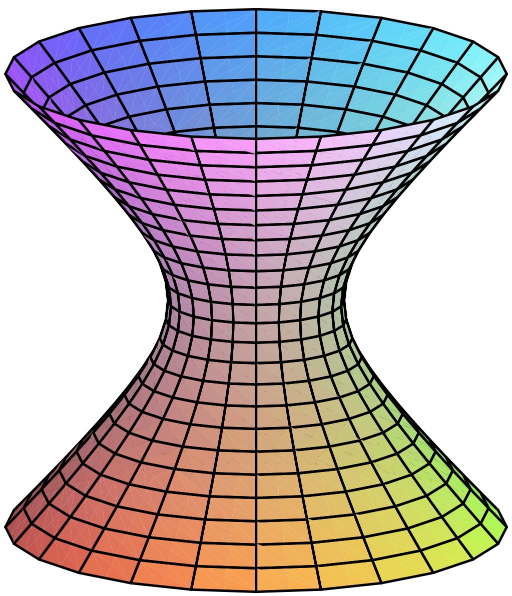
Representations of Lorentz group are ways to change under relativistic symmetry. Two families...

- Discrete series with frequency $F = \pm 1$ or ± 2 or....

$\leftrightsquigarrow$ holomorphic functions on hyperboloid of two sheets.

- Principal series with growth rate $z =$ complex number.

$\leftrightsquigarrow$ functions of homogeneity degree z on hyperboloid of one sheet.



That's all irreducible representations for the Lorentz group: two families, indexed by integer F or complex number z. Representations are infinite-dimensional, except principal series $z = \pm 1, \pm 2, \ldots$.

Morals of our story so far

Morals of our story so far

- Each representation identified by a few magic numbers, like. . .

Morals of our story so far

- Each representation identified by a few magic numbers, like. . .
  - rate of growth

Morals of our story so far

- Each representation identified by a few magic numbers, like. . .
  - rate of growth
  - frequency of oscillation

Morals of our story so far

- Each representation identified by a few magic numbers, like. . .
  - rate of growth
  - frequency of oscillation

group	representation	magic numbers

Morals of our story so far

- Each representation identified by a few magic numbers, like...
  - rate of growth
  - frequency of oscillation

group	representation	magic numbers
time symmetry	prime rate	$.0825$ (growth), 0.0 (frequency)

Morals of our story so far

- Each representation identified by a few magic numbers, like...
  - rate of growth
  - frequency of oscillation

group	representation	magic numbers
time symmetry	prime rate	$.0825$ (growth), 0.0 (frequency)
time symmetry	middle A	0.0 (growth), 440.0 (frequency)

Morals of our story so far

- Each representation identified by a few magic numbers, like. . .
  - rate of growth
  - frequency of oscillation

group	representation	magic numbers
time symmetry	prime rate	$.0825$ (growth), 0.0 (frequency)
time symmetry	middle A	0.0 (growth), 440.0 (frequency)
compact time symm.	EECS	6 (frequency)

Morals of our story so far

- Each representation identified by a few magic numbers, like...
  - rate of growth
  - frequency of oscillation

group	representation	magic numbers
time symmetry	prime rate	$.0825$ (growth), 0.0 (frequency)
time symmetry	middle A	0.0 (growth), 440.0 (frequency)
compact time symm.	EECS	6 (frequency)
rotations	d orbital electrons	2 (frequency)

Morals of our story so far

- Each representation identified by a few magic numbers, like. . .
  - rate of growth
  - frequency of oscillation

group	representation	magic numbers
time symmetry	prime rate	$.0825$ (growth), 0.0 (frequency)
time symmetry	middle A	0.0 (growth), 440.0 (frequency)
compact time symm.	EECS	6 (frequency)
rotations	d orbital electrons	2 (frequency)

- Magic numbers completely characterize the representation.

Morals of our story so far

- Each representation identified by a few magic numbers, like. . .
  - rate of growth
  - frequency of oscillation

group	representation	magic numbers
time symmetry	prime rate	$.0825$ (growth), 0.0 (frequency)
time symmetry	middle A	0.0 (growth), 440.0 (frequency)
compact time symm.	EECS	6 (frequency)
rotations	d orbital electrons	2 (frequency)

- Magic numbers completely characterize the representation.
- Group (partly) compact $\rightsquigarrow$ (some) magic numbers integers.

Morals of our story so far

- Each representation identified by a few magic numbers, like. . .
  - rate of growth
  - frequency of oscillation

group	representation	magic numbers
time symmetry	prime rate	$.0825$ (growth), 0.0 (frequency)
time symmetry	middle A	0.0 (growth), 440.0 (frequency)
compact time symm.	EECS	6 (frequency)
rotations	d orbital electrons	2 (frequency)

- Magic numbers completely characterize the representation.
- Group (partly) compact $\rightsquigarrow$ (some) magic numbers integers.

  Mathematical basis of integers in quantum physics.

What's a character table?

A description of all the representations.

What's a character table?

A description of all the representations.

One column for each irreducible representation, one row for each kind of symmetry. Here's the character table for time symmetry; the symbol T refers to moving forward T units of time.

What's a character table?

A description of all the representations.

One column for each irreducible representation, one row for each kind of symmetry. Here's the character table for time symmetry; the symbol T refers to moving forward T units of time.

	trivial	exp decay half-life H	exp growth doubling time D	oscillation frequency F	oscillatory growth
T	1	$2^{-T/H}$	$2^{T/D}$	$e^{2\pi iTF}$	$2^{T/D}e^{2\pi iTF}$

What's a character table?

A description of all the representations.

One column for each irreducible representation, one row for each kind of symmetry. Here's the character table for time symmetry; the symbol T refers to moving forward T units of time.

	trivial	exp decay half-life H	exp growth doubling time D	oscillation frequency F	oscillatory growth
T	1	$2^{-T/H}$	$2^{T/D}$	$e^{2\pi iTF}$	$2^{T/D}e^{2\pi iTF}$

Consolidate. . .

What's a character table?

A description of all the representations.

One column for each irreducible representation, one row for each kind of symmetry. Here's the character table for time symmetry; the symbol T refers to moving forward T units of time.

	trivial	exp decay half-life H	exp growth doubling time D	oscillation frequency F	oscillatory growth
T	1	$2^{-T/H}$	$2^{T/D}$	$e^{2\pi i T F}$	$2^{T/D} e^{2\pi i T F}$

Consolidate. . .

	z
T	$1 \cdot e^{zT}$

What's a character table?

A description of all the representations.

One column for each irreducible representation, one row for each kind of symmetry. Here's the character table for time symmetry; the symbol T refers to moving forward T units of time.

	trivial	exp decay half-life H	exp growth doubling time D	oscillation frequency F	oscillatory growth
T	1	$2^{-T/H}$	$2^{T/D}$	$e^{2\pi i T F}$	$2^{T/D} e^{2\pi i T F}$

Consolidate...

	z
T	$1 \cdot e^{zT}$

Atlas shorthand: $\left(1 \right)$.

Character table for Lorentz rotations

Character table for Lorentz rotations

Write θ for rotation, s for Lorentz boost.

Character table for Lorentz rotations

Write θ for rotation, s for Lorentz boost.

	positive discrete series repn $\#f$	negative discrete series repn $\#-f$	finite-dimensional $\#F$
θ	$-\frac{1\cdot e^{(2f+1)i\theta/2}}{2i\sin(\theta/2)}$	$\frac{1\cdot e^{-(2f+1)i\theta/2}}{2i\sin(\theta/2)}$	$\frac{1\cdot e^{(2F+1)i\theta/2}-1\cdot e^{-(2F+1)i\theta/2}}{2i\sin(\theta/2)}$
$s>0$	$\frac{e^{-(2f+1)s/2}}{2\sinh(s/2)}$	$\frac{e^{-(2f+1)s/2}}{2\sinh(s/2)}$	$\frac{1\cdot e^{(2F+1)s/2}-e^{-(2F+1)s/2}}{2\sinh(s/2)}$

Character table for Lorentz rotations

Write θ for rotation, s for Lorentz boost.

	positive discrete series repn $\#f$	negative discrete series repn $\#-f$	finite-dimensional $\#F$
θ	$-\frac{1 \cdot e^{(2f+1)i\theta/2}}{2i\sin(\theta/2)}$	$\frac{1 \cdot e^{-(2f+1)i\theta/2}}{2i\sin(\theta/2)}$	$\frac{1 \cdot e^{(2F+1)i\theta/2} - 1 \cdot e^{-(2F+1)i\theta/2}}{2i\sin(\theta/2)}$
$s > 0$	$\frac{e^{-(2f+1)s/2}}{2\sinh(s/2)}$	$\frac{e^{-(2f+1)s/2}}{2\sinh(s/2)}$	$\frac{1 \cdot e^{(2F+1)s/2} - e^{-(2F+1)s/2}}{2\sinh(s/2)}$

Atlas shorthand: $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$.

Character table for Lorentz rotations

Write θ for rotation, s for Lorentz boost.

	positive discrete series repn $\#f$	negative discrete series repn $\#{-f}$	finite-dimensional $\#F$
θ	$-\frac{1\cdot e^{(2f+1)i\theta/2}}{2i\sin(\theta/2)}$	$\frac{1\cdot e^{-(2f+1)i\theta/2}}{2i\sin(\theta/2)}$	$\frac{1\cdot e^{(2F+1)i\theta/2}-1\cdot e^{-(2F+1)i\theta/2}}{2i\sin(\theta/2)}$
$s>0$	$\frac{e^{-(2f+1)s/2}}{2\sinh(s/2)}$	$\frac{e^{-(2f+1)s/2}}{2\sinh(s/2)}$	$\frac{1\cdot e^{(2F+1)s/2}-e^{-(2F+1)s/2}}{2\sinh(s/2)}$

Atlas shorthand: $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$.

For applications, interesting representations are discrete series and trivial ($\#F = 1$). None has a simple physical interpretation like electron orbitals. . .

Character table for Lorentz rotations

Write θ for rotation, s for Lorentz boost.

	positive discrete series repn $\#f$	negative discrete series repn $\#-f$	finite-dimensional $\#F$
θ	$-\frac{1\cdot e^{(2f+1)i\theta/2}}{2i\sin(\theta/2)}$	$\frac{1\cdot e^{-(2f+1)i\theta/2}}{2i\sin(\theta/2)}$	$\frac{1\cdot e^{(2F+1)i\theta/2}-1\cdot e^{-(2F+1)i\theta/2}}{2i\sin(\theta/2)}$
$s>0$	$\frac{e^{-(2f+1)s/2}}{2\sinh(s/2)}$	$\frac{e^{-(2f+1)s/2}}{2\sinh(s/2)}$	$\frac{1\cdot e^{(2F+1)s/2}-e^{-(2F+1)s/2}}{2\sinh(s/2)}$

Atlas shorthand: $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$.

For applications, interesting representations are discrete series and trivial ($\#F=1$). None has a simple physical interpretation like electron orbitals. . .

. . . but discrete series $f=-1/4,-3/4 \leftrightsquigarrow$ quantum harmonic oscillator.

How do you write a character table?

RTFM (by Weyl and Harish-Chandra).

How do you write a character table?

RTFM (by Weyl and Harish-Chandra).

- Weyl and Harish-Chandra (1925, 1955): characters satisfy differential equations like $\frac{df}{dt} = z \cdot f$ (constant coefficient eigenvalue equations.) So solutions are combinations of functions like e^{zt}.

How do you write a character table?

RTFM (by Weyl and Harish-Chandra).

- Weyl and Harish-Chandra (1925, 1955): characters satisfy differential equations like $\frac{df}{dt} = z \cdot f$ (constant coefficient eigenvalue equations.) So solutions are combinations of functions like e^{zt}.
- Harish-Chandra (1965): wrote basic solns to differential equations $f_1, f_2, \ldots f_N$.

How do you write a character table?

RTFM (by Weyl and Harish-Chandra).

- Weyl and Harish-Chandra (1925, 1955): characters satisfy differential equations like $\frac{df}{dt} = z \cdot f$ (constant coefficient eigenvalue equations.) So solutions are combinations of functions like e^{zt}.
- Harish-Chandra (1965): wrote basic solns to differential equations $f_1, f_2, \ldots f_N$.

Any solution of differential equations (like a character) must be combination of basic solutions. Character matrix says which combinations are characters.

How do you write a character table?

RTFM (by Weyl and Harish-Chandra).

- Weyl and Harish-Chandra (1925, 1955): characters satisfy differential equations like $\frac{df}{dt} = z \cdot f$ (constant coefficient eigenvalue equations.) So solutions are combinations of functions like e^{zt}.
- Harish-Chandra (1965): wrote basic solns to differential equations $f_1, f_2, \ldots f_N$.

  Any solution of differential equations (like a character) must be combination of basic solutions. Character matrix says which combinations are characters.
- Langlands (1970): Character matrix is upper triangular matrix of integers, ones on diagonal.

How do you write a character matrix?

How do you write a character matrix?

RTFM (by Kazhdan and Lusztig).

How do you write a character matrix?

RTFM (by Kazhdan and Lusztig).

- Beilinson and Bernstein (1981): Character matrix is described by geometry of flag variety for G.

How do you write a character matrix?

RTFM (by Kazhdan and Lusztig).

- Beilinson and Bernstein (1981): Character matrix is described by geometry of flag variety for G.

Idea: flag variety is simplest/most complicated geometry for G. Understand the flag variety and understand everything!

How do you write a character matrix?

RTFM (by Kazhdan and Lusztig).

- Beilinson and Bernstein (1981): Character matrix is described by geometry of flag variety for G.

  Idea: flag variety is simplest/most complicated geometry for G. Understand the flag variety and understand everything!

  Classical groups: flag varieties $\leftrightsquigarrow$ projective Euclidean geometry of lines, planes. . .

How do you write a character matrix?

RTFM (by Kazhdan and Lusztig).

- Beilinson and Bernstein (1981): Character matrix is described by geometry of flag variety for G.

Idea: flag variety is simplest/most complicated geometry for G. Understand the flag variety and understand everything!

Classical groups: flag varieties $\leftrightsquigarrow$ projective Euclidean geometry of lines, planes. . .

Exceptional groups: flag varieties are more mysterious.

How do you write a character matrix?

RTFM (by Kazhdan and Lusztig).

- Beilinson and Bernstein (1981): Character matrix is described by geometry of flag variety for G.

  Idea: flag variety is simplest/most complicated geometry for G. Understand the flag variety and understand everything!

  Classical groups: flag varieties $\leftrightsquigarrow$ projective Euclidean geometry of lines, planes. . .

  Exceptional groups: flag varieties are more mysterious.

- Kazhdan/Lusztig (1979): how to compute char matrix.

How do you write a character matrix?

RTFM (by Kazhdan and Lusztig).

- Beilinson and Bernstein (1981): Character matrix is described by geometry of flag variety for G.

  Idea: flag variety is simplest/most complicated geometry for G. Understand the flag variety and understand everything!

  Classical groups: flag varieties $\leftrightsquigarrow$ projective Euclidean geometry of lines, planes...

  Exceptional groups: flag varieties are more mysterious.

- Kazhdan/Lusztig (1979): how to compute char matrix.

  Coxeter: simple Lie group $\rightsquigarrow$ regular polyhedron $\rightsquigarrow$ finite math.

How do you write a character matrix?

RTFM (by Kazhdan and Lusztig).

- Beilinson and Bernstein (1981): Character matrix is described by geometry of flag variety for G.

  Idea: flag variety is simplest/most complicated geometry for G. Understand the flag variety and understand everything!

  Classical groups: flag varieties $\leftrightsquigarrow$ projective Euclidean geometry of lines, planes...

  Exceptional groups: flag varieties are more mysterious.

- Kazhdan/Lusztig (1979): how to compute char matrix.

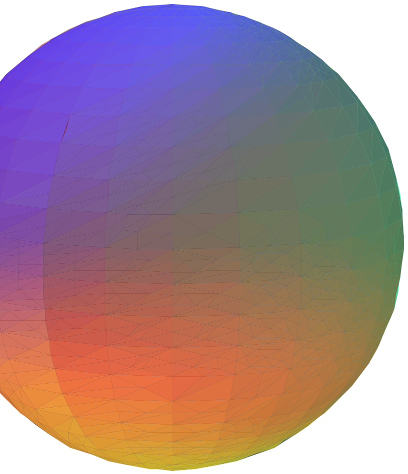
  Coxeter: simple Lie group $\rightsquigarrow$ regular polyhedron $\rightsquigarrow$ finite math.

  Kazhdan/Lusztig: finite math $\rightsquigarrow$ geometry of flag variety.

Example: Lorentz group

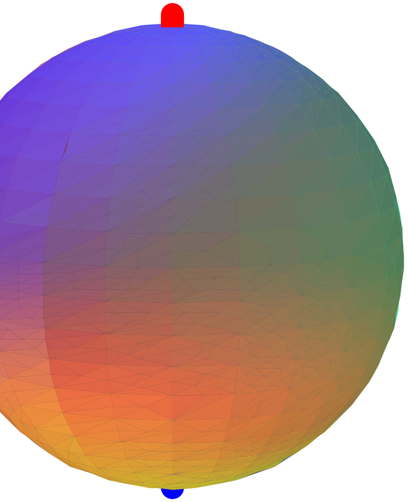
Example: Lorentz group

- Flag variety is sphere.



Example: Lorentz group

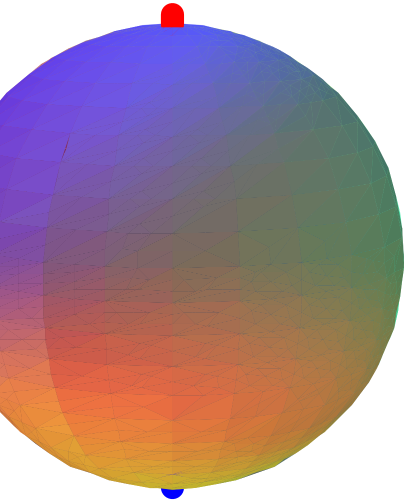
- Flag variety is sphere.



- Sphere divided in 3 parts: north pole, south pole, rest.

Example: Lorentz group

- Flag variety is sphere.



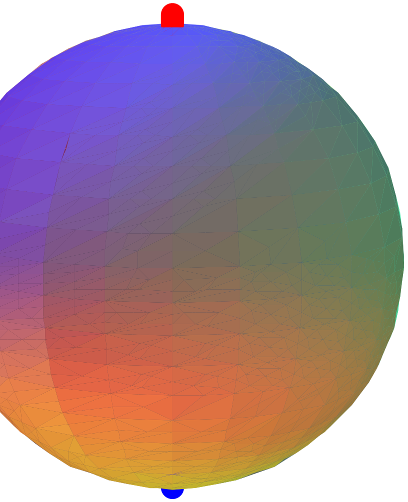
	NP	SP	rest
NP	1	0	1
SP	0	1	1
rest	0	0	1

- Sphere divided in 3 parts: north pole, south pole, rest.

  Each column describes one piece of sphere.

Example: Lorentz group

- Flag variety is sphere.



	NP	SP	rest
NP	1	0	1
SP	0	1	1
rest	0	0	1

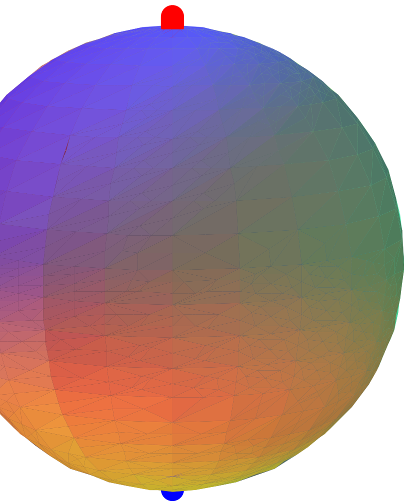
- Sphere divided in 3 parts: north pole, south pole, rest.

  Each column describes one piece of sphere.

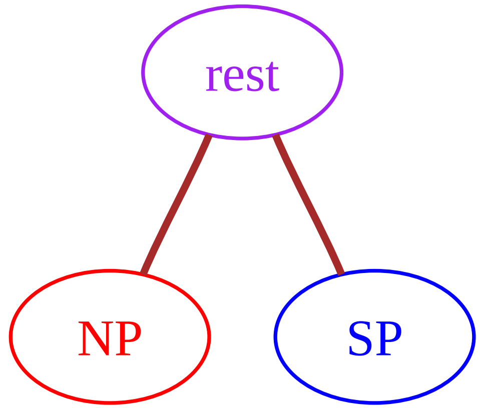
  Row entry describes how it looks near a smaller piece.

Example: Lorentz group

- Flag variety is sphere.



	NP	SP	rest
NP	1	0	1
SP	0	1	1
rest	0	0	1



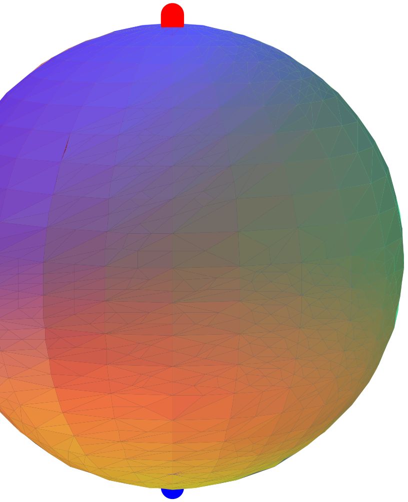
- Sphere divided in 3 parts: north pole, south pole, rest.

  Each column describes one piece of sphere.

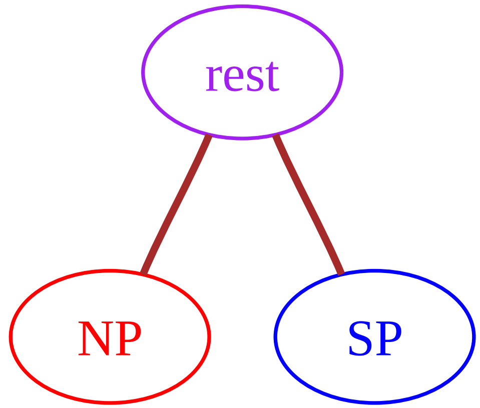
  Row entry describes how it looks near a smaller piece.
- Graph encodes geometry of sphere.

Example: Lorentz group

- Flag variety is sphere.



	NP	SP	rest
NP	1	0	1
SP	0	1	1
rest	0	0	1



- Sphere divided in 3 parts: north pole, south pole, rest.

  Each column describes one piece of sphere.

  Row entry describes how it looks near a smaller piece.

- Graph encodes geometry of sphere.

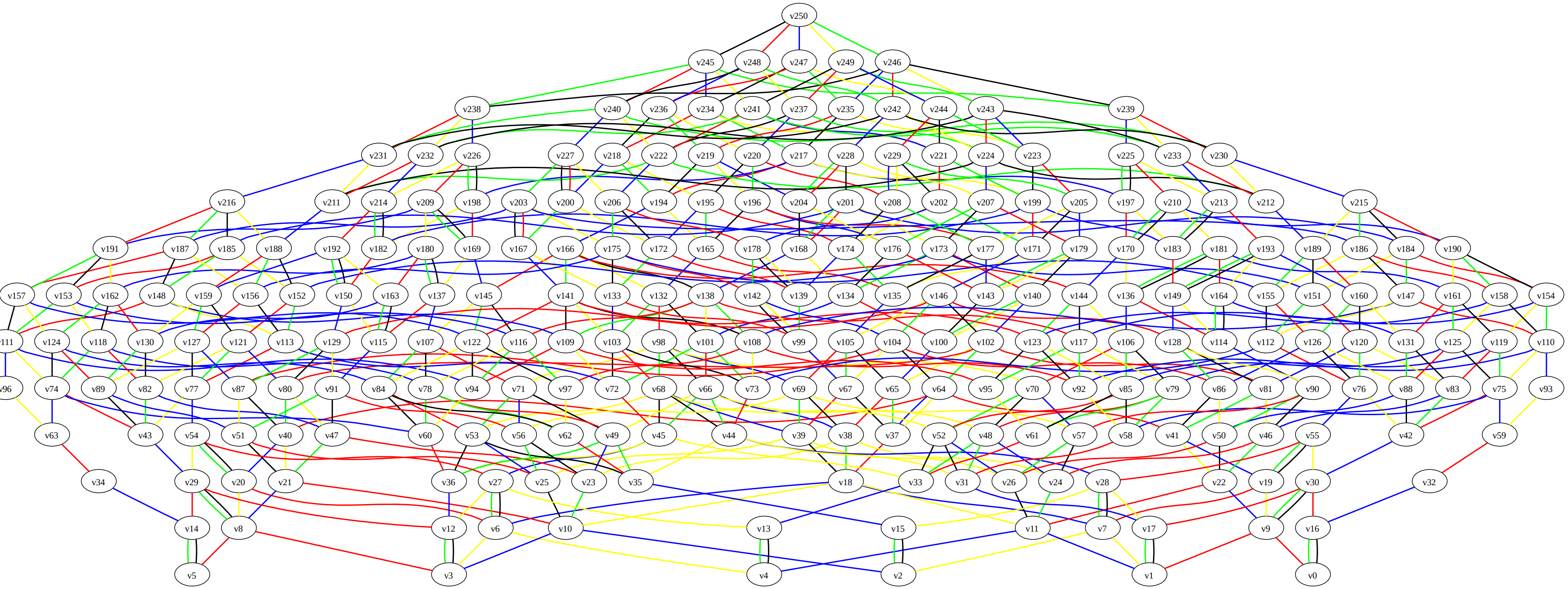
  For big groups: let graph tell you what algebra to do.

So what did you guys do exactly?

We read TFM.

So what did you guys do exactly?

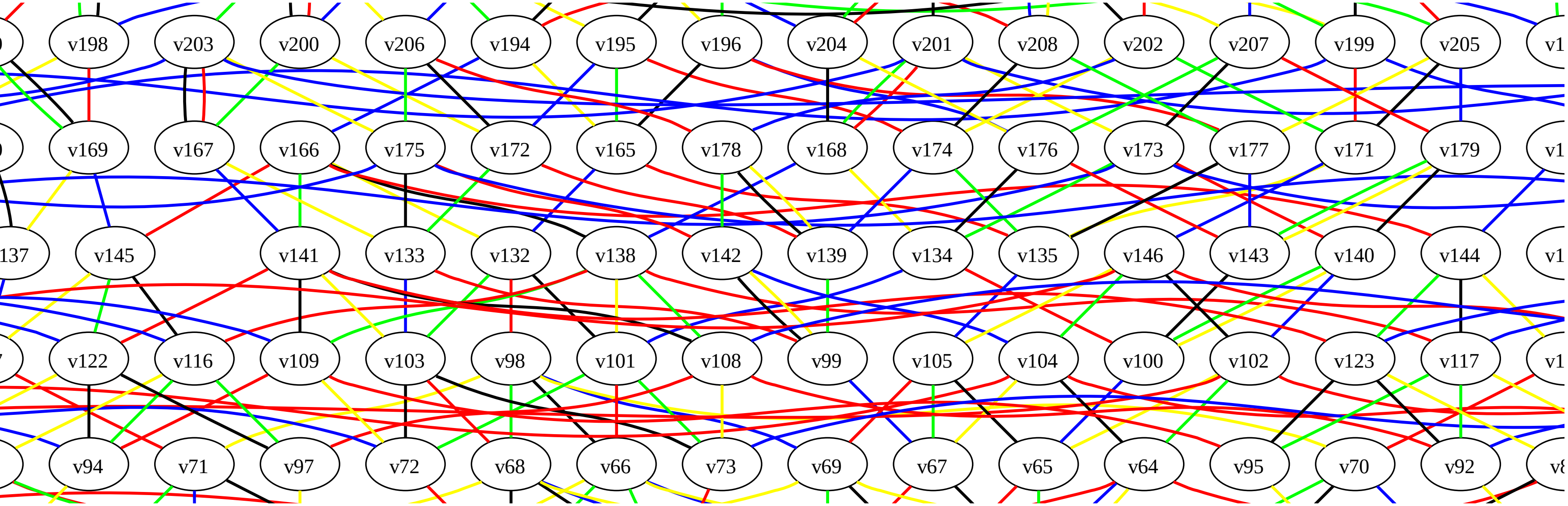
We read TFM.



Graph for group $SO(5,5)$ (corresponding to equilateral $\triangle$).

So what did you guys do exactly?

We read TFM.



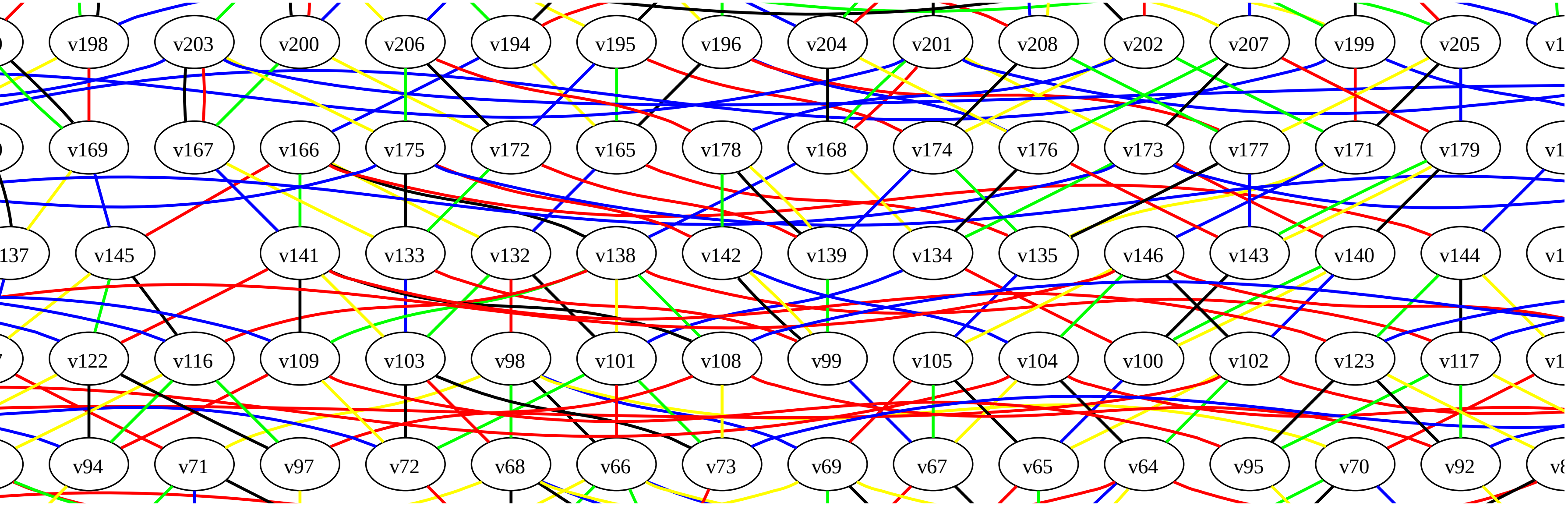
closeup view

Graph for group $SO(5,5)$ (corresponding to equilateral $\triangle$).

251 vertices $\rightsquigarrow$ 251 pieces of 40-dimensional flag variety.

So what did you guys do exactly?

We read TFM.



closeup view

Graph for group $SO(5,5)$ (corresponding to equilateral $\triangle$).

251 vertices $\rightsquigarrow$ 251 pieces of 40-dimensional flag variety.

E_8: $453,060$ vertices $\rightsquigarrow$ pieces of 240-dimensional flag variety.

How the computation works

How the computation works

- graph vertex $y \leftrightsquigarrow$ irreducible character

How the computation works

- graph vertex $y \leftrightsquigarrow$ irreducible character
- lower vertices $x \leftrightsquigarrow$ terms in character formula

How the computation works

- graph vertex $y \leftrightsquigarrow$ irreducible character
- lower vertices $x \leftrightsquigarrow$ terms in character formula
- For each pair (x, y), compute KL polynomial $P_{x,y}$.

How the computation works

- graph vertex $y \leftrightsquigarrow$ irreducible character
- lower vertices $x \leftrightsquigarrow$ terms in character formula
- For each pair (x, y), compute KL polynomial $P_{x,y}$.

  $P_{x,y}(1)$ is coefficient of term x in irreducible character y.

How the computation works

- graph vertex $y \leftrightsquigarrow$ irreducible character
- lower vertices $x \leftrightsquigarrow$ terms in character formula
- For each pair (x, y), compute KL polynomial $P_{x,y}$.

  $P_{x,y}(1)$ is coefficient of term x in irreducible character y.
- Induction: start with y's on bottom of graph, work up. For each y, start with $x = y$, work down.

How the computation works

- graph vertex $y \leftrightsquigarrow$ irreducible character
- lower vertices $x \leftrightsquigarrow$ terms in character formula
- For each pair (x, y), compute KL polynomial $P_{x,y}$.

  $P_{x,y}(1)$ is coefficient of term x in irreducible character y.
- Induction: start with y's on bottom of graph, work up.
  For each y, start with $x = y$, work down.
- Seek line up $\begin{array}{c} x' \\ | \\ x \end{array}$ same color as some line down $\begin{array}{c} y \\ | \\ y' \end{array}$.

How the computation works

- graph vertex $y \leftrightsquigarrow$ irreducible character
- lower vertices $x \leftrightsquigarrow$ terms in character formula
- For each pair (x, y), compute KL polynomial $P_{x,y}$.

  $P_{x,y}(1)$ is coefficient of term x in irreducible character y.
- Induction: start with y's on bottom of graph, work up.
  For each y, start with $x = y$, work down.
- Seek line up $\begin{matrix} x' \\ | \\ x \end{matrix}$ same color as some line down $\begin{matrix} y \\ | \\ y' \end{matrix}$.

  If it's there, then $P_{x,y} = P_{x',y}$ (known by induction).
  If not, (x, y) is primitive: no color down from y goes up from x.

How the computation works

- graph vertex $y \leftrightsquigarrow$ irreducible character
- lower vertices $x \leftrightsquigarrow$ terms in character formula
- For each pair (x, y), compute KL polynomial $P_{x,y}$.

  $P_{x,y}(1)$ is coefficient of term x in irreducible character y.
- Induction: start with y's on bottom of graph, work up.
  For each y, start with $x = y$, work down.
- Seek line up $\begin{array}{c} x' \\ | \\ x \end{array}$ same color as some line down $\begin{array}{c} y \\ | \\ y' \end{array}$.

  If it's there, then $P_{x,y} = P_{x',y}$ (known by induction).
  If not, (x, y) is primitive: no color down from y goes up from x.
- One hard calculation for each primitive pair (x, y).

What to do for primitive pair (x, y)

What to do for primitive pair (x, y)

- graph vertex $y \leftrightsquigarrow$ big piece F_y of flag variety.

What to do for primitive pair (x, y)

- graph vertex $y \leftrightsquigarrow$ big piece F_y of flag variety.
- lower vertex $x \leftrightsquigarrow$ little piece F_x of flag variety.

What to do for primitive pair (x, y)

- graph vertex $y \leftrightsquigarrow$ big piece F_y of flag variety.
- lower vertex $x \leftrightsquigarrow$ little piece F_x of flag variety.

  Want to know how singular F_y is near F_x.

What to do for primitive pair (x, y)

- graph vertex $y \leftrightsquigarrow$ big piece F_y of flag variety.
- lower vertex $x \leftrightsquigarrow$ little piece F_x of flag variety.

  Want to know how singular F_y is near F_x.

- Pick line down y; means $F_y \approx F_{y'} \times 2$-diml sphere.

$$\begin{array}{c} y \\ | \\ y' \end{array}$$

What to do for primitive pair (x, y)

- graph vertex $y \leftrightsquigarrow$ big piece F_y of flag variety.
- lower vertex $x \leftrightsquigarrow$ little piece F_x of flag variety.

  Want to know how singular F_y is near F_x.

- Pick line down y; means $F_y \approx F_{y'} \times 2$-diml sphere.

  y
  |
  y'

- Primitive means red line x is also down from x.

  x
  |
  x'

What to do for primitive pair (x, y)

- graph vertex $y \leftrightsquigarrow$ big piece F_y of flag variety.
- lower vertex $x \leftrightsquigarrow$ little piece F_x of flag variety.

  Want to know how singular F_y is near F_x.
- Pick line down $\begin{array}{c} y \\ | \\ y' \end{array}$; means $F_y \approx F_{y'} \times 2$-diml sphere.
- Primitive means red line $\begin{array}{c} x \\ | \\ x' \end{array}$ is also down from x.
- Geometry translates to algebra $P_{x,y} \approx P_{x',y'} + qP_{x,y'}$. Precisely:

What to do for primitive pair (x, y)

- graph vertex $y \leftrightsquigarrow$ big piece F_y of flag variety.
- lower vertex $x \leftrightsquigarrow$ little piece F_x of flag variety.

  Want to know how singular F_y is near F_x.
- Pick line down $\begin{array}{c} y \\ | \\ y' \end{array}$; means $F_y \approx F_{y'} \times 2$-diml sphere.
- Primitive means red line $\begin{array}{c} x \\ | \\ x' \end{array}$ is also down from x.
- Geometry translates to algebra $P_{x,y} \approx P_{x',y'} + qP_{x,y'}$. Precisely:

$$P_{x,y} = P_{x',y'} + qP_{x,y'} - \sum_{x' \leq z < y'} \mu(z, y') q^{(l(y') - l(z) - 1)/2} P_{x',z}.$$

What to do for primitive pair (x, y)

- graph vertex $y \leftrightsquigarrow$ big piece F_y of flag variety.
- lower vertex $x \leftrightsquigarrow$ little piece F_x of flag variety.

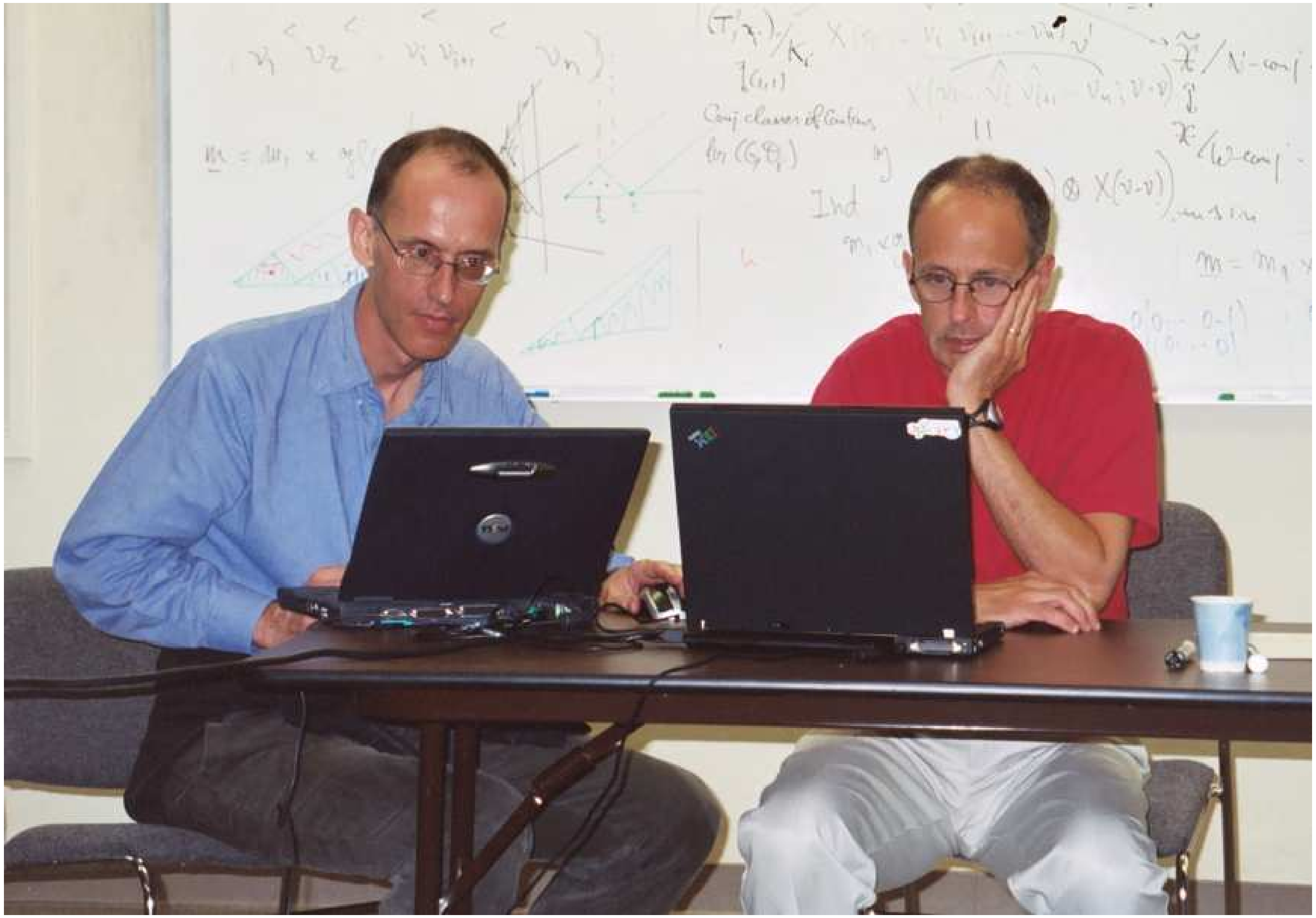
  Want to know how singular F_y is near F_x.
- Pick line down $\begin{array}{c} y \\ | \\ y' \end{array}$; means $F_y \approx F_{y'} \times 2$-diml sphere.
- Primitive means red line $\begin{array}{c} x \\ | \\ x' \end{array}$ is also down from x.
- Geometry translates to algebra $P_{x,y} \approx P_{x',y'} + qP_{x,y'}$. Precisely:

$$P_{x,y} = P_{x',y'} + qP_{x,y'} - \sum_{x' \leq z < y'} \mu(z, y') q^{(l(y')-l(z)-1)/2} P_{x',z}.$$

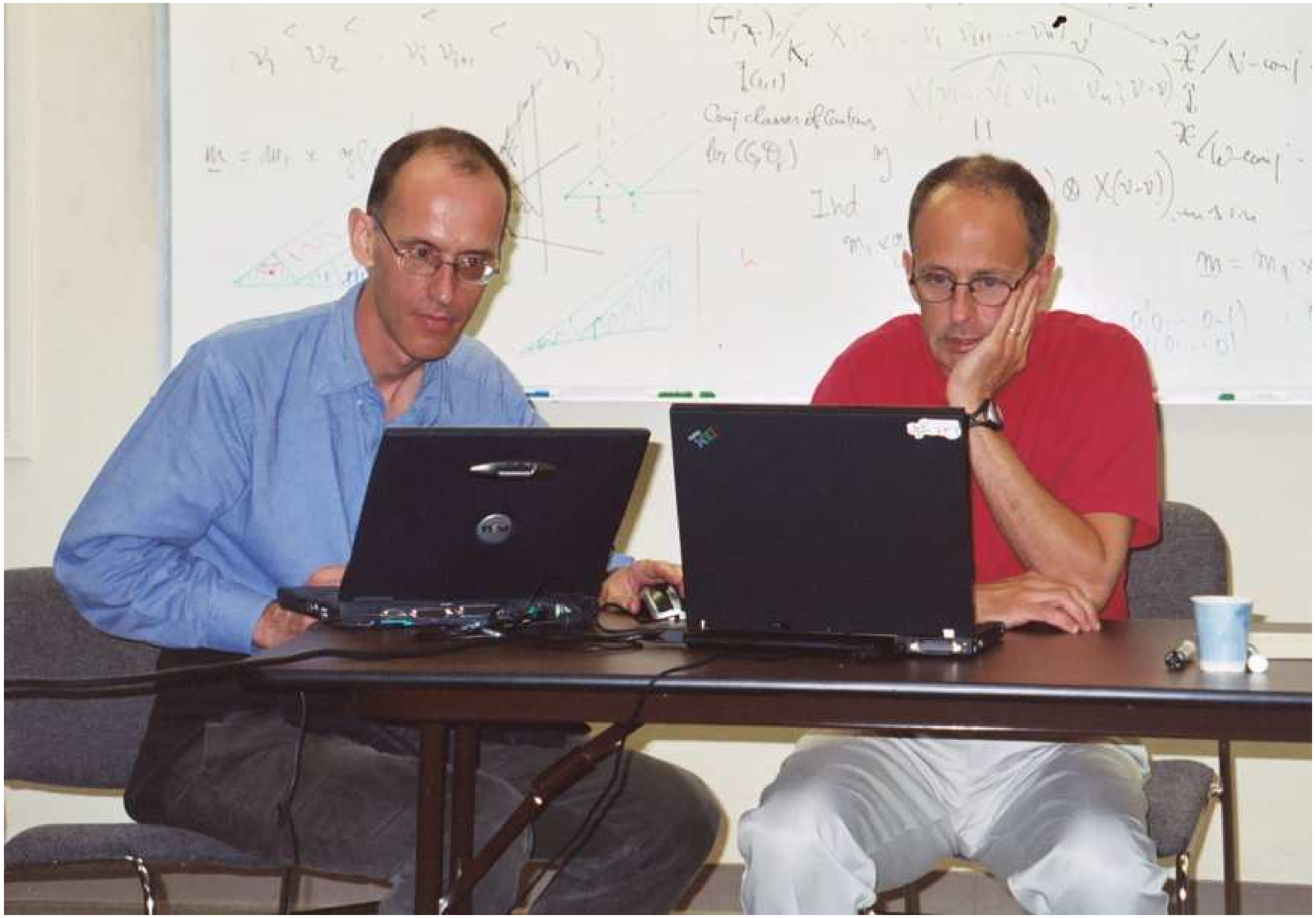
For E_8, the big sum averages about 150 nonzero terms.

How do you make a computer do that?

How do you make a computer do that?

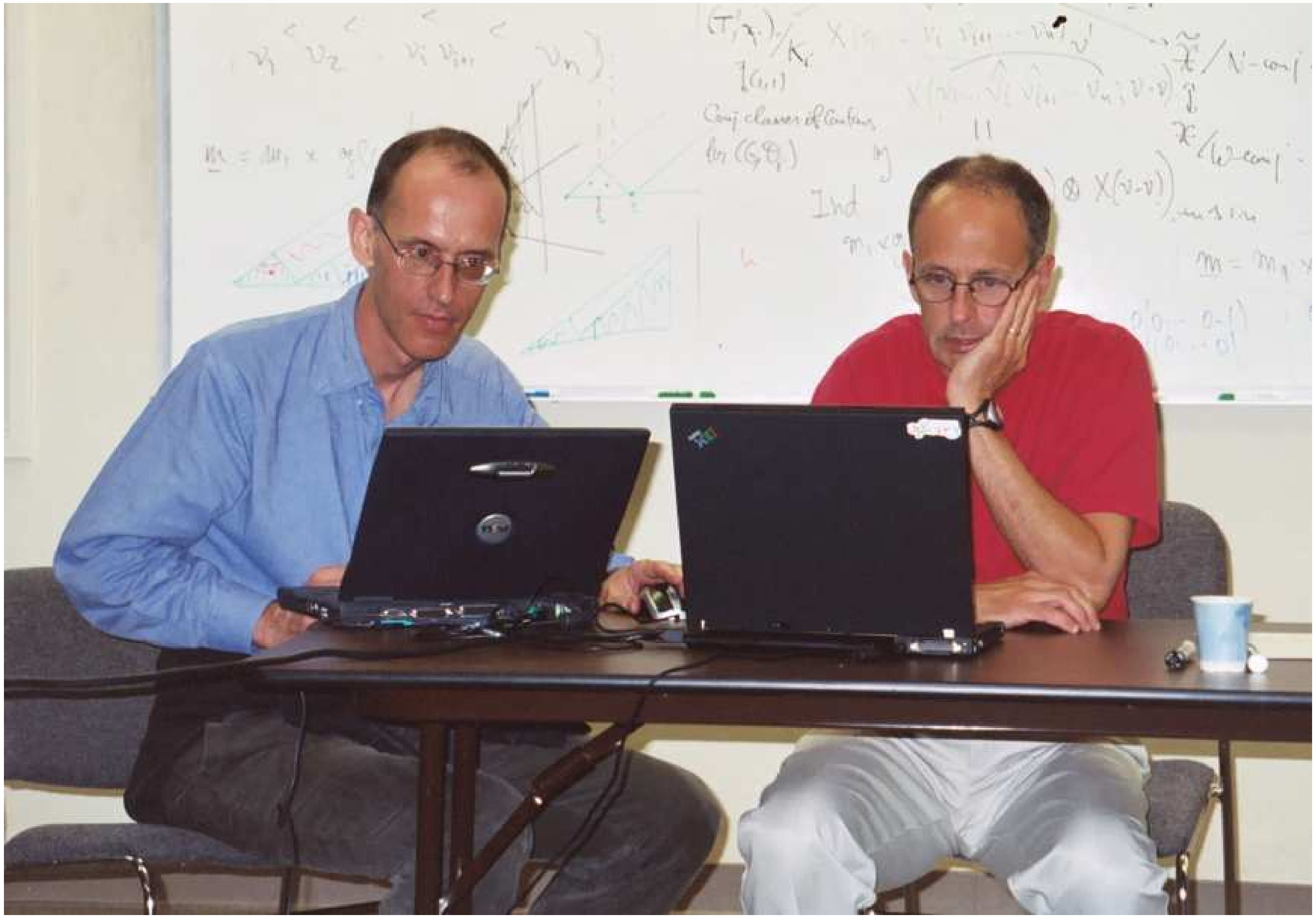


Iow do you make a computer do that?



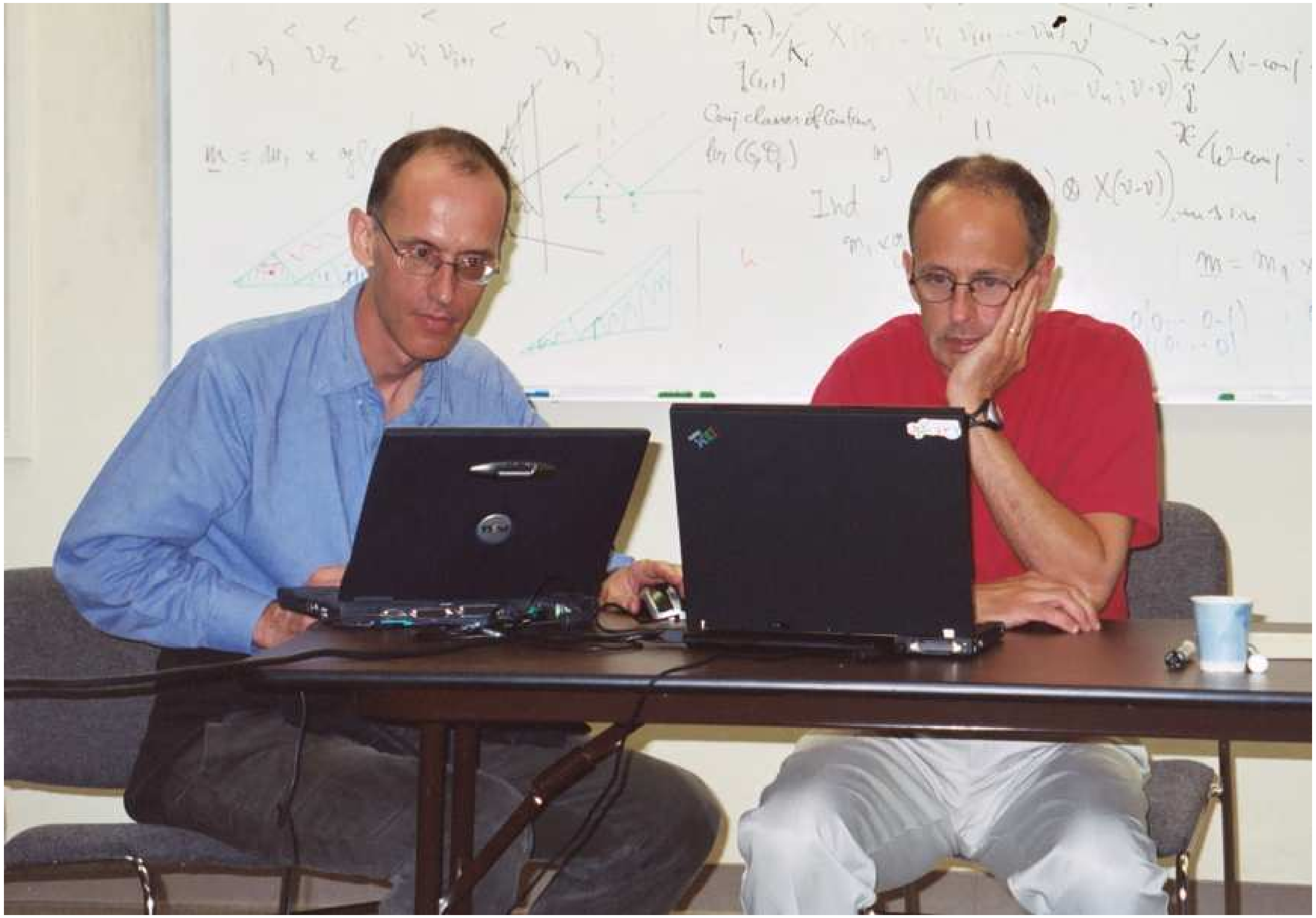
- In June 2002, Jeff Adams asked Fokko du Cloux.

Iow do you make a computer do that?



- In June 2002, Jeff Adams asked Fokko du Cloux.
- In November 2005, Fokko finished the program.

How do you make a computer do that?



- In June 2002, Jeff Adams asked Fokko du Cloux.
- In November 2005, Fokko finished the program.

Wasn't that easy?

What's the computer have to do?

TASK	COMPUTER RQMT

What's the computer have to do?

TASK	COMPUTER RQMT
Make graph: 453,060 nodes, 8 edges from each	250M RAM, 10 minutes

What's the computer have to do?

TASK	COMPUTER RQMT
Make graph: 453,060 nodes, 8 edges from each	250M RAM, 10 minutes
List primitive pairs of vertices: 6,083,626,944	450M RAM, few seconds

What's the computer have to do?

TASK	COMPUTER RQMT
Make graph: 453,060 nodes, 8 edges from each	250M RAM, 10 minutes
List primitive pairs of vertices: 6,083,626,944	450M RAM, few seconds
Calculate the polynomial for each primitive pair	Fetch few kB from memory, few thousand integer ops $\times$ 6 billion

What's the computer have to do?

TASK	COMPUTER RQMT
Make graph: 453,060 nodes, 8 edges from each	250M RAM, 10 minutes
List primitive pairs of vertices: 6,083,626,944	450M RAM, few seconds
Calculate the polynomial for each primitive pair	Fetch few kB from memory, few thousand integer ops $\times$ 6 billion
Look for polynomial in store, add if it's a new one	$4 \frac{\text{bytes}}{\text{coef}} \times 20 \frac{\text{coefs}}{\text{poly}} \times ??\ \text{polys}$ RAM

What's the computer have to do?

TASK	COMPUTER RQMT
Make graph: 453,060 nodes, 8 edges from each	250M RAM, 10 minutes
List primitive pairs of vertices: 6,083,626,944	450M RAM, few seconds
Calculate the polynomial for each primitive pair	Fetch few kB from memory, few thousand integer ops $\times$ 6 billion
Look for polynomial in store, add if it's a new one	$4 \frac{\text{bytes}}{\text{coef}} \times 20 \frac{\text{coefs}}{\text{poly}} \times ??\ \text{polys}$ RAM
Write number for poly in table	25G RAM

What's the computer have to do?

TASK	COMPUTER RQMT
Make graph: 453,060 nodes, 8 edges from each	250M RAM, 10 minutes
List primitive pairs of vertices: 6,083,626,944	450M RAM, few seconds
Calculate the polynomial for each primitive pair	Fetch few kB from memory, few thousand integer ops $\times$ 6 billion
Look for polynomial in store, add if it's a new one	$4\ \frac{\text{bytes}}{\text{coef}} \times 20\ \frac{\text{coefs}}{\text{poly}} \times\ ??\ \text{polys}$ RAM
Write number for poly in table	25G RAM

Big unknown: number of distinct polys.

What's the computer have to do?

TASK	COMPUTER RQMT
Make graph: 453,060 nodes, 8 edges from each	250M RAM, 10 minutes
List primitive pairs of vertices: 6,083,626,944	450M RAM, few seconds
Calculate the polynomial for each primitive pair	Fetch few kB from memory, few thousand integer ops $\times$ 6 billion
Look for polynomial in store, add if it's a new one	$4 \frac{\text{bytes}}{\text{coef}} \times 20 \frac{\text{coefs}}{\text{poly}} \times ??\ \text{polys}$ RAM
Write number for poly in table	25G RAM

Big unknown: number of distinct polys.

Hoped 400 million polys $\rightsquigarrow$ 75G total RAM.

What's the computer have to do?

TASK	COMPUTER RQMT
Make graph: 453,060 nodes, 8 edges from each	250M RAM, 10 minutes
List primitive pairs of vertices: 6,083,626,944	450M RAM, few seconds
Calculate the polynomial for each primitive pair	Fetch few kB from memory, few thousand integer ops $\times$ 6 billion
Look for polynomial in store, add if it's a new one	$4 \frac{\text{bytes}}{\text{coef}} \times 20 \frac{\text{coefs}}{\text{poly}} \times ??\ \text{polys}$ RAM
Write number for poly in table	25G RAM

Big unknown: number of distinct polys.

Hoped 400 million polys $\rightsquigarrow$ 75G total RAM.

Feared 1 billion $\rightsquigarrow$ 150G total RAM.

Saga of the end times

Saga of the end times

1/06 Experiments by Birne Binegar on William Stein's computer `sage` showed we needed 150G.

Saga of the end times

- 1/06 Experiments by Birne Binegar on William Stein's computer `sage` showed we needed 150G.
- 3/06 Asked about pure math uses for 256G computer.

Saga of the end times

l/06 Experiments by Birne Binegar on William Stein's computer `sage` showed we needed 150G.

3/06 Asked about pure math uses for 256G computer.

0/06 Noam Elkies told us we didn't need one. . .

Saga of the end times

1/06 Experiments by Birne Binegar on William Stein's computer `sage` showed we needed 150G.

3/06 Asked about pure math uses for 256G computer.

0/06 Noam Elkies told us we didn't need one. . .

$$\text{one 150G computation} \xrightarrow{\binom{\text{modular}}{\text{arithmetic}}} \text{four 50G computations}$$

Saga of the end times

1/06 Experiments by Birne Binegar on William Stein's computer `sage` showed we needed 150G.

3/06 Asked about pure math uses for 256G computer.

9/06 Noam Elkies told us we didn't need one. . .

$$\text{one 150G computation} \xrightarrow{\binom{\text{modular}}{\text{arithmetic}}} \text{four 50G computations}$$

3/06 Marc van Leeuwen made Fokko's code modular.

Saga of the end times

1/06 Experiments by Birne Binegar on William Stein's computer `sage` showed we needed 150G.

3/06 Asked about pure math uses for 256G computer.

0/06 Noam Elkies told us we didn't need one. . .

$$\text{one 150G computation} \xrightarrow{\binom{\text{modular}}{\text{arithmetic}}} \text{four 50G computations}$$

3/06 Marc van Leeuwen made Fokko's code modular.

9/06 mod 251 computation on `sage`. Took 17 hours:

Saga of the end times

1/06 Experiments by Birne Binegar on William Stein's computer `sage` showed we needed 150G.

3/06 Asked about pure math uses for 256G computer.

0/06 Noam Elkies told us we didn't need one. . .

$$\text{one 150G computation} \xrightarrow{\binom{\text{modular}}{\text{arithmetic}}} \text{four 50G computations}$$

3/06 Marc van Leeuwen made Fokko's code modular.

9/06 mod 251 computation on `sage`. Took 17 hours:

```
Total elapsed time = 62575s.  Finished at l = 64, y = 453059
d_store.size() = 1181642979, prim_size = 3393819659

VmData:  64435824 kB
```

Saga of the end times

1/06 Experiments by Birne Binegar on William Stein's computer `sage` showed we needed 150G.

3/06 Asked about pure math uses for 256G computer.

9/06 Noam Elkies told us we didn't need one. . .

$$\text{one 150G computation} \xrightarrow{\binom{\text{modular}}{\text{arithmetic}}} \text{four 50G computations}$$

3/06 Marc van Leeuwen made Fokko's code modular.

9/06 mod 251 computation on `sage`. Took 17 hours:

```
Total elapsed time = 62575s.  Finished at l = 64, y = 453059
d_store.size() = 1181642979, prim_size = 3393819659
VmData:  64435824 kB
```

Writing to disk took two days. Investigating why $\rightsquigarrow$ output bug, so mod 251 character table no good.

The Tribulation (continued)

The Tribulation (continued)

/06 9 P.M. Started mod 256 computation on `sage`. Computed 452,174 out of 453,060 rows of char table in 14 hours, then `sage` crashed.

The Tribulation (continued)

1/06 9 P.M. Started mod 256 computation on `sage`. Computed 452,174 out of 453,060 rows of char table in 14 hours, then `sage` crashed.

2/06 EVENING Restarted mod 256. Finished in just 11 hours

(hip, hip, HURRAH! hip, hip, HURRAH! `pthread_join(cheer[k], NULL);`):

The Tribulation (continued)

1/06 9 P.M. Started mod 256 computation on `sage`. Computed 452,174 out of 453,060 rows of char table in 14 hours, then `sage` crashed.

2/06 EVENING Restarted mod 256. Finished in just 11 hours

(hip, hip, HURRAH! hip, hip, HURRAH! `pthread_join(cheer[k], NULL);`):

```
Total elapsed time = 40229s.  Finished at l = 64, y = 453059
d_store.size() = 1181642979, prim_size = 3393819659
```

```
VmData:  54995416 kB
```

The Tribulation (continued)

/06 9 P.M. Started mod 256 computation on `sage`. Computed 452,174 out of 453,060 rows of char table in 14 hours, then `sage` crashed.

/06 EVENING Restarted mod 256. Finished in just 11 hours

(hip, hip, HURRAH! hip, hip, HURRAH! `pthread_join(cheer[k], NULL);`):

```
Total elapsed time = 40229s.  Finished at l = 64, y = 453059
d_store.size() = 1181642979, prim_size = 3393819659
```

```
VmData:  54995416 kB
```

/06 Started mod 255 computation on `sage`, which crashed.

The Tribulation (continued)

1/06 9 P.M. Started mod 256 computation on `sage`. Computed 452,174 out of 453,060 rows of char table in 14 hours, then `sage` crashed.

2/06 EVENING Restarted mod 256. Finished in just 11 hours

(hip, hip, HURRAH! / hip, hip, HURRAH! `pthread_join(cheer[k], NULL);`):

```
Total elapsed time = 40229s.  Finished at l = 64, y = 453059
d_store.size() = 1181642979, prim_size = 3393819659
VmData:  54995416 kB
```

3/06 Started mod 255 computation on `sage`, which crashed.

`sage` not rebooted until 12/26/06 (regional holiday in Seattle).

The Tribulation (continued)

1/06 9 P.M. Started mod 256 computation on sage. Computed 452,174 out of 453,060 rows of char table in 14 hours, then sage crashed.

2/06 EVENING Restarted mod 256. Finished in just 11 hours

(hip, hip, HURRAH! hip, hip, HURRAH! `pthread_join(cheer[k], NULL);`):

```
Total elapsed time = 40229s.  Finished at l = 64, y = 453059
d_store.size() = 1181642979, prim_size = 3393819659

VmData:  54995416 kB
```

3/06 Started mod 255 computation on sage, which crashed.

sage not rebooted until 12/26/06 (regional holiday in Seattle).

So we've got mod 256...

So we've got mod 256. . .

5/06 sage rebooted. Wrote character table mod 255.

So we've got mod 256. . .

6/06 sage rebooted. Wrote character table mod 255.

7/06 Started computation mod 253. Halfway, sage crashed.

So we've got mod 256. . .

6/06 sage rebooted. Wrote character table mod 255.

7/06 Started computation mod 253. Halfway, sage crashed.

consult experts $\rightsquigarrow$ probably not Sasquatch.

So we've got mod 256...

5/06 sage rebooted. Wrote character table mod 255.

7/06 Started computation mod 253. Halfway, sage crashed.

consult experts $\rightsquigarrow$ probably not Sasquatch.

Did I mention sage is in Seattle?

So we've got mod 256...

5/06 sage rebooted. Wrote character table mod 255.

7/06 Started computation mod 253. Halfway, sage crashed.

consult experts $\rightsquigarrow$ probably not Sasquatch.

Did I mention sage is in Seattle?

Decided not to abuse sage further for a year.

So we've got mod 256. . .

/06 sage rebooted. Wrote character table mod 255.

/06 Started computation mod 253. Halfway, sage crashed.

consult experts $\rightsquigarrow$ probably not Sasquatch.

Did I mention sage is in Seattle?

Decided not to abuse sage further for a year.

/07 Atlas members one year older $\rightsquigarrow$ thirty years wiser as team $\rightsquigarrow$ safe to go back to work.

So we've got mod 256...

/06 sage rebooted. Wrote character table mod 255.

/06 Started computation mod 253. Halfway, sage crashed.

consult experts ⇝ probably not Sasquatch.

Did I mention sage is in Seattle?

Decided not to abuse sage further for a year.

/07 Atlas members one year older ⇝ thirty years wiser as team ⇝ safe to go back to work.

Wrote character table mod 253 (12 hrs).

So we've got mod 256. . .

/06 sage rebooted. Wrote character table mod 255.

/06 Started computation mod 253. Halfway, sage crashed.

consult experts ⇝ probably not Sasquatch.

Did I mention sage is in Seattle?

Decided not to abuse sage further for a year.

/07 Atlas members one year older ⇝ thirty years wiser as team ⇝ safe to go back to work.

Wrote character table mod 253 (12 hrs).

Now we had answers mod 253, 255, 256.

Chinese Remainder Theorem (CRT)

gives answer mod 253·255·256 = 16,515,840.

So we've got mod 256...

5/06 sage rebooted. Wrote character table mod 255.

7/06 Started computation mod 253. Halfway, sage crashed.

consult experts ⇝ probably not Sasquatch.

Did I mention sage is in Seattle?

Decided not to abuse sage further for a year.

3/07 Atlas members one year older ⇝ thirty years wiser as team ⇝ safe to go back to work.

Wrote character table mod 253 (12 hrs).

Now we had answers mod 253, 255, 256.
Chinese Remainder Theorem (CRT)
gives answer mod 253·255·256 = 16,515,840.

One little computation for each of 13 billion coefficients.

The Chinese Remainder

The Chinese Remainder

4/07 Marc van Leeuwen started his CRT software. On-screen counter displayed polynomial number: $0, 1, 2, 3, \ldots, 1181642978$. Turns out that's a bad idea.

The Chinese Remainder

/07 Marc van Leeuwen started his CRT software.
On-screen counter displayed polynomial number:
$0, 1, 2, 3, \ldots, 1181642978$. Turns out that's a bad idea.

/07 MORNING Restarted CRT computation, with counter
$0, 4096, 8192, 12288, 16536, \ldots, 1181642752, 1181642978$.
Worked fine until sage crashed.

The Chinese Remainder

/07 Marc van Leeuwen started his CRT software.
On-screen counter displayed polynomial number:
$0, 1, 2, 3, \ldots, 1181642978.$ Turns out that's a bad idea.

/07 MORNING Restarted CRT computation, with counter
$0, 4096, 8192, 12288, 16536, \ldots, 1181642752, 1181642978.$
Worked fine until sage crashed.

William Stein (our hero!) replaced hard drive with one with backups of our 100G of files mod 253, 255, 256.

The Chinese Remainder

4/07 Marc van Leeuwen started his CRT software. On-screen counter displayed polynomial number: $0, 1, 2, 3, \ldots, 1181642978$. Turns out that's a bad idea.

5/07 MORNING Restarted CRT computation, with counter $0, 4096, 8192, 12288, 16536, \ldots, 1181642752, 1181642978$. Worked fine until sage crashed.

William Stein (our hero!) replaced hard drive with one with backups of our 100G of files mod 253, 255, 256.

5/07 AFTERNOON Re-restarted CRT computation.

The Chinese Remainder

4/07 Marc van Leeuwen started his CRT software. On-screen counter displayed polynomial number: $0, 1, 2, 3, \ldots, 1181642978$. Turns out that's a bad idea.

5/07 MORNING Restarted CRT computation, with counter $0, 4096, 8192, 12288, 16536, \ldots, 1181642752, 1181642978$. Worked fine until sage crashed.

William Stein (our hero!) replaced hard drive with one with backups of our 100G of files mod 253, 255, 256.

5/07 AFTERNOON Re-restarted CRT computation.

6/07 7 A.M. Output file 7G too big: BUG in output routine.

The Chinese Remainder

4/07 Marc van Leeuwen started his CRT software.
On-screen counter displayed polynomial number:
$0, 1, 2, 3, \ldots, 1181642978$. Turns out that's a bad idea.

5/07 MORNING Restarted CRT computation, with counter
$0, 4096, 8192, 12288, 16536, \ldots, 1181642752, 1181642978$.
Worked fine until sage crashed.

William Stein (our hero!) replaced hard drive with one
with backups of our 100G of files mod 253, 255, 256.

5/07 AFTERNOON Re-restarted CRT computation.

6/07 7 A.M. Output file 7G too big: BUG in output routine.

7/07 2 A.M. Marc found output bug. Occurred only after
polynomial 858,993,459; had tested to 100 million.

The Chinese Remainder

4/07 Marc van Leeuwen started his CRT software. On-screen counter displayed polynomial number: $0, 1, 2, 3, \ldots, 1181642978$. Turns out that's a bad idea.

5/07 MORNING Restarted CRT computation, with counter $0, 4096, 8192, 12288, 16536, \ldots, 1181642752, 1181642978$. Worked fine until sage crashed.

William Stein (our hero!) replaced hard drive with one with backups of our 100G of files mod 253, 255, 256.

5/07 AFTERNOON Re-restarted CRT computation.

6/07 7 A.M. Output file 7G too big: BUG in output routine.

7/07 2 A.M. Marc found output bug. Occurred only after polynomial 858,993,459; had tested to 100 million.

7/07 6 A.M. Re-re-restarted CRT computation.

n Which we Come to an Enchanted Place...

n Which we Come to an Enchanted Place...

/07 9 A.M. Finished writing to disk the character table of E_8.

n Which we Come to an Enchanted Place. . .

/07 9 A.M. Finished writing to disk the character table of E_8.

So what was the point?

n the fall of 2004, Fokko du Cloux was at MIT, rooming with fellow atlas member Dan Ciubotaru. Fokko was halfway through writing he software I've talked about: the point at which neither the end of he tunnel nor the beginning is visible any longer.

n Which we Come to an Enchanted Place. . .

/07 9 A.M. Finished writing to disk the character table of E_8.

So what was the point?

n the fall of 2004, Fokko du Cloux was at MIT, rooming with fellow atlas member Dan Ciubotaru. Fokko was halfway through writing he software I've talked about: the point at which neither the end of he tunnel nor the beginning is visible any longer.

Walking home after a weekend in the math department, Dan said, "Fokko, look at us. We're spending Sunday alone at work."

n Which we Come to an Enchanted Place. . .

/07 9 A.M. Finished writing to disk the character table of E_8.

So what was the point?

n the fall of 2004, Fokko du Cloux was at MIT, rooming with fellow atlas member Dan Ciubotaru. Fokko was halfway through writing he software I've talked about: the point at which neither the end of he tunnel nor the beginning is visible any longer.

Walking home after a weekend in the math department, Dan said, "Fokko, look at us. We're spending Sunday alone at work."

Fokko was startled by this remark, but not at a loss for words. "I don't know about you, but I'm having the time of my life!"

n Which we Come to an Enchanted Place…

3/07 9 A.M. Finished writing to disk the character table of E_8.

Fokko du Cloux

December 20, 1954–November 10, 2006