
- We define ‘stereographic projection’ as a self-mapping of R^4 determined by a basepoint b, linear functional ell, and a prescribed level set {ell=ell_select}. This mapping concentrates points not incident with the ell-level through b to the three-dimensional affine subspace ell=ell_select, and otherwise maps points on ell=ell(b) to the unique point-at-infinity of the affine subspace.
- Relative to the fixed basepoint b, the point x is pushed along the directed ray b + {R>0}.(x-b) until hitting {ell=ell_select}. In formulas, the stereographic projection relative to (b,ell, ell_select) becomes

  x -> b+ [ell_select - ell(b) / ell(x) - ell(b)]. (x-b).

  One checks immediately that the image satisfies ell(image[x])==ell_select.
- The function stereo defined above is a projection relative to basepoint {1,0,1,0}, functional x1+x2, and level set -2. This basepoint is the image of {0,0} under the square embedding map T^2 -> R^4. It is furthermore the unique maximum of the linear functional restricted to the embedded torus image. This is convenient for imagery. Our choice of basepoint and functional ensures that stereo (restricted to the square embedded torus image) is injective into the 3-dimensional affine subspace. Computing coordinates of this image via an orthonormal 3-frame of ker(ell) finally yields a 3d parametric surface.

- Among 2-dimensional flat tori, the hexagonal torus is at least (or more) relevant than the split square. To find a stereographic image of the hexagonal torus, we postcompose the image of the square embedded torus by the complexified action of A={{1,1/2},{0,Sqrt[3]/2}} on R^4 (the mapping hex above). Our projection mapping stereohex is defined relative to the hexbase point. Notice that our choice of hexbase corresponds again to the unique maximum of the linear functional x1+x2 restricted to the hex//squaremb torus image. Again, the stereohex mapping yields an embedding of the hex-torus onto the affine 3-space.

- [? Certifying/witnessing square and hexagonal symmetry of the stereo images]. In the ambient R^4, the embedded square and hexagonal torus possess order 2 and order 3 (orientation preserving) linear automorphisms, respectively. These are witnessed by the complexified actions of the plane-rotations at angles $2\pi/2$, $2\pi/3$. However the stereograph data is not invariant as the basepoint and level sets are displaced.

- We expect watermelons to have a collared black & green outer region, with deep pink/red flesh. Our ability to hijack and manipulate the RGB colour space associated to the built-in “WatermelonColors” gradient scheme is thus far limited. We are unsatisfied with the colouring of our watermelon hexagonal torus.

```
squareemb[{t1_, t2_}] := {Cos[t1], Sin[t1], Cos[t2], Sin[t2]};

lin[{x1_, y1_, x2_, y2_}] := x1 + x2;

stereo[{x1_, y1_, x2_, y2_}] :=
  {1, 0, 1, 0} + (-4 / (x1 + x2 - 2.0001)) {x1 - 1, y1, x2 - 1, y2};

normalcoord[{x1_, y1_, x2_, y2_}] := {y1, (x1 - x2) / Sqrt[2], y2};

hexnormalcoord[{x1_, y1_, x2_, y2_}] := {x2, (x1 - y1) / Sqrt[2], y2};

hex[{x1_, y1_, x2_, y2_}] := {x1 + x2 / 2, y1 + y2 / 2, Sqrt[3] / 2 x2, Sqrt[3] / 2 y2};

hexbase = hex[{1, 0, 1, 0}] // N;

stereohex[{x1_, y1_, x2_, y2_}] :=
  hexbase + (-3 - 2.36603) / (x1 + x2 - 2.36603) ({x1, y1, x2, y2} - hexbase);

squarefunk[{t1_, t2_}] := normalcoord@*stereo@*squareemb@{t1, t2};

hexfunk[{t1_, t2_}] := normalcoord@*stereohex@*hex@*squareemb@{t1, t2};

three[{x1_, y1_, x2_, y2_}] :=
  {(-1 / 2) x1 - (Sqrt[3] / 2) x2, (-1 / 2) y1 - (Sqrt[3] / 2) y2,
   (-1 / 2) x2 + (Sqrt[3] / 2) x1 , (-1 / 2) y2 + (Sqrt[3] / 2) y1};
```

```
{1, 0, 1, 0} // hex // three // three // three
```

$\left\{\frac{3}{2}, 0, \frac{\sqrt{3}}{2}, 0\right\}$

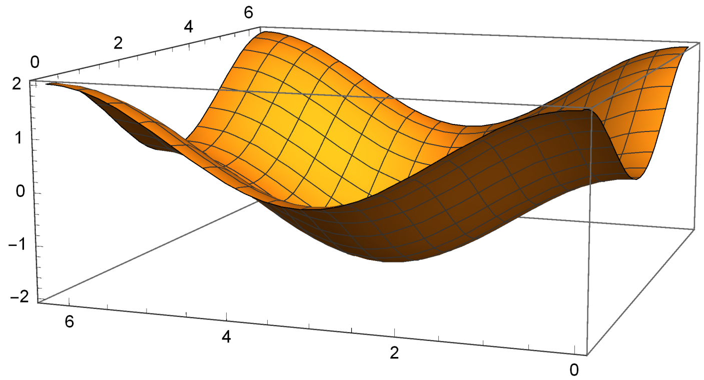
```
lin@*hex@*squareemb@{0, 0} // N
```

2.36603

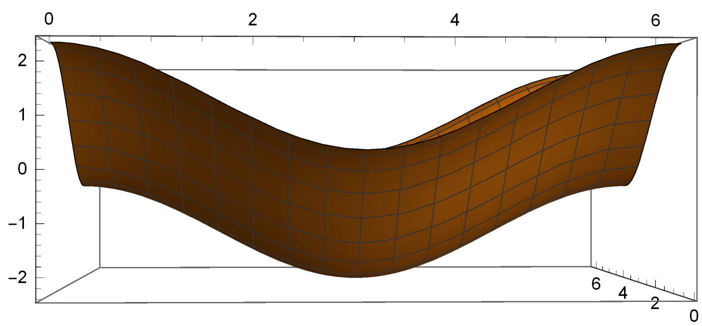
```
lin@*hex@*squareemb@{π, π}
```

$-\frac{3}{2}-\frac{\sqrt{3}}{2}$

```
Plot3D[lin@*squaremb@{t1, t2}, {t1, 0, 2 π}, {t2, 0, 2 π} ]
```

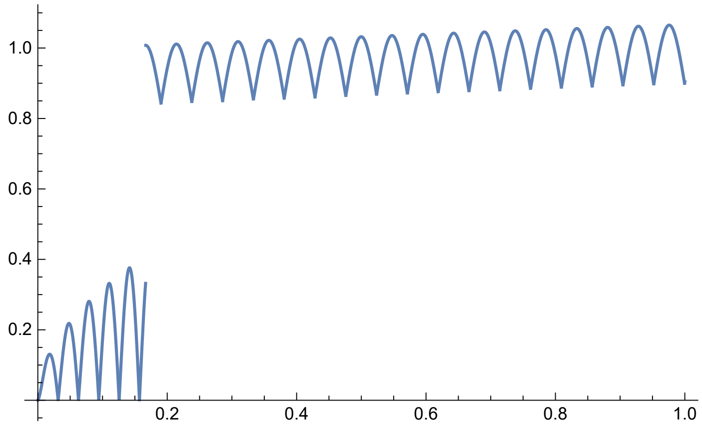



```
Plot3D[lin@*hex@*squaremb@{t1, t2}, {t1, 0, 2 π}, {t2, 0, 2 π} ]
```

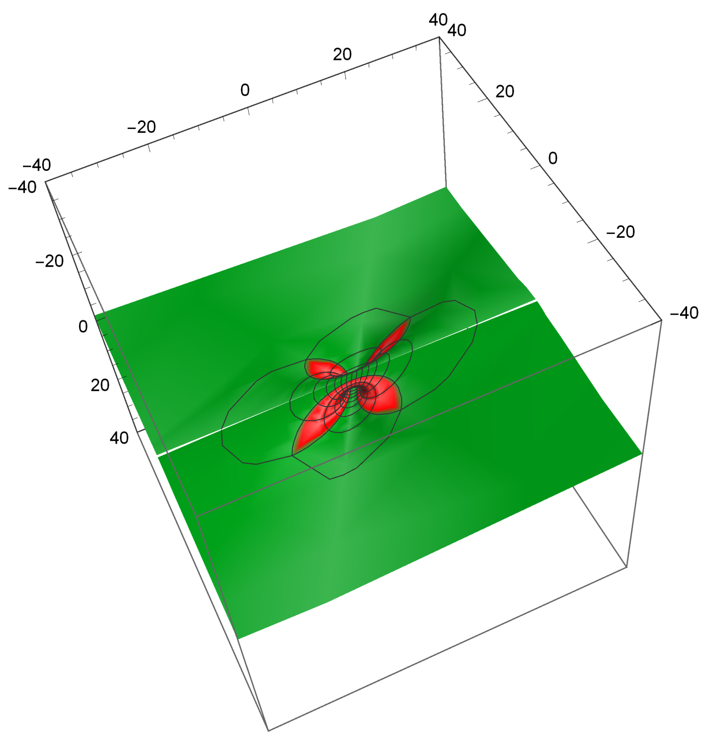



```
rap[z_] := Piecewise[{
    {z ^ (1 / 2) Abs[Sin[100 z]], z ≤ 1 / 2 - 1 / 3},
    {(1 + ((1 - 1 / 2 - 1 / 3) / (1 - 1 / 2 + 1 / 3)) (z - 1)) ^ (1 / 3) +
      Abs[Sin[66 z] / (6)] - 0.1, 1 / 2 - 1 / 3 < z}
  }];
```

```
Plot[rap[z], {z, 0, 1}, PlotRange → Full]
```




```
ParametricPlot3D[hexfunk[{t1, t2}],
 {t1, 0, 2 π}, {t2, 0.001, 1.999 π}, PlotRange → {-40, 40},
 ColorFunction → Function[{x, y, z, u, v}, ColorData["RedGreenSplit"][
    rap[(Max[Abs[π - u], Abs[π - v]] / (π)) ^ (12)] ]], ColorFunctionScaling → False]
```



```
ParametricPlot3D[squarefunk[{t1, t2}], {t1, 0, 2 π}, {t2, 0.001, 1.999 π},
 PlotRange → {-30, 30}, ColorFunction → Function[{x, y, z, u, v},
   RGBColor[Abs[Sin[4 (π - u)]] ^ (1) , ((Abs[π - u] + Abs[π - v]) / (4 π)) ^ (1 / 3) ,
    Abs[Cos[4 (π - v)]]]], ColorFunctionScaling → False]
```