

### Hands-on Tropical Geometry

H. Gangl, Durham University

Y. Ren, Durham University

Z. Urbancic, Durham University

herbert.gangl@durham.ac.uk,

yue.ren2@durham.ac.uk,

ziva.urbancic@durham.ac.uk



---

### Introduction

Tropical varieties are piecewise linear structures that arise in many areas inside and outside mathematics. They describe the loci of indifference prices in product mix auctions [1], spaces of phylogenetic trees [11, 15], or images of solutions of polynomial systems under component-wise valuation [9]. Geometrically, tropical varieties are strongly tied to their classical algebraic counterparts, they are strikingly similar in some aspects yet also fascinatingly different in others. However, while there has been a significant effort in modelling algebraic surfaces and curves - from Schilling's white plaster casts popularized by Klein in the 19th century [5, 14] to 3D-printed models of today [10] [3] - nothing comparable exists for tropical varieties despite the simplicity of their polyhedral nature.

This article aims to address this gap by presenting a comprehensive guide on how to create 3D-printable models of tropical surfaces, tropical curves, and combinations thereof. We will make use of the following open-source software, which are freely available on their respective websites for all platforms:

**POLYMAKE [4]** (<https://polymake.org>) for creating mathematically accurate tropical models.

**OPENSCAD [12]** (<https://openscad.org>) for solidifying the tropical models.

We will only focus on the creation of 3D models in OPENSCAD, which in turn is capable of exporting to a variety of file formats common in 3D printing such as STL. Exporting color is also supported by 3rd party scripts such as COLORSCAD. The actual 3D printing process depends significantly on the printer, the material, and the associated printing software used.

All scripts and templates in this report can be found in the official POLYMAKE repository and will be part of the official POLYMAKE distribution from version 4.6 onward. The notebook is also publicly available under

[https://polymake.org/doku.php/user\\_guide/tutorials/master/hands\\_on\\_tropical\\_geometry](https://polymake.org/doku.php/user_guide/tutorials/master/hands_on_tropical_geometry)

---

### Summary

This article assumes some familiarity with the basic objects in tropical geometry. If any terminology is unknown, refer to [9] and [7].

The starting point for creating a 3D-model is one or more polyhedral complexes in  $\mathbb{R}^3$ . In POLYMAKE, objects of type `fan::PolyhedralComplex` can always be constructed manually by specifying their POINTS and INPUT\_POLYTOPES. In special cases, which are covered in the upcoming sections, they can also be constructed via some shortcuts.

Given polyhedral complexes in POLYMAKE, the 3D-model can then be created in four steps:

1. Construct a bounding box in POLYMAKE. This can be done automatically using our script (which draws a box with a prescribed margin around all vertices) or manually by specifying any bounded polytope (which need not be a cuboid). Use “`->VISUAL`” to verify that the bounded box cuts off the polyhedral complex as desired.
2. Export the model from POLYMAKE to OPENSCAD using our script.
3. Adjust the thicknesses in OPENSCAD. Use the preview window to verify that the thickness is as desired.
4. Export the model from OPENSCAD to any 3D-printable file format.

In the following sections, we will specifically discuss the cases of

- a single tropical surface,
- a single tropical curve,
- a tropical surface containing a tropical curve.

The methods used can however be combined to cover arbitrary arrangements of surfaces and curves.

---

## Modelling Tropical Surfaces

---

For an example of producing a 3D-printable model of a tropical surface see the file

3d\_printing\_template\_surface.pl.

To run the example, simply copy its contents into any POLYMAKE session. In this section, we briefly explain some of its contents. Note that some variables are re-named in this article due to spacing.

### Constructing a tropical surface in POLYMAKE

An easy way to construct a tropical surface in  $\mathbb{R}^3$  of type `fan::PolyhedralComplex` in POLYMAKE is by constructing and converting a `tropical::Hypersurface`. This can be done either by specifying a tropical polynomial as a string, or by specifying an exponent matrix and a coefficient vector:

---

```
# Tropical quadratic surface via polynomial:
$f = toTropicalPolynomial("min(1+2*w,1+2*x,1+2*y,
                             1+2*z,w+x,w+y,w+z,
                             x+y,x+z,y+z)");

$T = new tropical::Hypersurface<Min>(POLYNOMIAL=>
                                     $f);

# Tropical quadratic surface via exponent matrix
# and coefficient vector:
$m = [[2,0,0,0], [1,1,0,0], [1,0,1,0], [1,0,0,1],
      [0,1,1,0], [0,1,0,1], [0,0,1,1], [0,2,0,0],
      [0,0,2,0], [0,0,0,2]];
$c = [1,0,0,0,0,0,0,1,1,1];
$T = new tropical::Hypersurface<Min>(MONOMIALS=>$m
                                     ,
                                     COEFFICIENTS=>$c);

# Converting Hypersurface to PolyhedralComplex
$T = new fan::PolyhedralComplex(
    VERTICES=>$T->VERTICES->minor(All,~[1]),
    MAXIMAL_POLYTOPES=>$T->MAXIMAL_POLYTOPES);

# Visualization:
$T->VISUAL;
```

---

**Important:** Since POLYMAKE uses homogeneous coordinates, i.e., it identifies polyhedra in  $\mathbb{R}^3$  with cones in  $\mathbb{R}^4$ , tropical polynomials need to be tetravariate and homogeneous instead of trivariate and inhomogeneous.

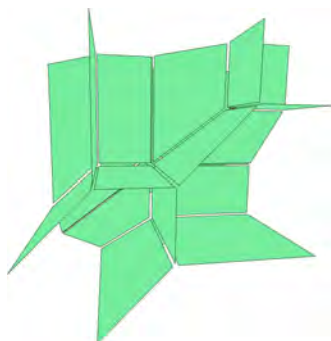


Figure 1: Tropical quadratic surface in POLYMAKE.

### Bounding a tropical surface in POLYMAKE and exporting it to OPENSCAD

Constructing a bounding box can be done using the command `generateBoundingBox` or by specifying a custom polytope, which need not be a cuboid:

---

```
# Constructing bounding box automatically:
$bBox = generateBoundingBox($T);
# Constructing bounding box manually:
$bBox = scale(cube(3),2);
```

---

Note: `generateBoundingBox($T)` draws a box around all vertices of `$T` with a margin of 1, `generateBoundingBox($T,3,4,5)` draws a box around all vertices of `$T` with margins 3, 4, 5 in  $x$ -,  $y$ -, and  $z$ -direction, respectively.

The command `intersectWithBoundingBox` intersects the `fan::PolyhedralComplex` with the bounding box, `->VISUAL` visualizes their intersection:

---

```
# Constructing bounded PolyhedralComplex:
$Tbounded = intersectWithBoundingBox($T,$bBox);
$Tbounded->VISUAL;
```

---

The command `generateSCADFileForSurface` exports the tropical surface to OPENSCAD. It requires a bounded polyhedral complex and a filename. If the file already exists, it will be overwritten:

---

```
$filename = "foo.scad";
generateSCADFileForSurface($Tbounded,$filename);
```

---

### Solidifying a tropical surface in OPENSCAD and exporting it for 3D-printing

To solidify the surface into a three-dimensional model, open the exported file in OPENSCAD. See Figure 2 for a preview of the model.

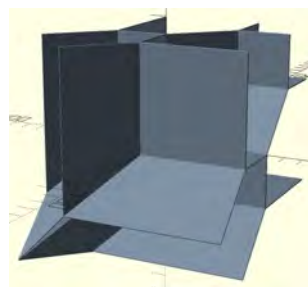


Figure 2: Solidified tropical quadratic surface in OPENSCAD.

The top of the file contains parameters which control the thickness of the model amongst other things:

---

```
colorSurface = "SlateGray"; // color of surface
scalingFactor = 1; // global scaling factor
thicknessSurface = 0.05; // thickness of surface
```

---

- `colorSurface` is the color of the surface in the render. It can be specified either by an HTML color name or by RGB value.
- `scalingFactor` is a factor by which the polyhedral complex is scaled. It can be used to scale the model to a desired size.
- `thicknessSurface` controls the thickness of the surface. OPENSCAD solidifies the surface by replacing every vertex with a ball with specified thickness and taking convex hulls.

**Important:** In order to 3D-print the tropical surface, every face needs to have a minimal thickness. As a rule of thumb for printing with PLA, we recommend a minimal thickness of 2mm for a print of size (10cm)<sup>3</sup>.

## Tropical Curves

For an example of producing a 3D-printable model of a tropical curve see the file

3d\_printing\_template\_curve.pl.

To run the example, simply copy its contents into any POLYMAKE session. In this section, we briefly explain some of its contents. Note that some variables are renamed due to spacing.

### Constructing a tropical curve in POLYMAKE

An easy way to construct a tropical curve in  $\mathbb{R}^3$  of type `fan::PolyhedralComplex` in POLYMAKE is by intersecting two tropical surfaces. Note however that not all tropical curves can be constructed this way.

```
# Tropical sextic curve that is the intersection
# of a tropical quadratic and cubic surface
$Mq = [[2,0,0,0], [0,2,0,0], [0,0,2,0], [0,0,0,2],
       [1,1,0,0], [1,0,1,0], [1,0,0,1],
       [0,1,1,0], [0,1,0,1], [0,0,1,1]];
$Cq = [1, -1/4, -2/4, -3/4, -3/4,
       -4/4, -5/4, 2/4, 0, -2/4];
$Tq = new tropical::Hypersurface<Min>(
    MONOMIALS=>$Mq,
    COEFFICIENTS=>$Cq);
$Mc = [[3,0,0,0], [0,3,0,0], [0,0,3,0], [0,0,0,3],
       [1,1,1,0], [1,1,0,1], [1,0,1,1], [0,1,1,1],
       [2,1,0,0], [2,0,1,0], [2,0,0,1],
       [1,2,0,0], [1,0,2,0], [1,0,0,2],
       [0,2,1,0], [0,2,0,1], [0,1,2,0],
       [0,1,0,2], [0,0,2,1], [0,0,1,2]];
$Cc = [3,3,3,3,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1];
$Tc = new tropical::Hypersurface<Min>(
    MONOMIALS=>$Mc,
    COEFFICIENTS=>$Cc);
$Ts = tropical::intersect($Tq,$Tc);
# Converting Cycle to PolyhedralComplex
$Ts = new fan::PolyhedralComplex(
    VERTICES=>$Ts->VERTICES->minor(All,~[1]),
    MAXIMAL_POLYTOPES=>$Ts->MAXIMAL_POLYTOPES);
```

**Important:** `intersect` is the intersection of tropical cycles. In terms of polyhedral complexes, it therefore computes the stable intersection, not the set-theoretic intersection.

Alternatively and in particular for tropical curves which are not realizable, they can also be manually constructed as a one-dimensional polyhedral complex by specifying vertices and maximal polyhedra:

```
# constructing a tropical genus-2 cubic curve
$V = [[1,-1,-1,0], [1,-4,-3,0], [1,-8,-5,0],
      [1,-9,-6,0], [1,-9,-7,0], [1,-8,-7,0],
      [1,-10,-8,0], [1,1,1,0], [1,4,4,1],
      [1,8,8,1], [1,9,9,2], [1,9,9,3],
      [1,8,8,3], [1,10,10,2], [0,-1,0,0],
      [0,0,-1,0], [0,0,0,-1], [0,1,1,1]];
$E = [[0,1], [1,2], [2,3], [3,4], [4,5],
      [5,1], [4,6], [0,15], [5,15], [6,15],
      [6,14], [3,14], [2,14], [0,7], [7,8],
      [8,9], [9,10], [10,11], [11,12], [12,8],
      [10,13], [7,16], [9,16], [13,16], [13,17],
      [11,17], [12,17]];
$Tc = new fan::PolyhedralComplex(VERTICES=>$V,
    MAXIMAL_POLYTOPES=>$E);
```

### Framing a tropical curve in POLYMAKE and exporting it to OPENSCAD

Constructing a bounding box can be done using the command `generateBoundingBox` as before:

```
$bBox = generateBoundingBox($Ts);
$Tbounded = intersectWithBoundingBox($Ts,$bBox);
```

For the sake of stability, we recommend constructing a frame for the tropical curve. This is best done using a tropical surface containing it:

```
$Tq = new fan::PolyhedralComplex(
    VERTICES=>$Tq->VERTICES->minor(All,~[1]),
    MAXIMAL_POLYTOPES=>$Tq->MAXIMAL_POLYTOPES);
$Tframe = intersectWithBoundingBoxForFraming($Tq,
    $bBox);
```

Exporting the frame and the tropical curve to OPENSCAD can be done using the command `generateSCADFileForCurve`. It requires two bounded polyhedral complexes and a filename. If the file already exists, it will be overwritten:

```
$filename = "foo.scad";
generateSCADFileForCurve($Tframe,$Ts,$filename);
```

### Solidifying a tropical curve in OPENSCAD and exporting it for 3D-printing

To solidify the curve into a three-dimensional model, open the exported file in OPENSCAD. See Figures 3 and 4 for a preview of the model.

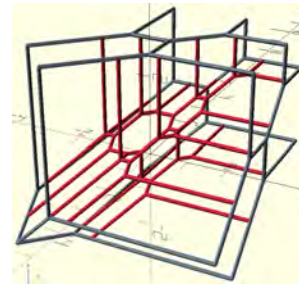


Figure 3: Framed and solidified tropical sextic curve in OPENSCAD.

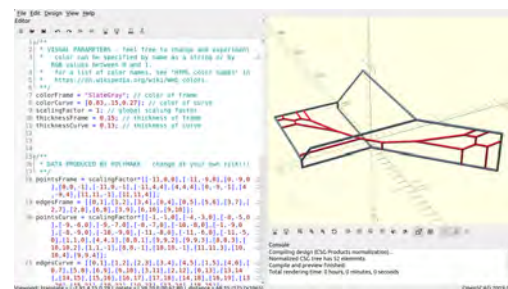


Figure 4: Framed and solidified genus-2 tropical cubic curve in OPENSCAD.

As before, the top of the file contains parameters which control the thickness of the model amongst other things:

```
colorFrame = "SlateGray"; // color of frame
colorCurve = [0.83,.15,0.27]; // color of curve
scalingFactor = 1; // global scaling factor
thicknessFrame = 0.05; // thickness of frame
thicknessCurve = 0.05; // thickness of curve
```

- `colorFrame` and `colorCurve` are the colors of the surface in the render. They can be specified either by an HTML color name or by RGB value.
- `scalingFactor` is a factor by which the polyhedral complex is scaled. It can be used to scale the model to the desired size.
- `thicknessFrame` and `thicknessCurve` control the thickness of the frame and curve. OPENSCAD solidifies both by replacing every vertex with a ball with diameter equal to the specified radius and taking convex hulls.

---

## Modelling Tropical Curves on Tropical Surfaces

---

For an example of producing a 3D-printable model of a tropical curve on a tropical surface see the file

`3d_printing_template_surface_and_curve.pl`.

To run the example, simply copy its contents into any POLYMAKE session. In this section, we briefly explain some of its contents. Note that some variables are renamed due to spacing.

### Constructing a curve and surface in POLYMAKE

The previous section showed an easy way to construct a tropical curve lying on a tropical surface in POLYMAKE: start with the surface and construct the curve by intersecting it with another surface. However, as mentioned before, this may not always be possible or easy to do. One option that always works is to construct the curve manually by specifying its vertices and edges:

```
# constructing tropical plane via polynomial
$t1=toTropicalPolynomial("max(y,z,w)",qw(w x y z));
$t1 = new tropical::Hypersurface<Max>(
    POLYNOMIAL=>$1);
$t1 = new fan::PolyhedralComplex(
    POINTS=>$t1->VERTICES->minor(All,~[1]),
    INPUT_POLYTOPES=>$t1->MAXIMAL_POLYTOPES,
    INPUT_LINEALITY=>$t1->LINEALITY_SPACE->
        minor(All,~[1]));

# constructing tropical curve manually:
$V = [[1,0,0,0],[1,-8,0,0],[1,-8,-6,0],
      [1,12,0,0],[0,0,-1,0],[1,-6,-6,0],
      [1,-11,-3,0],[1,-11,-6,0],[0,-1,0,0],
      [1,-12,-7,0],[1,4,0,-2],[1,10,0,-2],
      [1,8,0,-6],[0,0,0,-1],[0,1,1,1],
      [1,-4,6,6],[1,-4,5,5],[1,-3,5,5],
      [1,-5,4,4],[1,-3,3,3],[1,-5,3,3]];
$E = [[3,4],[0,5],[2,5],[4,5],[1,6],[6,7],
      [6,8],[2,7],[7,9],[8,9],[4,9],[10,11],
      [3,11],[11,12],[0,10],[10,12],[12,13],
      [1,13],[14,15],[8,15],[15,16],[14,17],
      [16,17],[16,18],[17,19],[8,18],[18,20],
      [0,19],[19,20],[1,20],[3,14]];
$tq = new fan::PolyhedralComplex(POINTS=>$V,
    INPUT_POLYTOPES=>$E);
```

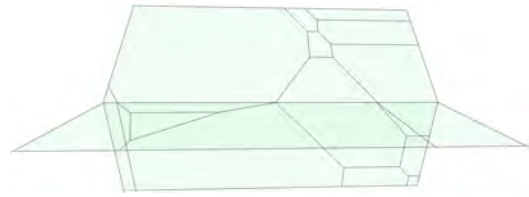
---

### Bounding a tropical surface in POLYMAKE and exporting it to OPENSCAD

Constructing a bounding box is easiest done using the command `generateBoundingBox` as before:

```
$bBox = generateBoundingBox($Tq);
$t1Bounded = intersectWithBoundingBox($t1,$bBox);
$tqBounded = intersectWithBoundingBox($tq,$bBox);
```

---



**Figure 5:** Tropical plane with a tropical quartic curve in POLYMAKE.

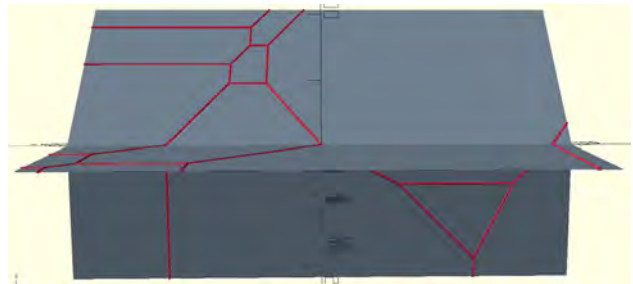
Exporting the tropical surface and curve to OPENSCAD can be done using the command `generateSCADFileForSurfaceAndCurve`. It requires two bounded polyhedral complexes and a file-name. If the file already exists, it will be overwritten:

```
$filename = "foo.scad";
generateSCADFileForSurfaceAndCurve($t1Bounded,
    $tqBounded,$filename);
```

---

### Solidifying a tropical surface with a tropical curve in OPENSCAD and exporting them for 3D-printing

To solidify the tropical surface with a tropical curve into a three-dimensional model, open the exported file in OPENSCAD. See Figure 6 for a preview of the model.



**Figure 6:** Solidified tropical plane with a tropical quartic curve in OPENSCAD.

The top of the file contains parameters which control the thickness of the model amongst other things:

```
colorSurface = "SlateGray"; // color of surface
colorCurve = [0.83,.15,0.27]; // color of curve
scalingFactor = 1; // global scaling factor
thicknessSurface = 0.01; // thickness of surface
thicknessCurve = 0.1; // thickness of curve
```

---

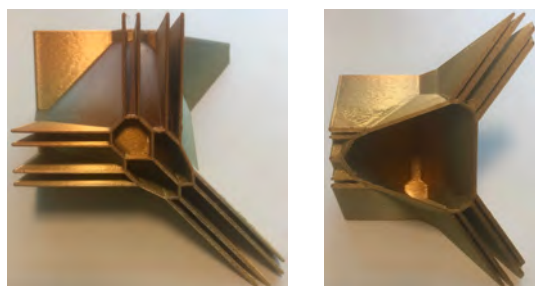
- `colorSurface` and `colorCurve` are the colors of the surface and curve in the render. It can be specified either by an HTML color name or by RGB value.
- `scalingFactor` is a factor by which the polyhedral complex is scaled. It can be used to scale the model to the desired size.
- `thicknessSurface` and `thicknessCurve` control the thickness of the surface and of the curve, respectively. OPENSCAD solidifies the surface by replacing every vertex with a ball with diameter equal to the specified radius and taking convex hulls.



## Gallery



**Figure 7:** A tropical cubic surface from [13] with 5 of 27 tropical lines drawn (one per motif).



**Figure 8:** A tropical quartic surface from [2], cut in half to reveal the inscribed  $K3$  polytope.



**Figure 9:** A tropical sextic curve from [6], the white frame traces the quadric surface containing it.

## References

[1] Elizabeth Baldwin, Paul Klempner: *Understanding Preferences: “Demand Types”, and the Existence of Equilibrium With Indivisibilities*. *Econometrica* **87**, No. 3, 867–932 (2019).

[2] Gabriele Balletti, Marta Panizzut, Bernd Sturmfels:  *$K3$  polytopes and their quartic surfaces*. *Adv. Geom.* **21**, No. 1, 85–98 (2021).

[3] *bertini\_real: a software for real algebraic sets*. Publicly available at <http://bertinireal.com>.

[4] Ewgenij Gawrilow, Michael Joswig: *polymake: a framework for analyzing convex polytopes*. In: *Polytopes—combinatorics and computation* (Oberwolfach, 1997), 43–73, DMV Sem., 29, Birkhäuser, Basel, 2000.

[5] Stefan Halverscheid, Oliver Labs: *Felix Klein’s Mathematical Heritage Seen Through 3D Models*. In: *The Legacy of Felix Klein*, 131–152, Springer International Publishing, 2019.

[6] Corey Harris, Yoav Len: *Tritangent planes to space sextics: the algebraic and tropical stories*. Smith, Gregory G. (ed.) et al., *Combinatorial algebraic geometry. Selected papers from the 2016 apprenticeship program, Ottawa, Canada, July–December 2016*. *Fields Institute Communications* **80**, 47–63 (2017).

[7] Michael Joswig: *Essentials of tropical combinatorics*. *Graduate Studies in Mathematics* **219**. Providence, RI: American Mathematical Society (2021).

[8] jschobben: *colorscad*. <https://github.com/jschobben/colorscad> (visited February 2022).

[9] Diane Maclagan, Bernd Sturmfels: *Introduction to tropical geometry*. *Graduate Studies in Mathematics* **161**. Providence, RI: American Mathematical Society (2015).

[10] *MO-Labs: The premium designer of 3D-printed and laser-in-glass mathematical models with a context in the history of mathematics*. <https://math-sculpture.com/> (visited December 2021).

[11] Anthea Monod, Bo Lin, Ruriko Yoshida, Qiwen Kang: *Tropical Geometry of Phylogenetic Tree Space: A Statistical Perspective* arXiv:1805.12400 (2018).

[12] *OpenSCAD: The Programmers Solid 3D CAD Modeller*. <https://openscad.org/>.

[13] Marta Panizzut, Magnus Dehli Vigeland: *Tropical lines on cubic surfaces*. *SIAM J. Discrete Math.* **36**, No. 1, 383–410 (2022).

[14] Martin Schilling: *Catalog mathematischer Modelle*. Martin Schilling Verlag, Leipzig, 1911. Gallery publicly available at <http://www.universitaetssammlungen.de/publikation/5290>.

[15] David Speyer, Bernd Sturmfels: *The tropical Grassmannian*. *Adv. Geom.* **4**, No. 3, 389–411 (2004).



## Antrag auf Mitgliedschaft in der Fachgruppe Computeralgebra

Die Fachgruppe Computeralgebra sieht es als ihre Aufgabe an, Lehre, Forschung, Entwicklung, Anwendungen, Informationsaustausch und Zusammenarbeit auf dem Gebiet der Computeralgebra in Deutschland zu fördern.

Eine Mitgliedschaft in der Fachgruppe Computeralgebra gibt es bereits ab 7,50 € pro Jahr (für Mitglieder von DMV, GI oder GAMM; ansonsten 9 €).

### Vorteile einer Mitgliedschaft:

- Sie fördern durch Ihren Beitrag die Workshops, Seminare, Tagungen und andere Aktivitäten auf dem Gebiet der Computeralgebra, die die Fachgruppe organisiert und unterstützt.
- Sie erhalten zweimal im Jahr den Computeralgebra-Rundbrief mit vielen interessanten Informationen rund um die Computeralgebra frei Haus.
- Sie verleihen unserer Stimme an Gewicht, die wir aktiv in Diskussionen um die Stellung der Computeralgebra in der Ausbildung in Schule und Hochschule einbringen.

Wir würden uns sehr über Ihre Unterstützung freuen. Die Mitgliedschaft in der Fachgruppe steht allen offen. Weiter Informationen zur Mitgliedschaft und einen Aufnahmeantrag finden Sie auf unserer Webseite unter folgender Adresse, oder scannen Sie einfach den QR-Code.

<https://fachgruppe-computeralgebra.de/aufnahmeantrag>

