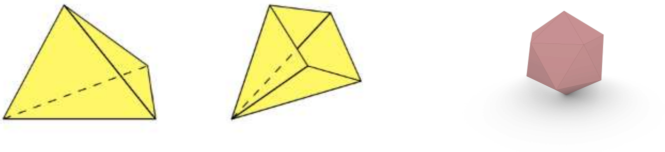
Simplicial Surfaces GAP Package

Reymond Akpanya, Tom Görtzen and Meike Weiß

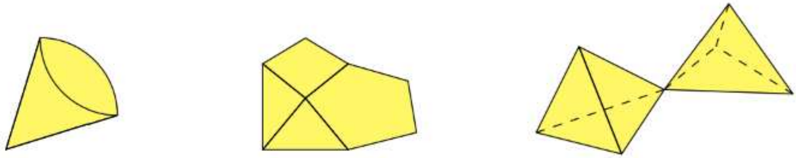
Lehrstuhl für Algebra und Darstellungstheorie - RWTH Aachen

GAP Days Summer 2022, 18th October 2022

Simplicial Surfaces



The package allows computations with generalisations of triangulations like polygonal complexes:



When examining simplicial surfaces, one can consider them:

1. In a combinatorial way $\rightarrow$ GAP package
2. By working with embeddings $\rightarrow$ Maple package by Daniel Robertz

Functionalities of the Package

- Constructing surfaces
- Properties of complexes:
  - Orientability
  - Connectivity
  - Degree-based properties
  - Euler-Characteristic
  - ...
- Modifying complexes
- Homomorphisms
- Graphs of simplicial surfaces
- Edge coloured surfaces
- Drawing simplicial surfaces
- ...

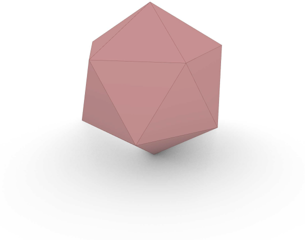
Functionalities of the Package

- Constructing surfaces
- Properties of complexes:
  - Orientability
  - Connectivity
  - Degree-based properties
  - Euler-Characteristic
  - ...
- Modifying complexes
- Homomorphisms
- Graphs of simplicial surfaces
- Edge coloured surfaces
- Drawing simplicial surfaces
- ...

Constructing surfaces

Different combinatorial ways for constructing surfaces:

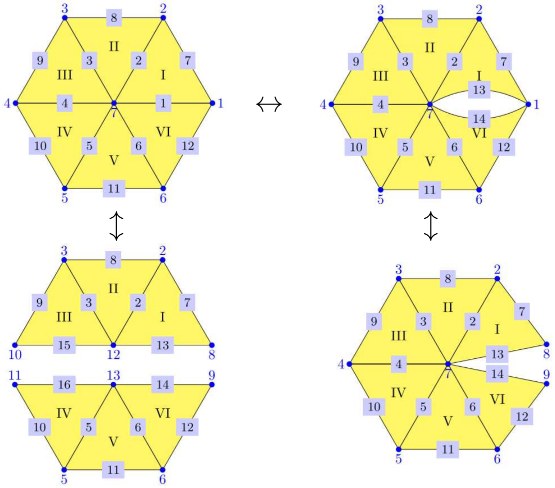
- With the vertices of edges and the edges of faces
- With the vertices of faces
- With the faces of vertices $\rightarrow$ Umbrella Descriptor



```
gap> SimplicialSurfaceByUmbrellaDescriptor([ (1,5,4,3,2), (1,6,11,7,2), (1,6,15,10,5),
> (2,7,12,8,3), (3,8,13,9,4), (4,9,14,10,5), (6,15,20,16,11), (7,12,17,16,11),
> (8,13,18,17,12), (9,14,19,18,13), (10,14,19,20,15), (16,17,18,19,20) ]);
simplicial surface (12 vertices, 30 edges, and 20 faces)
```

Modifying complexes

- Splitting/Joining along a path
- Removing faces
- Disjoint Union



Facegraph

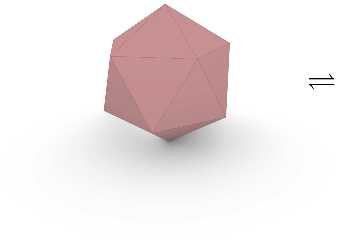
From surface to graph:

- Nodes: the faces of the simplical surfaces
- Edges: the edges between faces of the simplical surfaces

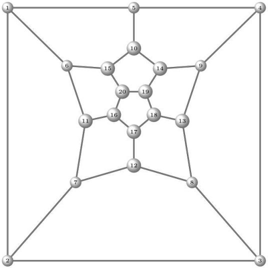
$\Rightarrow$ Facegraph of a simplicial surface is unique

From graph to surface: In general, the corresponding simplicial surface of a cubic graph is not unique.

Both ways can be calculated in the package.



$\rightleftharpoons$



Drawings/Animations in the SimplicialSurfaces Package

- DrawSurfaceToTikZ (folding plan)
- DrawFacegraphToTikZ (embedding of face graph)
- DrawSurfaceToJavascript (3D-animation of embedding)

Drawings/Animations in the SimplicialSurfaces Package

- DrawSurfaceToTikZ (folding plan)
- DrawFacegraphToTikZ (embedding of face graph)
- DrawSurfaceToJavascript (3D-animation of embedding)

Drawings/Animations in the SimplicialSurfaces Package

- DrawSurfaceToTikZ (folding plan)
- DrawFacegraphToTikZ (embedding of face graph)
- DrawSurfaceToJavascript (3D-animation of embedding)

Drawings/Animations in the SimplicialSurfaces Package

- DrawSurfaceToTikZ (folding plan)
- DrawFacegraphToTikZ (embedding of face graph)
- DrawSurfaceToJavascript (3D-animation of embedding)

Drawings/Animations in the SimplicialSurfaces Package

- DrawSurfaceToTikZ (folding plan)
- DrawFacegraphToTikZ (embedding of face graph)
- DrawSurfaceToJavascript (3D-animation of embedding)

Drawings/Animations in the SimplicialSurfaces Package

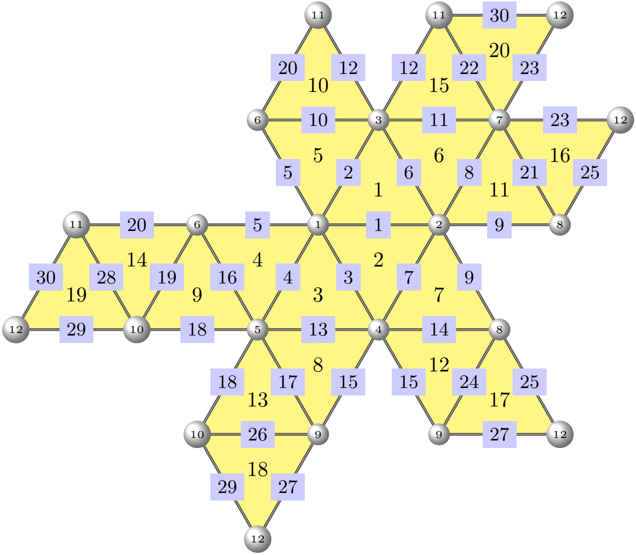
- DrawSurfaceToTikZ (folding plan)
- DrawFacegraphToTikZ (embedding of face graph)
- DrawSurfaceToJavascript (3D-animation of embedding)

DrawSurfaceToTikZ

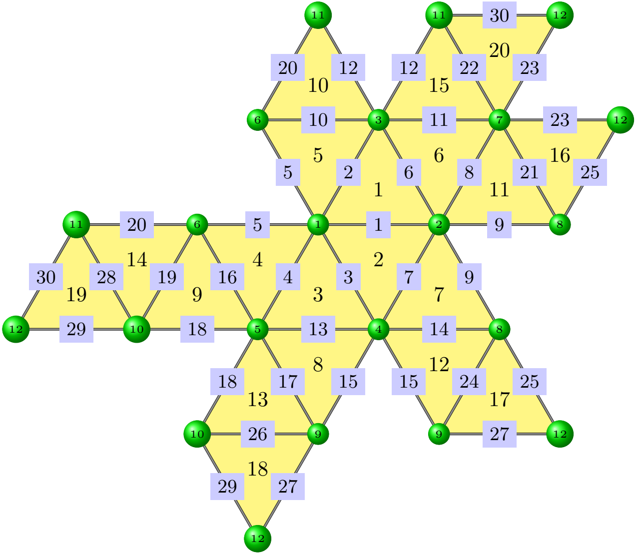
DrawSurfaceToTikZ draws the net of a surface into the plane.

```
gap> DrawSurfaceToTikz(Icosahedron(),"icosahedron");;
Picture written in TikZ.
```

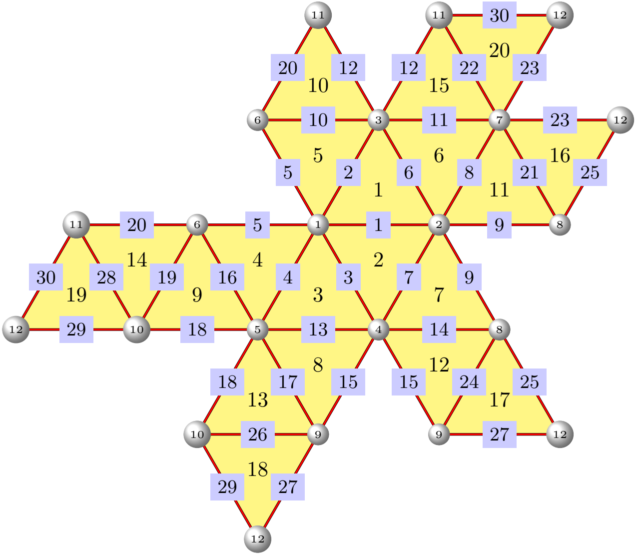
DrawSurfaceToTikZ



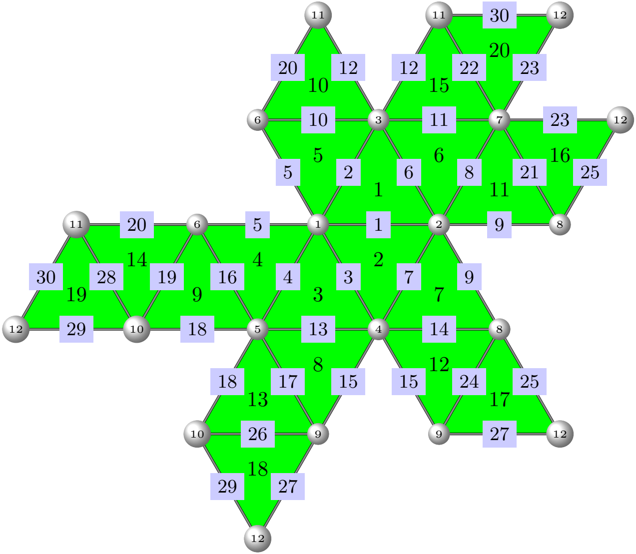
DrawSurfaceToTikZ



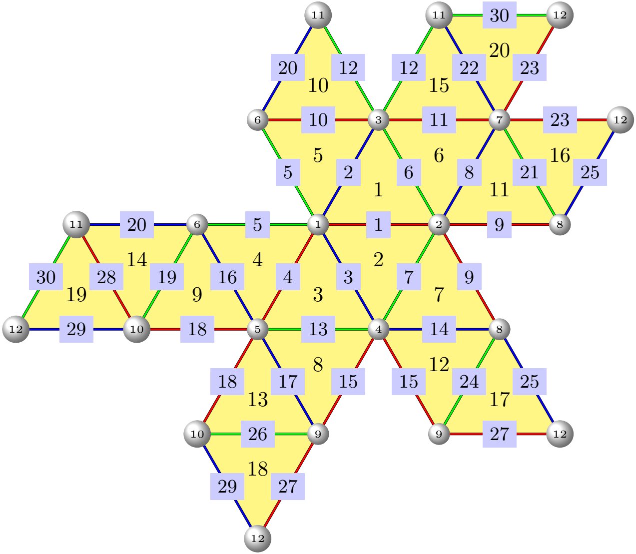
DrawSurfaceToTikZ



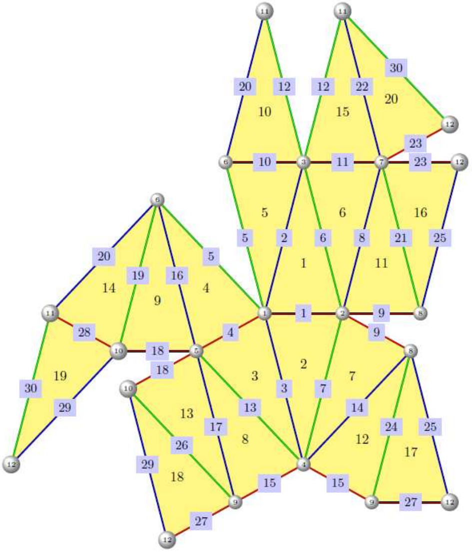
DrawSurfaceToTikZ



DrawSurfaceToTikZ

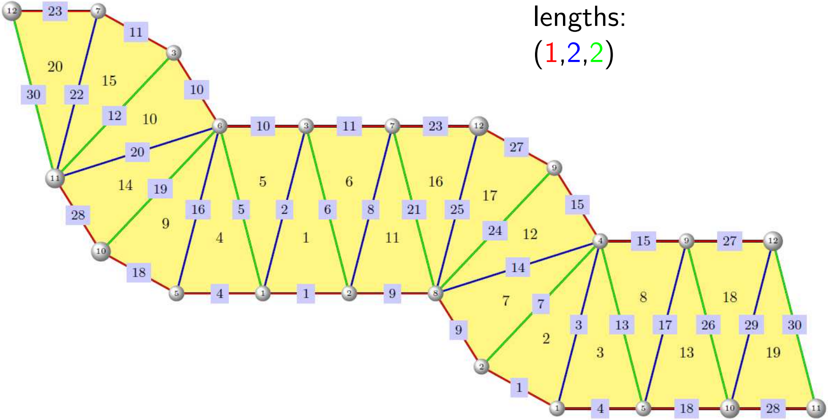


DrawSurfaceToTikZ



edge lengths: (1,2,2)

DrawSurfaceToTikZ



edge
lengths:
(1,2,2)

DrawFacegraphToTikZ

DrawFacegraphToTikZ draws the face graph of a simplicial surface into the euclidean plane. (straight line embedding)

```
gap> DrawFacegraphToTikz(Icosahedron(),"facegraph_presentation");;
Picture written in TikZ.
```

DrawFacegraphToTikZ

DrawFacegraphToTikZ draws the face graph of a simplicial surface into the euclidean plane. (straight line embedding)

```
gap> DrawFacegraphToTikz(Icosahedron(),"facegraph_presentation");;
Picture written in TikZ.
```

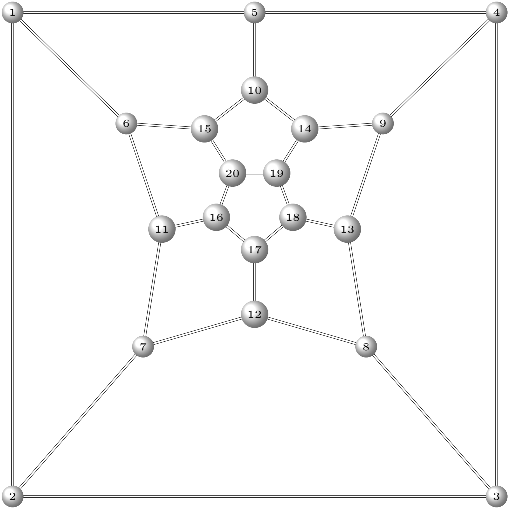
DrawFacegraphToTikZ

DrawFacegraphToTikZ draws the face graph of a simplicial surface into the euclidean plane. (straight line embedding)

```
gap> DrawFacegraphToTikz(Icosahedron(),"facegraph_presentation");;
Picture written in TikZ.
```

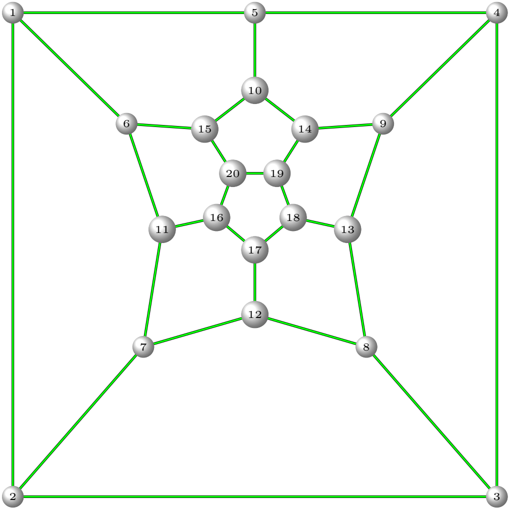
DrawFacegraphToTikZ

Example:



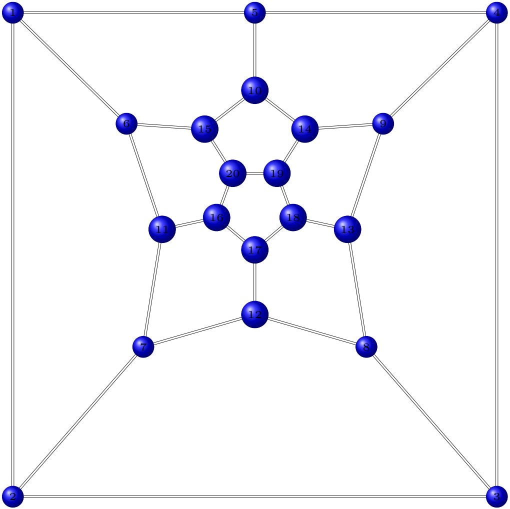
DrawFacegraphToTikZ

Example:



DrawFacegraphToTikZ

Example:



DrawSurfaceToJavascript

One central question about simplicial surfaces is whether they can be embedded into $\mathbb{R}^3$. To compute an embedding of a simplicial surface X (out of equilateral triangles), we have to find a map $\phi : X_0 \to \mathbb{R}^3$ so that

$$\|\phi(V) - \phi(V')\| = 1$$

for neighbouring vertices V and V'. This results in a system of quadratic equations.(Maple)

For example (Icosahedron):

- 30 quadratic equation
- $3 * 12$ indeterminates

DrawSurfaceToJavascript

One central question about simplicial surfaces is whether they can be embedded into $\mathbb{R}^3$. To compute an embedding of a simplicial surface X (out of equilateral triangles), we have to find a map $\phi : X_0 \to \mathbb{R}^3$ so that

$$\|\phi(V) - \phi(V')\| = 1$$

for neighbouring vertices V and V'. This results in a system of quadratic equations.(Maple)

For example (Icosahedron):

- 30 quadratic equation
- $3 * 12$ indeterminates

DrawSurfaceToJavascript

One central question about simplicial surfaces is whether they can be embedded into $\mathbb{R}^3$. To compute an embedding of a simplicial surface X (out of equilateral triangles), we have to find a map $\phi : X_0 \to \mathbb{R}^3$ so that

$$\|\phi(V) - \phi(V')\| = 1$$

for neighbouring vertices V and V'. This results in a system of quadratic equations.(Maple)

For example (Icosahedron):

- 30 quadratic equation
- $3 * 12$ indeterminates

DrawSurfaceToJavascript

One central question about simplicial surfaces is whether they can be embedded into $\mathbb{R}^3$. To compute an embedding of a simplicial surface X (out of equilateral triangles), we have to find a map $\phi : X_0 \to \mathbb{R}^3$ so that

$$\|\phi(V) - \phi(V')\| = 1$$

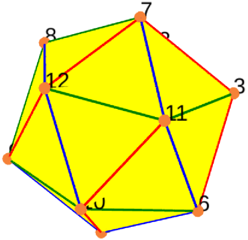
for neighbouring vertices V and V'. This results in a system of quadratic equations.(Maple)

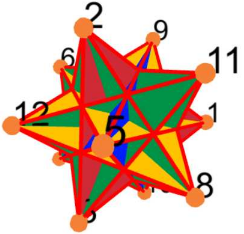
For example (Icosahedron):

- 30 quadratic equation
- $3 * 12$ indeterminates

By specifying the 3D-coordinates a animation of the surface can be generated.

By specifying the 3D-coordinates a animation of the surface can be generated.

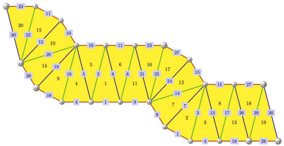




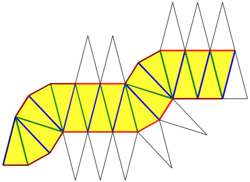
Folding and Cutting - Cricut Maker

Want to fold surfaces:

- Free choice of edge lengths
- Create SVG file with GAP
- Print → Cut → Fold → Glue :)



Tikz-Picture



SVG-Picture adapted to Cricut Maker

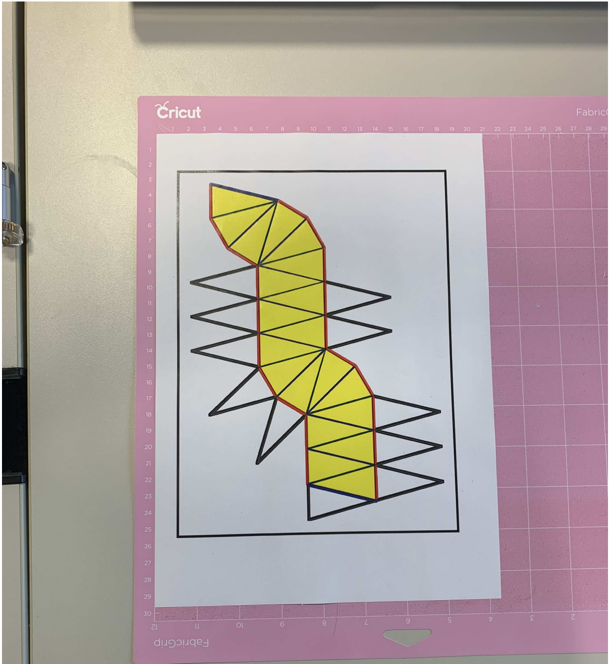
```
gap> ico:=AllWildColouredSurfaces(Icosahedron())[1];;
gap> pr:=rec();;
gap> pr.edgeDrawOrder:=[[29,26,17,13,3,7,14,24,25,21,8,6,2,5,16,19,20,12,22,30]];;
gap> pr.edgeColourClassLengths:=[1,2,2];;
gap> pr.edgeColourClassColours:=["red","green","blue"];;
gap> pr.faceColours:=List(Faces(ico),i->"yellow");
[ "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow", "yellow" ]
gap> pr.scale:=100;
100
gap> pr.AddFlapTriangle:=true;
true
gap> pr.AddFlaps:=false;
false
gap> pr.AddCircle:=false;
false
gap> pr:=DrawSurfaceToSVG(ico,"Examples/icosahedron_test",pr);;
Picture written in SVG.
```

Cricut Maker

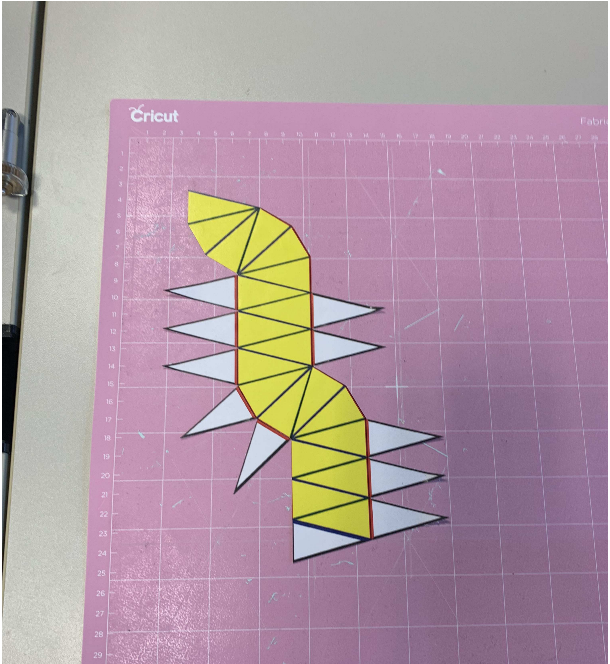
Cricut Maker

Cricut Maker

Cricut Maker



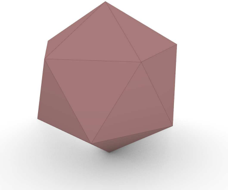
Cricut Maker



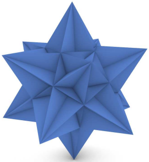
3D-Modelling and 3D-Printing of Surfaces

Want to 3D-model and 3D-print surfaces:

- Triangles can intersect
- Geometric properties
- Compute nice triangulation → STL-file → Print :)



Icosahedron (20 Faces)

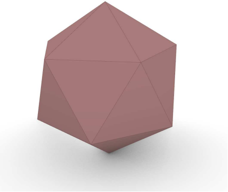


Self-Intersecting Icosahedron (20 Faces)

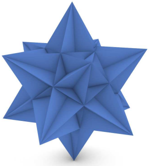
```
gap> Icosahedron();
simplicial surface (12 vertices, 30 edges, and 20 faces)
gap> data:=calculate_intersections(VerticesOfFaces(Icosahedron()),coordinates,false);;
gap> points:=data[1];;
gap> t:=TriangularComplexByVerticesInFaces(data[2]);
triangular complex (214 vertices, 1140 edges, and 1340 faces)
```

3D-Modelling and 3D-Printing of Surfaces

Want to 3D-model and 3D-print surfaces:



Icosahedron (20 Faces)



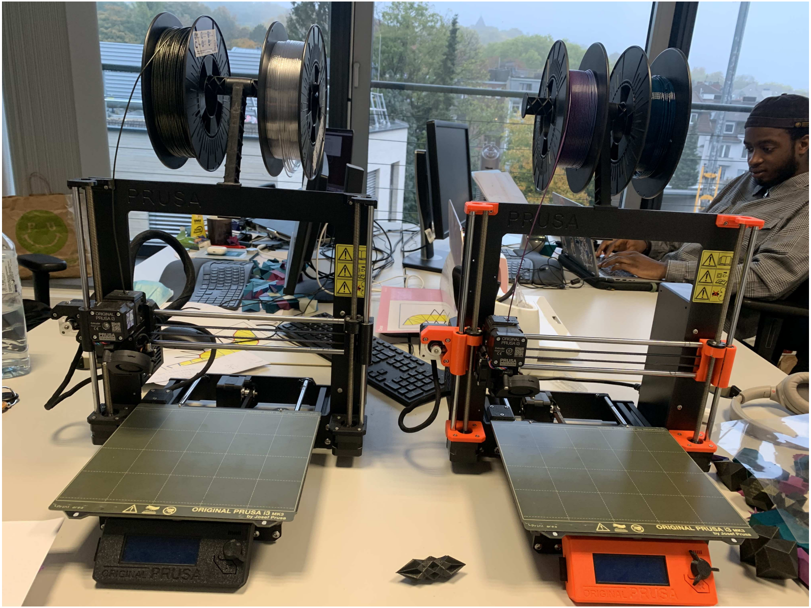
Self-Intersecting Icosahedron (20 Faces)

```
gap> Icosahedron();
simplicial surface (12 vertices, 30 edges, and 20 faces)
gap> data:=calculate_intersections(VerticesOfFaces(Icosahedron()),coordinates,false);;
gap> points:=data[1];;
gap> t:=TriangularComplexByVerticesInFaces(data[2]);
triangular complex (214 vertices, 1140 edges, and 1340 faces)

gap> components:=components(t,points);;
gap> Size(components[1]);
414
gap> components[1][2];
simplicial surface (92 vertices, 270 edges, and 180 faces)
gap> DrawSTLwithNormals(components[1][2],"ico_2_1",points,components[2][2],[]);

 Saved file
gap> # components[2][i] saves normals for each face contained in the surface components[1][i]
```

3D-Printing


PRUSA
PRUSA
ORIGINAL PRUSA i3 MK3
by Josef Prusa
ORIGINAL PRUSA i3 MK3
by Josef Prusa
ORIGINAL PRUSA
ORIGINAL PRUSA

Next steps and TODOs

- Include Folding and Printing functions
- Simplify compiling the manual (434 pages)
- Simplify installing the package
- Work on package dependencies (AttributeScheduler, GRAPE, DIGRAPH, NAUTYTRACES, GAPDOC)